Hyperion® Performance Suite™

Release 8.3

Installation Guide

Hyperion®

Hyperion Solutions Corporation

P/N: D832283000

Copyright © 1998–2004 Hyperion Solutions Corporation.
All rights reserved.

"Hyperion," the Hyperion "H" logo, and Hyperion's product names are trademarks of Hyperion. References to other companies and their products use trademarks owned by the respective companies and are for reference purpose only.

No portion hereof may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying, recording, or information storage and retrieval systems, for any purpose other than the recipient's personal use, without the express written permission of Hyperion.

The information contained herein is subject to change without notice. Hyperion shall not be liable for errors contained herein or consequential damages in connection with the furnishing, performance, or use hereof.

Any Hyperion software described herein is licensed exclusively subject to the conditions set forth in the Hyperion license agreement.

Use, duplication or disclosure by the U.S. Government is subject to restrictions set forth in the applicable Hyperion license agreement and as provided in DFARS 227.7202-1(a) and 227.7202-3(a) (1995), DFARS 252.227-7013(c)(1)(ii) (Oct 1988), FAR 12.212(a) (1995), FAR 52.227-19, or FAR 52.227-14, as applicable.

Hyperion Solutions Corporation
1344 Crossman Avenue
Sunnyvale, California 94089

Printed in the U.S.A.

Contents

Preface . . . . . **ix**
Purpose x
Audience x
Document Structure x
Where to Find Documentation xii
Conventions xiv
Additional Support xv
Documentation Feedback xv

CHAPTER 1 About Hyperion Performance Suite **17**
Overview 18
Licensed Packages 21
Suite Components 21
Integration with Hyperion Analyzer and Hyperion Reports 26

CHAPTER 2 Planning an Installation **27**
Hardware Requirements 28
Operating System Requirements 29
Repository and Data Source Databases 29
Web Environment Requirements 30
Web Browser Requirements 30
External Authentication Systems 31
Distributed and Replicated Components 32
Firewall and DMZ Support 33
APIs 34

Required Pre-Installation Configuration Procedures 35
Creating a Hyperion Performance Suite User Account 37
Configuring Database Software for the Hyperion Performance Suite Repository 37
Creating ODBC Connections 46
Creating a Database Schema for Row-Level Security Tables 46
Creating a Database Schema for Usage Tracking 47
Configuring a Hyperion SQR Server Installation 47
Installing and Configuring a Web Environment 48
Satisfying Temporary Space Requirements 50
Command Prompt Installation Considerations 51
UNIX Installation Mechanisms 51
Fonts on UNIX 52
About Trial Licenses 53

CHAPTER 3 Upgrading Hyperion Performance Suite 55
Upgrade Overview 56
General Upgrade Recommendations 57
Upgrade Recommendations for Distributed Systems 58
Upgrade Prerequisites 59
Upgrade Instructions 60

CHAPTER 4 Installing the Server Components: Express Mode 61
Express Installation Overview 62
Information Required for Express Server Installation 63
Installing Hyperion Performance Suite Server Components: New Express Installation 64

CHAPTER 5 Installing the Server Components: Custom Mode 75
Custom Installation Overview 76
Information Required for Custom Server Installation 77
Installing Hyperion Performance Suite Server Components: Custom Installation 78

CHAPTER 6 Adding, Reinstalling, or Replicating Server Components or Services 95
Adding Components or Services 96
Reinstalling Hyperion Performance Suite 97
Replicating Services 99

CHAPTER 7 Postinstallation Tasks 101
Testing the Hyperion Performance Suite Installation 102
Disabling Directory Browsing 104
Apache Tomcat 4.1.30 104
BEA WebLogic 7.x/8.x 104
IBM WebSphere 5.0.1 and 5.0.2 105
JRun 3.1 105
JRun 4 106
Sun Java System Web Server (aka iPlanet 6.0.6) 106
Security and Passwords 107
Increasing Performance with -Dbrio.content.path 108
Using Console Mode with Hyperion Performance Suite Components . . . 109
Installation Directory Structure 110
Registries 119
Windows Registry 119
VPD Registry 119
Cross-Platform Registry 119
Database Log 123
Core Installable Unit Software Components 123
Driver Components 124
JRE Components 124
Orbix Components 124
Services Components 124
SQR Components 124
SQR SQL 125
Start Scripts 125
Support Components 125
Servlet Configuration Details 126

CHAPTER 8 Removing a Server Installation 127
Removing Hyperion Performance Suite Server from Windows Systems . . 128
Removing Hyperion Performance Suite Server Using the Command Line 130
Installation Removal Details 130

CHAPTER 9 Client Tool Installation 131
Installing the Hyperion Performance Suite Client Tools 132
Removing Hyperion Performance Suite Client Tools 134
Hyperion Designer and Hyperion Explorer Installation Details 135

Directory Structure 135
Program Start Folders 135
Hyperion Intelligence Client Installation Details 136
Files Installed with the Hyperion Intelligence Client 136
Registry Entries 137

APPENDIX A Manually Setting Up the Hyperion Performance Suite Servlets 143
Running the Hyperion Performance Suite Servlets as a Standalone HTTP Server 144
Running the Hyperion Performance Suite Servlets on a Commercial Web Server 145
Supported Web Environments 145
About Manual Configuration 146
Servlet Engine Configuration Settings for Commercial Web Servers . . 146
Verifying Servlet Configuration 146
When Using JRE 1.3.x 147
Configuring Apache 1.3.x with JRun 3.1 148
Preparing the WAR file 148
Deploying the Application 149
Setting the Classpath 150
Adjusting the ws.conf and Logging Parameters 150
Copying Required Files 151
Creating a JRUN Connection Module 151
Testing the Servlets 153
Removing a JRun Connection Module 153
Configuring Apache HTTP Server 2.x with Apache Tomcat 4.1.30 and Later 154
Deploying the Servlets on Apache Tomcat 154
Adding a Jakarta Connector 154
Creating a workers2.properties File 155
Testing the Servlets 155
Configuring Apache Tomcat 4.1.30 and Later Standalone 156
Unpacking the Web Application Contents 156
Deploying Apache Tomcat 4.1.30 and Later 157
Adjusting the ws.conf and Logging Parameters 159
Precompiling the JSPs 159
Copying Required Files 161
Testing the Servlets 161

Configuring BEA WebLogic 6.x with WebLogic SP3 or Later 162
Unpacking the Web Application Contents 162
Modifying the startWebLogic Script 163
Updating the ws.config Parameters 164
Updating the Logging Parameters 164
Copying Required Files 164
Deploying the Servlets 165
Adding the wsmedia Virtual Directory 166
Precompiling WebLogic JSPs 167
Testing the Servlets 167
Configuring BEA WebLogic 7.x 168
Unpacking the Servlets 168
Modifying the startWebLogic Script 169
Modifying the startWLS Script 169
Modifying the CLASSPATH Variable 170
Updating the ws.conf Parameters 172
Updating the Logging Parameters 172
Copying Required Files 173
Deploying the Servlets 174
Adding the wsmedia Virtual Directory 174
Precompiling the JSPs 175
Testing the Servlets 175
Configuring BEA WebLogic 8.x 176
Unpacking the Servlets 176
Modifying the startWebLogic Script 177
Updating the ws.conf Parameters 178
Updating the Logging Parameters 178
Customizing the Servlets Context Path 179
Copying Required Files 179
Precompiling the JSPs 180
Deploying the Servlets 180
Adding the wsmedia Virtual Directory 181
Testing the Servlets 181
Configuring IBM HTTP Server with WebSphere 4.0.5 and Later 182
Prerequisites 183
Manually Deploying the Servlets on WebSphere 4.0.5 and Later 183
Configuring IBM WebSphere 5.0.1 or 5.0.2 187
Configuring WebSphere 5.0.1 or 5.0.2 188
Copying Required Files 189

Configuring the IBM HTTP Server 190
Configuring a Remote IBM HTTP Server 191
Testing the Servlets 192
Configuring the JRun 3.1 Built-in Web Server 193
Manually Installing JRun 3.1 193
Deploying Servlets onto JRun 3.1 196
Mapping the wsmedia Virtual Path 200
Configuring the Classpath 200
Starting the JRun Server Using the JRun Management Console 201
Configuring Microsoft IIS 6 with JRun 4.0 202
Unpacking the Web Application Contents 203
Specify the Java VM Classpath 203
Deploying the Servlets under JRun 204
Configuring Microsoft IIS 6 with JRun 4.0 205
Adding the wsmedia Virtual Directory 205
Testing the Servlets 205
Configuring Microsoft IIS 6 with Apache Tomcat 4.1.30 and Later 206
Unpacking the Web Application Contents 206
Adjusting the ws.conf and Logging Parameters 207
Copying Required Files 207
Precompiling the JSPs 208
Deploying Apache Tomcat 4.1.30 and Later 208
Configuring an Apache Tomcat to IIS 6 Connector 209
Adding the wsmedia Virtual Directory 211
Testing the Servlets 211
Configuring Sun Java System Web Server with Native Servlet Engine 212
Manual Configuration Steps for Sun Java System Web Server 212
Precompiling Sun Java System Web Server 217

APPENDIX B Troubleshooting 219
Using Log Files to Troubleshoot Problems During Installation 220
Troubleshooting Problems During Installation 222
Installation Error Messages 229
Using Log Files to Troubleshoot Web Server Configuration 231
Troubleshooting Problems with Web Servers/Servlet Engines 231
Troubleshooting Service Startup 241

Index 243

Preface

Welcome to the Hyperion Performance Suite Installation Guide. This preface discusses the following topics:

- "Purpose" on page x
- "Audience" on page x
- "Document Structure" on page x
- "Where to Find Documentation" on page xii
- "Conventions" on page xiv
- "Additional Support" on page xv
- "Documentation Feedback" on page xv

Purpose

This guide provides the following information that you need to install Hyperion Performance Suite software on both Microsoft Windows and UNIX systems:

- Operating system and third-party software requirements
- Installation instructions and prerequisite information
- Installation details including directory structure and contents, registry entries, and sample INI, configuration, log, and startup files
- Configuration information for administration tasks necessary to complete installation
- Troubleshooting information, including error messages

Audience

The *Hyperion Performance Suite Installation Guide* is intended for administrators who install and configure Hyperion Performance Suite. It assumes familiarity with distributed computing environments and terminology, server management, Web servers and servlet engines, and either Microsoft Windows or UNIX operating systems.

Document Structure

This guide provides everything that you need to install Hyperion Performance Suite. It contains:

Chapter 1, "About Hyperion Performance Suite," provides a brief architectural overview of Hyperion Performance Suite.

Chapter 2, "Planning an Installation," discusses the system requirements for server operating systems, JRE version, commercial Web servers, servlet engines, Web browsers, databases, and client operating systems. It includes an overview of the installation process, a pre-installation checklist, and required third-party software configuration procedures.

Chapter 3, "Upgrading Hyperion Performance Suite," provides step-by-step instructions for upgrading to Hyperion Performance Suite release 8.3.

Chapter 4, "Installing the Server Components: Express Mode," explains how to install all components authorized by your license keys on a single server. Step-by-step instructions enable the administrator to quickly and easily install all licensed software on a single Microsoft Windows or UNIX server.

Chapter 5, "Installing the Server Components: Custom Mode," explains how to install a distributed Hyperion Performance Suite system. It describes how to distribute the Hyperion Performance Suite components across multiple servers.

Chapter 6, "Adding, Reinstalling, or Replicating Server Components or Services," explains the steps involved in reinstalling, adding, replicating, and upgrading Hyperion Performance Suite components or services.

Chapter 7, "Postinstallation Tasks," provides details about registry entries and keys, INI files, configuration files, and log files.

Chapter 8, "Removing a Server Installation," provides instructions for removing the Hyperion Performance Suite server from your system.

Chapter 9, "Client Tool Installation," describes how to install the client-side software and lists the supporting folder structure.

Appendix A, "Manually Setting Up the Hyperion Performance Suite Servlets," lists the steps for manual configuration of Web servers and servlet engines supported by Hyperion Performance Suite.

Appendix B, "Troubleshooting," provides tips to help users troubleshoot installation problems along with a list of error messages and possible causes and solutions.

The **Index** contains a list of Hyperion Performance Suite terms and their page references.

Where to Find Documentation

All Hyperion Performance Suite documentation is accessible from the following locations:

- The HTML Information Map is located on the Hyperion Performance Suite Documentation CD.
- Online help is accessible from the product that it documents. Start the product and click the Help button or menu command.
- The Hyperion Solutions Web site is located at `http://www.hyperion.com`.
- The Hyperion Download Center can be accessed from `http://hyperion.subscribenet.com` or from `http://www.hyperion.com`.

To access documentation through the Hyperion Solutions Web site:

1. **Log on to** `http://www.hyperion.com`.
2. **Select the Support link and type your username and password to log on.**

Note New users must register to receive a username and password.

3. **Click the Hyperion Download Center link and follow the on-screen instructions.**

To access documentation from the Hyperion Download Center:

1. **Log on to** `http://hyperion.subscribenet.com`.
2. **In the Login ID and Password text boxes, enter your assigned login ID name and password.**
3. **In the Language list box, select the appropriate language and click Login.**
4. **If you are a member on multiple Hyperion Download Center accounts, select the account that you want to use for the current session.**
5. **Perform one of the following actions:**
   - To access documentation online, from the **Product List**, select the appropriate product and follow the on-screen instructions.
   - To order printed documentation, from the **Information** section in the left frame, select **Order Printed Documentation**, then follow the on-screen instructions.

To order printed documentation if you do not have access to the Hyperion Download Center:

- In the United States, call Hyperion Solutions Customer Support at 877-901-4975.
- From outside the United States, including Canada, call Hyperion Solutions Customer Support at 203-703-3600. Clients who are not serviced by support from North America should call their local support centers.

Conventions

The following table shows the conventions used in this document:

Item	Meaning
➤	Arrows indicate one-step procedures.
Bold	Bold highlights options, buttons, or tabs that you need to select and text that you need to type.
CAPITAL LETTERS	Capital letters denote commands and various IDs. For example: CLEARBLOCK command
[Ctrl + O]	Keystroke combinations shown with the plus sign (+) indicate that you should press the first key and hold it while you press the next key. Do not type the plus sign.
`Example text`	Courier font indicates that the example text is code or syntax. Courier font is also used for file names, directory/folder names, and path names. For example: `Sample1.bqy` is in the `\HYPERION_HOME\Samples` directory.
`Courier italics`	Courier italic text indicates a variable field in command syntax. Substitute a value in place of the variable shown in Courier italics.
Mouse orientation	This document provides examples and procedures using a right-handed mouse. If you use a left-handed mouse, adjust the procedures accordingly.
Menu options	Options in menus are shown in the following format. Substitute the appropriate option names in the placeholders, as indicated. **Menu name→Menu command→Extended menu command** For example: 1. Select **File→Desktop→Accounts**.

Additional Support

In addition to providing documentation and online help, Hyperion offers the following product information and support. For details on education, consulting, or support options, click the Services link on the Hyperion Web site at `http://www.hyperion.com`.

Education Services

Hyperion offers instructor-led training, custom training, and eTraining covering all Hyperion applications and technologies. Training is geared to administrators, end users, and information systems (IS) professionals.

Consulting Services

Experienced Hyperion consultants and partners implement software solutions tailored to clients' particular reporting, analysis, modeling, and planning requirements. Hyperion also offers specialized consulting packages, technical assessments, and integration solutions.

Technical Support

Hyperion provides enhanced electronic-based and telephone support to clients to resolve product issues quickly and accurately. This support is available for all Hyperion products at no additional cost to clients with current maintenance agreements.

Documentation Feedback

Hyperion strives to provide complete and accurate documentation. We value your opinions on this documentation and want to hear from you. Give us your comments by going to `http://www.hyperion.com/services/support_programs/doc_survey/index.cfm`.

1 About Hyperion Performance Suite

Hyperion Performance Suite stores, protects, and delivers information for your extended enterprise (employees, customers, suppliers, and partners). It enables users to analyze, transform, and report data; share, search for, retrieve, and use documents; and run queries, execute and schedule jobs, and distribute job output. This functionality can be accessed from the Hyperion Performance Suite application programs or from a Web browser.

For detailed information about the Hyperion Performance Suite system and how it works, see the *Hyperion Foundation Administrator's Guide* and the *Hyperion Foundation User's Guide.*

In This Chapter

Overview 18

Licensed Packages 21

Suite Components 21

Integration with Hyperion Analyzer and Hyperion Reports 26

Overview

Hyperion Performance Suite consists of a server, Java servlets that provide end-user and administrative applications across the Web, and Desktop tools for administrators and advanced end-users. Figure 1 illustrates the server components and Web applications of the system, and Figure 2 illustrates the desktop applications and tools.

The Hyperion Performance Suite server infrastructure consists of collaborative services running on one or more heterogeneous servers in a network. The services can be grouped into *core services* and *functional services.* The core services are required in every system, providing essential services such as authorization, authentication, session management, and publication of documents. The functional services perform more specialized work and may not be part of every Hyperion Performance Suite system.

The security in the system is maintained at three levels: authentication, service-based access, and object-based access.

More than one service of the same type can exist in the system. You can replicate these services:

- Service Broker
- Session Manager
- Authentication Service
- Authorization Service
- Job Service
- Usage Service
- Intelligence Service
- Data Access Service

In addition, the system replicates the Local Service Manager (LSM) automatically for each new server process.

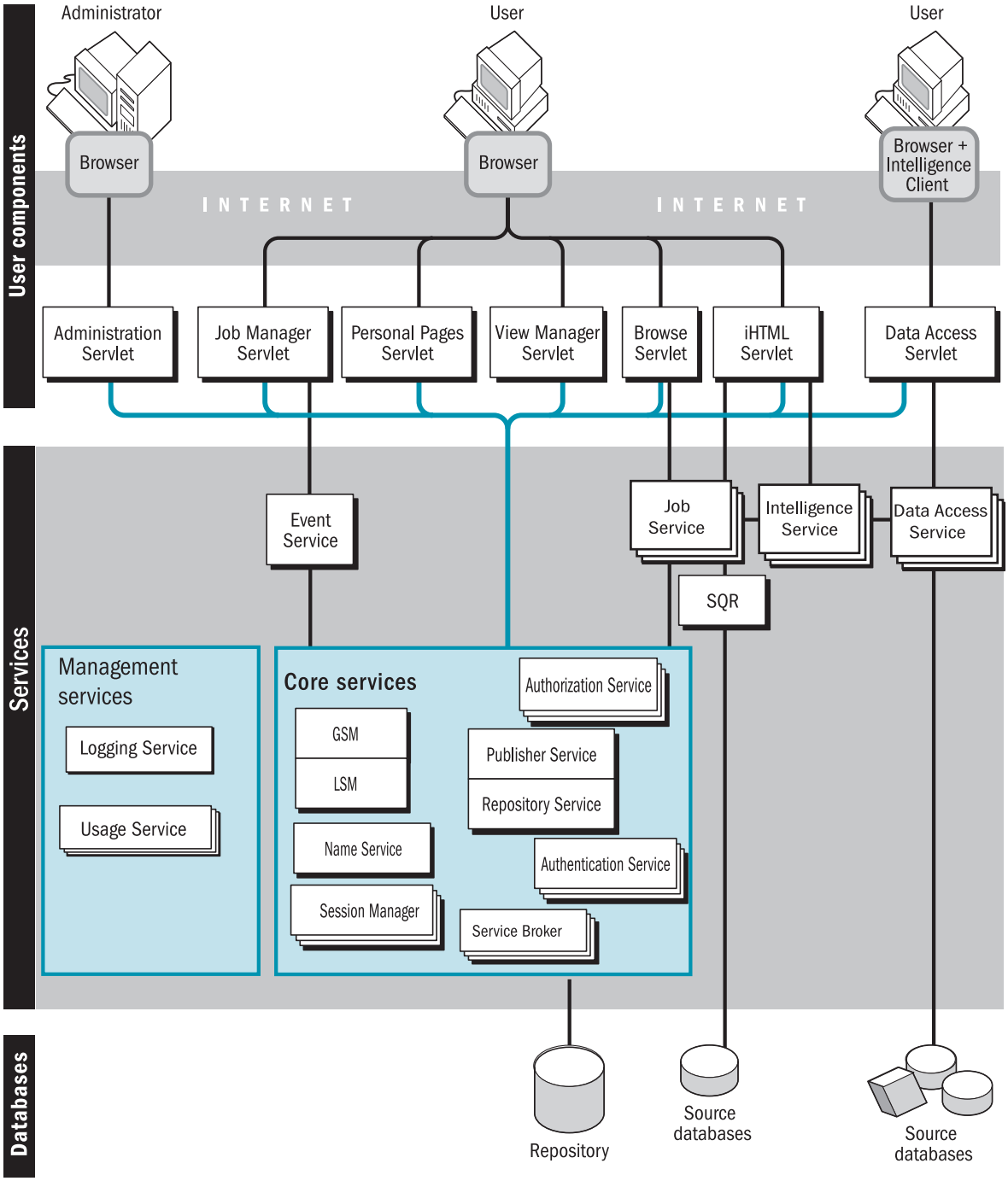

Figure 1 Hyperion Performance Suite

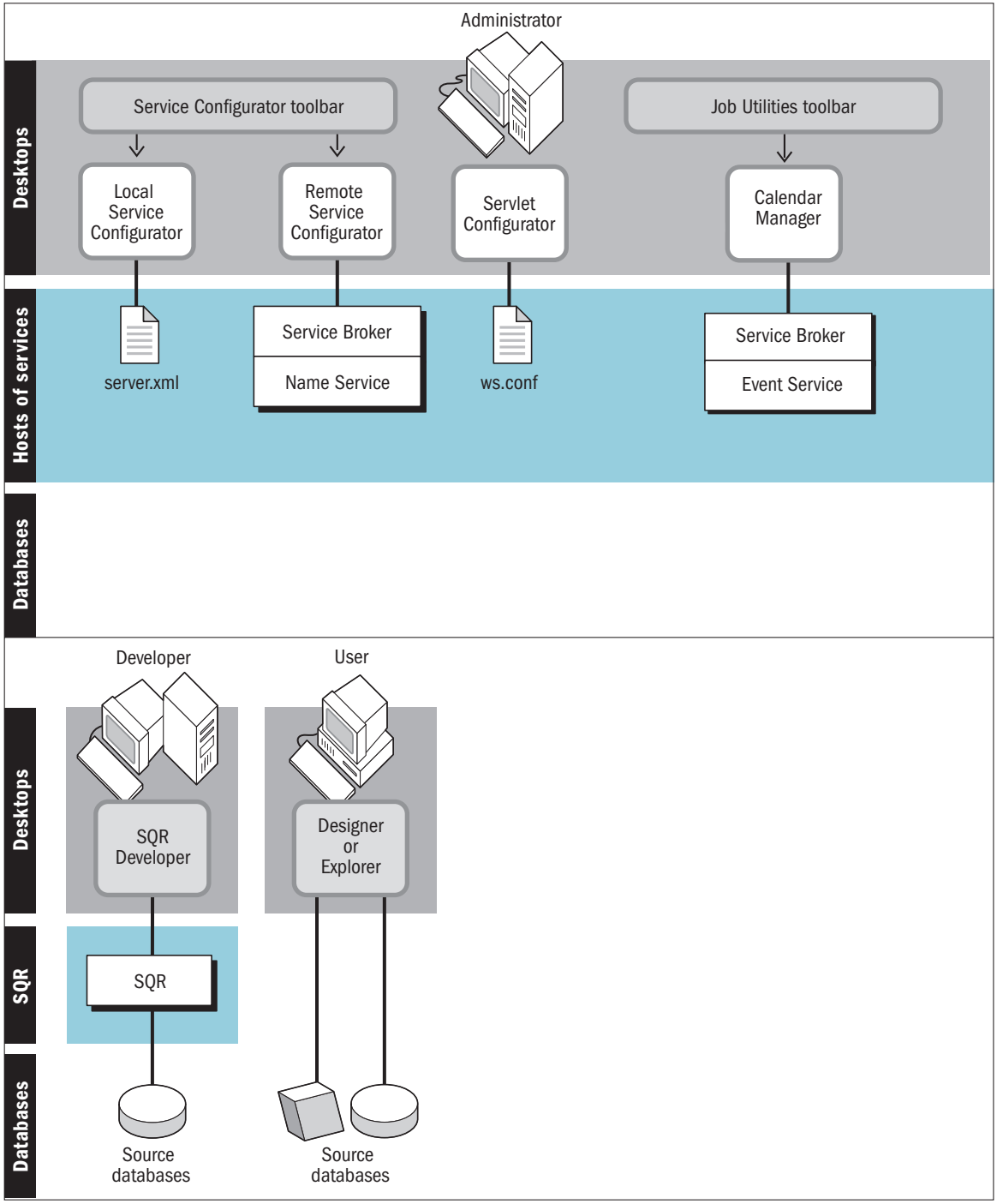

Figure 2 Desktop Applications and Tools

Licensed Packages

Hyperion Performance Suite consists of these product packages:

- Servers
  - Hyperion Intelligence iServer
  - Hyperion Intelligence Server
  - Hyperion SQR iServer
  - Hyperion SQR Server
- Clients
  - Hyperion Intelligence Designer
  - Hyperion Intelligence Explorer
  - Hyperion SQR Developer

For installation, you need the license keys for all of the packages that you purchased.

Suite Components

The Hyperion Performance Suite system comprises core services, functional services, client components, and a programmatic interface. Table 1 describes all the Hyperion Performance Suite server components.

Table 1 Hyperion Performance Suite Server Components

Service/Component	Description	Replicable	Installed with Product
Authentication Service	Checks the user's credentials at login time, and enables the user to connect if the credentials are valid. In addition, it determines the user's group membership, which affects what content and other system objects, or *resources*, the user can view and modify. A Hyperion Performance Suite system can have more than one Authentication Service. An Authentication Service does not have to be on the same server as other services.	Yes	Hyperion Intelligence iServer, Hyperion Intelligence Server, Hyperion SQR iServer

Table 1 Hyperion Performance Suite Server Components *(Continued)*

Service/Component	Description	Replicable	Installed with Product
Authorization Service	Provides security at the level of resources and actions. Authorization is enforced through access privileges, which are associated with actions and resources and then bestowed on users. The Authorization Service manages roles and their associations with operations, users, groups, and other roles. The system must have at least one Authorization Service.	Yes	Hyperion Intelligence iServer, Hyperion Intelligence Server, Hyperion SQR iServer
Client Core	Provides access to the administration tools, such as SDK and Job Utilities. The Client Core includes basic components for applications that communicate with services. Client Core can be replicated, but there is no reason to install Client Core by itself.	Yes	All nonservice applications (SDK, Job Utilities, servlets)
Core	Includes the Local and Remote Service Configurators, Management Services, and Job Utilities. Installed for every *INSTALL_HOME* except those that include only Intelligence Client, SDK, Job Utilities, or servlets. Core includes basic components required on all systems that host a service.	Yes	All services
Data Access Service (DAS)	Provides access to relational and multidimensional databases. It carries out database queries for the Intelligence Client plug-in, using the connection information in one or more OCE files. Each DAS typically supports connectivity to multiple data sources, so that a single DAS can process a document whose sections require various data sources. The DAS maintains a connection pool for database connections.	Yes	Hyperion Intelligence Server, Hyperion Intelligence iServer

Table 1 Hyperion Performance Suite Server Components *(Continued)*

Service/Component	Description	Replicable	Installed with Product
Event Service	Manages users' subscriptions and sends notifications of events involving subscriptions, jobs, and exceptions. It also manages the parameter lists for jobs and dispatches scheduled jobs to be executed by Job Services. In support of subscriptions and scheduled jobs, the Event Service distributes new or modified content through e-mail and FTP. The system must have one Event Service; it cannot have more than one.	No	Hyperion Intelligence iServer, Hyperion Intelligence Server, Hyperion SQR iServer
Global Service Manager (GSM) and Local Service Manager (LSM)	Act as the service dispatchers. They track all the services of the system and, when a client makes a request, they provide it with a suitable service. They perform load-balancing when dispatching services to clients. The GSM also holds the system configuration information and periodically checks all registered services to see if they are running. The system must have one GSM; it cannot have more than one. Every server process in the system must have one LSM (including the server process running the GSM); a server processor cannot have more than one LSM.	No	Hyperion Intelligence iServer, Hyperion Intelligence Server, Hyperion SQR iServer
Intelligence Service	The engine for the iHTML servlet. When the iHTML servlet requests an action involving a BQY document, the Intelligence Service fulfills that request. Specifically, this service gets the BQY document, processes it, and delivers HTML to the servlet for display in the user's browser.	Yes	Hyperion Intelligence iServer
Job Service	Executes reports, such as BQY, SQR, and generic jobs. It listens for a job request, initiated either by a user or the Event Service, and manages execution of the job. The Job Service publishes the job output to the repository, and returns the resulting output to the requester. The system can have more than one Job Service.	Yes	Hyperion Intelligence iServer, Hyperion Intelligence Server, Hyperion SQR iServer

Table 1 Hyperion Performance Suite Server Components *(Continued)*

Service/Component	Description	Replicable	Installed with Product
Job Utilities	Includes Calendar Manager for managing calendars and logs.	Yes	Hyperion Intelligence iServer, Hyperion Intelligence Server, Hyperion SQR iServer
Logging Service	Writes log messages from Hyperion Performance Suite components to one or more log files. Consolidates logs from all sources to a single location.	No	Hyperion Intelligence iServer, Hyperion Intelligence Server, Hyperion SQR iServer
Name Service	Keeps track of where the RSC services are installed and provides them with their configuration data. (They may have additional configuration settings obtained from the `server.xml` file.) The Name Service and Service Broker assist the GSM and LSMs by routing client requests for RSC services. A system must have one Name Service; it cannot have more than one.	No	Hyperion Intelligence iServer, Hyperion Intelligence Server, Hyperion SQR iServer
Publisher Service	Handles communications with the Repository Service for the LSC services. When any of these services needs to publish to or retrieve content from the repository, the service sends its request to the Publisher Service. The Publisher Service forwards the request to the Repository Service and passes the reply back to the initiating service. The system must have one Publisher Service; it cannot have more than one.	No	Hyperion Intelligence iServer, Hyperion Intelligence Server, Hyperion SQR iServer
Repository Service	Adds, stores, modifies, removes, retrieves, searches, and lists the content of the system, namely folders, files, jobs, channels, URLs, and shortcuts and their metadata. The content resides in the repository, which consists of files in the native file system and metadata for each object in a database in a Relational Database Management System (RDBMS). The system must have one Repository Service; it cannot have more than one.	No	Hyperion Intelligence iServer, Hyperion Intelligence Server, Hyperion SQR iServer

Table 1 Hyperion Performance Suite Server Components *(Continued)*

Service/Component	Description	Replicable	Installed with Product
SDK	Supports extension of the system. It includes APIs to the Hyperion Performance Suite services and Jython scripting language.	Yes	Hyperion Intelligence iServer, Hyperion Intelligence Server, Hyperion SQR iServer
Service Broker	Supports the Global Service Manager and the Local Service Managers by dispatching client requests to any of the RSC services and load-balancing those services. The Service Broker collaborates with the Name Service. A system can have more than one Service Broker.	Yes	Hyperion Intelligence iServer, Hyperion Intelligence Server, Hyperion SQR iServer
Servlets	Manage the control functions of Hyperion Performance Suite, such as dispatching requests, and handling nontextual data. The servlets are as follows: ■ **Browse**—End-user Web module for browsing the repository, finding and viewing items, and publishing content to the repository ■ **Personal Pages**—End-user Web module for user-customizable views of Hyperion Performance Suite content and links to favorite information Both the content and presentation are customizable. ■ **Job Manager**—End-user Web module for scheduling jobs ■ **View Manager**—End-user Web module for viewing subscribed and pushed items ■ **Administration**—Administrative Web module for managing users, groups, and roles, MIME types, and physical resources ■ **Data Access**—Servlet that works with the Intelligence Client browser plug-in, enabling users to use queries and reports over the Web ■ **iHTML**—Servlet that enables users to access BQY documents using only a Web browser	Yes	Hyperion Intelligence iServer, Hyperion Intelligence Server, Hyperion SQR iServer

Table 1 Hyperion Performance Suite Server Components *(Continued)*

Service/Component	Description	Replicable	Installed with Product
Session Manager	Tracks all the services of the system; holds the system configuration information and periodically checks that all registered services are running. More than one Session Manager can exist in a Hyperion Performance Suite deployment. Each one manages a set of sessions.	Yes	Hyperion Intelligence iServer, Hyperion Intelligence Server, Hyperion SQR iServer
Usage Service	Gathers historical information about system activities. It covers usage of the Hyperion Foundation as well as reporting and analytic activities. It keeps records on logins, opening documents, closing documents for select MIME types, running jobs, viewing job output, and queries processed by the Intelligence iServer.	Yes	Hyperion Intelligence iServer, Hyperion Intelligence Server, Hyperion SQR iServer
Web Services	Operational commands to administer Apache Tomcat that utilize SOAP message formats. **Note:** The Web Services feature is supported only with Apache Tomcat installations.	No	Hyperion Intelligence iServer, Hyperion Intelligence Server

For installation information on the Hyperion Intelligence Designer and Explorer clients, see Chapter 9, “Client Tool Installation.”

Integration with Hyperion Analyzer and Hyperion Reports

Hyperion Performance Suite has been integrated with Hyperion Analyzer and Hyperion Reports. You can access Hyperion Analyzer and Hyperion Reports content through the Browse module of Hyperion Performance Suite. You can also access Hyperion Performance Suite content through Hyperion Analyzer and Hyperion Reports.

Hyperion Hub integrates the applications. External authentication using the security platform driver is required to use Hyperion Hub.

For more information on integrating Hyperion applications, see the *Hyperion Foundation Administrator’s Guide*.

2 Planning an Installation

Before you can install and use Hyperion Performance Suite, your system must meet certain hardware and software requirements. You must also configure your system to facilitate the installation process. Make sure you carefully review this information before you install Hyperion Performance Suite.

Subsequent maintenance releases and service packs for third-party platform software may be used where the vendor asserts backward compatibility. Please be aware, however, that although these assertions are made in good faith, certain incompatibilities may exist. In the event that an incompatibility is identified, Hyperion may experience a delay in reproducing and fixing resultant issues for the affected versions.

In This Chapter

Hardware Requirements 28
Operating System Requirements 29
Repository and Data Source Databases 29
Web Environment Requirements 30
Web Browser Requirements 30
External Authentication Systems 31
Distributed and Replicated Components 32
Firewall and DMZ Support 33
APIs 34
Required Pre-Installation Configuration Procedures 35
Command Prompt Installation Considerations 51
UNIX Installation Mechanisms 51
Fonts on UNIX 52
About Trial Licenses 53

Hardware Requirements

By design, Hyperion Performance Suite is a distributed network system. For best performance, distribute Hyperion Performance Suite components across multiple servers. For evaluation purposes, however, you might find it useful to install Hyperion Performance Suite on a single computer. The minimum hardware requirements for installation of the Hyperion Performance Suite components are as follows:

- **CD-ROM drive**—Required for installation of Hyperion Performance Suite server or client components.
- **1.2 GHz CPU**—2 GHz is required if the repository database is on the same computer as all the Hyperion Performance Suite components.
- **RAM**—Each component (that is, Hyperion Foundation, Hyperion Intelligence iServer, Hyperion SQR iServer, and Hyperion Intelligence Server) requires a minimum of 256 MB of RAM. All services installed on the same computer requires 512 MB of RAM. All servlets installed on the same computer requires 512 MB of RAM. Everything installed on the same computer requires 1 GB of RAM.
- **Free disk space**—Foundation and Hyperion SQR iServer require 350 MB free disk space. Hyperion Intelligence iServer requires 450 MB. Hyperion Intelligence Server requires 400 MB. All servers on the same computer requires 350 MB of free disk space. SQR requires an additional 130 MB for all databases. Additional disk space is needed for files that users publish.

  Hyperion recommends that you have a minimum of 1 GB of free disk space before beginning the installation process.
- **Disk space for the installer**—125 MB required.
- **Cache/Temp space**—1 GB required.
- **TCP/IP connection**

Note These minimum requirements are only for Hyperion Performance Suite. In addition to these requirements, make sure that your hardware meets the systems requirements specified for your operating system.

Operating System Requirements

Before you begin your installation, you must view the Hyperion Performance Suite Support Matrix. The support matrix lists the operating system requirements for the various Hyperion Performance Suite components.

To view the support matrix, go to the Hyperion Download Center.

Repository and Data Source Databases

Repository databases are the relational databases that Hyperion Performance Suite uses to store and manage the application metadata. The network on which you run Hyperion Performance Suite must have one of the supported repository databases. You must view the Hyperion Performance Suite Support Matrix to review the supported repository and data source databases.

To view the support matrix, go to the Hyperion Download Center.

Note The amount of space needed depends on the size of the repository that you plan to build. Hyperion suggests starting with a minimum of 40 MB, which provides room to grow the Hyperion Performance Suite repository without having to increase the datafile or tablespace. Also, performance should be better if you initially create a larger data file; otherwise, pieces of it could be strewn across the disk.

Data source databases are the relational and multidimensional databases that the Hyperion Performance Suite business services and client components use to access data content. A wider variety of databases are supported for this purpose than is defined for the repository databases. The server and client components that access data sources include Hyperion Intelligence Server, Hyperion Intelligence iServer, Hyperion SQR iServer, Hyperion Designer, Hyperion Explorer, and HyperionSQR Developer.

Web Environment Requirements

The Web environment requirements are based on support for the World Wide Web Consortium (W3C) standards for servlets (Servlet 2.2) and JSP (JSP 1.1). The Web environment consists of an HTTP server and a servlet container. These two components may be installed on the same hardware, or they may be implemented on different computers. The Hyperion Performance Suite Support Matrix lists the supported Web environments and indicates whether they must be configured manually or are automatically configured by the installer. You must view the Hyperion Performance Suite Support Matrix to review the supported Web environments.

To view the support matrix, go to the Hyperion Download Center.

Please keep these points in mind when reviewing the support matrix:

- **JRun 3.1 built-in Web server**—Included with all JRun-based installations and is always installed with the Hyperion Performance Suite servlets because it is deployed as a demo/diagnostic tool. You should most likely turn off the service. Customer Support uses the JRun built-in Web server to better troubleshoot any problems you may encounter on your primary Web server.
- **IBM HTTP Server**—If on a different computer from IBM WebSphere, some manual post-configuration steps are required. See "Configuring a Remote IBM HTTP Server" on page 191.
- **Sun Java System Web Server**—Make sure that IWS_SERVER_HOME environment variable is set to the Sun Java System Web Server installation directory that contains `startconsole`.

Web Browser Requirements

The Hyperion Performance Suite Support Matrix lists the supported Web browsers. You must have a Web browser to use the Hyperion Performance Suite HTML-based applications (that is, the Browse, Publish, Administration, and Personal Pages modules and the Job Manager and View Manager), Hyperion SQR iServer, and Hyperion Intelligence iServer. Additionally, the Intelligence Client resides inside a browser as an Active-X or Netscape-style browser extension. You must view the Hyperion Performance Suite Support Matrix to review the supported Web browsers.

To view the support matrix, go to the Hyperion Download Center.

External Authentication Systems

The Hyperion Performance Suite Authentication service is built using a driver paradigm in which a variety of drivers may be built, by Hyperion Solutions or by customers/partners, to perform the actual authentication.

Hyperion Performance Suite provides out-of-box drivers for these external authentication systems:

- Security platform
- Windows 2000 Domains
- Microsoft Active Directory
- Netscape Directory Server
- Generic LDAP Server
- UNIX NIS

If you are integrating several Hyperion applications, single sign-on (through the security platform driver) and registration to Hyperion Hub allows users to access the reports or views of registered applications with one logon. For more information, refer to the *Hyperion Foundation Administrator's Guide*.

In addition to native Hyperion single sign-on, Hyperion applications can integrate with Web access management solutions such as the Netegrity SiteMinder product. The following security agents are tested and supported for single sign-on with Hyperion applications using the Hyperion security platform:

- SiteMinder Web Agent 5.5 Service Pack 2
- SiteMinder Policy Server 5.5 Service Pack 2

If your corporation has implemented SiteMinder to protect company Web resources, you can configure the security platform to require only that users authenticate through SiteMinder, after which they will not be required to present credentials again when logging on to Hyperion applications.

For information on configuring the configuring for SiteMinder single sign-on, see the *Hyperion Foundation Administrator's Guide*.

Distributed and Replicated Components

Hyperion Performance Suite server components are designed to work in a distributed and replicated manner. Distributed components make it possible to install components on separate computers. Replicated components make it possible to install multiple instances of the same component on separate computers where Hyperion Performance Suite automatically routes work across the multiple replicated components.

It is important to note that performance characteristics are not the same for all implementation configurations. Capacity planning and configuration planning information is not covered by this document.

This release of Hyperion Performance Suite supports the following configurations of distributed and replicated components:

- Distributed Configurations
  - All components on same computer
  - All servlets on one computer; all other components on a second computer
  - All servlets on one computer; Repository Service on a separate computer; business services on a separate computer
  - All components fully distributed
- Replicated Configurations
  - No replication; single implementation of all components
  - Servlets (as a unit) replicated
  - Authentication Service replicated
  - Authorization Service replicated
  - Data Access Service replicated
  - Intelligence Service replicated
  - Job Service replicated
  - Service Broker replicated
  - Session Manage replicated

The following limitations apply to distribution and replication:

- Publisher, Event, Repository, and Logging Services cannot be replicated.
- LSM is not directly installed; it comes with other services. You can have one LSM per process space.
- Publisher, Authorization, and Repository Services are required in one process space.
- Intelligence Service requires Job Service.

Firewall and DMZ Support

All elements of Hyperion Performance Suite work within configurations where one or more firewalls are deployed, and that communicate through both HTTP and HTTPS (at both 56- and 128-bit encryption levels). The supported firewall configurations are:

- No firewalls present
- Browser | Firewall | HTTP server with Hyperion Performance Suite servlets | Hyperion Performance Suite server components
- Browser | Firewall | HTTP server | Firewall | Application server with Hyperion Performance Suite servlets | Hyperion Performance Suite server components

Figure 3 depicts these supported firewall configurations.

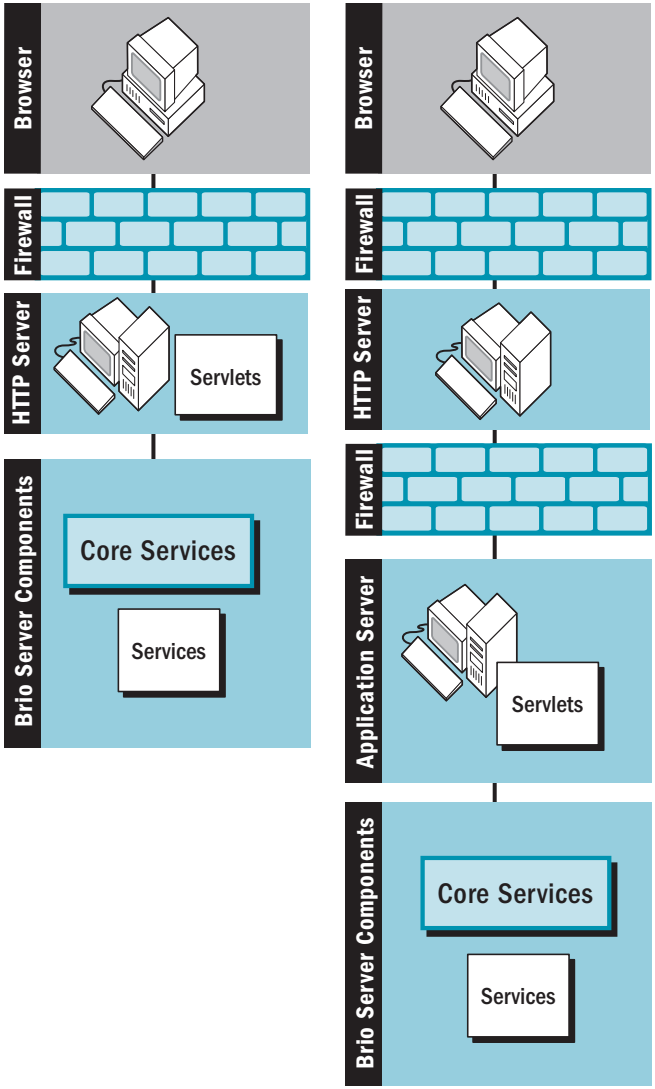

Figure 3 Supported Firewall Configurations

APIs

The Software Developer's Kit (SDK) contains the documentation and libraries necessary to extend the Hyperion Performance Suite Foundation using application programming interfaces (APIs).

Required Pre-Installation Configuration Procedures

The following sections describe the tasks you must complete prior to installing Hyperion Performance Suite:

- Creating a Hyperion Performance Suite User Account
- Configuring Database Software for the Hyperion Performance Suite Repository
- Creating ODBC Connections
- Creating a Database Schema for Row-Level Security Tables
- Creating a Database Schema for Usage Tracking
- Configuring a Hyperion SQR Server Installation (if your users want to use Hyperion SQR iServer to run SQR jobs)
- Installing and Configuring a Web Environment
- Satisfying Temporary Space Requirements

These sections detail how to configure your system prior to installing Hyperion Performance Suite. These procedures differ according to the RDBMS used for Hyperion Performance Suite metadata storage.

Note If you are running Hyperion Performance Suite services on multiple hosts, update the hosts file with an entry for each host in your Hyperion Performance Suite configuration. The hosts file usually resides at `c:\winnt\system32\drivers\etc\hosts` on Windows, or at `/etc/hosts` on UNIX. Listing each host in your Hyperion Performance Suite configuration increases the reliability of the Hyperion Performance Suite service agent startup procedure.

Table 2 provides additional information to consider before installing Hyperion Performance Suite.

Table 2 Pre-Installation Guidelines

Installer Environment	Guidelines
Hyperion Intelligence row-level security	If you want to enable row-level security in your database, set up a schema prior to installing Hyperion Performance Suite. Do not use the Hyperion Performance Suite user account.

Hyperion SQR 6.x and later (all RDBM systems)	On UNIX systems, perform these tasks:
	■ Set the SQR environment variable and path:
	C shell:
	`setenv SQRDIR "directory_name/db_code/bin"`
	`setenv PATH "${PATH}:$SQRDIR"`
	Bourne/Korn shell:
	`SQRDIR="directory_name/db_code/bin";`
	`export SQRDIRPATH="$PATH:$SQRDIR";export PATH`
	■ Check INPUT and ASK parameter specification.

IBM DB2	IBM DB2 is case-sensitive. Ensure that the IBM DB2 Client Application Enabler is installed. Make sure to create the required regular and system temporary tablespaces (see page 39). The default tablespace for the database user who owns the Hyperion Performance Suite repository must not be partitioned; it must be a single tablespace.
Microsoft SQL Server	The password for the Microsoft SQL Server repository database should not be null.
Oracle	Hyperion Performance Suite must not share user tablespaces with other applications or the SYSTEM tablespace.
RDBMS general caveats	Do not create superuser Hyperion Performance Suite accounts in your databases.
Solaris	Before running the installer on Solaris systems, make sure that the `cp -f` flag is enabled to prevent the installer from failing at the end of the installation when precompiling Web components.
UNIX	Before installing Hyperion Performance Suite, to ensure that BQY documents process correctly and have a consistent look and feel, you must make either Type1, TrueType, or OpenType fonts available to the Intelligence Service. See "Fonts on UNIX" on page 52 for more information.

Creating a Hyperion Performance Suite User Account

Select or create a Windows or UNIX user account for Hyperion Performance Suite. All Hyperion Performance Suite services on this computer should be started from this account.

For Hyperion Performance Suite to properly operate, it must be able to run using a user account. For security purposes, Hyperion suggests creating a new, unique user account for Hyperion Performance Suite named hypsuite. You can also use an existing user account.

Note Hyperion Performance Suite system permissions are the same as those of the user account with which it is associated.

Hyperion strongly recommends against installing Hyperion Performance Suite as a superuser because the services would then run with root privileges, thus creating a potential breach in system security.

On UNIX, the Hyperion Performance Suite user account does not need root privileges, but it does need selective write privileges to selected Web server directories to write configuration files.

On Windows, the Hyperion Performance Suite user account must have administrator privileges to update the registry.

Configuring Database Software for the Hyperion Performance Suite Repository

Hyperion Performance Suite requires a database to store its persistent data. If you have not already done so, install a database for the Hyperion Performance Suite repository. After the database is installed, follow the appropriate procedure to configure your database for use with Hyperion Performance Suite *prior* to installing Hyperion Performance Suite:

- Configuring IBM DB2
- Configuring Microsoft SQL Server
- Configuring Oracle

Configuring IBM DB2

After installing IBM DB2, record the IBM DB2 database host name, port number, and service name for use when you install the Hyperion Performance Suite products.

Note A database administrator should perform the following procedures.

To configure your system for use with IBM DB2:

1. **Create an operating system user account.**

   For instructions, see "Creating a Hyperion Performance Suite User Account" on page 37. Hyperion suggests that you name the database user **hypsuite**.

2. **Log on to the DB2 Control Center as the DB2 administrator and add the Hyperion Performance Suite user that you created in Step 1 to the DB2 database.**

   This provides Hyperion Performance Suite access to your database. The Hyperion Performance Suite user owns the Hyperion Performance Suite database tables that are built and populated and objects that are created during installation. Hyperion Performance Suite services use the same user account to connect to the database.

Note IBM DB2 is case-sensitive.

3. **Create a database within a DB2 instance for Hyperion Performance Suite.**

   Create a new database for Hyperion Performance Suite. You can create the database in an existing instance or in a new instance. See your database administrator for instructions on creating a new instance. To create a new database, log on to the instance account, run the DB2 command line processor, and enter the following commands:

   ```
   CREATE DATABASE database_name
   TERMINATE
   ```

Note Alternatively, you can use the Command Center instead of the Command prompt.

The amount of space needed depends on the size of the repository that you plan to build. Hyperion suggests starting with a minimum of 40 MB, which provides room to grow the Hyperion Performance Suite repository without increasing the datafile/tablespace. Also, performance should be better if you initially create a larger data file; otherwise, pieces of it could be strewn across the disk.

4 **Create a regular bufferpool and a regular tablespace with a pagesize of 8 KB.**

Rename *bufferpool_name* and *tablespace_name* as desired, but you must use the names consistently in both commands.

Caution! The following example is for DB2 version 8. If you are using DB7 version 7.2, you must remove IMMEDIATE from the command and manually bounce the DB2 database before installing Hyperion Performance Suite.

```
CREATE BUFFERPOOL bufferpool_name IMMEDIATE SIZE 256 PAGESIZE 8 K EXTENDED STORAGE
CREATE REGULAR TABLSPACE tablespace_name PAGESIZE 8 K MANAGED BY SYSTEM USING
('system_container')EXTENTSIZE 16 OVERHEAD 10.5 PREFETCHSIZE 16 TRANSFERRATE 0.33
BUFFERPOOL bufferpool_name DROPPED TABLE RECOVERY ON

COMMENT ON TABLESPACE tablespace_name IS 'regular tablespace with page size of 8K'
```

For example, you can submit these statements using the following parameters:

```
bufferpool_name = HPS_BP
tablespace_name = HPS-TS
system_container = 'C:\DB2\HPS_TS'
```

Note Do not start any of the *_name* variables with *SYS*.

5 **Create a bufferpool and a system temporary tablespace with a pagesize of 8 KB.**

Rename *tmp_bufferpool_name* and *tmp_tablespace_name* as desired, but you must use the names consistently in both commands.

Caution! The following example is for DB2 version 8. If you are using DB7 version 7.2, you need to remove IMMEDIATE from the command and manually bounce the DB2 database before installing Hyperion Performance Suite.

```
CREATE BUFFERPOOL tmp_bufferpool_name IMMEDIATE SIZE 512 PAGESIZE 8 K
CREATE SYSTEM TEMPORY TABLSPACE tmp_tablespace_name PAGESIZE 8 K MANAGED BY SYSTEM
USING ('system_tmp_container')EXTENTSIZE 16 OVERHEAD 10.5 PREFETCHSIZE 16
TRANSFERRATE 0.33 BUFFERPOOL tmp_bufferpool_name

COMMENT ON TABLESPACE TMP_TABLESPACE_NAME IS 'system temp tablespace with page
size of 8K'
```

For example, you can submit these statements using the following parameters:

```
tmp_bufferpool_name = TMP_HPS_BP
tmp_tablespace_name = TMP_HPS-TS
system_tmp_container = 'C:\DB2\TMP_HPS_TS'
```

Note Do not start any of the *_name* variables with *SYS*.

Not having this tablespace may cause the installation to fail, and not using these commands prevents the V8_JOBINPMULTVALS and VA_USAGE_ATTRIBUTES tables from being created.

Caution! The default tablespace for the database user that owns the Hyperion Performance Suite repository must not be partitioned; it must be a single tablespace.

6 **Grant database privileges for the Hyperion Performance Suite user.**

The user account created in Step 1 owns the Hyperion Performance Suite database tables and objects that are built and populated during this installation. Log on to the DB2 instance account and enter these lines from within the DB2 command line processor:

```
CONNECT TO database_name
GRANT USE OF TABLESPACE tablespace_name TO USER hypsuite_system_username
GRANT CONNECT ON DATABASE TO USER hypsuite_system_username
GRANT CREATETAB ON DATABASE TO USER hypsuite_system_username
TERMINATE
```

For information on how to define and set database parameters, consult your IBM DB2 documentation.

7 **Ensure that the IBM DB2 Client Application Enabler is installed on the computers on which you are about to install Hyperion Performance Suite services.**

8 **If you use an IBM DB2 RDBMS on a different computer than the Hyperion Performance Suite services, create a client connection to the Hyperion Performance Suite database using the Client Application Enabler.**

The Client Configuration Assistant feature of this application requires that you enter the host name, port number, and service name used in your database server installation. Note the full path to the installed driver, `db2java.zip`. The file is usually in `/DB2home/java`.

9 **Set all necessary environment variables:**

- **Windows systems**—Set the LIBPATH environment variable to the location of the DB2 libraries. Use the **System** option in Control Panel or the **Set** command from the Windows command prompt.
- **UNIX systems**—Set DB2COMM, DB2INSTANCE, DB2DIR, and LD_LIBRARY_PATH (or SHLIB_PATH on HP-UX) environment variables. For AIX, you must also set LIBPATH.

10 **Set the drda_heap_sz parameter to 2048 or higher using the Control Center or Command prompt, for example:**

```
update dbm cfg using drda_heap_sz 2048
```

Setting this instance level database parameter higher enables the services to start and run more cleanly and effectively.

11 **Set the stmtheap, applheapsz, and app_ctl_heap_sz parameters to 8096.**

```
update database configuration for database_name using stmtheap 8096
update database configuration for database_name using applheapsz 8096
update database configuration for database_name using app_ctl_heap_sz 8096
```

12 **If you are using DB2 version 7, set the DB2_INDEX_2BYTEVARLEN variable to ON using the db2set tool; for example, run "db2set DB2_INDEX_2BYTEVARLEN=ON".**

This ensures that DB2 creates indexes for columns with length greater than 255 bytes during Hyperion Performance Suite installation.

Note Connectivity to IBM DB2 on AIX 5.2 may fail if the installation is done on a remotely mounted location (a directory location mounted over the network). To resolve this issue, copy recursively the *INSTALL_HOME*`/BrioPlatform/connectivity` to a local directory like `/opt/odbc/connectivity` (this directory should be on a local hard disk of the computer), set the LIBPATH to point to `/opt/odbc/connectivity/lib`, then restart DAS.

Configuring Microsoft SQL Server

After you install Microsoft SQL Server, record the Microsoft SQL database host name, port number, and server for use when you install the Hyperion Performance Suite products. Microsoft SQL Server is not case-sensitive.

Note A database administrator should perform the following procedures.

To configure Microsoft SQL Server for use with Hyperion Performance Suite:

1 **Set up the database within your RDBMS.**

Decide whether there is a database device with sufficient disk space to store the Hyperion Performance Suite metadata. If not, create a database device. To do so, enter the following commands:

```
DISK INIT NAME = 'logical_name',
PHYSNAME = 'absolute_dir_path_and_file_name',
VDEVNO = virtual_device_number,
SIZE = size_in_2k_pages
go
```

Note You can use the Enterprise Manager instead of the Command prompt.

2 **Create a database in which to store the metadata.**

```
CREATE DATABASE hypsuiteDB
ON device_name = size_in_MB
go
```

The amount of space needed depends on the size of the repository that you plan to build. Hyperion suggests starting with a minimum of 40 MB, which provides room to grow the Hyperion Performance Suite repository without to increasing the datafile/tablespace. Also, performance should be better if you initially create a larger data file; otherwise, pieces of it could be strewn across the disk.

3 **Create a database user account for Hyperion Performance Suite.**

A user must be created. This user owns the Hyperion Performance Suite database tables and objects that are built and populated during this installation. The user created here is also the user that Hyperion Performance Suite services use to connect to the database. Hyperion suggests naming the user **hypsuite**.

Note The Hyperion Performance Suite repository database user cannot be administrator, nor can it be a user who is assigned the System Administrators role.

4 **Create the Microsoft SQL Server login:**

```
sp_addlogin login_id, password hypsuiteDB
go
```

Note The password for the Microsoft SQL Server repository database should not be null.

5 **Create the database user:**

```
use hypsuiteDB
go
sp_adduser login_id, login_id
go
```

6 **Grant privileges to the user:**

```
use hypsuiteDB
go
grant create table to login_id
go
grant select on sysobjects to login_id
go
grant create procedure to login_id
go
grant create view to login_id
go
```

where `login_id = UT_userlogin_id` (OR) `login_id = DB_userLogin_id` only if (`DB_userlogin_id` = `UT_userlogin_id`).

Configuring Oracle

After you install Oracle, record the Oracle database host name, port number, and Oracle SID (or Service Name) for use when you install the Hyperion Performance Suite products.

Note that when you install Oracle, SQL*Net is needed to run Hyperion SQR for Oracle.

Note A database administrator should perform these procedures.

To configure Oracle for use with Hyperion Performance Suite:

1 **Dedicate a tablespace to Hyperion Performance Suite.**

Determine the tablespaces to be used as the default tablespace and temporary tablespace for this user. For performance reasons, these should not share the same tablespace as other applications nor should they use the SYSTEM tablespace. To list the existing tablespaces, enter the following SQL statement from SQL*Plus:

```
select * from sys.dba_tablespaces;
```

If an existing tablespace is not suitable as the default or temporary tablespace for this user, then create a new tablespace by entering one of the following SQL statements from the SQL*Plus utility. (This action requires CREATE TABLESPACE system privileges.)

- If the database instance or file name (*filespec*) already exists, enter this statement:

```
create tablespace tablespace_name datafile 'filespec';
```

- If the database instance or file name (filespec) does not already exist, enter this statement:

```
create tablespace tablespace_name datafile 'filespec' size nnM;
```

where *nn* is the number of megabytes.

The amount of space needed depends on the size of the repository that you plan to build. Hyperion suggests starting with a minimum of 40 MB, which provides room to grow the Hyperion Performance Suite repository without increasing the datafile/tablespace. Also, performance should be better if you initially create a larger data file; otherwise; pieces of it could be strewn across the disk.

The sizing of the shared pool is also important, because it determines whether the installer can create all of the necessary repository tables. If the shared pool size on the database is set too small, you receive an error. Hyperion recommends that a shared pool size of 20 MB be used during the installation.

2 **Create an Oracle user account for Hyperion Performance Suite.**

A user must be created. This user owns the Hyperion Performance Suite database tables and objects that are built and populated during this installation. The user created here is also the user that Hyperion Performance Suite services use to connect to the database. Hyperion suggests naming the user **hypsuite**.

Note The Hyperion Performance Suite repository database user cannot be administrator.

Appropriate database privileges are required to perform the following actions. The tablespaces created or identified in the previous step are used as the default and temporary tablespaces. To create the user, enter the following commands from within the Oracle SQL*Plus utility:

```
create user username identified by password
default tablespace tablespace1 temporary tablespace tablespace2;
```

where *tablespace1* and *tablespace2* are the tablespaces previously created or identified.

3 **Give the user database connection privileges:**

```
grant connect, resource to username;
```

4 **Set all necessary environment variables:**

- **Windows systems**—Set the LIBPATH environment variable to the location of the Oracle Client libraries. Use the **System** option in Control Panel or the **Set** command from the Windows command prompt.
- **UNIX systems**—Set $ORACLE_HOME, LD_LIBRARY_PATH (or SHLIB_PATH on HP-UX), and PATH environment variables. For AIX, you must also set LIBPATH.

  $ORACLE_HOME identifies the location of the Oracle database software.

Creating ODBC Connections

If you are going to use ODBC to access your data source, you must create an ODBC connection. For information on installing and configuring the DataDirect ODBC drivers supplied with Hyperion Performance Suite, go to the following Web address:

`http://www.datadirect.com/download/docs/dochome/index.ssp`

Creating a Database Schema for Row-Level Security Tables

Row-level security enables users to view only those records that match their security profile, no matter what their search criteria. It enables administrators to tag data down in the row level of a database, thus controlling who has read-and-write access to specific information. Row-level security is critical to any application that displays sensitive data such as employee salaries, sales commissions, or customer details. Lack of row-level security could be a big detriment to organizations that want to distribute information to their user community using the Internet and intranets.

If you want to implement row-level security in Hyperion Performance Suite, keep these points in mind:

- The appropriate database client library should be installed on the computer where the Data Access Service is running.
- The data source for the Hyperion Performance Suite repository that has the row-level security table information should be configured.
- For security reasons, the username and password to access the data source should be different from that used for the Hyperion Performance Suite user account.

For more information about row-level security, see the *Hyperion Foundation Administrator's Guide*.

Creating a Database Schema for Usage Tracking

The Usage Service gathers historical information about system activities. It covers usage of the Hyperion Foundation as well as reporting and analytic activities. This service keeps records about logins, document opens, documents closes for select MIME types, jobs run, job output views, and queries processed by the Hyperion Intelligence iServer.

If you want to implement usage tracking in Hyperion Performance Suite, keep these points in mind:

- The username and password to access the usage tracking information may be different from that used for the Hyperion Performance Suite user account; however, an alternate schema is not required for usage tracking.
- An express installation uses the same schema for usage tracking and for the repository. If you want to use a different schema for usage tracking, use the custom installation option and change the schema specified during the installation process.

Configuring a Hyperion SQR Server Installation

If you intend to use Hyperion SQR iServer to run SQR jobs, you must have a valid Hyperon SQR Server license prior to running the Hyperion Performance Suite installer. Also, the version of Hyperion SQR Server that is installed must be the same as the version of Hyperion Performance Suite. During installation, the installer prompts for the following information:

- Root directory for the database client software for Job Service
- Connect string for Job Service (the string utilized by the Job Service to connect to a database when a connection is required during the execution of SQR jobs)
- Location of SQR Server `sqr.ini` file (only required if the installer cannot find it in the standard location, or if the environment variable SQRDIR is not set)

For instructions on installing Hyperion SQR Server, refer to the *Hyperion SQR Server Installation Guide*.

Installing and Configuring a Web Environment

Hyperion Performance Suite can run on various commercial Web servers or as a standalone HTTP server. If you are installing the Hyperion Performance Suite servlets, your Web server must be installed before you install Hyperion Performance Suite.

The Hyperion Performance Suite Support Matrix lists the supported Web environments and indicates whether they must be configured manually or are automatically configured by the installer. You must view the Hyperion Performance Suite Support Matrix to review the supported Web environments.

To view the support matrix, go to the Hyperion Download Center.

If you are using a servlet engine other than JRun 3.1 or Apache Tomcat 4.1.3.0, and you want the installer to automatically configure the Hyperion servlets, your servlet engine *must* be configured before you install Hyperion Performance Suite.

If you are using a Web server other than IIS 5 or Apache, and you want the installer to automatically configure the Hyperion servlets, your Web server must be installed and your servlet engine (if required) configured before you install Hyperion Performance Suite.

Hyperion recommends the use of a secure sockets-capable server (Microsoft IIS, for example) in any production environment, and in any environment where the local network is not protected by some other means (such as a firewall) or where public users are able to access the Web server. The absence of a secure sockets layer connection enables other users on the network to listen to your network traffic and thus access your data.

After the Web server/servlet engine is installed, follow the appropriate procedure to configure your Web environment for use with Hyperion Performance Suite prior to installing Hyperion Performance Suite.

Refer to the following sections for additional information:

- Configuring Apache
- Configuring IBM WebSphere
- Configuring Microsoft IIS 6
- Configuring Sun Java System Web Server

Configuring Apache

To run Apache servlets, Dynamic Shared Object (DSO) must be configured. DSO is a mechanism that provides a way to build a piece of program code in a special format for loading at runtime into the address space of an executable program. The DSO gets knowledge of the executable program symbol set as if it had been statically linked with it in the first place. DSO is required for connecting to external servlet engines such as JRun.

On Microsoft Windows systems, DSO is always enabled. On UNIX systems, instructions for configuring DSO are available at this Web address:

`http://httpd.apache.org/docs/dso.html`

Configuring Microsoft IIS 6

Hyperion Performance Suite supports the Microsoft IIS 6 Web server with JRun 4 through manual configuration. To use this Web environment, you must first acquire JRun 4 and install it. (For information on obtaining JRun 4, go to `http://www.macromedia.com/software/jrun/`.) Then, follow the instructions in "Configuring Microsoft IIS 6 with JRun 4.0" on page 202 to configure this Web environment manually.

Configuring IBM WebSphere

You must change the `wsadmin.sh` script (the last line that executes `com.ibm.ws.bootstrap.WSLauncher`) and add `-Xmx512M` for Java there. During servlet deployment on WebSphere, autodeployment may fail if the WebSphere administrator application (`wsadmin.sh`) is set to use less than 512 MB of memory.

When the Hyperion Performance Suite servlets are deployed to WebSphere on a UNIX system, the user who installs Hyperion Performance Suite should be the user who owns and runs the WebSphere installation. If a different user attempts the Hyperion Performance Suite installation, the installer cannot write to the directory where WebSphere is installed.

Configuring Sun Java System Web Server

If you are using the Sun Java System Web Server, make sure that the IWS_SERVER_HOME environment variable is set to the Sun Java System Web Server installation directory that contains `startconsole`.

For additional information about the setting up the Hyperion Performance Suite servlets with Sun Java System Web Server, see "Configuring Sun Java System Web Server with Native Servlet Engine" on page 212.

Satisfying Temporary Space Requirements

The installer uses a large amount of temporary space during the installation process. As a result, your temporary space may be low. You need approximately 1 GB of temporary space.

To redirect temporary space for the installation to somewhere else, take one of the following actions:

- **Windows systems**—Change the TEMP environment variable (**Start**→**Setting**→**Control Panel**→**System**, Advanced tab, Environment Variables button.)

  Alternatively, you can use this command:

  ```
  setupWinPlatform.exe -is:tempdir directory_name
  ```

  where *`directory_name`* is a directory on your system with space.

- **UNIX systems**—To avoid conflicts with `/tmp` directories created by other users, or access problems to the `/tmp` directory, use this syntax to specify the temporary directory you use when running the installer:

  ```
  setupSolarisPlatform.bin -is:tempdir /lhome/user/tmp
  ```

  When installing on UNIX systems, always use `-is:tempdir /lhome/user/tmp` where `/tmp` is a temporary directory with at least 1GB of space available. InstallShield also uses the `/var` directory for some temporary files, so make sure that `/var` is not full.

  When installing using a remote UNIX session, make sure that you set up display (-DISPLAY) to the terminal where you want the window displayed. Otherwise, the installation will seem to freeze, because the display is waiting for input on the UNIX host computer.

Command Prompt Installation Considerations

When installing using the Command prompt, keep these points in mind:

- To run the installer, you must set up a temporary directory where the installer can uncompress itself and run, for example:

```
mkdir /home/user/tmp
setupSolarisBrioPlatform.bin -is:tempdir /home/user/tmp -is:javaconsole -console
```

  where `setup`*`Solaris`*`BrioPlatform.bin` is the name of the installer and `/tmp` is a temporary directory with at least 1GB of space available. InstallShield also uses the `/var` directory for some temporary files, so make sure that `/var` is not full.

  You can also specify this in a Run instruction at a Command prompt.

- If you are running a UNIX installation from a Windows Telnet session, update your `LMHOSTS.SAM` file in `c:\winnt\system32\drivers\etc` to include the server where the installation is running.

UNIX Installation Mechanisms

In addition to the Java-based GUI installer, you can use these mechanisms to install Hyperion Performance Suite on UNIX systems:

- **Virtual Network Computing (VNC)**—A remote display system which enables you to view a computing desktop environment not only on the computer where it is running, but from anywhere on the Internet and from a wide variety of computer architectures. For more information, go to this Web address:

  `http://www.uk.research.att.com/vnc/`

- **Exceed®**—PC X enterprise connectivity software that enables you to connect Microsoft Windows desktops to UNIX servers and access UNIX applications.

- **Command prompt installer**—See "Command Prompt Installation Considerations" on page 51

Fonts on UNIX

On UNIX systems, to ensure that BQY documents process correctly and have a consistent look and feel, you must make either Type1, TrueType, or OpenType fonts available to the Intelligence Service. For a Windows-like look and feel you should download Microsoft's TrueType Web fonts.

The following procedure is for using Microsoft TrueType Web fonts. If you currently have Type1, TrueType, or OpenType fonts, perform Step 5 and Step 6 below to make your fonts available to the Intelligence Service. If you currently do not have a `fonts.dir` file, you will also need to perform Step 4 below.

To make Microsoft's TrueType Web fonts available to the Intelligence Service:

1. **Download the Microsoft TrueType web fonts from** `http://sourceforge.net/projects/corefonts/` **or some other source.**
2. **Create a new directory.**
3. **Extract each CAB file (`*.exe`) into the newly created directory using the cabextract utility located in** *`INSTALL_HOME`*`/BrioPlatform/bin`**.**

   ```
   INSTALL_HOME/BrioPlatform/bin/cabextract -d directory CAB file
   ```

4. **If you do not have a `fonts.dir` file, create a `fonts.dir` file in the directory containing the font files using the ttmkfdir utility.**
5. **The ttmkfdir utility can be dowloaded from the following URL:**

   `http://packages.qa.debian.org/t/ttmkfdir.html`

Note After installing Hyperion Performance Suite, a copy of this utility can be found in *`INSTALL_HOME`*`/BrioPlatform/bin`.

```
INSTALL_HOME/BrioPlatform/bin/ttmkfdir -d directory -o
directory/fonts.dir
```

6 **Set the environmental variable BQ_FONT_PATH to the directory where the `fonts.dir` file was created. Add this variable to the start-up script to save your changes. In the start-up script:**

```
BQ_FONT_PATH=directory; export BQ_FONT_PATH
```

This environmental variable can contain colon separated paths to directories containing `fonts.dir` files.

7 **Restart the Intelligence Service.**

About Trial Licenses

If you are an existing Hyperion customer and would like to try a new Hyperion product, do not install the trial software into the same *INSTALL_HOME* as your existing Hyperion installation. Install the trial Hyperion product into a new *INSTALL_HOME* using the Hyperion Performance Suite installer.

You cannot add on a trial product to an existing Hyperion product trial, nor can you combine permanent licenses.

To extend an installed trial license:

1 **Contact your local sales office for a new license key.**

2 **Invoke the License Key Extension tool (`lket.bat` on WINDOWS or `lket.sh` on UNIX) in the *`INSTALL_HOME`*`\etc` folder using this syntax:**

- **Windows** - `lket.bat new_license_key`
- **UNIX** - `lket.sh new_license_key`

3 **Restart your services and log on.**

3 Upgrading Hyperion Performance Suite

This chapter explains how to upgrade an existing installation of Hyperion Performance Suite to the 8.3 release.

Caution! These installation options rely on the InstallShield-generated `vpd.properties` file (in `C:\WINNT` or your UNIX *user home*) as well as the *`INSTALL_HOME`*`\log\InstallHistory.xml` file. Do not change or delete these files; doing so may cause undesirable results. In addition, Hyperion recommends that you do not cancel the installation process when doing an upgrade as doing so may reset the default values from the previous installation.

In This Chapter

Upgrade Overview 56

General Upgrade Recommendations 57

Upgrade Recommendations for Distributed Systems 58

Upgrade Prerequisites 59

Upgrade Instructions 60

Upgrade Overview

The upgrade option is available only when you have an existing *INSTALL_HOME* that is a different release from the Hyperion Performance Suite installer release. The installer automatically displays the upgrade option based on the *INSTALL_HOME* that you select as your destination directory. If you select an *INSTALL_HOME* whose release number does not match the installer release, the installer automatically sets the installation setup type to Upgrade. The Upgrade option updates binaries and the repository, adding new columns/indexes as necessary.

Note Upgrading to Hyperion Performance Suite release 8.3.x is only available for installations of Brio Performance Suite 8.1.x and Hyperion Performance Suite 8.2.x.

General Upgrade Recommendations

When upgrading to Hyperion Performance Suite release 8.3, please keep these points in mind:

- If you are upgrading your operating system to match one that is newly supported in release 8.3, first upgrade your operating system and test your Hyperion Performance Suite installation; then upgrade to Hyperion Performance Suite release 8.3.
- Hyperion Performance Suite release 8.3 supports different Web servers than previous releases. If your previous Web server is no longer supported in release 8.3, you can either upgrade your Web server or install a new Web server; or manually deploy the servlets on your older Web server.
- Add new components only after the upgrade is complete and tested. This helps to ensure the integrity of the data in your system.
- If you are upgrading from ReportMart 1.x, first upgrade to Brio Portal 6.0 using the 6.0 CD, then upgrade to Brio Portal 7.x, then install Hyperion Performance Suite and migrate your Brio Portal 7.x installation. Please contact Hyperion Solutions Consulting Services for assistance.

Upgrade Recommendations for Distributed Systems

If you are upgrading a distributed system, Hyperion recommends that you take these steps:

1. Shut down all services and Web servers.
2. Install or upgrade the services (or new Web server, if needed).
3. Test the servlets. (This can be done on a different Web server than the one used in your prerelease 8.3 installation; that is, it can be migrated at installation time.
4. Run the installer on each services host.
5. Test the services.

Upgrade Prerequisites

Table 3 lists the prerequisites that you must meet before upgrading to Hyperion Performance Suite release 8.3.

Table 3 Upgrade Requirements

Platform	Requirements
All platforms	When upgrading, please make sure that you complete all prerequisite tasks specified in "Required Pre-Installation Configuration Procedures" on page 35.

IBM DB2	Tables v8_param_values and v8_jobinpmultvals created before release 8.2 are in a tablespace with only 4KB page size. For release 8.2.x and later, these tables need to be in a tablespace with a page size of 8KB.
	
	Hyperion has provided a script to migrate these tables to your new tablespace. First, follow the DB2 pre-installation instructions for creating the required regular and system temporary tablespaces (see "Configuring IBM DB2" on page 38). Then, copy the script from the Servers CD Disk 1 `\Misc\DB2` directory to a temporary location on your system.
	
	Caution! To avoid corruption of the required tables, Hyperion recommends running `pre_install_db2_upgrade.cmd` line by line using the DB2 console. Make sure that the current statement processes successfully before running the next one.
	
	To use the script on Microsoft Windows systems:
	1. Edit this file to define proper values for parameters that appear in angle brackets.
	NOTE: Make sure that the user you are connected as to execute this script is the owner of modified tables and has appropriate permissions to perform import, export, and delete operations.
	2. Run this command:
	`db2cmd pre_install_db2_upgrade.cmd`
	
	To use the script on UNIX systems:
	1. Edit this file to define proper values for parameters that are enclosed in angle brackets.
	NOTE: Make sure that the user you are connected as to execute this script is the owner of modified tables and has appropriate permissions to perform import/export/delete operations.
	2. Run this command to add executable attributes to the `pre_install_db2_upgrade.cmd` file:
	`chmod +x pre_install_db2_upgrade.cmd`
	NOTE: If the file contains line feed characters, use the dos2unix utility to remove them.
	3. Run this command to define standard DB2 environment:
	`source <SQLLIB>/db2cshrc`
	4. Run this command:
	`./pre_install_db2_upgrade.cmd`

Upgrade Instructions

If you have a previous installation of Performance Suite (release 8.1.x or 8.2.x) and you want to upgrade to the Hyperion Performance Suite release 8.3:

1. **Run the setup file for your system from the Hyperion Performance Suite Servers CD.**

   Setup file names are:

   - **AIX**—`setupAixPlatform.bin`
   - **HP**—`setupHpPlatform.bin`
   - **Linux**—`setupLinuxPlatform.bin`
   - **Solaris**—`setupSolarisPlatform.bin`
   - **Windows**—`setupWinPlatform.exe`

2. **Install to your existing *INSTALL_HOME*.**

3. **If prompted, select the same installation setup type (Express or Custom) that you previously chose.**

   The installer performs an Express or Custom upgrade, prompting for information as necessary.

4. **After the upgrade is complete, proceed to Chapter 7, "Postinstallation Tasks," for information about additional tasks that you must perform and for instructions on testing your upgraded installation.**

4 Installing the Server Components: Express Mode

This chapter provides step-by-step instructions for installing Hyperion Performance Suite Servers using the Express installation option.

In This Chapter

Express Installation Overview 62

Information Required for Express Server Installation 63

Installing Hyperion Performance Suite Server Components: New Express Installation 64

Express Installation Overview

The Express installation option installs all server components authorized by your license keys on a single computer. Select this option if you want to install Hyperion Performance Suite in one *INSTALL_HOME* on one computer. A single-machine installation is best used only for demonstration purposes; it is not recommended for production environments.

Note If you have a previous installation of Hyperion Performance Suite and you want to upgrade to the newest release, see "Upgrading Hyperion Performance Suite" on page 55.

Do not use the Express installation option if you want to take any of the following actions:

- Use an authentication system other than the native authentication system.
- Change service agent names or port numbers.
- Implement row-level security.
- Implement a separate schema for usage tracking data.
- Install Hyperion Performance Suite on a distributed system.
- Replicate services.
- Install Web Services.
- Customize Web application and servlet URLs.

Use the Custom installation option instead. For more information, see Chapter 5, "Installing the Server Components: Custom Mode."

During an Express installation, where possible, the installation software detects and uses needed parameters from the system. Default values are provided for parameters that you can override.

Note The Hyperion Performance Suite installer is a Java-based installer that runs on both Microsoft and UNIX windowing systems. For information on using the installer in console mode, see "Command Prompt Installation Considerations" on page 51.

After you have installed Hyperion Performance Suite, you should test the installation to make sure it is working. Refer to Chapter 7, "Postinstallation Tasks" on page 101 for more information.

Information Required for Express Server Installation

Use the procedures in this section to install Hyperion Performance Suite using the Express installation option. Table 4 lists the information you need to successfully perform this installation. (All of this information is accessible to your database or system administrator.)

In addition to the information in Table 4, if you want the installer to configure Hyperion SQR, you need the ODBC data source name.

Table 4 Required Information

Repository Database	Required Information
For IBM DB2:	■ Your Hyperion Performance Suite license keys ■ The DB2 instance name, database name, username, and password ■ If you want to run SQR Jobs, the location of the `sqr.exe` or `sqrw.exe` file and the SQR release number
For Microsoft SQL Server:	■ Your Hyperion Performance Suite license keys ■ The Microsoft SQL Server database host name, server name, database name, port, username, and password ■ If you want to run SQR Jobs, the location of the `sqr.exe` or `sqrw.exe` file and the SQR release number **Note:** Hyperion recommends that the installation not be done as the *sa* user for the repository database
For Oracle:	■ Your Hyperion Performance Suite license keys ■ The Oracle database host, port, SID (or Service Name), and connect string (depending upon your system's configuration, you may find the Oracle database host, port, SID [or Service Name], and connect string in the Oracle `tnsnames.ora` file) ■ If you want to run SQR Jobs, the location of the `sqr.exe` or `sqrw.exe` file and the SQR release number

Installing Hyperion Performance Suite Server Components: New Express Installation

Follow these instructions to install Hyperion Performance Suite server components using the Express installation option, which uses default settings detected from your system.

These instructions assume that this is a new or first-time Express installation on a given host computer, with all Hyperion Performance Suite components being installed on a single computer that is running Microsoft Windows with the JRun 3.1 built-in Web server.

Caution! Make sure you set up adequate temporary space *before* running the installer. See "Satisfying Temporary Space Requirements" on page 50.

To install Hyperion Performance Suite server components:

1 **Run the setup file for your system from the Hyperion Performance Suite Servers CD.**

Setup file names are as follows:

- **AIX**—`setupAixPlatform.bin`
- **HP**—`setupHpPlatform.bin`
- **Linux**—`setupLinuxPlatform.bin`
- **Solaris**—`setupSolarisPlatform.bin`
- **Windows**—`setupWinPlatform.exe`

Note On UNIX systems, to avoid conflicts with `/tmp` directories created by other users, or access problems to the `/tmp` directory, use this syntax to specify the temporary directory you use when running the installer:

`setup`*`Solaris`*`Platform.bin -is:tempdir /home/`*`user`*`/tmp`

where `/tmp` is a temporary directory with at least 1GB of space available. InstallShield also uses the `/var` directory for some temporary files, so make sure that `/var` is not full. You should make sure this temporary directory does not contain important files because the directory contents may be removed during an uninstallation.

When installing using a remote UNIX session, make sure that you set up display (--DISPLAY) to the terminal where you want the window displayed. Otherwise, the installation seems to freeze because the display is waiting for input on the UNIX host computer.

The InstallShield wizard starts and displays a progress bar while the JVM is loaded. It is followed by the Hyperion Performance Suite Servers splash screen and then by the Installation Language dialog box.

2 **Select the language that you want the installer to use during the installation process, then click Next to display the Welcome dialog box.**

3 **Click Next on the Welcome dialog box to proceed with the installation.**

4 **Read the Software License Agreement. To continue with the installation, select the I accept the terms of the license agreement option and click Next.**

5 **In the Licence Key Entry dialog box, enter one or more license keys, then click OK.**

You must enter at least one license key to install the Hyperion Performance Suite files to your computer. When prompted by the installation software, enter the license key provided with the installation media. The key should be in the following format:

```
6L55-1P2E2
```

where dashes separate license keys. You may have multiple license keys, depending on the number of Hyperion Performance Suite components that you purchased. You may add more keys at any time.

Note For information about trial licenses, see "About Trial Licenses" on page 53.

License keys are stored in *user_home*`/InstallHistory.xml` where *user_home* means `C:/Documents and Settings/`*`username`* on Windows or your user home (often `/home/`*`username`*) on UNIX.

Note License keys are remembered for all subsequent installations on this computer. License keys are not case-sensitive.

6 **Verify that the correct components are listed, then click Next.**

To install a product that is not listed, click **Add License Key**, enter the appropriate license key, then click **OK**.

This sample installation installs the following products on a single computer:

- Hyperion Foundation
- Hyperion Intelligence Server
- Hyperion Intelligence iServer
- Hyperion SQR iServer

7 **In the Setup Type dialog box, select the Express installation setup type and click Next.**

8 **Select the destination directory (that is, an *INSTALL_HOME*), then click Next.**

Destination directory choices are as follows:

- **Default installation directory**—`C:\Program Files\Brio\Brio8` on Windows; `/opt/Brio/Brio8` on UNIX.
- **New directory**—Type the full path in the Directory Name text box.
- **Pre-existing directory**—Select **Browse,** then navigate to the desired directory and click **OK**.

If the directory does not exist, the installer creates it.

9 **Select the language and formatting locale for your Hyperion Performance Suite system to use.**

10 **Select the repository database type, then click Next.**

Select **IBM DB2**, **Microsoft SQL Server**, or **Oracle** as the Hyperion Performance Suite repository database type.

This sample installation uses Oracle for the repository database.

Note If configuring Hyperion SQR iServer and the installer has not found the location of the `sqr.ini` file, you are prompted to enter its location. Enter or browse to the file location, or select to not configure SQR iServer on this system. If the location you enter is wrong, the installer shows a warning panel; you should go back and enter the valid location.

11 **If configuring Hyperion SQR iServer, verify the path to the root directory for your database client software and click Next.**

If necessary, use the Browse button to locate the correct directory. This information must be provided to continue installation.

For this sample installation, the location of the Oracle client libraries is as follows:

`C:\Oracle\Ora8.1.6`

12 **Enter the repository database connection parameters, then click Next.**

You recorded this information when you configured your database. For more information, see the section specific to your database type under "Required Pre-Installation Configuration Procedures" on page 35.

If you are in doubt as to the correct information necessary to fill these fields, contact your database or system administrator.

Note Make sure you have an ODBC DSN before installing Job Service.

The terms displayed on this dialog box may vary, depending on which repository database type you selected, as each database vendor has a different nomenclature for describing the connectivity to their database. Table 5 describes the terms used by each of the supported database types.

Table 5 Database Nomenclature

Database Type	Database Service Name	Connect String for Job Service	Description
IBM DBS	Database Name	Data Source Name (ODBC DSN)	The data source is the name of the database created for Hyperion Performance Suite.
Microsoft SQL Server	Data Source Name (ODBC DSN)	Data Source Name (ODBC DSN)	The data source is the ODBC data source configured where the metadata is being installed. The database server could be on the local or remote computer.
Oracle	Service Name (SID)	Net Service Name (TNS Name)	The connect string in this case is a service name entry in the `tnsnames.ora` file on the computer where Hyperion Performance Suite is being installed.

- **User Name**—Username for the database account that was created as part of the pre-installation requirements

  The default user name that Hyperion previously recommended is *hypsuite*. Remember that IBM DB2 is case-sensitive.
- **Password**—Password that you assigned to the Hyperion Performance Suite user database account
- **Computer Running the Database**—Host name or IP address of the server utilized by the Repository Service
- **Port**—Database host port on which the database listens for connection requests from clients
- **Database Service Name**—Database or instance used to store the Hyperion Performance Suite metadata
- **Connect String for Job Service**—String utilized by the Job Service to connect to a database when a connection is required during the execution of SQR jobs

Note In an Express installation, the installer looks for the `sqr.ini` file in the standard location, then parses the file to get the SQR (that is `sqr.exe` or `sqrw.exe`) location. If it cannot find the SQR location, it also checks SQRDIR. If it finds the SQR location, it validates that the SQR EXE file really is in that directory, sets all the variables for you, and continues without prompting or having you confirm the location.

After you supply the database connectivity information, the installer verifies the information by connecting to the database.

- If the connection succeeds, proceed to Step 13.
- If the connection fails, the installer displays an error message. View the error log and correct your database connectivity parameters.

Note You cannot proceed with installation if the connection to your database fails. Database connectivity is required for successful installation of the Hyperion Performance Suite service components.

13 **Verify your installation choices, then click Next.**

Click **Back** to change any of the listed parameters.

Note You cannot change service names or port numbers in an Express installation. If you need to change service agent names or port numbers, or specify different components, go back to the installation Setup Type dialog box and select the Custom installation option.

14 **Select whether to configure the mail server for the Event Service, then click Next.**

15 **If you selected the option to set up the Event Service mail server, identify your SMTP or ASMTP mail server and the e-mail account name for sending notifications, then click Next.**

If you have an ASMTP mail server (which requires authentication), select **Requires Authentication**, then enter a username and password.

For the Event Service to provide e-mail notifications, you must specify the name of your outgoing mail server. If you do not know the name of your mail server, accept the default values to proceed with the installation. After you install all Hyperion Performance Suite components and start the Hyperion Performance Suite Service, you can use the Service Configurator to specify the correct mail server.

If desired, select **Send Test Email** to test the notification server, enter an e-mail address to send the test e-mail to, then click **OK**.

Note If you are using an ESMTP mail server, you need to do additional configuration. See the *Hyperion Foundation Administrator's Guide* for more information.

16 **If configuring Hyperion SQR and the connect string entered in Step 12 fails, re-enter the database connect string.**

You are prompted for the connect string only if necessary; otherwise, the installer validates the SQR connection.

If the connection fails again, the installer enables you to continue the installation without a successful connection; however, you must configure this information manually to run SQR jobs.

If you are not installing Hyperion Intelligence iServer or Hyperion Intelligence Server, proceed to Step 19.

17 **UNIX ONLY: If the BQ_FONT_PATH environment variable is not set, specify the directory that contains the `fonts.dir` file.**

For more information, see "Fonts on UNIX" on page 52.

18 **If you are installing Hyperion Intelligence iServer or Hyperion Intelligence Server, supply database parameters for data sources accessed by the Data Access Service.**

If you prefer to add this data after installation, proceed to Step 19.

On Windows, a data source that contains sample data is listed. The sample data is installed by default in an Express installation.

a. Click **Add Data Source**.

   Select the connectivity protocol and database type for the data source. Depending on the connectivity protocol and database type selected, you may also need to supply the data source name or the OLE DB server/file. The data source name may be the database alias, ODBC DSN, Database Server Name, DS Name, or Net Service Name (TNS Name).

b. Click **Add**.

   The data source is listed in the Data Access Service Data Source list.

c. Click **Next**.

19 **If you are installing the Hyperion Performance Suite servlets, select a Web server, then click Next.**

This sample installation uses the JRun built-in Web server.

Note Please keep this points in mind:

- If you are installing the Hyperion Performance Suite servlets and want the installer to automatically deploy the servlets, you must have a Web server installed prior to installing Hyperion Performance Suite unless you select to install the JRun 3.1 built-in Web server or Apache Tomcat 4.1.30 standalone Web server.
- When deploying the Hyperion Performance Suite servlets to IBM WebSphere on a UNIX system, the user who installs Hyperion Performance Suite should be the user who owns and runs the WebSphere installation. If a different user attempts the Hyperion Performance Suite installation, the installer cannot write to the directory where WebSphere is installed.
- Automatic deployment of the servlets on BEA WebLogic 8.x may take a while because of its complexity. Hyperion recommends that you view the Web server log file to monitor the progress of the deployment and confirm its completion.
- For information on increasing Web server performance, see "Increasing Performance with -Dbrio.content.path" on page 108.

The installer checks for a Web server and selects a default. Select from the supported Web servers, or other (requires manual configuration). The Web servers listed vary depending on your operating system. For a list of supported Web servers and servlet engines, view the support matrix on the Hyperion Download Center.

- If you are using BEA WebLogic 6.x, Hyperion recommends that you deploy the Hyperion Performance Suite servlets with precompiled JSP pages. This step must be completed *before* you continue the installer. For more information, see "Precompiling WebLogic JSPs" on page 167.
- If you select Other Web server as your Web server type, the installer does not prompt you to install and configure a servlet engine. You must manually configure these components. See Appendix A, "Manually Setting Up the Hyperion Performance Suite Servlets," for information on setting up your Web server.

20 **If you select the JRun 3.1 built-in Web server, enter the password for the JRun Management Console.**

21 **If you selected a Web server other than JRun 3.1 or Other, select whether to install and configure the servlet engine.**

If you selected a Web server other than JRun 3.1, the installer prompts you to install and configure a servlet engine. The options listed and the dialog boxes displayed vary, depending on the Web server you selected. It then detects the servlet engine type and confirms its location.

The servlet installation requires write access to the Web server configuration files. For more information, see Chapter 7, "Postinstallation Tasks."

22 **If requested, specify the JDK location, then click Next.**

This directory should exist and contain the `\lib\tools.jar` file.

23 **Verify the services and components to be installed.**

Click Back to change any of these choices or other parameters.

24 **Click Next to continue with the installation.**

The installer automatically copies files into target directories, populates the database, and configures the system for initial use. During this portion of the installation, a progress bar is displayed.

For a list of all Hyperion Performance Suite services and components and the products with which they are installed, see Table 1 on page 21.

If you are installing the servlets with automatic configuration, the installer copies servlet-specific files, creates virtual directory mappings in the Web server configuration files, and confirms the Web server host and port. (For details on these files, see "Servlet Configuration Details" on page 126.) The installer then configures the servlet engine, and precompiles and deploys the Web components.

Note If Sun Java System Web Server is being deployed automatically, the installer stops the Web server before starting the deployment.

This step may take a while, depending on your system configuration.

25 **If prompted, select whether to restart the Web server.**

Caution! The Web server must be restarted before you can use the Hyperion Performance Suite servlets. You should have sufficient privileges for restarting the Web Server.

Note If you elected to install the JRun 3.1 servlet engine with IIS 5 or Apache 1.3.x, then the JRun 3.1 built-in Web server is deployed as a diagnostic tool. For a production system, you should probably turn off the service. Customer Support uses the JRun built-in Web server to troubleshoot any problems that you encounter on your primary Web server.

26 **Click Next to complete the installation.**

Any errors that occurred during the installation process are listed on the dialog box that is displayed. The installer automatically selects the View Log option for you to review `Install.log`. If you encounter errors, view this log for information on diagnosing the failure.

For more information about installed files and logs, see Chapter 7, "Postinstallation Tasks."

27 **On Windows systems, if you installed services, specify whether to restart your system, then click Finish to complete the installation.**

If you installed services, you must restart your system to successfully complete the installation.

Note On UNIX, log off the account if you are prompted to do so.

Often this happens when there has been a change to an environment variable (made in the `.login` file).

28 Configure Job Service for Microsoft SQL Server:

Note Complete this step only if you intend to run SQR jobs against a Microsoft SQL Server database, or to use printers and network resources for BQY jobs.

To run any SQR engine, or to use printers and network resources for BQY jobs, configure the corresponding Job Service to run as a *user* instead of as *System*.

a. In the Control Panel, open **Administrative Tools**.

b. Open **Services**.

c. Select the **BrioPlatform**_*hostname_gsmhost_gsmport*, then select **Action→Properties**.

d. Select the **Log On** tab.

e. Select **This account** and specify a user other than Local System. Use the user account that you created for Hyperion Performance Suite (for example, hypsuite).

5 Installing the Server Components: Custom Mode

This chapter provides step-by-step instructions for installing Hyperion Performance Suite Server using the Custom installation option.

In This Chapter

Custom Installation Overview 76

Information Required for Custom Server Installation 77

Installing Hyperion Performance Suite Server Components: Custom Installation 78

Custom Installation Overview

Use the Custom installation option if you want to perform these operations:

- Distribute the Hyperion Performance Suite server components across multiple computers.
- Replicate services.
- Change service agent names or port numbers.
- Use an authentication system other than the native authentication system.
- Implement row-level security.
- Implement a separate schema for usage tracking data.
- Implement the Web Services feature for use with the Tomcat Web environment.

You must use the Custom installation option to install Hyperion Performance Suite Server components in multiple *INSTALL_HOMES* on one or more hosts.

Note If you have a previous installation of Hyperion Performance Suite and you want to upgrade to the newest release, see "Upgrading Hyperion Performance Suite" on page 55.

A Custom installation installs server products enabled by the license key on the host where the installer is being run, but does not automatically install other system dependent components. Over the total installation, the installer is unable to guarantee that all dependent components are present; this is flagged as the services are requested and started.

Note The Hyperion Performance Suite installer is a Java-based installer that runs on both Microsoft and UNIX windowing systems. For information on using the installer in console mode, see "Command Prompt Installation Considerations" on page 51.

After you have installed Hyperion Performance Suite Server, you should test the installation to make sure it is working. Refer to Chapter 7, "Postinstallation Tasks" on page 101 for more information.

Information Required for Custom Server Installation

Use the installation procedures in this section to install Hyperion Performance Suite Server components using the Custom installation option. Table 6 lists the information you need to successfully perform this installation. (All of this necessary information is accessible to your database or system administrator.)

In addition to the information in Table 6, if you want the installer to configure Hyperion SQR, you need the ODBC data source name.

Table 6 Required Information

Repository Database	Required Information
For IBM DB2:	■ Your Hyperion Performance Suite license keys ■ The DB2 instance name, database name, username, and password ■ If you want to run SQR Jobs, the location of the `sqr.exe` or `sqrw.exe` file and the SQR release number
For Microsoft SQL Server:	■ Your Hyperion Performance Suite license keys ■ The Microsoft SQL Server database host name, server name, database name, port, username, and password ■ If you want to run SQR Jobs, the location of the `sqr.exe` or `sqrw.exe` file and the SQR release number **Note:** The Hyperion Performance Suite repository database user cannot be administrator, nor can it be a user who is assigned the System Administrators role.
For Oracle:	■ Your Hyperion Performance Suite license keys ■ The Oracle database username and password, host, port, SID (or Service Name), and connect string (depending upon your system's configuration, you may find the Oracle database host, port, SID [or Service Name], and connect string in the Oracle `tnsnames.ora` file) ■ If you want to run SQR Jobs, the location of the `sqr.exe` or `sqrw.exe` file and the SQR release number

Installing Hyperion Performance Suite Server Components: Custom Installation

Follow these instructions to install Hyperion Performance Suite Server components using the Custom installation option.

These instructions assume that this is a new or first-time Custom installation on a given host computer, with all Hyperion Performance Suite server components being installed on a single computer that is running Microsoft Windows with the Apache 2.x/Tomcat 4.1.30 Web environment.

Caution! Make sure that you set up adequate temporary space *before* running the installer. See "Satisfying Temporary Space Requirements" on page 50.

To install Hyperion Performance Suite server components:

1. **Run the setup file for your system from the Hyperion Performance Suite Servers CD.**

   Setup file names are as follows:

   - **AIX**—`setupAixPlatform.bin`
   - **HP**—`setupHpPlatform.bin`
   - **Linux**—`setupLinuxPlatform.bin`
   - **Solaris**—`setupSolarisPlatform.bin`
   - **Windows**—`setupWinPlatform.exe`

Note On UNIX systems, to avoid conflicts with `/tmp` directories created by other users, or access problems to the `/tmp` directory, use this syntax to specify the temporary directory that you use when running the installer:

`setup`*`Solaris`*`Platform.bin -is:tempdir /home/`*`user`*`/tmp`

Also, always use `-is:tempdir /lhome/`*`user`*`/tmp` where `/tmp` is a temporary directory with at least 1GB of space available. InstallShield also uses the `/var` directory for some temporary files, so make sure that `/var` is not 100% full. You should make sure this temporary directory does not contain important files because the directory contents may be removed during an uninstallation.

When installing using a remote UNIX session, make sure that you set up display (-DISPLAY) to the terminal where you want the window displayed. Otherwise, the installation seems to freeze because the display is waiting for input on the UNIX host computer.

The InstallShield wizard starts and displays a progress bar while the JVM is loaded. It is followed by the Hyperion Performance Suite Servers splash screen and then by the Installation Language dialog box.

2 **Select the language that you want the installer to use during the installation process, then click Next.**

3 **Click Next on the Welcome dialog box to proceed with the installation.**

4 **Read the Software License Agreement. To continue with the installation, select the I accept the terms of the license agreement option and click Next.**

5 **Enter one or more license keys, then click OK.**

You must enter at least one license key to install the Hyperion Performance Suite files to your computer. The key should be in a format similar to:

```
6L55-1P2E2
```

where dashes separate license keys. You may have multiple license keys, depending on the number of Hyperion Performance Suite components that you purchased. You may add more keys at any time.

Note For information about trial licenses, see "About Trial Licenses" on page 53.

License keys are stored in `user_home/InstallHistory.xml` where `$D` means `C:/Documents and Settings/username` on Windows or your user home (often `/home/username`) on UNIX.

Note License keys are remembered for all subsequent installs on this computer. License keys are not case-sensitive.

6 **Verify that the correct components are listed, then click Next.**

To install a product that is not listed, select **Add License Key**, enter the appropriate license key and click **OK**.

This sample installation installs the following products on a single computer:

- Hyperion Foundation
- Hyperion Intelligence Server
- Hyperion Intelligence iServer
- Hyperion SQR iServer

7 **Select the Custom installation setup type, then click Next.**

8 **Select the destination directory (that is, an *INSTALL_HOME*), then click Next.**

Destination directory choices are as follows:

- **Default installation directory**—`C:\Program Files\Brio\Brio8` on Windows; `/opt/Brio/Brio8` on UNIX.
- **New directory**—Type the full path in the Directory Name text box.
- **Pre-existing directory**—Select **Browse,** then navigate to the desired directory and click **OK**.

If the directory does not exist, the installer creates it.

9 **Select the Hyperion Performance Suite components and services that you want to install on this computer.**

To select a component or service, select its check box. To deselect a component or service, clear its check box. To select all components click Select All. To deselect all components click, Clear All.

For a list of all Hyperion Performance Suite services and components, their replicability, and the products with which they are installed, see Table 1 on page 21. See also "Command Prompt Installation Considerations" on page 51.

Note In a distributed system, the GSM and its co-requisite Name Service should be installed first.

For optimal performance, keep in mind the following points:

- **Authentication Service**—Use the same name and port number for the installation and configuration of Authentication Service. (*Name* is case sensitive.)
- **Authorization Service**—For better performance, install Authorization Service on the same computer as Event Service or Repository Service.

- **Event Service**—Name Service, Repository Service, and Event Service should all be installed together. You can select Name Service and Repository Service without selecting Event Service, but you must reinstall them if you later want to install Event Service.
- **GSM/Name Service**—Installs GSM and Name Service on the same computer.
- **Intelligence Service**—Must be installed with Job Service.
- **Job Service**—Must reside on the same computer as the Intelligence Service, but it can be replicated by itself.

  Any changes to executables, SQR executables, and database connectivity are immediately propagated to the Job Services on the appropriate host and to all Service Brokers. Jobs using newly configured applications, executables, and subservices can be executed without restarting any service.
- **Logging Service**—There can be only one Logging Service per system.
- **Repository Service**—Must reside on the same computer as Authorization Service and Publisher Service.
- **Publisher Service**—Must reside on the same computer as the Repository Service and Authorization Service.
- **Session Manager**—Installation of Session Manager on the same computer as the GSM is desirable. Session Manager can be replicated by itself. For optimal load balancing, Hyperion recommends that every host include a Session Manager.
- **Web Services**—Requires Tomcat installation. Web services are Web-based enterprise applications that use open, XML-based standards and transport protocols to exchange data with calling clients. The operational commands to administer Tomcat are made available as Web services that utilize SOAP message formats.

During the installation process, you may customize the name and port number for each service that you install. When you specify a port number, ensure that it is unique and available for the host.

10 **Select the language and formatting locale for your Hyperion Performance Suite system to use.**

11 **Select a temporary directory for the services cache, then click Next.**

12 **Select the repository database type, then click Next.**

Select **IBM DB2**, **Microsoft SQL Server**, or **Oracle** as the Hyperion Performance Suite repository database type.

This sample installation uses Oracle for the repository database.

Note If configuring Hyperion SQR iServer and the installer has not found the location of the `sqr.ini` file, you are prompted to enter its location. Enter or browse to the file location, or select to not configure SQR iServer on this system. If the location you enter is wrong, the installer shows a warning panel; you should go back and enter the valid location.

13 **If configuring Hyperion SQR iServer, verify the path to the root directory for your repository database client software, then click Next.**

If necessary, use the Browse button to locate the correct directory. This information must be provided to continue installation.

For this sample installation, the location of the Oracle client libraries is as follows:

`C:\Oracle\Ora8.1.6`

14 **Enter the repository database parameters, then click Next.**

(If you are not installing Hyperion Intelligence iServer or Hyperion Intelligence Server, proceed to Step 18.)

You recorded this information when you configured your repository database. (See the section specific to your database type under "Required Pre-Installation Configuration Procedures" on page 35.) If you are in doubt as to the correct information necessary to fill these fields, contact your database or system administrator.

Note Make sure that you have an ODBC DSN before installing Job Service.

The terms displayed on this dialog box may vary, depending on which repository database type you selected. Table 7 describes the terms used by each of the supported database types.

Table 7 Database Nomenclature

Database Type	Database Service Name	Connect String for Job Service	Description
IBM DBS	Database Name	Data Source Name (ODBC DSN)	The data source is the name of the database created for Hyperion Performance Suite.
Microsoft SQL Server	Data Source Name (ODBC DSN)	Data Source Name (ODBC DSN)	The data source is the ODBC data source configured where the metadata is being installed. The database server could be on a local or remote computer.
Oracle	Service Name (SID)	Net Service Name (TNS Name)	The connect string is a service name entry in the `tnsnames.ora` file on the computer where Hyperion Performance Suite is being installed.

- **User Name**—Username for the database account that was created as part of the pre-installation requirements

  The default user name that Hyperion recommends is hypsuite. Remember that IBM DB2 is case-sensitive.
- **Password**—Password that you assigned to the Hyperion Performance Suite user database account
- **Computer Running the Database**—Host name or IP address of the server utilized by the Repository Service
- **Port**—Database host port on which the database listens for connection requests from clients
- **Database Service Name**—Database or instance used to store the Hyperion Performance Suite metadata
- **Connect String for Job Service**—String utilized by the Job Service to connect to a database when a connection is required during the execution of SQR jobs

After you supply the database connectivity information, the installer verifies the information by connecting to the database.

- If the connection succeeds, proceed to Step 15.
- If the connection fails, the installer displays an error message. View the error log and correct your database connectivity parameters.

Note You cannot proceed with installation if the connection to your database fails. Database connectivity is required for successful installation of the Hyperion Performance Suite service components.

15 **Enter the username and password for the database that you want to use for usage tracking, then click Next.**

The user name and password to access the usage tracking information may be different from that used for the Hyperion Performance Suite user account; however, an alternate schema is not required for usage tracking.

16 **If you are installing Hyperion Intelligence iServer or Hyperion Intelligence Server and you selected the option to implement row-level security, enter the connectivity information for your repository database, then click Next.**

Note The settings in this dialog box default to the first item in each drop-down list. Make sure you select the correct connectivity and database type.

If you want to implement row-level security, keep these points in mind:

- The appropriate database client library should be installed on the computer where the Data Access Service is running.
- The data source for the repository that has the row-level security tables information should be configured.
- For security reasons, the username and password to access it should be valid, and should be different from that used for the Hyperion Performance Suite user account.

17 **Enter the username and password for the row-level security database.**

Use a username and password that you created specifically for this purpose.

After you supply the row-level security database parameters, the installer attempts to verify the information.

If the connection fails, the installer displays an error message. View the error log and correct your row-level security database parameters.

Note You cannot proceed with installation if the connection to your data source database fails. Database connectivity is required for successful installation of Hyperion Performance Suite.

18 **Optional:** Customize the GSM and Name Service parameters if desired, then click **Next**.

Caution! The GSM hostname cannot begin with numerals. The BI and the DAS cannot communicate with the GSM if its hostname begins with numerals.

If you are installing a distributed Hyperion Performance Suite system with a remote GSM/Name Service, enter values for the remote GSM/Name Service. If the values you enter here do not match the values entered for the GSM/Name Service when the GSM was initially installed, the Hyperion Performance Suite system will not operate properly.

19 **Enter** the host name and port number for the Web server in use, then click **Next**.

20 **Optional:** Customize the service names and ports if desired, then click **Next**.

The default service names use the format *service_host*.

21 If you are installing Web Services, complete the following steps; otherwise, skip to Step 22. Note that the Web Services feature is supported only with Apache Tomcat installations.

a. Optional: Customize the application name and Web service name if desired, then click Next.

b. Enter or browse to the location of the Web server on which the servlets are to be installed.

c. Select whether to install and configure a new Tomcat servlet engine or use an existing installation, then click Next.

d. If requested, specify the JDK location, then click Next.

This directory should exist and contain the `\lib\tools.jar` file.

22 Select whether to configure the mail server for the Event Service, then click **Next**.

23 **If you selected to set up the Event Service mail server, identify your SMTP or ASMTP mail server and the e-mail account name for sending notifications, then click Next.**

If you mail server requires authentication, enter a username and password.

In order for the Event Service to provide e-mail notifications, you must specify the name of your outgoing mail server. If you do not know the name of your mail server, accept the default values to proceed with the installation. After you install all Hyperion Performance Suite components and start the Hyperion Performance Suite Service, you can use the Service Configurator to specify the correct mail server.

24 **Optional: Click Send Test Email to test the notification server, enter an e-mail address to send the test e-mail to, then click OK.**

Note If you are using an ESMTP mail server, you need to do additional configuration. See the *Hyperion Foundation Administrator's Guide* for more information.

If you are installing Hyperion SQR iServer, complete Step 25–Step 27; otherwise, proceed to Step 28.

25 **If you are installing Job Service for use with Hyperion SQR iServer, set the SQR location, then click Next.**

Note In a Custom installation, the installer looks for the `sqr.ini` file in the standard location, then parses the file to get the SQR (that is, `sqr.exe` or `sqrw.exe`) location. If it cannot find the SQR location, it also check SQRDIR. If it finds the SQR location, it validates that the SQR EXE file really is in that directory, sets all the variables for you; however, it always asks you to validate the location and the release of SQR that is installed.

26 **Verify the release of Hyperion SQR installed on your system, then click Next.**

If the Hyperion SQR release listed is not correct, click Back and change the SQR directory from the default to the one that you want to use.

27 **If installing Job Service and the connect string entered in Step 14 fails, re-enter the database connect string.**

You are prompted for the connect string only if necessary; otherwise, the installer validates the SQR connection.

If the connection fails again, the installer enables you to continue the installation without a successful connection; however, you must configure this information manually to run SQR jobs.

If you are not installing Hyperion Intelligence iServer or Hyperion Intelligence Server, proceed to Step 30.

28 **If you are installing Hyperion Intelligence iServer or Hyperion Intelligence Server, specify the location of the Intelligence Service cache directory.**

29 **If you are installing Hyperion Intelligence iServer or Hyperion Intelligence Server, enter data source information for the Data Access Service.**

If you prefer to add this data after installation, proceed to Step 30.

On Windows, a data source that contains sample data is listed.

a. Click **Add Data Source**.

   Select the connectivity protocol and database type for the data source. Depending on the connectivity protocol or database type selected, you may also need to supply the data source name or the OLE DB server/file. The data source name may be the database alias, ODBC DSN, Database Server Name, DS Name, or Net Service Name (TNS Name).

b. Click **Add**.

   The data source is added to the Data Access Service Data Source list.

c. Click **Next** to continue.

30 **If you are installing the Hyperion Performance Suite servlets, select a Web server, then click Next.**

Note Please keep this points in mind:

- If you are installing the Hyperion Performance Suite servlets and want to use the installer's automatic deployment feature, you must have a Web server installed prior to installing Hyperion Performance Suite unless you select to install the JRun 3.1 built-in Web server or the Apache Tomcat 4.1.30 standalone Web server.
- When deploying the Hyperion Performance Suite servlets to WebSphere on a UNIX system, the user who installs Hyperion Performance Suite should be the user who owns and runs the WebSphere installation. If a different user attempts the Hyperion Performance Suite installation, the installer cannot write to the directory where WebSphere is installed.
- Automatic deployment of the servlets on WebLogic 8.x may take a while because of its complexity. Hyperion recommends that you view the Web server log file to monitor the progress of the deployment and confirm its completion.
- For information on increasing Web server performance, see "Increasing Performance with -Dbrio.content.path" on page 108.

The installer checks for a Web server and selects a default. Select from the supported Web servers or other (requires manual configuration). The Web servers listed vary depending on your operating system. For a list of supported Web servers and servlet engines, view the Support Matrix on the Hyperion Download Center. Please note:

- If you are using **BEA WebLogic 6.x**, Hyperion Solutions recommends that you deploy the Hyperion Performance Suite servlets with precompiled JSP pages. This step must be completed *before* you continue the installer. For more information, see "Precompiling WebLogic JSPs" on page 167.
- If you select **Other Web server** as your Web server type, the installer does *not* prompt you to install and configure a servlet engine. You must manually configure these components. See Appendix A, "Manually Setting Up the Hyperion Performance Suite Servlets," for information on setting up your Web server.

31 **If you selected a Web server other than JRun 3.1 built-in or other, specify whether to install and configure the servlet engine.**

The options listed depend on the Web server that you selected.

If you selected a Web server other than JRun 3.1, the installer prompts you to install and configure a servlet engine. It then detects the servlet engine type and confirms its location.

The servlet installation requires write access to the Web server configuration files. For more information, see Chapter 7, "Postinstallation Tasks."

This sample installation uses the Apache Tomcat 4.1.30 servlet engine with the Apache 2.x Web server.

32 **If JRun 3.1 built-in is being installed as your servlet engine, enter a password for the JRun Management Console administration account.**

33 **UNIX ONLY: If the BQ_FONT_PATH environment variable is not set, specify the directory that contains the `fonts.dir` file.**

For more information, see "Fonts on UNIX" on page 52.

34 **Optional: Customize the application and Web component names if desired, then click Next.**

By customizing the names of the applications and the Hyperion Performance Suite servlets, you customize the URLs used to access your system's data. Each servlet uses a URL in this format:

`http://webserver:port/applicationname/componentname`

Note You may need to scroll to see all of the customizable items.

35 **For Windows systems, specify the Windows Start menu program folder, then click Next.**

36 **Verify the services and components to be installed.**

To change any of these choices or other parameters, click Back.

37 **Click Next to continue with the installation.**

The installer automatically copies files into target directories, populates the repository database, and configures the system for initial use. During this portion of the installation, a progress bar is displayed. This step takes generally 30 minutes or longer to complete, depending on your computer.

For a list of all Hyperion Performance Suite services and components and the products with which they are installed, see Table 1 on page 21.

After the installer finishes this portion of the installation, you must define your authentication system.

38 **Specify the Authentication Service information, then click Next.**

39 **If you select an external authentication system, supply driver information, then click Next.**

If you are using a native authentication system, proceed to Step 42.

Note The driver file must be in <*TARGET*>/lib. If you are using a custom driver, you must copy the driver file to the <*TARGET*>/lib directory.

Hyperion Solutions supplies these authentication system drivers:

- Security platform

  Use the security platform driver to provide single sign-on access when multiple Hyperion applications are deployed.
- LDAP
- Microsoft Active Directory
- Microsoft Domain
- NIS
- Database authentication

If you select to install a custom driver, you must specify the location of the external driver file and external driver class.

40 **Specify the external authentication system parameters, then click Next.**

- **Import external users**—Select whether the installer automatically imports external users from your authentication system when the authentication system is first installed.
- **Import external groups**—Select whether the installer imports external groups from your authentication system when the authentication system is first installed.

41 **If you selected the security platform driver for external authentication, read the additional information provided on screen and provide the URL for the configuration file.**

When using the security platform driver, you must modify the XML-based security configuration file (`css_config.xml`) associated with your product to specify correct information pertaining to your corporate authentication directory. For more information on using the security platform driver for external authentication, see the *Hyperion Foundation Administrator's Guide*.

42 **Select options for single sign-on, then click Next.**

Single sign-on (also referred to as pass-through) enables users to access multiple Hyperion applications after logging on just once, without additional prompting for authentication. Select one or both of the following options:

- Enable using end user's login credentials—Pass-through is enabled using the user's login credentials
- Allow end user to specify credentials for single sign-on—Pass-through is enabled using the credentials the user specifies in Browse Preferences.

  You can enable both pass-through options to allow users to choose in Browse Preference if they want to use their login credentials or use credentials specified in Browse Preference.

  For more information, see the *Hyperion Foundation Administrator's Guide* and the *Hyperion Foundation User's Guide*.

The installer then configures the authentication system, detects the servlet engine type, and confirms its location. If JRun 3.1 is selected as the servlet engine, the installer silently installs the JRun3.1 Web server and servlet engine. Generally, JRun 3.1 installs silently, but if your system configuration is nonstandard, the installer may prompt you for JRun installation parameters. See Appendix A, "Manually Setting Up the Hyperion Performance Suite Servlets," for more information.

The installer copies servlet-specific files, creates virtual directory mappings in the Web server configuration files, and confirms the Web server host and port. (For details on these files, see "Servlet Configuration Details" on page 126.) The installer then configures the servlet engine and precompiles and deploys the Web components.

Note If Sun Java System Web Server is being automatically deployed, the installer stops the Web server before starting the deployment.

This step may take a while, depending on your system configuration.

43 **If prompted, select whether to restart the Web server, then click Next.**

Caution! The Web server must be restarted before you can use the Hyperion Performance Suite servlets. You should have sufficient privileges for restarting the Web Server.

Note If you elected to install the JRun 3.1 servlet engine with IIS 5 or Apache 1.3.x, then the JRun 3.1 built-in Web server is deployed as a diagnostic tool. For a production system, you should probably turn off the service. Customer Support uses the JRun 3.1 built-in Web server to troubleshoot any problems on your primary Web server.

44 **Click Next to complete the installation.**

Any errors that occurred during the installation process are listed on the dialog box that is displayed at the end of the installation. The installer automatically selects the View Log option for you to review `Install.log`. If you encounter errors, view this log for information on diagnosing the failure.

For more information about installed files and logs, see Chapter 7, "Postinstallation Tasks."

45 **On Windows systems, if you installed services, specify whether to restart your system, then click Finish to complete the installation.**

If you installed services, you must restart your system to successfully complete the installation.

Note On UNIX, log off the account if you are prompted to do so.

Often this happens when there has been a change to an environment variable (made in the `.login` file).

46 **Configure Job Service for Microsoft SQL Server.**

Note Complete this step only if you intend to run SQR jobs against a Microsoft SQL Server database, or use printers and network resources for BQY jobs.

To run any SQR engine, or to use printers and network resources for BQY jobs, configure the corresponding Job Service to run as a *user* instead of as *System*.

a. In the Control Panel, open **Administrative Tools**.

b. Open **Services**.

c. Select the **BrioPlatform_*hostname_gsmhost_gsmport***, then select **Action→Properties**.

d. Select the **Log On** tab.

e. Select **This account** and specify a user other than *Local System*. Use the user account that you previously created for Hyperion Performance Suite (for example, *hypsuite*)

6 Adding, Reinstalling, or Replicating Server Components or Services

You can use the Hyperion Performance Suite installer to add components to an existing installation, reinstall or repair an existing installation, or replicate services.

Caution! These installation options rely on the InstallShield-generated `vpd.properties` file (in `C:\WINNT` or your UNIX *user home*) as well as the *`INSTALL_HOME`*`\log\InstallHistory.xml` file. Do not change or delete these files; doing so may cause undesirable results.

In This Chapter

Adding Components or Services 96

Reinstalling Hyperion Performance Suite 97

Replicating Services 99

Adding Components or Services

Use the Hyperion Performance Suite installer to add new components or services to an existing *INSTALL_HOME*. The components or services that you add must be the same release as the existing installation; for example, you cannot add a Hyperion Performance Suite release 8.3 service to a Hyperion Performance Suite release 8.1.x installation. You must first upgrade your Performance Suite 8.1.x installation to Hyperion Performance Suite release 8.3 before adding the new service.

Caution! Do not use the Add option to replicate a second copy of a service in an existing *INSTALL_HOME*. When adding components, the installer displays a filtered list of services and applications to prevent duplication of services in the same *INSTALL_HOME*. Only one service of a given type is supported for each *INSTALL_HOME*. The purpose of replicating services is to improve performance, and no performance gains would be realized by supporting multiple services of the same type within a single *INSTALL_HOME*.

To add additional components or services:

1. **Run the Hyperion Performance Suite installer from the Servers CD.**

   You must enter license keys for any additional products that you want to install.

Note For information about trial licenses, see "About Trial Licenses" on page 53.

2. **Select an *INSTALL_HOME* for the components that you want to add.**
3. **Select the Add new components option.**

Note If you initially installed Hyperion Performance Suite using the Express option, you must enter one or more new license keys that unlock additional items in order to add new components to your installation. If you do not enter additional license keys, the installer assumes that you want to reinstall Hyperion Performance Suite and does not give you the option to add new components.

4 **Click Next to continue.**

5 **If prompted, select either Express or Custom.**

In an Express installation, the Add option installs all the components enabled by your license keys that were not previously installed. If you want to select which components to install, select the Custom option.

The installer performs an Express or Custom installation, prompting for information as necessary.

Reinstalling Hyperion Performance Suite

You can reinstall Hyperion Performance Suite without first removing it. The Hyperion Performance Suite installer enables you to reinstall your Hyperion Performance Suite system in the same way on the host where the installation previously was run. You may need to reinstall your Hyperion Performance Suite system for any of these reasons:

- The computer where the *INSTALL_HOME* resides has a virus.
- One or more pertinent files were accidently deleted.
- You want to specify a different database for your Hyperion Performance Suite repository.
- Your initial installation attempt failed.

Note To perform a clean installation of Hyperion Performance Suite, uninstall Hyperion Performance Suite and then remove the `InstallHistory.xml` file. On Windows, this file is usually in `C:\Documents and Settings\`*`username`*; on UNIX, this file is usually in your user home directory (often `/home/username`).

To reinstall Hyperion Performance Suite:

1 **If JRun is running, shut it down before reinstalling any Hyperion Performance Suite components.**

2 **Run the Hyperion Performance Suite installer from the Servers CD.**

You do not need to re-enter your license keys.

3 **Select an *INSTALL_HOME* for the components that you want to reinstall.**

Do not select New INSTALL_HOME if you want to reinstall Hyperion Performance Suite.

4 **If prompted, select the Reinstall existing components option.**

Note If you initially installed Hyperion Performance Suite using the Express option and you did not enter any additional license keys, the installer assumes that you want to reinstall Hyperion Performance Suite and does not display this dialog box.

5 **Click Next to continue.**

The installer performs an Express or Custom installation (depending on the settings in your `InstallHistory.xml` file), prompting for information as necessary.

In a reinstallation, the old repository is left intact unless you select to drop it, but all the executable and configuration files are regenerated. Dropping the repository tables is an option only if you are reinstalling.

The installer prompts you again for database type and client library home in case you want to reinstall Hyperion Performance Suite using a different database than the one initially selected. However, the installer does not drop and recreate the Hyperion Performance Suite repository tables unless you select the option to do so, or unless they were not created in the initial installation process.

Replicating Services

More than one service of the same type can exist in a distributed Hyperion Performance Suite system. You can replicate services to handle increases in users and information, or so that you can configure them to perform different functions.

Caution! Use the Hyperion Performance Suite installer's Custom option to replicate services. Do not use the Add option to replicate a second copy of a service in an existing *INSTALL_HOME*. When adding components, the installer displays a filtered list of services and components to prevent duplication of services in the same *INSTALL_HOME*. Only one service of a given type is supported per *INSTALL_HOME*. The purpose of replicating services is to improve performance, and no performance gains would be realized by supporting multiple services of the same type within a single *INSTALL_HOME*.

You can replicate these services:

- Service Broker
- Session Manager

  For optimal load balancing, Hyperion recommends that every services host include a Session Manager.
- Authentication Service
- Authorization Service
- Job Service
- Usage Service
- Intelligence Service
- Data Access Service

In addition, the system automatically replicates the Local Service Manager (LSM) for each new server process.

Replicated components enable the installation of multiple instances of the same component on separate computers. Hyperion Performance Suite automatically routes work across the multiple replicated components.

Hyperion Performance Suite supports the following replicated configurations:

- No replication; single implementation of all components
- Servlets (as a unit) replicated
- Authentication Service replicated
- Authorization Service replicated
- Data Access Service replicated
- Intelligence Service replicated
- Job Service replicated
- Service Broker replicated
- Session Manage replicated

The following limitations apply to distribution and replication:

- Publisher, Event, and Repository Services cannot be replicated.
- LSM is not directly installed; it comes with other services. You can have one LSM per process space.
- Publisher, Authorization, and Repository Services are required in one process space.
- Intelligence Service requires Job Service.

7 Postinstallation Tasks

This chapter describes the procedures that you must perform after Hyperion Performance Suite is installed. It explains how to test your Hyperion Performance Suite installation. It also provides details on the directory structure implemented by the installer and the software components that it installs.

In This Chapter

Testing the Hyperion Performance Suite Installation 102

Disabling Directory Browsing 104

Security and Passwords 107

Increasing Performance with -Dbrio.content.path 108

Using Console Mode with Hyperion Performance Suite Components 109

Installation Directory Structure 110

Registries 119

Database Log 123

Core Installable Unit Software Components 123

Servlet Configuration Details 126

Testing the Hyperion Performance Suite Installation

To verify that Hyperion Performance Suite installed correctly:

1 **Verify the installed files.**

a. Verify the file system.

A hierarchy of subdirectories should exist in the installation directory. See "Installation Directory Structure" on page 110 for the Hyperion Performance Suite directory structure. Verify this directory structure.

b. Verify the contents of all `hosts` files.

If you are running Hyperion Performance Suite services on multiple hosts, ensure that the `hosts` file on each host computer contains entries for all Hyperion Performance Suite hosts. After you verified that all hosts are entered in the `hosts` file, you can copy that `hosts` file to all other Hyperion Performance Suite hosts. You can usually find the `hosts` file at `C:\winnt\system32\drivers\etc\hosts` on Windows or `/etc/hosts` on UNIX.

2 **Use the user account that you previously created for Hyperion Performance Suite (for example, *hypsuite*) to start the BrioPlatform Service (also referred to as the Hyperion Performance Suite Super Service), then verify that it has started.**

- **startCommonServices method**—The preferred startup method for UNIX, and an alternative method for Windows. To start all the Foundation services, run the startCommonServices script. This script, installed in *`INSTALL_HOME`*`/BrioPlatform/bin`, is named `startCommonServices.sh` (UNIX) or `startCommonServices.bat` (Windows).

  The startCommonServices method starts all of the Foundation services in an *`INSTALL_HOME`*, except any you have inactivated. Inactivating services is discussed in the *Hyperion Foundation Administrator's Guide*.

- **BrioPlatform service method**—The preferred startup method for Microsoft Windows. To start all the Foundation services, run the BrioPlatform Service from Services (in Administrative Tools, accessed from the Control Panel). The default setting is Manual startup mode. You can reset the Hyperion Performance Suite Super Service to Automatic startup mode if you want.

If the Server is not already running, select **BrioPlatform_*hostname_gsmhost_gsmport***, then select **Action→Start**. If other services are listed, you can start each one individually by selecting it and then selecting **Action→Start**.

Note The Hyperion Performance Suite Super Service includes all Hyperion Performance Suite services installed in an *INSTALL_HOME* except for Data Access Service and Intelligence Service. The Hyperion Performance Suite Super Service enables multiple services to run as a single process. If you need to start or stop individual services, use the start or stop scripts in the `/BrioPlatform/bin` directory. For more information on starting and stopping services, see the *Hyperion Foundation Administrator's Guide*.

If the Hyperion Performance Suite services do not start, see "Troubleshooting Service Startup" on page 241.

3 **If you installed the Hyperion Performance Suite servlets, verify that the servlet engine is properly configured:**

a. Stop and restart the servlet engine and the Web server.

b. Use this URL to access the Browse module of the Hyperion Performance Suite servlets:

```
http://host/Hyperion/browse/login
```

For more information on turning on the servlets and establishing a connection, see the *Hyperion Foundation Administrator's Guide*.

Disabling Directory Browsing

The current implementation of the installation does not restrict access to the internal directory structure of the server side of Hyperion Performance Suite. This means that anyone can browse the Hyperion Performance Suite server directory. The following topics explain how to disable directory browsing for these Web servers:

- "Apache Tomcat 4.1.30" on page 104
- "BEA WebLogic 7.x/8.x" on page 104
- "IBM WebSphere 5.0.1 and 5.0.2" on page 105
- "JRun 3.1" on page 105
- "JRun 4" on page 106
- "Sun Java System Web Server (aka iPlanet 6.0.6)" on page 106

Apache Tomcat 4.1.30

To disable directory browsing in Apache Tomcat 4.1.30:

1. **In the** *TOMCAT_HOME*`\conf\web.xml` **file, locate the listings initialization parameter of the** `org.apache.catalina.servlets.DefaultServlet`.
2. **Change it to false; for example:**

```
<init-param>
   <param-name>listings</param-name>
   <param-value>false</param-value>
</init-param>
```

BEA WebLogic 7.x/8.x

WebLogic 8.1 disables directory browsing by default, so there is no need for additional steps.

Although Web-application level directory browsing is controlled by the `<container-descriptor/index-directory-enabled>` property of `weblogic.xml`, the `<index-directory-enabled>` element controls whether an HTML directory listing is generated automatically if no suitable index file is found; for example:

```
<container-descriptor>
<index-directory-enabled>false</index-directory-enabled>
</container-descriptor>
```

Note `Weblogic.xml` for the servlets explicitly sets `<index-directory-enabled>` to false.

IBM WebSphere 5.0.1 and 5.0.2

Modify `<App.ear>\<App.war>\WEB-INF\ibm-web-ext.xmi` so that it contains `directoryBrowsingEnabled="false"`; for example:

```
<webappext:WebAppExtension xmi:version="2.0"
xmlns:xmi="http://www.omg.org/XMI"
xmlns:webappext="webappext.xmi"
xmlns:webapplication="webapplication.xmi"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmi:id="WebApp_1_Ext" reloadInterval="3"
reloadingEnabled="true" fileServingEnabled="true"
directoryBrowsingEnabled="false"
serveServletsByClassnameEnabled="false">
  <defaultErrorPage xsi:nil="true"/>
  <additionalClassPath xsi:nil="true"/>
  <webApp href="WEB-INF/web.xml#WebApp_1"/>
</webappext:WebAppExtension>
```

Note This setting is Web-application specific, so `ibm-web-ext.xmi` for the servlets already contains `directoryBrowsingEnabled="false"`.

JRun 3.1

To disable directory browsing in JRun 3.1:

1. In the `<JRUN_HOME>\lib\global.properties` file, look for **`file.browsedirs=true`**.
2. Change this to **`file.browsedirs=false`**.

Note The Hyperion Performance Suite 8.3 installer is shipped with installable media for Tomcat 4.1.30 and JRun 3.1. If you select the option to install one of these application servers, then the installer disables directory browsing during installation.

JRun 4

Modify the `<JRun4_Home>\servers\<server_name>\SERVER-INF\default-web.xml` file so that the `browseDirs` parameter is set to `false` for `FileServlet`. For JRun 4, `browseDirs` is set to false by default.

Sun Java System Web Server (aka iPlanet 6.0.6)

To disable directory browsing in Sun Java System Web Server:

1. **Add the following lines to the** *IPLANET_HOME*`/httpacl/generated.https-`*SERVER_NAME*`.acl` **file:**

```
acl "uri=/Hyperion";
allow (read, execute, info) user = "anyone";
deny (list, write, delete) user = "anyone";
```

2. **Restart Sun Java System Web Server.**

Security and Passwords

Table 8 lists the default passwords that you should change after you have successfully installed Hyperion Performance Suite.

Table 8 Default Passwords to Change

Password for User	Default Password
globaladministrator	globaladministrator
administrator	administrator
Config.file	administrator
ServletUser	administrator

For instructions on changing these passwords, see the *Hyperion Foundation Administrator's Guide*.

Caution! Users can browse all the directories below the Hyperion Performance Suite deployment directory, regardless of the role assigned to them. To prevent users from browsing these directories, you must block access through your Web server. See "Disabling Directory Browsing" on page 104 for more information.

Increasing Performance with -Dbrio.content.path

To increase performance of your Hyperion Performance Suite installation, you can serve static files (such as GIFs) from your Web server instead of from the application server. Hyperion recommends that you do so if your Web environment is Microsoft IIS with JRun 3.x.

Serving static files from your Web server also can be helpful on large enterprise systems where there is a separate Web server before an application server. In this case, moving static content to the Web server side is good practice. Very often such systems have several secure zones; serving static files from the Web server reduces traffic between the Web server and the application server, as well as the load on the firewall, network, and the application server.

To serve static files from your Web server instead of from the application server, you use the `-Dbrio.content.path` command line option with the servlets.

For example, with the IBM HTTP Server and WebSphere:

1. Create a virtual directory or alias on your Web server resembling this one in `httpd.conf` for the IBM HTTP Server:

   ```
   Alias /brio_content "/usr/WebSphere/AppServer/installedApps/
   Brio.ear/Brio.war"
   ```

2. On the application server, in this case WebSphere, add a Java command line property:

   ```
   -Dbrio.content.path=brio_content
   ```

   where *`brio_content`* matches the alias name on the Web server.

The exact instructions depend on the Web and application servers used; see your application server documentation for instructions on setting a Java System Property in the application server.

Using Console Mode with Hyperion Performance Suite Components

Components of Hyperion Performance Suite can be run on systems with no X windows capabilities; the Local Service Configurator (LSC) is an exception.

The LSC is a GUI-mode Java application that configures services residing on a particular system by modifying the `server.xml` file on that system; therefore, the LSC cannot be run remotely. If you run any LSC services on a system with no GUI capability, you cannot use the Local Service Configurator to modify the `server.xml` file. LSC includes the following services:

- Global Service Manager
- Local Service Manager
- Authentication Service
- Session Manager
- Publisher Service
- Authorization Service
- (Super Service)
- Intelligence Service
- Logging Service
- Data Access Service
- Usage Service

Installation Directory Structure

Figure 4 shows the directory structure created during the installation process.

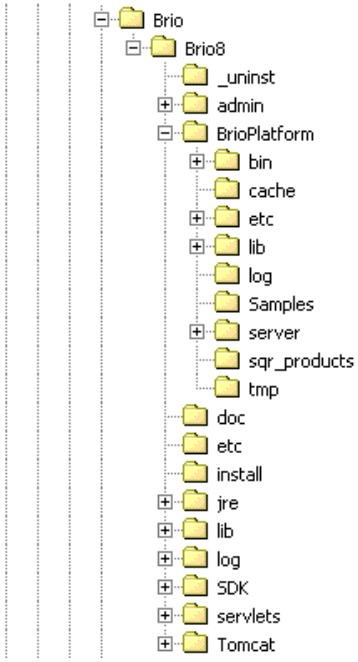

Figure 4 Hyperion Performance Suite Directory Structure

Hyperion Performance Suite uses these directories and subdirectories:

- `/BrioPlatform/bin`—Binary executables and DLLs
- `/BrioPlatform/cache`—Caching files
- `/BrioPlatform/etc`—Configuration items such as `server.xml` and `config.dat`

  Sample `config.dat` and `server.xml` files are included only for example purposes.

- /BrioPlatform/lib—Service-only JAR files
- /BrioPlatform/log—Service logs
- /BrioPlatform/tmp—Temporary files
- /doc—Documents
- /install—Installation only items
- /jre—Java runtime environment
- /lib—Common JAR files

Sample config.dat File

```
[Setup]
Key=29981372
Password=4E44D8DC983E6C89BAF1E7F4ADE32C3E8EFC37C3D1C7CBCE2D0726C0CE1FB8E7F71CA249
[NameServer]
Name=NS1_PAndrews
Login=hypsuite
Password=4ADEF1A3F6AFA452542FC3B936FBECE1FB8E7F71CA249
JDBC_URL=jdbc:oracle:thin:@10.210.1.10:1521:sea817
JDBC_DRIVER=oracle.jdbc.driver.OracleDriver
Host=PAndrews
Port=1498
SAPassword=5CA04F00042B9098676E8A4D39E01506D819074BC2F2318AE3F530CE1FB8E7F71CA249
[AGENT=RM1_PAndrews]
Password=83AF169274608F2ACF9DA3B34239E01506D819074BC2F2318AE3F530CE1FB8E7F71CA249
[AGENT=SB1_PAndrews]
Password=E58C374F96F1B36567259C6B4239E01506D819074BC2F2318AE3F530CE1FB8E7F71CA249
[AGENT=JF1_PAndrews]
Password=36DEB1E2F9CFCDE7CCD383284239E01506D819074BC2F2318AE3F530CE1FB8E7F71CA249
[AGENT=ES1_PAndrews]
Password=643687AC0EB70406B0D67A024239E01506D819074BC2F2318AE3F530CE1FB8E7F71CA249
[AGENT=SS1_PAndrews]
Password=632E65482026CEC28729B2194239E01506D819074BC2F2318AE3F530CE1FB8E7F71CA249
```

Sample server.xml File


```
<?xml version='1.0' encoding='utf-8'?>
<BRIO>
  <SETUP>
    <key>50629081</key>
    <filecontrol>0987473927474cb0de884801688fb61206bcc71130db9d7e48adf8a95ec51bb1
    </filecontrol>
    <tmpfilelocation>C:\Program Files\Brio\Brio8\BrioPlatform\tmp
    </tmpfilelocation>
    <FILEID>Host pandrews2</FILEID>
    <gsmcontextid>cc3330f85cc727cfcfcfb62120b757d372033e6dad4b35963788fde4071c387
</gsmcontextid>
    <properties>
      <property name="ISO_BQ_COUNTRY_CODE">us</property>
        <propertylist name="LOCALE_LIST" defid="000000f324d3875d-0000-114a-
        0ad72214">
          <propertystruct name="LOCALE" defid="000000f324d56aa3-0000-114a-
          0ad72214">
            <property name="ISO_LANG_CODE" defid="000000f324d65997-0000-114a-
            0ad72214">en</property>
            <property name="ISO_COUNTRY_CODE" defid="000000f324d6f616-0000-114a-
            0ad72214">US</property>
            <property name="DEFAULT_LOCALE" defid="000000f324d778d3-0000-114a-
            0ad72214">true</property>
          </propertystruct>
        </propertylist>
    </properties>
    <logging>
      <redirectoutput>true</redirectoutput>
      <xmllogging>true</xmllogging>
      <level>warning</level>
    </logging>
    <GSM>
      <runtype>start</runtype>
      <fedName>
        <instanceid>000000f4934d641b-0000-0460-0ad72dd8</instanceid>
        <typeid>0ad72298-0000-0ccc-000000e750e523ab</typeid>
        <versionid>00000000-0000-0000-0000000000000000</versionid>
        <pathname>GlobalServiceManager</pathname>
      </fedName>
      <host>pandrews2</host>
      <port>1800</port>
      <ServiceTestInterval>60</ServiceTestInterval>
    </GSM>
    <bqyrowlevelconnect>
      <db_connectivity>API_CODE_NONE</db_connectivity>
      <db_type>SVR_CODE_NONE</db_type>
      <db_host>NONE</db_host>
      <db_user>NONE</db_user>
    </bqyrowlevelconnect>
  </SETUP>
  <SERVICES app="bq">
    <LSM>
      <fedName>
```

```
    <instanceid>000000f4935f67fb-0000-0460-0ad72dd8</instanceid>
    <typeid>0ad72298-0000-0ccc-000000e750e616d5</typeid>
    <versionid>00000000-0000-0000-0000000000000000</versionid>
    <pathname>LSM_BI1_pandrews2</pathname>
    <servicedesc>Generated service name for BI LSM</servicedesc>
  </fedName>
  <dbconfig>
    <username>hypsuite</username>
     <password>ab8386f57c0bcb08837a4e875991b95fa448ba8ecbb4f424ede5d433c56769
     </password>
    <dburl>jdbc:oracle:thin:@10.213.1.39:1521:sea817</dburl>
    <dbdriver>oracle.jdbc.driver.OracleDriver</dbdriver>
  </dbconfig>
  <login>
    <username>Server on host pandrews2</username>
    <password>10e0ed8be87442367f466f8abb05f8f9</password>
  </login>
  <properties>
    <property name="directory">C:\Program Files\Brio\Brio8\BrioPlatform
    </property>
    <property name="GsmSyncTime" defid="0">30</property>
    <property name="ServiceTestInterval" defid="0">60</property>
    <property name="ServiceRegistrationInterval" defid="0">15</property>
  </properties>
  <firewallport>1802</firewallport>
</LSM>
<service type="BrioQuery">
  <runtype>start</runtype>
  <fedName>
    <instanceid>000000f4935f676f-0000-0460-0ad72dd8</instanceid>
    <typeid>0ad70312-0000-0934-000000ea4ac65069</typeid>
    <versionid>00000000-0000-0000-0000000000000000</versionid>
    <pathname>BI1_pandrews2</pathname>
    <servicedesc>BI1_pandrews2</servicedesc>
  </fedName>
  <classInfo>
    <infotype>com.brio.one.services.brioquery.info.BrioQueryServiceInfo
    </infotype>
  </classInfo>
  <properties>
    <propertylist name="BQ_EVENT_MONITOR_PROPERTY_LIST" defid="0ad70321-0002-
    08aa-000000e738090110">
      <property name="EVENT_MONITORING" defid="0ad70321-0001-08aa-
      000000e738090110">OFF</property>
      <property name="MAXIMUM_DOCUMENTS_THRESHOLD" defid="0ad70321-0001-08aa-
      000000e738090110">10000</property>
      <property name="MAXIMUM_JOBS_THRESHOLD" defid="0ad70321-0001-08aa-
      000000e738090110">10000</property>
      <property name="MAXIMUM_UP_TIME_THRESHOLD" defid="0ad70321-0001-08aa-
      000000e738090110">180</property>
      <property name="SPECIFIC_SHUTDOWN_TIME" defid="0ad70321-0001-08aa-
      000000e738090110">1290</property>
    </propertylist>
    <property name="CACHE_LOCATION" defid="0ad70321-0001-08aa-000000e
    738090110">C:\Program Files\Brio\Brio8\BrioPlatform\cache</property>
```

```
      <property name="MAX_CONCURRENT_REQUESTS" defid="0ad70321-0001-08aa-
      000000e738090110">5000</property>
      <property name="MIN_DISK_SPACE" defid="0ad70321-0001-08aa-
      000000e738090110">10</property>
      <property name="DOCUMENT_UNLOAD_TIMEOUT" defid="0ad70321-0001-08aa-
      000000e738090110">900</property>
      <property name="DOCUMENT_UNLOAD_THRESHOLD" defid="0ad70321-0001-08aa-
      000000e738090110">15</property>
      <property name="POLLING_INTERVAL" defid="0ad70321-0001-08aa-
      000000e738090110">180</property>
      <property name="LOG_FILE" defid="0ad70321-0001-08aa-
      000000e738090110">C:\Program Files\Brio\Brio8\BrioPlatform\log\
      BI1_pandrews2.log</property>
      <property name="LOG_LEVEL" defid="0ad70321-0001-08aa-
      000000e738090110">Informational</property>
      <property name="LOG_FILES_TO_KEEP" defid="0ad70321-0001-08aa-
      000000e738090110">2</property>
      <property name="LOG_FILES_RECYCLE_TIME" defid="0ad70321-0001-08aa-
      000000e738090110">24</property>
    </properties>
  </service>
</SERVICES>
<SERVICES app="cs">
  <service type="Authentication">
    <runtype>start</runtype>
    <fedName>
      <instanceId>000000f4934d5c2b-0000-0460-0ad72dd8</instanceId>
      <typeid>0ad72298-0000-0ccc-000000e750e6b845</typeid>
      <versionid>00000000-0000-0000-0000000000000000</versionid>
      <pathname>AN1_pandrews2</pathname>
      <servicedesc>AN1_pandrews2</servicedesc>
    </fedName>
    <classInfo>
      <infotype>com.brio.one.services.authentication.info.AuthenticationService
      Info</infotype>
      <serviceclass>com.brio.one.services.authentication.idl.impl.
      AuthenticationServiceInstance</serviceclass>
    </classInfo>
    <properties>
      <property name="com.brio.one.services.authentication.AuthenticationConst.
      AuthenticationSystem" defid="0ad72237-0087-0d59-000000e4d26d0b6c">
      000000f493919b2d-0000-0460-0ad72dd8</property>
    </properties>
  </service>
  <service type="SessionManager">
    <runtype>start</runtype>
    <fedName>
      <instanceid>000000f4934d5bce-0000-0460-0ad72dd8</instanceid>
      <typeid>0ad72298-0000-0ccc-000000e750e6ffdd</typeid>
      <versionid>00000000-0000-0000-0000000000000000</versionid>
      <pathname>SM1_pandrews2</pathname>
      <servicedesc>SM1_pandrews2</servicedesc>
    </fedName>
    <properties>
      <property name="SessionIdleTime" defid="0">1800</property>
```

```
    </properties>
    <classInfo>
      <infotype>com.brio.one.services.sessionmanager.info.SessionManagerInfo
      </infotype>
      <serviceclass>com.brio.one.services.sessionmanager.idl.impl.Session
      ManagerInstance</serviceclass>
    </classInfo>
  </service>
  <service type="Publisher">
    <runtype>start</runtype>
    <fedName>
      <instanceid>000000f4934d5b41-0000-0460-0ad72dd8</instanceid>
      <typeid>0ad72298-0000-0ccc-000000e750ea6705</typeid>
      <versionid>00000000-0000-0000-0000000000000000</versionid>
      <pathname>PUB1_pandrews2</pathname>
      <servicedesc>PUB1_pandrews2</servicedesc>
    </fedName>
    <classInfo>
      <infotype>com.brio.one.services.publisher.info.PublisherServiceInfo
      </infotype>
      <serviceclass>com.brio.one.services.publisher.idl.impl.PublisherService
      Instance</serviceclass>
    </classInfo>
  </service>
  <service type="Authorization">
    <runtype>start</runtype>
    <fedName>
      <instanceid>000000f4934d5ca8-0000-0460-0ad72dd8</instanceid>
      <typeid>0ad72298-0000-0ccc-000000e750e654d8</typeid>
      <versionid>00000000-0000-0000-0000000000000000</versionid>
      <pathname>AZ1_pandrews2</pathname>
      <servicedesc>AZ1_pandrews2</servicedesc>
    </fedName>
    <classInfo>
      <infotype>com.brio.one.services.authorization.info.AuthorizationService
      Info</infotype>
      <serviceclass>com.brio.one.services.authorization.idl.impl.Authorization
      ServiceInstance</serviceclass>
    </classInfo>
  </service>
  <service type="BrioPortal">
    <runtype>start</runtype>
    <fedName>
      <instanceid>000000f4934d63dc-0000-0460-0ad72dd8</instanceid>
      <typeid>0ad72298-0000-0ccc-000000e750e8c6c2</typeid>
      <versionid>00000000-0000-0000-0000000000000000</versionid>
      <pathname>BPS1_pandrews2</pathname>
    </fedName>
    <classInfo>
      <serviceclass>com.sqribe.transformer.SuperServerImpl</serviceclass>
    </classInfo>
    <properties>
      <propertylist type="local" name="SystemProperties">
        <property name="directory">C:\Program Files\Brio\Brio8\BrioPlatform
        </property>
```

```
      </propertylist>
    </properties>
  </service>
  <LSM>
    <fedName>
      <instanceid>000000f4934d1f42-0001-0460-0ad72dd8</instanceid>
      <typeid>0ad72298-0000-0ccc-000000e750e616d5</typeid>
      <versionid>00000000-0000-0000-0000000000000000</versionid>
      <pathname>CORE1_pandrews2</pathname>
      <servicedesc>CORE1_pandrews2</servicedesc>
    /fedName>
    <dbconfig>
      <username>hypsuite</username>
      <password>ab8386f57c0bcb08837a4e875991b95fa448ba8ed433c5644769</password>
      <dburl>jdbc:oracle:thin:@10.213.1.39:1521:sea817</dburl>
      <dbdriver>oracle.jdbc.driver.OracleDriver</dbdriver>
    </dbconfig>
    <login>
      <username>Server on host pandrews2</username>
      <password>10e0ed8be87442367f466f8abb05f8f9</password>
    </login>
    <properties>
      <property name="directory">C:\Program Files\Brio\Brio8\BrioPlatform
      </property>
      <property name="GsmSyncTime" defid="0">30</property>
      <property name="ServiceTestInterval" defid="0">60</property>
      <property name="ServiceRegistrationInterval" defid="0">15</property>
    </properties>
    <firewallport>1801</firewallport>
  </LSM>
</SERVICES>
<SERVICES app="DAS">
  <LSM>
    <fedName>
      <instanceid>000000f4936193a5-0000-0460-0ad72dd8</instanceid>
      <typeid>0ad72298-0000-0ccc-000000e750e616d5</typeid>
      <versionid>00000000-0000-0000-0000000000000000</versionid>
      <pathname>LSM_DAS1_pandrews2</pathname>
    <servicedesc>Generated service name for DAS LSM</servicedesc>
    </fedName>
    <dbconfig>
      <username>hypsuite</username>
      <password>ab8386f57c0bcb08837a4e875991cbb4f424ede5d433c5644769</password>
      <dburl>jdbc:oracle:thin:@10.213.1.39:1521:sea817</dburl>
      <dbdriver>oracle.jdbc.driver.OracleDriver</dbdriver>
    </dbconfig>
    <login>
      <username>Server on host pandrews2</username>
      <password>10e0ed8be87442367f466f8abb05f8f9</password>
    </login>
    <properties>
      <property name="directory">C:\Program Files\Brio\Brio8\BrioPlatform
      </property>
      <property name="GsmSyncTime" defid="0">30</property>
      <property name="ServiceTestInterval" defid="0">60</property>
```

```
    <property name="ServiceRegistrationInterval" defid="0">15</property>
  </properties>
  <firewallport>1803</firewallport>
</LSM>
<service type="DataAccess">
<runtype>start</runtype>
<fedName>
    <instanceid>000000f493619347-0000-0460-0ad72dd8</instanceid>
    <typeid>0ad70312-0000-0934-000000ea4ac5e670</typeid>
    <versionid>00000000-0000-0000-0000000000000000</versionid>
    <pathname>DAS1_pandrews2</pathname>
  <servicedesc>DAS1_pandrews2</servicedesc>
</fedName>
  <classInfo>
    <infotype>com.brio.one.services.das.info.DataAccessServiceInfo</infotype>
    <proxy>com.brio.one.services.das.proxy.DataAccessServiceProxy</proxy>
  </classInfo>
  <properties>
    <propertylist name="DAS_EVENT_MONITOR_PROPERTY_LIST" defid="0ad70321-
    0002-08aa-000000e738090110">
      <property name="EVENT_MONITORING" defid="0ad70321-0001-08aa-
      000000e738090110">OFF</property>
      <property name="MAXIMUM_RELATIONAL_PROCESS_THRESHOLD" defid="0ad70321-
      0001-08aa-000000e738090110">20000</property>
      <property name="MAXIMUM_MDD_PROCESS_THRESHOLD" defid="0ad70321-0001-
      08aa-000000e738090110">20000</property>
      <property name="MAXIMUM_RELATIONAL_OTHER_THRESHOLD" defid="0ad70321-
      0001-08aa-000000e738090110">50000</property>
      <property name="MAXIMUM_MDD_OTHER_THRESHOLD" defid="0ad70321-0001-08aa-
      000000e738090110">50000</property>
      <property name="MAXIMUM_UP_TIME_THRESHOLD" defid="0ad70321-0001-08aa-
      000000e738090110">180</property>
      <property name="SPECIFIC_SHUTDOWN_TIME" defid="0ad70321-0001-08aa-
      000000e738090110">1290</property>
    </propertylist>
    <property name="LOG_FILE" defid="0ad70321-0001-08aa-
    000000e738090110">C:\Program Files\Brio\Brio8\BrioPlatform\log\
    DAS1_pandrews2.log</property>
    <property name="LOG_LEVEL" defid="0ad70321-0001-08aa-
    000000e738090110">Informational</property>
    <property name="LOG_FILES_TO_KEEP" defid="0ad70321-0001-08aa-
    000000e738090110">2</property>
    <property name="LOG_FILES_RECYCLE_TIME" defid="0ad70321-0001-08aa-
    000000e738090110">24</property>
    <property name="DAS_RELATIONAL_DB_PARTIAL_RESULT_CELL_COUNT"
    defid="0ad70321-0001-08aa-000000e738090110">50000</property>
    <property name="DAS_MULTIDIMENSIONAL_DB_PARTIAL_RESULT_ROW_COUNT"
    defid="0ad70321-0001-08aa-000000e738090110">512</property>
    <property name="DAS_RESULT_RESOURCES_REAP_INTERVAL_SECONDS"
    defid="0ad70321-0001-08aa-000000e738090110">180</property>
    <property name="DAS_RESULT_RESOURCE_MINIMUM_IDLE_TIME_SECONDS"
    defid="0ad70321-0001-08aa-000000e738090110">180</property>
    <propertylist type="local" name="DAS_DB_SERVER_PROPERTIES_LIST"
    defid="0ad72289-0000-0415-000000ea038ab234">
      <propertystruct name="DAS_DB_SERVER_PROPERTY" defid="0ad72289-0000-
```

```
        0415-000000ea038bac72">
          <property name="DAS_DB_CONNECTIVITY" defid="0ad72289-0001-0415-
          000000ea038baef3">API_CODE_ODBC</property>
          <property name="DAS_DB_TYPE" defid="0ad72289-0000-0415-
          000000ea038badd9">SVR_CODE_ODBC</property>
          <property name="DAS_DB_HOST" defid="0ad72289-0000-0415-
          000000ea038baef3">Brio 8.0 Sample</property>
          <property name="DAS_OLEDB_DATASOURCE" defid="0ad72289-0002-0415-
          000000ea038baef3">DATASOURCENAME_PARAM</property>
          <property name="DAS_MAXIMUM_CONNECTIONS_TO_DB_LIMIT" defid="0ad72289-
          0000-0415-000000ea038baffc">2000</property>
          <property name="DAS_MAXIMUM_CONNECTION_WAIT_QUEUE_SIZE_LIMIT"
          defid="0ad72289-0000-041d-000000ea0392288a">100</property>
          <property name="DAS_MINIMUM_CONNECTION_IDLE_TIME_SECONDS"
          defid="0ad72289-1000-041d-000000ea0392288a">180</property>
          <property name="DAS_CONNECTION_RESOURCE_REAP_INTERVAL_SECONDS"
          defid="0ad72289-1001-041d-000000ea0392288a">180</property>
          <property name="DAS_MAXIMUM_NUMBER_OF_CONNECTIONS_IN_CONNECTION_POOL"
          defid="0ad72289-1002-041d-000000ea0392288a">1000</property>
          <property name="DAS_MINIMUM_EMPTY_CONNECTION_POOL_IDLE_TIME_SECONDS"
          defid="0ad72289-1003-041d-000000ea0392288a">180</property>
        </propertystruct>
        <propertystruct name="DAS_DB_SERVER_PROPERTY" defid="0ad72289-0000-
        0415-000000ea038bac72">
          <property name="DAS_DB_CONNECTIVITY" defid="0ad72289-0001-0415-
          000000ea038baef3">API_CODE_NET8</property>
          <property name="DAS_DB_TYPE" defid="0ad72289-0000-0415-
          000000ea038badd9">SVR_CODE_ORA8</property>
          <property name="DAS_DB_HOST" defid="0ad72289-0000-0415-
          000000ea038baef3">sea817</property>
          <property name="DAS_OLEDB_DATASOURCE" defid="0ad72289-0002-0415-
          000000ea038baef3">DATASOURCENAME_PARAM</property>
          <property name="DAS_MAXIMUM_CONNECTIONS_TO_DB_LIMIT" defid="0ad72289-
          0000-0415-000000ea038baffc">2000</property>
          <property name="DAS_MAXIMUM_CONNECTION_WAIT_QUEUE_SIZE_LIMIT"
          defid="0ad72289-0000-041d-000000ea0392288a">100</property>
          <property name="DAS_MINIMUM_CONNECTION_IDLE_TIME_SECONDS"
          defid="0ad72289-1000-041d-000000ea0392288a">180</property>
          <property name="DAS_CONNECTION_RESOURCE_REAP_INTERVAL_SECONDS"
          defid="0ad72289-1001-041d-000000ea0392288a">180</property>
          <property name="DAS_MAXIMUM_NUMBER_OF_CONNECTIONS_IN_CONNECTION_POOL"
          defid="0ad72289-1002-041d-000000ea0392288a">1000</property>
          <property name="DAS_MINIMUM_EMPTY_CONNECTION_POOL_IDLE_TIME_SECONDS"
          defid="0ad72289-1003-041d-000000ea0392288a">180</property>
        </propertystruct>
      </propertylist>
    </properties>
  </service>
 </SERVICES>
</BRIO>
```

Registries

The following topics discuss the three registries utilized by Hyperion Performance Suite:

- Windows Registry
- VPD Registry
- Cross-Platform Registry

Windows Registry

The Windows registry contains an entry for *INSTALL_HOME*, which you can use to find the `brio.rc` file. There may be more than one *INSTALL_HOME* entry if there is more than one GSM context installed on a computer. An example of this registry entry is as follows:

```
HKEY_LOCAL_MACHINE→SOFTWARE→Brio
Software→BrioPlatform→BRIO_HOME#1
```

VPD Registry

The VPD registry (`vpd.properties`) tracks the software installed, the level of the software, and the path to the software.

Caution! Never delete the VPD registry. Upgrades, Add installations, and other software patch actions may not work if this registry is deleted or removed.

Cross-Platform Registry

The cross-platform registry, `brio.rc`, is an XML file that contains useful information about common service components. This information is not stored in the VPD registry.

The registry consists of general information organized by installable units for use by vendors and customers to integrate their own software. There is one section for each installable unit that is installed. The information stored reflects the values entered or generated during the most recent installation.

This `brio.rc` registry stores information on the following items:

- Product IDs, names, and release numbers
- License keys
- Languages installed
- JRE home
- Orbix home
- Other configuration information written by the installable unit installers

A sample `brio.rc` file is included only for example purposes.

Sample brio.rc File

```
<?xml version="1.0" encoding="utf-8"?>
<BRIO>
<AuthenticationService>
<version>8.3.0.294</version>
<path>C:\Program Files\Brio\Brio8</path>
<name>AN1_PAndrew2</name>
<description>AN1_PAndrew2</description>
</AuthenticationService>
<CoreService>
<version>8.3.0.294</version>
<path>C:\Program Files\Brio\Brio8</path>
<name>CORE1_PAndrew2</name>
<description>CORE1_PAndrew2</description>
</CoreService>
<GSMNameService>
<version>8.3.0.294</version>
<name>NS1_PAndrew2</name>
<description>NS1_PAndrew2</description>
<path>C:\Program Files\Brio\Brio8</path>
<host>PAndrew2</host>
</GSMNameService>
<AuthorizationService>
<version>8.3.0.294</version>
<name>AZ1_PAndrew2</name>
<description>AZ1_PAndrew2</description>
<path>C:\Program Files\Brio\Brio8</path>
<host>PAndrew2</host>
</AuthorizationService>
<AuthenticationService>
<version>8.3.0.294</version>
<name>AN1_PAndrew2</name>
<description>AN1_PAndrew2</description>
<path>C:\Program Files\Brio\Brio8</path>
<host>PAndrew2</host>
```

```
</AuthenticationService>
<PublisherService>
<version>8.3.0.294</version>
<name>PUB1_PAndrew2</name>
<description>PUB1_PAndrew2</description>
<path>C:\Program Files\Brio\Brio8</path>
<host>PAndrew2</host>
</PublisherService>
<SessionManager>
<version>8.3.0.294</version>
<name>SM1_PAndrew2</name>
<description>SM1_PAndrew2</description>
<path>C:\Program Files\Brio\Brio8</path>
<host>PAndrew2</host>
<sessionTimeout>1800</sessionTimeout>
</SessionManager>
<LoggingService>
<version>8.3.0.294</version>
<name>LS1_PAndrew2</name>
<description>LS1_PAndrew2</description>
<path>C:\Program Files\Brio\Brio8</path>
<host>PAndrew2</host>
</LoggingService>
<UsageService>
<version>8.3.0.294</version>
<name>UT1_PAndrew2</name>
<description>UT1_PAndrew2</description>
<path>C:\Program Files\Brio\Brio8</path>
<host>PAndrew2</host>
</UsageService>
<RepositoryService>
<version>8.3.0.294</version>
<name>RM1_PAndrew2</name>
<description>RM1_PAndrew2</description>
<path>C:\Program Files\Brio\Brio8</path>
<host>PAndrew2</host>
</RepositoryService>
<ServiceBroker>
<version>8.3.0.294</version>
<name>SB1_PAndrew2</name>
<description>SB1_PAndrew2</description>
<path>C:\Program Files\Brio\Brio8</path>
<host>PAndrew2</host>
</ServiceBroker>
<JobService>
<version>8.3.0.294</version>
<name>JF1_PAndrew2</name>
<description>JF1_PAndrew2</description>
<path>C:\Program Files\Brio\Brio8</path>
<host>PAndrew2</host>
</JobService>
<EventService>
<version>8.3.0.294</version>
<name>ES1_PAndrew2</name>
<description>ES1_PAndrew2</description>
```

```
<path>C:\Program Files\Brio\Brio8</path>
<host>PAndrew2</host>
</EventService>
<IntelligenceService>
<version>8.3.0.294</version>
<name>BI1_PAndrew2</name>
<description>BI1_PAndrew2</description>
<path>C:\Program Files\Brio\Brio8</path>
<host>PAndrew2</host>
<cacheDir>C:\Program Files\Brio\Brio8\BrioPlatform\cache</cacheDir>
<BIsectionTimeout>36000</BIsectionTimeout>
<BIdocumentTimeout>36000</BIdocumentTimeout>
</IntelligenceService>
<DataAccessService>
<version>8.3.0.294</version>
<name>DAS1_PAndrew2</name>
<description>DAS1_PAndrew2</description>
<path>C:\Program Files\Brio\Brio8</path>
<host>PAndrew2</host>
</DataAccessService>
<ClientCore>
<version>8.3.0.294</version>
<path>C:\Program Files\Brio\Brio8</path>
<host>PAndrew2</host>
<classpath/>
</ClientCore>
<SDK>
<version>8.3.0.294</version>
<classpath>.;C:\Program Files\Brio\Brio8\SDK\lib\rmapi.jar;C:\Program
Files\Brio\Brio8\lib\xmlParserAPIs.jar;C:\Program
Files\Brio\Brio8\lib\xercesImpl.jar;C:\Program
Files\Brio\Brio8\lib\foundation.jar;C:\Program
Files\Brio\Brio8\lib\comutil1_01.jar;C:\Program
Files\Brio\Brio8\lib\crawlerurl.jar;C:\Program
Files\Brio\Brio8\lib\spf60.zip;C:\Program Files\Brio\Brio8\SDK</classpath>
</SDK>
<JobUtilities>
<version>8.3.0.294</version>
<classpath>C:\Program Files\Brio\Brio8\admin\lib;C:\Program
Files\Brio\Brio8\lib</classpath>
</JobUtilities>
<RemoteServiceConfigurator>
<version>8.3.0.294</version>
<classpath>C:\Program Files\Brio\Brio8\admin\lib;C:\Program
Files\Brio\Brio8\lib</classpath>
</RemoteServiceConfigurator>
<Servlets>
<version>8.3.0.294</version>
<appname>brio</appname>
<SVwebserverType>APACHE2</SVwebserverType>
<SVwebserverLocation>C:\Inetpub\AdminScripts</SVwebserverLocation>
<SVservletEngineType>TOMCAT</SVservletEngineType>
<SVservletConfigFileLocation>C:\Program
Files\Brio\Brio8\servlets\deployment</SVservletConfigFileLocation>
<SVselectedJRunServer/>
```

```
<servlet>administration</servlet>
<servlet>jobmanager</servlet>
<servlet>viewmanager</servlet>
<servlet>browse</servlet>
<servlet>personalpages</servlet>
<servlet>dataaccess</servlet>
<servlet>ihtml</servlet>
</Servlets>
</BRIO>
```

Database Log

The installer creates and maintains two tables, `v8_install_svc_log` and `v8_install_svc_gsm`, that track installation activity for a GSM context. Because the database must be up and running for all server installations, this table enables some additional checking that cannot be done within a host. Use of this log enables Hyperion Performance Suite to detect if a nonreplicable item is being installed twice.

Caution! Do not modify these database log tables manually unless instructed to do so by Hyperion Solutions Customer Support.

Core Installable Unit Software Components

This section lists the software components installed by the Core Services installer, including:

- "Driver Components" on page 124
- "JRE Components" on page 124
- "Orbix Components" on page 124
- "Services Components" on page 124
- "SQR Components" on page 124
- "SQR SQL" on page 125
- "Start Scripts" on page 125
- "Support Components" on page 125

Driver Components

The following Driver components are installed:

```
INSTALL_HOME\lib\authdriver.zip
INSTALL_HOME\lib\css.jar
```

JRE Components

The Java Runtime Environment (JRE) components consists of JRE 1.4.2, installed in *INSTALL_HOME*`\jre\jre142`.

Orbix Components

The following Orbix components, which provide the CORBA platform, are installed:

```
INSTALL_HOME\lib\default-domain.cfg
INSTALL_HOME\lib\iona61.jar
INSTALL_HOME\lib\licenses.txt
```

Services Components

The following Services components are installed:

```
INSTALL_HOME\lib\foundation.jar
INSTALL_HOME\lib\clientservlet.jar
INSTALL_HOME\lib\logJavaUtil.jar
```

SQR Components

The following SQR component is installed:

```
INSTALL_HOME\lib\spf82.jar
```

SQR SQL

SQR SQL is used for manually adding Hyperion SQR releases. It is loaded into the `sqr_products` directory:

```
INSTALL_HOME\BrioPlatform\sqr_products
```

It includes a batch utility for loading new Hyperion SQR releases.

Start Scripts

Individual common services start scripts are loaded into the `\bin` directory:

```
INSTALL_HOME\BrioPlatform\bin\StartCommonServices.bat
INSTALL_HOME\BrioPlatform\bin\StartCommonServices.sh
```

Support Components

The following Support components are installed:

```
INSTALL_HOME\lib\castor-0.9.5-xml.jar
INSTALL_HOME\lib\commons_collections.jar
INSTALL_HOME\lib\comutil1_01.jar
INSTALL_HOME\lib\concurrent.jar
INSTALL_HOME\lib\dom.jar
INSTALL_HOME\lib\jaxp-api.jar
INSTALL_HOME\lib\jdom.jar
INSTALL_HOME\lib\jecf.jar
INSTALL_HOME\lib\log4j.jar
INSTALL_HOME\lib\logi.crypto1.1.2.jar
INSTALL_HOME\lib\sax.jar
INSTALL_HOME\lib\sfc.jar
INSTALL_HOME\lib\swing.jar
INSTALL_HOME\lib\swingall.jar
INSTALL_HOME\lib\symbeans.jar
INSTALL_HOME\lib\xercesImpl.jar
INSTALL_HOME\lib\xmlParserAPIs.jar
```

Servlet Configuration Details

If you select the option to install the Hyperion Performance Suite servlets, the installer preconfigures your servlet engine to improve performance and copies servlet-specific files to your system, including these:

- A standard servlet configuration file containing standard defaults (formerly known as `WS.conf`) and the Java executable used to modify it.
- Intelligence Client installer and JavaScript files, bundled in the WAR-compliant file structure of the Intelligence servlet.
- WAR-compliant file structures for each of the seven Hyperion Performance Suite servlets.

The installer then configures the servlet engine. The installer creates virtual directory mappings in the Web server configuration files and confirms the Web server host and port.

The virtual directory mappings are as follows:

```
WebClient.Applications.URL.Browser=/Hyperion/browse/
WebClient.Applications.URL.Admin=/Hyperion/administrator/
WebClient.Applications.URL.Personal=/Hyperion/personalpages/
WebClient.Applications.URL.Scheduler=/Hyperion/jobmanager/
WebClient.Applications.URL.iHTMLServlet=/Hyperion/ihtml/
WebClient.Applications.URL.DAServlet=/Hyperion/dataaccess/
WebClient.Applications.URL.Viewmanager=/Hyperion/viewmanager
```

The URLs for the servlets are as follows:

```
http://hostname/Hyperion/browse
http://hostname/Hyperion/administrator
http://hostname/Hyperion/personalpages
http://hostname/Hyperion/jobmanager
http://hostname/Hyperion/ihtml
http://hostname/Hyperion/dataaccess/
http://hostname/Hyperion/viewmanager
```

8 Removing a Server Installation

This chapter explains how to remove your Hyperion Performance Suite Server installation.

Note If your installation failed, you do not need to uninstall Hyperion Performance Suite. For more information, see "Reinstalling Hyperion Performance Suite" on page 97.

In This Chapter

Removing Hyperion Performance Suite Server from Windows Systems 128
Removing Hyperion Performance Suite Server Using the Command Line. 130
Installation Removal Details 130

Removing Hyperion Performance Suite Server from Windows Systems

Follow this procedure to remove your Hyperion Performance Suite Server installation. For a custom installation, repeat this procedure on each platform where software was installed in order to completely remove the Hyperion Performance Suite system.

To remove your Hyperion Performance Suite Server installation from a Windows system:

1 **Stop all related services on the computer.**

This frees the files for deletion.

2 Select **Start**→**Settings**→**Control Panel**→**Add/Remove Programs**.

3 **If the servlets and JRun or Apache Tomcat were installed on this computer using the Hyperion Performance Suite installer, remove JRun or Apache Tomcat first.**

4 Select **Hyperion Performance Server1** and click **Change/Remove**.

A command window is displayed, followed by the Hyperion Performance Suite uninstaller.

5 **Click Next to remove Hyperion Performance Suite Server from your system.**

6 **Select whether to remove all installed components, including customized and backup files, then click Next to continue.**

Customized files and backup files normally are not removed during the uninstall process. To remove all Hyperion Performance Suite services and components, select Yes.

7 **When the InstallShield wizard warns you that it is removing required services, click Next to continue.**

8 **If the InstallShield wizard detects JRun on your system, it displays a warning message; click Next to continue.**

The InstallShield wizard starts removing Hyperion Performance Suite Server components from your system.

9 **When prompted, specify whether to remove existing files that have been modified.**

The InstallShield wizard continues to remove Hyperion Performance Suite Server components from your system. This may take several minutes.

10 **When prompted, click Next to finish.**

11 **Select the Yes, restart my system option, then click Finish.**

You must restart your system to remove all Hyperion Performance Suite Server components.

12 **On Windows systems, after your system restarts, verify that these components have been removed. If not, delete them manually.**

- `\Brio8` directory
- In the Windows registry under HKEY_LOCAL_MACHINE→SOFTWARE:

  `Allaire` (for the JRun software)

  `Brio Software`
- In the Windows registry, for all listed **Control Sets** under HKEY_LOCAL_MACHINE→System:

  `Services→BPS BI_`*`host__host_port`*

  `Services→BPS DAS_`*`host_host_port`*

  `Services→BPS `*`host_host_port`*

  `Services→JRun Admin`

  `Services→JRun Default`

13 **Verify that the file `C:\Winnt\vpd.properties` exists but does not contain BRIO entries. This file is a registry used by all programs installed with ISMP.**

Removing Hyperion Performance Suite Server Using the Command Line

You can remove your Hyperion Performance Suite installation from a Command prompt. To do so, run these scripts:

- **Windows**—Run *INSTALL_HOME*_uninst\Uninstall.bat.
- **UNIX**—Run *INSTALL_HOME*/_uninst/Uninstall.sh.

Installation Removal Details

During the removal of a Hyperion Performance Suite server installation, the uninstall process writes log messages to BrioPlatformInstall.log. On Windows systems, the file is usually in \Documents and Settings*username*. On UNIX, it is in your user home (usually /home/*username*).

9 Client Tool Installation

Follow the instructions in this chapter to install Hyperion Intelligence Designer or Hyperion Intelligence Explorer.

In This Chapter

Installing the Hyperion Performance Suite Client Tools 132

Hyperion Designer and Hyperion Explorer Installation Details 135

Hyperion Intelligence Client Installation Details 136

Removing Hyperion Performance Suite Client Tools 134

Installing the Hyperion Performance Suite Client Tools

To install the Hyperion Performance Suite client tools:

1 **Run `setupWinPlatform.exe` from the Hyperion Performance Suite Intelligence Clients CD.**

The InstallShield Wizard starts and displays a progress bar while the JVM loads, followed by the Hyperion Performance Suite Client Tools splash screen, then by the Language dialog box.

2 **Select a language, then click Next.**

The Welcome dialog box is displayed.

3 **Click Next to proceed with the installation.**

The Software License Agreement dialog box is displayed.

4 **Read the Software License Agreement. To continue with the installation, select the I accept the terms of the license agreement option and click Next.**

The Enter License Key dialog box is displayed.

5 **If prompted, enter your license keys, then click OK.**

You must enter at least one license key to install the Hyperion Performance Suite files to your computer. When prompted by the installation software, enter the license key provided with the installation media. The key should be in the following format:

```
6L55-1P2E2
```

where dashes separate license keys. You may have multiple license keys, depending on the number of Hyperion Performance Suite components you purchased. You may also add keys at any time.

Note License keys are retained by your system throughout an installation session.

6 **Verify that the correct components are listed, then click Next.**

To install a product that is not listed, click **Add License Key**, enter the appropriate license key, then click **OK**.

7 **If your license key unlocks more than one client tool, select a client tool, then click Next.**

8 **Select an installation setup type, then click Next:**

- **Express** —Installs all licensed Hyperion Performance Suite client tools on this host computer, including program files and sample files.
- **Custom**—Enables you to select which Hyperion Performance Suite client tool files you want to install.

9 **Select the destination directory, then click Next:**

- **Default installation directory**—The default is `C:\Program Files\Brio\Brio8` on Windows or `/opt/Brio/Brio8` on UNIX.
- **New directory**—Type the full path in the **Directory Name** text box. If the directory does not exist, the installer creates it for you.
- **Pre-existing directory**—Select **Browse,** then navigate to the directory and click **OK**.

If the directory does not exist, click **Yes** when the installer asks if it should be created.

10 **If you selected the Custom installation option, select the client tools/files you want to install, then click Next.**

Note Sample files are available only for English.

11 **Select a Windows Start menu program folder, then click Next.**

12 **Verify your installation choices, then click Next. Click Back to change any of the listed parameters.**

The installer displays a progress bar while it automatically copies files into target directories.

13 **Complete the installation.**

Any errors that occur during the installation are listed on the dialog box that is displayed. The installer automatically selects the View Log option for you to review `Install.log`. If you encounter errors, view this log for information on diagnosing the failure.

Removing Hyperion Performance Suite Client Tools

Follow this procedure to remove the Hyperion Performance Suite client tools installation.

To remove the Hyperion Performance Suite client tools from a Windows system:

1. Select **Start**→**Settings**→**Control Panel**→**Add/Remove Programs**.
2. Select **Hyperion Intelligence Designer** or **Hyperion Intelligence Explorer** and click **Change/Remove**.

   A command window is displayed, followed by the Hyperion Performance Suite Install wizard.
3. Click **Next** to remove the Hyperion Performance Suite client tool from your system.
4. Select whether to remove all installed components, including customized and backup files, then click **Next** to continue.

   Customized files and backup files normally are not removed during the uninstall process. To remove all Hyperion Performance Suite customized files, select **Yes**.
5. When prompted, click **Finish**.

Hyperion Designer and Hyperion Explorer Installation Details

This section explains the directory structure implemented by the Hyperion Performance Suite Client Tools installer and lists the software components that it installs.

Directory Structure

For Hyperion Intelligence Designer and Hyperion Intelligence Explorer, the client tools files are installed in this directory:

```
C:\Program Files\Brio\Brio8\Client
```

The `\Client` directory may include up to three additional subdirectories, depending on the installation options you chose:

- `\Program`—Contains the `brioqry.exe` application file and the following directories:
  - `\BrioHelp`—Hyperion Designer and Hyperion Explorer help files
  - `\Connectivity`—Connectivity files and DataDirect Connect® for ODBC help files
  - `\Open Catalog Extensions`—Open catalog extension (OCE) files
- `\Samples`—Contains the Hyperion Performance Suite sample databases and sample BQY documents (English only)
- `\Template`—Contains templates that provide page styles when exporting a BQY section (or sections) using the HTML Wizard.

Program Start Folders

For Hyperion Intelligence Designer and Hyperion Intelligence Explorer, a Windows shortcut is added to the Start menu:

```
Start > Programs > Hyperion Solutions > Hyperion Performance
Suite > Developer Applications > Hyperion Intelligence Designer
```

or

```
Start > Programs > Hyperion Solutions > Hyperion Performance
Suite > Developer Applications > Hyperion Intelligence Explorer
```

Hyperion Intelligence Client Installation Details

This section lists the files installed and the Windows Registry entries made when you install Hyperion Intelligence Client (Web client).

All information contained in this section assumes that you are using the Microsoft Internet Explorer browser.

Files Installed with the Hyperion Intelligence Client

Table 9 lists the files installed during an Hyperion Intelligence Client installation.

Table 9 Files Installed with Hyperion Intelligence Client

File or Folder Location	File or Folder Name	Purpose
`C:\Program Files\Internet Explorer \PLUGINS\Brio8`	`axbqs32.dll`	Intelligence Client executable
`C:\Program Files\Internet Explorer \PLUGINS\Brio8`	`BILauncher.dll`	Intelligence Client executable
`C:\Program Files\Internet Explorer \PLUGINS\Brio8`	`briottbl.txt`	Localization data
`C:\WINNT`	`bqformat.ini`	Storage file for standard and user-defined numeric display formats
`C:\WINNT`	`bqmeta0.ini`	Information for metadata integration
`C:\WINNT`	`brioqplg.ini`	Storage file for system and user-defined numeric display formats

When you uninstall Hyperion Intelligence Client, all files and folders listed in Table 9 are removed.

Registry Entries

The installation process also writes certain information into the Windows registry. These registry entries define:

- Default display icons for the various types of Hyperion Intelligence files
- The way Hyperion Intelligence Client operates within the user's browser
- Hyperion Intelligence Client user settings

The complete listing of registry entries added by the installation of the Hyperion Intelligence Client is as follows:

```
[HKEY_CLASSES_ROOT\.bqd]
@="bqdfile"
"Content Type"="application/x-brioquerydata"
[HKEY_CLASSES_ROOT\.bqy]
@="bqyfile"
"Content Type"="application/x-brioquery"
[HKEY_CLASSES_ROOT\bqdfile]
@="Intelligence Data File"
"EditFlags"=dword:00010000
[HKEY_CLASSES_ROOT\bqdfile\CLSID]
@="{18FCE64F-A5F7-40e7-94B4-5D35C9687A64}"
[HKEY_CLASSES_ROOT\bqdfile\DefaultIcon]
@="C:\\Program Files\\Internet Explorer\\Plugins\\Brio8\\BILauncher.dll,2"
[HKEY_CLASSES_ROOT\bqdfile\DocObject]
@="0"
[HKEY_CLASSES_ROOT\bqyfile]
@="BrioQuery Document"
"CLSID"="{18FCE64F-A5F7-40e7-94B4-5D35C9687A63}"
"EditFlags"=dword:00010000
[HKEY_CLASSES_ROOT\bqyfile\CLSID]
@="{18FCE64F-A5F7-40e7-94B4-5D35C9687A63}"
[HKEY_CLASSES_ROOT\bqyfile\DocObject]
@="0"
[HKEY_CLASSES_ROOT\BQxLauncher.InsightLauncher]
@="BrioQuery Document"
[HKEY_CLASSES_ROOT\BQxLauncher.InsightLauncher\CLSID]
@="{18FCE64F-A5F7-40e7-94B4-5D35C9687A63}"
[HKEY_CLASSES_ROOT\BQxLauncher.InsightLauncher\CurVer]
@="BQxLauncher.InsightLauncher.2"
[HKEY_CLASSES_ROOT\BQxLauncher.InsightLauncher.2]
@="BrioQuery Document"
"EditFlags"=dword:00010000
[HKEY_CLASSES_ROOT\BQxLauncher.InsightLauncher.2\CLSID]
@="{18FCE64F-A5F7-40e7-94B4-5D35C9687A63}"
[HKEY_CLASSES_ROOT\BQxLauncher.InsightLauncher.2\DocObject]
@="0"
[HKEY_CLASSES_ROOT\BQxLauncher.InsightVersion]
@="InsightVersion Class"
[HKEY_CLASSES_ROOT\BQxLauncher.InsightVersion\CLSID]
```

```
@="{E434186E-E152-4292-A670-018360B8D82E}"
[HKEY_CLASSES_ROOT\BQxLauncher.InsightVersion\CurVer]
@="BQxLauncher.InsightVersion.1"
[HKEY_CLASSES_ROOT\BQxLauncher.InsightVersion.1]
@="InsightVersion Class"
[HKEY_CLASSES_ROOT\BQxLauncher.InsightVersion.1\CLSID]
@="{E434186E-E152-4292-A670-018360B8D82E}"
[HKEY_CLASSES_ROOT\BQxPlugin.PluginCtl]
@="PluginCtl Class"
[HKEY_CLASSES_ROOT\BQxPlugin.PluginCtl\CLSID]
@="{E460F621-92AB-11D5-80B4-00C04F209D79}"
[HKEY_CLASSES_ROOT\BQxPlugin.PluginCtl\CurVer]
@="BQxPlugin.PluginCtl.2"
[HKEY_CLASSES_ROOT\BQxPlugin.PluginCtl.2]
@="PluginCtl Class"
[HKEY_CLASSES_ROOT\BQxPlugin.PluginCtl.2\CLSID]
@="{E460F621-92AB-11D5-80B4-00C04F209D79}"
[HKEY_CLASSES_ROOT\CLSID\{18FCE64F-A5F7-40e7-94B4-5D35C9687A63}]
@="BrioQuery Document"
[HKEY_CLASSES_ROOT\CLSID\{18FCE64F-A5F7-40e7-94B4-5D35C9687A63}\Control]
[HKEY_CLASSES_ROOT\CLSID\{18FCE64F-A5F7-40e7-94B4-5D35C9687A63}\DocObject]
@="0"
[HKEY_CLASSES_ROOT\CLSID\{18FCE64F-A5F7-40e7-94B4-5D35C9687A63}\EnableFullPage]
[HKEY_CLASSES_ROOT\CLSID\{18FCE64F-A5F7-40e7-94B4-
5D35C9687A63}\EnableFullPage\.bqd]
[HKEY_CLASSES_ROOT\CLSID\{18FCE64F-A5F7-40e7-94B4-
5D35C9687A63}\EnableFullPage\.bqy]
[HKEY_CLASSES_ROOT\CLSID\{18FCE64F-A5F7-40e7-94B4-5D35C9687A63}\Implemented
Categories]
[HKEY_CLASSES_ROOT\CLSID\{18FCE64F-A5F7-40e7-94B4-5D35C9687A63}\Implemented
Categories\{7DD95801-9882-11CF-9FA9-00AA006C42C4}]
[HKEY_CLASSES_ROOT\CLSID\{18FCE64F-A5F7-40e7-94B4-5D35C9687A63}\Implemented
Categories\{7DD95802-9882-11CF-9FA9-00AA006C42C4}]
[HKEY_CLASSES_ROOT\CLSID\{18FCE64F-A5F7-40e7-94B4-5D35C9687A63}\InprocHandler32]
@="ole32.dll"
[HKEY_CLASSES_ROOT\CLSID\{18FCE64F-A5F7-40e7-94B4-5D35C9687A63}\InprocServer32]
@="C:\\Program Files\\Internet Explorer\\Plugins\\Brio8\\BILauncher.dll"
"ThreadingModel"="Apartment"
[HKEY_CLASSES_ROOT\CLSID\{18FCE64F-A5F7-40e7-94B4-5D35C9687A63}\Insertable]
[HKEY_CLASSES_ROOT\CLSID\{18FCE64F-A5F7-40e7-94B4-5D35C9687A63}\MiscStatus]
@="0"
[HKEY_CLASSES_ROOT\CLSID\{18FCE64F-A5F7-40e7-94B4-5D35C9687A63}\MiscStatus\1]
@="131473"
[HKEY_CLASSES_ROOT\CLSID\{18FCE64F-A5F7-40e7-94B4-5D35C9687A63}\ProgID]
@="bqyfile"
[HKEY_CLASSES_ROOT\CLSID\{18FCE64F-A5F7-40e7-94B4-5D35C9687A63}\Programmable]
[HKEY_CLASSES_ROOT\CLSID\{18FCE64F-A5F7-40e7-94B4-5D35C9687A63}\ToolboxBitmap32]
@="C:\\Program Files\\Internet Explorer\\Plugins\\Brio8\\BILauncher.dll, 101"
[HKEY_CLASSES_ROOT\CLSID\{18FCE64F-A5F7-40e7-94B4-5D35C9687A63}\TypeLib]
@="{AD3A2EBC-D363-4ad2-B4BF-9286B744A3B9}"
[HKEY_CLASSES_ROOT\CLSID\{18FCE64F-A5F7-40e7-94B4-5D35C9687A63}\Version]
@="1.0"
[HKEY_CLASSES_ROOT\CLSID\{18FCE64F-A5F7-40e7-94B4-
5D35C9687A63}\VersionIndependentProgID]
@="BQxLauncher.InsightLauncher"
```

```
[HKEY_CLASSES_ROOT\CLSID\{18FCE64F-A5F7-40e7-94B4-5D35C9687A64}]
@="Intelligence Data File"
[HKEY_CLASSES_ROOT\CLSID\{18FCE64F-A5F7-40e7-94B4-5D35C9687A64}\Control]
[HKEY_CLASSES_ROOT\CLSID\{18FCE64F-A5F7-40e7-94B4-5D35C9687A64}\DocObject]
@="0"
[HKEY_CLASSES_ROOT\CLSID\{18FCE64F-A5F7-40e7-94B4-5D35C9687A64}\EnableFullPage]
[HKEY_CLASSES_ROOT\CLSID\{18FCE64F-A5F7-40e7-94B4-
5D35C9687A64}\EnableFullPage\.bqd]
[HKEY_CLASSES_ROOT\CLSID\{18FCE64F-A5F7-40e7-94B4-
5D35C9687A64}\EnableFullPage\.bqy]
[HKEY_CLASSES_ROOT\CLSID\{18FCE64F-A5F7-40e7-94B4-5D35C9687A64}\Implemented
Categories]
[HKEY_CLASSES_ROOT\CLSID\{18FCE64F-A5F7-40e7-94B4-5D35C9687A64}\Implemented
Categories\{7DD95801-9882-11CF-9FA9-00AA006C42C4}]
[HKEY_CLASSES_ROOT\CLSID\{18FCE64F-A5F7-40e7-94B4-5D35C9687A64}\Implemented
Categories\{7DD95802-9882-11CF-9FA9-00AA006C42C4}]
[HKEY_CLASSES_ROOT\CLSID\{18FCE64F-A5F7-40e7-94B4-5D35C9687A64}\InprocHandler32]
@="ole32.dll"
[HKEY_CLASSES_ROOT\CLSID\{18FCE64F-A5F7-40e7-94B4-5D35C9687A64}\InprocServer32]
@="C:\\Program Files\\Internet Explorer\\Plugins\\Brio8\\BILauncher.dll"
"ThreadingModel"="Apartment"
[HKEY_CLASSES_ROOT\CLSID\{18FCE64F-A5F7-40e7-94B4-5D35C9687A64}\Insertable]
[HKEY_CLASSES_ROOT\CLSID\{18FCE64F-A5F7-40e7-94B4-5D35C9687A64}\MiscStatus]
@="0"
[HKEY_CLASSES_ROOT\CLSID\{18FCE64F-A5F7-40e7-94B4-5D35C9687A64}\MiscStatus\1]
@="131473"
[HKEY_CLASSES_ROOT\CLSID\{18FCE64F-A5F7-40e7-94B4-5D35C9687A64}\ProgID]
@="bqdfile"
[HKEY_CLASSES_ROOT\CLSID\{18FCE64F-A5F7-40e7-94B4-5D35C9687A64}\Programmable]
[HKEY_CLASSES_ROOT\CLSID\{18FCE64F-A5F7-40e7-94B4-5D35C9687A64}\ToolboxBitmap32]
@="C:\\Program Files\\Internet Explorer\\Plugins\\Brio8\\BILauncher.dll, 101"
[HKEY_CLASSES_ROOT\CLSID\{18FCE64F-A5F7-40e7-94B4-5D35C9687A64}\TypeLib]
@="{AD3A2EBC-D363-4ad2-B4BF-9286B744A3B9}"
[HKEY_CLASSES_ROOT\CLSID\{18FCE64F-A5F7-40e7-94B4-5D35C9687A64}\Version]
@="1.0"
[HKEY_CLASSES_ROOT\CLSID\{E460F621-92AB-11D5-80B4-00C04F209D79}]
@="PluginCtl Class"
[HKEY_CLASSES_ROOT\CLSID\{E460F621-92AB-11D5-80B4-00C04F209D79}\Control]
[HKEY_CLASSES_ROOT\CLSID\{E460F621-92AB-11D5-80B4-00C04F209D79}\EnableFullPage]
[HKEY_CLASSES_ROOT\CLSID\{E460F621-92AB-11D5-80B4-
00C04F209D79}\EnableFullPage\.bqd]
[HKEY_CLASSES_ROOT\CLSID\{E460F621-92AB-11D5-80B4-
00C04F209D79}\EnableFullPage\.bqy]
[HKEY_CLASSES_ROOT\CLSID\{E460F621-92AB-11D5-80B4-00C04F209D79}\Implemented
Categories]
[HKEY_CLASSES_ROOT\CLSID\{E460F621-92AB-11D5-80B4-00C04F209D79}\Implemented
Categories\{7DD95801-9882-11CF-9FA9-00AA006C42C4}]
[HKEY_CLASSES_ROOT\CLSID\{E460F621-92AB-11D5-80B4-00C04F209D79}\Implemented
Categories\{7DD95802-9882-11CF-9FA9-00AA006C42C4}]
[HKEY_CLASSES_ROOT\CLSID\{E460F621-92AB-11D5-80B4-00C04F209D79}\InprocHandler32]
@="ole32.dll"
[HKEY_CLASSES_ROOT\CLSID\{E460F621-92AB-11D5-80B4-00C04F209D79}\InprocServer32]
@="C:\\Program Files\\Internet Explorer\\Plugins\\Brio8\\axbqs32.dll"
"ThreadingModel"="Apartment"
[HKEY_CLASSES_ROOT\CLSID\{E460F621-92AB-11D5-80B4-00C04F209D79}\Insertable]
```

```
[HKEY_CLASSES_ROOT\CLSID\{E460F621-92AB-11D5-80B4-00C04F209D79}\MiscStatus]
@="0"
[HKEY_CLASSES_ROOT\CLSID\{E460F621-92AB-11D5-80B4-00C04F209D79}\MiscStatus\1]
@="131473"
[HKEY_CLASSES_ROOT\CLSID\{E460F621-92AB-11D5-80B4-00C04F209D79}\ProgID]
@="BQxPlugin.PluginCtl.2"
[HKEY_CLASSES_ROOT\CLSID\{E460F621-92AB-11D5-80B4-00C04F209D79}\Programmable]
[HKEY_CLASSES_ROOT\CLSID\{E460F621-92AB-11D5-80B4-00C04F209D79}\ToolboxBitmap32]
@="C:\\Program Files\\Internet Explorer\\Plugins\\Brio8\\axbqs32.dll, 101"
[HKEY_CLASSES_ROOT\CLSID\{E460F621-92AB-11D5-80B4-00C04F209D79}\TypeLib]
@="{E460F60B-92AB-11D5-80B4-00C04F209D79}"
[HKEY_CLASSES_ROOT\CLSID\{E460F621-92AB-11D5-80B4-00C04F209D79}\Version]
@="2.0"
[HKEY_CLASSES_ROOT\CLSID\{E460F621-92AB-11D5-80B4-
00C04F209D79}\VersionIndependentProgID]
@="BQxPlugin.PluginCtl"
[HKEY_CLASSES_ROOT\CLSID\{E434186E-E152-4292-A670-018360B8D82E}]
@="InsightVersion Class"
[HKEY_CLASSES_ROOT\CLSID\{E434186E-E152-4292-A670-018360B8D82E}\Implemented
Categories]
[HKEY_CLASSES_ROOT\CLSID\{E434186E-E152-4292-A670-018360B8D82E}\Implemented
Categories\{7DD95801-9882-11CF-9FA9-00AA006C42C4}]
[HKEY_CLASSES_ROOT\CLSID\{E434186E-E152-4292-A670-018360B8D82E}\Implemented
Categories\{7DD95802-9882-11CF-9FA9-00AA006C42C4}]
[HKEY_CLASSES_ROOT\CLSID\{E434186E-E152-4292-A670-018360B8D82E}\InprocServer32]
@="C:\\Program Files\\Internet Explorer\\Plugins\\Brio8\\BILauncher.dll"
"ThreadingModel"="Apartment"
[HKEY_CLASSES_ROOT\CLSID\{E434186E-E152-4292-A670-018360B8D82E}\ProgID]
@="BQxLauncher.InsightVersion.1"
[HKEY_CLASSES_ROOT\CLSID\{E434186E-E152-4292-A670-018360B8D82E}\TypeLib]
@="{AD3A2EBC-D363-4ad2-B4BF-9286B744A3B9}"
[HKEY_CLASSES_ROOT\CLSID\{E434186E-E152-4292-A670-
018360B8D82E}\VersionIndependentProgID]
@="BQxLauncher.InsightVersion"
[HKEY_CLASSES_ROOT\Interface\{7889013D-AA6C-4DB7-B805-6D94DE96C5F3}]
@="ICtlHost6x"
[HKEY_CLASSES_ROOT\Interface\{7889013D-AA6C-4DB7-B805-
6D94DE96C5F3}\ProxyStubClsid]
@="{00020424-0000-0000-C000-000000000046}"
[HKEY_CLASSES_ROOT\Interface\{7889013D-AA6C-4DB7-B805-
6D94DE96C5F3}\ProxyStubClsid32]
@="{00020424-0000-0000-C000-000000000046}"
[HKEY_CLASSES_ROOT\Interface\{7889013D-AA6C-4DB7-B805-6D94DE96C5F3}\TypeLib]
@="{AD3A2EBC-D363-4AD2-B4BF-9286B744A3B9}"
"Version"="1.0"
[HKEY_CLASSES_ROOT\Interface\{ACB48BE9-68E4-40D1-A2D2-7456F1798CE1}]
@="IInsightVersion"
[HKEY_CLASSES_ROOT\Interface\{ACB48BE9-68E4-40D1-A2D2-
7456F1798CE1}\ProxyStubClsid]
@="{00020424-0000-0000-C000-000000000046}"
[HKEY_CLASSES_ROOT\Interface\{ACB48BE9-68E4-40D1-A2D2-
7456F1798CE1}\ProxyStubClsid32]
@="{00020424-0000-0000-C000-000000000046}"
[HKEY_CLASSES_ROOT\Interface\{ACB48BE9-68E4-40D1-A2D2-7456F1798CE1}\TypeLib]
@="{AD3A2EBC-D363-4AD2-B4BF-9286B744A3B9}"
```

```
"Version"="1.0"
[HKEY_CLASSES_ROOT\Interface\{E460F619-92AB-11D5-80B4-00C04F209D79}]
@="ILauncher"
[HKEY_CLASSES_ROOT\Interface\{E460F619-92AB-11D5-80B4-
00C04F209D79}\ProxyStubClsid]
@="{00020424-0000-0000-C000-000000000046}"
[HKEY_CLASSES_ROOT\Interface\{E460F619-92AB-11D5-80B4-
00C04F209D79}\ProxyStubClsid32]
@="{00020424-0000-0000-C000-000000000046}"
[HKEY_CLASSES_ROOT\Interface\{E460F619-92AB-11D5-80B4-00C04F209D79}\TypeLib]
@="{AD3A2EBC-D363-4AD2-B4BF-9286B744A3B9}"
"Version"="1.0"
[HKEY_CLASSES_ROOT\TypeLib\{AD3A2EBC-D363-4AD2-B4BF-9286B744A3B9}]
[HKEY_CLASSES_ROOT\TypeLib\{AD3A2EBC-D363-4AD2-B4BF-9286B744A3B9}\1.0]
@="BQxLauncher 1.0 Type Library"
[HKEY_CLASSES_ROOT\TypeLib\{AD3A2EBC-D363-4AD2-B4BF-9286B744A3B9}\1.0\0]
[HKEY_CLASSES_ROOT\TypeLib\{AD3A2EBC-D363-4AD2-B4BF-9286B744A3B9}\1.0\0\win32]
@="C:\\Program Files\\Internet Explorer\\Plugins\\Brio8\\BILauncher.dll"
[HKEY_CLASSES_ROOT\TypeLib\{AD3A2EBC-D363-4AD2-B4BF-9286B744A3B9}\1.0\FLAGS]
@="0"
[HKEY_CLASSES_ROOT\TypeLib\{AD3A2EBC-D363-4AD2-B4BF-9286B744A3B9}\1.0\HELPDIR]
@="C:\\Program Files\\Internet Explorer\\Plugins\\Brio8"
[HKEY_LOCAL_MACHINE\SOFTWARE\Classes\CLSID\{E434186E-E152-4292-A670-
018360B8D82E}]
@="InsightVersion Class"
[HKEY_LOCAL_MACHINE\SOFTWARE\Classes\CLSID\{E434186E-E152-4292-A670-
018360B8D82E}\Implemented Categories]
[HKEY_LOCAL_MACHINE\SOFTWARE\Classes\CLSID\{E434186E-E152-4292-A670-
018360B8D82E}\Implemented Categories\{7DD95801-9882-11CF-9FA9-00AA006C42C4}]
[HKEY_LOCAL_MACHINE\SOFTWARE\Classes\CLSID\{E434186E-E152-4292-A670-
018360B8D82E}\Implemented Categories\{7DD95802-9882-11CF-9FA9-00AA006C42C4}]
[HKEY_LOCAL_MACHINE\SOFTWARE\Classes\CLSID\{E434186E-E152-4292-A670-
018360B8D82E}\InprocServer32]
@="C:\\Program Files\\Internet Explorer\\Plugins\\Brio8\\BILauncher.dll"
"ThreadingModel"="Apartment"
[HKEY_LOCAL_MACHINE\SOFTWARE\Classes\CLSID\{E434186E-E152-4292-A670-
018360B8D82E}\ProgID]
@="BQxLauncher.InsightVersion.1"
[HKEY_LOCAL_MACHINE\SOFTWARE\Classes\CLSID\{E434186E-E152-4292-A670-
018360B8D82E}\TypeLib]
@="{AD3A2EBC-D363-4ad2-B4BF-9286B744A3B9}"
[HKEY_LOCAL_MACHINE\SOFTWARE\Classes\CLSID\{E434186E-E152-4292-A670-
018360B8D82E}\VersionIndependentProgID]
@="BQxLauncher.InsightVersion"
```

When you uninstall Hyperion Intelligence Client, all related registry entries are removed.

A Manually Setting Up the Hyperion Performance Suite Servlets

The Hyperion Performance Suite installer automatically configures Web environments to run with the Hyperion Performance Suite servlets. The servlets can be run either on a standalone HTTP server or as servlets on a commercial Web server.

Normally the installer performs these configurations automatically; however, manual configuration may be required if the installation fails, or if you selected the option to configure the Web environment manually during the installation process.

In This Appendix

Running the Hyperion Performance Suite Servlets as a Standalone HTTP Server 144

Running the Hyperion Performance Suite Servlets on a Commercial Web Server 145

Configuring Apache 1.3.x with JRun 3.1 148

Configuring Apache HTTP Server 2.x with Apache Tomcat 4.1.30 and Later. 154

Configuring Apache Tomcat 4.1.30 and Later Standalone 156

Configuring BEA WebLogic 6.x with WebLogic SP3 or Later. 162

Configuring BEA WebLogic 7.x 168

Configuring BEA WebLogic 8.x 176

Configuring IBM HTTP Server with WebSphere 4.0.5 and Later 182

Configuring IBM WebSphere 5.0.1 or 5.0.2. 187

Configuring the JRun 3.1 Built-in Web Server. 193

Configuring Microsoft IIS 6 with JRun 4.0 202

Configuring Microsoft IIS 6 with Apache Tomcat 4.1.30 and Later 206

Configuring Sun Java System Web Server with Native Servlet Engine 212

Running the Hyperion Performance Suite Servlets as a Standalone HTTP Server

You can implement the Hyperion Performance Suite servlets on a standalone HTTP server. Generally, this is not recommended, but it can be done for demonstration or diagnostic purposes.

To run the Hyperion Performance Suite servlets on a standalone HTTP server, select the option to install and deploy the JRun 3.1 built-in Web server or the Apache Tomcat 4.1.30 standalone Web server during the installation process.

To install and deploy Apache Tomcat standalone manually, see "Configuring Apache Tomcat 4.1.30 and Later Standalone" on page 156. To install and deploy JRun 3.1 manually, see "Configuring the JRun 3.1 Built-in Web Server" on page 193.

After JRun 3.1 is installed and deployed, start the JRun default server.

On Windows, the JRun default server is installed with a default Manual startup mode. You may change this to Automatic so that it starts whenever the host computer is brought up.

To run the Hyperion Performance Suite servlets, open your browser and enter this URL in the address bar:

```
http://host:8100/Hyperion/browse
```

On UNIX, use the following commands to start and stop the server:

```
INSTALL_HOME/JRun/bin/jrun -start
INSTALL_HOME/JRun/bin/jrun -stop
```

To run the Hyperion Performance Suite servlets on a standalone HTTP server on UNIX systems, insert the JRun Start Default Server command line into the UNIX server's startup script.

For more information, go to this URL:

```
http://host:8000/docs/dochome.htm
```

Running the Hyperion Performance Suite Servlets on a Commercial Web Server

The most efficient and secure way to configure the Hyperion Performance Suite servlets is as servlets running as part of a commercial Web server. This permits the use of secured sockets (HTTPS), which are supported by most commercial Web servers.

Secured sockets virtually eliminate the threat of someone on the network being able to view data transfers between the browser and the Web server. This solution also scales better, because it can be used in conjunction with Web server load balancing techniques when running multiple instances of a Web server.

This configuration is ideal for large, secure production environments. When configuring Web servers to run servlets, refer to specific Web server product documentation.

Supported Web Environments

View the Hyperion Performance Suite Support Matrix for the most up-to-date list of supported Web environments.

You can view the support matrix on the Hyperion Download Center.

Note Hyperion Solutions supports only the Web environment configurations listed in the Support Matrix for Hyperion Performance Suite Release 8.3. While other configuration combinations are possible, they are not supported by Hyperion Solutions; however, the Hyperion Performance Suite servlets are based on Sun Microsystems' Java Servlet API and therefore can be run on any Web server that supports this API.

JRun 3.1 is distributed on the Hyperion Performance Suite Servers CD and is installed and configured automatically depending on which installation option you click. Additional information about JRun is available at this Web address:

`http://www.macromedia.com/software/jrun/`

About Manual Configuration

You must configure your Web environment manually if the Hyperion Performance Suite installer did not automatically configure any of the Web environments that can be automatically configured, or if you selected Other Web server during the installation process.

The following sections provide step-by-step instructions on manually configuring the supported Web environments:

Servlet Engine Configuration Settings for Commercial Web Servers

If you elected to configure JRun 3.1 during the installation process, the installer automatically configures the servlet engine settings.

For JRun 3.1, you must manually configure additional document directories or virtual directories (depending on the terminology your Web server uses) after the installer completes the installation and configuration processes.

Verifying Servlet Configuration

To verify that the servlet engine is properly configured:

1. **Stop and restart the servlet engine and the Web server.**

2. **Use this URL to access the Hyperion Performance Suite servlets, but use either the well-known port 80 or no port assignment after the host name:**

   `http://host:port/Hyperion/browse/login`

   The Web server listens on port 80 for HTTP requests and communicates with JRun 3.1 for servlet-specific tasks through the JRun Connection Module.

When Using JRE 1.3.x

Hyperion Performance Suite release 8.3 makes use of Java Components for Encryption (JCE), which is bundled along with the standard JRE 1.4.2 installation. When using JRE 1.3.x with the servlets, manual changes are required.

To manually configure JRE 1.3.x to work with JCE:

1. **Install JCE by following option 1 as described in http://java.sun.com/products/jce/jce122_install.html.**

2. **Modify the JAVA_HOME/jre/lib/security/java.properties file by adding the SunJCE to the end of the list of providers, for example:**

```
#
# List of providers and their preference orders (see above):
#
security.provider.1=sun.security.provider.Sun
security.provider.2=com.sun.rsajca.Provider
security.provider.3=com.sun.crypto.provider.SunJCE
```

Caution! It is very important that you make sure that the number on the left side of the equal sign (which is 3 in the example), is unique for the SunJCE `security.provider` entry.

3. **Restart your Web server.**

Configuring Apache 1.3.x with JRun 3.1

This section explains how to configure the servlets manually on Apache 1.3.x with JRun 3.1. To configure this Web environment, you must perform the tasks discussed in the following sections:

1. "Preparing the WAR file" on page 148
2. "Deploying the Application" on page 149
3. "Setting the Classpath" on page 150
4. "Adjusting the ws.conf and Logging Parameters" on page 150
5. "Copying Required Files" on page 151
6. "Creating a JRUN Connection Module" on page 151
7. "Testing the Servlets" on page 153

Information on removing a JRun Connection Module is also provided.

Note Hyperion Solutions supports Apache 1.3.x with JRun 3.1 on those operating systems listed in the Hyperion Performance Suite 8.3 Support Matrix. While other configuration combinations are possible, they are not supported by Hyperion Solutions.

Information on required modifications for JRun-supported Web server types is included in the *JRun 3.1 Advanced Configuration Guide* available at this Web address:

`http://www.macromedia.com/support/jrun/documentation.html`

Preparing the WAR file

To prepare the WAR file:

➤ **Extract the `Brio.war` file from the `Brio.ear` file in the Hyperion Performance Suite Servers CD Disk 3 `\misc` directory to the root directory of the Web application.**

Deploying the Application

To deploy the application:

1 **Run the JRun Management Console from your Web browser.**

This is generally found at localhost:8000. (For instructions on manually deploying a WAR file, see "Manually Starting the WarDeploy Utility" on page 197.)

Note Before manually deploying the servlets, if a partial deployment has already occurred, use the `redeploy` command instead of the `deploy` command.

2 **Log on using the administrator password that you set up during installation.**

3 **Go to the JRun Default Server and click Web Applications.**

4 **Click Deploy an Application.**

5 **Supply the following information:**

- **Servlet War File or Directory**—Location of the WAR archive
- **JRun Server Name**—Server on which the Web application will be deployed; usually the JRun Default Server
- **Application Name**—Hyperion
- **Application Host**—All Hosts
- **Application URL**—`/Hyperion` (This is the Web application context.)
- **Application Deploy Directory**—*INSTALL_HOME*`\servlets\deployment` (The WAR file is extracted to this location.)

6 **Click deploy.**

If deployment is successful, the following message is displayed, and Hyperion is listed under Web Applications:

```
You have successfully deployed Hyperion into
INSTALL_HOME\servlets\deployment
```

If the message is not displayed, double-check all entries and try again.

7 **Select Web Applications→Hyperion→Servlet URL Mappings and verify that the `web.xml` file was properly read.**

Administrator, Browser, Scheduler, Personal Page, and Viewer should be defined.

8 **Select Web Applications→Hyperion→Servlet Definitions and verify that the `web.xml` file was properly read.**

Administrator, Browser, Scheduler, Personal Page, and Viewer should be defined.

Setting the Classpath

To set the classpath:

➤ **Select JRun Default Server→Java Settings and edit the classpath to include these settings:**

```
INSTALL_HOME\servlets\deployment\WEB-INF\lib\foundation.jar
INSTALL_HOME\servlets\deployment\WEB-INF\lib
INSTALL_HOME\servlets\deployment\WEB-INF\classes
```

Adjusting the ws.conf and Logging Parameters

To adjust the `ws.conf` and logging parameters:

1 **Open the *INSTALL_HOME*\servlets\config\ws.conf file and modify the `WebClient.Diagnostics.Log.LoggingService` and `WebClient.UserInterface.Login.RMServer` parameters to have the correct values (GSM_*computer_name*:GSM_*port_number*).**

2 **Open the *INSTALL_HOME*\servlets\deployment\config\ws.conf file and modify the path to `servletLog4JConfig.xml` to the directory in which the file is located.**

3 **Open the *INSTALL_HOME*\servlets\deployment\config\servletLog4JConfig.xml file and modify the path to the `appenders.xml` and `debugLoggers.xml` files to the directory in which these files are located.**

4 **Open the *INSTALL_HOME*\servlets\deployment\config\appenders.xml file and modify the path to the log files to the directory in which the file is located.**

Copying Required Files

To copy required files:

1. **Copy** `Brio.Insight.en.cab`, `Brio.InsightNoHelp.en.cab`, `Brio.Insight.en.xpi`, **and** `Brio.InsightNoHelp.en.xpi` **from the Servers CD Disk 3** `\misc\zeroadmin` **directory to the** *`INSTALL_HOME`*`\servlets\deployment\zeroadmin\component` **directory.**

2. **Extract the help files from the Servers CD Disk 3** `\misc\help` **to the** *`INSTALL_HOME`*`\servlets\deployment\help directory`**.**

Note To simplify the process of unzipping help, the installation includes a file in the *`INSTALL_HOME`*`/install` directory called `unzipWebComponentHelp.bat` or `unzipWebComponentHelp.sh`. In this file, fill in the location of the help files (for example, `mountedDrive\serverCD\disk3\misc\help`), the location of a JDK on your system, and the target deployment location (for example, `/home/websphere/installedApps/Brio.ear/Brio.war/`). When you run the script, the help is unzipped.

Creating a JRUN Connection Module

To create a JRun connection module:

1. **In the JRun Management Console, select the JRun server that you want to connect to an external Web server.**

2. **Specify the Web server type, version, and platform, then click Next.**

Note For Apache Web servers, you must select 1.3.9+ for your Web server version even if you are using a lower version of Apache.

A warning message is displayed that instructs you to stop your Web server before proceeding.

Note For IIS **5**, you also must stop the World Wide Web Publishing Service. Use Control Panel→Services to stop this service.

3 **Stop the Web server (and publishing service if IIS 5) as instructed, then click OK.**

4 **In Step 2 of the Connector wizard, accept the default settings for the JRun Server IP address and connector port, then click Next.**

You may change these settings if there is a conflict in your setup.

5 **In Step 3 of the Connector wizard, enter the path where the JRun Connection Module will be installed.**

- **Apache**—Enter the location of the Apache `\conf` directory. Also make sure the **DSO Support** check box is selected

  To run Apache servlets, Dynamic Shared Object (DSO) must be configured. DSO is a mechanism which provides a way to build a piece of program code in a special format for loading at run time into the address space of an executable program. The DSO gets knowledge of the executable program symbol set as if it had been statically linked with it in the first place. DSO is required for connecting to external servlet engines such as JRun.

  On Microsoft Windows systems, DSO is always enabled. On UNIX systems, instructions for configuring DSO are available at:

  `http://httpd.apache.org/docs/dso.html.`

- **IIS 5**—Enter the location of the IIS `\scripts` directory. This is usually in `C:\Inetpub\scripts`.

6 **Click Next.**

The JRun Connection Module is successfully installed.

7 **Restart the JRun Default server and the Web server to make sure that the updated configurations are loaded.**

Testing the Servlets

Test the servlets by running JRun.

To test the servlets:

1. **Restart the JRun Default Server.**
2. **Use the original URL to access the servlets, but use either the well known port 80 or no port assignment after the host name:**

   `http://host/Hyperion/browse/login`

   The Web server now listens on port 80 for HTTP request and communicates with JRun for servlet-specific tasks through the JRun Connection Module.

Removing a JRun Connection Module

JRun provides utilities to modify the registry and metabase; full instructions for their use are included in the `Advanced JRun Configuration.pdf` file available from Macromedia.

The following example shows the command lines used to remove a JRun Connection Module. In this example, IIS 5 is the Web server, JRun is installed in `D:\Program Files\Allaire\JRun`, and the IIS `scripts` file is in `C:\inetpub\scripts`:

```
"D:\Program Files\Allaire\JRun\connectors\isapi\intel-win\regset" -d
"c:\inetpub\scripts"
"D:\Program Files\Allaire\JRun\connectors\isapi\intel-win\metaset" -d /1
"c:\inetpub\scripts"
"D:\Program Files\Allaire\JRun\connectors\isapi\intel-win\metaset" -d /0
"c:\inetpub\scripts"
```

Configuring Apache HTTP Server 2.x with Apache Tomcat 4.1.30 and Later

To configure and test the servlets on Apache HTTP Server 2.x with Tomcat 4.1.30 and later release, you must perform the tasks discussed in the following sections:

1. "Deploying the Servlets on Apache Tomcat" on page 154
2. "Adding a Jakarta Connector" on page 154
3. "Creating a workers2.properties File" on page 155
4. "Testing the Servlets" on page 155.

The following sections provide step-by-step instructions for completing these tasks.

Note The Web Services feature is supported only with Apache Tomcat installations.

Note Hyperion Solutions supports Apache 2.x HTTP Server with Apache Tomcat 4.1.30 and later on those operating systems listed in the Hyperion Performance Suite 8.3 Support Matrix. While other configuration combinations are possible, they are not supported by Hyperion Solutions.

Deploying the Servlets on Apache Tomcat

For instructions on deploying the servlets on Apache Tomcat, see "Configuring Apache Tomcat 4.1.30 and Later Standalone" on page 156.

Adding a Jakarta Connector

To add a Jakarta connector:

1 **Download the Jakarta Tomcat connector for your platform.**

Download the connector from `http://jakarta.apache.org/site/binindex.cgi`. You also can build the connector from source code at `http://jakarta.apache.org/site/sourceindex.cgi`.

2 **Place the connector file in the `APACHE2_HOME\modules` folder.**

3 **Add this line to the `APACHE2_HOME\conf\httpd.conf` file:**

```
LoadModule jk2_module module/CONNECTOR_FILE
```

where *`CONNECTOR_FILE`* is the name of the connector file for your platform.

Creating a workers2.properties File

To create a `workers2.properties` file:

1 **In the `APACHE2_HOME\conf` directory, use a text editor to create a new file and enter these lines:**

```
[shm:]
disabled=1

[channel.socket:localhost:8009]
info=Ajp13 forwarding over socket
tomcatId=localhost:8009

# Map the Tomcat examples webapp to the Web server uri space
[uri:/examples/*]
[uri:/admin/*]
[uri:/manager/*]
[uri:/Hyperion/*]
[uri:/wsmedia/*]
```

2 **Save the file as `workers2.properties`.**

3 **Restart Apache.**

Testing the Servlets

Test the servlets by running Apache/Tomcat.

To test the servlets:

➤ **In a Web browser, go to `http://localhost/Hyperion/`.**

Configuring Apache Tomcat 4.1.30 and Later Standalone

To configure and test the servlets on Apache Tomcat 4.1.30 and later standalone Web server, you must perform the tasks described in the following sections:

1. "Unpacking the Web Application Contents" on page 156
2. "Deploying Apache Tomcat 4.1.30 and Later" on page 157
3. "Adjusting the ws.conf and Logging Parameters" on page 159
4. "Precompiling the JSPs" on page 159
5. "Copying Required Files" on page 161
6. "Testing the Servlets" on page 161

Note The Web Services feature is supported only with Tomcat installations.

Unpacking the Web Application Contents

Steps 1 and 2 are required only if you chose the "Do not configure servlet engine at this time" option during Hyperion Performance Suite installation. Skip these steps if you chose the "Install servlet files in preparation for manual deployment" option. If you are uncertain which option you selected, check whether the *INSTALL_HOME*`\servlets\deployment` directory exists.

To unpack the Web application contents:

1. **Copy the `Brio.ear` from the `\misc` directory on the Servers CD Disk 3 to a temporary directory and uncompress it.**
2. **Move the `Brio.war` file from the temporary directory you created in Step 1 to the *INSTALL_HOME*`\servlets\deployment` directory and uncompress it.**

Note On UNIX, use this command: `jar -xvf Brio.war`

This recreates the application root and WEB-INF root for the servlets.

3 **Remove the** `Brio.war` **file from the** *INSTALL_HOME*`\servlets\deployment` **directory.**

Deploying Apache Tomcat 4.1.30 and Later

To deploy Apache Tomcat:

1 **Start Apache Tomcat.**

You can use the following start scripts:

- **Windows**—*TOMCAT_HOME*`\bin\startup.bat`
- **UNIX**—*TOMCAT_HOME*`/bin/startup.sh`

2 **In a Web browser, go to** `http://`*COMPUTER_NAME*`:8080/admin`.

3 **Enter the administrator username and password, then click Login to start the Apache Tomcat Web Server Administration Tool.**

In order to use the Tomcat Web Server Administration Tool, you may need to create a user with appropriate privileges. To add an admin user, open the *TOMCAT_HOME*`\conf\tomcat-users.xml` file and add the following lines to the `<tomcat-users>` element:

```
<role rolename="manager"/>
<role rolename="admin"/>
<user username="admin" password="admin_password"
fullName="Tomcat admin" roles="admin,manager"/>
```

Save the modified XML file, then restart Apache Tomcat.

4 **Expand the Tomcat Server→Service (Tomcat Standalone) entry in the left Navigation frame, then select Host (localhost).**

5 **Select Create New Context from the Host Actions drop-down list to create a context for the servlets.**

6 **Change the Context Properties information as follows, then click Save:**

- **Document Base**—Path to your servlets directory; for example:
  - **Windows**—`C:\Program Files\Brio\Brio8\servlets\deployment`
  - **UNIX**—`/export/home/brio/brio8/servlets/deployment`
- **Path**—`/Hyperion`

Do not change any other settings.

7 **Click Commit Changes.**

8 **Select Create New Context from the Host Actions drop-down list to create a context for wsmedia.**

9 **Change the Context Properties information as follows, then click Save:**

- **Document Base**—Path to your wsmedia directory; for example:
  - **Windows**—`C:\Program Files\Brio\Brio8\servlets\deployment\wsmedia`
  - **UNIX**—`/export/home/brio/brio8/servlets/deployment/wsmedia`
- **Path**—`/wsmedia`

Do not change any other settings.

10 **Click Save, then click Commit Changes.**

11 **Stop Apache Tomcat.**

You can use the following stop scripts:

- **Windows**—*TOMCAT_HOME*`\bin\shutdown.bat`
- **UNIX**—*TOMCAT_HOME*`/bin/shutdown.sh`

12 **Change the DEPLOYMENT_LIB and DEPLOYMENT_PATH variables in the** *INSTALL_HOME*`/servlets/bin/servletconfig.bat` **file on Windows, or the** *INSTALL_HOME*`\servlets\bin\servletconfig.sh` **file on UNIX, to point to the deployment directory.**

Adjusting the ws.conf and Logging Parameters

To adjust the `ws.conf` and logging parameters:

1. **Open the *INSTALL_HOME*\servlets\config\ws.conf file and modify the WebClient.Diagnostics.Log.LoggingService and WebClient.UserInterface.Login.RMServer parameters to have the correct values (GSM_*computer_name*:GSM_*port_number*).**
2. **Open the *INSTALL_HOME*\servlets\deployment\config\ws.conf file and modify the path to servletLog4JConfig.xml to the directory in which the file is located.**
3. **Open the *INSTALL_HOME*\servlets\deployment\config \servletLog4JConfig.xml file and modify the path to the appenders.xml and debugLoggers.xml files to the directory in which these files are located.**
4. **Open the *INSTALL_HOME*\servlets\deployment\config\ appenders.xml file and modify the path to the log files to the directory in which the file is located.**

Precompiling the JSPs

Precompiling JSPs is optional, but it ensures better response time the first time a JSP page is requested.

To precompile the JSPs:

1. **In the precompile.bat file (Windows) or the precompile.sh file (UNIX), specify values for these variables:**
   - *JAVA_HOME*
   - *TOMCAT_HOME*
   - *SERVLETS_HOME*
2. **Save the changes.**
3. **Run the precompile.bat or precompile.sh file to compile all JSP pages.**

Sample precompile.bat

```
SET JAVA_HOME=<Java home directory>
SET TOMCAT_HOME=<Tomcat directory>
SET SERVLETS_HOME=<Servlets home>
SET CLASSPATH=./ant-contrib-0.4.jar;./ant.jar;./
optional.jar;%JAVA_HOME%\lib\tools.jar;%CLASSPATH%

echo %JAVA_HOME%\bin\java -classpath "%classpath%"
org.apache.tools.ant.Main -Dtomcat.home="%TOMCAT_HOME%" -
Dwebapp.name=Hyperion -Dwebapp.path=%SERVLETS_HOME% -buildfile
tomcat-precompile-jsp.xml
%JAVA_HOME%\bin\java -classpath "%classpath%"
org.apache.tools.ant.Main -Dtomcat.home="%TOMCAT_HOME%" -
Dwebapp.name=Hyperion -Dwebapp.path=%SERVLETS_HOME% -buildfile
tomcat-precompile-jsp.xml
```

Sample precompile.sh

```
#!/bin/sh
JAVA_HOME=<Java home directory>
TOMCAT_HOME=<Tomcat directory>
SERVLETS_HOME=<Servlets home>
CLASSPATH=./ant-contrib-0.4.jar:./ant.jar:./
optional.jar:"$JAVA_HOME"\lib\tools.jar:"$CLASSPATH"
echo "$JAVA_HOME"/bin/java -classpath "$CLASSPATH"
org.apache.tools.ant.Main -Dtomcat.home="$TOMCAT_HOME" -
Dwebapp.name=Hyperion -Dwebapp.path="$SERVLETS_HOME" -buildfile
tomcat-precompile-jsp.xml
"$JAVA_HOME"/bin/java -classpath "$CLASSPATH"
org.apache.tools.ant.Main -Dtomcat.home="$TOMCAT_HOME" -
Dwebapp.name=Hyperion -Dwebapp.path="$SERVLETS_HOME" -buildfile
tomcat-precompile-jsp.xml
```

Copying Required Files

To copy required files:

1. **Copy** `Brio.Insight.en.cab`, `Brio.InsightNoHelp.en.cab`, `Brio.Insight.en.xpi`, **and** `Brio.InsightNoHelp.en.xpi` **from the Servers CD Disk 3** `\misc\zeroadmin` **directory to the** *`INSTALL_HOME`*`\servlets\deployment\zeroadmin\component` **directory.**

2. **Extract the help files from the Servers CD Disk 3** `\misc\help` **to the** *`INSTALL_HOME`*`\servlets\deployment\help directory`.

Note To simplify the process of unzipping help, the installation includes a file in the *`INSTALL_HOME`*`\install` directory called `unzipWebComponentHelp.bat` or `unzipWebComponentHelp.sh`. In this file, fill in the location of the help files (for example, `mountedDrive\serverCD\disk3\misc\help`), the location of a JDK on your system, and the target deployment location (for example, *`INSTALL_HOME`*`\servlets\deployment`). When you run the script, the help is unzipped.

Testing the Servlets

Test that the servlets run in the Web server.

To test the servlets:

1. **Start Apache Tomcat.**

   You can use the following start scripts:

   - **Windows**—*`TOMCAT_HOME`*`\bin\startup.bat`
   - **UNIX**—*`TOMCAT_HOME`*`/bin/startup.sh`

2. **In a Web browser, go to** `http://`*`COMPUTER_NAME`*`:8080/Hyperion/`.

Configuring BEA WebLogic 6.x with WebLogic SP3 or Later

To manually configure BEA WebLogic 6.x with WebLogic SP3 or later, you must perform the tasks described in the following sections:

- "Unpacking the Web Application Contents" on page 162
- "Modifying the startWebLogic Script" on page 163
- "Updating the ws.config Parameters" on page 164
- "Updating the Logging Parameters" on page 164
- "Copying Required Files" on page 164
- "Deploying the Servlets" on page 165
- "Adding the wsmedia Virtual Directory" on page 166
- "Precompiling WebLogic JSPs" on page 167
- "Testing the Servlets" on page 167

Follow the instructions in the these topics only if you need to configure WebLogic 6.x manually.

Note Hyperion Solutions supports BEA WebLogic 6.x with WebLogic SP3 or later on those operating systems listed in the Hyperion Performance Suite 8.3 Support Matrix. While other configuration combinations are possible, they are not supported by Hyperion Solutions.

Unpacking the Web Application Contents

You need to unpack the Web application contents only if you selected the "Do not configure servlet engine at this time" option during Hyperion Performance Suite installation. If you are not sure you selected this option, check whether the *INSTALL_HOME*`\servlets\deployment` directory exists.

To unpack the Web application contents:

1. **Copy the `Brio.ear` from the `\misc` directory on the Servers CD Disk 3 to a temporary directory and uncompress it.**

   This extracts the `Brio.war` file and `\META-INF` directory from the `Brio.ear` file.

Note On UNIX, use this command: *JAVA_HOME*/bin/jar -xvf *FILENAME*

2 **Create an** *INSTALL_HOME*\servlets\deployment **directory.**

3 **Move the** Brio.war **file from the temporary directory that you created in Step 2 to the** *INSTALL_HOME*\servlets\deployment **directory and uncompress it.**

4 **Remove the** Brio.war **file from the** *INSTALL_HOME*\servlets\deployment **directory.**

Modifying the startWebLogic Script

To modify the startWebLogic script:

1 **Open the** startWebLogic.bat **(Windows) or** startWebLogic.sh **(UNIX) script.**

On Windows, this script may be in C:\bea\wlserver6.1\config\mydomain; on UNIX, the script may be in /home/user/bea/wlserver6.1/config/mydomain.

2 **In the** startWebLogic **script:**

- Add one of the following additional classpath settings to the beginning of the current CLASSPATH:
  - **Windows:**

```
CLASSPATH=.;.\lib\weblogic_sp.jar;.\lib\weblogic.jar;INSTALL_HOME\WEB-INF\lib
\iona61.jar;INSTALL_HOME\WEB-INF\lib
```

  - **UNIX:**

```
CLASSPATH=.:./lib/weblogic_sp.jar:./lib/weblogic.jar:INSTALL_HOME/WEB-INF/lib
/iona61.jar:INSTALL_HOME/WEB-INF/lib
```

- Modify the WebLogic startup parameters:
  - Enter values for the JAVA_HOME location and the WL_HOME variable. These values may have been set by the WebLogic installer.
  - Set STARTMODE=true. This starts the WebLogic Application Server in production mode.
  - Set WLS_PW value. This is the system user password required to start the Administration Console. If you do not provide a value, you will be prompted for it at startup.

3 **Save and close the** startWebLogic **script.**

Updating the ws.config Parameters

You must update the `ws.config` parameters only if you selected the "Do not configure servlet engine at this time" option during Hyperion Performance Suite installation.

To update the `ws.config` parameters:

➤ **Open the *INSTALL_HOME*\servlets\deployment\config\ws.conf file and modify the WebClient.Diagnostics.Log.LoggingService and WebClient.UserInterface.Login.RMServer parameters to have the correct values (GSM_computer_name:GSM_port_number value).**

Updating the Logging Parameters

To update the logging parameters:

1. **Open the *INSTALL_HOME*\servlets\deployment\config\ws.conf file and modify the path to servletLog4JConfig.xml to the directory in which the file is located.**
2. **Open the *INSTALL_HOME*\servlets\deployment\config\servletLog4JConfig.xml file and modify the path to the appenders.xml and debugLoggers.xml files to the directory in which these files are located.**
3. **Open the *INSTALL_HOME*\servlets\deployment\config\appenders.xml file and modify the path to the log files to the directory in which the files are located.**

Copying Required Files

You must copy required files only if you selected the "Do not configure servlet engine at this time" option during Hyperion Performance Suite installation.

To copy required files:

1. **Copy Brio.Insight.en.cab, Brio.InsightNoHelp.en.cab, Brio.Insight.en.xpi, and Brio.InsightNoHelp.en.xpi from the Servers CD Disk 3 \misc\zeroadmin directory to the *INSTALL_HOME*\servlets\deployment\zeroadmin\component directory.**

2 **Extract the help files from the Servers CD Disk 3** `\misc\help` **directory to the** *INSTALL_HOME*`\servlets\deployment\help` **directory.**

Note To simplify the process of unzipping help, the installation includes a file in the *INSTALL_HOME*`\install` directory called `unzipWebComponentHelp.bat` or `unzipWebComponentHelp.sh`. In this file, fill in the location of the help files (for example, `mountedDrive\serverCD\disk3\misc\help`), the location of a JDK on your system, and the target deployment location (for example, *INSTALL_HOME*`\servlets\deployment`). When you run the script, the help will be unzipped. You may also obtain this script from Hyperion Customer Support.

Deploying the Servlets

To modify the startWebLogic script:

1 **Use** `sh startWebLogic.sh` **to start the Administration Console.**

Note You need permission to execute scripts and write to directories and log files.

2 **Use a Web browser to access the Administration Console.**

This is generally at `http://`*`localhost`*`:7001/console`.

3 **Enter the username and password.**

4 **In the Administrator Console, select Deployment→Web Applications.**

5 **Select Configure a new Web Application.**

6 **On the Configuration General page, enter these parameters, then click Create:**

- Name—Hyperion
- Path—*INSTALL_HOME*`/servlets/deployment`

7 **Select the Targets tab, move one of the Available servers to the Chosen list, then click Apply.**

The servlets are deployed.

Note To improve performance, Hyperion recommends that you change the Reload Period to -1 (on the Files page).

Adding the wsmedia Virtual Directory

To add the `\wsmedia` virtual directory:

1. **Create a new WEB-INF directory in** *INSTALL_HOME*`\servlets\deployment\wsmedia`.

2. **In the newly created** `\WEB-INF` **directory, use a text editor to create a new file and enter these lines:**

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE web-app PUBLIC "-//Sun Microsystems, Inc.//DTD Web Application 2.2//EN"
"http://java.sun.com/dtd/web-app_2_2.dtd">
<web-app>
</web-app>
```

3. **Save the file as** `web.xml`.

4. **In a Web browser, go to** `http://`*localhost*`:7001/console`.

5. **In the left Navigation frame, select Deployments→Web Applications, then click Configure a new Web Application.**

6. **On the Configuration General page, enter these parameters, then click Create.**
   - Name—wsmedia
   - Path—*INSTALL_HOME*`/servlets/deployment/`wsmedia

7. **Select the Targets tab, move one of the available servers to the Chosen list, then click Apply.**

8. **The** `\wsmedia` **directory is deployed.**

Precompiling WebLogic JSPs

Hyperion Solutions recommends that you deploy the Hyperion Performance Suite servlets with precompiled JSP pages. This boosts page rendering performance, because pages will already be in a compiled state.

- **For WebLogic 6.0 JSP precompilation**—`web.xml` must have the following tags for JSP precompilation upon deployment of the servlets:

  ```
  <context-param>
     <param-name>weblogic.jsp.precompile</param-name>
     <param-value>true</param-value>
  </context-param>
  ```

  This `context-param` is explicitly for WebLogic installations; the base `web.xml` does not contain these tags.

- **For WebLogic 6.1 and later JSP precompilation**—The Hyperion Performance Suite installer automatically precompiles the JSP pages. WebLogic uses *INSTALL_HOME*`/servlets/deployment/WEB-INF/weblogic.xml` for precompilation.

You must restart WebLogic or redeploy the servlets for the precompilation to take effect.

Some Web servers automatically precompile after deployment, and some have a check box which lets you elect to precompile or not. After you manually deploy the Hyperion Performance Suite servlets, you may need to run the precompile steps. To determine this, look in the folder `/deployment/jsp` for `.class` files. If the `.class` files are there, the servlets are already precompiled.

Precompilation is an optional step. It has a big effect on performance, but for a development or test system you do not need to do it.

Testing the Servlets

Test that the servlets run in the Web server.

To test the servlets:

➤ **In a Web browser, go to** `http://`*`localhost`*`:7001/Hyperion/browse/login`.

Configuring BEA WebLogic 7.x

To configure and test the servlets on BEA WebLogic 7.x, you must perform the tasks described in the following sections:

1. "Unpacking the Servlets" on page 168
2. "Modifying the startWebLogic Script" on page 169
3. "Modifying the startWLS Script" on page 169
4. "Modifying the CLASSPATH Variable" on page 170
5. "Updating the ws.conf Parameters" on page 172
6. "Updating the Logging Parameters" on page 172
7. "Copying Required Files" on page 173
8. "Deploying the Servlets" on page 174
9. "Adding the wsmedia Virtual Directory" on page 174
10. "Precompiling the JSPs" on page 175
11. "Testing the Servlets" on page 175

Note If you are running WebLogic 7.x in a secure environment, please read the troubleshooting item on page 237 for important deployment information.

Unpacking the Servlets

You must unpack the servlets only if you selected the "Do not configure servlet engine at this time" option during Hyperion Performance Suite installation. Skip this step if you selected the "Install servlet files in preparation for manual deployment" option. If you are uncertain which option you selected, check whether the *INSTALL_HOME*`\servlets\deployment` directory exists.

To unpack the servlets:

1 **Copy the `Brio.ear` from the `\misc` directory on the Servers CD Disk 3 to a temporary directory and uncompress it.**

This extracts the `Brio.war` file and `\META-INF` directory from the `Brio.ear` file.

Note On UNIX, use this command: `jar -xvf Brio.war`

2 **Create an** *INSTALL_HOME*`\servlets\deployment` **directory.**

3 **Move the** `Brio.war` **file from the temporary directory to the** *INSTALL_HOME*`\servlets\deployment` **directory and uncompress it.**

4 **Remove the** `Brio.war` **file from the** *INSTALL_HOME*`\servlets\deployment` **directory.**

Modifying the startWebLogic Script

To modify the `startWebLogic` script:

1 **Open the** `startWebLogic.cmd` **(Windows) or** `startWebLogic.sh` **(UNIX) script.**

On Windows, this script may be in `C:\bea\user_projects\domains\mydomain`; on UNIX, the script may be in `/home/user/bea/user_projects/domains/mydomain`.

2 **In the** `startWebLogic` **script:**

- Set STARTMODE=true. This starts the WebLogic Application Server in production mode.
- Enter values for WLS_USER and WLS_PW. This is the system user name and password required to start the Administration Console.

3 **Save and close the** `startWebLogic` **script.**

Modifying the startWLS Script

To modify the `startWLS` script:

1 **Open the** `startWLS.cmd` **(Windows) or** `startWLS.sh` **(UNIX) script.**

On Windows, this script may be in `C:\BEA\weblogic700\server\bin`; on UNIX, the script may be in `/home/user/bea/weblogic700/server/bin`.

2 **In the** `startWLS` **script,** enter values for `JAVA_HOME` and `WL_HOME`.

These values may have been set by the WebLogic installer.

3 **Save and close the** `startWLS` **script.**

Modifying the CLASSPATH Variable

To modify the CLASSPATH variable on **Windows**:

➤ **Add the following lines to the `startWLS.cmd` file, directly before the line that starts the WebLogic server:**

```
set HPS_SERVLETS=<INSTALL_HOME>\servlets\deployment
set HPS_SERVLETS_LIB=%HPS_SERVLETS%\WEB-INF\lib
set HPS_CLASSPATH=%HPS_SERVLETS%\WEB-INF\classes;%HPS_CLASSPATH%
set HPS_CLASSPATH=%HPS_CLASSPATH%;%HPS_SERVLETS_LIB%\administrator.jar
set HPS_CLASSPATH=%HPS_CLASSPATH%;%HPS_SERVLETS_LIB%\aods.jar
set HPS_CLASSPATH=%HPS_CLASSPATH%;%HPS_SERVLETS_LIB%\bqservice.jar
set HPS_CLASSPATH=%HPS_CLASSPATH%;%HPS_SERVLETS_LIB%\bqxmlutil.jar
set HPS_CLASSPATH=%HPS_CLASSPATH%;%HPS_SERVLETS_LIB%\browser.jar
set HPS_CLASSPATH=%HPS_CLASSPATH%;%HPS_SERVLETS_LIB%\castor-0.9.5-xml.jar
set HPS_CLASSPATH=%HPS_CLASSPATH%;%HPS_SERVLETS_LIB%\clientservlet.jar
set HPS_CLASSPATH=%HPS_CLASSPATH%;%HPS_SERVLETS_LIB%\commons-httpclient.jar
set HPS_CLASSPATH=%HPS_CLASSPATH%;%HPS_SERVLETS_LIB%\commons-logging.jar
set HPS_CLASSPATH=%HPS_CLASSPATH%;%HPS_SERVLETS_LIB%\commons_collections.jar
set HPS_CLASSPATH=%HPS_CLASSPATH%;%HPS_SERVLETS_LIB%\comutil1_01.jar
set HPS_CLASSPATH=%HPS_CLASSPATH%;%HPS_SERVLETS_LIB%\concurrent.jar
set HPS_CLASSPATH=%HPS_CLASSPATH%;%HPS_SERVLETS_LIB%\css-2_5_3.jar
set HPS_CLASSPATH=%HPS_CLASSPATH%;%HPS_SERVLETS_LIB%\das.jar
set HPS_CLASSPATH=%HPS_CLASSPATH%;%HPS_SERVLETS_LIB%\dashboard.jar
set HPS_CLASSPATH=%HPS_CLASSPATH%;%HPS_SERVLETS_LIB%\foundation.jar
set HPS_CLASSPATH=%HPS_CLASSPATH%;%HPS_SERVLETS_LIB%\interop-common.jar
set HPS_CLASSPATH=%HPS_CLASSPATH%;%HPS_SERVLETS_LIB%\interop-sdk.jar
set HPS_CLASSPATH=%HPS_CLASSPATH%;%HPS_SERVLETS_LIB%\iona61.jar
set HPS_CLASSPATH=%HPS_CLASSPATH%;%HPS_SERVLETS_LIB%\jecf.jar
set HPS_CLASSPATH=%HPS_CLASSPATH%;%HPS_SERVLETS_LIB%\jhall.jar
set HPS_CLASSPATH=%HPS_CLASSPATH%;%HPS_SERVLETS_LIB%\log4j.jar
set HPS_CLASSPATH=%HPS_CLASSPATH%;%HPS_SERVLETS_LIB%\logi.crypto1.1.2.jar
set HPS_CLASSPATH=%HPS_CLASSPATH%;%HPS_SERVLETS_LIB%\personalpage.jar
set HPS_CLASSPATH=%HPS_CLASSPATH%;%HPS_SERVLETS_LIB%\scheduler.jar
set HPS_CLASSPATH=%HPS_CLASSPATH%;%HPS_SERVLETS_LIB%\servletutil.jar
set HPS_CLASSPATH=%HPS_CLASSPATH%;%HPS_SERVLETS_LIB%\sfc.jar
set HPS_CLASSPATH=%HPS_CLASSPATH%;%HPS_SERVLETS_LIB%\slide-webdavlib.jar
set HPS_CLASSPATH=%HPS_CLASSPATH%;%HPS_SERVLETS_LIB%\spf82.jar
set HPS_CLASSPATH=%HPS_CLASSPATH%;%HPS_SERVLETS_LIB%\swingall.jar
set HPS_CLASSPATH=%HPS_CLASSPATH%;%HPS_SERVLETS_LIB%\symbeans.jar
set HPS_CLASSPATH=%HPS_CLASSPATH%;%HPS_SERVLETS_LIB%\tc.jar
set HPS_CLASSPATH=%HPS_CLASSPATH%;%HPS_SERVLETS_LIB%\xalan.jar
set HPS_CLASSPATH=%HPS_CLASSPATH%;%HPS_SERVLETS_LIB%\xercesImpl.jar
set HPS_CLASSPATH=%HPS_CLASSPATH%;%HPS_SERVLETS_LIB%\xml-apis.jar
set HPS_CLASSPATH=%HPS_CLASSPATH%;%HPS_SERVLETS_LIB%\xmlParserAPIs.jar

set CLASSPATH=%HPS_CLASSPATH%;%CLASSPATH%
```

On Windows, you may get the following error message:

```
The input line is too long.
```

This occurs when the path for *INSTALL_HOME*\servlets\deployment is too long. To work around this issue, move the \servlets\deployment directory closer to the root level; for example: C:\Hyperion\HPSU8.

To modify the CLASSPATH variable on **UNIX**:

➤ **Add the following lines to the** `startWLS.sh` **file, directly before the line that starts the WebLogic server:**

```
HPS_SERVLETS=<INSTALL_HOME>/servlets/Brio
HPS_SERVLETS_LIB="${HPS_SERVLETS}/WEB-INF/lib"
HPS_CLASSPATH="${HPS_SERVLETS}/WEB-INF/classes:${HPS_CLASSPATH}"
HPS_CLASSPATH="${HPS_CLASSPATH}:${HPS_SERVLETS_LIB}/administrator.jar"
HPS_CLASSPATH="${HPS_CLASSPATH}:${HPS_SERVLETS_LIB}/aods.jar"
HPS_CLASSPATH="${HPS_CLASSPATH}:${HPS_SERVLETS_LIB}/bqservice.jar"
HPS_CLASSPATH="${HPS_CLASSPATH}:${HPS_SERVLETS_LIB}/bqxmlutil.jar"
HPS_CLASSPATH="${HPS_CLASSPATH}:${HPS_SERVLETS_LIB}/browser.jar"
HPS_CLASSPATH="${HPS_CLASSPATH}:${HPS_SERVLETS_LIB}/castor-0.9.5-xml.jar"
HPS_CLASSPATH="${HPS_CLASSPATH}:${HPS_SERVLETS_LIB}/clientservlet.jar"
HPS_CLASSPATH="${HPS_CLASSPATH}:${HPS_SERVLETS_LIB}/commons-httpclient.jar”
HPS_CLASSPATH="${HPS_CLASSPATH}:${HPS_SERVLETS_LIB}/commons-logging.jar”
HPS_CLASSPATH="${HPS_CLASSPATH}:${HPS_SERVLETS_LIB}/commons_collections.jar"
HPS_CLASSPATH="${HPS_CLASSPATH}:${HPS_SERVLETS_LIB}/comutil1_01.jar"
HPS_CLASSPATH="${HPS_CLASSPATH}:${HPS_SERVLETS_LIB}/concurrent.jar"
HPS_CLASSPATH="${HPS_CLASSPATH}:${HPS_SERVLETS_LIB}/css-2_5_3.jar”
HPS_CLASSPATH="${HPS_CLASSPATH}:${HPS_SERVLETS_LIB}/das.jar"
HPS_CLASSPATH="${HPS_CLASSPATH}:${HPS_SERVLETS_LIB}/dashboard.jar"
HPS_CLASSPATH="${HPS_CLASSPATH}:${HPS_SERVLETS_LIB}/foundation.jar"
HPS_CLASSPATH="${HPS_CLASSPATH}:${HPS_SERVLETS_LIB}/interop-common.jar”
HPS_CLASSPATH="${HPS_CLASSPATH}:${HPS_SERVLETS_LIB}/interop-sdk.jar”
HPS_CLASSPATH="${HPS_CLASSPATH}:${HPS_SERVLETS_LIB}/iona61.jar"
HPS_CLASSPATH="${HPS_CLASSPATH}:${HPS_SERVLETS_LIB}/jecf.jar"
HPS_CLASSPATH="${HPS_CLASSPATH}:${HPS_SERVLETS_LIB}/jhall.jar"
HPS_CLASSPATH="${HPS_CLASSPATH}:${HPS_SERVLETS_LIB}/log4j.jar"
HPS_CLASSPATH="${HPS_CLASSPATH}:${HPS_SERVLETS_LIB}/logi.crypto1.1.2.jar"
HPS_CLASSPATH="${HPS_CLASSPATH}:${HPS_SERVLETS_LIB}/personalpage.jar"
HPS_CLASSPATH="${HPS_CLASSPATH}:${HPS_SERVLETS_LIB}/scheduler.jar"
HPS_CLASSPATH="${HPS_CLASSPATH}:${HPS_SERVLETS_LIB}/servletutil.jar"
HPS_CLASSPATH="${HPS_CLASSPATH}:${HPS_SERVLETS_LIB}/sfc.jar"
HPS_CLASSPATH="${HPS_CLASSPATH}:${HPS_SERVLETS_LIB}/slide-webdavlib.jar”
HPS_CLASSPATH="${HPS_CLASSPATH}:${HPS_SERVLETS_LIB}/spf82.jar"
HPS_CLASSPATH="${HPS_CLASSPATH}:${HPS_SERVLETS_LIB}/swingall.jar"
HPS_CLASSPATH="${HPS_CLASSPATH}:${HPS_SERVLETS_LIB}/symbeans.jar"
HPS_CLASSPATH="${HPS_CLASSPATH}:${HPS_SERVLETS_LIB}/tc.jar"
HPS_CLASSPATH="${HPS_CLASSPATH}:${HPS_SERVLETS_LIB}/xalan.jar"
HPS_CLASSPATH="${HPS_CLASSPATH}:${HPS_SERVLETS_LIB}/xercesImpl.jar"
HPS_CLASSPATH="${HPS_CLASSPATH}:${HPS_SERVLETS_LIB}/xml-apis.jar"
HPS_CLASSPATH="${HPS_CLASSPATH}:${HPS_SERVLETS_LIB}/xmlParserAPIs.jar"

CLASSPATH="${HPS_CLASSPATH}:${CLASSPATH}"
export CLASSPATH
```

Updating the ws.conf Parameters

You must update `ws.conf` parameters only if you selected the "Do not configure servlet engine at this time" option during Hyperion Performance Suite installation.

To update `ws.conf` parameters:

➤ **Open the *INSTALL_HOME*`\servlets\deployment\config\ws.conf` file and modify the `WebClient.Diagnostics.Log.LoggingService` and `WebClient.UserInterface.Login.RMServer` parameters to have the correct values (GSM_*computer_name*:GSM_*port_number*).**

Updating the Logging Parameters

To update the logging parameters:

1. **Open the *INSTALL_HOME*`\servlets\deployment\config\ws.conf` file and modify the path to `servletLog4JConfig.xml` to the directory in which the file is located.**
2. **Open the *INSTALL_HOME*`\servlets\deployment\config\servletLog4JConfig.xml` file and modify the path to the `appenders.xml` and `debugLoggers.xml` files to the directory in which these files are located.**
3. **Open the *INSTALL_HOME*`\servlets\deployment\config\appenders.xml` file and modify the path to the log files to the directory in which the files are located.**

Copying Required Files

You must copy required files only if you selected the "Do not configure servlet engine at this time" option during Hyperion Performance Suite installation.

To copy required files:

1. **Copy** `Brio.Insight.en.cab`, `Brio.InsightNoHelp.en.cab`, `Brio.Insight.en.xpi`, **and** `Brio.InsightNoHelp.en.xpi` **from the Servers CD Disk 3** `\misc\zeroadmin` **directory to the** *`INSTALL_HOME`*`\servlets\deployment\zeroadmin\component` **directory.**

2. **Extract the help files from the Servers CD Disk 3** `\misc\help` **directory to the** *`INSTALL_HOME`*`\servlets\deployment\help` **directory.**

   You must extract the help files to a directory named with the language extension used in the ZIP file name; for example:

   - Extract `Insight_Help_sc.zip` to the following directory:

     ```
     INSTALL_HOME\servlets\deployment\help\Insight\sc\BrioHelp
     ```

   - Extract `Intelligence_iServer_Help_fr.zip` to the following directory:

     ```
     INSTALL_HOME\servlets\deployment\help\iHTMLServlet\fr
     ```

Note To simplify the process of unzipping help, the installation includes a file in the *`INSTALL_HOME`*`\install` directory called `unzipWebComponentHelp.bat` or `unzipWebComponentHelp.sh`. In this file, fill in the location of the help files (for example, `mountedDrive\serverCD\disk3\misc\help`), the location of a JDK on your system, and the target deployment location (for example, *`INSTALL_HOME`*`\servlets\deployment`). When you run the script, the help is unzipped.

Deploying the Servlets

To deploy the servlets:

1 **Use `startWebLogic.cmd` (Windows) or `startWebLogic.sh` (UNIX) to start the WebLogic Server Administration Console.**

Note You must have permissions to execute scripts and write to directories and log files.

2 **In a Web browser, go to `http://localhost:7001/console`.**

3 **Enter a username and password, then click Sign In.**

4 **In the left Navigation frame, select Deployments→Web Applications, then select Configure a new Web Application.**

5 **Select *INSTALL_HOME*`\servlets\deployment` as the deployment location, then click [select].**

6 **Select a target server, review your choices, then click Configure and Deploy.**

The servlets are deployed.

Adding the wsmedia Virtual Directory

To add the `\wsmedia` virtual directory:

1 **Create a new WEB-INF directory in *INSTALL_HOME*`\servlets\deployment\wsmedia`.**

2 **In the newly created `\WEB-INF` directory, use a text editor to create a new file and enter these lines:**

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE web-app PUBLIC "-//Sun Microsystems, Inc.//DTD Web Application 2.3//EN"
"http://java.sun.com/dtd/web-app_2_3.dtd">
<web-app>
</web-app>
```

3 Save the file as `web.xml`.

4 In a Web browser, go to `http://localhost:7001/console`.

5 In the left Navigation frame, select **Deployments→Web Applications**, then click **Configure a new Web Application**.

6 Select *INSTALL_HOME*`\servlets\deployment\wsmedia` as the deployment location, then click **[select]**.

7 Review your choices, then click **Configure and Deploy**.

8 The `\wsmedia` directory is deployed.

Precompiling the JSPs

To turn on JSP precompilation during deployment:

➤ In the *INSTALL_HOME*`/servlets/deployment/WEB-INF/weblogic.xml` file, set the `precompile` parameter value to `true`.

Testing the Servlets

Test that the servlets run in the Web server.

To test the servlets:

➤ In a Web browser, go to `http://localhost:7001/Hyperion/browse/login`.

Configuring BEA WebLogic 8.x

To configure and test the servlets on BEA WebLogic 8.x, you must peform the tasks described in the following sections:

1. "Unpacking the Servlets" on page 176
2. "Modifying the startWebLogic Script" on page 177
3. "Updating the ws.conf Parameters" on page 178
4. "Updating the Logging Parameters" on page 178
5. "Customizing the Servlets Context Path" on page 179
6. "Copying Required Files" on page 179
7. "Precompiling the JSPs" on page 180
8. "Deploying the Servlets" on page 180
9. "Adding the wsmedia Virtual Directory" on page 181
10. "Testing the Servlets" on page 181

Note If you are running WebLogic 8.x in a secure environment, please read the troubleshooting item on page 237 for important deployment information.

Unpacking the Servlets

You must unpack the servlets only if you selected the "Do not configure servlet engine at this time" option during Hyperion Performance Suite installation. Skip this step f you selected the "Install servlet files in preparation for manual deployment" option. If you are uncertain which option you selected, check whether the *INSTALL_HOME*`\servlets\deployment` directory exists.

To unpack the servlets:

1 **Copy the `Brio.ear` from the `\misc` directory on the Servers CD Disk 3 to a temporary directory and uncompress it.**

This extracts the `Brio.war` file and `\META-INF` directory from the `Brio.ear` file.

Note On UNIX, use this command: `jar -xvf Brio.war`

2. **Create an** *INSTALL_HOME*`\servlets\deployment` **directory.**

3. **Move the** `Brio.war` **file from the temporary directory to the** *INSTALL_HOME*`\servlets\deployment` **directory and uncompress it.**

4. **Remove the** `Brio.war` **file from the** *INSTALL_HOME*`\servlets\deployment` **directory.**

Modifying the startWebLogic Script

To modify the `startWebLogic` script:

1. **Open the** `startWebLogic.bat` **(Windows) or** `startWebLogic.sh` **(UNIX) script.**

   On Windows, this script may be in `C:\bea\user_projects\domains\mydomain`; on UNIX, the script may be in `/home/user/bea/user_projects/domains/mydomain`.

2. **In the** `startWebLogic` **script, modify the WebLogic startup parameters:**

   - Enter values for the JAVA_HOME location and the WL_HOME variable. These values may have been set by the WebLogic installer.
   - Set PRODUCTION_MODE=true. This starts the WebLogic Application Server in production mode.
   - Modify WebLogic class path and Java options. Add the following lines before the line WebLogic uses for running Java:
     - On Windows (`startWebLogic.bat`):

```
set SERVLETS_LIB=INSTALL_HOME\servlets\deployment\WEB-INF\lib
set CLASSPATH=%SERVLETS_LIB%\iona61.jar;%SERVLETS_LIB%;%CLASSPATH%
set JAVA_OPTIONS=-Dorg.omg.CORBA.ORBClass=com.iona.corba.art.artimpl.ORBImpl
-Dorg.omg.CORBA.ORBSingletonClass=com.iona.corba.art.artimpl.ORBSingleton
```

     - On UNIX (`startWebLogic.sh`):

```
SERVLETS_LIB=INSTALL_HOME/servlets/deployment/WEB-INF/lib
CLASSPATH=${SERVLETS_LIB}/iona61.jar: ${SERVLETS_LIB}:${CLASSPATH}
export CLASSPATH
JAVA_OPTIONS=-Dorg.omg.CORBA.ORBClass=com.iona.corba.art.artimpl.ORBImpl
-Dorg.omg.CORBA.ORBSingletonClass=com.iona.corba.art.artimpl.ORBSingleton
```

3 **Save and close the `startWebLogic` script.**

4 **In the *BEA_HOME*/`user_projects/domains/my_domain/boot.properties` file, enter values for username and password.**

This is the system username and password required to start the Administration Console. After you run WebLogic, the username and password values from the `boot.properties` file are encrypted.

If you do not provide these values, WebLogic prompts you for them at startup.

5 **Save and close the `boot.properties` file.**

Updating the ws.conf Parameters

You must update `ws.conf` parameters only if you selected the "Do not configure servlet engine at this time" option during Hyperion Performance Suite installation.

To update `ws.conf` parameters:

➤ **Open the *INSTALL_HOME*`\servlets\deployment\config\ws.conf` file and modify the `WebClient.Diagnostics.Log.LoggingService` and `WebClient.UserInterface.Login.RMServer` parameters to have the correct values (GSM_*computer_name*:GSM_*port_number*).**

Updating the Logging Parameters

To update the logging parameters:

1 **Open the *INSTALL_HOME*`\servlets\deployment\config\ws.conf` file and modify the path to `servletLog4JConfig.xml` to the directory in which the file is located.**

2 **Open the *INSTALL_HOME*`\servlets\deployment\config\servletLog4JConfig.xml` file and modify the path to the `appenders.xml` and `debugLoggers.xml` files to the directory in which these files are located.**

3 **Open the *INSTALL_HOME*`\servlets\deployment\config\appenders.xml` file and modify the path to the log files to the directory in which the files are located.**

Customizing the Servlets Context Path

You can designate a custom context path for the servlets (the default is Hyperion).

To customize the servlets context path:

1. **Open the** `ws.conf` **file and replace** `/Hyperion` **in the** `WebClient.Applications.URL.*` **parameters with your custom context path.**
2. **Open the** `weblogic.xml` **file and set the value of the** `<context-root>` **element to your custom context path.**

Copying Required Files

You must copy required files only if you selected the "Do not configure servlet engine at this time" option during Hyperion Performance Suite installation.

To copy required files:

1. **Copy** `Brio.Insight.en.cab`, `Brio.InsightNoHelp.en.cab`, `Brio.Insight.en.xpi`, **and** `Brio.InsightNoHelp.en.xpi` **from the Servers CD Disk 3** `\misc\zeroadmin` **directory to the** *`INSTALL_HOME`*`\servlets\deployment\zeroadmin\component` **directory.**
2. **Extract the help files from the Servers CD Disk 3** `\misc\help` **directory to the** *`INSTALL_HOME`*`\servlets\deployment\help` **directory.**

Note To simplify the process of unzipping help, the installation includes a file in the *`INSTALL_HOME`*`\install` directory called `unzipWebComponentHelp.bat` or `unzipWebComponentHelp.sh`. In this file, fill in the location of the help files (for example, `mountedDrive\serverCD\disk3\misc\help`), the location of a JDK on your system, and the target deployment location (for example, *`INSTALL_HOME`*`\servlets\deployment`). When you run the script, the help is unzipped.

Precompiling the JSPs

To turn on JSP precompilation during deployment:

➤ **In the** *INSTALL_HOME*`/servlets/deployment/WEB-INF/weblogic.xml` **file, set the** `precompile` **parameter value to** `true`**.**

Deploying the Servlets

To deploy the servlets:

1. **Change the DEPLOYMENT_LIB and DEPLOYMENT_PATH variables in the** `/servlets/bin/servletconfig.bat` **file to point to the deployment directory.**
2. **Use** `startWebLogic.cmd` **(Windows) or** `startWebLogic.sh` **(UNIX) to start the WebLogic Server Administration Console.**

Note You must have permissions to execute scripts and write to directories and log files.

3. **In a Web browser, go to** `http://localhost:7001/console`**.**
4. **Enter a username and password, then click Sign In.**
5. **In the left Navigation frame, select Deployments→Web Applications Modules, then select Deploy a new Web Application Module.**
6. **Select** *INSTALL_HOME*`\servlets\deployment` **as the deployment location, then click Target Module.**
7. **Review your choices, then click Deploy.**

   The servlets are deployed.

Adding the wsmedia Virtual Directory

To add the `\wsmedia` virtual directory:

1. **Create a new WEB-INF directory in** *INSTALL_HOME*`\servlets\deployment\wsmedia`.

2. **In the newly created** `\WEB-INF` **directory, use a text editor to create a new file and enter these lines:**

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE web-app PUBLIC "-//Sun Microsystems, Inc.//DTD Web Application 2.3//EN"
"http://java.sun.com/dtd/web-app_2_3.dtd">
<web-app>
</web-app>
```

3. **Save the file as** `web.xml`.

4. **In a Web browser, go to** `http://localhost:7001/console`.

5. **In the left Navigation frame, select Deployments→Web Applications Modules, then click Deploy a new Web Application Module.**

6. **Select** *INSTALL_HOME*`\servlets\deployment\wsmedia` **as the deployment location, then click Target Module.**

7. **Review your choices, then click Deploy.**

8. **The** `\wsmedia` **directory is deployed.**

Testing the Servlets

Test that the servlets run in the Web server.

To test the servlets:

➤ **In a Web browser, go to** `http://localhost:7001/Hyperion/browse/login`.

where `Hyperion` is the context path. If you used a custom context path for the servlets, replace `Hyperion` with your custom context path.

Configuring IBM HTTP Server with WebSphere 4.0.5 and Later

To configure the IBM HTTP Server with WebSphere 4.0.5 and later releases manually, you must peform the tasks described in the following sections:

- "Prerequisites" on page 183
- "Manually Deploying the Servlets on WebSphere 4.0.5 and Later" on page 183

Note Hyperion Solutions supports IBM HTTP WebServer with WebSphere 4.0.5 and later releases on those operating systems listed in the Hyperion Performance Suite 8.3 Support Matrix. While other configuration combinations are possible, they are not supported by Hyperion Solutions.

Follow these instructions only if you need to configure IBM HTTP Server with WebSphere manually. On UNIX, the correct path for WebSphere 4.0.5 and later is usually `/usr/WebSphere/AppServer`.

Note Some Web servers automatically precompile after deployment, and some have a check box which lets you elect to precompile or not. After you manually deploy the servlets, you may need to run the precompile steps. To determine this, look in the folder `/usr/WebSphere/AppServer/installedApps/Brio.ear/Brio.war/jsp` for `.class` files. If the `.class` files are there, the servlets are already precompiled.

Precompilation is an optional step. It has a big effect on performance, but for a development or test system you do not need to do it.

Prerequisites

Before deploying the servlets on WebSphere 4.0.5 and later, make sure you have these items:

- The host name and port number of the computer on which the GSM is running (that is, GSM_*computer_name*:GSM_*port_number*)
- Root access to the server (for restarting WebSphere)
- At least 200 MB of free disk space

Manually Deploying the Servlets on WebSphere 4.0.5 and Later

To manually deploy the servlets on WebSphere 4.0.5 and later:

1. **Copy the `Brio.ear` file from the Hyperion Performance Suite Servers CD Disk 3 `/misc` directory to /tmp.**

2. **Use the WebSphere Administrative Console to deploy the Hyperion Web enterprise application to the WebSphere Application Server:**

   a. Use the `startupServer.sh` command (`/usr/WebSphere/AppServer/bin`) to run your WebSphere Server.

   b. Right-click **Enterprise Applications** and select **Install Enterprise Application** from the pop-up menu to start the Install Enterprise Application.

   c. Enter the following information, then click **Next**:

      - **Path**—The path to the `Brio.ear` file (See Step 1.)
      - **Application Name**—Brio

   d. Follow the wizard instructions and confirm your application installation settings by clicking **Finish** in the Completing the Application Installation Wizard window.

3. **In the WebSphere Advanced Administrative Console (for `\nodes\<host>`), click Application Servers, then select the JVM Settings tab.**

4 **Enter the classpath for the** *WAS_HOME*`/installedApps/Brio.ear/Brio.war/WEB-INF/classes` **directory into a table row, then click Apply.**

5 **Regenerate the plug-in configuration: Right-click the node (**`\nodes\<host>`**) and select Regen Webserver Plugin from the pop-up menu.**

6 **Open the** *WAS_HOME*`/installedApps/Brio.ear/Brio.war/config/ws.conf` **file and update these parameters:**

- `WebClient.Diagnostics.Log.LoggingService` and `WebClient.UserInterface.Login.RMServer` to `GSM_`*computer_name*`:GSM_`*port_number* value
- `WebClient.Internal.Redirect.RedirectPolicy` from `HttpRedirect` to `JavaScriptRedirect`

7 **Update these logging parameters:**

- In the *WAS_HOME*`/installedApps/Brio.ear/Brio.war/config/ws.conf` file, modify the path to `servletLog4JConfig.xml` to the directory in which the file is located.
- In the *WAS_HOME*`/installedApps/Brio.ear/Brio.war/config/servletLog4JConfig.xml` file, modify the path to `appenders.xml` and `debugLoggers.xml` to the directory in which these files are located.

  `servletLog4jConfig.xml` contains two variables marked by `${}`; for example:

```
<!ENTITY appenders SYSTEM "file:${log4j_config_directory}\appenders.xml">
```

  If your `servletLog4jConfig.xml` (and by association, `appenders.xml`) is in the `/home/websphere/installedApps/Brio.ear/Brio.war/config` directory, then replace `:${log4j_config_directory}` with `/home/websphere/installedApps/Brio.ear/Brio.war/config`.

The replacement happens in two places in the file:

- In the *WAS_HOME*/installedApps/Brio.ear/Brio.war/config/appenders.xml, modify the path to the log files to the directory in which these files are located.
- In appenders.xml there are two variables ${directory} and ${name} that need to be replaced with actual values. For example, to make your backup log files appear in the directory /home/websphere/installedApps/Brio.ear/Brio.war/config, and have the name servlets, you would change this line:

```
<param name="File" value="${directory}/log/${name}_backupMessages.log" />
```

To read as follows:

```
<param name="File" value="/home/websphere/installedApps/Brio.ear/Brio.war/config/
log/servlets_backupMessages.log" />
```

8 **Change the DEPLOYMENT_LIB and DEPLOYMENT_PATH variables in the** /servlets/bin/servletconfig.bat **file to point to the deployment directory.**

9 **Copy necessary files to** *WAS_HOME*/installedApps/Brio.ear/Brio.war/zeroadmin/**components and** *WAS_HOME*/installedApps/Brio.ear/Brio.war/help.

Depending on what language you want for your plug-in and help files, select the files from the Servers CD Disk 3 \misc\zeroadmin and \misc\help directories.

For the plug-in, copy four English files: Brio.Insight.en.cab, Brio.Insight.en.xpi, Brio.InsightNoHelp.en.cab, and Brio.InsightNoHelp.en.xpi to (for example) /home/websphere/installedApps/Brio.ear/Brio.war/zeroadmin/component.

For the help, there are also four English files: admin_help_en.zip, Insight_Help_en.zip, user_help_en.zip, Intelligence_iServer_Help_en.zip which need to be unzipped to (for example) /home/websphere/installedApps/Brio.ear/Brio.war/help.

Note To simplify the process of unzipping help, the installation includes a file in the *INSTALL_HOME*/`install` directory called `unzipWebComponentHelp.bat` or `unzipWebComponentHelp.sh`. In this file, fill in the location of the help files (for example, `mountedDrive\serverCD\disk3\misc\help`), the location of a JDK on your system, and the target deployment location (for example, `/home/websphere/installedApps/Brio.ear/Brio.war/`). When you run the script, the help is unzipped. You also can obtain these files by contacting Hyperion Customer Support.

10 **Shut down WebSphere.**

11 **Move the** *WASHOME*/`installedApps/Brio.ear/Brio.war/WEB-INF/lib/iona61.jar` **file to** *WASHOME*/`lib/ext`.

Note Make sure that you delete the `iona61.jar` file from the `WASHOME/installedApps/Brio.ear/Brio.war/WEB-INF/lib` directory.

12 **To avoid the problem with xerces parsers, apply WebSphere fix PQ76047 from this Web address:**

`http://www-1.ibm.com/support/docview.wss?rs=180&context=SSEQTP&q1=JSPG0103E&uid=swg24005064&loc=en_US&cs=utf-8&lang=en+en`

13 **Start WebSphere.**

14 **In a Web browser, go to** `http://`*servername:port*`/Hyperion` **to run the servlets.**

Configuring IBM WebSphere 5.0.1 or 5.0.2

To configure and test the servlets on IBM WebSphere 5.0.1 or 5.0.2, you must perform the tasks described in the following sections:

1. "Configuring WebSphere 5.0.1 or 5.0.2" on page 188
2. "Copying Required Files" on page 189
3. "Configuring the IBM HTTP Server" on page 190
4. "Configuring a Remote IBM HTTP Server" on page 191
5. "Testing the Servlets" on page 192

Please make note of the following items:

- Hyperion does not support WebSphere Application Server Global Security, so you are not prompted for a username or password. You must disable WebSphere Application Server Global Security (if enabled) before installation if you are using the Hyperion Performance Suite installer.
- The Hyperion Performance Suite servlets will not start properly after installation if Java 2 Security is enabled.
- Before installing the Hyperion Performance Suites servlets, verify that the WebSphere Application Server and the IBM HTTP Server are configured and working correctly by testing one of the following URLS:

  `http://server/snoop` or `https://server/snoop`

  If access to one of these URLS was successful, continue with the WebSphere Application Server and the IBM HTTP Server manual configuration steps. If you could not access one of these URLs, refer to the IBM documentation on how to configure ere Application Server and the IBM HTTP Server.

The following steps assume that WebSphere Application Server and the IBM HTTP Server are installed correctly.

Configuring WebSphere 5.0.1 or 5.0.2

To configure WebSphere:

1. **Start your WebSphere Server.**

   For example, on Windows, run `/IBM/AppServer/bin/startServer.bat server1`. On UNIX, run `/usr/WebSphere/AppServer5/bin/startServer.sh server1`.

2. **Enter the User ID, then click OK.**

3. **In the left Navigation frame, select Applications→Install New Application.**

4. **Select Server path, enter the path to the `Brio.ear` file in the `\misc` directory on the Servers CD Disk 3, then click Next.**

5. **Select Generate Default Bindings, then click Next.**

6. **Select the Pre-compile JSP (if you want to precompile JSPs) and Deploy EJBs AppDeployment options, then click Next.**

7. **Select Web Module and select a name from the Virtual Host drop-down list, click Next twice, then click Finish.**

   Wait while WebSphere deploys the application.

8. **Select Save in the Messages area, then select Save in the Save to Master Configuration area.**

9. **Open the `\WebSphere\AppServer\installedApps\SERVERNAME\Brio.ear/Brio.war\config\ws.conf` file and make these changes:**

   - Modify the `WebClient.Diagnostics.Log.Configuration` parameter to include the path to `servletLog4JConfig.xml`, which is /WebSphere/AppServer/installedApps/*SERVERNAME*/`Brio.ear/Brio.war/config/`.

Caution! Note that when specifying the path, you must use slashes (/), not backslashes (\).

- Modify the `WebClient.Diagnostics.Log.Configuration` parameter to GSM_computer_name:GSM_port_number value.
- Modify the `WebClient.UserInterface.Login.RMServer` parameter to GSM_computer_name:GSM_port_number value.

10 **Open the** `\WebSphere\AppServer\installedApps\`*`SERVERNAME`*`\Brio.ear\Brio.war\config\servletLog4JConfig.xml` **file and set the path for the** `appenders.xml` **and** `debugLoggers.xml` **files to the directory in which these files are located, which is** `/WebSphere/AppServer/installedApps/`***SERVERNAME***`/Brio.ear/Brio.war/config/`.

11 **Open the** `/WebSphere/AppServer/installedApps/`***SERVERNAME***`/Brio.ear/Brio.war/config/appenders.xml` **file and set the path for the log files to the directory in which this file is located.**

Note that when specifying the path, you must use forward slashes (/), not backslashes (\).

12 **Change the DEPLOYMENT_LIB and DEPLOYMENT_PATH variables in the** `/servlets/bin/servletconfig.bat` **file to point to the deployment directory.**

Copying Required Files

To copy required files:

1 **Copy the** `Brio.Insight.en.cab`, `Brio.InsightNoHelp.en.cab`, `Brio.Insight.en.xpi`, **and** `Brio.InsightNoHelp.en.xpi` **files from the Servers CD Disk 3** `\misc\zeroadmin` **directory to the** `\WebSphere\AppServer\installedApps\`*`SERVERNAME`*`\Brio.ear\Brio.war\zeroadmin\component` **directory.**

2 **Extract the help files from the Servers CD Disk 3** `\misc\help` **directory to the** `\WebSphere\AppServer\installedApps\`*`SERVERNAME`*`\Brio.ear\Brio.war\help` **directory.**

Note To simplify the process of unzipping help, the installation includes a file in the *INSTALL_HOME*/`install` directory called `unzipWebComponentHelp.bat` or `unzipWebComponentHelp.sh`. In this file, fill in the location of the help files (for example, `mountedDrive\serverCD\disk3\misc\help`), the location of a JDK on your system, and the target deployment location (for example, `/home/websphere/installedApps/Brio.ear/Brio.war/`). When you run the script, the help is unzipped.

3 **If you changed the default context root, in the `ws.conf` file, replace `/Hyperion` in the `WebClient.Applications.URL.*` parameters with your custom context path.**

4 **Restart WebSphere.**

Configuring the IBM HTTP Server

If your IBM HTTP server is on a different computer from the WebSphere application, see "Configuring a Remote IBM HTTP Server" on page 191 for instructions.

To configure the IBM HTTP Server:

1 **Add this line to the `<UriGroup>` section of the *WEBSPHERE_APPSERVER_ROOT*`\config\cells\plugin-cgf.xml` file:**

```
<Uri Name="/Context_root/*"/>
```

2 **Add this line to the `\`*IBM_HTTP_SERVER_ROOT*`\conf\httpd.conf` file to create an alias for `/wsmedia`:**

```
Alias /wsmedia "WEBSPHERE_APPSERVER_ROOT\installedApps
\SERVERNAME\Brio.ear\Brio.war\wsmedia\"
```

Configuring a Remote IBM HTTP Server

If you are using the IBM HTTPserver on a different computer from the WebSphere application server, you must manually configure the remote IBM HTTP Server after you have installed Hyperion Performance Suite.

To manually configure the IBM HTTP Server:

1. **Log on to the WebSphere Administrative Console.**
2. **Go to Environment→Update Web Server Plugin and click OK to generate the** `plugin-cfg.xml` **file.**
3. **Create a directory in** *IBM_HTTP_SERVER_HOME*, **for example:**

   IBM_HTTP_SERVER_HOME/brio
4. **Move the regenerated** `plugin-cfg.xml` **file from** *WAS_HOME*/config/cells/plugin-cfg.xml **to the directory that you created in Step 3.**
5. **Add the path to the moved** `plugin-cfg.xml` **file to the** *IBM_HTTP_SERVER_HOME*/conf/httpd.conf **file; for example:**

   WebSpherePluginConfig *IBM_HTTP_SERVER_HOME*/brio/plugin-cfg.xml
6. **Move the** *WAS_HOME*/bin/mod_ibm_app_server_http.so **(UNIX) or the** *WAS_HOME*/bin/mod_ibm_app_server_http.dll **(Windows) library file to the directory that you created in Step 3.**
7. **Add the path to the moved library file to the** *IBM_HTTP_SERVER_HOME*/conf/httpd.conf **file. For example, on UNIX:**

   LoadModule ibm_app_server_http_module *IBM_HTTP_SERVER_HOME* /brio/mod_ibm_app_server_http.so
8. **Copy the WebSphere** /wsmedia **directory on the remote computer to the local directory that you created in Step 3, for example:**

   IBM_HTTP_SERVER_HOME/brio/wsmedia
9. **Add this line to the** `httpd.conf` **file to create an alias for** /wsmedia**:**

   Alias /wsmedia "*IBM_HTTP_SERVER_HOME*/brio/wsmedia"
10. **Create an alias for the Hyperion application (which is the real path on the remote computer with WebSphere):**

    Alias Brio "*WAS_HOME*/installedApps/*NODE_NAME*/Brio.ear/Brio.war"

11 **Correct the path to the log file in the** *IBM_HTTP_SERVER_HOME*/brio/plugin-cfg.xml **file:**

```
<Log LogLevel="Error" Name="IBM_HTTP_SERVER_HOME/brio
/http_plugin.log"/>
```

12 **Correct the transport host name in the** *IBM_HTTP_SERVER_HOME*/brio/plugin-cfg.xml **file:**

```
<Transport Hostname="WebSphere_Machine_IP_Address" Port="9080" Protocol="http"/>
<Transport Hostname="WebSphere_Machine_IP_Address" Port="9443" Protocol="https">
```

13 **Copy these files from the remote WebSphere to the** *IBM_HTTP_SERVER_HOME*/brio/etc **directory:**

- plugin-key.kdb
- plugin-key.sth

14 **Correct the path to these files in the** *IBM_HTTP_SERVER_HOME*/brio/plugin-cfg.xml **file:**

```
<Property name="keyring" value="IBM_HTTP_SERVER_HOME/brio/etc/plugin-key.kdb"/>
<Property name="stashfile" value="IBM_HTTP_SERVER_HOME/brio/etc/plugin-key.sth"/>
```

15 **Start the IBM HTTP Server.**

Testing the Servlets

Test that the servlets run in the Web server.

To test the servlets:

➤ **In a Web browser, go to** http://localhost:7001/Hyperion/browse/login.

Configuring the JRun 3.1 Built-in Web Server

This section explains how to manually configure the stand-alone JRun 3.1 built-in Web server. It contains these topics:

- "Manually Installing JRun 3.1" on page 193
- "Deploying Servlets onto JRun 3.1" on page 196
- "Mapping the wsmedia Virtual Path" on page 200
- "Configuring the Classpath" on page 200
- "Starting the JRun Server Using the JRun Management Console" on page 201

Normally, the installer performs these configurations; however, manual configuration is required in these situations:

- The installation fails.
- You selected the option to configure the servlets manually during the installation process.
- You selected the option to configure the Web server manually during the installation process

Manually Installing JRun 3.1

This section contains the following topics:

- "Manual Installation Steps for JRun 3.1" on page 194
- "JRun 3.1 Installation Information Required for Windows" on page 195
- "Removing JRun 3.1" on page 195

You may want to manually install JRun 3.1 in these situations:

- Automatic installation of JRun 3.1 from the installer fails.
- You want to set custom values for the JRun 3.1 installation directory or the JRE directory or both.

Manual Installation Steps for JRun 3.1

To launch the JRun installer:

1. **Open the `JrunInstallMedia` directory on the Hyperion Performance Suite Servers CD Disk 3, then open the appropriate directory for your operating system (Windows or UNIX).**

2. **Run the setup file for your system.**

   - Windows—`setup.exe`
   - UNIX—`jrun-31-unix-us.sh`

   The InstallShield wizard starts and displays a progress bar while it prepares your system for installation.

3. **Select the destination directory, then click Next.**

   Hyperion Solutions suggests that you use the JRun folder in your *INSTALL_HOME*.

   The JRun installer prompts for the JRE location if it cannot find *INSTALL_HOME*`\jre`.

4. **Select a Windows Program folder, then click Next.**

5. **Enter a password for the JRun Management Console administration account, then click Next.**

6. **Click Finish to complete the JRun installation and start the JRun server.**

Troubleshooting the JRun 3.1 Installation

Errors and output that occurred during JRun installation can be found in the *INSTALL_HOME*`\log\install` directory, in the files `jRunInstallation.log`, `jRunInstallationError.log`, and `jrunInstallStdout.log`.

The main log file for the installation contains the command line used to run the installer, in this format:

```
"CDLOCATION\ JRunInstallMedia$\win\Setup.exe" -s -f1
INSTALL_HOME\install\setup.iss
```

where `setup.iss` contains input values for the installer (for example, the *INSTALL_HOME* location).

JRun 3.1 Installation Information Required for Windows

On Windows, the installer asks for the following four pieces of information:

- **Installation directory**—Hyperion Solutions recommends that the installation directory be placed under the *INSTALL_HOME* directory. The terms of the JRun license enable this servlet engine to be used with Hyperion Solutions products only. The default path for JRun installation is *INSTALL_HOME*\JRun.
- **JRE directory**—The JRE used by JRun is set by default to the Hyperion Solutions JRE version 1.4.2. This is automatically installed in *INSTALL_HOME*\jre.
- **Start menu folder**—The Start menu folder is generally set to Hyperion\Foundation Services\JRun.
- **Administrator password**—The administrator password should be set during installation.

All other information, such as the license key for using JRun 3.1 with Hyperion Performance Suite, the ports, and the service options, are automatically set. You can use the JRun Management Console to change any of these values later.

Removing JRun 3.1

Use one of these methods to remove JRun 3.1:

- Select **Start→Program Files→Hyperion Solutions→Foundation Services→JRun→Uninstall JRun**.
- Relaunch the JRun installer from the Hyperion Performance Suite Servers CD Disk 3.

  The installer asks if you want to remove JRun 3.1 then prompts you to restart your computer.

Deploying Servlets onto JRun 3.1

This section explains how to deploy the servlets manually. It contains these topics:

- "Troubleshooting Your JRun 3.1 Deployment" on page 196
- "Precompiling JRun 3.1" on page 197
- "Manually Starting the WarDeploy Utility" on page 197
- "Deploying the Application Using the JRun Management Console" on page 199

You may need to manually deploy the servlet engine if automatic deployment of servlets from the installer failed.

Note Before manually deploying the servlets, if a partial deployment has already occurred, use the `redeploy` command instead of the `deploy` command.

Troubleshooting Your JRun 3.1 Deployment

Errors and output that occurred during JRun deployment are in the *INSTALL_HOME*`/log/install` directory, in the files `jRunDeploy.log` and `jRunDeployError.log`.

The main log file for the installation contains the command line used to run the installer, in this format:

```
Finished running JRun deployment command: [deployment command
line].
```

Note The JRun Management console is known to cache pages. If you get a "page not found" error even after fixing a configuration problem, it could be the cache. You can clear it by stopping and restarting the JRun admin and default servers from the Windows Control Panel.

Precompiling JRun 3.1

Precompiling JRun greatly improves the performance of the Hyperion Performance Suite servlets. If you need to manually precompile JRun, the installer creates the script that you need to run, as follows:

- **Windows**—*INSTALL_HOME*`\install\jrun_precompile.bat`
- **UNIX**—*INSTALL_HOME*`/install/jrun_precompile.sh`

Manually Starting the WarDeploy Utility

This section describes the WarDeploy utility and explains how to use it to deploy Web applications manually. You can also use the JRun Management Console to deploy Web applications (see "Deploying the Application Using the JRun Management Console" on page 199).

Name

WarDeploy

Usage

```
java -cp [classpath] allaire.jrun.tools.WarDeploy
[-deploy|redeploy
[-config=property_file_name]
[deploy.war.path=value
deploy.server.name=value
deploy.webapp.name=value
deploy.context.path=value
deploy.webapp.rootdir=value
deploy.jrun.rootdir=value]
]
[-remove
deploy.server.name=value
deploy.jrun.rootdir=value
deploy.webapp.name=value
]
```

Description

The WarDeploy utility deploys, redeploys, or removes Web applications.

You can deploy an application with WarDeploy either by passing the six required parameters on the command line or with the `-config=`*`filename`* option where the indicated file contains these parameters.

To remove an application with WarDeploy, use the `-remove` option and pass the three required parameters on the command line or use the `-config=`*`filename`* option where *`filename`* is the name of a file that contains those three parameters.

Package

`allaire.jrun.tools`. You must include the `jrun.jar` file in your classpath.

Example

Command line used by the installer to deploy the servlets:

```
C:\PROGRA~1\JavaSoft\JRE\1.3.1\bin\java.exe -cp
".;f:\Progra~1\Allaire\JRun\lib\jrun.jar" allaire.jrun.tools.WarDeploy -deploy
deploy.war.path="d:\progra~1\Brio8\Servlets\deployment"
deploy.server.name=default deploy.webapp.name=Hyperion deploy.context.path=/
Hyperion deploy.webapp.rootdir="d:\progra~1\Brio8\Servlets\deployment"
deploy.jrun.rootdir="f:\Progra~1\Allaire\JRun"
```

Note Before manually deploying the servlets, if a partial deployment has already occurred, use the `redeploy` command instead of the `deploy` command.

Notes

If you deploy the application but it does not appear in the Admin Console (`http://localhost:8000/index.jsp`), check the `local.properties` file under the server you selected for deployment. For example, if you have a new JRun installation, the file is `C:\Program Files\Allaire \JRun\servers\default\local.properties`.

If you selected a JRun server during the installation, then it is in `C:\Program Files\Allaire\JRun\servers\`*`SERVERNAME`* `\local.properties`.

If the application is deployed properly, entries in the `local.properties` contains entries similar to these:

```
servlet.webapps=default-app,Hyperion
HPS.rootdir=D:\\Program Files\\Brio8\\ServletDeploy\\deployment
HPS.class={webapp.service-class}
webapp.mapping./HPS=HPS
```

If there was an error during deployment, look for log files in *`INSTALL_HOME`*`\log\install\jRunDeploy.log` and `jRunDeployErr.log`.

`Error unable to open ORB configuration file 'default-domain.cfg'` means that the classpath did not get set properly. In the JRun Admin UI, this is ServerName - Java Settings - Classpath.

Deploying the Application Using the JRun Management Console

To deploy the application:

1. **Expand the Web Application node and select Deploy an Application from the right frame.**

2. **Provide the requested information:**

   - **Servlet War File or Directory**—*INSTALL_HOME*/servlets/deployment
   - **JRun Server Name**—JRun Default Server
   - **Application Name**—Hyperion
   - **Application Host**—All Hosts
   - **Application URL**—/URL (not \URL) (use "/" and not "\")
   - **Application Deploy Directory**—*INSTALL_HOME*/servlets/deployment

Note Double-check every field before you click deploy because JRun sometimes resets some values.

3. **If deployment is successful, a message is displayed:**

   `You have successfully deployed Hyperion Performance Suite into` *INSTALL_HOME*`/servlets/deployment.`

   If the message is not displayed, double-check all the entries and try again.

Note Before manually deploying the servlets, if a partial deployment has already occurred, use the `redeploy` command instead of the `deploy` command.

4. **Expand the left frame and verify that Hyperion is listed under Web Applications.**

Mapping the wsmedia Virtual Path

To map the wsmedia virtual path:

1. **In the left frame, expand Default User Application, then select Virtual Mapping.**

   You may need to scroll.

2. **Click Edit.**

   The Edit Window is displayed.

3. **Add the following information, then click Update.**

   - **Virtual Path**—`/wsmedia`
   - **Mapping**—*`INSTALL_HOME`*`\servlets\deployment\wsmedia`

4. **Verify that wsmedia is listed in the left frame under Virtual Path.**

Configuring the Classpath

To configure the classpath:

1. **In the left frame, select JRun Default Server→Java Settings.**

2. **In the right frame, select Classpath, then click Edit.**

   The Edit Window is displayed.

3. **Add the following to the last entry in the Input text box, then click Update:**

   - `INSTALL_HOME/servlets/deployment/WEB-INF/lib/foundation.jar`
   - `INSTALL_HOME/servlets/deployment/WEB-INF/lib`
   - `INSTALL_HOME/servlets/deployment/WEB-INF/classes`

4. **Verify that the path you just entered is listed in the Classpath drop-down list in the right frame.**

Starting the JRun Server Using the JRun Management Console

To start the JRun server:

1. **In the left frame, select Default Server.**
2. **In the right frame, click Restart Server.**

   After you start JRun, you can invoke the servlets through a Web browser. The URL for your servlets would be similar to these:

   - `http://`*`jrun_host_name`*`:8100/Hyperion/administrator/login`
   - `http://`*`jrun_host_name`*`:8100/Hyperion/browse/login`
   - `http://`*`jrun_host_name`*`:8100/Hyperion/personalpage/login`
   - `http://`*`jrun_host_name`*`:8100/Hyperion/scheduler/login`
   - `http://`*`jrun_host_name`*`:8100/Hyperion/viewer/login`
3. **To replace the English language files, copy these files:**
   - Template language files—For example, for French, copy `\deployment\_templates\cctemplates_fr_FR` to `\deployment\cctemplates`.
   - Zero administration language files—For example, for French, copy these files from the Hyperion Performance Suite Servers CD Disc 3 `\misc\zeroadmin` directory to the *`INSTALL_HOME`*`\servlets\deployment\zeroadmin\component` directory:
     - `Brio.Insight.fr.cab`
     - `Brio.Insight.fr.xpi`
     - `Brio.InsightNoHelp.fr.cab`
     - `Brio.InsightNoHelp.fr.xpi`
   - `defaultdomain.cfg` and `licenses.txt` from *`INSTALL_HOME`*`\lib` to `\WEB-INF\classes`

Configuring Microsoft IIS 6 with JRun 4.0

Hyperion Performance Suite supports the Microsoft IIS 6 Web server with JRun 4.0 through manual configuration. To use this Web environment, you must first acquire JRun 4.0 and install it. For information on obtaining JRun 4.0, go to:

`http://www.macromedia.com/software/jrun/`

To configure and test the servlets on Microsoft IIS 6 with JRun 4.0, you must perform the tasks described in the following sections:

1. "Unpacking the Web Application Contents" on page 203
2. "Specify the Java VM Classpath" on page 203
3. "Deploying the Servlets under JRun" on page 204
4. "Configuring Microsoft IIS 6 with JRun 4.0" on page 205
5. "Adding the wsmedia Virtual Directory" on page 205
6. "Testing the Servlets" on page 205

Before you start the configuration process, make sure you have JRun 4 Updater 2. To check the JRun 4.0 version, run `jrun.exe` with the `-info` parameter, for example:

```
C:\JRun4\bin>jrun.exe -info
```

If the version is 61650 or later, you have JRun 4 Updater 2.

For more information, go to this Web address:

`http://www.macromedia.com/support/jrun/releasenotes/4/releasenotes_4_updater2.html`

Unpacking the Web Application Contents

Steps 1 and 2 are required only if you chose the "Do not configure servlet engine at this time" option during Hyperion Performance Suite installation. Skip these steps if you chose the "Install servlet files in preparation for manual deployment" option. If you are uncertain which option you selected, check whether the *INSTALL_HOME*`\servlets\deployment` directory exists.

To unpack the Web application contents:

1. **Copy the `Brio.ear` from the `\misc` directory on the Servers CD Disk 3 to a temporary directory and uncompress it.**
2. **Move the `Brio.war` file from the temporary directory that you created in Step 1 to the *INSTALL_HOME*`\servlets\deployment` directory and uncompress it.**

Note On UNIX, use this command: `jar -xvf Brio.war`

This recreates the application root and WEB-INF root for the servlets.

3. **Remove the `Brio.war` file from the *INSTALL_HOME*`\servlets\deployment` directory.**

Specify the Java VM Classpath

To specify a classpath for the JVM:

1. **Start the administration server.**
2. **In a Web browser, go to `http://localhost:8000/`.**
3. **Enter the administrator username and password, then click Login to start the JRun Managment Console.**
4. **On the left Navigation frame, select Default→Settings.**
5. **In the Classpaths for Java VM area, click Browse.**
6. **Navigate to the *INSTALL_HOME*`\servlets\deployment\WEB-INF\classes\` directory, click Open, then click Add.**

7 In the Classpaths for Java VM area, click **Browse** again.

8 Navigate to the *INSTALL_HOME*\servlets\deployment\WEB-INF\lib directory, click **Open**, then click **Add**.

9 Click **Update** (at the bottom of the Java VM Settings dialog box).

10 Open the *INSTALL_HOME*\servlets\deployment\config\ws.conf file and modify the WebClient.Diagnostics.Log.LoggingService and WebClient.UserInterface.Login.RMServer parameters for next values (GSM_*computer_name*:GSM_*port_number*).

11 Open the *INSTALL_HOME*\servlets\deployment\config\ws.conf file and set the servletLog4JConfig.xml path to the directory in which the file is located.

12 Open the *INSTALL_HOME*\servlets\deployment\config\servletLog4JConfig.xml file and set the path to the appenders.xml and debugLoggers.xml files to the directory in which these files are located.

13 Open the *INSTALL_HOME*\servlets\deployment\config\appenders.xml file and set the path for the log files to the directory in which this file is located.

Deploying the Servlets under JRun

To deploy the servlets under JRun:

1 On the left Navigation frame, select **default→J2EE Components**.

2 In the Web Applications area, click **Add**.

3 Enter (or browse to) the path to the *INSTALL_HOME*\servlets\deployment directory, then click **Deploy**.

4 In the General Settings for Hyperion dialog box, enter /Hyperion as the Context Path, then click **Apply**.

/Hyperion is listed in the Web Applications area.

Configuring Microsoft IIS 6 with JRun 4.0

To configure Microsoft IIS 6 with JRun 4.0:

1. Select **Start**→**Programs**→**Macromedia JRun 4**→**Web Server Configuration Tool**.
2. Click **Add**.
3. Change the connection parameter, if necessary, then click **OK**.
4. Click **Yes** to restart the Web server.
5. Click **Exit**.

Adding the wsmedia Virtual Directory

To add the `wsmedia` virtual directory:

1. Start the Internet Information Services (IIS) Manager.
2. Expand **Web Sites** on the Navigation frame, select your Web site (for example, Default Web Site), then right-click and select **New**→**Virtual Directory**.
3. Click **Next** in the Virtual Directory Creation wizard.
4. Enter `wsmedia` as the virtual directory alias, then click **Next**.
5. Enter (or browse to) the path to the *INSTALL_HOME*`\servlets\deployment\wsmedia` directory, then click Next.
6. Select **Read** for the virtual directory access permissions, then click **Next**.
7. Click **Finish** to close the Virtual Directory Creation wizard.

Testing the Servlets

Test that the servlets run in the Web server.

To test the servlets:

1. Restart the JRun server.
2. In a Web browser, go to `http://localhost/Hyperion/`.

Configuring Microsoft IIS 6 with Apache Tomcat 4.1.30 and Later

To configure and test the servlets on Microsoft IIS 6 with Apache Tomcat 4.1.30 and later, you must perform the described in the following sections:

1. "Unpacking the Web Application Contents" on page 206
2. "Adjusting the ws.conf and Logging Parameters" on page 207
3. "Copying Required Files" on page 207
4. "Precompiling the JSPs" on page 208
5. "Deploying Apache Tomcat 4.1.30 and Later" on page 208
6. "Configuring an Apache Tomcat to IIS 6 Connector" on page 209
7. "Adding the wsmedia Virtual Directory" on page 211
8. "Testing the Servlets" on page 211

Note The Web Services feature is supported only with Tomcat installations.

Unpacking the Web Application Contents

Steps 1 and 2 are required only if you chose the "Do not configure servlet engine at this time" option during Hyperion Performance Suite installation. Skip these steps if you chose the "Install servlet files in preparation for manual deployment" option. If you are uncertain which option you selected, check whether the *INSTALL_HOME*`\servlets\deployment` directory exists.

To unpack the Web application contents:

1. **Copy the `Brio.ear` from the `\misc` directory on the Servers CD Disk 3 to a temporary directory and uncompress it.**

2. **Move the `Brio.war` file from the temporary directory you created in Step 1 to the *INSTALL_HOME*`\servlets\deployment` directory and uncompress it.**

   On UNIX, use this command: *JAVA_HOME*`\bin\jar -xvf` *filename*

   This recreates the application root and WEB-INF root for the servlets.

3. **Remove the `Brio.war` file from the *INSTALL_HOME*`\servlets\deployment` directory.**

Adjusting the ws.conf and Logging Parameters

To adjust the `ws.conf` and logging parameters:

1. **Open the** *INSTALL_HOME*`\servlets\deployment\config\ws.conf` **file and modify the path to** `servletLog4JConfig.xml` **to the directory in which the file is located.**
2. **Open the** *INSTALL_HOME*`\servlets\deployment\config\servletLog4JConfig.xml` **file and modify the path to the** `appenders.xml` **and** `debugLoggers.xml` **files to the directory in which these files are located.**
3. **Open the** *INSTALL_HOME*`\servlets\deployment\config\appenders.xml` **file and modify the path to the log files to the directory in which the file is located.**

Copying Required Files

To copy required files:

1. **Copy** `Brio.Insight.en.cab`, `Brio.InsightNoHelp.en.cab`, `Brio.Insight.en.xpi`, **and** `Brio.InsightNoHelp.en.xpi` **from the Servers CD Disk 3** `\misc\zeroadmin` **directory to the** *INSTALL_HOME*`\servlets\deployment\zeroadmin\component` **directory.**
2. **Extract the help files from the Servers CD Disk 3** `\misc\help` **to the** *INSTALL_HOME*`\servlets\deployment\help` **directory.**

   You must extract the help files to a directory named with the language extension used in the ZIP file name; for example:

   - Extract `Insight_Help_sc.zip` to the following directory:

     `INSTALL_HOME\servlets\deployment\help\Insight\sc\BrioHelp`

   - Extract `Intelligence_iServer_Help_fr.zip` to the following directory:

     `INSTALL_HOME\servlets\deployment\help\iHTMLServlet\fr`

Note To simplify the process of unzipping help, the installation includes a file in the *INSTALL_HOME*`/install` directory called `unzipWebComponentHelp.bat` or `unzipWebComponentHelp.sh`. In this file, fill in the location of the help files (for example, `mountedDrive\serverCD\disk3\misc\help`), the location of a JDK on your system, and the target deployment location (for example, *INSTALL_HOME*`\servlets\deployment`), to run the script and unzip the help.

Precompiling the JSPs

Precompiling JSPs is optional, but it ensures better response time the first time a JSP page is requested.

To precompile the JSPs:

1 **In the `precompile.bat` file, specify values for these items:**

- *JAVA_HOME*
- *TOMCAT_HOME*
- *SERVLETS_HOME*

2 **Save the changes.**

3 **Run the `precompile.bat` file to compile all JSP pages.**

Deploying Apache Tomcat 4.1.30 and Later

To deploy Apache Tomcat:

1 **Open the Apache Tomcat Administration Console in a Web browser using this URL:**

`http://COMPUTER_NAME:8080/admin`

2 **Enter the administrator username (admin) and password (leave blank), then click Login to start the Apache Tomcat Web Server Administration Tool.**

3 **Expand the Tomcat Server→Service (Tomcat Standalone) entry in the left Navigation frame, then select Host (localhost).**

4 **Select Create New Context from the Host Actions drop-down list to create a context for the servlets.**

5 **Change the Context Properties information as follows, then click Save:**

- **Document Base**—Path to your servlets directory; for example:

  `C:\Program Files\Brio\Brio8\servlets\deployment`

- **Path**—`/Hyperion`

Do not change any other settings.

6 **Click Commit Changes.**

Configuring an Apache Tomcat to IIS 6 Connector

To configure a connector for Apache Tomcat and IIS:

1. **Download the Jakarta Tomcat ISAPI connector.**

   Download `jakarta-tomcat-connectors-jk2.0.2-win32-IIS.zip` from `http://www.apache.org/dist/jakarta/tomcat-connectors/jk2/binaries/win32/`.

2. **Create a `TOMCAT_INSTALL\isapi` directory and extract the `jakarta-tomcat-connectors-jk2.0.2-win32-IIS.zip` file to this directory.**

3. **In the `TOMCAT_INSTALL\isapi` directory, create a file named `workers2.properties` with the following contents:**

```
[shm:]
info=Shared memory file. Required for multiprocess servers
file=${serverRoot}\logs\jk2.shm
size=1000000

[channel.socket:localhost:8009]
info=Ajp13 worker, connects to tomcat instance using AJP 1.3
protocol
tomcatId=localhost:8009

[uri:/Hyperion/*]
info=HPS Servlets
context=/Hyperion
```

   Note that if your connector is configured to use a port other then 8009, you must update the `[channel.socket:localhost:8009]` and `tomcatId=localhost:8009` lines to match your port number. This number can be verified in `TOMCAT_INSTALL\conf\server.xml` in the following section:

```
<!-- Define a Coyote/JK2 AJP 1.3 Connector on port 8009 -->
<Connector
className="org.apache.coyote.tomcat4.CoyoteConnector"
           port="8009" minProcessors="5" maxProcessors="75"
           enableLookups="true" redirectPort="8443"
           acceptCount="10" debug="0" connectionTimeout="20000"
           useURIValidationHack="false"
protocolHandlerClassName="org.apache.jk.server.JkCoyoteHandler"
/>
```

   If you want to map additional Apache Tomcat applications to work with IIS, you must create additional URI mappings; for example, for the `/admin` application, the URI group must be configured as follows (in `workers2.properties`):

```
[uri:/admin/*]
info=Administration module
context=/admin
```

4 **In the Windows registry, create the following keys (if they do not already exist) for ISAPI connector values:**

- `HKEY_LOCAL_MACHINE→SOFTWARE→Apache Software Foundation`
- `HKEY_LOCAL_MACHINE→SOFTWARE→Apache Software Foundation→Jakarta Isapi Redirector`
- `HKEY_LOCAL_MACHINE→SOFTWARE→Apache Software Foundation→Jakarta Isapi Redirector→2.0`

5 **In HKEY_LOCAL_MACHINE→SOFTWARE→Apache Software Foundation→Jakarta Isapi Redirector→2.0, create a new String Values with the following data:**

Name	Value
extensionUri	/jakarta/isapi_redirector2.dll
logFile	TOMCAT_INSTALL\isapi\isapi.log
logLevel	Error
serverRoot	TOMCAT_INSTALL
workersFile	TOMCAT_INSTALL\isapi\workers2.properties

6 **Configure the connector with IIS.**

a. Select **Start→Control Panel→Administrative Tools→Internet Services Manager**.

b. Select the Web site name that you want to configure (for example, Default Web Site), then select **Action→New→Virtual Directory** to open the Virtual Directory Creation wizard.

c. Click **Next**.

d. Enter `jakarta` as the Directory Alias and click **Next**.

e. Enter `TOMCAT_INSTALL\isapi` as the path to the directory with contents and click **Next**.

f. On the Access Permission dialog box, select the **Read**, **Run Scripts**, and **Execute** options, click **Next**, then click **Finish**.

g. Select the Web site name, then select **Action→Properties**.

h. On the ISAPI Filters tab, click **Add**.

i. Enter `jakarta` as the Filter Name.

j. Select `TOMCAT_INSTALL\isapi\isapi_redirectory2.dll` as the Executable, then click **OK** to close the Filter Properties dialog box.

k. Click **OK** to close the Web site Properties dialog box.

Adding the wsmedia Virtual Directory

To add the `wsmedia` virtual directory:

1. **Start the Internet Information Services (IIS) Manager.**
2. **Expand Web Sites on the Navigation frame, select your Web site (for example, Default Web Site), then right-click and select Action→New→Virtual Directory.**
3. **Click Next in the Virtual Directory Creation wizard.**
4. **Enter `wsmedia` as the virtual directory alias, then click Next.**
5. **Enter (or browse to) the path to the *`INSTALL_HOME`*`\servlets\deployment\wsmedia` directory, then click Next.**
6. **Select Read for the virtual directory access permissions, then click Next.**
7. **Click Finish to close the Virtual Directory Creation wizard.**

Testing the Servlets

Test that the servlets run in the Web server.

To test the servlets:

➤ **In a Web browser, go to `http://`*`COMPUTER_NAME`*`:`*`IIS_Port`*`/Hyperion`.**

Configuring Sun Java System Web Server with Native Servlet Engine

This section explains how to manually configure the Sun Java System Web Server with its native servlet engine (formerly Sun ONE Web Server and iPlanet Web). It contains these topics:

- "Manual Configuration Steps for Sun Java System Web Server" on page 212
- "Precompiling Sun Java System Web Server" on page 217

Follow these instructions only if you need to configure the Sun ONE Web Server manually.

Note Hyperion Solutions supports Sun Java System Web Server with its native servlet engine only on those operating systems listed in the Support Matrix. While other configuration combinations are possible, they are not supported by Hyperion Solutions.

Manual Configuration Steps for Sun Java System Web Server

To configure Sun Java System Web Server with its native servlet engine manually, you must complete the tasks described in the following sections

- "Setting the JRE 1.4.2 Path" on page 213
- "Unpacking the Web Application Contents" on page 213
- "Deploying the Application" on page 214
- "Setting the Classpath" on page 215
- "Adding the wsmedia Virtual Directory" on page 215
- "Updating the ws.conf and Logging Parameters" on page 216
- "Replacing the English Language Files (Optional)" on page 216

Configuring JRE 1.4.x

Hyperion Performance Suite release 8.3 makes use of Java Components for Encryption (JCE), which is bundled along with the standard JRE 1.4.x installation. Sun Java System Web Server 6.1 works with JCE correctly; however, if you are using version 6.0, manual changes are required.

To manually configure Sun Java System Web Server 6.0 to work with JCE:

1. **Copy `JAVA_HOME/jre/lib/jce.jar` to `JAVA_HOME/jre/lib/ext`.**
2. **Modify `Jvm12.conf` to include the correct location of `jce.jar`, for example:**

   ```
   jvm.classpath=JAVA_HOME/jre/lib/ext/jce.jar:<...>
   ```

3. **Restart the Sun Java System Web Sever.**

Setting the JRE 1.4.2 Path

To set the path to the JRE 1.4.2:

1. **Start the Sun Java System Web Server and open the Administration Console using a URL of this format:**

   ```
   http://host:port/
   ```

2. **Select the Web Server Administration Server from the Server Manager drop-down list.**
3. **Click the Global Settings tab.**
4. **In the JRE Path text box, specify the path to the JRE 1.4.2 that you previously installed.**

Note Some Administration Console pages may be inaccessible because of the use of JRE 1.4.2 (which is required by the Hyperion servlet) instead of JRE 1.3.x.

Unpacking the Web Application Contents

To unpack the Web application contents:

➤ **Extract the `Brio.war` file from the Hyperion Performance Suite Servers CD Disk 3 `\misc\Brio.ear` archive to the root directory of the Web application; for example:**

```
/export/home/user/Brio
```

Deploying the Application

To deploy the application:

1. In the Administration Console, click the **Servers** tab.
2. Select **Manage Virtual Servers**.
3. Select a virtual server to manage from the **Manage Virtual Servers** drop-down list, then click **Manage**.
4. Click the Class Manager **Apply** button (at the top right).
5. Select the server name entry under **Tree View of the Server** to manage that server.
6. Select the **Web Applications** tab.
7. On the Web Applications page, click **Deploy Web Applications**.
8. Enter the following information:
   - **WAR File On**—Server/Local Server
   - **WAR File Path**—Location of the WAR file (`Brio.war`)

     This should be the root directory of the Web application (see Step ‰).
   - **Application URL**—/Hyperion (the context where your deployment is located)
   - **Installation Directory**—Location where you want to place the expanded deployment directory of the Web application

Note Before manually deploying the servlets, if a partial deployment has already occurred, use the `redeploy` command instead of the `deploy` command.

9. Click **OK** to deploy the application.

Setting the Classpath

After you deploy the application, you must add the JAR files in `/WEB-INF/lib` directory to the classpath. You can use the Server Manager to add the JAR files to the JVM classpath, or you can add the JAR files manually.

Note Make sure that IWS_SERVER_HOME environment variable is set to the Sun Java System Web Server installation directory that contains `startconsole`.

To update the classpath:

1. **In the Administration Console, click the button (at the top left) that bears the name of your managed server.**
2. **Select the Java tab, then click Configure JVM Attributes.**
3. **Add these items to the classpath:**
   - `tools.jar`—For JSP runtime compilation
   - Path to the Web applications `/classes` and `/lib` directories; for example:

     ```
     /usr/jdk141/lib/tools.jar:WEB_APPLICATION_DIRECTORY
     /WEB-INF/classes:WEB_APPLICATION_DIRECTORY/WEB-INF/lib
     ```
4. **Click the defaultclass button (at the top).**

Adding the wsmedia Virtual Directory

To add the `wsmedia` virtual directory:

1. **In the Administration Console, click the Content Management tab, then click Additional Document Directories.**
2. **Specify these parameters, then click OK:**
   - **URL Prefix**—`/wsmedia`
   - **Map To Directory**—Path to the `/wsmedia` directory
3. **Click the Apply button (at the top right).**

Updating the ws.conf and Logging Parameters

To update the `ws.conf` and logging parameters:

1. **In the *WEB_APPLICATION_DIRECTORY*\config\ws.conf file:**
   - Locate `WebClient.Internal.Redirect.RedirectPolicy=HttpRedirect` and change the value to `JavaScriptRedirect`.
   - Modify the `WebClient.Diagnostics.Log.LoggingService` and `WebClient.UserInterface.Login.RMServer` parameters for next values (GSM_computer_name:GSM_port_number).
   - Set the `servletLog4JConfig.xml` path to the directory in which the file is located.
2. **Open the *WEB_APPLICATION_DIRECTORY*\config\servletLog4JConfig.xml file and set the path to the appenders.xml and debugLoggers.xml files to the directory in which these files are located.**
3. **Open the *WEB_APPLICATION_DIRECTORY*\config\appenders.xml file and set the path for the log files to the directory in which this file is located.**
4. **Change the DEPLOYMENT_LIB and DEPLOYMENT_PATH variables in the /servlets/bin/servletconfig.bat file to point to the deployment directory:**

Replacing the English Language Files (Optional)

To replace the English language files:

1. **Copy the template language files.**

   For example, for French, copy *INSTALL_HOME*\servlets\deployment_templates\cctemplates_fr_FR to *INSTALL_HOME*\servlets\deployment\cctemplates.

2. **Copy the Zero administration language files.**

   For example, for French, copy these files from the Hyperion Performance Suite Servers CD Disk 3 \misc\zeroadmin directory to the *INSTALL_HOME*\servlets\deployment\zeroadmin\component directory:
   - `Brio.Insight.fr.cab`
   - `Brio.Insight.fr.xpi`

- `Brio.InsightNoHelp.fr.cab`
- `Brio.InsightNoHelp.fr.xpi`

3 **Copy the** `defaultdomain.cfg` **and** `licenses.txt` **from** *INSTALL_HOME*`\lib` **to** *INSTALL_HOME*`\servlets\deployment\WEB-INF\classes`.

Precompiling Sun Java System Web Server

Precompiling greatly improves the performance of the Hyperion Performance Suite servlets. Use this sample script to precompile Sun Java System Web Server.

iplanet6_precompile_solaris.sh

```
#!/bin/csh
# iplanet6_precompile_solaris.sh -- this file was generated by the Hyperion
installation program
# precompile Brio8 servlet for iPlanet/Solaris
setenv JAVA_HOME $E(JAVA_HOME)
setenv CLASSPATH $JAVA_HOME/lib/dt.jar:$JAVA_HOME/lib/tools.jar
setenv IWS_SERVER_HOME /lhome/dybe0/webservers/iPlanet
setenv BRIOHOME /lhome/dybe0/brio8
setenv URIBASE $BRIOHOME/servlets/deployment/WEB-INF/jsp
setenv SERVER_NAME https-sunburn.ohio.sqribe.com
setenv APP_NAME Brio
setenv APP_ROOT $BRIOHOME/servlets/deployment
setenv OUTPUT_DIR $IWS_SERVER_HOME/$SERVER_NAME/ClassCache/$SERVER_NAME/$APP_NAME
#echo Precompiling JSPs for $APP_NAME
$IWS_SERVER_HOME/bin/https/bin/jspc -genclass -v2 -uribase $URIBASE -d $OUTPUT_DIR
-webapp $APP_ROOT
```

Precompilation is an optional step. It has a big effect on performance, but for a development or test system you do not need to do it.

B Troubleshooting

Review the following troubleshooting topics if you encounter problems during the installation process:

- Using Log Files to Troubleshoot Problems During Installation
- Troubleshooting Problems During Installation
- Installation Error Messages
- Using Log Files to Troubleshoot Web Server Configuration
- Troubleshooting Problems with Web Servers/Servlet Engines
- Troubleshooting Service Startup

Using Log Files to Troubleshoot Problems During Installation

During installation, for all activities that go on while the progress bar is displayed, and for activities specific to a particular component, the `<INSTALL_HOME>\log\install.log` and `BrioPlatformInstall.log` file entries contain prefixes (see Table 10) that identify the activity and component.

If you have a problem with your Event Service, you can search the log for entries where `ES_` precedes the action name, for example:

```
installing Files (ES_ESnotification)
```

You can also search `CORE_`, because that applies to all services.

Table 10 Installation Log Prefixes

Prefix	Installation Activity/Component
CORE_	All services
AN_	Authentication Service
AZ_	Authorization Service
SM_	Session Manager Service
PUB_	Publisher Service
GSM_	Global Service Manager and Name Service
RM_	Repository Service
SB_	Service Broker
ES_	Job Service
DAS_	Data Access Service
BI_	Intelligence Service
CC_	Client Core (non-service items on the Servers CD)
SV_	Servlets
SDK_	Software Development Kit
JU_	Job utilities

Table 10 Installation Log Prefixes *(Continued)*

Prefix	Installation Activity/Component
RS _	Remote Service Configurator
LSC _	Local Service Configurator
BQ _	Designer/Explorer

In addition to the logs provided with Hyperion Performance Suite, you also can use the following InstallShield `is:log` and `is:javaconsole` commands for troubleshooting:

- `-is:log` *<file name>*—This option is useful for setup launchers that hide the Java console, because it logs the detailed information about the launcher's processing, including the actual Java commands that were used to start the Java program, to the specified file name. This includes all of the `stdout` and `stderr` messages from the Java process; for example:

  ```
  setup.exe -is:log myLogFile.txt
  ```

  or

  ```
  ./setup.sh -is:log "%HOME//ismp.log"
  ```

- `-is:javaconsole`—When you are running in GUI or Wizard mode, the `is:javaconsole` option makes a DOS box (Windows) or a console box (UNIX) pop up behind it, to show error messages and `stdout`. The box disappears when you click Finish or Cancel in the installer. For example:

  ```
  setup.exe -is:javaconsole
  ```

Troubleshooting Problems During Installation

Problem: The installer suddenly freezes on UNIX with no error message, before any prompts are displayed.

Solution: The installer may crash if you mistype a parameter. The most common error is omitting the dash or using any capital letters for the parameter `-is:tempdir`. This parameter must be specified in all lowercase letters, preceded by a dash.

Problem: The installer fails on HP-UX 11.11 with this error message:

```
ErrOut:
Assertion failed: so->so_hard_contents, file softobj.c, line 363
sh: 5046 Abort
```

Solution: This error usually occurs when the correct OS patch is not installed for a particular JVM. Try running your installer with a different JVM. A list of all required HP-UX Java patches is available at this Web address:

`http://www.hp.com/products1/unix/java/patches/index.html`

Problem: The installer prompts for `is:tempdir`.

Solution: The installer uses a large amount of temporary space during the installation process and requires that you specify a temporary directory for the installer. See "Satisfying Temporary Space Requirements" on page 50 for more information.

Problem: When installing from the Servers CD to a Windows 2000 Server system that has its display set to 16 Colors, the installation aborts after the InstallShield preparation dialog boxes.

Solution: Set the color settings properties for your laptop display to something other than 16 Colors, then rerun the Hyperion Performance Suite servers installer.

Problem: The installers for many other software packages (database clients, MKS, and so on) do not launch on computers where the `BIService.exe` is running.

Solution: Stop the BI Service before running any software installation programs.

Problem: The installer displays a “no temp space” error.

Solution: The installer uses a large amount of temporary space during the installation process. As a result, your temporary space may be low. You need approximately 1 GB of temporary space.

To redirect temporary space for the installation to somewhere else, use one of these settings:

- **Windows**

  Change the TEMP environment variable (**Start→Setting→Control Panel→System**, **Advanced** tab, **Environment Variables** button.)

  Alternatively, you can use this command:

  ```
  setupWinPlatform.exe -is:tempdir <directory_name>
  ```

  where *`directory_name`* is a directory on your system with space.

- **UNIX**

  ```
  setupSolarisPlatform.bin -is:tempdir/tmp
  ```

  When installing on UNIX systems, always use `-is:tempdir xxx` where `xxx` is a temporary directory with at least 1 GB of space available. InstallShield also uses `/var` directory for some temporary files, so make sure that `/var` is not full.

  When installing using a remote UNIX session, make sure you set up display to the terminal where you want the window the be displayed. Otherwise, the installation will seem to freeze because the display is really on the UNIX host computer waiting for input.

Problem: The installer on Solaris fails during the SQR Job Service connect string test.

Solution: You must have SQRDIR set to `/SQR_HOME/bin` before installing Hyperion Performance Suite. Complete the installation, then use the Service Configurator to set up Hyperion SQR.

Problem: The installer displays a "Test Connection Failed" error after you enter the user name and password for an IBM DB2 row-level security database. The log file displays this error message:

```
java.sql.SQLException: [BrioSoftware][DB2 JDBC Driver]Failed to
begin the bind process
```

Solution: Make sure that you granted the DB2 user permission to create packages.

Problem: An SQL error message is listed in the installation summary.

Solution: Check in the *INSTALL_HOME*`/log/installSql.log` file. Search for *Excp* (short for exception). Following are some common errors in this file:

- **Unique Constraint errors**—These are generally harmless. The installer may not check for a row before inserting it (especially during reinstallations), and the database integrity constraints prevent it from being re-entered.
- **Tablespace Full errors**—Have your DBA increase the amount of space available in the database, then rerun the installer (uninstall) to remove the partially created repository and *INSTALL_HOME*.
- **Permissions errors**—The Hyperion Performance Suite database user does not have permission to create tables or indexes or both. Have your DBA give the user the required permissions, then rerun the installer (uninstall) to remove the partially created repository and *INSTALL_HOME*.

Problem: SQR does not connect.

Solution: This does not stop the installer, but SQR jobs will not work until the problem is resolved. Search the `install.log` file for *VerifySqr*. The installer logs a command there similar to this one:

```
VerifySQR failed, command line was: /lhome/bi/username/sqr/sqr81rc4/ora/bin/sqr
/tmp/connectivityTest.sqr ****/****@mytnsnamesentry /tmp/
```

Run this command from a Command prompt to determine the source of the error. Correct the problem in the environment (for example, SQRDIR not set, bad `SQR.ini` parameter, client database software not available, client database variables not set). You may also need to set variables in the Remote Service Configurator (Job Service/SQR tab) to run SQR jobs.

Problem: Unable to deploy servlets.

Solution: You can manually deploy servlets (see Appendix A, "Manually Setting Up the Hyperion Performance Suite Servlets"), or try this solution:

Search the `install.log` file for CommandLine. The installer logs a command line similar to this one:

```
System Property 'SVdeployjRunCommandLine' set to
'/lhome/bi/jgiglio0/231/jre/jre131/bin/java -classpath
.:/lhome/bi/jgiglio0/231/JRun/lib/jrun.jar  allaire.jrun.tools.WarDeploy -deploy
-config=/lhome/bi/jgiglio0/231/install/deployment.txt'
```

Run this command from a Command prompt to determine the source of the error. Correct the error, then rerun the installer to redeploy the servlets.

Problem: Unable to precompile servlets.

Solution: You can manually precompile the servlets (see Appendix A, "Manually Setting Up the Hyperion Performance Suite Servlets"), or try this:

Search the `install.log` file for Precompile. The installer logs a command line, or the name of a script, which precompiles the classes.

Run this command (or script) from a command prompt to determine the source of the error. Correct the error, then rerun the script or command. There is no need to reinstall.

Problem: The installer stops after a dialog box where you need to select a directory.

Solution: The installer cannot find a required file in the specified directory.

Check the last line of the `install.log` or the pop-up informational message for a relative path (such as `bin/orant.dll`) to a file that must be in the selected directory. For example, database drivers and Web server configuration files are requested in this manner.

Enter the directory that contains the file. If the file is not part of your Web server or database installation, check the *Hyperion Performance Suite Installation Guide* to ensure that the release is supported or to learn where to obtain the file.

Problem: Failed database connection.

Solution: The installer logs the connection error in the `install.log` file.

Check the last line of `install.log` to see the returned database error. A ClassNotFound exception indicates that the database driver in `c:\overrides` is not compatible with the database, or the driver is missing. Database-specific errors should be checked against the database documentation.

Problem: Connectivity to IBM DB2 on AIX5.2 fails.

Solution: Connectivity to IBM DB2 on AIX5.2 may fail if the installation is done on a remotely mounted location (a directory location mounted over the network). To resolve this issue, copy the *`INSTALL_HOME`*`/BrioPlatform/connectivity` recursively to a local directory like `/opt/odbc/connectivity` (a directory on a local hard disk), set the LIBPATH to point to `/opt/odbc/connectivity/lib`, then restart DAS.

Problem: The installer gives you an "installation failed" message.

Solution: The `install.log` contains a failure point and exception trace.

Search the `install.log` backwards for installing Files. These messages occur throughout the product installation progress bar. At some point, the messages in a failed installation change from installing to uninstalling; this is the failure point. For example, if you click No to all, the installer may treat this as a failure, or if there is a problem with disk space or permissions. Correct the problem and reinstall. You may want to drop any partly created repository tables.

Problem: During a reinstallation, errors (such as missing options or dialog boxes) occur before the installation progress bar is displayed.

Solution: Search `BrioPlatformInstall.log` (found in the *userhome* directory).

SUITEinstallType should be set to REINSTALL. SUITEBrioHome should be set to the home directory you selected. Ensure that the products licensed and the existing *INSTALL_HOMES* are accurate.

The installer relies on the `InstallHistory.xml` file and the `vpd.properties` file to reset itself for a reinstall. If either of these files is missing or corrupted, problems can occur.

Problem: Files are missing during an international installation.

Solution: Search the `install.log` for language settings.

Look for SUITEinstallerLanguageSelection. Ensure that the target language set corresponds to one of these supported languages:

Chinese (Simplified)	Italian
Chinese (Traditional)	Japanese
English	Korean
French	Portuguese (Brazilian)
German	Spanish

Problem: Hyperion Performance Suite help files are missing.

Solution: Servlet help is added to the folders using a batch/shell script in the *INSTALL_HOME* installation folder. If help is missing or improperly named (wrong subfolder, for example), check the script `unzipWebComponentHelp.bat` (or `unzipWebComponentHelp.sh` on UNIX) for troubleshooting. You can rerun the script manually and correct any problems.

Problem: After installation, a "JVM not found" error is displayed.

Solution: The directory referenced by `is:tempdir` is nearly full. Increase the amount of space allocated to this directory.

Problem: During installation on Solaris, the installer fails after precompiling the servlets. The JRun Installation Error.log gives this error message:

```
Installing Servlet/JSP components ... cp:  illegal option -- f
```

Solution: For installations on Solaris, ensure that the `cp -f` flag is enabled.

Problem: The following message is presented when installing Hyperion Performance Suite with Oracle:

```
-17 14:08:30 SQL Excp: java.lang.Exception: java.sql.SQLException:
[BrioSoftware][Oracle JDBC Driver][Oracle]ORA-04031: unable to allocate 4160 bytes
of shared memory ("shared pool","unknown object","sga heap(1,0)","state objects")
```

Solution: This message is displayed when the shared pool size on the database is set to too small a number. The sizing of the shared pool is important, because it affects the ability of the installer to create all of the necessary repository tables. Hyperion recommends that a shared pool size of 20 MB be used during the installation.

Installation Error Messages

Table 11 lists error messages that the installer may display during the installation process.

Table 11 Installer Error Messages

Error Number/Type	Error Message	Explanation
SUITE01000	CD incomplete, data directory missing, database cannot be loaded.	The installer did not find the necessary files or directories on the product CD. Contact Customer Support.
SUITE01001	License Key Validation Error	Check that you entered the correct validation key.
AN1010	Native System Create ERROR	A postinstallation creation of a native authentication system failed.
AN1020	External System Create ERROR	A postinstallation creation of an external authentication system failed.
AN1040	External driver default properties ERROR	Failure occurred in retrieving or updating external driver properties.
AN1050	External driver properties ERROR	Failure occurred in retrieving or updating external driver properties.
AN1060	Field validation ERROR: Field must not be empty	Check that all required fields have entered values.
AN1070	Authentication System validation Error: System already exists	An authentication system already exists.
AN1100	Failure in creating external system, see log for details.	A postinstallation creation of an external authentication system failed.
SUITE1000	Database Check ERROR	A failure occurred while querying the database for information.
SUITE1020	XML file delete	The installer encountered an error while updating the XML configurator files.
SUITE1040	Field Validation Error	Check that you entered valid values in each field.
SUITE1050	must not be empty	Enter the requested value for all required fields.
SUITE1060	Field must be 5 chars or more	Make sure that the field contains five or more characters.

Table 11 Installer Error Messages *(Continued)*

Error Number/Type	Error Message	Explanation
SUITE1070	The referenced service is already installed, are you sure you want to proceed?	Installing a service twice into your *INSTALL_HOME* directory is not recommended.
SUITE1080	Database tables exist from a prior installation, are you sure you want to proceed?	The installer detected the presence of database tables. A new installation should have a clean database.
SUITE1090	File Check Error	The installer check for a file on the system failed.
SUITE1100	File already exists.	The already exists on the system.
SUITE1110	File doesn't exist.	The files does not exist on the system.
SUITE1120	Tables already exist.	Database tables already exist on the system.
SUITE1130	Tables do not exist.	Database tables do not already exist on the system.
SUITE2020	Service validation Error: Service already exists.	Do not give a new service the same name as an existing service.
SUITE2120	Service name must be entered.	Enter a service name in the dialog text box.
SUITE2130	Non-replicable service cannot be installed twice.	The installer found the current service when checking for existing non-replicable services.
SUITE2140	Service cannot be upgraded or patched if it is not installed already.	The service cannot upgrade or patch a service that is not already installed.
SUITE2190	Cannot install this service without the Name Server being installed prior or during this installa-tion	GSM/Name Service must be installed before or at the same time as other services.
SUITE2200	qryPortal error	This is a query database error; the erring SQL will be logged.

Using Log Files to Troubleshoot Web Server Configuration

Errors and output that occurred during JRun deployment can be found in the *INSTALL_HOME*/`log` directory, in the files *Webserver*`Update.log` and *Webserver*`Update Error.log`.

The main log file for the installation contains the command line used to run the installer; it uses the following format:

```
Finished running Web server update command: deployment command
line
```

Troubleshooting Problems with Web Servers/Servlet Engines

This section lists some of the problems you may encounter related to Web servers and servlet engines.

Problem: The installer fails to precompile JRun.

Solution: Manually precompile JRun:

1. Check the `install.log` for this line:

   ```
   Precompile JRun servlets complete using script
   ```

   This line gives you the name of the precompilation script that you need to run.

2. Check the `\log` directory for the files `jrunPrecompile.log` and `jrunPrecompileError.log`.

3. Correct the errors in the logs, then manually run the script.

   For examples of the precompilation scripts, see"Precompiling JRun 3.1" on page 197.

Problem: JRun deployment fails.

Solution: If you get a message that deployment failed on JRun, you can deploy it manually. Look in the installation log for this line:

```
SVdeployjRunCommandLine
```

The setting resembles this example:

```
d:\PROGRA~1\brio8\jre\jre131\bin\java.exe -cp .;d:\program
files\brio8\JRun\lib\jrun.jar allaire.jrun.tools.WarDeploy -deploy
deploy.war.path=f:\stagedCD\disk1\misc\WebComponents.war
deploy.server.name=default deploy.webapp.name=Brio deploy.context.path=/brio
deploy.webapp.rootdir=F:\Installer\ISMP Projects\Unit
Testing\Servlet\disk1\misc\WebComponents.war deploy.jrun.rootdir=d:\program
files\brio8\JRun
```

Copy the command line and run it from a Command prompt. Make sure there are quotation marks around any paths with spaces in them (or, in Windows, use the short DOS path). This should give you an error message that you can troubleshoot.

Note Before manually deploying the servlets, if a partial deployment has already occurred, use the `redeploy` command instead of the `deploy` command.

Problem: You want to change the log levels for JRun.

Solution: These JRun log files are created in the `\JRun\Log` directory:

- `default-out.log`
- `default-event.log`
- `default-err.log`
- `jrunInstallStdout.log`

You can use the JRun Management Console to change the JRun log levels, or you can change the `local.properties` file; for example:

```
logging.loglevel=debug.
```

Problem: Errors occurred while installing JRun on UNIX.

Solution: If any errors occur during JRun installation on UNIX systems, the installer creates a file that contains the output from the silent installer:

INSTALL_HOME/log/jrunInstallStdout.log

Problem: You receive JRun messages after removing JRun from your system.

Solution: If JRun is not properly removed from your system, you may get messages because some services or registry entries point to the wrong JRUN_HOME.

Also, sometime the EJB (Enterprise Java Bean) properties are not fully removed. Check the JRun logs for lines resembling these:

```
Java Virtual Machine: build JDK-1.2.2_011, native threads, symcjit from Sun
Microsystems Inc.
JRun 3.1 3.1.15506 Starting admin...
Current Locale: en_GB
Loading scheduler
Loading logging
Warning: d:\program files\brio8\JRun\servers\admin\deploy\deploy.properties not
found
Warning: d:\program files\brio8\JRun\servers\default\deploy\deploy.properties not
found
Warning: D:\Program Files\Brio8\JRun\servers\admin\lib\ejipt.properties not found
Error: Ejb directory .\servers\default not found
```

Problem: The installer fails to precompile the Sun Java System Web Server (iPlanet) Web components.

Solution: There are two possible solutions:

- Check *INSTALL_HOME*/log/iplanetDeployError.log for errors. Permission errors and disk space errors are reported here.
- Check *INSTALL_HOME*/log/install.log If it contains an "Exception in thread "main" java.lang.OutOfMemoryError," increase the heap size for the precompile command. Set the script `$IWS_SERVER_HOME/bin/https/bin/jspc` to use a max heap size of 1 GB. For example, set the variable `IWS_JAVA=${IWS_JRE}/bin/java -Xmx1024m`.

Problem: The installer fails to precompile Sun Java System Web Server (iPlanet).

Solution: Manually precompile Sun Java System Web Server:

1. Check the installation log for this line:

   `Precompile iPlanet servlets complete using` *`script`*

   Where *script* is the name of the precompilation script that you need to run.

2. Check the `\log` directory for the files `iplanetPrecompile.log` and `iplanetPrecompileError.log`.

3. Correct the errors in the logs, then manually run the script.

   For examples of the precompilation scripts, see"Precompiling Sun Java System Web Server" on page 217.

Problem: The installer fails to precompile WebLogic.

Solution: Precompilation is handled by an addition to `weblogic.xml` (for version 6.1 this is fully supported by the installer) or `web.xml` (for 6.0, this requires manual steps). For more information, see "Configuring BEA WebLogic 6.x with WebLogic SP3 or Later" on page 162.

If the precompilation fails, correct the problem in the Web server logs and manually redeploy.

Problem: The following exception occurs after installing the servlets on Solaris with WebLogic:

```
org.omg.CORBA.INITIALIZE:
Ensure that you have either (1) Set IT_DOMAIN_NAME to your <domain_name> and you
are using it_java or
(2) You have passed the '-ORBdomain_name <domain_name>' to the process.
Unable to open ORB configuration file "/default-domain.cfg" for reading, check
file existence (on CLASSPATH) and read
permissions minor code: 1230242048 completed: No
at
com.iona.corba.art.configuration.art_config_file.ConfigFile.readConfigSource(Unkn
own Source)
at
com.iona.corba.art.configuration.art_config_file.ConfigFile.readConfigSource(Unkn
own Source)
at
com.iona.corba.art.configuration.art_config_file.ArtConfigFilePlugInImpl.get_repo
sitory(Unknown Source)
at com.iona.corba.art.artimpl.ORBDelegate.set_parameters(Unknown Source)
at com.iona.corba.art.artimpl.ORBImpl.set_parameters(Unknown Source)
at org.omg.CORBA.ORB.init(ORB.java:324)
at com.brio.one.utils.CorbaClient.<init>(Unknown Source)
at com.brio.one.web.launcher.util.LauncherUtil.<init>(Unknown Source)
at com.sqribe.WS.WSBaseServlet.loadConfiguration(Unknown Source)
at com.sqribe.WS.WSBaseServlet.init(Unknown Source)
at
weblogic.servlet.internal.ServletStubImpl.createServlet(ServletStubImpl.java:731)
at
weblogic.servlet.internal.ServletStubImpl.createInstances(ServletStubImpl.java:65
8)
at
weblogic.servlet.internal.ServletStubImpl.prepareServlet(ServletStubImpl.java:603
)
at weblogic.servlet.internal.ServletStubImpl.getServlet(ServletStubImpl.java:370)
at
weblogic.servlet.internal.ServletStubImpl.invokeServlet(ServletStubImpl.java:240)
at
weblogic.servlet.internal.ServletStubImpl.invokeServlet(ServletStubImpl.java:198)
at
weblogic.servlet.internal.WebAppServletContext.invokeServlet(WebAppServletContext
.java:2637)
at
weblogic.servlet.internal.ServletRequestImpl.execute(ServletRequestImpl.java:2359
)
at weblogic.kernel.ExecuteThread.execute(ExecuteThread.java:139)
at weblogic.kernel.ExecuteThread.run(ExecuteThread.java:120)
```

Solution: Double-check the manual configuration steps in your `startWebLogic.sh` file. Check the `install.log` file (search for `SVwlogicManualConfigInstructions`) and verify that you performed the steps listed.

Problem: After you deploy WebLogic, a “404:page not found” error is displayed.

Solution: Follow these steps:

1. Log on to the WebLogic Administration Console.
2. Find the Hyperion Performance Suite application.
3. Select **Edit Web Application Descriptor**.
4. Drill down to **Servlet Mappings**.

   The servlet mappings should be /administration/*, /browse/* …
5. If the “/*” characters are missing at the end of the mapping, add them.
6. Apply the changes and log on to the Hyperion Performance Suite applications.

Problem: The installer freezes while deploying servlets to WebLogic on AIX systems.

Solution: WebLogic has run out of resources while trying to precompile the servlets. To validate that this is the problem, look in the `/logs` subdirectory of the `/config/domain` directory. The log should show precompilation steps, and the date stamp will show they stopped processing.

Follow these steps to correct the problem:

1. Terminate the installation process.
2. Create an empty **`precompile.txt`** file in your `/tmp/overrides` directory.
3. Rerun the installer with the `precompile.txt` file in place.

Note that without precompilation, there will be a significant performance hit the first time the servlets are accessed and anytime the Web server is restarted. This is because the Web server must compile the JSPs with each restart.

Problem: When running in a secure environment, manual deployment of the servlets on WebLogic fails.

Solution: Follow these steps to deploy WebLogic in a secure environment:

1. Create an `applications` directory within WebLogic where Hyperion Performance Suite will be deployed.

   This must be done during WebLogic installation or from the WebLogic console.

2. Decompress the `Brio.ear` file into a temporary directory.

3. Decompress the resulting `Brio.war` file into a temporary `Brio` directory.

4. Move or copy the resulting temporary `Brio` directory tree to the WebLogic `.../applications` directory created by WebLogic for Hyperion Performance Suite.

5. Modify `..../Brio/config/ws.conf`:

   `WebClient.Diagnostics.Log.Configuration=file://localhost/C:/CCV/ced0_apps/fnd_commonclient/CommonClient/config/servletLog4JConfig.xml` must point to the deployment tree `...../Brio/config/servletLog4JConfig.xml`

6. Modify `..../Brio/config/servletLog4JConfig.xml`:

   - `file:${log4j_config_directory}\appenders.xml` must point to the deployment tree `..../Brio/config/appenders.xml`
   - `file:${log4j_config_directory}\debugLoggers.xml` must point to the deployment tree `..../Brio/config/debugLoggers.xml`

7. Modify `..../Brio/config/appenders.xml`:

   Change the paths to the log files for all occurrences of `${directory}/log/${name}`. Substitute the directory path such as `/apps01/BEA/user_projects/brio_apps1_1/applications/Brio/logs/` or your preference of location.

8. Download and FTP the `sun xml` DTDs to the installation host

   Use these definitions or a later version:

   `http://java.sun.com/j2ee/dtds/web-app_2_2.dtd`

9. Modify `..../Brio/WEB-INF/web.xml` to point to the downloaded DTDs:

```
 PUBLIC "-//Sun Microsystems, Inc.//DTD Web Application
2.2//EN"
```

   Change the version to match the download if need be:

   Change `http://java.sun.com/j2ee/dtds/web-app_2_2.dtd` to `file://apps01/BEA/user_projects/brio_apps1_1/applications/Brio/web-app_2_2.dtd`

10. Download and FTP the `weblogic.xml` DTDs to the installation host:

   Use these definitions or a later version:

```
http://www.bea.com/servers/wls600/dtd/weblogic-web-jar.dtd
```

11. Modify `..../Brio/WEB-INF/weblogic.xml` to point to the downloaded DTDs:

   Change `http://www.bea.com/servers/wls600/dtd/weblogic-web-jar.dtd` to `file://apps01/BEA/user_projects/brio_apps1_1/applications/Brio/weblogic-web-jar.dtd`.

12. For WebLogic 8.1.0 and 8.1.1, retrieve patch `CR136204_810sp1.zip` from Hyperion:

   Copy `CR136204_810sp1` to `/apps01/BEA/user_projects/brio_apps1_1/applications/Brio/WEB-INF/lib`.

13. Change the WebLogic classpath in the WebLogic startup script (for example, `startWebLogic.sh`):

```
CLASSPATH="/apps01/CR136204_810sp1.jar:${WEBLOGIC_CLASSPATH}:${
JAVA_HOME}/jre/lib/rt.jar:${WL_HOME}/server/lib/webservices.jar
:${CLASSPAH}"
```

14. Navigate within the WebLogic application console to Web Application Modules and deploy Brio.

Problem: The installer fails to precompile the servlets for WebSphere.

Solution: Use one of these methods to run the precompilation script manually:

- If your servlets were deployed automatically by the installer, use this command:

  ```
  JspBatchCompiler.bat -enterpriseApp Brio -webModule Brio.war
  ```

- If you deployed your servlets manually, check the names you assigned to the Enterprise Application and Web Module. Use the preceding command for automatic deployments with the enterprise application and Web module names used in the manual deployment.

Problem: The installer fails to deploy the servlets on WebSphere.

Solution: During servlet deployment on WebSphere, autodeployment may fail if the WebSphere administrator application (`wsadmin.sh`) is set to use less than 512 MB of memory. You must change the `wsadmin.sh` script (the last line that executes `com.ibm.ws.bootstrap.WSLauncher`) and add `-Xmx512M` for Java there.

Problem: After you deploy the servlets on WebSphere version 4.0.6, the Administration module does not work. A ClassCastException or JSPG0103E: Unable to open taglibrary error is displayed at runtime.

Solution: These errors occur because the version of xerces in the servlet differs from the WebSphere version of xerces. If you received either of these error messages, you need to download and install WebSphere package PQ76047 from `http://www-1.ibm.com/support/docview.wss?rs=180&context=SSEQTP&q1=JSPG0103E&uid=swg24005064&loc=en_US&cs=utf-8&lang=en+en`.

Problem: You get this error message:

```
Please correct the following errors and click 'deploy' again:·
unexpected EOF
```

Solution: The WAR file is incomplete. This can happen if there are FTP errors or interruptions during the transfer of this large file.

Problem: Your servlets are working but they have codes such as LOGIN_BUTTON instead of proper messages.

Solution: The deployment was incomplete and is lacking the files in *INSTALL_HOME*`/servlets/deployment/WEB-INF/classes/msgs`, or the folder is missing from the servlet classpath. If the files exist, add the directory to the classpath. If the files do not exist, redeploy the application.

Problem: You want to redeploy the application.

Solution: Look in your `install.log` (*INSTALL_HOME*`/log/install.log`) and search for `SVdeployjRunCommandLine`. Copy the value to the command prompt. If the original value uses `-deploy`, you should change this value to `-redeploy` to force update of the servlets.

Problem: Images are missing from my Hyperion Performance Suite system.

Solution: Missing images indicate that the `\wsmedia` directory is not mapped. If you are using the IBM HTTP server and it is on a separate computer from WebSphere, you must map the `\wsmedia` directory manually.

Troubleshooting Service Startup

Read this section for help with starting the Hyperion Performance Suite services.

Problem: When you start from Control Panel→Services, the BrioPlatform service fails to start up; however, the service starts from the batch script.

Solution: The services are starting up, but they take longer than two minutes to do so. At this point, Windows interprets this to mean that the service has failed and throws an error. This problem is generally an underpowered computer that requires more RAM or a faster processor to support the services. For example, to run all the services on a single computer, that computer requires at least a 2 GHz processor and 1 GB of RAM.

Problem: The Hyperion Performance Suite services do not start after installation is complete.

Solution: Use one of these two options to obtain more troubleshooting information:

- In the `server.xml` file, change the log level from "warning" to "info."

```
<SETUP>
<key>19564513</key>
<filecontrol>6c004ca12d24186868d6d5a84926b649e8b8be66ff4
bbab9b17437770df3adbb</filecontrol>
<tmpfilelocation>C:\Program
Files\Brio\Brio8\BrioPlatform\tmp</tmpfilelocation>
<FILEID>Host PAndrews</FILEID>
    <logging>
      <redirectoutput>true</redirectoutput>
      <xmllogging>true</xmllogging>
      <level>warning</level>
    </logging>
```

Change "warning" to "info"

You also can use the Remote Service Configurator (RSC) to change the log level. See the *Hyperion Foundation Administrator's Guide* for information on using the RSC.

- In the start script (`startCommonServices.sh` or `startCommonServices.bat`), set the start super service options, for example:

  ```
  -Dprint_query=true
  ```

  You may also want to increase the heap size settings (BP_MIN_HEAP and BP_MAX_HEAP) in these start scripts.

Look in these files for additional information:

- `stdout_console.log`—If you find errors in this log, check the log for the specified service.
- `BI startup.log` and `DAS startup.log`—Startup logs for BI and DAS services.

Problem: The BI and DAS services will not start.

Solution: The BI and the DAS services cannot communicate with the GSM if the hostname (of the Foundation Services GSM) begins with numerals. Verify that the GSM name does not begin with numerals. If it does, rename it.

Problem: The BI service is slow to respond.

Solution: When you connect to a machine to start the BI Service on Windows, make sure the color property setting for the display is 16 bits or higher. If the color property setting is less than 16 bits, users may encounter extremely long response times when opening Chart sections of BQY documents with the Zero-footprint Client. This is an important pre-startup step, especially when starting the services remotely (for example using VNC, Terminal Services, Remote Administrator or Timbuktu, and so on), because many remote administration clients connect with only 8-bit colors by default.

Index

A

access, directory, blocking, 107
accounts, user, creating, 37
adding components or services, 96
aliases, database, 67, 83
Apache
  configuration requirements prior to installation, 49
  configuring DSO, 49, 152
  servlet configuration, 148
APIs, 34
applications, redeploying, 240
audience for this guide, x
authentication systems, external, 31
axbqs32.dll, 136

B

BEA WebLogic, precompilation errors, 234
beans, enterprise Java, 233
BILauncher.dll, 136
BP_MAX_HEAP, 242
BP_MIN_HEAP, 242
bqformat.ini, 136
bqmeta0.ini, 136
brio.rc registry, 119 – 120
BrioPlatformInstall.log, 220
brioqplg.ini, 136
briottbl.txt, 136
browsers, Web, requirements, 30
built-in Web server, JRun, 30

C

cache requirements, 28
cache space, 28
case sensitivity, 36, 38
CD-ROM drive requirements, 28
ClassCastException error, 239
ClassNotFound errors, 226
classpaths, configuring, 200
client tools installation, 131
color setting properties, during installation, 222
command line installer, 51
commercial Web servers
  configuring servlets, 145
  servlet engine configuration settings, 146
commercial Web servers, servlet engine configuration settings, 146
communications protocol requirements, 28
components
  adding, 96
  core installable unit software, 123
  distributed, 32
  JRE, 124
  Orbix, 124
  replicated, 32
  services, 124
  suite, 21
  support, 125

config.dat, 111
configuration procedures, required, 35
configuration settings, servlet engine, 146
configuration, servlet, verifying, 146
Configurator, Remote Service, 242
configuring
  Apache
    JRun, 148
    pre-installation, 49
    servlets on, 148
  classpaths, 200
  database software for the Repository, 37
  DSO, 49
  IBM DB2, 38
  IIS
    pre-installation, 49
  Microsoft IIS with JRun, 148
  Microsoft SQL Server, 42
  Oracle, 44
  servlets on a commercial Web server, 145
  Sun Java System Web Server
    pre-installation, 50
connect strings, for Job Service, 67, 83
connection errors
  database, 226
  SQR, 224
connection failures, SQR, 87
connection requirements, TCP/IP, 28
connections, ODBC, creating, 46
constraint errors, unique, 224
consulting services, xv
conventions, typographic, xiv
core installable unit software components, 123
creating
  database schema for row-level security tables, 46 – 47
  ODBC Connections, 46
  user accounts, 37
cross-platform registry, 119
custom installation, servers, 75

D

data source names, 67, 83
database aliases, 67, 83
database connection errors, 226
database logs, 123
database nomenclature, 67, 83
databases
  data source, 29
  repository
    size recommendations, 29
DB2, connection errors, 226
DB2. *See* IBM DB2., 38
DB2COMM, 41
DB2DIR, 41
DB2INSTANCE, 41
-Dbrio.content.path, 108
default installation directory, 66, 80
default installation directory, client tools, 133
default-err.log, 232
default-event.log, 232
default-out.log, 232
deploying
  applications using the JRun management console, 199
  servlets onto JRun, 196
deployment directory, blocking access to, 107
deployment errors, servlet, 225
deployment failures, WebSphere, 239
details, server installation, 101
directories
  default installation, 66, 80, 133
  temporary, 50, 223
directory access, blocking, 107
directory mappings, virtual, 126
directory structure
  client installation, 135
  server installation, 110

disk space requirements, 28
distributed components, 32
distributed system upgrade, 58
DMZ support, firewall and, 33
documents
  feedback, xv
  ordering print documents, xiii
  structure of, x
documents, accessing
  documents, accessing
    online help, xii
  Hyperion Download Center, xii – xiii
  Hyperion Solutions Web site, xii
drive requirements, CD-ROM, 28
drivers, components, 124
dynamic shared objects (DSO), 49, 152

E

education services, xv
enterprise java beans, 233
environment requirements, Web server, 30
environment variables
  IWS_SERVER_HOME, 30, 50, 215
  SQR, 36
  Sun Java System Web Server settings, 30, 50, 215
  TEMP, 50, 223
  UNIX, 41, 45
  Windows, 41, 45
EOF error, unexpected, 239
error messages, 45, 228
errors
  ClassCastException, 239
  color settings properties, 222
  database connection, 226
  deployment
    WebSphere, 239
  file missing, 225
  FTP, 239
  HP-UX, 222
errors *(continued)*
  identifying
    installation, 220
  images, missing, 240
  installation
    failures, 223 – 226
    identifying, 220
    message numbers, 229
  internationalization, 227 – 228
  JRun, 232 – 233
  JSPG0103E, 239
  language setting, 227 – 228
  language settings, 227 – 228
  LOGIN_BUTTON, 240
  missing file, 225
  no temp space, 223
  permissions, 224
  precompilation
    iPlanet, 234
    JRun, 231
    WebLogic, 234
    WebSphere, 236, 239
  reinstallation, 227
  servlets
    deployment, 225
    precompilation, 225
  Solaris, 228
  SQL, 224
  SQR connection, 224
  tablespace full, 224
  temporary space, 223
  third-part installers, problems launching, 222
  unexpected EOF, 239
  unique constraint, 224
  UNIX, 222
  WebLogic, 235
  WebSphere, 239
  Windows 2000 Server, 222
  wsmedia, 240

Exceed, 51
exceptions, ClassNotFound, 226
express installation option, 62
express installation, servers, 61
external authentication systems, 31

F

failures, installation, 223 – 224, 226
file missing errors, 225
file structures, WAR-compliant, 126
files, installed with Hyperion Intelligence Client, 136
firewalls
  DMZ support and, 33
  Web server installation and, 48
folders, Program Start, 135
fonts, on Unix, 52
FTP errors, 239

G

guidelines, pre-installation, 36

H

hardware requirements, 28
heap size settings, 242
help files, scripts, 151, 161, 165, 173, 179, 186, 190, 207
host name, 68, 83
hosts files, updating, 35
HP-UX, installation errors, 222
HTTP servers, servlets and, 144
Hyperion Consulting Services, xv
Hyperion Download Center, accessing documents, xiii
Hyperion Education Services, xv
Hyperion Hub, 31
Hyperion Intelligence Client, files installed with, 136
Hyperion Intelligence Client, registry entries, 137
Hyperion Performance Suite
  before installing, 27
  components
    distributing, 32
    replicating, 32
  directory structure, 110
  increasing performance of, 108
  overview, 17 – 18
  reinstalling, 97
  services, verifying installation, 102
  servlets
    running as standalone HTTP server, 144
    setting up, 143
  servlets, running as standalone HTTP server, 144
  uninstalling, 127
Hyperion product information, xv
Hyperion Solutions Web Site, accessing documents, xii
Hyperion SQR, configuring, 63, 77
Hyperion support, xv
Hyperion Technical Support, xv
hypsuite, 37

I

IBM DB2
  configuring, 38
  connection errors, 226
  required information, 63, 77
  troubleshooting, 38
  upgrade requirements, 59
identifying installation errors, 220
IIS, configuration requirements prior to installation, 49
images, missing, 240
increasing performance, 71, 88
information, required, 63, 77
  for custom installation, 77
  for express installation, 63
install.log, 73, 92, 220, 240

INSTALL_HOME, defined, 66, 80
installable unit software components, core, 123
installation
  available types, 66, 80, 132 – 133
  client tools, 131
  problems, troubleshooting, 220
  UNIX, 51
  verification, services, 102
installation details
  client tools, 135
  server, 101
installation directory, default, 66, 80, 133
installation errors, 229
  identifying, 220
  installer failures, 223 – 224, 226
installation options, express, 62
installations, single-computer, 62
installer, command line, 51
InstallHistory.xml, 55, 65, 79, 95, 227
installing
  Hyperion Performance Suite
    before, 27
    client tools, 132
    custom, 78
    express, 64
  JRun, manually, 193
  SQR, 47
  Web servers, 48
Intelligence Client executable, 136
internationalization errors, 227 – 228
iPlanet. *See* Sun Java System Web Server
iplanet6_precompile_solaris.sh, 217
-is
  javaconsole, 221
  log, 221
is
  javaconsole, 221
  log, 221
is:tempdir, 222 – 223
IWS_SERVER_HOME environment variable, 30, 50, 215

J

java beans, enterprise, 233
JDBC driver, and IBM DB2, 41
JRE, components, 124
JRun
  configuration
    on Apache, 148
  deployment failures, 232
  errors, 232 – 233
  installation information required for Windows, 195
  installing, manually, 193
  log files, 232
  precompilation errors, 231
  UNIX installation errors, 233
JRun 4, 49, 202
JRun built-in Web server, 30
JRun server, starting using the JRun Management Console, 201
jrun_precompile.bat, 197
jrun_precompile.sh, 197
jRunInstallation.log, 194
jRunInstallationError.log, 194
jrunInstallStdout.log, 194, 233
JSPG0103E error, 239

K

keys, license, 120

L

language setting errors, 227 – 228
languages, supported, 227
LD_LIBRARY_PATH, 41, 45
levels, message, in error logs, 241
LIBPATH, 41, 45

License Key Extension tool, 53
license keys, location of, 65, 79, 120
licensed packages, 21
licenses, trial, 53
limitations, to distributing and replicating components, 33
lket.bat, 53
lket.sh, 53
log files, troubleshooting with, 222
log levels, 241
LOGIN_BUTTON errors, 240
logs, database, 123

M

manually configuring, JRun built-in Web server, 193
manually installing JRun, 193
manually starting wardeploy utility, 197
mapping wsmedia virtual path, 200
mappings, virtual directory, 126
Microsoft SQL Server
  configuring, 42
  required information, 63, 77
missing file errors, 225
missing images, 240

N

names
  data source, 67, 83
  host, 68, 83
  net service, 67, 83
  service, 67, 83
net service names, 67, 83
Netegrity SiteMinder, 31
network computing, virtual, 51
no temp space error, 223
nomenclature, database, 67, 83
numbers, port, 68, 83

O

objects, dynamic shared, 49
ODBC connections, creating, 46
ODBC data source name, 63, 77
ODBC DSNs, 67, 83
operating systems, requirements, 29
optimizing performance, 80
Oracle, 45, 228
  configuring, 44
  error messages, 45, 228
  required information, 63, 77
  tnsnames.ora file, 63, 77
Orbix components, 124

P

packages, licensed, 21
parameters, -is:tempdir, 222
pass-through, 91
path, mapping wsmedia virtual, 200
performance, increasing, 71, 88, 108
performance, optimizing, 80
permissions errors, 224
port numbers, 68, 83
pre_install_db2_upgrade.cmd file, 59
precompilation errors, servlet, 225
precompilation failures
  iPlanet, 234
  JRun, 231
  WebLogic, 234
  WebSphere, 236, 239
precompile.bat, sample, 160
precompile.sh, sample, 160
pre-installation guidelines, 36
prerequisites for using this guide, x
prerequistes, UNIX, 36
problems
  installation, troubleshooting, 220
  using log files to troubleshoot, 222
  Web server, troubleshooting, 231

procedures, required configuration, 35
processor requirements, 28
products, trial, 53
Program Start folders, 135
properties, color setting, during installation, 222
protocol requirements, communications, 28

R

RAM requirements, 28
redeploying applications, 240
registries
  brio.rc, 119 – 120
  cross-platform, 119
  VPD, 119
  Windows, 119
registry entries, Hyperion Intelligence Client installation, 137
reinstallation errors, 227
reinstalling
  Hyperion Performance Suite, 97
  reasons for, 97
Remote Service Configurator, 242
removing
  client tools, 134
  Hyperion Performance Suite, 128
  JRun, 195
  JRun connection modules, 153
replicated components, 32
replicating services, 99
repository databases
  size recommendations
    general, 29
    IBM DB2, 39
    Microsoft SQL Server, 42
    Oracle, 44
required configuration procedures, 35
required information, 63, 77
requirements
  cache, 28
  CD-ROM drive, 28
  communications protocol, 28
  disk space, 28
  hardware, 28
  operating system, 29
  processor, 28
  RAM, 28
  software, 29
  TCP/IP connection, 28
  temporary space, 28, 50
  Web browsers, 30
  Web servers, 30
requirements, upgrade, 59
running servlets as standalone HTTP server, 144

S

samples
  precompile.bat, 160
  precompile.sh, 160
scripts, for help files, 151, 161, 165, 173, 179, 186, 190, 207
scripts, start, 125, 242
scripts, StartCommonServices, 125
secure environments, deploying WebLogic in, 168, 176
secure sockets-capable servers, 48
security platform driver, 90
server installation details, 101
server.xml, 112, 241
Service Configurator, Remote, 242
service names, 67, 83
services
  adding, 96
  components, 124
  installation verification, 102
  replicating, 99

services, Web, 81
servlet configuration, verifying, 146
servlet deployment errors, 225
servlet engines, configuration settings, 146
servlet precompilation errors, 225
servlets
  configuring Apache to run with JRun, 148
  configuring on commercial Web servers, 145
  invoking using a Web browser, 201
  running as a standalone HTTP server, 144
  setting up, 143
  URL for, 201
settings
  heap size, 242
  language, errors, 227 – 228
  servlet engine configuration, 146
setup.iss, 194
setupAixPlatform.bin, 60, 64, 78
setupHpBrioPlatform.bin, 60, 64, 78
setupLinuxBrioPlatform.bin, 60, 64, 78
setupSolarisPlatform.bin, 60, 64, 78
setupWinPlatform.exe, 60, 64, 78, 194
shared pool size, Oracle and, 45, 228
SHLIB_PATH, 41, 45
SIDs, 67, 83
single sign-on, 31, 90 – 91
single-computer installations, 62
SiteMinder Policy Server 5.5 Service Pack 2, 31
SiteMinder Web Agent 5.5 Service Pack 2, 31
software components, core installable unit, 123
software requirements, 29
Solaris
  installation problems
    Solaris, prerequistes, 228
  prerequistes, 36
space requirements
  disk, 28
  temporary, 28, 50
space, cache, 28
SQL errors, 224
SQL Server, Microsoft, configuring, 42
SQR
  components, 124
  connection errors, 224
  connection failures, 87
  installing, 47
  SQL, 125
sqr.exe, 63, 68, 77, 86
sqr.ini, 68, 86
SQRDIR, 68, 86, 223
sqrw.exe, 63, 68, 77, 86
standalone HTTP server, running servlets as, 144
start scripts, 125, 242
StartCommonServices scripts, 125
startCommonServices.bat, 242
startCommonServices.sh, 242
startconsole, 50
starting
  JRun server using JRun Management Console, 201
  wardeploy utility, manually, 197
startWebLogic.sh, 235
strings, connect, 67, 83
structures, WAR-compliant file, 126
suite components, 21
Sun Java System Web Server
  configuration requirements prior to installation, 50
  IWS_SERVER_HOME, 30, 50, 215
  precompilation errors, 234
superuser accounts, 36
support components, 125
supported
  external authentication systems, 31
  firewall configurations, 34
  firewalls and DMZs, 33
  languages, 227
  operating systems, 29

SVdeployjRunCommandLine, 240
system permissions, for Hyperion Performance Suite, 37
SYSTEM tablespace, 36
systems, external authentication, 31

T

tablespace full errors, 224
tablespaces, sharing, 36
TCP/IP connection requirements, 28
technical support, xv
temporary space
  errors related to, 223
  requirements, 28, 50
third-party installers, problems launching, 222
TNS names, 67, 83
tnsnames.ora, 63, 77
trial licenses, 53
troubleshooting
  installation problems, 220
  JRun
    deployment, 196, 231
    installation, 194
  log files and, 222
  service startup, 241
  Web server configuration problems, 231
typographic conventions, xiv

U

unexpected EOF error, 239
uninstalling Hyperion Performance Suite, 127
unique constraint errors, 224
unit software components, core installable, 123
UNIX
  installation errors, 222
  installation mechanisms, 51
  prerequistes, 36
UNIX, fonts on, 52
unzipWebComponentHelp.bat, 151, 161, 165, 173, 179, 186, 190, 207
unzipWebComponentHelp.sh, 151, 161, 165, 173, 179, 186, 190, 207
Upgrade Prerequisites, 59
upgrade requirements, 59
upgrading, 60, 62, 76
upgrading, distributed system, 58
URLs, servlets, 126
user accounts, creating, 37

V

variables, environment, 30, 50, 215
verifying
  services installation, 102
  servlet configuration, 146
virtual directory mappings, 126
virtual network computing, 51
virtual paths, mapping wsmedia, 200
VNC, 51
VPD Registry, 119
vpd.properties, 55, 95, 129, 227

W

WAR files, incomplete, 239
WAR-compliant file structures, 126
wardeploy utility, manually starting, 197
Web access management solutions, 31
Web browsers
  requirements, 30
  servlets, invoking through, 201
Web components. *See* servlets.
Web server configuration, problems, troubleshooting, 231

Web servers
commercial
configuring, 145
servlet engine configuration settings, 146
environment requirements, 30
increasing performance of, 71, 88
JRun built-in, 30
Web services, 76, 81
WebLogic
deploying in secure environments, 237
precompilation errors, 234, 236
WebSphere
configuration requirements prior to installation, 49
deployment errors, 239
errors, 239
precompilation errors, 239
Windows Registry, 119
Windows user accounts, creating, 37
ws.conf, 126
wsmedia virtual path, mapping, 200