

*IBM SPSS Modeler Server 16  
Administration and Performance Guide*

**IBM**

**Note**

Before using this information and the product it supports, read the information in "Notices" on page 67.

**Product Information**

This edition applies to version 16, release 0, modification 0 of IBM(r) SPSS(r) Modeler and to all subsequent releases and modifications until otherwise indicated in new editions.

---

# Contents

<b>Preface</b> . . . . .	<b>v</b>	SPSS Modeler Server Connections . . . . .	20
<b>Chapter 1. About IBM SPSS Modeler</b> . . . . .	<b>1</b>	SPSS Modeler Server Configuration . . . . .	20
IBM SPSS Modeler Products . . . . .	1	SPSS Modeler Server Monitoring . . . . .	25
IBM SPSS Modeler . . . . .	1	Using the options.cfg File. . . . .	25
IBM SPSS Modeler Server . . . . .	1	Closing Unused Database Connections . . . . .	25
IBM SPSS Modeler Administration Console . . . . .	2	Using SSL to secure data transfer . . . . .	26
IBM SPSS Modeler Batch . . . . .	2	How SSL works . . . . .	26
IBM SPSS Modeler Solution Publisher . . . . .	2	Securing client/server and server-server	
IBM SPSS Modeler Server Adapters for IBM SPSS		communications with SSL . . . . .	26
Collaboration and Deployment Services . . . . .	2	Server Log. . . . .	30
IBM SPSS Modeler Editions . . . . .	2	<b>Chapter 5. Performance Overview</b> . . . . .	<b>31</b>
IBM SPSS Modeler Documentation . . . . .	3	Server Performance and Optimization Settings . . . . .	31
SPSS Modeler Professional Documentation . . . . .	3	Client Performance and Optimization Settings. . . . .	31
SPSS Modeler Premium Documentation . . . . .	4	Database Usage and Optimization. . . . .	33
Application Examples . . . . .	4	SQL Optimization . . . . .	33
Demos Folder . . . . .	4	<b>Chapter 6. SQL Optimization</b> . . . . .	<b>35</b>
<b>Chapter 2. Architecture and Hardware</b>		How SQL Generation Works. . . . .	36
<b>Recommendations</b> . . . . .	<b>5</b>	SQL Generation Example . . . . .	36
IBM SPSS Modeler Architecture . . . . .	5	Configuring SQL Optimization . . . . .	38
Architecture Description . . . . .	5	Previewing Generated SQL . . . . .	39
Hardware Recommendations . . . . .	6	Viewing SQL for Model Nuggets . . . . .	39
Temporary Disk Space and RAM Requirements. . . . .	7	Tips for Maximizing SQL Generation. . . . .	39
Data Access. . . . .	8	Nodes Supporting SQL Generation . . . . .	40
Referencing Data Files . . . . .	9	CLEM Expressions and Operators Supporting SQL	
Importing IBM SPSS Statistics Data Files . . . . .	10	Generation . . . . .	44
Installation Instructions . . . . .	10	Using SQL Functions in CLEM Expressions. . . . .	46
<b>Chapter 3. IBM SPSS Modeler Support</b>	<b>11</b>	Writing SQL Queries . . . . .	46
Connecting to IBM SPSS Modeler Server. . . . .	11	<b>Appendix A. Configuring Oracle for</b>	
Configuring Single Sign-On . . . . .	12	<b>UNIX Platforms</b> . . . . .	<b>49</b>
Adding and Editing the IBM SPSS Modeler		Configuring Oracle for SQL Optimization . . . . .	49
Server Connection . . . . .	13	<b>Appendix B. Configuring UNIX Startup</b>	
Searching for Servers in IBM SPSS Collaboration		<b>Scripts</b> . . . . .	<b>51</b>
and Deployment Services. . . . .	14	Introduction . . . . .	51
Data and File Systems. . . . .	15	Scripts . . . . .	51
User Authentication . . . . .	15	Automatically Starting and Stopping IBM SPSS	
Permissions . . . . .	15	Modeler Server . . . . .	51
File Creation . . . . .	16	Manually Starting and Stopping IBM SPSS Modeler	
Differences in Results . . . . .	16	Server . . . . .	52
<b>Chapter 4. IBM SPSS Modeler Server</b>		Editing Scripts . . . . .	52
<b>Administration</b> . . . . .	<b>17</b>	Controlling Permissions on File Creation . . . . .	52
Starting and Stopping IBM SPSS Modeler Server . . . . .	17	IBM SPSS Modeler Server and the Data Access Pack	52
To Start, Stop, and Check Status on Windows . . . . .	17	Troubleshooting ODBC Configuration . . . . .	54
To Start, Stop, and Check Status on UNIX . . . . .	17	Library Paths. . . . .	57
Handling Unresponsive Server Processes (UNIX		Configuring a Driver for the Enterprise View Node	57
Systems) . . . . .	18	<b>Appendix C. Configuring and Running</b>	
Administration . . . . .	18	<b>SPSS Modeler Server as a Non-Root</b>	
Using IBM SPSS Modeler Administration Console	19	<b>Process on UNIX</b> . . . . .	<b>59</b>
Starting Modeler Administration Console . . . . .	19	Introduction . . . . .	59
Configuring Access with Modeler Administration			
Console. . . . .	19		

Configuring as non-root without a private password database . . . . .	59
Configuring as non-root using a private password database . . . . .	60
Running SPSS Modeler Server as a non-root user . . . . .	61
Troubleshooting user authentication failures . . . . .	61

<b>Appendix D. Configuring and Running SPSS Modeler Server with a private password file on Windows . . . . .</b>	<b>63</b>
Introduction . . . . .	63

Configuring using a private password database . . . . .	63
---	----

<b>Appendix E. Load Balancing with Server Clusters . . . . .</b>	<b>65</b>
--	-----------

<b>Notices . . . . .</b>	<b>67</b>
Trademarks . . . . .	68

<b>Index . . . . .</b>	<b>71</b>
------------------------	-----------

---

## Preface

IBM® SPSS® Modeler is the IBM Corp. enterprise-strength data mining workbench. SPSS Modeler helps organizations to improve customer and citizen relationships through an in-depth understanding of data. Organizations use the insight gained from SPSS Modeler to retain profitable customers, identify cross-selling opportunities, attract new customers, detect fraud, reduce risk, and improve government service delivery.

SPSS Modeler's visual interface invites users to apply their specific business expertise, which leads to more powerful predictive models and shortens time-to-solution. SPSS Modeler offers many modeling techniques, such as prediction, classification, segmentation, and association detection algorithms. Once models are created, IBM SPSS Modeler Solution Publisher enables their delivery enterprise-wide to decision makers or to a database.

### About IBM Business Analytics

IBM Business Analytics software delivers complete, consistent and accurate information that decision-makers trust to improve business performance. A comprehensive portfolio of business intelligence, predictive analytics, financial performance and strategy management, and analytic applications provides clear, immediate and actionable insights into current performance and the ability to predict future outcomes. Combined with rich industry solutions, proven practices and professional services, organizations of every size can drive the highest productivity, confidently automate decisions and deliver better results.

As part of this portfolio, IBM SPSS Predictive Analytics software helps organizations predict future events and proactively act upon that insight to drive better business outcomes. Commercial, government and academic customers worldwide rely on IBM SPSS technology as a competitive advantage in attracting, retaining and growing customers, while reducing fraud and mitigating risk. By incorporating IBM SPSS software into their daily operations, organizations become predictive enterprises - able to direct and automate decisions to meet business goals and achieve measurable competitive advantage. For further information or to reach a representative visit <http://www.ibm.com/spss>.

### Technical support

Technical support is available to maintenance customers. Customers may contact Technical Support for assistance in using IBM Corp. products or for installation help for one of the supported hardware environments. To reach Technical Support, see the IBM Corp. website at <http://www.ibm.com/support>. Be prepared to identify yourself, your organization, and your support agreement when requesting assistance.



---

## Chapter 1. About IBM SPSS Modeler

IBM SPSS Modeler is a set of data mining tools that enable you to quickly develop predictive models using business expertise and deploy them into business operations to improve decision making. Designed around the industry-standard CRISP-DM model, IBM SPSS Modeler supports the entire data mining process, from data to better business results.

IBM SPSS Modeler offers a variety of modeling methods taken from machine learning, artificial intelligence, and statistics. The methods available on the Modeling palette allow you to derive new information from your data and to develop predictive models. Each method has certain strengths and is best suited for particular types of problems.

SPSS Modeler can be purchased as a standalone product, or used as a client in combination with SPSS Modeler Server. A number of additional options are also available, as summarized in the following sections. For more information, see <http://www.ibm.com/software/analytics/spss/products/modeler/>.

---

### IBM SPSS Modeler Products

The IBM SPSS Modeler family of products and associated software comprises the following.

- IBM SPSS Modeler
- IBM SPSS Modeler Server
- IBM SPSS Modeler Administration Console
- IBM SPSS Modeler Batch
- IBM SPSS Modeler Solution Publisher
- IBM SPSS Modeler Server adapters for IBM SPSS Collaboration and Deployment Services

### IBM SPSS Modeler

SPSS Modeler is a functionally complete version of the product that you install and run on your personal computer. You can run SPSS Modeler in local mode as a standalone product, or use it in distributed mode along with IBM SPSS Modeler Server for improved performance on large data sets.

With SPSS Modeler, you can build accurate predictive models quickly and intuitively, without programming. Using the unique visual interface, you can easily visualize the data mining process. With the support of the advanced analytics embedded in the product, you can discover previously hidden patterns and trends in your data. You can model outcomes and understand the factors that influence them, enabling you to take advantage of business opportunities and mitigate risks.

SPSS Modeler is available in two editions: SPSS Modeler Professional and SPSS Modeler Premium. See the topic “IBM SPSS Modeler Editions” on page 2 for more information.

### IBM SPSS Modeler Server

SPSS Modeler uses a client/server architecture to distribute requests for resource-intensive operations to powerful server software, resulting in faster performance on larger data sets.

SPSS Modeler Server is a separately-licensed product that runs continually in distributed analysis mode on a server host in conjunction with one or more IBM SPSS Modeler installations. In this way, SPSS Modeler Server provides superior performance on large data sets because memory-intensive operations can be done on the server without downloading data to the client computer. IBM SPSS Modeler Server also provides support for SQL optimization and in-database modeling capabilities, delivering further benefits in performance and automation.

## IBM SPSS Modeler Administration Console

The Modeler Administration Console is a graphical application for managing many of the SPSS Modeler Server configuration options, which are also configurable by means of an options file. The application provides a console user interface to monitor and configure your SPSS Modeler Server installations, and is available free-of-charge to current SPSS Modeler Server customers. The application can be installed only on Windows computers; however, it can administer a server installed on any supported platform.

## IBM SPSS Modeler Batch

While data mining is usually an interactive process, it is also possible to run SPSS Modeler from a command line, without the need for the graphical user interface. For example, you might have long-running or repetitive tasks that you want to perform with no user intervention. SPSS Modeler Batch is a special version of the product that provides support for the complete analytical capabilities of SPSS Modeler without access to the regular user interface. SPSS Modeler Server is required to use SPSS Modeler Batch.

## IBM SPSS Modeler Solution Publisher

SPSS Modeler Solution Publisher is a tool that enables you to create a packaged version of an SPSS Modeler stream that can be run by an external runtime engine or embedded in an external application. In this way, you can publish and deploy complete SPSS Modeler streams for use in environments that do not have SPSS Modeler installed. SPSS Modeler Solution Publisher is distributed as part of the IBM SPSS Collaboration and Deployment Services - Scoring service, for which a separate license is required. With this license, you receive SPSS Modeler Solution Publisher Runtime, which enables you to execute the published streams.

## IBM SPSS Modeler Server Adapters for IBM SPSS Collaboration and Deployment Services

A number of adapters for IBM SPSS Collaboration and Deployment Services are available that enable SPSS Modeler and SPSS Modeler Server to interact with an IBM SPSS Collaboration and Deployment Services repository. In this way, an SPSS Modeler stream deployed to the repository can be shared by multiple users, or accessed from the thin-client application IBM SPSS Modeler Advantage. You install the adapter on the system that hosts the repository.

---

## IBM SPSS Modeler Editions

SPSS Modeler is available in the following editions.

### SPSS Modeler Professional

SPSS Modeler Professional provides all the tools you need to work with most types of structured data, such as behaviors and interactions tracked in CRM systems, demographics, purchasing behavior and sales data.

### SPSS Modeler Premium

SPSS Modeler Premium is a separately-licensed product that extends SPSS Modeler Professional to work with specialized data such as that used for entity analytics or social networking, and with unstructured text data. SPSS Modeler Premium comprises the following components.

**IBM SPSS Modeler Entity Analytics** adds an extra dimension to IBM SPSS Modeler predictive analytics. Whereas predictive analytics attempts to predict future behavior from past data, entity analytics focuses on improving the coherence and consistency of current data by resolving identity conflicts within the records themselves. An identity can be that of an individual, an organization, an object, or any other



entity for which ambiguity might exist. Identity resolution can be vital in a number of fields, including customer relationship management, fraud detection, anti-money laundering, and national and international security.

**IBM SPSS Modeler Social Network Analysis** transforms information about relationships into fields that characterize the social behavior of individuals and groups. Using data describing the relationships underlying social networks, IBM SPSS Modeler Social Network Analysis identifies social leaders who influence the behavior of others in the network. In addition, you can determine which people are most affected by other network participants. By combining these results with other measures, you can create comprehensive profiles of individuals on which to base your predictive models. Models that include this social information will perform better than models that do not.

**IBM SPSS Modeler Text Analytics** uses advanced linguistic technologies and Natural Language Processing (NLP) to rapidly process a large variety of unstructured text data, extract and organize the key concepts, and group these concepts into categories. Extracted concepts and categories can be combined with existing structured data, such as demographics, and applied to modeling using the full suite of IBM SPSS Modeler data mining tools to yield better and more focused decisions.

---

## IBM SPSS Modeler Documentation

Documentation in online help format is available from the Help menu of SPSS Modeler. This includes documentation for SPSS Modeler, SPSS Modeler Server, and SPSS Modeler Solution Publisher, as well as the Applications Guide and other supporting materials.

Complete documentation for each product (including installation instructions) is available in PDF format under the *\Documentation* folder on each product DVD. Installation documents can also be downloaded from the web at <http://www-01.ibm.com/support/docview.wss?uid=swg27038316>.

Documentation in both formats is also available from the SPSS Modeler Information Center at <http://publib.boulder.ibm.com/infocenter/spssmodl/v16r0m0/>.

## SPSS Modeler Professional Documentation

The SPSS Modeler Professional documentation suite (excluding installation instructions) is as follows.

- **IBM SPSS Modeler User's Guide.** General introduction to using SPSS Modeler, including how to build data streams, handle missing values, build CLEM expressions, work with projects and reports, and package streams for deployment to IBM SPSS Collaboration and Deployment Services, Predictive Applications, or IBM SPSS Modeler Advantage.
- **IBM SPSS Modeler Source, Process, and Output Nodes.** Descriptions of all the nodes used to read, process, and output data in different formats. Effectively this means all nodes other than modeling nodes.
- **IBM SPSS Modeler Modeling Nodes.** Descriptions of all the nodes used to create data mining models. IBM SPSS Modeler offers a variety of modeling methods taken from machine learning, artificial intelligence, and statistics.
- **IBM SPSS Modeler Algorithms Guide.** Descriptions of the mathematical foundations of the modeling methods used in IBM SPSS Modeler. This guide is available in PDF format only.
- **IBM SPSS Modeler Applications Guide.** The examples in this guide provide brief, targeted introductions to specific modeling methods and techniques. An online version of this guide is also available from the Help menu. See the topic "Application Examples" on page 4 for more information.
- **IBM SPSS Modeler Python Scripting and Automation.** Information on automating the system through Python scripting, including the properties that can be used to manipulate nodes and streams.
- **IBM SPSS Modeler Deployment Guide.** Information on running IBM SPSS Modeler streams and scenarios as steps in processing jobs under IBM SPSS Collaboration and Deployment Services Deployment Manager.

- **IBM SPSS Modeler CLEF Developer's Guide.** CLEF provides the ability to integrate third-party programs such as data processing routines or modeling algorithms as nodes in IBM SPSS Modeler.
- **IBM SPSS Modeler In-Database Mining Guide.** Information on how to use the power of your database to improve performance and extend the range of analytical capabilities through third-party algorithms.
- **IBM SPSS Modeler Server Administration and Performance Guide.** Information on how to configure and administer IBM SPSS Modeler Server.
- **IBM SPSS Modeler Administration Console User Guide.** Information on installing and using the console user interface for monitoring and configuring IBM SPSS Modeler Server. The console is implemented as a plug-in to the Deployment Manager application.
- **IBM SPSS Modeler CRISP-DM Guide.** Step-by-step guide to using the CRISP-DM methodology for data mining with SPSS Modeler.
- **IBM SPSS Modeler Batch User's Guide.** Complete guide to using IBM SPSS Modeler in batch mode, including details of batch mode execution and command-line arguments. This guide is available in PDF format only.

## SPSS Modeler Premium Documentation

The SPSS Modeler Premium documentation suite (excluding installation instructions) is as follows.

- **IBM SPSS Modeler Entity Analytics User Guide.** Information on using entity analytics with SPSS Modeler, covering repository installation and configuration, entity analytics nodes, and administrative tasks.
- **IBM SPSS Modeler Social Network Analysis User Guide.** A guide to performing social network analysis with SPSS Modeler, including group analysis and diffusion analysis.
- **SPSS Modeler Text Analytics User's Guide.** Information on using text analytics with SPSS Modeler, covering the text mining nodes, interactive workbench, templates, and other resources.

---

## Application Examples

While the data mining tools in SPSS Modeler can help solve a wide variety of business and organizational problems, the application examples provide brief, targeted introductions to specific modeling methods and techniques. The data sets used here are much smaller than the enormous data stores managed by some data miners, but the concepts and methods involved should be scalable to real-world applications.

You can access the examples by clicking **Application Examples** on the Help menu in SPSS Modeler. The data files and sample streams are installed in the *Demos* folder under the product installation directory. See the topic “Demos Folder” for more information.

**Database modeling examples.** See the examples in the *IBM SPSS Modeler In-Database Mining Guide*.

**Scripting examples.** See the examples in the *IBM SPSS Modeler Scripting and Automation Guide*.

---

## Demos Folder

The data files and sample streams used with the application examples are installed in the *Demos* folder under the product installation directory. This folder can also be accessed from the IBM SPSS Modeler program group on the Windows Start menu, or by clicking *Demos* on the list of recent directories in the File Open dialog box.

---

## Chapter 2. Architecture and Hardware Recommendations

---

### IBM SPSS Modeler Architecture

This section describes the architecture of IBM SPSS Modeler Server, including the server software, the client software, and the database. It includes information about how IBM SPSS Modeler Server is designed for optimal performance and provides recommendations for maximizing this performance by choosing appropriately sized hardware. It concludes with a section on data access, which describes where to set up the necessary ODBC drivers.

---

### Architecture Description

IBM SPSS Modeler Server uses a three-tier, distributed architecture. Software operations are shared between the client and the server computers. The advantages of installing and using IBM SPSS Modeler Server (versus the standalone IBM SPSS Modeler), especially when dealing with large data sets, are numerous:

- IBM SPSS Modeler Server can run on UNIX, in addition to Windows, allowing more flexibility in deciding where to install it. On any platform, you can dedicate a faster, larger server computer to data mining processes.
- IBM SPSS Modeler Server is optimized for fast performance. When operations cannot be pushed into the database, IBM SPSS Modeler Server stores the intermediate results as temporary files on disk rather than in RAM. Because servers usually have significant disk space available, IBM SPSS Modeler Server can perform sort, merge, and aggregation operations on very large data sets.
- Using the client-server architecture, you can centralize data-mining processes in your organization. Centralization can help to formalize the role of data mining in your business processes.
- Using administrator tools like the IBM SPSS Modeler Administration Console (included with IBM SPSS Modeler Server) and IBM SPSS Collaboration and Deployment Services (sold separately), you can monitor data mining processes, ensuring that adequate computing resources are available. With IBM SPSS Collaboration and Deployment Services you can automate certain data mining tasks, manage access to data models, and share results across your organization.

The components of IBM SPSS Modeler's distributed architecture are shown in the "IBM SPSS Modeler Server Architecture" graphic.

- **IBM SPSS Modeler.** The client software is installed on the end user's computer. It provides the user interface and displays the data mining results. The client is a complete installation of IBM SPSS Modeler software, but when it is connected to IBM SPSS Modeler Server for distributed analysis, its execution engine is inactive. The IBM SPSS Modeler runs on Windows operating systems only.
- **IBM SPSS Modeler Server.** The server software installed on a server computer, with network connectivity to both the IBM SPSS Modeler(s) and the database. IBM SPSS Modeler Server runs as a service (on Windows) or a daemon process (on UNIX), waiting for clients to connect. It handles the execution of streams and scripts created using the IBM SPSS Modeler.
- **Database server.** The database server could be a live data warehouse (for example, Oracle on a large UNIX server) or, to reduce impact on other operational systems, a data mart on a local/departmental server (for example, SQL Server on Windows).

### IBM SPSS Modeler Server Architecture

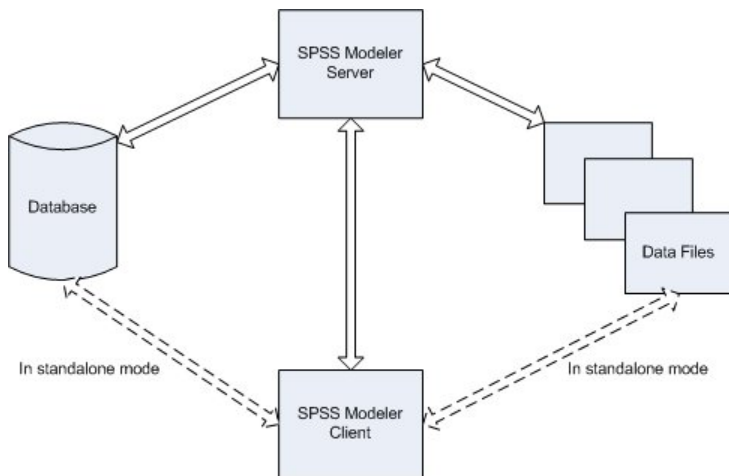


Figure 1. IBM SPSS Modeler Server architecture

With the distributed architecture, most of the processing occurs on the server computer. When the end user executes a stream, IBM SPSS Modeler sends a description of the stream to the server. The server determines which operations can be executed in SQL and creates the appropriate queries. These queries are executed in the database, and the resulting data are passed to the server for any processing that cannot be expressed using SQL. Once the processing is complete, only the relevant results are passed back to the client.

If necessary, IBM SPSS Modeler Server can execute all IBM SPSS Modeler operations outside of the database. It automatically balances its use of RAM and disk memory to hold data for manipulation. This process makes IBM SPSS Modeler Server fully compatible with flat files.

Load balancing is also available by using a cluster of servers for processing. Clustering is available starting in IBM SPSS Collaboration and Deployment Services 3.5 through the Coordinator of Processes plug-in. See the topic Appendix E, "Load Balancing with Server Clusters," on page 65 for more information. You can connect to a server or cluster managed in the Coordinator of Processes directly through IBM SPSS Modeler's Server Login dialog. See the topic "Connecting to IBM SPSS Modeler Server" on page 11 for more information.

### Standalone Client

IBM SPSS Modeler may also be configured to run as a self-contained desktop application, shown in the graphic below. See Chapter 3, "IBM SPSS Modeler Support," on page 11 for more information.

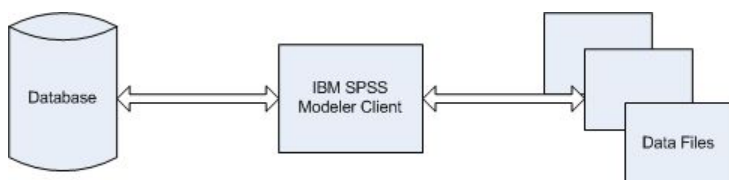


Figure 2. IBM SPSS Modeler standalone

## Hardware Recommendations

As you plan your IBM SPSS Modeler Server installation, you should consider the hardware that you will use. Although IBM SPSS Modeler Server is designed to be speedy, you can maximize its efficiency by using hardware that is sized appropriately for your data mining tasks. Upgrading hardware is often the simplest and most economical way to improve performance across the board.

**Dedicated server.** Install IBM SPSS Modeler Server on a dedicated server machine where it will not compete for resources with other applications, including any databases to which IBM SPSS Modeler Server may be connecting. Model-building operations in particular are resource-intensive and perform much better when not in competition with other applications.

*Note:* Although installing IBM SPSS Modeler Server on the same computer as the database can reduce data-transfer time between the database and the server by avoiding network overhead, in most cases the best configuration is to have the server and database on separate machines to avoid competition for resources. Provide a fast connection between the two to minimize the cost of data transfer.

**Processors.** The number of processors on the machine should be no less than the number of concurrent tasks (simultaneously executing streams) you expect to run on a regular basis. In general, the more processors, the better.

- A single instance of IBM SPSS Modeler Server will accept connections from multiple clients (users), and each client connection can initiate multiple stream executions. One server can therefore have several execution tasks in progress at any one time.
- As a rule of thumb, allow one processor for one or two users, two processors for up to four users, and four processors for up to eight users. Add one additional processor for every two to four users beyond that, depending on the mix of work.
- To the extent that some processing may be pushed back to the database through SQL optimization, it may be possible to share a CPU between two or more users with minimal loss in performance.
- Multithreading capabilities make it possible for a single task to take advantage of multiple processors, so adding processors can improve performance even in cases where only one task is running at a time. Generally, multithreading is used for C5.0 model building and certain data preparation operations (sort, aggregate, and merge).

**64-bit platforms.** If you plan to process or build models on very large volumes of data, use 64-bit Solaris, Windows, or Linux as your IBM SPSS Modeler Server platform, and maximize the amount of RAM for the machine. For larger data sets, the server can quickly exhaust the per-process memory limits imposed by 32-bit platforms, forcing data to be spilled to disk and significantly increasing the running time. 64-bit server implementations can take advantage of additional RAM; a minimum of 8 gigabytes (GB) is recommended. 64-bit support is available for Solaris, Windows, and Linux platforms.

**Future needs.** Whenever feasible, make sure that server hardware is expandable in terms of memory and CPUs, both to accommodate increases in usage (for example, increased numbers of simultaneous users or increases in the existing users processing requirements) and increased multithreading capabilities of IBM SPSS Modeler Server in the future.

## Temporary Disk Space and RAM Requirements

IBM SPSS Modeler Server uses temporary disk space to process large volumes of data. The amount of temporary space that you need depends on the volume and type of data that you process and the type of operations you perform. The data volume is proportional to both the number of rows *and* the number of columns. The more rows and columns that you process, the more disk space you need.

This section describes the conditions under which temporary disk space and extra RAM are required, and how to estimate the amount required. Note that this section does not discuss the temporary disk space requirements for processes that occur in a database, since these requirements are specific to each database.

### Conditions That Require Temporary Disk Space

IBM SPSS Modeler Server's powerful SQL optimization feature allows processing to occur in the database (rather than on the server) whenever possible. However, when any of the following conditions are true, SQL optimization cannot be used:

- The data to be processed are held in a flat file rather than in a database.

- SQL optimization is turned off.
- The processing operation cannot be optimized using SQL.

When SQL optimization cannot be used, the following data manipulation nodes and CLEM functions create temporary disk copies of some or all of the data. If the streams used at your site contain these processing commands or functions, you may need to set aside additional disk space on your server.

- Aggregate node
- Distinct node
- Binning node
- Merge node when using the merge-by-key option
- Any modeling node
- Sort node
- Table output node
- @OFFSET functions in which the lookup condition uses @THIS.
- Any @ function, such as @MIN, @MAX, and @AVE, in which the offset parameter is calculated.

### Calculating the Amount of Temporary Disk Space

In general, IBM SPSS Modeler Server needs to be able to write a temporary file that is at least *three times as large* as the original data set. For example, if the data file is 2GB and SQL generation is not used, IBM SPSS Modeler Server will require 6GB of disk space to process the data. Because each concurrent user account creates its own temporary files, you will need to increase the disk space accordingly for each concurrent user.

If you find that your site frequently uses large temporary files, consider using a separate file system for IBM SPSS Modeler's temporary files, created on a separate disk. For best results, a RAID 0 or striped data set that spans multiple physical disks can be used to speed up disk operations, ideally with each disk in the striped file system on a separate disk controller.

### RAM Requirements

For most processing that cannot be performed in the database, IBM SPSS Modeler Server stores the intermediate results as temporary files on disk rather than in memory (RAM). However, for modeling nodes, RAM is used if possible. The Neural Net, Kohonen, and K-Means nodes require large amounts of RAM. If these nodes are frequently used at your site, consider installing more RAM on the server.

In general, the number of bytes of RAM needed can be estimated by

$$(\text{number\_of\_records} * \text{number\_of\_cells\_per\_record}) * \text{number\_of\_bytes\_per\_cell}$$

where `number_of_cells_per_record` can become very large when there are nominal fields.

Refer to the system requirements section of the server installation guide for current RAM recommendations. For four or more simultaneous users, even more RAM is recommended. Memory must be shared between concurrent tasks, so scale up accordingly. In general, adding memory is likely to be one of the most cost-effective ways to improve performance across the board.

---

## Data Access

To read or write to a database, you must have an ODBC data source installed and configured for the relevant database, with read or write permissions as needed. The IBM SPSS Data Access Pack includes a set of ODBC drivers that can be used for this purpose, and these drivers are available on the IBM SPSS Data Access Pack DVD or from the download site. If you have questions about creating or setting permissions for ODBC data sources, contact your database administrator.

Supported ODBC Drivers

For the latest information on which databases and ODBC drivers are supported and tested for use with IBM SPSS Modeler 16, see the product compatibility matrices on the corporate Support site (<http://www.ibm.com/support>).

## Where to Install Drivers

Note that ODBC drivers must be installed and configured on each computer where processing may occur.

- If you are running IBM SPSS Modeler in local (standalone) mode, the drivers must be installed on the local computer.
- If you are running IBM SPSS Modeler in distributed mode against a remote IBM SPSS Modeler Server, the ODBC drivers need to be installed on the computer where IBM SPSS Modeler Server is installed. For IBM SPSS Modeler Server on UNIX systems, see also "Configuring ODBC drivers on UNIX systems" later in this section.
- If you need to access the same data sources from both IBM SPSS Modeler and IBM SPSS Modeler Server, the ODBC drivers must be installed on both computers.
- If you are running IBM SPSS Modeler over Terminal Services, the ODBC drivers need to be installed on the Terminal Services server on which you have IBM SPSS Modeler installed.

## Configuring ODBC drivers on UNIX systems

By default, the DataDirect Driver Manager is not configured for IBM SPSS Modeler Server on UNIX systems. To configure UNIX to load the DataDirect Driver Manager, enter the following commands:

```
cd modeler_server_install_directory/bin
rm -f libspssodbc.so
ln -s libspssodbc_datadirect.so libspssodbc.so
```

Doing so removes the default link and creates a link to the DataDirect Driver Manager.

*Note:* The above rules apply specifically to accessing data in a database. Other types of file operations, such as opening and saving streams, projects, models, nodes, PMML, output, and script files, are always done on the client and are always specified in terms of the file system of the client computer. In addition, the Set Directory command in IBM SPSS Modeler sets the working directory for *local* client objects (for example, streams) but does not affect the server's working directory.

**UNIX.** For information about how to configure IBM SPSS Modeler Server on UNIX to work with the IBM SPSS Statistics data access technology, see Appendix B, "Configuring UNIX Startup Scripts," on page 51.

## Referencing Data Files

**Windows.** If you store data on the same computer as IBM SPSS Modeler Server, we recommend that you give the path to the data from the perspective of the server computer (for example, *C:\ServerData\Sales 1998.csv*). Performance is faster when the network is not used to locate the file.

If the data is stored on a different host, we recommend using UNC file references (for example, *\\mydataserver\ServerData\Sales 1998.csv*). Note that UNC names work only when the path contains the name of a shared network resource. The referencing computer must have permission to read the specified file. If you switch frequently from distributed to local analysis mode, use UNC file references because they work regardless of the mode.

**UNIX.** To reference data files that reside on a UNIX server, use the full file specification and forward slashes (for example, */public/data/ServerData/Sales 1998.csv*). Avoid using the backslash character in the UNIX directory and in filenames for data used with IBM SPSS Modeler Server. It does not matter whether a text file uses UNIX or DOS format—both are handled automatically.

## Importing IBM SPSS Statistics Data Files

If you are also running IBM SPSS Statistics Server at your site, users may want to import or export IBM SPSS Statistics data while in distributed mode. Recall that when the IBM SPSS Modeler runs in distributed mode, it presents the server's file system. The IBM SPSS Statistics client works in the same way. For importing and exporting to take place between the two applications, both clients must be operating in the same mode. If they are not, their views of the file systems will be different and they will not be able to share files. IBM SPSS Modeler's IBM SPSS Statistics nodes can automatically start the IBM SPSS Statistics client, but users must first ensure that the IBM SPSS Statistics client is operating in the same mode as IBM SPSS Modeler.

---

## Installation Instructions

For information on installing IBM SPSS Modeler Server, see the instructions in the `\documentation\installation\` folder on the Server CD. Separate documents are available for Windows and UNIX.

For complete information on installing and using the IBM SPSS Modeler, see the client CD.



---

## Chapter 3. IBM SPSS Modeler Support

This section is intended for administrators and help-desk personnel who support users of IBM SPSS Modeler. It covers the following topics:

- How to log on to IBM SPSS Modeler Server (or run standalone by disconnecting from a Server)
- Data and file systems that users may need
- User accounts and file permissions pertaining to IBM SPSS Modeler Server
- Differences in results that users may see when switching between IBM SPSS Modeler Server and IBM SPSS Modeler

---

### Connecting to IBM SPSS Modeler Server

IBM SPSS Modeler can be run as a standalone application, or as a client connected to IBM SPSS Modeler Server directly or to an IBM SPSS Modeler Server or server cluster through the Coordinator of Processes plug-in from IBM SPSS Collaboration and Deployment Services. The current connection status is displayed at the bottom left of the IBM SPSS Modeler window.

Whenever you want to connect to a server, you can manually enter the server name to which you want to connect or select a name that you have previously defined. However, if you have IBM SPSS Collaboration and Deployment Services, you can search through a list of servers or server clusters from the Server Login dialog box. The ability to browse through the Statistics services running on a network is made available through the Coordinator of Processes.

To Connect to a Server

1. On the Tools menu, click **Server Login**. The Server Login dialog box opens. Alternatively, double-click the connection status area of the IBM SPSS Modeler window.
2. Using the dialog box, specify options to connect to the local server computer or select a connection from the table.
  - Click **Add** or **Edit** to add or edit a connection. See the topic “Adding and Editing the IBM SPSS Modeler Server Connection” on page 13 for more information.
  - Click **Search** to access a server or server cluster in the Coordinator of Processes. See the topic “Searching for Servers in IBM SPSS Collaboration and Deployment Services” on page 14 for more information.

**Server table.** This table contains the set of defined server connections. The table displays the default connection, server name, description, and port number. You can manually add a new connection, as well as select or search for an existing connection. To set a particular server as the default connection, select the check box in the Default column in the table for the connection.

**Default data path.** Specify a path used for data on the server computer. Click the ellipsis button (...) to browse to the required location.

**Set Credentials.** Leave this box unchecked to enable the **single sign-on** feature, which attempts to log you in to the server using your local computer username and password details. If single sign-on is not possible, or if you check this box to disable single sign-on (for example, to log in to an administrator account), the following fields are enabled for you to enter your credentials.

**User ID.** Enter the user name with which to log on to the server.

**Password.** Enter the password associated with the specified user name.

**Domain.** Specify the domain used to log on to the server. A domain name is required only when the server computer is in a different Windows domain than the client computer.

3. Click **OK** to complete the connection.

To Disconnect from a Server

1. On the Tools menu, click **Server Login**. The Server Login dialog box opens. Alternatively, double-click the connection status area of the IBM SPSS Modeler window.
2. In the dialog box, select the Local Server and click **OK**.

## Configuring Single Sign-On

You can connect to an IBM SPSS Modeler Server that is running on any supported platform using Single Sign-On. To connect using Single Sign-On, you must first configure your IBM SPSS Modeler server and client machines.

If you are using Single Sign-On to connect to both IBM SPSS Modeler Server and IBM SPSS Collaboration and Deployment Services, you must connect to IBM SPSS Collaboration and Deployment Services before you connect to IBM SPSS Modeler.

**Note:** Before you configure your IBM SPSS Modeler server and client machines for Single Sign-On, you must make sure that the machines have access to the domain controller server.

### Configuring Single Sign-On on Windows

#### To Configure IBM SPSS Modeler Server

1. Ensure that the Windows server machine is a member of the Active Directory (AD) domain.
2. In the IBM SPSS Modeler Server installation location, locate the folder called config.
3. In the config folder, create a subfolder called sso.
4. In the sso folder, create a krb5.conf file. Instructions for how to create the krb5.conf file can be found at [http://web.mit.edu/kerberos/krb5-current/doc/admin/conf\\_files/krb5\\_conf.html](http://web.mit.edu/kerberos/krb5-current/doc/admin/conf_files/krb5_conf.html). An example of a krb5.conf file is given below:

```
[libdefaults]
    default_realm = MODELERSSO.COM
    dns_lookup_kdc = true
    dns_lookup_realm = true

[realms]
    MODELERSSO.COM = {
        kdc = modelersso.com:88
        admin_server = modelersso.com:749
        default_domain = MODELERSSO.COM
    }

[domain_realm]
    .modelersso.com = MODELERSSO.COM
```

#### To Configure IBM SPSS Modeler Client

1. Ensure that the local Windows machine that is running IBM SPSS Modeler is a member of the AD domain.
2. Add the domain user as an administrator on the local machine.
3. Enable Windows to access the TGT session key:
  - a. From the **Start** menu, click **Run**.
  - b. Enter regedit and click **OK** to open the Registry Editor.
  - c. Navigate to the registry location appropriate to the operating system of the local machine:
    - On Windows XP: My Computer\HKEY\_LOCAL\_MACHINE\System\CurrentControlSet\Control\Lsa\Kerberos
    - On Windows Vista, or Windows 7: My Computer\HKEY\_LOCAL\_MACHINE\System\CurrentControlSet\Control\Lsa\Kerberos\Parameters
  - d. Right click the folder and select **New > DWORD**. The name of the new value should be allowtgtsessionkey.

- e. Set the value of `allowtgtsessionkey` to a hexadecimal value of 1, that is 0x0000001.
  - f. Close the Registry Editor.
  - g. If the operating system of the local machine is Windows Vista or Windows 7, you must run `kinit.exe`, which can be found in `<IBM SPSS Modeler installation location>\jre\bin`.
4. In the `config` folder of the IBM SPSS Modeler installation location, create a folder called `sso`.
  5. Copy the `krb5.conf` file from the server in to the `sso` folder.
  6. Restart the local machine and the server machine.

## Configuring Single Sign-On on UNIX

To configure Single Sign-On for UNIX server machines, you can add the UNIX machine to the Windows AD domain, then follow the instructions for configuring Single Sign-On on Windows. See the topic “Configuring Single Sign-On on Windows” on page 12 for more information. Alternatively, you can perform the following steps:

1. Create a domain user account for the UNIX machine.
2. Change the host name. If you are using RedHat Linux, open the `/etc/sysconfig/network` file and modify `HOSTNAME` to the form `<name>.<realm>`. This enables the AD to find the server credentials.
3. To enable the DNS server to find the UNIX machine, take one of the following steps:
  - Open the `%windows%/system32/drivers/etc/hosts` file and add the IP/host mapping, for example:
 

```
192.168.1.102 test.modelersso.com test
```
  - Or
  - Add a new reverse lookup zone entry. This will add an IP/host mapping on the DNS server.

If the DNS entry for the UNIX machine is not correct, you can manually add the reverse lookup entry on the DNS server.

4. Follow the instructions for configuring IBM SPSS Modeler Client, which can be found in the topic “Configuring Single Sign-On on Windows” on page 12.

## Single Sign-On for Data Sources

You can connect to databases from IBM SPSS Modeler using single sign-on. If you want to create a database connection using single sign-on, you must first use your ODBC management software to properly configure a data source and single sign-on token. Then when connecting to a database in IBM SPSS Modeler, IBM SPSS Modeler will use that same single sign-on token and the user will not be prompted to log on to the data source.

However, if the data source was not configured properly for single sign-on, IBM SPSS Modeler will prompt the user to log on to the data source. The user will still be able to access the data source after providing valid credentials.

For complete details about configuring ODBC data sources on your system with single sign-on enabled, see your database vendor documentation. Following is an example of the general steps that may be involved:

1. Configure your database so it can support Kerberos single sign-on.
2. On the IBM SPSS Modeler Server machine, create an ODBC data source and test it. The DSN connection should not require a user ID and password.
3. Connect to IBM SPSS Modeler Server using single sign-on and begin using the ODBC data source created and validate in step 2.

## Adding and Editing the IBM SPSS Modeler Server Connection

You can manually edit or add a server connection in the Server Login dialog box. By clicking Add, you can access an empty Add/Edit Server dialog box in which you can enter server connection details. By selecting an existing connection and clicking Edit in the Server Login dialog box, the Add/Edit Server dialog box opens with the details for that connection so that you can make any changes.

**Note:** You cannot edit a server connection that was added from IBM SPSS Collaboration and Deployment Services, since the name, port, and other details are defined in IBM SPSS Collaboration and Deployment Services. Best practice dictates that the same ports should be used to communicate with both IBM SPSS Collaboration and Deployment Services and SPSS Modeler Client. These can be set as `max_server_port` and `min_server_port` in the `options.cfg` file.

To Add Server Connections

1. On the Tools menu, click **Server Login**. The Server Login dialog box opens.
  2. In this dialog box, click **Add**. The Server Login Add/Edit Server dialog box opens.
  3. Enter the server connection details and click **OK** to save the connection and return to the Server Login dialog box.
- **Server.** Specify an available server or select one from the list. The server computer can be identified by an alphanumeric name (for example, *myserver*) or an IP address assigned to the server computer (for example, 202.123.456.78).
  - **Port.** Give the port number on which the server is listening. If the default does not work, ask your system administrator for the correct port number.
  - **Description.** Enter an optional description for this server connection.
  - **Ensure secure connection (use SSL).** Specifies whether an SSL (**Secure Sockets Layer**) connection should be used. SSL is a commonly used protocol for securing data sent over a network. To use this feature, SSL must be enabled on the server hosting IBM SPSS Modeler Server. If necessary, contact your local administrator for details.

To Edit Server Connections

1. On the Tools menu, click **Server Login**. The Server Login dialog box opens.
2. In this dialog box, select the connection you want to edit and then click **Edit**. The Server Login Add/Edit Server dialog box opens.
3. Change the server connection details and click **OK** to save the changes and return to the Server Login dialog box.

## Searching for Servers in IBM SPSS Collaboration and Deployment Services

Instead of entering a server connection manually, you can select a server or server cluster available on the network through the Coordinator of Processes, available in IBM SPSS Collaboration and Deployment Services. A server cluster is a group of servers from which the Coordinator of Processes determines the server best suited to respond to a processing request.

Although you can manually add servers in the Server Login dialog box, searching for available servers lets you connect to servers without requiring that you know the correct server name and port number. This information is automatically provided. However, you still need the correct logon information, such as username, domain, and password.

*Note:* If you do not have access to the Coordinator of Processes capability, you can still manually enter the server name to which you want to connect or select a name that you have previously defined. See the topic “Adding and Editing the IBM SPSS Modeler Server Connection” on page 13 for more information.

To search for servers and clusters

1. On the Tools menu, click **Server Login**. The Server Login dialog box opens.
2. In this dialog box, click **Search** to open the Search for Servers dialog box. If you are not logged on to IBM SPSS Collaboration and Deployment Services when you attempt to browse the Coordinator of Processes, you will be prompted to do so.
3. Select the server or server cluster from the list.
4. Click **OK** to close the dialog box and add this connection to the table in the Server Login dialog box.

---

## Data and File Systems

Users working with IBM SPSS Modeler Server will probably need to access data files and other data sources on the network, as well as save files on the network. They may need the following information, as applicable:

- **ODBC data source information.** If users need access to ODBC data sources defined on the server computer, they will need the names, descriptions, and login information (including database login IDs and passwords) for the data sources.
- **Data file access.** If users need to access data files on the server computer or elsewhere on the network, they will need the names and locations of the data files.
- **Location for saved files.** When users save data while connected to IBM SPSS Modeler Server, they may attempt to save files on the server computer. However, this is often a write-protected location. If so, let users know where they should save data files. (Typically, the location is the user's home directory.)

---

## User Authentication

IBM SPSS Modeler Server uses the operating system on the server machine to authenticate users who connect to the server. When a user connects to IBM SPSS Modeler Server, all operations performed on behalf of the user are performed in the user's security context. Access to database tables is subject to user and/or password privileges in the database itself.

**Windows.** On Windows, any user with a valid account on the host network may log on. With the default authentication, users must have modify access rights to the `<modeler_server_install>\Tmp` directory. Without these rights, users cannot log on to IBM SPSS Modeler Server from the client using the default authentication on Windows.

**UNIX.** By default, IBM SPSS Modeler Server is assumed to run as root on UNIX. This allows any user with a valid account on the host network to log on and limits users' file access to their own files and directories. However, you can configure IBM SPSS Modeler Server to run without root privilege. In this case, you must create a private password database to be used for authentication, and all IBM SPSS Modeler users must share a single UNIX user account (and, consequently, share access to data files). See the topic "Configuring as non-root using a private password database" on page 60 for more information.

On Solaris, Linux and AIX platforms, IBM SPSS Modeler Server uses PAM for authentication. You can use the service name `modelerserver` to configure the PAM modules for IBM SPSS Modeler Server if required.

## Permissions

**Windows.** A user connecting to server software that is installed on an *NTFS* drive must login with an account that has the following permissions.

- Read and execute permissions to the server's installation directory and its subdirectories
- Read, execute, and write permissions to the directory location for temporary files.

In Windows Server 2008 and later, you cannot assume that users have these permission. Be sure to explicitly set permissions as needed.

If the server software is installed on a *FAT* drive, you do not need to set permissions because all files allow users to have full control.

**UNIX.** If you are not using internal authentication, a user connecting to the server software must login with an account that has the following permissions:

- Read and execute permissions to the server's installation directory and its subdirectories
- Read, execute, and write permissions to the directory location for temporary files.

## File Creation

When IBM SPSS Modeler Server accesses and processes data, it often has to keep a temporary copy of that data on disk. The amount of disk space that will be used for temporary files depends on the size of the data file that the end user is analyzing and the type of analysis that he or she is performing. See the topic “Temporary Disk Space and RAM Requirements” on page 7 for more information.

**UNIX.** The UNIX versions of IBM SPSS Modeler Server use the UNIX `umask` command to set file permissions for the temporary files. You can override the server's default permissions. See the topic “Controlling Permissions on File Creation” on page 52 for more information.

---

## Differences in Results

Users who run analyses in both modes may see slight differences in the results between IBM SPSS Modeler and IBM SPSS Modeler Server. The discrepancy usually occurs because of record ordering or rounding differences.

**Record ordering.** Unless a stream explicitly orders records by sorting them, the order in which records are presented may vary between streams executed locally and those executed on the server. There may also be differences in order between operations run within a database and those run in IBM SPSS Modeler Server. These differences are due to the different algorithms used by each system to implement functions that may reorder records, such as aggregation. Also, note that SQL does not specify the order in which records are returned from a database in cases where there is no explicit ordering operation.

**Rounding differences.** IBM SPSS Modeler running in local mode uses a different internal format for storing floating point values than does IBM SPSS Modeler Server. Due to rounding differences, results might vary slightly between each version.

---

## Chapter 4. IBM SPSS Modeler Server Administration

This chapter contains information about starting and stopping IBM SPSS Modeler Server, configuring various server options, and interpreting the log file. It describes how to use IBM SPSS Modeler Administration Console, an application that facilitates server configuration and monitoring. For installation instructions for this component, refer to the installation instructions for IBM SPSS Modeler Server, available with that product.

---

### Starting and Stopping IBM SPSS Modeler Server

IBM SPSS Modeler Server runs as a service on Windows or as a daemon process on UNIX.

*Scheduling note:* Stopping IBM SPSS Modeler Server disconnects end users and terminates their sessions, so try to schedule server restarts during periods of low usage. If this is not possible, be sure to notify users before stopping the server.

#### To Start, Stop, and Check Status on Windows

On Windows, you control IBM SPSS Modeler Server with the Services dialog box in the Windows Control Panel.

1. **Windows XP.** Open the Windows Start menu. Choose **Settings** and then **Control Panel**. Double-click **Administrative Tools** and then **Services**.  
**Windows 2003 or 2008.** Open the Windows Start menu. Choose **Control Panel**, then **Administrative Tools**, then **Services**.
2. Select the **IBM SPSS Modeler Server <nn.n>** service. You can now check its status, start or stop it, and edit startup parameters, as appropriate.

By default, the service is configured for automatic startup, which means that if you stop it, it will restart automatically when the computer is rebooted. When started this way, the service runs unattended, and the server computer can be logged off without affecting it.

#### To Start, Stop, and Check Status on UNIX

On UNIX, you start or stop IBM SPSS Modeler Server by running the *modelersrv.sh* script in the IBM SPSS Modeler Server installation directory.

1. Change to the IBM SPSS Modeler Server installation directory. For example, at a UNIX command prompt, type  
`cd /usr/modelersrv`  
where *modelersrv* is the IBM SPSS Modeler Server installation directory.
2. To start the server, at the command prompt, type  
`./modelersrv.sh start`
3. To stop the server, at the command prompt, type  
`./modelersrv.sh stop`
4. To check the status of IBM SPSS Modeler Server, at a UNIX command prompt, type  
`./modelersrv.sh list`

and look at the output, which is similar to what the UNIX `ps` command produces. The first process in the list is the IBM SPSS Modeler Server daemon process, and remaining processes are IBM SPSS Modeler sessions.

The IBM SPSS Modeler Server installation program includes a script (*auto.sh*) that configures your system to start the server daemon automatically at boot time. If you have run that script and then stop the server, the server daemon will restart automatically when the computer is rebooted. See the topic “Automatically Starting and Stopping IBM SPSS Modeler Server” on page 51 for more information.

## UNIX kernel limits

You must ensure that kernel limits on the system are sufficient for the operation of IBM SPSS Modeler Server. The data, memory, and file ulimits are particularly important and should be set to unlimited within the IBM SPSS Modeler Server environment. To do this:

1. Add the following commands to *modelersrv.sh*:

```
ulimit -d unlimited
ulimit -m unlimited
ulimit -f unlimited
```

2. Restart IBM SPSS Modeler Server.

---

## Handling Unresponsive Server Processes (UNIX Systems)

IBM SPSS Modeler Server processes may become unresponsive for several reasons, including situations where they make a system call or ODBC driver call that becomes blocked (call never returns, or takes a very long time to return). When UNIX processes enter this state, they can be cleaned up using the UNIX `kill` command (interrupts initiated by IBM SPSS Modeler client, or the closing of IBM SPSS Modeler client, will have no effect). A `kill` command is provided as an alternative to the normal stop command, and enables an administrator to use *modelersrv.sh* to easily issue the appropriate `kill` command.

On systems which are susceptible to the accumulation of unusable (“zombie”) server processes, we recommend that IBM SPSS Modeler Server is stopped and restarted at regular intervals, using the following sequence of commands:

```
cd modeler_server_install_directory
./modelersrv.sh stop
./modelersrv.sh kill
```

Those IBM SPSS Modeler processes that are ended using the `modelersrv.sh kill` command will leave behind temporary files (from the temporary directory) that will need to be removed manually. Temporary files may be left behind in some other situations too, including application crashes due to resource exhaustion, user interrupts, system crashes, or other reasons. Therefore we recommend that, as part of the process of restarting IBM SPSS Modeler Server at regular intervals, all remaining files are removed from the IBM SPSS Modeler temporary directory.

Once all server processes have been closed and temporary files have been removed, IBM SPSS Modeler Server can be safely restarted.

---

## Administration

IBM SPSS Modeler Server has a number of configurable options that control its behavior. You can set these options in two ways:

- Use the IBM SPSS Modeler Administration Console application, which is available free of charge to current IBM SPSS Modeler customers. See the topic “Using IBM SPSS Modeler Administration Console” on page 19 for more information.
- Use the *options.cfg* text file, located in the [*server install path*]/*config* directory. See the topic “Using the options.cfg File” on page 25 for more information.

We recommend that you install and use IBM SPSS Modeler Administration Console as your administration tool, rather than editing the *options.cfg* file. Editing the file requires access to the IBM SPSS



Modeler Server file system, but IBM SPSS Modeler Administration Console allows you to authorize anyone with a user account to adjust these options. Also, IBM SPSS Modeler Administration Console provides additional information about the server processes, allowing you to monitor usage and performance. Unlike when editing the configuration file, most configuration options can be changed without restarting IBM SPSS Modeler Server.

More information about using IBM SPSS Modeler Administration Console and the *options.cfg* file is provided in the following sections.

---

## Using IBM SPSS Modeler Administration Console

The Modeler Administration Console application provides a console user interface to monitor and configure your SPSS Modeler Server installations, and is available free-of-charge to current SPSS Modeler Server customers. The application can be installed only on Windows computers; however, it can administer a server installed on any supported platform.

Many of the options available through Modeler Administration Console can also be specified in the *options.cfg* file, which is located in the SPSS Modeler Server installation directory under */config*. However, the Modeler Administration Console application provides a shared graphical interface that allows you to connect, configure, and monitor multiple servers.

## Starting Modeler Administration Console

From the Windows Start menu, choose **[All] Programs**, then **IBM SPSS Collaboration and Deployment Services**, then **Deployment Manager**, then **IBM SPSS Modeler Administration Console <nn.n>**.

When you first run the application, you see empty Server Administration and Properties panes (unless you already have Deployment Manager installed with an IBM SPSS Collaboration and Deployment Services server connection already set up). After you configure Modeler Administration Console, the Server Administrator pane on the left displays a node for each SPSS Modeler Server that you want to administer. The right-hand pane shows the configuration options for the selected server. You must first set up a connection for each server that you want to administer.

## Configuring Access with Modeler Administration Console

Administrator access to SPSS Modeler Server through Modeler Administration Console is controlled with the *administrators* line in the *options.cfg* file, located in the SPSS Modeler Server installation directory under */config*. This line is commented out by default, so you must edit this line to allow access to specific people, or use *\** to allow access to all users, as shown in the following examples:

```
administrators, "*"
administrators, "jsmith,mjones,achavez"
```

- The line must begin with *administrators*, and the entries must be contained in quotation marks. Entries are case sensitive.
- Separate multiple user IDs with commas.
- For Windows accounts, do not use domain names.
- Use the asterisk with care. It allows anyone with a valid user account for IBM SPSS Modeler Server (which, in most cases, is anyone on the network) to log in and change the configuration options.

## Configuring Access with User Access Control

To use the Modeler Administration Console to make updates to a SPSS Modeler Server configuration installed on a Windows machine that has User Access Control (UAC) enabled you must have read, write, and execute permissions defined on the *config* directory and on the *options.cfg* file. These (NTFS) permissions must be defined at the specific user level and not at group level, this is due to the way that UAC and NTFS permissions interact.

## SPSS Modeler Server Connections

You must specify a connection to each SPSS Modeler Server on your network that you want to administer. You must then log in to each server. Although the server connection is remembered across Modeler Administration Console sessions, the login credentials are not. You must log in every time you start Modeler Administration Console.

### To Set Up a Server Connection

1. Ensure that the IBM SPSS Modeler Server service is started.
2. From the File menu, choose **New** and then **Administered Server Connection**.
3. On the first page of the wizard, enter a name for your server connection. The name is for your own use and should be something descriptive; for example, *Production Server*. Ensure that Type is set to **Administered IBM SPSS Modeler Server**, then click **Next**.
4. On the second page, enter the hostname or IP address of the server. If you have changed the port from the default, enter the port number. Click **Finish**. The new server connection is shown in the Server Administrator pane.

To perform administration tasks, you must now log in.

### To Log in to the Server

1. In the Server Administrator pane, double-click to select the server to which you want to log in.
2. In the Login dialog box, enter your credentials. (Use your user account for the server host.) Click **OK**.

If the login fails with the message **Unable to obtain administrator rights on server**, the most likely cause is that administrator access has not been configured correctly. See the topic “Configuring Access with Modeler Administration Console” on page 19 for more information.

If the login fails with the message **Failed to connect to server '<server>'**, make sure that the user ID and password are correct, then make sure that the IBM SPSS Modeler Server service is running. For example, under Windows, go to Control Panel > Administrative Tools > Services and check the entry for IBM SPSS Modeler Server. If the Status column does not show **Started**, select this line on the screen and click **Start**, then retry the login.

Once you log in to your IBM SPSS Modeler Server, two options are shown below the server name, Configuration and Monitoring. Double-click one of these options.

## SPSS Modeler Server Configuration

The Configuration pane shows configuration options for SPSS Modeler Server. Use this pane to change the options as desired. Click **Save** on the toolbar to save the changes. (Note that changing the port number requires a server restart in order to take effect.)

The options are described in the following sections, with the corresponding line in *options.cfg* given in parentheses for each option. Options that are visible only in *options.cfg* are described at the end of this section.

### Connections/Sessions

**Maximum number of connections.** (*max\_sessions*) Maximum number of server sessions at one time. A value of  $-1$  indicates no limit.

**Port number.** (*port\_number*) The port number for SPSS Modeler Server to listen on. Change if another application already uses the default. End users must know the port number in order to use SPSS Modeler Server.

## Analytic Server Connection

**Analytic Server URL.** (as\_url) The URL of the Analytic Server, including the IP address of the server, the port, and the tenant the SPSS Modeler Server installation is a member of.

**Prompt for Password.** (as\_prompt\_for\_password) Specify *N* if the SPSS Modeler Server is configured with the same authentication system for users and passwords as that used on Analytic Server; otherwise, *Y*.

## Data File Access

**Restrict access to only data file path.** (data\_files\_restricted) When set to *yes*, this option restricts data files to the standard data directory and those listed in the **Data File Path** below.

**Data file path.** (data\_file\_path) A list of additional directories to which clients are allowed to read and write data files. This option is ignored unless the **Restrict Access to Only Data File Path** option is turned on. Note that you should use forward slashes in all pathnames. On Windows, specify multiple directories using semicolons (for example, *[server install path]/data;c:/data;c:/temp*). On Linux and UNIX, use colons (:) instead of semicolons.

**Restrict access to only program files path.** (program\_files\_restricted) When set to *yes*, this option restricts program file access to the standard *bin* directory and those listed in the **Program Files Path** below.

**Program files path.** (program\_file\_path) A list of additional directories from which clients are allowed to execute programs. This option is ignored unless the **Restrict Access to Only Program Files Path** option is turned on. Note that you should use forward slashes in all pathnames. Specify multiple directories using semicolons.

**Maximum file size (MB).** (max\_file\_size) Maximum size of temporary and exported data files created during stream execution (does not apply to SAS and SPSS Statistics data files). A value of *-1* indicates no limit.

**Temporary directory.** (temp\_directory) The directory used to store temporary data files (cache files). Ideally, this directory should be on a separate high-speed drive or controller because speed of access to this directory may have a significant impact on performance. You may specify multiple temporary directories, separating each with a comma. These should be located on different disks; the first directory is used most often, and additional directories are used to store temporary work files when certain data preparation operations (such as sort) use parallelism during execution. Allowing each execution thread to use separate disks for temporary storage can improve performance. Use forward slashes in all path specifications. *Note:* Temporary files are generated in this directory during startup of SPSS Modeler Server. Ensure that you have the necessary access rights to this directory (for example, if the temporary directory is a shared network folder), otherwise SPSS Modeler Server startup will fail.

**Python Exe Path.** (python\_exe\_path) Full path to the python executable including the executable name. Note that **Restrict access to only program files path** may need to be set to *No* depending on the location where python is installed.

## Performance/Optimization

**Stream rewriting.** (stream\_rewriting\_enabled) Allows the server to optimize streams by rewriting them. For example, the server might push data reduction operations closer to the source node to minimize the size of the dataset as early as possible. Disabling this option is normally recommended only if the optimization causes an error or other unexpected results. This setting overrides the corresponding client optimization setting. If this setting is disabled in the server, then the client cannot enable it. But if it is enabled in the server, the client can choose to disable it.

**Parallelism.** (max\_parallelism) Describes the number of parallel worker threads that SPSS Modeler is allowed to use when running a stream. Setting this to 0 or any negative number causes IBM SPSS Modeler to match the number of threads to the number of available processors on the computer; the

default value for this option is -1. To turn off parallel processing (for machines with multiple processors), set this option to 1. To allow limited parallel processing, set it to a number smaller than the number of processors on your machine. Note that a hyperthreaded or dual-core processor is treated as two processors.

**Buffer size (bytes).** (`io_buffer_size`) Data files transferred from the server to the client are passed through a buffer of this number of bytes.

**Cache compression.** (`cache_compression`) An integer value in the range 0 to 9 that controls the compression of cache and other files in the server's temporary directory. Compression reduces the amount of disk space used, which can be important when space is limited, and improves performance by reducing the amount of disk activity needed to create and read the cache. Compression increases processor time, but this is almost always made up by the reduction in disk access time. Note that only certain caches, those accessed sequentially, can be compressed. This option does not apply to random-access caches, such as those used by the network training algorithms. A value of 0 disables compression entirely. Values from 1 upward provide increasing degrees of compression but with a corresponding cost in access time. The default value is 1; higher values are rarely needed except where disk space is at a premium.

**Memory usage multiplier.** (`memory_usage`) Controls the proportion of physical memory allocated for sorting and other in-memory caches. The default is 100, which corresponds to approximately 10% of physical memory. Increase this value to improve sort performance where free memory is available, but be careful of increasing it so high as to cause excessive paging.

**Modeling memory limit percentage.** (`modelling_memory_limit_percentage`) Controls the proportion of physical memory allocated for training neural net, Kohonen, and *k*-means models. The default is 25%. Increase this value to improve training performance where free memory is available, but be careful of increasing it so high as to cause excessive paging when data spills onto the disk.

**Allow modeling memory override.** (`allow_modelling_memory_override`) Enables or disables the **Optimize for Speed** option in certain modeling nodes. The default is enabled. This option allows the modeling algorithm to claim all available memory, bypassing the percentage limit option. You may want to disable this if you need to share memory resources on the server machine.

**Maximum and minimum server port.** (`max_server_port` and `min_server_port`) Specifies the range of port numbers that can be used for the additional socket connections between client and server that are required for interactive models and stream execution. These require the server to listen on another port; not restricting the range could cause problems for users on systems with firewalls. Default value for both is -1, meaning "no restriction." Thus, for example, to set the server to listen on port 8000 or above, you would set `min_server_port` to 8000 and `max_server_port` to -1.

Note that you must open additional ports over the main server port to open or execute a stream, and correspondingly more ports if you want to open or execute concurrent streams. This is required in order to capture feedback from the stream execution.

By default, IBM SPSS Modeler will use any open port that is available; if it does not find one (for example, if they are all closed by a firewall), an error is displayed when you execute the stream. To configure the range of ports, IBM SPSS Modeler will need two open ports (in addition to the main server port) available per concurrent stream, plus an additional port for each connected client that has an ODBC connection. To execute more concurrent streams, you can widen the range of port numbers and open up more ports on the firewall accordingly.

**Note:** Best practice dictates that the same ports should be used to communicate with both IBM SPSS Collaboration and Deployment Services and SPSS Modeler Client. These can be set as `max_server_port` and `min_server_port`.

**Note:** If you change these parameters, you need to restart SPSS Modeler Server for the change to take effect.

**Array fetch optimization.** (`sql_row_array_size`) Controls the way that SPSS Modeler Server fetches data from the ODBC datasource. The default value is 1, which fetches a single row at a time. Increasing this value causes the server to read the information in larger chunks, fetching the specified number of rows into an array. With some operating system/database combinations, this can result in improvements to the performance of SELECT statements.

## SQL

**Maximum SQL string length.** (`max_sql_string_length`) For a string imported from the database with SQL, the maximum number of characters that are guaranteed to be passed successfully. Depending on the operating system, string values longer than this may be truncated on the right without warning. The valid range is between 1 and 65,535 characters. This property also applies to the Database export node.

**Note:** The default value for this parameter is 2048. If the text you are analyzing is longer than 2048 characters (for example, this may occur if using the SPSS Modeler Text Analytics Web Feed node) we recommend increasing this value if working in native mode otherwise your results may be truncated. If you are using a database and user-defined functions (UDF), this restriction does not occur; this can account for differences in results between native and UDF modes.

**Automatic SQL generation.** (`sql_generation_enabled`) Allows automatic SQL generation for streams, which may substantially improve performance. The default is enabled. Disabling this option is recommended only if the database is not able to support queries submitted by SPSS Modeler Server. Note that this setting overrides the corresponding client optimization setting; also note that for purposes of scoring, SQL generation must be enabled separately for each modeling node regardless of this setting. If this setting is disabled in the server, then the client cannot enable it. But if it is enabled in the server, the client can choose to disable it.

**Default SQL string length.** (`default_sql_string_length`). Specifies the default width of string columns that will be created within database cache tables. String fields in database cache tables will be created with a default width of 255 if there is no upstream type information. If you have wider values than this in your data, either instantiate an upstream Type node with those values, or set this parameter to a value that is large enough to accommodate those string values.

**Support SQL type logging.** (`supported_sql_type_logging`). For debug purposes only; do not modify unless instructed to do so by a technical-support representative.

**Enable Database UDF.** (`db_udf_enabled`). If set to Y (default), causes the SQL generation option to generate user-defined function (UDF) SQL instead of pure SPSS Modeler SQL. UDF SQL usually outperforms pure SQL.

## SSL

**Enable SSL.** (`ssl_enabled`) Enables SSL encryption for connections between SPSS Modeler and SPSS Modeler Server.

**Keystore.** (`ssl_keystore`) The SSL key database file to be loaded when the server starts (either a full path or a relative path to the SPSS Modeler installation directory).

**Keystore stash file.** (`ssl_keystore_stash_file`) The name of the key database password stash file to be loaded when the server starts up (either a full path or a relative path to the SPSS Modeler installation directory).

**Keystore label.** (`ssl_keystore_label`) Label for the specified certificate.

**Note:** To use the Administration Console with a server setup for SSL, you must import any certificates required by SPSS Modeler Server into the Deployment Manager trust store (under `../jre/lib/security`).

## Coordinator of Processes Configuration

**Host.** (`cop_host`) The hostname or IP address of the Coordinator of Processes service. The default “`spsscop`” is a vanity name which administrators can choose to add as an alias for the IBM SPSS Collaboration and Deployment Services host in DNS.

**Port number.** (`cop_port_number`) The port number of the Coordinator of Processes service. The default, 8080, is the IBM SPSS Collaboration and Deployment Services default.

**Context root.** (`cop_context_root`) The URL of the Coordinator of Processes service.

**Login name.** (`cop_user_name`) The user name for authentication to the Coordinator of Processes service. This is an IBM SPSS Collaboration and Deployment Services login name so may include a security-provider prefix (for example: `ad/jsmith`).

**Password.** (`cop_password`) The password for authentication to the Coordinator of Processes service.

**Enabled.** (`cop_enabled`) Determines whether the server should attempt to register with the Coordinator of Processes. The default is *not* to register because the administrator should choose which services are advertised through the Coordinator of Processes.

**Server name.** (`cop_service_name`) The name of this SPSS Modeler Server instance; the default is the host name.

**Description.** (`cop_service_description`) A description of this instance.

**Update interval (min).** (`cop_update_interval`) The number of minutes between keep-alive messages; the default is 2.

**Weight.** (`cop_service_weight`) The weight of this instance, specified as an integer between 1 and 10. A higher weight attracts more connections. The default is 1.

**Service host.** (`cop_service_host`) The fully-qualified host name of the IBM SPSS Modeler Server host. The default of the host name is derived automatically; the administrator can override the default for multi-homed hosts.

## Options Visible in `options.cfg`

administrators. Specify the usernames of those users to whom you want to grant administrator access. See the topic “Configuring Access with Modeler Administration Console” on page 19 for more information.

`allow_config_custom_overrides`. Do not modify unless instructed to do so by a technical-support representative.

`fips_encryption`. Enables FIPS compliant encryption; the default is N.

`max_transfer_size`. For internal system use only; do not modify.

`shell`. (UNIX servers only) Overrides the default setting for the UNIX shell, for example `shell`, `"/usr/bin/ksh"`. By default, IBM SPSS Modeler uses the shell defined in the user profile of the user who is connecting to IBM SPSS Modeler Server.

`start_process_as_login_user`. Set this to Y if you are running SPSS Modeler Server with a private password database.

`time_ecode_execution_log`. Set this to Y if you want the *message.log* file (in the *log* folder of the SPSS Modeler Server installation) to include individual execution times for each node in the stream.

`use_bigint_for_count`. When the number of the records to be counted is larger than a normal integer ( $2^{32}-1$ ) can hold, set this option to Y. When this option is set to Y, and a stream is connected to either DB2 or SQL Server, `COUNT_BIG()` will be used where a record count is needed (for example, the **Record\_Count** field generated by the Aggregate node).

## SPSS Modeler Server Monitoring

The monitoring pane of Modeler Administration Console shows a snapshot of all processes running on the SPSS Modeler Server computer, similar to the Windows Task Manager. To activate the monitoring pane, double-click the Monitoring node beneath the desired server in the Server Administrator pane. This populates the pane with a current snapshot of data from the server. The data refreshes at the rate shown (one minute by default). To refresh the data manually, click the **Refresh** button. To show only SPSS Modeler Server processes in this list, click the **Filter out non-SPSS Modeler processes** button.

## Using the options.cfg File

The *options.cfg* file is located in the *[server install path]/config* directory. Each setting is represented by a comma-separated name-value pair, where the **name** is the name of the option and the **value** is the value for the option. Pound (hash) signs (#) indicate comments.

*Note:* Most configuration options can be changed using IBM SPSS Modeler Administration Console, rather than this configuration file, but there are a few exceptions. See the topic “Options Visible in options.cfg” on page 24 for more information.

By using IBM SPSS Modeler Administration Console, you can avoid server restarts for all options except the server port. See the topic “Using IBM SPSS Modeler Administration Console” on page 19 for more information.

### Configuration Options That Can Be Added to the Default File

By default, in-database caching is enabled with IBM SPSS Modeler Server. You can disable this feature by adding the following line to the *options.cfg* file:

```
enable_database_caching, N
```

Doing so causes temporary files to be created on the server and not in the database.

To View or Change the IBM SPSS Modeler Server Configuration Options:

1. Open the *options.cfg* file with a text editor.
2. Locate the options of interest. For a full list of options, see “SPSS Modeler Server Configuration” on page 20.
3. Edit the values, as appropriate. Note that all pathname values must use a forward slash (/) rather than a backslash as the pathname separator.
4. Save the file.
5. Stop and restart IBM SPSS Modeler Server so that the changes will take effect. See the topic “Starting and Stopping IBM SPSS Modeler Server” on page 17 for more information.

## Closing Unused Database Connections

By default, IBM SPSS Modeler caches at least one connection to a database once that connection has been accessed. The database session is held open even when streams requiring database access are not being executed.

Caching database connections can improve execution times by removing the need for IBM SPSS Modeler to reconnect to the database each time a stream is executed. However, in some environments, it is important for applications to release database resources as quickly as possible. If too many IBM SPSS Modeler sessions maintain connections to the database that are no longer used, database resources may become exhausted.

You can avoid this possibility by turning off the IBM SPSS Modeler option `cache_connection` in a custom database configuration file. Doing so can also make IBM SPSS Modeler more resilient to faults in the database connection (such as timeouts) that can occur when connections are used over a long period of time by an IBM SPSS Modeler session.

To cause unused database connections to be closed:

1. Locate the `[server install path]/config` directory.
2. Add the following file (or open it, if it already exists):  
`odbc-custom-properties.cfg`
3. Add the following line to the file:  
`cache_connection, N`
4. Save and close the file.
5. Restart IBM SPSS Modeler Server to cause the changes to take effect.

---

## Using SSL to secure data transfer

Secure Sockets Layer (SSL) is a protocol for encrypting data transferred between two computers. SSL ensures that communication between the computers is secure. SSL can encrypt the authentication of a username/password and the contents of an exchange between a server and client.

### How SSL works

SSL relies on the server's public and private keys, in addition to a public key certificate that binds the server's identity to its public key.

1. When a client connects to a server, the client authenticates the server with the public key certificate.
2. The client then generates a random number, encrypts the number with the server's public key, and sends the encrypted message back to the server.
3. The server decrypts the random number with its private key.
4. From the random number, both the server and client create the session keys used for encrypting and decrypting subsequent information.

The public key certificate is typically signed by a certificate authority. Certificate authorities, such as VeriSign and Thawte, are organizations that issue, authenticate, and manage security credentials contained in the public key certificates. Essentially, the certificate authority confirms the identity of the server. The certificate authority usually charges a monetary fee for a certificate, but self-signed certificates can also be generated.

### Securing client/server and server-server communications with SSL

The main steps in securing client/server and server-server communications with SSL are:

1. Obtain and install the SSL certificate and keys.
2. Enable and configure SSL in the server administration application (IBM SPSS Collaboration and Deployment Services Deployment Manager).
3. If necessary, install unlimited strength encryption on the client computers.
4. Instruct users to enable SSL when connecting to the server.



**Note:** Occasionally a server product acts as a client. An example is IBM SPSS Statistics Server connecting to the IBM SPSS Collaboration and Deployment Services Repository. In this case, IBM SPSS Statistics Server is the *client*.

## Obtaining and installing SSL certificate and keys

The first steps you must follow to configure SSL support are:

1. Obtain an SSL certificate and key file. There are three ways you can do this:
  - Purchase them from a public certificate authority (such as VeriSign, Thawte, or Entrust). The public certificate authority (CA) signs the certificate to verify the server that uses it.
  - Generate the key and certificate files with a third-party certificate authority. If this approach is taken then the third-party CA's root certificate must be imported into the client and server keystore files. See the topic "Importing a third-party root CA certificate" on page 28 for more information.
  - Generate the key and certificate files with an internal self-signed certificate authority. The steps to do this are:
    - a. Prepare a key database. See the topic "Creating an SSL key database" on page 28 for more information.
    - b. Create the self-signed certificate. See the topic "Creating a self-signed SSL certificate" on page 28 for more information.
2. Copy the .kdb and .sth files created in step 1 into a directory to which the IBM SPSS Modeler Server has access and specify the path to that directory in the *options.cfg* file.

**Note:** Use forward slashes as separators in the directory path.

3. Set the following parameters in the *options.cfg* file:
  - `ssl_enabled, Y`
  - `ssl_keystore, "<filename>.kdb"` where *<filename>* is the name of your key database.
  - `ssl_keystore_stash_file, "<filename>.sth"` where *<filename>* is the name of the key database password stash file.
  - `ssl_keystore_label, <label>` where *<label>* is the label of your certificate.
4. For self-signed or third-party certificates install the certificate on client systems. For purchased public CA certificates, this step is not required. Ensure that access permissions deny casual browsing of the directory that contains the certificate. See the topic "Installing a self-signed SSL certificate" on page 28 for more information.

**Configuring the environment to run GSKit:** The GSKCapiCmd is a non-Java-based command-line tool, and Java™ does not need to be installed on your system to use this tool; it is located in the *<Modeler installation directory>/bin* folder. The process to configure your environment to run IBM Global Security Kit (GSKit) varies depending on the platform in use.

To configure for Linux/Unix, add the shared libraries directory *<Modeler installation directory>/lib* to your environment:

```
$export <Shared library path environment variable>=<modeler_server_install_path>/lib
$export PATH=$PATH:<modeler_server_install_path>/bin
```

The shared library path variable name depends on your platform:

- AIX uses the variable name: LIBPATH
- HP-UX uses the variable name: SHLIB\_PATH
- Linux, Solaris use the variable name: LD\_LIBRARY\_PATH

For example, to set the environment on Linux, use:

```
$export LD_LIBRARY_PATH=/path/to/gskit/lib
$export PATH=$PATH:/path/to/gskit/bin
```

**Creating an SSL key database:** Use the GSKCapiCmd tool to create your key database. Before using the tool you must configure your environment; see the topic “Configuring the environment to run GSKit” on page 27 for more information

To create the key database, run GSKit and enter the following command:

```
gsk<ver>capicmd[_64] -keydb -create -populate -db <filename>.kdb -pw <password> -stash
```

where <ver> is the GSKit version number, <filename> is the name you want to use for the key database file, and <password> is the password for the key database.

The *-stash* option creates a stash file at the same path as the key database, with a file extension of .sth. GSKit uses the stash file to obtain the password to the key database so that it doesn't have to be entered on the command line each time.

**Note:** You should use strong file system protection on the .sth file.

**Creating a self-signed SSL certificate:** To generate a self-signed certificate and store it in the key database, use the following command:

```
gsk<ver>capicmd[_64] -cert -create -db <filename>.kdb -stashed -dn "CN=myserver,OU=mynetwork,O=mycompany,C=mycountry" -expire <Number of days certificate is valid> -label <label> -default_cert yes
```

where <ver> is the GSKit version number, <filename> is the name of the key database file, <Number of days certificate is valid> is the physical number of days that the certificate is valid, and <label> is a descriptive label to help you identify the file (for example, you could use a label such as: *myselfsigned*).

**Installing a self-signed SSL certificate:** For the client machines that connect to your server using SSL, you must distribute the public part of the certificate to the clients so that it can be stored in their key databases. To do this:

1. Extract the public part to a file using the following command:  

```
gsk8capicmd -cert -extract -db <filename>.kdb -stashed -label <label> -format ascii -target mycert.arm
```
2. Distribute *mycert.arm* to the clients. It should be copied to their *jre/bin* directory.
3. Add the new certificate to the clients' key database using the following command:  

```
keytool -import -alias <label> -keystore cacerts -file mycert.arm
```

If prompted for a password, use: *changeit*. The keytool is located in the <Modeler installation directory>\jre\bin directory.

4. Copy the newly generated cacerts files to the *jre\lib\security* directory to replace the existing version.

**Importing a third-party root CA certificate:** Instead of purchasing a certificate from a well known certificate authority (CA) or creating a self-signed certificate, you can use a third-party certificate authority to sign your server certificates. The client and server must have access to the third-party CA's root certificate to verify the server certificates that are signed by the third-party CA. To do this:

1. Obtain the third-party CA root certificate. The process for this varies depending on the third-party CA's procedures. Third-party CAs often make their root certificates available for download.
2. Add the certificate to the servers' key database using the following command:  

```
gsk<ver>capicmd[_64] -cert -add -db <filename>.kdb -stashed -label <label> -file <ca_certificate>.cert -format binary -t
```
3. Add the certificate to the clients' key database using the following command:

```
C:> cd <Modeler Client installation path>\jre\bin  
C:> keytool -import -keystore ..\lib\security\cacerts -file <ca_certificate>.cert -alias <label>
```

If prompted for a password, use: *changeit*. The keytool is located in the <Modeler installation directory>\jre\bin directory.

4. Validate the server's key database with the root CA certificate using the following command:

```
gsk<ver>capicmd[_64] -cert -validate -db <filename>.kdb -stashed -label <label>
```

A successful validation is indicated by the returned message: OK.

**Note:** The commands explained above use a third-party CA root certificate that is in a binary format. If the certificate is in an ASCII format, use the `-format ascii` option.

The `-db` parameter specifies the name of the key database into which you import the third-party CA root certificate.

The `-label` parameter specifies the label to use for the third-party CA root certificate inside the key database file. The label you use here can be anything because it does not have any relation to the labels used in the IBM SPSS Modeler options.cfg file.

The `-file` parameter specifies the file that contains the third-party CA root certificate

## Enable and Configure SSL in the IBM SPSS Collaboration and Deployment Services Deployment Manager

1. If installing a self-signed SSL certificate, copy the cacerts file that you created to the *<Deployment Manager installation directory>\jre\lib\security* directory. See the topic “Installing a self-signed SSL certificate” on page 28 for more information.
2. Start the server administration application ( IBM SPSS Collaboration and Deployment Services Deployment Manager ) and connect to the server.
3. On the configuration page, set **Secure Sockets Layer** to Yes.
4. In **SSL Public Key File**, specify the full path to the public key file.
5. In **SSL Private Key File**, specify the full path to the private key file.  
*Note:* If the public and private keys are stored in one file, specify the same file in **SSL Public Key File** and **SSL Private Key File**.
6. From the menus choose:  
**File > Save**
7. Restart the server service or daemon. When you restart, you will be prompted for the SSL password. On Windows, you can select **Remember this password** to store the password securely. This option eliminates the need to enter the password every time the server is started.

## Installing unlimited strength encryption

The Java Runtime Environment shipped with the product has US export-strength encryption enabled. For enhanced security of your data, upgrading to unlimited-strength encryption is recommended.

### IBM J9

1. Download the Java Cryptography Extension (JCE) Unlimited Strength Jurisdiction Policy Files for your version of the SDK from the IBM.com website.
2. Extract the unlimited jurisdiction policy files that are packaged in the compressed file. The compressed file contains a *US\_export\_policy.jar* file and a *local\_policy.jar* file. In your WebSphere Application Server installation, go to the *\$JAVA\_HOME/jre/lib/security* directory and back up your *US\_export\_policy.jar* and *local\_policy.jar* files.
3. Replace your *US\_export\_policy.jar* and *local\_policy.jar* files with the two files that you downloaded.
4. Enable security in the WebSphere Application Server administration console. Make sure that all node agents within the cell are active beforehand. For more information, see WebSphere documentation. Note that you must select an available realm definition from the list in **Security > Secure administration, applications, and infrastructure**, and then click **Set as current** so that security is enabled upon a server restart.
5. Log off the administrative console.
6. Stop the server.

7. Restart the server.

### Sun Java

1. Download the Java Cryptography Extension (JCE) Unlimited Strength Jurisdiction Policy Files for your version of the SDK from Sun Java website.
2. Extract the downloaded file.
3. Copy the two *.jar* files *local\_policy.jar* and *US\_export\_policy.jar* into *<installation folder>/jre/lib/security*, where *<installation folder>* is the folder in which you installed the product.

### Instructing users to enable SSL

When users connect to the server through a client product, they need to enable SSL in the dialog box for connecting to the server.

---

## Server Log

IBM SPSS Modeler Server keeps a record of its important actions in a log file called *server\_logging.log*. On UNIX this file is located in a log folder in the installation directory, on Windows this file is located in *%ALLUSERSPROFILE%/IBM/SPSS/Modeler Server/<version>/log*. When IBM SPSS Modeler Server is started, the log file is updated to indicate the time of the start and the port number it is listening on. From that point forward, the log file is updated with the result whenever a client attempts to connect to the server.

The server writes the following events to the log file by default:

- Server started
- Server stopped
- Connection accepted
- Connection refused
- Session started
- Session stopped
- System error
- Login succeeded
- Login failed
- Database login succeeded
- Database login failed
- Database logout
- File open/create
- File close

---

## Chapter 5. Performance Overview

Real performance in analyzing data is affected by a number of factors, from server and database configuration to the ordering of individual nodes within a stream. In general, you can obtain the best performance by doing the following:

- Store your data in a DBMS, and use SQL generation and optimization whenever possible.
- Use hardware that meets or exceeds the recommendations given in Chapter 2, “Architecture and Hardware Recommendations,” on page 5.
- Ensure that the client and server performance and optimization settings are properly configured. Note that when SPSS Modeler is connected to an SPSS Modeler Server installation, the server performance and optimization settings override the client equivalents.
- Design streams for maximum performance.

More information about each of these performance factors is available in the following sections.

---

### Server Performance and Optimization Settings

Certain IBM SPSS Modeler Server settings can be configured to optimize performance. You can adjust these settings using the IBM SPSS Modeler Administration Console application, which is included with IBM SPSS Modeler Server. See the topic “Using IBM SPSS Modeler Administration Console” on page 19 for more information.

The settings are grouped under the heading **Performance and Optimization** in the IBM SPSS Modeler Administration Console configuration window. The settings are preconfigured for optimal performance for most installations. However, you may need to adjust them depending on your particular hardware, the size of your data sets, and the contents of your streams. See the topic “Performance/Optimization” on page 21 for more information.

---

### Client Performance and Optimization Settings

The client performance and optimization settings are available from the Options tab of the Stream Properties dialog box. To display these options, choose the following from the client menu.

**Tools > Stream Properties > Options > Optimization**

You can use the Optimization settings to optimize stream performance. Note that the performance and optimization settings on IBM SPSS Modeler Server (if used) override any equivalent settings in the client. If these settings are disabled in the server, then the client cannot enable them. But if they are enabled in the server, the client can choose to disable them.

*Note:* Database modeling and SQL optimization require that IBM SPSS Modeler Server connectivity be enabled on the IBM SPSS Modeler computer. With this setting enabled, you can access database algorithms, push back SQL directly from IBM SPSS Modeler, and access IBM SPSS Modeler Server. To verify the current license status, choose the following from the IBM SPSS Modeler menu.

**Help > About > Additional Details**

If connectivity is enabled, you see the option **Server Enablement** in the License Status tab.

See the topic “Connecting to IBM SPSS Modeler Server” on page 11 for more information.

*Note:* Whether SQL pushback and optimization are supported depends on the type of database in use. For the latest information on which databases and ODBC drivers are supported and tested for use with IBM SPSS Modeler 16, see the corporate Support site at <http://www.ibm.com/support>.

**Enable stream rewriting.** Select this option to enable stream rewriting in IBM SPSS Modeler. Four types of rewriting are available, and you can select one or more of them. Stream rewriting reorders the nodes in a stream behind the scenes for more efficient operation, without altering stream semantics.

- **Optimize SQL generation.** This option enables nodes to be reordered within the stream so that more operations can be pushed back using SQL generation for execution in the database. When it finds a node that cannot be rendered into SQL, the optimizer will look ahead to see if there are any downstream nodes that can be rendered into SQL and safely moved in front of the problem node without affecting the stream semantics. Not only can the database perform operations more efficiently than IBM SPSS Modeler, but such pushbacks act to reduce the size of the data set that is returned to IBM SPSS Modeler for processing. This, in turn, can reduce network traffic and speed stream operations. Note that the **Generate SQL** check box must be selected for SQL optimization to have any effect.
- **Optimize CLEM expression.** This option enables the optimizer to search for CLEM expressions that can be preprocessed before the stream is run, in order to increase the processing speed. As a simple example, if you have an expression such as  $\log(\text{salary})$ , the optimizer would calculate the actual salary value and pass that on for processing. This can be used both to improve SQL pushback and IBM SPSS Modeler Server performance.
- **Optimize syntax execution.** This method of stream rewriting increases the efficiency of operations that incorporate more than one node containing IBM SPSS Statistics syntax. Optimization is achieved by combining the syntax commands into a single operation, instead of running each as a separate operation.
- **Optimize other execution.** This method of stream rewriting increases the efficiency of operations that cannot be delegated to the database. Optimization is achieved by reducing the amount of data in the stream as early as possible. While maintaining data integrity, the stream is rewritten to push operations closer to the data source, thus reducing data downstream for costly operations, such as joins.

**Enable parallel processing.** When running on a computer with multiple processors, this option allows the system to balance the load across those processors, which may result in faster performance. Use of multiple nodes or use of the following individual nodes may benefit from parallel processing: C5.0, Merge (by key), Sort, Bin (rank and tile methods), and Aggregate (using one or more key fields).

**Generate SQL.** Select this option to enable SQL generation, allowing stream operations to be pushed back to the database by using SQL code to generate execution processes, which may improve performance. To further improve performance, **Optimize SQL generation** can also be selected to maximize the number of operations pushed back to the database. When operations for a node have been pushed back to the database, the node will be highlighted in purple when the stream is run.

- **Database caching.** For streams that generate SQL to be executed in the database, data can be cached midstream to a temporary table in the database rather than to the file system. When combined with SQL optimization, this may result in significant gains in performance. For example, the output from a stream that merges multiple tables to create a data mining view may be cached and reused as needed. With database caching enabled, simply right-click any nonterminal node to cache data at that point, and the cache is automatically created directly in the database the next time the stream is run. This allows SQL to be generated for downstream nodes, further improving performance. Alternatively, this option can be disabled if needed, such as when policies or permissions preclude data being written to the database. If database caching or SQL optimization is not enabled, the cache will be written to the file system instead.
- **Use relaxed conversion.** This option enables the conversion of data from either strings to numbers, or numbers to strings, if stored in a suitable format. For example, if the data is kept in the database as a string, but actually contains a meaningful number, the data can be converted for use when the pushback occurs.

*Note:* Due to minor differences in SQL implementation, streams run in a database may return slightly different results from those returned when run in IBM SPSS Modeler. For similar reasons, these differences may also vary depending on the database vendor.

---

## Database Usage and Optimization

**Database server.** If possible, create a dedicated database instance for data mining so that the production server is not impacted by IBM SPSS Modeler queries. SQL statements generated by IBM SPSS Modeler can be demanding—multiple tasks on the IBM SPSS Modeler Server machine can be executing SQL in the same database.

**In-database mining.** Many database vendors provide data mining extensions for their products. These extensions allow data mining activities (such as model-building or scoring) to run within the database server, or within a separate dedicated server. IBM SPSS Modeler’s in-database mining features complement and extend its SQL generation capability, providing a way to drive the vendor-specific database extensions. In some cases, taking this approach avoids the potentially expensive overhead of data transfer between IBM SPSS Modeler and the database. Database caching can further increase the benefits. For more information, see the file *DatabaseMiningGuide.pdf*, available under the `\Documentation` folder on the IBM SPSS Modeler installation disk.

## SQL Optimization

For best performance, you should always try to maximize the amount of SQL generated to exploit the performance and scalability of the database. Only the parts of the stream that cannot be compiled to SQL should be executed within IBM SPSS Modeler Server. For more information, see Chapter 6, “SQL Optimization,” on page 35.

### Uploading File-Based Data

Data that is not stored in a database cannot benefit from SQL optimization. If the data you want to analyze is not already in a database, you can upload it using a Database Output node. You can also use this node to store intermediate data sets from data preparation and the results of deployment.

IBM SPSS Modeler can interface with the external loaders for many common database systems. Several scripts are included with the software and are available (with documentation) in the `/scripts` subdirectory under your IBM SPSS Modeler installation folder.

The following table shows the potential performance benefit of bulk-loading. The figures show the elapsed time to export 250,000 records and 21 fields to an Oracle database. The external loader was Oracle’s `sqlldr` utility.

*Table 1. Performance benefit of bulk-loading*

Export option	Time (in seconds)
Default (ODBC)	409
Bulk-load via ODBC	52
Bulk-load via external loader	33





---

## Chapter 6. SQL Optimization

One of the most powerful capabilities of IBM SPSS Modeler is the ability to perform many data preparation and mining operations directly in the database. By generating SQL code that can be pushed back to the database for execution, many operations, such as sampling, sorting, deriving new fields, and certain types of graphing, can be performed in the database rather than on the IBM SPSS Modeler or IBM SPSS Modeler Server computer. When you are working with large datasets, these **pushbacks** can dramatically enhance performance in several ways:

- By reducing the size of the result set to be transferred from the DBMS to IBM SPSS Modeler. When large result sets are read through an ODBC driver, network I/O or driver inefficiencies may result. For this reason, the operations that benefit most from SQL optimization are row and column selection and aggregation (Select, Sample, Aggregate nodes), which typically reduce the size of the dataset to be transferred. Data can also be cached to a temporary table in the database at critical points in the stream (after a Merge or Select node, for example) to further improve performance.
- By making use of the performance and scalability of the database. Efficiency is increased because a DBMS can often take advantage of parallel processing, more powerful hardware, more sophisticated management of disk storage, and the presence of indexes.

Given these advantages, IBM SPSS Modeler is designed to maximize the amount of SQL generated by each stream so that only those operations that cannot be compiled to SQL are executed by IBM SPSS Modeler Server. Because of limitations in what can be expressed in standard SQL (SQL-92), however, certain operations may not be supported. See the topic “Tips for Maximizing SQL Generation” on page 39 for more information.

**Note:** Because of minor differences in SQL implementation, streams executed in a database may return slightly different results when executed in IBM SPSS Modeler. These differences may also vary depending on the database vendor, for similar reasons. For example, depending on the database configuration for case sensitivity in string comparison and string collation, IBM SPSS Modeler streams executed using SQL pushback may produce different results from those executed without SQL pushback. Contact your database administrator for advice on configuring your database. To maximize compatibility with IBM SPSS Modeler, database string comparisons should be case sensitive.

*Note:* Database modeling and SQL optimization require that IBM SPSS Modeler Server connectivity be enabled on the IBM SPSS Modeler computer. With this setting enabled, you can access database algorithms, push back SQL directly from IBM SPSS Modeler, and access IBM SPSS Modeler Server. To verify the current license status, choose the following from the IBM SPSS Modeler menu.

### Help > About > Additional Details

If connectivity is enabled, you see the option **Server Enablement** in the License Status tab.

See the topic “Connecting to IBM SPSS Modeler Server” on page 11 for more information.

**Note:** When streams are executed in a Netezza database, date and time details are taken from that database. This may differ from your local or IBM SPSS Modeler Server date and time if, for example, the database is on a machine that is located in a different country or time zone.

### Database requirements

For the latest information on which databases and ODBC drivers are supported and tested for use with IBM SPSS Modeler, see the product compatibility matrices on the corporate Support site at <http://www.ibm.com/support>.

Note that you may gain additional performance improvements by using database modeling.

## ODBC driver setup

To ensure that time details (such as HH:MM:SS) are processed correctly when using SQL 2012 on Windows 8 32bit systems, when setting up your ODBC SQL Server Wire Protocol Driver, you should select both the **Enable Quoted Identifiers** and **Fetch TWFS as Time** options.

---

## How SQL Generation Works

The initial fragments of a stream leading from the database source nodes are the main targets for SQL generation. When a node is encountered that cannot be compiled to SQL, the data are extracted from the database and subsequent processing is performed by IBM SPSS Modeler Server.

During stream preparation and prior to execution, the SQL generation process happens as follows:

- The server reorders streams to move downstream nodes into the “SQL zone” where it can be proven safe to do so. (This feature can be disabled on the server.)
- Working from the source nodes toward the terminal nodes, SQL expressions are constructed incrementally. This phase stops when a node is encountered that cannot be converted to SQL or when the terminal node (for example, Table node or Graph node) is converted to SQL. At the end of this phase, each node is labeled with an SQL statement if the node and its predecessors have an SQL equivalent.
- Working from the nodes with the most complicated SQL equivalents back toward the source nodes, the SQL is checked for validity. The SQL that was successfully validated is chosen for execution.
- Nodes for which all operations have generated SQL are highlighted in purple on the stream canvas. Based on the results, you may want to further reorganize your stream where appropriate to take full advantage of database execution. See the topic “Tips for Maximizing SQL Generation” on page 39 for more information.

### Where Improvements Occur

SQL optimization improves performance in a number of data operations:

- **Joins (merge by key).** Join operations can increase optimization within databases.
- **Aggregation.** The Aggregate, Distribution, and Web nodes all use aggregation to produce their results. Summarized data uses considerably less bandwidth than the original data.
- **Selection.** Choosing records based on certain criteria reduces the quantity of records.
- **Sorting.** Sorting records is a resource-intensive activity that is performed more efficiently in a database.
- **Field derivation.** New fields are generated more efficiently in a database.
- **Field projection.** IBM SPSS Modeler Server extracts only fields that are required for subsequent processing from the database, which minimizes bandwidth and memory requirements. The same is also true for superfluous fields in flat files: although the server must read the superfluous fields, it does not allocate any storage for them.
- **Scoring.** SQL can be generated from decision trees, rulesets, linear regression, and factor-generated models.

## SQL Generation Example

The following stream joins three database tables by key operations and then performs an aggregation and sort.

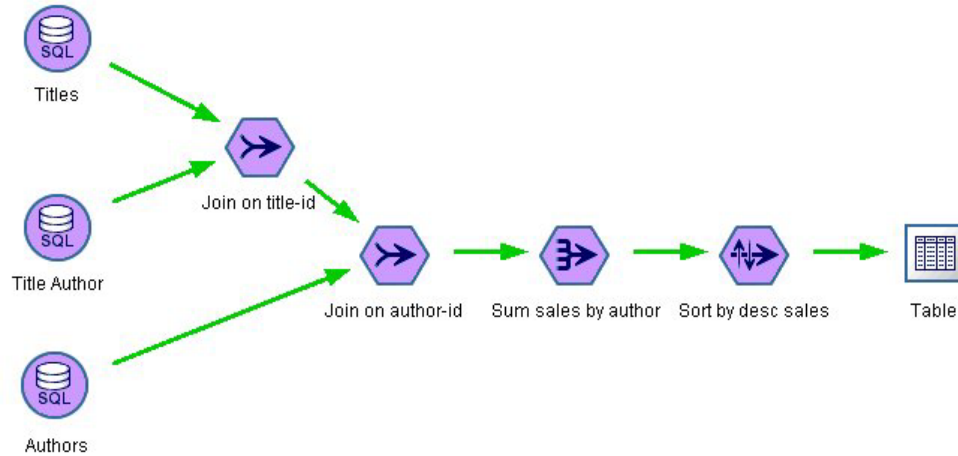


Figure 3. Optimized stream with purple nodes indicating SQL pushbacks (operations performed in database)

### Generated SQL

The generated SQL for this stream is as follows:

```

SELECT
    T2. au_lname AS C0,
    T2. au_fname AS C1,
    SUM({fn CONVERT(T0. ytd_sales ,SQL_BIGINT)}) AS C2
FROM
    dbo . titles T0,
    dbo . titleauthor T1,
    dbo . authors T2
WHERE
    (T0. title_id = T1. title_id )
    AND (T1. au_id = T2. au_id )
GROUP BY T2. au_lname ,T2. au_fname
ORDER BY 3 DESC

```

### Executing the Stream

When the stream is terminated with a database export node, it is possible for the whole stream to be executed in the database.

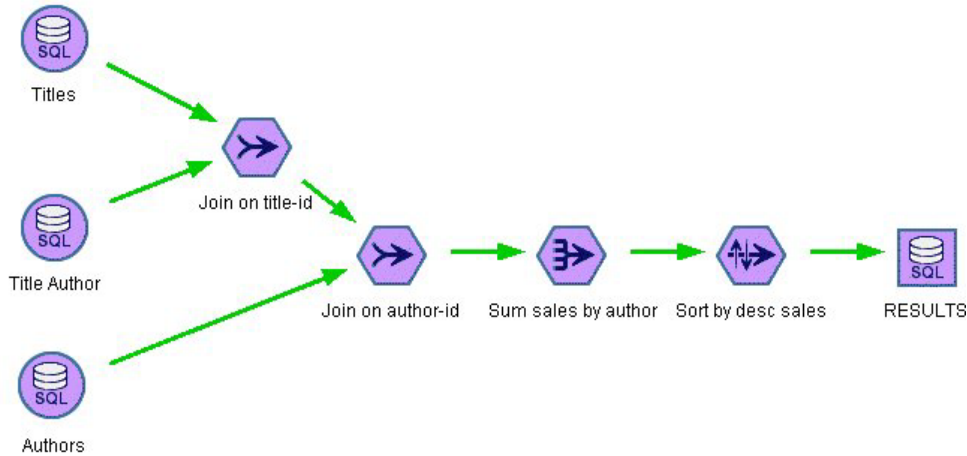


Figure 4. Entire stream executed in database

## Configuring SQL Optimization

1. Install an ODBC driver and configure a data source for the database you want to use. See the topic “Data Access” on page 8 for more information.
2. Create a stream that uses a source node to pull data from that database.
3. Check to be sure that SQL generation is enabled on the client and server, if applicable. It is enabled by default for both.

### To Enable SQL Optimization on the Client

1. From the Tools menu, choose **Stream Properties > Options**.
2. Click the Optimization tab. Select **Generate SQL** to enable SQL optimization. Optionally, you can select other settings to improve performance. See the topic “Client Performance and Optimization Settings” on page 31 for more information.

### To Enable SQL Optimization on the Server

Because server settings override any specifications made on the client, the server configuration settings **Stream Rewriting** and **Automatic SQL Generation** must both be turned on. For more information about how to change IBM SPSS Modeler Server settings, see the section “Performance/Optimization” on page 21. Note that if these settings are disabled in the server, then the client cannot enable them. But if they are enabled in the server, the client can choose to disable them.

### To Enable Optimization When Scoring Models

For purposes of scoring, SQL generation must be enabled separately for each modeling node, regardless of any server or client-level settings. This is done because some models generate extremely complex SQL expressions that may not be evaluated effectively within the database. The database may report errors when trying to execute the generated SQL, due to the size or complexity of the SQL.

A certain amount of trial-and-error may be needed to determine whether SQL generation improves performance for a given model. This is done on the Settings tab after adding a generated model to a stream.

---

## Previewing Generated SQL

You can preview generated SQL in the message log before executing it in the database. This may be useful for debugging purposes, and it allows you to export the generated SQL to edit or run in the database at a future date. It also indicates which nodes will be pushed back to the database, which may help you determine whether the stream can be reordered to improve performance.

1. Make sure that **Display SQL in the messages log during stream execution** and **Display SQL generation details in the messages log during stream preparation** are selected in the User Options dialog box. See the topic “Client Performance and Optimization Settings” on page 31 for more information.

2. On the stream canvas, select the node or stream that you want to preview.

3. Click the **Preview SQL** button on the toolbar.

All nodes for which SQL is generated (and that will be pushed back to the database when the stream is executed) are colored purple on the stream canvas.

4. To preview the generated SQL, from the menus choose:

**Tools > Stream Properties > Messages...**

---

## Viewing SQL for Model Nuggets

For some models, SQL for the model nugget can be generated, pushing back the model scoring stage to the database. The main use of this feature is not to improve performance, but to allow streams containing these nuggets to have their full SQL pushed back. See the topic “Nodes Supporting SQL Generation” on page 40 for more information.

To view the SQL for a model nugget that supports SQL generation:

1. Select the Settings tab on the model nugget.
2. Choose one of the options **Generate with (no) missing value support** or **Generate SQL for this model**, as appropriate.
3. In the model nugget menu, choose:  
**File > Export SQL**
4. Save the file.
5. Open the file to view the SQL.

---

## Tips for Maximizing SQL Generation

To get the best performance boost from SQL optimization, pay attention to the following items.

**Stream order.** SQL generation may be halted when the function of the node has no semantic equivalent in SQL because IBM SPSS Modeler’s data-mining functionality is richer than the traditional data-processing operations supported by standard SQL. When this happens, SQL generation is also suppressed for any downstream nodes. Therefore, you may be able to significantly improve performance by reordering nodes to put operations that halt SQL as far downstream as possible. The SQL optimizer can do a certain amount of reordering automatically (just make sure stream rewriting is enabled), but further improvements may be possible. A good candidate for this is the Select node, which can often be brought forward. See the topic “Nodes Supporting SQL Generation” on page 40 for more information.

**CLEM expressions.** If a stream cannot be reordered, you may be able to change node options or CLEM expressions or otherwise recast the way the operation is performed, so that it no longer inhibits SQL generation. Derive, Select, and similar nodes can commonly be rendered into SQL, provided that all of the CLEM expression operators have SQL equivalents. Most operators can be rendered, but there are a number of operators that inhibit SQL generation (in particular, the sequence functions [“@ functions”).

Sometimes generation is halted because the generated query has become too complex for the database to handle. See the topic “CLEM Expressions and Operators Supporting SQL Generation” on page 44 for more information.

**Multiple source nodes.** Where a stream has multiple database source nodes, SQL generation is applied to each input branch independently. If generation is halted on one branch, it can continue on another. Where two branches merge (and both branches can be expressed in SQL up to the merge), the merge itself can often be replaced with a database join, and generation can be continued downstream.

**Database algorithms.** Model estimation is always performed on IBM SPSS Modeler Server rather than the database, except when using database-native algorithms from Microsoft, IBM, or Oracle.

**Scoring models.** In-database scoring is supported for some models by rendering the generated model into SQL. However, some models generate extremely complex SQL expressions that aren’t always evaluated effectively within the database. For this reason, SQL generation must be enabled separately for each model node. If you find that a model node is inhibiting SQL generation, go to the Settings tab on the node dialog box and select **Generate SQL for this model** (with some models, you may have additional options controlling generation). Run tests to confirm that the option is beneficial for your application. See the topic “Nodes Supporting SQL Generation” for more information.

When testing modeling nodes to see if SQL generation for models works effectively, we recommend first saving all streams from IBM SPSS Modeler. Some database systems may hang while trying to process the (potentially complex) generated SQL, requiring IBM SPSS Modeler to be closed from the Windows task manager.

**Database caching.** If you are using a node cache to save data at critical points in the stream (for example, following a Merge or Aggregate node), make sure that database caching is enabled along with SQL optimization. This will allow data to be cached to a temporary table in the database (rather than the file system) in most cases. See the topic “Configuring SQL Optimization” on page 38 for more information.

**Vendor-specific SQL.** Most of the generated SQL is standards-conforming (SQL-92), but some nonstandard, vendor-specific features are exploited where practical. The degree of SQL optimization can vary, depending on the database source.

---

## Nodes Supporting SQL Generation

The following tables show nodes representing data-mining operations that support SQL generation. With the exception of the database modeling nodes, if a node does not appear in these tables, it does not support SQL generation.

You can preview the SQL that is generated before executing it. See the topic “Previewing Generated SQL” on page 39 for more information.

*Table 2. Sources*

Node supporting SQL generation	Notes
Database	This node is used to specify tables and views to be used for further analysis. This node enables entry of SQL queries. Avoid results sets with duplicate column names. See the topic “Writing SQL Queries” on page 46 for more information.

Table 3. Record operations

Node supporting SQL generation	Notes
Select	Supports generation only if SQL generation for the select expression itself is supported (see expressions below). If any fields have nulls, SQL generation does not give the same results for discard as are given in native IBM SPSS Modeler.
Sample	Simple sampling supports SQL generation to varying degrees depending on the database. See Table 4.
Aggregate	SQL generation support for aggregation depends on the data storage type. See Table 5 on page 42.
RFM Aggregate	Supports generation except if saving the date of the second or third most recent transactions, or if only including recent transactions. However, including recent transactions does work if the <code>datetime_date(YEAR,MONTH,DAY)</code> function is pushed back.
Sort	
Merge	<p>No SQL generated for merge by order.</p> <p>Merge by key with full or partial outer join is only supported if the database/driver supports it. Non-matching input fields can be renamed by means of a Filter node, or the Filter tab of a source node.</p> <p>Supports SQL generation for merge by condition.</p> <p>For all types of merge, <code>SQL_SP_EXISTS</code> is not supported if inputs originate in different databases.</p>
Append	Supports generation if inputs are unsorted.
Distinct	

Table 4. SQL generation support in the Sample node for simple sampling.

Mode	Sample	Max size	Seed	DB2 for z/OS	DB2 for OS/400	DB2 for Windows/UNIX	Netezza	Oracle	SQL Server	Teradata	
Include	First	n/a		Y	Y	Y	Y	Y	Y	Y	
	1-in-n	off		Y	Y	Y	Y	Y		Y	
		max		Y	Y	Y	Y	Y		Y	
	Random %	off	off			Y	Y	Y			Y
			on			Y		Y			
		max	off			Y	Y	Y	Y		Y
on					Y		Y				
Discard	First	off					Y	Y			
		max					Y	Y			
	1-in-n	off		Y	Y	Y	Y	Y		Y	
		max		Y	Y	Y	Y	Y		Y	
	Random %	off	off			Y	Y	Y			Y
			on			Y		Y			
		max	off			Y	Y	Y	Y		Y
			on			Y		Y			

Table 5. SQL generation support in the Aggregate node.

Storage	Sum	Mean	Min	Max	SDev	Median	Count	Variance	Percentile
Integer	Y	Y	Y	Y	Y		Y	Y	
Real	Y	Y	Y	Y	Y		Y	Y	
Date			Y	Y			Y		
Time			Y	Y			Y		
Timestamp			Y	Y			Y		
String			Y	Y			Y		

Table 6. Field operations

Node supporting SQL generation	Notes
Type	Supports SQL generation if the Type node is instantiated and no ABORT or WARN type checking is specified.
Filter	
Derive	Supports SQL generation if SQL generated for the derive expression is supported (see expressions below).
Ensemble	Supports SQL generation for Continuous targets. For other targets, supports generation only if the "Highest confidence wins" ensemble method is used.
Filler	Supports SQL generation if the SQL generated for the derive expression is supported (see expressions below).
Anonymize	Supports SQL generation for Continuous targets, and partial SQL generation for Nominal and Flag targets.
Reclassify	
Binning	Supports SQL generation if the "Tiles (equal count)" binning method is used and the "Read from Bin Values tab if available" option is selected.
RFM Analysis	Supports SQL generation if the "Read from Bin Values tab if available" option is selected, but downstream nodes will not support it.
Partition	Supports SQL generation to assign records to partitions.
SetToFlag	
Restructure	

Table 7. Graphs

Node supporting SQL generation	Notes
Graphboard	SQL generation is supported for the following graph types: Area, 3-D Area, Bar, 3-D Bar, Bar of Counts, Heat map, Pie, 3-D Pie, Pie of Counts. For Histograms, SQL generation is supported for categorical data only.
Distribution	
Web	
Evaluation	

For some models, SQL for the model nugget can be generated, pushing back the model scoring stage to the database. The main use of this feature is not to improve performance, but to allow streams containing these nuggets to have their full SQL pushed back. See the topic "Viewing SQL for Model Nuggets" on page 39 for more information.



Table 8. Model nuggets

Model nugget supporting SQL generation	Notes
C&R Tree	Supports SQL generation for the single tree option, but not for the boosting, bagging or large dataset options.
QUEST	
CHAID	
C5.0	
Decision List	
Linear	Supports SQL generation for the standard model option, but not for the boosting, bagging or large dataset options.
Neural Net	Supports SQL generation for the standard model option (Multilayer Perceptron only), but not for the boosting, bagging or large dataset options.
PCA/Factor	
Logistic	Supports SQL generation for Multinomial procedure but not Binomial. For Multinomial, generation is not supported when confidences are selected, unless the target type is Flag.
Generated Rulesets	
Auto Classifier	If a User Defined Function (UDF) scoring adapter is enabled, these nuggets support SQL pushback. In addition, If either SQL generation for Continuous targets, or the "Highest confidence wins" ensemble method are used, these nuggets support further pushback downstream.
Auto Numeric	

Table 9. Output

Node supporting SQL generation	Notes
Table	Supports generation if SQL generation is supported for highlight expression (see expressions below).
Matrix	Supports generation except if "All numerics" is selected for the Fields option.
Analysis	Supports generation, depending on the options selected.
Transform	
Statistics	Supports generation if the Correlate option is not used.
Report	
Set Globals	

Table 10. Export

Node supporting SQL generation	Notes
Database	
Publisher	The published stream will contain generated SQL.

## CLEM Expressions and Operators Supporting SQL Generation

The following tables show the mathematical operations and expressions that support SQL generation and are often used during data mining. Operations absent from these tables do not support SQL generation in the current release.

Table 11. Operators

Operation supporting SQL generation	Notes
+	
-	
/	
*	
><	Used to concatenate strings.

Table 12. Relational operators

Operation supporting SQL generation	Notes
=	
/=	Used to specify "not equal."
>	
>=	
<	
<=	

Table 13. Functions

Operation supporting SQL generation	Notes
abs	
allbutfirst	
allbutlast	
and	
arccos	
arcsin	
arctan	
arctanh	
cos	
div	
exp	
fracof	
hasstartstring	
hassubstring	
integer	
intof	
isaplhacode	

Table 13. Functions (continued)

Operation supporting SQL generation	Notes
islowercode	
isnumbercode	
isstartstring	
issubstring	
isuppercode	
last	
length	
locchar	
log	
log10	
lowertoupper	
max	
member	
min	
negate	
not	
number	
or	
pi	
real	
rem	
round	
sign	
sin	
sqrt	
string	
strmember	
subscrs	
substring	
substring_between	
uppertolower	
to_string	

Table 14. Special functions

Operation supporting SQL generation	Notes
@NULL	
@GLOBAL_AVE	The special global functions are used to retrieve global values computed by the Set Globals node.
@GLOBAL_SUM	

Table 14. Special functions (continued)

Operation supporting SQL generation	Notes
@GLOBAL_MAX	
@GLOBAL_MEAN	
@GLOBAL_MIN	
@GLOBALSDEV	

Table 15. Aggregate functions

Operation supporting SQL generation	Notes
Sum	
Mean	
Min	
Max	
Count	
SDev	

## Using SQL Functions in CLEM Expressions

The @SQLFN function can be used to add named SQL functions within CLEM expressions, for purposes of database execution only. This can be useful in special cases where proprietary SQL or other vendor-specific customizations are required.

Use of this function is not covered by the standard IBM SPSS Modeler support agreement, since execution relies on external database components beyond the control of IBM Corp., but may be deployed in special cases, typically as part of a Services engagement. Contact <http://www.ibm.com/software/analytics/spss/services/> for more information if necessary.

---

## Writing SQL Queries

When using the Database node, you should pay special attention to any SQL queries that result in a dataset with duplicate column names. These duplicate names often prevent SQL optimization for any downstream nodes.

IBM SPSS Modeler uses nested SELECT statements to push back SQL for streams that use an SQL query in the Database source node. In other words, the stream nests the query specified in the Database source node inside of one or more SELECT statements generated during the optimization of downstream nodes. Therefore, if the result set of a query contains duplicate column names, the statement cannot be nested by the RDBMS. Nesting difficulties occur most often during a table join where a column with the same name is selected in more than one of the joined tables. For example, consider this query in the source node:

```
SELECT e.ID, e.LAST_NAME, d.*
FROM EMP e RIGHT OUTER JOIN
DEPT d ON e.ID = d.ID;
```

The query will prevent subsequent SQL optimization, since this SELECT statement would result in a dataset with two columns called ID.

In order to allow full SQL optimization, you should be more explicit when writing SQL queries and specify column aliases when a situation with duplicate column names arises. The statement below illustrates a more explicit query:

```
SELECT e.ID AS ID1, e.LAST_NAME, d.*  
FROM EMP e RIGHT OUTER JOIN  
DEPT d ON e.ID = d.ID;
```



---

## Appendix A. Configuring Oracle for UNIX Platforms

---

### Configuring Oracle for SQL Optimization

When running IBM SPSS Modeler Server on UNIX platforms and reading from an Oracle database, consider the following tips to ensure that generated SQL is being fully optimized within the database.

#### Proper Locale Specification

When running IBM SPSS Modeler Server in a locale other than that shipped with the Connect ODBC drivers, you should reconfigure the machine to enhance SQL optimization. Connect ODBC drivers ship only with the *en\_US* locale files. Consequently, if the IBM SPSS Modeler Server machine is running in a different locale or if the shell in which IBM SPSS Modeler Server was started did not have the locale fully defined, generated SQL may not be fully optimized within Oracle. The reasons are as follows:

- IBM SPSS Modeler Server uses the ODBC locale files corresponding to the locale in which it is running to translate the codes returned from the database into text strings. It then uses these text strings to determine which database it is actually connecting to.
- If the locale (as returned to IBM SPSS Modeler Server by the system `$LANG` query) is not *en\_US*, IBM SPSS Modeler cannot translate the codes it receives from the ODBC driver into text. In other words, an untranslated code, rather than the string *Oracle*, is returned to IBM SPSS Modeler Server at the start of a database connection. This means that IBM SPSS Modeler is unable to optimize streams for Oracle.

To check and reset locale specifications:

1. In a UNIX shell, run:

```
#locale
```

This will return the locale information for the shell. For example:

```
$ locale
LANG=en_US.ISO8859-15
LC_CTYPE="en_US.ISO8859-15"
LC_NUMERIC="en_US.ISO8859-15"
LC_TIME="en_US.ISO8859-15"
LC_COLLATE="en_US.ISO8859-15"
LC_MONETARY="en_US.ISO8859-15"
LC_MESSAGES="en_US.ISO8859-15"
LC_ALL=en_US.ISO8859-15
```

2. Change to your Connect ODBC/locale directory. (Here you will see a single directory, *en\_US*.)
3. Create a soft link to this *en\_US* directory, specifying the name of the locale setup in the shell. An example is as follows:

```
#ln -s en_US en_US.ISO8859-15
```

For a non-English locale, such as *fr\_FR.ISO8859-1*, you should create the soft link as follows:

```
#ln -s en_US fr_FR.ISO8859-1
```

4. Once you have created the link, restart IBM SPSS Modeler Server from this same shell. (IBM SPSS Modeler Server receives its locale information from the shell from which it is started.)

#### Notes

When optimizing a UNIX machine for SQL pushbacks to Oracle, consider the following tips:

- The full locale must be specified. In the above example, you must create the link in the form `language_territory.code-page`. The existing *en\_US* locale directory is not sufficient.

- To fully optimize in-database mining, both LANG and LC\_ALL must be defined in the shell used to start IBM SPSS Modeler Server. LANG may be defined in the shell as you would any other environment variable before restarting IBM SPSS Modeler Server. For example, see the following definition:  
`#LANG=en_US.ISO8859-15; export LANG`
- Each time you start IBM SPSS Modeler Server you will need to check that the shell locale information is fully specified and that the appropriate soft link exists in the ODBC/locale directory.



---

## Appendix B. Configuring UNIX Startup Scripts

---

### Introduction

This appendix describes some of the scripts that ship with the UNIX versions of IBM SPSS Modeler Server, and it explains how to configure the scripts. Scripts are used to:

- Configure IBM SPSS Modeler Server to start automatically when the server computer is restarted.
- Manually stop and restart IBM SPSS Modeler Server.
- Change permissions on files created by IBM SPSS Modeler Server.
- Configure IBM SPSS Modeler Server to work with the ODBC Connect drivers provided with IBM SPSS Modeler Server. See the topic “IBM SPSS Modeler Server and the Data Access Pack” on page 52 for more information.
- Configure IBM SPSS Modeler Server to use the IBM SPSS Collaboration and Deployment Services Enterprise View Driver when reading data with an Enterprise View node. See the topic “Configuring a Driver for the Enterprise View Node” on page 57 for more information.

---

### Scripts

IBM SPSS Modeler Server uses several scripts, including:

- **modelersrv.sh.** The manual startup script for IBM SPSS Modeler Server is located in the IBM SPSS Modeler Server installation directory. It configures the environment for the server when the server daemon process is *manually* started. Run it when you want to manually start and stop the server. Edit it when you need to change the configuration for manual startup.
- **auto.sh.** This is a script that configures your system to start the server daemon process automatically at boot time. Run it once to configure your system for automatic startup. You do not need to edit it. The script is located in the IBM SPSS Modeler Server installation directory.
- **rc.modeler.** When you run *auto.sh*, this script is created in a location that depends on your server's operating system. It configures the environment for the server when it is *automatically* started. Edit it when you need to change the configuration for automatic startup.

Table 16. Location of *rc.modeler* by operating system.

Operating system	Location
AIX	<i>/etc/rc.modeler</i>
Solaris	<i>/etc/init.d/rc.modeler</i>

---

### Automatically Starting and Stopping IBM SPSS Modeler Server

IBM SPSS Modeler Server must be started as a daemon process. The installation program includes a script (*auto.sh*) that you can run to configure your system to automatically stop and restart IBM SPSS Modeler Server.

#### To configure the system for automatic startup and shutdown

1. Log on as root.
2. Change to the IBM SPSS Modeler Server installation directory.
3. Run the script. At the UNIX prompt, type:  
`./auto.sh`

An automatic startup script, *rc.modeler*, is created in the location shown in the table above. The operating system will use *rc.modeler* to start the IBM SPSS Modeler Server daemon process whenever the server computer is rebooted. The operating system will also use *rc.modeler* to stop the daemon whenever the system is shut down.

---

## Manually Starting and Stopping IBM SPSS Modeler Server

You can manually start and stop IBM SPSS Modeler Server by running the *modelersrv.sh* script.

### To manually start and stop IBM SPSS Modeler Server

1. Change to the IBM SPSS Modeler Server installation directory.
2. To start the server, at the UNIX command prompt, type:  

```
./modelersrv.sh start
```
3. To stop the server, at the UNIX command prompt type:  

```
./modelersrv.sh stop
```

---

## Editing Scripts

If you use both manual and automatic startup, make parallel changes in both *modelersrv.sh* and *rc.modeler*. If you use only manual startup, make changes in *modelersrv.sh*. If you use only automatic startup, make changes in *rc.modeler*.

### To edit the scripts

1. Stop IBM SPSS Modeler Server. (See the topic “Manually Starting and Stopping IBM SPSS Modeler Server” for more information. )
2. Locate the appropriate script. (See the topic “Scripts” on page 51 for more information. )
3. Open the script in a text editor, make changes, and save the file.
4. Start IBM SPSS Modeler Server, either automatically (by restarting the server computer) or manually.

---

## Controlling Permissions on File Creation

IBM SPSS Modeler Server creates temporary files with read, write, and execute permissions for everyone. You can override this default by editing the *UMASK* setting in the startup script, either in *modelersrv.sh*, *rc.modeler*, or in both. (For more information, see “Editing Scripts” above.) We recommend 077 as the most restrictive *UMASK* setting to use. Settings that are more restrictive could cause permissions problems for IBM SPSS Modeler Server.

---

## IBM SPSS Modeler Server and the Data Access Pack

If you want to use the ODBC drivers with IBM SPSS Modeler Server, the ODBC environment must be configured by *odbc.sh* when the IBM SPSS Modeler Server process starts. You do this by editing the appropriate IBM SPSS Modeler startup script, either in *modelersrv.sh*, *rc.modeler*, or in both. (See the topic “Editing Scripts” for more information. )

For more information, see the Technical Support web site at <http://www.ibm.com/support>. If you have questions about creating or setting permissions for ODBC data sources, contact your database administrator.

### To configure ODBC to start with IBM SPSS Modeler Server

1. Stop the IBM SPSS Modeler Server host if it is running.

2. Download the relevant compressed tar archive for the platform on which you have IBM SPSS Modeler Server installed. Make sure you download the drivers for the correct version of IBM SPSS Modeler Server that you have installed. Copy the file to the location where you want to install the ODBC drivers (for example, */usr/spss/odbc*).
3. Extract the archive.
4. Run the *setodbcpath.sh* script that is extracted from the archive.
5. Edit the script *odbc.sh* to add the definition of ODBCINI to the bottom of this script and export it, for example:

```
ODBCINI=/usr/spss/odbc/odbc.ini; export ODBCINI
```

ODBCINI must point to the full pathname of the *odbc.ini* file that you want IBM SPSS Modeler to read to get a list of the ODBC datasources that you define (a default *odbc.ini* is installed with the drivers).

6. Save *odbc.sh*.
7. (64-bit IBM SPSS Modeler Server installations only; for other installations, continue from the next step) Define and export LD\_LIBRARY\_PATH\_64 in *odbc.sh*:

```
if [ "$LD_LIBRARY_PATH_64" = "" ]; then
    LD_LIBRARY_PATH_64=<library_path>
else
    LD_LIBRARY_PATH_64=<library_path>:$LD_LIBRARY_PATH_64
fi
export LD_LIBRARY_PATH_64
```

where *library\_path* is the same as for the LD\_LIBRARY\_PATH definition already in the script that has been initialized with your installation path (for example, */usr/spss/odbc/lib*). The easiest way to do this is to copy the if and export statements for LD\_LIBRARY\_PATH in your *odbc.sh* file, append them to the end of the file, and then replace the "LD\_LIBRARY\_PATH" strings in the newly appended if and export statements with "LD\_LIBRARY\_PATH\_64".

Thus your final *odbc.sh* file on a 64-bit IBM SPSS Modeler Server installation might look like this:

```
if [ "$LD_LIBRARY_PATH" = "" ]; then
    LD_LIBRARY_PATH=/usr/spss/odbc/lib
else
    LD_LIBRARY_PATH=/usr/spss/odbc/lib:$LD_LIBRARY_PATH
fi
export LD_LIBRARY_PATH
if [ "$LD_LIBRARY_PATH_64" = "" ]; then
    LD_LIBRARY_PATH_64=/usr/spss/odbc/lib
else
    LD_LIBRARY_PATH_64=/usr/spss/odbc/lib:$LD_LIBRARY_PATH_64
fi
export LD_LIBRARY_PATH_64
ODBCINI=/usr/spss/odbc/odbc.ini; export ODBCINI
```

Remember to export LD\_LIBRARY\_PATH\_64, as well as defining it with the if loop.

8. Edit the *odbc.ini* file that you defined earlier using \$ODBCINI. Define the data source names that you require (these depend on the database that you are accessing).
9. Save the *odbc.ini* file.
10. Configure IBM SPSS Modeler Server to use these drivers. To do so, edit *modelersrv.sh* and add the following line immediately below the line that defines SCLEMDNAME:

```
. <odbc.sh_path>
```

where *odbc.sh\_path* is the full path to the *odbc.sh* file that you edited near the beginning of this procedure, for example:

```
. /usr/spss/odbc/odbc.sh
```

*Note:* The syntax is important here; be sure to leave a space between the first period and the path to the file.

11. Save *modelersrv.sh*.

#### To test the connection

1. Restart IBM SPSS Modeler Server.
2. Connect to IBM SPSS Modeler Server from a client.
3. On the client, add a Database source node to the canvas.
4. Open the node and verify that you can see the data source names that you defined in the *odbc.ini* file earlier in the configuration procedure.

If you do not see what you expect here, or you get errors when you try to connect to a data source that you have defined, follow the Troubleshooting procedure. See the topic “Troubleshooting ODBC Configuration” for more information.

#### To configure ODBC to start with IBM SPSS Modeler Solution Publisher Runtime

When you can successfully connect to the database from IBM SPSS Modeler Server, you can configure a IBM SPSS Modeler Solution Publisher Runtime installation on the same server by referencing the same *odbc.sh* script from the startup script of IBM SPSS Modeler Solution Publisher Runtime.

1. Edit the *modelerrun* script in IBM SPSS Modeler Solution Publisher Runtime to add the following line immediately above the last line of the script:

```
. <odbc.sh_path>
```

where *odbc.sh\_path* is the full path to the *odbc.sh* file that you edited near the beginning of this procedure, for example:

```
. /usr/spss/odbc/odbc.sh
```

*Note:* The syntax is important here; be sure to leave a space between the first period and the path to the file.

2. Save the *modelerrun* script file.

#### To configure ODBC to start with IBM SPSS Modeler Batch

No configuration of the IBM SPSS Modeler Batch script is necessary for ODBC. This is because you connect to IBM SPSS Modeler Server from IBM SPSS Modeler Batch in order to run streams. Ensure that the IBM SPSS Modeler Server ODBC configuration has been made and is working correctly, as described earlier in this section.

#### To add or edit a data source name

1. Edit the *odbc.ini* file to include the new or changed name.
2. Test the connection as described earlier in this section.

When the connection with IBM SPSS Modeler Server is working correctly, the new or changed data source should also work correctly with IBM SPSS Modeler Solution Publisher Runtime and IBM SPSS Modeler Batch.

## Troubleshooting ODBC Configuration

### No data sources listed, or random text displayed

If you open a Database source node and the list of available data sources is empty or contains unexpected entries, it may be due to a problem with the startup script.

1. Check that \$ODBCINI is defined within *modelersrv.sh*, either explicitly in the script itself, or in the *odbc.sh* script that is referenced in *modelersrv.sh*.
2. In the latter case, ensure that ODBCINI points to the full path to the *odbc.ini* file that you have used to define your ODBC data sources.

3. If the path specification in ODBCINI is correct, check the value of \$ODBCINI that is being used in the IBM SPSS Modeler Server environment by echoing the variable from within *modelersrv.sh*. To do so, add the following line to *modelersrv.sh* after the point where you define ODBCINI:

```
echo $ODBCINI
```

4. Save and then execute *modelersrv.sh*. The value of \$ODBCINI that is being set in the IBM SPSS Modeler Server environment is written to *stdout* for verification.
5. If no value at all is returned to *stdout*, and you are defining \$ODBCINI in the *odbc.sh* script that you are referencing from *modelersrv.sh*, check that the referencing syntax is correct. It should be:

```
. <odbc.sh_path>
```

where *odbc.sh\_path* is the full path to the *odbc.sh* file that you edited near the beginning of this procedure, for example:

```
. /usr/spss/odbc/odbc.sh
```

*Note:* The syntax is important here; be sure to leave a space between the first period and the path to the file.

When the correct value is echoed to *stdout* on running *modelersrv.sh*, you should be able to see the data source names in the Database source node when you restart IBM SPSS Modeler Server and connect to it from the client.

### IBM SPSS Modeler client hangs on clicking Connect in Database Connections dialog box

This behavior can be caused by your library path not being correctly set to include the path to the ODBC libraries. The library path is defined by \$LD\_LIBRARY\_PATH (and \$LD\_LIBRARY\_PATH\_64 on 64-bit versions).

To see the value of the library path in the IBM SPSS Modeler Server daemon environment, echo the value of the appropriate environment variable from within *modelersrv.sh*, after the line where you are appending the ODBC library path to the library path, and execute the script. The library path value will be echoed to the terminal when you next execute the script.

If you are referencing *odbc.sh* from *modelersrv.sh* to set up your IBM SPSS Modeler Server ODBC environment, echo your library path value from the line after the one where you reference the *odbc.sh* script. To echo the value, add the following line to the script, then save and execute the script file:

```
echo ${library_path_variable}
```

where *<library\_path\_variable>* is the appropriate library path variable for your server operating system.

The returned value of your library path must include the path to the *lib* subdirectory of your ODBC installation. If it does not, append this location to the file.

If you are running the 64-bit version of IBM SPSS Modeler Server, \$LD\_LIBRARY\_PATH\_64 will override \$LD\_LIBRARY\_PATH if it is set. If you are having this problem on one of these 64-bit platforms, echo LD\_LIBRARY\_PATH\_64 as well as \$LD\_LIBRARY\_PATH from *modelersrv.sh* and, if required, set \$LD\_LIBRARY\_PATH\_64 so that it includes the *lib* subdirectory of your ODBC installation, and export the definition.

### Data source name not found and no default driver specified

If you see this error on clicking Connect in the Database Connections dialog box, it usually indicates that your *odbc.ini* file is incorrectly defined. Check that the data source name (DSN) as defined within the [ODBC Data Sources] section at the top of the file matches the string specified between the square brackets further down in *odbc.ini* to define the DSN. If these are different in any way, you will see this error when you try to connect using the DSN from within IBM SPSS Modeler. Following is an example of an *incorrect* specification:

```
[ODBC Data Sources]
Oracle=Oracle Wire Protocol
```

```
....
....
```

```
[Oracle Driver]
Driver=/usr/ODBC/lib/XEora22.so
Description=SPSS 5.2 Oracle Wire Protocol
AlternateServers=
....
```

You need to change one of the two strings in bold so that they match exactly. Doing so should resolve the error.

### Specified driver could not be loaded

This error also indicates that the *odbc.ini* file is incorrectly defined. One possibility is that the Driver parameter within the driver stanza is incorrectly set, for example:

```
[ODBC Data Sources]
Oracle=Oracle Wire Protocol
```

```
....
....
```

```
[Oracle]
Driver=/nosuchpath/ODBC/lib/XEora22.so
Description=SPSS 5.2 Oracle Wire Protocol
AlternateServers=
```

1. Check that the shared object specified by the Driver parameter exists.
2. Correct the path to the shared object if it is incorrect.
3. If the Driver parameter is specified in this format:

```
Driver=ODBCHOME/lib/XEora22.so
```

this indicates that you have not initialized your ODBC-related scripts. Run the *setodbcpath.sh* script that is installed with the drivers. See the topic “IBM SPSS Modeler Server and the Data Access Pack” on page 52 for more information. When you have run this script, you should see that the string “ODBCHOME” has been substituted with the path to your ODBC installation. This should resolve the issue.

Another cause may be a problem with the driver's library. Use the *ivtestlib* tool provided with ODBC to confirm that the driver can't be loaded. For Connect64, use the *ddtestlib* tool. Correct the problem by setting the library path variable in the startup script.

For example, if the Oracle driver cannot be loaded for a 32-bit installation, follow these steps:

1. Use *ivtestlib* to confirm that the driver cannot be loaded. For example, at the UNIX prompt, type:

```
sh
cd ODBC DIR
. odbc.sh
./bin/ivtestlib MFor815
```

where OBCDIR is replaced by the path to your ODBC installation directory.

2. Read the message to see if there is an error. For example, the message:

```
Load of MFor815.so failed: ld.so.1: bin/ivtestlib: fatal: libclntsh.so: open failed: No such file or directory
```

indicates that the Oracle client library, *libclntsh.so*, is missing or that it is not on the library path (for example, not on LD\_LIBRARY\_PATH on Solaris).

3. Confirm that the library exists. If it doesn't, reinstall the Oracle client. If the library is there, type the following sequence of commands from the UNIX command prompt:

```
LD_LIBRARY_PATH=$LD_LIBRARY_PATH:/bigdisk/oracle/product/8.1.6/lib
export LD_LIBRARY_PATH
./bin/ivtestlib Mfor815
```

where */bigdisk/oracle/product/8.1.6/lib* is replaced by the path to *libclntsh.so* and *LD\_LIBRARY\_PATH* is the library path variable for your operating system.

Note that if you are running IBM SPSS Modeler 64-bit on Linux or Solaris, the library path variable contains the suffix *\_64*. Therefore, the first two lines in the previous example would become:

```
LD_LIBRARY_PATH_64=$LD_LIBRARY_PATH_64:/bigdisk/oracle/product/8.1.6/lib
export LD_LIBRARY_PATH_64
```

4. Read the message to confirm that the driver can now be loaded. For example, the message:

**Load of MFor815.so successful, qehandle is 0xFF3A1BE4**

indicates that the Oracle client library can be loaded.

5. Correct the library path in the IBM SPSS Modeler startup script.
6. Restart the IBM SPSS Modeler Server with the startup script that you edited (*modelersrv.sh* or *rc.modeler*).

## Library Paths

The name of the library path variable varies depending on your operating system. The following table is included as a guide to help you make appropriate substitutions when you are configuring or troubleshooting on your system.

Table 17. Library path by operating system.

Operating system	Library path variable name
AIX	LIBPATH
Solaris or Linux 64-bit	LD_LIBRARY_PATH_64

## Configuring a Driver for the Enterprise View Node

The Enterprise View node enables you to create and maintain a connection between an IBM SPSS Modeler session and an IBM SPSS Collaboration and Deployment Services Enterprise View in a shared IBM SPSS Collaboration and Deployment Services Repository.

To use the Enterprise View node with IBM SPSS Modeler Server, an IBM SPSS Collaboration and Deployment Services Enterprise View Driver must be installed and configured. For details on installing the driver, contact your local administrator. After installation, a reference to *pev.sh* must be added in the startup script—either in *modelersrv.sh*, *rc.modeler*, or in both—following the format:

```
./usr/odbc/pev.sh
```





---

## Appendix C. Configuring and Running SPSS Modeler Server as a Non-Root Process on UNIX

---

### Introduction

These instructions provide information on running IBM SPSS Modeler Server as a non-root process on UNIX systems.

**Running as root.** The default installation of IBM SPSS Modeler Server assumes that the server daemon process will run as root. Running as root allows IBM SPSS Modeler to authenticate reliably each user login and start each user session on the corresponding UNIX user account. This ensures that users have access only to their own files and directories.

**Running as non-root.** Running IBM SPSS Modeler Server as a non-root process means having the real and effective user IDs of the server daemon process set to an account of your choice. All user sessions started by SPSS Modeler Server will use the same UNIX account and this means that any file data read or written by SPSS Modeler is shared by all SPSS Modeler users. Access to database data is not affected because users have to authenticate themselves independently to each of the database data sources they use. Without root privilege, IBM SPSS Modeler operates in one of two ways:

- **Without a private password database.** With this method, SPSS Modeler uses the existing UNIX password database, NIS, or LDAP server that is normally used for user authentication on the UNIX system. See the topic “Configuring as non-root without a private password database” for more information.
- **With a private password database.** With this method, SPSS Modeler authenticates users against a private password database, distinct from the UNIX password database, NIS, or LDAP server that is normally used for authentication on UNIX. See the topic “Configuring as non-root using a private password database” on page 60 for more information.

---

### Configuring as non-root without a private password database

To configure IBM SPSS Modeler Server to run on a non-root account without the need for a private password database, follow these steps:

1. Open the SPSS Modeler Server *options.cfg* file for editing.
2. Set the option `start_process_as_login_user` to Y.
3. Save and close the *options.cfg* file.

By default, SPSS Modeler Server tries each authentication method until it finds one that works. However, if desired you can use the `authentication_methods` option in *options.cfg* to configure the server to try only one specific authentication method. Possible values for the option are `pasw_modeler`, `gss`, `pam`, `sspi`, `unix`, or `windows`.

Note that running as non-root is likely to require configuration updates. See the topic “Troubleshooting user authentication failures” on page 61 for more information.

*Caution:* Do not enable the `start_process_as_login_user` setting and then start up IBM SPSS Modeler Server as *root*. Doing so would mean that, for all users that are connected to the server, their server processes would run as *root*; this is a security risk.

---

## Configuring as non-root using a private password database

If you choose to authenticate users by means of a private password database, all user sessions are started on the same non-root user account.

To configure IBM SPSS Modeler Server to run on a non-root account in this way, follow these steps:

1. Create a group to contain all of your users. You can name this group whatever you'd like, but for this example, let's call it *modelerusers*.
2. Create the user account on which to run IBM SPSS Modeler Server. This account is for the sole use of the IBM SPSS Modeler Server daemon process. For this example, let's call it *modelerserv*.

When creating the account, note that:

- The primary group should be the *<modelerusers>* group created previously.
  - The home directory can be the IBM SPSS Modeler installation directory or any other convenient default (consider using something other than the installation directory if you need the account to survive upgrades).
3. Next, configure the startup scripts to start IBM SPSS Modeler Server using the newly created account. Locate the appropriate startup script and open it in a text editor. See the topic "Scripts" on page 51 for more information.

- a. Change the umask setting to allow at least group read access on created files:

```
umask 027
```

4. Edit the server options file, *config/options.cfg*, to specify authentication against the private password database by appending the line:

```
authentication_methods, "pasw_modeler"
```

5. Set the option *start\_process\_as\_login\_user* to Y.

6. Next, you'll need to create a private password database stored in the file *config/passwords.cfg*. The password file defines the user name/password combinations that are allowed to login to IBM SPSS Modeler. *Note:* These are private to IBM SPSS Modeler and have no connection with the user names and passwords used to login to UNIX. You can use the same user names for convenience, but you cannot use the same passwords.

To create the password file, you will need to use the password utility program, *pwutil*, located in the *bin* directory of the IBM SPSS Modeler Server installation. The synopsis of this program is:

```
pwutil [ username [ password ] ]
```

The program takes a user name and plain-text password and writes the user name and encrypted password to the standard output in a format suitable for inclusion in the password file. For example, to define a user *modeler* with the password "data mining" you would type:

```
bin/pwutil modeler "data mining" > config/passwords.cfg
```

Defining a single user name is sufficient in most cases, where all users log in with the same name and password. However additional users can be created by using the >> operator to append each to the file, for example:

```
bin/pwutil modeler "data miner2" >> config/passwords.cfg
```

*Note:* If a single > is used, the contents of *passwords.cfg* will be overwritten each time, replacing any users set previously. Remember that all users share the same UNIX user account regardless.

*Note:* If you add new users to the private passwords database while SPSS Modeler Server is running, you will need to restart SPSS Modeler Server so that it can recognize the newly defined users. Until you do so, logins will fail for any new users added via *pwutil* since the last restart of SPSS Modeler Server.

7. Recursively change the ownership of the IBM SPSS Modeler installation directory and its contents to be user *<modelerserv>* and group *<modelerusers>* where the names referenced are those you created earlier. For example:

```
chown -R -h modelerserv:modelerusers .
```

8. Consider creating subdirectories in the data directory for your IBM SPSS Modeler users so that they have somewhere to store working data without interference. These directories should be group-owned by the `<modelerusers>` group and have group read, write, and search permissions. For example, to create a working directory for user `bob`:

```
mkdir data/bob
chown bob:modelerusers data/bob
chmod ug=rwx,o= data/bob
```

Additionally, you can set the set-group-ID bit on the directory so that any data files copied into the directory will be automatically group-owned by `<modelerusers>`:

```
chmod g+s data/bob
```

---

## Running SPSS Modeler Server as a non-root user

To run SPSS Modeler Server as a non-root user, follow these steps:

1. Log in using the non-root user account created earlier.
2. If you are running with the configuration file option `start_process_as_login_user` enabled, you can start, stop, and check the status of SPSS Modeler Server. See the topic “To Start, Stop, and Check Status on UNIX” on page 17 for more information.

End users connect to SPSS Modeler Server by logging in from the client software. You must give end users the information that they need to connect, including the IP address or host name of the server machine.

### Running IBM SPSS Modeler Server as a non-root user on AIX

When starting IBM SPSS Modeler Server as a non-root user on AIX, there is the chance that the error `lsattr not found` will be output to the console. This is caused because `lsattr` is located in the `/usr/sbin` directory, but when IBM SPSS Modeler Server logs in a new user (to run as non-root) it resets the path to include just `/usr/bin` and `/bin`.

If this error occurs, the solution is to make the following link under `/usr/bin`:

```
su
cd /usr/bin
ln -s /usr/sbin/lsattr lsattr
```

After making the link, restart IBM SPSS Modeler Server.

---

## Troubleshooting user authentication failures

Depending on how the operating system is configured to perform authentication, you may experience failures to log on to SPSS Modeler Server when running in a non-root configuration. For example, this may occur if your operating system is configured (using the `/etc/nsswitch.conf` file or similar) to check the local shadow password file, rather than use NIS or LDAP. This occurs because SPSS Modeler Server requires read access to the files used to perform authentication, including the `/etc/shadow` file or its equivalent (for example, `/etc/security/passwd` on AIX), which stores secure user account information. However, the operating system file permissions are generally set so that `/etc/shadow` is accessible only by the root user. Under these circumstances a non-root process cannot read `/etc/shadow` to validate user passwords, resulting in an authentication error.

There are several ways to resolve this issue:

- Ask your system administrator to configure the operating system to use NIS or LDAP for authentication.
- Change the file permissions on the protected files, for example by granting read access to the `/etc/shadow` file so that the local user account used to run SPSS Modeler Server can access the file. While

this workaround might be deemed unsuitable in production environments, it could be temporarily applied to a test environment to verify whether the authorization failure is linked to the operating system configuration.

- Specify an access control list (ACL) for the */etc/shadow* file (see the example later in this section).
- Run SPSS Modeler Server as root, to enable the server processes to read the */etc/shadow* file. *Caution:* In this case, ensure that the *options.cfg* file for SPSS Modeler Server contains the option `start_process_as_login_user, N` to avoid the security issue explained earlier.

### Example of an access control list (ACL)

On Solaris operating systems, you can use the `setfacl` command to create an access control list (check with your administrator to ensure that this command is available on your system). The following example shows how to create such an ACL for the user *modserv*:

```
# vi /tmp/shadow.acl
user::r--
user:modserv:r--
group::--
mask:r--
other:--

# setfacl -f /tmp/shadow.acl /etc/shadow

# getfacl /etc/shadow

# file: /etc/shadow
# owner: root
# group: sys
user::r--
user:modserv:r--      #effective:r--
group:---             #effective:---
mask:r--
other:---
```

---

## Appendix D. Configuring and Running SPSS Modeler Server with a private password file on Windows

---

### Introduction

These instructions provide information on running IBM SPSS Modeler Server using a private password file on Windows systems. With this method, IBM SPSS Modeler authenticates users against a private password database, distinct from the system authentication on Windows.

---

### Configuring using a private password database

If you choose to authenticate users by means of a private password database, all user sessions are started on the same user account.

To configure IBM SPSS Modeler Server in this way, follow these steps:

1. Create the user account on which to run IBM SPSS Modeler Server. This account is for the sole use of the IBM SPSS Modeler Server daemon process. For this example, let's call it *modelerserv*.
2. Edit the server options file, *config/options.cfg*, to set the option `start_process_as_login_user` to `Yand` to specify authentication against the private password database by appending the line:  
`authentication_methods, "pasw_modeler"`
3. Next, you'll need to create a private password database stored in the file *config/passwords.cfg*. The password file defines the user name/password combinations that are allowed to login to IBM SPSS Modeler. *Note:* These are private to IBM SPSS Modeler and have no connection with the user names and passwords used to login to Windows. You can use the same user names for convenience, but you cannot use the same passwords.

To create the password file, you will need to use the password utility program, *pwutil*, located in the *bin* directory of the IBM SPSS Modeler Server installation. The synopsis of this program is:

```
pwutil [ username [ password ] ]
```

The program takes a user name and plain-text password and writes the user name and encrypted password to the standard output in a format suitable for inclusion in the password file. For example, to define a user *modeler* with the password "data mining" you would type:

```
bin/pwutil modeler "data mining" > config/passwords.cfg
```

Defining a single user name is sufficient in most cases, where all users log in with the same name and password. However additional users can be created by using the `>>` operator to append each to the file, for example:

```
bin/pwutil modeler "data miner2" >> config/passwords.cfg
```

*Note:* If a single `>` is used, the contents of *passwords.cfg* will be overwritten each time, replacing any users set previously. Remember that all users share the same UNIX user account regardless.

*Note:* If you add new users to the private passwords database while SPSS Modeler Server is running, you will need to restart SPSS Modeler Server so that it can recognize the newly defined users. Until you do so, logins will fail for any new users added via *pwutil* since the last restart of SPSS Modeler Server.

4. Give the user created in step 1 full control over the server options file: *config/options.cfg* and the `%ALLUSERSPROFILE%\IBM\SPSS` directory.
5. In the system services, stop the *IBM SPSS Modeler Server* service and change the *Log on* from the Local System Account to the user account created in step 1 and restart the service.



## Appendix E. Load Balancing with Server Clusters

With IBM SPSS Collaboration and Deployment Services, a plug-in called the Coordinator of Processes can be used to manage services on the network. The Coordinator of Processes provides server management capabilities designed to optimize client-server communication and processing.

Services to be managed, such as IBM SPSS Statistics Server or IBM SPSS Modeler Server, register with the Coordinator of Processes upon starting and periodically send updated status messages. Services can also store any necessary configuration files in the IBM SPSS Collaboration and Deployment Services Repository and retrieve them when initializing.

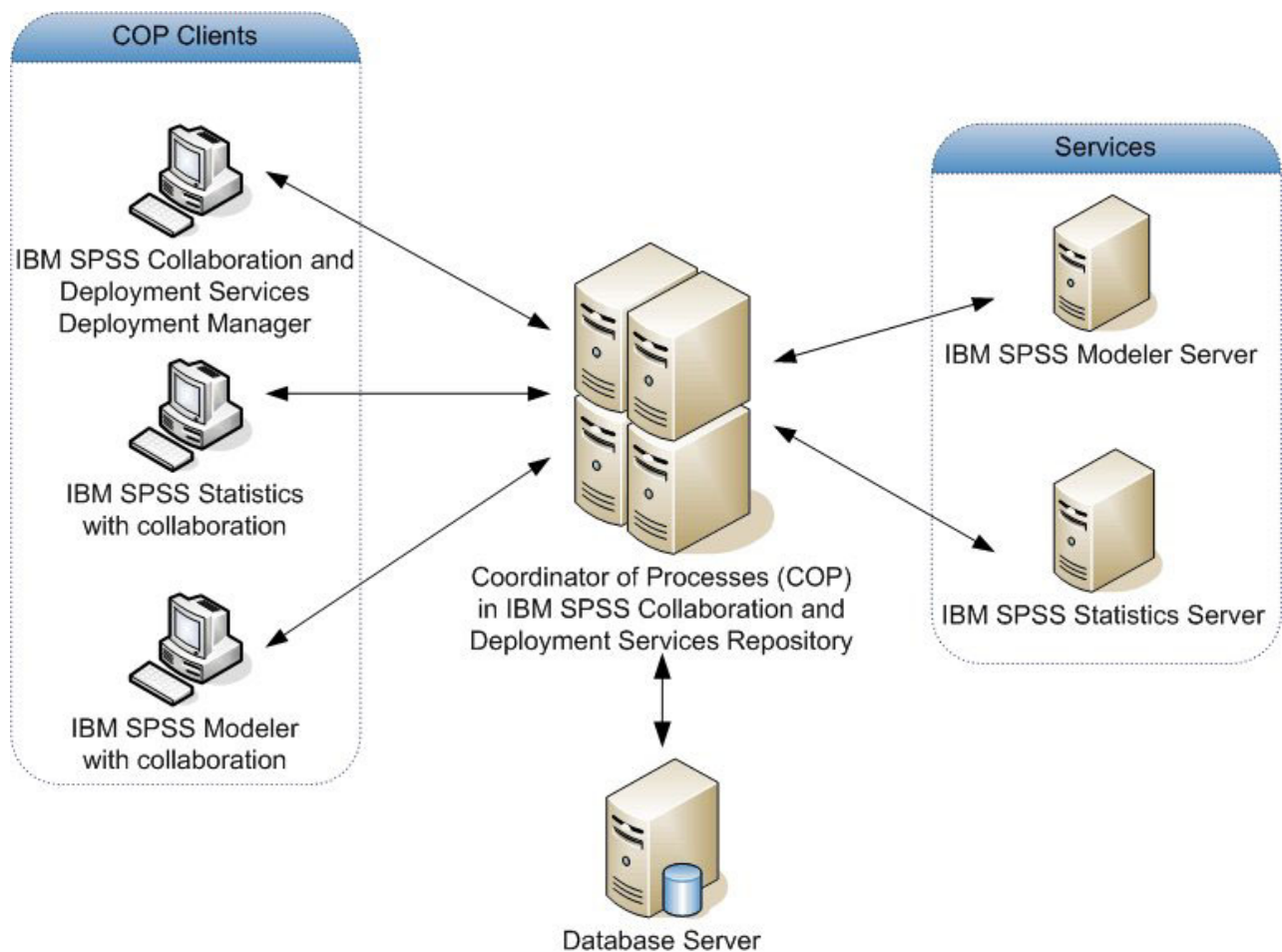


Figure 5. Coordinator of Processes Architecture

Executing your IBM SPSS Modeler streams on a server can increase performance. In some cases, you may have only the choice of one or two servers. In other cases, you might be offered a larger choice of servers because there is a substantive difference between each server, such as owner, access rights, server data, test versus production servers, and so on. In addition, if you have the Coordinator of Processes on your network, you might be offered a server cluster.

A server cluster is a group of servers that are interchangeable in terms of configuration and resources. The Coordinator of Processes determines which server is best suited to respond to a processing request, using an algorithm that will balance the load according to several criteria, including the server weights,

user priorities, and current processing loads. For more information, see the *Coordinator of Processes Service Developer's Guide* available in the IBM SPSS Collaboration and Deployment Services documentation suite.

Whenever you connect to a server or server cluster in IBM SPSS Modeler, you can enter a server manually or search for a server or cluster using the Coordinator of Processes. See the topic "Connecting to IBM SPSS Modeler Server" on page 11 for more information.



---

## Notices

This information was developed for products and services offered worldwide.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not grant you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing  
IBM Corporation  
North Castle Drive  
Armonk, NY 10504-1785  
U.S.A.

For license inquiries regarding double-byte (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

Intellectual Property Licensing  
Legal and Intellectual Property Law  
IBM Japan Ltd.  
1623-14, Shimotsuruma, Yamato-shi  
Kanagawa 242-8502 Japan

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law: INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM Software Group  
ATTN: Licensing  
200 W. Madison St.  
Chicago, IL; 60606  
U.S.A.

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this document and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement or any equivalent agreement between us.

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurements may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

All statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

If you are viewing this information softcopy, the photographs and color illustrations may not appear.

---

## Trademarks

IBM, the IBM logo, and [ibm.com](http://www.ibm.com) are trademarks or registered trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the Web at "Copyright and trademark information" at [www.ibm.com/legal/copytrade.shtml](http://www.ibm.com/legal/copytrade.shtml).

Intel, Intel logo, Intel Inside, Intel Inside logo, Intel Centrino, Intel Centrino logo, Celeron, Intel Xeon, Intel SpeedStep, Itanium, and Pentium are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.

Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both.

Microsoft, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

UNIX is a registered trademark of The Open Group in the United States and other countries.

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

Other product and service names might be trademarks of IBM or other companies.



---

# Index

## Special characters

@SQLFN function 46

## Numerics

64-bit operating systems 6

## A

adding IBM SPSS Modeler Server  
connections 13, 14  
administration  
of IBM SPSS Modeler Server 18  
administrator access  
for IBM SPSS Modeler Server 19  
with User Access Control (UAC) 19  
allow\_modelling\_memory\_override  
options.cfg file 21  
application examples 3  
architecture  
components 5  
authentication 15  
auto.sh (UNIX)  
location of 51  
automatic server startup  
configuring on UNIX 51

## C

cache compression 21  
cache\_compression  
options.cfg file 21  
cache\_connection option 25  
caching, in-database 25  
chemsrv.sh (UNIX)  
location of 51  
CLEM expressions  
SQL generation 44  
configuration options  
automatic SQL generation 23  
connections and sessions 20  
coordinator of processes 24  
COP 24  
data file access 21  
login attempts 20  
memory management 21  
of IBM SPSS Modeler Server 18  
overview 20, 21  
parallel processing 21  
performance and optimization 21  
port number 20  
SQL string length 23  
SSL data encryption 23  
stream rewriting 21  
temp directory 21  
connections  
server cluster 14  
to IBM SPSS Modeler Server 11, 13,  
14

Coordinator of Processes 14  
load balancing 65  
server clusters 65  
coordinator of processes configuration  
for IBM SPSS Modeler Server 24  
COP 14  
load balancing 65  
server clusters 65  
COP configuration  
for IBM SPSS Modeler Server 24  
cop\_enabled  
options.cfg file 24  
cop\_host  
options.cfg file 24  
cop\_password  
options.cfg file 24  
cop\_port\_number  
options.cfg file 24  
cop\_service\_description  
options.cfg file 24  
cop\_service\_host  
options.cfg file 24  
cop\_service\_name  
options.cfg file 24  
cop\_service\_weight  
options.cfg file 24  
cop\_update\_interval  
options.cfg file 24  
cop\_user\_name  
options.cfg file 24

## D

data access 8  
data access pack  
and UNIX library paths 57  
configuring UNIX for 52  
ODBC, configuring on UNIX 52  
troubleshooting ODBC on UNIX 54  
data files  
IBM SPSS Statistics 10  
importing and exporting 10  
data sources  
single sign-on 13  
data\_file\_path  
options.cfg file 21  
data\_files\_restricted  
options.cfg file 21  
database caching  
controlling from options.cfg 25  
SQL generation 39  
database connections  
closing 25  
database servers 33  
databases  
accessing 8  
DB2  
SQL optimization 35, 36  
disk space 7  
calculating 8  
documentation 3

domain name (Windows)  
IBM SPSS Modeler Server 11

## E

encryption  
FIPS 24  
SSL 26  
error on stream execution 21  
examples  
Applications Guide 3  
overview 4

## F

file permissions  
configuring on UNIX 52  
on IBM SPSS Modeler Server 16  
filenames  
UNIX 9  
Windows 9  
FIPS encryption 24  
firewall settings  
options.cfg file 21

## H

hard drives 8  
hardware recommendations  
for IBM SPSS Modeler Server 6  
host name  
IBM SPSS Modeler Server 11, 13

## I

IBM SPSS Analytic Server  
configuration options 21  
IBM SPSS Modeler 1  
documentation 3  
IBM SPSS Modeler Administration  
Console 19  
administrator access 19  
User Access Control access 19  
IBM SPSS Modeler Server 1  
administration of 19  
administration options 18  
administrator access 19  
configuration options 20  
coordinator of processes  
configuration 24  
COP configuration 24  
different results than client 16  
domain name (Windows) 11  
file creation 16  
host name 11, 13  
information for end users 15  
monitoring usage 25  
password 11  
permissions 15

- IBM SPSS Modeler Server (*continued*)
  - port number 11, 13, 20
  - server processes 25
  - single sign-on 12, 13
  - single sign-on for data sources 13
  - temp directory 21
  - unresponsive processes 18
  - User Access Control access 19
  - user accounts 15
  - user authentication 15
  - user ID 11
- IBM SPSS Statistics data access technology 8
- IBM SPSS Statistics data files
  - importing and exporting 10
- in-database caching 25
- in-database mining 33
- io\_buffer\_size
  - options.cfg file 21

## K

- kernel limits on UNIX 17

## L

- log files
  - displaying generated SQL 39
  - for IBM SPSS Modeler Server 30
- logging in to IBM SPSS Modeler Server 11

## M

- max\_file\_size
  - options.cfg file 21
- max\_login\_attempts
  - options.cfg file 20
- max\_parallelism
  - options.cfg file 21
- max\_sessions
  - options.cfg file 20
- max\_sql\_string\_length
  - options.cfg file 23
- memory 8
- memory management
  - administration options 21
- memory\_usage
  - options.cfg file 21
- messages
  - displaying generated SQL 39
- Microsoft SQL Server
  - SQL optimization 35, 36
- model nuggets
  - viewing SQL for 39
- modeling
  - memory management 21
- modelling\_memory\_limit\_percentage
  - options.cfg file 21
- multiple stream execution 21

## N

- node caching
  - SQL generation 39

- node caching (*continued*)
  - writing to database 39
- nodes
  - that support SQL generation 40

## O

- ODBC
  - configuring on UNIX 52
- ODBC data sources
  - and UNIX 52
  - ODBC and UNIX scripts 52
- ODBC driver setup 35
- operating systems
  - 64-bit 6
- operators
  - SQL generation 44
- optimization
  - SQL generation 35, 36, 38
- options.cfg file 25
- Oracle
  - SQL optimization 35, 36, 49

## P

- parallel processing
  - controlling 21
- password
  - IBM SPSS Modeler Server 11
- paths 9
- performance
  - of IBM SPSS Modeler Server 31
- permissions 15
- port number
  - IBM SPSS Modeler Server 11, 13, 20
- port settings
  - options.cfg file 21
- port\_number
  - options.cfg file 20
- preview
  - SQL generation 39
- processes, unresponsive 18
- processors 6
  - multiple 21
- program\_file\_path
  - options.cfg file 21
- program\_files\_restricted
  - options.cfg file 21
- purple nodes
  - SQL optimization 36
- pushbacks 35, 36, 38, 40
  - CLEM expressions 44

## R

- RAM 8
- rc.modeler (UNIX)
  - location of 51
- results
  - differences between Client and Server 16
  - record order 16
  - rounding of 16

## S

- searching COP for connections 14
- Secure Sockets Layer 26
- security
  - configuring file creation on UNIX 52
  - file creation 16
  - SSL 26
- server
  - adding connections 13
  - logging in 11
  - searching COP for servers 14
  - single sign-on 12, 13
- server port settings
  - options.cfg file 21
- server\_logging.log file 30
- single sign-on 11
- SQL
  - duplicate column names 46
  - optimizing Oracle 49
  - previewing generated 39
  - queries 46
  - viewing for model nuggets 39
- SQL generation 35, 36, 40
  - CLEM expressions 39, 44
  - enabling 38
  - enabling for IBM SPSS Modeler Server 23
  - logging 39
  - previewing 39
  - stream rewriting 39
  - tips 39
  - viewing for model nuggets 39
- SQL pushback. See also SQL generation 35
- SQL Server
  - SQL optimization 35, 36
- sql\_generation\_enabled
  - options.cfg file 23
- SSL 26
  - overview 26
  - securing communications 26
- SSL data encryption
  - enabling for IBM SPSS Modeler Server 23
- ssl\_certificate\_file
  - options.cfg file 23
- ssl\_enabled
  - options.cfg file 23
- ssl\_private\_key\_file
  - options.cfg file 23
- ssl\_private\_key\_password
  - options.cfg file 23
- starting IBM SPSS Modeler Server
  - on UNIX 17
  - on Windows 17
- status
  - of IBM SPSS Modeler Server on UNIX 17
  - of IBM SPSS Modeler Server on Windows 17
- stopping IBM SPSS Modeler Server
  - on UNIX 17
  - on Windows 17
- stream rewriting 39
- stream\_rewriting\_enabled
  - options.cfg file 21

## T

- temp directory
  - for IBM SPSS Modeler Server 21
- temp\_directory
  - options.cfg file 21
- temporary files 7
  - permissions for (IBM SPSS Modeler Server) 16

## U

- UNC filenames 9
- UNIX
  - configuring file permissions 52
  - library paths 57
  - permissions 15
  - single sign-on 13
  - user authentication 15
- UNIX kernel limits 17
- UNIX scripts
  - auto.sh 51
  - editing 52
  - modelersrv.sh 51
  - rc.modeler 51
- user accounts
  - IBM SPSS Modeler Server 15
  - permissions 15
- user authentication 15
- user ID
  - IBM SPSS Modeler Server 11

## Z

- zombie processes, IBM SPSS Modeler Server 18









Printed in USA