



TIBCO Spotfire® Statistics Services Installation and Administration

*Software Release 7.11.2
April 2019*

Important Information

SOME TIBCO SOFTWARE EMBEDS OR BUNDLES OTHER TIBCO SOFTWARE. USE OF SUCH EMBEDDED OR BUNDLED TIBCO SOFTWARE IS SOLELY TO ENABLE THE FUNCTIONALITY (OR PROVIDE LIMITED ADD-ON FUNCTIONALITY) OF THE LICENSED TIBCO SOFTWARE. THE EMBEDDED OR BUNDLED SOFTWARE IS NOT LICENSED TO BE USED OR ACCESSED BY ANY OTHER TIBCO SOFTWARE OR FOR ANY OTHER PURPOSE.

USE OF TIBCO SOFTWARE AND THIS DOCUMENT IS SUBJECT TO THE TERMS AND CONDITIONS OF A LICENSE AGREEMENT FOUND IN EITHER A SEPARATELY EXECUTED SOFTWARE LICENSE AGREEMENT, OR, IF THERE IS NO SUCH SEPARATE AGREEMENT, THE CLICKWRAP END USER LICENSE AGREEMENT WHICH IS DISPLAYED DURING DOWNLOAD OR INSTALLATION OF THE SOFTWARE (AND WHICH IS DUPLICATED IN THE LICENSE FILE) OR IF THERE IS NO SUCH SOFTWARE LICENSE AGREEMENT OR CLICKWRAP END USER LICENSE AGREEMENT, THE LICENSE(S) LOCATED IN THE "LICENSE" FILE(S) OF THE SOFTWARE. USE OF THIS DOCUMENT IS SUBJECT TO THOSE TERMS AND CONDITIONS, AND YOUR USE HEREOF SHALL CONSTITUTE ACCEPTANCE OF AND AN AGREEMENT TO BE BOUND BY THE SAME.

ANY SOFTWARE ITEM IDENTIFIED AS THIRD PARTY LIBRARY IS AVAILABLE UNDER SEPARATE SOFTWARE LICENSE TERMS AND IS NOT PART OF A TIBCO PRODUCT. AS SUCH, THESE SOFTWARE ITEMS ARE NOT COVERED BY THE TERMS OF YOUR AGREEMENT WITH TIBCO, INCLUDING ANY TERMS CONCERNING SUPPORT, MAINTENANCE, WARRANTIES, AND INDEMNITIES. DOWNLOAD AND USE OF THESE ITEMS IS SOLELY AT YOUR OWN DISCRETION AND SUBJECT TO THE LICENSE TERMS APPLICABLE TO THEM. BY PROCEEDING TO DOWNLOAD, INSTALL OR USE ANY OF THESE ITEMS, YOU ACKNOWLEDGE THE FOREGOING DISTINCTIONS BETWEEN THESE ITEMS AND TIBCO PRODUCTS.

This document is subject to U.S. and international copyright laws and treaties. No part of this document may be reproduced in any form without the written authorization of TIBCO Software Inc.

TIBCO, Two-Second Advantage, TIBCO Spotfire, TIBCO Enterprise Runtime for R, TIBCO Spotfire Server, TIBCO Spotfire Web Player, TIBCO Spotfire Statistics Services, S-PLUS, and TIBCO Spotfire S+ are either registered trademarks or trademarks of TIBCO Software Inc. in the United States and/or other countries.

Enterprise Java Beans (EJB), Java Platform Enterprise Edition (Java EE), Java 2 Platform Enterprise Edition (J2EE), and all Java-based trademarks and logos are trademarks or registered trademarks of Oracle Corporation in the U.S. and other countries.

All other product and company names and marks mentioned in this document are the property of their respective owners and are mentioned for identification purposes only.

This software may be available on multiple operating systems. However, not all operating system platforms for a specific software version are released at the same time. Please see the readme.txt file for the availability of this software version on a specific operating system platform.

THIS DOCUMENT IS PROVIDED "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NON-INFRINGEMENT.

THIS DOCUMENT COULD INCLUDE TECHNICAL INACCURACIES OR TYPOGRAPHICAL ERRORS. CHANGES ARE PERIODICALLY ADDED TO THE INFORMATION HEREIN; THESE CHANGES WILL BE INCORPORATED IN NEW EDITIONS OF THIS DOCUMENT. TIBCO SOFTWARE INC. MAY MAKE IMPROVEMENTS AND/OR CHANGES IN THE PRODUCT(S) AND/OR THE PROGRAM(S) DESCRIBED IN THIS DOCUMENT AT ANY TIME.

THE CONTENTS OF THIS DOCUMENT MAY BE MODIFIED AND/OR QUALIFIED, DIRECTLY OR INDIRECTLY, BY OTHER DOCUMENTATION WHICH ACCOMPANIES THIS SOFTWARE, INCLUDING BUT NOT LIMITED TO ANY RELEASE NOTES AND "READ ME" FILES.

This and other products of TIBCO Software Inc. may be covered by registered patents. Please refer to TIBCO's Virtual Patent Marking document (<https://www.tibco.com/patents>) for details.

Copyright © 1999-2019 TIBCO Software Inc. All rights reserved.

Contents

Figures	9
Documentation and support services	10
Recommendations for safeguarding your environment	13
Administration overview	14
Introduction to TIBCO Spotfire Statistics Services	14
Server features	14
Compatible engines	15
Supporting software	15
Supported platforms and system requirements	16
Server architecture	16
Job process workflow	17
Client APIs	18
General client developers	18
R developer package deployment	19
Package availability on the server	20
Package development references	20
Installation types	20
Standalone installation	21
Cluster installation	21
Standalone server installation	23
Preparing for installing a standalone service	23
Establishing log on as a service right	23
Establishing permissions to write to a directory	24
Installing standalone TIBCO Spotfire Statistics Services on Windows	24
Installing standalone TIBCO Spotfire Statistics Services on Linux	26
Migrating your Spotfire Statistics Services configuration and data	27
Post-installation configuration	28
Engine location	28
Changing the engine type	28
Changing the engine count on Windows	29
Changing the engine count on Linux	30
Simple no configuration installation	30
Validation	30
Cluster deployment	32
Cluster scaling design	33
Manager nodes	34

Cluster installation checklist	34
Cluster file share	35
Installing the first manager node on Windows	35
Installing the first manager node on UNIX or Linux	35
Running the installer	36
Migrating your Spotfire Statistics Services configuration and data	37
Post-installation configuration settings	38
Granting service access	39
Register init scripts	39
Set the job database	39
Change the engine type in a cluster	39
Specify engine count for the manager node	40
Start and validate the service	41
Add nodes	41
Installing an additional node on Windows	41
Installing an additional node on UNIX or Linux	42
Validate nodes	43
Load balancer	43
Configuring the load balancer	44
Restricting the load balancer	45
Automate cluster installations	46
Recording and running a response file	46
Response file template	46
Service configuration and properties	49
Simple no configuration installation	49
TIBCO Spotfire Statistics Services data files	49
Configuration and log files	50
Configure service logging	50
On-the-fly logging	51
Configure engine logging	51
Creating a log4j engine logging file	52
Server properties	52
authentication.required	53
cluster.jms.port	53
cluster.multicast.group.address	53
cluster.multicast.port	54
db.cleanup.run.interval.minutes	54
db.cleanup.time.to.live.days	55
email.from	55

email.host	55
email.reply.to	55
engine.appdata.dir	55
engine.init.expr	56
engine.init.file	56
engine.java.options	56
engine.max.elapsed.seconds	57
engine.max.requests	57
engine.startup.timeout.seconds	57
engine.type	58
engine.verbose.log	58
expression.service.enabled	58
function.service.whitelist	59
isclustered	60
jmxremote.enabled	61
jmxremote.password	61
jmxremote.port	61
job.list.xslt.enabled	61
max.jobs.to.return	61
notification.delay.time	62
notification.max.times	62
service.id	62
service.url	62
terr.restricted.execution.mode	63
webdav.cleanup.run.interval.minutes	64
webdav.cleanup.time.to.live.minutes	64
xml.version	65
Authentication in TIBCO Spotfire Statistics Services	65
Determine authentication setting	65
Active Directory properties	66
activeDirectory.enabled	66
ldap.domain	66
ldap.host	67
LDAP authentication implementation	67
LDAP properties	68
ldap.accountNameAttr	68
ldap.basedn	68
ldap.customSearchFilter	68
ldap.enabled	69

ldap.groupMemberAttr	69
ldap.groupRoleAttr	69
ldap.groupnum	70
ldap.host	70
ldap.manager.anonymous	71
ldap.manager.cn	71
ldap.manager.pass	72
ldap.path	72
ldap.port	72
ldap.protocol	73
ldap.searchBase	73
ldap.searchSubTree	73
Configuring LDAPS	74
SSL configuration	74
Server environment properties	75
WebDAV security configuration	76
Remote monitoring and management with JMX	76
Enabling or disabling remote monitoring and management with JMX	77
Changing the username or password for remote monitoring and management	77
Modifying the JMX port number	77
Changing the protocol or port	78
Configuring an open-source R engine	78
Configuring an S-PLUS engine	79
Configuring a SAS engine	80
Configuring a MATLAB engine	81
Configuring TIBCO Spotfire to use TIBCO Spotfire Statistics Services	82
Engine affinity	83
Database configuration	85
Job database requirements	85
Configure the H2 database server	85
Configuring a non-default database	85
Configure the package for a Teradata Aster database	86
Configuring the AsterDB package	86
Configuring the JDBC driver	87
Database properties	87
database.maxActive	88
database.maxIdle	89
database.type	89
database.url	89

database.username	90
database.password	91
Service administration	92
Stopping the service on Microsoft Windows	92
Starting the service on Microsoft Windows	92
Uninstalling the server from Windows	93
Configuring the service to run at start up on UNIX or Linux	93
Starting the service on UNIX or Linux	93
Stopping the service on UNIX or Linux	93
Restarting the service on UNIX or Linux	94
Uninstalling the server from UNIX or Linux	94
Deleting protected packages	95
Monitor the service	95
Using Procrun to modify a Java property	96
Troubleshoot the service	96
MySQL dropped connections	96
Troubleshoot R engine failures	96

Figures

Example engine clusters configured by engine type.	17
Spotfire Statistics Services with single manager server	21
Spotfire Statistics Services single-manager server with workers.	32
Spotfire Statistics Services two manager servers with load balancer.	33
Spotfire Statistics Services ultimate cluster deployment with failover.	34

Documentation and support services

How to Access TIBCO Documentation

Documentation for TIBCO products is available on the TIBCO Product Documentation website, mainly in HTML and PDF formats.

The TIBCO Product Documentation website is updated frequently and is more current than any other documentation included with the product. To access the latest documentation, visit <https://docs.tibco.com>.

TIBCO Spotfire Statistics Services documentation

The following documents for this product can be found in the TIBCO Documentation Library:

- *Release Notes*
- *System Requirements*
- *Installation and Administration*
- *Package Management for the TIBCO Spotfire® Environment*
- *User's Guide*
- *Java API Reference*
- *C# API Reference*
- *URL API Reference*
- *SpotfireUtils Package Reference*
- *Release Notes*
- *License Agreement*

TIBCO Enterprise Runtime for R documentation

The following documents for TIBCO® Enterprise Runtime for R can be found in the TIBCO Product Documentation website.

- *TIBCO® Enterprise Runtime for R Technical Documentation*
- *Language Reference (HTML)*
- *Differences Between TIBCO® Enterprise Runtime for R and Open-Source R (HTML)*
- *Release Notes (PDF)*
- *License Agreement (PDF)*

The following documents for integrating with third-party products are also available in PDF format.

- *Technical Note: Configure KNIME to use TIBCO® Enterprise Runtime for R*
- *Technical Note: Configure RStudio to use TIBCO® Enterprise Runtime for R*
- *Technical Note: Configure SparkR to use TIBCO® Enterprise Runtime for R*

Spotfire documentation

The following documents for this product can be found in the TIBCO Documentation Library for TIBCO Spotfire® Analyst and TIBCO Spotfire® Server:

- *TIBCO Spotfire® Analyst User's Guide*

- *TIBCO Spotfire® API Reference*
- *TIBCO Spotfire® Administration Manager User's Guide*
- *TIBCO Spotfire® Cobranding*
- *Running TIBCO Spotfire® on Citrix XenApp™*
- *TIBCO Spotfire® Server Release Notes*
- *TIBCO Spotfire® License Agreement*

TIBCO Spotfire Server documentation

The following documents for this product can be found in the TIBCO Documentation Library:

- *TIBCO Spotfire® Server and Environment Installation and Administration*
- *TIBCO Spotfire® Server and Environment Basic Installation*
- *TIBCO Spotfire® Cobranding*
- *TIBCO Spotfire® Server Web Services API*
- *TIBCO Spotfire® Server Server Platform API*
- *TIBCO Spotfire® Server Information Services API*
- *TIBCO Spotfire® Server REST API*
- *TIBCO Spotfire® Server Release Notes*
- *TIBCO Spotfire® Server License Agreement*

System Requirements for Spotfire Products

For information about the system requirements for Spotfire products, visit <http://spotfi.re/sr>.

Release Version Support

Some release versions of TIBCO Spotfire products are designated as long-term support (LTS) versions. LTS versions are typically supported for up to 36 months from release. Defect corrections will typically be delivered in a new release version and as hotfixes or service packs to one or more LTS versions. See also <https://docs.tibco.com/products/tibco-spotfire-general>.

How to Contact TIBCO Support

You can contact TIBCO Support in the following ways:

- For an overview of TIBCO Support, visit <http://www.tibco.com/services/support>.
- For accessing the Support Knowledge Base and getting personalized content about products you are interested in, visit the TIBCO Support portal at <https://support.tibco.com>.
- For creating a Support case, you must have a valid maintenance or support contract with TIBCO. You also need a user name and password to log in to <https://support.tibco.com>. If you do not have a user name, you can request one by clicking Register on the website.

How to Join TIBCO Community

TIBCO Community is the official channel for TIBCO customers, partners, and employee subject matter experts to share and access their collective experience. TIBCO Community offers access to Q&A forums, product wikis, and best practices. It also offers access to extensions, adapters, solution accelerators, and tools that extend and enable customers to gain full value from TIBCO products. In addition, users can submit and vote on feature requests from within the [TIBCO Ideas Portal](https://community.tibco.com). For a free registration, go to <https://community.tibco.com>.

For quick access to TIBCO Spotfire content, see <https://community.tibco.com/products/spotfire>.

Recommendations for safeguarding your environment

When installing and configuring Spotfire® Statistics Services to use a statistical engine, you can take steps to protect the deployment, to minimize the risk of unauthorized access, and to minimize the possibility of malicious acts.

Statistical engines provide functions to access data and packages on the internet. Additionally, they have functions that access the host computer system, such as those for executing system commands, and those for reading and writing files. By their very design, these languages can expose computer systems to risk from bad actors, unless the deployer takes steps to secure the environments in which they run. We strongly recommend reviewing and implementing the practices described in these topics.

Protecting the deployment

- Always run Spotfire Statistics Services behind a firewall to hide its ports. Using a firewall and/or configuring the operating system to limit accessible ports ensures that the system and Spotfire Statistics Services are not exposed externally.
- Limit access (using the firewall) to ONLY those clients necessary. For example, if the intended use of Spotfire Statistics Services is only to support running Spotfire Data Functions from the TIBCO Spotfire® Web Player, block all connections from any machine other than the Spotfire® Web Player server.

Restricting user access

- Run Spotfire Statistics Services using an account that limits network access to only required external data sources and services. (Note that taking this step can limit availability to data and package updates.)
- ALWAYS run the Spotfire Statistics Services server as non-root user on the host operating system. (That is, not as root or under an Administrative account.)
- If you are running a system where other servers have access to computers running Spotfire Statistics Services, disable passwordless access between the Spotfire Statistics Services server and other servers.

Configuring for tighter engine control

The following configuration settings are found in the file `conf/spserver.properties`.

- Carefully curate the white-listed functions listed in the server property `function.service.whitelist`. See [function.service.whitelist](#) on page 59.
- Preserve the default settings for using TERR in restricted mode (`expression.service.enabled=false` and `terr.restricted.execution.mode=true`). The configuration property `terr.restricted.execution.mode` applies only to deployments that are configured to use the TERR engine. For more information, see [expression.service.enabled](#) on page 58 and [terr.restricted.execution.mode](#) on page 63.

Administration overview

TIBCO Spotfire® Statistics Services can be deployed as a stand-alone server against which you can run analyses in one of its engines, or it can be deployed as part of a Spotfire stack.

Spotfire Statistics Services is a light-weight, flexible server that provides a communication layer, a service layer, and TIBCO® Enterprise Runtime for R (TERR™) or open-source R engine pool, among other features. Spotfire Statistics Services does not include user interface features.

You can also use external engines, such as S-PLUS, SAS®¹, or MATLAB®,² if you have access to the corresponding software.



Open-source R is available under separate open source software license terms and is not part of TERR. As such, open-source R is not within the scope of your license for TERR. Open-source R is not supported, maintained, or warranted in any way by TIBCO Software Inc. Download and use of open-source R is solely at your own discretion and subject to the free open source license terms applicable to open-source R.

Introduction to TIBCO Spotfire Statistics Services

Spotfire Statistics Services builds on the comprehensive library of statistical and mathematical algorithms provided by TERR engines (or other engines) by offering enterprise-ready features.

These features include the following.

- Centralized management.
- Event logging.
- Support for integration through a variety of programming languages.
- Authentication and authorization.
- Clustering.
- Load balancing.
- The capacity for multiple nodes to handle client requests.

Users submit jobs to an engine through the server, control job execution, and retrieve results. Administrators configure and maintain a cluster or a standalone installation.



A Spotfire Statistics Services job is an arbitrary expression (script) or function, bundled with the data it is to execute upon.

Even though external engines based on S-PLUS, SAS®, or MATLAB® software can be used to provide calculations, this documentation is primarily focused on the use of TERR or open-source R engines. For more information on other engines, see [Changing the engine type](#).

Server features

This list contains some of the most important features of the Spotfire Statistics Services architecture:

- Simple and lightweight. Users can pass data to the server, execute code on the server, and retrieve results.
- Easy to install, configure, and maintain. The engines run as separate, restartable processes.
- Can be extended through executing a script or function written in TERR, open-source R, S-PLUS, SAS, or MATLAB, and submitted through Java™, C#, or the URL API. In this way, the code remains separate from Spotfire Statistics Services.

¹ SAS and all other SAS Institute Inc. product or service names are registered trademarks or trademarks of SAS Institute Inc. in the USA and other countries. ® indicates USA registration.

² MATLAB is a trademark or registered trademark of The MathWorks, Inc.

- Configurable to access multiple TERR engines, open-source R engines, S-PLUS, SAS, or MATLAB engines on each server. You can easily install and configure the server in a cluster. After you establish the cluster, you can install new Worker nodes as needed. Each new Worker node immediately begins processing requests from the centralized job queue.
- Version 7.11.2 of Spotfire Statistics Services is designed to work with open-source R version 3.4.4.
- Provides centralized management and access. The computational resources are exposed as RESTful services. You can deploy commonly used libraries and store all execution results in a centralized location.
- Provides configurable user and permission settings. You can configure the settings so that credentials are evaluated before certain actions can be performed. Spotfire Statistics Services supports different types of data stores that can hold the user and permission information, for example a property file or an LDAP directory.
- Adheres to external standards whenever possible, which makes the server easier to use.
- Robust and flexible. This enables you to create high quality user interfaces that are tailored to your requirements.

Compatible engines

Learn which engines you can use with Spotfire Statistics Services. Learn the engine limitations and considerations.

Engines

You can use either TERR, S-PLUS, or open-source R as the engine in both standalone and cluster installations. In a standalone or an individual cluster, you must have just one engine type. For more information about maintaining, updating, or deploying engines in the server, see [Post-installation configuration](#) and [Change the engine type](#).



If you have developers who plan to use the package RinR, which is provided with TERR, you must configure Spotfire Statistics Services to use the TERR engine, but you must also ensure that the open-source R engine is installed on the server (but not configured as the engine to use).



Open-source R is available under separate open source software license terms and is not part of TERR. As such, open-source R is not within the scope of your license for TERR. Open-source R is not supported, maintained, or warranted in any way by TIBCO Software Inc. Download and use of open-source R is solely at your own discretion and subject to the free open source license terms applicable to open-source R.

SAS® and MATLAB® Engines

If you have a working SAS software or MATLAB installation you can use that software as the engine in both standalone and cluster installations of Spotfire Statistics Services. However, within each cluster, you can use only one engine type. For more information, see [Configuring a SAS engine](#) or [Configuring a MATLAB engine](#).

Supporting software

Spotfire Statistics Services is packaged in a single installer file that includes third-party products.

- Java Runtime Environment 8 Update 181.
- Apache Tomcat 8.5.32.

The following software is useful but it is not required:

- Apache Chainsaw v2 log viewer.
- Apache HTTP Server or Microsoft Internet Information Services (IIS) as a front end or load balancer.

Supported platforms and system requirements

Your system must meet the basic requirements to work with Spotfire Statistics Services.

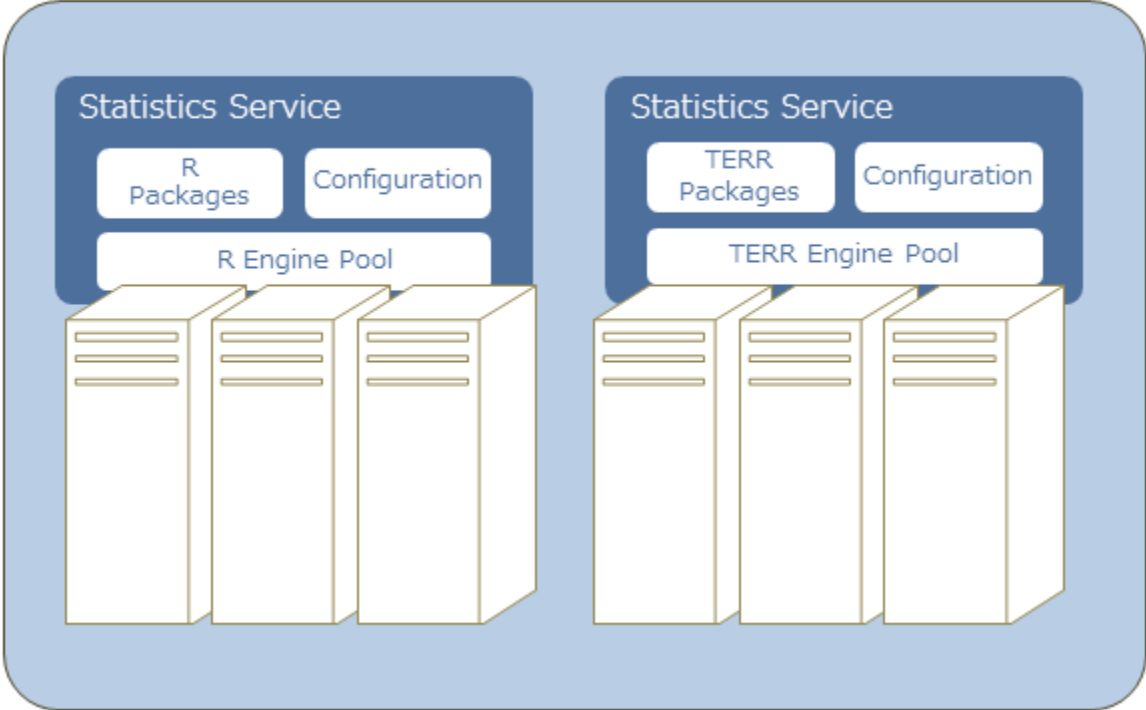
Before you install Spotfire Statistics Services, verify that your system complies with the latest system requirements listed at [TIBCO Spotfire® Products System Requirements](#).

Refer to the *TIBCO Spotfire® Statistics Services Release Notes* for changes in this version.

Server architecture

TIBCO Spotfire® Statistics Services is a web service application that exposes a set of resources that allow a user to interact with an R engine. This section provides an overview of the components that make up the server architecture.

TIBCO Spotfire® Statistics Services architecture component	Description
Processing runtime	<p>The processing runtime is the core set of components included in every Spotfire Statistics Services installation.</p> <p>The Processing Runtime implements the functionality of the server by accepting and processing jobs (requests submitted by supported APIs), providing support for node administration and maintenance, performing event logging to provide full traceability, and retrieving job results. The engine runs in the context of the processing runtime.</p>
Data access layer	<p>The data access layer enables communication to a job database on a database server. When a job is persisted in the job database, a user can access that job as needed.</p>
Job queue	<p>The job queue handles job scheduling of jobs in the job database. Users can schedule and monitor jobs.</p>
Periodic cleanup	<p>The periodic cleanup component makes sure that the necessary maintenance tasks can and do run. You can configure settings related to the preservation or deletion of job definitions and job artifacts, which are results from the execution of jobs.</p>
Configuration store	<p>The configuration store is a component that manages server configuration settings. This component stores the current configuration in the <code>spserver.properties</code> file and provides JMX access to the current configuration.</p>

TIBCO Spotfire® Statistics Services architecture component	Description
Engine pool	<p>The engine pool component manages the engines. You can run multiple engines in an engine pool, but you cannot mix engines of different types. For example, you cannot run an open-source R engine and a TERR engine in a single engine pool. The following image shows two clusters, each configured to run a specific engine type.</p> <p><i>Example engine clusters configured by engine type.</i></p> 

Job process workflow

Understanding how Spotfire Statistics Services processes jobs can help you troubleshoot problems or just understand how the results are derived. The basic steps of the typical job process and the expected results you can expect are described in this topic.

1. A client uploads the data necessary to complete a job to the server using a WebDAV service and submits a job request to a Manager node using a web request.
2. Execution is asynchronous, and the Manager node stores the job in the job database, places it in a queue for execution, and marks the job as waiting.
3. The job ID and the status are returned to the client.

Using the job ID, a client can manage the job and view the status.

4. When the job is submitted to a Worker node, the status is changed to running.
5. When the job completes, the client is notified and, if it completed successfully, the client can retrieve the results.

If an error occurred, the status is set to either failed or done with error.

- Done with error means that an error occurred while parsing and executing the job inside the engine.
- Failed means that an unexpected internal error occurred while processing the job.)



The client APIs provide for synchronous execution. The server achieves this by blocking the call until the job completion notification is received from the server. Synchronous jobs have a higher priority over asynchronous jobs.

When you deploy Spotfire Statistics Services as part of your Spotfire deployment, Spotfire Analyst or Spotfire Web Player is the client sending the job request to the cluster. The Manager node passes it on to an available Worker node, and then returns the result when the job is complete.

Client APIs

TERR, open-source R, S-PLUS, and client application developers can write applications to integrate with the server using one of the Spotfire Statistics Services client APIs.

Spotfire Statistics Services includes the following client APIs:

- Java™
- C#
- URL API

The Java and C# APIs provide support for managing code packages in a server package repository. The APIs contain essentially the same functionality with the exception of differences related to special language constructs that are available only in C#, such as properties, events, and delegates. A Java developer can implement the equivalent of these interfaces in Java as getter/setter pairs, single method interfaces, and classes.

These client APIs provide a layer that shields the client application developers from the mechanics of accessing the underlying web services exposed by the server applications.

You can access some of the underlying RESTful resources in other programming languages directly by passing in the request as an HTTP request to the URL API.

Clients can support both synchronous and asynchronous method invocations. In the case of the default asynchronous invocation, the client submits the request to a server and control returns to the client code after the server adds the request to a job queue. The client code can then register a callback method that can be invoked when the server completes the execution of the job.

In the synchronous invocation case, the client submits the request to the server exactly as in the above case, but instead of returning control to the client code after the server sends an initial response, the server generates a response to the client call only upon completion of the job.

General client developers

Spotfire Statistics Services is a flexible system that developers can use to expand upon and build clients using the available programming interfaces.

Client developers depend on the administrator for obtaining the programming interfaces and associated documentation. Spotfire Statistics Services is designed to minimize the impact on administrators by making these tools and all documentation available to remote users from the Spotfire Statistics Services landing page. You need to provide to the client developers just the URL to the server landing page:

```
http://servername:port/<service_name>
```

From this URL, client developers can obtain the tools and documentation that they need to build a client application.

The links on the server landing page access the following files and directories in SPSEVER_HOME:

```
/docs
```

```
/ext
```

CsharpAPI

JavaAPI

URLAPI

R developer package deployment

An R developer can expand on the available functions and libraries by creating TERR functions, open-source R functions, or S-PLUS functions in packages for processing requests from their clients.

Typically, an R developer might want to download packages available from the CRAN repository, a curated public R package repository accessed through a web browser. Your company can easily establish and maintain an in-house repository to contain publicly-available and customized packages, administered by the administrator or by a specified package curator. One easy way to establish an in-house repository is by using the drat package, which is also available on CRAN. The drat package can establish a package repository on a web server or on a file system. See its help, included with the package, for more information.



Remember that when installing and configuring Spotfire Statistics Services to use a statistical engine, consider the configurations and strategies to restrict access. See [Recommendations for safeguarding your environment](#) on page 13 for detailed information.

The administrator and (if granted permissions) R developers can deploy, update, or remove packages on Spotfire Statistics Services using the following techniques.

- Logging in as administrator on the computer where Spotfire Statistics Services is installed, and opening the TERR or open-source R console application to call `install.packages()` or `remove.packages()`.
- Using the Administration APIs provided with Spotfire Statistics Services. The APIs include C# and Java versions. See their help for more information.
- Installing Eclipse, and then using the Eclipse plugin for TERR (available with Spotfire Statistics Services).



These tools do not support uploading or updating packages on a Spotfire Statistics Services server that has authentication enabled. If your Spotfire Statistics Services uses authentication, you should make your plugins and package repository available on a web server that does not require authentication. Be sure to provide the server location to your R developers.

See the *TIBCO® Enterprise Runtime for R Technical Documentation* on <https://docs.tibco.com/products/tibco-enterprise-runtime-for-r> for more detailed information about managing packages in your organization.

Spotfire Statistics Services supports only binary packages, which are pre-compiled and have a `.zip` extension on Windows or a `.tar.gz` extension on UNIX/Linux. Packages are not cross-platform compatible, so the developer must build the package specifically for each platform. In the case of open-source R binary packages, the developer must use an open-source R engine configured with the option `--enable-R-shlib` enabled.



Open-source R is available under separate open source software license terms and is not part of TERR. As such, open-source R is not within the scope of your license for TERR. Open-source R is not supported, maintained, or warranted in any way by TIBCO Software Inc. Download and use of open-source R is solely at your own discretion and subject to the free open source license terms applicable to open-source R.

Package availability on the server

R packages are deployed on the server by the package developer, who must decide when and how the package can be accessed by the code.

Developers can use the Administration APIs to deploy, update, or remove their packages from the server. In a clustered environment, when a package is deployed to a Manager node, the package is automatically made available to all Worker nodes in the cluster.

Every time a request is sent to the server, an available engine starts to process the request.

You can ensure that a package is available in the running engine in one of two ways:


- **Loads when needed:** To load the package only when an application requires it, the client developer can add a call to `library(packagename)` in the application prior to accessing any of the package functions.
- **Always loaded when an engine starts:** To ensure that a package always loads whenever an engine starts, set the server property `engine.init.expr` or `engine.init.file` to define initialization code. (To use `engine.init.file`, you must create an initialization file and specify its path in `engine.init.file`. See the description of this [server property](#) for more information.)



This method should be used only if needed, because loading a package every time the engine starts can have a detrimental effect on performance.

Package development references

The following resources can help developers creating and managing packages that use the TERR engine or the open-source R engine.

Document Name	Description
<i>TIBCO Spotfire® Package Management</i>	Contains guidance for managing packages developed for the TERR engine or downloaded from a CRAN-like repository. It also contains guidance for deploying these packages to the Spotfire® Server.
<i>Writing R Extensions</i>	Contains instructions for creating packages in open-source R.  Open-source R is available under separate open source software license terms and is not part of TERR. As such, open-source R is not within the scope of your license for TERR. Open-source R is not supported, maintained, or warranted in any way by TIBCO Software Inc. Download and use of open-source R is solely at your own discretion and subject to the free open source license terms applicable to open-source R.

Installation types

Installation types for Spotfire Statistics Services include standalone and cluster.

Before you begin the Spotfire Statistics Services installation process, you must determine whether you need a standalone installation or the scalability and fail-over features offered in a cluster.

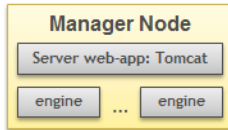
Spotfire Statistics Services is built from a basic set of reusable components as a true component-based application. From these basic components, we built a single installation file, a standard Java web application archive (WAR), for each supported platform. You can install either the standalone or the cluster server types. See [Cluster deployment](#) for more guidance in deciding which installation type you need.

Standalone installation

Determine if the typical standalone installation of Spotfire Statistics Services is the right installation for your needs.

In a standalone installation, you install and configure Spotfire Statistics Services on a single computer. This installation is ideal if you expect your users to send only a small number of jobs, and those jobs do not have large data sets. See [Preparing for installing a standalone service](#).

Spotfire Statistics Services with single manager server



Cluster installation

Determining if a cluster installation of Spotfire Statistics Services is the right installation for your needs.

In your plans for your deployment, if your requirements include the scalability, high-availability, and reliability that go beyond the benefits of a standalone installation, consider deploying Spotfire Statistics Services in a clustered environment.

You can install Spotfire Statistics Services Manager and Worker nodes on multiple computers so that they will operate together in a clustered environment. You can also configure a Manager node to fill the role of a Worker node so that it responds to administration requests and processes jobs. In this way a node can function as an entire system on a single computer or as many computers participating in a cluster.

Cluster Manager node

You must have at least one Manager node in each cluster, and it must be installed on a computer separate from the Worker nodes. When you install additional Manager or Worker nodes the configuration settings from this first Manager node is read in during the installation process.

The role of a Manager node is to expose a set of RESTful resources on the network that provide the ability to submit an expression or function for execution to an engine, enable remote job management, and configure and manage a cluster.

The Manager node also exposes a WebDAV service for remote file management. This service allows clients to upload a data set to a server prior to job execution and then, once the job is complete, they can obtain the files generated as an output of their job execution.

The Manager node has the following components.

Manager node component	Description
HTTP Request Handler	Translates HTTP requests into processing runtime or cluster management component calls.
Processing Runtime	Queues jobs and executes job requests.
WebDAV Service	Provides server directory and file access.
Cluster Management	Manages all of the nodes in a cluster and distributes load between nodes.

Cluster Worker node

The Worker node is a command-line application wrapped in a service that listens for commands from Manager nodes and processes queued jobs. A Worker node does not expose any externally accessible services other than those available through a Manager node.

The Worker node has the following components.

Worker node component	Description
Processing runtime	Queues jobs and executes job requests.
Cluster Agent	Listens for commands from the Manager node; Starts, stops, or restarts the processing runtime as needed; and returns job status change notifications.

For more information, see [Cluster deployment](#).

Standalone server installation

Select the Spotfire Statistics Services installation path that suits your needs.

For a stand-alone installation, you can install on a Windows server or a UNIX or Linux server.

If you need to deploy a cluster, see [Cluster deployment](#)

Preparing for installing a standalone service

Your computer must meet certain criteria for a Spotfire Statistics Services installation.

Procedure

1. Make sure that your computer meets the minimum system requirements. For more information, see <http://support.spotfire.com/sr.asp>.
2. For a Microsoft Windows® installation, determine whether to run the service using the local system account, or to run it under a service account.

- If the service does not need to access resources on a network, you can install using the local system account.
- If the service must access the network for resources, you must specify a service account.

For a Linux® installation, you have access to all resources that you normally have access to when you are logged on as the user of the account, so you do not need to specify this option.

3. Verify that the port numbers you specify for the **Main Service Port** and the **JMX Monitoring Port** during the Spotfire Statistics Services installation are not being used by other applications on the computer.



Remember that when installing and configuring Spotfire Statistics Services to use a statistical engine, consider the configurations and strategies to restrict access. See [Recommendations for safeguarding your environment](#) on page 13 for detailed information.

Establishing log on as a service right

On Microsoft Windows, If you need to specify a service account, ensure that the service account has the appropriate rights.

Procedure

1. On the computer where the standalone Spotfire Statistics Services is to be installed, log on with a user name that is a member of the local administrators group.
2. On the taskbar, click **Start > Control Panel > Administrative Tools > Local Security Policy**.
3. In the **Local Security Settings** dialog box, under **Security Settings**, expand **Local Policies**.
4. Select **User Rights Assignment**, and then double click **Log on as a Service**.
5. In the **Log on as a service Properties** dialog box, click **Add User or Group**, and then enter the service account you will specify during installation.



If you need to configure this option after you install Spotfire Statistics Services, you can do so, but you must stop the service before following these steps, and then restart the service.

6. Ensure the service account has all appropriate access permissions for any local or external resources (that is, the Spotfire Statistics Services installation directory, the domain resources, and so on).

Establishing permissions to write to a directory

On a Linux server, You must make sure that the user has appropriate permissions to install Spotfire Statistics Services.

Procedure

1. Log on as the user under which Spotfire Statistics Services is to run. This must be a valid user account. (Never log in as root. See [Security recommendations](#) for more information.)
2. Make sure that the user has permissions to write to the desired installation directory.
3. Follow the procedure in [UNIX or Linux installation](#).

Installing standalone TIBCO Spotfire Statistics Services on Windows

Follow these instructions to install a standalone version of Spotfire Statistics Services on a Windows computer.



Do not use this procedure if you intend to install a cluster. If you want to perform a cluster installation, see [Cluster deployment](#).



Remember that when installing and configuring Spotfire Statistics Services to use a statistical engine, consider the configurations and strategies to restrict access. See [Recommendations for safeguarding your environment](#) on page 13 for detailed information.

Procedure

1. From the Spotfire Statistics Services installation media, use an application appropriate for unpacking the installation .zip package, TIB_sf_statsvcs_7.11.2_win_x86_64.zip.
Review the accompanying release notes (if you have not done so already) and run the installation executable TSSS_7.11.2_win_x86_64.exe.
In the resulting Spotfire Statistics Services installation wizard, when you complete a step, click **Next** to proceed to the next panel.
2. **Introduction:** We recommend closing all running programs before proceeding with the installation.
3. **License Agreement:** Read the license agreement and if you agree to the terms, select **I accept the terms of the License Agreement**.
4. **Installation Type:** Select **Standalone**.
If you select **Cluster**, you should be following the instruction to install a cluster. See [Cluster deployment](#).
5. **Installation Path:** Either accept the default, type a path, or choose a path.
6. **Service Name:** Accept the default name (**SplusServer**) or type a different *service_name*.

The *service_name* is used in several places. It is used as the name of a new directory under the path that you typed in the Installation Path panel. For example, for a 64-bit Windows installation, the service name the path is:

C:\Program Files\TIBCO\statsvcs711\<service_name> where *service_name* is the value you supply, or the default, **SplusServer**.

The name of the Windows service is TSSS711<service_name>. The <service_name> is also appended to the service display name; for example, TIBCO Spotfire Statistics Services (SplusServer). It is also included in the server URL:

`http://servername:port/service_name/`



You can install multiple instances of Spotfire Statistics Services on a single computer, but you must make sure that the *service_name* for each service is unique.

7. **Ports** (panel 1): Specify the HTTP port that is used by clients or APIs to access the service.
 - a) For **Main Service Port**, either accept the default or type a port number for the service.
From here, we will refer to this value as *port*, as it appears in the server URL as `http://servername:port/service_name`. For example, `http://MyServer:8080/TERRServer`.
 - a) In **JMX Monitoring Port**, either accept the default or type a port number.
Remember that the port numbers must be unique for each Spotfire Statistics Services instance on the machine.
8. **Ports** (panel 2): Select **HTTP**.
If you plan to use a front-end proxy or load balancer, you should select **AJP**. For more information, see [Load balancer](#).
9. **Management User**: Provide the user name and password for authentication when using a JMX tool such as TIBCO Hawk or jconsole (for example). The default user name is `admin`.
These credentials are written to a server configuration file to provide for authentication when an authorized user accesses monitoring services. The value you provide here is encrypted when it is stored in the configuration file.

These credentials are specific to this Spotfire Statistics Services installation, and because they are stored in a server configuration file, we recommend not using your login credentials.
10. **Service URL**: This is the URL that accesses the Spotfire Statistics Services landing page. It is a concatenation of the server name, the *port*, and the *service_name*. For a standalone installation, just accept the default.
11. **Service User** (panel 1):
 - If all of your resources are on your local machine, and you want to configure the Spotfire Statistics Services (*service_name*) to run using the credentials for the Local System account, accept the default **Local System**.
 - If you must have access to resources on the network, select **Other Account**. (If you are not sure which option to select, review [Preparing for installing a standalone service](#).)
 - a) **Service User** (panel 2): If you selected **Other Account**, specify the service user and password. The specified account must have permission to access resources required for Spotfire Statistics Services.
12. **Pre-Installation Summary**: Review your options, and then click **Install**.
The **Installing Spotfire Statistics Services** (*service_name*) panel appears while the server is installed and configured.
13. **Start Service**: If **Start Service** is selected, Spotfire Statistics Services (*service_name*) is started when installation completes. If you do not want the service to start at this time, clear **Start Service**.

If the Service User you specified in step 12 has not been granted the Log on as a service right, clear **Start Service** or grant the user the Log on as a service right. See [Preparing for installing a standalone service](#) for more information.

If you need to change the number of engines configured on the server, see [Changing the engine count on Windows](#). (You must stop the service to change the engine count.)

The Spotfire Statistics Services (*service_name*) service is set to start automatically, which means that the service starts when you reboot your computer.
14. **Install Complete**: Click **Done**. If you want to set some post-installation configuration options, such as changing the engine type or count, stop the service, and then restart it following your changes.

Installing standalone TIBCO Spotfire Statistics Services on Linux

Follow these instructions to install a standalone version of TIBCO Spotfire® Statistics Services on a Linux or a UNIX computer.



Do not use this procedure if you intend to install a cluster. If you want to perform a cluster installation, see [Cluster deployment](#).



Remember that when installing and configuring Spotfire Statistics Services to use a statistical engine, consider the configurations and strategies to restrict access. See [Recommendations for safeguarding your environment](#) on page 13 for detailed information.

Procedure

1. From the installation media, unpack Spotfire Statistics Services .tar.
The operating system executable for 64-bit Linux is
`TIB_sf_statsvcs_7.11.2_linux24gl23_x86_64.tar`
2. Review the release notes (if you have not already done so), and then run `.bin` to install.
3. Follow the steps in the Spotfire Statistics Services installation wizard. (As you complete steps in the console, press ENTER to proceed to the next step.)
4. **License Agreement:** Read the license agreement and if you agree to the terms, select **I accept the terms of the License Agreement**.
5. **Installation Type:** Select **Standalone**.
If you select **Cluster**, you should be following the instruction to install a cluster. See [Cluster deployment](#).
6. **Installation Path:** Either accept the default, type a path, or choose a path.
7. **Ports** (panel 1): Specify the HTTP port that is used by clients or APIs to access the service.
 - a) For **Main Service Port**, either accept the default or type a port number for the service.
From here, we will refer to this value as `port`, as it appears in the server URL as `http://servername:port/service_name`. For example, `http://MyServer:8080/TERRServer`.
 - a) In **JMX Monitoring Port**, either accept the default or type a port number.
Remember that the port numbers must be unique for each Spotfire Statistics Services instance on the machine.
8. **Ports** (panel 2): Select **HTTP**.
If you plan to use a front-end proxy or load balancer, you should select **AJP**. For more information, see [Load balancer](#).
9. **Management User:** Provide the user name and password for authentication when using a JMX tool such as TIBCO Hawk or jconsole (for example). The default user name is `admin`.
These credentials are written to a server configuration file to provide for authentication when an authorized user accesses monitoring services. The value you provide here is encrypted when it is stored in the configuration file.

These credentials are specific to this Spotfire Statistics Services installation, and because they are stored in a server configuration file, we recommend not using your login credentials.
10. **Service URL:** This is the URL that accesses the Spotfire Statistics Services landing page. It is a concatenation of the server name, the port, and the `service_name`. For a standalone installation, just accept the default.
11. **Service User:** Accept the default. The service runs in the context of the user that you are using to run this installer. The specified account must have permission to access any other Spotfire Statistics Services resources. (See [Preparing for installing a standalone service](#) for more information.)

12. **Pre-Installation Summary:** Review your options, and then click **Install**.
The **Installing Spotfire Statistics Services (service_name)** panel appears while the server is installed and configured.
13. Optional: Run the following command as root: `SPSERVER_HOME/init.d/register-service.sh`. Alternately, you can configure post-installation options before completing this step.



On UNIX/Linux, you must configure the service to run when the computer starts by registering the init scripts with system init.

14. Manually start the service by running the following command: `SPSERVER_HOME/init.d/spserver start`.

Migrating your Spotfire Statistics Services configuration and data

Upgrading from a previous release of Spotfire Statistics Services requires a new installation. However you can retain and migrate configuration and data.

We recommend shutting down the previous version service and installing the new version in a different location. After you have installed the new version, follow this guidance.

Prerequisites

Install the new version, test that the service works, and then shut down the new version's service before completing these steps.

Procedure

1. Copy any `.jar` files from your endorsed directory to your new installation.
2. From the older installation, open the folder `SPSERVER_HOME/conf` and copy the appropriate configuration and data from the old installation to the new.

For most configuration and data, you can migrate from the previous installation to the new one, with the cautions and exceptions listed in here.



Do not modify the following entries in the file `conf/server.properties`:

- `service.id`
- `service.url`

Do not modify the following properties, unless you need to change them from the values you selected during installation.

- `jmxremote.port`
- `jmxremote.password`
- `jmxremote.username`

If you are upgrading from a version of Spotfire Statistics Services earlier than version 6.0, do not copy the older versions of the files `conf/database.properties` and `conf/ldap.properties`. These files have changed. Specifically, do not copy the entry `database.url` in the file `conf/database.properties` from the older installation. Instead, just change the `SPSERVER_SHARE` location to point to the new installation.

If you are installing a cluster, do not copy the file `conf/cluster.properties`.

3. Locate previous user data in `SPSERVER_HOME/data/common` and `TSSS_HOME/data/users`, and then copy these directories directly from the previous installation to the new installation.



All other directories under `SPSERVER_HOME/data` contain temporary data from the previous installation and should not be copied.

What to do next

Complete any remaining configuration tasks. If no further tasks are needed, start the service.

Post-installation configuration

After you have finished running the Spotfire Statistics Services installer, you can make minor configuration adjustments.

After completing the installation, you have the option to change the engine count and (on UNIX/Linux) configure the service to start when the machine is restarted. (On Windows, the service automatically starts when you reboot the computer.)

Engine location

Whether you deploy a stand-alone or a cluster Spotfire Statistics Services installation, the engines reside in compressed files in a location from where the server can deploy them automatically, and from where they can be easily referenced, maintained, and updated.

The engines are stored in the directory listed below, according to your installation type.

- **Stand-alone engine directory:** `SPSERVER_HOME/data/binaries`.
- **Cluster engine directory:** `SPSERVER_SHARE/data/binaries`.

The following table shows the compressed engine types by operating system.

Windows	UNIX/Linux
<code>TERREngine.zip</code>	<code>TERREngine.tar.gz</code>
<code>SplusEngine.zip</code>	<code>SplusEngine.tar.gz</code>



Spotfire Statistics Services provides the TERR engine by default. You must download and maintain the open-source R engine separately. See [Configuring an open-source R engine](#) for more information.

Changing the engine type

You can change the engine that Spotfire Statistics Services uses as part of your post-installation configuration.



By default, during installation, the engine type is set to TERR. Changing the engine type is optional. You can specify only one engine type.



Remember that when installing and configuring Spotfire Statistics Services to use a statistical engine, consider the configurations and strategies to restrict access. See [Recommendations for safeguarding your environment](#) on page 13 for detailed information.

Procedure

1. Browse to the directory `SPSERVER_HOME/conf/` (where `SPSERVER_HOME` is your server installation directory).
2. Using a text editor, open the file `spserver.properties`.
3. Locate the property `engine.type`, and set it to the value to use.

Engine Type	Description
TERR	TERR engine. The default. Provided with your Spotfire Statistics Services installation.
R	Open-source R engine. Use this engine if developers are deploying R packages to the service, and the packages contain R code that will not run in TERR. Requires additional steps. See Configure an open-source R engine for more information.  In Spotfire Statistics Services version 7.11.2, open-source R is configurable.
S-PLUS	S-PLUS engine. Provided with your Spotfire Statistics Services installation. Use this engine if developers are deploying S-PLUS packages to the service.  This engine is deprecated and will be removed in future releases.
SAS	SAS® engine. Requires additional steps. See Configure a SAS engine for more information.
MATLAB	MATLAB® engine. Requires additional steps. See Configure a MATLAB engine for more information.

When you update or change the `engine.type` property, you must restart the server to initiate the deployment.

For clusters, when the server starts, the engine is copied from its location and deployed to each node in the cluster.

For more information about the location of the engines, see [Engine location](#).

Changing the engine count on Windows

The engine count specifies the number of engines configured on the server to handle engine-related requests. For a standalone Spotfire Statistics Services installation, by default, the engine count is set to the number of cores on the server.

If you want your standalone installation to handle other tasks, you can change processing resources by setting the Java property `-Dengine.count`. On Windows, we use the Procrun service to add or modify Java properties.

Procedure

1. Browse to the directory `SPSERVER_HOME/tomcat/bin` (where `SPSERVER_HOME` is your server installation directory).
2. Open `TSSS<service_name>.exe`, where `service_name` is the value that you specified during installation.
3. Select the **Java** tab.
4. Under **Java Options**, add the specific Java property `-Dengine.count=val` where `val` is the desired value.
5. Start or Restart the service.

Changing the engine count on Linux

The engine count specifies the number of engines configured on the server to handle engine-related requests. For a standalone Spotfire Statistics Services installation, by default, the engine count is set to the number of cores on the server.

Procedure

1. Edit the script file `SPSERVER_HOME/init.d/spserver` (where `SPSERVER_HOME` is your server installation directory).
2. Set the value of `SPSERVER_ENGINES` to the desired value.
3. Start or restart the service.

Simple no configuration installation

A simple standalone Spotfire Statistics Services installation requires no additional configuration.

If you install a standalone Spotfire Statistics Services on a computer that meets the minimum requirements, you require no post-installation configuration. (See <http://support.spotfire.com/sr.asp> for up-to-date information about system requirements.)

You can customize the configuration after installing Spotfire Statistics Services and check that it is operating normally. To verify that your installation is running normally, see [Validation](#) on page 30.

Review the service property information to determine which settings to customize. For example, by default, the configuration file requires no authentication. If you want to enable authentication, change the default value to `true`.



By default, if your engine type is TERR, then TERR is set to use restricted execution mode. To change the properties that control this option, you must edit the `spserver.properties` configuration file. See [Server properties](#) for more information.

Validation

After you complete the Spotfire Statistics Services installation, we recommend that you validate the installation to ensure that Spotfire Statistics Services is operating normally.

To verify that Spotfire Statistics Services is operating normally, open a web browser and, in the address bar, send a request using the following URLs. These URLs exercise all services: Administration, Expression, Function, and WebDAV.

URL	Description
<code>http://servername:port/service_name</code>	Returns the Spotfire Statistics Services landing page with links to the documentation and programming interfaces.
<code>http://servername:port/service_name/api/v8/administration/ExtendedServerInfo</code>	Displays server version information useful for support purposes.
<code>http://servername:port/service_name/api/v8/expression/eval?cmd=file.create(%22validate.txt%22)</code>	Exercises the engine pool using the Expression service.
<code>http://servername:port/service_name/api/v8/function/file.create?file=validate.txt</code>	Exercises the engine pool using the Function service.



The `servername:port` corresponds to the Service URL you specified during installation. In some examples, we use the default `service_name`, `SplusServer`; however, you might have named your service something else.

The results of the third and fourth URLs in Table 2.4 should be nearly identical. Examine the resulting XML to find the value for the `ResultDir` element. It should look like the following:

```
<ResultsDir>  
http://servername:port/<service_name>/webdav/results/966CD2713342CD35/  
</ResultsDir>
```

Copy the URL from the `ResultDir` element and paste it in the address box of your browser. This URL should display a page that lists the files in the directory and you should see a 0.0 kb file named `validate.txt`. Spotfire Statistics Services is operating normally if the validation does not produce an error or result in any other unexpected behavior.



If you encounter issues, see [Configure engine logging](#). For additional assistance contact `http://servername:port/service_name` technical support at <http://www.tibco.com/services/support>.

Cluster deployment

Clustering Spotfire Statistics Services can offer benefits for an expanding project or organization. When you consider deploying a cluster, review the network requirements and potential points of failure as well as the benefits.

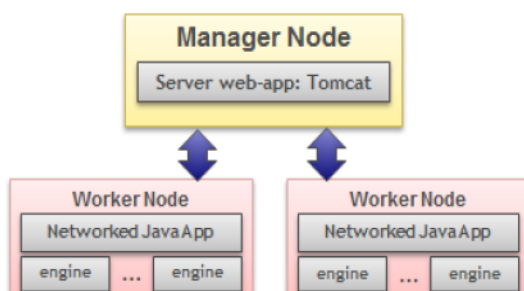
Benefits of Clusters

Benefit	Description
Expands to meet growth.	You are not required to identify a single computer that will meet growth requirements. You can start conservatively and grow the system as needed.
Provides failover.	Jobs running on a failed node are marked as failed. You can resubmit these jobs without needing to bring the failed node back online.
Offers high availability.	You do not need to rely on a single computer. If a Spotfire Statistics Services node fails, other nodes continue processing jobs.
Offers high reliability.	You can identify a growing job queue easily. Adding a Spotfire Statistics Services node is a simple process. Once the service starts, the new node immediately begins processing jobs.
Provides load balancing.	Jobs are distributed among the Spotfire Statistics Services nodes to ensure optimal resource utilization.

Considerations for Clustering

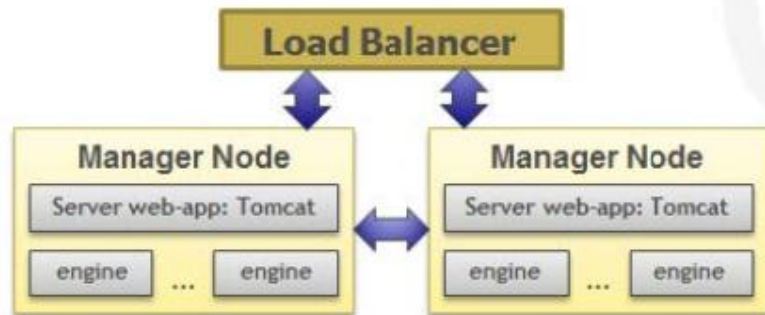
Consideration	Description
Requires a fast network.	The cluster depends heavily on the network used by the systems in the cluster. Additional planning might be necessary to ensure the systems use appropriate interconnects and they are in close proximity.
Includes potential points of failure.	<p>The basic cluster deployment includes the following three potential points of failure.</p> <ul style="list-style-type: none"> • Database server used for the job database. • File system used for the shared cluster configuration store. • Single point for handling incoming requests.

Spotfire Statistics Services single-manager server with workers.



You can eliminate these points of failure, but you might need additional planning and configuration. In the case of the job database and the shared cluster configuration store, most database servers and file systems offer solutions for failover and high availability. For the consideration of single point for handling incoming requests, you might consider using a replicated load balancer with two Manager nodes.

Spotfire Statistics Services two manager servers with load balancer.



Cluster Configuration Requirements

In a Spotfire Statistics Services cluster, each node must share the same configuration store and job database. It is critical that the results generated by a node are accessible to other nodes and clients in the event that node is not available. To achieve this, the configuration store, job database, and the cluster storage must be centralized.

The `spserver.properties` file contains the key configuration properties for Spotfire Statistics Services. See [Server properties](#) for more information.

Cluster scaling design

As you plan a Spotfire Statistics Services cluster deployment, consider whether use the vertical or horizontal scaling design.

The Spotfire Statistics Services cluster is no different, conceptually, than a standalone deployment. The standalone, a single computer, processes all incoming requests. The job database manages the job queue and holds information needed during the job lifecycle. The engines execute the jobs, and the local file system contains all of the required files, along with the configuration and any results generated during job execution.

Vertical Scaling

The simple design of Spotfire Statistics Services allows vertical scaling. That is, you can provide more capacity by upgrading hardware on the standalone computer. For example, adding processors, memory, or storage could increase overall computing power or allow the system to support more users.

In many cases, vertically scaling Spotfire Statistics Services is a reasonable choice and certainly has its benefits. For example, you need to maintain only a single computer. The drawback to this approach is that it usually requires careful initial planning to find a computer that will meet growth requirements. In addition, a single computer generally indicates a single point of failure.

Horizontal Scaling

To address issues of growth requirements and scalability, as well as those of single point of failure, Spotfire Statistics Services supports horizontal scaling. In other words, increasing capacity does not require modifications to a single computer. Rather, it only requires installing the software on additional computers that work together. This is the basis for the Spotfire Statistics Services cluster.

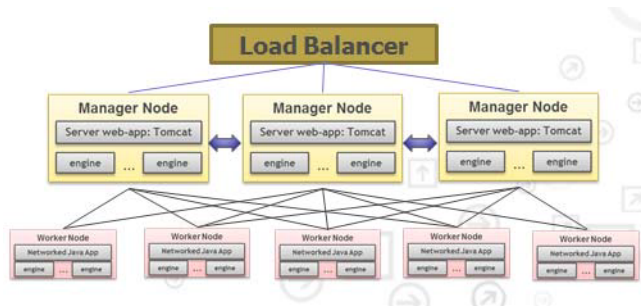
Manager nodes

To implement a Spotfire Statistics Services cluster, you configure Manager nodes and Worker nodes.

A Spotfire Statistics Services Manager node is dedicated to handling all incoming requests. You must have at least one Manager node per cluster, and it must be installed on a separate computer from worker nodes. After you establish the first Manager node, you can add one or more Worker nodes at any time, because the cluster uses a centralized storage location for the configuration store.

A dedicated Spotfire Statistics Services Manager node that handles all incoming requests introduces a single point of failure. We recommend, for failover purposes, that you configure an additional Manager node. However, when you have more than one Manager node in your cluster, you must also install a load balancer. A load balancer can ensure failover and distribute incoming requests between two Manager nodes. The nodes share the responsibilities of queuing jobs and forwarding results back to the load balancer. The load balancer sends the results to clients.

Spotfire Statistics Services ultimate cluster deployment with failover.



Cluster installation checklist

Before you establish the Spotfire Statistics Services cluster, confirm that you meet all of the requirements for a successful deployment.

- Verify that a computer meets the Spotfire Statistics Services minimum requirements to use as the first Manager node in your cluster.
- Verify that the port numbers to specify for the Main Service Port and the JMX Monitoring Port during the Spotfire Statistics Services installation are not being used by other applications on the computer.



Remember that when installing and configuring Spotfire Statistics Services to use a statistical engine, consider the configurations and strategies to restrict access. See [Recommendations for safeguarding your environment](#) on page 13 for detailed information.

- If you use Multicast for messaging, ensure that the combination of the Cluster Multicast Group Address and Cluster Multicast Discovery Port, specified during installation, is unique for each Spotfire Statistics Services cluster in your organization.
- If you use JMS messaging, ensure that the Cluster Messaging Port is not being used by other applications.
- If you plan to use a native Web server as a proxy or load balancer, identify the server name and the port it uses to service requests. You use this information to construct the Service URL during installation.
- Identify an adequate database server for the jobs database. Spotfire Statistics Services includes an H2 database as the default. We recommend at least a gigabit connection and that the database server is within close proximity with Spotfire Statistics Services nodes.



Because the jobs database represents a potential single point of failure, you might consider replicating the database server or deploying a database server cluster.

- Ensure that you have the JDBC driver for connecting to the job database server. For more information, see [Configuring the JDBC driver](#).
- Identify the file server to be used for the centralized configuration store. We recommend a modern file server that has a gigabit connection and close proximity between it and the Spotfire Statistics Services nodes.



Because the file server is a potential single point of failure, you might consider implementing a clustered file server.

Cluster file share

A Spotfire Statistics Services cluster must have a file share established prior to the cluster installation.

On the file server, create and share a directory to use as the centralized configuration store. Each node in the cluster must have both read and write access to this directory.

Make a note of this path because you will need this path when you install the Manager and Worker nodes.

Installing the first manager node on Windows

Establish your Spotfire Statistics Services cluster by installing the first Manager node.

Prerequisites

- To add a Manager or Worker node, you must use the same version of the installer that you used to install the first Manager node in the cluster.
- You must install the Manager node on a separate computer from the Worker nodes.

Procedure

1. Log on with a user name that is a member of the local administrators group.
2. From the Spotfire Statistics Services installation media, use an application appropriate for unpacking the installation .zip package named `TIB_sf_statsvcs_7.11.2_win_x86_64.zip`.
3. Review the accompanying release notes (if you have not done so already) and run the installation executable named `TSSS_7.11.2_win_x86_64.exe`.
4. Follow the procedure [Running the installer](#). When you complete a step in the Spotfire Statistics Services installation wizard, click **Next** to proceed to the next panel.

Installing the first manager node on UNIX or Linux

Establish your Spotfire Statistics Services cluster by installing the first Manager node.

Prerequisites

- To add a Manager or Worker node, you must use the same version of the installer that you used to install the first Manager node in the cluster.
- You must install the Manager node on a separate computer from the Worker nodes.

Procedure

1. Log on as the user under which Spotfire Statistics Services will run. This must be a valid user account. (We do not recommend using root, because doing so can result in problems.)

2. Make sure that the user has permissions to write to the desired installation directory.
3. From the installation media, unpack Spotfire Statistics Services .tar.
The operating system executable for 64-bit Linux is
TIB_sf_statsvcs_7.11.2_linux24gl23_x86_64.tar
4. Review the release notes (if you have not already done so), and then run .bin to install.
5. Follow the procedure [Running the installer](#). During the installation, when you complete a step in the console press **ENTER** to proceed to the next step.

Running the installer

This procedure walks you through running the Spotfire Statistics Services installer as you add the first node in a new cluster.

Procedure

1. **License Agreement:** Read the license agreement and if you agree to the terms, select **I accept the terms of the License Agreement**.
2. **Installation type:** Select **Cluster: Manager Node**.
3. **Installation Path:** Either accept the default, type a path, or choose a path.
4. **Cluster Share:** Type or choose a path that is available to all computers in the cluster. It can be a local or a network path.
5. **Service Name:** Accept the default name (SplusServer) or type a different *service_name*.



You can install multiple instances of Spotfire Statistics Services on a single computer with the following conditions:

- The service name for each service must be unique.
- They cannot share the same home directory or cluster share directory.

The *service_name* is used in several places. It is used as the name of a new directory under the path that you typed in the **Installation Path** panel. For example, if you accept the default service name, in 64-bit Windows, the path is C:\Program Files\TIBCO\statsvcs711\<service_name>.

- On Windows computers, the name of the Windows service is TSSS711<service_name>. The <service_name> is also appended to the service display name; for example, http://servername:port/(SplusServer).
 - On UNIX/Linux computers, this string is used when you register the init script.
6. **Ports** (Panel 1): For the **Main Service Port**, either accept the default or type a port number for the service. For the **JMX Monitoring Port**, either accept the default or type a port number.



The port numbers must be unique for each Spotfire Statistics Services instance on the computer.

7. **Ports** (Panel 2): If you plan to use a front-end proxy or load balancer, select **AJP**; otherwise, select **HTTP**.
8. **Ports** (Panel 3): Select the communication method to use.
 - **Multicast** protocol communicates by sending short packets (datagrams), which can be received by all the nodes listening on a specific Multicast address and port. This protocol uses Java built-in support for object serialization to serialize and deserialize the messages sent using this protocol. This protocol provides a simple messaging bus that allows all nodes to stay in sync and distribute the workload with a very low overhead. However, the Multicast protocol uses UDP to transmit messages, and UDP packets can be dropped by switches and firewalls. To

minimize packet loss, we recommend you use Multicast only if all the nodes of the cluster are physically located on the same subnet. Multicast protocol cannot be used for cloud computing.

- **Messaging (JMS)** is a very reliable messaging protocol, but it requires more resources from both manager and worker nodes to provide support for cluster messaging. You must specify JMS Messaging if your server is running on a cloud. Otherwise, you should choose this protocol only if Multicast is unavailable or unreliable in a given networking context.

Spotfire Statistics Services has its own built-in JMS support and does not require a separate JMS server product. Note that you cannot alter or substitute embedded JMS service that is based on Apache Active MQ.

9. **Ports** (Panel 4):

- If you chose **Multicast**, for **Cluster Multicast Discovery Port**, either accept the default or type a port number. For **Cluster Multicast Group Address**, either accept the default or type an Multicast IP address.



The combination of the Multicast IP address and port must be unique for each Spotfire Statistics Services cluster in your organization.

- If you chose **Messaging (JMS)**, for **Cluster Messaging Port**, either accept the default or type a port number.

10. **Service URL**: Provide the URL for your load balancer. If you are not using a load balancer, accept the default.

11. **Management User**: (Manager nodes only.) Provide the user name and password for authentication when using a JMX tool such as TIBCO Hawk or jconsole (for example). The default user name is admin. These credentials are written to a server configuration file to provide secure job monitoring. The credentials provided for this management node are used cluster-wide.

These credentials are specific to this Spotfire Statistics Services cluster installation, and because they are stored in a server configuration file, we highly recommend not using your login credentials.

12. **Service User**:

- On Windows computers, to configure Spotfire Statistics Services, type the credentials for the service account.
- On UNIX/Linux, the service runs in the context of the user that you are using to run this installer.

For both Windows and UNIX/Linux, the specified account must have read/write access to `SPSERVER_SHARE`, as well as permission to access any other Spotfire Statistics Services resources.

13. **Pre-Installation Summary**: Review your selected options, and then click **Install** or press **ENTER**. The Installing TIBCO Spotfire Statistics Services<service_name> panel appears while the server is installed and configured.

14. **Install Complete**: Click **Done** or press **ENTER**.



If you encounter issues with the installation, you should review the installation log file at `SPSERVER_HOME/Install_SplusServer.log`.

Migrating your Spotfire Statistics Services configuration and data

Upgrading from a previous release of Spotfire Statistics Services requires a new installation. However you can retain and migrate configuration and data.

We recommend shutting down the previous version service and installing the new version in a different location. After you have installed the new version, follow this guidance.

Prerequisites

Install the new version, test that the service works, and then shut down the new version's service before completing these steps.

Procedure

1. Copy any `.jar` files from your endorsed directory to your new installation.
2. From the older installation, open the folder `SPSERVER_HOME/conf` and copy the appropriate configuration and data from the old installation to the new.

For most configuration and data, you can migrate from the previous installation to the new one, with the cautions and exceptions listed in here.



Do not modify the following entries in the file `conf/server.properties`:

- `service.id`
- `service.url`

Do not modify the following properties, unless you need to change them from the values you selected during installation.

- `jmxremote.port`
- `jmxremote.password`
- `jmxremote.username`

If you are upgrading from a version of Spotfire Statistics Services earlier than version 6.0, do not copy the older versions of the files `conf/database.properties` and `conf/ldap.properties`. These files have changed. Specifically, do not copy the entry `database.url` in the file `conf/database.properties` from the older installation. Instead, just change the `SPSERVER_SHARE` location to point to the new installation.

If you are installing a cluster, do not copy the file `conf/cluster.properties`.

3. Locate previous user data in `SPSERVER_HOME/data/common` and `TSSS_HOME/data/users`, and then copy these directories directly from the previous installation to the new installation.



All other directories under `SPSERVER_HOME/data` contain temporary data from the previous installation and should not be copied.

What to do next

Complete any remaining configuration tasks. If no further tasks are needed, start the service.

Post-installation configuration settings

On both Windows and UNIX/Linux, you must make a few configuration adjustments to the Spotfire Statistics Services cluster installation and configuration, and you must configure the job database.

Immediately after you finish your installation of a Spotfire Statistics Services cluster, you should make additional configuration changes, first depending on your operating system, and then to set up the job database.

After reviewing and changing any post-installation settings, your first Manager node configuration is complete. All subsequent Worker nodes use the same configuration. You can further customize the configuration (for example, enable authentication). See [Server properties](#).

Granting service access

On Windows computers, you must grant **Log on as a service** to the Spotfire Statistics Services service account that you specified when you installed the Manager node.

Procedure

1. On the task bar, click **Start >Control Panel > Administrative Tools > Local Security Policy**.
2. In the **Local Security Settings** dialog box, under **Security Settings**, expand **Local Policies**, select **User Rights Assignment**, and then double click **Log on as a service**.
3. In the **Log on as a service Properties** dialog box, click **Add User or Group**, and then enter the service account you specified in step 12 of [Running the installer](#).

Register init scripts

On UNIX/Linux, make additional Spotfire Statistics Services post-installation configuration changes for a successful cluster installation.

To configure the service to run when the computer starts, you must register the `init` scripts with system `init` by running the following command as root:

```
SPSERVER_HOME/init.d/register-service.sh
```

Set the job database

After your successful cluster installation of Spotfire Statistics Services, configure the job database.

By default, the cluster jobs database is configured to use the H2 database type, and the installation prepopulates the JDBC connection string for your environment. See [database.url](#) on page 89 for more information about the default.



If you are using the H2 database implementation in a cluster, you must configure it. If you start the H2 cluster manager node without first configuring the H2 database cluster options, no errors are reported; however, the server will not work correctly.

You can change the default configuration of the jobs database on a database server by setting the properties and configuring the options. See [Job database requirements](#).

Change the engine type in a cluster

The TERR engine is maintained and deployed automatically by the server. The compressed file containing the engine resides in the `SPSERVER_SHARE/data/binaries` directory, where it can be referenced and updated easily.

The compressed engine types by operating system include the following.

Windows	UNIX/Linux
<code>TERREngine.zip</code>	<code>TERREngine.tar.gz</code>
<code>SplusEngine.zip</code>	<code>SplusEngine.tar.gz</code>



You can specify any of the engine types described in the following table. If you are using an engine other than those maintained and deployed by the server, you must install them explicitly on the machines where they are used. You must download and maintain the open-source R engine separately.



Remember that when installing and configuring Spotfire Statistics Services to use a statistical engine, consider the configurations and strategies to restrict access. See [Recommendations for safeguarding your environment](#) on page 13 for detailed information.



Open-source R is available under separate open source software license terms and is not part of TERR. As such, open-source R is not within the scope of your license for TERR. Open-source R is not supported, maintained, or warranted in any way by TIBCO Software Inc. Download and use of open-source R is solely at your own discretion and subject to the free open source license terms applicable to open-source R.

Engine Type	Description
TERR	TERR engine. The default. Provided with your Spotfire Statistics Services installation.
R	Open-source R engine. Use this engine if developers are deploying R packages to the service, and the packages contain R code that will not run in TERR. Requires additional steps. See Configuring an open-source R engine for more information.  In Spotfire Statistics Services version 7.11.2, open-source R is configurable.
S-PLUS	S-PLUS engine. Provided with your Spotfire Statistics Services installation. Use this engine if developers are deploying S-PLUS packages to the service.  This engine is deprecated and will be removed in future releases.
SAS	SAS® engine. Requires additional steps. See Configuring a SAS engine for more information.
MATLAB	MATLAB® engine. Requires additional steps. See Configuring a MATLAB engine for more information.

By default, `engine.type` is set to TERR. To specify a different engine type, set the `engine.type` property in the file `SPSERVER_SHARE/conf/spserver.properties`.



You must specify only one engine type per cluster.

See [Server properties](#) for more information about setting properties.

When the server starts, the engine is unpacked and copied from its location in the `SPSERVER_SHARE/data/binaries` directory, and then deployed to each node in the cluster. If you update or change the `engine.type` property, you must restart the server to initiate the deployment.

Specify engine count for the manager node

The engine count is the number of engines configured on Spotfire Statistics Services to handle engine-related requests.

When you install a Manager node, this value is set to zero. If you want your Manager node to be able to process jobs, you can assign processing resources by setting the `-Engine.count` Java runtime system property. You should always leave at least one processor free for administrative tasks in the Manager node.

When you install a Worker node, the engine count is set to the number of cores in the server.

- To specify the engine count on Windows, using Procrun, set `-Engine.count` to the desired value. For example, `-Engine.count=4`. See [Monitor the service](#) for more information.
- To specify the engine count on UNIX/Linux, in `SPSERVER_SHARE/init.d/spserver`, modify the value of `SPSERVER_ENGINES`.

Start and validate the service

After you have completed the Spotfire Statistics Services installation the post-installation configuration steps, you are ready to start the service.

Start the service on the first Manager node.

- For Windows installations, see [Starting the service on Microsoft Windows](#) for details.
- for UNIX/Linux installations, see [Starting the service on Unix or Linux](#) for details.

Validate the Installation

You should validate the Spotfire Statistics Services installation to make sure that it is working properly. See [Validation](#) on page 30 for more information.



On Windows, if you want to validate your Manager node before you install Worker nodes, you can set the `-Dengine.count` Java property to at least 1 using the Procrun application; although it is possible to get a partial validation without setting the `engine.count`. See [Monitor the service](#).

Add nodes

When you finish installing and verifying the first Spotfire Statistics Services Manager node, the settings and configuration for the cluster are complete. Now, you can add additional Manager and/or Worker nodes.

The same settings are used when you install an additional Manager node or when you add a Worker node to the cluster. To provide failover capability and to minimize downtime we recommend that you use more than one Manager node in your cluster and set up a front-end proxy or load balancer.



If you want to add a Manager or Worker node, you must use the same version of the installer that you used to install the first Manager node in the cluster.

Installing an additional node on Windows

Complete these tasks to add a Manager or Worker node to a Microsoft® Windows installation of a Spotfire Statistics Services cluster.

Procedure

1. Log on with a user name that is a member of the local administrators group.
2. Spotfire Statistics Services requires that you to designate an account for running the service. You must verify that the designated service user account has been granted Log on as a service rights before starting the service. To grant a user Log on as a service rights, follow the steps in [Establishing log on as a service right](#) on page 23.
3. You must use the same installer file that you used in the To install the initial Manager node procedure. From the Spotfire Statistics Services installation media, run the installation executable `TSSS_7.11.2_win_x86_64.exe`.
4. **License Agreement:** Read the license agreement and if you agree to the terms, select **I accept the terms of the License Agreement**.
5. Installation Type: Select the node type.
 - **Cluster: Manager Node**
 - **Cluster: Worker Node**
6. **Cluster Share:** Type or choose the path that you defined for your cluster file share ([Cluster file share](#)). If you are adding a Worker node to an existing cluster, provide the same Cluster Share Path you specified for the Manager node.

7. Service user:

- Windows: To configure the Spotfire Statistics Services (`service_name`), type the credentials for the service account and then click **Next**.
- UNIX/Linux: The service runs in the context of the user that you are using to run this installer.

For both Windows and UNIX/Linux, this account must have read/write access to `SPSERVER_SHARE` as well as permission to access any other Spotfire Statistics Services resources.



All nodes in the cluster must use the same credentials. In Windows, you must use the same service account and in UNIX/Linux, you must run the installer under the same account on each node.

8. **Pre-Installation Summary:** Review your selected options, and then click **Install** or press **ENTER**. The Installing TIBCO Spotfire Statistics Services<`service_name`> panel appears while the server is installed and configured.
9. **Start Service:** Click **Next** or press **ENTER**.
If you selected **Start Service**, Spotfire Statistics Services (`service_name`) is started when installation completes. If you do not want the service to start at this time, clear **Start Service**.



If the Service User has not been granted the Log on as a service right, clear **Start Service** or grant the user the Log on as a service right.

On Windows computers, the Spotfire Statistics Services(`service_name`) service is set to start automatically, which means that the service starts when you reboot your computer.

10. **Install Complete:** Click **Done** or press **ENTER**.



If you encounter issues with the installation, you should review the installation log file at `SPSERVER_HOME/Install_SplusServer.log`.

Installing an additional node on UNIX or Linux

Complete these tasks to add a Manager or Worker node to a UNIX or Linux installation of a Spotfire Statistics Services cluster.

Procedure

1. Log on as the user under which Spotfire Statistics Services will run. You should make sure to use the same user account to install all the nodes in your cluster.
2. Make sure that the user has permissions to write to the desired installation directory.
3. You must use the same installer file that you used in the procedure To install the initial Manager node. From the Spotfire Statistics Services installation media, run the installation executable.
4. **License Agreement:** Read the license agreement and if you agree to the terms, select **I accept the terms of the License Agreement**.
5. Installation Type: Select the node type.
 - **Cluster: Manager Node**
 - **Cluster: Worker Node**
6. **Installation Path:** Either accept the default, type a path, or choose a path.
7. **Cluster Share:** Type or choose the path that you defined for your cluster file share ([Cluster file share](#)). If you are adding a Worker node to an existing cluster, provide the same Cluster Share Path you specified for the Manager node.
8. **Service user:**

- Windows: To configure the Spotfire Statistics Services (*service_name*), type the credentials for the service account and then click **Next**.
- UNIX/Linux: The service runs in the context of the user that you are using to run this installer.

For both Windows and UNIX/Linux, this account must have read/write access to `SPSERVER_SHARE` as well as permission to access any other Spotfire Statistics Services resources.



All nodes in the cluster must use the same credentials. In Windows, you must use the same service account and in UNIX/Linux, you must run the installer under the same account on each node.

9. **Pre-Installation Summary:** Review your selected options, and then click **Install** or press **ENTER**. The Installing TIBCO Spotfire Statistics Services<*service_name*> panel appears while the server is installed and configured.

10. **Start Service:** Click **Next** or press **ENTER**.

If you selected **Start Service**, Spotfire Statistics Services (*service_name*) is started when installation completes. If you do not want the service to start at this time, clear **Start Service**.



If the Service User has not been granted the Log on as a service right, clear **Start Service** or grant the user the Log on as a service right.

11. **Install Complete:** Click **Done** or press **ENTER**.



If you encounter issues with the installation, you should review the installation log file at `SPSERVER_HOME/Install_SplusServer.log`.

12. To configure the service to run when the computer starts, you must register the init scripts with system init by running the following command as root:
`SPSERVER_HOME/init.d/register-service.sh`

Validate nodes

You can use the Spotfire Statistics Services URL API to verify the nodes in the cluster.

Open a browser, and in the address text box, type:

`http://servername:port/service_name/api/v8/nodes`

It takes a few minutes after a node is started for it to appear in the XML returned from this URL API call.

The service URL that you use for validation is `http://servername:port/<service_name>`, where *servername* depends on whether you set up a load balancer.

- If you did set up a load balancer, you must use the *servername* and port from the load balancer.
- If you did not install a load balancer, you can use the *servername* and port of the first manager node.

Load balancer

Use a load balancer with your Spotfire Statistics Services cluster deployment to distribute job requests equally, or if you have multiple Manager nodes.

To distribute job requests equally to all the Manager nodes, use a load balancer. A load balancer can be configured with any Spotfire Statistics Services cluster but if your cluster has more than one Manager node, you must install and configure a load balancer. The load balancer that you select for your environment depends on how much traffic you expect and the sophistication that you require in the load balancing algorithms.

You can use a load balancing solution based on Apache HTTP Server, which supports AJP (Apache JServ Protocol), with the `mod_jk` module enabled.

Prerequisites

To set up load balancing using Apache HTTP Server, you will need the following.

- A computer with Apache HTTP Server and the mod_jk module installed. You can download these from the following locations.
 - Apache HTTP Server: <http://httpd.apache.org>.
 - mod_jk module: <http://tomcat.apache.org/download-connectors.cgi>.
- At least one Spotfire Statistics Services Manager node must be installed and configured.

Cluster Node Protocol Configuration

During the installation of the first Manager node in your Spotfire Statistics Services cluster, while planning for a load balancer, you would have selected the **AJP** protocol so that the cluster can communicate with a load balancer. No other changes are required on the Spotfire Statistics Services nodes. (After you install the Manager node, you can modify settings by editing `SPSERVER_SHARE/tomcat/conf/server.xml`. For more information, see the Apache Tomcat documentation.)

If, during installation, you selected **HTTP** and now you want to configure a load balancer, you must modify all of the Manager nodes to use AJP. For more information, see [Changing the protocol or port](#).

Configuring the load balancer

You must configure the load balancer so that it can find and communicate with the Spotfire Statistics Services cluster.

Procedure

1. Install an Apache HTTP Server.
2. Install the mod_jk module. For more detailed instructions on adding modules to Apache HTTP Server, see the Apache HTTP Server documentation.
3. Edit the `workers.properties` file to add the following properties and values. This file is usually located in the Apache HTTP Server configuration directory. If this file is not present, you must create it and then add the settings.

```
# Define worker list
# (All workers with additional exposed applications
# must also be added here, and do not forget to add
# the corresponding JkMount option in mod_jk.conf!)
worker.list=jkstatus, loadbalancer
# Example: the /admin application on worker1 should
# be exposed through the load balancer
# worker.list=jkstatus, loadbalancer, worker1

# Set status
worker.jkstatus.type=status

# Set properties for the load balancer
worker.loadbalancer.type=lb
worker.loadbalancer.balance_workers=worker1, worker2

# Set properties for worker1 (ajp13)
worker.worker1.type=ajp13
worker.worker1.host=[Managernode1_name]
worker.worker1.port=[Managernode1_port]
worker.worker1.max_packet_size=65536
worker.worker1.lbfactor=1

# Set properties for worker2 (ajp13)
worker.worker2.type=ajp13
worker.worker2.host=[Managernode2_name]
```

```
worker.worker2.port=[Managernode2_port]
worker.worker2.max_packet_size=65536
worker.worker2.lbfactor=1
```

You must change [Managernode1_name] to the hostname or IP address of the first Manager node, and [Managernode2_name] to the name of your second Manager node. In every instance of the worker.worker2.port property, you must specify the port number that you set as the **Main Service Port** when you installed the first Manager node. If you want to add an additional Manager node, copy the Set properties for workern (ajp13) section and modify the values as needed.

4. Add the following lines to the mod_jk.conf file. If the file is not present, you must create the file and then add the settings.

```
# Load the mod_jk module
LoadModule jk_module modules/mod_jk.so

# Load the workers configuration
JkWorkersFile conf/workers.properties

# The mod_jk module's log file
JkLogFile logs/mod_jk.log

# The mod_jk module's log level
# (trace, debug, info, warn, error)
JkLogLevel info

# Let the load balancer worker handle all requests
# to the TSS web applications
JkMount /<service_name> loadbalancer
JkMount /<service_name>/* loadbalancer
# replace <service_name> with the name of
# your server instance.

# Define Apache environment variables to be
# exported by mod_jk to Tomcat web applications
JkEnvVar REMOTE_USER
JkEnvVar SSL_CLIENT_CERT
#JkEnvVar SSL_CLIENT_CERT_CHAIN
#JkEnvVar SSL_CLIENT_S_DN
#JkEnvVar SSL_CLIENT_S_DN_CN
```

5. The Apache HTTP Server configuration must include mod_jk.conf. For example, if mod_jk.conf is in the same directory as the Apache HTTP Server configuration, httpd.conf, add Include conf/mod_jk.conf.
6. Restart the Apache HTTP Server. Check for any errors that were generated when the system started.
7. Open a web browser and, in the address bar, send a request for the nodes in the cluster by using `http://servername:port/<service_name>/api/v8/nodes`, where `servername:port` corresponds to the load balancer name and port. Scan the XML for `nodeType="Manager"` and make sure that all of your Manager nodes are in the list.

Restricting the load balancer

For increased security, you can configure the load balancer to authenticate when it communicates with Spotfire Statistics Services.

You can restrict access to the load balancer by setting up an AJP Connector secret keyword for the load balancers to use to authenticate with the Manager nodes. This is a secret keyword that the load balancers and Manager nodes all know.

Procedure

1. Add the keyword to all the Manager nodes.
2. In the SPSEVER_SHARE/tomcat/conf/server.xml file, find the section specifying `Service name = "Catalina"`, and in that section, find the Connector configuration:

```
<Connector port="8009" protocol="AJP/1.3" packetSize="65536"/>
```

3. Add the keyword definition

```
<Connector port="8009"
protocol="AJP/1.3"
packetSize="65536"
request.useSecret="true"
request.secret="SecretKeyword" />
```

4. Add the keyword to the `worker.properties` file on the load balancer computer.
5. Above the properties for individual workers, add a keyword that all the nodes will use:

```
# Enable secret keyword
worker.loadbalancer.secret="SecretKeyword"
```

Now Spotfire Statistics Services accepts only AJP connections from load balancers that know the secret keyword.

Automate cluster installations

If you want to install Spotfire Statistics Services on several computers, you can perform a silent Spotfire Statistics Services installation by providing a response file to the installer. You can use this approach to automate Spotfire Statistics Services installations.

There are two methods for creating a response file for automated cluster installations.

- Create a valid response file by automatically recording the responses that you select or type when you run the installation wizard, and then use it in subsequent installations.
- To create a response file based on the Response File template. We recommend this method.

Recording and running a response file

To create a valid Spotfire Statistics Services response file manually, you can set a switch to automatically record the responses that you select or type when you run the installation wizard. Then you can use another switch to perform a silent installation.

Procedure

1. To capture your responses in a response file, execute the Spotfire Statistics Services installation file name at the command line with the `-r` switch followed by a full path to the destination of the response file. For example, to create a response file for the Windows 64-bit installer you would type the following command:
`TSSS_7.11.2_win_x86_64.exe -r c:\foo.txt`
2. To perform a silent installation, execute the Spotfire Statistics Services installation file with the `-f` switch followed by the path to a valid response file. For example, to execute a silent installation on a Windows 64-bit computer, you would type the following command:
`TSSS_7.11.2_win_x86_64.exe -f c:\foo.txt`
For more information, refer to the *FLEXERA SOFTWARE InstallAnywhere Users Guide* (<http://www.flexerasoftware.com/>).

Response file template

You can create a valid Spotfire Statistics Services response file for automating cluster installations by basing it on the response file template.

The following printout is a template of a valid Spotfire Statistics Services response file.

```
#####
#
# Install_SplusServer.properties
#
# TIBCO Spotfire Statistics Services
# Automated installation properties file
#
# To use this file with Install_SplusServer,
```

```

# issue the following command:
#
# Install_SplusServer -f Install_SplusServer.properties
#
# Provide a full path to the installer and the file.
#####

#INSTALLER_UI: Set to silent for automated install
INSTALLER_UI=silent

#CHOSEN_INSTALL_SET: Install set options are:
#Solo, Manager, or Worker.
CHOSEN_INSTALL_SET=Solo

#USER_INSTALL_DIR: Enter the installation path
USER_INSTALL_DIR=/opt/TIBCO/statsvcs711

#SERVICE_NAME: The name of the service instance you
#are installing. The installation is placed into a #subdirectory of
USER_INSTALL_DIR with this name.
SERVICE_NAME=<service_name>

#SERVICE_URL: Specify the main URL used to access
#this server.
#Note: This may differ from the default when requests
#come through a load balancer such as Apache HTTP Server
#or Microsoft IIS.
SERVICE_URL=http://servername:port/service_name

#SERVICE_USER: Specify the user account that you want to
#run the service under.
#On Windows, this should be in the format DOMAIN\USERNAME.
#To account for the format of Java properties files you
#must escape the backslashes.
#If blank, the default account is used.
# On Windows this is LocalSystem
# On UNIX this is the user account that is running
# the installer).
#SERVICE_PASSWORD: On Windows, enter the SERVICE_USER
#password.
SERVICE_USER=
SERVICE_PASSWORD=

#JMX_USER: Specify the username of the management user
#for JMX.
#JMX_PASSWORD: Specify the password of the management
#user for JMX.
JMX_USER=admin
JMX_PASSWORD=

#START_*_SERVICE: To start Spotfire Statistics Services
#after installation, specify 'true', 'false' not to start
#the service.
START_TOMCAT_SERVICE=true

#CLUSTER_METHOD: Specify the method that cluster nodes
#should use to communicate among themselves. This should
#be either 'multicast' or 'jms'.
#Only consulted if the "Cluster" feature is used.
CLUSTER_METHOD=multicast

#PORT_*: Specify the ports required for Spotfire Statistics
#Services.
#Note: If you are installing a Manager or Worker node,
#the values you set here are overridden by values that
#you specify in the shared cluster storage directory in
#the CLUSTER_DIR parameter.
PORT_SPSERVER=8080
PORT_JMX=9004
PORT_MULTICAST=5000
PORT_MULTICAST_GROUPADDR=224.0.0.224

```

PORT_JMS=61616

#TOMCAT_CONNECTOR: Specify either HTTP/1.1 or AJP to
 #define the Tomcat connection protocol to place into
 #the server.xml in the embedded Tomcat server.
 #The value that you specify is placed directly
 #into the protocol attribute of the Connector
 #element of USER_INSTALL_DIR/tomcat/conf/server.xml.
 TOMCAT_CONNECTOR=HTTP/1.1

#CLUSTER_DIR: Specify the path to a directory that
 #contains the shared cluster storage directory for
 #this cluster. This setting is only consulted if you
 #specify Manager or Worker in CHOSEN_INSTALL_SET.
 #CLUSTER_DIR=/path/to/cluster/share

The Response File specifies properties and values that provide the details for the installation.

Service configuration and properties

After installing Spotfire Statistics Services, you can configure it for your particular deployment needs.

In planning your configuration, you should familiarize yourself with the installation folders and files, set the properties for your installation needs, and identify the service and engine log files. You can also configure the service for LDAP authentication, or change the analytic engine.

Simple no configuration installation

A simple standalone Spotfire Statistics Services installation requires no additional configuration.

If you install a standalone Spotfire Statistics Services on a computer that meets the minimum requirements, you require no post-installation configuration. (See <http://support.spotfire.com/sr.asp> for up-to-date information about system requirements.)

You can customize the configuration after installing Spotfire Statistics Services and check that it is operating normally. To verify that your installation is running normally, see [Validation](#) on page 30.

Review the service property information to determine which settings to customize. For example, by default, the configuration file requires no authentication. If you want to enable authentication, change the default value to `true`.



By default, if your engine type is TERR, then TERR is set to use restricted execution mode. To change the properties that control this option, you must edit the `spserver.properties` configuration file. See [Server properties](#) for more information.

TIBCO Spotfire Statistics Services data files

You might be called on to manage data files associated with Spotfire Statistics Services.

These data files are located in `SPSERVER_HOME/data/` (in the case of a stand-alone) or in `SPSERVER_SHARE/data/` (in the case of a cluster).

Directory	Description
<code>appdata</code>	For internal use only.
<code>artifacts</code>	For internal use only.
<code>binaries</code>	Contains the compressed directories for the engines available to use with Spotfire Statistics Services. Used only for updating TERR engines, or for adding or updating an S-PLUS engine or an open -source R engine.
<code>common</code>	Contains the common persistent data that is shared with all users, as specified by calling the function <code>spserver.file(user.name="")</code> . For more information, see the <i>TIBCO Spotfire® Statistics Services User's Guide, Managing Transient and Persistent Data</i> .
<code>packages</code>	For internal use. Contains any uploaded language packages. Created only when a package is uploaded
<code>requests</code>	For internal use. Contains data objects sent via a request to the server. Created only when a request containing a data object is submitted to the server. (Simple validation tests do not cause this directory to be created.)
<code>results</code>	For internal use. Contains job results. Specified in the function <code>spserver.results.file()</code> . This directory is managed by the server and automatically cleaned up.
<code>sessions</code>	For internal use. Contains temporary session data. This directory is managed by the server and automatically cleaned up.

Directory	Description
spserver	For internal use. Contains the functions used by the server engine processes.
users	Contains user-specific persistent data. Specified by spserver.file(user.name="<user>").

Configuration and log files

The conf and Spotfire Statistics Services directories of your Spotfire Statistics Services installation contain configuration and log files you can use manage and review the your Spotfire Statistics Services installation. In most cases, you should not have to change property settings.

Configuration and Log Directories

Directory and File	Description
conf/database.properties	Contains database properties.
conf/ldap.properties	Contains LDAP properties.
conf/log4j.properties	For internal use only.
conf/spserver.properties	Contains the server properties
conf/users.properties	For internal use only.
tomcat/conf/catalina.policy	For internal use only.
tomcat/conf/catalina.properties	For internal use only.
tomcat/conf/context.xml	For internal use only.
tomcat/logging.properties	For internal use only.
tomcat/conf/server.xml	Contains the setting for Tomcat's HTTP port. You can change the port number in this file after installation, if necessary.
tomcat/conf/tomcat-users.xml	For internal use only.
tomcat/conf/web.xml	For internal use only.
tomcat/logs/SpplusServer.log	The main server log. Useful for troubleshooting.
tomcat/bin	Contains the Procrun executables.

Configure service logging

The Spotfire Statistics Services service generates logs containing information you can use to troubleshoot. You can set logging to different levels.

Spotfire Statistics Services uses log4j (<http://logging.apache.org/log4j>) as the primary logging mechanism for the service. The log files roll over, but by default only three are kept. (This default is configurable).

Other log files in SPSEVER_HOME/tomcat/logs occasionally generate Spotfire Statistics Services-related information but, in general, most information is written to SpplusServer.log

The log levels provided by Spotfire Statistics Services are the standard log4j levels. In the case of Spotfire Statistics Services, these levels can be interpreted as follows:

Log Threshold	Description
DEBUG	Messages for troubleshooting/debugging purposes.
INFO	Messages indicating the server is healthy and active.
WARN	Messages indicating a potential issue.
ERROR	Messages indicating an issue has occurred.
FATAL	Messages indicating a catastrophic event has occurred.

Log Levels

The root logger is set to DEBUG by default. The default appender for the `SpplusServer.log` file is labeled ROLLFILE. It has a default value of INFO for the logging threshold. To change this to DEBUG, for example, update the threshold to DEBUG:

```
log4j.appender.ROLLFILE.Threshold=DEBUG
```

For your convenience, the `log4j.properties` file includes brief documentation for the ROLLFILE appender and other commented-out appenders.

If you need assistance setting up one of these appenders, contact Spotfire support.

For information on engine logging, see [configure engine logging](#).

On-the-fly logging

You can modify the properties for logging, and Spotfire Statistics Services applies the changes without having to be restarted.

You can modify all properties in the `log4j.properties` file and Spotfire Statistics Services applies the changes on the fly. The server checks the file for changes every 10 seconds. If the file has changed, the server automatically begins using the updated configuration. The properties file is located at `SPSERVER_HOME/conf` for a standalone installation or `SPSERVER_SHARE/conf` on a cluster.

Configure engine logging

The Spotfire Statistics Services engine generates a log containing information for troubleshooting. Each job receives its own log file with information pertaining to that job.

Engine Log File

You can use the information in the `engine.log` file to troubleshoot issues related to the engine. The file contains information about the engine process life cycle, such as loading of libraries, establishing communication with the server, and details of the job execution inside the engine, which include all the messages sent and received by the engine process, the actual code executed, arguments passed, and other details.

The `engine.log` is located in the results directory of each specific job. To review the log file for a job, find the link in the `ResultsDir` property for that job. For example:

```
<ResultsDir>
http://localhost:8966/<service_name>/webdav/results/9C347CC9BF333EEF/
</ResultsDir>
```

While you cannot change the logging level provided by the default `engine.log`, you can create a log4j engine logging file with configurable settings.

Creating a log4j engine logging file

The default TIBCO Spotfire® Statistics Services engine log file can be configured to provide a logging level different from the default level (which is `DEBUG`). If the default level of logging displayed in `engine.log` is not the level you need, you can create a file to use for configuring engine logging.

Procedure

1. Create a file named `engine-log4j.properties`.
2. Provide in the file a basic configuration, such as the following:

```
#Loggers.  
log4j.logger.com.insightful.splussserver.engine=DEBUG
```

The engine's log configuration file does not include appender information (unlike the `log4j.properties` file). It need only include a directive for package-level logging. To disable most logging, set the configuration value as follows:

```
#Loggers.  
log4j.logger.com.insightful.splussserver.engine=INFO
```



The `INFO` log option results in only exceptional cases being logged. In most cases, such a setting produces a zero-length `engine.log` file for each job.

3. Save the file in the folder `SPSERVER_HOME/conf` (for a standalone installation) or `SPSERVER_SHARE/conf` (for a cluster installation).

Result

You can use this file in the future to set the logging level to one of the four available levels:

Log Threshold	Description
DEBUG	Messages for troubleshooting/debugging purposes.
INFO	Messages indicating the server is healthy and active.
WARN	Messages indicating a potential issue.
ERROR	Messages indicating an issue has occurred.
FATAL	Messages indicating a catastrophic event has occurred.



If you change the logging level in the `engine-log4j.properties` file, you must restart the service to force the change to take effect.

Server properties

The server configuration properties for Spotfire Statistics Services are contained in the file `spserver.properties`. We use the standard Java properties file conventions to define each server property.



Database properties are separate from the server configuration, therefore they are stored in the file `database.properties`. For more information about the startup time property file `database.properties`, see [Database properties](#).

- In a cluster installation, the file `spserver.properties` is stored in the `conf` directory under `SPSERVER_SHARE`. When you install a new node in a cluster, provide this location so that all the nodes in the cluster have the same values.
- In a standalone installation, the file is stored in the `conf` directory under `SPSERVER_HOME`.

In a default installation, the full path to the `spserver.properties` file is:

Windows	Spotfire Statistics Services: C:\Program Files\TIBCO\statsvcs711\<service_name>\conf
UNIX/Linux	/opt/TIBCO/statsvcs711/<service_name>/conf



Not every server property documented in the reference is included in the default version of `spserver.properties`. To include a property listed below in your configuration, manually insert the property into your `spserver.properties` file.



You must restart the Spotfire Statistics Services service to pick up any changes that you make in the `spserver.properties` file. When you restart the Spotfire Statistics Services service, the server resumes processing queued jobs. However, because restarting the Spotfire Statistics Services service also stops the pool of engines, any job that has not completed is marked as failed. To prevent this, we recommend that you wait until all jobs have completed before you restart the service. Alternatively, you can resubmit failed jobs after restarting the service. To monitor job status, use the `jobs` function. For more information about the `jobs` function, see the URL API documentation, available from the Spotfire Statistics Services landing page.

authentication.required

Set `authentication.required` to `true` to require remote clients to authenticate when they access Spotfire Statistics Services.

Property Name	Default Value	Property Description
<code>authentication.required</code>	<code>false</code>	This option requires additional planning and setup if set to <code>true</code> . The default is <code>false</code> . See Determine authentication setting for more information.

cluster.jms.port

Set `cluster.jms.port` to the port number to use for cluster communication.

Property Name	Default Value	Property Description
<code>cluster.jms.port</code>	3308	Use this setting in cases where you cannot use Multicast IP addressing. You must specify a JMS port number to use for cluster communication if you use cloud computing to implement your Spotfire Statistics Services cluster.

Details

To use the `cluster.jms.port` property you must:

- Enable clustering by setting `isclustered=true`.
- Make sure that the `cluster.multicast.group.address` property does not contain a value. If it does contain a value, cluster communication will use Multicast IP instead of JMS.

cluster.multicast.group.address

Set `cluster.multicast.group.address` to specify the Multicast IP group address that Spotfire Statistics Services should use for cluster communication.

Property Name	Default Value	Property Description
<code>cluster.multicast.group.address</code>	224.0.0.224	This property specifies the Multicast IP group address to use for cluster communication. To

Property Name	Default Value	Property Description
		use this property you must enable clustering by setting <code>isclustered=true</code> .

Details

The combination of `cluster.multicast.group.address` and `cluster.multicast.port` must be unique for each Spotfire Statistics Services cluster on your local area network. To ensure uniqueness, we recommend that you specify a unique port number.

cluster.multicast.port

Set `cluster.multicast.port` to specify the Multicast port number to use for cluster communication.

Property Name	Default Value	Property Description
<code>cluster.multicast.port</code>	5000	To use this property you must enable clustering by setting <code>isclustered=true</code> .

Details

The combination of `cluster.multicast.group.address` and `cluster.multicast.port` must be unique for each Spotfire Statistics Services cluster on your local area network. To ensure uniqueness, we recommend that you specify a unique port number.

db.cleanup.run.interval.minutes

Set `db.cleanup.run.interval.minutes` to the interval for monitoring and cleaning up the Spotfire Statistics Services job queue.

Property Name	Default Value	Property Description
<code>db.cleanup.run.interval.minutes</code>	0	This value, set to minutes, specifies how much time should elapse before the job queue is cleaned up. This property works in combination with <code>db.cleanup.time.to.live.days</code> . The default value of 0 specifies that the Spotfire Statistics Services job queue database should grow indefinitely.

Details

Alternatively, you can manage the job queue using the `delete` function included with the URL API. (See the Help for the URL API, available from the server landing page.)

This property might be necessary to automate the process. For example, this property might be useful in a Spotfire Statistics Services cluster or if your group generates thousands of requests. The optimal value usually depends on the rate of incoming requests and the amount of time the server has been active.

db.cleanup.time.to.live.days

Set the number of days to elapse before a job in the Spotfire Statistics Services job database is considered old enough to delete when the scheduled cleanup runs.

Property Name	Default Value	Property Description
db.cleanup.time.to.live.days	7	Set the number of days to elapse before a job in the Spotfire Statistics Services job database is considered old enough to delete when the scheduled cleanup runs. The optimal value usually depends on the rate of incoming requests and the amount of time the server has been active. By default, this is set to 7 days.

email.from

Set the email.from property to the email address to use in the **from** field for Spotfire Statistics Services job notifications.

Property Name	Default Value	Property Description
email.from	notification@service_name.com	The value you set for this property populates the from field for job notifications.

email.host

Set the email.host property to the host to use for sending notification emails.

Property Type	Default Value	Property Description
email.host	na	This property specializes the email host name to use for sending notification emails from Spotfire Statistics Services. The default is blank.

email.reply.to

Set the email.reply.to property to the email address to use in the **reply to** field for Spotfire Statistics Services job notifications.

Property Name	Default Value	Property Description
email.reply.to	notification@service_name.com	Set this property to the email address for replies to job notifications. This value populates the reply to field.

engine.appdata.dir

Set the engine.appdata.directory to identify the directory used for storing TERR, S-PLUS, or open-source R packages installed to Spotfire Statistics Services via pkgutils.

Property Name	Default Value	Property Description
engine.appdata.dir	na	The engine.appdata.dir specifies the directory where TERR, S-PLUS or open-source R packages are installed using the pkgutils utility function. For more information about adding packages to the server, see Package availability on the server . The default is blank.

Details

The `S_USER_APPDATA_DIR` environment variable is set to this value when the processing runtime creates the engine process. If this property is not provided, or if it is an empty string, Spotfire Statistics Services uses the default directory `SPSERVER_HOME/data/appdata`. If the installation is a cluster, or if you configured the `SPSERVER_SHARE/spserver.share` property, then the directory is set to `SPSERVER_SHARE/data/appdata`.

engine.init.expr

Provide to `engine.init.expr` any open-source R, S-PLUS, SAS, or MATLAB code to run when the engine first starts in Spotfire Statistics Services.

Property Name	Default Value	Property Description
<code>engine.init.expr</code>	na	The default for <code>engine.init.expr</code> is blank. The value for <code>engine.init.expr</code> can be open-source R, TERR, S-PLUS, SAS, or MATLAB code. This value can be used to provide initialization code for the engine the server uses. For example, it can specify language commands to be executed when an engine is started.

Details

Separate multiple commands using semicolons. Any special XML characters (like quotation marks) must be changed to the appropriate XML character sequence (like `"`;) if they are specified in this file.

Commands in `engine.init.expr` are always issued before any commands in the file defined by `engine.init.file`.

engine.init.file

Set `engine.init.file` to the full path to a file containing initialization commands.

Property Type	Default Value	Property Description
<code>engine.init.file</code>	na	By default, this property is blank. If you provide this file, it should be the full path to a file containing the initialization commands. The commands in the file specified by <code>engine.init.file</code> are either open-source R, TERR, S-PLUS, SAS, or MATLAB scripts, depending on the engine type specified for the server. The file specified by <code>engine.init.file</code> is processed after any expressions defined by <code>engine.init.expr</code> .

engine.java.options

Set `engine.java.options` to specify the Java runtime options used when the engine process is created.

Property Name	Default Value	Property Description
<code>engine.java.options</code>	na	<code>engine.java.options</code> specify the Java runtime options used when the engine process is created.

Details

For example, setting this property to `"-Xmx600m -Xss2000k"` sets the Java maximum memory heap size to 600MB and the maximum thread stack size to 2000KB (these limits are normally 400MB and

1280KB). There are other Java runtime options to set the initial Java heap size or raise the limit on the Java stack size. These are documented in Sun's documentation on the Java application launcher.

engine.max.elapsed.seconds

Set `engine.max.elapsed.seconds` to the maximum number of seconds since the engine was started before it is restarted.

Property Name	Default Value	Property Description
<code>engine.max.elapsed.seconds</code>	0	Use this property to set the maximum number of seconds since the engine was last started to when it should be restarted. The default of 0 indicates no limit.

Details

Use the default value for this property, unless you understand the stateful nature of the TERR engine, the S-PLUS engine, or the open-source R engine, and you accept the possible problems that can occur when reusing an engine for multiple requests.

engine.max.requests

Set the maximum number of requests the engine processes before it is restarted.

Property Name	Default Value	Property Description
<code>engine.max.requests</code>	1	Use this property to set the maximum number of requests the engine processes before it should be restarted. The default of 1 indicates that the engine restarts after each job.

Details

Use the default value for this property, unless you understand the stateful nature of the TERR engine, the S-PLUS engine, or the open-source R engine, and you accept the possible problems that can occur when reusing an engine for multiple requests.

engine.startup.timeout.seconds

Set the maximum time to elapse before Spotfire Statistics Services indicates that the engine startup has failed.

Property Name	Default Value	Property Description
<code>engine.startup.timeout.seconds</code>	180 or 480	Use this property to set the maximum number of seconds that Spotfire Statistics Services waits before indicating that the engine has failed to start.

Details

The default of 180 seconds (3 minutes) applies to TERR, open-source R, or SAS engine types; 480 (8 minutes) applies to the MATLAB engine.

engine.type

Set the property `engine.type` to the statistical engine you want to use.

Property Name	Default Value	Property Description
<code>engine.type</code>	TERR	Set this property to the statistical engine you want to use.

Details

The engine type can be TERR (the TERR engine), open-source R, S-PLUS, SAS, or MATLAB. If you want to use an open-source R engine, an S-PLUS engine, a SAS engine, or a MATLAB engine, you must complete additional steps.

- For more information on open-source R, see [Configuring an open-source R engine](#).
- For more information on S-PLUS, see [Configuring an S-PLUS engine](#).
- For a SAS engine see [Configuring a SAS engine](#)
- For MATLAB see [Configuring a MATLAB engine](#)

The engines are maintained and deployed automatically by the server. If you update or change the `engine.type` property, you must restart the server to initiate the deployment.

engine.verbose.log

Set the `engine.verbose.log` to `true` if you want to produce verbose log files.

Property Name	Default Value	Property Description
<code>engine.verbose.log</code>	false	This property determines whether the engine produces verbose logs. By default, it does not.


Details

If the value of this property is `true`, the engines produce verbose log files. In addition, the `engine.max.requests` property is set to 1 automatically, so that each engine is shut down after each request. (This property is applicable only for Spotfire Statistics Services engines.)

expression.service.enabled

Set `expression.service.enabled` to `true` to be able to use the Expression Service API with Spotfire Statistics Services. Applies to any of the tested statistical engine types in your Spotfire Statistics Services installation.

Property Name	Default Value	Property Description
<code>expression.service.enabled</code>	false	<p>The Expression Service is used to send any valid expression to the engine. It can be useful for checking the health of the service, or performing other administrative tasks.</p> <p>This option is set to <code>false</code> by default to remove the ability to run unrestricted expressions. If this option is set to any value other than <code>true</code> (or <code>TRUE</code>), then it is read as <code>false</code>.</p>

Property Name	Default Value	Property Description
		<p>When Spotfire executes a data function, it also calls <code>ExtendedServerInfo</code>, which sends expressions to the engine to get the engine version information. If the Expression Service is disabled, then this generates the following error and traceback in the DEBUG log.</p> <pre>ExtendedServerInfoResource Unable to retrieve runtime engine info</pre> <p>To prevent this from filling up the logs with useless tracebacks, this error is logged only once. To remedy this error, set <code>expression.service.enabled=true</code>.</p> <p>See Recommendations for safeguarding your environment on page 13 for more information.</p>

function.service.whitelist

The property `function.service.whitelist` provides the means to specify the functions that can run on Spotfire Statistics Services. Applies to any of the tested statistical engine types in your Spotfire Statistics Services installation when Spotfire calls a data function using `SpotfireConnector/` `execSpotfireAnalytic`.

Property Name	Default Value	Property Description
<code>function.service.whitelist</code>	<code>SpotfireConnector/</code> <code>execSpotfireAnalytic</code>	<p>This option provides the means to limit which functions can be called using the Function Service API to run on Spotfire Statistics Services.</p> <ul style="list-style-type: none"> If <code>function.service.whitelist</code> is not given, or if it is empty, then it is the same as the default, accepting only <code>SpotfireConnector</code> and <code>execSpotfireAnalytic</code>, so Spotfire data functions can run. If <code>function.service.whitelist</code> is none, then no functions are permitted. If <code>function.service.whitelist</code> is set to <code>*</code>, then all functions are permitted. <p>The default value is a function used only for data functions that are sent to Spotfire Statistics Services from Spotfire.</p> <p>You can set the value to a string with one or more <i>package/function</i> patterns, separated by commas. Each pattern is a Java-style regex string. It can use <code>*</code> to match anything.</p>

Property Name	Default Value	Property Description
		<p>Example</p> <p>The whitelist <code>rnorm,stats/rn.*,*/aa</code> allows the following.</p> <ul style="list-style-type: none"> • The function <code>rnorm</code> in any package. • Any function starting with <code>rn</code>, followed by any characters, in the <code>stats</code> package. • The function <code>aa</code> in any package. <p>If you are using the TERR engine, then any function that is accepted by this whitelist is also evaluated by the property <code>terr.restricted.execution.mode</code> if it is set to <code>true</code> (the default). If a function is determined to engage in potentially malicious behavior, then the function fails when it tries to perform a restricted operation, unless you change <code>terr.restricted.execution.mode</code> to <code>false</code>. (Applies only if you are using the TERR engine.)</p> <p>For more information, and a non-exhaustive list of the types of behavior not permitted when <code>terr.restricted.execution.mode=true</code>, see its help.</p> <p>See Recommendations for safeguarding your environment on page 13 for more information.</p>

isclustered

Set `isclustered` to `true` to enable the server as a node in a Spotfire Statistics Services cluster.

Property Name	Default Value	Property Description
<code>isclustered</code>	<code>false</code>	This property is <code>false</code> for a standalone server installation, and <code>true</code> for a node in a cluster.

Details

If you set this value to `true`, you must conduct additional planning and perform additional steps. You must also configure either Multicast IP addressing or JMS settings for node-to-node communication.

- To configure Multicast IP addressing you must specify values for the `cluster.multicast.group.address` and `cluster.multicast.port` properties.
- To configure for cloud computing, you must specify a value in the `cluster.jms.port` property, and you must make sure that the `cluster.multicast.group.address` property contains no value.

For more information, see [Cluster installation](#) and [Configuring a non-default database](#).

jmxremote.enabled

Set `jmxremote.enabled` to `true` to enable remote server monitoring and management using JMX.

Property Name	Default Value	Property Description
<code>jmxremote.enabled</code>	<code>true</code>	If you are using JMX for remote server monitoring and management, accept the default value of <code>true</code> .

jmxremote.password

Set during installation, the value for `jmxremote.password` identifies the password to use with the username to access JMX remote monitoring and management..

Property Name	Default Value	Property Description
<code>jmxremote.password</code>	(set during installation)	This property identifies the password to use with the username for JMX remote monitoring.

jmxremote.port

Set `jmxremote.port` to indicate which port JMX should use for remote monitoring and management.

Property Name	Default Value	Property Description
<code>jmxremote.port</code>	9004	The value of <code>jmxremote.port</code> is set to 9004 by default during installation, and specifies the port that JMX uses for remote monitoring and management. If this port is used, change the value to an available port.

job.list.xslt.enabled

Set `job.list.xslt.enabled` to `true` if you are using a style sheet to display the results of calls to the Spotfire Statistics Services Administration Service function `jobs`.

Property Name	Default Value	Property Description
<code>job.list.xslt.enabled</code>	<code>true</code>	If you use a style sheet to display the results of calls to the Administration Service function <code>jobs</code> , set this property value to <code>true</code> . Otherwise, set it to <code>false</code> . (The style sheet makes the job output easier to read than the raw XML output.)

max.jobs.to.return

Set `max.jobs.to.return` to specify the number of jobs to display as a result of a call to the Spotfire Statistics Services URL API.

Property Name	Default Value	Property Description
<code>max.jobs.to.return</code>	1000	This property specifies the maximum number of jobs to display as a result of a call to the URL API function <code>jobs</code> .

Details

You can learn more about the URL API functions in its help, available from the Spotfire Statistics Services landing page.

notification.delay.time

Set the value of `notification.delay.time` to the notification time, in milliseconds.

Property Name	Default Value	Property Description
<code>notification.delay.time</code>	5000	This property specifies the value of the notification time, in milliseconds.

notification.max.times

Set `notification.max.times` to the number of times Spotfire Statistics Services should attempt to send a notification to the client.

Property Name	Default Value	Property Description
<code>notification.max.times</code>	10	This property specifies the number of times Spotfire Statistics Services tries to send a notification to a client.

service.id

The property `service.id` distinguishes Spotfire Statistics Services instances on your local area network. Do not change this value.

Property Name	Default Value	Property Description
<code>service.id</code>	<code>service_name::<GUID></code>	The property <code>service.id</code> identifies Spotfire Statistics Services instances (standalone or cluster) to your local area network.

Details

The default value for the property `service.id` is `service_name::<GUID>`, where the `service_name` is the value you specified at installation and `<GUID>` is a globally unique identifier. For example, the default installation would produce a service ID similar to `statsvcs::32f2c569-8cd2-4f62-926d-ed13a2af9472`. Do not change this value.

service.url

The value for the property `service.url` is the base URL of a standalone or clustered installation of Spotfire Statistics Services.

Property Name	Default Value	Property Description
<code>service.url</code>	<code>http://servername:port/service_name</code>	This setting is the base URL of a standalone or a clustered installation of Spotfire Statistics Services. It is this URL that clients use to access the landing page for Spotfire Statistics Services.


Details

The URL identified by `service.url` is composed of the server name and port number specified in the installation, and of the service name, which is also specified in the installation. You should not change the value for `service.url`.

terr.restricted.execution.mode

The property `terr.restricted.execution.mode` is set by default to `true` to ensure that executing certain operations defined as restricted generates an error. Applies only if you are using the TERR engine in your Spotfire Statistics Services installation (that is, you set `engine.type=TERR`).

Property Name	Default Value	Property Description
<code>terr.restricted.execution.mode</code>	<code>true</code>	<p>This option specifies that TERR evaluates expressions by calling the function <code>terrUtils::evalREX</code>, which ensures that they do not perform I/O on the file system and cannot engage in potentially malicious behavior (such as deleting files or uploading confidential data to the internet). If such an evaluation is attempted, TERR generates an error, such as "Error: restricted call to Native[tempfile]", and execution of the expression is terminated.</p> <p>Restricted behavior includes the following non-exhaustive list of operations.</p> <ul style="list-style-type: none"> • Calling <code>evalREX</code> itself. • Performing any I/O to the file system or the internet. • Loading new packages, except for the libraries included with TIBCO Enterprise Runtime for R (stats, terrUtils, and so on). • Spawning new OS processes (calling 'system'). • Calling <code>.Call</code>, which is used to call Rapi code in CRAN packages. • Calling <code>.C</code> or <code>.Fortran</code>. • Calling into Java using the <code>terrJava</code> package (which allows executing arbitrary Java methods). • Calling any functions in the <code>parallel</code> package (which uses <code>terrJava</code>). • Accessing any function environments in the stack above the call to <code>evalREX</code> using <code>sys.frame</code> or <code>parent.frame</code>. This prevents malicious code from installing functions or expressions that could be executed after leaving restricted execution mode. • Changing the variable lookup path by setting <code>parent.env</code> of an environment, or reading or setting the environment of a closure. • Defining S4 classes and methods using <code>setClass</code> or <code>setMethod</code>.

Property Name	Default Value	Property Description
		<p>If <code>terr.restricted.execution.mode=true</code>, then Spotfire Statistics Services attempts to detect if the version of TERR it is running has <code>terrUtils::evalREX</code> defined. If it does not, then it cannot start any TERR engines, and the system generates many logging errors reading "TERR engine does not support evalREX (update TERR or set <code>spserver.properties</code> item '<code>terr.restricted.execution.mode=false</code>')"</p> <p>See Recommendations for safeguarding your environment on page 13 for more information.</p> <p>For more information about <code>terrUtils::evalREX</code>, see its help file in the <i>TIBCO® Enterprise Runtime for R Language Reference</i>.</p>

webdav.cleanup.run.interval.minutes

Set the value of `webdav.cleanup.run.interval.minutes` to the interval to use for monitoring transient data to be cleaned out of the Spotfire Statistics Services WebDAV root directory.

Property Name	Default Value	Property Description
<code>webdav.cleanup.run.interval.minutes</code>	10	This property specifies the interval used for monitoring transient data that needs to be cleaned out of the WebDAV root directory.

Details

In a standalone installation, the WebDAV root directory is `SPSERVER_HOME/data` and in a cluster, it is `SPSERVER_SHARE/data`. This property works in combination with `webdav.cleanup.time.to.live.minutes`.

The optimal value usually depends on the rate of incoming requests.

On the UNIX/Linux platform, a directory can have no more than 32000 subdirectories. If you are in an environment where many requests are generated in a short period, more result subdirectories will be created than are eligible for cleanup. If the maximum number of subdirectories is reached, Spotfire Statistics Services cannot process additional requests. In this situation, you should estimate maximum throughput and adjust this value and the `webdav.cleanup.time.to.live.minutes` value so that subdirectories are removed more often.

When you change this value, you should carefully monitor the performance of your computer.

webdav.cleanup.time.to.live.minutes

Set `webdav.cleanup.time.to.live.minutes` to indicate when any file is old enough to delete when cleanup runs.

Property Name	Default Value	Property Description
<code>webdav.cleanup.time.to.live.minutes</code>	1440	This property indicates when a file is old enough to delete when cleanup runs.

Details

The optimal value for `webdav.cleanup.time.to.live.minutes` usually depends on the rate of incoming requests.

On UNIX/Linux platforms, a directory can have no more than 32000 subdirectories. If you are in an environment where many requests are generated in a short period, more result subdirectories will be created than are eligible for cleanup. If the maximum number of subdirectories is reached, Spotfire Statistics Services cannot process additional requests. In this situation, you should estimate maximum throughput and adjust this value and the `webdav.cleanup.run.interval.minutes` value so that subdirectories are removed more often.

When you change this value, you should carefully monitor the performance of your computer.

xml.version

Set the version of XML that Spotfire Statistics Services uses. Do not change.

Property Name	Property Value	Property Description
<code>xml.version</code>	1.0	The value of this property sets the XML version for the internal Spotfire Statistics Services settings. Do not change this setting.

Authentication in TIBCO Spotfire Statistics Services

Spotfire Statistics Services uses user properties, Active Directory (AD), or LDAP to authenticate users. Whether Spotfire Statistics Services checks for credentials depends on the authentication property settings.

Determine authentication setting

Spotfire Statistics Services uses user properties, Active Directory (AD), or LDAP to authenticate users. Whether Spotfire Statistics Services checks for credentials depends on the authentication property settings.

The files controlling enabling authentication are the properties files `server.properties`, `users.properties` and `ldap.properties`.

- In a cluster, the properties files (such as `ldap.properties`) are stored in the `conf` directory under `SPSERVER_SHARE`.
- In a standalone installation, the properties files are stored in the `conf` directory under `SPSERVER_HOME`.

In the file `spserver.properties`, enable authentication globally by setting the `authentication.required` property to `true`. (It is set to `false` by default.)

The simplest form of user authentication for Spotfire Statistics Services is an in-memory authentication list, controlled by the file `users.properties`. If authentication is enabled, this file is always checked first, before LDAP or Active Directory is checked. If neither LDAP or Active Directory is specified, Spotfire Statistics Services checks this file for the user login ID and role. Users for this form of authentication can have access as the roles `ROLE_USER`, `ROLE_ADMIN`, or both.

For more sophisticated systems, user credentials for login authentication, e-mail access, and other such activities requiring user access are verified against either the Active Directory or the LDAP provider. To establish authentication with one of these systems, you must set the following properties in the file `ldap.properties`, according to your organization's requirements.

- If you are using Active Directory for authentication, set the property `activeDirectory.enabled` to `true`. (It is set to `false` by default.)
- If you are using LDAP for authentication, set the property `ldap.enabled` to `true`. (It is set to `false` by default.)

If you enable authentication on the server, and then set these properties to `false`, the `user.properties` authentication is used.

If you enable authentication on the server, set either of these properties to `true`, and then configure the appropriate service correctly, the authentication process accesses the enabled service, and the server searches the database.

If you set either of these properties to `true`, but if the corresponding service (that is Active Directory or LDAP) is not configured correctly (for example, because the `ldap.host` property in the `ldap.properties` file is empty), the authentication process fails silently, and it appears as if the user credentials are wrong. Check the log file for more information. (See [configure engine logging](#).)

Active Directory properties

You can use Active Directory (AD) to authenticate users for Spotfire Statistics Services. The AD configuration properties are contained in the file `ldap.properties`. Set these properties if you intend to use AD to authenticate users.



The file `ldap.properties` also contains the properties used for both Active Directory and LDAP authentication. If you intend to use Active Directory rather than LDAP, be sure to set `activeDirectory.enabled` to `true` and `ldap.enabled` to `false`.

- In a cluster, the properties files (such as `ldap.properties`) are stored in the `conf` directory under `SPSERVER_SHARE`.
- In a standalone installation, the properties files are stored in the `conf` directory under `SPSERVER_HOME`.

activeDirectory.enabled

The property `activeDirectory.enabled` indicates whether Spotfire Statistics Services uses Active Directory to authenticate users.

Property Name	Property Description
<code>activeDirectory.enabled</code>	If this property is set to <code>true</code> , and <code>ldap.enabled</code> is set to <code>false</code> , Active Directory is used to authenticate users. If you want to use LDAP for authentication, set <code>activeDirectory.enabled</code> to <code>false</code> . If you set this property to <code>true</code> , you need only set property values for <code>ldap.domain</code> and <code>ldap.host</code> . The default is <code>false</code> .

Example

```
activeDirectory.enabled=true
```

ldap.domain

The property `ldap.domain` identifies the domain within which to search for Active Directory users.

Property Name	Property Description
<code>ldap.domain</code>	Use this property only for Active Directory authentication. This property contains the complete domain name where searches for Active Directory users is kept. For example, <code>usa.mycompany.com</code> is a complete domain name. The default is <code>example.com</code> , which exists to just demonstrate the format.

Example

```
ldap.domain=mycompany.com
```

ldap.host

The property `ldap.host` identifies the host name of the Active Directory server or the LDAP server.

Property Name	Property Description
<code>ldap.host</code>	This property contains the host name of the LDAP or Active Directory server. (Set this property whether you are using LDAP or Active Directory.)

The LDAP URL

This image shows the construction of the LDAP URL, which is constructed of the properties you define. (You need not supply the characters depicted in red; they are not included in the property definitions.)



For Active Directory servers, you need only specify the `ldap.host` property.

Example

```
ldap.host=myauthserver
```

LDAP authentication implementation

As currently implemented for Spotfire Statistics Services, LDAP authentication has two steps.

1. Lookup/search using Distinguished Name (DN) patterns runs if at least one of the corresponding properties `ldap.group0` through `ldap.group9` are defined.

The search pattern is defined as

```
(ldap.accountNameAttr={0}, ldap.groupX)
```

where `{0}` is a username placeholder and `X` is a respective group number. If the specified user is found and authenticated in this first step, step two is omitted.

2. This step is performed if the specified user is not found in the first step. It allows for more complex search scenarios.

The property `ldap.searchBase` defines the context name in which to search, which is relative to the root DN (that is, the property `ldap.path`). This step has two options:

- Use a standard/default filter.
- Use a custom filter.

The default filter is assembled based on the property `ldap.accountNameAttr={0}`.

However, if the property `ldap.customSearchFilter` is defined, it is used in place of the default filter. The `ldap.customSearchFilter` property should contain a placeholder `{0}` for a username.

LDAP properties

You can use LDAP to authenticate users for Spotfire Statistics Services. The LDAP configuration properties are contained in the file `ldap.properties`. Set these properties if you intend to use LDAP to authenticate users.



The file `ldap.properties` also contains the properties used for Active Directory and LDAP authentication. If you intend to use LDAP rather than Active Directory, be sure to set `activeDirectory.enabled` to `false` and `ldap.enabled` to `true`.

- In a cluster, the properties files (such as `ldap.properties`) are stored in the `conf` directory under `SPSERVER_SHARE`.
- In a standalone installation, the properties files are stored in the `conf` directory under `SPSERVER_HOME`.



Spotfire Statistics Services implements LDAP authentication. LDAP authorization is not implemented at this time.

`ldap.accountNameAttr`

The property `ldap.accountNameAttr` identifies the user.

Property Name	Property Description
<code>ldap.accountNameAttr</code>	This property identifies the user. The value represents a placeholder (for example, <code>uid</code> or <code>sAMAccountName</code>); the actual value depends on the particular authentication. The default is <code>sAMAccountName</code> .

Example

```
ldap.accountNameAttr=uid
```

`ldap.basedn`

The property `ldap.basedn` specifies the LDAP base distinguished name (DN), or the top level of the LDAP directory tree.

Property Name	Property Description
<code>ldap.basedn</code>	This property specifies the top level of the LDAP directory tree (the base distinguished name (DN) is the group search base; for example <code>OU=groups</code>). (OU specifies Organizational Unit.) You can leave this blank. The default is <code>OU\=All Users</code> .

Example

```
ldap.basedn=OU\=All Users
```

`ldap.customSearchFilter`

The property `ldap.customSearchFilter` defines a filter for searching across the user database.

Property Name	Property Description
<code>ldap.customSearchFilter</code>	If this property is defined, it is used in place of the default filter. It should have a placeholder <code>{0}</code> for the username. If this property is defined, then a filter-based LDAP user search is performed using the

Property Name	Property Description
	<code>ldap.searchBase</code> property value and the custom search filter in the LDAP context (the host definition). If the custom search is not defined, the search is performed using the variable <code>ldap.accountNameAttr={0}</code> . The default is blank.

Example

```
ldap.customSearchFilter={0}
```

ldap.enabled

The property `ldap.enabled`, when set to `true`, indicates that LDAP is used as the authentication source.

Property Name	Property Description
<code>ldap.enabled</code>	This property must be set to <code>true</code> if you have an LDAP server and you want to use it for authentication. If you have an Active Directory server, and you want to use it for authentication, set this property to <code>false</code> . The default is <code>false</code> .

Example

```
ldap.enabled=true
```

ldap.groupMemberAttr

The property `ldap.groupMemberAttr` specifies the pattern for the user search.

Property Name	Property Description
<code>ldap.groupMemberAttr</code>	This property specifies the pattern to be used for the user search. That is, it defines the <code>Group Search Filter</code> attribute. The default is <code>sAMAccountName</code> .

Example

```
ldap.groupMemberAttr=sAMAccountName
```

ldap.groupRoleAttr

The property `ldap.groupRoleAttr` specifies the ID of the attribute that contains the role name for a group.

Property Name	Property Description
<code>ldap.groupRoleAttr</code>	This property specifies the ID of the attribute that contains the role name for a group. The default is <code>OU</code> (Organizational Unit) or <code>cn</code> (Common Name). The default is <code>OU</code> .

Example

```
ldap.groupRoleAttr=OU
```

ldap.groupnum

The `ldap.group#` property specifies the group number(s) that the user can be in. (# specifies a number 0 through 9.)

Property Name	Property Description
<code>ldap.group#</code>	This property specifies the groups (where # is a number 0 through 9) that a user can be in. These groups must be properly formatted, because invalid formats can cause any user search filtering to be skipped.

Example

```
ldap.group2=OU\=US_NewYork, OU\=All Users
ldap.group1=OU\=US_SanFrancisco, OU\=All Users
ldap.group0=OU\=US_Denver, OU\=All Users
```

ldap.host

The property `ldap.host` identifies the host name of the Active Directory server or the LDAP server.

Property Name	Property Description
<code>ldap.host</code>	This property contains the host name of the LDAP or Active Directory server. (Set this property whether you are using LDAP or Active Directory.)

The LDAP URL

This image shows the construction of the LDAP URL, which is constructed of the properties you define. (You need not supply the characters depicted in red; they are not included in the property definitions.)




For Active Directory servers, you need only specify the `ldap.host` property.

Example

```
ldap.host=myauthserver
```

ldap.manager.anonymous

The property `ldap.manager.anonymous` allows for anonymous read-only access of the LDAP database.

Property Name	Property Description
<code>ldap.manager.anonymous</code>	<p>Setting this property to <code>true</code> allows anonymous read-only access to the LDAP database. When set to <code>true</code>, the server directs the authentication mechanism to attempt to authenticate users (if authentication is required), using an anonymous mechanism. The default is <code>false</code>.</p> <p>If anonymous access is enabled using this property, the value for <code>ldap.manager.cn</code> user does not need a valid value for <code>ldap.manager.pass</code>.</p> <div>  <p>Some LDAP servers might not support this option; they require additional configuration to allow anonymous read-only access.</p> </div>

ldap.manager.cn

The property `ldap.manager.cn` specifies the name of the manager user.

Property Name	Property Description
<code>ldap.manager.cn</code>	This property specifies the Common Name (cn) of the manager user. It is used only for the LDAP connection. Set the user distinguished name (DN) for accessing an authenticated LDAP server. (The example builds the entire DN.)

Example

```
ldap.manager.cn=CN\=TestUser,OU\=TestAccounts,OU\=All Users Other,dc
\=company,dc\=com
```

Term	Definition	Example	Comment
CN	Common Name	TestUser	Can be a generic name and a specific user's name, such as John Smith
OU	Organizational Unit	Test Accounts, All Users Other	Indicates the organization units where CN is located
dc	Domain Component	company, com	Builds the domain name (company.com)

ldap.manager.pass

The property `ldap.manager.pass` contains the password for manager access.

Property Name	Property Description
<code>ldap.manager.pass</code>	This property contains encrypted password for manager access. If an unencrypted password is provided, the first time Spotfire Statistics Services starts, it encrypts the password and writes the encrypted result to this property.

Example

```
ldap.manager.pass=3DES{ZFeEJVWgSE6ctm3nkDycsg\=\=}
```

ldap.path

The property `ldap.path` specifies the path to the LDAP directory.

Property Name	Property Description
<code>ldap.path</code>	This property specifies the path to the LDAP directory. This is part of the URL used to construct the path to the server. Used to construct the LDAP address. The default is <code>dc\=company,dc\=com</code> .

The LDAP URL

This image shows the construction of the LDAP URL, which is constructed of the properties you define. (You need not supply the characters depicted in red; they are not included in the property definitions.)



For Active Directory servers, you need only specify the `ldap.host` property.

Example

```
ldap.path=dc\=myldap,dc\=mycompany,dc\=com
```

`dc`, or domain component, builds the path as `myldap.mycompany.com`

ldap.port

The property `ldap.port` specifies the port over which the LDAP service is provided.

Property Name	Property Description
<code>ldap.port</code>	This property specifies the port over which LDAP service is provided. For LDAP, the default port is 389 (the default). For LDAPS, the default port is 636.

The LDAP URL

This image shows the construction of the LDAP URL, which is constructed of the properties you define. (You need not supply the characters depicted in red; they are not included in the property definitions.)





For Active Directory servers, you need only specify the `ldap.host` property.

Example

```
ldap.port=389
```

ldap.protocol

The `ldap.protocol` property specifies whether to use LDAP or LDAPS.

Property Name	Property Description
<code>ldap.protocol</code>	This property specifies whether to use LDAP or LDAPS. LDAPS uses SSL connections instead of plain (that is, unprotected) connections. The default is <code>ldap</code> .

The LDAP URL

This image shows the construction of the LDAP URL, which is constructed of the properties you define. (You need not supply the characters depicted in red; they are not included in the property definitions.)



For Active Directory servers, you need only specify the `ldap.host` property.

Example

```
ldap.protocol=ldap
```

ldap.searchBase

The property `ldap.searchBase` specifies the location in a directory from which to start the search for a user.

Property Name	Property Description
<code>ldap.searchBase</code>	This property specifies the location in a directory from which to start the search for a user. This property is typically left blank, but it can be used to define a more specific domain name. The default is <code>OU\=All Users</code>

Example

```
ldap.searchBase=OU\=All Users
```

ldap.searchSubTree

If the property `ldap.searchSubTree` is set to `true`, it searches across the entire subtree.

Property Name	Property Description
<code>ldap.searchSubTree</code>	If this property is set to <code>true</code> , searches across the entire subtree (given the context) are performed. If it is set to <code>false</code> , then searches are performed only on the level identified by the context, which means that the configuration must be exact. the default is <code>true</code> .

Example

```
ldap.searchSubTree=true
```

Configuring LDAPS

If you are using an SSL connection, you can configure Spotfire Statistics Services to use LDAPS.

Procedure

1. Obtain the LDAP server's CA certificate. For example, if you were using a Windows Server 2008 domain controller that is also acting as the CA, you can obtain the root certificate by issuing the following command at the Command Prompt.

```
certutil -ca.cert ca.cert
```

2. On the Spotfire Statistics Services server, import the certificate. For example, using the example above with the `ca.cert` file, you would issue the following commands from the `SPSERVER_HOME\jre\bin` directory, at the server's Command Prompt:

```
keytool -import -alias dc -keystore C:\(keystore_dir)\keystore.jks -file C:\(path_to)\ca.cert -trustcacerts
keytool -import -alias dc -keystore C:\(keystore_dir)\cacerts.jks -file C:\(path_to)\ca.cert -trustcacerts
```

where `(keystore_dir)` is the path where you want to import and register the trusted certificate and `(path_to)` is the directory where you have copied `ca.cert`.

These two commands generate the files `keystore.jks` and `cacerts.jks`. When you issue the commands, specify passwords for the files. Make a note of the passwords you use, because you will need them in the step next step.

3. For the Apache Tomcat JVM, set the Java options pointing to the keystore and truststore files created in the previous step.

On Windows, do this by opening `SPSERVER_HOME\tomcat\bin\<service_name>.exe`, selecting the Java tab, and then adding the following properties under Java Options.

```
-Djavax.net.ssl.keyStore=C:\some\directory\keystore.jks
-Djavax.net.ssl.keyStorePassword=somesecret
-Djavax.net.ssl.trustStore=C:\(keystore_dir)\cacerts.jks
-Djavax.net.ssl.trustStorePassword=somesecret
```

On UNIX/Linux, add these properties to the appropriate location in the startup script.

4. Modify `SPSERVER_HOME/conf/ldap.properties`, to change the following properties.

```
ldap.protocol=ldaps
ldap.port=636
```

5. Restart the Spotfire Statistics Services service.

SSL configuration

If you need to configure a secure socket layer (SSL) for a stand-alone installation or a single-Manager cluster installation of Spotfire Statistics Services, you can follow Apache Tomcat documented processes.

In the case of the single Manager node cluster with no load balancer, you do not need to perform this configuration on Worker nodes. If you have a load balancer, you must consult your load balancer documentation concerning SSL configuration.

For detailed information and procedures on configuring SSL with Apache Tomcat, follow the procedures in the SSL Configuration HOW-TO on the Apache Tomcat Web site.

<http://tomcat.apache.org/tomcat-8.0-doc/ssl-howto.html>

You can use this information to configure SSL with Spotfire Statistics Services. If you are using a default installation, you can set `SPSERVER_HOME/jre` as `JAVA_HOME` and use the Apache Tomcat configuration file, `server.xml`, located in `SPSERVER_HOME/tomcat/conf`.

After you have configured SSL, you must modify the properties in the `spserver.properties` configuration file at `SPSERVER_HOME/conf/` (for a standalone installation) or in `SPSERVER_HOME/conf` (if you installed a cluster with only one Manager node and no load balancer). In most cases, you need to update the `service.url` property only. For example, the following non-default `service.url` assignment indicates that Spotfire Statistics Services is using an SSL Connector on port 8443:

```
service.url=https://servername:8443/<service_name>/
```

Server environment properties

When Spotfire Statistics Services starts, it establishes the environment by checking for the existence certain Java properties. These Java properties are specified during Spotfire Statistics Services installation.

Java Property	Description
<code>spserver.share</code>	Path to the centralized configuration and shared storage area for a Spotfire Statistics Services cluster. In a standalone installation, the default value is set to the base installation directory.
<code>spserver.home</code>	Path to the base installation directory of Spotfire Statistics Services . The default values are: <ul style="list-style-type: none"> Windows: Spotfire Statistics Services on Windows: C:\Program Files\TIBCO\statsvcs711\<service_name> UNIX/Linux: /opt/TIBCO/statsvcs711
<code>spserver.terrrhome</code>	Path to the installation for the TERR engine. The default values are: <ul style="list-style-type: none"> Windows: C:\Program Files\TIBCO\statsvcs711\<service_name>\engines\terr UNIX/Linux: /opt/TIBCO/statsvcs711/engines/terr
<code>spserver.rhome</code>	Path to the installation for the R engine. For more information, see Configuring an open-source R engine .
<code>spserver.sashome</code>	Path to the installation for the SAS engine. For more information, see Configuring a SAS engine .
<code>spserver.matlabhome</code>	Path to the installation for the MATLAB engine. For more information, see Configuring a MATLAB engine .

In most cases, the values set by the installer should be sufficient; however, you can customize these values. If you want to modify the values on a Windows computer, see [Using Procrun to modify a Java property](#). On UNIX/Linux computers, you can modify the values by editing the `SPSERVER_HOME/init.d/spserver` file. You can modify the values for any of the properties and the next time you restart the service your changes will appear. We recommend that you make a backup of the file before you make any modifications.

In some cases, you might find it necessary to override the Java properties temporarily without altering the values set by the installer. You can do this by setting any one of the environment variables.

Each environment variable is equivalent to its corresponding Java property except the environment variable takes precedence. For example, if you want to override the value that the installer set for `spserver.terrrhome`, you can specify the `SPSERVER_terrrhome` environment variable as in the following example:

```
SPSERVER_terrrhome=/path/to/alternate/terr.
```

We recommend that you avoid using environment variables because when you mix environment variables and Java properties it can be confusing when you need to troubleshoot your installation. If

you do need to verify your environment, you can review the Spotfire Statistics Services log for text indicating the values used to establish the environment, these settings are added to the log file soon after startup. See [Configure engine logging](#) for more information.

WebDAV security configuration

If you want to control the roles that use the webdav path in the Spotfire Statistics Services installation, create a file called `webdav.security.config` and set properties to manage that control.

The file `webdav.security.config` is used to configure the roles that are required for using the `/webdav/` path within the Tomcat container (that is, `http://server/<server_name>/webdav`, where `server` is the server name.). The file contains a specific format for providing access. The settings are similar to the following two options.

- `/webdav/.*=ROLE_USER,ROLE_ADMIN`
- `/webdav/directoryA/=ROLE_USER,ROLE_ADMIN`

If you do not create the file and specify a different configuration, the first example is used as the default.



The syntax for your configuration must be exactly as demonstrated above. That is, `/directory/regexp` (on the left) equals (=) the comma-delimited role definition(s) on the right.

Remote monitoring and management with JMX

You can monitor Spotfire Statistics Services using TIBCO Hawk or any other Java Management Extensions (JMX) -compliant monitoring tool, like JConsole, which is a part of the Java SDK.

When you install Spotfire Statistics Services, Remote Monitoring and Management is enabled by default. During installation, you are prompted for a required user name (by default, `admin`) and password. These credentials are specific to this standalone server installation or to the entire cluster. We recommend against using your login/domain credentials.

You must access the JMX server from your chosen monitoring tool using the following URL:

```
service:jmx:rmi:///jndi/rmi://[SERVER_NAME]:[JMX_PORT]/jmxrmi
```

Using remote management and monitoring through JMX, you can perform the following tasks:

- Restart Manager or Worker nodes.
- View and edit the server configuration.
- Perform job and package management.
- Manage nodes in a cluster.

After installation, you can use the `spserver.properties` settings to perform the following tasks:

- [Change JMX options to enable or disable remote monitoring and management.](#)
- [Change the JMX username or password.](#)
- [Change the JMX port number.](#)

For more information on additional configuration options for Remote Management and Monitoring via JMX, see <http://java.sun.com/javase/7/docs/technotes/guides/management/agent.html>.

Enabling or disabling remote monitoring and management with JMX

Use the techniques described in this topic to manage remote monitoring with JMX in a Spotfire Statistics Services installation.

Procedure

1. Open the `spserver.properties` file from the following location
 - **Cluster:** `SPSERVER_SHARE/conf`
 - **Standalone:** `SPSERVER_HOME/conf`
2. Set the `jmxremote.enabled` value:
 - `true` to enable remote monitoring and management.
 - `false` to disable remote monitoring and management.
3. Restart the Spotfire Statistics Services service.
 After changing the server properties, you must restart the Spotfire Statistics Services service for the changes to take effect. In the case of a cluster, you must restart the Spotfire Statistics Services service on each node.

Changing the username or password for remote monitoring and management

You can change the JMX username or password by changing a Spotfire Statistics Services property.

Procedure

1. Open the `spserver.properties` file from the following location
 - **Cluster:** `SPSERVER_SHARE/conf`
 - **Standalone:** `SPSERVER_HOME/conf`
2. Edit the following server properties to set the new username and password values:
 - `jmxremote.username=[JMX_USERNAME]`
 - `jmxremote.password=[JMX_PASSWORD]` (encrypted on first startup)
3. Restart the Spotfire Statistics Services service.
 After changing the server properties, you must restart the Spotfire Statistics Services service for the changes to take effect. In the case of a cluster, you must restart the Spotfire Statistics Services service on each node.

Modifying the JMX port number

You can change the JMX port number by changing a TIBCO Spotfire® Statistics Services property.

Procedure

1. Open the `spserver.properties` file from the following location
 - **Cluster:** `SPSERVER_SHARE/conf`
 - **Standalone:** `SPSERVER_HOME/conf`
2. Edit the `jmxremote.port` value to be the port number that you want to use.
3. Restart the Spotfire Statistics Services service.

After changing the server properties, you must restart the Spotfire Statistics Services service for the changes to take effect. In the case of a cluster, you must restart the Spotfire Statistics Services service on each node.

Changing the protocol or port

You can change either the protocol or the port provided when Spotfire Statistics Services was installed.

Procedure

1. Edit `SPSERVER_HOME/tomcat/conf/server.xml` to modify the values for protocol or port. For example, the default installation includes the following.

```
<Connector port="8080" URIEncoding="UTF-8"
connectionTimeout="20000"
protocol="org.apache.coyote.http11.Http11NioProtocol"/>
```

Possible values for protocol are `Http11NioProtocol` or `AJP/1.3`. If you are not using a native Web Server as a proxy or load balancer, you must specify `Http11NioProtocol`.

2. If you modify the port, you must change the `service.url` property in the `spserver.properties` configuration file to reflect the new value. See [service.url](#) on page 62 for more information.
3. Restart the Spotfire Statistics Services service.

Configuring an open-source R engine

To use an open-source R engine with Spotfire Statistics Services, rather than the TERR™ engine provided with the installation, you must perform additional configuration steps.

Spotfire Statistics Services version 7.11.2 was tested with open-source R version 3.4.4.



Remember that when installing and configuring Spotfire Statistics Services to use a statistical engine, consider the configurations and strategies to restrict access. See [Recommendations for safeguarding your environment](#) on page 13 for detailed information.



Open-source R is available under separate open source software license terms and is not part of TERR. As such, open-source R is not within the scope of your license for TERR. Open-source R is not supported, maintained, or warranted in any way by TIBCO Software Inc. Download and use of open-source R is solely at your own discretion and subject to the free open source license terms applicable to open-source R.

Procedure

1. Open the `spserver.properties` file from the following location:
 - **Cluster:** `SPSERVER_SHARE/conf`
 - **Standalone:** `SPSERVER_HOME/conf`
2. Set the property `engine.type=R`.
3. Download and install the tested version of open-source R, as listed on the **Systems Requirements** page, available from [TIBCO Spotfire® Products System Requirements](#).
You can download open-source R from <http://www.r-project.org/>.



If you plan to build open-source R from source, it must be configured with the option `--enable-R-shlib`. If you are downloading a distribution, this option was likely included in the build process.

4. Install the `rJava` package.
 - a) Start an open-source R desktop session.

- b) In the open-source R Console, type `install.packages("rJava", install_path)` where *install_path* is the library path where open-source R running under Spotfire Statistics Services expects to find it. For example, it might be `C:/Program Files/R/R-3.4.4/library`.




If you do not supply a path for this argument, you might encounter problems using `rJava`.

- c) Select a CRAN mirror, and then open-source R installs the `rJava` package.
d) Validate the `rJava` installation in the R console by running the following command:

```
library(rJava)
```

5. Perform the following to include the path to the open-source R installation

Environment	Description
Windows	<p>Add <code>-Dspserver.rhome=path-to-R</code> to the Java options for the service. For more information, see Using Procrun to modify a Java property.</p> <p> The <i>path-to-R</i> where you installed open-source R, at the level of its version number, and no lower (for example, <code>R-N.n.n</code>).</p> <p>See the following examples.</p> <pre>-Dspserver.rhome=C:\Program Files\R\R-3.4.4</pre> <p>or</p> <pre>-Dspserver.rhome=C:\R\R-3.4.4</pre> <p>If jobs fail, and you see the following error in <code>SplusServer.log</code>, the path you provided in <code>-Dspserver.rhome</code> to your installation of open-source R is incorrect.</p> <pre>java.lang.IllegalStateException: Required R executable file is not found</pre>
UNIX/Linux	Edit <code>SPSERVER_HOME/init.d/spserver</code> to specify <code>SPSERVER_RHOME=path-to-R</code> .

6. Restart Spotfire Statistics Services.

Configuring an S-PLUS engine

To use an S-PLUS engine with Spotfire Statistics Services, rather than the TERR™ engine provided with the installation, you must perform additional configuration steps.

This release of Spotfire® Statistics Services includes the Spotfire S+ (S-PLUS) language package for backwards compatibility only. It will be removed in future releases.



Remember that when installing and configuring Spotfire Statistics Services to use a statistical engine, consider the configurations and strategies to restrict access. See [Recommendations for safeguarding your environment](#) on page 13 for detailed information.

Procedure

1. Open the `spserver.properties` file from the following location:


- **Cluster:** `SPSERVER_SHARE/conf`
- **Standalone:** `SPSERVER_HOME/conf`

2. Set the property `engine.type=SPLUS`.

3. Optional: Include the path to the S-PLUS installation.



This step is required only if you are using an S-PLUS installation separate from the S-PLUS engine included with Spotfire Statistics Services.

Environment	Description
Windows	<p>Add <code>-Dspserver.shome=path-to-S-PLUS</code> to the Java options for the service. For more information, see Using Procrun to modify a Java property.</p> <p> The <code>path-to-S-PLUS</code> is at the level of its version number, and no lower.</p> <p>See the following examples.</p> <pre>-Dspserver.shome=C:\Program Files\splus\splus-8.2.0</pre> <p>or</p> <pre>-Dspserver.shome=C:\splus\splus-8.2.0</pre> <p>If jobs fail, and you see the following error in <code>SplusServer.log</code>, the path you provided in <code>-Dspserver.shome</code> to your installation of S-PLUS is incorrect.</p> <pre>java.lang.IllegalStateException: Required S-PLUS executable file is not found</pre>
UNIX/Linux	Edit <code>SPSERVER_HOME/init.d/spserver</code> to specify <code>SPSERVER_SHOME=path-to-S-PLUS</code> .

- Restart Spotfire Statistics Services.

Configuring a SAS engine

To use SAS[™] software as the Spotfire Statistics Services engine, you must perform additional configuration steps for each Spotfire Statistics Services installation.

Prerequisites

The SAS software must be installed on the same machine as the server. For cluster installations, SAS software must be installed on every machine where a server node is run. The only part of the SAS software that Spotfire Statistics Services requires is Base SAS. Other SAS software packages are not required, but they can be installed on the machine.

For more information about tested versions, see the Spotfire Statistics Services System Requirements section of the support web site at <http://support.spotfire.com/sr.asp>.



On Microsoft Windows[®], Spotfire Statistics Services normally would run as a Windows service. The properties window for a service contains a **Log On** tab, specifying how the service process should be executed; either **Log in as: Local System account**, or **Log in as: This account**. Use **Log in as: This account** for the service if the time to start up a SAS software task needs to be minimized.

Procedure

- Edit the `spserver.properties` file to set `engine.type=SAS`.
The properties file is located at `SPSERVER_HOME/conf` for a standalone installation or `SPSERVER_SHARE/conf` on a cluster. For more information, see [engine.type](#).
- Set the following property to identify the path to the SAS software installation:

Operating System	Description
Microsoft Windows [®]	Add <code>-Dspserver.sashome=/path/to/SAS</code> to the Java options for the service. For more information, see Procrun .
UNIX/Linux	Edit <code>SPSERVER_HOME/init.d/spserver</code> to specify <code>SPSERVER_SASHOME=/path/to/SAS</code> .



SASHOME must be set not to the SAS installation, but to the location of the SAS executable (sas.exe).

3. Restart Spotfire Statistics Services.
4. Test Spotfire Statistics Services with SAS software operations by sending a simple SAS script to a Web browser. For example, use the following script (assuming the server is accessed via the URL `http://myserver:8966/server_name`).

```
http://myserver:8966/server_name/api/v8/expression
/eval?cmd=DATA out;DO x=1 to 3;y=x*100;rnum =
UNIFORM(-1);OUTPUT;END;RUN;
```

If Spotfire Statistics Services is running correctly, it should return XML for a data frame with three rows and three columns.

It should also return text giving the SAS software log for the script, listing the SAS code executed, execution time, and errors, as well as any text produced by the SAS PRINT procedure.

Configuring a MATLAB engine

To use MATLAB™ software as a Spotfire Statistics Services engine, you must perform additional configuration steps for each Spotfire Statistics Services installation.

Prerequisites

MATLAB must be installed on the same machine as the server. For cluster installations, MATLAB must be installed on every machine where a server node is run. Spotfire Statistics Services has been tested on machines with MATLAB installed as an **Individual - Standalone Named User**. The only part of MATLAB that Spotfire Statistics Services requires is basic MATLAB. Other MATLAB toolkits are not required, but they can be installed on the machine.

For more information about tested versions, see the Spotfire Statistics Services System Requirements section of the support web site at <http://support.spotfire.com/sr.asp>.

On Microsoft Windows®, Spotfire Statistics Services normally would run as a Windows service. The properties window for a service contains a **Log On** tab specifying how the service process should be executed; either **Log in as: Local System account**, or **Log in as: This account**. When using Spotfire Statistics Services with MATLAB, you must use the **Log in as: This account** option for MATLAB to detect the appropriate user license.

Procedure

1. Edit the `spserver.properties` file to set `engine.type=MATLAB`.
The properties file is located at `SPSERVER_HOME/conf` for a standalone installation or `SPSERVER_SHARE/conf` on a cluster. For more information, see [engine.type](#)
2. Set the following property to identify the path to the MATLAB installation:

Operating System	Description
Microsoft Windows	Add <code>-Dspserver.matlabhome=/path/to/MATLAB</code> to the Java options for the service. For more information, see Procrun
UNIX/Linux	Edit <code>SPSERVER_HOME/init.d/spserver</code> to specify <code>SPSERVER_MATLABHOME=/path/to/MATLAB</code> .

3. Restart Spotfire Statistics Services.
4. Test Spotfire Statistics Services with MATLAB by sending a simple MATLAB script to a Web browser. For example, use the following script (assuming the server is accessed via the URL `http://myserver:8966/server_name`).

```
http://myserver:8966/server_name/api/v8/expression/eval?cmd=
struct('x',transpose(1:3),'y',100*transpose(1:3),'rnum',randn(3,1))
```

If the server is running correctly, it should return XML for a data frame with three rows and three columns. It should also return text giving the MATLAB text output for the script.

Configuring TIBCO Spotfire to use TIBCO Spotfire Statistics Services

If statisticians in your company develop and use data functions, or if they use the predictive analytics tools in Spotfire, and if they run these analyses through Spotfire Statistics Services, you must provide the service URL to the Spotfire Server administrator.

If you are the licensed Spotfire administrator, you can configure this option.



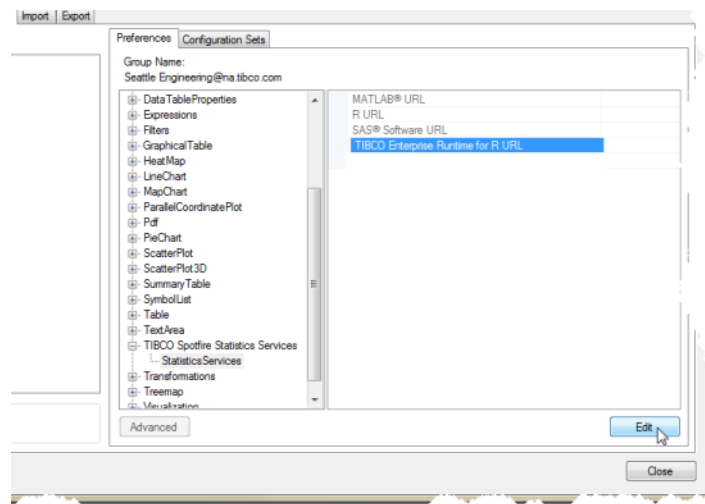
If you are the administrator for Spotfire Web Player, remember that it does not include an statistical engine. Rather, it relies on the engine configured in Spotfire Statistics Services and specified in Spotfire® Analyst application, as described below.

Prerequisites

- You must have installed, configured, and validated Spotfire Statistics Services, you must know which statistical engine the service uses, and have you must know the service URL.
- You must have access to a computer on which Spotfire Professional is installed.
- You must have administrative privileges for Spotfire.

Procedure

1. Log in to Spotfire Professional using administrator credentials.
2. Click **Tools > Administration Manager**.
3. Select the main **Preferences** tab.
4. Select the group for which you want to set the URL, and then select the secondary **Preferences** tab.
5. Scroll through the preferences until you find **TIBCO Spotfire Statistics Services** Spotfire Statistics Services. Expand this option and select **Statistics Services**.
The right pane displays the default URLs for all engine types.
6. Select the engine for which you want to set the URL, and then click **Edit**.



7. In the resulting box, type the URL for Spotfire Statistics Services. Include the service name. For example, `http://MyServer:8080/TERRServer`.
8. Inform the Spotfire Analyst users that you have set the default URL.

Engine affinity

To improve performance on a multi-CPU Microsoft Windows® computer running multiple engines at a time, Spotfire Statistics Services manages the affinity of engine processes by assigning each process to a particular CPU, rather than moving engine assignments over time. This design is configurable for cases where the default behavior is not desirable.

By default, the engine affinities are assigned from any of the available CPUs. Thus, if you are running Spotfire Statistics Services with four engines on an eight-processor computer, the default settings would probably assign engine affinities to the first four CPUs.

However, this behavior can cause a problem if you want to run multiple copies of on a single multi-CPU machine: for example, if you want one server with TERR engines, and another server with R engines on the same machine. If the two independent servers assign engine affinities to the same CPUs, you could end up with some CPUs running multiple engines, and other CPUs left idle.

To address this situation, you can control which CPUs are used for assigning engine affinities by setting the Java property `-Engine.affinity`. This property is set using Procrun. (See [Monitor the service.](#))

The value of the `-Engine.affinity` lists the CPU numbers that are available for assigning engine affinity (starting from one).

Examples

For the settings in the following table, assume that the server computer has eight CPUs.

Java Property Setting	Description
<code>-Dengine.affinity=</code>	Use CPUs 1,2,3,4,5,6,7,8 for affinity. If the value is blank, the default is to use all CPUs.
<code>-Dengine.affinity=1,2,3</code>	use CPUs 1,2,3.
<code>-Dengine.affinity=1-3</code>	an alternative to specify using CPUs 1,2,3.
<code>-Dengine.affinity=1,3,6-20</code>	Use CPUs 1,3,6,7,8 (numbers greater than the number of CPUs are ignored).
<code>-Dengine.affinity=1-9999</code>	an alternative to specify all CPUs.
<code>-Dengine.affinity=0</code>	disable assigning engine affinity

Suppose that you want to run two servers on a computer with eight CPUs.

In this case, the first server could be set up with the following:

```
-Dengine.count=4
-Dengine.affinity=1-4
```

The second server could be set up with the following:

```
-Dengine.count=4
-Dengine.affinity=5-8
```

With these properties, the engines from the two servers would run on totally different sets of CPUs.

Usually, you would want to define `-Dengine.affinity` to specify as many CPUs as the server has engines, but it is not necessary. For example, if you specify the following on an eight-CPU machine, then two engines run on each of the first two CPUs.

```
-Dengine.count=4
-Dengine.affinity=1-2
```



Spotfire Statistics Services does not set engine affinities on non-Windows machines, nor on single-CPU machines.

Database configuration

Select the type of database you intend to use and configure it for Spotfire Statistics Services.

After you have configured your Spotfire Statistics Services installation, indicate whether you are using the default or the non-default database, and then install and configure the JDBC driver for the database. Check to make sure you have set all appropriate properties for the database in the properties file.

Job database requirements

By default, Spotfire Statistics Services relies on H2 database implementation to maintain the job queue.

- The standalone installation configures H2 in the in-process mode. This installation option requires no additional configuration, and simply works out of the box.
- The cluster installation configures H2 in the server mode. This installation option requires minor post-installation configuration.

When H2 is running in a Spotfire Statistics Services cluster (that is, in server mode), one of the managed servers is configured to run a true JDBC-compliant database, which is used by all other nodes in the cluster.



H2 provides no failover capability for a Spotfire Statistics Services cluster.

Optionally, you can use an enterprise-class database server. Spotfire Statistics Services works with the following supported enterprise-class database servers:

- Oracle (10g or later).
- Microsoft SQL Server (2005 or later).
- MySQL (5.0.89 or later).

Configure the H2 database server

Configure an H2 database to use as the Spotfire Statistics Services job queue database.

Configuring an H2 database requires that you set only the port number and password only in the database properties described in the database properties options. There are no configuration files in the database where you need to match these settings, as there are for databases such as MYSQL and ORACLE. You need only set or change the port number and/or the password in the server's `database.properties` file.

Configuring a non-default database

If you are using a database other than H2, you must provide additional configuration.



If you are using the default H2 database with a stand-alone Spotfire Statistics Services, you can disregard this information. See [Configure the H2 database server](#) on page 85.

Procedure

1. Create a dedicated database to use with Spotfire Statistics Services.
2. In the new database, run the script we provide to create the tables and triggers for Spotfire Statistics Services.
Scripts for the supported database servers are located in `SPSERVER_HOME/db_scripts`.
3. You must grant all privileges to the user account that you will use to connect to the database from Spotfire Statistics Services.

Configure the package for a Teradata Aster database

Spotfire provides a connector to a Teradata™ database. Spotfire Statistics Services also includes an out-of-the-box data function example for using the Teradata Aster database for analytic processing with data stores. For developers who want to create and use data functions for the Teradata Aster Database, Spotfire Statistics Services includes the TERR package AsterDB.

The AsterDB package requires special configuration, and the same package configuration must be used by all clients, whether they are using it via Spotfire Professional to develop data functions, or whether they are accessing it on Spotfire Statistics Services via Spotfire Web Player. For these reasons, the AsterDB package is included with Spotfire Statistics Services instead of TERR.

The AsterDB package also requires Java 6 or higher, with the JVM, installed on the developer's machine, and that JAVA_HOME be set to that installation location.



If your users want to run the Aster Data Function examples in Spotfire, you must have installed on your Aster Database the nPath Map Reduce application, as well as have the Aster-provided bank.web.clicks database table loaded.

As Spotfire Statistics Services Administrator, you can install and configure the package, and then redeploy the configured package to Spotfire Server (for distribution to Spotfire Professional users developing data functions) and to Spotfire Statistics Services clusters (accessed by Spotfire Web Player users).

Before you follow the steps to complete these tasks, make sure you have the required information about the Aster Database (including its host, its port, its name, its username, and its password.)


Configuring the AsterDB package

For TIBCO Spotfire® Data Function developers to be able to use the TERR AsterDB package, you must follow the installation and configuration steps to make it available to the TERR engine on Spotfire Statistics Services, and to deploy it to use with Spotfire Data Functions.

Procedure

1. Run TERR engine console as administrator.
In the Spotfire Statistics Services installation, you can find the engine in TSSS_HOME/engines/Terr/bin directory. Right-click TERR.exe and select **Run as administrator**.
(The function `install.packages()` must write to the TERR installation directory, and if TERR is installed under C:\Program Files, administrator access is required.)
2. From within the TERR console, download and install the package from the Spotfire Statistics Services repository by running the following function:
 - a) `install.packages("AsterDB", repos="http://yourserver:port#/service_name/update/TERR")`.
 - b) Quit the TERR console by typing `q()` at the command prompt.
3. Browse to the Teradata® Web site, and then locate and download the Aster JDBC driver ("noarch-aster-jdbc-driver.jar").
4. Copy this driver to TSSS_HOME/engines/Terr/library/AsterDB/java.
5. Make sure JAVA_HOME is set (for example, C:\Program Files\Java\jre1.8.0_171). (AsterDB requires the terrJava package, which requires this environment variable to be set.)
6. Run the TERR engine console again, and then load the package:
`library(AsterDB)`
You should see the following:

```
> library(AsterDB)
Loading required package: sjdbc
Loading required package: terrJava
>
```

7. Test that the library loaded successfully using the TERR command prompt, type `?aster.connect`. A browser window should open and display the help topic for this AsterDB function.
 8. Configure your database settings.
 - a) In TERR, call the function `encryptString("AsterDatabasePassword")` where `AsterDatabasePassword` is the unencrypted password to your Aster database. The encrypted string for this password is returned.
 - b) Copy this encrypted string to your clipboard.
 - c) Browse to the `TSSS_HOME/engines/Terr/library/AsterDB` directory, and using a text editor, open and edit the file `DatabaseConfig.dcf`.
 1. Set the alias property to the value data function developers should use to refer to this database configuration from the first argument of the TERR function `aster.connect()`. (Be sure to give this alias to the data function developers writing code to call that function.)
 2. Set the password property value by pasting the encrypted password you copied.
-  You can specify multiple Aster databases in this configuration file by copying and pasting the entire block (separated by a blank line), specifying a unique alias for each, and supplying the appropriate encrypted password for each.
- See the Aster Database documentation for more information.
9. Test the AsterDB package by repeating steps 6 and 7, using one of the examples in the help file for `aster.connect()`.
 10. Now that you have the AsterDB package configured correctly, you can deploy it to a Spotfire Statistics Services cluster, and you can deploy it to your Spotfire servers, to be distributed to Spotfire Analyst users. See the [TERR Package Management](#) documentation for detailed information about performing these tasks.

Configuring the JDBC driver

You must make sure that the JDBC driver for your database server is available in your environment.

Procedure

1. Obtain the JDBC driver for your database server. For more information, see the documentation for your database server.
2. Copy the JDBC driver to the following directory:

Option	Description
Cluster	SPSERVER_SHARE/endorsed
Standalone	SPSERVER_HOME/endorsed

3. Restart the service. (For cluster configurations, you must restart the service for each node.)

Database properties

The database configuration properties for Spotfire Statistics Services are contained in the file `database.properties`. We use the standard Java properties file conventions to define each database property.



If you are using the H2 database implementation in a standalone installation, you can disregard setting properties. The default implementation works as is and requires no configuration.

If you are using the H2 database implementation in a cluster, you must configure it.

- In a cluster installation, the file `database.properties` is stored in the `conf` directory under `SPSERVER_SHARE`. When you install a new node in a cluster, provide this location so that all the nodes in the cluster can find the centralized configuration.
- In a standalone installation, the file is stored in the `conf` directory under `SPSERVER_HOME`.

In a default installation, the full path to the `database.properties` file is:

Spotfire Statistics Services installed on Windows	C:\Program Files\TIBCO\statsvcs711\<service_name>\conf
Spotfire Statistics Services installed on UNIX/Linux	/opt/TIBCO/statsvcs711/conf

You must restart the Spotfire Statistics Services service to include any changes that you make in the `database.properties` file.



When you restart the Spotfire Statistics Services service, the server resumes processing queued jobs. However, because restarting the Spotfire Statistics Services service also stops the pool of S engines, any job that has not completed is marked as failed. To prevent this, we recommend that you wait until all jobs have completed before you restart the service. Alternatively, you can resubmit failed jobs after restarting the service.

To monitor job status, use the `jobs` function. For more information about the `jobs` function, see the URL API documentation, available from the Spotfire Statistics Services landing page.

database.maxActive

This property specifies the number of active connections to a Spotfire® Statistics Services job queue.

Property Name	Default Value	Property Description
<code>database.maxActive</code>	1	Set the maximum number of active connections that can be allocated at the same time. The default is 1. You can set to a negative value to specify no limit.

database.maxIdle

This property indicates the maximum number of connections to the Spotfire® Statistics Services that can remain idle in the pool without extra connections being destroyed.

Property Name	Default Value	Property Description
database.maxIdle	1	<p>The maximum number of connections that can remain idle in the pool, without extra ones being destroyed.</p> <p>If database.maxIdle is set too low on heavily-loaded systems, you might see connections being closed and new connections opened almost immediately. This is a result of the active threads momentarily closing connections faster than connections are opened, causing the number of idle connections to rise above database.maxIdle. For heavily-loaded systems, the best value for database.maxIdle can vary, but the default is a good starting point.</p> <p>The default is 1. You can set to a negative value to specify no limit.</p>

database.type

This property indicates the type of database you intend to use for managing the Spotfire® Statistics Services job queue.

Property Name	Default Value	Property Description
database.type	H2	<p>Set the database server for managing the Spotfire® Statistics Services job queue. The default, an in-process H2DB database (H2) requires no configuration for a stand-alone installation. (If you have a cluster, it requires configuration.) Other database options that you can configure to manage the job queue include:</p> <ul style="list-style-type: none"> • Oracle (ORACLE) • Microsoft SQL (MSSQL) • MySQL (MYSQL)

database.url

This property specifies the JDBC connection string for accessing the Spotfire® Statistics Services job database.

Property Name	Default Value	Property Description
database.url	Can be blank for a standalone only. Upon first server start, the value is populated with the path to the embedded database. See Details for cluster default.	Contains a JDBC connection string for accessing the Spotfire Statistics Services job queue database. The string you enter depends on the value you set in the database.type property.

Details

If you use the default H2 as the database type in a cluster, the installation prepopulates with the following pattern to build the JDBC connection string for your environment:

```
jdbc:h2:tcp://<manager_node>:<dbport>/<spserver.cluster.share>/h2/db
```

where

- `<manager_node>` is the host name of the primary manager node running the database instance (that is, the manager node to run the database).
- `<dbport>` is the port number to use (by default, 9092 but you can specify any open port)



The port you specify is the actual port used by the server, so be sure it is available.

- `<spserver.cluster.share>` is the location of the `SPSERVER_SHARE` for the specified manager node.
 - On Microsoft Windows®, for H2, this share requires 4 preceding backslashes, as follows:
`\\\\\\hostname/path/to/share/h2/db`
 - On UNIX/Linux, for H2, this share appears as follows:
`/path/to/share`
- The subdirectory `h2` is created by the installer.
- `db` is used to create the first part of the filenames of the database files.

Examples

- H2 database in a Windows cluster:
`database.url=jdbc:h2:tcp://myserver:9999/\\\\\\hostname/path/to/share/h2/db`
- H2 in a UNIX/Linux cluster, set up as nfs share:
`database.url=jdbc:h2:tcp://path/to/share/h2/db`
- Oracle database pattern:
`database.url=jdbc:oracle:thin:@//
<db_servername>:<db_serverport>:<SID>`
- MSSQL database pattern:
`database.url=jdbc:sqlserver://
<db_servername>:<db_serverport>;databaseName=<database_name>`
- MYSQL database pattern:
`database.url=jdbc:mysql://<db_servername>:<db_serverport>/
<database_name>`

For more information on JDBC connection strings, refer to the documentation for your database server.

database.username

This property specifies the user who has all privileges to the job queue database.

Property Name	Default Value	Property Description
<code>database.username</code>	sa	This property specifies the user with GRANT ALL privileges on the job queue database. Its setting depends on <code>database.type</code> . Do not modify the default value if you use <code>database.type=H2</code> .

database.password

This property specifies the password for the user specified by `database.username`.

Property Name	Default Value	Property Description
<code>database.password</code>	<code>3DES{encrypted_password}</code>	Sets the password for the user identified by <code>database.username</code> . Its value depends on the <code>database.type</code> .

Details

The password is encrypted using 3DES encryption. You do not need to encrypt the passwords manually before placing them in the configuration file. When the server is started, any unencrypted password is automatically replaced with an encrypted password and written back to the configuration file.

If you need to change the password, just replace the existing value `3DES{encrypted_password}` with the new, unencrypted password.

If you use the default `database.type=H2`, but in cluster mode, a value is required. The cluster must be down when you change the password. For H2, defining this value actually sets the password.

Service administration

After the installing and configuring Spotfire® Statistics Services, and the service is up and running, you might be called on to perform certain standard or specialized administration tasks. needs occasional administration to keep it running without problems.

Standard administration tasks include the following.

- Stopping and restarting the service
- Uninstalling the server.
- Deleting protected packages.
- Apply special configuration for database connections.
- Troubleshoot the service or the engine.

Stopping the service on Microsoft Windows

Occasionally, you must stop Spotfire Statistics Services.

The display name for the service is Spotfire Statistics Services(service_name) where service_name is the value you specified during installation. The default is **SplusServer**.

Procedure

1. Click **Start > Control Panel > Administrative Tools**, and then select **Services**
2. In the **Service Manager** console, right-click Spotfire Statistics Services, and then click **Stop**.

Result

The service is stopped for maintenance, upgrade, or other necessary tasks.

What to do next

Remember to restart the service or, if you restart the server, ensure that it restarts automatically.

Starting the service on Microsoft Windows

Occasionally, you must start Spotfire Statistics Services, although by default the service starts automatically.

The display name for the service is TIBCO Spotfire Statistics Services(service_name) where service_name is the value you specified during installation. The default is **SplusServer**.

Procedure

1. Click **Start > Control Panel > Administrative Tools**, and then select **Services**.
2. In the **Service Manager** console, right-click TIBCO Spotfire Statistics Services, and then click **Start**.

Result

The service is running and ready to be used.



If you encounter issues, see the service dedicated log file, `SplusServer.log`, located in `SPSERVER_HOME/tomcat/logs` (for standalone installations) or `SPSERVER_HOME/worker/logs` (for Worker nodes)

Uninstalling the server from Windows

Follow the steps to remove Spotfire Statistics Services from your computer.

Prerequisites

Back up the `SPSERVER_HOME/data` directory.

Procedure

1. Click **Start > Control Panel**, and then double-click **Add or Remove Programs**.
2. In **Currently installed programs**, select **TIBCO Spotfire Statistics Services(service_name)**, and then click **Remove**.

If you are prompted to confirm the removal of the program, click **Yes**.

Configuring the service to run at start up on UNIX or Linux

You can configure the Spotfire Statistics Services installation to run when the service starts by registering the init scripts with system init.

Procedure

1. Log in as root.
2. Type the following command:

```
SPSERVER_HOME/init.d/register-service.sh
```

When the machine starts, the service will run.

Starting the service on UNIX or Linux

Occasionally, you must start Spotfire Statistics Services on your UNIX or Linux machine.



The user who performs this task must own the entire Spotfire Statistics Services installation. When the system is ready, issue the following commands in a terminal with the appropriate value for `SPSERVER_HOME`

Procedure

1. Open a terminal and log in with the user account that will be used to run Spotfire Statistics Services.
2. When the system is ready, issue the following commands in a terminal with the appropriate value for `SPSERVER_HOME`:

```
SPSERVER_HOME=/opt/TIBCO/statsvcs711/<server_name>
$SPSERVER_HOME/init.d/spserver start
```

The service should start and be ready to use.

Stopping the service on UNIX or Linux

Occasionally, you must stop Spotfire Statistics Services on your UNIX or Linux machine.



The user who performs this task must own the entire Spotfire Statistics Services installation. When the system is ready, issue the following commands in a terminal with the appropriate value for `SPSERVER_HOME`

Procedure

1. Open a terminal and log in with the user account that will be used to run Spotfire Statistics Services
2. When the system is ready, issue the following commands in a terminal with the appropriate value for *SPSERVER_HOME*:

```
SPSERVER_HOME=/opt/TIBCO/statsvcs711/<server_name>
$SPSERVER_HOME/init.d/spserver stop
```

The service is stopped for maintenance, upgrade, or other necessary tasks.

What to do next

Remember to restart the service or, if you restart the server, ensure that it is configured to restart automatically.

Restarting the service on UNIX or Linux

If the Spotfire Statistics Services service is stopped, you must restart it manually, unless you have configured it to start automatically after restarting the machine.



The user who performs this task must own the entire Spotfire Statistics Services installation. When the system is ready, issue the following commands in a terminal with the appropriate value for *SPSERVER_HOME*

Procedure

1. Open a terminal and log in with the user account that will be used to run Spotfire Statistics Services.
2. When the system is ready, issue the following commands in a terminal with the appropriate value for *SPSERVER_HOME*:

```
SPSERVER_HOME=/opt/TIBCO/statsvcs711/<server_name>
$SPSERVER_HOME/init.d/spserver restart
```

The service should start and be ready to use.



If you encounter issues, see the service dedicated log file, *SplusServer.log*, located in *SPSERVER_HOME/tomcat/logs* (for standalone installations) or *SPSERVER_HOME/worker/logs* (for Worker nodes).

Uninstalling the server from UNIX or Linux

Follow the steps to remove Spotfire Statistics Services from your computer.

Procedure

1. If you registered the service to start at boot, you should unregister the service by running the following command as root:
`SPSERVER_HOME/init.d/unregister-service.sh`
2. Run `SPSERVER_HOME/Uninstall_SplusServer/Uninstall_SplusServer`
3. Manually remove the installation directory.

Deleting protected packages

Follow these rules to delete a protected package from Spotfire Statistics Services.

Procedure

1. From your Java Development Kit installation, start the Java Monitoring & Management Console by executing `jconsole`.
The JConsole application opens.
2. In the **JConsole: New Connection** dialog box, select **Remote Process**, and type your server name and the **JMX port**. For example, `servername:9004`, and then click **Connect**.
You are connected to the server via this remote process.
3. In the **JConsole**, click the **MBeans** tab.
The **MBeans** tab is available.
4. Expand **tibco**, **packageRepository**, **Operations**, and then select **deleteProtectedPackage**.
The **Operation invocation** area is available.
5. In the **Operation invocation** area, type the name of the protected package, and then click **deleteProtectedPackage**.
The protected package is deleted.



If you do not have access to the JConsole application included in JDK, it is possible—but not recommended—to manually delete the file from the `packages/Protected` directory. If you use this method, after you remove the package you should restart the service for all of the nodes in the cluster.

Monitor the service

Discover the tools you can use to monitor Spotfire Statistics Services.

JMX

The default installation of Spotfire Statistics Services enables remote monitoring and administration via JMX on port 9004. If you use a JMX-compliant monitoring tool such as TIBCO Hawk or JConsole, you can monitor the server's health and status, edit the configuration, and manage the job queue of Spotfire Statistics Services, all remotely.

Procrun

On Windows, we use the Procrun service and monitor applications for all node types.



Procrun is Windows only. To monitor or modify settings on UNIX/Linux, you must edit `SPSERVER_HOME/init.d/spserver`.

We use the service application, `Prunsvr`, to automate updating and modifying the service configuration. For example, we automate setting the service to run as a specified user and changing a Java property through this service application. You can start the monitor application, `Prunmgr`, to change settings interactively.

If you decide that you want to automate a change, you can use the service application to automate, at the command line, any change that you can make through the monitor application.

The Procrun service application is `TSSS<servername>.exe` and the monitor application is `TSSS<service_name>w.exe`, where `<service_name>` is the service name that you specify during installation. For example, the default service name is `SplusServer` and yields `TSSSSplusServer.exe` and `TSSSSplusServerw.exe`.

The respective locations are:

- Standalone/Manager: `SPSERVER_HOME\tomcat\bin`
- Worker: `SPSERVER_HOME\worker\bin`

For more information, see [Using Procrun to modify a Java property](#). For more general information about Procrun, see <http://commons.apache.org/daemon/procrun.html>.

Using Procrun to modify a Java property

On Windows, for managing Spotfire Statistics Services, you can add or modify a Java property using Procrun.

Procedure

1. Open `TSSS<service_name>.exe`, where `service_name` is the value that you specified during installation.
2. Select the **Java** tab.
3. Under **Java Options**, add the specific Java property. Examples of properties and values that you could set include:

- `-Dengine.count=4`
- `-Dspserver.rhome=/path/to/r`



See [Configuring an open-source R engine](#) for more detailed information about setting the path to an open-source R engine.

Troubleshoot the service

When you encounter specific problems with Spotfire Statistics Services, you can use support and the issues listed in the release notes to help you find the answers.

If you encounter problems with Spotfire Statistics Services, register for a support account at <http://spotfire.tibco.com/support>.

The Release Notes (`relnotes.pdf`), available from both the server landing page and on the installation media, contain a list of known issues in the version at the time of release.

MySQL dropped connections

This known issue is a characteristic of MySQL when it is left idle for a given period of time.

If you use MySQL as the job database you may experience dropped connections if Spotfire Statistics Services is idle for more than eight hours. As a result, the first request after the time-out occurs will fail. However, the failed request automatically re-establishes the connection and subsequent requests succeed.

To work around this issue, we recommend that you change the MySQL server `wait_timeout` environment variable. As an alternative, you can edit the `spserver.properties` file to modify the `database.url` value by appending `?autoReconnect=true` to the server URL. For more information on the `wait_timeout` or `autoReconnect` properties, see your MySQL Reference Manual.

Troubleshoot R engine failures

When you encounter problems with jobs running, and you are using the open-source engine, you should check to make sure that the `rJava` package is installed and working correctly.



Packages are not cross-platform compatible. The package developer must build the package specifically for each platform. In the case of open-source R binary packages, the developer must use an open-source R engine configured with the option `--enable-R-shlib` enabled.



Open-source R is available under separate open source software license terms and is not part of TERR. As such, open-source R is not within the scope of your license for TERR. Open-source R is not supported, maintained, or warranted in any way by TIBCO Software Inc. Download and use of open-source R is solely at your own discretion and subject to the free open source license terms applicable to open-source R.

If your open-source R jobs fail, and your log or error messages indicate that it cannot find the rJava package, you can check the following conditions:

On Windows

Make sure the rJava package is not installed in the `User` directory. When you call `install.packages`, you must provide the installation path, as recommended in [Configuring an open-source R engine](#).

In open-source R, you can run the following commands to discover if rJava is installed, where it is installed, and if it is installed correctly:

```
.find.package("rJava")
system.file("jri", package="rJava")
```

On UNIX/LINUX

When you try to install rJava using `install.packages()`, you might get the following error if your Java environment for R is not properly configured:

```
md64 -L/usr/local/lib64 -L/usr/java/packages/lib/amd64 -L/usr/lib64 -L/lib64 -L/lib
-L/usr/lib -ljvm'
```

```
configure: error: Java Development Kit (JDK) is missing or not registered in R
```

Make sure R is configured with full Java support (including the JDK). run the following command as root to add Java support to R:

```
R CMD javareconf
```

If you do not have root privileges, run the following command to set all Java-related variables, and then install rJava.

```
R CMD javareconf.
```

If you see the following error

```
ERROR: configuration failed for package "rJava"
```

```
* removing "/usr/local/lib64/R/library/rJava"
```

```
"/tmp/RtmpkNW5/downloaded_packages"
```

```
Updating HTML index of packages in '.Library'
```

```
Warning message:
```

```
In install.packages("rJava") :
```

```
installation of package 'rJava' had non-zero exit status
```

run the following command:

```
env JAVA_HOME=/usr/java/jdk1.8.0_161 ./R CMD javareconf
```

Where `JAVA_HOME` is set to the path to the installed JDK.

Index

A

access 39
 activeDirectory.enabled 65, 66
 AJP 43–45
 Apache Tomcat 15, 27, 37, 38
 API 18
 Aster 86
 AsterDB 86
 asynchronous 17
 authentication 65
 authentication.required 53, 65
 automation 46
 autoReconnect 96

B

bin 50
 binaries 49

C

C# 18
 catalina.policy 50
 catalina.properties 50
 certificate 74
 cluster 32–35, 39–41, 43–45
 Cluster agent 21
 Cluster management 21
 cluster.jms.port 53, 60
 cluster.multicast.group.address 53, 54, 60
 cluster.multicast.port 53, 54, 60
 cluster.properties 27, 37, 38
 common 49
 configuration 39
 context.xml 50
 CsharpAPI 18

D

database 39, 85
 database configuration 86
 database.maxActive 88
 database.maxIdle 89
 database.password 91
 database.properties 50, 87
 database.type 89–91
 database.url 89, 96
 database.username 90
 db.cleanup.run.interval.minutes 54
 db.cleanup.time.to.live.days 55
 DEBUG 50
 deleting 95

E

email.from 55
 email.host 55
 email.reply.to 55
 engine count 29, 30, 40
 engine type 39
 engine.appdata.dir 55
 engine.count 41
 engine.init.expr 56
 engine.init.file 20, 56
 engine.java.options 56
 engine.log 51, 52
 engine.max.elapsed.seconds 57
 engine.max.requests 57, 58
 engine.type 58, 80–82
 engine.verbose.log 58
 ERROR 50
 expression.service.enabled 58

F

failover 32–34
 FATAL 50
 feature overview 14
 function.service.whitelist 59

G

Group Search Filter 69

H

H2 85, 89
 H2 database 39
 HTTP request handler 21
 Http11NioProtocol 78

I

INFO 50
 initialization 39
 installation 26, 30, 34, 49
 installing 36
 isclustered 60

J

Java 18
 JavaAPI 18
 JConsole 95
 JDBC 85, 87
 JDK 96
 JMX 76, 77, 95
 jmx.remote.enabled 61

jmxremote.password 27, 37, 38, 61
 jmxremote.port 27, 37, 38, 61, 77
 jmxremote.username 27, 37, 38
 job.lost.xslt.enabled 61
 jobs 61
 JRE 15, 27, 37, 38

L

ldap 67, 68
 ldap.accountNameAttr 68
 ldap.basedn 68
 ldap.customSearchFilter 67, 68
 ldap.domain 66
 ldap.enabled 65, 66, 69
 ldap.group 67
 ldap.group# 70
 ldap.groupMemberAttr 69
 ldap.groupRoleAttr 69
 ldap.host 65, 67, 70
 ldap.manager.anonymous 71
 ldap.manager.cn 71
 ldap.manager.pass 71, 72
 ldap.path 67, 72
 ldap.port 72
 ldap.properties 50, 65, 66
 ldap.protocol 73
 ldap.searchBase 68, 73
 ldap.searchSubTree 73
 LDAPS 74
 Linux 26, 30
 load balancing 41, 43–45
 local security rights 39
 local system 24
 log4j 51, 52
 log4j levels 50
 log4j.properties 50
 logging 51, 52
 logging.properties 50

M

manager 35
 manager node 39
 Manager node 17, 21
 MATLAB 14, 15, 81
 max.jobx.to.return 61
 mod_jk 43–45
 monitoring 77, 95
 MSSQL 89
 Multicast 53
 MYSQL 89, 96

N

node 34, 35
 nodes 41, 43

notification.delay.time 62
 notification.max.times 62

O

Open-source R 28
 Oracle 89

P

package resources 20
 packages 20, 49, 95
 permissions 24
 ports 24
 procrun 95, 96
 properties 49, 52
 Prunsrv 95

R

R engine 96
 R package 19
 requests 49
 response 46
 response file 46
 REST API 14
 restarting 94
 RESTful 18
 results 49
 RinR package 15
 rJava 96
 ROLLFILE 50

S

S_USER_APPDATA_DIR 55
 sAMAccountName 68, 69
 SAS 14, 15, 80
 security 39
 server monitoring 76
 server.properties 27, 37, 38
 server.xml 50, 78
 service registration 39
 service_name 24
 service.id 27, 37, 38, 62
 service.url 27, 37, 38, 62
 sessions 49
 SplusServer.log 50, 92, 94
 Spotfire 82
 spserver 49
 spserver.cluster.share 89
 spserver.home 75
 spserver.properties 32, 33, 50, 52, 65, 77, 80, 96
 spserver.results.file 49
 spserver.share 75
 spserver.terhome 75
 SSL 74
 standalone 30, 49

- standalone installation 24
- starting 92
- startup 93
- stopping 92, 93
- synchronous 17

T

- Teradata 86
- terr.restricted.execution.mode 63
- TERREngine.tar.gz 28
- TERREngine.zip 28
- testing 30
- TIBCO Enterprise Runtime for R 28
- tomcat-users.xml 50
- troubleshooting 96
- troubleshooting 96

U

- uid 68
- uninstalling 93, 94
- URLAPI 18
- users 49
- users.properties 50, 65

V

- validating 43
- validation 30, 41

W

- wait_timeout 96
- WARN 50
- web.xml 50
- WebDAV 17, 76
- WebDAV service 21
- webdav.cleanup.run.interval.minutes 64
- webdav.cleanup.time.to.live.minutes 64
- webdav.security.config 76
- Windows 29
- Worker node 17, 21
- worker.properties 45
- workers.properties 44

X

- xml.version 65