STIBO SYSTEMS
MASTER DATA MANAGEMENT

SYSTEM ADMINISTRATION GUIDE

2024.1 – March 2024

Table of Contents

System Administration

There are many administrative tasks that are needed to keep a STEP system running at optimum levels. This section of the documentation introduces those responsible to the common tasks, challenges, and issues that often occur when working with administration of a complete STEP system and infrastructure.

Some of the information presented may not be applicable to all systems. Any questions should be directed to your Stibo Systems' representative or the Stibo Systems Technical Support department.

Important: The majority of the documentation in this guide is for on-premise systems. SaaS information can be found within the self-service documentation provided with SaaS instances.

The System Administration section includes the main topics below:

- Infrastructure Requirements
- Installing STEP
- Linux Administration
- Migration
- Optimizing Performance
- Patching STEP
- STEP Architecture
- Externally Used Open Source Components

Infrastructure Requirements

Meeting the following infrastructure requirements allows you to ensure a successful STEP setup.

- Platform and Software Support is addressed for each release. For example, refer to the **11.2 Platform and Software Support** topic within the **System Update and Patch Notes / Platform and Software Support** documentation.
- Virtualization and Cloud Support
- Network Requirements
- Storage Performance Requirements
- High Availability Considerations

Virtualization and Cloud Support

This is one of the infrastructure requirements for a successful STEP setup. The full list is defined in the **Infrastructure Requirements** topic.

Supporting STEP via virtualization or cloud involves separate requirements. For details, refer to the following topics:

- Virtualization Support
- Cloud Support

Virtualization Support

Virtualization of STEP has officially been released on the following platforms:

- VMware ESXi 5+ for the following guest OS platforms:
  - Red Hat Enterprise Linux 8.6 / 7.9 x86-64
  - Oracle Enterprise Linux 8.6 / 7.5 x86-64

The only supported virtualization environments are VMware and Hyper-V on x86-64 based platforms.

Scope and Restrictions

Consider the following restrictions regarding virtualization:

- Virtualization of the STEP application uses a supported OS platform, as listed above.
- Virtualization of the STEP Oracle database is supported in non-production environments on all released OS platforms, as indicated above.
- Virtualization of the production database must be evaluated on a case-by-case basis, based on system load and sizing expectations. It is conditional upon a performance test conducted jointly by the parties and finally approved by Stibo Systems.
- Virtualization of a DTP server is supported only on the Windows Server 2019 / 2016 OS platform.

Licensing and Support Using VMware

Regarding licensing of the Oracle Database on the VMware virtualization platform, the VMware host environment must have no license violations. This is independent of the supported OS being used. Oracle does not license at the OS level.

VMware licensing and 'expanded support' when virtualizing the Oracle Database on their platform is described in the document **Understanding Oracle Certification, Support and Licensing for VMware Environments** on the web at http://www.vmware.com/files/pdf/solutions/oracle/Understanding_Oracle_Certification_Support_Licensing_VMware_environments.pdf.

Requirements

The following requirements must be met by the system and infrastructure when considering virtualization using VMware or Hyper-V:

- Hypervisor must be VMware ESXi 5+ or Microsoft Hyper-V 2019 / 2016.
- VMware or Hyper-V servers must be guaranteed to provide a constant capacity in terms of RAM, CPU power and I/O that is equivalent to that of a physical system.
- STEP version must be 7.0 or newer.

- Stibo Standalone application server is based on AdoptOpenJDK.
- OpenJDK 11 (64-bit)
- Host processor must match (or be similar to) what is recommended in this document. Processors like the Intel low-power CPUs are not supported for production environments.
- Storage and network requirements must be met as described in this document.

If a performance problem occurs and Stibo Systems and the client come to the conclusion that it is caused by the virtualization of the Oracle Database, the client is responsible for resolving the problem. Stibo Systems will assist where possible.

Recommendations for Best VMware Performance

It is critical that VMware is configured correctly for maximum performance.

This is done by following the recommendations in the white paper **Performance Best Practices for VMware vSphere**. Especially the sections on Host Power Management in ESXi and Running Network Latency Sensitive Applications.

This includes:

- Configuring ESXi policy to use 'High Performance' power option
- Reserving CPU and memory resources

The **Performance Best Practices for VMware vSphere** are online, based on versions, as follows:

- v5.5: https://www.vmware.com/pdf/Perf_Best_Practices_vSphere5.5.pdf
- v6.0: https://www.vmware.com/content/dam/digitalmarketing/vmware/en/pdf/techpaper/vmware-perfbest-practices-vsphere6-0-white-paper.pdf
- v6.5: https://www.vmware.com/content/dam/digitalmarketing/vmware/en/pdf/techpaper/performance/Perf_Best_Practices_vSphere65.pdf

Cloud Support

STEP is supported in the following major cloud providers:

- **Amazon Web Services** – AWS
- **Microsoft Azure Cloud** – When deploying STEP in Azure it must be observed that the Azure feature Accelerated Networking is a requirement for servers being part of a STEP environment.
- **Oracle Cloud Infrastructure - OCI** – The DBaaS on VM shapes has not been certified.
- **Oracle Cloud Infrastructure Classic - OCI Classic** – The Database Cloud Service (DBCS) on SSD storage has not been certified.

All requirements (network and storage) must be fulfilled as described and in the Infrastructure Recommendation being generated. These requirements are the same regardless of how and where the environment is hosted, and whether it is a physical or a virtual environment.

Hybrid Setup

A hybrid setup, where some servers are placed on the premises and others in the cloud, is not supported. Such a setup does not fulfill the performance requirements, especially related to network performance.

Oracle Licensing in AWS and Azure

When licensing Oracle in the cloud, refer to the **Licensing Oracle Software in the Cloud Computing Environment** paper published by Oracle, which describes how licensing works. This paper is available online at http://www.oracle.com/us/corporate/pricing/cloud-licensing-070579.pdf.

Oracle updated this document on January 23, 2018, indicating that within AWS (EC2 and RDS) and Azure, Oracle Standard Edition 2 may only be licensed on instances up to 8 vCPUs. If licensing Database Standard Edition 2 by Named User Plus metric, the minimums are 10 NUP licenses per 8 Amazon vCPUs or 8 Azure vCPUs. Contact Stibo Systems for the latest update on how AWS and Azure handles this.

Network Requirements

This is one of the infrastructure requirements for a successful STEP setup. The full list is defined in the **Infrastructure Requirements** topic.

Network requirements include the following:

- Server-to-Server Connections
- Client-to-Server Connections
- Server-to-External Server Network Requirements
- SSL Certificates

Server-to-Server Connections

This is one of the network requirements for a successful STEP setup. The full list is defined in the **Network Requirements** topic.

All connectivity between servers within each environment must be direct connections. This can be achieved either through dedicated NICs or through point-to-point switches without the interference from any additional hardware and/or software.

The introduction of load balancing or firewall hardware and/or software known to cause the following issues due to limitations and malfunctions in the introduced hardware and/or software:

- degradation of performance
- server instability
- environment instability

Servers must have permanent IP addresses assigned, either by manually configuring network settings on the servers, or by dedicating a specific IP address for each server on the DHCP server. Server IP addresses must not change while the application is in operation. This is particularly important for systems with multiple application and/or database servers.

All application servers in a STEP application cluster must be located on the same network segment / VLAN, so a firewall may never be located between application servers.

It is not uncommon that application and database are placed on different networks, and thus separated by a firewall. This is accepted by Stibo Systems as long as the network latency requirements mentioned below are still met.

Network Bandwidth Requirement

Minimum requirement is 1 Gbit dedicated network between all servers, except to file servers. For enterprise systems with a heavy workload profile, a 10 Gbit dedicated network is recommended.

Important: Using the STEP In-Memory component requires a 10 Gbit network between application servers and database server.

Network Latency Requirements

The requirements for network latency between application servers in a STEP application cluster and between application server(s) and database server(s) are:

- Maximum 0.2 ms with a 64 bytes packet size
- Maximum 0.4 ms with a 8K packet size

- Maximum 1.-0 ms with a 32K packet size

For more information, refer to the **Network Latency Measurements and Tools** section of the **Infrastructure Checks** topic.

Server-to-Server Network Requirements

The following table summarizes the inter-server communication requirements:

Source	Target	Protocol	Port	Comment
Application Server	Application Server	Cache coherency	5636	Used for internal STEP application clustering cache. Only relevant when having more than one application server.
Application Server	Application Server	In-Memory	5999	STEP In-Memory data communication / exchange. Only relevant when there is more than one application server and using STEP In-Memory.
Application Server(s)	Database Server	SQL Net	1521	Communication between application and database. 1521 is the default port, so it could be different on any given installation.

Client-to-Server Connections

This is one of the network requirements for a successful STEP setup. The full list is defined in the **Network Requirements** topic.

The following table summarizes the network requirements between the STEP client and the STEP servers per concurrent user (refer to the **Concurrent User** section below):

Application	Bandwidth	Recommended Max. Latency	Protocol	Port	Encryption Option
STEP Workbench	20 MBit/s *(1)*	25 ms	HTTP	80	HTTPS (Port 443)
STEP Web UI	20 MBit/s *(1)*	25 ms	HTTP	80	HTTPS (Port 443)
Print Publisher (STEP'n'design) Client	10 MBit/s *(2)*	25 ms	HTTP	80	HTTPS (Port 443)
Stibo Systems Support Client	20 MBit/s	25 ms	HTTP RDP SSH VNC SQL	80 3389 22 5900-5909 1521	HTTPS (Port 443)
Surveillance Agent	~30 MBit/s *(3)*	25 ms	TCP/IP	5666	SSH (Port 22)

(1) May be higher for clients that are uploading images. Any increase of the bandwidth requirement depends on the number and size of the images to be uploaded. The overall upload frequency with which images are uploaded contributes to the requirement as well.

(2) Data is transmitted only when page data is downloaded and saved back. Image data will be downloaded from a local image repository (a 1GBit LAN connection is recommended).

(3) The surveillance agent will be running with only one instance per server and the required bandwidth depends on the individual surveillance agent software. Stibo Systems uses Nagios for server surveillance. For information, refer to http://www.nagios.org.

Concurrent User

With the knowledge of how the client to server connection works, Stibo Systems defines the term 'concurrent user' as a user that is actively working in STEP by uploading and/or requesting data. This term is important when scaling or predicting network bandwidth for the network infrastructure. Only concurrent users contribute to the overall network load.

For example, the following behaviors constitute a concurrent user:

- Select a product
- Wait 15 sec
- Select another product
- Wait 15 sec
- Select an attribute
- Wait 15 sec
- Change attribute data and save changes
- Wait 15 sec
- …

STEP Workbench and Web UI

The STEP client only opens a connection to the STEP server when the user performs actions on the client. This can be by clicking buttons, browsing the product hierarchy, uploading data, etc. Clients that are only logged on to STEP, but otherwise are idle, do not transmit or receive any data (so there is no network usage).

A typical workbench user is conducting data maintenance tasks, enriching data, and performing import and exports based on the STEP import and export managers, as well as uploading and downloading images.

A typical Web UI user is conducting data maintenance tasks and enriching data as well as uploading and downloading images.

Print Publisher (STEP’n’design) Client (DTP) Client

This user manually builds pages in Adobe® InDesign® and through the use of Print Flatplanner. Print Publisher (also known as STEP’n’design) is the STEP integration with Adobe® InDesign® and allows users to mount database information to an Adobe® InDesign® page.

The DTP client only exchanges data with the STEP system when:

- Loading page data
- Saving page data back to STEP

Important: Image data loaded from a local image repository vastly improves performance for remote users and reduces network usage.

Stibo Systems Support Client

Stibo Systems must be able to access the STEP server(s) hosted by the customer in order to provide support. When publishing is included in the solution, the customer must provide an InDesign client license that is available to the Stibo Systems support / consulting teams.

Surveillance Agent

A surveillance agent will only be installed on customer servers when a DBA agreement is signed between the customer and Stibo Systems.

Server-to-External Server Network Requirements

This is one of the network requirements for a successful STEP setup. The full list is defined in the **Network Requirements** topic.

This table summarizes the network between the STEP server and external servers for additional functionalities:

STEP Server	External Server	Protocol	Port	Comment
Application Server	Mail Server	SMTP	25	Used to send mail from the STEP system.
All STEP Servers	Time Server	UDP NTP	123	Access to time synchronization service.
Application Server	SFTP Server	SFTP	22	Import / Export of STEP and patch data.
Application Server	Stibo Systems Update Server	HTTPS	443	Update service for STEP software: https://updates.stibosystems.com.
Application Server	LDAP Server	LDAP LDAPS	389 636	External authentication using LDAP.
Asset Push File Server	Remote File Servers	RSYNC	873	Image replication / copying to remote locations.
All STEP Servers	Backup Server	?	?	Must be provided by the customer in order to perform system backup.
BeyondTrust Jumpoint proxy	BeyondTrust Appliance (Stibo)	HTTPS	443	BeyondTrust (formerly Bomgar) remote support access must be opened to the following addresses: • 217.28.168.103 • 217.28.163.56
All STEP servers	BeyondTrust Jumpoint proxy	HTTPS	443	Application and database servers uses the BeyondTrust Jumpoint proxy to connect to the BeyondTrust appliance at Stibo Systems.

SSL Certificates

This is one of the network requirements for a successful STEP setup. The full list is defined in the **Network Requirements** topic.

When configuring a STEP solution to use SSL encryption (typically done for external supplier access), it is important to only use certificates from a trusted CA authority, such as Verisign or Thawte.

Self-signed certificates are not supported by the STEP solution.

It is the responsibility of the customer / hosting partner to provide the required SSL certificates.

Storage Performance Requirements

This is one of the technical infrastructure data gathering methodologies and performance recommendations for server machines. The full list is defined in the **Server Environment Recommendations** topic.

Performance of the storage used for the database has been measured when testing scalability of the STEP solution. The I/O performance has been measured as IOPS.

The requirements for the database are:

- up to 2,500 IOPS during normal operations
- up to 5,000 IOPS during peak operations

The STEP Oracle Database uses a block size of 8 KB. The read / write ratio is typically between 60/40 and 70/30.

The application server (including shared storage) requires 500 – 1,000 IOPS.

For more information, refer to the **CPU and Storage** section of the **Infrastructure Checks** topic.

AWS EBS Storage Considerations

The following elements should be considered:

- EBS bandwidth — Depending on the chosen instance type, the EBS storage bandwidth varies. For example, the r4.4xlarge offers 3000 Mbit/s EBS storage bandwidth, whereas the r4.8xlarge offers 6000 Mbit/s.

  If EBS bandwidth becomes a bottleneck, it is necessary to switch to the larger instance type.

- EBS IOPS (Provisioned IOPS or General Purpose SSD) — Depending on the chosen SSD volume type, it is possible to control how many IOPS that are available, which is important to the database storage performance.
  - Provisioned IOPS SSD (io1) offers 50 IOPS/GB.
  - General Purpose SSD (gp2) offers 3 IOPS/GB and volumes can be striped to achieve higher IOPS than this.

Choose the solution that fulfills the storage performance requirements of 5.000+ IOPS at peak performance as described above.

Azure Storage Considerations

Performance testing of STEP in Azure shows that SSD storage is required for the Oracle database and for the shared storage covering workarea and upload for the application servers. In addition, SSD storage is preferred for the OS and software partitions of application servers as well as database servers. In Azure, SSD storage is referred to as Premium storage. Instance types in Azure that supports the use of SSD storage will have ‘s’ in its name like 'DSv3' or 'ESv3.'

In Azure, asset data can reside on non-SSD storage like Managed Disks or Unmanaged Disks, depending on if data must be highly available by managing replicas in other data centers or not.

High Availability Considerations

This is one of the infrastructure requirements for a successful STEP setup. The full list is defined in the **Infrastructure Requirements** topic.

To achieve system redundancy and an increased amount of system resources for large server solutions, STEP supports the following options for active / active and active / passive clustering.

Active / Active

This option requires both the database and application layer to become fully redundant with at least two (2) active nodes. This can be accomplished via the following ways:

Application

Two or more application servers are configured as an Application Cluster. The cluster is managed by the STEP application (no OS cluster) and will load balance user sessions and processes automatically. In the case of one of the nodes failing, user sessions are automatically transferred and background processes are automatically restarted on the healthy node.

Database

Database clustering is fully managed by Oracle. The active / active clustering option from Oracle is 'Oracle Real Application Cluster' (Oracle RAC).

Active / Passive

Since active / active clustering can be very complex to set up and manage, and has a very high price tag on the Oracle part, an alternative is an active / passive setup, featuring at least two nodes.

Application

An active / passive setup is not directly possible with the application. However, it is possible to setup multiple nodes and configure these for specific purposes. For example, on a three (3) node application setup, the nodes can be dedicated individually for background processes, internal Java client users, and external Web UI users. In case of a breakdown, the system can be reconfigured to use the remaining nodes.

Database

Active / passive clustering on the database is possible with different technologies and setups. The most commonly used is to set up an Oracle 'standby' database server that either shares the database storage with the active node or features its own storage. When using its own storage, the passive node is set up as a standalone

database server that must be updated at regular intervals or real-time from the active node. There are multiple ways to update the passive node. The most common way is to use Oracle Data Guard.

Infrastructure Checks

The following tools can be used to verify that infrastructure requirements are met.

Network Latency Measurements and Tools

Use the appropriate option below to measure latency and ensure the results meet the **Network Latency Requirements** defined in the **Server-to-Server Connections** topic.

Measure latency on Linux using ping

Run the following commands from application server to database server (results are in microseconds (ms)):

```
ping -c20 <db-host> |grep rtt
ping -c20 -s8000 <db-host> |grep rtt
ping -c20 -s32000 <db-host> |grep rtt
```

Measure latency on Windows using 'hrping'

Download the tool using SPOT:

```
spot --baseurl=updates:/install/software --download=hrping/windows/hrpingv506.
zip --output=L:/install-software
```

Run the following commands from the application server to database server (results are in microseconds (ms)):

```
hrping -l 64 -n 20 <database-host>
hrping -l 8000 -n 20 <database-host>
hrping -l 32000 -n 20 <database-host>
```

Measure latency using ‘benchnet5.sql’ script in Linux

Executed from the application server against the database, with different package-sizes, to measure any network influence (latency). Results are in seconds (sec).

1. From a Linux server, log on to the application server as 'stibosw'.
2. Copy the following SQL script using the file name 'benchnet5.sql' to `/home/stibosw` on the application server.

```
set timing on

set autotrace traceonly statistics
set arrays 5000
set pages 1000
select null from dual connect by rownum < 100000;
```

```
prompt Small without network influence
select null from dual connect by rownum < 100000;
set arrays 10
prompt Small with network influence
select null from dual connect by rownum < 100000;

set arrays 5000
prompt Medium without network influence
select
'01234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901'||rownum
from dual connect by rownum < 1000000;
set arrays 100
prompt Medium with network influence
select
'01234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901'||rownum
from dual connect by rownum < 1000000;

set arrays 5000
prompt Large without network influence
select
'01234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890
12345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901
34567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123
56789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789'||rownum from dual connect by rownum
< 1000000;
set arrays 100
prompt Large with network influence
select
'01234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890
12345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901
34567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123
56789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789'||rownum from dual connect by rownum
< 1000000;
```

3. Run the script on the application server:

```
sqlplus stepsys/stepsys
@/home/stibosw/benchnet5.sql
```

4. Collect all output and attach it to a Jira issue.

The following table shows the Stibo Systems recommended limits for network influence. The script outputs in msek, limits are displayed in sek (seconds).

Size	Including Network Influence	Maximum Recommended Limit
Small	No	00:00:00.15
Small	Yes	00:00:02.00
Medium	No	00:00:08.00
Medium	Yes	00:00:12.00
Large	No	00:00:24.00
Large	Yes	00:00:30.00

CPU and Storage

Use the appropriate option below to measure read, write, and CPU performance, and to ensure the results meet the requirements defined in the **Storage Performance Requirements** topic.

Measure read, write and CPU performance using 'stibo-ts-load-test.pl' script

On a Linux application server, run the following commands (script outputs in msek, limits are in sek (seconds)):

1. Log on to the database server as 'oracle'.
2. Locate db-server-toolbox, for example: `/opt/stibo/step/admin/db-server-toolbox/`
3. Run this command:

```
/opt/stibo/step/admin/db-server-toolbox/stibo-ts-load-test.pl -stepsys-schema
stepsys -stepsysdata-tbs stepsysdata
```

4. Run this command to report results:

```
/opt/stibo/step/admin/db-server-toolbox/stibo-ts-load-test.pl -stepsys-schema
stepsys -stepsysdata-tbs stepsysdata -report
```

Storage Limits

I/O for reads / writes	Classification
< 5 s	very good

I/O for reads / writes	Classification
< 10s	good
< 20s	moderate

CPU Limits

For CPU, Stibo Systems defines 60 sek (seconds) as the 'standard' level.

Measure IOPS on Storage Devices

Performance on storage devices can be evaluated using 'fio' with results in iops.

Windows

1. Download the tool using spot:

```
spot --baseurl=updates:/install/software --download=fio/windows/fio-2.2.10-x64.zip --output=L:/install-software
spot --baseurl=updates:/install/software --download=fio/windows/fio-examples.zip --output=L:/install-software
```

2. Configure fio by editing `settings.fio` for your server type.
3. Execute fio:

```
L:\install-software\fio\windows\fio-examples\windows\fio_test.bat
```

Linux

1. Download the tool using spot:

```
spot --baseurl=updates:/install/software --download=fio/linux/fio-2.1.7-1.el7.rf.x86_64.rpm --output=/workarea/install-software
spot --baseurl=updates:/install/software --download=fio/linux/fio-examples.zip --output=/workarea/install-software
```

2. Log in as a user with admin privilege (root) and install fio using this command:

```
yum install /workarea/install-software/fio/linux/fio-2.1.10-1.el7.rf.x86_64.rpm
```

3. Configure fio by editing `settings.fio` for your server type.
4. Execute fio:

```
/workarea/install-software/fio/linux/fio-examples/linux/fio_test.sh
```

Example

This example is the output from fio, executed with 8k block size:

```
trivial-readwrite-1g: (g=0): rw=randrw, bs=8K-8K/8K-8K/8K-8K,
ioengine=posixaio, iodepth=64
fio-2.1.10
Starting 1 thread
trivial-readwrite-1g: Laying out IO file(s) (1 file(s) / 1024MB)
Jobs: 1 (f=1): [m] [100.0% done] [12122KB/12241KB/0KB /s] [1515/1530/0
iops] [eta 00m:00s]
trivial-readwrite-1g: (groupid=0, jobs=1): err= 0: pid=55096: Tue Dec  6
12:37:25 2016
  read : io=525088KB, bw=11687KB/s, iops=1460, runt= 44928msec
```

Installing STEP

The installation of a STEP system is comprised of multiple components:

- An Oracle 19c RDBMS server software installation is recommended
- A STEP database
- An installation of the Java Runtime Environment that runs the application
- The latest STEP version

Important: The complete list of platform and software support is available in the **2023.3 (11.3) Platform and Software Support** section of the **System Update and Patch Notes** documentation.

To install a STEP system you must perform the following steps:

1. Review the system requirements. These requirements can be found in the **System Requirements** section of the documentation.
2. Run all required pre-installation tasks. These tasks are detailed in the **Pre-Installation Tasks** section of the documentation.
3. Run the installation script (install.pl).

Before proceeding with the installation, however, an installation scenario must be decided on that best fits the client's needs. The two primary questions that must be answered are: should you use a Single or a Clustered Application Server setup, and who needs to be involved in the setup?

Single vs. Clustered Application Server Setup

Determine whether the system being installed consists of one or more Application Servers. The installation must be performed on all servers in the setup. For more information, refer to the **Pre-Installation Tasks** section of the documentation.

Who Needs to be Involved in the Installation?

When running the installation script to install a STEP system (including all third-party software) a number of pre-installation steps are performed before actually installing the software itself.

The pre-installation steps require that the user running the installation scripts is a privileged user. This might be an issue on Linux where certain kernel parameters must be set, and OS groups and users must be created. On Windows, the STEP Installer must be a member of the Administrator group. For more information, refer to the **System Requirements** section of the documentation.

System Requirements

The system requirements for running STEP on a host machine largely depends on the operating system.

The supported operating systems include:

- Linux
- Windows
- MacOS

The complete list of platform and software support is available in the 2023.4 Platform and Software Support topic of the **System Update and Patch Notes** documentation.

Linux System Requirements

The supported operating systems are:

- Red Hat Enterprise Linux 8.9 (or higher **minor** release) 64-bit
- Oracle Enterprise Linux 8.9 (or higher **minor** release) 64-bit
- Red Hat Enterprise Linux 7.9 (or higher **minor** release) 64-bit
- Oracle Enterprise Linux 7.9 (or higher **minor** release) 64-bit

Unless explicitly mentioned, OS prerequisites are implemented by the installation scripts (when run as root or with sudo on Linux).

The install scripts will (if needed):

- Create OS user(s) and group
- Adapt kernel parameters
- Install required RPMs if the server has a connection to Red Hat Network (RHN)

Important: The Infrastructure Recommendations guide provides more detailed information regarding requirements.

User Accounts

The OS user accounts include:

User Account Name	Must Be Member of Group(s)	Description
oracle	dba	Account must be created on the database server and must use the Bash shell. The user is the owner of the Oracle database software. If the user and/or group does not exist, the installation scripts will automatically create them. Group ID (GID) default: 501

User Account Name	Must Be Member of Group(s)	Description
		User ID (UID) default: 501
stibosw	stibosw	Account must be created on the application server(s) and must use the Bash shell. The user is the owner of the STEP application software. If the user and/or group does not exist, the installation scripts will automatically create them. Group ID (GID) default: 500 User ID (UID) default: 500

Note: The installation scripts are implemented and tested using the Bash shell. Therefore, it is important that this is used if pre-creating the OS user accounts.

Server Setup

In a single server setup, with the database and application server running on one machine, both accounts must be created. This is often the case for test systems.

As indicated in the table above, there might be more than one application server, such as in the case of a clustered system setup. In these instances, the user account must exist on all application servers.

Service Accounts vs. Logon Accounts

If security policies require separation of service and logon accounts, two separate accounts must be created. The service account password may never expire and must not be changed.

User Account Names and IDs

User account names do not need to match what is shown in the table. If required, create account names that adhere to corporate standards.

Group and user IDs must adhere to corporate standards if the default values are not compliant. Properties for these include:

- user.db (username)
- user.db-id

- user.db-primary-group-id
- user.app (username)
- user.app-id
- user.app-primary-group-id

Note: User names are case sensitive.

Windows System Requirements

Important: Microsoft Windows-based servers are no longer an option for new customers and are now desupported for existing customers (as of the 11.0 release). Contact Stibo Systems to review your options for making the transition to a supported platform.

The following accounts are required for a complete STEP installation. The user accounts can either be local accounts (recommended) or domain accounts. The requirements below are valid in both scenarios.

User Account Name	Must Be Member of Group(s)	Description
`oracle`	`Administrators` `Remote Desktop Users`	Account must be created on the database server. This user is the owner of the Oracle database software. If this user does not already exist, it will automatically be created by the installation (local user).
`stibosw`	`Administrators` `Remote Desktop Users`	Account must be created on the application server(s). This user is the owner of the STEP application and HTTP server software. If this user does not already exist, it will automatically be created by the installation (local user).

Password Properties

Because both users act as logon users for Windows services, it is important that the password does not change or expire. Doing so would prevent the required services from starting.

Single Server Setup

In a single server setup, with the database and application server running on one machine, both accounts must be created. This is often the case for test systems.

As indicated in the table above, there might be more than one application server, such as in the case of a clustered system setup. In these instances, the user account must exist on all application servers.

Service Accounts vs. Logon Accounts

If security policies require separation of service and logon accounts, two separate accounts must be created. The service account password may never expire and must not be changed.

Windows OS Services

The installer package is dependent on the 'Windows Management Instrumentation' Windows Service being enabled and started.

MacOS System Requirements

The supported operating systems are:

- macOS 14 (Sonoma)
- macOS 13 (Ventura)
- macOS 12 (Monterey)

User Account Name	Must Be Member of Group(s)	Description
stibosw	stibosw	Account must be created on the DTP server. This user is the owner of the DTP server software.

A server running MacOS X is only required if there is a need for running Adobe InDesign Server on MacOS.

Installation Prerequisites

Before installing a STEP system, the following software must be available on the servers:

- OpenJDK / AdoptJDK Java (JDK or JRE) 64-bit (refer to 2024.1 Platform and Software Support)
- Perl 64-bit
- SPOT foothold distribution file (Stibo software)

Java

The STEP application requires OpenJDK* 64-bit to run. As part of the STEP deployment (patch procedure), AdoptOpenJDK* will be downloaded and installed. This Java version is regularly tested and released with the STEP application, but the SPOT tool itself also requires Java. Therefore, OpenJDK / AdoptOpenJDK Java must be installed manually on the servers in order to download and install third-party software and the STEP application.

Java for Windows

For Windows, download AdoptJDK Java JRE* from https://adoptopenjdk.net/. It must be 64-bit.

Follow the steps below to install AdoptJDK Java on Windows:

1. Login with an Administrator account.
2. Download AdoptOpenJDK binaries (64-bit) and save it to any directory (refer to the above URL).
3. Unzip the binaries in the chosen directory (E: is used in the below example commands) by running the following commands in command prompt:

```
unzip -d E:\java OpenJDK%U-jdk_x64_windows_hotspot_%u%b%.zip>
```

4. Create a link for the newly unzipped Java binaries to allow for easy update when a newer version is installed:

```
mklink /d E:\java\jre.current E:\java\jdk%u%-b%-jre
```

5. Add the following to the System Environment variable PATH:

```
E:\java\jre.current\bin
```

Java for Linux

On Linux, OpenJDK 1.8 64-bit can be downloaded using Yum. OpenJDK Java is located in the standard Red Hat and Oracle Linux repositories.

Follow the steps below to install OpenJDK Java on Linux:

1. Login as the root user.
2. Install Java by running the following command in a Terminal:

```
yum install java-1.8.0-openjdk.x86_64
```

Perl

Perl for Windows

When running SPOT the first time ActivePerl will be downloaded and the installer will be launched. For more information on installing ActivePerl, refer to the **Installing ActivePerl on Windows** section of the documentation.

Perl for Linux

As Perl is standard on Linux, nothing has to be done.

SPOT Foothold Distribution

Running the installation requires an Installation Package provided by Stibo Systems. The Installation Package contains all the required third-party software and scripts used to run the installation. For more information, refer to the **Pre-Installation Tasks** section of the documentation.

The installation package is automatically downloaded to the servers from the Stibo Systems release server using a foothold distribution.

The foothold zip file contains required scripts and customer specific certificates that allow for the download of third-party software from the Stibo Systems release server. The foothold zip can be received via e-mail from Stibo.

Important: All servers must be able to access the Stibo Update Server via HTTPS (port 443): https://updates.stibosystems.com.

Pre-Installation Tasks

Before beginning the installation, the system must have been set up according to the **System Requirements** section of the documentation. Additionally, the installation instructions described below assume that the volumes / partitions have been configured as listed in the **Default Filesystem Layout** section of the documentation.

Important: The foothold distribution file (received by email) is required to get started. The email also includes a required password to unzip the foothold file. This file should be provided by Stibo Systems.

User Account Controls

If using Windows, the Use Account Control must be disabled during installation.

1. Go to **Control Panel** > **User Accounts** > **Change User Account Control settings** and set notification to 'Never notify'.
2. Restart server to activate settings.

Deploying the Installation Package

The foothold distribution file must be placed in a specific directory on the server to get started. This directory is referred to as STEP_HOME.

Important: The steps described in this topic must be executed on all servers included in the setup: Database, Application, and DTP servers.

Create STEP_HOME Directory

On Linux (as a root user)

1. Go to `/opt` directory:

   ```
   cd /opt
   ```

2. Create `stibo/step` directories:

   ```
   mkdir -p stibo/step
   ```

On Windows

1. Go to E: drive
2. Create `stibo\step` directories using Explorer or from command line:

   ```
   mkdir stibo\step
   ```

Unzip Foothold Distribution File

The foothold zip file (`<customer-name>-foothold.zip`) must now be placed in the STEP_HOME folder. This can be done using SCP/SFTP (Linux) and FTP, or copy / paste through RDP (Windows).

On Linux (as root user)

1. Go to the STEP_HOME directory created above:

```
cd /opt/stibo/step
```

2. Unzip the foothold file using the supplied password:

```
unzip -P <password> <customer-name>-foothold.zip
```

On Windows

1. Go to the STEP_HOME directory created above:

```
E: <enter>
cd stibo\step <enter>
```

2. Unzip the foothold file by right-clicking the file and selecting `Extract All`. Enter the supplied password when prompted.

Now that the foothold distribution file has been deployed on all servers, the user can download and install third-party software and the STEP software.

Prepare SPOT and Download Installation Scripts

Prepare SPOT for the first time and download the installation scripts.

On Linux

Run the following commands:

```
[root@test step]# ./spot --wrapper
Generating wrapper /opt/stibo/step/spot with baked in JAVA_HOME=/usr/lib/jvm/java-1.8.0-openjdk-1.8.0.191.b12-1.el7_6.x86_64/jre
[root@test step]# ./spot --install
To apply this recipe 1 bundle(s) need to be installed.
The prepared recipe was stored as: /opt/stibo/step/admin/spot/recipes/prepared/2019-01-11-20-19-55.spr
```

On Windows

Run the following commands:

```
E:\stibo\step>spot --wrapper
Generating wrapper E:\stibo\step\spot.bat with baked in JAVA_HOME=E:\java\jdk8u192-b12
```

```
E:\stibo\step>spot --install
To apply this recipe 1 bundle(s) need to be installed.
The prepared recipe was stored as: E:\stibo\step\admin\spot\recipes\prepared\2019-
01-11-20-19-55.spr
```

Results

After the last command has run, the following STEP_HOME/admin folder will be created:

```
STEP_HOME/
       |-- admin/
               |-- install/
               |      |-- config/
               |      |-- documentation/
               |      |-- oracle-database-19c/
               |      |-- oracle-instant-client/
               |      |-- oracle-instant-client-19c/
               |      |-- oracle-single-server/
               |      |-- os-prerequisites/
               |      |-- pm/
               |      |-- step-audit
               |      |-- step/
               |      |-- step-database/
               |      |-- toolbox-backup/
               |      |-- toolbox-db/
               |      |-- tools/
               |-- install.pl
               |-- spot/
               |-- var/
                       |-- <servername>/
```

These installation scripts comprise what is known as the Installation Package.

Installation

Once the Installation Package has been deployed, the actual installation can begin. The STEP_HOME/admin folder contains the install.pl (Perl) script, which is used to launch the download and install menu.

For more information on the STEP_HOME directory and the Installation Package, refer to the **Pre-Installation Tasks** section of the documentation.

The installation of STEP is divided into two phases:

- Downloading the Software
- Installing the Software

Downloading the Software

The software must be downloaded before installation.

Note: The third-party software downloaded and used for the STEP installation can be found as HTML documentation in: `STEP_HOME/admin/install/documentation/html/software.html`

1. Log onto the server (database or application server) as `root` (Linux) or `Administrator` (Windows)
2. In a terminal window / command prompt, go to STEP_HOME/admin:

   On Linux

   ```
   cd /opt/stibo/step/admin
   ```

   On Windows

   ```
   E: <enter>
   cd stibo\step\admin <enter>
   ```

3. Run the install.pl script:

   On Linux

   ```
   ./install.pl
   ```

   On Windows

   ```
   install.pl
   ```

When running the script for the first time, the user is prompted to create a folder to hold the downloaded software.

The `Software folder` defaults to `<workarea>/install-software`. This directory can be created if you select 'y', or you can select 'n' and specify another location:

```
[root@test admin]# ./install.pl
2019-01-14 21:19:53 Info     install.pl -> /opt/stibo/step/admin/install.pl
   -> Arguments:
2019-01-14 21:19:53 Command  install.pl ->
/opt/stibo/step/admin/install/config/fetch-software.pl --
log=/opt/stibo/step/diag/logs/test/install.log
2019-01-14 21:19:53 ERROR    fetch-software.pl -> Default software folder not
found: /workarea/install-software
Do you want to create '/workarea/install-software' [y/n]:
```

Note: The workarea directory defaults to `/workarea` (Linux) and `L:` (Windows). A different location must be specified in the database server, usually, `/opt/install-software` (Linux), or `E:\install-software` (Windows) since `/workarea` and `L:\` are only standard partitions for the application server.

The software directory holds all the downloaded software. The default directory can be changed later by pressing `'c'` and entering a new location.

Sharing the Software Directory Between Servers

If possible, the software directory should be shared between all servers (both database and application servers) in the setup. This way the installation script will be able to determine the same directory and install configuration.

It is common for the database and application server(s) to not have a shared filesystem. In these scenarios, you will have to use a local filesystem and replicate the configuration file (install.properties). On the database server, you would only download the database-related software and on the application server(s) you would only download the application related software.

Choosing Software to Download

After the Software folder is created, the following menu is displayed:

```
--------------------------------------------------------------------------------
The following software have been marked for download

Software directory: /opt/install-software
--------------------------------------------------------------------------------
    No software marked for download



--------------------------------------------------------------------------------
Choose software for download by selecting a platform

    Id: Platform
--------------------------------------------------------------------------------
     1: Generic
     2: Linux
     3: Mac
     4: Windows
--------------------------------------------------------------------------------
     c: Change software directory
     d: Download files (if not already downloaded) and exit
     q: Exit without downloading
     a: Switch to advanced mode (not recommended)
--------------------------------------------------------------------------------

Choice:
```

Depending on the role of the server you are connected to, and what platform you are installing on, you now must choose what software to download. This is done by choosing the **Id** (1 – 4) in the menu.

Example (Linux)

In the following example the user is downloading software required for a database server installation on the Linux platform (sub-menu Id=2):

```
--------------------------------------------------------------------------------
Please select software to download for 'Linux'
--------------------------------------------------------------------------------

    Id: Software                                                Action
--------------------------------------------------------------------------------
     a: ** Oracle Database 12c **
        oracle-database-12c                                     skip
        oracle-database-12c-patchset-update                     skip

     b: ** Oracle Instant Client 12c **
        oracle-instant-client                                   skip

     c: ** Oracle tools **
        oracp                                                   skip

--------------------------------------------------------------------------------
   a-z: Toggle between 'download' and 'skip'
     0: Back to main menu
--------------------------------------------------------------------------------

Choice:
```

1. Once Linux has been selected, the user would type `'a'` to mark the group 'Oracle Database 12c' for download as this group is required software for the database server.

```
--------------------------------------------------------------------------------
Please select software to download for 'Linux'
--------------------------------------------------------------------------------

    Id: Software                                                Action
--------------------------------------------------------------------------------
     a: ** Oracle Database 12c **
        oracle-database-12c                                     skip
        oracle-database-12c-patchset-update                     skip

     b: ** Oracle Instant Client 12c **
        oracle-instant-client                                   skip

     c: ** Oracle tools **
        oracp                                                   skip

--------------------------------------------------------------------------------
   a-z: Toggle between 'download' and 'skip'
     0: Back to main menu
--------------------------------------------------------------------------------

Choice: a
```

2. Next, the user would enter `'0'` to return to the main menu which has updated to reflect the previous selections.

```
--------------------------------------------------------------------------------
Please select software to download for 'Linux'
--------------------------------------------------------------------------------

    Id: Software                                                Action
--------------------------------------------------------------------------------
     a: ** Oracle Database 12c **
        oracle-database-12c                                     download
        oracle-database-12c-patchset-update                     download

     b: ** Oracle Instant Client 12c **
        oracle-instant-client                                   skip

     c: ** Oracle tools **
        oracp                                                   skip

--------------------------------------------------------------------------------
   a-z: Toggle between 'download' and 'skip'
     0: Back to main menu
--------------------------------------------------------------------------------

Choice: 0
```

3. The user also requires a 'starter' database (referred to as STEP standard) which must be downloaded as well. It is located under the Generic sub-menu (Id=1).

```
--------------------------------------------------------------------------------
The following software have been marked for download

Software directory: /opt/install-software
--------------------------------------------------------------------------------
** linux **
    oracle-database-12c
    oracle-database-12c-patchset-update


--------------------------------------------------------------------------------
Choose software for download by selecting a platform

    Id: Platform
--------------------------------------------------------------------------------
     1: Generic
     2: Linux
     3: Mac
     4: Windows
--------------------------------------------------------------------------------
     c: Change software directory
     d: Download files (if not already downloaded) and exit
     q: Exit without downloading
     a: Switch to advanced mode (not recommended)
--------------------------------------------------------------------------------

Choice: 1
```

4. There are multiple versions of the 'starter' database available from the menu. Each choice is compatible with a range of STEP versions. Choose `'c'` if installing the latest version of STEP.

```
--------------------------------------------------------------------------------
Please select software to download for 'Generic'
--------------------------------------------------------------------------------

    Id: Software                                                Action
--------------------------------------------------------------------------------
     a: ** STEP base (>= step-8.1) **
        stepbase-datapump.step-8.1                              skip

     b: ** STEP base (>= step-8.2) **
        stepbase-datapump.step-8.2                              skip

     c: ** STEP base (>= step-8.3) **
        stepbase-datapump.step-8.3                              skip

--------------------------------------------------------------------------------
   a-z: Toggle between 'download' and 'skip'
     0: Back to main menu
--------------------------------------------------------------------------------

Choice: c
```

5. Type `'0'` to return to the main menu.
6. To start the download, type `'d'`.

```
--------------------------------------------------------------------------------
The following software have been marked for download

Software directory: /opt/install-software
--------------------------------------------------------------------------------
** generic **
    stepbase-datapump.step-8.3
** linux **
    oracle-database-12c
    oracle-database-12c-patchset-update




--------------------------------------------------------------------------------
Choose software for download by selecting a platform

    Id: Platform
--------------------------------------------------------------------------------
     1: Generic
     2: Linux
     3: Mac
     4: Windows
--------------------------------------------------------------------------------
     c: Change software directory
     d: Download files (if not already downloaded) and exit
     q: Exit without downloading
     a: Switch to advanced mode (not recommended)
--------------------------------------------------------------------------------

Choice: d
```

7. The download process begins.

```
2019-01-15 18:54:21 Command  fetch-software.pl -> /opt/stibo/step/spot --
baseurl=updates:/install --download=stepbase-datapump/stepbase-datapump-11.zip -
-output=/opt/install-software
Stibo Patch Operations Tool - test - test - test
Downloading [########################################] 100% 1724/1724 k
/workarea/install-software/stepbase-datapump/stepbase-datapump-11.zip
2019-01-15 18:54:24 Command  fetch-software.pl -> /opt/stibo/step/spot --
baseurl=updates:/install/software --download=oracle-database-
12c/linux/linuxx64_12201_database_1of2.zip,oracle-database-
12c/linux/linuxx64_12201_database_2of2.zip,oracle-database-12c-patchset-
update/linux/p6880880_122010_Linux-x86-64.zip,oracle-database-12c-patchset-
update/linux/p28163133_122010_Linux-x86-64.zip --output=/opt/install-software
Stibo Patch Operations Tool - test - test
Downloading [########################################] 100% 3856/3856 M
```

The files are downloaded under the chosen **`Software folder`** which, in this example, is **`/opt/install-software`**. In an application server installation, this defaults to **`/workarea/install-software`** (Linux), or **`L:\install-software`** (Windows).

```
[root@test install-software]# pwd
/opt/install-software
[root@test install-software]# ls -l
total 16
drwxrwxrwx. 3 root root 4096 Jan 15 18:57 oracle-database-12c
drwxrwxrwx. 3 root root 4096 Jan 15 18:57 oracle-database-12c-patchset-update
drwxrwxrwx. 3 root root 4096 Jan 15 18:58 oracle-instant-client
drwxrwxrwx. 2 root root 4096 Jan 15 18:54 stepbase-datapump
```

Installing the Software

Once the software has been downloaded to the `Software folder`, the installation will begin.

> **Note:** All installation properties are documented as HTML documentation in: `STEP_HOME/admin/install/documentation/html/install-properties.html`

Initial Installation Steps

1. The initial installation steps require you to provide the names of the database, application, and DTP servers. This must be the FQDN (Fully Qualified Domain Name) of the server(s).

```
--------------------------------------------------------------------------------
Please enter minimum required properties:
--------------------------------------------------------------------------------
host.application-servers =
   Group 'STEP installation setup'
   Specifies the application server(s) in the STEP system.
   Should be fully qualified hostnames (in lower case).
   NOTE: Comma separated.
   - Default value ''
   - Must match pattern
   '^[a-z0-9-]*(\.[a-z0-9-]+)*(\s*,\s*[a-z0-9-]*(\.[a-z0-9-]+)*)*$'
   - This property is mandatory (not null)

Enter value: <Application Server FQDN>

host.database-servers =
   Group 'STEP installation setup'
   Specifies the database server in the STEP system.
   Should be fully qualified hostnames (in lower case).
   NOTE: Comma separated.
   - Default value ''
   - Must match pattern
   '^[a-z0-9-]*(\.[a-z0-9-]+)*(\s*,\s*[a-z0-9-]*(\.[a-z0-9-]+)*)*$'
   - This property is mandatory (not null)

Enter value: <Database Server FQDN>
```

   In the event that there is more than one application server, specify them all as a comma-separated list. This is the host name(s) of the server(s). If a DTP server is not part of the setup, just leave the property empty.

2. You will be prompted to provide the **step.system-name**. This is a unique name that identifies the system, and must exist in Stibo's internal systems as the license is mapped to this system name. This name must be

agreed upon by the customer and created as a licensed system in Stibo.

Some examples include:

- <customer>-development / <customer>-dev
- <customer>-production / <customer>-prod

```
step.system-name =
   Group 'STEP configuration setup'
   The STEP system name
   - Default value ''
   - This property is mandatory (not null)

Enter value []: test-install
```

3. You will be prompted to provide the **step.takeout-ur**. This is the version of STEP to be installed to the system. Pressing `Enter` will set the value as default, which is the latest version.

```
Enter value [to:step/trailblazer/latest.spr]: [Enter]
```

Installation Menu

After the **step.takeout-ur**is provided and pressing `Enter`, the following installation menu is shown:

```
--------------------------------------------------------------------------------
Date and time              : 2019-01-15 19:16:16
Server                     : testdb.ec2.internal
Server role(s)             : database-server
OS User(s)/OS Group(s)     : oracle:dba
OS Caption                 : Red Hat Enterprise Linux Server release 7.6 (Maipo)
Virtual server             : yes
SPOT is online             : yes
STEP version               : 0.0
Current user               : root
Switch user method         : sudo -u <user> <cmd>
Tablespace (stepsysdata)   : 8192 MB (initial 8192 MB), 1 file(s)
Tablespace (stepsysblob)   : 8192 MB (initial 8192 MB), 1 file(s)

Primary database server    : testdb.ec2.internal
Primary application server : testapp.ec2.internal

Configuration file         : /workarea/install-software/install.properties
Log file                   :
/opt/stibo/step/diag/logs/testdb.ec2.internal/install.log

First/last script          : 1 -> 10
Dry run                    : no
--- -------- ---------------------------------------------------------------
Id  Action   Script description
--- -------- ---------------------------------------------------------------
 1  Run      OS prerequisites
 2  Run      Apply the latest STEP DB Server toolbox
 3  Run      Apply the latest STEP DB Backup toolbox
 4  Run      Install and configure Oracle 12c database software
 5  Run      Install Oracle OPatch and Patch Set Update for Oracle 12c Database
 6  Run      Create database instance with Oracle 12c DBCA
 7  Run      Create STEP database schemas
 8  Run      Create STEP database users
 9  Run      Configure STEP database
10  Run      Import STEP Base database

--- --------------------------------------------------------------------------
Key Command
--- --------------------------------------------------------------------------
** DOWNLOAD **
 d  Download third party software, tools etc. required for the installation
 u  Download and install latest install scripts

** CONFIGURATION **
 e  Edit configuration file
 p  List properties by property name
 v  List properties by property value
 c  List properties that differ from default

** INSTALL **
 f  Change first script
 l  Change last script
 i  Start installation
 q  Quit

** HELP/DEBUG **
 h  Help
 n  Toggle dry run mode
 r  Revision log for install package
 s  Show/hide script
 w  Show/hide reason why steps are marked with action 'Skip'
--- --------------------------------------------------------------------------

Choice:
```

The line `First/last` script indicates which installation scripts are run or skipped. If a script is skipped, you can type ‘w’ to identify why this script is being skipped. If you only want to run some scripts, you can use ‘f’ and ‘l’ to indicate the first and last script to run.

Default Configuration Properties

The installation consists of a large set of default configuration properties which can be modified by overwriting the defaults in the following file: `<Software folder>/install.properties`.The file can be modified directly from the installation menu by selecting option ‘e.’ Upon saving the file, the installation process will return to the installation menu.

Options ‘p,'v,’ and ’c’ are optional to list the various available properties. The wildcard '*’ can be used with ‘p’ and ‘v’ options to show all properties.

The OS user(s) and group(s) are examples of this, and can be changed to match the customer's requirements.

For some examples, refer to the **Configuration Examples** section of the documentation.

For information on the filesystem layout, refer to the **Default Filesystem Layout** section of the documentation.

Toggle Dry Run

Option ‘n’ will toggle dry run mode on or off. The default is off. When dry run mode is enabled, and an installation is started by selecting option ‘i,’ the installation goes through its install routine based on the supplied properties, and builds a file that self-documents the scripts and commands that will be executed during the installation process without installing any components.

The file created during a dry run is `<STEP_HOME>/logs/<hostname>/dry-run/dry-run.txt`.

Any files that are generated as part of the installation routine will be placed in: `<STEP_HOME>/diag/logs/<hostname>/dry-run/<username>`.

Note: Ensure to toggle dry run to 'off' to start an actual installation that physically installs the components.

Completing the Installation

When ready, typing ‘i’ will start the installation. If errors are found during installation, they must be solved before the installation can continue. Upon a successful installation, STEP will be started automatically on the application server.

An installation log file is found in `<STEP_HOME>/logs/<hostname>/install.log`.

Note: When installing the database server in Windows, the system administrator will have to run all options (excluding 'OS prerequisites') as the oracle user or equivalent, otherwise the actions will be marked as ‘Skip’.

Application Server Prompt Examples

```
--------------------------------------------------------------------------------
The following software have been marked for download

Software directory: /workarea/install-software
--------------------------------------------------------------------------------
** linux **
    oracle-instant-client




--------------------------------------------------------------------------------
Choose software for download by selecting a platform

    Id: Platform
--------------------------------------------------------------------------------
     1: Generic
     2: Linux
     3: Mac
     4: Windows
--------------------------------------------------------------------------------
     c: Change software directory
     d: Download files (if not already downloaded) and exit
     q: Exit without downloading
     a: Switch to advanced mode (not recommended)
--------------------------------------------------------------------------------

Choice: d
```

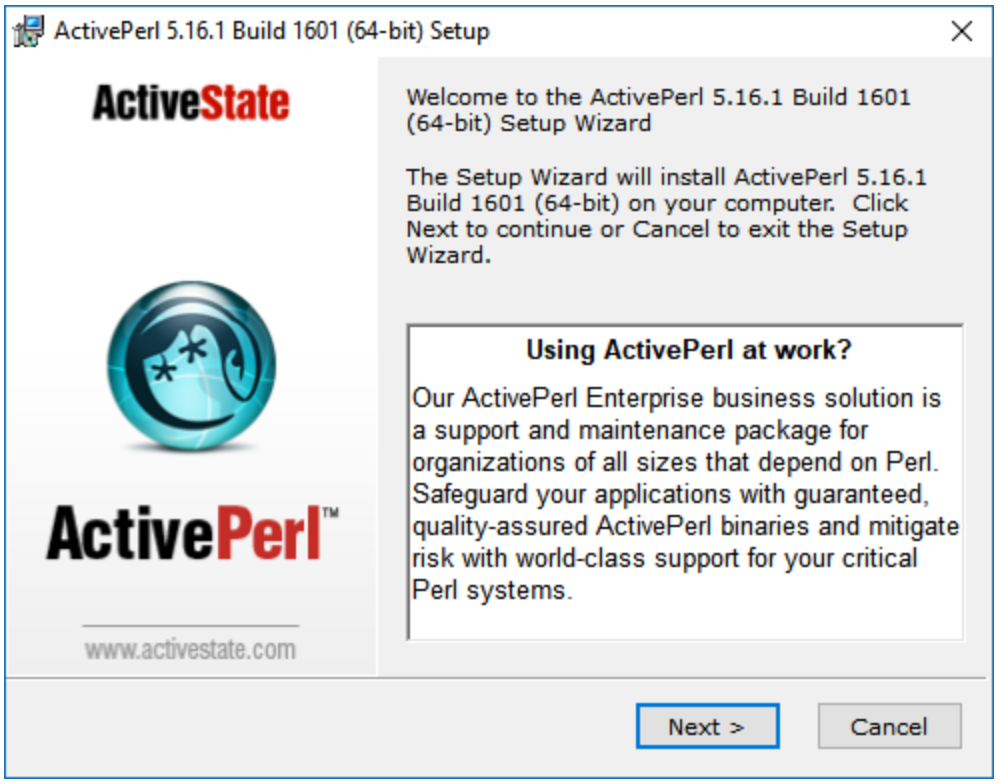
```
--------------------------------------------------------------------------------
Date and time              : 2019-01-22 15:02:28
Server                     : ip-172-31-36-178.ec2.internal
Server role(s)             : application-server
OS User(s)/OS Group(s)     : stibosw:stibosw
OS Caption                 : Red Hat Enterprise Linux Server release 7.6 (Maipo)
Virtual server             : yes
SPOT is online             : yes
STEP version               : 0.0
Current user               : root
Switch user method         : sudo -u <user> <cmd>

Primary database server    : testdb.ec2.internal
Primary application server : testapp.ec2.internal

Configuration file         : /workarea/install-software/install.properties
```

```
Log file                  : /opt/stibo/step/diag/logs/ip-172-31-36-
178.ec2.internal/install.log

First/last script         : 1 -> 4
Dry run                   : no
--------------------------------------------------------------------------------

--- -------- ---------------------------------------------------------------
Id  Action   Script description
--- -------- ---------------------------------------------------------------
 1  Run      OS prerequisites
 2  Run      Install and configure the Oracle instant client software
 3  Run      Install and configure the Apache HTTP server software
 4  Run      Install and configure the STEP software

--- -------------------------------------------------------------------------
Key Command
--- -------------------------------------------------------------------------
** DOWNLOAD **
 d  Download third party software, tools etc. required for the installation
 u  Download and install latest install scripts

** CONFIGURATION **
 e  Edit configuration file
 p  List properties by property name
 v  List properties by property value
 c  List properties that differ from default

** INSTALL **
 f  Change first script
 l  Change last script
 i  Start installation
 q  Quit

** HELP/DEBUG **
 h  Help
 n  Toggle dry run mode
 r  Revision log for install package
 s  Show/hide script
 w  Show/hide reason why steps are marked with action 'Skip'
--- -------------------------------------------------------------------------
```

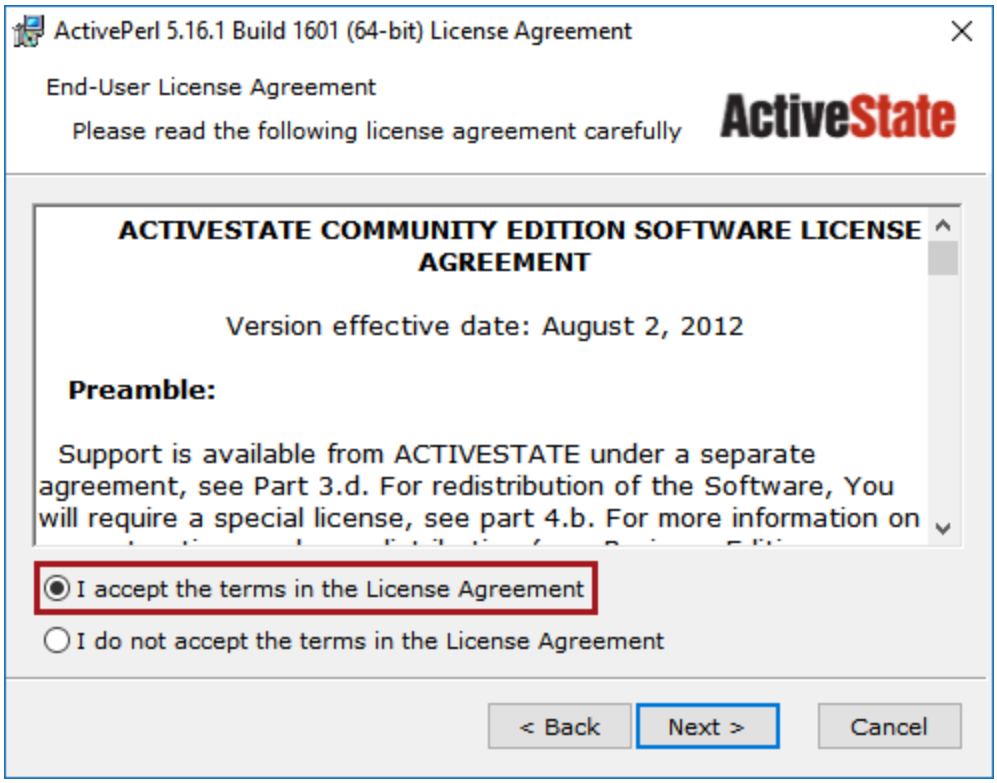
Installing ActivePerl on Windows

The ActivePerl installer is automatically launched when running `STEP_HOME\spot-install` for the first time.

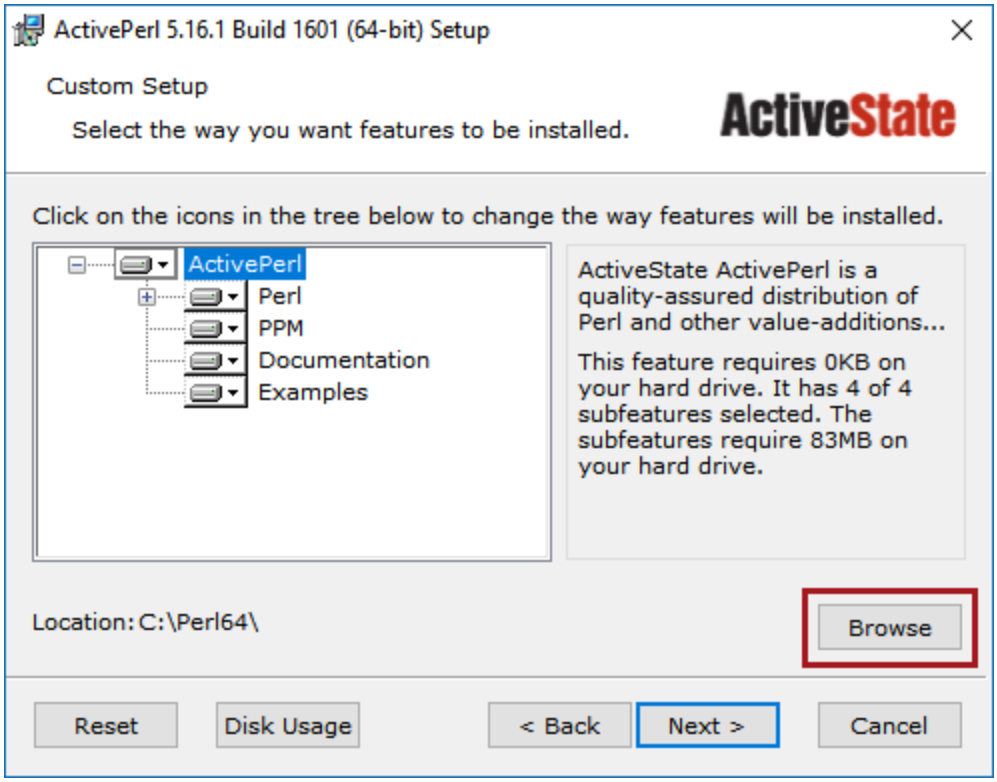
1. Click **Next** to proceed with the installation.



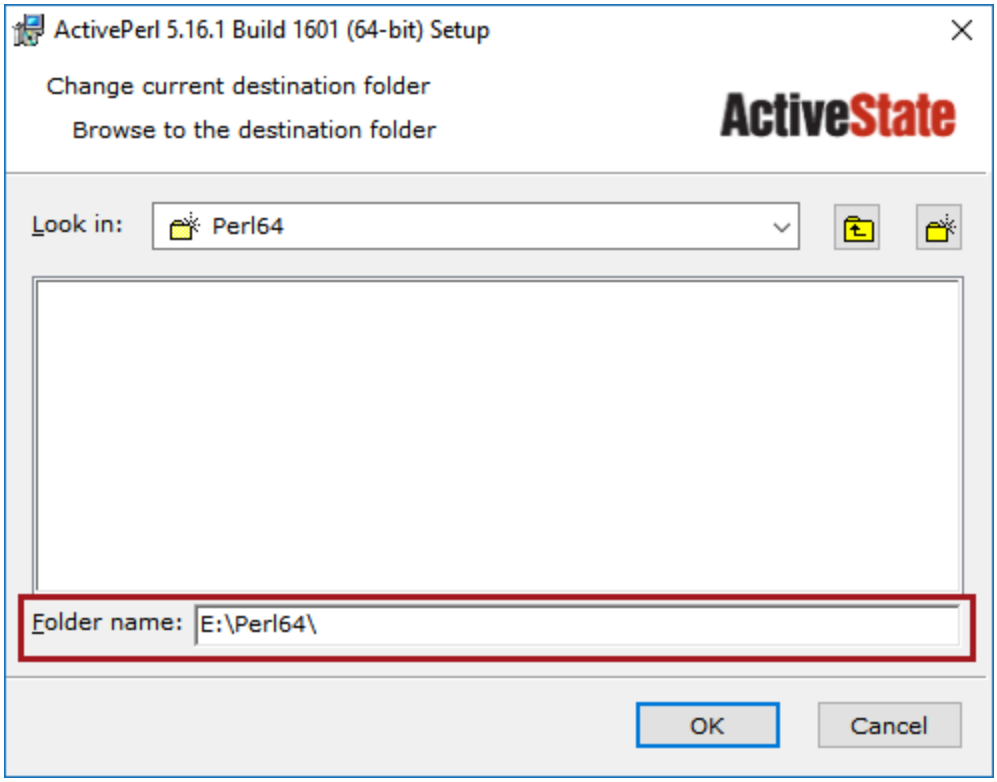
2. Accept the End-User License Agreement and click **Next**.



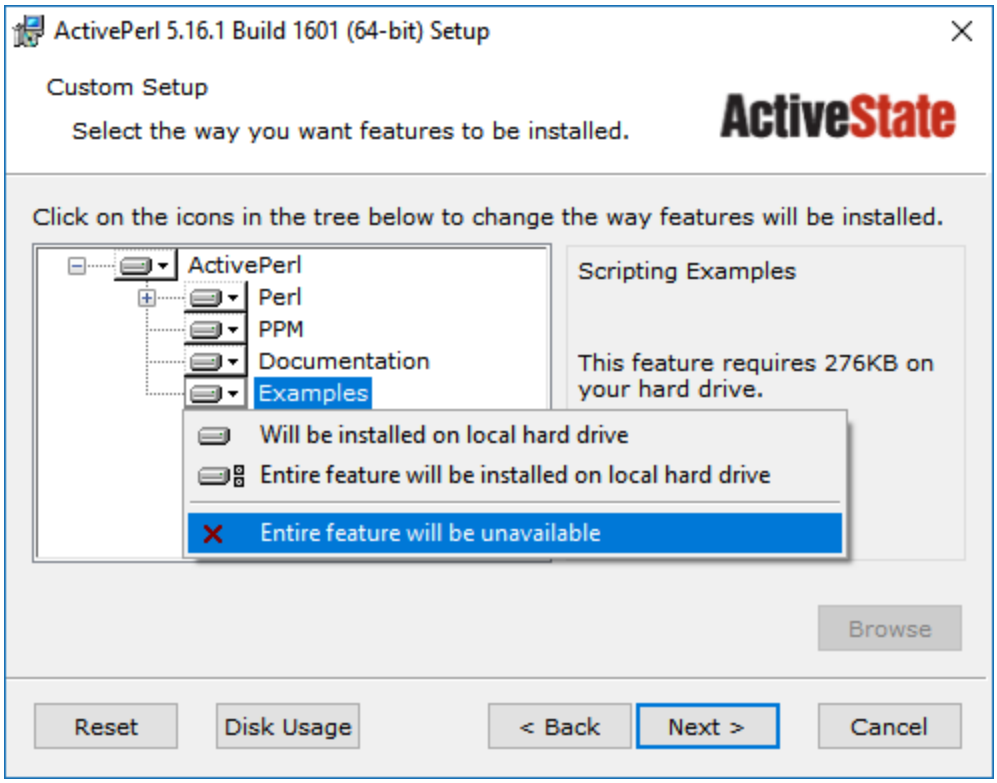
3. Click **Browse** to open the 'Change current destination' folder dialog.



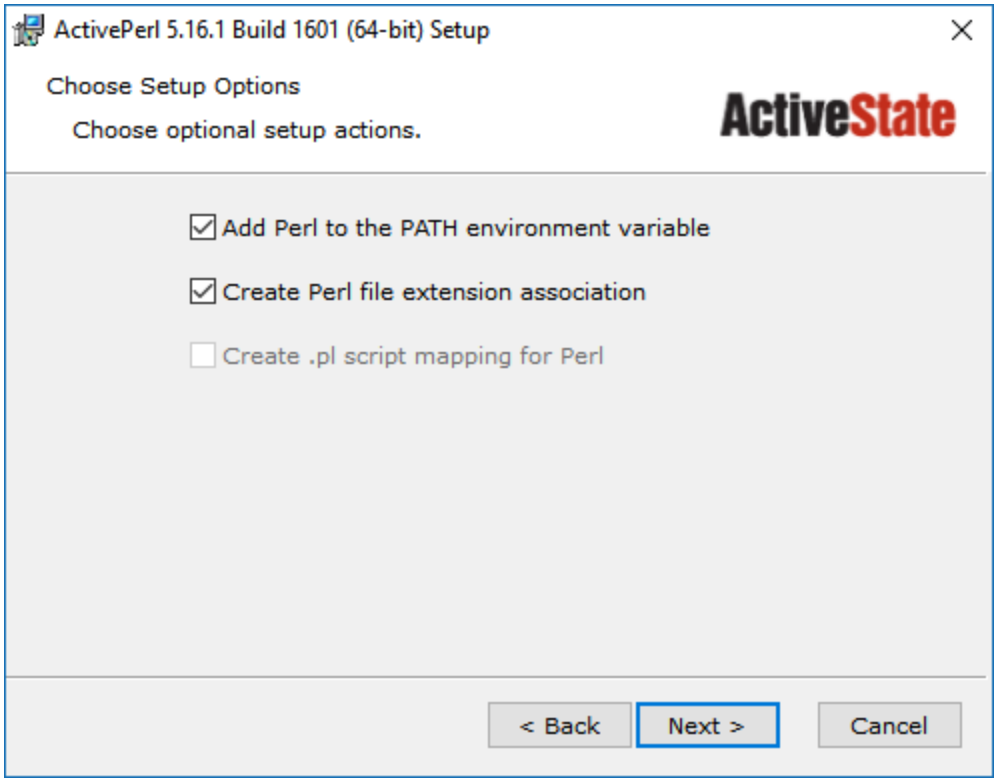
4. Enter the correct path in the 'Folder name' parameter and click **OK**. The location in the image is for example purposes.



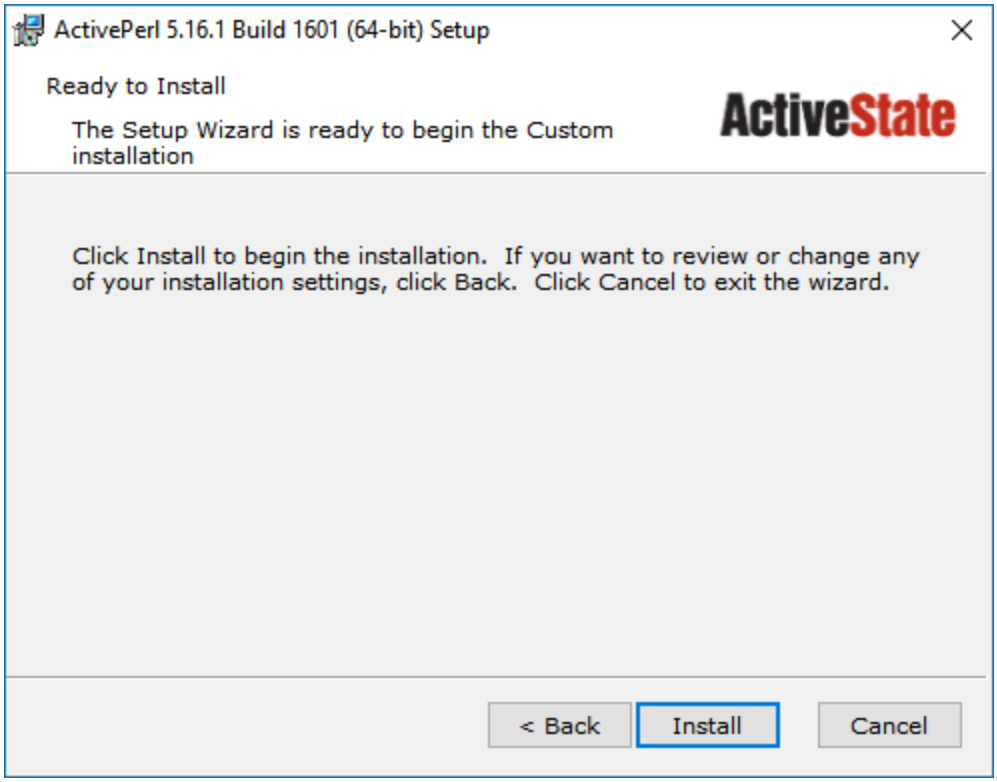
5. Right-click on 'Examples' in the navigation tree, and select 'Entire feature will be unavailable'. Click **Next**.



6. On the 'Choose Setup Options' screen, leave the defaults selected, and click **Next**.



7. On the 'Ready to Install' screen, click **Install**.



8. Once the installation is complete, click **Finish** to exit the installer. Check the 'Display the release notes' box to view the release notes upon closing the installer.

Default Filesystem Layout

The default filesystem layouts expected by the installation scripts are detailed in the tables below.

This information can also be found in the Hardware Recommendations document (provided by Stibo Systems).

Linux

Database Server

Server Internal Storage:

Partition	Description	Size (Net)
/	System	10 GB (or more)
/opt	Software	40 GB
/tmp	Temp file system	10 GB
/var	Variable files	10 GB
/home	Home	5 GB

Server External Storage:

Partition	Description	Size (Net)
/database/dbredo1	Redo1	10 GB
/database/dbredo2	Redo2	10 GB
/database/db1	Database	XX GB
/database/dblog	DB-Log	XX GB
/database/backup	Online Backup	XX GB

Note: The online backup partition (staging area for RMAN) is optional depending on any existing backup solutions on an existing storage setup.

Application Server

Server Internal Storage:

Partition	Description	Size (Net)
/	System	10 GB (or more)
/opt	Software + Application temp	100 GB
/tmp	Temp file system	10 GB
/var	Variable files	10 GB
/home	Home	5 GB

Server External Shared Storage:

Partition	Description	Size (Net)
/workarea	Workarea	100 GB
/upload	Upload	50 GB

Note: In a clustered application server setup, this storage must be shared with read / write access by all application servers.

Windows

Database Server

Server Internal Storage:

Partition	Description	Size (Net)
C:\	System	40 GB
E:\	Software	40 GB

Server External Storage (SAN):

Partition	Description	Size (Net)
F:\	Redo1	10 GB
G:\	Redo2	10 GB
H:\	Database	XX GB
I:\	DB-Log	XX GB
K:\	Online Backup	XX GB

Note: The online backup partition (staging area for RMAN) is optional depending on any existing backup solutions on an existing storage setup.

Application Server

Server Internal Storage:

Partition	Description	Size (Net)
C:\	System	40 GB
E:\	Software + application temp	100 GB

Server External Storage (SAN):

Partition	Description	Size (Net)
L:\	Workarea	100 GB
M:\	Upload	50 GB

Note: In a clustered application server setup, this storage must be shared with read / write access by all application servers.

Cluster SSH Configuration

When configuring a Linux application server cluster, the OS account that runs the STEP software (default 'stibosw') must be able to login via SSH (Secure Shell) between the application servers without being prompted for a password. This secure access is accomplished by using public key authentication.

SSH Setup

1. Select one machine to be the temporary master. This machine is where the initial commands will be run.
2. On the master machine, create a temporary folder in the shared storage area (default `/workarea`), for example:

```
cd /workarea
mkdir tmpkeys
```

3. On the same machine, enter the following commands with the stibosw account:

```
ssh-keygen -t rsa -b 3072
cp -pa ~/.ssh/id_rsa.pub /workarea/tmpkeys/id_rsa-$(hostname).pub
```

> **Note:** Accept all defaults.

4. On **each** other machine in the cluster *except* the master, enter the following commands with the stibosw account:

```
ssh-keygen -t rsa -b 3072
cp -pa ~/.ssh/id_rsa.pub /workarea/tmpkeys/id_rsa-$(hostname).pub
touch ~/.ssh/authorized_keys && chmod 600 ~/.ssh/authorized_keys
```

> **Note:** Accept all defaults.

5. On the master machine again, enter the following commands with the stibosw account:

```
cd /workarea/tmpkeys
for k in *.pub; do cat $k >> id_rsa.all; done
cp -pa id_rsa.all >> ~/.ssh/authorized_keys
```

6. Still on the master machine, run the following command with the stibosw account:

```
cat /workarea/tmpkeys/id_rsa.all | ssh stibosw@{SERVER_NAME} 'cat >>
~/.ssh/authorized_keys'
```

In the command, the `{SERVER_NAME}` refers to the name of the next application server in the cluster. For example, if application server 1 is the temporary master, `{SERVER_NAME}` would refer to application server 2.

7. Repeat the command of the above step on **all** of the remaining application servers in the cluster.
8. Once the SSH key exchange is done, it is possible for the stibosw user to SSH to any other machine in the cluster using public key authentication.
9. After confirming that public key authentication is working, the `/workarea/tmpkeys` directory can be deleted from the system.

Configuration Examples

Application Server

The following configuration steps are typically required:

- Specify the fully qualified hostnames of the STEP servers (mandatory).
- Specify the fully qualified hostname of the database server (mandatory).
- Specify the System Name (mandatory).
- Specify a STEP takeout URL.
- Specify a filesystem for software if it differs from the default.
- Specify a filesystem for Workarea and Upload if they differ from default.
- Specify a filesystem for STEP temp-files if it differs from default.
- Specify Java heapsize for STEP if it differs from default.
- Specify database name and/or service if they differ from default step.
- Specify name and/or password for STEP database users if they differ from default.

Examples

```
host.application-servers = stepapp.stibo.com
host.database-servers = stepdb.stibo.com
step.system-name = step-test
#step.takeout-url =
#filesystem.app-software =
#filesystem.app-workarea =
#filesystem.app-upload =
#filesystem.step-temp-files =
#step.applicationserver-maximumheapsize =
#step.applicationserver-initialheapsize =
```

If the name of the STEP OS user differs from the default 'stibosw':

```
user.app = <my-os-app-user>
```

If the service name of the database differs from dbca.db-name:

```
dbca.service-name = <my-db-service-name>
```

If the name and/or password of the stepsys / stepview database user differs from default:

```
database.stepsys-owner = <my-db-stepsys>
database.stepsys-owner-passwd  = <my-db-stepsys-password>
```

```
database.stepview-owner = <my-db-stepview>
database.stepview-owner-passwd = <my-db-stepview-password>
```

Note: Leading blanks are not valid.

Database Server

The following configuration steps are typically required:

- Specify the fully qualified hostnames of the STEP servers (mandatory).
- Specify the fully qualified hostname of the database server (mandatory).
- Specify the System Name (mandatory).
- Specify a filesystem for software if it differs from the default.
- Specify a filesystem for DB-Workarea and DB-Upload if they differ from default, or set to blank if they are not present on the database server (ignored for single server installations).
- Specify filesystems for the database if they differ from default.
- Configure Oracle edition if it differs from default. Refer to the Configure Oracle Server Edition section below for more information.
- Manually specify the amount of memory that is available to the database and/or use memory target. Refer to the Configure Oracle Server Edition section below for more information.
- Specify the size of the tablespaces STEPSYSDATA and STEPSYSBLOB.

Examples

```
host.application-servers = stepapp.stibo.com
host.database-servers = stepdb.stibo.com
step.system-name = step-test
#filesystem.db-software =
filesystem.db-workarea =
filesystem.db-upload =
#filesystem.database =
#filesystem.dblog =
#filesystem.redo1 =
#filesystem.redo2 =
database.stepsysdata-size-gb = 32
database.stepsysblob-size-gb = 320
database.stepsysblob-initial-file-size-gb = 16
```

If the name of the database OS user differs from the default Oracle:

```
user.db = <my-os-db-user>
```

If the name of the database differs from the default step:

```
dbca.db-name = <my-db-name>
```

If the service name of the database differs from dbca.db-name:

```
dbca.service-name = <my-db-service-name>
```

If the instance name of the database differs from dbca.db-name:

```
oracle.sid = <my-instance-name>
```

If the name and/or password of the stepsys / stepview database user differs from default:

```
database.stepsys-owner = <my-db-stepsys>
database.stepsys-owner-passwd = <my-db-stepsys-password>
database.stepview-owner = <my-db-stepview>
database.stepview-owner-passwd = <my-db-stepview-password>
```

If the database is a RAC database, then specify the SCAN name:

```
oracle.database-listener-reference = <scan-name>
```

Note: Leading blanks are not valid.

Configure Oracle Instance Memory

By default, SGA_TARGET and PGA_AGGREGATE_TARGET are used. SGA and PGA are set to 5/6 and 1/6 of the accountable amount of memory for the database instance.

If you want to use MEMORY_TARGET, then you must specify:

```
dbca.use-memory-target = true
host.db-use-hugepages = false  (if Linux)
```

The available amount of memory for the database instance is calculated from either of the following rules (available-host-memory):

- For a dedicated database server, the available amount of memory is set to 2/3 of host.memory-ram.
- For a server which is both an application and database server, the available amount of memory is set to 2/5 of host.memory-ram.

The maximum allowable amount of memory for the database instance (dbca.max-db-memory) is set to 68719476736 (64GB) by default.

The accountable amount of memory for the database instance is calculated from the following rules:

- dbca.max-db-memory < available-host-memory
  - The calculation is based on dbca.max-db-memory
- dbca.max-db-memory >= available-host-memory

- The calculation is based on available-host-memory

- dbca.max-db-memory is set to 0
  - The calculation is based on available-host-memory

If the amount of available-host-memory is higher than dbca.max-db-memory and you want to use it, raise dbca.max-db-memory or set it to 0 to disable the limit. Alternatively, you can lower the value if you want your database instance to use less than the calculated available-host-memory.

Configure Oracle Server Edition

By default, the Oracle server software is installed as the standard edition (SE). To install the enterprise edition, you must specify:

```
oracle.install-edition-short-name = EE
```

The Oracle enterprise edition has a list of optional components. These components come at an additional cost. All components are installed, but you have the choice to enable / disable specific or all components. By default, the install script is configured to install Oracle enterprise with all optional components disabled.

To disable all optional components:

```
oracle.enterprise-edition-options-selection = true
oracle.enable-enterprise-edition-options-list
```

Note: The options list must be blank.

To enable specific optional components:

```
oracle.enterprise-edition-options-selection = true
oracle.enable-enterprise-edition-options-list = <comp1-name>:version, <comp2-
name>:version
```

The components can also be enabled or disabled after installation by using the command utility chopt.

The optional components are registered in the v$option view.

Upgrading to Oracle Enterprise Edition

To upgrade to the Enterprise Edition, you must be able to install the exact same Oracle server software (including patches). If the required software is unavailable or you want to upgrade to a new Oracle release, you must upgrade both the server software and the database. Refer to the Oracle upgrade guide for instructions.

Note: The Oracle upgrade guide is provided by Oracle's own documentation.

Upgrading in Linux

1. Stop all Oracle services related to the Oracle home you intend to uninstall. Note that the listener may be servicing other Oracle homes.
2. Backup the Oracle home you intend to uninstall.
3. Copy the db-parameter files, password files, listener.ora, tnsnames.ora, and sqlnet.ora from the Oracle home you intend to uninstall, and send them to a location outside Oracle base.
4. Comment out <ORACLE_SID> entries, in /etc/oratab, related to the Oracle home you intend to uninstall.
5. Uninstall the current Oracle server software installation from a location outside the Oracle home you intend to uninstall:

```
$ <ORACLE_HOME-TO-UNINSTALL>/uninstall/uninstall
```

Important: When the program says 'de-configure', the term means to remove. If the list of Oracle databases shown is not empty, then there is something wrong. You should not continue unless you truly want to remove the database.

6. Install the Oracle server software again (including patches).
7. Stop the listener if it is running.
8. Restore db-parameter files, password files, listener.ora, tnsnames.ora, and sqlnet.ora.
9. Start the listener.
10. Start the database.

The database is now upgraded to Enterprise Edition.

Upgrading in Windows

1. Stop all Oracle services related to the Oracle home you intend to uninstall. Note that the listener may be servicing other Oracle homes.
2. Backup the Oracle home you intend to uninstall.
3. Copy db-parameter files, password files, listener.ora, tnsnames.ora, and sqlnet.ora from the Oracle home you intend to uninstall, and send them to a location outside Oracle base.
4. Backup registry settings 'ORA_<ORACLE_SID>_%'.

5. Remove the listener if it has the same name as a listener in another Oracle home. To do this, manually execute Net Configuration Assistant (NETCA) in the Oracle home you intend to uninstall.
6. Delete Windows service 'OracleService<ORACLE_SID>', using ORADIM, belonging to the Oracle home you intend to uninstall.
7. Leave all command prompts and Explorer files that are located in the Oracle home you intend to uninstall.
8. Uninstall the current Oracle server software installation from a location outside the Oracle home you intend to uninstall:

```
$ <ORACLE_HOME-TO-DEINSTALL>\deinstall\deinstall.bat
```

Important: When the program says 'de-configure', the term means to remove. If the list of Oracle databases shown is not empty, then there is something wrong. You should not continue unless you truly want to remove the database.

9. Install the Oracle server software again (including patches).
10. Recreate the listener by executing Net Configuration Assistant (NETCA) in the re-installed ORACLE_HOME.
11. Stop the listener if it is running.
12. Restore db-parameter files, password files, listener.ora, tnsnames.ora, and sqlnet.ora.
13. Start the listener.
14. Recreate Windows service 'OracleService<ORACLE_SID>' using ORADIM. Restore registry settings for the service as needed.
15. Start the database.

The database is now upgraded to Enterprise Edition.

Offline Installations

Customers with strict security policies that prohibit external access to https://updates.stibosystems.com will prevent the standard installation from working, resulting in a failed installation. In these cases, it may be necessary to perform an 'offline' installation of STEP for both Application and Database servers.

An offline installation comprises two steps. The first step is to prepare the relevant installation recipes (install the latest STEP version) as well as creating a software repository of the software required during the installation. This software must be fetched from Stibo Systems' updates.stibosystem.com server. This step must be performed in a location and on a machine that has access to https://updates.stibosystem.com. The content created during this stage will need to be copied to the target server intended for the installation. The second step is to perform the installation on the target server using the files / content created from the first step using the online machine.

Note: Java JRE 1.8 is required for the tasks listed in the below sections.

Online Machine Tasks

1. Unzip the specific customer foothold.zip file into '<STEPHOME>'
2. `<STEPHOME>/spot --wrapper`
3. `<STEPHOME>/spot --install`
4. `<STEPHOME>/spot --offline-recipe --snapshot=install.spr`
5. `<STEPHOME>/spot --accessmode=offline --prepare=to:step/trailblazer/latest.spr --target=install.spr --output=latest-fullfat.spr`
6. `<STEPHOME>/admin/install/config/fetch-software.pl`

   Download the required 3rd party software (e.g., /workarea/install-software).
7. The following files will need to be transferred to the target server and will be used during the installation of STEP:
   - install.spr
   - latest-fullfat.spr
   - zipped / tarball of directory containing the downloaded software (e.g., workarea/install-software)

Target Server Tasks

1. Unzip the specific customer foothold.zip file into '<STEPHOME>'
2. `<STEPHOME>/spot --wrapper`
3. `<STEPHOME>/spot --apply=install.spr --sync`
4. `<STEPHOME>/admin/install.pl --swdir=<SWDIR>`

> **Note:** When prompted for the 'step.takout-url' property, enter `latest-fullfat.spr` including the full path to the file.

5. The installation will continue to process as normal.

> **Note:** During the installation of the STEP software, an error will be thrown testing a connection to the Stibo Updates server. Press ‘c’ to ignore and continue with the installation.

Linux Administration

This section describes administration procedures for a STEP system running on Red Hat® Enterprise Linux (x64) and Oracle® Enterprise Linux (x64), and also, for the third-party software components that it uses.

The provided information is intended for the users responsible for the administration of the STEP system.

The administrator must be familiar with the following:

- Red Hat / Oracle Enterprise Linux: general administration, monitoring, and maintenance.
- Oracle database administration. In the STEP environment, this means that the administrator must be able to perform the tasks mentioned in this document, e.g., using SQL*Plus for adding data files to a tablespace.
- The STEP Architecture Guide.

Administration for Linux

- Managing the STEP Application Server
- Start and Stop Procedures
- Verify the Installed STEP Version
- Send a Diagnostic Snapshot to Stibo Systems
- STEP System Configuration
- Health Check Facility
- Benchmark Facility
- Patching STEP
- Network Setup
- Troubleshooting
- STEP Monitoring Recommended Practices
- System Maintenance
- Backup and Recovery
- Security
- Live-to-Test Synchronization

Backup and Recovery

Database backup, restore, and recovery are critical processes for any business in the marketplace.

Database Backup Strategy

The backup strategy depends largely on the amount of time the system is permitted to be offline after any type of system crash.

A well-implemented database backup and recovery strategy is the cornerstone for every deployment, making it possible to restore and recover all or part of a database without data loss.

In the Oracle landscape, there are several options available to support backup strategy. The basic concepts of four well-known strategies are described in the subsections below.

Cold database backup (offline backup)

In order to do an offline backup, it is necessary to shut down the database because the files must remain unchanged. All database files, redo logs, and control files are backed up to a secure medium.

It is not common to use cold database backups as a backup strategy because it is not possible to perform a point-in-time recovery; the database is unavailable during backup, and with the Oracle database shut down, it means that all in-memory information is lost.

Hot database backup

The 'hot backup' strategy supports backing up a running database. It is a three-phase, manual process, where the tablespace is put in backup mode - that is, issue the `alter tablespace <TABLESPACE_NAME> begin backup`; statement, then back up the data files of the tablespace by OS copy, and lastly, take the tablespace out of backup mode — `alter tablespace <TABLESPACE_NAME> end backup`;

It should be noted that Oracle will continue to write to the database's data files even if a tablespace is in backup mode. In this case, Oracle will write out full change blocks to the redo log files. Normally, only deltas (changes) are logged to the redo log files. This is done to enable reconstruction of a block if only half of it was backed up, causing a split block. Because of this, there should be an increase in log activity and archiving during online backups.

When the procedure for all permanent tablespaces is complete, make an archive log switch, create a control file backup, and copy all archive logs and control files to the backup destination. This approach will back up data files, control files, and archive files. During the restore process, the data files affected by the failure and all the associated archive files should be restored.

The advantage of this backup strategy is that you can still work on the database while it is being backed up, meaning that the 'normal' window for backup is expanded time-wise. Hot database backups are also an optimal choice if SAN snapshots are being used as part of the overall strategy.

Database backup using Oracle Recovery Manager (RMAN)

The Oracle Recovery Manager (RMAN) backup tool backs up, restores, and recovers database files; and supports incremental backups. This approach not only optimizes backup times but also improves recovery operations since RMAN intelligently decides what combination of full, incremental, and archive-log backups will produce the optimal recovery path.

Recovery Manager maintains information about the backup files in either the recovery catalog or the control and archive redo log files, thereby freeing the DBA from having to track all the backup copies and archive logs (except for keeping track of tapes). It then uses the stored information about the backups to automate both restore and recovery operations so that there is no chance of accidentally restoring from the wrong backup.

The advantages of using RMAN compared to hot database backups are:

- It is part of Oracle's core software and it handles the entire backup - in other words, both the reading and writing of data blocks, which makes it more efficient than the manual OS copy methods used for hot database backups.
- It administers the backups from a repository, which is stored in the control file.
- It handles both restore and recovery.

Note: The only reasonable backup strategies for a critical system are online backup methods. Though these methods can run online, they will typically degrade performance. For this reason, backups should ideally be performed during maintenance windows only.

Disk backup

This backup strategy is a complete backup of each individual hard disk. This process could be very time consuming, depending on the number of disks. For this reason, this technique will normally be more suitable as an application backup strategy.

Backing up to disk first and then to tape

It is highly recommended that the database be backed up to a designated disk partition, and afterward be written to tape for long-term storage and disaster recovery purposes.

Storing tapes off site

The ideal backup strategy ensures that there are multiple tapes available in the event an initial tape is bad, corrupted, or similar. In addition, tapes should be stored in multiple, off site safety facilities to ensure backup availability at all times.

Test restore frequently

It is good policy to assume that a backup that has not been restored and tested is not a good backup. In other words, it is bad practice to overwrite a tape before having tested that a more recent backup can successfully be restored.

Exclude general OS backup from online Oracle files

Backing up the Oracle database is tricky and should only be done with the provided hot backup scripts or third-party backup tool that has database plug-ins. Attempting to back up Oracle data files while Oracle is running will cause it to shut down fatally if a file is locked by backup software during file copy. Oracle data files can be backed up safely only when Oracle is shut down.

The only exception to this rule is the hot backup script, which uses special functionality provided by Oracle to safely copy files while the database is running.

Backup responsibilities

It is the responsibility of the system administrator to perform the following backup tasks:

- Ensure that the database backup script is run every night (or other agreed-upon schedule) and that the content of the 'backupspace' partition is backed up to tape every night (or other agreed-upon schedule) after the script has finished.
- Make sure that the generated backup log files do not contain any errors that may indicate that the backup was not successful.
- Be certain that the area designated for the hot backup resides on disks that have enough storage capacity to hold the backup.
- Be sure that the hot backup folder includes all files necessary to perform a complete Oracle database restore.
- Certify that all the files in the hot backup folder are backed up to tape, that they can be restored, and that the Oracle database can be completely restored from said tape.
- Restore the contents of the hot backup folder (in the event of a system disaster) to hardware that is similar in configuration to that of the failed system.

Stibo Systems Recommendations

Stibo Systems recommends the following storage layout combined with one of the following backup strategies.

Storage layout

Storage area should be divided into four separate areas:

1. Operating System (OS)
2. Other software (SW), for example STEP and Oracle
3. Database files only (DB)
4. Online backups (staging area)

No OS / SW files should be placed on volumes used for DB / staging and vice versa. The staging area must be large enough to hold a full backup of the database – including the archive log files and incremental backups.

Strategies

1. Hot backup is performed using the script mentioned in the **RMAN backup script** section below. The script uses RMAN for performing an RMAN hot backup to disk (staging area). This backup must then be backed up during the file system backup.
2. Backup is performed using third-party backup software (BSW). For example, NetVault, Legato, or Veritas. When doing so, backups can be taken in two ways:
   - Using built-in features in BSW, which are integrated with RMAN, allow RMAN to perform hot backups directly to tape device. This will enable you to save the extra costs of creating a staging area.

     A disadvantage of performing backups this way is the price of the license for using RMAN via BSW, as this license is usually expensive. Another disadvantage of using BSW for backups is that it requires the software to be present on the target environment in case the database is to be cloned / restored on other servers.

     A third disadvantage of this approach is that backing up directly to tape will most likely increase the restore time.

   - Using an RMAN script to perform hot backups to disk (staging area), and thereafter, utilizing the BSW to back up the OS, SW, and staging area volumes to tape device.

Whether daily / weekly database backups are to be performed as full or incremental backups depends on the size of the database, the backup window, and the time given to restore / recover the database in case of disaster. In general, Stibo Systems recommends RMAN for internal database backups, but in practice, the scheduling and type of backups is to be decided based on the Service Level Agreement (SLA), database size, and backup window.

Note: For special-purpose backups – for example, if a database copy needs to be sent to Stibo Systems for implementation – or support-related activities, the recommended strategy is using STEP's own export / import utility, STEP DB Toolbox, which can be found in the STEP DB Toolboxes Guide and Examples documentation.

Backup strategies are always subject to discussion since it is dependent on various parameters such as: database size, server architecture, storage capacity, and uptime requirements. Regardless of what backup strategy is chosen, it should always be carefully tested before being considered a reliable backup.

Backup Procedures

Focusing on the STEP system, the backup procedure must be designed to back up enough data from the system to restore any type of stored application data hierarchy.

RMAN backup script

Stibo Systems has written a hot backup script called `rman_backup.pl` as an alternative to a Client / Customer written one. However, upon first use – including, but not limited to configuration of the Stibo Systems script, the Client / Customer understands and accepts the disclaimer below:

When using the Stibo Systems backup script:

- All backup files are placed in a sub directory on the partition labeled 'backupspace' – also known as the staging area (the default path is `/backupspace/<ORACLE_SID>`).
- An RMAN restore script is generated.

Important: When using the `rman_backup.pl` script, the client / customer accepts that Stibo Systems cannot be liable for any use whatsoever of the script. Therefore, it is also the sole responsibility of the client / customer to ensure that the result of any backup is fully operational and useful after changes made to the aforementioned script or the STEP software – including, but not limited to patching, updating, configuring, etc.

File systems that must be backed up

To be able to restore the system, the following file systems must be backed up during cold backup:

File System (default)	Description
Local disk	System volume
`/home` `/opt`	Contains STEP installation and third-party software
`/workarea`	Workarea for STEP
`/upload`	Hotfolders for STEP
`/backupspace`	Oracle RMAN backup

The following file systems can optionally be backed up during cold backup:

File System (default)	Description
`/database/db1`	Oracle data files containing the database files and control files
`/database/db2`	Oracle data files containing the database files and control files
`/database/db3`	Oracle data files containing the database files and control files
`/database/dbredo1`	Oracle online redo log files
`/database/dbredo2`	Oracle online redo log files

Backup frequency

Stibo Systems recommends extending the chosen strategy to include some sort of backup covering the entire system – especially the installed software and its configuration information. The minimum recommended backup schedule is as listed below.

Data type	How often	Notes
OS	When changed, minimum once a week	
Applications	When changed, minimum once a week	Normally located on the `<STEP_HOME>` volume (default is `/opt`).
Configuration	When changed, minimum once a week	Normally located on the `<STEP_HOME>` volume (default is `/opt`).
Database	Daily	Complete including data, configuration, and archive logs. Use hot backup scripts.

Recovery

There is no easy answer to a recovery strategy since it depends on the type of failure. For example, a disk crash, a corrupt file system, a bad block in the database, etc. can have different solutions.

In any situation where there is a system failure, Stibo Systems strongly recommends contacting Stibo Systems Technical Support in order to recover as quickly and seamlessly as possible. As an additional resource, the backup script mentioned in the **RMAN backup script** section generates an RMAN restore script for use when restoring and recovering the database is required.

Benchmark Facility

To ensure a STEP environment meets Stibo Systems' operating requirements, SPOT provides a mechanism to run a benchmark on an application server that measures I / O throughput against associated file systems and network latency between STEP's application server(s) and the database server.

Typically, Stibo Systems Technical Support will request that these tests be run if they feel that the STEP system is not performing optimally. Results from these tests are automatically sent to Stibo Systems via the Updates server.

On Linux platforms, note that the benchmark tool is dependent on the FIO package, which can either be installed via a YUM repository or downloaded from Stibo Systems Updates server if not currently present on the system. If the package is not installed, SPOT will prompt you with instructions. Because it will take several minutes to perform the benchmark tests, it is best to execute them when STEP is not running.

Run SPOT benchmark

1. Go to the directory where STEP is installed. The default path is `/opt/stibo/step`
2. Run the SPOT command as follows:

   ```
   ./spot --benchmark
   ```

Healthcheck Facility

SPOT provides a customized healthcheck facility upon start-up of the STEP Application Server. Customized scripts can be developed to test upstream or downstream services that have a dependency on STEP. If a healthcheck fails, the application server will fail to start up.

Scripts placed in `/admin/healthcheck.d` of the directory where STEP is installed are executed by SPOT before attempting to start the application server.

The healthcheck scripts must conform to the following rules:

- Failed checks must exit with a code >0; otherwise, the script needs to exit with a code of 0.
- Multiple scripts will be executed in ascending order based on the script name.
- Ensure script output is written to STDOUT as this will be captured and written during execution to `/diag/logs/pre-start-toolbox.app.jar.log` of the directory where STEP is installed.

Live-To-Test Synchronization

There are several steps involved in synchronizing a source system to a target system. In most cases, this would be from a live (production) system to a test system. If requested, Stibo Systems can provide a detailed, customer-specific document.

This documentation is considered a billable service, which is not included in the standard support agreement. Be aware that certain requirements to the STEP setup must be met for Stibo Systems to be able to deliver this service. Contact Stibo Systems for more information.

Back up the source system

The database and file system must be backed up. This can be done according to the procedures described in the **System Maintenance** documentation or any other backup procedure currently used by your organization.

Restore the database on the target system

The database must be restored on the target system. The specific approach needed to do this depends on the strategy used to back up the database in question. As per the **Backup and Recovery** documentation, some examples of backup strategies are:

- Oracle Import (import or DataPump import)
- Oracle RMAN restore
- Restore of cold backup

Copy the source configuration to the target server

The target system must be identical to the source system, and therefore, the configuration on the target system must match that of the source – except for those parameters that are server specific.

Patch the target system

Unless the entire directory where STEP is installed has been restored from a backup on the source system, the target system should be patched to the same STEP patch level as the source. For patching the STEP system, refer to the **Patching STEP** documentation.

Managing the STEP Application Server

The primary tool for managing a STEP Application server, either a single server or within a clustered STEP environment, is the Stibo Patch Operations Tool (SPOT). The SPOT tool offers the following capabilities:

- Start and Stop Procedures
- Verify the Installed STEP Version
- Send a Diagnostic Snapshot to Stibo Systems
- STEP System Configuration
- Health Check Facility
- Benchmark Facility
- Patching STEP

The majority of SPOT's capabilities require access to Stibo Systems' Updates Server (https://updates.stibosystems.com), either directly or via a SPOT mirror. Direct access via proxy is supported, but does require additional properties to be manually added to SPOT's properties file.

Configure SPOT to use a proxy server

1. Go to the directory where STEP is installed. The default path is `/opt/stibo/step`
2. Edit the file, `spot.properties`, and append the following properties with values that correspond to the proxy's requirement:

```
Proxy.HTTP.Host=<PROXY_HOST>
Proxy.HTTP.Port=<PROXY_PORT>
Proxy.HTTP.User=<PROXY_USERNAME>
Proxy.HTTP.Password=<PROXY_USERNAME_PASSWORD>
```

Test the connectivity to STEP's Updates Server

1. Go to the directory where STEP is installed. The default path is `/opt/stibo/step`
2. Run the SPOT command as follows:

```
./spot --ping
```

Network Setup

All communication between the STEP Application Server and the STEP Client is based on the HTTP / HTTPS protocol. The same goes for the DTP InDesign clients communicating with the PIM application through the respective DTP extensions. The default port is `tcp/80`, but this can be changed in the `sharedconfig.properties` file via the `System.ServicePort` property. If SSL is enabled, the default port is `tcp/443`, but this default can also be changed via the `System.SSLServicePort` property.

Administration

The best way to access the Linux server setup is by using an SSH via port `tcp/22`. Alternatively, VNC or other X server utilities can be used.

HTTP server vs. application server

The system is always set up to run an Apache HTTP proxy as a front end to the application server. The HTTP server receives the incoming requests from the PIM or DTP clients and forwards them to the STEP application server, which then receives the responses and then passes them on to the client applications.

STEP uses Apache 2.4.37 HTTP Server (on RHEL / OEL 8.x) and Apache 2.4.6 HTTP Server (on RHEL / OEL 7.x.)

Important: This HTTP proxy is running on the application server itself as part of the STEP application and cannot be separated or removed. If a proxy is required in a DMZ, this must be an additional Apache instance that proxies all requests to the local STEP HTTP proxy.

DTP server

The back-end DTP Server (Adobe InDesign®) listens for requests from the STEP application server on the port specified by the `InDesign.Port` property (in the `sharedconfig.properties` file.)

The URL is not configured explicitly, since this is part of the sidecar deployment JAR file generated when adding the `InDesign.Instances = <DTP_SERVER_HOSTNAME>` property.

The response is delivered back to the STEP application server from the DTP Server, which uses the application server's URL – taken from the `System.Servers` property that is also present in the sidecar's deployment JAR file.

Patching STEP

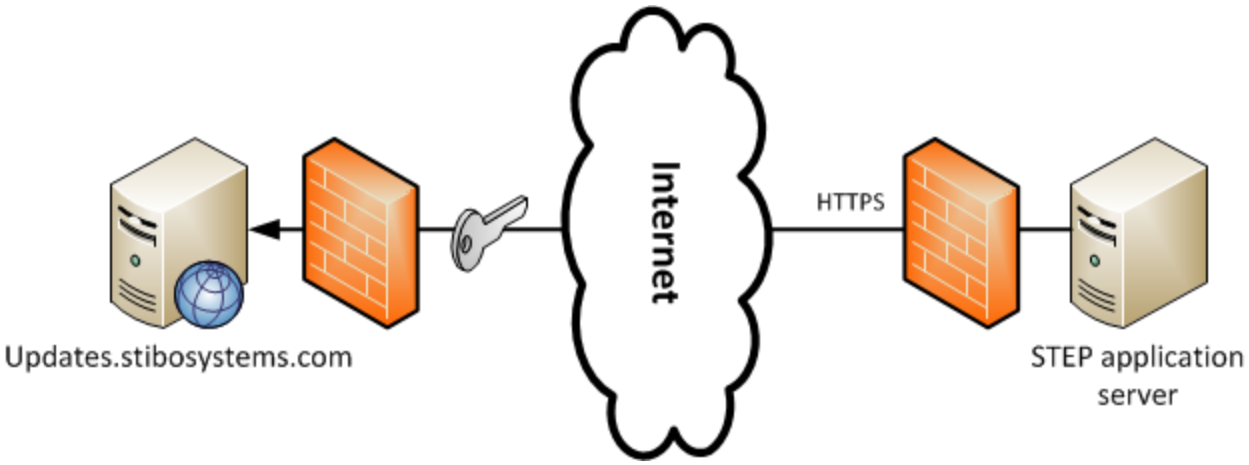
In STEP, the architecture is split up into components, with each component having the ability to access other components through a new set of component APIs. This component-based architecture satisfies otherwise contradictory requirements for longer time between releases, and faster introduction of new improvements.

By using components, it is possible to keep the core and other components unchanged for a longer period of time while choosing to upgrade selected components to take advantage of new features and/or important updates within those components. Thus, the risk and workload involved in testing new updates is reduced.

There are three methods that can be used to apply a STEP patch:

Direct Connection method

The default method for patching is the Direct Connection method, where the STEP environment is configured to directly allow an encrypted connection via HTTPS to the release (updates) server. This method offers the recommended security.

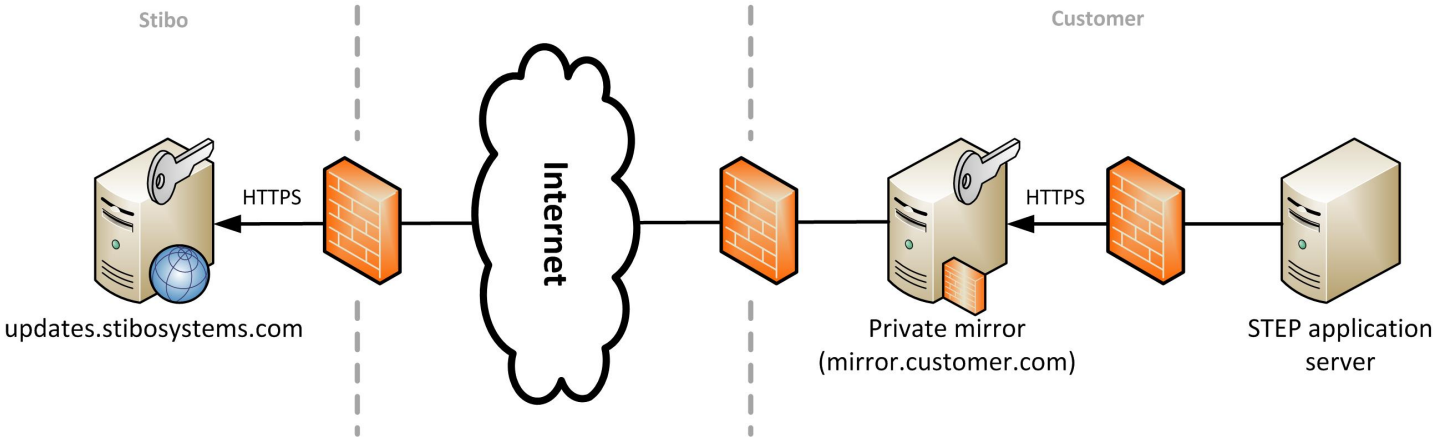


The advantages of using the Direct Connection method are:

- Faster support from Stibo Systems by providing complete version information.
- Simplified infrastructure.

Private Updates Mirror method

As an alternative to accessing the release server directly, it is possible to set up a Private Updates Mirror and configure SPOT on the internal STEP servers to use this mirror instead.



The advantages of using the Private Updates Mirror method are:

- If the internet connection or the global updates server breaks down, the files that have already been downloaded will still be available.
- The consumed internet connection bandwidth is reduced by avoiding repeated downloads.
- The network configuration is simpler since only the mirror requires access to the updates server, while the individual SPOT instances can be configured to talk only to the private mirror on the internal network.

Offline method

In some cases, especially production environments, access to Stibo Systems' release server directly or via a private updates mirror is not permitted often due to stringent security policies. In this scenario, the only acceptable method available to patch a STEP environment is to use the Offline method.

The Offline method requires SPOT to be installed on a remote Desktop / Server that has access to Stibo Systems' release server, but not necessarily a direct connection to the required STEP environment(s) that need (s) patching.

The Offline method is a manual, three step process:

1. Using SPOT, take a snapshot of the STEP environment that requires patching and transfer the snapshot file to the remote desktop / server that has access to Stibo Systems' release server.
2. On the desktop / server, use SPOT to prepare the required patch files using the supplied snapshot and transfer them back to the STEP environment.
3. On the application server, apply the patch files to the environment using SPOT.

Use of the Offline method is not recommended based on the following information:

- Additional hardware in terms of the SPOT Desktop / Server is required.
- Risk that the thin snapshot found on the SPOT Desktop / Server is not recent and thereby results in downloading / applying and out of date patch.
- A more complex infrastructure is needed.

- Increased risk of human error due to the presence of more commands to execute.
- More manual work / automation is required.

Using SPOT to Patch a STEP System

It is recommended to test the connectivity to Stibo Systems' release server prior to attempting to patch the environment.

Patch STEP on all servers in the STEP cluster to the latest release

1. Go to the directory where STEP is installed. The default path is `/opt/stibo/step`
2. Using the appropriate substitutions, run the following SPOT command:

   `./spot --apply=to:step/platform/step-<STEP_VERSION>-[mp<LEVEL>-]<RELEASE_TIMESTAMP>.spr`

Example:

```
./spot --apply=to:step/platform/step-11.1-mp5-2023-11-08-15-54-27.spr
```

It should be noted that the SPOT command above shows the general patch recipe format – i.e., the STEP version, monthly maintenance patch level (if applicable), and the release timestamp – for informational purposes. In most cases, Stibo Systems will provide the necessary recipe for your situation.

Use the `--local` option to patch individual servers one at a time. All servers in the cluster MUST be at the same STEP release.

Applying the latest release of STEP or a monthly maintenance patch (mp) will restart the STEP application, so be sure to plan for a bit of down time or perform these only during a proper maintenance window.

Patch a specific STEP component

In the example below, the RESTAPI component is being patched to 7.0.40.

1. Go to the directory where STEP is installed. The default path is `/opt/stibo/step`
2. Run the SPOT command as follows:

   `./spot --showupdates` (This is an optional step to verify the latest component release.)

   ```
   ./spot --apply=to:restapi/7.0/restapi-7.0.40.spr
   ```

Certain STEP components can be applied using the `--hot` option. The Hot option allows a component to be upgraded without restarting STEP. Only certain components support the Hot option so to determine whether a component can be applied “Hot”, run the following command:

Determine whether a component can be applied hot

Some components can be applied hot, allowing the update to run without having to restart the STEP server.

1. Go to the directory where STEP is installed. The default path is `/opt/stibo/step`

2. Run the SPOT command as follows:

   `./spot --prepare=<COMPONENT_RECIPE>`

If the component can be applied hot, the following message will be returned:

Notice: This upgrade can be applied without restarting STEP. To do so, rerun prepare using `--hot` when applying the recipe. `--hot` ensures that the system is not restarted by accident.

Patch STEP using the Offline method

The following example details the steps required to patch a STEP environment to the latest release using the Offline method. The same steps can be applied when patching a specific STEP component, hotfix, or customer add-on.

Important: Since you will typically use a single SPOT Desktop / Server to manage multiple STEP environments, the Offline method introduces the possible risk of downloading an update based on the thin snapshot from one STEP environment and applying it to another by mistake. Therefore, extreme caution must be exercised in order to not mix up the environments.

Patch an offline STEP environment to the latest release

Take a snapshot of the STEP Application Server:

1. Go to the directory where STEP is installed. The default path is `/opt/stibo/step`
2. Run the SPOT command as follows:

   `./spot --snapshot=<THIN_SNAPSHOT_FILENAME>.spr`

3. Copy the thin snapshot file to the SPOT Desktop / Server.

Prepare a patch file from the snapshot on SPOT Desktop / Server:

1. Go to the directory where STEP is installed. The default path is `/opt/stibo/step`
2. Using the appropriate substitutions, run the following SPOT command:

   `./spot --accessmode=offline --apply= to:step/trailblazer/step-trailblazer-<STEP_VERSION>-[mp<LEVEL>-]<RELEASE_TIMESTAMP>.spr --target=<THIN_SNAPSHOT_FILENAME>.spr --output=<PATCH_FILENAME>.spr`

3. Copy the prepared patch file to the offline STEP Environment.

Apply the patch on the STEP Application Server:

1. Go to the directory where STEP is installed. The default path is `/opt/stibo/step`
2. Run the SPOT command as follows:

   `./spot --apply=<PATCH_FILENAME>.spr`

Security

It is the responsibility of the system administrator to establish a security policy or extend an existing one to include STEP.

Antivirus

Running antivirus software on the STEP server(s) is encouraged – except in the cases described below. Scanning the filesystems should be done outside office hours.

Filesystem exclusions

The filesystem exclusions described are for both 'on-demand' scans and the 'on-access' scans feature. On-demand scans may be run if the system is offline. On-access scans must be completely disabled.

Files	Filesystems (default)	On-demand scans	On-access scans
Database Server			
Oracle Data files	`/database/db1`	When the database and listener are offline	Never
Online redo log files	`/database/dbredo1` `/database/dbredo2`		
Archive log files	`/database/dblog`		
Database backup	`/database/backupspace`		
$ORACLE_BASE	`/opt/oracle`		
Application Server			
Software installation directories	`/opt`	When STEP and third-party software are offline	Never
WorkArea	`/workarea`		
Upload	`/upload`		
Temp	`/tmp`		

Note: In a single server setup – i.e., the database and application are on the same machine, all filesystems listed above must be excluded.

Send a Diagnostic Snapshot to Stibo Systems

SPOT can be used to pull diagnostic information about the STEP environment and subsequently upload the snapshot automatically so that Stibo Systems Support can quickly identify any issues. If a Jira ticket has already been created with Stibo Systems, there is an option to assign the snapshot directly to a ticket number.

Note: To use this capability, the STEP environment must be able to access Stibo Systems' updates server (https://updates.stibosystems.com).

Pull all diagnostic information and upload it to Stibo Systems

1. Go to the directory where STEP is installed. The default path is `/opt/stibo/step`
2. Run the SPOT command as follows:

   ```
   ./spot -d
   ```

   or alternatively,

   ```
   ./spot --diag
   ```

Pull all diagnostic information and store it in a file

1. Go to the directory where STEP is installed. The default path is `/opt/stibo/step`
2. Run the SPOT command as follows:

   ```
   ./spot -d --output=<FILE_NAME>
   ```

   or alternatively,

   ```
   ./spot --diag --output=<FILE_NAME>
   ```

Pull all diagnostic information and include profiling data or heap dumps

1. Go to the directory where STEP is installed. The default path is `/opt/stibo/step`
2. Run the SPOT command as follows to include Profiling Data:

   ```
   ./spot -d --target=profiling
   ```

   or alternatively,

   ```
   ./spot --diag --target=profiling
   ```

3. Run the SPOT command as follows to include Heap Dumps:

   ```
   ./spot -d --target=heapdumps
   ```

   or alternatively,

```
./spot --diag --target=heapdumps
```

4. Run the SPOT command as follows to include All Data:

   ```
   ./spot -d --target=all
   ```

   or alternatively,

   ```
   ./spot --diag --target=all
   ```

Additional options

- `--issue=<JIRA_ISSUE>` – the Stibo Jira issue (e.g., STIBO-1) to which this diagnostics snapshot relates
- `--history=<TIME_PERIOD>` – sets how far back in time to collect diagnostics; the default is 1 day. The history option takes days, hours, minutes, and seconds – e.g., 2d, 3h, 60m, 86400s

Start and Stop Procedures

As indicated in the system architecture diagram (refer to the **STEP Architecture Guide**), the dependencies between the components dictate the start-up procedure. Following this procedure ensures the system starts and functions properly.

The normal sequence of starting the main components of the system is:

1. Oracle Database
2. STEP Application
3. HTTP Web Server

The SPOT script placed in the directory where STEP is installed handles the start / stop procedure of the application server. This includes all application servers in a STEP cluster.

Note: Except for other STEP application servers in a cluster, SPOT does not handle start / stop sequences on remote servers. If a DTP server exists, it will run on a Windows or a Mac server as a service and start up automatically on server boot.

The following start up options are available and are defined in the sections below:

- Starting all application components in a single or clustered setup
- Starting the application component (STEP) in a single or clustered environment
- Starting STEP on a single server in a clustered environment
- Stopping all components except the database
- Checking the status of all components

Start all application components in a single or clustered setup

1. Go to the directory where STEP is installed. The default path is `/opt/stibo/step`
2. Run the SPOT command as follows:

```
./spot --start
```

Start the application component (STEP) in a single or clustered environment

1. Go to the directory where STEP is installed. The default path is `/opt/stibo/step`
2. Run the SPOT command as follows:

```
./spot --local --service=app --start
```

Start STEP on a single server in a clustered environment

1. Go to the directory where STEP is installed. The default path is `/opt/stibo/step`

2. Run the SPOT command as follows:

   ```
   ./spot --local --service=app --start
   ```

Note: In this scenario, it is possible to start / stop only the STEP component on the currently logged-in node.

Stop all components except the database

Note: The database can only be controlled in a single server environment, in other words, where both STEP and Oracle are running on the same server.

1. Go to the directory where STEP is installed. The default path is `/opt/stibo/step`
2. Run the SPOT command as follows:

   ```
   ./spot --stop
   ```

Note: Refer to the **System Shutdown** section in the **Troubleshooting** documentation if the stop command is hanging.

Check the status of all components

1. Go to the directory where STEP is installed. The default path is `/opt/stibo/step`
2. Run the SPOT command as follows:

   ```
   ./spot --status
   ```

STEP Monitoring Recommended Practices

It is important to monitor STEP for reliability and performance efficiency. These are the recommended practices for monitoring STEP and integrating with third-party management system tools. The following topics are discussed in this section:

- STEP Sensors
- Log File Monitoring
- Port Monitoring
- Integrating STEP with Application Performance Monitoring (APM) Tools
- Monitoring STEP Using Amazon Web Services
- Monitoring STEP Using STEP's REST API

STEP Sensors

STEP Sensors provide the ability to monitor the availability and performance of STEP and are the preferred and simplest methods for integrating with third-party management system tools. There is a vast array of STEP Sensors that can be used to monitor various aspects of STEP and related dependencies. For example, below is a list of available sensor categories:

- Java heap usage
- HTTP
- Event queues
- Schedules jobs
- STEP Web Services API
- Security

Each sensor provides a separate URL that can be integrated with any monitoring tool that can make an HTTP request and parse the response. These sensor URLs do not require authentication and provide three output formats:

1. Simple Status String ('OK,' 'WARNING,' and 'CRITICAL')
2. Nagios Plugin Output
3. Full XML

The format used will depend on the monitoring tool's level of sophistication and the ability to parse the output.

Example:

To monitor and integrate the HTTP sensor, Http-local

1. Use this sensor URL for simple output:

```
http(s)://<APPLICATION_SERVER_NAME>/admin/monitoring/Http-local/status
```

2. Use this sensor URL for Nagios plugin output:

```
http(s)://<APPLICATION_SERVER_NAME>/admin/monitoring/Http-local/nagios
```

3. Use this sensor URL for full XML output:

```
http(s)://<APPLICATION_SERVER_NAME>/admin/monitoring/Http-local/xml
```

Note: In general, the URL format used to access a given sensor is as follows: `http(s)://<APPLICATION_SERVER_NAME>/admin/monitoring/<SENSOR_NAME>/<SENSOR_OUTPUT_FORMAT>`, where <SENSOR_NAME> is the sensor's name as it appears on the http(s)://<APPLICATION_SERVER_NAME>/admin/monitoring page and the <SENSOR_OUTPUT_FORMAT> is status, nagios, or xml.

You can review the available sensors or check the status of a sensor by logging into the STEP Administration Portal, selecting the Monitoring tab, and clicking the *Sensors for external monitoring* link at the bottom of the page. For further documentation on accessing the Administration portal – or STEP Sensors in general, refer to the *STEP Administration Portal* section.

Although sensors cannot be customized, it is possible to modify the thresholds of certain categories of sensors. Thresholds are set by adding additional properties and values to STEP's property files. Refer to the **STEP System Configuration** section for additional details on this topic.

The table below provides a matrix of configurable sensors and their associated properties:

Properties (Listed by Category)	Threshold / Item (Default)
Event Queue Sensor	
`Monitor.EventQueue.NoOfUnreadEvents.Critical`	100000
`Monitor.EventQueue.NoOfUnreadEvents.Warning`	80000
File System Sensor	
`Monitor.FileSystem.Critical`	150 ms
`Monitor.FileSystem.Warning`	50 ms
STEP Scheduler Sensor	
`Monitor.ScheduledBackgroundProcess.MaxEndedAgeInDays`	-1

Properties (Listed by Category)	Threshold / Item (Default)
`Monitor.ScheduledBackgroundProcess.ReportWaitingAsWarning`	TRUE
Oracle Sensor	
`Oracle.Sensor.AlertLog.Critical`	None
`Oracle.Sensor.AlertLog.Warning`	None

Log File Monitoring

All STEP application logging is managed in a file in a central location on each application server. Housekeeping (file rotation, size of log files, and history) are configurable through STEP properties. By default, STEP will keep 20 log files in total, including the current log, and each one can be a maximum of 10 MB in size.

The default location for STEP logs is `<STEP HOME>/diag/logs` and the most current log will be `step.0.log` (the `step.1.log` -> `step.19.log` are the rotated logs).

Logging verbosity is set to 'INFO' by default, which means that all informational, warning, and severe actions will be logged.

Available STEP properties to manage log files

1. Log Verbosity

   `Log.Level=<FINEST|FINER|FINE|CONFIG|INFO|WARNING|SEVERE>`

Default: `INFO`

2. Log Location

   `Log.Root=<PATH>`

Default: diag/logs

3. Log Size

   `Log.Size=<INTEGER_VALUE> (in MB)`

Default: 10 MB

4. Log Count

   `Log.Count=<INTEGER_VALUE>`

Default: 20

Port Monitoring

The table below represents key TCP ports that are used by STEP and are available for monitoring:

TCP Port (Default)	Component	Server
80	Apache	STEP Application Server
443	Apache	STEP Application Server
9870	STEP Application	STEP Application Server
5636	STEP Cluster	STEP Application Server
22	Secure Shell (used by SPOT)	STEP Application Server
1521	Oracle Database	STEP Database
9090-9093	InDesign Sidecar	InDesign Server

Integrating STEP with Application Performance Monitoring (APM) Tools

To integrate STEP with APM tools - such as New Relic and Computerware's Dynatrace - STEP typically requires an external Java agent to be loaded alongside STEP's JVM. This can be achieved by using STEP's standalone.JVMArgs property along with any required parameters, path, and agent.

The following is an example:

Integrate STEP with new Relic APM

Add the following property to the **config.properties** or **sharedconfig.properties** file:

```
Standalone.JVMArgs= -javaagent:/<PATH>/newrelic.jar
```

Important: Stibo Systems does not provide any support with loading external agents alongside STEP's JVM that might become unstable or cause instability in the STEP application itself.

Monitoring STEP Using Amazon Web Services

Amazon CloudWatch is a monitoring service for Amazon Web Services (AWS) resources and hosted applications that provides real-time, system-wide visibility into resource utilization, application performance, and overall operational health. Amazon CloudWatch can monitor AWS resources by providing out of the box metrics for Amazon EC2 instances, Amazon S3 volumes, and Amazon RDS Database instances, as well as the ability to generate and collect custom application metrics, either directly or indirectly via collected log files.

With the current trend of IT provisioning more and more IT services within AWS, Stibo Systems has recognized the need to tightly integrate its current application monitoring and performance capabilities with AWS's CloudWatch offering by providing an AWS toolkit. The Stibo AWS Toolkit is a suite of utilities designed to integrate STEP's application metrics, application logs, database tablespace usage, and statistics with CloudWatch's functions for storing and visualizing metrics and defining alert based notifications.

The Stibo AWS Toolkit offers the following functionalities:

- The ability to consume key log files generated by the STEP application into CloudWatch logs, enabling monitoring in near real time across multiple STEP instances for diagnosing and centralizing log management.
- A mechanism for consuming Oracle's alert log into CloudWatch logs for centralized log management across multiple Oracle instances.
- Custom metrics based on STEP's database tablespace usage that can be collected and viewed through CloudWatch.
- A standard set of filtering metrics that provides custom metrics from STEP-generated log streams for collecting, visualizing, and defining alarms.
- An AWS Lambda function for remotely monitoring STEP's sensors.
- An architecture that enables it to run on STEP's application server, be delivered and applied / upgraded seamlessly using Stibo's Patch Operation Tool (SPOT).

For further information on using the AWS Toolkit, refer to the **Integrating STEP with AWS CloudWatch** document.

Monitoring STEP Using STEP's REST API

Managing STEP Integration Endpoints (IEP) via external enterprise schedulers, such as BMC Control-M and Tivoli Workload Scheduler, is common practice within organizations. These organizations want the ability to schedule, control, and monitor batch processing where there are often dependencies between STEP and upstream / downstream systems.

Out of the box, STEP provides the ability to invoke an IEP remotely utilizing STEP's REST API. However, this needs to be achieved programmatically (not covered by this guide) by first sequentially combining several API calls, invoking the required IEP queue, and continuously monitoring the status of the triggered background processes to its completion.

Note: STEP's REST API is a licensed component.

The following example details the required REST API requests to invoke an Integration Endpoint, gather information about the background process (BGP), and monitor the status of said process by its ID. For added security, STEP's REST API requires authentication.

To access a STEP server's REST API, the required URL is formed by the protocol (http or https), the STEP application server name, and the corresponding REST API request URI, as displayed below:

```
http(s)://<STEP_APPLICATION_SERVER>/<RESTAPI_URI>
```

Invoking an integration endpoint

When invoking an IEP, the `<RESTAPI_URI>` should be of this format:

```
restapi/integrationendpoints/<QUEUE_NAME>/invoke?context=<CONTEXT>
```

Note: Adding a context may not be required and depends on the configuration.

Retrieve background process information associated with integration endpoint queue

When retrieving information about an IEP's background process, the following format should be used for the `<RESTAPI_URI>`:

```
restapi/integrationendpoints/<QUEUE_NAME>/backgroundprocesses?context=<CONTEXT>
```

Note: Adding a context may not be required and depends on the configuration.

Retrieve the status of a specific background process by ID

When checking the status of an IEP's background process, the `<RESTAPI_URI>` format should be as follows:

```
restapi/bgpinstance/<BGP_ID>/status?context=<CONTEXT>
```

Note: Adding a context may not be required and depends on the configuration.

STEP System Configuration

SPOT provides the ability to manage STEP's properties files (such as config.properties and sharedconfig.properties) by checking syntax, searching available properties, encrypting security- exposed properties, and viewing / updating license information.

The STEP property files are key for configuring / customizing the STEP environment. For example, adding additional servers to the cluster, increasing STEP's JVM heap size, housekeeping, and managing background processes' queue sizes.

Each application server has its own property file, config.properties, that is used to customize STEP on the local application server only. A shared property file, sharedconfig.properties, is read from the config.properties file by using the `@include` option and is the preferred file to change when customizing a STEP environment, as the file is shared between all application servers when in a clustered configuration.

Additional property files can be included using the `@include` option and added to the config.properties file. Any changes made to sharedconfig.properties or additional properties files require the application's configuration to be updated (as displayed below).

Check the syntax of STEP's property files / update the configuration

1. Go to the directory where STEP is installed. The default path is `/opt/stibo/step`
2. Run the SPOT command as follows:

```
./spot --config
```

Note: If an unknown property has been added, an error will be displayed after executing the above command.

Search available properties and associated documentation

1. Go to the directory where STEP is installed. The default path is `/opt/stibo/step`
2. Run the SPOT command as follows:

```
./spot --prop=<SEARCH_STRING>
```

Encrypt a property value within a property file

1. Go to the directory where STEP is installed. The default path is `/opt/stibo/step`
2. Run the SPOT command as follows:

```
./spot --encrypt=<PROPERTY_NAME>
```

View license information

1. Go to the directory where STEP is installed. The default path is `/opt/stibo/step`
2. Run the SPOT command as follows:

   ```
   ./spot --license
   ```

Update license information

1. Go to the directory where STEP is installed. The default path is `/opt/stibo/step`
2. Run the SPOT command as follows:

   ```
   ./spot --updatelicense
   ```

Note: Prior to running the above command, any change in customer license must first be made internally by Stibo Systems personnel. Once that has been done, this command is then used to update the intended STEP environment(s).

System Maintenance

In order to maintain the system, there are several daily tasks and other routine tasks that are recommended. The following subsections list several routine tasks that must be performed in order to detect any system irregularities before they become disastrous.

Daily Tasks

- Check the backup logs to ensure that the backup to disk - as well as to tape - was successful.
- Check the Oracle alerts log on the database server for any possible errors related to Oracle. The alerts log is located in the `$ORACLE_BASE/diag/rdbms/<DB_UNIQUE_NAME>/<ORACLE_SID>/trace/alert_<ORACLE_SID>.log` directory. If any error is encountered, it is necessary to determine if the error is of a disastrous nature or if it can be ignored.
- Check the system logs for any possible system and/or application errors that are of a disastrous nature.
- Check the RAID utility (if applicable) for any errors related to disks.
- Gather statistics on tables owned by the stepsys user. This is automatically done by a scheduled Oracle job, which is created as part of the STEP installation.

Other Tasks

- To ensure that the Oracle database has been backed up properly, a test restore from backup should be performed regularly. The schedule may vary by the chosen backup strategy, but no tape should be overwritten before a more recent backup has been properly tested and successfully restored.
- Check tablespace allocations in Oracle to ensure that enough disk space is available for database operations. Refer to the **Oracle Database** section of the **Troubleshooting** documentation for details.
- Depending on the system load, archived log files can quickly accumulate and therefore, consume significant disk space. It is the responsibility of the system administrator to delete old archive log files.
- Gather statistics on indexes owned by the stepsys user. This is automatically done by a scheduled Oracle job, which is created as part of the STEP installation.

Database Maintenance

Database Maintenance related to performance is described in the STEP Database Maintenance Guide documentation.

Troubleshooting

This section contains some of the most common errors experienced during system start up and how to fix them.

System Start Up

Unless a specific error is suspected, follow the troubleshooting order below. The most common problems are caused by not properly following the start up procedures.

Operating system

It is common for start up issues to be caused by problems with the operating system and hardware. Below are a few basic checks that cover most problems caused by the operating system and hardware:

- Check the system logs for any errors or warnings (`/var/log/messages, dmesg`).
- Check RAID (if applicable) for any errors or warnings.
- Check for free disk space on all volumes (`df`).
- Check the memory usage (`top, prstat`).
- Check for processes that may be consuming the system's resources (`top, prstat`).
- Check that all network connections can be established (use `ping` and/or browse the network).
- Check the NFS mounted filesystems by testing the file locks (`flock`).

Note: Before proceeding with the following troubleshooting steps, make sure all STEP components are shut down.

Using SPOT when troubleshooting STEP components

The SPOT command can be used for checking the status of one or more components.

Oracle

The listing below shows the processes that must be running for Oracle to function properly:

- Oracle Listener (`tnslsnr`)
- Oracle Server Processes (`ora_XXXX_<ORACLE_SID>`)

```
ps -ef | grep oracle | grep ora_
```

Checking log files

The main Oracle log file is called the alert log and can be found in the following directory:

```
<ORACLE_BASE>/diag/rdbms/<DB_UNIQUE_NAME>/<ORACLE_SID>/trace/alert_<ORACLE_SID>.log
```

Note: By default, `<ORACLE_BASE>` is `/opt/oracle`, but may be different in some environments depending upon company standards.

STEP

The STEP application is started by the 'standalone' component as mentioned in the start and stop procedures. Standalone is the name used for STEP running directly on an Oracle Java or AdoptOpenJDK VM.

When STEP has been started, a Java process is running. This can be verified by running the following command:

```
ps -ef | grep [j]ava
```

Checking STEP log files

When starting the Java process, the following log files are generated:

Log file	Description
`<STEP_HOME>/diag/logs/startup-stderr.log`	Standard error output from starting the Java process is piped to this log file.
`<STEP_HOME>/diag/logs/startup.log`	Standard out from starting the Java process is piped to this log file.
`<STEP_HOME>/diag/logs/step.0.log`	Main STEP application log file. This file is rotated based on size (default 10 MB). By default, a maximum of 20 files are generated before overwriting the first file.
<*STEP_HOME*>/diag/gc/gc.log.0.current	Output from the Java garbage collector

System Shutdown

STEP

If the stop command is hanging, check the STEP log file on every node of the cluster. If the log is reporting that services are still running, then this could be a background process (BGP) that is not able to stop. In this case, you will need to manually kill the Java process on the node in question. You can get the id (pid) of the Java process by running the command below:

```
ps -ef | grep [j]ava
```

The output of the Java process will contain the word 'standalone' if you are using a standalone STEP server.

Once STEP is completely stopped, you must manually verify that all STEP database sessions are gone in the database. If there are any STEP database sessions they must be removed. If you have only stopped certain nodes in the cluster, then the database sessions in question are the ones which are established from these nodes.

STEP Administration Portal

The Administration Portal (commonly referred to as the admin portal) provides a number of useful tools, enabling administrators to support and troubleshoot STEP without physical access to the system. Some functions available within the admin portal are useful only for Stibo Systems Technical Support and/or R&D groups, while others are applicable for all system administrators. Specifically, the admin portal allows administrators to:

- View and download system information, including activities and processes running on the STEP application and their impact on the servers via the **Activity** tab.
- View the amount and types of requests made to the system, and how long it takes for the requests to be answered via the **Activity Dashboards** tab.
- View and download system logs via the **Logs** tab
- Monitor servers, events and components via the **Monitoring** tab - including providing data for external monitoring systems.
- View the system properties' configurations (such as the properties and values in the system property files) via the **Configuration** tab.
- Trace business rules, compare system configurations, generate a system snapshot, and test user authentication via the **Tools** tab.
- Perform healthchecks for common system issues via the **Health Check** tab.
- Send diagnostics and healthcheck information to Stibo Systems via the **Send Diagnostics** tab.
- Extract static texts for localization or interfaces and import translations via the **Localization** tab.

Logging on to the Administration Portal

Prerequisites

To access the admin portal, the person attempting to log in must have a user created in STEP that is a member of a user group with privileges that include the following setup actions:

- View Administration
- View Context

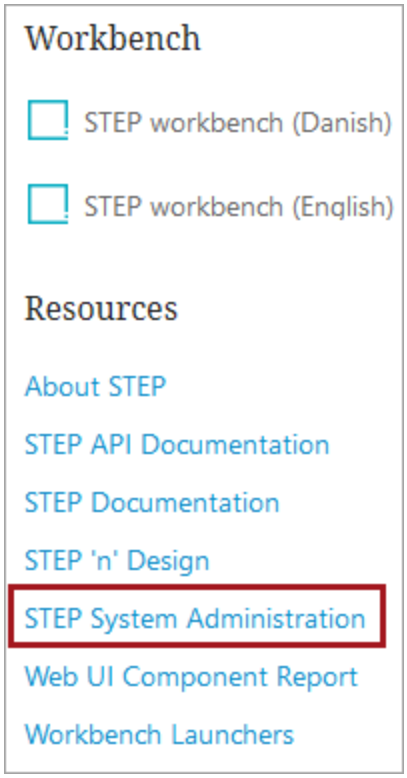
Accessing the Administration Portal

The admin portal can be accessed two different ways:

- Typing the appropriate URL into any browser:

```
http(s)://<APPLICATION_SERVER_NAME>/adminportal
```

- Clicking the **STEP System Administration** button from the Start page:



Both methods will yield a login prompt:

Valid credentials must be entered for login as indicated above in the **Prerequisites** section of this topic. Once logged in, the admin portal displays a series of tabs with various functions available on each one.

Refer to this online help page for further comprehensive documentation on the Administration Portal:

```
http(s)://<APPLICATION_SERVER_NAME>/help/#admin_portal/administration_
portal.html%3FTocPath%3DAdministration%20Portal%7C_____0
```

Oracle Database Troubleshooting

As part of a STEP implementation, Stibo Systems provides a database toolbox to assist administrators in diagnosing issues that might be related to the STEP database. There are two types of toolboxes available, a toolbox that resides on the application server for tasks related to database structure, session / instance information, maintenance, and reporting; and, an additional toolbox that resides on the database server for tasks that can only be resolved when executed from the database server. Both toolboxes are installed, updated, and patched using the SPOT utility just like any other STEP component.

In summary, the database toolboxes provide the following areas of functionality to assist in troubleshooting the STEP database:

- Tablespace management — for example, viewing usage information and adding additional table space files.
- Identification and management of database sessions.
- SQL Management — for example, checking SQL execution times, performance, and plans.
- Oracle events.

For more information on using Stibo Systems' Database Toolbox, refer to the related **STEP DB Toolboxes Guide and Examples** document.

Verify the Installed STEP Version

SPOT can be used to check STEP's baseline version, component versions, core hotfixes, and customer add-ons that have been applied to the environment, as well as components that have available updates.

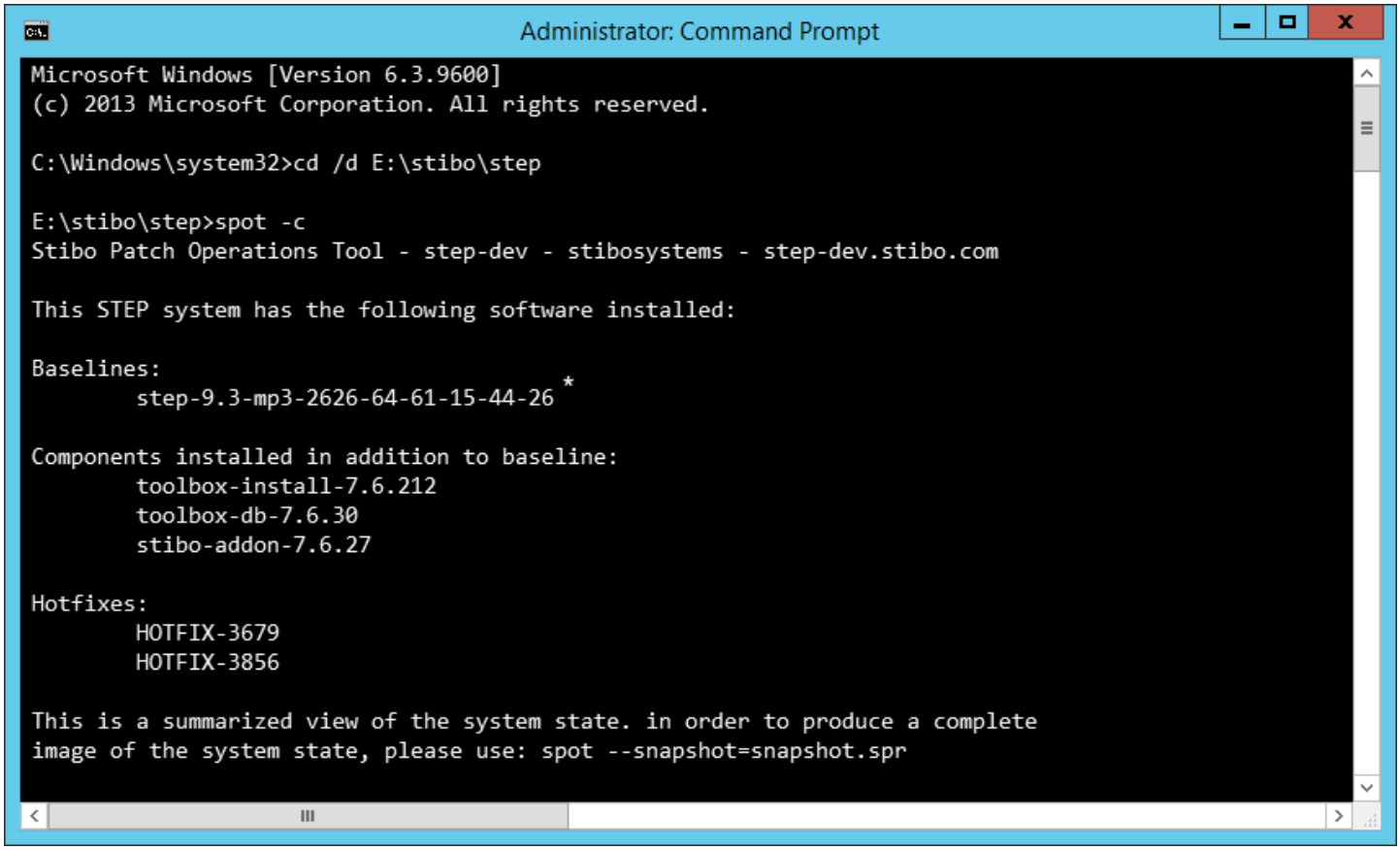
1. Go to the directory where STEP is installed. The default path is `/opt/stibo/step`
2. Run the SPOT command as follows:

   `./spot -c`

   or alternatively,

   `./spot --components`

Example:



Note: The baseline specifies the STEP version, which often includes a monthly maintenance patch (MP) level. In this example, the system is running STEP 9.3 with maintenance patch 3.

Verify components available for updates

1. Go to the directory where STEP is installed. The default path is `/opt/stibo/step`
2. Run the SPOT command as follows:

   `./spot -u`

   or alternatively,

   `./spot --showupdates`

Verify if all servers in the cluster are at the same software level

1. Go to the directory where STEP is installed. The default path is `/opt/stibo/step`
2. Run the SPOT command as follows:

   `./spot --clustercheck`

Optimizing Performance

Perceived bad performance of the STEP system is seldom the result of a single cause. Usually, a wide range of improvements, in a variety of areas, will result in an improved performance of the system. Therefore, it is impossible to predict which recommendations will result in a particular performance improvement percentage.

Note: Optimizing one area of the system may negatively affect performance in another area. Consider system-wide testing before and after making optimization changes to verify the end result meets your expectations.

This section includes the following recommended data gathering methods and performance improvement tactics:

1. **Performance assessment** involves analyzing the system from a performance perspective, identifying the pain points, and giving actionable recommendations to optimize performance.
2. **Performance troubleshooting** involves methods defined to troubleshoot performance issues.
3. **Technical performance recommendations** involve client computers, internet connection, and STEP application and database server setup and connections.
4. **Performance recommendations** involve configuring the data mode and the recommended use of application features.
5. **Healthcheck performance recommendations** involve identifying and resolving system and performance issues. The tests available are defined in the **Healthcheck** section of the **Administration Portal** documentation.
6. **Monitoring performance recommendations** involve managing the status of external systems used to process data as well as identify possible problem areas of the STEP application server. The monitoring sensors available are defined in the **Monitoring** section of the **Administration Portal** documentation.

Additionally, the In-Memory Database Component for STEP allows the system to take advantage of the availability of inexpensive memory to read all data into memory, and then read data from the In-Memory database instead of querying Oracle. For more information, refer to the **In-Memory Database Component for STEP** topic in the **Resource Materials** section of online help.

Technical Performance Recommendations

This topic describes the data gathering methodologies and performance recommendations from a technical infrastructure viewpoint for the server machines:

- Infrastructure Performance Recommendations
- Client Environment Recommendations
- Server Environment Recommendations
- Server Hygiene

Infrastructure Performance Recommendations

This is one of the technical infrastructure data gathering methodologies and performance recommendations. The full list is defined in the **Technical Performance Recommendations** topic.

Stibo Systems provides the STEP Infrastructure Recommendation document to describe how STEP environments should be set up to meet performance and scalability requirements.

The document should be updated when the load on the STEP environments has significantly increased after the infrastructure recommendation was delivered. For example, a significant increase of the load could include a significant increase in the number of STEP users, or a significant increase of the number of objects in STEP. As part of this engagement, review the loads documented in the original infrastructure recommendation document and contact Stibo Systems if that needs to be updated before the start of the engagement.

For on-premise systems, refer to the **Infrastructure Requirements** section in the **System Administration Guide** on the **Downloadable Documentation**.

Client Environment Recommendations

This is one of the technical infrastructure data gathering methodologies and performance recommendations. The full list is defined in the **Technical Performance Recommendations** topic.

This topic describes the data gathering methodologies and performance recommendations for the client machines using workbench and Web UI, including:

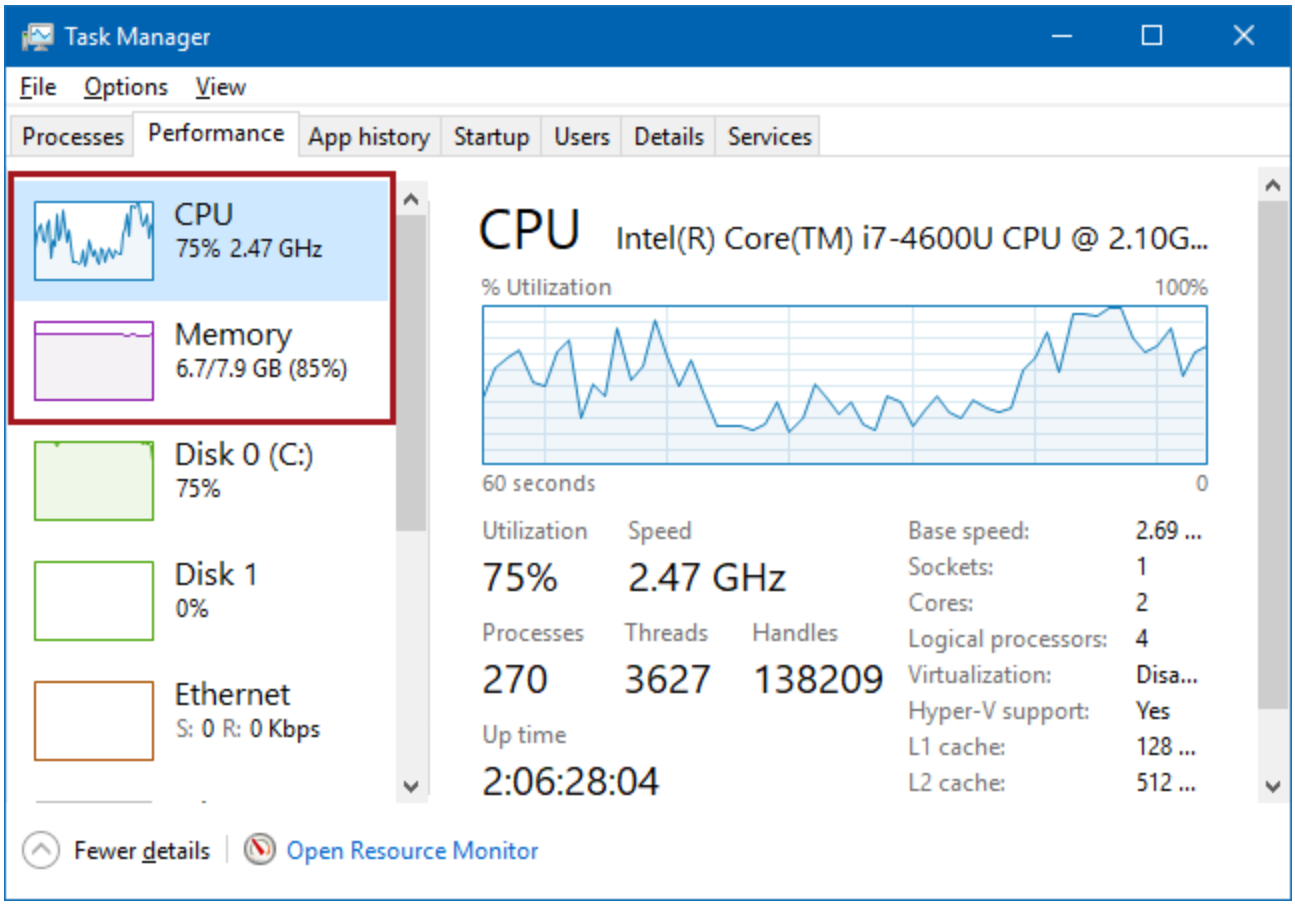
- Client computer
- Client internet browser
- Client internet connection

Client Computer

Poor performance in workbench and Web UI can be caused by a client computer that runs slowly.

Check the performance of a Windows computer using the following steps:

1. On the keyboard, press the following three keys simultaneously: **Ctrl**+**Alt**+**Delete**.
2. Click the **Task Manager** option.
3. On the Performance tab, analyze the client computer **CPU** utilization and **Memory** usage.



Recommendation

If the CPU and/or Memory percentage stays high (near 100%), the client computer is generally slow. Workbench and Web UI do not perform well on a slow computer. Consult your IT department for assistance.

Client Internet Browser

Poor performance of an internet browser can hinder the performance of workbench and Web UI. An internet browser can perform slowly due to plug-ins or an outdated version.

Check internet browser performance using the following steps:

1. Verify that the browser version being used is supported by the installed version of STEP. Search online help for the relevant Platform and Software Support topic.
2. Verify that the browser version being used is included in the Infrastructure Recommendations.
3. Open the browser and view a number of website, observing the response time.
4. If the browser is slow, try a different supported browser for comparison.

Recommendation

Update the internet browser version, disable plug-ins, or install a different browser. Consult your IT department for assistance.

Client Internet Connection

A slow or unstable internet connection can cause poor performance in workbench and Web UI. This check can be performed from a Windows machine via a command prompt, or from the workbench regardless of the machine type. The response times of the ping and/or network latency check should meet normal internet connection figures.

For example, if the application server has a high response time (such as 125 milliseconds), the internet connection from the client computer to STEP is about the same speed as a dial-up connection.

Important: You can estimate an additional 1 millisecond in network latency for every 100 kilometers the data travels. So, if the application server is in Venlo, Netherlands and the client is in Seattle, USA (a distance of about 8,000 km), then a response time of about 30 milliseconds from client to application server corresponds with a response time of about 110 milliseconds from client in Seattle, USA to application server in Venlo, Netherlands.

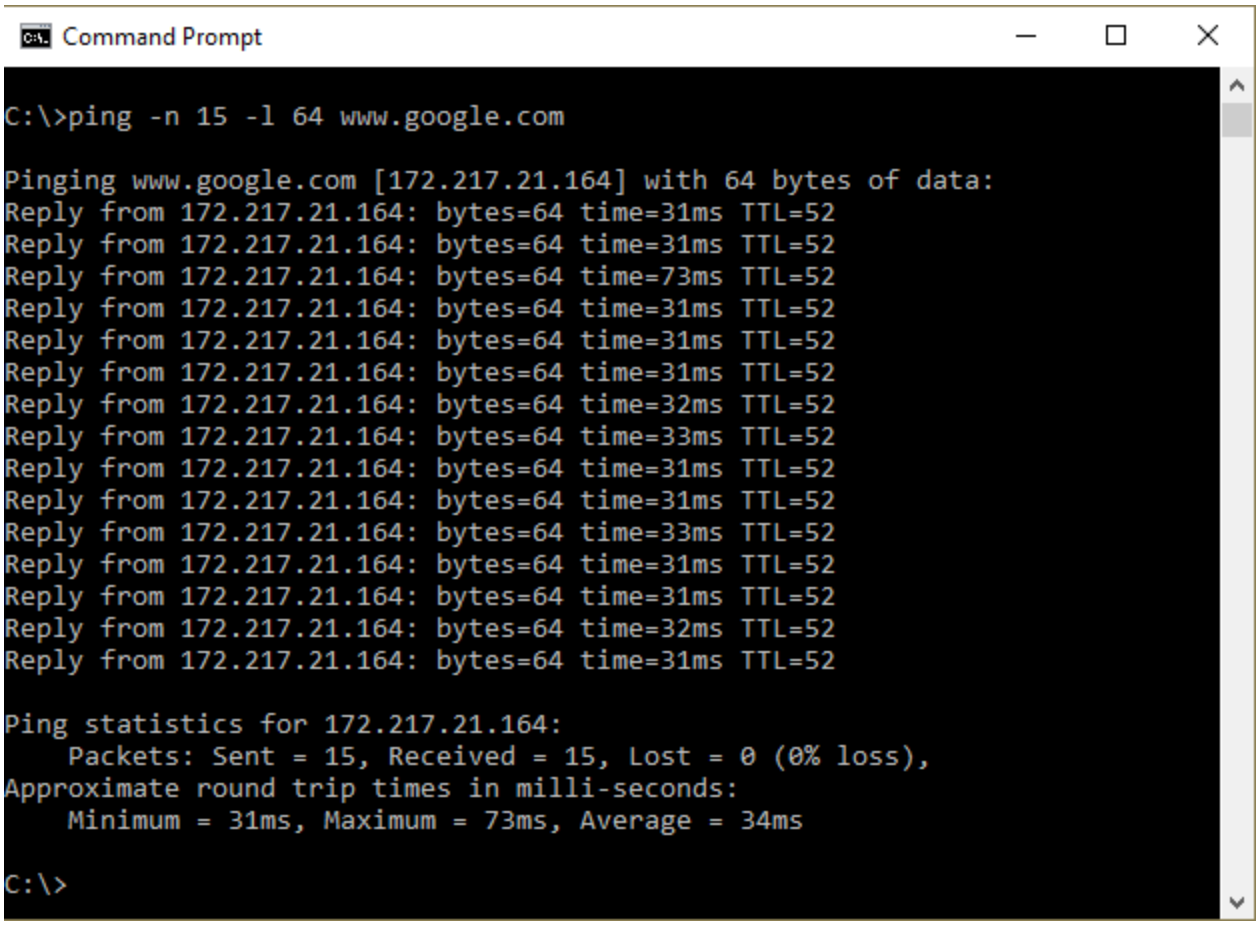
Windows Command Prompt Check

Check internet connection performance using the following steps:

1. On a Windows machine, click **Start** and then type 'cmd' to display the Command Prompt option.
2. In the command prompt dialog, type the following command to 'ping' the Google website.

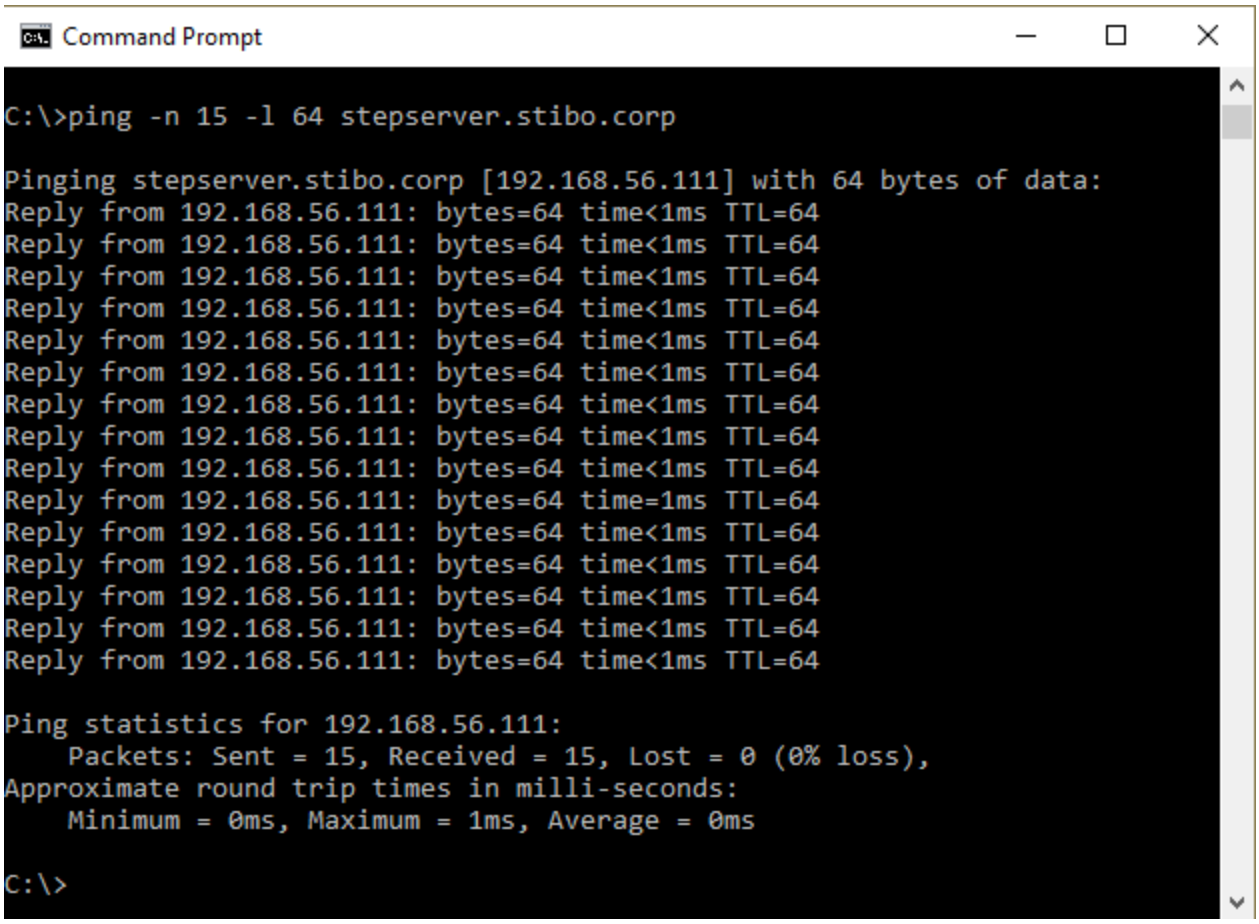
   ```
   ping -n 15 -l 64 www.google.com
   ```

   In this command, '-n 15' means 15 echo requests instead of the default 4; and '-1 64' means 64 bytes requests instead of the default 32 bytes.



3. In the command prompt dialog, type the following command to 'ping' the application server.

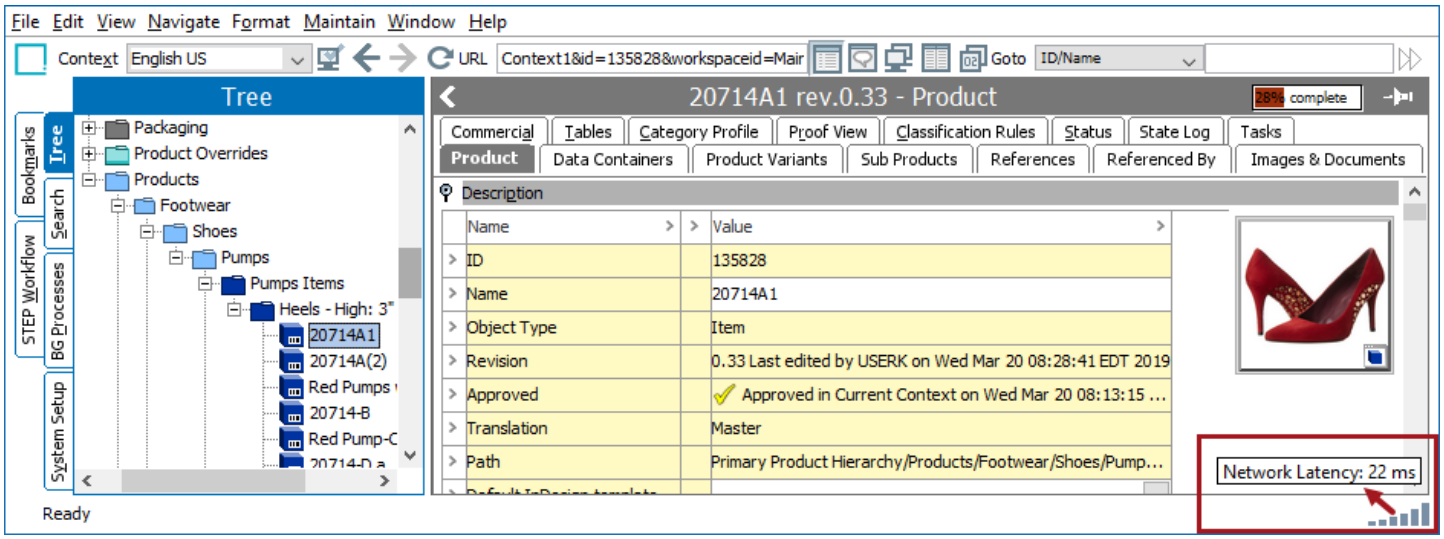
```
ping -n 15 -l 64 [STEP_SERVER_NAME]
```



Workbench Check

Check internet connection performance using the following steps:

1. Log in to the workbench.
2. Hover over the icon on the bottom right corner to display the Network Latency value.



Recommendation

The easiest way to improve a bad internet connection is to connect via a wired (cable) connection instead of wireless Wi-Fi. If the client machine still experiences a bad internet connection with response times frequently above 125 milliseconds, then the internet connection should be improved. Consult your IT department for assistance.

Server Environment Recommendations

This is one of the technical infrastructure data gathering methodologies and performance recommendations. The full list is defined in the **Technical Performance Recommendations** topic.

This topic describes the data gathering methodologies and performance recommendations from a technical infrastructure viewpoint for the server machines:

- Server Configuration File
- Server Virtualization
- Server Network Latency
- Folders as Shared Locations
- Server Log File Settings
- Components
- Application Server Load
- Garbage Collection Log Files
- Database Long-Lasting SQL Queries
- Database Server Redo Log
- Database Server Alert Log
- Database Server Oracle Adaptive Features
- Database Server Oracle Statistics
- Database Server Oracle DataGuard
- DTP Server
- Custom Extensions

Server Configuration File

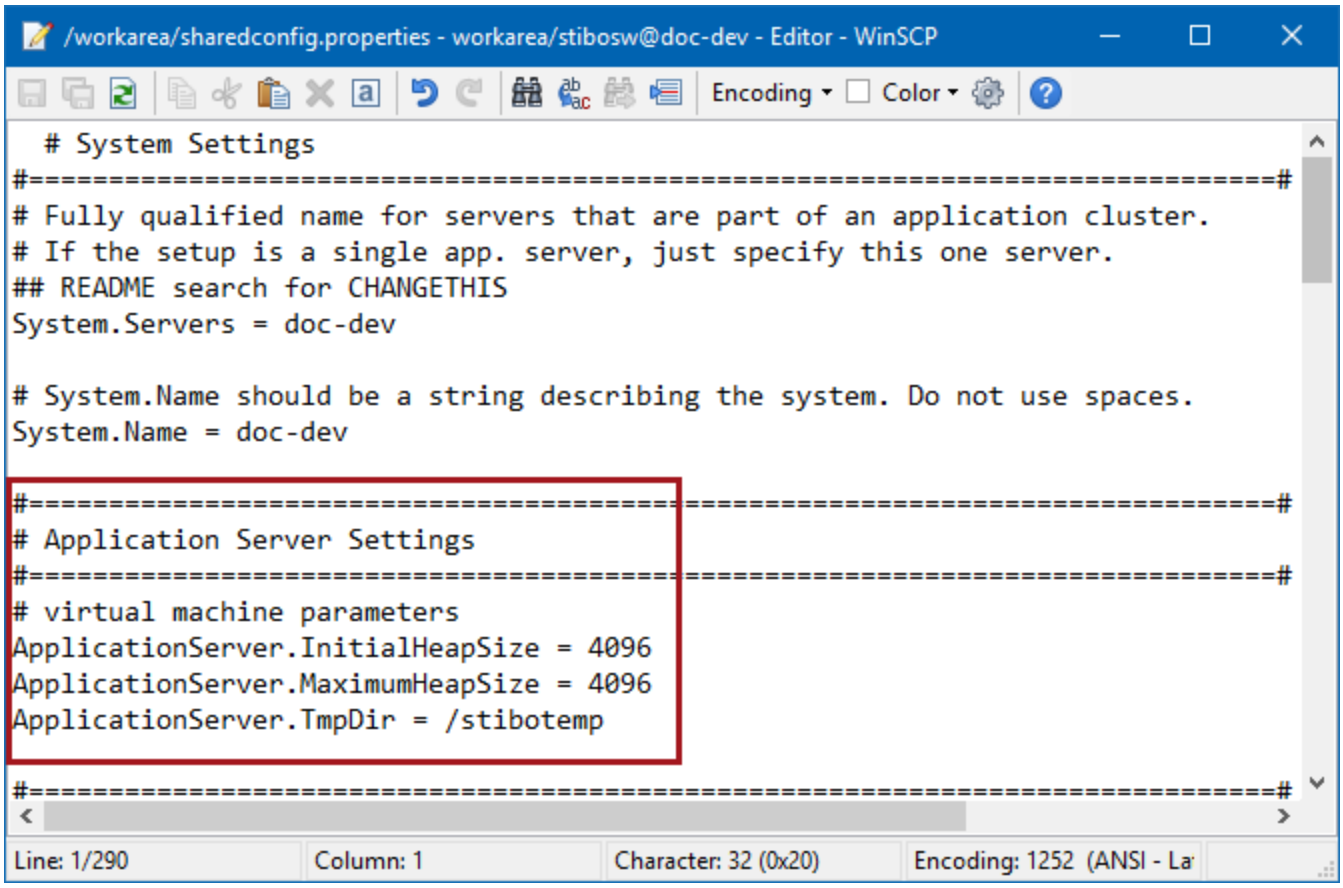
This is one of the technical infrastructure data gathering methodologies and performance recommendations for server machines. The full list is defined in the **Server Environment Recommendations** topic.

The server configuration settings are defined by Stibo Systems. For example, the initial and maximum heap sizes are set by Stibo Systems and should remain unchanged.

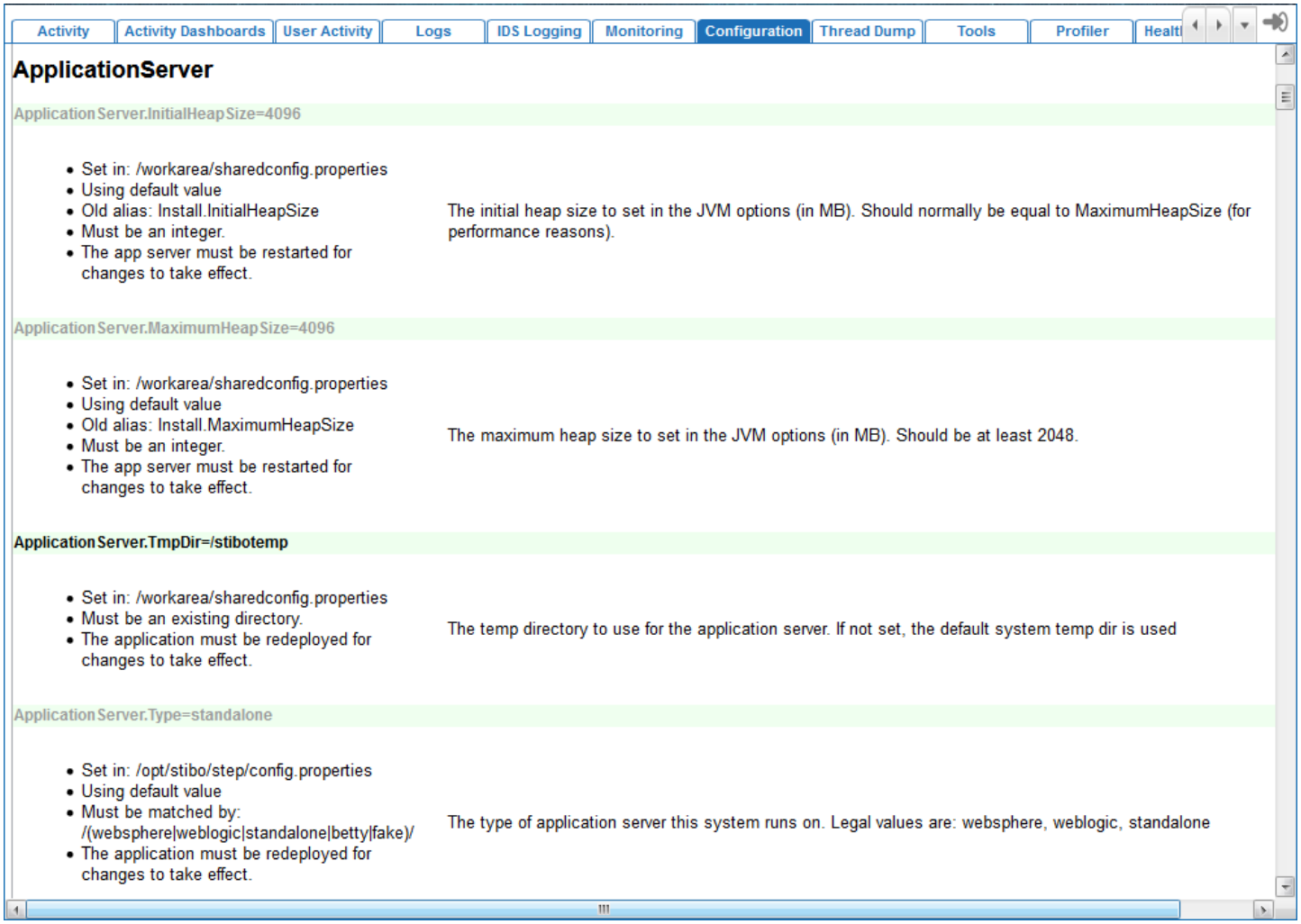
Note: Before modifying the server configurations in the sharedconfig.properties file, first consult your Stibo Systems representative.

To check the server configurations, view the ApplicationServer settings in the sharedconfig.properties file settings using one of these options:

- On the application server, locate and view the **sharedconfig.properties** file and locate the application server section.



- From the Start Page, click the STEP System Administration button and supply the login credentials. On the Configuration tab, click the **Fetch data** button to load the current system configuration values. Under the '**ApplicationServer**' section, review the default and non-default settings. If the configurations are shown in black text, then the setting for these are non-default and set in a config file. Gray text indicates properties that have default values. For more information, refer to the **Configuration** topic in the **Administration Portal** documentation.



Important: Whenever changing a value in a property that contains special characters, such as ':', '=', or '\', those characters must be escaped. For example, the value 'j=:geh' should be entered as 'j\=\:geh'. If these characters are not escaped properly, the server will not interpret them in the intended manner. There is an exception to this guidance: Passwords should not be escaped.

Server Virtualization

This is one of the technical infrastructure data gathering methodologies and performance recommendations for server machines. The full list is defined in the **Server Environment Recommendations** topic.

Stibo Systems recommends to set up server virtualization according to the provided STEP Infrastructure Recommendation document and/or Platform and Software Support document. Furthermore, Stibo Systems recommends taking the following white paper for virtualization on VMware into consideration for latency-sensitive systems.

Search the web for relevant details regarding dealing with latency-sensitive applications while running in a VMware environment. For example, technical white papers on the https://www.vmware.com website.

Server Network Latency

This is one of the technical infrastructure data gathering methodologies and performance recommendations for server machines. The full list is defined in the **Server Environment Recommendations** topic.

The flexibility of STEP gives rise to a potentially large number of ad hoc requests between the application server and the database server. Network influence (latency) and bandwidth between the application and database servers are important because STEP performance is negatively impacted when the network latency is not stable and frequently above the latency requirements.

The ability to synchronize across the application server cluster is crucial. In an environment with two application servers, to avoid 'Optimistic Locking' errors, a data change via one application server needs to be reflected as close to instantly as possible.

Although full network recommendations are stated in the Infrastructure Recommendation document and/or Platform and Software Support document, of particular interest on this topic are:

- Network bandwidth requirements: Minimum requirement is 1 gigabit dedicated network between all STEP servers except to file servers; the recommended requirement is 10 gigabit for enterprise systems with a heavy workload profile.
- Network latency requirements: The requirements for network latency between application servers in an application cluster and between application server(s) and database server(s) are:
  - Maximum 0.2 milliseconds with a 64 bytes packet size
  - Maximum 0.4 milliseconds with an 8K packet size
  - Maximum 1.0 milliseconds with a 32K packet size

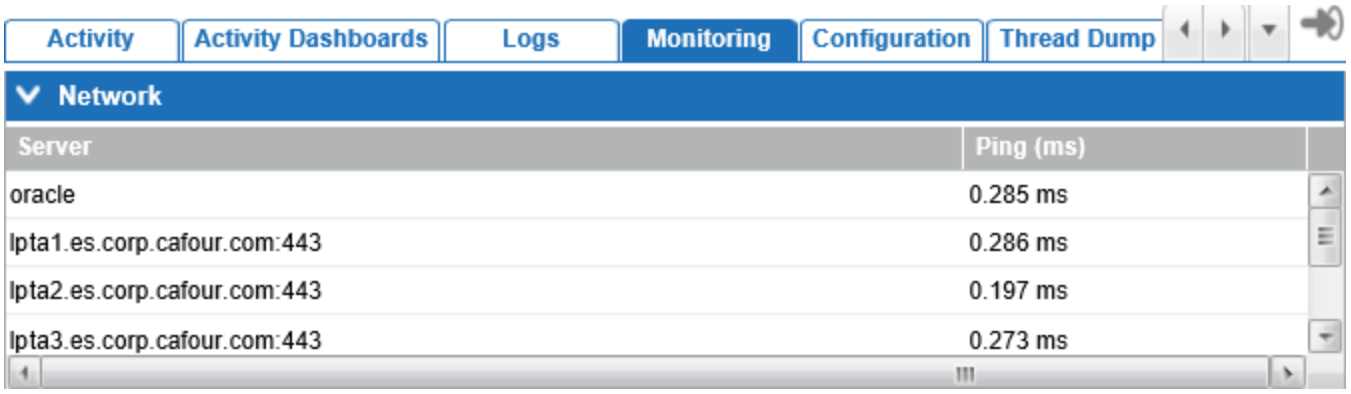
Network latency can be analyzed using the options defined below. If network latency is not stable and/or is frequently above the latency requirements, consult your IT department for optimization.

Network Latency Analysis via STEP

From the Start Page, click the **STEP System Administration** button and log in.

1. Click the **Monitoring** tab to view an instant indication of the network latency between the application and database servers. This is a near real-time check of the latency between the servers. For more information, refer to the **Monitoring** topic in the **Administration Portal** documentation.

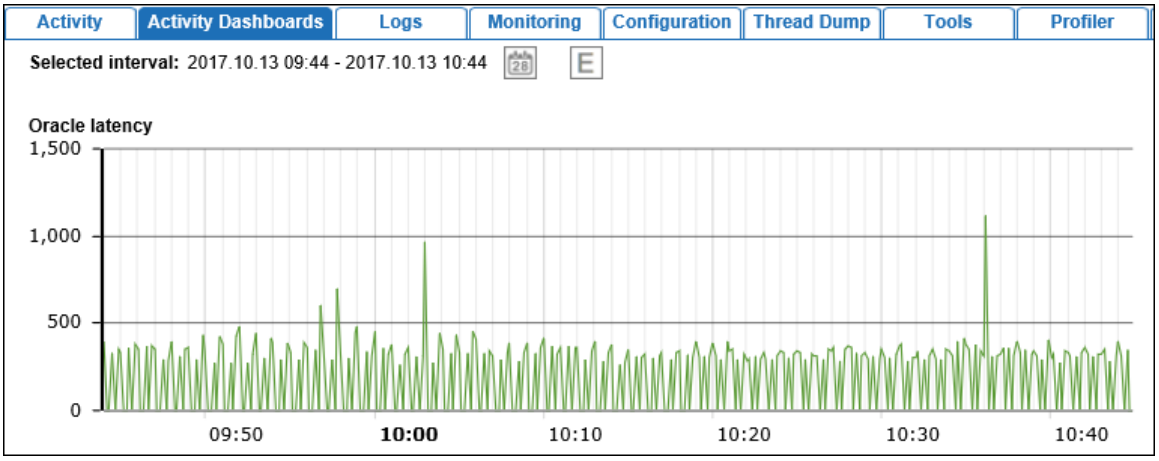
   The following example shows a network latency which is not significantly above the 0.2 milliseconds and therefore indicates a good network latency:



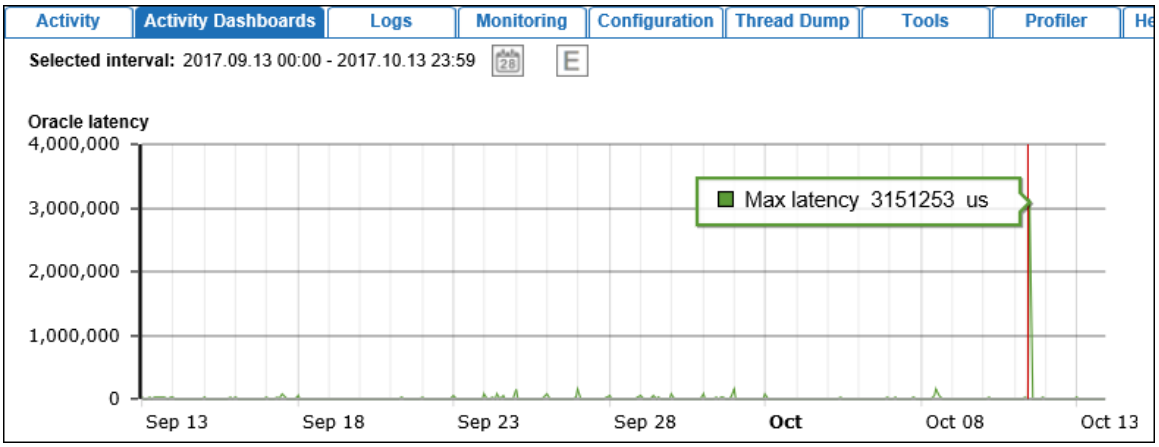
2. Click the **Activity Dashboard** tab to view a historic network latency analysis. This graphic provides the average time it takes to execute 10 simple database requests. High latency numbers can indicate network issues, but can also indicate the database server is having trouble keeping up with the number of requests. For more information, refer to the **Activity Dashboard** topic in the **Administration Portal** documentation.

   The following images provide historic insights in the network latency:

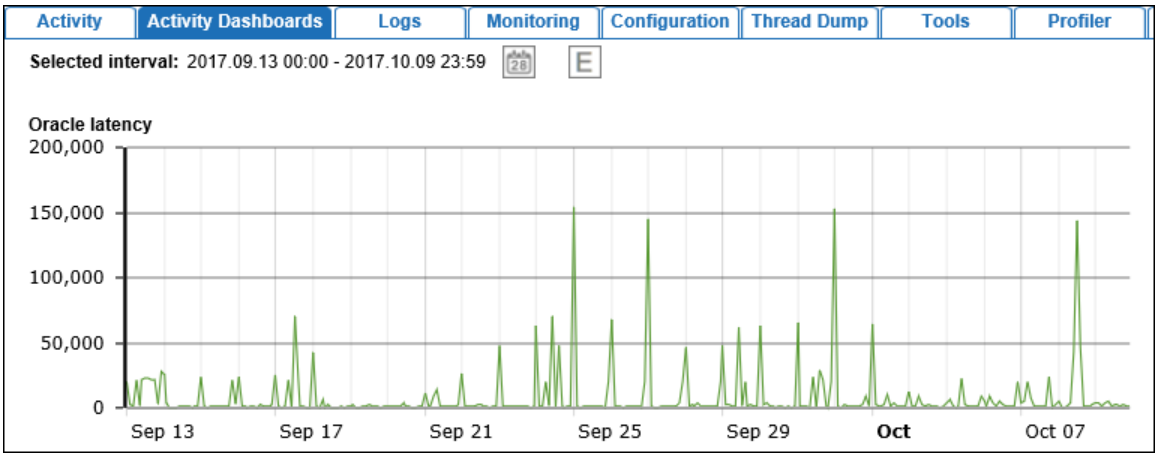
   - The time interval 'last hour' shows that latency had a couple of spikes near to the 1 millisecond, which is something to be concerned about.



   - The time interval 'last thirty days' shows a large spike around October 10 of about 3 seconds which is concerning.



- The time interval before the spike above shows many other spikes, frequently above the 50 milliseconds, which is concerning as well.



Network Latency Analysis via Ping

Based on your type of server, use ping to measure network latency between application server(s) and database server(s) in an application cluster, as defined below.

- On a Linux server, run the following ping commands from application server to database server:

```
ping -c20 <db-host> |grep rtt
ping -c20 -s8000 <db-host> |grep rtt
ping -c20 -s32000 <db-host> |grep rtt
```

- On a Windows server, run the following hrping commands from application server to database server:

```
hrping -l 64 -n 20 <database-host>
hrping -l 8000 -n 20 <database-host>
hrping -l 32000 -n 20 <database-host>
```

Network Latency Analysis via Script

A script executed from the application server against the database and with different package-sizes can be used to measure any network latency.

1. From a Linux server, log on to the application server as 'stibosw'.
2. Copy the following SQL script using the file name 'benchnet5.sql' to `/home/stibosw` on the application server.

```
set timing on

set autotrace traceonly statistics
set arrays 5000
set pages 1000
select null from dual connect by rownum < 100000;
prompt Small without network influence
select null from dual connect by rownum < 100000;
set arrays 10
prompt Small with network influence
select null from dual connect by rownum < 100000;

set arrays 5000
prompt Medium without network influence
select
'01234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901'||rownum
from dual connect by rownum < 1000000;
set arrays 100
prompt Medium with network influence
select
'01234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901'||rownum
from dual connect by rownum < 1000000;

set arrays 5000
prompt Large without network influence
select
'01234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901
12345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012
34567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234
56789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789'||rownum from dual connect by rownum
< 1000000;
set arrays 100
prompt Large with network influence
select
'01234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901
12345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012
34567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234
56789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789'||rownum from dual connect by rownum
< 1000000;
```

3. Run the script on the application server:

```
sqlplus stepsys/stepsys
@/home/stibosw/benchnet5.sql
```

4. Collect all output and attach it to a Jira issue.

Folders as Shared Locations

This is one of the technical infrastructure data gathering methodologies and performance recommendations for server machines. The full list is defined in the **Server Environment Recommendations** topic.

Shared locations on other servers can be used as STEP folders. For example, using a server other than the STEP application server to house the hotfolders for import files. However, placing STEP files on shared locations can be unpredictable, depending on the latency of the network. Therefore, if STEP files are on shared locations, then the infrastructure setup should comply with the Infrastructure Recommendation document and/or Platform and Software Support document, and the latency of the network should meet the stated recommendations.

For example, if the application server has a high response time, such as 220 milliseconds, then this means the internet connection from the client computer to STEP is about the same speed as a dial-up connection.

Important: You can estimate an additional 1 millisecond in network latency for every 100 kilometers the data travels. For example, consider that the application server and the database server are both located in Amsterdam in the Netherlands and the latency between both is 0.2 milliseconds on average, which is an optimal latency. The shared file location, however, is located in Seattle, USA, which is 8,000 km away. In this case, an additional 80 milliseconds needs to be added to the network latency, resulting in a drastic decrease of the optimal network latency of 80.2 milliseconds instead of 0.2 milliseconds.

To check network latency from a browser, refer to the **Network** section of the **Monitoring** topic in the **Administration Portal** documentation.

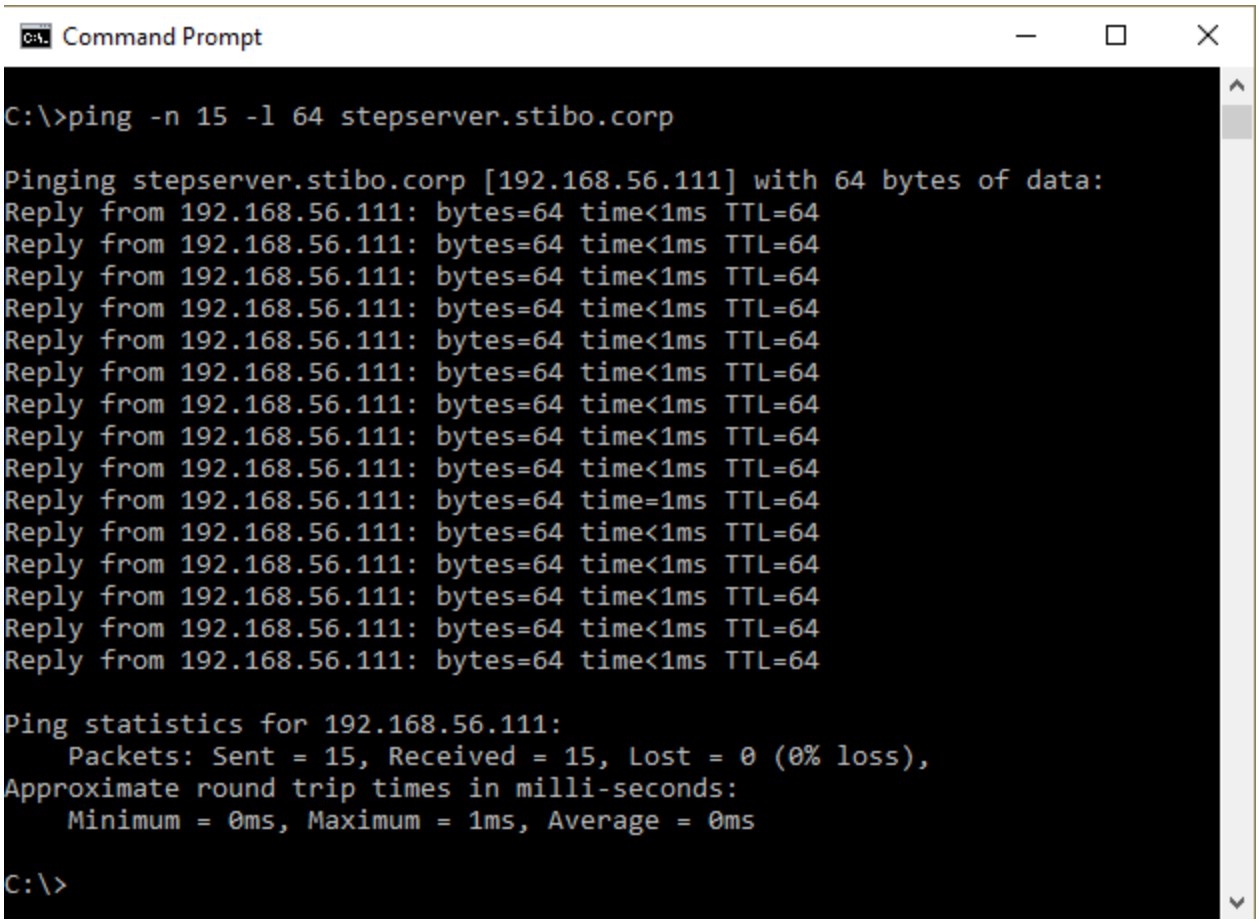
Windows Check

Check the internet connection performance on a Windows application server machine as follows:

In a command prompt dialog, type the following command to ping the shared file location:

```
ping -n 15 -l 64 shared.file.location
```

In this command, '-n 15' means 15 echo requests instead of the default 4; and '-l 64' means 64 bytes requests instead of the default 32 bytes.

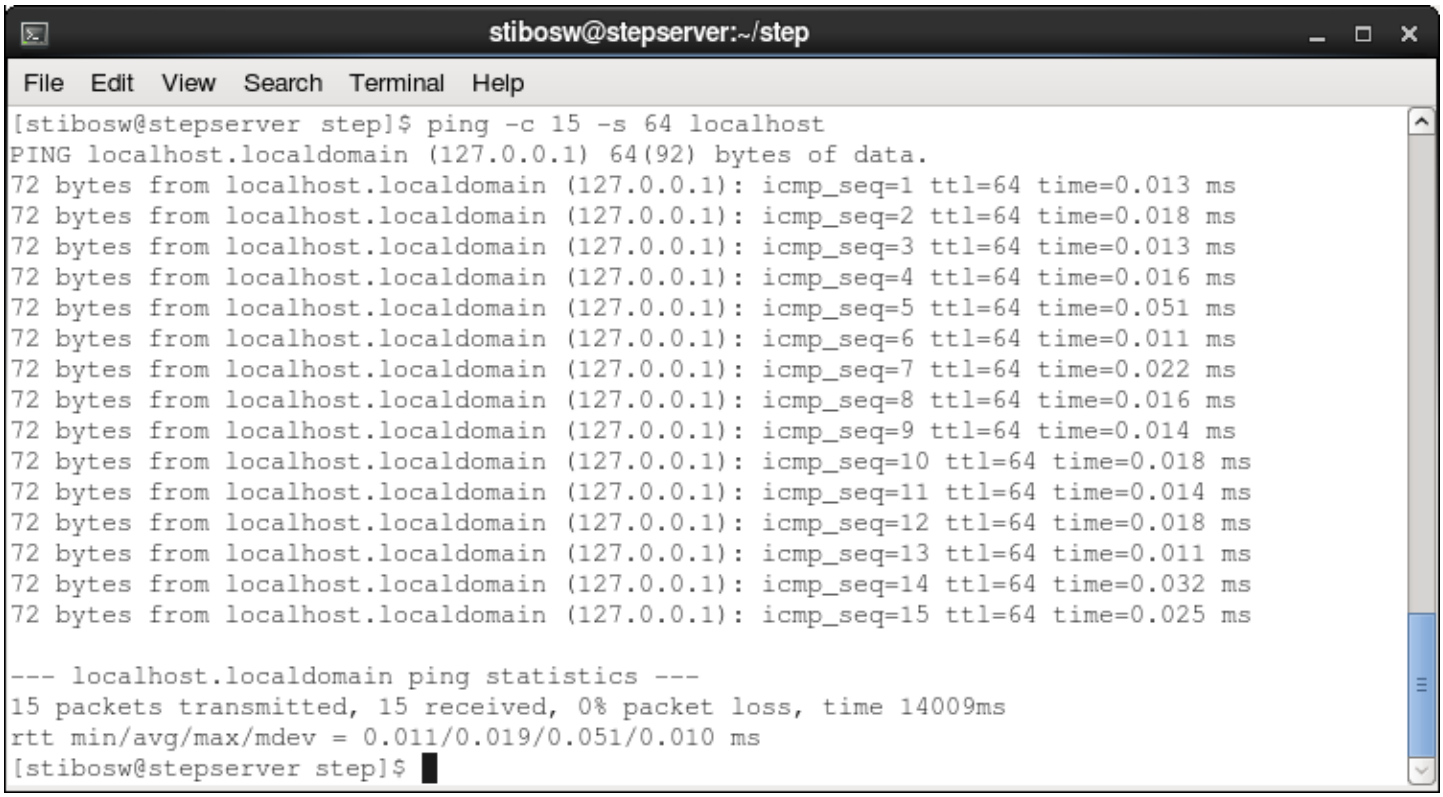


Linux Check

Check the internet connection performance on a Linux application server machine as follows:

```
ping -c 15 -s 64 shared.file.location
```

In this command, '-c 15' means 15 echo requests instead of the default 4; and '-s 64' means 64 bytes requests instead of the default 32 bytes.


stibosw@stepserver:~/step
File Edit View Search Terminal Help
[stibosw@stepserver step]$ ping -c 15 -s 64 localhost
PING localhost.localdomain (127.0.0.1) 64(92) bytes of data.
72 bytes from localhost.localdomain (127.0.0.1): icmp_seq=1 ttl=64 time=0.013 ms
72 bytes from localhost.localdomain (127.0.0.1): icmp_seq=2 ttl=64 time=0.018 ms
72 bytes from localhost.localdomain (127.0.0.1): icmp_seq=3 ttl=64 time=0.013 ms
72 bytes from localhost.localdomain (127.0.0.1): icmp_seq=4 ttl=64 time=0.016 ms
72 bytes from localhost.localdomain (127.0.0.1): icmp_seq=5 ttl=64 time=0.051 ms
72 bytes from localhost.localdomain (127.0.0.1): icmp_seq=6 ttl=64 time=0.011 ms
72 bytes from localhost.localdomain (127.0.0.1): icmp_seq=7 ttl=64 time=0.022 ms
72 bytes from localhost.localdomain (127.0.0.1): icmp_seq=8 ttl=64 time=0.016 ms
72 bytes from localhost.localdomain (127.0.0.1): icmp_seq=9 ttl=64 time=0.014 ms
72 bytes from localhost.localdomain (127.0.0.1): icmp_seq=10 ttl=64 time=0.018 ms
72 bytes from localhost.localdomain (127.0.0.1): icmp_seq=11 ttl=64 time=0.014 ms
72 bytes from localhost.localdomain (127.0.0.1): icmp_seq=12 ttl=64 time=0.018 ms
72 bytes from localhost.localdomain (127.0.0.1): icmp_seq=13 ttl=64 time=0.011 ms
72 bytes from localhost.localdomain (127.0.0.1): icmp_seq=14 ttl=64 time=0.032 ms
72 bytes from localhost.localdomain (127.0.0.1): icmp_seq=15 ttl=64 time=0.025 ms
--- localhost.localdomain ping statistics ---
15 packets transmitted, 15 received, 0% packet loss, time 14009ms
rtt min/avg/max/mdev = 0.011/0.019/0.051/0.010 ms
[stibosw@stepserver step]$

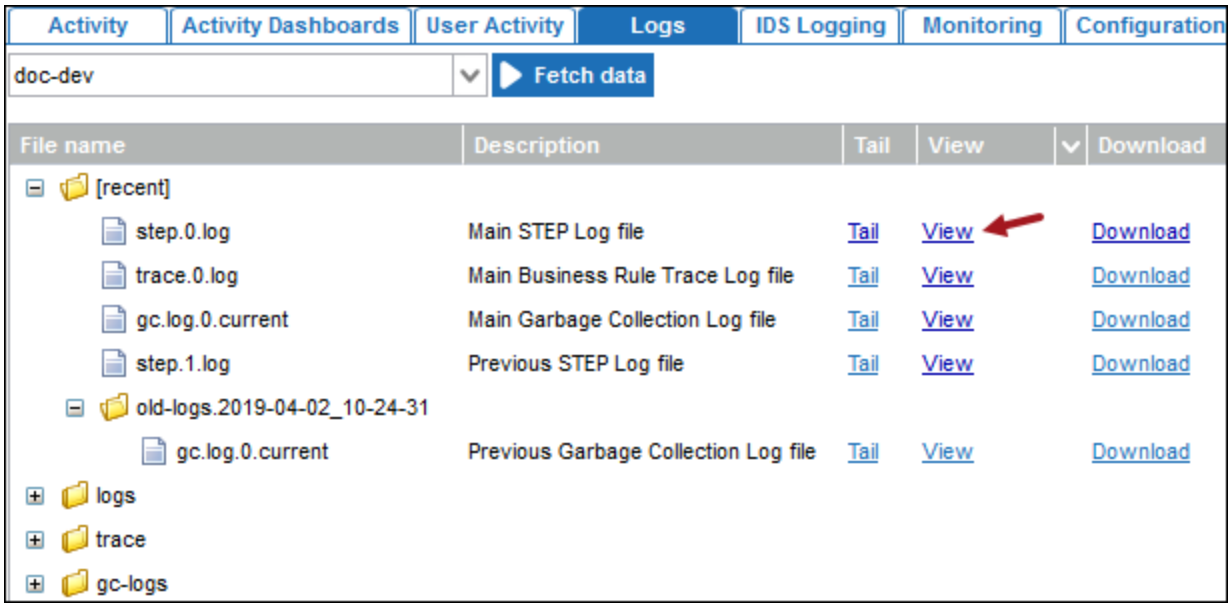
Server Log File Settings

This is one of the technical infrastructure data gathering methodologies and performance recommendations for server machines. The full list is defined in the **Server Environment Recommendations** topic.

STEP uses a powerful logging subsystem tool to provide troubleshooting information if the system is not working as expected. For more information, refer to the **Logs** topic in the **Administration Portal** documentation.

View Main STEP Log File

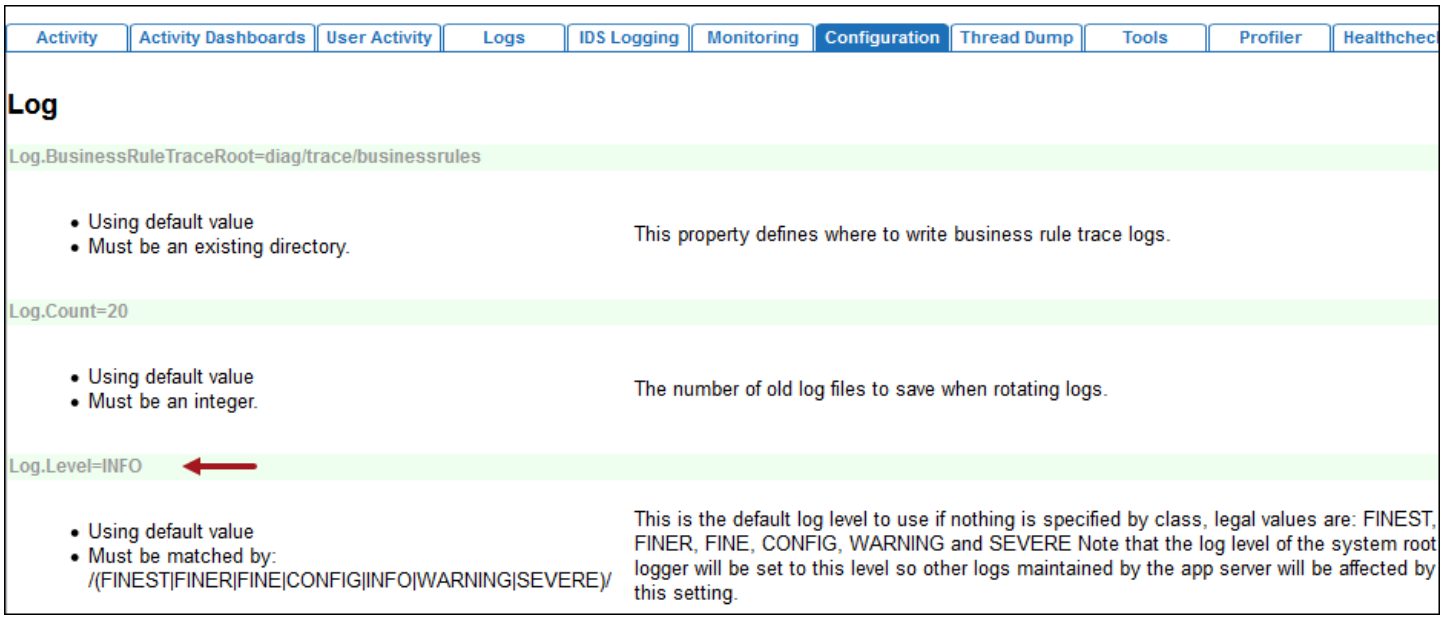
From the Start Page, click the **STEP System Administration** button and supply the login credentials. On the Logs tab, click the **Fetch data** button to load the logs. On the **Main STEP Log file** row, click the **View** link.



Logging Levels

The Log.Level configuration property in the sharedconfig.properties file controls the level of detail generated in the logs. A high level log contains only the most important messages while a low level provides more detail. Each level implicitly logs all other levels above that level as well.

To determine the current level being logged, from the Start Page, click the **STEP System Administration** button and supply the login credentials. On the Configuration tab, search for Log.Level.



Important: Log level detail FINE, FINER, and FINEST can create a large number of log entries in a short period of time. Typically these levels are not used for the whole system.

The available logging levels are:

- SEVERE: Errors where the program is unable to recover and continue in a normal manner, such as a workflow looking for a property that does not exist.
- WARNING: Conditions which are possible errors or where the program is able to recover and continue in a normal manner.
- INFO: Messages indicating normal progress of the program, the default setting. These entries can be of interest to the end user.
- CONFIG: Messages not generally of interest to the end user, but often useful for diagnosing issues on a running system.
- FINE: Messages of interest to most developers.
- FINER: Messages of interest to only a few developers.
- FINEST: Messages of interest to only a very few developers. This can be extremely verbose.

Recommendation

The log file settings typically differ per environment, as follows:

- On development (STEP DEV) and test (STEP TEST) environments, set the log to FINE to trace errors. This log level assists in identifying potential issues during development and testing. If there is an error and the current level log file setting does not provide enough detail, set the log to FINER or FINEST.
- On a QA environment (STEP QA), set the log to INFO or WARNING.

- On a production environment (STEP PROD), the log level is typically set to log SEVERE issues during normal operation. This avoids the risk of clogging the production system with large log files.

Package-Specific Logging

Set a logging level for a package to allow a more granular log for a specific area. In the sharedconfig.properties file, add the name of the package to the end Log.Level property.

For example, the setting **Log.Level.com.stibo.centralhotfolder=FINE** will set the level of everything under com.stibo.centralhotfolder to FINE.

Stack Trace Information

In the event of an error, the log will commonly have a stack trace of some sort. While the stack trace data is mostly of interest to developers, it is a good indicator of the most interesting log messages. The following is an example log entry with a stack trace. Real log entries and stack traces will look somewhat different and will not have identifying information trimmed out as is in this illustration.

```
Feb 18, 2009 6:28:33 PM com.stibo.servicemanager.beans.CentralHotFolder doRun
SEVERE: Unable to handle [trimmed] exception message Could not rename [trimmed] to
[trimmed]
java.lang.RuntimeException: Could not rename [trimmed] to [trimmed]
at com.stibo.customer.[trimmed].moveFile([trimmed].java:143)
at com.stibo.customer.[trimmed].handleFile([trimmed].java:76)
at com.stibo.servicemanager.beans.CentralHotFolder.doRun(CentralHotFolder.java:277)
at com.stibo.servicemanager.beans.CentralHotFolder$4$1.run
(CentralHotFolder.java:211)
...
```

- The first line shows the date and time as well as what part of the program the message was logged from, namely the doRun method in com.stibo.servicemanager.beans.CentralHotFolder.
- The next line gives the error message where the cause was one file not being able to be renamed and the error happened in a specific customer hotfolder.

Components

This is one of the technical infrastructure data gathering methodologies and performance recommendations for server machines. The full list is defined in the **Server Environment Recommendations** topic.

STEP components include:

- Core functionality: the STEP version, e.g., STEP 11.1, STEP 11.2, etc.
- Hotfixes (optional)
- Monthly Patches (optional)
- Custom Extensions (optional)

Run the './spot --components' command to display the applied components, as shown below:

```
This STEP system has the following software installed:

Baselines:
        sdl-with-framework-8.3
        step-9.0-mp2-2018-06-28-17-26-22

Components installed in addition to baseline:
        restapiv2-7.0.1
        mongodbadapter-7.0.48
        integration-amazon-sqs-7.0.3
        fda-esignature-7.0.1
        assetdownload-7.0.1
        assetanalyzer-7.0.1
        acrolinx-7.0.2
```

Considerations

- The DTAP street needs to be based on the same components. The Development, Test, Acceptance, and Production system should be based on the same components and versions. If not, at a minimum, ensure the baseline is the same on all environments.
- Consider that there can be unnecessary components on the STEP system after an upgrade. It may well be that after an upgrade, some of the hotfixes are not needed anymore. Read the Release Notes carefully to be sure that all unnecessary components are removed.

Application Server Load

This is one of the technical infrastructure data gathering methodologies and performance recommendations for server machines. The full list is defined in the **Server Environment Recommendations** topic.

The load of the systems in the cluster can be analyzed using the STEP Admin Portal.

- When the STEP server load is high, the performance is degraded and further actions are required.
- When the STEP environment is a cluster environment, and one of the systems in the cluster is not balanced, performance is degraded and further actions are required.

Use these steps to analyze the CPU and Memory load and distribution of the STEP Application Servers:

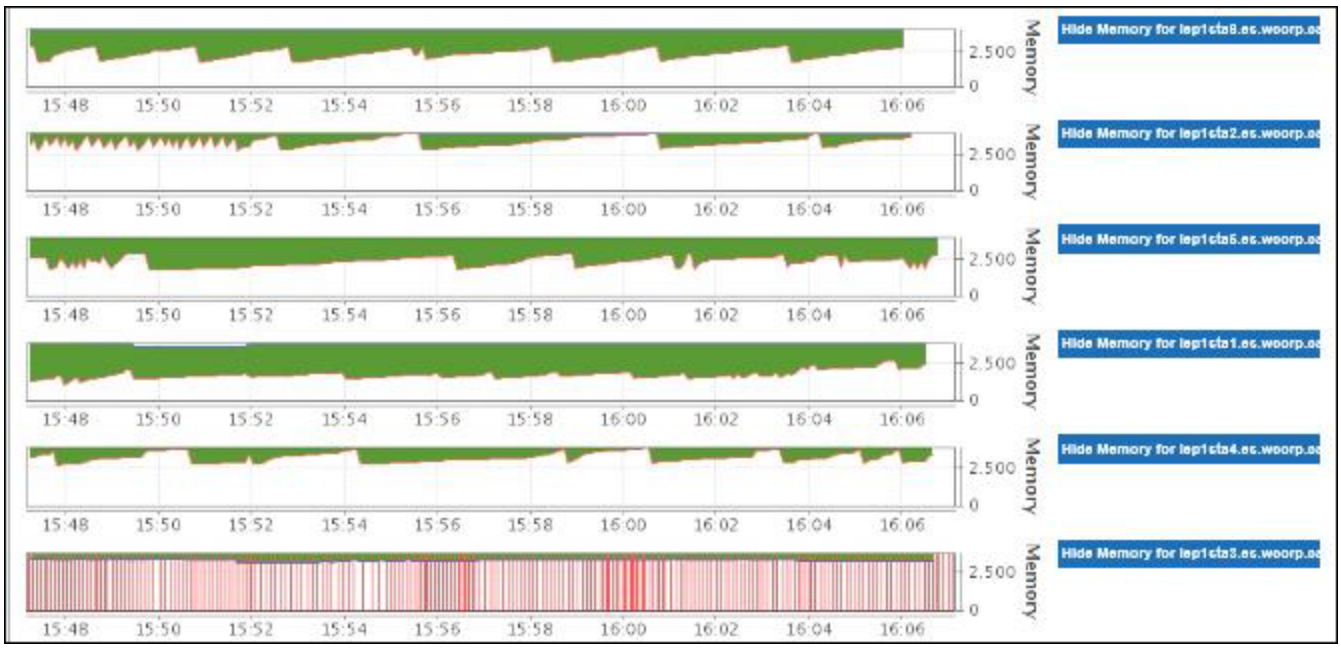
1. From the Start Page, click the **STEP System Administration** button.
2. Supply the login credentials and click the **Login** button.
3. On the **Activity** tab, click the **Fetch data** button to load CPU, Memory, and Thread data.

   For more information, refer to the **Activity** topic in the **Administration Portal** documentation.

For example, the CPU load of the servers may look like the following image:

The CPU load reported above is unbalanced on the different servers as part of the cluster. Mainly, the last server lep1sta3.es, is doing all the work and the rest of the servers are hardly loaded.

In another example, the Memory load of the servers may look like the following image:



The memory load reported above is unbalanced on the different servers as part of the cluster. Mainly, the last server lep1sta3.es is doing all the work and the rest of the servers are hardly loaded.

In these examples, the server lep1sta3.es is running back-to-back garbage collection due to the high load, potentially combined a too small heap-allocation. So, in this example, the recommendation is to analyze the heap size further and determine if it is configured correctly.

Additionally, there seems to be incorrect load balancing between the servers in the cluster for the background processes. The system selects which of the application servers should handle the background processes. Therefore, it is also recommended to further analyze the load balancing.

Garbage Collection Log Files

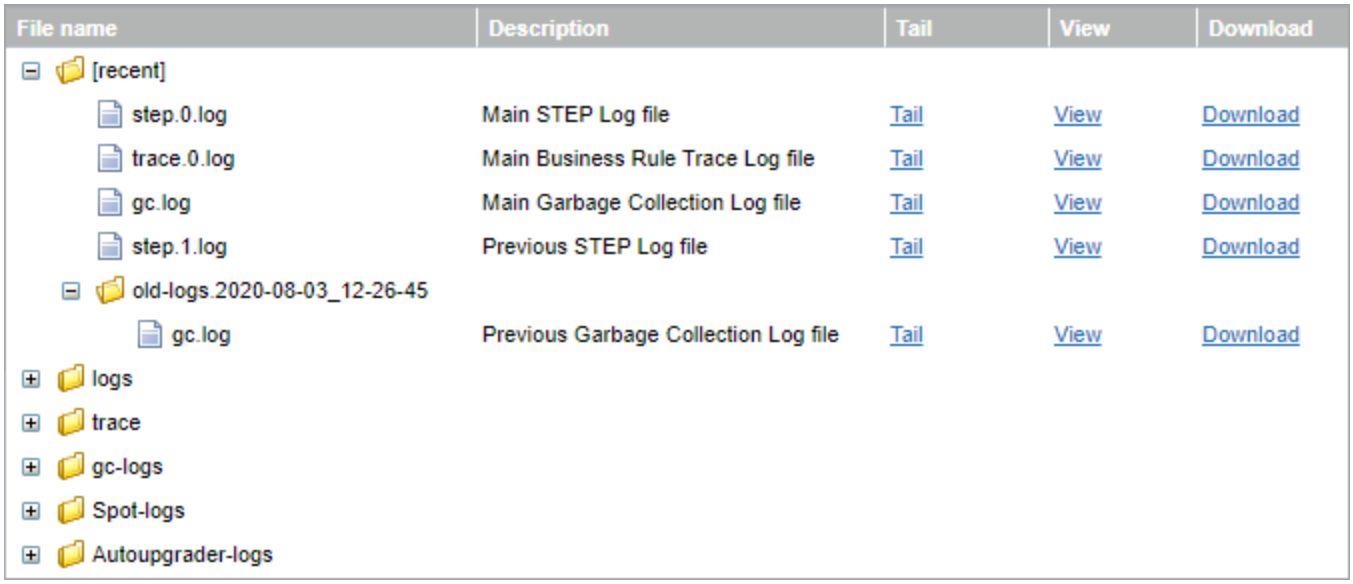
This is one of the technical infrastructure data gathering methodologies and performance recommendations for server machines. The full list is defined in the **Server Environment Recommendations** topic.

Healthy systems process garbage collections with speed. Long-running and frequent garbage collection processes indicate that system speed can be improved.

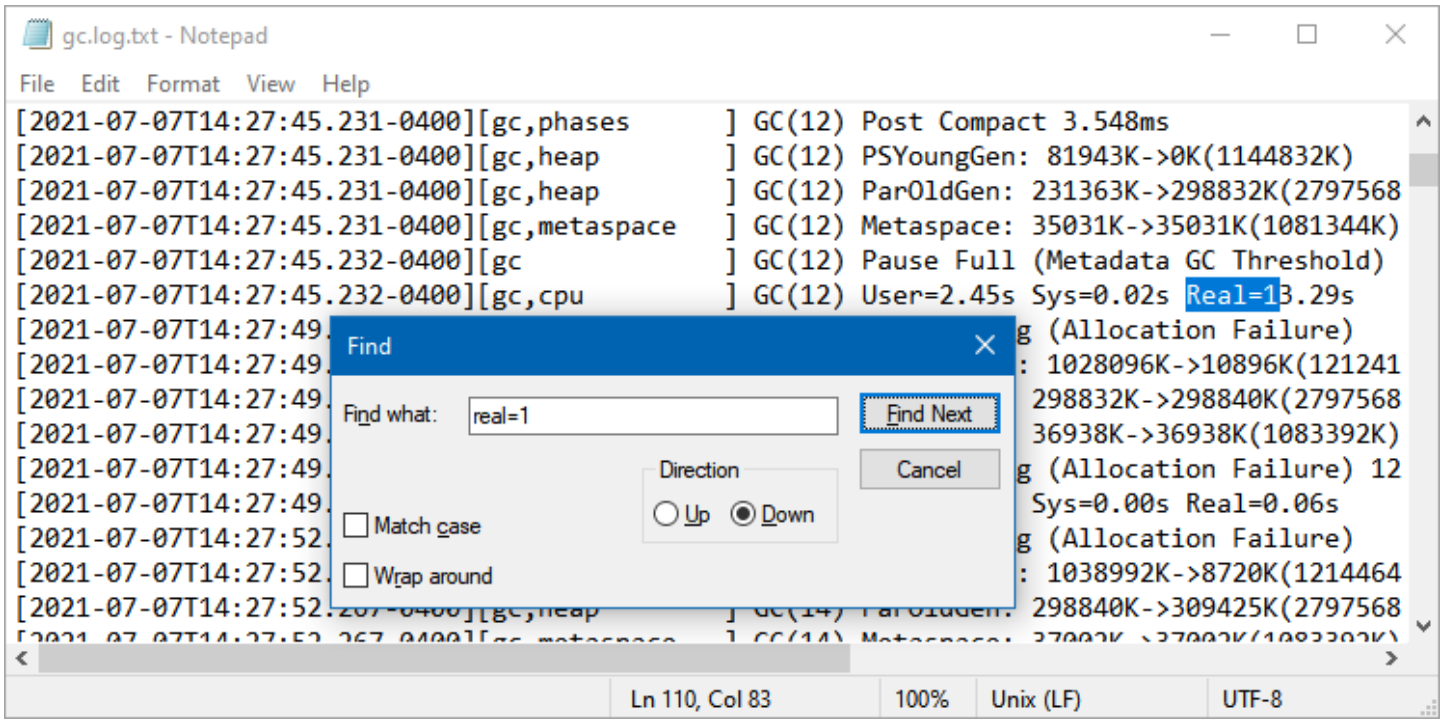
Analyze the Logs

Use the following steps to review the garbage collection log files.

1. From the Start page, click the **STEP System Administration** link and log in.
2. On the Logs tab, click the **Fetch Data** button.



3. On the **gc.log** row (Main Garbage Collection log file) click the **Download** link.
4. Open the downloaded file (for example, with Notepad++) and search for **real=1** repeatedly, noting the date to review the status.



Based on your findings, consider the following actions:

- Garbage collections that take more than 10 seconds are an indication of a problem.
- Within a single day, more than 20 garbage collections with a real time >10 seconds indicates that heap size should be reconsidered. Submit a Jira ticket for assistance.

Database Long-Lasting SQL Queries

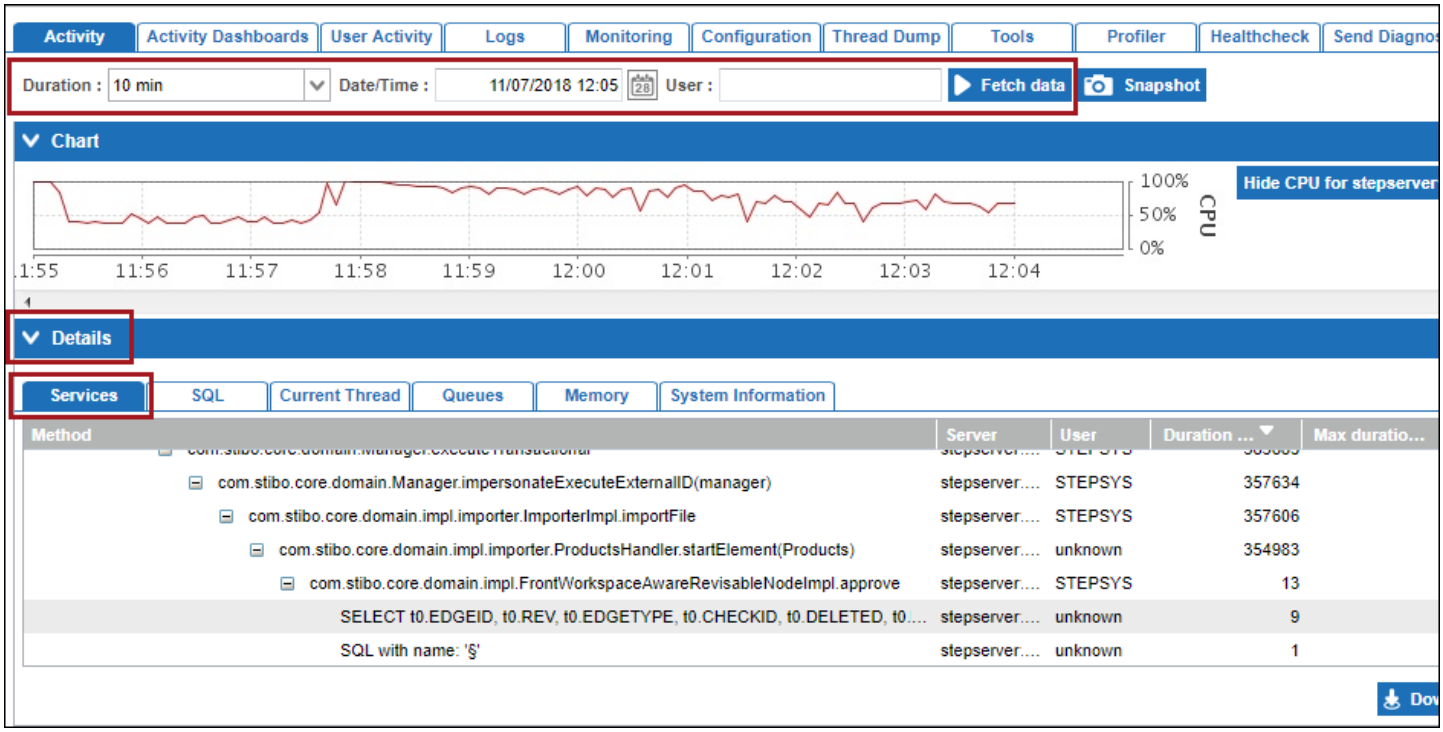
This is one of the technical infrastructure data gathering methodologies and performance recommendations for server machines. The full list is defined in the **Server Environment Recommendations** topic.

Any long-lasting SQL queries that reach the hundreds of milliseconds or more indicate that the application server is waiting for the Oracle database. Waiting for the Oracle database can be caused by latency issues, optimistic locking issues, etc. The cause should be analyzed further.

For more information, refer to the **Activity** topic in the **Administration Portal** documentation.

Analyze Long-Lasting SQL Queries

1. From the Start Page, click the **STEP System Administration** button and supply the login credentials.
2. On the **Activity** tab, set the Duration and Date / Time parameters according to when the performance issues were noticed.
3. Click the **Fetch data** button to load the data.

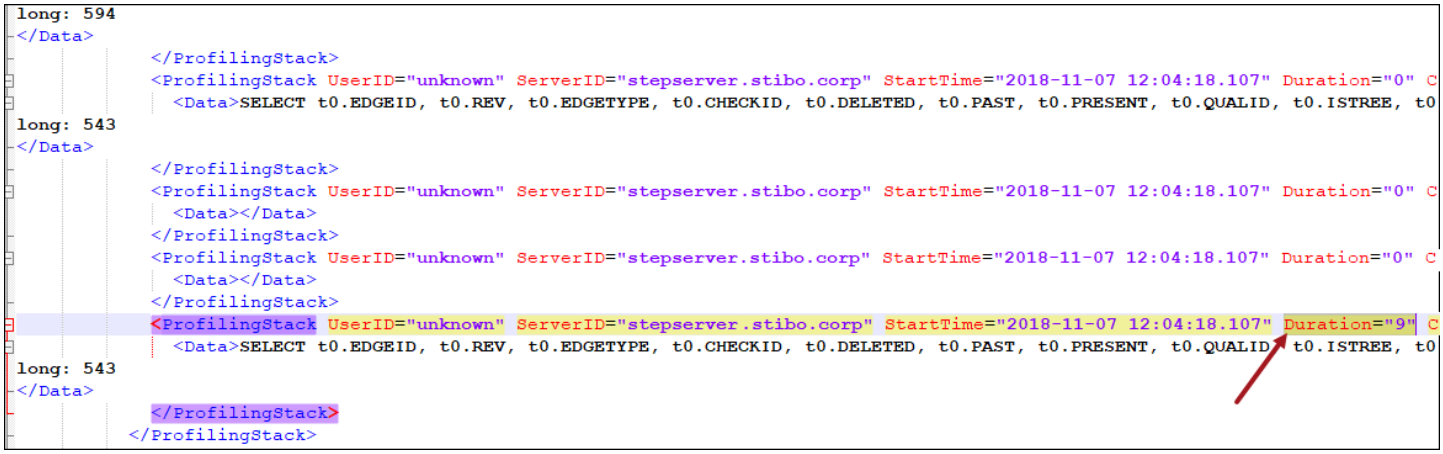


4. Click the Details heading to display the Services tab.
5. Click the Services tab and drill down to the level where the SQL queries are visible to determine if the SQL query is long-lasting, specifically, more than hundreds of milliseconds.
6. Compare the longest-lasting SQL query with the longest-lasting SQL query under the **SQL** tab using the 'Max duration' column.

In this example, the maximum duration of a SQL query was 9 milliseconds, which is fast. Nevertheless, if long-lasting SQL queries are found with hundreds of milliseconds, then this indicates that the solution is waiting for the Oracle database and further analysis is necessary.

Alternatively, you can download the profiling data and look up the longest-lasting query in the XML, as follows:

1. Click the **Download data** button.
2. Open the downloaded file.
3. Search for the largest duration which was displayed under the 'Max Duration' column in the SQL tab.



Database Server Redo Log

This is one of the technical infrastructure data gathering methodologies and performance recommendations for server machines. The full list is defined in the **Server Environment Recommendations** topic.

The STEP database has a redo log for recovery operations. The redo log consists of two or more pre-allocated files that store all changes made to the database as they occur. Every instance of the database has an associated redo log to protect the database in case of an instance failure.

Every time the database performs, a redo log switch triggers a database checkpoint, which brings the database to a halt while the checkpoint runs. The general performance of the database is impacted if the redo log switch is performed too often.

Important: The database should perform a redo log switch of about 50 times per day (every 30 minutes) and the redo log switches per day should be stable.

Follow these steps to analyze the redo log file of the database via the DB ToolBox:

1. On the application server, go to the DB ToolBox, which is typically saved in the '/opt/stibo/step/admin/app-server-toolbox/' folder.
2. Start the ToolBox using one of these commands:
   - `./db-toolbox.pl`
   - `perl db-toolbox.pl`
3. Execute DB ToolBox number 42: 'Show redolog history'

Example

This database performed a redo log switch 2-to-3 times per minute in several days over the last 28 days. That negatively impacts the general performance of the database. Additionally, over the last 28 days, the redo log switch varies from about 20 times per day to about 500 times per day.

The redo log file size is 256 MB.

```
Redolog groups, members and sizes
++++++++++++++++++++++++++++++++

    GROUP# MEMBER                                              MB_SIZE
---------- -------------------------------------------------- ----------
         1 /database/dbredo1/oradata/step/redo01a.log                256
         1 /database/dbredo2/oradata/step/redo01b.log                256
         2 /database/dbredo1/oradata/step/redo02a.log                256
         2 /database/dbredo2/oradata/step/redo02b.log                256
         3 /database/dbredo1/oradata/step/redo03a.log                256
         3 /database/dbredo2/oradata/step/redo03b.log                256
         4 /database/dbredo1/oradata/step/redo04a.log                256
         4 /database/dbredo2/oradata/step/redo04b.log                256
         5 /database/dbredo1/oradata/step/redo05a.log                256
         5 /database/dbredo2/oradata/step/redo05b.log                256
         6 /database/dbredo1/oradata/step/redo06a.log                256
         6 /database/dbredo2/oradata/step/redo06b.log                256
```

The statistics of the number of redo log switches per day are as follows:

```
Redolog switches per day (last 28 days)
+++++++++++++++++++++++++++++++++++++++

DAY               COUNT(*)
--------------- ----------
20170930                60
20171001                75
20171002               149
20171003                84
20171004                81
20171005                99
20171006               407
20171007               445
20171008               351
20171009               459
20171010               389
20171011               500
20171012                49
20171013               113
20171014               384
20171015               419
20171016               384
20171017               344
20171018               324
20171019               431
20171020               453
20171021               142
20171022                24
20171023                79
20171024                49
20171025                98
20171026                70
20171027                23
```

In this example, the recommendation is to increase the size of the redo log files, and to minimize the number of redo log switches to one every half hour. This will positively impact the database performance and can be achieved by:

- Increasing the size of the redo log files from 256 MB to 5 GB.
- Setting the database parameter ARCHIVE_LAG_TARGET=1800, which will make it switch log files every 30 minutes.

Database Server Alert Log

This is one of the technical infrastructure data gathering methodologies and performance recommendations for server machines. The full list is defined in the **Server Environment Recommendations** topic.

The DB Toolbox includes options for many troubleshooting tools.

Show DB Alert Log

If the alert log feature is enabled in the database, the database alert log can be analyzed via DB ToolBox, as follows:

1. On the application server, go to the DB ToolBox, which is typically saved in the '/opt/stibo/step/admin/app-server-toolbox/' folder.
2. Start the ToolBox using one of these commands:
   - `./db-toolbox.pl`
   - `perl db-toolbox.pl`
3. Execute DB ToolBox number 36: 'Show DB alert-log'.

Show Tablespace Usage Report

Determine the size of the TEMP tablespace using the DB Toolbox as follows:

1. On the application server, go to the DB ToolBox, which is typically saved in the '/opt/stibo/step/admin/app-server-toolbox/' folder.
2. Start the ToolBox using one of these commands:
   - `./db-toolbox.pl`
   - `perl db-toolbox.pl`
3. Execute DB ToolBox number 1: 'Show tablespace usage report'.

Example

Use the DB ToolBox and run the 'Show DB alert log' on the database to show out-of-space errors for the TEMP tablespace.

```
07-OCT-2017 02:01:03 ORA-1652: unable to extend temp segment by 128 in tablespace
TEMP
07-OCT-2017 02:07:05 ORA-1652: unable to extend temp segment by 128 in tablespace
TEMP
07-OCT-2017 02:12:52 ORA-1652: unable to extend temp segment by 128 in tablespace
TEMP
07-OCT-2017 02:18:21 ORA-1652: unable to extend temp segment by 128 in tablespace
```

```
TEMP
07-OCT-2017 02:24:35 ORA-1652: unable to extend temp segment by 128 in tablespace
TEMP
```

This mainly occurs every weekend, but has also happened several times during weekdays. These errors indicate that important STEP tasks are failing, and should therefore be avoided.

Running the 'Show tablespace usage report' shows the current size of TEMP tablespace is one file of 32 GB.

```
Tablespace usage report
++++++++++++++++++++++

Number in parantheses specifies instance number

TBS                 TYPE            MB_MAXTBS_SIZE MB_FILE_SIZE    MB_USED   MB_AVAIL
------------------- --------------- -------------- ------------ ---------- ----------
UNDOTBS1 (1)        UNDO                     15357        15360      15302         55
USERS               PERMANENT                   99          100          4         95
PATROL              PERMANENT                   97           98          1         97
SYSTEM              PERMANENT                 1499         1500        689        810
SYSAUX              PERMANENT                 8191         8192       6071       2120
TEMP                TEMPORARY                30719        30720        131      30588
STEPSYSDATA         PERMANENT               399256       399360     362848      36408
STEPSYSBLOB         PERMANENT               139104       139264      80960      58144
                                                     ------------ ----------
sum                                                      594594     466006

8 rows selected.
```

In this example, the recommendation is to add at least one extra file of 32 GB, so the total size of the TEMP tablespace will be increased to 64 GB.

Additionally, other database alter log errors may include, for instance:

- ORA-00060 errors: deadlock detected
- ORA-01555 errors: snapshot too old

The further recommendation would be to involve a Stibo Systems system architect to analyze, recommend, and execute on the database alert errors, as well as on generic infrastructure and system setup.

Database Server Oracle Adaptive Features

This is one of the technical infrastructure data gathering methodologies and performance recommendations for server machines. The full list is defined in the **Server Environment Recommendations** topic.

Oracle version 12.1.0.2. includes a known issue (bug) in which a 'directive' is used. For some queries, this means a poor execution plan, and response times go from milliseconds to seconds.

This has quite an unfortunate impact on performance, and when displaying execution plans for SQL queries, shows up as an 'adaptive plan' by Oracle.

Recommendation

- In Oracle version 12.1.0.2, the parameter should be set to FALSE.
- Once Oracle is upgraded to 12.2.0.1 or greater, reassess the parameter setting.

Database Server Oracle Statistics

This is one of the technical infrastructure data gathering methodologies and performance recommendations for server machines. The full list is defined in the **Server Environment Recommendations** topic.

After upgrade, the Oracle data statistics should be renewed for STEP in order to collect a complete new set of statistics for all indexes / tables in STEPSYS schema.

The following information is about STEP and the way database statistics are maintained for the STEPSYS schema:

- STEP has its own database job (GATHER_STATS_STEP) which, by default, is executed (executing the function stepsys.adminpck.table_manage) every night at 23.30h. This job collects new statistics on indexes / tables in STEPSYS schema, where more than 10 percent of rows has been inserted / updated / deleted since last time, statistics were collected on the given object.
- The normal Oracle built-in scheduled job for collecting statistics on all indexes / tables in the database must be configured to only collect statistics for Oracle's own objects (SYS schema). Therefore, the parameter AUTOSTATS_TARGET, must be set to ORACLE.
- Some core tables in the STEPSYS schema are locked and without any statistics. This because the number of rows in the tables change too frequently, and the statistics would therefore never be up to date. In order to make Oracle choose the right path of execution, when executing queries against these tables, STEP instead uses the built-in Oracle feature 'dynamic samplic' (Oracle init parameter DYNAMIC_SAMPLING=2), which makes Oracle generate some statistics for the given object at runtime.

Collect a complete new set of statistics for all indexes / tables in STEPSYS schema as follows:

1. Log onto the database as user STEPSYS (or sqlplus, or the like).
2. Execute the following command:

```
Exec adminpck.analyze(true);
```

Database Server Oracle DataGuard

This is one of the technical infrastructure data gathering methodologies and performance recommendations for server machines. The full list is defined in the **Server Environment Recommendations** topic.

The Oracle database can run with DataGuard protection and is configured with 'SYNC AFFIRM' (Maximum Availability mode).

However, running with DataGuard introduces an overhead on the STEP system, namely:

- The database must run in 'Force Logging' mode. This ensures that ALL writings performed on Primary database are mirrored correctly to Standby database. STEP uses some internal tables for building up temporary result sets for searches and exports, and these are created with 'nologging' in order to reduce impact on performance. Data stored in these tables is considered as interim and not important in a restore / recover scenario.
- DataGuard can be configured to run with different levels of protection, depending on the SLA requirements. Choosing 'SYNC AFFIRM' means going for nearly the highest protection level. With this level, all writings (even a write of a temporary data set built for a search) has to be written to the other site in order to keep the two databases in sync. 'SYNC AFFIRM' also means that the writing will have to be written to the disk on secondary database server (Standby) before the Primary database server can send back an acknowledgment to the App server.

A test is recommended where the protection level is lowered and the results in performance are analyzed.

For more information, refer to the **High Availability Implementation** topic.

DTP Server

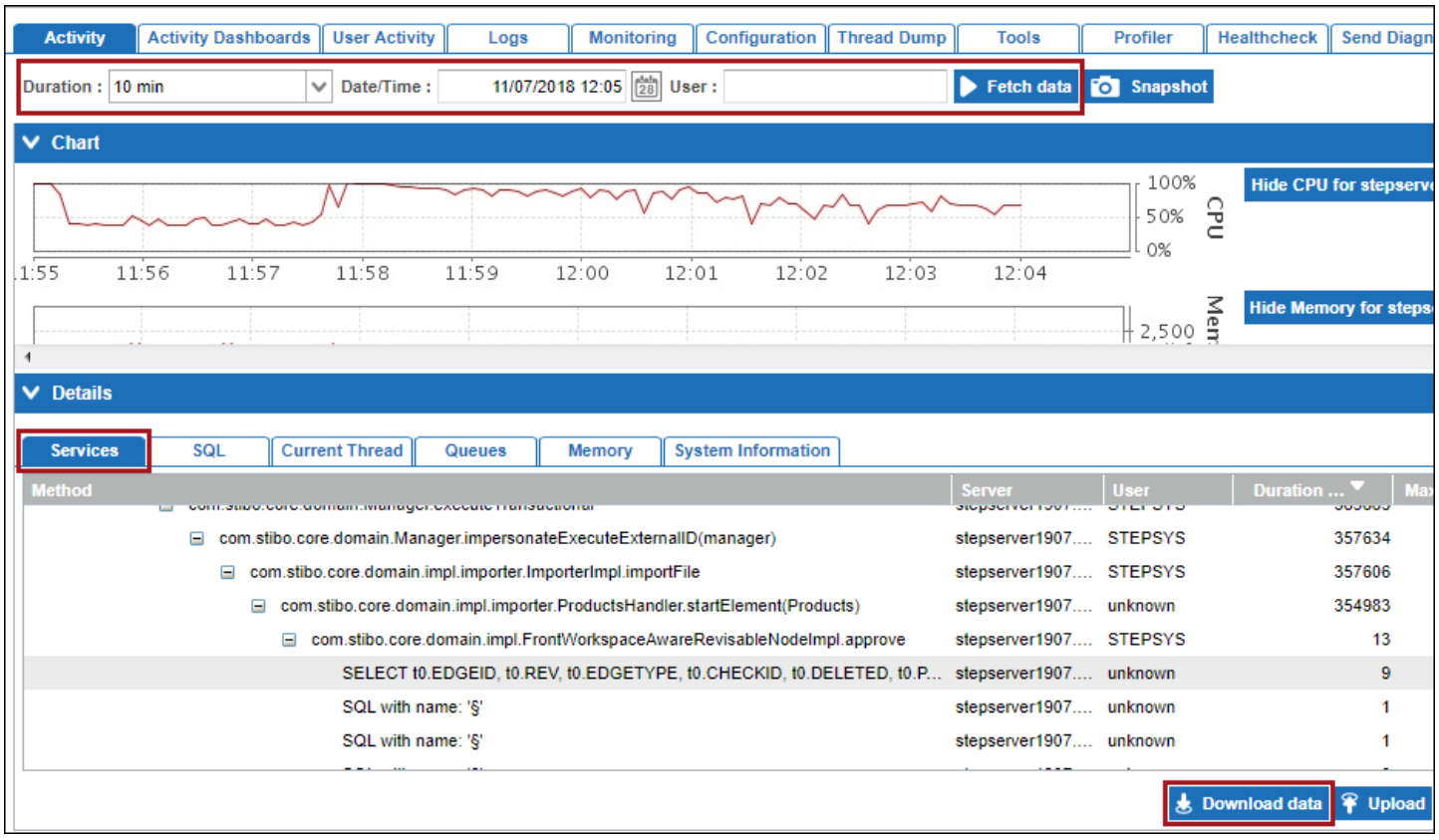
This is one of the technical infrastructure data gathering methodologies and performance recommendations for server machines. The full list is defined in the **Server Environment Recommendations** topic.

A DTP server can cause performance loss when the application server is waiting for the DTP server, for example, due to latency issues. The cause should be analyzed further.

For more information, refer to the **Activity** topic in the **Administration Portal** documentation.

Analyze the DTP server waiting as follows:

1. From the Start Page, click the **STEP System Administration** button and supply the login credentials.
2. On the **Activity** tab, set the Duration and Date / Time parameters according to when the performance issues were noticed.



3. Click the **Fetch data** button to load the data.
4. Click the Details heading to display the Services tab.
5. Click the **Download data** button.
6. Open the downloaded XML file.
7. Search for the 'com.stibo.ddsconnector' or 'com.stibo.idsconnector' service calls.

8. Determine if the duration is significantly longer than expected. If so, consult Stibo Systems to look into the DTP Server.

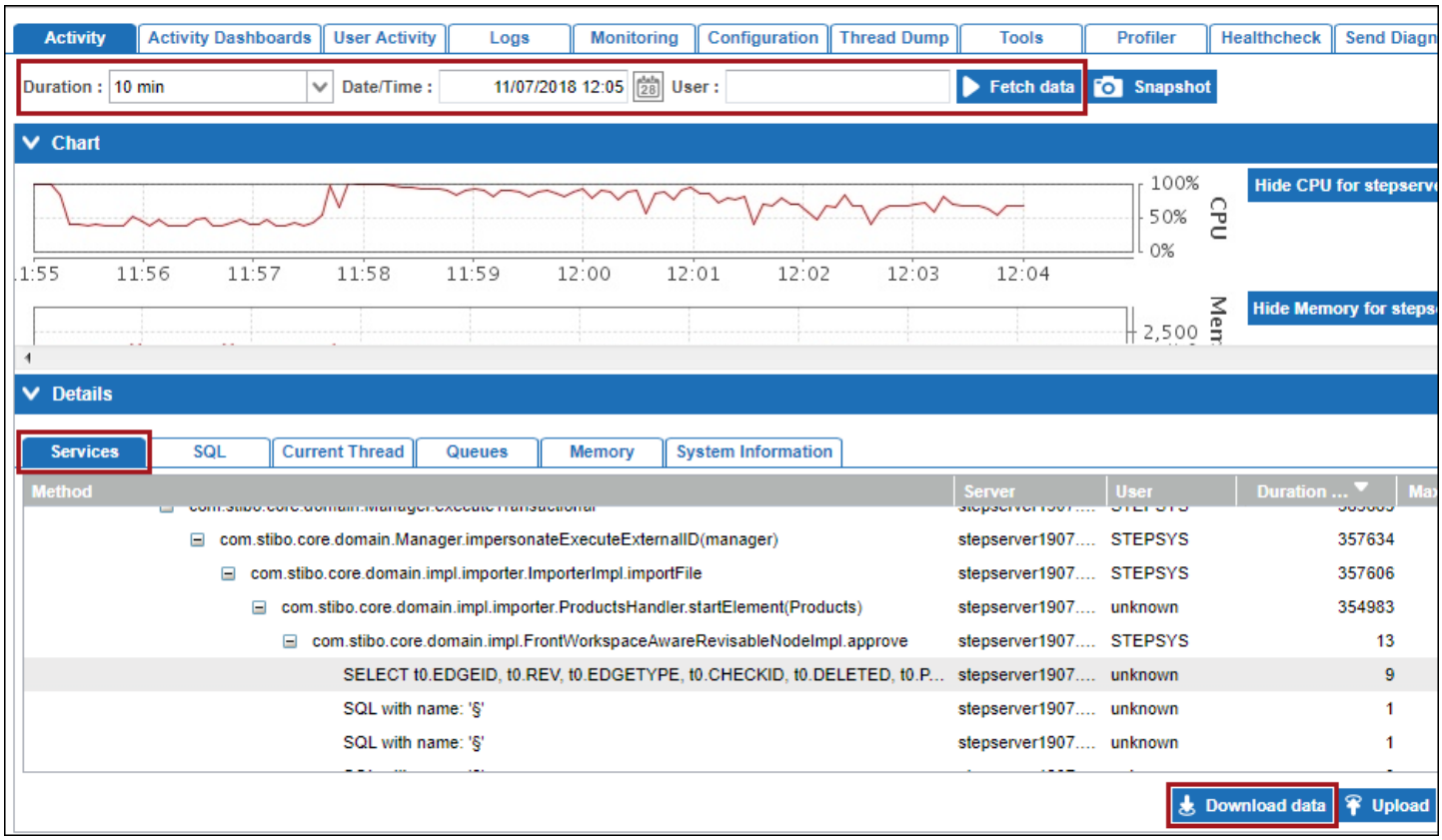
Custom Extensions

This is one of the technical infrastructure data gathering methodologies and performance recommendations for server machines. The full list is defined in the **Server Environment Recommendations** topic.

Custom extensions can cause performance loss. For example, this can occur when custom extensions are relatively old, were delivered for an out-of-date STEP version, and if STEP has been upgraded to newer versions many times.

Long-lasting custom extensions can be analyzed via the STEP Admin Portal.

1. From the Start Page, click the **STEP System Administration** button and supply the login credentials.
2. On the **Activity** tab, set the Duration and Date / Time parameters according to when the performance issues were noticed.



3. Click the **Fetch data** button to load the data.
4. Click the Details heading to display the Services tab.
5. Click the **Download data** button.
6. Open the downloaded XML file.
7. Search for the 'com.stibo.customer' service call.

8. Determine if the duration is significantly longer than expected. If so, consult Stibo Systems to look into the DTP Server.

Server Hygiene

This is one of the technical infrastructure data gathering methodologies and performance recommendations. The full list is defined in the **Technical Performance Recommendations** topic.

This section describes the data gathering methodologies from a technical infrastructure viewpoint for the clean up of the STEP servers.

- Analyze Application Server Disk Space
- Clean Up Import Files
- Clean Up Unnecessary Files
- Analyze Background Processes
- Create a Background Processes Maintenance Plan
- Clean Up Background Processes

Analyze Application Server Disk Space

This is one of the technical infrastructure data gathering methodologies and performance recommendations for server hygiene. The full list is defined in the **Server Hygiene** topic.

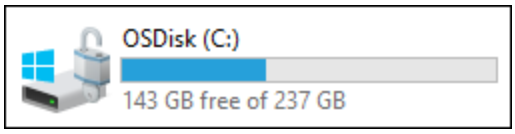
When the system is generally slow, it can be caused by an over-burdened system.

Analyze every application server for ample available disk space as follows:

1. Based on the type of server, check the available disk space on every application server:
   - For a Linux application server, use the `df -h` command.

```
Filesystem            Size  Used Avail Use% Mounted on
/dev/mapper/vg_vsv1g186-lv_root
                       50G  8.5G   39G  19% /
tmpfs                 7.8G   72K  7.8G   1% /dev/shm
/dev/sda1             477M   42M  410M  10% /boot
/dev/mapper/vg_vsv1g186-lv_home
                       25G   45M   24G   1% /home
/dev/mapper/vg_vsv1g186-lvstiprd01
                       40G   13G   26G  33% /opt/stibo
10.1.1.150:stibosw    2.9T   65G  2.8T   3% /opt/stiprd
```

   - For a Windows application server, use Windows File Explorer.



> **Note:** If the analysis shows 90 percent or more of the volume is used, it is a good indication that the system is overloaded.

2. From the Start Page, click the STEP System Administration button and supply the log in credentials.
3. On the Logs tab, click the **Fetch data** button to load the logs.
4. On the **Main STEP Log file** row, click the **View** link.
5. Search for the following error message:

```
… FileUtils.copyFile(): failed to copy file (tried multi) …
```

> **Note:** If this error is found, a folder has reached its maximum limit of files and the system is overloaded.

6. If the system is overloaded, continue with the **Clean Up Import File** topic to analyze and resolve the issue.

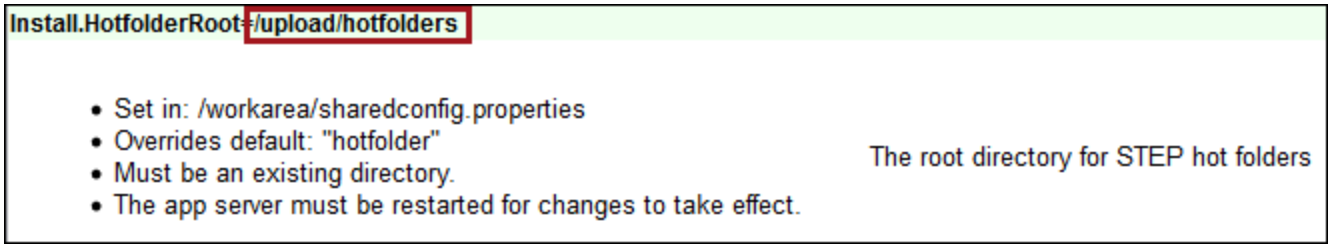
Clean Up Import Files

This is one of the technical infrastructure data gathering methodologies and performance recommendations for server hygiene. The full list is defined in the **Server Hygiene** topic.

When the STEP application server is overloaded, it can be caused by the number of import files in hotfolders. Furthermore, large import files can also cause the system to be slow.

Analyze the import files situation as follows:

1. Determine where the import files are stored. On the Start Page, click the **STEP System Administration** button and supply the login credentials.
2. On the Configuration tab, click the **Fetch data** button to load the data.
3. Search for 'Install.HotfolderRoot' and note the hotfolder location.

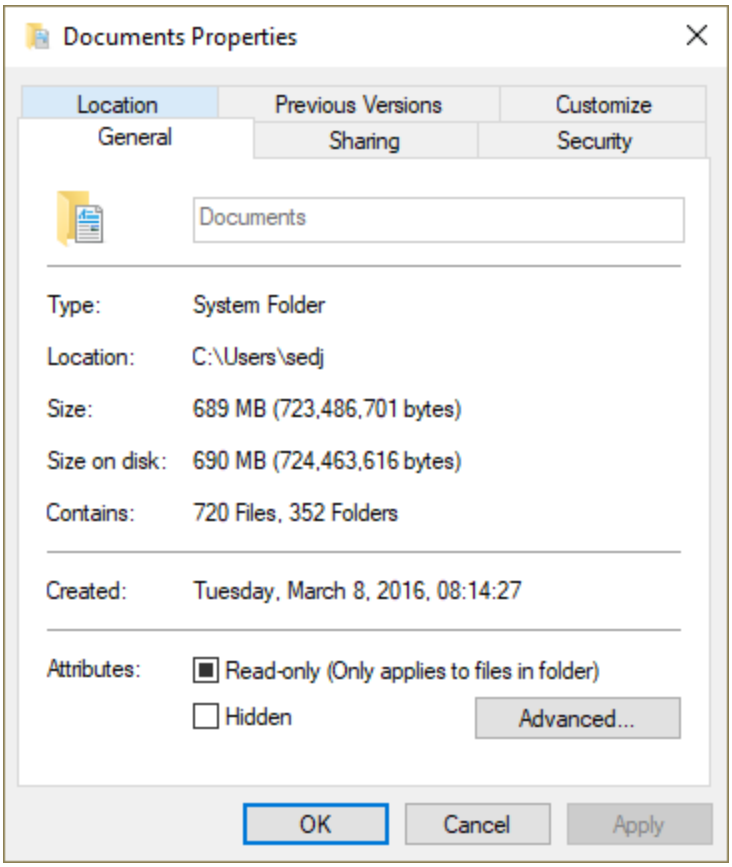


4. Find the number of import files in the hotfolder location.
   - For a Linux application server, use the following command:

```
cd /upload
find hotfolders/ -type f | wc -l
```

```
[stibosw@euuktestepapp01p upload]$ find hotfolders/ -type f | wc -l
find: `hotfolders/ZippedTest': Permission denied
456580
```

   - For a Windows application server, open the Windows File Explorer, and go to the '/upload' location. Right-click the directory and click **Properties**.

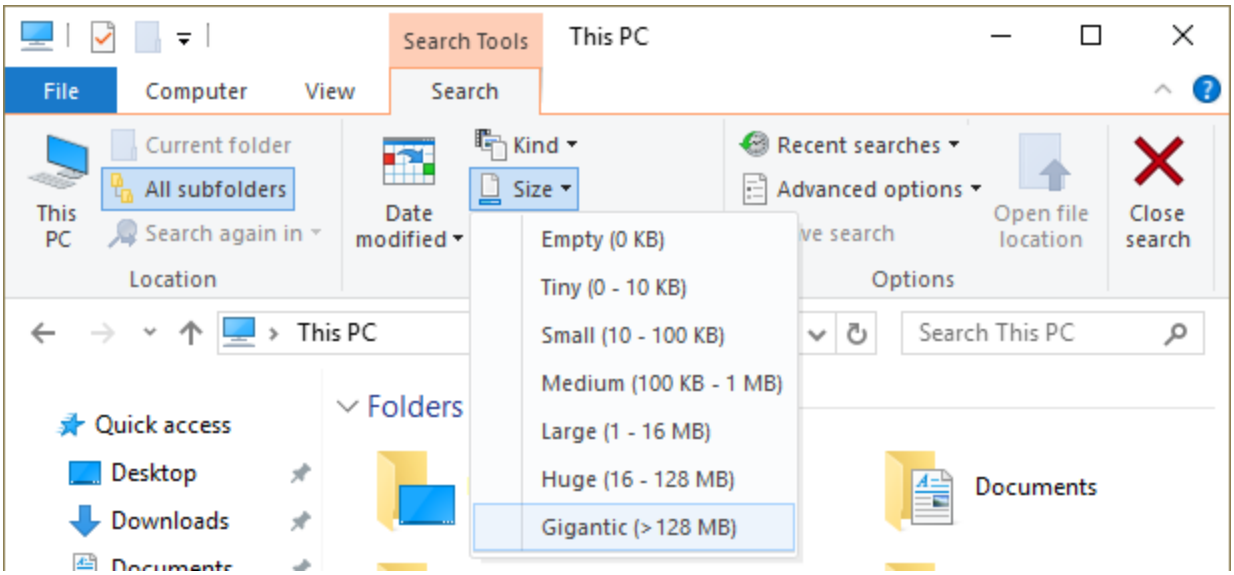


5. Find the size of import files in the hotfolder location.
   - For a Linux application server, use the following command:

```
cd /upload
du -hsx * | sort -rh | head -10
```

```
[stibosw@euuktestepapp01p upload]$ du -hsx * | sort -rh | head -10
du: cannot read directory `hotfolders/ZippedTest': Permission denied
du: cannot read directory `lost+found': Permission denied
69G     hotfolders
16K     lost+found
0       test2
```

   - For a Windows application server, open the Windows File Explorer, and go to the '/upload' location and view the size as shown in the following image.



Identify the Causes

If the number of import files in the hotfolder is in the hundreds of thousands (e.g., 250k files), or if the size of the import files is too large, continue with the following activities to determine the cause.

IIEP 'Keep file after load'

Review the active hotfolder and REST Receiver IIEPs and verify that they are configured to automatically clean up the import files. Set the 'Keep file after load' parameter to 'No' to remove import files after success import. For more information, refer to the **Hotfolder Receiver** topic of the **Data Exchange** documentation.

IIEP failed or completed with errors

Review the IIEPs for those that regularly complete with errors, or fail. In these cases, the import process is saved along with the imported file for error tracing. When the import process errors, the accompanying background processes are not removed. This means neither the import files nor the background processes are not being removed. When an IIEP successfully completes without errors, both the background processes and the import files can be deleted.

IIEPs that encounter Optimistic Locking errors can result in background processes that fail or 'complete with errors.' In this case, consider setting the 'Reference Target Lock Policy' to 'Relaxed' on the object types for which the long transaction applies. For more information, refer to the **Reference Target Lock Policy on Object Types** topic in the **System Setup** documentation.

IIEP non-standard configuration

IIEPs are created with a standard import directory structure as illustrated below. Modifying that structure can prevent the removal of the imported files after successful import and processing.

Folder	Example of Standard Config	Description
root	/upload/hotfolders/products/	
in	/upload/hotfolders/products/in	import files reside, and are removed after processing
save	/upload/hotfolders/products/save	import files remain when they are imported and when configured to keep the import files after successful processing
error	/upload/hotfolders/products/error	import files remain when the import process completed with errors
failed	/upload/hotfolders/products/failed	import files will remain when the import process failed

Remove Import Files

Based on the reason for the excess import files, follow the steps below to remove them manually:

1. If the import files were not removed because the 'Keep file after load' was previously set to 'Yes,' use the following directions to remove the files from the 'save' folder:
   - For a Linux server, use the command in the save folder holding the excess files:

     ```
     rm -rf *
     ```

   - For a Windows server, open Windows File Explorer and delete the files from the 'save' folder.
2. If the import files were not removed for a reason other than the 'Keep file after load' setting, this indicates the import files are tied to background processes.

   Remove the background processes of the IIEP, which also removes the corresponding import files. Refer to the **Clean Up Background Processes** topic for details.

Important: Removing these import files by deleting them directly from the file system will not delete the corresponding background processes.

Clean Up Unnecessary Files

This is one of the technical infrastructure data gathering methodologies and performance recommendations for server hygiene. The full list is defined in the **Server Hygiene** topic.

When the application server is overloaded and it is not due to too many and/or too large import files, it can be caused by large number of leftover or temporary files, e.g., from previous migrations. A large number of these unnecessary files can also cause the system to be slow.

There is no single location for these unnecessary files to reside on the application server. However, sometimes unnecessary files are located in temporary directories such as tmp, temp, or temporary.

Important: Be careful with temporary files the STEP system does use, such as the temporary files in the **step/diag** folder or in the **step/admin/cache** folders. These temporary files must not be removed.

Analyze Background Processes

This is one of the technical infrastructure data gathering methodologies and performance recommendations for server hygiene. The full list is defined in the **Server Hygiene** topic.

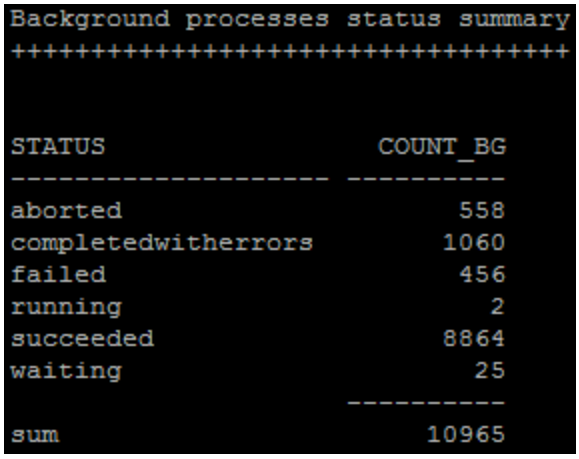
When a process ends in either 'failed' or 'completed with errors' state, by design, background process log information is not deleted automatically. The retained information can be used for manual processing, which allows the background process owner to address the errors before deleting the background process log information manually.

However, keeping many background processes can slow down the system, because each background process creates a sub-directory on the file system, which over time, can overload the system. If the number of background processes are excessive, such as hundreds of thousands, these background processes should be deleted.

Note: The number of background processes in the database and on the application server file system may differ. This typically happens when background processes are not removed properly from the application server.

Check the number of background processes in the database and on the application server as follows:

1. On the application server, go to the DB ToolBox, which is typically saved in the '/opt/stibo/step/admin/app-server-toolbox/' folder.
2. Start the ToolBox using one of these commands:
   - `./db-toolbox.pl`
   - `perl db-toolbox.pl`
3. Execute DB ToolBox number 38: 'Show background process information.'



```
Background processes status summary
+++++++++++++++++++++++++++++++++++


STATUS                 COUNT_BG
-------------------- ----------
aborted                     558
completedwitherrors        1060
failed                      456
running                       2
succeeded                  8864
waiting                      25
                     ----------
sum                       10965
```

4. On the application server file system, determine the number of files in directory 'background-processarea' recursively, based on your server type:
   - On a Linux server, run the command:

```
find background-processarea/ -type f | wc -l
```

```
[stibosw@lep1sta1 workarea]$ find background-processarea/ -type f | wc -l
1196277
```

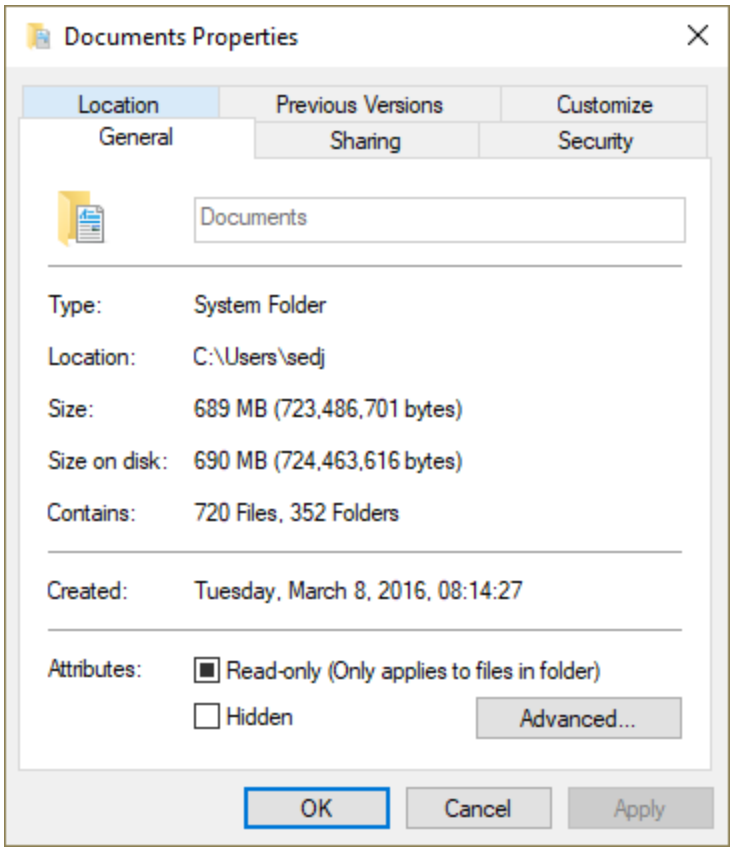
   Optionally, analyze the sub-directories as well. Use the following command to display the files created last:

```
ls -lat
```

   And this command to display the largest files:

```
du -hsx * | sort -rh | head -10
```

   - On a Windows server, open the Windows File Explorer, go to the 'background-processarea' directory, right-click and select the Properties option.



Once you determine that there is an excessive number of background processes in the database and/or on the application server file system, the recommendation is to create a maintenance plan to ensure the number will be controlled and clean up the existing background processes. For more information, refer to the **Create a Background Processes Maintenance Plan** in this guide.

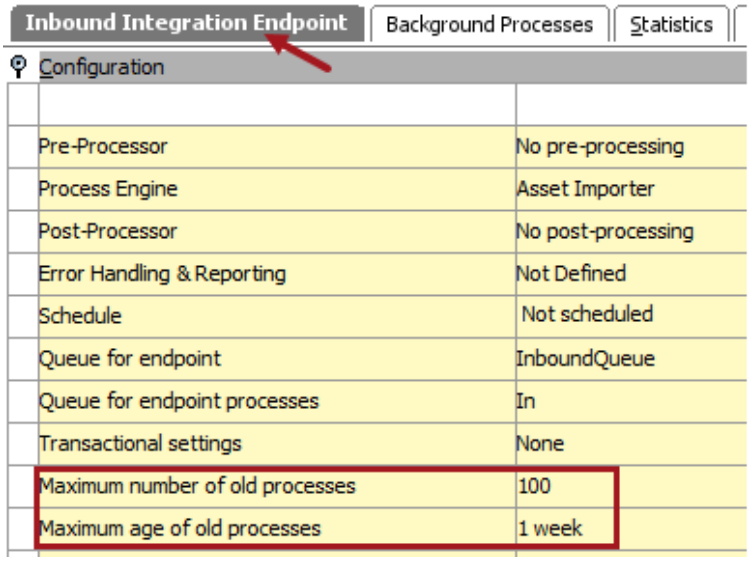
Create a Background Processes Maintenance Plan

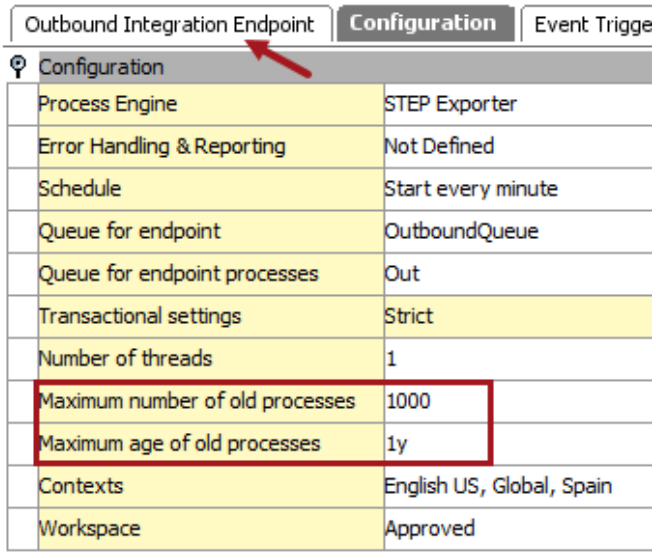
This is one of the technical infrastructure data gathering methodologies and performance recommendations for server hygiene. The full list is defined in the **Server Hygiene** topic.

Settings on IEPs and on background process allow configuration of automatic clean-up activities.

IEP Auto Delete Settings

Use the following parameters to configure automatic deletion of background processes on IIEPs and OIEPs:





- **Maximum number of old processes**: Specify the number of ended processes the system will keep. Succeeded and ended processes are deleted when the number exceeds the specified limit. The oldest processes are deleted first. Setting this number too high may eventually degrade performance.

  For example, if 'Maximum number of old processes' is set to '1000,' a maximum of 1000 succeeded and ended background processes will be retained. The oldest background processes are deleted automatically when the number exceeds the specified limit of 1000.

- **Maximum age of old processes**: Specify the maximum age of ended processes that the system will keep. Ended processes are deleted when the maximum age is exceeded. Setting this number too high may eventually degrade performance.

  For example, if 'Maximum age of old processes' is set to '1 y,' the background processes older than 1 year are deleted automatically.

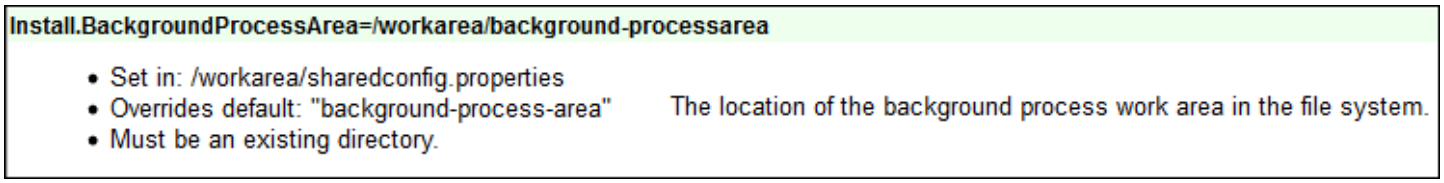
Ensure the configurations represent a realistic situation as demonstrated below:

- In an environment with a large number of small imports and exports, use '1000' as a maximum number of old processes and '1 w' as a maximum age of old processes.
- In an environment with small number of large imports and exports, use '50' as a maximum number of old processes and '1 m' as a maximum age of old processes.

Background Process Auto Delete Settings

Automatic deletion of background processes is also managed in the sharedconfig.properties file. The configuration defines the number of hours after the background processes ends that it will be deleted, and is based on the specified background process template.

1. Determine the location of the background processes on the application server as follows:
   - From the Start Page, click the **STEP System Administration** button and supply the login credentials.
   - On the **Configurations** tab, set the Duration and Date / Time parameters according to when the performance issues were noticed.
   - Click the **Fetch data** button to load the data.
   - Search for the 'Install.BackgroundProcessArea' parameter and note the location. In the following image, the location is '/workarea/background-processarea.'



2. On the application server, use the following commands to display the available process templates.

```
cd /workarea/background-processarea
ls -l
```

```
[stibosw@lep1sta1 ~]$ cd /workarea/
[stibosw@lep1sta1 workarea]$ cd background-processarea/
[stibosw@lep1sta1 background-processarea]$ ls -l
total 2960
drwxrwxr-x    3 stibosw stibosw    4096 Oct  9 20:43 AnalyzeChangePackage
drwxrwxr-x    2 stibosw stibosw    4096 Oct  4 15:35 ApproveRecursive
drwxrwxr-x    3 stibosw stibosw    4096 Oct 10 11:13 AssetDownload
drwxrwxr-x  166 stibosw stibosw  131072 Oct 15 07:35 BulkUpdate
drwxrwxr-x    2 stibosw stibosw    4096 Oct  6 13:59 CreateColFromFile
drwxr-xr-x    2 stibosw stibosw    4096 Jun 13 12:49 DataProfiler
drwxrwxr-x   15 stibosw stibosw    4096 Oct 13 13:03 DownloadReport
drwxrwxr-x    2 stibosw stibosw    4096 Oct 13 16:35 EmptyRecycleBin
drwxrwxr-x    2 stibosw stibosw    4096 Oct 13 16:01 EventProcBGP
drwxrwxr-x    2 stibosw stibosw    4096 Jul  5  2016 Event Queue Delete
drwxrwxr-x   78 stibosw stibosw   12288 Oct 14 13:04 ExcelExportDownload
drwxrwxr-x  498 stibosw stibosw   69632 Oct 13 17:41 Exporter
drwxrwxr-x 4990 stibosw stibosw  397312 Oct 13 15:12 Importer
drwxrwxr-x 7338 stibosw stibosw 1970176 Oct 15 06:46 Inbound
drwxrwxr-x    2 stibosw stibosw    4096 Sep 14 19:30 InboundPoller
drwxrwxr-x    2 stibosw stibosw    4096 Oct  4 19:36 InstallChangePackage
drwxrwxr-x    2 stibosw stibosw    4096 Sep 14 14:31 NodeListToCollection
drwxrwxr-x 3107 stibosw stibosw  294912 Oct 15 07:40 Outbound
drwxr-xr-x    2 stibosw stibosw    4096 Sep  4 09:52 OutboundPoller
drwxrwxr-x    2 stibosw stibosw    4096 May 12  2016 PortalDownload
drwxrwxr-x   37 stibosw stibosw    8192 Mar 22  2016 PortalUpload
drwxrwxr-x    2 stibosw stibosw    4096 Oct 14 01:30 PurgeRevisions
drwxrwxr-x    2 stibosw stibosw    4096 Oct  5 10:37 Refresh Node Collection
drwxrwxr-x    2 stibosw stibosw    4096 Feb 10  2017 RemoveFromSTEPWorkflow
drwxrwxr-x    2 stibosw stibosw    4096 Oct 12  2016 Report
drwxrwxr-x    2 stibosw stibosw    4096 Feb 13  2017 RepublishService
drwxrwxr-x    2 stibosw stibosw    4096 Jun 22 13:23 ReviveRecursive
drwxrwxr-x   10 stibosw stibosw    4096 Jul 17 20:37 Schedule
drwxrwxr-x    2 stibosw stibosw    4096 Mar  1  2017 SealChangePackage
drwxrwxr-x   59 stibosw stibosw   20480 Oct 13 18:11 SheetImporter
drwxrwxr-x    2 stibosw stibosw    4096 Oct 14 23:06 StateflowDeadline
```

3. Configure the auto-delete hours of the background process templates as follows:
   - Open the sharedconfig.properties file.
   - Search for a section that holds the case-sensitive parameter(s), where the name of template and number of hours should be replaced by the placeholders indicated in the brackets:

     ```
     AutoDeleteBackgroundProcess.AgeInHours.[template]=[hours]
     ```

   - Add or edit the templates to define the number of hours that should elapse prior to the automatic deletion of the template's background process.

     For example, the most important templates are generally Exporter, Importer, Outbound, Inbound, and WebPublisher. The parameter for Exporter can be configured as follows:

     ```
     AutoDeleteBackgroundProcesses.AgeInHours.Exporter = 168
     ```

Clean Up Background Processes

This is one of the technical infrastructure data gathering methodologies and performance recommendations for server hygiene. The full list is defined in the **Server Hygiene** topic.

The configurations for automatically removing the background processes (BGPs) only apply to the succeeded background processes. The 'Failed' or 'Completed with errors' background processes are not deleted automatically to allow for analysis. Once analyzed, these background processes should be removed. The following procedures are available for removing BGPs and each is defined below:

- Remove a small number of BGPs
- Remove a large number of BGPs via Search
- Remove a large number of IEP BGPs via Copy
- Remove a large number of BGPs via Script

Note: Deleting aborted BGPs could lead to multiple (aborted) pollers. The current running endpoint needs to be stopped before deleting a BGP.

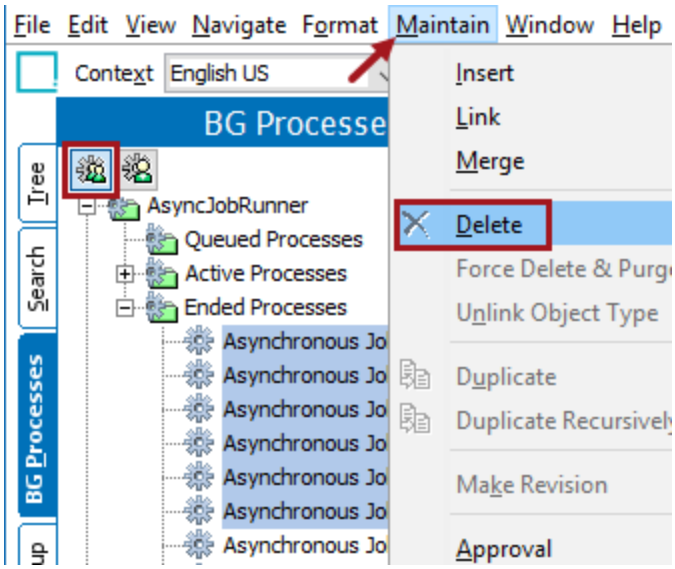
Remove a Small Number of BGPs

If only a small number of BGPs need to be removed, use the following procedures:

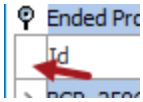
1. For non-IEP processes, in workbench open the BG Processes tab.
2. Click the Background Process View option to display all BGPs (), instead of only the BGPs run by the current user.
3. Open a node, open an 'Ended Processes' folder, and select one or more background processes.

   Note: The 'Queued Processes' and 'Active Processes' background processes should not be deleted.

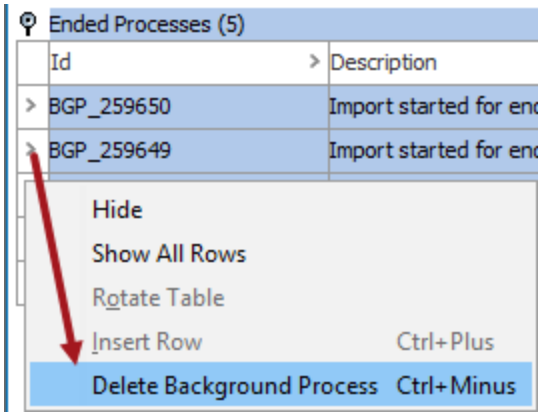
4. Click the Maintain menu and select **Delete** to remove the ended processes.



5. For the IEP processes, on the System Setup tab, select an IEP and open the Background Processes tab.
6. Open the 'Ended Processes' flipper, select the row to be deleted, or select all rows by clicking the empty title on the first row as shown.



7. Right-click on the first column and select the **Delete Background Processes** option.

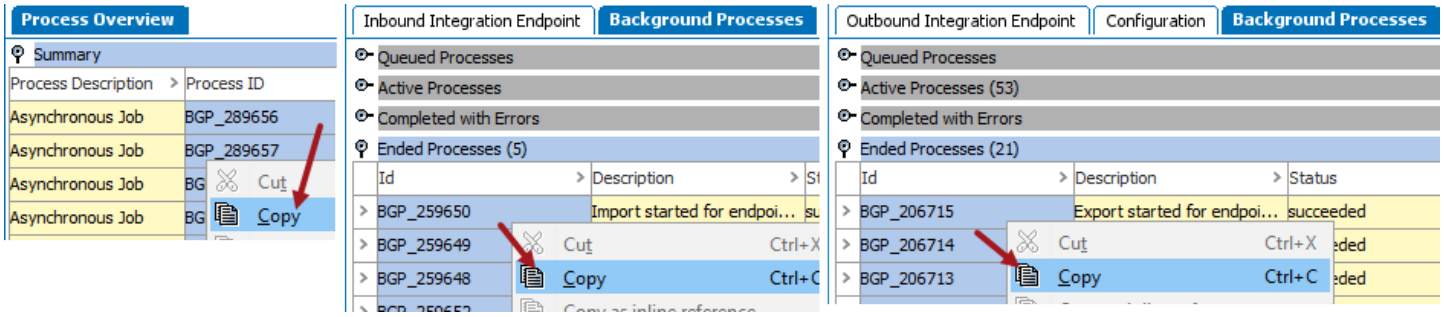


Remove a Large Number of BGPs via Search

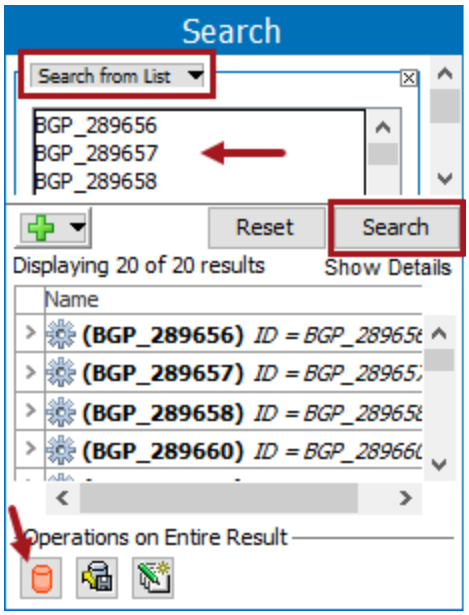
The workbench Search and Collection features can be used to delete a large number of BGPs, using the following procedures.

1. Display the BGPs to be deleted, either on the BG Processes tab or within an IEP (all shown below).
2. Click the header of the ID column (sometimes labeled Process ID) to select all of the BGPs to be deleted.

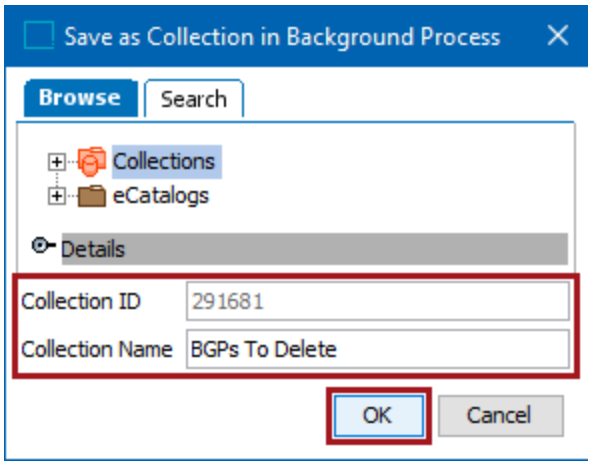
3. Right-click within the selected column and click **Copy** from the menu to copy the list to the clipboard.



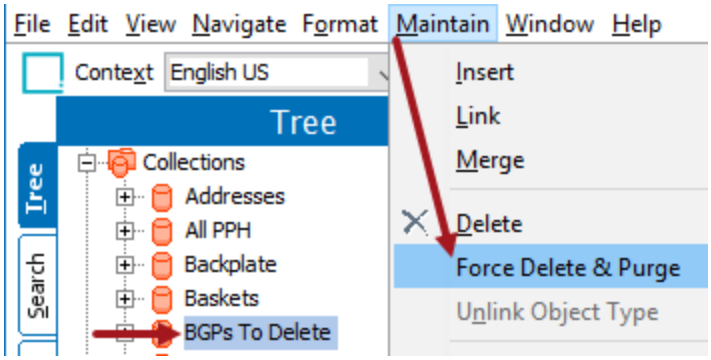
4. Paste the copied list to a text file, for example, in Notepad.
5. On the Search tab, use the dropdown to choose 'Search from List' option.
6. Paste the copied list into the text box and click the **Search** button.



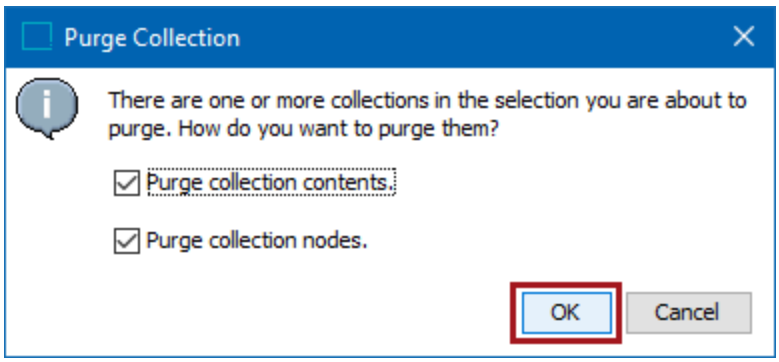
7. Click the collection button ( ) at the bottom of the search panel to add the search results to a collection.
8. Select a folder for the collection, add a collection ID and name, click **OK**. This image shows that ID are auto-generated, so it is automatically supplied by the system.



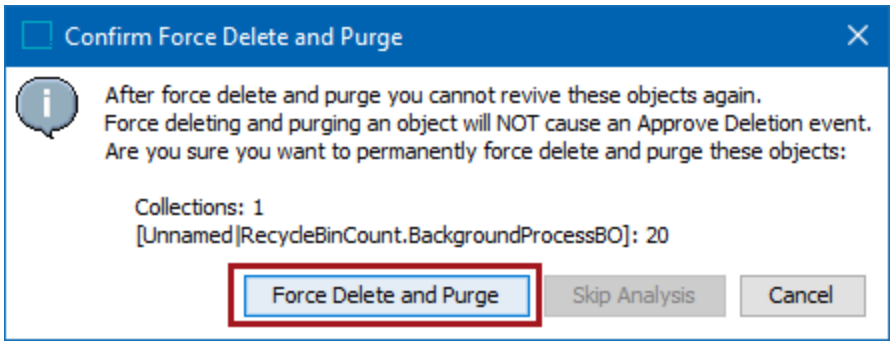
9. On the Tree tab, select the new collection, click the Maintain menu, and click the **Force Delete and Purge** option.



10. On the Purge Collection dialog, check both checkbox options and click **OK**.



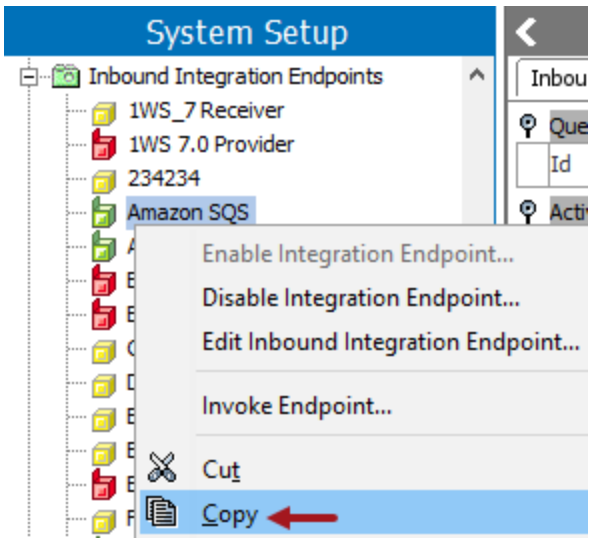
11. On the Confirm Force Delete and Purge dialog, click the **Force Delete and Purge** button. The collection and the listed BGPs are deleted from the system.



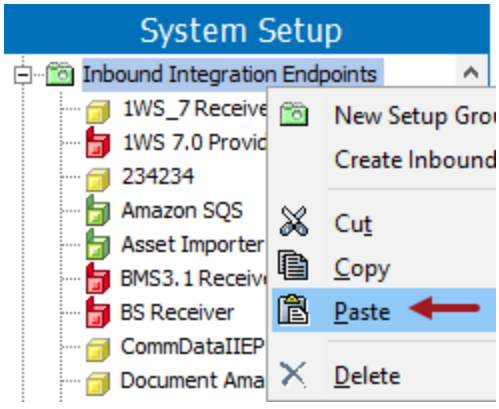
Remove a Large Number of IEP BGPs via Copy

The workbench Copy feature can be used with IEPs to delete a large number of BGPs, using the following procedures:

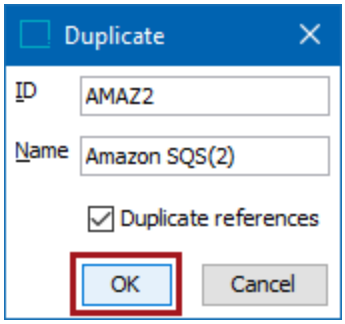
1. On the System Setup tab, select an IEP with a large number of BGPs to be deleted.
2. Right-click the IEP and click **Copy** from the menu.



3. Click the parent IEP folder, right-click and then click **Paste** from the menu.



4. On the Duplicate dialog, add an ID and Name, then click **OK**.



5. When successfully pasted, enable and invoke the new IEP as defined in the **Running an Inbound Integration Endpoint** topic of the **Data Exchange** documentation.
6. Select the original endpoint, right-click and click **Delete** from the menu. The original IEP and the related BGPs are deleted.

Remove a Large Number of BGPs via Script

If the previous methods for removing BGPs are not feasible due to the large number, consult your Stibo Systems representative for assistance in creating a script to remove them.

Performance Recommendations

This section describes the data gathering methodologies from a technical infrastructure viewpoint for the clean up of the STEP servers.

- Attribute and Attribute Group Recommendations
- Background Process Queue Recommendations
- Base Setup Recommendations
- Business Rule Recommendations
- Export Recommendations
- Event Processor and Event Queue Recommendations
- Import Recommendations
- Matching and Linking Recommendations
- Optimistic and Pessimistic Locking Recommendations
- Privilege Recommendations
- Profiling Recommendations
- Revision Control Recommendations
- Scheduled Process Recommendations
- File Hygiene Recommendations
- Web UI Configuration Recommendations

Attribute and Attribute Group Recommendations

This is one of the data gathering methodologies and recommendations for functional performance improvement. The full list is defined in the **Performance Recommendations** topic.

Configured attributes are:

- Specification or description
- Of a certain value type (text, list of value, number, date, etc.)
- Multi-valued or single valued
- Associated with a unit of measure or not.

While none of these configurations independently have a direct impact on performance, some attribute group and attribute configurations can influence performance.

Note: A review of the configured attribute groups and attributes is a good start in analyzing the configuration. Export the attribute groups to a STEPXML file and the attributes to an Excel file for review.

Attribute Groups

The following parameters can affect performance and apply to attribute groups:

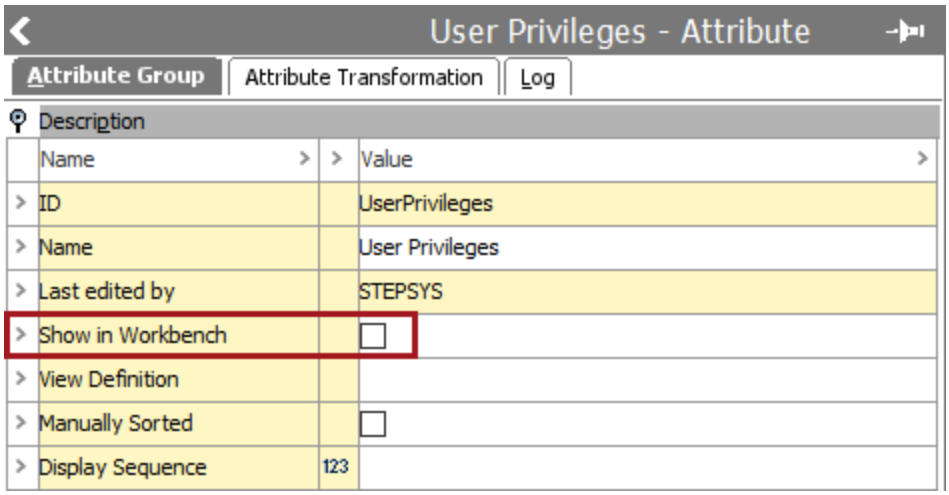
- Show in Workbench
- Manually Sorted

Show in Workbench

In System Setup, the 'Show in Workbench' parameter on an attribute allows the groups to be included or excluded from the workbench display. Showing many attribute values in the workbench requires additional processing power, and can become expensive in terms of performance, especially for the values of calculated attributes.

To review the attributes being displayed in the workbench, export all attribute groups as STEPXML and in the output file search for **ShowInWorkbench ="true"**. Determine if any of the attributes being displayed should be hidden.

Disable the 'Show in Workbench' option to prevent the attribute group from being displayed.



For better performance, set the 'Show in Workbench' parameter as enabled only for attribute groups that need to be displayed in the workbench.

Manually Sorted

'Manually Sorted' adds complexity and can cause performance issues. Refer to the **Manually Sorted Recommendations** topic for more information.

Attributes

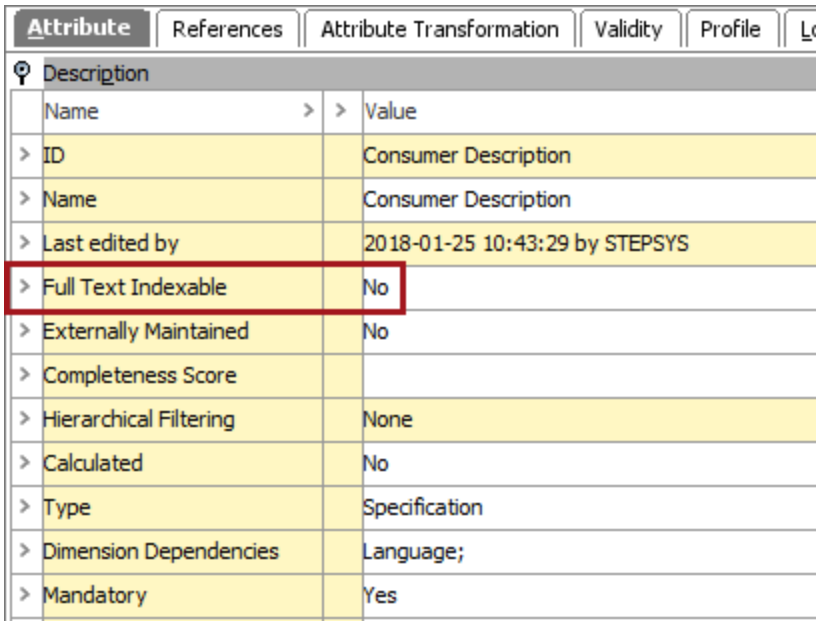
The following parameters can affect performance and apply to attributes:

- Full Text Indexable
- Dimension Dependencies
- Large Lists of Values (LOVs)

Full Text Indexable

In System Setup, the 'Full Text Indexable' parameter can be set on an attribute with a Validation Base Type of 'text' so you can search for words within 'text' values. For example, it allows the user to search for objects based on a word, or set of words, in a sentence within an attribute value. Without the full text indexable option, you can still search for values.

When an attribute is made as Full Text Indexable, performance can be negatively affected. For more information, refer to the **Full Text Indexable Attributes** topic in the **System Setup** documentation.



To review the attributes with this option enabled, export all attributes as STEPXML and in the output file search for **FullTextIndexed="true"**. Determine if the parameter can be disabled.

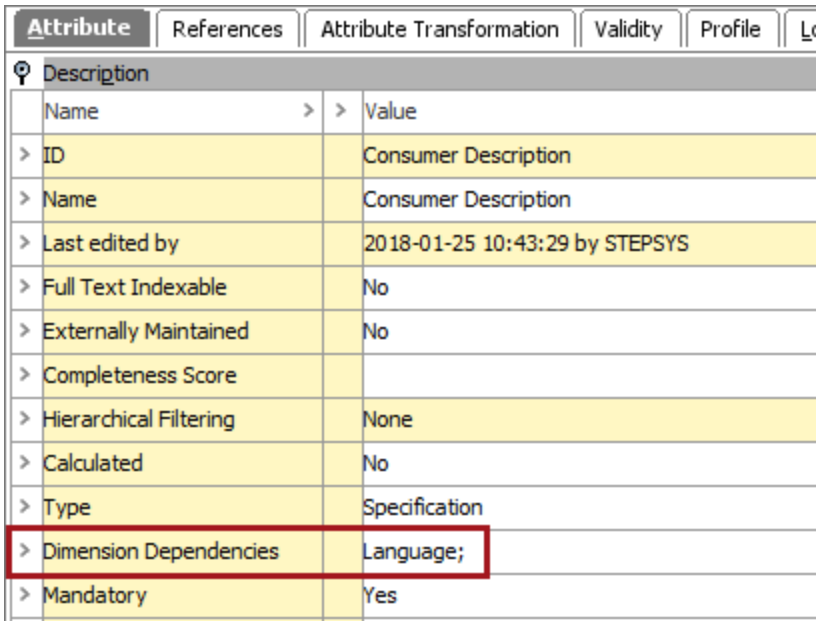
For better performance, set the 'Full Text Indexable' parameter to 'Yes' only if needed.

Dimension Dependencies

Dimension Dependencies add complexity and cause performance issues, and should only be used on attributes when necessary.

To review the attributes with this option enabled, export all attributes as Excel and in the output file filter on **<Attribute Dimension dependencies>**.

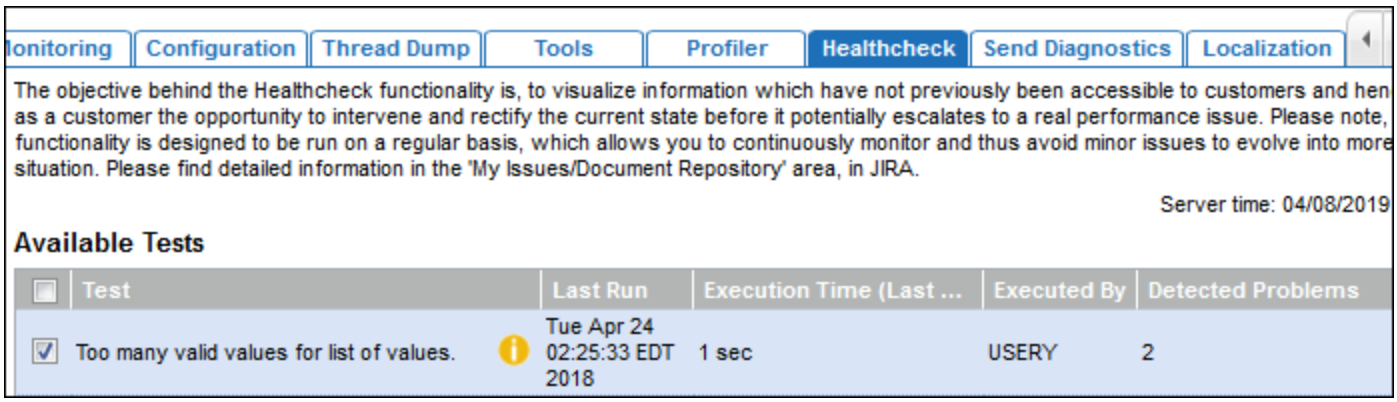
Refer to the **Dimension and Context Recommendations** topic for more information.



Large Lists of Values (LOVs)

Special attention is required for attributes with large List Of Values (LOVs)—those with thousands of values. LOVs with many values can lead to performance degradation. Additionally, from a usability perspective, large LOVs should be avoided.

Use the Healthcheck in the Admin Portal to find LOVs with too many values. For more information, refer to the **Healthcheck** topic in the **Administration Portal** documentation.



For better performance, split large LOVs into smaller LOVs, or use a text field instead of an LOV.

For more information, refer to the **LOV Filtering Recommendations** topic.

Calculated Attributes

Calculated attributes provide a lot of flexibility, but they also have a direct impact on downstream deliveries. Additionally, calculated attributes can degrade performance. For more information, refer to the **Calculated Attribute Recommendations** topic in this documentation.

Calculated Attribute Recommendations

This is one of the data gathering methodologies and recommendations for base setup improvement. The full list is defined in the **Base Setup Recommendations** topic.

Note: These recommendations are valid for systems without In-Memory. The performance costs are different with In-Memory. For more information, refer to the **In-Memory Database Component for STEP** section of the **Resource Materials** in online help.

Calculated attributes are a special type of attributes whose values are not written in the database but are calculated on the fly. Calculated values are dynamic based on the context, version, product selected in the GUI or included in the export. Calculated values are generated when clicking on the object in GUI or when exported and are not stored in the database. The foundation for a calculated attribute is a functional programming language very similar to the language used for defining functions in Excel.

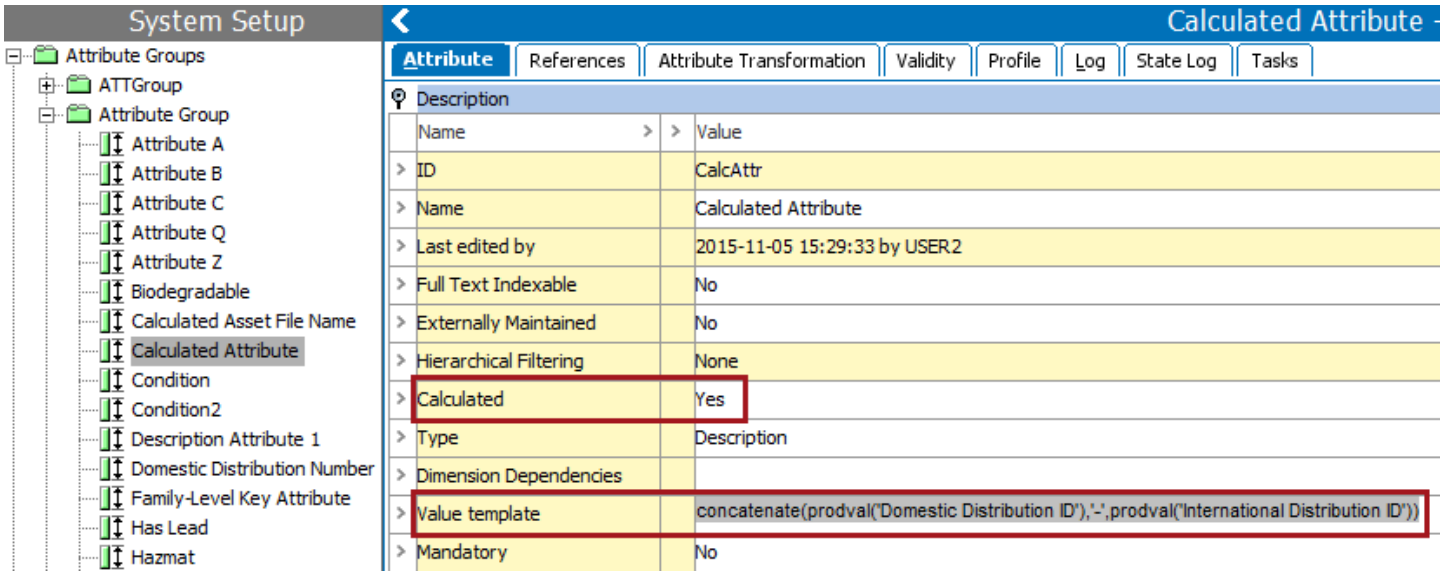
Note: Change Flags for events require that data is present in the database. Since calculated attribute values are not stored, no change flags are generated. For more information, refer to the **Limitations and Exceptions** section of the **Core Events** topic in the **System Setup** documentation.

For assistance in determining if a calculated attribute is the most efficient way to meet your requirements, refer to the **Calculated Attribute Considerations and Limitations** topic in the **Calculated Attributes** section of the **System Setup** documentation.

The value of a calculated attribute is determined by a value template which can be written in the Function Editor. Many functions can include an argument, which specifies the STEP data to be accessed. Functions can access information from references, compare data between objects, round numbers, modify text, and many other options. For a step-by-step guide to creating a calculated attribute, refer to the **Creating a Calculated Attribute** topic in the **Calculated Attributes** section of the **System Setup** documentation.

As an example, the following functions and arguments are included in the workbench image below:

- The 'concatenate' function takes a comma-delimited list of arguments. In this example, the three (3) arguments are: prodval('Domestic Distribution ID'), '-', and prodval('International Distribution ID').
- The 'prodval' function takes an argument of an attribute ID and retrieves the value of the attribute identified. In this example, the two (2) attribute IDs are: 'Domestic Distribution ID' and 'International Distribution ID'.
- Assuming that the 'Domestic Distribution ID' value is 1234 and the 'International Distribution ID' value is 5678, the result of this 'Value template' would be 1234-5678.



For additional use cases for calculated attributes, refer to the **Calculated Attribute Use Cases** topic.

In Web UI, calculated attributes can be identified, and their values can be overridden on the Attribute Management screen as described in the **Calculated Attributes in Web UI** section of the **Web User Interfaces** documentation.

Considerations and Limitations

Generally, business actions are preferred to calculated attributes with the understanding that the value is not always up to date, typically immediately following approval.

While calculated attributes provide a lot of flexibility, they also have a direct impact on downstream deliveries. Additionally, calculated attributes can degrade performance as defined below.

- For cross-context exports, STEP attempts optimizations to avoid repetitive calculations per context, but the following limitations exist:

  Attributes used in the calculated attribute must not be dimension dependent. The cross-context exporter does not impact performance significantly when extracting non-calculated attributes in many contexts compared to a single context. However, the time it takes to extract calculated attribute values grows in a nearly linear way with the number of contexts. STEP cannot automatically detect if the formula that expresses the calculation is dimension dependent or not. This means that introducing many calculated attributes for new contexts to a downstream delivery can significantly degrade performance.

  Iterations in a calculation must not include references or child objects. Only attribute values directly on the selected object or inherited values for a product are allowed. Locality refers to the distance between the values required by a calculation and the calculated attribute. Calculated attributes that access only values on the local object can be very efficient, but calculated attributes that navigate to other objects (via references and/or hierarchy links) are more expensive in terms of performance. Therefore, calculated attributes that navigate to other objects should be carried out in a controlled and limited manner.

Calculated attributes must not rely on the value from another calculated attribute. Chained calculated attributes depend on other calculated attributes and can make performance unpredictable.

- Calculated attributes that cross transitive closures can potentially grow very large cause performance to degrade. For example, a calculated attribute that involves all children of a large (and growing) entity or product hierarchy.
- Calculated attributes are calculated each time they are viewed.
- Complex calculated attributes increase the load time of an object (product or entity) in the Web UI and workbench.

Recommendations

Review calculated attributes by exporting all attributes as Excel or CSV and filter on 'Attribute Calculated'. Add the identified calculated attributes to a collection and export the collection as STEPXML to evaluate the value templates for the following elements.

- Identify the products and entities that take a long time to load. In workbench, from the View menu, click **Disable calculated values** and verify if there is a measurable difference between viewing these objects without the calculated attribute values. Significant differences in load time indicate further analysis is needed as outlined in the following items. Refer to the **View Menu** topic in the **Getting Started** documentation.
- Check the log files for SEVERE errors of 'DAPTreeWalker.' This indicates that a calculated attribute value template uses the function 'stepurl' to call a reference instead of a node. Correct the calculated attribute value template to resolve this error. For more information, refer to the **Logs** topic in the **Administration Portal** documentation.
- If the calculated attribute is included in cross-context exports, consider performing the calculation on approval via a business action, and then copy the result to a non-calculated attribute.
- Avoid calculated attributes that navigate to other objects via references and/or hierarchy links across children. Replace the calculates attribute with a business action.
- Avoid calculated attributes across transitive closures that potentially grow very large.
- Avoid having chained calculated attributes and use business actions for better performance.
- Consider In-Memory to optimize the performance of reading data on complex data models via calculated attributes which navigate references. For more information, refer to the **In-Memory Database Component for STEP** section of the **Resource Materials** section of online help.

LOV Filtering Recommendations

This is one of the recommendations for attribute and attribute groups. The full list is defined in the **Attribute and Attribute Group Recommendations** topic.

Filtering a List of Values (LOV) can eliminate duplicate LOV values, while restricting the values available for selection based on the location of an object in the hierarchy, or based on the attribute using the LOV. However, this functionality can result in poor performance and be difficult to manage.

Avoid using LOV filters to enforce data population as the functionality is intended to support filtered end user selection rather than globally restricted data population. Specifically, when LOV filters are implemented, users are prevented from manually selecting values outside of the applicable filter.

For more information, refer to the **Filtering LOVs** topic in the **System Setup** documentation.

The following sections discuss filters for LOVs that are set on the LOV itself or using the LOV attribute and hierarchical filtering.

Filtering on the LOV

The LOV itself can be filtered via the 'Value Filter' on the List Of Values, as defined in the **Filtering LOV Values** topic in the **System Setup** documentation and shown in the table below. This enables you to define legal values based on the attribute using the LOV.

'Color' LOV Values	Legal Values for Attribute 1	Legal Values for Attribute 2
• Green • Red • Yellow • Blue • White	Filter Includes • Green • Yellow • White	Filter Includes • Green • Red • Blue

Filtering on the LOV attribute and hierarchical filtering

The LOV attribute can be filtered via the Hierarchical Filtering parameter, as defined in the **Filtering by Hierarchy on LOV Attributes** topic in the **System Setup** documentation and shown in the table below. This enables you to define legal values based on product and/or classification hierarchies.

'ColorLOV' Values	Legal Values for the 'Color' attribute on the Plastic Products Node	Legal Values for the 'Color' attribute on the Wood Products Node
• Green • Red • Yellow • Oak • Mahogany	Filter Includes • Green • Red • Yellow	Filter Includes • Oak • Mahogany

Note: Hierarchical LOV filtering operates as an intersection. This means that when a product is associated with multiple classifications, only the common values that are shared across all the filters within these classifications will be considered valid for the given product.

Recommendations

Export all attributes and attribute groups as STEPXML and in the output file search for **HierarchicalFiltering="true"** to review areas for improvement.

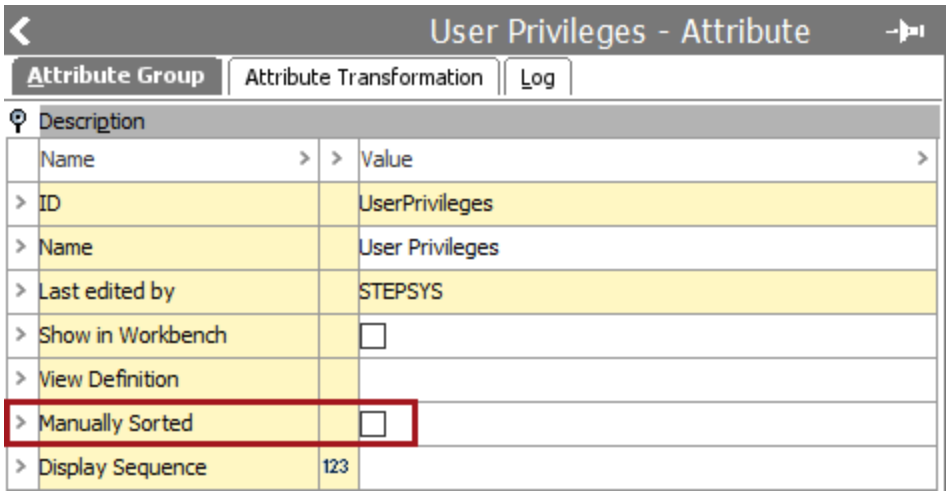
Avoid hierarchical filtering on product and classification hierarchy where possible since hierarchical filtering in both product and classification hierarchies can negatively affect performance.

Avoid using the 'Ignore LOV filters' option on the relevant product-to-classification link types. While it restricts which parts of the classification hierarchies are used for LOV filter inheritance, it can become difficult to manage.

Manually Sorted Recommendations

This is one of the recommendations for attribute and attribute groups. The full list is defined in the **Attribute and Attribute Group Recommendations** topic.

The Manually Sorted option requires additional processing power and can have an impact on performance when using extensively.



Manually Sorted can apply to attribute groups as well as object types.

The limitations are:

- Manual sequence is only set up in the workbench.
- Manual sequence can only be enabled for attribute groups that have no sub-groups.
- Manual sequence is only supported for attributes. Sequencing of 'tag groups,' LOV groups and reference types below an attribute group are not supported.

For more information, refer to the Manually Sorted section of the **Attribute Groups** topic in the **System Setup** documentation.

To review the attribute groups with this option enabled, export all attribute groups as STEPXML and in the output file search for **ManuallySorted="true"**. Determine if the parameter can be disabled.

To review the object types with this option enabled, export all object types as Advanced STEPXML using the following template and in the output file search for **ManuallySorted="true"**. Determine if the parameter can be disabled.

```
<STEP-ProductInformation>
<UserTypes ExportSize="All"/>
<EdgeTypes/>
```

```
<CrossReferenceTypes ExportSize="All"/>
</STEP-ProductInformation>
```

Recommendations

Use the Manually Sorted option only when required to sequence object types or attributes in their attribute group.

Background Process Queue Recommendations

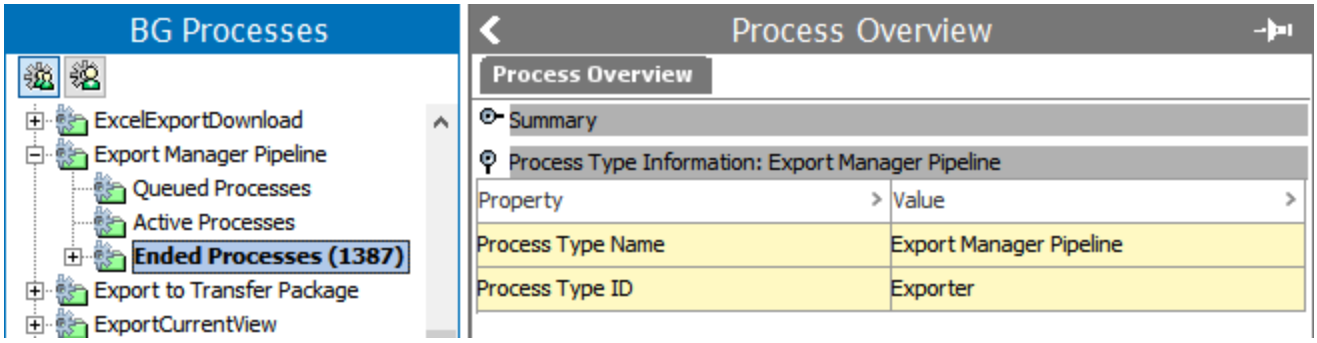
This is one of the data gathering methodologies and recommendations for functional performance improvement. The full list is defined in the **Performance Recommendations** topic.

Configuration properties allow admin users to manage system behavior for a wide range of functionality, for example, the management of background processes and web services.

> **Note:** Ensure changes in the settings are applied with care and properly tested before promoting to production. Parallel setting and multithreading can easily result in optimistic locking issues, which can have a negative impact on performance, rather than the desired performance gain.

Parallel and multithreading optimizations can be applied to background process (BGP) queues used by integration endpoints, event processors, and scheduled processes.

To determine the background process type and the queue handling the process, in workbench, on the BG Processes tab, click the all users button (), select a node, open the Process Type Information flipper, and find the 'Process Type ID' parameter.



Recommendations

Legacy background process (BGP) functionality uses specified queues, while the recommended BGP execution mechanism runs BGPs based on the priority of the BGP and the created time. Refer to the **BG Processes Execution Management** topic in the **System Setup** documentation.

The following legacy recommendations apply to specific types of BGPs.

- Long-running integration endpoints: use the legacy 'Queue for the endpoint process' parameter, the queue size, the 'Transactional settings' parameter, and CPU cores as appropriate to limit optimistic locking issues.
- Event processors: use the legacy queue size and CPU cores as appropriate to limit optimistic locking issues.

- Scheduled processes (types include BulkUpdate, Exporter, etc., and others like AsyncJobRunner): use the queue size, the 'Transactional settings' parameter, queue parallel, and CPU cores as appropriate to limit optimistic and pessimistic locking issues.

Review the topics below for information on running parallel and multithreading processes.

In the **System Setup** documentation:

- **Background Processes and Queues** - for explanations and examples of parallel and multithreading properties
- **Default Configuration for Background Process Queues** - for standard configuration settings
- **Modifying Background Process Queue Configuration** - for instructions on updating queue settings to use parallel and multithreading

In the **System Administration** documentation:

- **Event Processor and Event Queue Recommendations** - for instructions on managing queues and queue size
- **Import Elements to Use** - for information about using separate queues for important IIEPs
- **Export Elements to Use** - for information about multithreading and using separate queues for important IIEPs

In the **Data Exchange** documentation:

- **Integration Endpoint Transactional Settings** - for information about managing the data processing order

Base Setup Recommendations

This is one of the data gathering methodologies and recommendations for functional performance improvement. The full list is defined in the **Performance Recommendations** topic.

The base setup of the contexts and dimensions, product and entity data model, assets and classifications are based on business requirements. Nevertheless, the base setup should also be designed based on performance since a complex base setup can have a negative impact on general performance.

An analysis of the base setup is worthwhile, as changes on the base setup for performance reasons usually have a great impact.

For example, changing the data model may impact attributes and references, but can also impact business rules, workflows, Web UI configurations, exports, and imports.

Oracle KODO Database Layer

The Oracle KODO layer is the database layer which allows for Java Persistence API (JPA) specification and Java Data Objects (JDO). By default, the Oracle KODO layer has a cache of 10,000 data relations on an object. An excess of 10,000 relations overwhelms the cache and negatively impacts performance. Examples include a hierarchy with more than 10,000 children, a product with more than 10,000 references, a list of values with more than 10,000 values, and an outbound integration endpoint with a batch size of more than 10,000 events.

Storing large data relations can have a significant negative performance impact due to the likelihood of having a large number of the related objects that are no longer in the cache.

If your system performance is significantly impacted due to this limitation, contact Stibo Systems Support to request a configuration setting to increase the data cache size in the sharedconfig.properties file.

Important: Always weigh the pros and cons of changing the base setup prior to making the change, and test the effects on a lower system before making change on a production system.

For detailed information, refer to the following topics:

- Asset Recommendations
- Classification Recommendations
- Data Model Recommendations
- Dimension and Context Recommendations
- Global Count of Object and Attribute Recommendations
- Reference Recommendations

Asset Recommendations

This is one of the data gathering methodologies and recommendations for base setup improvement. The full list is defined in the **Base Setup Recommendations** topic.

> **Note:** These recommendations are valid for systems without In-Memory. The performance costs are different with In-Memory. For more information, refer to the **In-Memory Database Component for STEP** section of the **Resource Materials** in online help.

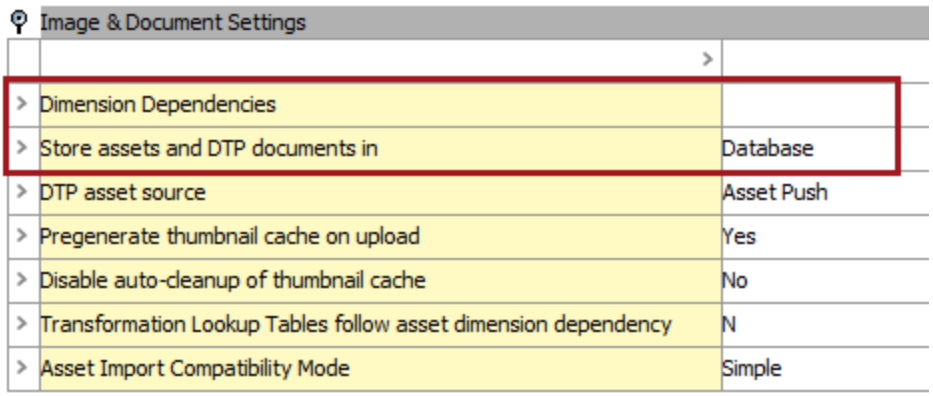
Digital assets are media files like images, videos, documents, etc. In STEP, these are asset objects which can hold attributes (metadata) and have a reference to the actual asset (binary file).

Storage Location

The asset binaries can be stored outside STEP in an external DAM system. The storage of the asset binary can also be stored on the file system or in the STEP database. While storing the asset binaries in the database might seemingly allow faster access, accessing many asset binaries in the database also results in higher database traffic. Therefore, asset binary storage in the file system or in the STEP database does not determine the performance of STEP.

Dimension Dependency

Dimension dependency on assets should be avoided when it is not required. In workbench, on System Setup, asset dimension dependency is defined in the Users & Groups node under the Image & Document Settings.



The Dimension Dependencies option is used if there is a requirement to store images or documents in different contexts. For example, you may have a requirement store an image that has English labels in the image and the same image with French labels, as found in the examples below.

To set up, click in the Dimension Dependencies value row, then click the ellipsis button (...). Select the Language option. Now the image in English can be stored in the English context and the image in French can be stored in the French context.

Note: In the image examples used above, both must have the same image ID when loading into STEP and the appropriate dimension point must be selected for each image.

Import and Export

STEP allows for asset transformations during import and export. Asset transformations require system resources and therefore should be avoided unless necessary. For more information, refer to the inbound and outbound sections of the **Digital Assets** documentation.

Asset import optimizations include:

- Using the standard asset importer, as defined in the **Asset Importer** topic of the **Digital Assets** documentation.
- Import assets into a folder hierarchy instead of using a single folder, including saving uncategorized assets into subfolders. The standard asset importer can automatically create an asset folder structure.

Classification Recommendations

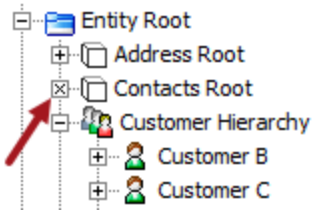
This is one of the data gathering methodologies and recommendations for base setup improvement. The full list is defined in the **Base Setup Recommendations** topic.

> **Note:** These recommendations are valid for systems without In-Memory. The performance costs are different with In-Memory. For more information, refer to the **In-Memory Database Component for STEP** section of the **Resource Materials** in online help.

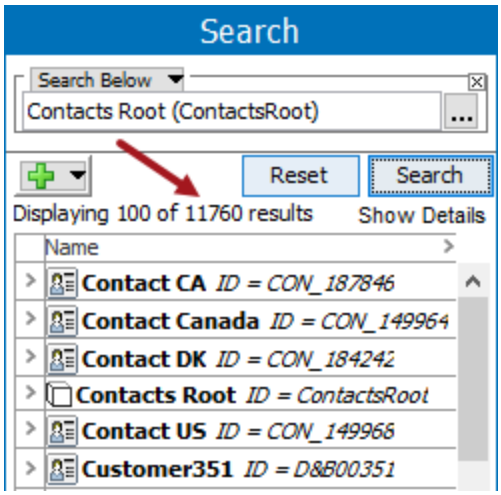
STEP allows for setting up one primary hierarchy (blue folders) and multiple classifications (yellow folders). However, many children in a level of a hierarchy is expensive from a performance perspective. The built-in STEP unique name check means that having many items in one node of the hierarchy can degrade performance.

A classification or hierarchy with more than 10,000 children should be avoided. For details and an alternative, refer to the **Oracle KODO Database Layer** section of the **Base Setup Recommendations** topic.

In Tree, after attempting to expand a hierarchy with 'too many' children, the node is identified with ⊠ (shown in the image below). Nodes that cannot be expanded are automatically set to non-readable and cannot be browsed. 'Too many' for workbench is determined by the case-sensitive property **Workbench.TreeNode.MaxNumberOfChildrenToShowInTree** and for Web UI, the case-sensitive property **Portal.TreeNode.MaxNumberOfChildrenToShowInTree**, both are displayed by accessing the STEP System Administration button on the Start page. For more information, refer to the **Configuration** topic in the **Administration Portal** documentation.



For nodes that cannot be expanded, Search Below can be used to analyze how many children are in the hierarchy. For example, searching below the 'Contacts Root' hierarchy shows it has 11,760 children.



Recommendations

Use the following recommendations to optimize performance:

- Avoid classification hierarchies with 10,000 children by organizing and categorizing children in sub hierarchies which allow you to manage millions of child nodes.
- Instead of a functional classification, use an alphanumeric classification naming structure to accelerate the listing of many objects by creating subcategories at import or by applying auto-classification.
- Identify long-running business rules via the Activity Dashboards tab as defined in the **Admin Portal Business Rule Activity Dashboard** section of the **Business Rule Analysis** topic. In workbench, analyze the business rule to determine if the nodes being modified are excessively large.

Data Model Recommendations

This is one of the data gathering methodologies and recommendations for base setup improvement. The full list is defined in the **Base Setup Recommendations** topic.

Note: These recommendations are valid for systems without In-Memory. The performance costs are different with In-Memory. For more information, refer to the **In-Memory Database Component for STEP** section of the **Resource Materials** in online help.

The data model of STEP is very flexible and allows for complex data model setups. While the data model is designed and set up based on business requirements, the data model should also avoid unnecessary complexity and ensure optimal performance. A complex data model can have a negative impact on the general performance.

Important: For performance reasons, do not over-design the data model since a complex data model requires more extensive processing. Also, use the 'Manually Sorted' parameter on product objects and entities sparingly. For more information, refer to the **Attribute Groups** topic in the **System Setup** documentation.

The following example data models highlight the strengths of various approaches for achieving the business requirements.

Data Containers

Consider a required 'Company' entity that has a visiting and distribution address with country, city, street, and zip code. The addresses can be defined using one of the following methods:

- Attributes directly on the 'Company' entity, such as visiting address street, visiting address city, distribution address zip code, etc.
- A separate 'Address' entity for each address type (e.g., 'visiting address' and 'distribution address'), street, city, zip code, etc., and reference the 'Company' to the 'Address.'
- Two data containers 'Visiting Address' and 'Distribution Address' with country, city, street, zip code, etc.

The preferred data model to use is likely the data containers, since:

- Reusing the same address over multiple companies is minimal. Therefore, the data model with the 'Address' as a separate entity is unnecessarily complex.
- Some companies might not have a 'Distribution Address,' making the usage of separate attributes on 'Company' somewhat inflexible.

Product References

Consider that a product belongs to one product family in most cases, but in some exceptional cases might also belong to another product family. The products can be defined using one of the following methods:

- The product in the blue hierarchy belongs to a parent product family in the blue hierarchy. For the exceptional cases (where the product also belongs to another product family), a product reference is used where the product is also related to the other product family.
- The product-override is used where the product-overrides are alternate versions of products and product families that may have differing values, references, links, and structures. Attributes and values applied to the product family are inherited to the product-override and can be replaced with local values and references on the product-override.

The preferred data model to use is likely the product reference, when inheritance of the second product family is not required, since it is less complex.

Dimension and Context Recommendations

This is one of the data gathering methodologies and recommendations for base setup improvement. The full list is defined in the **Base Setup Recommendations** topic.

Note: These recommendations are valid for systems without In-Memory. The performance costs are different with In-Memory. For more information, refer to the **In-Memory Database Component for STEP** section of the **Resource Materials** in online help.

The setup of contexts and dimensions is flexible and is often designed and configured based on business requirements. Ideally, the setup of contexts and dimensions should be based on avoiding unnecessary complexity and optimal performance as well, since a complex setup can have a negative impact on general performance.

Usually, in multilingual setups, two dimension points are defined by:

- Country
- Language

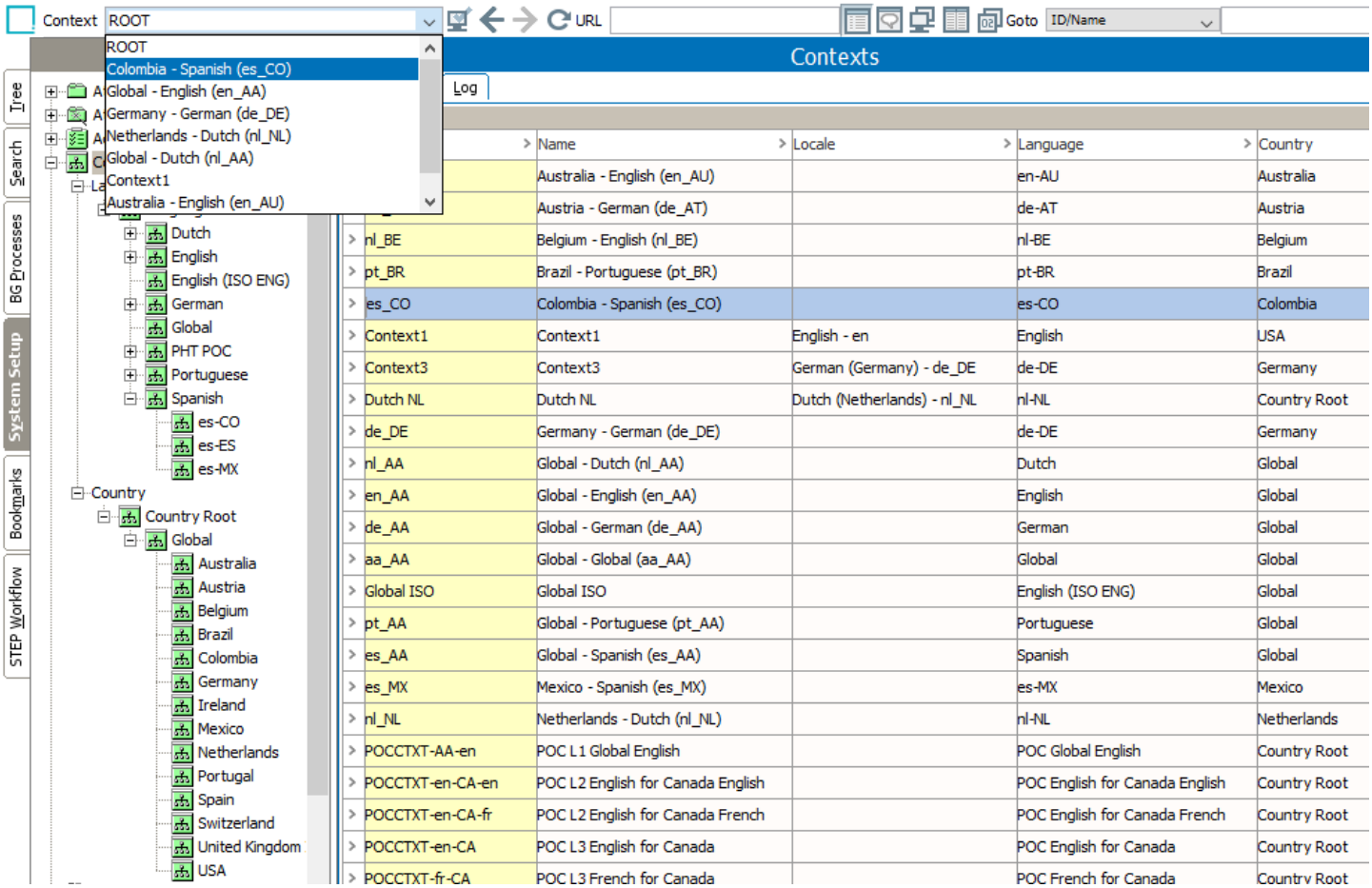
For this example, every object (both entities and products) can have a country and a language dimension:

- The country dimension allows differences in countries, such as different accessories of the product in different countries via suppressing a reference.
- The language dimension allows translations in different languages where the same object has an English, German, Spanish, etc., translation of the data.

The context defines the combination between the defined dimension points.

For example, the context **Colombia - Spanish (es_CO)** is defined with:

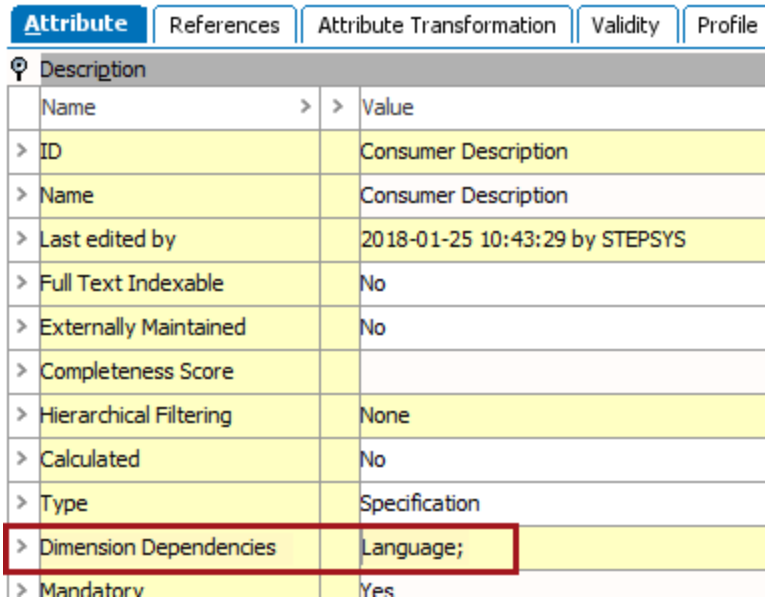
- Country = Colombia for Colombia (es-CO)
- Language = Spanish for Columbia (es-CO)



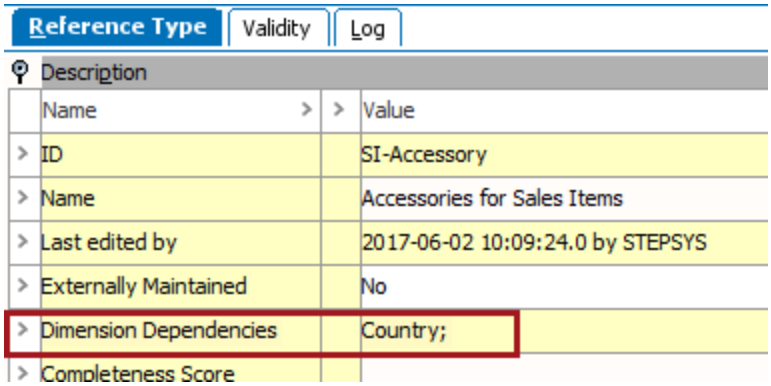
Important: The dimensions allow setup in a hierarchy using inheritance. This means that the dimension point inherits the data from its parent unless it has its own data. For example, the language 'Spanish for Colombia (es-CO)' is setup below the language 'Spanish (es).' This means that all data in the language 'Spanish for Colombia (es-CO)' is inherited from the language 'Spanish (es)' until local data in the language 'Spanish for Colombia (es-CO)' is set.

Once the dimensions and contexts are configured, then attributes, references, etc., can be configured to be dimension dependent.

For example, an attribute 'Consumer Description' can be configured to be language dependent.



And a reference such as 'Accessory' can be configured to be country dependent.



Recommendations

The following scenarios should be considered when determining the necessary dimension points for the system:

- Setting more than one dimension on an object can result in a complex situation and should be avoided. For instance, if a metadata attribute for an asset is defined to be language and country dependent, then it is possible to have a translated metadata attribute on the asset in different languages, and have a different metadata attribute value for different countries. It is hard to predict values when making an attribute multi-dimensional, such as when the metadata attribute inherits the values from its parents (e.g., from language Spanish and from country Global).
- Only define two dimensions (e.g., language and country) and only configure dimension dependency when required. For example, if the objects do not differ much in the countries, then only define a language dimension.

The more dimensions defined, the more complex the solution will be. A complex dimension setup requires more extensive processing in imports, exports, bulk updates, running business rules, etc.

The number of dimension points (e.g., number of languages) is of less importance, as well as the number of contexts.

For more information, refer to the **Dimensions, Dimension Points, and Contexts** topic in the **System Setup** documentation.

Global Count of Object and Attribute Recommendations

This is one of the data gathering methodologies and recommendations for base setup improvement. The full list is defined in the **Base Setup Recommendations** topic.

Note: These recommendations are valid for systems without In-Memory. The performance costs are different with In-Memory. For more information, refer to the **In-Memory Database Component for STEP** section of the **Resource Materials** in online help.

From the Start page, click the About STEP link, and log in to display counts of a variety of objects.

- System name: **dreloc**
- STEP version: **10.2-mp1-2021-07-08-14-09-06**
- Detailed version information
- Number of user accounts (including suppliers): **63**
- Number of supplier accounts: **7**
- Allowed number of user accounts: **100**
- Number of products (approx.): **900**
- Number of classifications (approx.): **8000**
- Number of assets (approx.): **500**
- Number of entities (approx.): **7000**
- Number of contexts: **12**
- Number of dimensions: **2**
- Is InMemory Enabled: **false**
- Number of languages: **10**

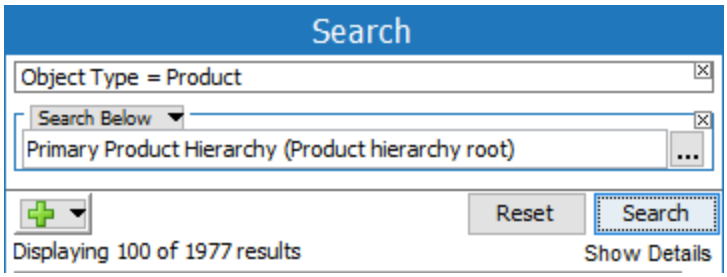
Additionally, the following are examples of using search to count the number of objects in STEP. The search displays the total number of results found.

An example is included for the following types of counts:

- Products
- Products by object type
- Entities
- Entities by object type
- Assets
- Attributes

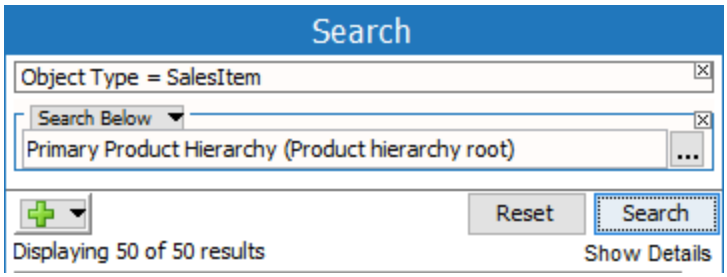
Count the number of products

Search criteria: 'Object Type = Product' and 'Search Below = Primary Product Hierarchy'



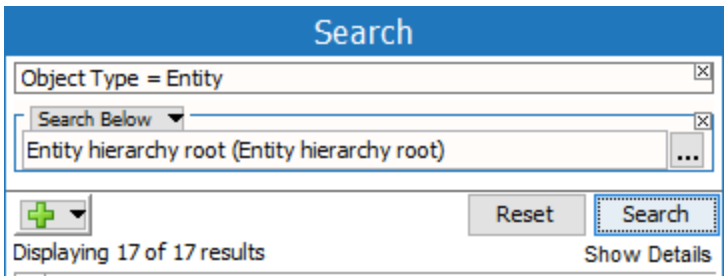
Count the number of products by object type

Search criteria: 'Object Type = [*Product Object Type*]' and 'Search Below = Primary Product Hierarchy'



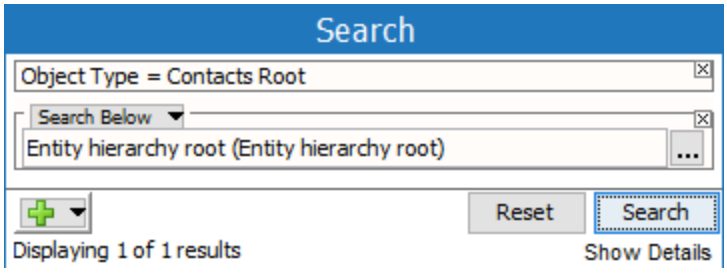
Count the number entities

Search criteria: 'Object Type = Entity' and 'Search Below = Entity Hierarchy Root'



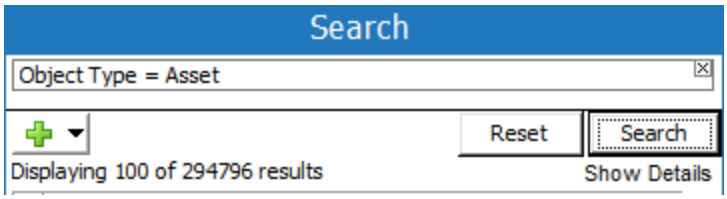
Count the number entities by object type

Search criteria: 'Object Type = [*Entity Object Type*]' and 'Search Below = Entity Hierarchy Root'



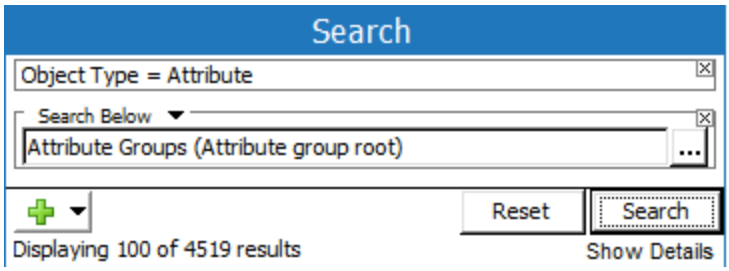
Count the number of assets

Search criteria: 'Object Type = Asset'



Count the number of attributes

Search criteria: 'Object Type = Attribute' and 'Search Below = Attribute Groups'



Reference Recommendations

This is one of the data gathering methodologies and recommendations for base setup improvement. The full list is defined in the **Base Setup Recommendations** topic.

> **Note:** These recommendations are valid for systems without In-Memory. The performance costs are different with In-Memory. For more information, refer to the **In-Memory Database Component for STEP** section of the **Resource Materials** in online help.

A few of the flexible elements of references include:

- Ability to configure with multiple different sources and targets (reference and reference by)
- Are of a certain type (product, asset, classification, etc.)
- Can have metadata attributes

None of these reference configurations independently have a direct impact on performance.

However, objects with many references can have a negative impact on performance since the object needs to be re-approved every time a reference changes. Also, since the approval of an object checks each reference, when the object has thousands of references, approval can become costly in performance. For details, refer to the **Oracle KODO Database Layer** section of the **Base Setup Recommendations** topic.

Inheritance also has an impact on performance because the hierarchy is navigated to identify valid values.

Additionally, navigation across a reference to another object, to fetch attribute values and/or references, is expensive on performance. This is because another database access, and potentially another disk access, is required. For example, navigation via references in business rules to resolve values, resolving calculated attributes, or resolving references during export.

Recommendations

When encountering either of the cases above:

- Verify that references are owned by the object type that requires the fewest number of approvals. An object that owns many references carries the load of the approval process, including verifying approval for all referenced objects.
- Limit the usage of references in business rules, calculated attributes and during export.

Business Rule Recommendations

This is one of the data gathering methodologies and recommendations for functional performance improvement. The full list is defined in the **Performance Recommendations** topic.

Business rules allow extension of the STEP functionality with customer-specific business logic. Business rules provide a flexible way to tailor the core functionality in a very precise manner and can be tied to, e.g., bulk updates, events, imports and IIEPs, approval processes, workflows, Web UI screens, etc.

Important: The system load of the business rule execution can have effect on the performance of STEP. Running multiple complex business rules degrades performance while STEP is processing the business rules.

Database lock issues are common when very long business rules perform changes while running in parallel. To avoid locking issues on a busy, multi-user system, a business rule must run inside a single transaction and the transactions must be short to allow for several commits during the overall processing. For more information, refer to the **Avoid Large Transactions** section in the **Business Rule Elements to Avoid** topic.

The available types of business rules, and the purposes for which each is optimized, are defined in the **Business Rules** topic in the **Business Rules** documentation.

Details about where business rules can be used are defined in the **Using Business Rules in STEP** topic in the **Business Rules** documentation.

Optimizing performance in business rules involves the following:

- Business Rule Analysis
- Business Rule Elements to Use
- Business Rule Elements to Avoid

Extending STEP Functionality

If the business rule recommended practices have been implemented, and additional performance improvement is required, consider the following additional ways to extend STEP functionality.

Scripting API	Allows customers and partners to expand STEP functionality with JavaScript Business Rules. Involves a low level of customization and extended functionality.
Extension API	Allows customers and partners to extend STEP functionality with

	development of plug-ins and components. Involves a moderate level of customization and extended functionality.
Custom Extensions	Allows Stibo Systems to extend STEP functionality with development of custom extensions. Involves a high level of customization and extended functionality.

Using a Java extension can sometimes improve performance over a JavaScript business rule.

For example, this provides the ability to develop a custom extension instead of executing many complex business rules on import of data.

Note: Only cases where extreme amounts of logic are executed should be considered for this solution. In such cases, consider why it is necessary to run such complex logic.

The possible benefits of using a custom extension include:

- Event batching is used by the processor to ensure that business logic is executed exactly once per product per batch, even if multiple imports were executed for a single product.
- Optionally, the example plugin uses the parallel processing framework to maximize performance. This can potentially allow a 'strict' transaction endpoint to still benefit from parallel processing in that rule execution can be multi-threaded while still ensuring data integrity.
- Resolve optimistic locks and deadlocks in cases where the locks are caused not by the import itself, but by logic that accesses and writes to objects shared among parallel import processes.

  A common example of this occurs with tree structures. If parallel imports are executing on children of a common parent, each of which executes business logic causing an update to the parent, the result is deadlocks and sever performance degradation.

  Another performance-related issue with this pattern is that the business logic is executed once per child. However, if the import logic is changed to republish the parent to the event queue instead of each child, event batching will result in a single update to the parent object, regardless of the number of children imported.

- The same concept can be applied using an OIEP configured with a business rule pre-processor. Execute import business logic in the pre-processor, then discard the event so that no exports are produced by the endpoint. Since endpoints do not use multithreading while executing the pre-processor, the execution time may be longer than if a custom extension is used.

This solution also has some consequences:

- There will be a delay between importing data onto the object, and the business rule running. If timing is an issue, consider other options.
- The object cannot be inspected in both its previous and current form as easily, the business rules running in such a processor will have less information available than the same rule running during import.

Business Rule Analysis

This is one of the recommendations for performing analysis on business rule to improve performance. The full list is defined in the **Business Rule Analysis** topic.

Business rules are units of business logic that are stored as objects in System Setup. Business rules are used for many different purposes in STEP and come in three variants:

	Input	Output	Side effects allowed
Business actions	Current object, current event batch, etc. provided by the context in which the action is executed. For more on business actions, refer to the **Business Actions** topic.	None	Yes
Business conditions	Current object, current event, etc. provided by the context in which the condition is evaluated. For more on business conditions, refer to the **Business Conditions** topic.	Boolean result of evaluating the condition and a message for the user	No
Business functions	Input parameters defined by the function and provided by the functionality evaluating the function. For more on business functions, refer to the **Business Functions** topic.	Result of evaluating the function	No

A fourth type of business rule, **business library**, allows users to define JavaScript library functions that can be called from other JavaScript-based business rules. For more information on business libraries, refer to **Business Libraries** topic documentation.

For more information on differentiating between scenarios where one business rule is more useful than another, refer to the **Business Rule Use Cases** topic in the **Business Rules** documentation.

Using Business Rules in STEP

Wherever business rules are used, they are always tested or executed in relation to one object at a time, and in a specific context / workspace. The most common places to use business rules are included in the list below.

- **Approvals** - Conditions can be tested when approval is attempted on an object under revision control, and the condition can allow or prevent the approval. Actions executed on approval and can modify data in STEP (typically data on the object being approved), send emails etc. related to the approval.

  For more information, refer to **Business Rules on Approval** documentation.

- **Automatic Classifications** - Actions can be executed to automatically classify objects, and can be applied, for example, on approval, during an import, or as a part of a workflow.

  For more information, refer to **Using Automatic Classification with Business Actions** documentation.

- **Bulk Updates** - Conditions can be tested as a precondition for executing an action. Actions can be executed.

  For more information, refer to **Run Business Rule Operation** documentation.

- **Conditional Attributes** - The JavaScript business action 'Conditionally Invalid Values' bind resolves to a set of all values.

  For more information, refer to **Business Rules with Conditional Attributes** documentation.

- **Data Profiles** - Conditions can be tested against all objects in a category (for example, a part of the Product hierarchy), and the result of the tests can be displayed on the Profile Dashboard.

  For more information, refer to **Business Conditions in Data Profiling** documentation.

- **Event Processors** - Actions can determine when and how to act upon the events from the event processor. This is useful when changes occur on objects and custom actions need to occur.

  For more information, refer to **Execute Business Action Processing Plugin Parameters and Triggers** documentation or **Execute Business Action for Event Batch Processing Plugin Parameters and Triggers** documentation.

- **Gateway Integration Endpoints** - Accessed from JavaScript in business rule conditions and actions, the bind can work with a variety of the REST methods.

  For more information, refer to the **Gateway Integration Endpoint Bind** documentation in the **Resource Materials** online help. *Gateway Integration Endpoints (GIEPs) do not always use the REST plugin; the Gateway Integration Endpoint Bind topic only applies for GIEPs that use the REST plugin.*

- **Imports and Inbound Integration Endpoints** - Conditions can be tested during imports, and the condition can allow or prevent the creation or update of objects. Actions executed during import can modify the objects being imported, apply actions to objects being imported, send emails, and start workflows, etc.

  For more information, refer to **Business Rules in an Import Configuration** documentation.

- **Matching, Linking, and Merging** - Actions and Conditions can be used to normalize the data for comparison to identify the duplicate products in STEP. Actions can also be used in relation to Golden Records Survivorship Rules.

  For more information, refer to Matching section of the **JavaScript Binds** topic in the online help **Resource Materials** documentation or the **Golden Records Survivorship Rules** topic in the **Matching, Linking, and Merging** documentation.

- **Outbound Integration Endpoint** - Conditions can be used as event filters for an event-based OIEP. Actions can be executed via a pre-processor for any OIEP, or as a event generator to generate derived events to export or publish for an event-based OIEP.

  For more information, refer to **OIEP - Event-Based - Event Triggering Definitions Tab** documentation or **OIEP - Pre-processor - Business Action** documentation.

- **Web UI** - Actions can be executed to update the object. Conditions can be evaluated to improve data validity.

  For more information, refer to **Business Rules in Web UI** documentation.

- **Workflows** - Conditions can be tested within workflows to allow or prevent transitions from one state to another. Actions can be executed when entering a state, when leaving a state, when performing a specific transition, and when a deadline is met. Actions can also modify the object being tracked by the workflow, modify other objects in STEP, modify the workflow behavior, send email, start other workflows, etc.

  For more information, refer to **Business Rules in Workflows** documentation.

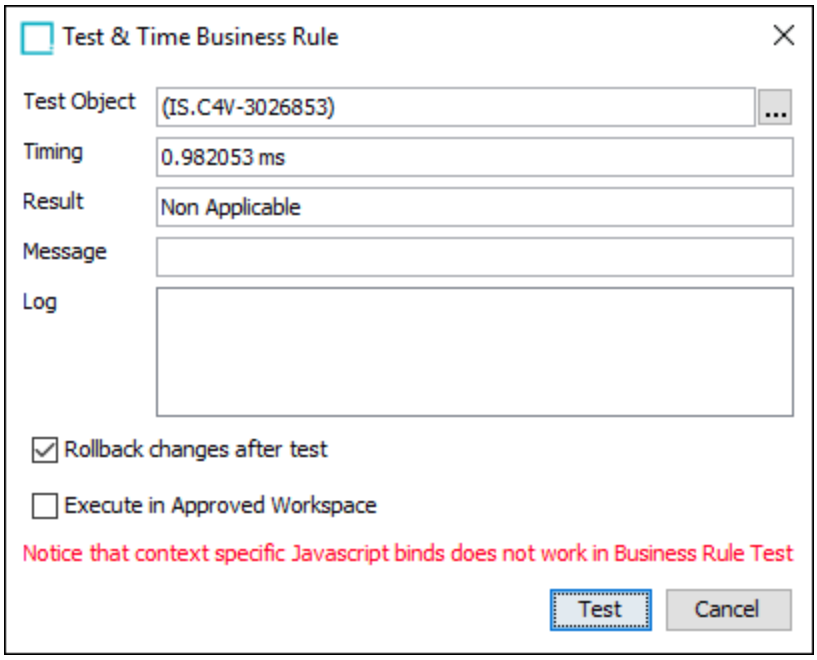
Test & Time Business Rule

Use the 'Test & Time Business Rule' dialog, as defined in the **Testing a Business Rule** topic of the **Business Rules** documentation. This option gives a first indication of the performance of the business rule for a certain item.

Test a business rule multiple times against objects that will either fail or pass and analyze the performance timing.

Test a long-running business rule to verify the performance timing.

For example, the business rule shown below took about 0.98 milliseconds to complete for the selected object. The same business rule may take more or less time for other objects.

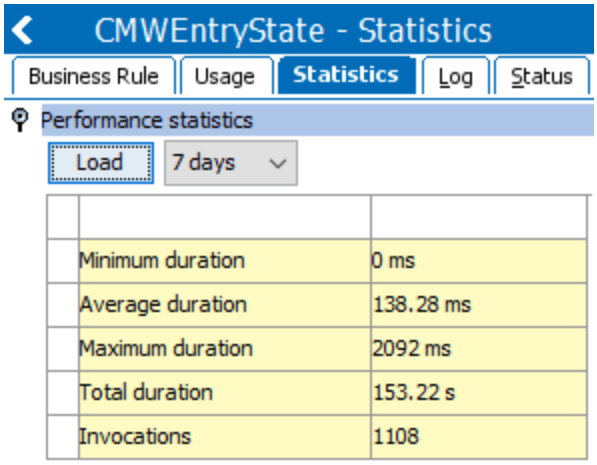


Business Rules Statistics

Use the business rule 'Statistics' tab, as defined in the **Maintaining a Global Business Rule** topic of the **Business Rules** documentation. This tab shows the performance of the business rule over time.

The minimum, maximum, average, and total duration of the business rule, as well as the number of invocations per selected period are displayed. The period can be configured to be between an hour up to a week.

For example, the image below shows the same business rule was invoked more than 100 times during the last 7 days. That average duration was about 138 ms. Click on the maximum duration of about 2092 milliseconds to show which item the business rule took longest to execute.

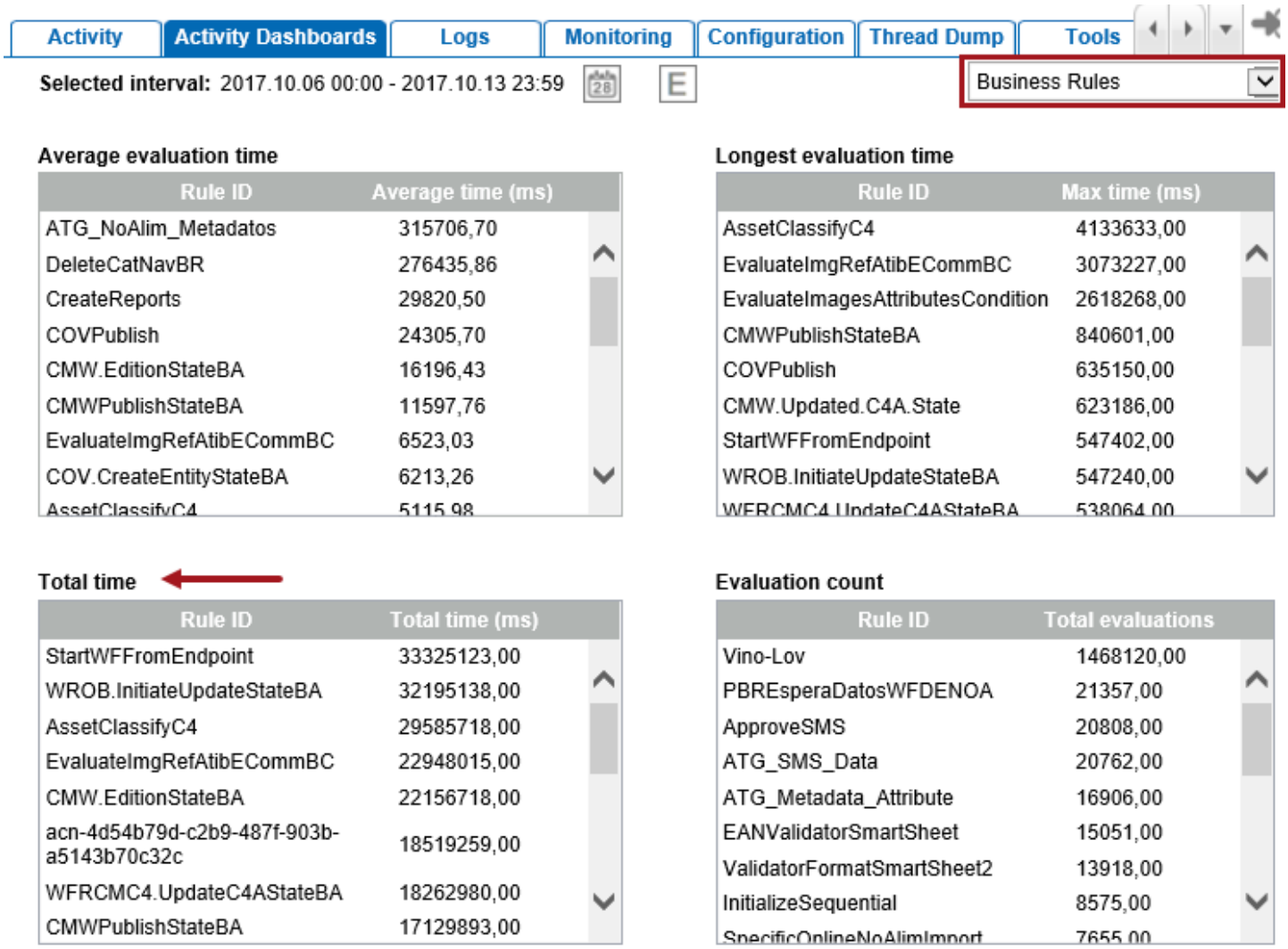


Admin Portal Business Rule Activity Dashboard

Use the Admin Portal 'Activity Dashboards' tab to display business rules information, as defined in the **Activity Dashboards** topic of the **Administration Portal** documentation. This option allows you to track and trace the most demanding business rules performance over a given period.

The dashboard shows the top business rules over the configured period, with the longest average evaluation time, the longest maximum evaluation time, the longest total time, and the number of invocations.

Start with the 'Total time' section to review the business rules with the longest average evaluation time and the most number of invocations.



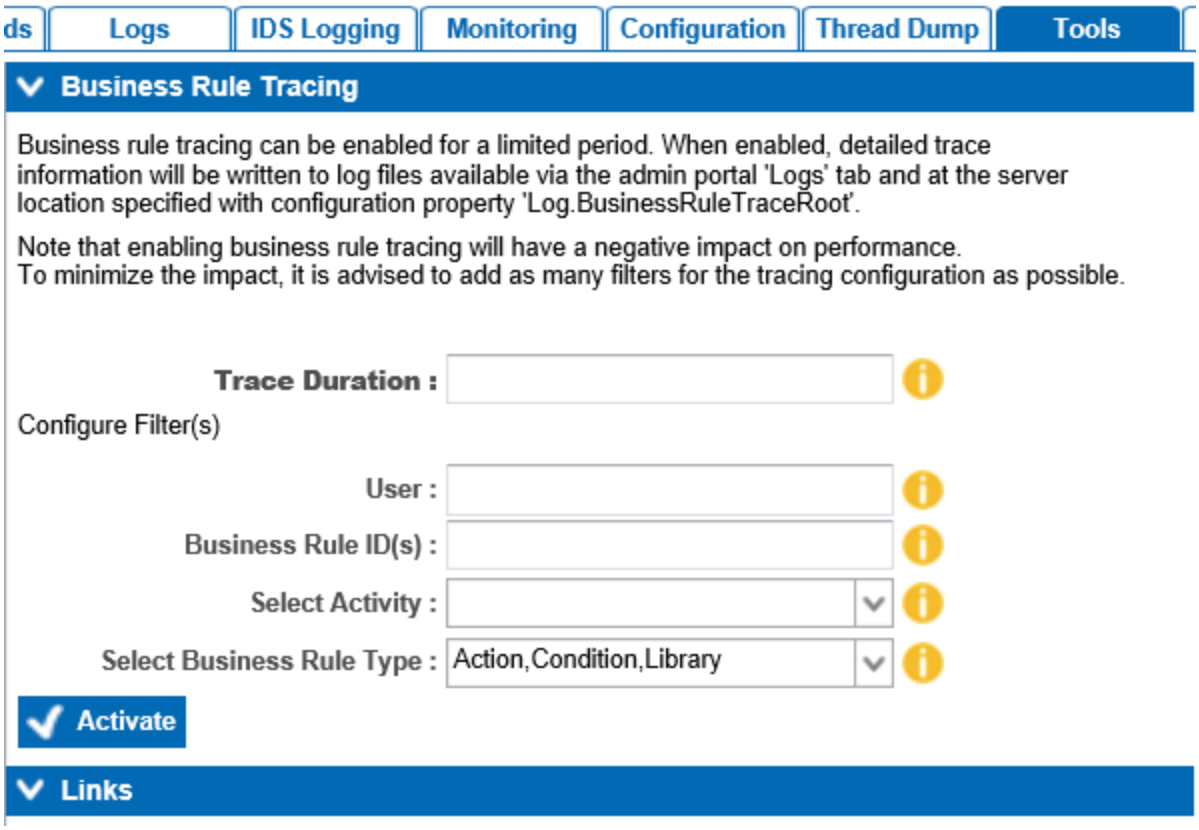
Admin Portal Business Rule Tracing

Use the Admin Portal 'Tools' tab to run the Business Rule Tracing functionality. When enabled, business rule tracing writes to log files.

Important: Enabling business rule tracing has a negative impact on performance. To minimize the impact, leave it enabled for a limited time and add as many tracing configuration filters as possible.

Define the location of the tracing log files by setting the case-sensitive property **Log.BusinessRuleTraceRoot** in the sharedconfig.properties file.

To enable business rule tracing, click the yellow information icon next to each parameter for a description of the parameter / filter. Supply the relevant information and click the **Activate** button. Then within the 'Trace Duration' time frame, from workbench, trigger the business rule(s) being traced.



Tracing stops when the system is stopped or restarted, and when the Trace Duration expires. Once a trace is activated a stop button (Stop) is displayed, which allows a user to manually stop tracing.

Business Rule Elements to Use

This is one of the recommendations for performing analysis on business rule to improve performance. The full list is defined in the **Business Rule Analysis** topic.

The following list includes business rule elements known to assist system performance. This list can be used to troubleshoot existing long-running business rules, and can also be reviewed prior to writing new business rules to prevent performance problems.

Use Exception Handling

If an error occurs during approval, an exception is thrown from the domain layer. If this exception is caught in a business rule but not re-thrown, it will not reach the exception approval handler. In this case, objects can be inconsistently approved (some parts are approved and other parts are not). This behavior also has a negative effect on the performance of the business rule.

When writing JavaScript business rules, it is important that 'try…catch' statements are designed correctly. 'Try...catch' statements should not swallow (catch and ignore) exceptions that should cause changes made by the script to be rolled back and cause the business rule to fail.

Additionally, only catching an exception without rethrowing it can lead to inconsistent data or the error: 'The transaction cannot be committed, because it was already marked for rollback only. The transaction will be rolled back instead.'

Important: Carelessly catching and ignoring exceptions will lead to derived errors, which usually makes it difficult to determine the root cause.

The sample code in the table below demonstrates a way to log exceptions in JavaScript without swallowing those that should be handled by the framework.

Correct	Incorrect

`try {`
`  // Some code`
`} catch (e) {`
`  logger.info(e);`
`  throw(e); // REQUIRED`
`}`
	`try {`
	`  // Some code`
	`} catch(e) {`
	`  logger.info(e);`
	
	`}`

The only types of JavaScript exceptions that may be caught and handled locally are:

- Checked exceptions thrown by an invoked method (including any subclasses of the specified checked exception) (refer to the example below)
- Exceptions not generated by calls to the STEP API

Important: Any other exception must always be re-thrown if caught, or not caught at all.

Correct Handling of Checked Exceptions

The code snippet below demonstrates how a checked exception can be caught and handled locally, including the necessary Rhino specific '**e.javaException**' notation.

```
    try {
    currentObject.createReference(targetAsset, primaryImageRefType.getID());
    } catch (e) {
    if (e.javaException instanceof com.stibo.core.domain.UniqueConstraintException) {
    logger.info("Reference could not be created");
    } else {
    throw(e); // All other exceptions MUST be re-thrown
    }
    }
```

Proper exception handling correctly re-throws the exception when using 'try-catch' in business rules and avoids inconsistent objects.

For more information, refer to the **Recommended Error Handling Practices** topic of the **Resource Materials** section of online help.

Use Logging Carefully

Warnings and errors encountered while executing business rules can be written in the Main STEP Log File. While logging many details can have a negative impact on performance, using the correct logging level can aid in troubleshooting and resolving unexpected outcomes.

Logging can be managed for all business rules or on a per-business rule basis.

Manage Logging for all Business Rules

Define the appropriate logging level for each STEP server environment using the following case-sensitive entries in the sharedconfig.properties file:

- **BusinessRule.Warning.Threshold** - a threshold in milliseconds for business action execution. If it takes longer to execute or test a given business action, a warning is posted in the Main STEP Log File.

For example, the following setting writes a warning to the STEP log file whenever a business rule execution takes longer than 10 seconds.

```
BusinessRule.Warning.Threshold=10000
```

- **Log.Level** - specifies the level of detail for logging business rules results, the entry shown below would include log entries for info, warning, and severe:

```
Log.Level = INFO
```

- **Log.Level.org.mozilla.javascript.MemberBox** - specify the logging level for business rule results, as shown below:

```
Log.Level.org.mozilla.javascript.MemberBox = SEVERE
```

The logging values are ALL, FINEST, FINER, FINE, CONFIG, INFO, WARNING, SEVERE, and OFF. The most granular logging provided using ALL, while minimal logging is written using SEVERE. Setting a level also includes logging for the levels to the right. INFO is the default setting.

Suggested settings are FINE to trace errors on a development or test system, INFO or WARNING for a QA system, and SEVERE on a production system.

For more details about the log levels, refer to the **Server Log File Settings** topic in this guide.

Manage Logging per Business Rule

Log the result of business rules during development on the development server, but turn off the logging when deploying to the test, quality, and production servers. An easy and transparent way to control logging per business rule is to set a debug flag in the business rule code, as illustrated below.

```
//Debug 'flag' DO NOT use unless you develop or test
//When doing tests DO NOT test on large amount of products
//REMEMBER to set to 'false' when development and testing phases are complete
var isDebug = false;

//Function to handle whatever logging of debug information should occur or not
function logDebug(message) {
if(isDebug) {logger.info(message)}
}
...
logDebug("A message for the log file")
...
```

Use Arrays, Not Multiple Read Calls

Business rules repeatedly using calls to the database for large sets of data significantly degrades performance. Instead, use one call to get the data, and push it into arrays and work from there. Minimizing the number of calls to the database aids performance.

When multiple business rules are executed sequentially (e.g., as part of an approval process), and these business rules fetch the same data from the database multiple times, it is beneficial to rewrite the business rules to fetch the data once, and push the data into (multi-dimensional) arrays.

In-Memory

In-Memory can improve performance of the business rules because it provides faster operations on complex data models where business rules navigate references.

In-Memory may improve performance on business rules that still perform poorly after implementing the recommended practices for business rules. For more information, refer to the **In-Memory Database Component for STEP** topic of the **Resource Materials** section of online help.

Consider Using an Extension

An extension should be considered when additional performance improvement on business logic in the system is required and all previous recommendations on business rules are implemented (including In-Memory). Some of the code using in JavaScript (business rules) might run faster in Java (extensions).

Important: Only cases where extreme amounts of logic are executed should be considered for this solution. And in such cases, first consider why it is necessary to run such complex logic.

For example, you could develop a custom extension instead of executing many complex business rules on import of data. The possible benefits of such a solution are:

- Event batching is used by the processor to ensure that business logic is executed exactly once per product per batch, even if multiple or many imports are executed for a single product.
- The example plugin optionally uses the parallel processing framework to maximize performance. Potentially, this allows a 'strict' transaction endpoint to benefit from parallelization in that rule execution can be multi-threaded while still ensuring data integrity.
- Optimistic locks and deadlocks can be resolved in cases where the locks are caused not by the import itself, but by logic that accesses and writes to objects shared among parallel import processes.

  A common example of this occurs with tree structures. If parallel imports are executing on children of a common parent, each of which executes business logic causing an update to the parent, deadlocks (and sever performance degradation) will result.

  Another performance-related issue with this pattern is that the business logic is executed once per child. However, if the import logic is changed to republish the parent to the event queue instead of each child, event batching will result is a single update to the parent object, regardless of the number of children imported.

- For STEP 7.4+ solutions, the same concept can be applied using an outbound integration endpoint configured with a business rule pre-processor. First, execute import business logic in the pre-processor, then discard the event so that no exports are produced by the endpoint.

Note: Endpoints do not use multithreading while executing the pre-processor so the execution time may be longer than if a custom extension is used.

However, a pre-processor solution also has some consequences:

- There will be a delay between importing data onto the object and the business rule running. If timing is an issue, consider another option.
- Running business rules in a pre-processor does not allow the object to be inspected in both its previous and current form as easily as running the same rule running during import.

Business Rule Elements to Avoid

This is one of the recommendations for performing analysis on business rule to improve performance. The full list is defined in the **Business Rule Analysis** topic.

The following list includes business rule elements known to degrade system performance. This list can be used to troubleshoot existing long-running business rules, and can also be reviewed prior to writing new business rules to prevent performance problems.

Avoid Large Transactions

Business actions involve a transaction which allows you to manipulate data in STEP. Business actions with long-running transactions degrade the performance. Additionally, since STEP runs with optimistic locking policy, large transactions increase the probability of optimistic locking failures when running the business action simultaneously. For more information, refer to the **Optimistic and Pessimistic Locking Recommendations** topic.

Keeping business rule transactions small means it is necessary to develop business rules while considering the worst-case scenario. This includes the following recommendations:

- Avoid traversing a substantial percentage of the complete data range.
- Keep changes to data local (to the nodes nearest your main data object).
- Avoid business rules that follow and potentially change the objects of transitive closures of referential structures, such as everything underneath a high-level folder in the product hierarchy containing thousands of children. Instead, use a bulk update or a customer specific background process.

If it seems that the business rule transactions cannot follow these recommendations, refer to the **'Reference Target Lock Policy' Parameter** section of the **Optimistic and Pessimistic Locking Recommendations** topic.

Avoid Large Libraries

On each execution, JavaScript libraries are compiled. Each dependency is stacked into a script which is also compiled using the ScriptEngine.eval() method before each execution of the business rule.

For example, Script S depends on Library A and Library B. Library A depends on B. In this case, the script is stacked as follows:

- A::script
- A::B::script
- B::script
- S::bindings
- S::script

If a business rule depends on a library, the library is compiled as well. By extension, if a business rule uses only a single function within a library, the whole library is still compiled.

And if a library depends on another library, the other library needs to be compiled as well.

STEP caches the scripts instead of reloading them from the database. By default, 100 business rules are cached. Generally, it takes about 500 milliseconds to compile about 8,500 lines of code at each business rule execution.

Important: Be aware that libraries are compiled every time the business rule is executed, which is especially burdensome to performance when libraries depend on each other, and are used in many business rules. Dividing a large library into multiple libraries, but keeping the dependencies, does not resolve the issue.

To improve performance, consider making the library functions local to the business rule.

Avoid Many Inclusions of a Library

Avoid smaller libraries with functions used multiple times by the business rule. For example, a business rule that is applied to 10,000 objects can take up to five times as long to complete compared to running the same function locally in the business rule, instead of from a library.

Evaluate poor performing business rules to determine if one of these scenarios applies:

- If one or more function(s) are called from a **large** library (thousands of lines), make the library functions local to the business rule itself or use a business function and analyze the level of performance improvement.
- If one or more function(s) are called from a **small** library but the function is used many times (thousands of executions), make the library functions local to the business rule itself or move the business rule itself to the library or use a business function and analyze the performance improvement.

Avoid Infinite Loops

Infinite loops lead to severe degradation of system performance on the affected application server(s). Ultimately, an infinite loop can make the entire STEP installation unresponsive.

Analyze business rules to determine if infinite loops exist.

Avoid the 'getChildren' Function with Many Nodes

Business rules that use 'getChildren' on a large number of children (more than 10,000) can cause memory problems because the function reads all the children. Instead, use 'queryChildren' function, which limits the number of children affected.

Analyze business rules to determine if 'getChildren' is used on a selection with more than 10,000 children, and update the 'queryChildren' function as needed.

Avoid Updating Data via Business Conditions

Business conditions are optimized for determining the true / false result of read-only scenarios. For details, refer to the **Business Conditions** topic in the **Business Rules** documentation.

Event Processor and Event Queue Recommendations

This is one of the data gathering methodologies and recommendations for functional performance improvement. The full list is defined in the **Performance Recommendations** topic.

Event processors and queues are used for asynchronous processing, such as auto-purging revisions on schedule or performing a delayed approval after import. Warnings and/or critical issues in asynchronous processing can have an influence on the performance of the system.

For general information on event processors and event queues, refer to the **Event Processors** section and **Event Queues** section, both in the **System Setup** documentation.

For more information, refer to the **Monitoring** topic of the **Administration Portal** documentation.

Run Without Warnings and Errors

Warnings and errors are visible in the event processor background processes. For optimal processing, no warnings or errors should be reported.

For example, the following image shows the business rule execution report for an event processor. In this illustration, the event processor initiated a business rule with a setting 'Valid Object Types = All object types valid' but the event processor is not valid for all object types. Business rules initiated via an event processor should be defined with specific object type(s) rather than setting all object types as valid.

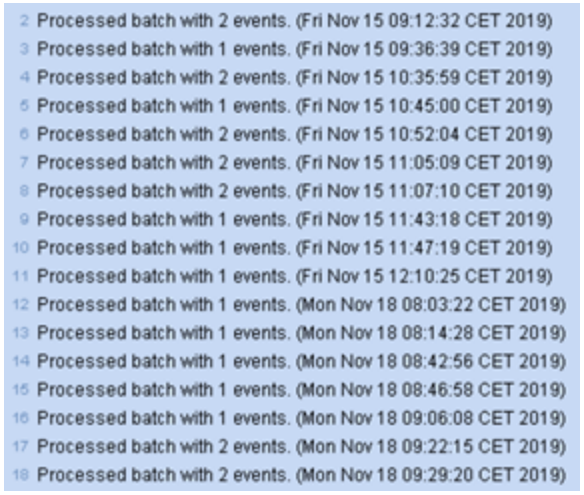
Optimize STEP Setup for Performance

Use the following setup recommendations within the event processor configuration to optimize performance. For detail on all parameters, refer to the **Event Processors** topic in the **System Setup** documentation.

- The **User Running Event Processor Plugin** parameter defines the privileges that are checked when the event processor is invoked. For each piece of information processed, all privileges are checked for the selected user. Creating a user with relatively few and broad permissions explicitly for event processors can improve performance dramatically.
- The **Number of Events to Batch** parameter defines the number of events processed in a batch. While the events are handled one-by-one, committing large batches are more effective and improve performance. For the event processors:

- Set the 'Number of events to batch' parameter no larger than the max size defined in the sharedconfig.properties file for the OutboundMessageProcessor.BatchEventsMaxSize property. By default, this is 10,000. Lower this number if you experience system memory problems.
- Analyze the schedules and the background processes to determine how many events are processed in each batch.

For example, in the following image, the number of events in a batch is 1 or 2, so the number of events to batch could be set to five (5). However, if the schedule is increased, e.g., once per 15 minutes instead of once every minute, then the number of events in a batch could be an average of 30. In that case, the 'Number of events to batch' parameter could be set to 30.



- The **Days to Retain Events** parameter is the number of days to keep events once processed. However, the value should be set to 0 since this parameter has no impact on event processors.
- The **Schedule** parameter can be set to run every minute, but this will have a negative impact on performance. The 'Start every minute' option should be used only in rare and necessary situations.

Analyze Asynchronous Processing

Analyze the state of asynchronous processing using the 'Sensors for external monitoring' option as follows.

1. On the Start page, click the **STEP System Administration** button, and supply the login credentials.
2. On the **Monitoring** tab, open the Additional Links section and click the 'Sensors for external monitoring' link to display the list of monitors.
3. For each sensor with a 'Warning' or 'Critical' status, click the sensor name link to display the details of the status.
4. Resolve the issue indicated in the sensor log.

For example, the following sensor has a 'Critical' status.

Sensor	Status	Message
EventQueueSensor-DynamoDBEvents	**Critical**	1649000 events have been queued, which is more than the scheduled limit of100000

The details show that event queue 'DynamoDBEvents' has too many events queued.

Sensor status for EventQueueSensor-DynamoDBEvents

Plugin	EventQueueSensor
Sensor	DynamoDBEvents
Status	**Critical**
Created	Wed Oct 31 12:49:15 UTC 2018 (0 seconds ago)
TTL	30 seconds
Short message	1810000 events have been queued, which is more than the scheduled limit of100000

Performance data

Name	Value	Unit	Warning	Critical	Min	Max
Estimated number of unread events	1810000.0					

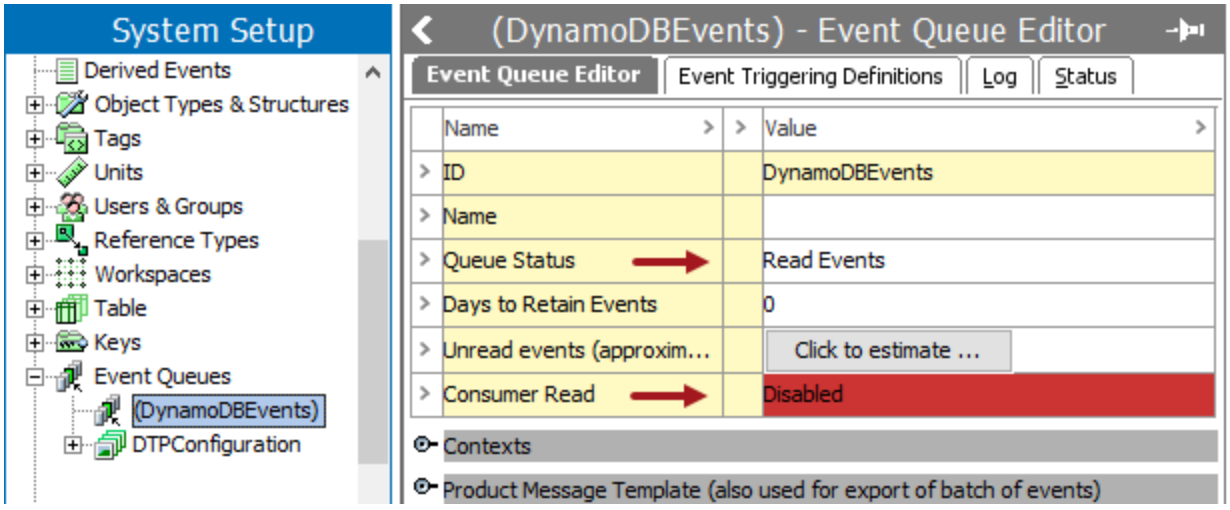
Formats

The status shown on this page is also available in the following machine-friendly formats:

- A simple status string, Possible values: OK, WARNING, CRITICAL, UNKNOWN.
- Nagios plugin output, output formatted for easy integration with Nagios.
- Full xml all available data in xml for easy parsing by ad-hoc monitoring tools.

Please do not rely on the output of this page for automated monitoring, use one of the formats above.

In the workbench, this event queue shows that the event queue is disabled but is still reading events. Since the latest change on this event queue is months ago, and it is on the production environment, the event queue is no longer being used and can be removed.



Legacy - Queues and Queue Size

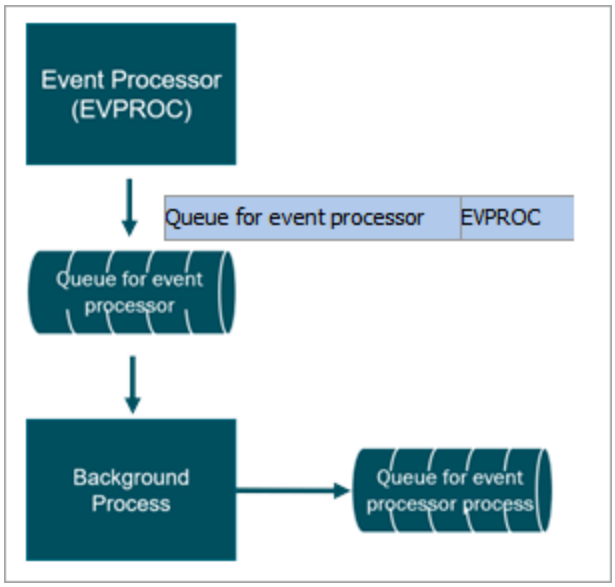
Legacy background process (BGP) functionality uses specified queues, while the recommended BGP execution mechanism runs BGPs based on the priority of the BGP and the created time. Refer to the **BG Processes Execution Management** topic in the **System Setup** documentation.

The event processor initiates a background process to handle the processing.

- The legacy **Queue for event processor** parameter stores the queue that is used by the background process to poll the event processor. The default value is **EVPROC**.

- The queue for associated processes is used by the background processes to handle the actual processing. The queue is identified on the BG Processes tab: select the Event Processor BGP node to display the Process Overview tab and open the 'Process Type Information' flipper. The default value is **EventProcBGP**. For details on identifying and editing the queue being used, refer to the **Modifying Background Process Queue Configuration** topic in the **System Setup** documentation.

STEP legacy functionality allows you to define a separate queue for event processors to ensure only events for this process are read.



The first time you activate the event processor, a queue with the specified name is created if it does not already exist.

A background process (BGP) queue allows prioritizing system processes to ensure high- level system performance. BGP queues are named and configured to control which processes run on which server and how many can run at the same time on each server.

Note: Parallel processing is set by increasing the queue size for the event processor background processes and can improve performance, but can also cause optimistic locking issues when parallel processes attempt to view / update the same node at the same moment. A queue size of 1 means only one process can run at a time, while a queue size of 2 means two processes can run concurrently.

Recommendations

Configure the simplified BGP execution mechanism to run BGPs based on the priority of the BGP and the created time. Refer to the **BG Processes Execution Management** topic in the **System Setup** documentation.

In the event processor settings, legacy functionality uses separate queues for event processors, especially for those with long-running events.

Test parallel background processing using an increased the queue size in a test environment before increasing it in production. Increase the queue size to 2 for the event processor background process via the following case-sensitive properties in the sharedconfig.properties file:

- BackgroundProcess.ProcessType.[*process type ID*].Queue=[*queue*]

  For example:

  ```
  BackgroundProcess.ProcessType.EventProcBGP.Queue=EventProcBGPQ
  ```

- BackgroundProcess.Queue.[*name*].Size=[*number of allowed concurrent processes*]

  For example:

  ```
  BackgroundProcess.Queue.EventProcBGPQ.Size=2
  ```

- Restart the application server to apply the changes to the properties file.

Refer to the **Background Processes and Queues** topic in the **System Setup** documentation for examples of parallel and multithreading properties.

Consider In-Memory for Event Processors

When the other event processor recommendations have been followed and additional performance improvement is needed, consider In-Memory since it provides faster read operations.

For more information, refer to the **In-Memory Database Component for STEP** topic in the **Resource Materials** section of online help.

Export Recommendations

This is one of the data gathering methodologies and recommendations for functional performance improvement. The full list is defined in the **Performance Recommendations** topic.

One of the primary goals of most STEP implementations is to reduce the time to market required by the flow of data from STEP to external systems such as ERP systems and e-commerce systems.

Important: Optimizing the export performance may adversely affect the performance of the system as a whole. For example, multithreading can increase export performance, although it may have a negative impact on the overall system performance.

The available tools for exporting data are defined in the **Data Exchange** documentation.

Optimizing performance in exports involves the following:

- Export Elements to Use
- Export Elements to Limit

Export Elements to Use

The following list includes export elements known to minimize impact on system performance. This list can be used to troubleshoot existing exports, and can also be reviewed prior to creating new exports to prevent performance problems.

To ensure maximum export performance, the usual strategy is to limit the amount of information exported, or limit the number of times the same node is exported with the same data.

For general information on exports, refer to the **Data Exchange** documentation.

Optimize Object Type Triggering Definitions

Ensure that only required objects are exported using the Triggering Object Types flipper on the outbound integration endpoint (OIEP) Triggering Definitions tab. For example, when exporting to an e-commerce system, only the product itself is potentially relevant. Triggering on other object types risks spending time exporting objects which would be irrelevant downstream.

For more information, refer to the **OIEP - Event-Based - Event Triggering Definitions Tab** topic of the **Data Exchange** documentation.

Optimize Attribute and Reference Triggering Definitions

Applying triggering definitions can ensure that objects are only exported when necessary (based on the attributes changed) and minimize performance issues. On the outbound integration endpoint (OIEP) Event Triggering Definitions tab, set the object types in the Triggering Object Types flipper. For example, an e-commerce system is probably only interested when certain attributes, like the description, of a product object changes. So the Triggering Object Type is set to 'product' and the Triggering Attributes is set to 'description.'

Additionally, set the OIEP triggering definitions specifically for the export recipient. Create attribute groups specifically for export channels, and then trigger attributes specified by attribute group. For example, the attribute 'WebsiteAttribute' contains only the attribute relevant for the website export channel.

For more information, refer to the **OIEP - Event-Based - Event Triggering Definitions Tab** topic of the **Data Exchange** documentation.

Optimize Event Filter and Event Generator Triggering Definitions

Event filters and event generators are business rules that are executed during export.

For example, consider that inheritance affects changes to attributes residing higher up the hierarchy, and can result in a significant number of products if all child products are exported. In this case, replacing a single high-level attribute change event with multiple child-level events is recommended. Using the event filter / event generator options means the significant volume of events can be handled in multiple batches, which enables output files to be kept to a reasonable size. It also allows exporting multiple batches in parallel reducing overall export times.

Typically, the 'approval' mechanism should not determine if data is suitable for publishing since approval is much earlier within the product lifecycle. Often, it is desirable to limit the data exported to just 'published' objects. The event filter and pre-processor functionality can meet both of these requirements, although they operate in slightly different ways. Alternatively, all data can be exported downstream (related to the underlying events) and a middleware-based solution can determine which data to route to which target system.

For more information, refer to the **OIEP - Event-Based - Event Triggering Definitions Tab** topic of the **Data Exchange** documentation.

Use Multiple Dedicated OIEPs

In many cases, a single OIEP is used to monitor all product changes, i.e., both attribute and reference changes.

In this case, it is easy to understand how the assignment of many products to a classification, e.g., 'Christmas Gifts' can cause a significant volume of data to be exported, delivered, and processed by the receiving system, even though no actual product data has changed.

This issue can be avoided through the use of multiple OIEPs:

- One OIEP listens for traditional product (attribute) changes and outputs all data for the product.
- Another OIEP listens for reference changes and outputs minimal data, e.g., product ID and target ID.

Note: Generally speaking, the performance gain can be degraded if using more than 10 OIEPs.

For more information, refer to the **Creating an Event-Based Outbound Integration Endpoint** topic of the **Data Exchange** documentation.

Use Multithreading

Increasing the number of threads for a given OIEP can increase export performance for event-based OIEPs.

Multithreading is effective when a large amount of data goes to a downstream system on a regular basis and the downstream system can handle the load.

Consider the following points before increasing the number of threads to more than one:

• STEP system hardware requires enough resources to perform multithreading.

• The downstream (receiving) system must be able to handle parallel events.

Since the batch-fetching of the events runs serially, increasing the thread number results in each batch size being divided by the thread number so that the contents of a batch can be processed in parallel. The data is distributed as evenly as possible to each thread. From fetching the batch through the delivery stage, the pre-processing, main processing, and post-processing take place in parallel. When all threads are complete, the batch yields a single message as is consistent with the 'Strict' transactional setting required by event-based OIEPs. This allows events to be delivered as if everything was executed serially.

Thread size is typically increased for critical information where the speed with which the message is produced is key. However, multithreading can have a negative impact on the overall system performance when a large amount of data is involved.

The recommendation is to consider using multithreading to improve production of messages to a downstream system on a regular basis to improve export performance. Run the settings in a test environment, starting with a small amount of data and then increasing it, before implementing it in the production system.

For more information, refer to the **Event-Based OIEP Multithreading Support** topic of the **Data Exchange** documentation.

Note: Setting the 'Maximum Number of Threads' parameter on the Configuration tab has no impact on a Select Object OIEP (which uses a static set of data, not events).

Optimize the Batch Size

The appropriate batch size is typically based on the size of messages and downstream system processing capabilities:

- Use a smaller batch size for larger messages.
- Use a larger batch size for smaller messages.

For individual OIEPs, the batch size is set using the 'Number of events to batch' parameter on the Event Queue Configuration flipper. For more information, refer to the **OIEP - Event-Based - Event Queue Configuration Flipper** topic of the **Data Exchange** documentation.

Note: Use the system-wide case-sensitive **OutboundExportService.BatchEventsMaxSize** property, set in the sharedconfig.properties file, to limit the number of events included within any batch. When the batch size in the workbench is greater than this setting, this setting overrides the workbench to limit data exported in a single export invocation.

Small Batch Size Considerations

A small batch-size, or no batching at all, invokes the export engine and surrounding framework many times (once per batch), incurring a significant overhead. A small batch-size can also cause the same product to be exported multiple times in cases where a product has been approved multiple times in quick succession, or there have been changes to externally maintained data (in this case one event will be created for each external change).

Note: While setting the batch size = 1 may appear to deliver the required results, in practice, there are problems associated with this approach as defined above. Generally, small batch sizes should be avoided.

To publish one STEPXML file per item (product or entity) while using a batch size larger than 1, use the 'Generic XML Splitter' or the 'STEPXML Splitter' post-processor. For more information, refer to the **Configure the Pre-processor and Post-processor** section of the **OIEP - Select Objects - Output Templates Flipper** topic or the **OIEP - Event-Based - Output Templates Flipper** topic in the **Data Exchange** documentation.

Large Batch Size Considerations

A large batch-size (e.g., 1,000 - 10,000 events) invokes the export engine and associated framework fewer times incurring less overhead and reduces the amount of duplicate product data exported.

The database Oracle KODO cache default limit is 10,000. For example, a hierarchy with more than 10,000 children, a product with more than 10,000 references, a list of value with more than 10,000 values, an OIEP with a batch size of more than 10,000 events, etc. Storing very large data relations can have great negative performance impacts, since there is a high likelihood that a large number of the related objects are no longer in the cache.

Recommendations

- Avoid a batch-size of 1 and if required, use a post-processor to publish separate XML files per item.
- Avoid a batch-size of more than 10,000. If there is a reason for a larger batch size, contact Stibo Systems Support to discuss raising the **Install.DataCache.MaxRelationSize** setting to allow for a larger cache size.

Use Cross-Context Exports

One way to limit how much data is being exported, is to export cross-context STEPXML on an endpoint rather than having multiple endpoints, each exporting in their own context. This is because having multiple endpoints also means exporting the non-context sensitive data multiple times.

Do not configure multiple OIEPs each outputting product details for a single context. Instead, for significantly faster overall export time, use a single OIEP configured with multiple contexts. Subsequently, the context splitter post-processor can be used if separate context output files are required.

For more information, refer to the 'Contexts' parameter section on the **OIEP - Configuration Flipper** topic of the **Data Exchange** documentation.

Use Event-Based Exports Over Static Exports

While static exports can be used, it is usually much more efficient to use event messaging via event-based exports.

In reality, for a larger STEP system with several million products, an event-based integration is the only viable approach. Not only from a STEP perspective (the time taken from the STEP side to export all data), but also from the receiving system perspective (the time to import all data).

It is recommended to use event-based exports over selection-based exports whenever possible.

For more information, refer to the **Outbound Integration Endpoints** topic in the **Data Exchange** documentation.

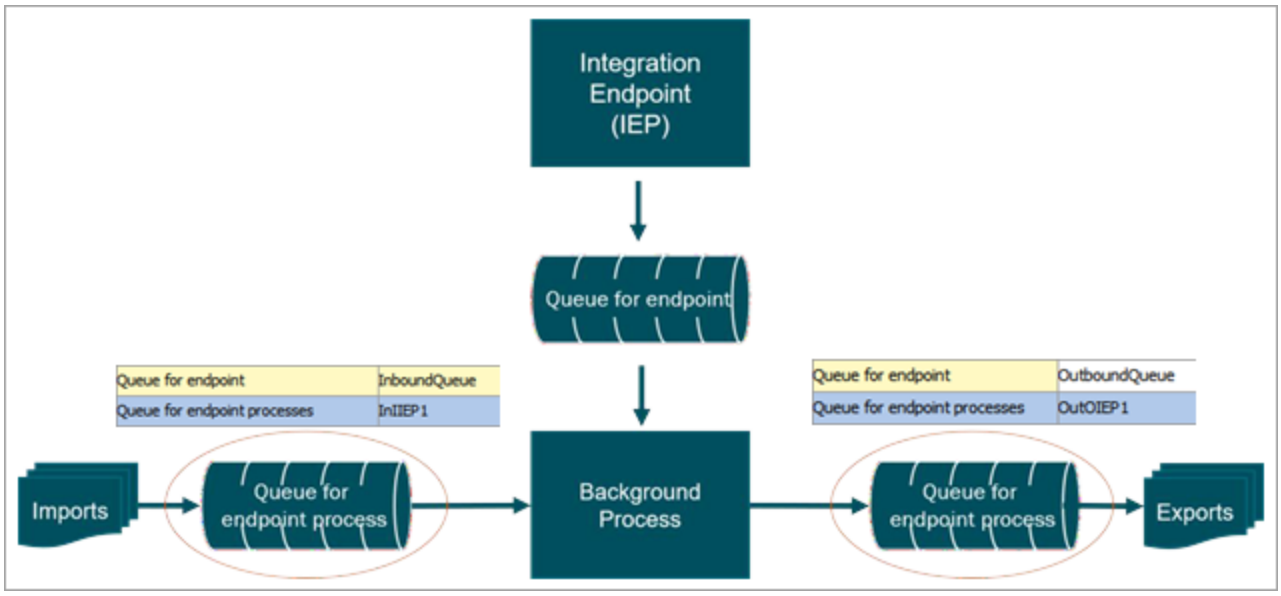
Legacy - Use Separate Queues for Important OIEPs

Legacy background process (BGP) functionality uses specified queues, while the recommended BGP execution mechanism runs BGPs based on the priority of the BGP and the created time. Refer to the **BG Processes Execution Management** topic in the **System Setup** documentation.

The legacy outbound integration endpoint (OIEP) initiates a background process which handles the actual export.

- The legacy **Queue for endpoint** parameter stores the queue that is used by the background process to poll the outbound integration endpoint. The default value is **OutboundQueue**.
- The legacy **Queue for endpoint processes** parameter stores the queue that is used by the background processes to handle the actual export. The default value is **Out**.

STEP allows you to define separate queues for the endpoint and the endpoint process of an OIEP in the Configuration flipper of the OIEP editor. In this example, the queue for endpoint processes for the outbound integration endpoint is renamed to OutOIEP1.



The first time you activate the endpoint, a queue with the specified name is created if it does not already exist. Events are not lost if a separate queue for endpoint process is defined.

Changing the legacy **Queue for the endpoint process** for OIEPs means each OIEP background process uses this named queue for the actual export, causing OIEPs to run simultaneously and not wait for other OIEPs to finish processing. Do not use this when OIEPs require sequential export processing.

Recommendations

Configure the simplified BGP execution mechanism to run BGPs based on the priority of the BGP and the created time. Refer to the **BG Processes Execution Management** topic in the **System Setup** documentation.

Configure separate **Queue for endpoint processes** (to handle the actual export) for all high-priority or long-running OIEPs that do not require sequential processing (exports).

Refer to the **Background Processes and Queues** topic in the **System Setup** documentation for examples of parallel and multithreading properties.

Consider In-Memory for Exports

In-Memory can improve export performance because In-Memory provides faster read operations. This is beneficial for complex data models, such as export configurations with data aggregations and export configurations with data that navigates references.

For more information, refer to the **In-Memory Database Component for STEP** topic in the **Resource Materials** section of online help.

Optimize STEP Setup for Export Performance

Use the following setup recommendations to optimize export performance.

- Evaluate the export schedules of all exports and schedule long running exports sequentially (one after the other) to balance the STEP system load.
- Remove inactive and unused OIEPs (which includes queues, background processes, and export files, etc.) on operational production environments. When an OIEP is removed, the corresponding background processes are also removed.
- Set OIEPs to limit the number of background processes kept after export using the 'Maximum number of old processes' and 'Maximum age of old processes in hours' parameters, as defined in the **IEP Auto Delete Settings** section of the **Create a Background Processes Maintenance Plan** topic.

Export Elements to Limit

The following list includes export elements known to degrade system performance. This list can be used to troubleshoot existing exports, and can also be reviewed prior to creating new exports to prevent performance problems.

To ensure maximum export performance, the usual strategy is to limit the amount of information exported, or limit the number of times the same node is exported with the same data.

For general information on exports, refer to the **Data Exchange** documentation.

Limit Event-Based OIEPs

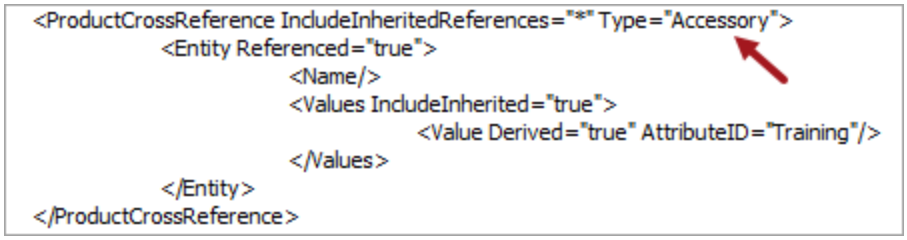
Event-based outbound integration endpoints (OIEPs) can adversely affect the performance of the system as a whole. The number of OIEPs can adversely affect the approval / external attribute change process because each change must be checked against the triggering definitions on all OIEPs to determine if any is interested in that specific change. A general guide is to have no more than 10 event-based OIEPs.

For more information, refer to the **Outbound Integration Endpoints** topic in the **Data Exchange** documentation.

Limit the Volume of Exported Data

Advanced STEPXML makes it possible to define what data to export while leaving out the other data. For example, only include the export attribute groups instead of including all attributes. The CSV export format and the Generic XML format allow configuring only the data that is needed.

In the following image, the Advanced STEPXML exports only the 'Accessory' reference type:



For more information, refer to the **Advanced STEPXML Format** topic and the **Data Formats** topic, both in the **Data Exchange** documentation.

Limit Unnecessary Data

Consider the impact of additional objects and determine if they are required, as many additional objects slows the export process. Generally, limit unnecessary data in the export using these methods:

- An event generator can be used to add additional objects via derived events, as defined in the **Generate Event** section of the **OIEP - Event-Based - Event Triggering Definitions Tab** topic within the **Data Exchange** documentation.
- A pre-processor can add additional objects to the export set associated with the event batch, as defined in the **Configure the Pre-processor and Post-processor** section of the **OIEP - Event-Based - Output Templates Flipper** topic within the **Data Exchange** documentation.
- The Advanced STEPXML template can include additional objects, for example, via references. For more information, refer to the **Filter References in STEPXML** topic within the **Data Exchange** documentation.

Limit Multiple Output Templates

Multiple output templates can be associated with a single outbound integration endpoint (OIEP) enabling the OIEP to handle different types of objects / events, each having the ability to be output in a different format. This flexibility allows the user to change how data is exported for different kinds of objects.

Important: While STEP allows multiple output templates per OIEP, it can have a severe impact on performance since a new batch is created each time a new output template is required.

Alternate approaches include:

- When possible, use different OIEPs to handle the 'family' and 'item' changes separately. This is not an option if the events need to be processed in sequence since the 'item' change could be exported, and delivered, before the 'family' change.
- Use STEPXML to easily contain the 'family' and 'item' data in the same file.

For more information, refer to the **OIEP - Event-Based - Output Templates Flipper** topic within the **Data Exchange** documentation.

Avoid Complex Export Privileges

When exporting, for each piece of information exported, all privileges are checked for the export user. This privilege check often includes a hierarchy check (e.g. 'is the product below a certain root node') which can take a significant amount of time for large exports. Limiting the number of privileges for the exporting user can improve performance dramatically.

Consider creating a user with relatively few and broad permissions explicitly for exports. Since only the user configured on the endpoint is relevant, this export user generally ensures improved performance.

Import Recommendations

This is one of the data gathering methodologies and recommendations for functional performance improvement. The full list is defined in the **Performance Recommendations** topic.

Bulk data imports can be executed manually or automatically on a schedule via an inbound integration endpoint (IIEP). Bulk data imports can vary from fast to slow, depending on the expectations and on following the recommendations for imports described in this section.

A clean, simple, optimized import that does no processing can process about a hundred records per second, whereas a complex import with complex processing may only process one record per second, or perhaps be even slower.

A slow import could be problematic based on the situation. Slow import performance of about one record per second, for example, is fine when a small number of records are imported on a daily basis. However, if the import contains a large number of records and it will take weeks to finalize, then this is likely a problem. This illustrates why it is important to clearly describe the use cases for imports.

If the STEP system suffers from perceived bad performance during import, these recommendations should be considered to optimize performance.

The available tools for importing data are defined in the **Data Exchange** documentation.

Optimizing performance in imports involves the following:

- Import Elements to Use
- Import Elements to Avoid
- Importing for Migration

Import Elements to Use

The following list includes import elements known to minimize impact on system performance. This list can be used to troubleshoot existing imports and can also be reviewed prior to creating new imports to prevent performance problems.

For general information on imports, refer to the **Data Exchange** documentation.

Use Term Lists for Price Data

The structure of the import file can be optimized for maximum import performance by using commercial data (also called terms lists) for price data where appropriate. Especially for complex, time-limited / quantity limited price data, terms lists can speed up imports.

For more information, refer to topics in the **Commercial Data** section of the **Publisher (Adobe InDesign Integration)** documentation.

Use Business Rules Designed for Import Performance

If business rules are required on import, carefully map the business rule execution of each import (especially endpoints) to understand the full impact of the configurations. Ensure any business rules running on import (via approval, import actions, or through a workflow), has acceptable performance.

Review the following items to ensure business rules are used efficiently:

- Simplify complex business rule JavaScript logic being executed on the import.
- When possible, move business actions running on the import to event processors. This only applies if the actions can be performed asynchronously from import.
- Business rules that read or update objects other than the one being imported (reference sources or targets, parents, children, etc.) will reduce performance because only the imported product is likely to be resident in cache at the time. Business rules that iterate through children of the imported product are particularly common, but very expensive performance-wise.
- Consider if it is possible to ensure that no business conditions exist, allowing a logical exception or a lock contention. Business rule execution for the purpose of automation or transformation is secondary and should be treated as fail-tolerant. This means that the criticality of a rule failure is far less than that of the import operation itself. In the event of an exception, rules can be fixed and executed again, whereas import records that are skipped can be difficult to rectify on a busy system.

For more information, refer to the **Business Rule Recommendations** topic.

Use Workflow Initiations Designed for Import Performance

When a product is initiated into a workflow, or a state transition is triggered by an import, all business rules configured on exit of an existing state, transition between, or entry to the next state execute as part of the import process.

The following scenarios can heavily impact the import performance:

- If workflow initiations and/or transitions are necessary, consider if the business rules trigger on entry or exit. The import performance may be heavily impacted by the executing of these transitions.
- If business rules initiate more workflows or auto submit to other states, a single workflow submission may cascade into hundreds (or even thousands) of lines of executed business rule logic.
- When a workflow business rule conditions fails, the current transaction is rolled back.

Use Approvals Designed for Import Performance

If approvals are necessary at import, then consider which approval conditions and actions will be executed.

Ensure that endpoints are importing externally-maintained data, since it requires no approvals. In this scenario, consideration for approval conditions and actions is not required. Also, externally-maintained data has no revision history. Therefore, revision history growth is not a consideration.

Use Event-Based Exports Designed for Import Performance

When importing externally maintained data, or importing and approving, all changes must be checked against any event-based outbound integration end points (OIEPs), to verify if an event should be generated.

Consider which approval events are queued on OIEPs and which event filter / generator rules will be triggered. If a large number of OIEPs exists, and if these have Event Triggering Definitions on attribute groups, for each check, the system must check if the attribute exists under the given attribute group. This can lead to performance degradation.

Ensure that the OIEPs are triggered to as specific and few attributes as possible. For more information, refer to the **Creating an Event-Based Outbound Integration Endpoint** topic in the **Data Exchange** documentation.

Legacy - Use Separate Queues for Important IIEPs

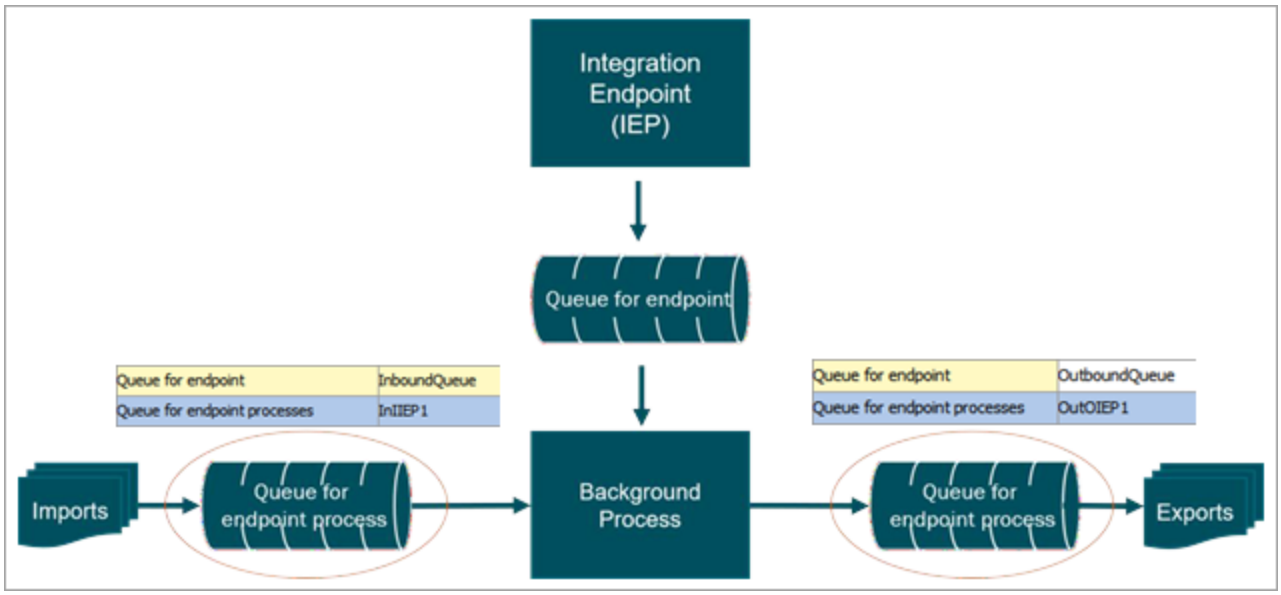
Legacy background process (BGP) functionality uses specified queues, while the recommended BGP execution mechanism runs BGPs based on the priority of the BGP and the created time. Refer to the **BG Processes Execution Management** topic in the **System Setup** documentation.

The inbound integration endpoint (IIEP) initiates a background process which handles the actual import.

- The legacy **Queue for endpoint** parameter stores the queue that is used by the background process to poll the inbound integration endpoint. The default value is **InboundQueue**.

- The legacy **Queue for endpoint processes** parameter stores the queue that is used by the background processes to handle the actual import. The default value is **In**.

STEP allows you to define separate queues for the endpoint and the endpoint process of an IIEP in the Configuration flipper of the IIEP editor. In this example, the queue for endpoint processes for the inbound integration endpoint is renamed to InIIEP1.



The first time you activate the endpoint, a queue with the specified name is created if it does not already exist. Events are not lost if a separate queue for endpoint process is defined.

When changing the **Queue for the endpoint process** for IIEPs, each IIEP background process uses this named queue for the actual import. This means that the IIEPs run simultaneously and not wait for other OIEPs to finish processing. Do not use this when IIEPs require sequential export processing.

Recommendations

Configure separate **Queue for endpoint processes** (to handle the actual import) for all high-priority or long-running IIEPs that do not require sequential processing (imports).

Refer to the **Background Processes and Queues** topic in the **System Setup** documentation for examples of parallel and multithreading properties.

Optimize STEP Setup for Import Performance

Use the following setup recommendations to optimize import performance.

- Remove inactive and unused IIEPs (which includes queues, background processes, and import files, etc.) on operational production environments. When an IIEP is removed, the corresponding background processes are also removed.
- Set hotfolder and REST Receiver IIEPs to remove files after import using the 'Keep file after load' parameter, as defined in the **Hotfolder Receiver** topic of the **Data Exchange** documentation.
- Set IIEPs to limit the number of background processes kept after import using the 'Maximum number of old processes' and 'Maximum age of old processes in hours' parameters, as defined in the **IEP Auto Delete Settings** section of the **Create a Background Processes Maintenance Plan** topic.
- Use the standard asset importer, instead of the legacy asset importer is used, as defined in the **Importing Assets** topic within the **Digital Assets** documentation.

- Assets and other objects are not imported into a single folder, as defined in the **Hierarchy Builder** topic within the **Digital Assets** documentation.
- Evaluate the import schedules of all imports and schedule long running imports sequentially (one after the other) to balance the STEP system load.
- Review high-priority integrations and integrations with long-running processes. The recommended simplified background process (BGP) execution mechanism prioritizes the order in which background processes are handled. Refer to the **BG Processes Execution Management** topic in the **System Setup** documentation.

  In a legacy background process (BGP) implementation, these integrations should have their own queue for endpoint processes where the background process handles the actual import. The queue is automatically created on the system if it does not already exist. For more information, refer to the **IIEP - Configure Endpoint** topic within the **Data Exchange** documentation.

- Consider In-Memory to improve import performance, as defined in the **In-Memory Database Component for STEP** topic in the **Resource Materials** section of online help.
- Limit the number of OIEPs listening on modify events to avoid excessive checking of event validity, as defined in the **Events** topic of the **System Setup** documentation.
- Use the 'Relaxed' locking policy for commonly-referenced products to improve performance of imports, as defined above.
- Use parallel imports only as needed, considering the performance degradation to the rest of the system, as defined above.

Import Elements to Avoid

The following list includes import elements known to degrade system performance. This list can be used to troubleshoot existing imports and can also be reviewed prior to creating new imports to prevent performance problems.

For general information on imports, refer to the **Data Exchange** documentation.

Avoid Typical Import Errors

The following errors are typical for imports:

- Invalid attribute values - These errors can be caused by the attribute setup parameter in the import not matching the parameter in the system. For example, the attribute's validation type, mask, minimum and maximum values, maximum length, LOV, the object's own object type, and so on.
- Object type validity - Even if you load attribute values for an object, not all values will be automatically available in STEP. Attributes must be made valid for an object before the values can be accessed via references. STEP will always load attribute values if the object's type is valid for the attribute, and the attribute values meet the validity criteria. But, the attribute itself must be a valid attribute for that object, that is, linked somewhere in the object hierarchy or classification hierarchy where the object resides.
- Inaccurate STEPXML statements - Performance degrades when the STEPXML import file has the wrong case for the property 'UserTypeID' (such as 'UserTypeId'). If the STEPXML import file has no Parent IDs for the products, then new products are not imported. To skip importing new products, add the reject new product tag in the file header (RejectNewProducts="true") to avoid the missing Parent ID error.

Background Process (BGP) progress and the number of errors and warnings are shown on the Background Processes tab under the Execution Report flipper as follows:

- For manual imports using Import Manager - on the BG Processes tab under the 'Import Manager Pipeline' node
- For IIEPs - on the Background Processes tab click the BGP link to display information for this particular background process. In both cases, on the Background Processes tab, open the Execution Report flipper to are also displayed as shown below.

For a list of common errors, refer to the **Import Error Messages** topic and the **Import Error Message Examples** topic in the **Data Exchange** documentation.

Avoid Missing Reference Targets

Ensure that import files do not have reference nodes that are missing in the file (missing reference targets).

If the reference target does not exist in the import file, then the target does not exist at the time of import. The importer skips the product and triggers a second import pass after the initial completion (testing if the target exists later in the same file). If the reference target does not exist in the system at the time of the second import pass, the background process logs an error due to a missing reference target. A second import pass caused by missing targets bears a heavy performance impact.

This type of error illustrates the importance of the sequence of nodes in the import file. Refer to the **Avoid Forward Declarations** section below.

Avoid Forward Declarations

The structure of the import file can be optimized for import performance by ensuring that referenced nodes are not being created in the file after they are needed (forward declaration).

If the reference target exists later in the same import file, then the target does not exist at the time of import. The importer skips the product and triggers a second import pass after the initial completion. The reference target will exist in the system at the time of the second import pass, which will allow the import to succeed.

A second import pass caused by missing targets bears a heavy performance impact. This illustrates the importance of the sequence of the nodes in the import file. Refer to the **Avoid Missing Reference Targets** section above.

Avoid Multiple Updates of Same Object

The structure of the import file can be optimized for maximum import performance by ensuring that the same product or entity is not updated several times in a single file or in multiple files.

- When creating import files, structure the files by modified nodes, not by attribute. It is faster to update a single node with 10 attributes once, than to update each attribute individually.
- Update the same node with all relevant information in one file, rather than splitting it over multiple files or imports. Thus, understand the data patterns of busy endpoints and attempt to structure import files so that any given product or entity appears in the fewest number of import files as possible.

  If a given product or entity is distributed across several import files instead of consolidated into one, this will cause duplicate execution of business rules, approvals and cache load. It also has the potential to dramatically increase revision history, which is a performance detriment of its own.

  Updating the same product or entity at the same time across multiple imports can result in Optimistic Lock failures.

Avoid Unnecessary Business Rules on Import

Using business rules on imports is a powerful tool, but like all powerful tools comes with some risks. Complex JavaScript business actions add execution time to imports.

Review the following items to ensure business rules are used efficiently:

- Are automated actions *required* on import? The fewer automated actions on import, the faster the import performs.
- Does the business process *require* actions on import? When possible, design your solution so imports can run free of any business actions. An event processor can be used to perform business actions that are not necessary to be run on import.
- Are the business rules running on import via workflows or approvals *required*? While extremely useful, all of these options impact performance.

Once it is determined that a business rule is required, verify that there are no unnecessary business actions being performed, such as:

- Are there duplicate executions? For example, an approval triggered in a global business rule as well as in a local state transition rule.
- Are business rules reading or updating objects other than the one being imported (reference sources or targets, parents, children, etc.)? This reduces performance because only the imported product is likely to be resident in cache at the time. Business rules that iterate through children of the imported product are particularly common but very expensive performance-wise.
- Do the business conditions allow a logical exception or lock contention? Verify business rule execution for the purpose of automation or transformation is secondary and is fail tolerant. This means that the criticality of a rule failure is far less than that of the import operation itself. In the event of an exception, rules can be fixed and executed again, whereas import records that are skipped can be difficult to rectify on a busy system.

For more information, refer to the **Business Rule Recommendations** topic.

Avoid Optimistic Locking in Business Rules on Import

If business rules on import are determined to be required, minimize the risk of optimistic locking caused by business actions on import:

- Business actions that execute on tree structures or commonly referenced objects may contribute to optimistic locking or even deadlock failures.
- Any business action that triggers an approval inside of a try / catch block can cause an optimistic lock (as well as subsequent retries and eventual failure) if an approval condition failure is caught and the original exception is not re-thrown.
- Any action that throws an error with the word 'exception' in the error message results in the importer functioning as if an optimistic lock occurred, resulting in retries and eventual failure.

For more information, refer to the **Business Rule Recommendations** topic and the **Optimistic and Pessimistic Locking Recommendations** topic.

Avoid Complex Privileges on Import

When importing, all privileges are checked for each piece of information imported, even when the data is not imported. For example, when performing a hierarchy check (e.g., is the product below a certain root node), performance can be impacted significantly for large imports.

To avoid excessive privilege-checking and improve performance dramatically, the import user should have as broad and as few privileges as possible. For IIEPs, this applies only to the user configured on the endpoint.

Importing for Migration

The import recommendations apply to operational scenarios on production environments. However, the recommendations can also be applied to migration scenarios, although the migration scenarios are usually performed on a separate environment where the results are copied to a production environment via database copies.

Initial data migration is typically handled differently from standard imports because it is a one-time operation and the volume of data is generally far greater than a typical import would be expected to process. It is also generally expected that a greater level of effort will be invested in preparing the import messages or files so that the migration can be completed over a reasonable period.

When preparing migration data files consider the following:

- Transformation business rules may be required specifically for migration. Attempt to avoid using rules that read from or write to many related objects or children. Business rules should, wherever possible, only transform data on the object being imported.
- With serial endpoints, attempt to load products in the smallest number of import files possible. For example, load each product exactly once, providing full attribution. This reduces the number of times each product must read / flush from cache.
- If necessary, set the migration endpoint to parallel processing to use multiple concurrent background processes.
- If there are many references between objects, optimistic locks or deadlocks may occur. Even in the absence of actual locking errors in the logs, you may find that performance is slower than expected due to the lock waits required to update reference targets.
- Consider using two passes. The first set of import files is loaded for an IIEP running parallel and contain all information about the products except for references. The second set of import files is loaded for a separate serialized IIEP and contain only references. Start the product information import IIEP first and allow it to progress enough to load the products being referenced by the second set of files. Both endpoints can run at the same time as long as the products are not being processed by the two IIEPs at the same time.
- When using parallelized endpoints, avoid business rules that update products other than the one being loaded. This scenario can produce optimistic locking errors. If this type of business rule is required, consider using bulk updates to execute the logic after import is complete.

Matching and Linking Recommendations

This is one of the data gathering methodologies and recommendations for functional performance improvement. The full list is defined in the **Performance Recommendations** topic.

Matching and Linking functionality relies on these underlying components:

- **Match Codes**: define a string of attributes to compare, such as name, address, birth date, and email for customer data; or GTIN, EAN, part number, and SAP entity for product data.
- **Matching Algorithm**: evaluates match codes for potential duplicates and displays a percentage to indicate if they are the same record (via auto threshold) or likely the same record (requiring clerical review). The matching algorithm also defines how the duplicates are handled, such as merge or generate a golden record. Survivorship rules of the matching algorithm determine which attributes are copied to the golden record (name, attributes, references).
- **Event Processor**: triggers events which can be acted upon by the matching and linking business rules or matching and linking outbound integration endpoint.

Match codes limit the number of comparisons that the matching algorithm must make, and an event processor launches the associated business rule or OIEP. The system load of matching and linking can have a negative effect on the performance of STEP, especially if there are a lot of supplier items imported, if the matching and linking process is complex, or if the approval of the golden record is part of the process. For more information, refer to the **Matching, Linking, and Merging** documentation.

While the matching and linking and golden record functionalities are designed and based on business requirements, use the following recommendations when a matching and linking process performs poorly.

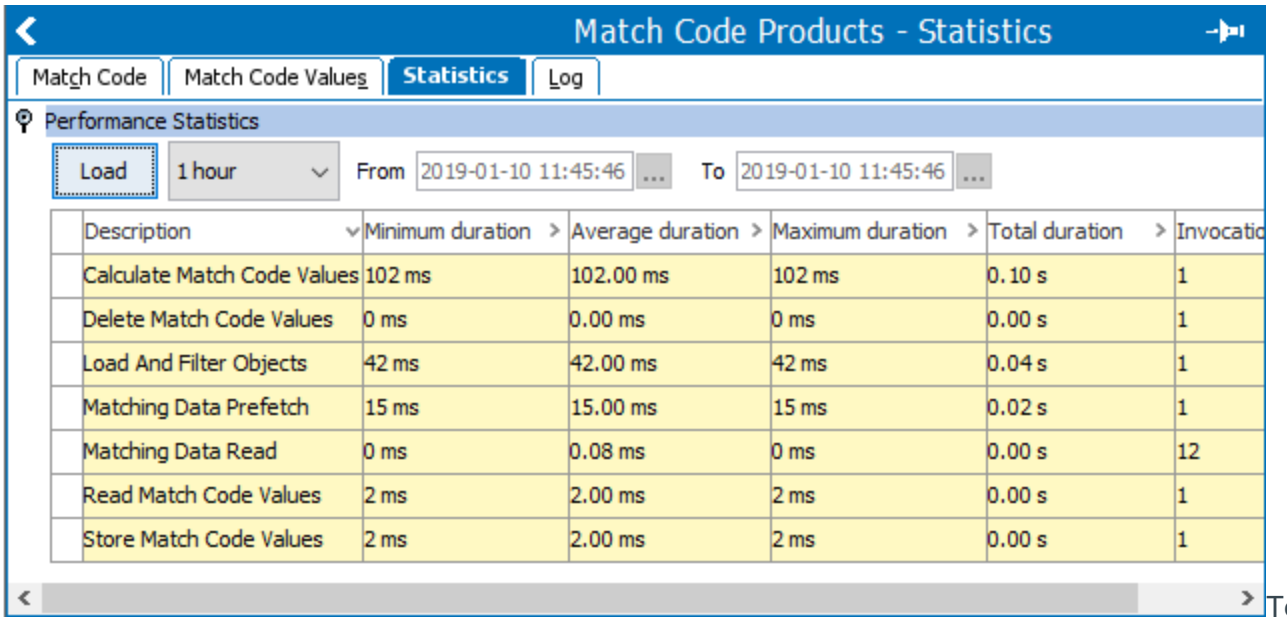
Optimize Match Codes and Matching Algorithm

For details on match codes and matching algorithms, refer to the **Match Codes** section within the **Matching, Linking, and Merging** documentation.

- Evaluate the performance of match code generation via the 'Maintain Match Code Values' option by reviewing the Match Code object's Statistics tab 'Performance Statistics' flipper details. Modify the items reported to have poor performance.
- Evaluate the performance of the matching algorithm execution via the 'Generate Events for Matching' option by reviewing the Matching Algorithm object's Statistics tab 'Performance Statistics' flipper details. Modify the items reported to have poor performance.

Statistics on Match Codes Generation

The performance of the generation of the Match Codes can be measured in STEP. The Match Code Statistics tab allows you to verify the Match Codes are performing well and identify performance problems.

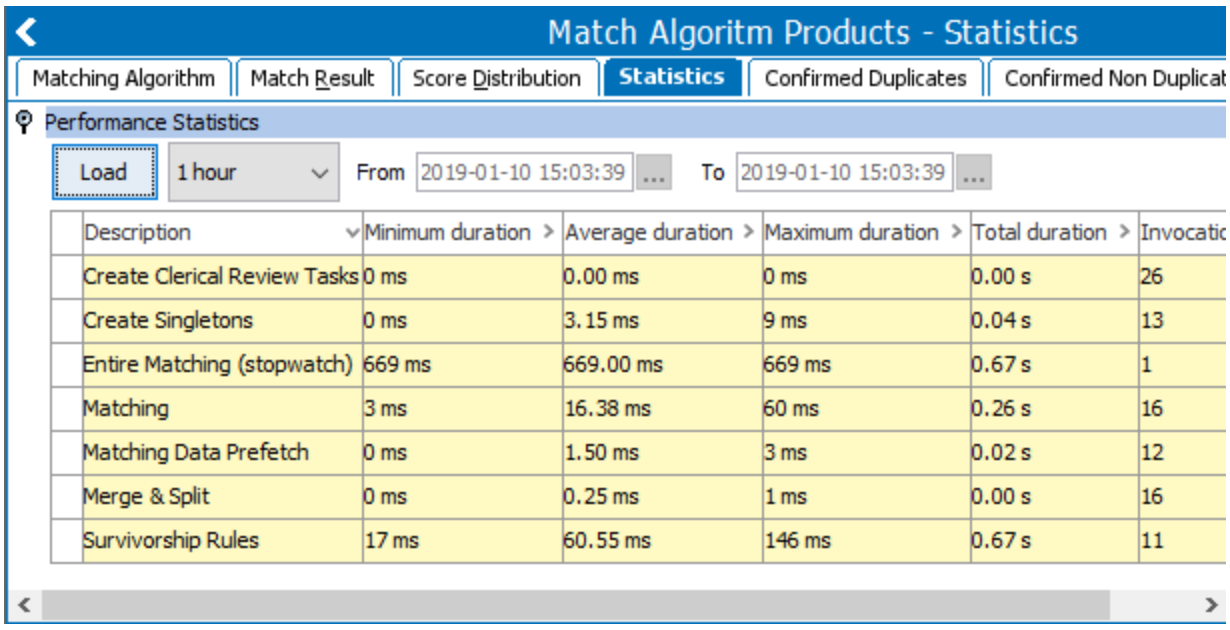


Te

For detailed information on the 'Maintain Match Code Values' option and the 'Match Code Values' tab, start by going to the **Match Codes** section within the **Matching, Linking, and Merging** documentation.

Statistics on Running the Matching Algorithm

The performance of the running of the Matching Algorithm can be measured in STEP. The Matching Algorithm Statistics tab allows you to verify that Matching Algorithm generation is performing well and identify performance problems.



Limit Attributes Promoted to Golden Records

Attributes can be promoted to golden records via the 'Trusted Source Value' survivorship rule. When the attributes belong to an attribute group with thousands of attributes (or more), such as an ETIM attribute group, the golden record process can degrade performance significantly. To avoid this performance issue, limit the number of attributes in the attribute group that is subject for promotion.

In most cases, this can be achieved by defining an attribute group that includes a limited set of attributes.

If the number of attributes in the attribute group cannot be limited (such as ETIM attribute groups), use a business action during the survivorship rules, instead of using the 'Trusted Source' option.

Approve Golden Records Outside of Matching and Linking

Golden records can be approved within the matching and linking process, separate from the process, or not approved at all.

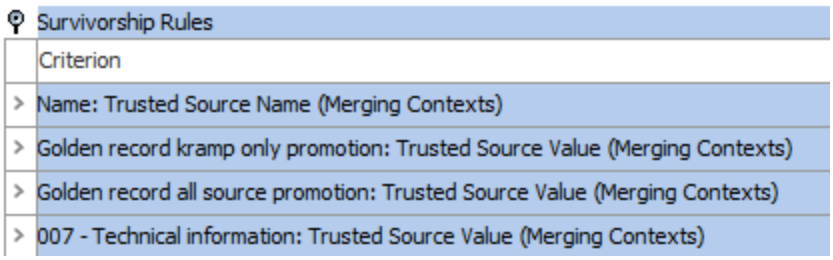
When approving the golden record is required, for recommended performance, use an event processor trigger to run a bulk update after the matching and linking process is complete.

Avoid Multi-Context Survivorship Rules

STEP allows multi-context survivorship rules for context specific names, context specific attribute values, context specific image references, and context specific web-hierarchy links.

However, when a multi-context survivorship rule is executed and the global value is not available, the matching and linking process continues through all contexts to find an attribute value in a local context. Ultimately, the local value is copied to the golden record in that local context. This process negatively effects performance.

In legacy STEP versions, the multi-context survivorship rules (available via a custom extension) were called 'Merging Contexts' as shown below.



In current versions, multi-context survivorship rules are standard and are called 'Multi Context' as shown below.

Multi Context Trusted Source Value

If multiple languages are defined, then a multi-context survivorship rule for a language-dependent name, attribute values, primary image references, or web-hierarchy links attempts to promote the correct value of the supplier record in the right language of golden record.

By default the 'Allow values from …' checkbox is selected, and allows sub-dimension point to be updated from less trusted sources.

The following examples illustrate this effect:

Assume the name is language dependent and the trusted sources are 'SLRV' and 'CNET,' in that order.

- The SLRV source (silver record) only has a name for 'Language Root' context.
- The CNET source has a name for 'English' context (which is a sub-dimension point to 'Language Root').

The English name will only be allowed to be promoted to the golden record if the checkbox is checked.

Optimistic and Pessimistic Locking Recommendations

This is one of the data gathering methodologies and recommendations for functional performance improvement. The full list is defined in the **Performance Recommendations** topic.

STEP uses locks to prevent data corruption due to concurrent updates by multiple processes. For example, consider a bank account where one person deposits $100 and another person concurrently withdraws $100. If the two (2) processes read the balance at the same time, then add or subtract $100 and update the balance, the balance is either increased or decreased by $100. But the correct result is for the balance to remain the same.

In STEP, when concurrent threads attempt to generate new node revisions, locks protect the node's revision history to avoid data corruption. Contention for row locks indicate a thread is locked (often by business logic) while waiting for another thread to commit data. To avoid row lock contention, business logic that runs updates (like bulk updates, approvals, workflow transitions, imports, etc.) should be fast to execute and limit the updates to a small number of objects. For example, updating a field on child products when approving a product may work reasonably well when updating a product with 10 children. However, it is not uncommon for some products to have thousands of children. With the 'update child field' business logic, the approval would likely run for a very long time, and as a result, the product being approved would be locked, blocking other attempts to update that product until the approval operation is complete.

The available locking options include:

- **Optimistic** - This is the default option for STEP and involves no locking but increases the version number of an object in the database. When a process attempts to update a value on a node, it reads the version number, updates the value, and then re-reads the version number. If another process has updated the value in the meantime, then the first process retries up to 10 times (including 2000 ms waiting time) until successful.

  Optimistic locking exceptions indicate that additional attempts were required before the update was successful. The exceptions prevent two (2) identical updates from starting concurrently which would result in an infinite loop. However, even with a successful update, the update runs slowly.

  - Success / Failure: If the version number is the same as it was before the update, the version number is incremented. If, in the meantime, another process has updated the value, an optimistic locking exception is thrown and the update is not committed.
  - User Interface / Automatic Process: If an optimistic locking error occurs in the user interface, the user is immediately notified that another process has already updated the same data. If an optimistic locking error occurs in an automatic process, the lock is reported after the process retries 10 times. The result is that optimistic locking errors can slow down processes without reporting any errors. The longer a transaction takes, the higher the probability of optimistic locking issues when other processes run simultaneously.
- **Pessimistic** - Locking occurs when two or more processes attempt to update a value in the database at exactly the same time. Other processes must wait for the lock to be released before proceeding which can degrade performance. STEP uses pessimistic locking when creating a new revision of a node and when

creating a link to a node unless the target node type is set to a relaxed locking policy.

The disadvantage to pessimistic locking is that until the lock is released, the locked data cannot be edited even if the user that originally locked the row has completed their task. There is also a higher risk for deadlocks.

Deadlocks

Database deadlocks occur when two or more processes have locked a resource, and each process requests a lock on a resource that another process has already locked. One or more transactions are waiting for one another to release a lock on their process.

For example, Process A has resource 1 locked and is requesting a lock on resource 2. Process B has resource 2 locked and is requesting a lock on resource 1. A deadlock occurs and is not resolved until one of the processes releases a lock. Although this example is simplistic, it shows how deadlocks work and can be resolved.

This topic includes the following sections:

- 'Reference Target Lock Policy' Parameter
- Optimistic Locking - Log Level Setting
- Optimistic Locking - Analyze Failures
- Optimistic and Pessimistic Locking Recommendations

'Reference Target Lock Policy' Parameter

Both optimistic and pessimistic locking are impacted by the 'Reference Target Lock Policy' parameter on the object types of the objects being handled. For automated processes that create references to target objects, the 'Relaxed' setting is beneficial on the Reference Target Lock Policy parameter of the target object types.

The 'Reference Target Lock Policy' parameter for an asset, entity, classification, and product object type manages how objects are locked while they are being referenced.

When long-running transactions cannot be simplified easily, consider setting the 'Reference Target Lock Policy' parameter to 'Relaxed' on the object types to which the long transaction applies.

Strict or Relaxed

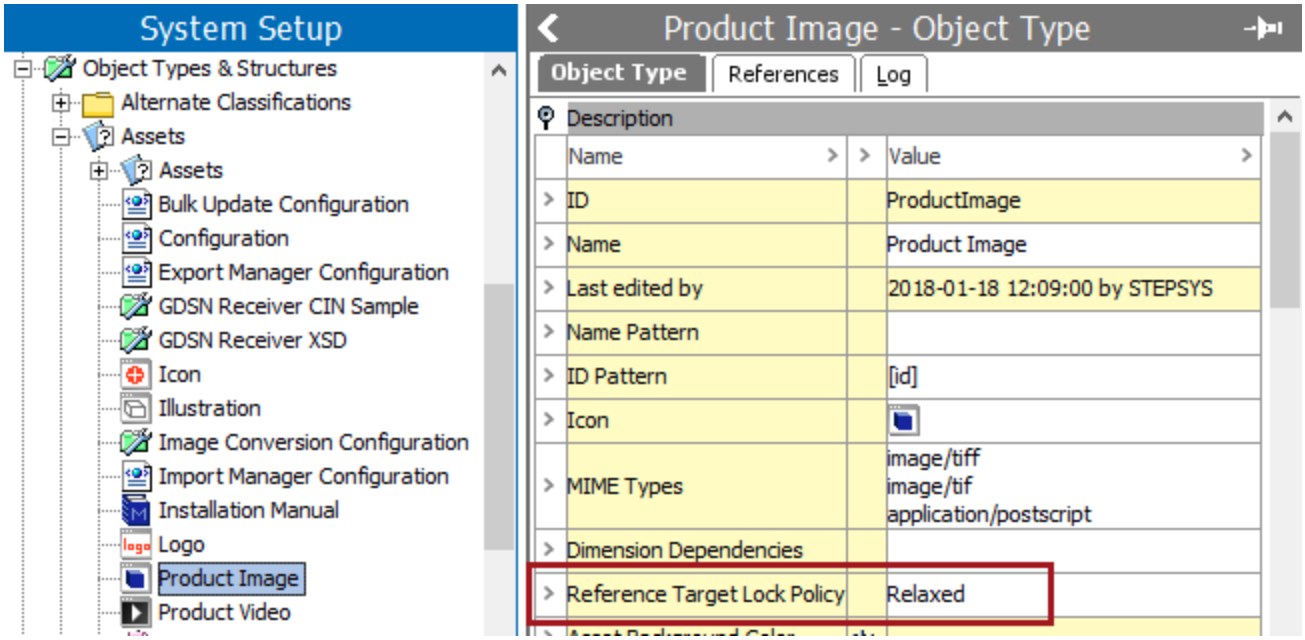
The 'Strict' setting is default for the 'Reference Target Lock Policy' parameter. When a reference between two objects is being created, the target object is locked to ensure that it is not being deleted while the reference is being created. This means that only one process or user can edit it at a time. Often, deletion of the object is not a deletion from the database, but a deletion due to revision control when the insert of a new history entry occurs.

When using the Optimistic Locking option, a 'Strict' setting can result in an issue running parallel imports where the first import locks the object type being referenced, and the second import eventually stops running because it cannot access the locked object. Since STEP continues to retry the import, this can negatively impact inbound feeds.

The 'Relaxed' setting uses a less restrictive lock (a shared 'user lock') on the reference target objects being edited so that concurrent updates by multiple processes and/or users are allowed. A full lock is used only when a deletion is attempted. The Relaxed setting allows faster parallel creation of references to the same target object. This setting, when used with object types that are frequently referenced but rarely deleted, improves the performance and stability of parallel inbound imports, bulk updates, and users concurrently creating references to the same objects.

Important: Do not use the 'Relaxed' setting on an object type for objects that are often deleted as this can result in poor performance, and risks locks and deadlocks.

For example, the image below shows the 'Reference Target Lock Policy' is set to 'Relaxed' for object type 'Product Image (ProductImage)' since the many of the optimistic locking errors in the asset push apply to this object type.



Optimistic Locking - Log Level Setting

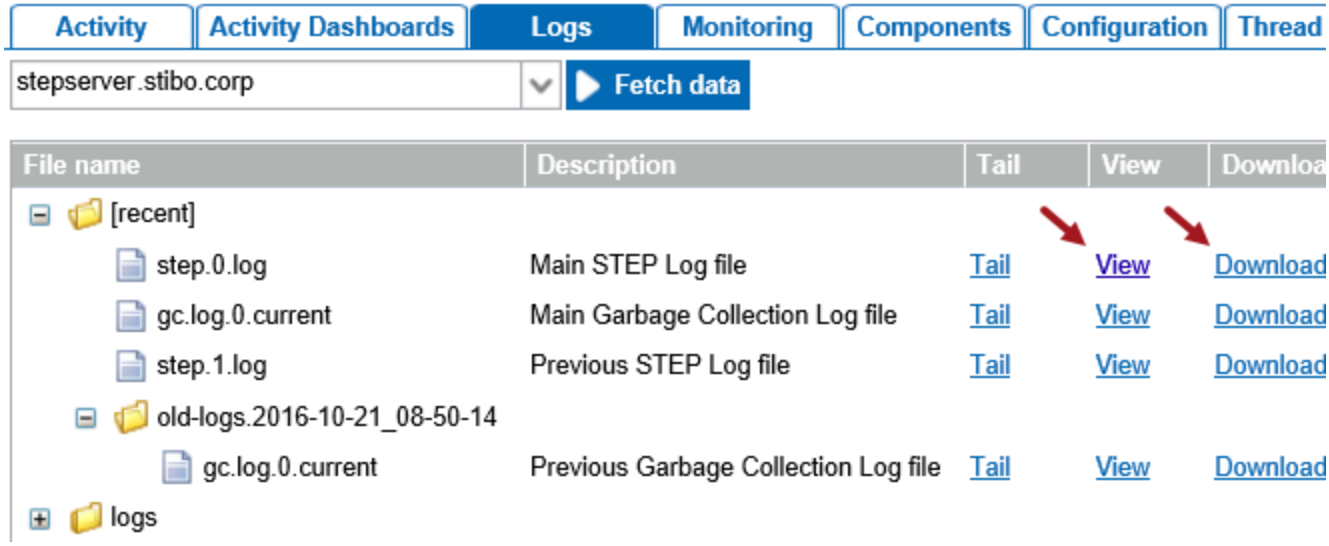
STEP puts a transaction on hold when optimistic locking occurs and after some time, tries to process the transaction again. This illustrates how optimistic locking errors can degrade the performance of imports, exports, asset push, business rules, etc. and can degrade the performance of the STEP system.

The message **Optimistic lock exception occurred, retrying** occurs for optimistic locking errors with fewer than 10 retries. When the **Log.Level.com.stibo.core.domain.impl.ManagerImpl** property is set to FINER or FINEST in the sharedconfig.properties file on the application server, these errors are written to the step.log files. For more information, refer to the **Logs** topic in the **Administration Portal** documentation.

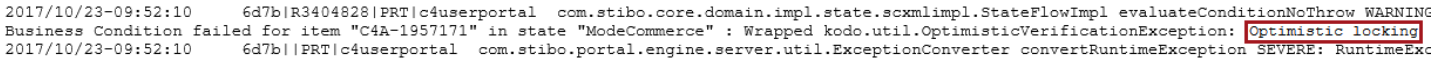
Optimistic Locking - Analyzing Failures

Use the STEP Admin Portal to analyze optimistic locking errors.

1. From the Start Page, click the STEP System Administration button and supply the login credentials.
2. On the **Logs** tab, click the **Fetch data** button to load the data.
3. Select the **Main STEP Log File** and click either the **View** link or the **Download** link to review the log.



4. Search the log for the instance of 'optimistic locking' text.



5. Determine the cause of the optimistic locking error as shown in the **Examples** section below.
6. Follow the steps in the **Optimistic Locking Recommendations** section to reduce optimistic locking errors.

Examples

Optimistic locking errors are caused by long transactions and may be an import, export, business rule, asset push, etc. The following examples illustrate how to identify what caused the optimistic locking errors.

- Warning for optimistic locking caused by Asset Push

```
2017/10/25-16:34:07 9e8e com.stibo.services.assetpush.beans.AbstractServiceBean
wrapUnexpectedException
WARNING Caught unexpected: kodo.util.OptimisticVerificationException: Optimistic
locking errors were
detected when flushing to the data store. This indicates that some objects were
concurrently modified
in another transaction. Failed objects: [AssetPO@7caa505: 3615004291612_1,
AssetPO@526cb477:
3615004302998_3, AssetPO@3c0fccc3: 3615004302998_pantone, AssetPO@3bd3211d:
3615004291612_3,
AssetPO@696359ef: 3615004302998_1, AssetPO@5e9b87b8: 3615004291612_pantone,
```

```
AssetPO@1cb0217c:
3615004302998_2, AssetPO@3e49203e: 3615004291612_2] [java.util.ArrayList]
```

- Severe optimistic locking error caused by a Web UI transaction in Web UI 'C4userportal'

```
2017/10/17-12:34:59 795|L3022356|PRT|C4userportal
com.stibo.portal.engine.server.util.ExceptionConverter
convertExceptionSerializeAndLocalize SEVERE Type: Error no esperado, Message:
Optimistic locking errors
were detected when flushing to the data store. This indicates that some objects
were concurrently
modified in another transaction. Failed objects: [ProductPO@21d744cc: C4A-305195,
Value7PO@5b8b4f21,
valno: 231765818, qualifier: -5, rev: 0, Value7PO@344dc391, valno: 231765817,
qualifier: -5, rev: 0,
Value7PO@3f2b8947, valno: 231765815, qualifier: -5, rev: 0, Value7PO@5478ad23,
valno: 231765816,
qualifier: -5, rev: 0, com.stibo.core.persistence.NodeStatePO@36bc02a7]
[java.util.ArrayList]
```

- Warning for optimistic locking caused by business condition transaction in state 'ModeCommerce' in Web UI 'C4userprotal'

```
2017/10/23-09:52:10 6d7b|R3404828|PRT|c4userportal
com.stibo.core.domain.impl.state.scxmlimpl.StateFlowImpl
evaluateConditionNoThrow WARNING Business Condition failed for item "C4A-1957171"
in state
"ModeCommerce" : Wrapped kodo.util.OptimisticVerificationException: Optimistic
locking errors were
detected when flushing to the data store. This indicates that some objects were
concurrently modified
in another transaction. Failed objects: [ProductPO@3c0b71a7: C4V-1957172,
ProductPO@2f9030b2:
C4V-1038657, ProductProductReferencePO@284a0999, edgeid: 23130110, rev: 0]
[java.util.ArrayList]
(Script#305) in Script at line number 25 at column number 0
```

Optimistic and Pessimistic Locking Recommendations

For both locking options:

- **Modify the reference target lock policy** - set the parameter to 'Relaxed' when all of these points are true:
  - optimistic locking failures occur frequently in the log file
  - transactions cannot be optimized
  - the object types that are frequently referenced are rarely deleted

  For more information, refer to the **'Reference Target Lock Policy' Parameter** section.

For the optimistic locking option:

- **Optimize large transactions** - keep transactions small to limit the probability of introducing an optimistic locking failure by optimizing the business rules and functions used at e.g., imports, exports, bulk updates.
- **Temporarily set** `Log.Level.com.stibo.core.domain.impl.ManagerImpl = FINER or FINEST` to analyze all optimistic locking issues as defined in the **Optimistic Locking - Log Level Setting** section.
- **Analyze optimistic locking problems** - defined in the **Optimistic Locking - Analyzing Failures** section.

Privilege Recommendations

This is one of the data gathering methodologies and recommendations for functional performance improvement. The full list is defined in the **Performance Recommendations** topic.

While STEP allows a very granular privilege system and privilege setups, complex privilege models can lead to a degradation in performance. Running STEP as a user with a large number of very specific privileges influences the performance of any action in STEP that goes across a large number of nodes, values, or references. This performance impact includes export, import, bulk update, recursive approval, matching, and 'multi views' like task list and multi editor.

For more information, refer to the **Privilege Rules** topic in the **System Setup** documentation.

Privilege Configurations

Privileges are additive only, which means that whenever a basic action is executed, STEP looks for the first privilege that provides the permission.

In terms of performance, the most expensive privilege check is attempting a task for which the user does not have access. The least expensive privilege check is when a user has global permission to everything.

Additionally, consider the following when setting privileges:

- Very specific and granular permissions result in a longer search for the appropriate privilege.
- Assigning privileges on a group of objects (and using the hierarchy to access these objects) provides a less expensive listing than assigning privileges on each object separately.
- Avoid excessive privilege checking to improve performance.

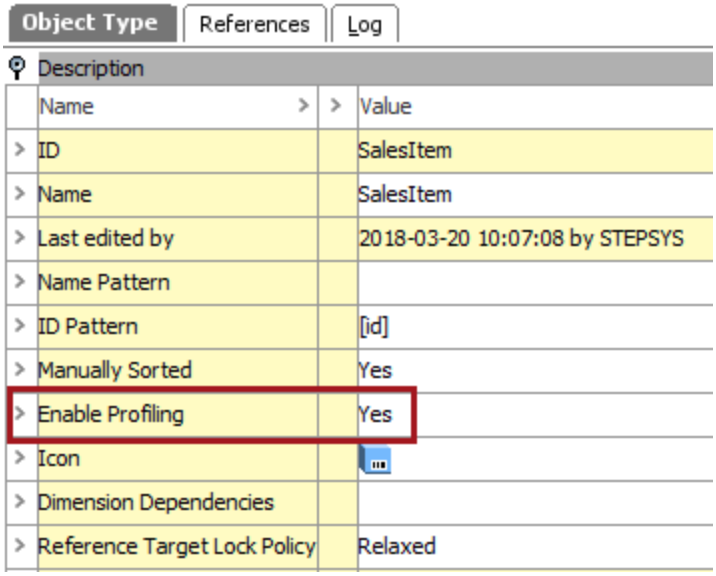
Profiling Recommendations

This is one of the data gathering methodologies and recommendations for functional performance improvement. The full list is defined in the **Performance Recommendations** topic.

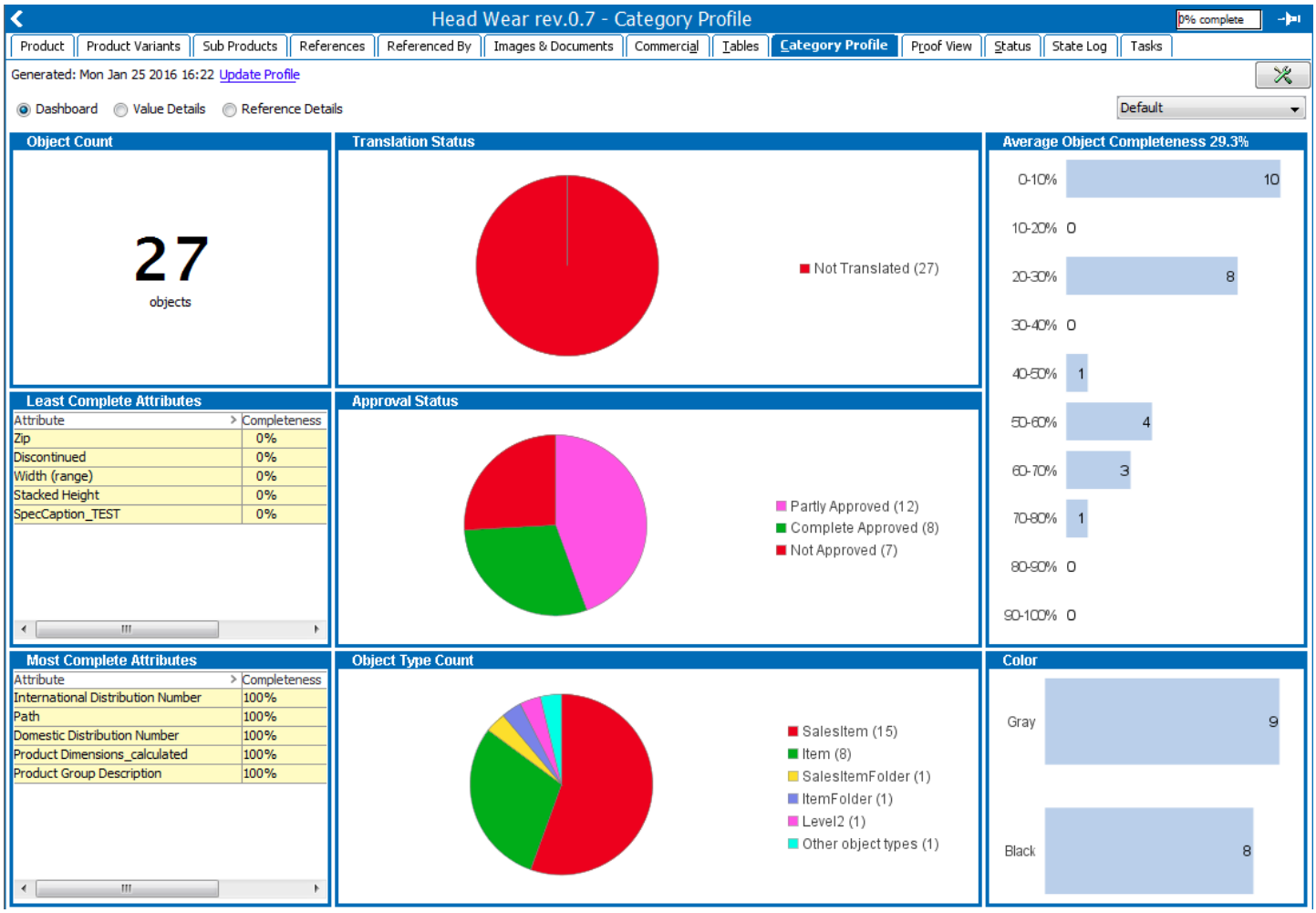
The Category Profile and Data Profile functionality provides a detailed overview of data in a specific branch of the hierarchy in Tree. However, when large categories are profiled, the system uses a lot of memory which can have a negative impact on system performance.

To identify and analyze object types with

Profiling is enabled in System Setup, using the 'Enable Profiling' parameter with the Object Types & Structures node.



When a profile is run, information about the data is displayed as follows and provides access to correct data errors:



For more information, refer to the **Data Profiling** documentation.

Recommendations

Review object types with profiling enabled by exporting all object types using the STEPXML template below and search for 'IsCategory="true".

```
<STEP-ProductInformation>
<UserTypes ExportSize="All"/>
<EdgeTypes/>
<CrossReferenceTypes ExportSize="All"/>
</STEP-ProductInformation>
```

- **Only enable profiling when required**.
- **Limit memory usage when profiling is enabled** via the following case-sensitive sharedconfig.properties:

  DataProfile.MaxDistinctAttributeValuesConsideredDuringProfileGeneration - Sets the maximum number of distinct attribute values per attribute considered during profile generation. The default setting is 100. When the limit is reached, the following happens:

  - Frequent value counts can become inaccurate. STEP uses a counting implementation dedicated for counting in big data collections with a limited memory usage from Clearspring Analytics.
  - The rare value count is disabled because only a frequent count can be maintained. In the profile, the frequent and rare values cells for attributes with too many distinct values are displayed with a light red background color. The attribute completeness and count, and the value instance counts for profiled attributes are correct

  DataProfile.MaxDistinctTargetsConsideredDuringProfileGeneration - Sets the maximum number of distinct targets for the reference or link type that is profiled. The default setting is 100.
- **Consider In-Memory** - Optimizes profiling performance. For more information, refer to the **In-Memory Database Component for STEP** topic of the **Resource Materials** section of online help.

Revision Control Recommendations

This is one of the data gathering methodologies and recommendations for functional performance improvement. The full list is defined in the **Performance Recommendations** topic.

'Revisions' in STEP are historical versions of objects. A revision represents a historical 'snapshot' of an object. Major data objects in STEP (i.e., products, assets, classifications, entities, etc.) are under revision control. This means that previous versions of an object can be viewed, compared, and revived.

For more information, refer to the **Revisions** topic and the **Generating Revisions** topic in the **System Setup** documentation.

Storing many revisions can have a negative impact on system performance. Use the following recommendations to keep the number of revisions under control and to remove unnecessary revisions.

Optimizing performance in revision control involves the following steps and each is defined below:

- Setting the revision threshold
- Maintaining object revisions
- Maintaining integration endpoint revisions

Setting the Revision Threshold

By default, revisions on objects are created when a user makes a change to the object and the number of hours set in the threshold is exceeded (starting when the object was first touched after the last revision was made).

This is particularly useful in cases where an object is primarily maintained by a single user and would only be made by that user choosing to do so manually. The threshold functionality ensures that changes are recorded, without creating an excessive number of revisions, which can have a negative effect on system performance.

In workbench, the revision threshold parameter is in the System Setup 'Users & Groups' node under the Revisability Settings flipper. For details about this threshold, refer to the **Revisability Settings** topic within the **System Setup** documentation.

To avoid excessive creation of the revisions, keep the default setting (168 hours) for the threshold parameter.

Maintaining Object Revisions

As the revision history grows, it impacts the time it takes to retrieve the front revision of the object, including its name, attribute values, and references. This may have a negative effect on the system performance.

To reduce poor system performance due to excessive revisions, define a revision policy as follows:

1. Manually delete revisions that are no longer needed. This creates a baseline for ongoing maintenance.
2. Automatically purge old revisions via one or more event processors to keep the number of revisions under control (i.e., schedule the event process to run monthly, and to delete revisions older than one month).

For details about purging old revisions manually, refer to the **Maintaining Revisions** topic in the **System Setup** documentation.

For details about purging old revisions automatically, refer to the **Creating an Event Processor** topic and the **Revision Management Processing Plugin Parameters and Triggers** topic in the **System Setup** documentation.

Maintaining Integration Endpoints Revisions

Integration endpoints (IIEP and OIEP) with revisions in the thousands (or more) can have a negative impact on system performance and IEPs open very slowly. Typically, this is a result of the IEP poller being started by a different user than the one configured in the IEP.

Identify IEPs with different users

1. Run the Administration Portal Healthcheck for pollers started by a different user than the one configured in the IEP to identify the scenario.
1. From the Start Page, click the **STEP System Administration** button and supply the login credentials.
2. On the **Healthcheck** tab, open the Data Error section and select the 'Pollers started by a different user than the one configured in the IEP' test.
3. Click the Run Selected Tests button (

).
4. When the 'Last Run' column for the test shows today's date / time, review the results in the Detected Problems area at the bottom of the screen.

For more information, refer to the **Healthcheck** topic in the **Administration Portal** documentation.

Update IEPs to resolve different users

1. Log in to the workbench and open an IEP that was reported in the Healthcheck above.
2. Identify the user on the IEP and, if necessary, log in to workbench again as that user.
3. Disable the IEP.
4. Enable the IEP again to set the same user under Revisions (on the Status tab) and the User parameter.

Purge old IEP revisions

After verifying the excessive number of revisions will no longer be created, remove the old IEP revisions that exist in the database.

Use the 'Purge old revisions' option to delete revisions for 'Inbound Integration Endpoint Type' and 'Outbound Integration Endpoint Type' as defined in the **Generating Revisions** topic in the **System Setup** documentation.

Important: Purging a large number of revisions on IEPs can require an index rebuild in the database. Contact your database administrator or contact Stibo Systems.

Scheduled Process Recommendations

This is one of the data gathering methodologies and recommendations for functional performance improvement. The full list is defined in the **Performance Recommendations** topic.

Scheduled processes can negatively influence system performance if, for example:

- the bulk update, business rule, or search query to fill the collection is not optimized.
- the data profile processes are no longer required by the business.
- the exporter process returns warning or errors.
- the structured translation using Excel or XML checks for translation export requests every few minutes.

Analyzing scheduled processes and resolving the identified issues can potentially improve performance.

For details on scheduling a background process, refer to the **Creating a Scheduled Background Process** topic in the **System Setup** documentation.

Recommendations

Resolve degraded performance with these improvements:

- If an automated schedule is necessary, verify the frequency is warranted. If possible, use the 'Later and Repeat' option to set a recurring schedule to daily (or less frequently), rather than hourly or by minutes.
- Balance the load on the STEP system by coordinating the scheduled processes. If possible, schedule long-running scheduled processes to run sequentially and during system down time.
- Delete scheduled processes without a business value.
- Resolve errors and warnings in scheduled processes (both are visible in the '# of errors' column and also in the log file by searching for the text 'Caught exception handling bg-process').

Optimizing performance in scheduled processes can also include the following:

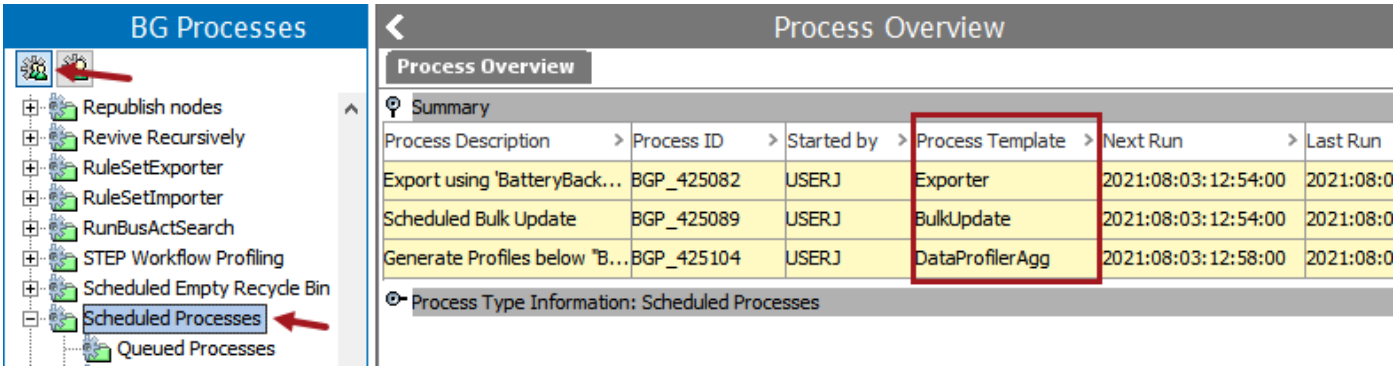
- Structured Translation - Excel or XML, as defined in the **Scheduling a Data Translation** topic of the **Translations** documentation.
- Improve BGP performance using parallel and multithreading properties, as defined in the **Background Processes and Queues** topic in the **System Setup** documentation.
- Consider In-Memory for scheduled BGPs since it can improve performance particularly with long-running search queries. For more information, refer to the **In-Memory Database Component for STEP** topic in the **Resource Materials** section of online help.

Create an Excel Overview of Scheduled Processes

Review the scheduled processes on your system.

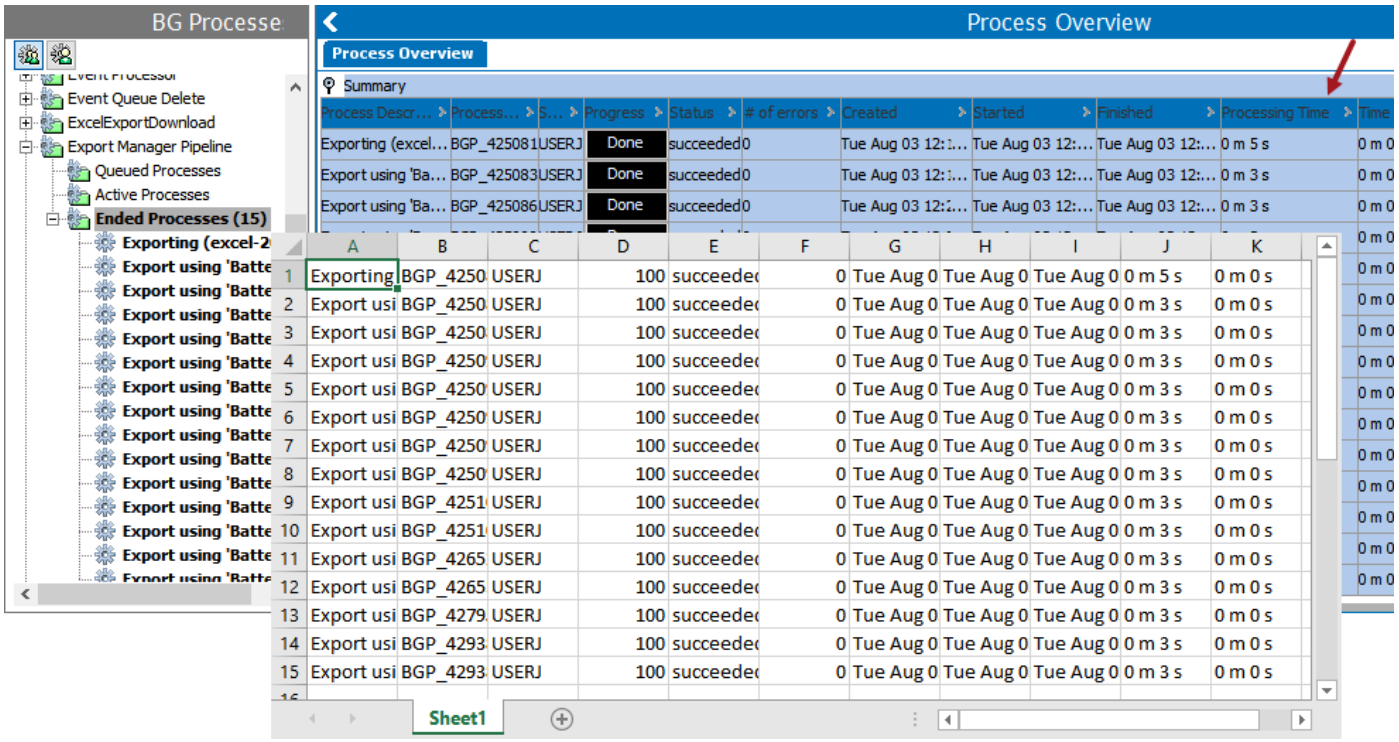
1. In workbench, on the BG Processes tab, click the all users button ( ) to display processes for all users.

2. Expand the **Scheduled Processes** node to view the scheduled processes.



3. For each Process Template entry, copy the scheduled processes summary and paste to Excel:

   - Click the node type on the list of scheduled processes (in this example, the Exporter queue is used by the Export Manager Pipeline, so click the Export Manager Pipeline node, and then repeat for the others) and click the Ended Processes folder.
   - Select all columns in the Process Overview tab in the Summary flipper (including the **Processing Time** column), right-click, and click **Copy** (or use Ctrl+C) to copy the data.
   - Paste the copied process data to Excel and add a column header row.



- Repeat for each type of template.

4. For each Process Template type, follow the steps below to analyze and optimize the scheduled processes.

Analyzing and Optimizing Scheduled Processes

Review the scheduled processes for your system.

1. In the Excel file, review the total number of scheduled processes.
   - Performance issues are likely when 25 or more processes are scheduled to run more often than daily.
2. In the Excel file, review the scheduled processes and verify each is valid for current business requirements.
   - Delete unnecessary scheduled processes as defined in the **Deleting a Scheduled Background Process** topic in **System Setup** documentation.
3. In the Excel file, sort data by the **# of errors** column data to address the scheduled processes that are running with errors and warnings.
   - Resolve the problems identified.
4. In the Excel file, sort data by the **Processing Time** column data to identify long-running scheduled processes.
   - Determine if the scheduled processes are still necessary. Delete processes that no longer provide the intended business value.
   - Continue with the **Analyzing and Optimizing Long-Running Processes** section below.

Analyzing and Optimizing Long-Running Processes

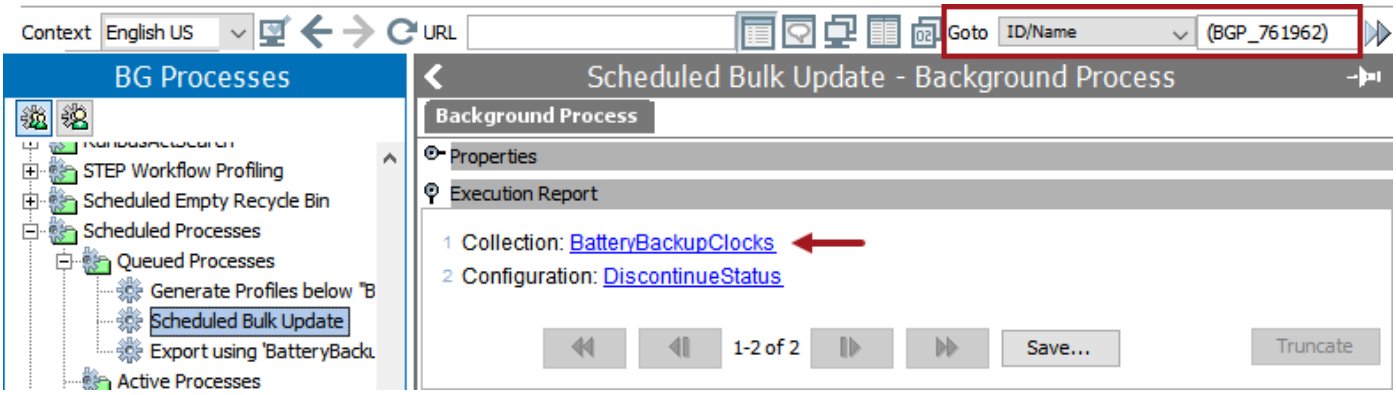
Since repeated searches are typically used in scheduled processes, it is important to analyze the run time of the scheduled processes to identify the searches that might be improved.

Note: The database Oracle KODO cache default limit is 10,000. For example, a hierarchy with more than 10,000 children, a product with more than 10,000 references, a list of value with more than 10,000 values, etc. Storing very large data relations can have great negative performance impacts, since there is a high likelihood that a large number of the related objects are no longer in the cache. For more information, refer to the **Base Setup Recommendations** topic.

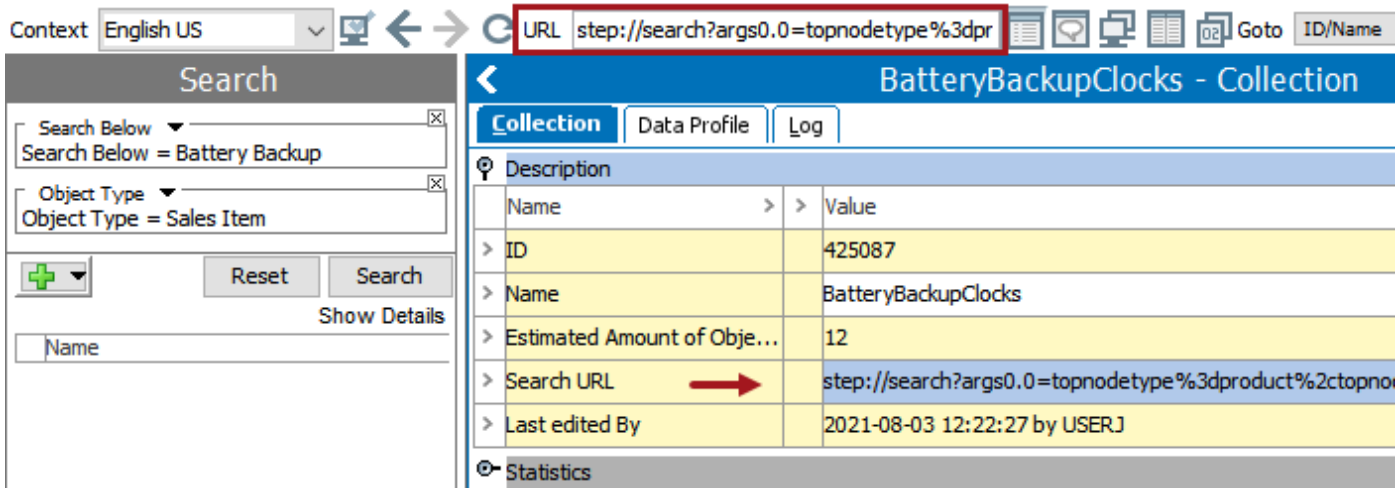
Business rules can also be used in scheduled bulk update processes. Analyze the run time of the scheduled processes and verify which business rules in bulk updates might be subject to improvements.

Review the scheduled long-running processes for your system.

1. In the Excel file, copy the BGP ID for a Scheduled Bulk Update process.
2. In workbench, search for the BGP ID, view the Background Process tab Execution Report flipper, and click the **Collection** link.

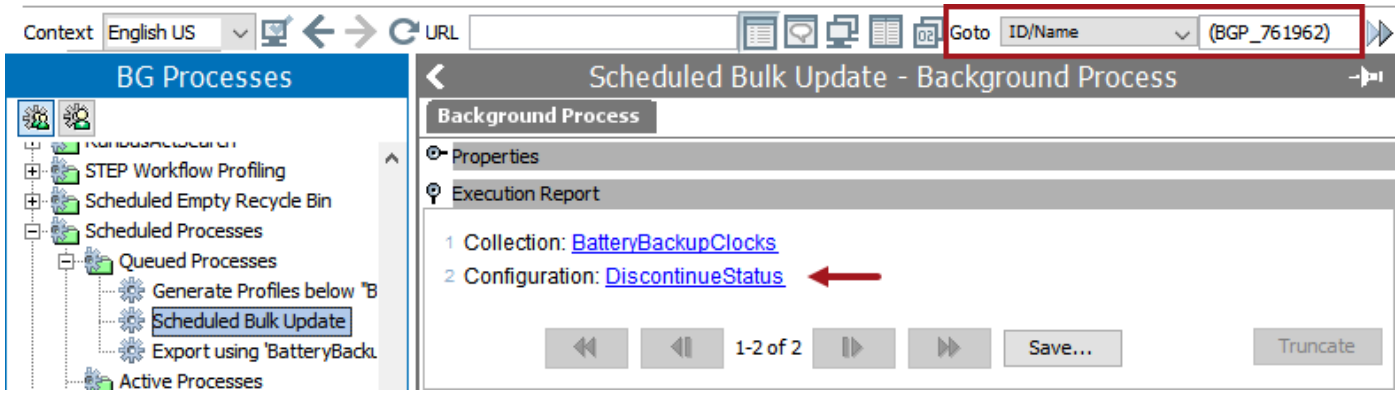


3. If the collection includes a Search URL, copy the **Search URL** parameter value, paste it into the URL parameter, and press **Enter**.

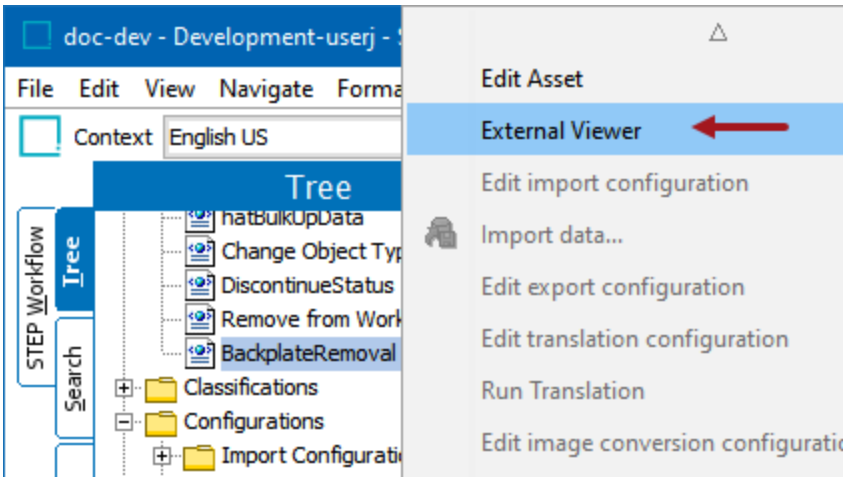


- Click the **Search** button and evaluate the length of time the search takes to respond. For a long-running search
  - If many objects are being searched, use as specific search criteria as possible.
  - Review the **Search Elements to Use** topic or the **Search Elements to Avoid** topic and implement the recommendations.

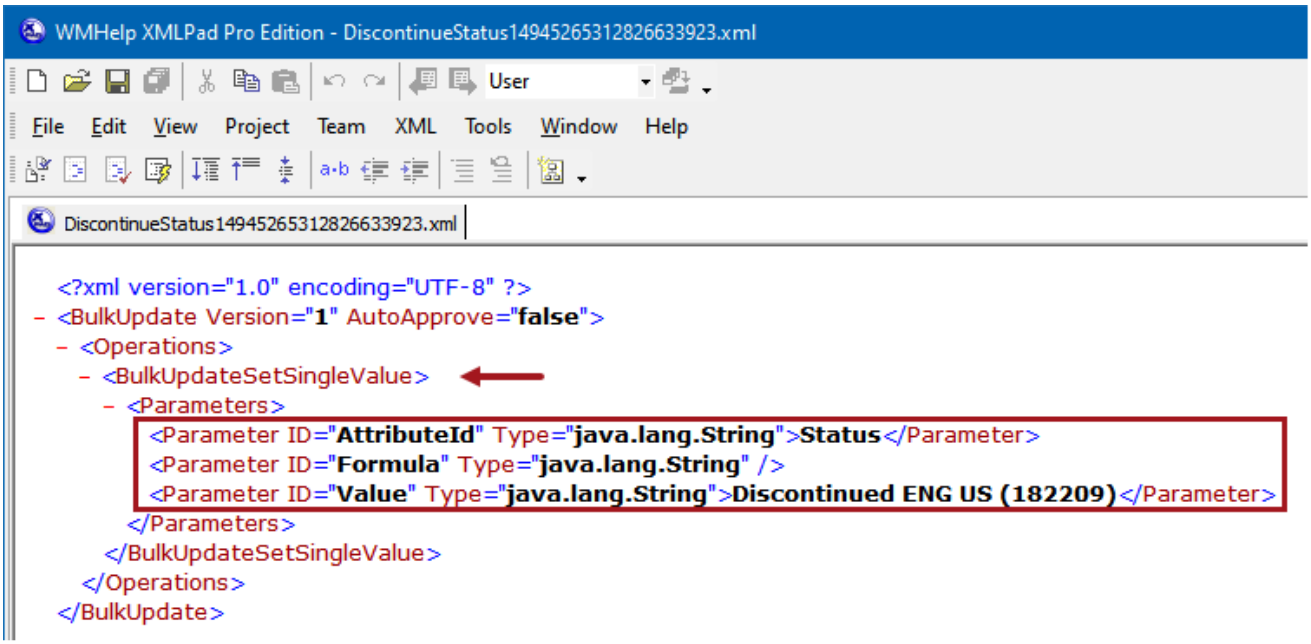
4. Search for the background process again and on the Execution Report flipper click the **Configuration** link.

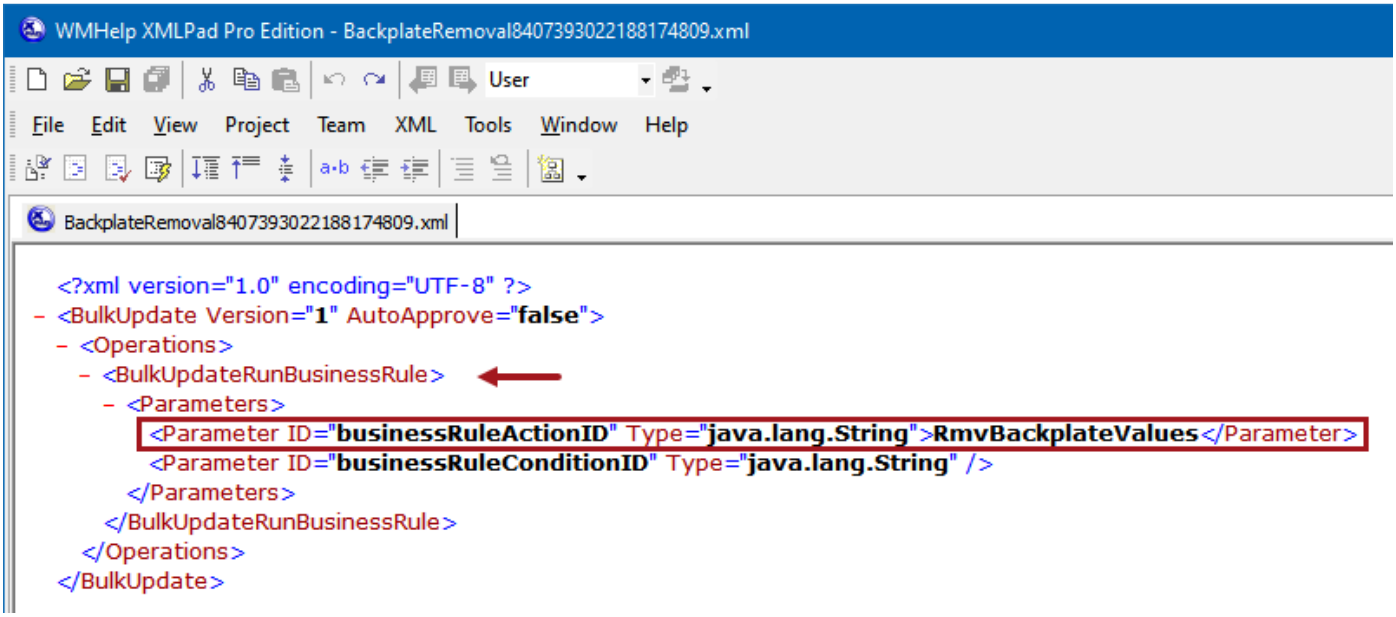


5. Click the Tree tab to show the highlighted configuration, right-click and select **External Viewer**.



6. In the XML viewer displayed, observe the configuration details of the Bulk Update operation. In the examples below, the Set Single Value operation and the Run Business Rule operation are shown below the Operations tag.





- For a **BulkUpdateRunBusinessRule** operation, in workbench, search for the business rule(s) by ID (for example, 'RmvBackplateValues' in the previous image). A business condition can also be included. Open the business rule(s) and analyze the statistics. For details, refer to the **Business Rule Recommendations** topic and the **Business Rule Analysis** topic.

Search Elements to Use

The following list includes search elements known to minimize impact on system performance. This list can be used to troubleshoot existing searches, and can also be reviewed prior to creating new searches to prevent performance problems.

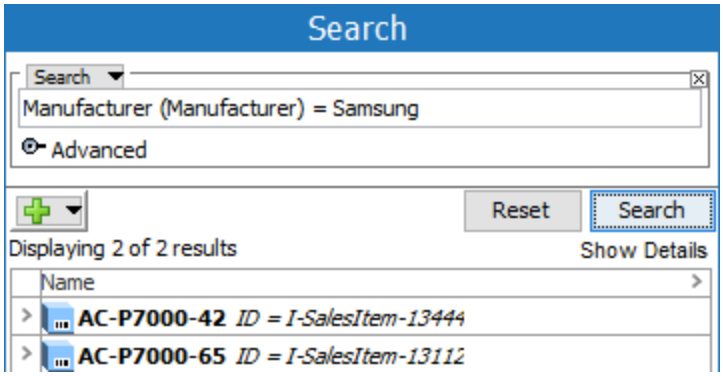
For general information on searches, refer to the **Search** topic in the **Getting Started** documentation.

Basic Searches

When possible, use the following syntax to achieve the best search performance. Searching for a value without specifying the attribute, ID, or name searches for the value in all attribute values, IDs, and names.

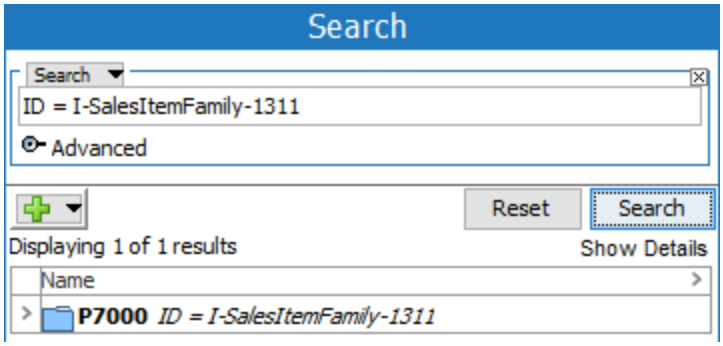
Attributes

Search using 'Attribute-ID = value' to search in values of this one specified attribute only. For example, search for the 'Samsung' value only in the Manufacturer attribute, rather than in all attributes.



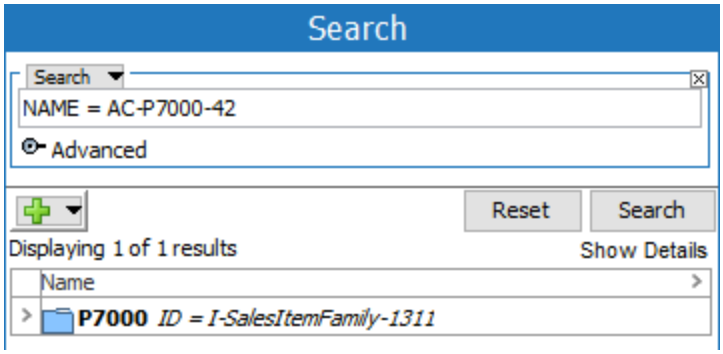
Objects by ID

Search using 'ID = value' to search for objects by looking in the ID parameter only. For example, search for the 'I-SalesItemFamily-1311' value only in the ID field.



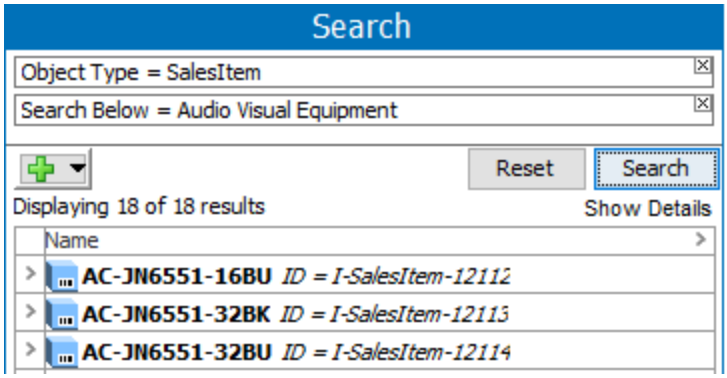
Objects by Name

Search using 'Name = value' to search for objects by looking in the Name parameter only. For example, search for the 'AC-P7000-42' value only in the Name field.



Use Specific Search Criteria

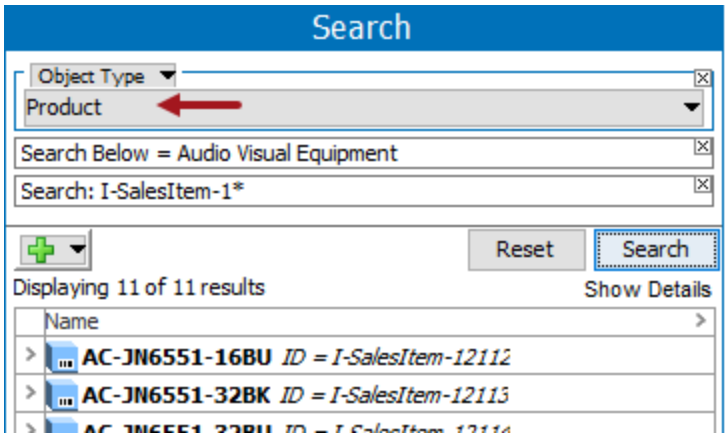
When searching a database with many objects, use as specific a search criteria as possible. For example, specifying the object type to search for in a specific hierarchy (as shown below) is faster than searching for all object types in all hierarchies.



Use Object Super Types

The search function performs slightly better when specifying the object super type (such as product, entity, asset, classification, and so on) instead of using the specific object type below the main object types. For more information, refer to the **Object Super Types** topic in the **Getting Started** documentation.

In the example shown below, the product super object type is used, instead of SalesItem (which is an object type below the product super type).

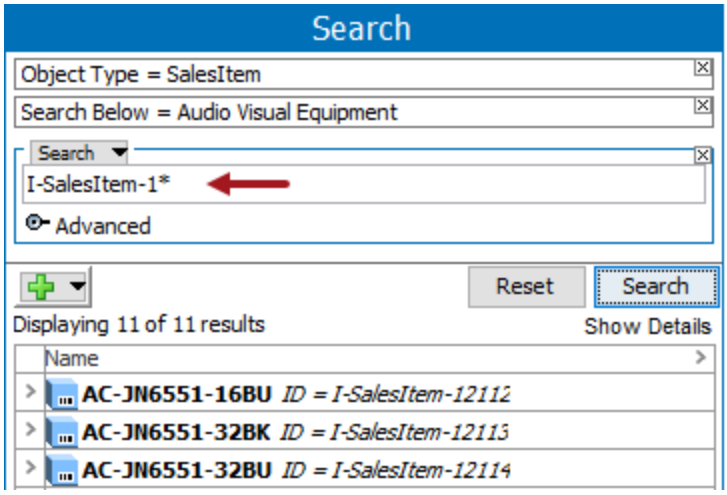


Searches with Wildcards

Specify as many characters as possible before using a wildcard (*) to optimize search performance results.

Important: Specifying a wildcard as first character prevents the use of the database indexing query, which means the search must traverse all objects in the database. This is a dramatic performance impact.

For example, use a wildcard (*) after more than three characters, as shown below.



Optimize Search Below with a Configuration Property

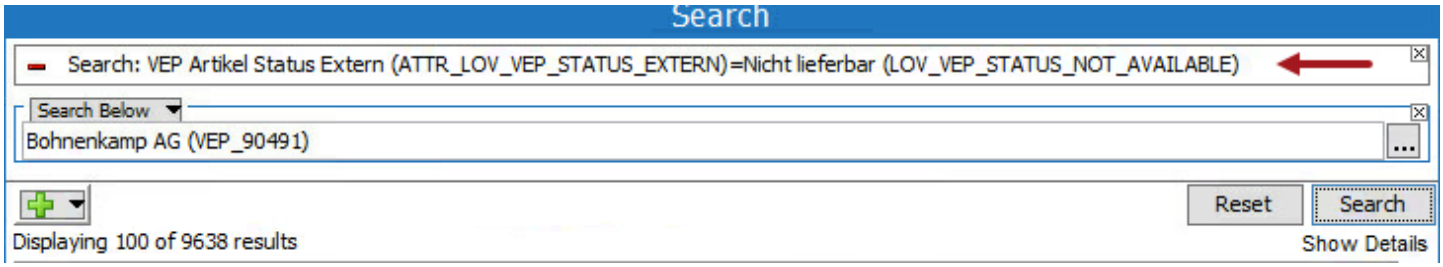
In some cases, a bad execution plan is implemented by Oracle when using the 'Search Below' criteria. The following case-sensitive entry in the sharedconfig.properties file will improve the search performance:

```
Domain.BelowCriteria.UseRecursiveWith=true
```

Setting this property to 'true' forces Oracle to choose another (and in some cases better) exclusion plan, and may have a positive effect on search performance.

Optimize Combined Search Below and Value Search

Combining the 'Search Below' criteria with a value search criteria may not perform well. However, this type of search can be optimized by first doing a value exclusion search, and then combining the result with the 'Search Below' criteria. For example, review the image below:



In-Memory

Consider In-Memory to improve import performance, as defined in the **In-Memory Database Component for STEP** topic in the **Resource Materials** section of online help.

Search Elements to Avoid

The following list includes import elements known to degrade system performance. This list can be used to troubleshoot existing imports, and can also be reviewed prior to creating new imports to prevent performance problems.

For general information on searches, refer to the **Search** topic in the **Getting Started** documentation.

Avoid Inherited Values and Regular Expressions

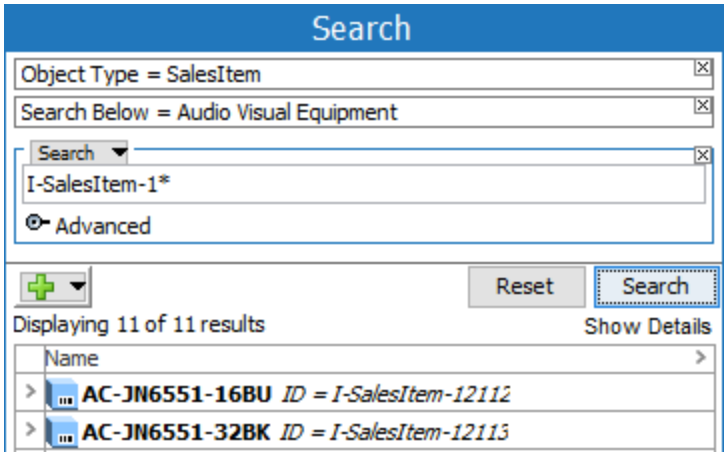
Advanced searches in the workbench and Web UI should avoid the following options to achieve the recommended performance:

- 'Include Inherited Values' - only use when required.
- 'Regular Expression' - disables database indexing, only use when required.

Avoid Root Hierarchies in Search Below

The 'Search Below' option should be used with caution and only when necessary to achieve the required result. Specifying many root nodes in Web UI searches is specifically not recommended.

For example, use a granular hierarchy node instead of searching the full primary product hierarchy.



Web UI Configuration Recommendations

This is one of the data gathering methodologies and recommendations for functional performance improvement. The full list is defined in the **Performance Recommendations** topic.

The Web UI Designer is flexible and can configure a Web UI in many ways. However, not all configurations perform equally well. Therefore, it is important to consider performance when configuring a Web UI. A screen with little data and limited functionality will load faster than a screen with lots of data and functionality.

For more information, refer to the **Healthcheck** section of the **Administration Portal** documentation.

The following actions can limit the performance impacts while using Web UI:

- Web UI Component Report
- Use multiple Web UIs
- Use small dedicated Web UI screens
- Avoid using images in multi-select Web UI screens
- Use 'lazy' loading for Web UI screens
- Use type ahead for LOVs in Web UI screens
- Correctly configure status selectors in Web UI
- Privileges in Web UI configuration
- Consider In-Memory for Web UI screens

Web UI Component Report

The Web UI Component Report provides an in-depth look at the components configured for all Web UIs. Of particular interest are those in the end-stages of their lifecycle. Use this report to replace outdated components and update configurations improve the upgrade process. The reported components are those that have been:

- Deleted entirely: will not render
- Withdrawn: once removed cannot be added again
- Superseded: will work as usual, but a new and improved component is available

The Web UI Component Report link is displayed at the top-right of the system Start Page and in the popup that displays when accessing the Web UI designer. You can also access it using the URL http://[ENVIRONMENT]/webui/componentreport (modified with your environment name).

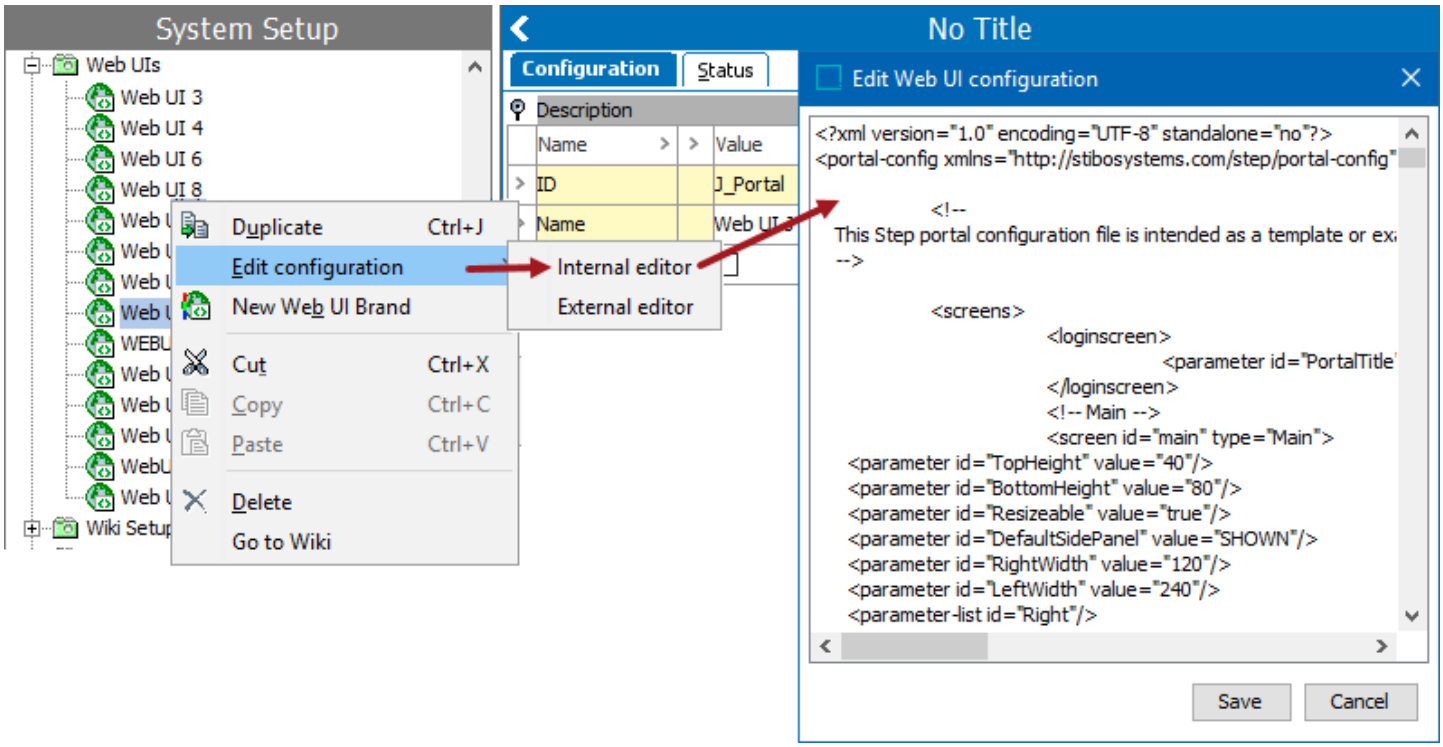
For more information, refer to the **Web UI Component Report** section of the **Web UI Component Basics** topic in the **Web User Interfaces** documentation.

Recommendation

For optimal Web UI performance, remove the superseded, withdrawn, and deleted Web UI components or replace with the newer available Web UI components.

Use Multiple Web UIs

Web UI configurations are stored in System Setup under the Web UIs node. When a user logs into the Web UI and opens the homepage, the corresponding configuration XML file is loaded. If the Web UI configuration XML file is large, then the Web UI homepage takes additional time to load. Edit the Web UI configuration to view the XML, as shown below.



For more information, refer to the **Managing Web UI Configurations** topic in the **Web User Interfaces** documentation.

Recommendation

When the Web UI configuration XML is more than 25,000 lines, split the configuration into separate Web UIs where each is used for a specific purpose.

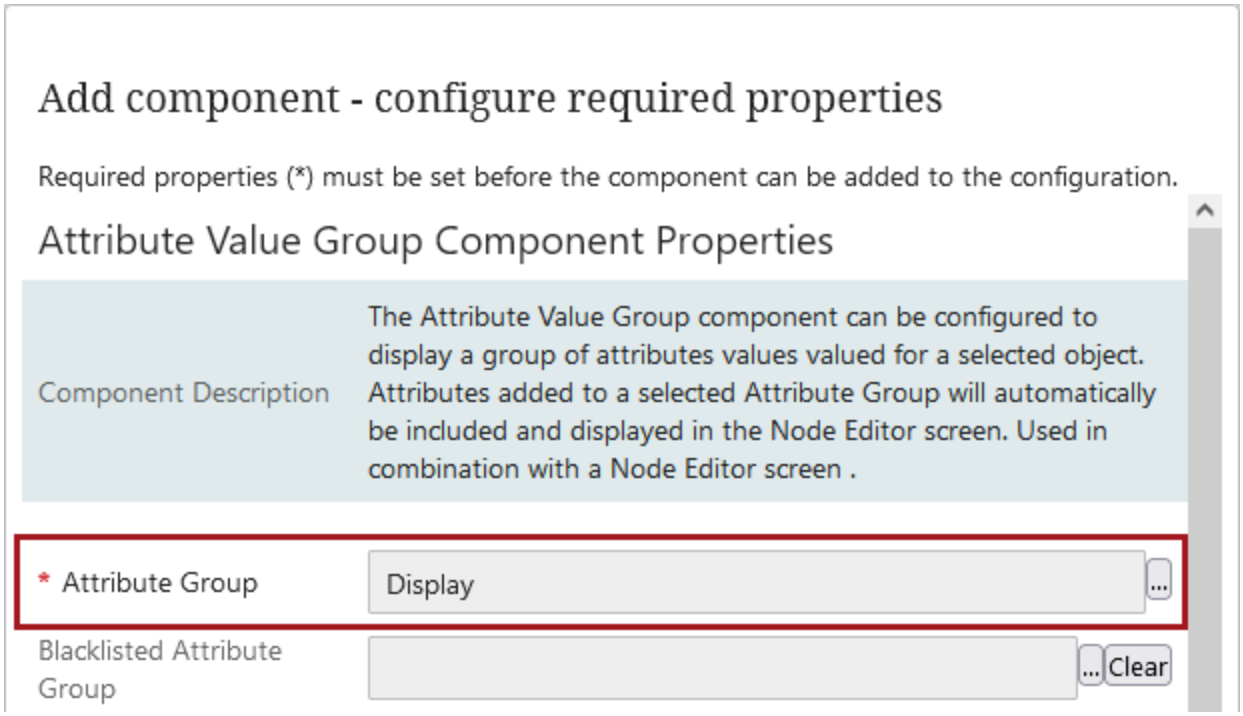
Use Small Dedicated Web UI Screens

A typical Web UI screen fetches all attribute values from the attribute group defined in the Web UI screen. The screen then filters out the attributes based on the validity of the product type and user privileges.

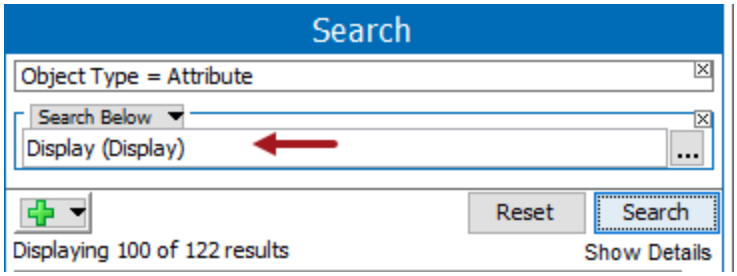
Configuring attribute groups for the Web UI screen which contain a large number of attributes can have a negative impact on the loading time of the screen because it involves:

- fetching all attributes from the attribute groups, and then
- filtering out attributes for display based on the validity of the product type and user privileges.

To reduce the screen load time, first determine which attribute groups are used in the Web UI screen, as shown below.



Then analyze how many attributes each attribute group contains, using the Search Below option.



Recommendation

Avoid using attribute groups with more than 100 attributes. When configured attribute groups contain more than 100 attributes, consider creating attribute groups specifically for Web UI display. Organize the attributes to be displayed on the Web UI screens into these specific attribute groups.

Avoid Using Images in Multi-Select Web UI Screens

A multi-select screen displays a selection of items in a table. The table view also allows for thumbnails of the items to be displayed. However, loading many items with their thumbnails naturally requires fetching these thumbnails from the file system or database, which may have a negative impact on the load time of the screen.

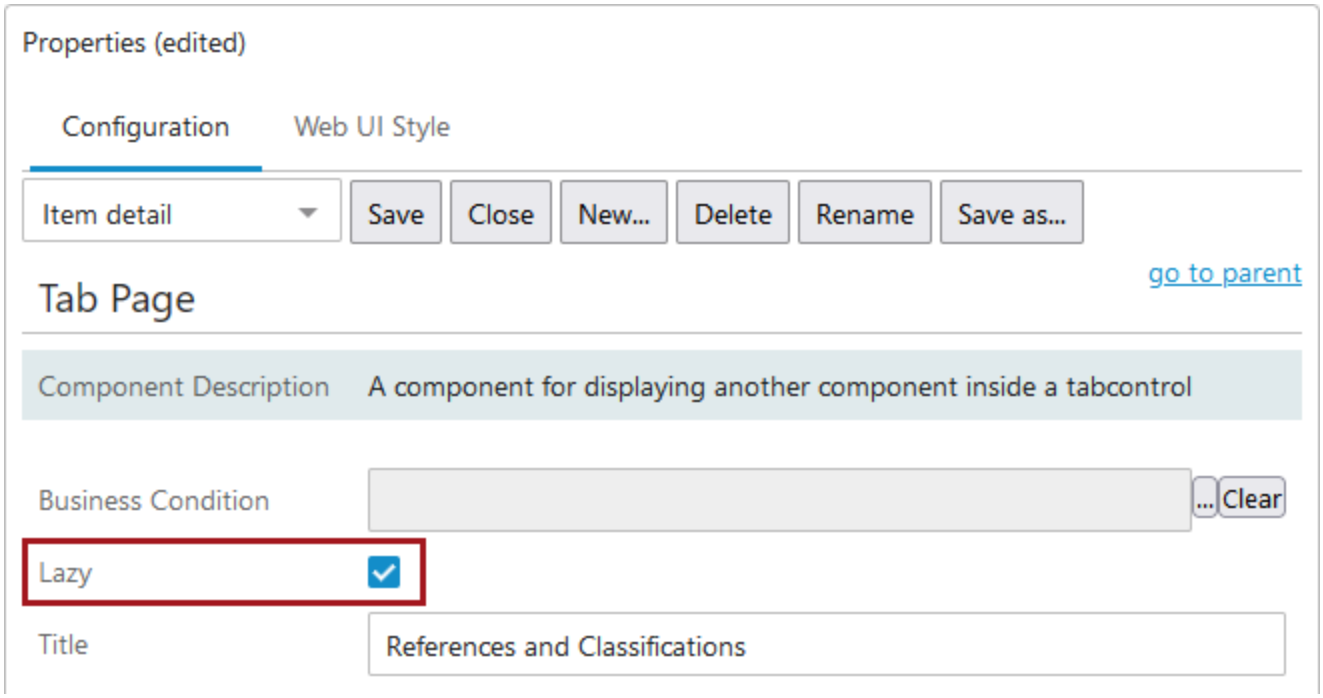
Recommendation

Reconsider displaying thumbnails (or no images at all) in multi-select screens when many items are displayed and load time of the screen is considered slow.

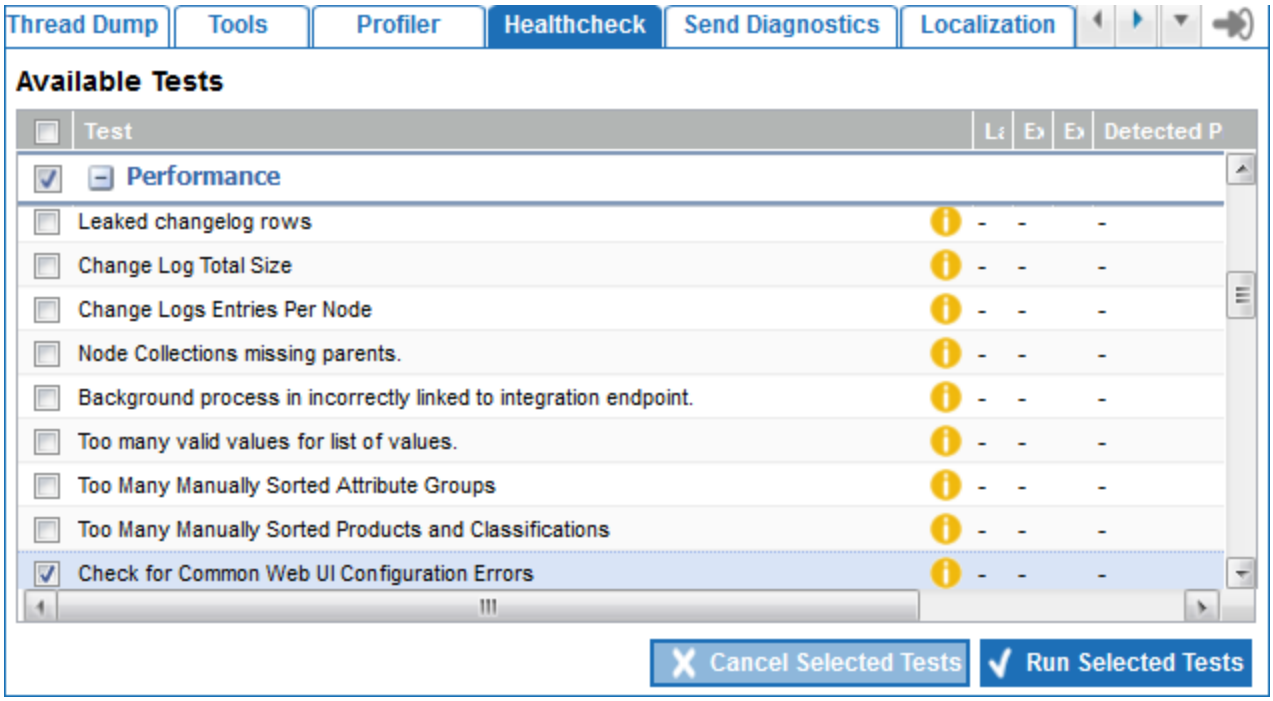
Use 'Lazy' Loading for Web UI Screens

For Web UI tasks where the user needs to inspect many aspects of an item, the recommendation is to split the information out into multiple tabs. However, by default, the Web UI loads the screen including all tabs, which may result in a slow loading time for the Web UI screen.

In Web UI designer, the 'Lazy' parameter allows you to apply 'lazy loading' on these Tab Page screens. When enabled, components are 'lazy loaded' and rendered only when a tab is displayed, which can reduce screen load time.



Use the Performance section of the STEP Health Check in the Admin Portal to analyze which Web UI screens can be lazy loaded by running the 'Check for Common Web UI Configuration Errors' healthcheck shown below.



For more information, refer to the **Healthcheck** topic in the **Administration Portal** documentation.

Recommendation

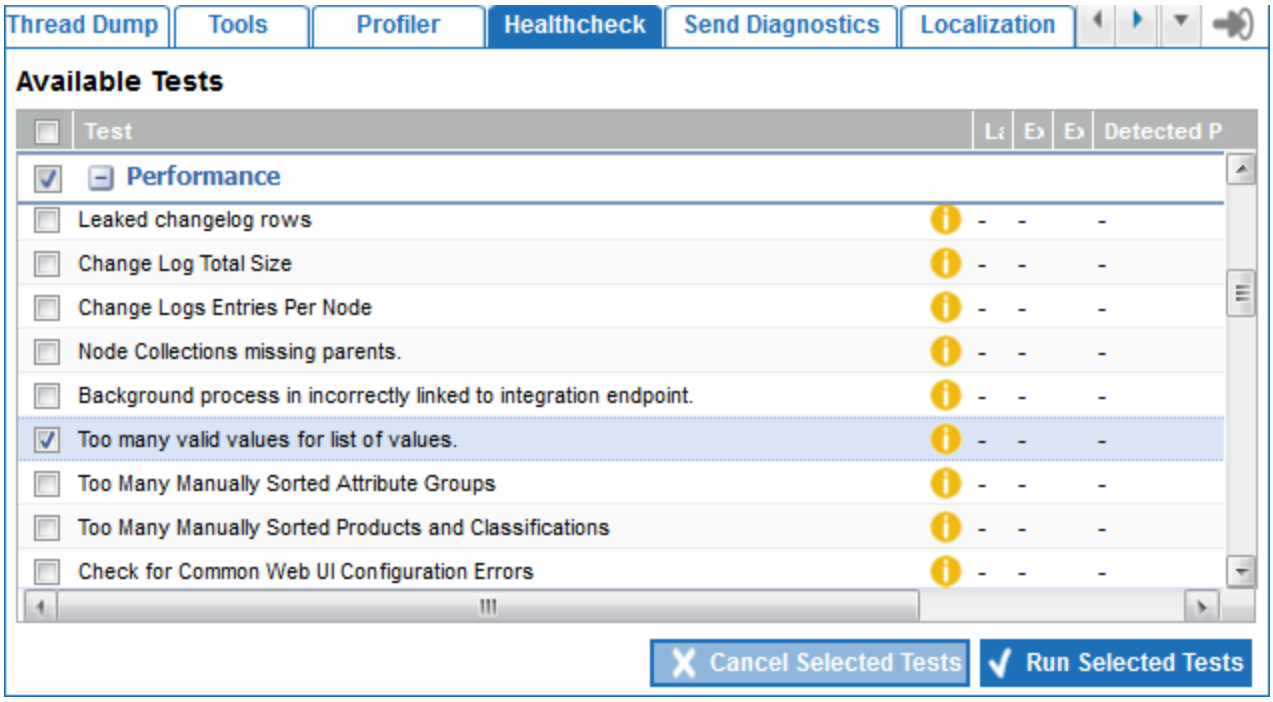
When possible, lazy load data on tabs.

Use Type Ahead for LOVs in Web UI Screens

Web UI screens using attributes with a large list of values (LOVs), the load time of the Web UI screen can become slower. The following LOV type-ahead sharedconfig.properties are available to optimize the Web UI screen load time:

- **Portal.ValueGroup.LOV.ForceTypeahead** forces type-ahead for all LOVs in the Web UI. Set this property when there are attributes used in Web UI screens and many of the LOVs used are large.
- **Portal.ValueGroup.LOV.ForceTypeahead.Exclude** forces type-ahead for certain LOVs in the Web UI. Certain specific LOVs can be configured not to be forced to user type-ahead functionality. Use this for the LOVs with small number of values and for the LOVs which are required to not be typed-ahead.

Analyze which LOVs have more than 5,000 values by running the 'Too many valid values for list of values' healthcheck. Also review Web UI screens to find where large LOVs with fewer than 5,000 values are being used.



For more information, refer to the **Healthcheck** topic in the **Administration Portal** documentation.

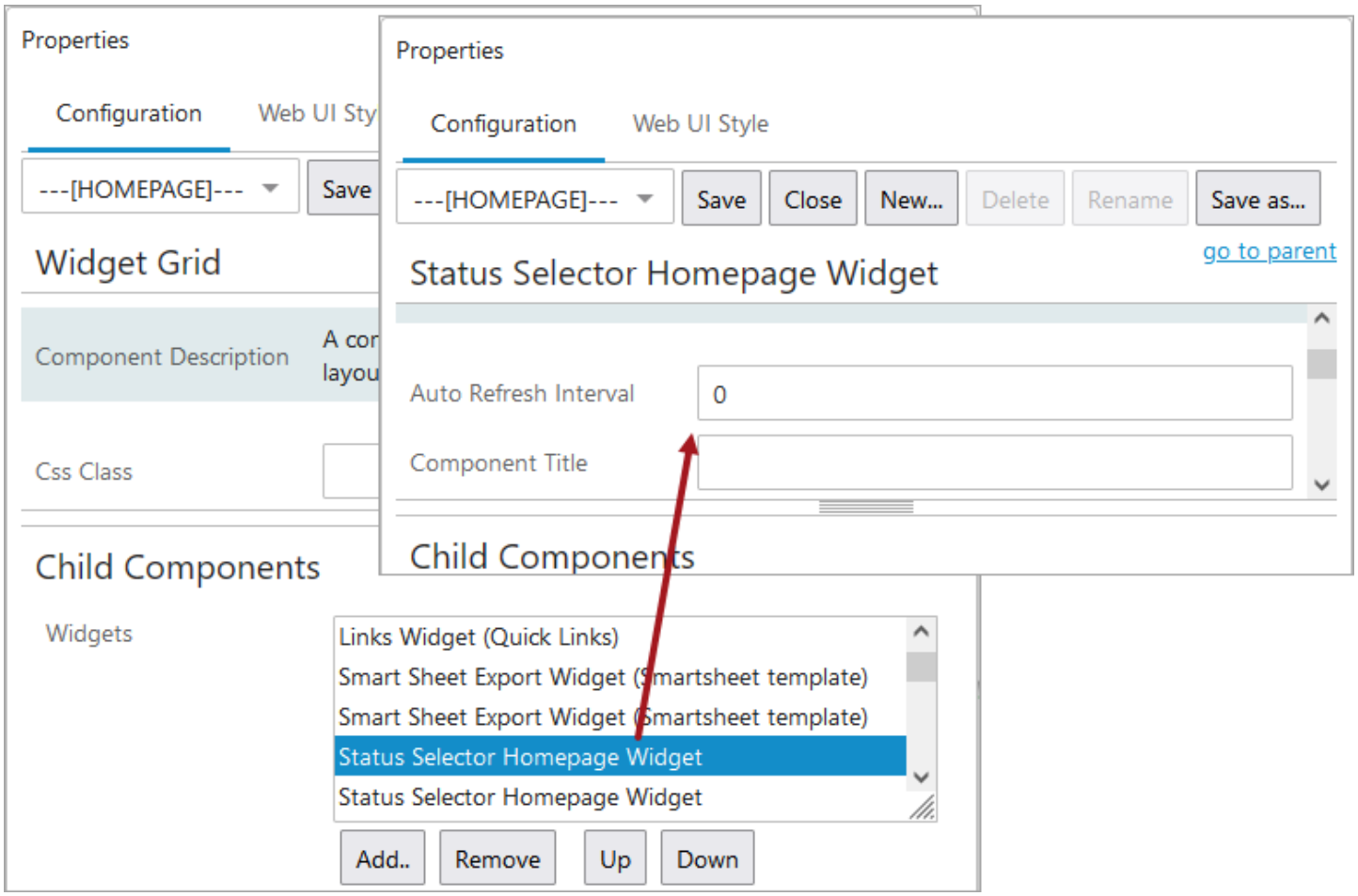
Recommendation

Set up typeahead for large LOVs.

Correctly Configure Status Selectors in Web UI

The workflow Status Selectors Homepage Widget is configured to poll for updates in the background. The statuses are updated whenever the user submits an item. However, if there are many status selectors and/or the update polling is set very fast, the Web UI Homepage screen may suffer from poor performance.

In design mode, double-click the Status Selector Homepage Widget to display the 'Auto Refresh Interval' parameter. The recommended value is 60 seconds. Disable the refresh interval by setting the value to 0.



For more information, refer to the **Status Selector Homepage Widget** topic in the **Web User Interfaces** documentation.

Recommendations

- Limit the number of status selectors, generally no more than 25 status selectors on the Homepage screen.

  Download the Web UI XML to review status selectors used in a Web UI and search for:

  - **StatusSelectorHomepageWidget** to find the status selectors on the homepage.
  - **StatusSelectorWidget** to find the status selector on another page.

- Do not set the Status Selector Homepage Widget 'Auto Refresh Interval' parameter to less than 60 seconds.

  Download the Web UI XML and search for **AutoRefreshInterval** to find auto refresh intervals on status selectors.

- Be aware that including many nodes in a workflow being calculated in the Status Selector may increase loading times.
- Reduce load times. On the Web UI designer for the Status Selector Homepage Widget properties screen, Advanced section, check the 'Use Content Indicator' parameter. This shows a button to manually load the count, instead of displaying the exact count automatically.

  Download the Web UI XML and search for **UseContentIndicator** to find the content indicator used on status selectors.

Privileges in Web UI configuration

Although privilege restrictions can be set in the Web UI configuration itself, excessive privilege checking in the Web UI XML configuration can degrade performance.

Recommendations

Download the Web UI XML and search for **restrict=** and analyze if these user restrictions are necessary.

Consider In-Memory for Web UI Screens

In-Memory can improve performance of the Web UI screens. In-Memory makes the Web UI more responsive in general, but especially when the Web UI screens include multi-object display, complex data models, and/or data with a long history of writes.

For more information, refer to the **In-Memory Database Component for STEP** section of the **Resource Materials** section of online help.

Performance Assessment

The goal of the performance assessment is to analyze the system from a performance perspective, identify the pain points, and give actionable recommendations to optimize performance.

Performance testing is highly recommended and is expected to be part of the implementation process and release. You should perform the testing, or hire a recommended third-party who is skilled at the process.

STEP provides the means to perform some performance testing, such as measuring import and export execution time (background process times), and measuring the execution of business rules.

Prerequisites

1. Run each of the healthchecks available from the Administration Portal, as defined in the **Healthcheck** topic of the **Administration Portal** documentation.
2. Define the goal of the performance test including a good test result and a bad test result. Typically, the absolute results of the performance tests do not represent realistic behaviors, but performance test results should be compared with previous performance test results to interpret the STEP system behavior over time.

Assessing Performance

Review the following topics for information on performance:

- Performance Assessment Process
- Non-Functional Performance Requirements
- Large Scale Performance Testing

Troubleshooting Performance Issues

Use the following topics to assist in resolving issues:

- Performance Troubleshooting
- Troubleshooting Checklists

Optimizing Performance

For information on improving performance, refer to the following topics:

- (Functional) Performance Recommendations
- Technical Performance Recommendations

Performance Assessment Process

The following process is recommended to analyze STEP from a performance perspective:

1. Contact Stibo Systems to begin the process for executing a performance assessment.
2. Complete the provided template for identifying the main performance problems, non-functional performance requirements, and performance use cases.
3. Schedule an online call (approximately 2 hours) to discuss the completed template and establish performance goals.
4. Your Stibo Systems representative uses the template to reproduce the performance problems off-site and performs the analysis. This step can take three (3) days.
5. Your Stibo Systems representative schedules a 2-day on-site visit which includes:
   - Presenting the initial performance results of the off-site analysis.
   - Determining the most important recommendations with you.
   - Implementing the most important recommendations as feasible.
   - Executing the defined performance use cases again and discussing the improvements.
   - Determining the next steps.
6. Your Stibo Systems representative writes the conclusions of the performance assessment. This can take one (1) day.
7. Your Stibo Systems representative schedules an online call (approximately 2 hours) to hand over the final assessment results document.

Non-Functional Performance Requirements

Defining non-functional requirements allow you to indicate what is expected from a STEP performance perspective.

Each non-functional performance requirement must be supported by one or more use cases. Each use case will include step-by-step instructions to verify the non-functional requirement.

For example, the following tables shows a requirement and the use case to verify it.

Performance Requirement	Use Case
Import 'Supplier A (SupplierA)' via Excel; complete within 3 hours	1. Log in to workbench 2. Run import using 'Import Supplier A' configuration and Excel sheet 'Import Supplier A' 3. Wait for import to complete

Large Scale Performance Testing

Large scale performance testing can be performed by you or your Stibo Systems representative using NeoLoad. For more information, search the web.

Prerequisite

Before running large-scale performance testing, run each of the healthchecks available from the Administration Portal, as defined in the **Healthcheck** topic of the **Administration Portal** documentation.

Test Guidelines

Large-scale performance testing must respect the following basic rules for a realistic test:

1. A good representative and realistic data set must be in place on a system, observing all hardware requirements specified by Stibo Systems.
2. Each time a performance test has run the database has to be reset, which allows every test to start from the same point. For example, manually deleting what is thought to be the test data is not sufficient; there must be a restore from backup which holds the agreed test data.
3. Simulating the behavior of multiple users working in parallel will require that the data being worked is parameterized. For example, it is not a good idea to start 1,000 users working on the same product.
4. If a test robot is used to perform the user scenarios, delays between consecutive user actions must reflect realistic user behavior. For example, a human being does not make multiple mouse-clicks per millisecond.
5. Consider carefully how to start the test. For example, it is not advised to start 1,000 users within the same millisecond in bulk.
6. Running strongly parallel bulk updates or imports requires careful planning so that the test does not end up in massive updates of the same resources leading to massive and not very realistic locking in the database, leading to terrible performance timings.
7. Gather relevant monitoring data from the test. At a minimum, have a complete copy of the STEP 'diag' folder for every test run. It is useful to collect Oracle data statspack/awr and OS information sar/nmon. It is a good idea to keep this data at least until a few more tests have been completed.

Performance Troubleshooting

Use the methods defined in the following table to troubleshoot performance issues. When all data is gathered, include the information on a Jira ticket assigned to R&D.

Performance Test	Windows	Linux / Solaris
IO usage	Process Explorer	sar - d sar - d - f /var/adm/sa/sa dd *(where dd is the day of month)* iostat - zdxnk 2
CPU usage	Process Explorer Task Manager	sar - u sar - f /var/adm/sa/sa dd *(where dd is the day of month)* top, mpstat - P - ALL 2
Memory usage	Process Explorer	v mstat sar - r - f /var/adm/sa/sa dd *(where dd is the day of month)*
Ping	ping dbserver ping – l 8000 dbserver ping – l 32000 dbserver	ping – c5 dbserver ping – c5 – s8000 dbserver ping – c5 – s32000 dbserver

Latency Requirements

- Max 0.2ms with a 64 bytes packet size
- Max 1.0ms with a 32K packet size

Additional latency information is included in the **Server Network Latency** topic.

Identify the Problem Layer

1. From the Start Page, click the **STEP System Administration** button and supply the login credentials.
2. On the **Profiler** tab, set the **From** and **To** date / time parameters to define when the performance issues were noticed.
3. Click the **Generate** button to load the data.
4. In the results section, expand the service methods to determine the issue:
   - 'com.stibo.customer' indicates that Custom Solutions should be involved.
   - a long-running SQL statement indicates that it is waiting for the Oracle database. Refer to the **Database Server** section below.
   - 'com.stibo. ddsconnector' or 'com.stibo.idsconnector' indicates it is waiting for the DTP server. Refer to the **DTP Server** section below.
   - a problem that occurs for only one specific user or group of users indicates that the problem is related to the user or workstation. Refer to the **User Interface / Client Application** section below.
   - If none of the previous scenarios fit your issue, consider that the problem might be on the application server. Refer to the **Application Server** section below.

Database Server

After reviewing the profiling data (as defined above), when the issue is a long-running SQL statement, follow these steps.

1. From the Start Page, click the **STEP System Administration** button and supply the login credentials.
2. On the **Activity** tab, set the Duration and Date / Time parameters according to when the performance issues were noticed.
3. Click the **Fetch data** button to load the data.
4. In the **Details** section, on the **SQL** tab, identify the same SQL statement with the same duration.
5. Attach the SQL statement, including the execution plan, to the Jira issue.

For more information, refer to the **Database Long-Lasting SQL Queries** topic.

Additional Steps

With the DB ToolBox (typically saved in the '/opt/stibo/step/admin/app-server-toolbox/' folder), you may be able to access the additional information outlined below.

- Check if the system waits for locks in the database.
- Check the CPU usage of the database server.
- Check the I/O usage on the database server.
- Examine the alert log for errors, as defined in the **Database Server Alert Log** topic.

Record this information using the **Technical Support Tasks** table in the **Troubleshooting Checklists** topic.

DTP Server

Perform the following checks:

- Check that the configuration for QuarkXPress Server / Adobe InDesignServer is using asset push. For more information, refer to the **Asset Push** topic in the **Digital Assets** documentation.
- Check the logging configuration. The debug level should only be used when required as it will have a negative performance impact. For more information, refer to the **Server Log File Settings** topic.
- Compare the number of DTP server renders with license, hardware, and user load.

In case of emergency, use the restart action command: restart dtp-servers.

Services representing sidecars can be restarted or even the server itself. Services should be checked after a restart.

Record this information using the **Technical Support Tasks** table in the **Troubleshooting Checklists** topic.

Application Server

Perform the following checks:

- Check the sensor warnings and critical errors, as defined in the **Monitoring** topic within the **Administration Portal** documentation.
- Check the memory graph, as defined in the **Activity** topic within the **Administration Portal** documentation.

  If constantly high with only small 'garbage collections'

  - Consider increasing heap memory.
  - Check for heap dumps on the server and put them in a location in garm.stibo.dk (default stibosw log on).
- Check the CPU usage via Task Manager as well as the admin portal, as defined in the **Activity** topic in the **Administration Portal** documentation.

  If CPU usage is high, check if image GraphicsMagick processing (gm.exe, etc.) are starting and stopping. If so, this can be caused by excessive image conversions. Increasing the sizes of image caches can potentially fix this.
- Check the thread graph, as defined in the **Activity** topic in the **Administration Portal** documentation.

  Compare the number of threads to the number of CPU cores in the application server. If the number of threads is higher, processes will queue and waits are experienced.
- Ping the database server using ping command via the **Help Desk Tasks** table.
- Trace route to dbserver to look for unwanted network equipment.
- Check the number of background processes, as defined in the **Analyze Background Processes** topic.
- Examine the IO usage of the application server. If it is high:
  - Check if the memory on the server is used up, i.e., more memory in use than real memory available will make the server swap.

- Check if log files are filling up quickly. If so, check if debug logging is enabled, as defined in the **Server Log File Settings** topic.

Add all of this information to the **Technical Support Tasks** table in the **Troubleshooting Checklists** topic.

User Interface / Client Application

Perform the following checks:

- Examine task manager on the user workstation while reproducing the problem.
  - If the java process has high CPU usage > 90 percent. If so, the problem might be in the UI.
  - Do other processes have a high CPU usage? If so, ask the user to shut down these processes and attempt to reproduce.
  - Does the system have a memory usage beyond the real memory in the system? If so, shutdown the other processes until the usage is below real memory and ask the user to reproduce.
  - Reboot the PC and start only STEP on the PC and attempt to reproduce.
- Check the Network Latency indicator in the lower right corner of the STEPworkbench client. If it is above 125ms, examine the network.
- Open the Java Control Panel and enable ‘Show console’ and reproduce. Attach the console output to the Jira issue.

If these checks and adjustments do not solve the problem, add this information to the **Technical Support Tasks** table in the **Troubleshooting Checklists** topic.

Troubleshooting Checklists

Use the following checklists to verify you have the appropriate information for a Jira ticket.

Help Desk Tasks	Initials	Date	Comments
Is the problem tied to one specific user or a group of users?			
Is the problem reproducible?			
What is the expected duration of the operation?			
What date / time did the problem occur?			
When was the last time the system was patched?			
Can this be reproduced on a test system?			
Is Stibo Systems allowed to reproduce the problem?			

Technical Support Tasks	Initials	Date	Comments
Symptom Clarification			
What are the specific events (user ID, BGP ID, date / time)?			
Every time? Periodic? Rare?			

Technical Support Tasks	Initials	Date	Comments
Database Server			
Waiting for locks?			
Is CPU usage - normal / high?			
Is IO usage - normal / high?			
Are there Alert log errors?			
DPT Server			
Is DPT server using Asset Push?			
Is debug logging enabled?			
Is the number of renders OK?			
Application Server			
Are there sensor warnings or critical errors?			
Is Memory usage normal / high?			
Are heap dumps available?			
Is CPU usage of JVM normal / high?			
Is CPU usage of server normal / high?			
Is the number of threads higher than number of CPU cores?			
What are ping times?			
What is the total number of BGPs?			
Swapping?			
Is excessive debug logging enabled?			
User Interface / Client Application			
Is CPU usage normal/ high?			
Is memory usage too high / swapping?			
What is network latency to application server?			

Patching STEP

The system architecture of the STEP platform is split up into separate components, each of which may access other components through a set of component APIs. This component-based architecture satisfies the otherwise contradictory requirements for longer time between releases and fast introduction of new improvements. Customers can choose to upgrade specific components in order to take advantage of new features and important updates, whilst keeping the core of STEP and other unrelated components unchanged. This approach reduces the risk and workload involved in testing new updates.

When upgrading components individually, the customer may choose not to upgrade to the newest STEP release. If the feature is available in a new component, that component can be installed on its own. If the feature is available as an upgrade to an existing component, that component may be upgraded while keeping other components as they are.

Note: Available component updates are made visible on a STEP system similarly to the way updates are to mobile phone apps, i.e., with release notes detailing the new features and fixes available relative to the current installation, and with instructions on how to perform the update.

Components have separate release cycles limited only by the dependencies introduced when one component uses another component. Each component declares its dependency on other components through principles where a given component version may depend on a specified range of versions of another component.

For additional information on patching STEP components, refer to the following topics within this documentation:

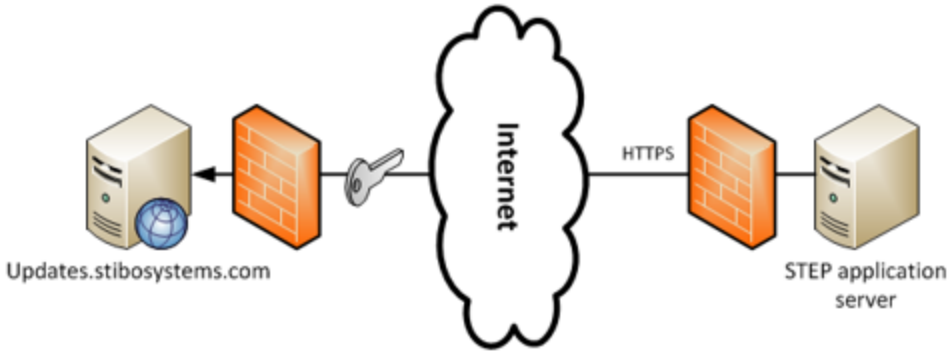
- Patching Methods
- SPOT Program
- STEP Patching Procedures
- Patching Security
- Configuring a Private Updates Mirror

Patching Methods

Patch operations in the STEP system are defined by the specific component(s) being installed / upgraded. These component updates are downloaded either directly from one of the Stibo Systems Global Updates Mirrors (Release Server) or from a private updates mirror at the customer can be used to execute these operations. The connection to either of the two uses an encrypted network connection over HTTPS. Connections are always initiated from the customer side. The update mirror will at no time initiate a connection to the STEP environment.

Direct Connection to Release Server

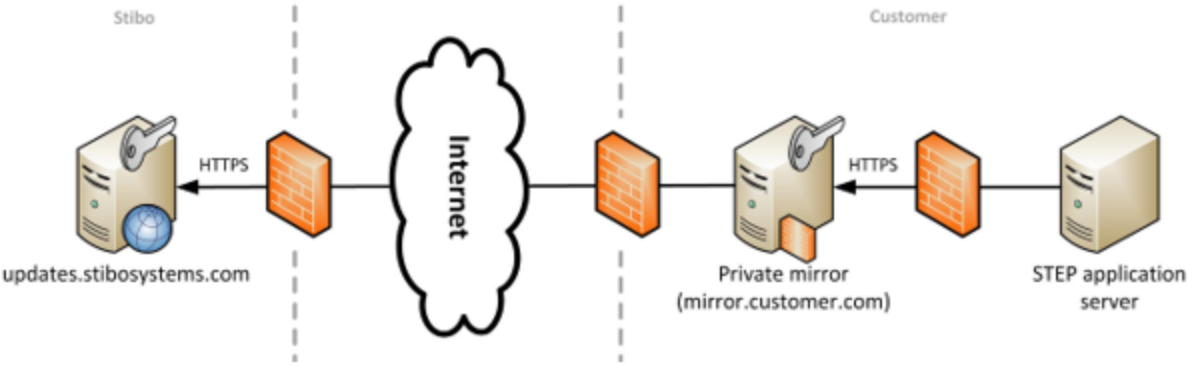
Downloading updates directly from a Release Server is the default method for patching. Using this method, the STEP environment is configured to allow an encrypted connection by HTTPS to the release server. This method offers the best security.



Advantages to using the Direct Connection method include: faster support from Stibo Systems by providing complete version information and a simplified infrastructure.

Private Updates Mirror

As an alternative to accessing the release server directly, it is possible to set up a Private Updates Mirror and configure SPOT on the internal STEP servers to use the mirror instead.



Advantages

The advantages of using the Private Updates Mirror method are:

- If the internet connection or the global updates server breaks down, the already downloaded files will still be available.
- The internet connection bandwidth consumed is reduced by avoiding repeated downloads.
- The network configuration is simpler as only the mirror needs to access the updates server, while the individual SPOT instances can be configured to talk only to the private mirror on the internal network.

Requirements

To run a private mirror server, you need:

- A 64-bit Linux host, not shared with STEP.
- Java 11 64-bit (and updated version will be installed by SPOT, so the OS version is okay for bootstrapping).
- Enough storage to hold the entire mirror (400 GB will suffice).
- Outgoing internet access to the Stibo Systems updates servers on port 443.
- Incoming aces from the private network on port 443 for the SPOT hosts.
- A DNS entry on the local network that can be expected to never change, so *mirror.customer.com* would be preferable to *pc2016-02-13-room7-linux-test-dl120g9.dhcp.customer.com*.

Upstream Root Mirrors

The root mirrors that the private mirror connects to can be listed using `spot --mirrors`, but these are the current hosts:

- *dk1.updates.stibosystems.com*: Primary root mirror.
- *dk2.updates.stibosystems.com*: Secondary root mirror.
- *updates.stibosystems.com*: Fail-over mirror on a shared IP between the two root mirrors.

Outgoing TCP access on port 443 must be allowed to each of the root mirror IP addresses from the private mirror, this way the mirror has more upstream mirrors to pick from if one fails.

SPOT Program

The Stibo Patch Operations Tool (SPOT) program initiates an encrypted exchange between the customer site and the update mirrors at Stibo Systems. This program runs off either the STEP application server or on a dedicated SPOT support installation PC.

The communication sequence between the SPOT program and the update mirror is as follows:

1. SPOT stores the current thin snapshot of version information to updates.stibosystems.com.
2. SPOT fetches the desired recipe of the software bundles to download.
3. SPOT downloads the actual bundles.
4. SPOT stores the updated thin snapshot of the version information to updates.stibosystems.com.

Storing the thin snapshots to updates.stibosystems.com serves two purposes. It enables:

- Stibo Systems to support the STEP environment by providing complete version information.
- Easy creation of the exact software configuration for additional Test / QA environments and in the case of disaster recovery.

Both the metadata (including the thin snapshot) and the bundle recipe, together with the actual bundles, are cached by SPOT and only the files that are actually needed are ever downloaded, so the amount of data transferred is as low as possible.

The SPOT program can be found in the home directory of the STEP installation on the application server. On a Linux server, this will typically be in `/opt/stibo/step`. On a Windows server, this will typically be in `E:\stibo\step`.

Using the Upgrade Command to Add Components

To help simplify the analysis process and make it easier to work with components, the `--upgrade` command can be executed to look for possible upgrades to the components installed on a STEP system. Users can also run the command to search for a component not already installed to verify availability and compatibility with their version of STEP.

The `--upgrade` (or `-u`) command is used to calculate the newest possible version of the listed components that can be installed given different restrictions on how large of an upgrade is allowed. This command never changes the STEP system or performs any automatic upgrading.

To further facilitate the process, the upgrade options output that is displayed upon running the `--upgrade` command includes a recipe file that can later be applied to the system.

Outlined below are the different upgrade command options with examples. The name of the actual component (s) should be used in place of what is shown in the examples.

Upgrade Options	Examples
Upgrading one component	To upgrade the Experian component, use: `--upgrade=experian`
Upgrading several components	To upgrade both Experian and Loqate, use: `--upgrade=experian,local-loqate`
Upgrading the baseline (the STEP version such as 9.0, 9.1)	The baseline can also be upgraded by using the component name *step*: `--upgrade=step` Pick the release of the baseline by specifying a prefix: `--upgrade=step:9.1` When a prefix is specified, the newest version matching the prefix will be tried.

Installation candidates (components that have not yet been installed) can also be found by using the `--upgrade` command, as described above.

Upgrade levels

The calculation used to determine upgrades can potentially produce suggestions for up to five levels of upgrade. Only the upgrades that bring newer versions of the listed components will be shown in the result. Below, the options shown are sorted by how aggressive the update would be with regard to introducing new component versions.

Level	Description
Listed	This is the most conservative upgrade possible where only the listed components are touched.
Dependents	This level allows upgrading of: • The listed components • The components that depend on the listed components
Dependencies	This level, listed with `DEPENDENCIES_BUT_NOT_BASELINE` in the file name, allows upgrading of: • The listed components • The components that depend on the listed components

Level	Description
	• The components that the listed components depend on, but not STEP baseline
Baseline within Maintenance Patch	This level, listed with `BASELINE_WITHIN_MP` in the file name, allows upgrading of all components, including the STEP baseline, but only to the latest maintenance patch of the same release as the one currently installed. For example, if the system has step-8.2-mp1 installed, then this level would look for the newest MP of that release, possibly 8.2-mp3, but not 8.3.
Baseline	This is the least conservative upgrade level, which allows upgrading of all components, including the STEP baseline to the latest released version.

The upgrade options are shown on screen with the upgrade file recipes listed. Users can use standard commands to view a detailed change log, prepare for an installation, and to apply changes to their STEP system.

For example: `--upgrade=inmemory`

```
Found 3 possible upgrades to choose from:
================================================================================
Option 1: Upgrade only the listed components
Components:
       * assetloader: Keep at 7.0.14 (newest available: 7.0.24)
       * inmemory: Upgrade from 7.0.10 (newest available: 7.0.23)
       * spot: Keep at 7.0.48 (newest available 7.0.65)
File: /home/step/admin/spot/recipes/upgrade/upgrade.LISTED.2017-11-01-15-24-28.spr

================================================================================

Option 2: Upgrade to latest maintenance patch within the same STEP release
+ All components
Components:
       * assetloader: Keep at 7.0.14 (newest available: 7.0.24)
       * inmemory: Upgrade from 7.0.10 to 7.0.15 (newest available: 7.0.23)
       * spot: Keep at 7.0.48 (newest available 7.0.65)
       * step: Upgrade from 8.0-mp3-2016-09-06-14-12-00 to 8.0-mp4-2016-10-04-10-10-27
(newest available: 8.2-mp3-2017-11-02-07-39-51)

File: /home/step/admin/spot/recipes/upgrade/upgrade.BASELINE_WITHIN_MP.2017-11-01-
15-24-28.spr

================================================================================

Option 3: Upgrade to latest STEP release (full upgrade)
```

```
+ All components
Components:
       * assetloader: Keep at 7.0.14 (newest available: 7.0.24)
       * inmemory: Upgrade from 7.0.10 to 7.0.14 (newest available: 7.0.23)
       * spot: Keep at 7.0.48 (newest available 7.0.65)
       * step: Upgrade from 8.0-mp3-2016-09-06-14-12-00 to 8.1-mp5-2017-10-02-16-10-00
(newest available: 8.2-mp3-2017-11-02-07-39-51)

File: /home/step/admin/spot/recipes/upgrade/upgrade.BASELINE.2017-11-01-15-24-28.spr
```

If the system is ignoring any components or if the system cannot find a way to upgrade the components specified, the applicable messaging will be shown on the screen. All ignored versions will not be considered when trying to find an upgrade.

Starting with STEP 8.3 and with all subsequent versions, the `--upgrade` command can be used in place of the installation commands. For example, `--upgrade=wikimetadata` or `--upgrade=acrolinx`.

Using SPOT to Remove Components

SPOT is able to remove components by naming the specific components to remove. For example, to remove the 'acrolinx' component, you will add 'rm:acrolinx' to the argument '--apply,' like this:

spot --apply=rm:acrolinx

It is possible to apply a recipe along with removing a component, but the parts of the recipe need to be separated with a comma. For example:

spot --apply=to:wikimetadata/7.0/wikimetadata-7.0.5.spr,rm:acrolinx

Using SPOT to Clear Cache

The cache maintained by SPOT for files downloaded from the updates server can grow quite large, and on some systems with very little disk space, it is possible to exhaust the free space on the system leading to errors. To mitigate the problem with too small file systems, SPOT deletes the entire cache if there is less than five (5) GB of free disk space when SPOT is started.

A manual option to trigger the cleanup has also been added: --cleancache

Using --cleancache will remove the SPOT cache before using the cache.

Note: The best solution to running out of disk space is to ensure that the file system for STEP is large enough. There are many other things that can cause space to be consumed that SPOT can do nothing about.

STEP Patching Procedures

All commands listed are valid for any STEP environment, counting single application server setups and clusters.

Back Up the STEP Database and Application

Before patching STEP, a fallback procedure should be developed to mitigate any risk. Ideally, full back ups of the STEP database should be maintained, and the option to restore the database to a specific point in time should be available.

Note: The requirement for each individual patch may vary. Refer to the relevant release note for more information.

Back ups of the STEP application should also be maintained, including all files provided in STEP_HOME/config.properties and a snapshot of the STEP system itself.

To take a snapshot of the STEP system:

```
cd /opt/stibo/step
./spot --snapshot=/workarea/<snapshot-env-date>.spr
```

Prepare the Patch

The patch should be downloaded in advance to avoid unnecessary downtime for deployment.

A STEP core patch may look like the following command:

```
./spot --prepare=to:step/platform/step-<release>.spr
```

Sometimes, customers have their own components in addition to the STEP core, and the command could look something like this:

```
./spot --prepare=to:step/platform/step-
<release>.spr,to:customer/<customer>/<customer>-addon/7.0/<customer>-addon-7.0.x.spr
```

Install the Patch

The patch should be installed by the following command:

```
./spot --apply=to:step/platform/step-<release>.spr
```

With customer components included, the command looks like this:

```
./spot --apply=to:step/platform/step-
<release>.spr,to:customer/<customer>/<customer>-addon/7.0/<customer>-addon-7.0.x.spr
```

STEP will automatically stop and start during the patch session.

In case of any deprecated parameters in the configuration, follow the instructions on the screen that explain how to correct and restart STEP.

```
./spot --start
```

Fallback

In the event of errors during patching, it may be necessary to restore STEP to a previous state.

Depending on the contents of the patch the following steps should be completed when reverting the patch:

1. Stop STEP

   ```
   ./spot --stop
   ```

2. Restore database
3. Restore configuration files
4. Redeploy STEP using a snapshot

   ```
   ./spot --apply=/workarea/<snapshot-env-date>.spr --sync --syncmode=delete
   ```

Using the snapshot and the above `--sync --syncmode=delete` command will entirely recover STEP and delete any files related to a failed patch-session.

Note: Refer to the relevant release note to check if restoring the database is required for the patch in question.

Patching Security

Stibo Systems only distributes software via the updates.stibosystems.com server or one of the official mirrors.

The update mirror web server is configured to only communicate via HTTPS (never plain HTTP) on port 443, with only the high security cipher suites using the Apache SSLCipherSuite 'HIGH' option and only communicating with clients which have a proper client certificate issued by the build system certificate authority (CA) of Stibo Systems. This Stibo-specific CA was created solely for the purpose of certifying various STEP-related infrastructures.

Unlike a standard website where an external CA-signed certificate is used for ease of access by multiple clients (users), the updates server has only one client that is allowed to communicate with it: the SPOT client. For this reason, Stibo Systems believes this to be a safer and stronger security approach – over using an external CA certificate – as it is not possible for a cyberattacker to use a fake certificate from a compromised external CA to gain access.

By taking this approach, some auditing tools may register a false positive and flag the server's certificate as self-signed. Because of this, security teams should configure these tools to trust Stibo Systems' CA to certify *stibosystems.com* domains

The client certificate required for communicating with the update mirror is included in the STEP installation package, and is used by the SPOT program to fetch both the software required for the initial installation and future application updates. Only the certificate used by the updates server will be trusted by SPOT for downloading these installation bits and updates.

All the certificates involved use 2048-bit RSA keys, so the system is considered secure against any man-in-the-middle attacker for the foreseeable future. Even with a valid client certificate, the operations allowed are severely limited to downloading only the licensed software produced by Stibo Systems and to saving customer-specific thin snapshots that do not contain software, so a compromised client would not be able to affect other customers or compromise other clients.

The SPOT program caches all files locally and validates contents using a SHA-1 hash before using the cached files, so the amount of traffic is kept as low as possible while ensuring the integrity of the cached files.

At no point will the STEP software communicate customer data back to the update mirrors at Stibo Systems. The thin snapshots uploaded to the release server contain only a list of versions of the installed STEP software components and they are only used by Stibo Systems to provide the best support to the STEP system.

Configuring a Private Updates Mirror

Requirements for running a private mirror server include:

- A 64-bit Linux host, not shared with STEP.
- 64-bit Java 11 (an updated version will be installed by SPOT, so the OS version is okay for bootstrapping).
- Enough storage to hold the entire mirror (currently 400 GB will suffice).
- Outgoing internet access to the Stibo Systems updates servers on port 443.
- Incoming access from the private network on port 443 for the SPOT hosts.
- A DNS entry on the local network that can be expected to never change, so *updates.example.com* would be preferable to *pc2016-02-13-room7-linux-testdl120g9.dhcp.example.com*.

The root mirrors that the private mirror connects to can be listed using `spot --mirrors`, but these are the current hosts:

- *dk1.updates.stibosystems.com*: Primary root mirror.
- *dk2.updates.stibosystems.com*: Secondary root mirror.
- *updates.stibosystems.com*: Fail-over mirror on a shared IP between the two root mirrors.

Outgoing TCP access on port 443 must be allowed to each of the root mirror IP addresses from the private mirror, this way the mirror has more upstream mirrors to pick from if one fails.

The mirror server listens on three ports:

- *10080*: The Admin port of dropwizard, which is used to serve HTTP requests that allow monitoring the health of the server. The init script uses this port to check if the server is running.
- *10081*: The stop port of Jetty. The init script uses this port to shut down the server in an orderly fashion. This port should not be accessed from outside the machine itself.
- *10082*: The HTTPS service port that serves the actual mirror. This port should not be accessed from outside the machine itself.

These ports are all internal to the host that the server runs on and external systems should not connect directly to them (with the possible exception of having a monitoring system talking to port 10080.)

Important: Do not configure any STEP systems to talk to the mirror on port 10082. Port redirection (as described in the next section) must be set up.

IPTables Rules

It is impossible to listen to port 443 when running the Java process as an unprivileged user. To account for this, a set of iptables rules must be used.

There are two ways to install the required rules: either run the mirror script as root when starting the server or set up iptables at the OS level. If the init script is called by root, then it will install the needed port redirection, but if the administrator tasked with maintaining the mirror does not have sudo access to this script, then the rules can be inserted into the */etc/sysconfig/iptables* config file, allowing the OS to load the rules at boot time.

These rules redirect all incoming requests to TCP port 443 over to port 10082 where the server listens.

To configure iptables on the server, switch to the root user and run the following command to view the current settings:

```
[root@mirror mirror]# /sbin/iptables -L -n --line-numbers
Chain INPUT (policy ACCEPT)
num  target     prot opt source      destination
1    ACCEPT     all  --  0.0.0.0/0   0.0.0.0/0   state RELATED,ESTABLISHED
2    ACCEPT     icmp --  0.0.0.0/0   0.0.0.0/0
3    ACCEPT     all  --  0.0.0.0/0   0.0.0.0/0
4    ACCEPT     tcp  --  0.0.0.0/0   0.0.0.0/0   state NEW tcp dpt:22
5    REJECT     all  --  0.0.0.0/0   0.0.0.0/0   reject-with icmp-host-
prohibited

Chain FORWARD (policy ACCEPT)
num  target     prot opt source       destination
1    REJECT     all  --  0.0.0.0/0    0.0.0.0/0   reject-with icmp-host-
prohibited

Chain OUTPUT (policy ACCEPT)
num  target     prot opt source       destination
```

In the output, there will be a line that displays **REJECT** as the **INPUT** type, and in its first column **(num)**, the line number is shown ('5' in the above example.) This line number will be the starting line for adding entries to the iptables configuration.

Once this information is known, run the following commands to add the needed port-opening entries:

```
[root@mirror mirror]# /sbin/iptables -I INPUT <line_number> -p tcp -m tcp --dport
443 -j ACCEPT
[root@mirror mirror]# /sbin/iptables -I INPUT <line_number> -p tcp -m tcp --dport
10082 -j ACCEPT
```

In the example above, the line number shown is '5', and therefore, the commands would look like the following commands:

```
[root@mirror mirror]# /sbin/iptables -I INPUT 5 -p tcp -m tcp --dport 443 -j
ACCEPT
[root@mirror mirror]# /sbin/iptables -I INPUT 6 -p tcp -m tcp --dport 10082 -j
ACCEPT
```

Afterward, add the entries for redirection by executing these commands:

```
[root@mirror mirror]# /sbin/iptables -t nat -A PREROUTING -p tcp -m tcp --dport
443 -j REDIRECT --to-ports 10082
[root@mirror mirror]# /sbin/iptables -t nat -A OUTPUT -o lo -p tcp -m tcp --dport
443 -j REDIRECT --to-ports 10082
```

Once that has been done, the added entries can be checked by running the commands that follow:

```
[root@mirror mirror]# /sbin/iptables -L -n --line-numbers
Chain INPUT (policy ACCEPT)
num  target     prot opt source       destination
1    ACCEPT     all  --  0.0.0.0/0    0.0.0.0/0   state RELATED,ESTABLISHED
2    ACCEPT     icmp --  0.0.0.0/0    0.0.0.0/0
3    ACCEPT     all  --  0.0.0.0/0    0.0.0.0/0
4    ACCEPT     tcp  --  0.0.0.0/0    0.0.0.0/0   state NEW tcp dpt:22
5    ACCEPT     tcp  --  0.0.0.0/0    0.0.0.0/0   tcp dpt:443
6    ACCEPT     tcp  --  0.0.0.0/0    0.0.0.0/0   tcp dpt:10082
7    REJECT     all  --  0.0.0.0/0    0.0.0.0/0   reject-with icmp-host-prohibited

Chain FORWARD (policy ACCEPT)
num  target     prot opt source       destination
1    REJECT     all  --  0.0.0.0/0    0.0.0.0/0   reject-with icmp-host-prohibited

Chain OUTPUT (policy ACCEPT)
num  target     prot opt source       destination
```

```
[root@mirror mirror]# /sbin/iptables -L -n --line-numbers -t nat
Chain PREROUTING (policy ACCEPT)
num  target     prot opt source       destination
1    REDIRECT   tcp  --  0.0.0.0/0    0.0.0.0/0    tcp dpt:443 redir ports 10082

Chain INPUT (policy ACCEPT)
num  target     prot opt source       destination

Chain OUTPUT (policy ACCEPT)
num  target     prot opt source       destination
1    REDIRECT   tcp  --  0.0.0.0/0    0.0.0.0/0    tcp dpt:443 redir ports 10082

Chain POSTROUTING (policy ACCEPT)
num  target     prot opt source       destination
```

If everything looks to be correct, save the configuration so that it will be loaded each time the system reboots using these commands:

```
[root@mirror mirror]# /sbin/service iptables save
[root@mirror mirror]# /sbin/service iptables stop
[root@mirror mirror]# /sbin/service iptables start
```

Important: If the mirror server is a RHEL 7.x system, the above `/sbin/service iptables stop` and `/sbin/service iptables start` commands should be replaced with the following: `/bin/systemctl stop iptables` and `/bin/systemctl start iptables`.

Once complete, the */etc/sysconfig/iptables* config file should look similar to the following:

```
# Generated by iptables-save v1.4.7 on Tue Jun 21 14:16:10 2016
*nat
:PREROUTING ACCEPT [4595:497811]
:INPUT ACCEPT [1:28]
:OUTPUT ACCEPT [0:0]
:POSTROUTING ACCEPT [0:0]
-A PREROUTING -p tcp -m tcp --dport 443 -j REDIRECT --to-ports 10082
-A OUTPUT -o lo -p tcp -m tcp --dport 443 -j REDIRECT --to-ports 10082
COMMIT
# Completed on Tue Jun 21 14:16:10 2016
# Generated by iptables-save v1.4.7 on Tue Jun 21 14:16:10 2016
*filter
:INPUT ACCEPT [0:0]
:FORWARD ACCEPT [0:0]
:OUTPUT ACCEPT [434:47393]
-A INPUT -m state --state RELATED,ESTABLISHED -j ACCEPT
-A INPUT -p icmp -j ACCEPT
-A INPUT -i lo -j ACCEPT
-A INPUT -p tcp -m state --state NEW -m tcp --dport 22 -j ACCEPT
-A INPUT -p tcp -m tcp --dport 443 -j ACCEPT
-A INPUT -p tcp -m tcp --dport 10082 -j ACCEPT
-A INPUT -j REJECT --reject-with icmp-host-prohibited
-A FORWARD -j REJECT --reject-with icmp-host-prohibited
COMMIT
# Completed on Tue Jun 21 14:16:10 2016
```

Installing a Private Mirror

1. Satisfy all requirements mentioned above.
2. Make a note of the DNS name which all the SPOT hosts will be using. In this example we will call it *mirror.customer.com*.
3. Contact Stibo Systems Technical Services and request that a system name is created for the mirror. This must be human readable and unique. The system name in this example is *your-mirror*.
4. Create a directory for the mirror.
5. Unzip the SPOT foothold (must be newer than the March 2016 release).
6. Run: `./spot --enroll=mirror:your-mirror:mirror.customer.com`.
7. Run: `./spot --apply=to:updates/mirror/latest.spr`.
8. Edit the *mirror.yaml* file and review the options in the file. Some of them, particularly those dealing with mailing of errors, will need to be changed.
9. Run: `./mirror start`.
10. Your mirror should now be running on *mirror.customer.com*.
11. On a system with STEP installed, run the following command:

```
./spot --updates=https://mirror.customer.com --ping
```

12. As `root`, create a symlink to the mirror script into the appropriate sysv init directories using a command similar to the following:

```
ln -s <mirror_home>/mirror /etc/rc3.d/S90stibo-updatesmirror
```

For example:

```
ln -s /home/mirror/mirror /etc/rc3.d/S90stibo-updatesmirror
```

Important: Do not run any of these commands as `root`. Make sure an unprivileged user exists for this mirror — e.g., `mirrorsw`.

Once the private mirror has been configured, it can be verified by running the following command on the STEP application server as the `stibosw` (or equivalent) user:

```
[stibosw@app1 step]$ ./spot --mirrors
Stibo Patch Operations Tool
Priority Id          Name                          Url                                [X]
100      <customer>  <customer> local mirror       https://mirror.customer.com        X
30       global      Auto failover mirror          https://updates.stibosystems.com
20       dk1         Primary mirror in Aarhus DK   https://dk1.updates.stibosystems.com
10       dk2         Secondary mirror in Aarhus DK https://dk2.updates.stibosystems.com

Please use spot --mirrors --updates={Url} to set the upstream mirror
```

The mirror can also upgrade itself using the init script by running: `./mirror upgrade`. The upgrade command calls the `spot --apply=to:updates/mirror/latest.spr` and `./mirror restart` commands.

If the iptables' rules have been added to the RHEL config file, the init script no longer needs root access and can be started by an unprivileged user. This is done by editing said user's *crontab* entries (such as `crontab -e`) and adding the following line:

```
@reboot <mirror_home>/mirror start
```

For example:

```
@reboot /home/mirror/mirror start
```

Preemptive Download

The mirror server can download files before the STEP systems ask for them. Doing this allows most files to be served from the local mirror without waiting for the upstream mirror, so better performance can be expected. This comes at the cost of more disk space being utilized and the possibility of downloading files that end up never being needed.

The download option has three possible values:

- `download: HISTORIC`: Downloads all the files available from the upstream mirror, regardless of age. This requires about 1.5 TB of space.
- `download: RELEASED`: Downloads newly released code as soon as it becomes available, this is the default and will steadily consume space. About 2 GB are consumed per month.
- `download: ON_DEMAND`: Nothing is downloaded until a client asks for it.

Note: When new content (e.g., monthly maintenance patches, add-on components, hotfixes, etc.) is downloaded to the mirror, it will be saved to the server's `<mirror_home>/content/takeout` directory (e.g., `/home/mirror/content/takeout`).

As no user is actively waiting for the preemptive downloads to complete and because the downloads can be quite large, the bandwidth consumed by the background downloads can be limited via the `bulkDownloadSpeedInMbitPerSecond` configuration option. The default limit is 10 Mb/s, so the expected lag after a release of STEP until the mirror is in sync should be less than an hour.

The bulk download speed limit is applied to the download of newly released files and historically released files separately, so if a historic download is running, then the two bulk processes can consume twice the speed limit in total.

If downloads take a long time to complete, it could be because the network or the upstream mirror is overloaded. To avoid contributing to the problem the bulk download threads will sleep for a while after completing a download. The amount of time to sleep after a download can be specified using the `bulkBackoffFactor` option, which defaults to '1.5'.

For example, if a download takes two seconds then a *bulkBackoffFactor* of '1.5' means that the process will sleep three seconds before downloading the next file.

STEP Architecture

This section is intended to be used by experienced Information Technology Professionals. An intermediate-to-advanced understanding of technical architecture is assumed. This is a guide but does not address all situations. Consult your implementation team or your Stibo Systems representative for further information.

This section describes the architecture of STEP and addresses the following questions that are vital for successful implementation of the Master Data Management (MDM) solution:

- What options allow STEP to integrate with other systems in the infrastructure?
- What issues must be considered before making final decisions about fundamental topics, such as hardware, network design, backup and recovery, clustering, and security?
- What is required to maintain a STEP installation that is fail-safe and scalable, and that behaves in a predictable manner even in a worst-case scenario?
- What are the recommended procedures for keeping the STEP software and any third-party software, up to date?

Architecture Elements

For more information, refer to the following topics:

- Architecture Layout
- Integration
- Storing Asset Content Externally
- Business Process Management
- Clustering
- High Availability Implementation
- Network Consideration
- Security

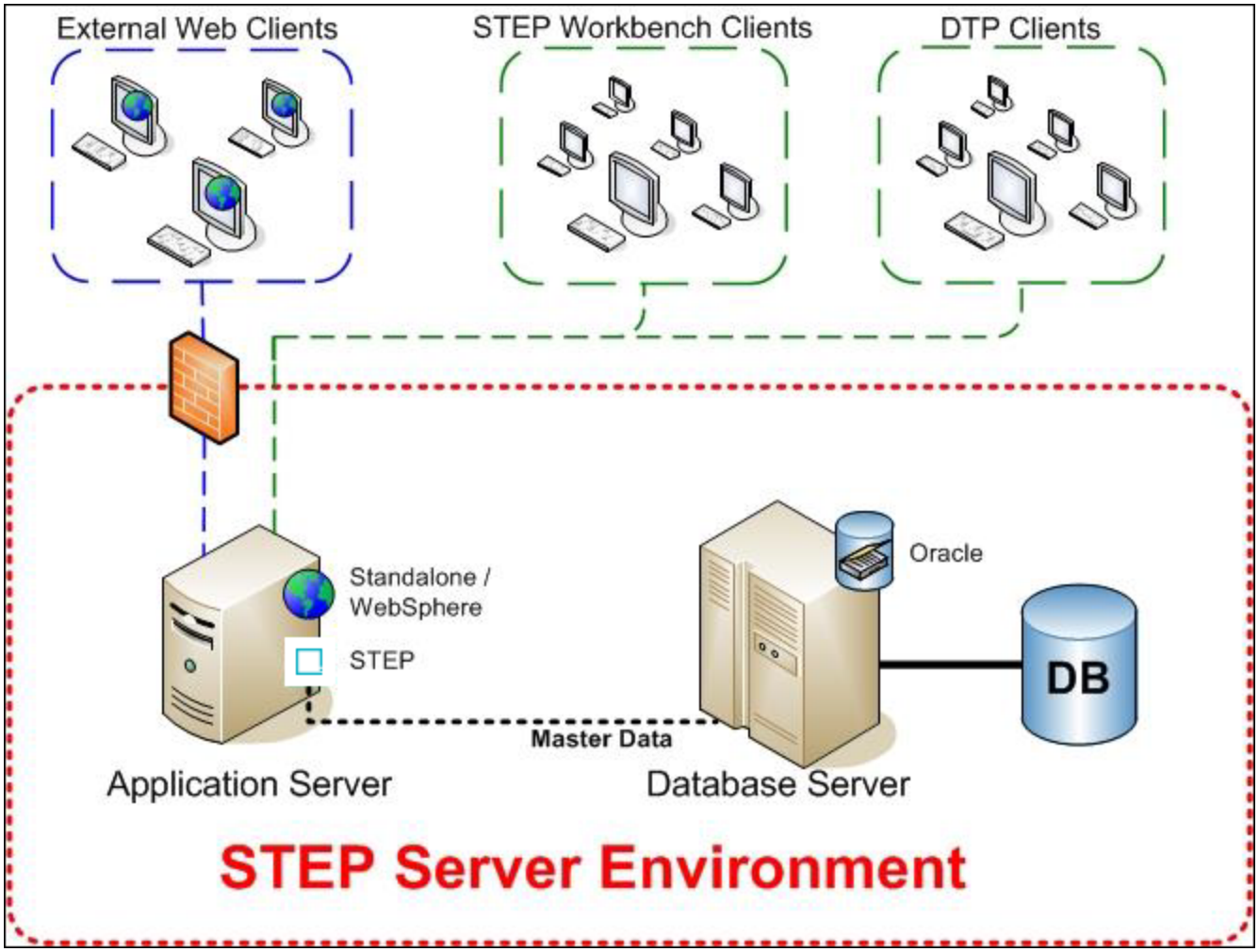
Additionally, platform and software support is addressed for each release. For example, refer to the **11.2 Platform and Software Support** topic within the **System Update and Patch Notes / Platform and Software Support** documentation.

Architecture Layout

This is one of the topics that describes the architecture of the STEP solution. The full list is defined in the **STEP Architecture** topic.

STEP Architecture

This illustration shows a simple STEP system setup with one application server and one database server.



The Application Server is a physical server running either standalone (on Oracle Java) or a WebSphere Server. Within the application server, an instance of the STEP application is running, supporting three types of clients. The STEP Workflow component supporting business process management is part of the STEP application.

The Oracle Database Server provides the primary storage for all information to be stored persistently by the STEP system.

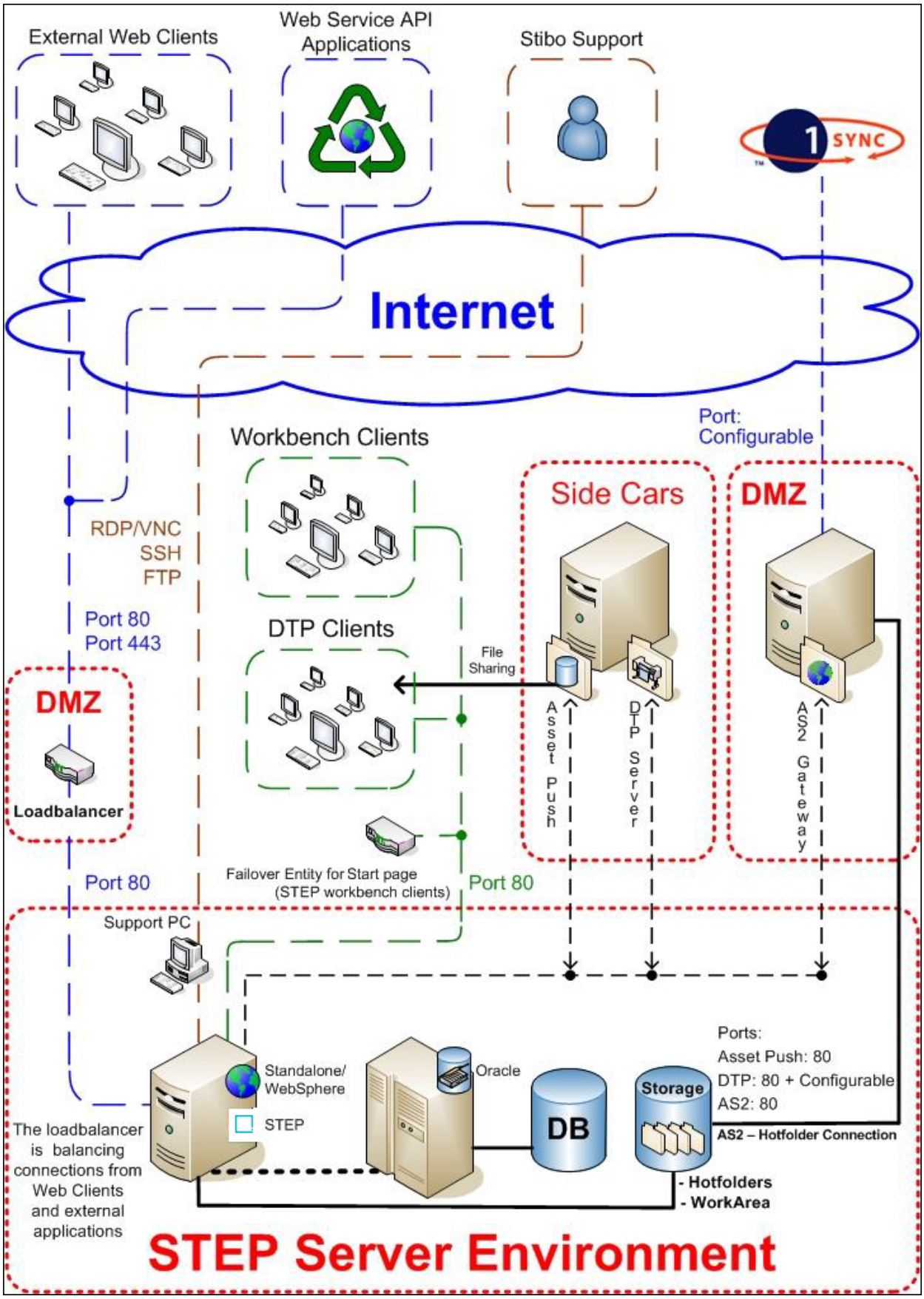
These types of clients are represented:

- STEP Workbench Client is a cross-platform, Java client providing the full set of functionality of the STEP application.
- External Web Client is an internet browser-based client providing functionalities used by users, suppliers, and vendors.
- Desktop Publishing (DTP) Client represents the InDesign application extended with a STEP DTP plug-in which allows the DTP application to communicate with STEP via a dedicated Web Service API.

Enterprise Architecture

The following comments apply to the illustration below which shows how the STEP system relates to its immediate surroundings.

- External applications communicate with the STEP system using the Web Service API or, as an alternative, the REST API. Additionally, other ways to integrate with the system are mentioned in the **Integration** topic.
- For simplicity, STEP is only represented as a single box. Refer to the **STEP Architecture** section above for more details.
- Sidecars denote a number of applications supporting the STEP Enterprise application functionality, such as:
  - DTP server (either InDesign or QuarkXPress) provides operations such as proof views and PDF renderings of the DTP documents for users without DTP clients. It is possible to install multiple, physical DTP servers for load balancing and fail over.
  - Asset Push Client listens for creation and updates of assets stored in STEP and automatically extracts a configurable version and subset of these assets to the file system. To allow DTP clients and servers faster access to images and further performance improvements, the output images can be synchronized to a file server near the DTP users if they are far away from the STEP system.
- AS2 gateway handles the AS2 communication with 1Sync. It can either be the OpenAS2 side car supplied by Stibo Systems or a customer-supplied (and perhaps pre-existing) AS2 gateway. The AS2 gateway should be placed in a separate DMZ for tighter control on the network traffic, so only incoming traffic from 1Sync is allowed on the configured port.
- DB represents the disks necessary for storing the data of the Oracle database being part of STEP.
- Failover Entity ensures that the Start page of the workbench does not become a single point of failure. The workbench application handles load balancing and failover directly and does not use this device once started, (refer to the **Clustering** topic). The entity can either be a dedicated piece of hardware or a computer running an Apache web server with an appropriate proxy configuration.
- Loadbalancer provides an additional point of entry for external users (through the internet) accessing STEP. This point of entry is either a network box or a physical server capable of performing URL filtering together with session affinity, ensuring that the same session goes to the same server until the session times out. In effect, this box will perform load balancing and failover tasks for the STEP Web Client.
- For Stibo Systems Technical Support to be able to assist with troubleshooting problems, configure a remote connection from Stibo Systems to the customer hosting the system. This can be a Citrix-based connection or a direct VPN connection to a Support PC in the customer's network.


External Web Clients
Web Service API Applications
Stibo Support
1 SYNC
Internet
Workbench Clients
Port: Configurable
Side Cars
DMZ
RDP/VNC
SSH
FTP
Port 80
Port 443
DTP Clients
File Sharing
Asset Push
DTP Server
AS2 Gateway
DMZ
Loadbalancer
Failover Entity for Start page (STEP workbench clients)
Port 80
Port 80
Support PC
Standalone/ WebSphere
STEP
Oracle
DB
Storage
Ports:
Asset Push: 80
DTP: 80 + Configurable
AS2: 80
AS2 – Hotfolder Connection
- Hotfolders
- WorkArea
The loadbalancer is balancing connections from Web Clients and external applications
STEP Server Environment

Integration

This is one of the topics that describes the architecture of the STEP solution. The full list is defined in the **STEP Architecture** topic.

The components listed below all contribute to the data exchange between STEP and external sources / applications. Together, these components create a suite that exposes the STEP system as an open ended and flexible system, which easily integrates with an existing infrastructure.

Inbound Integration Endpoints

Inbound integration endpoints (IIEPs) read from a queue to retrieve data from an external system. When active, an IIEP tracks errors, measures throughput and latency times. For more information, refer to the **Inbound Integration Endpoints** section of the **Data Exchange** documentation.

IIEPs allow a variety of receiver methods, as defined in the **IIEP Receiver Methods** topic of the **Data Exchange** documentation.

Outbound Integration Endpoints

Outbound integration endpoints (OIEPs) sends data via a queue to an external system. For more information, refer to the **Outbound Integration Endpoints** section of the **Data Exchange** documentation.

OIEPs allow a variety of delivery methods, as defined in the **OIEP - Delivery Method Flipper** topic of the **Data Exchange** documentation.

Export Manager

The Export Manger wizard allows the user to export STEP data via a background process. For more information, refer to the **Export Manager** section of the **Data Exchange** documentation.

Export Manager can export data via a number of formats, as defined in the **Export Manager - Select Delivery Method** topic of the **Data Exchange** documentation.

Import Manager

The Import Manger wizard allows the user to import STEP data via a background process. For more information, refer to the **Import Manager** section of the **Data Exchange** documentation.

Import Manager can import data via a number of formats, as defined in the **Import Manager - Select Format** topic of the **Data Exchange** documentation.

Web Service API

Custom applications can be built using the STEP Web Services API to read, update and delete data in the STEP PIM. The communication takes place via a standard SOAP protocol providing a maximum of decoupling between the external applications and STEP. For more information, click the **STEP API Documentation** button on the Start Page.

REST API

REST (Representational State Transfer) covers simple transfer of data over HTTP by using URLs to identify the data, the same way typing a URL into a web browser returns HTML which is rendered by the web browser. With the REST API, the URL points to data in a STEP system. The STEP system returns the data in STEPXML format, which enables viewing and navigating data in STEP via a web browser. The same can be done by programs that want to integrate with STEP by issuing an HTTP GET request and parsing XML. It is also possible to issue an HTTP PUT request to update and approve STEP data. There is also an option to apply an XSL stylesheet stored in STEP to the data before it is returned. This enables easy creation of web content proofing by storing XSL and CSS stylesheets as assets in STEP. For more information, click the **STEP API Documentation** button on the Start Page.

SQL API

The SQL API can be used to generate customized reports and extractions. The API consists of a series of views and stored packages. These structures can be accessed via SQL, PL/SQL, or other database-enabled programming languages such as Visual Basic or Java in order to extract data. To secure the read-only property of the API, all of the API components are installed under a separate Oracle user account. This user is granted limited rights, excluding update, insert, or delete rights on the tables underlying the views. For more information, click the **STEP API Documentation** button on the Start Page.

GDSN Integration

Data can be imported from and exported to the data pool 1SYNC, a subsidiary of GS1 US, which is a Global Data Synchronization Network (GDSN)-certified Data Pool that offers a range of data synchronization solutions that eliminates costly data errors, increases supply chain efficiencies, and provides a foundation for the implementation of technologies, such as the Electronic Product Code™ (EPC) and Radio Frequency Identification (RFID).

Imports are configured as 'subscriptions' in STEP with different criteria. These subscriptions are sent to 1SYNC and if products in 1SYNC match a subscription, STEP receives an import file with the product data included. This synchronization of product data continues until the data receiver or the data provider wishes to stop it.

Exports of STEP products can also be registered with the 1SYNC GDSN data pool. These products may subsequently be published to a specific trading partner via the GDSN network while tracking the status inside STEP. Updates to published products are tracked so that the products may be re-published.

For more information, refer to the **GDSN Receiver Solution Enablement** documentation and the **Product Data Exchange** section of the **Data Integration** documentation.

Storing Asset Content Externally

This is one of the topics that describes the architecture of the STEP solution. The full list is defined in the **STEP Architecture** topic.

The following options are available to store asset content externally:

- External File Structure (EFS)
- External Digital Asset Management (DAM)

Business Process Management

This is one of the topics that describes the architecture of the STEP solution. The full list is defined in the **STEP Architecture** topic.

The following support is available to model and integrate data management processes tailored towards specific business needs.

Background Processing

Complex and time consuming processes can be run as background processes (BGPs). Examples of fundamental core features that are implemented as background processes are import, export, translation, approve recursive, and report generation. BGPs can be started from the workbench and are passed onto a JMS-based queue system. The jobs in the queue are then processed in FIFO order, allowing processes to run in parallel to the extent defined by an adjustable queue configuration. For example, a system can be set up to have two separate queues (one for imports and one for exports) and the 'size' of the queues can be configured so that three (3) imports and six (6) exports max are allowed to run in parallel. For more information, refer to the **Background Processes and Queues** section of **System Setup** documentation.

Business Rules

Many business processes necessitate the ability to pass approved data automatically from STEP to other systems. This poses extra requirements on the validity of data - otherwise data will possibly be rejected by the receiving system. In order to ensure validity, STEP provides an opportunity to add business rules that are applied at the time of approval. Such business rules may perform a check across all data defined on an object and reject or accept approval. In addition, STEP has support for creating approve triggers that are also executed as an integral part of the approval process. While business rules are designed for read-only validity checks on data before and after approval, the approve triggers are able to change data as an integral part of the approval process. If any business rule fails, approval does not take place, and the user is notified. Business rules can be written using a subset of the Java-based Core Domain API. For more information, refer to the **Business Rules** documentation, and also review the API documentation by clicking the **STEP API Documentation** button on the Start Page.

Workflows

The workflow functionality is used for modeling processes where tasks are moved between people. Workflows are 'data-centric' which means that a 'workflow instance' is defined as the relation between a workflow definition and an object in the database. For more information, refer to the **Workflows** documentation.

Clustering

This is one of the topics that describes the architecture of the STEP solution. The full list is defined in the **STEP Architecture** topic.

Clustering creates a STEP system with high availability (HA) and scalability. In a clustered setup, the load is distributed so that all members of a cluster take a fair part of the load. A successful cluster setup requires accurate configuration of many parameters: number of servers, bandwidth, network latency, etc. A fail-over strategy must be available for all of the essential components of the system.

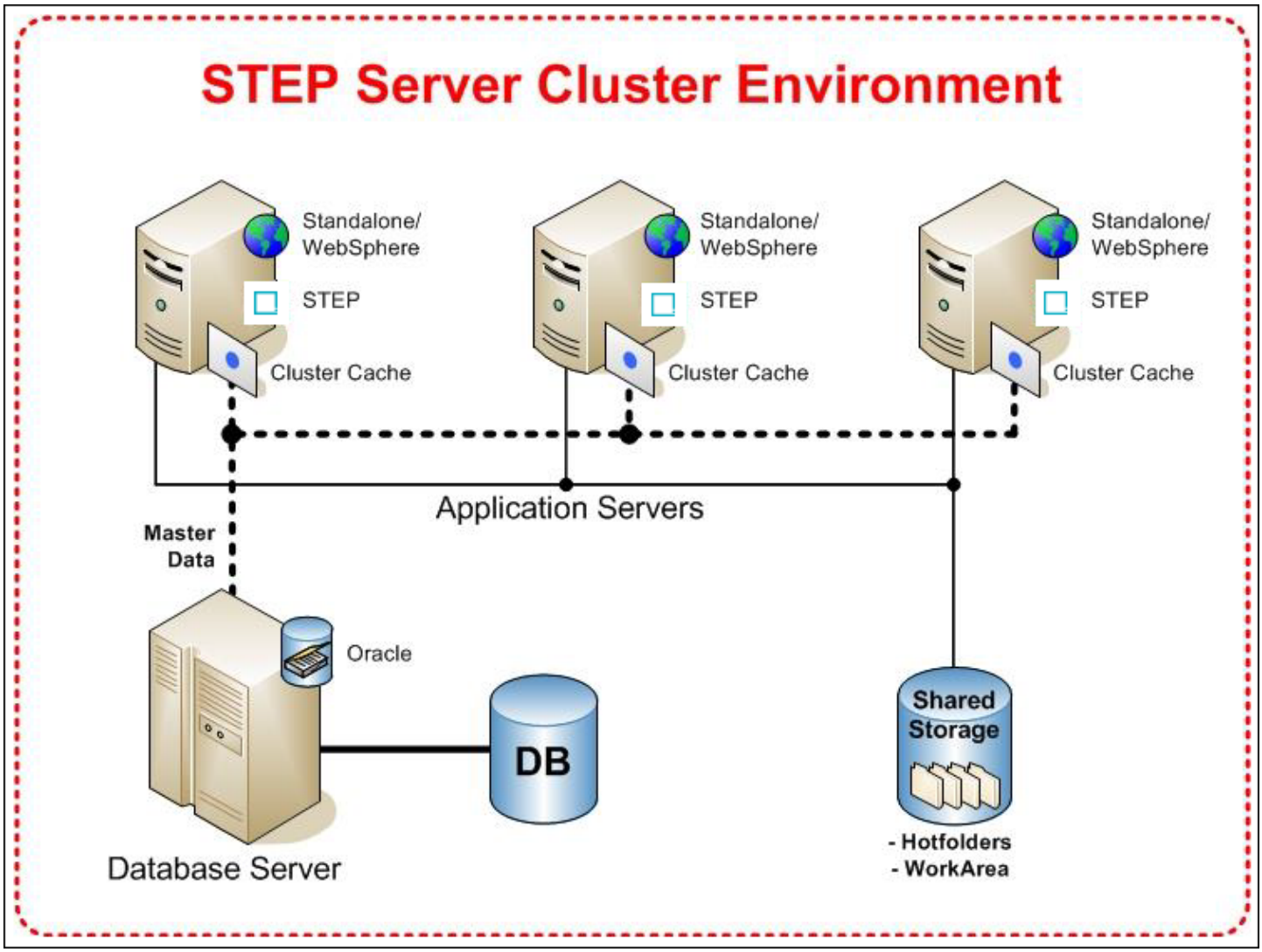
Homogenous cluster nodes are recommended in a cluster setup. The simplest type of cluster is created by adding an additional application server that does everything the first application server does. It is recommended that clustered applications servers are set up as clones and kept on identical hardware.

Important: Clustering a STEP system is handled by the STEP application itself. It does not use clustering technology provided by a commercial application server.

Architecture

The following comments apply to the illustration below which shows a STEP system running as a cluster setup with three application servers.

- Application Servers are physical servers running either standalone (using Oracle Java) or a WebSphere Server. Within each of the application servers, an instance of STEP is running, supporting all three types of clients mentioned earlier (workbench clients, web clients and DTP clients). Each application server computer has a cache of its own that is synchronized whenever write operations take place.
- Oracle Database Server provides the primary storage for all information to be stored persistently by the STEP system. Optionally, this can be an Oracle Real Application Cluster setup to compensate for the risk of failure in the underlying hardware.
- Shared Storage is a file system that is shared by all nodes in the cluster. The 'step/workarea' folder contains, for example, an image cache where thumbnails of images are generated. It also stores intermediate files generated by the STEP Workflow component and files used by background and batch processes. The hotfolders are dynamic folders that automatically process files based on a hotfolder configuration. A good example is uploading assets, where the assets are dropped into a folder and automatically imported into STEP. The hotfolders can be exposed directly on the internet making it possible to place content for import directly, for example, via FTP (not shown).



Application Server Roles

Application Servers in the cluster can be configured as a server for any of the following services:

- Background services
- Workbench client services
- Web client services

The preferred setup is that all servers handling a specific type of service handle the same set of services. For instance, a single server should not handle background processing and workbench client processing, while another server handles only workbench client processing. In this case, the load balancer may not recognize the differences between the two servers and may put equal numbers of workbench clients on both servers even though one of them also serves background services.

Implementation

Running STEP in a cluster means that one instance of the application is running on each application server. Each of the application instances maintains a cache residing in memory on the corresponding application server in order to minimize the number of database requests on repetitive read operations. Whenever a client or a background batch process changes data via one clustered application instance, all of the other application instances must know about it and update their caches accordingly. The mechanism used for this cache synchronization is implemented in the JDO (Java Data Object) layer of the application and basically relies on the same implementation no matter what application server software is being used. In other words, the cluster implementation used by STEP does not use any of the application server specific clustering facilities provided by WebSphere.

Load Balancing

As shown on the architecture diagram in **Architecture Layout** topic, the Web Client is load-balanced through a hardware load balancer. Such a load balancer must support session affinity to ensure that the same session goes to the same server until the session times out. This is important since session state is not replicated among servers and a user directed to a different server will be asked to log on again.

In contrast, the workbench client has built-in load balancing capabilities building on the CPU load on each server. Once connected to a server, the client will keep using that server unless the client GUI has been idle for 15 minutes. The standard configuration for this applies for most setups. STEP needs to be provided with the list of server names as known by clients in the STEP configuration file.

The workbench client has no session state on the server, thus moving from one server to the other is not a problem, except for performance reasons (the caches on the server the user last accessed is most likely to contain the data the user will request next). The system will ensure that any writes to the database are viewed consistently across all application servers in the cluster.

Scalability

STEP is designed to support both large, initial deployments and the growth of smaller systems through horizontal scalability. A cornerstone of the design is to provide a cost-effective solution that supports both. Regardless if the number of users, the amount of data, or both should grow significantly, STEP is not likely to be the system that limits activities.

Serious and resource-demanding tests have been and are continuously executed to prove that the scalability potential of STEP meets these goals.

High Availability Implementation

This is one of the topics that describes the architecture of the STEP solution. The full list is defined in the **STEP Architecture** topic.

STEP can be set up in a way that allows it to quickly and automatically recover from most types of failures as they occur. Running multiple application servers in a cluster is just one of the cornerstones towards achieving high availability (HA).

Elements

Technically, the key elements of a HA implementation are:

- Eliminating single points of failure.
- Applying multi-pathing by creating redundant physical path components (such as adapters, cables, switches, and interfaces) to create logical 'paths' between the server and the storage device.
- Using load balancing to ensure that servers are not overwhelmed to the point of being unable to function properly.

Considerations

Consider the following items when performing a cost / benefit analysis when considering or planning for high availability:

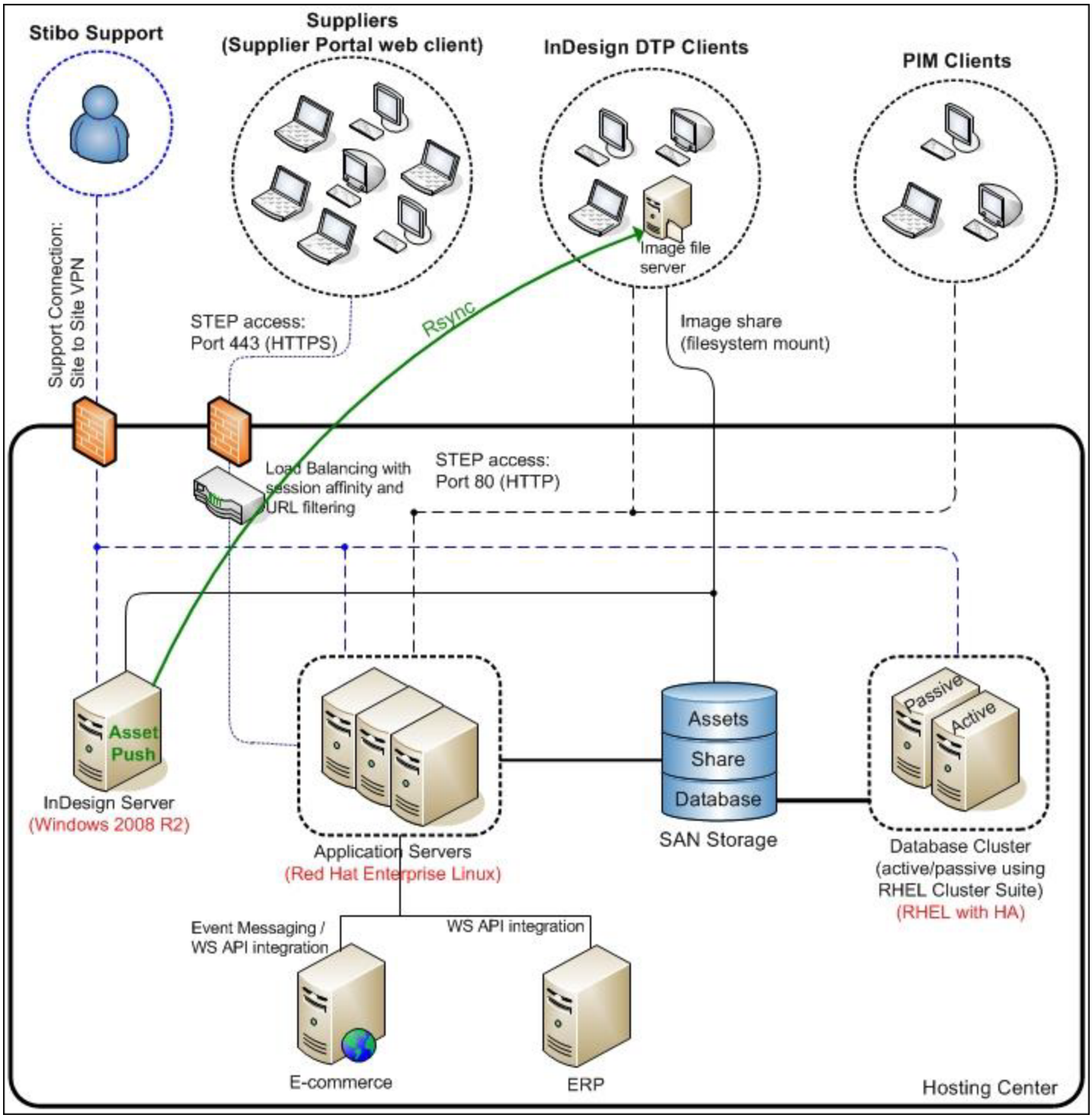
- Any HA option can lead to higher complexity, which may introduce new risks of instability and change maintenance requirements.
- Choosing a large number of smaller servers versus a small number of large servers (horizontal vs. vertical scaling).
- Business need for a level of performance while one of the cluster nodes is down.

In general, servers with more resources (faster CPU, more memory etc.) are better, but larger hardware is also more expensive. Therefore it is common to choose several smaller servers. Stibo Systems has performed scalability tests with 1-, 2- and 4-server setups that show that STEP scales almost linearly with the number of servers (horizontal scaling).

The following sections focus on HA options for each of the main software and hardware components that contribute to the broad range of services provided by STEP.

Two other factors impact high availability but are not considered in these options:

- Human failure - when a user mistakenly deletes vital parts of the data stored in the system.
- Planned downtime for system maintenance.



Application Server

If one of the application servers goes offline, STEP will continue operation on one of the other servers without needing a server restart.

Web client users may get an HTTP error (or other error) if a server is unavailable, depending on the load balancer in use. The user will have to log in again since the session state is not replicated between servers (for performance reasons). They will subsequently be able to continue working.

Workbench users may get an internal error message if a server is unavailable. After ignoring this, they can continue working. No additional log in is necessary as the workbench client is stateless, in the sense that application servers have no session state on behalf of a workbench client.

Failover handling does not handle the case where a server stops answering the client. In this case, the user must log in again (possibly after terminating their client). Upon log in, the system disregards servers not responding in a timely fashion to the log in request, but such servers will delay the log in slightly until resolved (either by taking the server down or resolving the issue).

If an application server running background services goes down, the background services will automatically be rerun when the other servers discover that the server has failed (this may take a while). No tasks are lost, but execution of the task may be delayed. In general, it is difficult to ensure that a server failure is noticed by the other servers, since it can have stopped processing the single task but keeps answering to ping requests, etc. In that case, there is no way to automatically detect the failure.

Oracle Database Server

Oracle provides these options to support clustering and/or failover of the database:

- Oracle Real Application Cluster (RAC)
- Oracle Data Guard

Oracle RAC provides fault tolerance, load balancing, and scalability. In an Oracle RAC environment, two or more computers (each with one instance) concurrently access a single database. This allows the application to connect to either computer and have access to a single, coordinated database. When one of the nodes in the cluster fails, the database is still available through one or more other nodes.

Oracle RAC is supported on both Oracle Standard Edition and Enterprise Edition.

The Standard Edition includes these limitations:

- The maximum number of CPUs defined by the license is for the entire cluster; it is not a per node maximum.
- Automatic Storage Management (ASM) based on RAW or block devices must be used to manage all database-related files. Third-party volume managers and file systems are not supported for this purpose. Stibo Systems does not provide support for ASM on RAW devices; this is the responsibility of the customer.

The Enterprise Edition does not have these limitations, therefore Stibo Systems recommends using Oracle RAC only on Oracle Enterprise Edition.

The Oracle Data Guard solution provides high availability, data protection, and disaster recovery. DataGuard uses a standby database, which is a copy of the production database. The standby database is kept up to date by applying redo log data from the production database. If the production database fails, DataGuard will switch to the standby database so this now becomes the production database.

Data Guard is only available with the Oracle Enterprise Edition. For more information, refer to the **Database Server Oracle DataGuard** topic.

Oracle and third-party software alternatives

The Oracle Enterprise Edition and the extra cost options mentioned above are expensive and other solutions exist. Using Oracle RAC provides close to 100 percent uptime, but if minimal downtime is tolerated by the business, these solutions are worth considering as they do not require the Enterprise Edition and are much less expensive to implement.

Possible solutions for implementing high availability are:

- Red Hat Enterprise Linux 8.6 with High Availability Add-On

This solution uses an active / passive setup as illustrated in the image above. If the active database node fails, the clustering functionality will failover the database storage to the passive node and start up the database instance. The passive node is now the active node.

An alternative to Oracle DataGuard for disaster recovery using a standby database is 'DBvisit standby.' For more information, refer to the web at www.dbvisit.com.

DTP Server Failover

The STEP application load balances across the DTP servers that are online. A DTP server crash affects only the users who are executing an operation against that particular DTP server. These users will subsequently be able to redo the failing operation. The STEP application will dispatch to another DTP server that is still online, and the operation can complete.

Shared Storage

Disk crashes can be handled in a RAID setup, allowing a crashed disk to be hot swapped with a new one without leading to any system breakdown. To get even better protection against unexpected accidents, an IP-Storage Area Network (IP-SAN) can establish two instances of the same storage at two different locations, thereby achieving box-to-box redundancy. One storage instance is the active one and the other one is passive. In the event of irrecoverable failure of the active system, the passive instance becomes the active one. A variety of advanced techniques are available for keeping the active and passive storage in sync.

Network Consideration

This is one of the topics that describes the architecture of the STEP solution. The full list is defined in the **STEP Architecture** topic.

Barriers set up for security reasons but that are not aligned with the STEP system requirements can result in a non-functional system or a very poorly performing system. While STEP is not particularly demanding, the following rules must be observed for the system to work properly.

Network Boundaries and Firewalls

Several distinct network realms are involved in the STEP system. It is assumed that firewalls only exist at the boundaries between the networks, so network traffic is only defined at those boundaries.

As the architecture diagram hints, most application traffic happens as plain HTTP on port 80 from workbench, web browsers, and other clients, directly to the application servers. There are other requirements for network connectivity that depend on the particular configuration of the system. Examples of such connection requirements are:

- Incoming SSH or RDP (depending on system platform) to allow administrators to manage the systems in the cluster.
- Incoming FTP for file upload if hotfolders are used.
- Outgoing HTTPS to the Stibo Systems update server, so new software can be downloaded.
- Outgoing HTTP on a configurable port to the DTP sidecar.
- File exchange with the AS2 gateway if 1SYNC integration is used.

Important: The exact configuration of the network boundaries can only be determined after the specific features needed on a particular system are known.

Contact Stibo Systems to determine the latest port requirements.

Intra-Cluster Network: One Switch, One VLAN

The application servers access the database using JDBC connections. The communication is a combination of small requests and larger batches that can take hours to process. The process of sending a lot of small requests will be very sensitive to latency in the network, while timeouts may kill the large batch jobs. Therefore, any latency, TCP timeouts, or incorrect configuration in the intra-cluster network can severely impact performance, stability, and functionality of the system, and additionally can be difficult to detect and debug. Generally, it is important to keep the network that binds the cluster together as simple as possible. The application servers and the database servers must all be located on the same switch with no filtering or routing between them.

In a standard implementation, it is assumed that no firewalls exist between the application server(s) and the database server.

Redundancy in the Network Infrastructure

The principle that simple components are less prone to failure than complex ones is important when implementing network infrastructure. For example, the network switch is not expected to fail very often, and if it does, it can be swapped out. If the network infrastructure needs to be made redundant, the standard tools (such as STP or OSPF with multiple NICs in each server) are needed. For more information, search the web.

Security

This is one of the topics that describes the architecture of the STEP solution. The full list is defined in the **STEP Architecture** topic.

This section briefly covers the main aspects of security in relation to STEP.

Authentication

Authentication of users in STEP can be configured in the following ways:

- Default user database maintained internally in the STEP database
- LDAP
- LDAP with Kerberos

The LDAP authentication methods allow STEP to be integrated with any existing user authentication infrastructure. The advantages are that users do not need to maintain an extra password for STEP and that the password policies of the central authentication mechanism are inherited to STEP.

Data Protection

The access to data in STEP is under fine-grained control by the application servers and is configured via the workbench client. User actions are set up and attached to user groups and the specific users then get permissions due to membership of one or more of these groups. Granting of permissions is positive in the sense that the user actions specify what the users are allowed to do (not what they are not allowed to do). If a user is a member of two different user groups with overlapping sets of privileges, the resulting set of privileges will be the union of the two groups. For more information, refer to the **Users and Groups** section of **System Setup** documentation.

Network Security Considerations

The STEP cluster consists of the Oracle database server and the application servers which must be on the same physical network with no firewalls between them. For more information, refer to the **Network Consideration** topic.

Local OS Security

As with any secure system, only trusted users should be granted access to any of the servers in the cluster (Oracle database and the application servers). This protects against local privilege escalation attacks which can be numerous and hard to guard against, but also limits the possibility for performance interference.