StiboSystems

StiboSystems

STEP Performance

Assessment and Recommendations

VERSION: 3.0

AUTHOR: Stibo Systems Professional Services

CONFIDENTIALITY LEVEL: Confidential

Content

1 Document Control

1.1 Document purpose

This document describes the STEP performance assessment, analysis and recommendations for [CLIENT].

The performance guidelines and recommendations are based on the facts and figures gathered and analysis done by Stibo Systems for the configuration and implementation of a STEP system optimized for performance.

All analysis and recommendations in this document apply to the [CLIENT] STEP PRODUCTION environment.

Stibo Systems recommends [CLIENT] to take corrective actions based on the performance assessment results from this document.

1.2 Standard performance assessment process

Normally, the standard performance assessment process is as follows:

1. The performance assessment process starts with an agreement between the client and Stibo Systems to execute a performance assessment process.
2. Stibo Systems sends the template for identifying the main performance problems, non-functional performance requirements and performance use cases to the client for the client to fill out. Please use appendix D Template for main performance problems and non-functional requirements for this purpose.
3. An online call of approximately 2 hours is scheduled to discuss the main performance problems, non-functional performance requirements and performance use cases the client had filled out with the goal to fully understand the performance problems and be able to reproduce these.
4. Stibo System uses the filled-out template to reproduce the performance problems offsite and performs the performance analysis based on this document. This initial performance analysis offsite takes about 3 days.
5. Stibo Systems has an onsite visit with client for 2 days to
    - Present the initial performance analysis done by Stibo Systems offsite
    - Determine the most important recommendations with client
    - Implement the most important recommendations with the client as far as feasible

- Execute the defined performance use cases again and discuss possible performance improvements with the client
- Determine next steps with the client

6. Stibo System finalizes the performance assessment by finalizing the execution of the performance use cases and writing the conclusions this document. The finalization takes about 1 day.
7. Stibo System hands over the document to the client in an online call of about 2 hours.

It takes about a month in throughput time at minimum to finalize the standard performance assessment process from performance requirement and use case definition to delivery of the performance findings and recommendations.

1.3 Causes of bad performance

A perceived bad performance of the STEP system is very seldom the result of one single cause. Usually a wide range of improvements in a variety of areas will together result in an improved performance of the STEP system.

Therefore, it's impossible to predict which single performance recommendation will result in a certain improvement percentage of performance of the STEP system.

1.4 How to read and use this document

This document is used during the performance assessment and handed over to [CLIENT] after finalization. The performance assessment document includes the following sections:

- Summary of most important performance related findings and recommendations for the client
- Non-functional requirements and performance use cases of the client
- Performance assessment with performance related findings and recommendations for the client
- Appendices with analysis methodologies and recommendations in general (non-client specific)

1.4.1 Summary of most important performance related findings and recommendations for the client

The document starts with a summary of the most important issues and recommendations, which are the issues and recommendations with expected relative high positive impact on performance and / or expected relative low effort on executing.

1.4.2 Non-functional requirements and performance use cases of the client

If available, the non-functional requirements and performance use cases are listed. The use cases describe the reproduceable situation where perceive bad performance occurs.

Normally, the use cases are executed at the start of the assessment and the response times are logged before any performance improvement recommendations are implemented (initial). The same use cases are executed again after the assessment and after the performance improvement recommendations are implemented (final). The results of the performance improvements are visible when comparing both initial and final use cases.

1.4.3 Performance assessment with performance related findings and recommendations for the client

The performance assessment tasks to execute are stated in the following chapters

- Chapter 4: overall STEP health check
- Chapter 5: from a technical environment / infrastructure viewpoint
- Chapter 6: from a functional viewpoint.

The performance assessment tasks, and corresponding analysis and recommendations are condensed into a table. Every line in the table is a performance assessment task. The tables are structured as follows:

- Task: Short description of the performance assessment task to be executed.
- How to reference: Reference to the detailed description of the task in the appendices A and B. Here the performance assessment task is described in detail, together with the reasoning why it may impact performance, how to execute the task, what's normal or what it should be, and which corrective actions can be taken to improve the performance.
- Baseline: Short description of what is normal / standard.
- Analysis: Short description of the result of the executed task (outcome of the analysis). Optionally, references to the detailed descriptions of the analysis can be added in appendix C.
- Recommendation: Short description of the recommendation in case the outcome of executed task doesn't meet the baseline (corrective action).
- Urgency: An indication whether the corrective action (recommendation) will have a significant performance improvement or a less significant performance improvement.

Here's an example of a performance assessment task as stated in the table.

Task	How to reference	Baseline	Analysis	Recommendation	Urgency

Check client computer CPU	A.2.1 Client computer	The CPU of the client computers should not reach 100% when working with STEP Workbench and / or WebUI		Not STEP related. Setup computers according to the Infrastructure Recommendation document.	

The client specific performance findings and recommendation details are described below the table in each paragraph.

1.4.4 Appendices with analysis methodologies and recommendations in general (non-client specific)

The remaining part of the document includes appendices describing the details of the analysis methodologies and recommendation in general.

The appendices are general (non-client specific) and give the insights and background on how to analyze the STEP system and what the good practices are in in general. The examples in the appendices are therefore generic and non-client specific as well.

1.5 Document versions

Version	Date	Author	Comment

1.6 Document distribution

Version	Date	Send to

2 Summary

[SUMMARY TO BE FILLED OUT AFTER THE PERFORMANCE ASSESSMENT]

3 Non-functional requirements

This chapter describes the non-functional requirements, based on the top most important use cases, and performance testing which has or has not been executed. It also provides an overview of the number of objects and attributes in the STEP environment.

3.1 Non-functional requirements

This paragraph provides an overview of the non-functional requirements regarding performance. The non-functional requirements should indicate what is expected by the client from a STEP performance perspective. The non-functional requirements should have been defined and agreed upon in an earlier stage.

The performance related non-functional requirements are listed in the table below.

Nbr.	Non-functional requirement	Date	Agreed by
NFR1			
NFR2			
NFR3			

For more information and examples, please refer to appendix D Template for main performance problems and non-functional requirements

3.2 Use cases

This paragraph provides an overview of the reproduceable use cases which show whether the non-functional requirements are met or not.

The most important use cases are described which are reproduceable and proof a performance degrade.

Each use case is related to a non-functional requirement.

Each use case is executed before any performance improvement recommendation is executed (as-is situation). After the performance recommendations are executed, each use case is executed again to proof a performance improvement.

For more information and examples, please refer to appendix D Template for main performance problems and non-functional requirements

3.2.1 Use case 1

Non-functional requirement: ….

Description use case: ….

Environment: ….

3.2.1.1 Initial execution

Execution of the use case before performance improvement recommendations are processed.

Date and time of execution: ….

Nbr.	Task executed to reproduce	Expected response time	Actual response time	Pass / Fail	Remarks
US.1.1					
US.1.2					
US.1.3					

3.2.1.2 Final execution

Execution of the use case after performance improvement recommendations are processed.

Date and time of execution: ….

Nbr.	Task executed to reproduce	Expected response time	Actual response time	Pass / Fail	Remarks
US.1.1					
US.1.2					
US.1.3					

3.2.2 Use case 2

Non-functional requirement: ….

Description use case: ….

Environment: ….

3.2.2.1 Initial execution

Execution of the use case before performance improvement recommendations are processed.

Date and time of execution: ….

Nbr.	Task executed to reproduce	Expected response time	Actual response time	Pass / Fail	Remarks
US.2.1					
US.2.2					

US.2.3					

3.2.2.2 Final execution

Execution of the use case after performance improvement recommendations are processed.

Date and time of execution: ….

Nbr.	Task Executed to Reproduce	Expected Response Time	Actual Response Time	Pass / Fail	Remarks
US.2.1					
US.2.2					
US.2.3					

3.2.3 Use case 3

Non-functional requirement: ….

Description use case: ….

Environment: ….

3.2.3.1 Initial execution

Execution of the use case before performance improvement recommendations are processed.

Date and time of execution: ….

Nbr.	Task Executed to Reproduce	Expected Response Time	Actual Response Time	Pass / Fail	Remarks

US.3.1					
US.3.2					
US.3.3					

3.2.3.2 Final execution

Execution of the use case after performance improvement recommendations are processed.

Date and time of execution: ….

Nbr.	Task Executed to Reproduce	Expected Response Time	Actual Response Time	Pass / Fail	Remarks
US.3.1					
US.3.2					
US.3.3					

3.3 Performance testing

This paragraph describes if performance testing has been executed and whether or not the performance testing results indicated a performance degrade.

The most important information about the performance tests are listed in the table below.

Nbr.	Performance test	Date execution	Results and issues found
PT1			
PT2			
PT3			

For more information and examples, please refer to appendix D Template for main performance problems and non-functional requirements

3.4 Volume

This paragraph provides an overview of the number of objects in STEP. Please refer to appendix B.1.6 Global count of objects and attributes to see how these statistics can be gathered.

3.4.1 Products

Overview of number of product objects.

Product object	Environment	Date of count	Count
Products			

3.4.2 Entities

Overview of number of entity objects.

Entity object	Environment	Date of count	Count
Entities			

3.4.3 Assets and attributes

Overview of total number of assets and attributes.

Type	Environment	Date of count	Count
Assets			
Attributes			

4 Health Check

There's an option in the Admin Portal of STEP version 8.1 and higher to check the health of the STEP system. Before any other analysis are done and recommendations are followed up, it's recommended to run the STEP health checks one by one. The methods to gather the performance data from the STEP Health Check are described in appendix C Analysis methodologies and recommendations via STEP Health Check.

After the initial health checks are executed, the issues identified should be exported to a CSV file and embedded in this document in appendix C Analysis methodologies and recommendations via STEP Health Check.

Since some of the issues found may not be related to performance, and because some of the issues found may be resolved after following up the recommendations stated in this document, the embedded CSV file with the issues found serve the purpose of bringing them to light.

After the recommendations stated in this document are followed up, a final STEP Health Check should be executed by running the health checks once again. After these final health checks are executed, the issues identified are be exported again to a CSV file and embedded as a second health check file in this document in appendix C Analysis methodologies and recommendations via STEP Health Check.

Task	Reference	Environment	Date	Issues found
Initial STEP Health Check	C Analysis methodologies and recommendations via STEP Health Check			• … • … • …
Initial STEP Health Check	C Analysis methodologies and recommendations via STEP Health Check			• … • … • …

5 Technical environment viewpoint

When STEP users suffer poor performance, especially in operation on STEP production, it's necessary to check the performance of STEP from a technical infrastructure viewpoint, which includes the client computers, internet connection and the STEP servers and the STEP server network.

The methods to gather the performance data from a technical infrastructure viewpoint, as well as the detailed description of the recommendations with examples are described in appendix A - Analysis methodologies and recommendations from technical viewpoint.

5.1 Client environment

The client workstations are analyzed together with the internet connection of the client computers and connection between client computers and the STEP application server.

It's the responsibility of the client to make sure the client computers and internet connections are well performing.

Task	Reference	Baseline	Analysis	Recommendation	Urgency
Check client computer CPU	A.2.1 Client computer	The CPU of the client computers should not reach 100% when working with STEP Workbench and / or WebUI		Setup computers according to the Infrastructure Recommendation document.	
Check client computer memory	A.2.1 Client computer	The memory of the client computers should not reach 100% when working with STEP Workbench and / or WebUI		Setup computers according to the Infrastructure Recommendation document.	

Check client internet browser response	A.2.2 Client internet browser	The internet browser of the client computers should respond swiftly in a normal way		Update the existing internet browser, disable browser plug-ins, or install another browser.	
Check client internet browser compatibility	A.2.2 Client internet browser	The browser and browser version should meet the STEP infrastructure recommendation document, usually all browsers are supported apart from Internet Explorer versions below 9.		Use browsers according to the Infrastructure Recommendation document.	
Check client internet connection speed via ping test	A.2.3 Client internet connection	The difference in ping response times between a random website and the STEP application server should not differ much, and below 200 milliseconds		Switch to wired network when wi-fi is used.	
Check client internet connection speed via Workbench	A.2.3 Client internet connection	The internet connection should be below 200 milliseconds		If response times are below 200 milliseconds and wi-fi is used, then switch to wired network	
Check internet connection setup	A.2.3 Client internet connection	The internet connection should meet the STEP infrastructure recommendation document, usually 1 GB to servers in normal environments and 10 GB in enterprise environments		If internet connection is slow and wi-fi is used, then switch to wired network	

5.2 STEP server environment

The STEP server environments are analyzed via basic analysis on the STEP application server as well as the STEP database server.

The STEP server environments should be setup according to the delivered STEP Infrastructure Recommendation document. If the STEP Infrastructure Recommendation document is outdated, then it should be updated and the STEP servers should be setup accordingly. The STEP server configurations should be according to the recommendations.

For deeper system and infrastructure analysis, recommendations and execution, it's highly recommended to involve a Stibo Systems System Architect (Infrastructure Architect) together with the System Architect of the client.

Task	Reference	Baseline	Analysis	Recommendation	Urgency
Check STEP Infrastructure Recommendation document	A.1 STEP infrastructure recommendation document	The STEP servers are setup according to infrastructure recommendation document		Setup STEP servers according to infrastructure recommendation document	
Update STEP Infrastructure Recommendation document	A.1 STEP infrastructure recommendation document	In case the STEP load is significantly increased, then update STEP infrastructure recommendation document		Update STEP infrastructure recommendation document and Setup STEP servers accordingly	
Check STEP server configurations	A.3.1 STEP server configuration file	The STEP server configurations in the shared configuration properties file hasn't been changed without consulting Stibo Systems		Configurations in shared configuration properties file are according to Stibo Systems recommendations	

Check STEP server virtualization	A.3.2 STEP server virtualization	The STEP production environment should not be virtualized without consulting Stibo Systems		No virtualization of STEP production environment without consulting Stibo Systems	
Check STEP server virtualization setup	A.3.2 STEP server virtualization	In case the STEP production environment is virtualized, then it should be virtualized according to virtualization recommendations		Virtualization of STEP production environment according to the provided documentation	
Check STEP folders in shared locations	A.3.3 STEP folders as shared locations	The STEP environment should not use shared folders on other servers		No usage of shared folders without consulting Stibo Systems	
Check STEP folders in shared locations latency	A.3.3 STEP folders as shared locations	In case the STEP environment uses shared folders on other servers, then the latency should meet the recommendations described in infrastructure recommendation document		Consult Stibo Systems to setup shared folders with optimal latency	
Check CPU and Memory load on STEP servers	A.3.4 STEP server load	The STEP servers should have normal CPU and Memory load, otherwise this should be analyzed further by Stibo Systems and probably the heap size should be increased		Consult Stibo Systems to normalize CPU and Memory load	
Check CPU and Memory load on STEP servers are well balanced	A.3.4 STEP server load	The STEP servers should have balanced CPU and Memory loads, otherwise this should be analyzed further by Stibo Systems and probably the load		Consult Stibo Systems to balance CPU and Memory load in cluster environment	

		balanced configuration should be reconsidered			
Check STEP server network latency	A.3.5 STEP server network latency	The network latency of the STEP server environment should be stable and about 0.2 milliseconds		Consult Stibo Systems to optimize the STEP server environment network latency	
Check STEP database redo log	A.3.6 STEP database server redo log	The STEP database should perform a redo log switch of about 50 times per day (every 30 minutes)		Consult Stibo Systems to minimize the number of redo log switches	
Check STEP database alert log	A.3.7 STEP database server alert log	The STEP database alert log should be analyzed on errors such as TEMP tablespace errors and deadlock errors		Consult Stibo Systems to analyze the STEP database alert log further	
Set STEP logfile detail properly	A.3.8 STEP server logfile settings	Set the logfile details on the STEP environment according to recommendations		Set the logfile detail level on STEP production to only log severe problems	
Check STEP server components	A.3.9 STEP components	The STEP server components should be the same in every STEP environment (DTAP)		Differences in STEP baseline components on the separate environments (DTAP) can lead to differences in behavior	High
Check STEP server components	A.3.9 STEP components	No unnecessary STEP server components should remain after system upgrade		Unnecessary components can lead to strange server behavior	High
Oracle Adaptive Plans	A.3.10 Oracle Adaptive Features	This will show up as Oracle using “adaptive plans”, when you display execution plans for SQL		Known issue also at Oracle (known bug). Parameter should be	High

		queries. Oracle chooses to use an “directive”, which for some queries means a very bad execution plan, and response times goes from milliseconds to seconds.		set to FALSE in Oracle version 12.1.0.2.	
Oracle data statistics	A.3.11 Oracle Statistics	After upgrade the Oracle data statistics should be renewed for STEP		Collect a complete new set of statistics for all indices / tables in STEPSYS schema	High
Oracle DataGuard	A.3.12 Oracle DataGuard	Running with DataGuard however introduces an overhead on the STEP system and may case performance issues		Recommended is to run tests where the protection level of the DataGuard is lowered and performance results are measured	High

5.3 STEP server hygiene

The STEP server environments are analyzed via basic analysis on the STEP application server as well as the STEP database server.

The STEP server environments should be setup according to the delivered STEP Infrastructure Recommendation document. If the STEP Infrastructure Recommendation document is outdated, then it should be updated and the STEP servers should be setup accordingly. The STEP server configurations should be according to the recommendations.

For deeper system and infrastructure analysis, recommendations and execution, it's highly recommended to involve a Stibo Systems System Architect (Infrastructure Architect) together with the System Architect of the client.

Task	Reference	Baseline	Analysis	Recommendation	Urgency
Check STEP application server disk space	A.4.1 STEP application server disk space	All the STEP application servers have enough disk space left		Cleanup files on STEP application servers (see next)	
Check number and size of import files	A.4.2 Clean up unnecessary import files	The number of other unnecessary files should not exceed the hundreds of thousands and the files sizes should be less than gigabytes.		Check the root cause and cleanup import files (see next)	
Check root cause of too many and/or too big import files	A.4.2 Clean up unnecessary import files	Most common reasons for too many import files are - IIEP is configured to keep import files are import - IIEP suffers from failed or completed with errors import processes - IIEP suffers from non-standard directory structure		Fix root cause of too many import files on STEP application servers	

Cleanup too many and/or too big import files	A.4.2 Clean up unnecessary import files	The number of other unnecessary files should not exceed the hundreds of thousands and the files sizes should be less than gigabytes.		Remove import files from file system (in case IIEP is configured to keep import files) or remove background processes.	
Cleanup too many and/or too big other files	A.4.3 Clean up other unnecessary files	The number of other unnecessary files should not exceed the hundreds of thousands and the files sizes should be less than gigabytes.		Find and remove other unnecessary files from file system	
Check number of background processes	A.4.4 Clean up background processes	The number of background processes should not exceed the hundreds of thousands.		Check the root cause and configure auto-delete of background processes (see next)	
Check root cause and configure auto-delete of background processes	A.4.4 Clean up background processes	Most common reasons for too many background processes is - Many failed or completed with errors background processes - Wrong configuration of auto-delete of background processes		Fix the root cause why many background processes fail and set the auto-delete configuration properly	
Cleanup too many background processes	A.4.4 Clean up background processes	The number of background processes should not exceed the hundreds of thousands.		Remove background processes from STEP database and file system	

6 Functional viewpoint

When STEP users suffer poor performance, especially in operation on STEP production, it's necessary to check the performance of STEP from a functional viewpoint which includes the setup and configuration of the STEP application.

The methods to gather the performance data from a client's technical environment viewpoint are described in appendix B - Analysis methodologies and recommendations from a functional viewpoint.

6.1 Base configurations

The base setup of the STEP contexts and dimensions, product and entity data model, assets and classifications are designed based on business requirements. Nevertheless, the base setup should be designed based on performance as well since a complex base setup can have a negative impact on the performance in general.

An analysis of the base setup is therefore worthwhile, however changes on the base setup to improve performance does sometimes have a huge impact. For example, changing the data model might have impact on attributes, references, assets, but also on business rules, workflows, WebUI configurations, exports and imports. It's therefore highly recommended to always weight the pros and cons before changing the base setup.

Task	Reference	Baseline	Analysis	Recommendation	Urgency
Check need of dimensions	B.1.1 Dimensions and contexts	Use as little as dimensions as possible (max. two, e.g. language and country)		Use as little as dimensions as possible (max. two, e.g. language and country)	

Check dimension dependencies	B.1.1 Dimensions and contexts	Only configure dimension dependency when required		Only configure dimension dependency when required	
Check data model	B.1.2 Data model	Design for simplicity		Keep the data model as simple and meaningful as possible	
Check data model	B.1.2 Data model	Consciously use Manually Sorted		Avoid using Manually Sorted when not required	
Check asset storage method	B.1.3 Assets	The asset storage methodology doesn't have impact on performance as such		Make a conscious decision on storing assets on file system or in the database	
Check dimension dependency of assets	B.1.3 Assets	Only use dimension dependency for assets when required		Make a conscious decision on asset dimension dependency	
Check asset transformations at import and export	B.1.3 Assets	Only use asset transformations when required		Make a conscious decision on asset transformations	
Check for hierarchies with many children	B.1.4 Classifications	Avoid hierarchies with many children		Use a more granular hierarchy structure	

Check usage of references	B.1.5 References	Be careful with navigation via references in business rules to resolve values, resolving calculated attributes or resolving references during export		Consider avoiding the usage of these references as much as possible	

6.2 Attribute and attribute configurations

There are attribute and attribute group configuration which might have influence on the STEP performance.

Task	Reference	Baseline	Analysis	Recommendation	Urgency
Configuration of the display of attribute groups	B.2.1 Disable the display of attribute groups	Displaying many attribute values, especially of calculated attributes, may have impact on performance		Disable display of attribute groups where possible	
Configuration of manually sorting of attributes	B.2.2 Disable manually sorted	Processing manually sorted attributes may have impact on performance when used extensively		Disable manually sorting of attributes where possible	
Configuration of full text indexing of attributes	B.2.3 Disable full text indexing where possible	When an attribute is made as Full Text Indexable, there are some chances of having performance issues.		Disable Full Text Indexing of attributes where possible	

Check for need of dimension dependency of attributes	B.2.4 Disable dimension dependency where possible	Only use dimension dependency for attributes when required		Make a conscious decision on attribute dimension dependency	
Check for LOVs with too many values	B.2.5 Avoid using large lists of values	Avoid usage of LOVs with thousands of values, also from a usability perspective		Consider splitting up large LOVs or consider using another attribute value type	
Check for calculated attributes in cross-contexts exports	B.2.6 Avoid using calculated attributes	Avoid usage of calculated attributes in cross-contexts exports.		Consider evaluating the necessity for the calculated attribute in the cross-context export, or consider using "copy fields"	
Check the locality of calculated attributes	B.2.6 Avoid using calculated attributes	Be careful with calculated attributes that navigate to other objects via references and / or hierarchy links, especially which go across many children		Consider using business actions instead	
Check whether the calculated attributes depend on other calculated attributes	B.2.6 Avoid using calculated attributes	Avoid the usage of calculated attributes which depend on other calculated attributes		Consider using more efficient business actions to come to the same result	

6.3 Optimistic locking

STEP runs with optimistic locking policy. Optimistic locking errors may be caused by long transactions of e.g. imports, exports, asset push, business rules. The longer a transaction, the higher the probability of introducing an optimistic locking failure when running long transactions simultaneously.

STEP will put the transaction on hold when optimistic locking occurs, and will try to process the transaction again after some time. Optimistic locking errors therefore degrade the performance of imports, exports, asset push, business rules, etc. and degrade the performance of the STEP system.

Task	Reference	Baseline	Analysis	Recommendation	Urgency
Check optimistic locking	B.3 Optimistic locking	Optimistic locking failures due to long transactions degrade performance		Optimize long transactions by optimizing the business rules and functions	
Check optimistic locking	B.3 Optimistic locking	Optimistic locking failures due to long transactions degrade performance		Avoid optimistic locking failures by the Reference Target Lock Policy to Relaxed	

6.4 Data profiling

Use data profiling only when required. If used, then optimize the memory usage for data profiling.

Task	Reference	Baseline	Analysis	Recommendation	Urgency

Configuration of data profiling	B.4 Data profiling	When large categories are profiled, the system uses a lot of memory.		Disable data profiling where possible	
Optimize memory usage when using data profiling	B.4 Data profiling	When large categories are profiled, the system uses a lot of memory		Optimize memory usage for data profiling	

6.5 WebUI configurations

The WebUI Designer is flexible and can configure a WebUI in many ways. However, not all configurations perform equally well. Therefore, it is important to consider performance when configuring a WebUI.

Task	Reference	Baseline	Analysis	Recommendation	Urgency
Multiple WebUIs	B.5.1 Use multiple WebUIs	A rule of thumb is to avoid WebUIs with more than 25k XML lines.		Split up the WebUI into multiple WebUIs, each with their own specific purpose	
Small dedicated WebUI screens	B.5.2 Use small dedicated WebUI screens	A rule of thumb is to avoid using attribute groups with more than 100 attributes in WebUI screens.		Configure attribute groups specifically for WebUI display	

Images in multi-select screens	B.5.3 Avoid using images in multi-select WebUI screens	A rule of thumb is not to use thumbnails in multi-select screens with many items.		Reconsider using thumbnails in multi-select screens with many items	
Lazy loading of WebUI screens	B.5.4 Use Lazy Loading for WebUI screens	With lazy loading, components will be 'lazy loaded' from STEP and rendered only when needed		Activate lazy loading for each tab page	
Type-ahead for large LOVs in WebUI	B.5.5 Use Type Ahead for LOVs in WebUI screens	Use type-ahead for LOVs with many values in WebUI		Configure the type-ahead options to make large LOVs typed-ahead	
Configuration of Status Selectors in WebUI	B.5.6 Configure Status Selectors in WebUI correctly	Configure refresh interval of status selectors to 60 seconds or more		Set the refresh interval on 60 seconds (default) or more	
Configuration of Status Selectors in WebUI	B.5.6 Configure Status Selectors in WebUI correctly	Rule of thumb is not to include more than 25 status selectors on one WebUI screen		Keep the number of status selectors to a minimum (less than 25).	

6.6 Business rules

The flexibility of business rules comes a high level of responsibility. The system load of the business rule execution may have effect on the performance of STEP. If there're a lot of business rules running and if the business rules are complex, then this will degrade performance simply because STEP is very busy processing these business rules.

Task	Reference	Baseline	Analysis	Recommendation	Urgency
Check business conditions usage	B.6.2 Do not update data via business conditions	Wrong usage of business conditions versus business actions may impact performance.		Do not use business conditions to manipulate data or implement side effects in a business condition.	
Keep business action transactions small	B.6.3 Keep business rule transactions small	Long business rule translations may lead to optimistic locking failures and degrade of performance		Business rule transactions must be kept small and be developed full control of even the worst-case scenario.	
Keep business action transactions small	B.6.3 Keep business rule transactions small	Long business rule translations may lead to optimistic locking failures and degrade of performance		Traversal of a substantial percentage of the complete data range should be avoided, and changes to data must be kept local (to the nearest surrounding of your main data object).	

Keep business action transactions small	B.6.3 Keep business rule transactions small	Long business rule translations may lead to optimistic locking failures and degrade of performance		Consider setting `Reference Target Lock Policy` to `Relaxed` if the business rule transactions cannot be simplified and optimistic locking failures still frequently occur	
Avoid large business rule libraries	B.6.4 Avoid large business rule libraries	As a rule of thumb, it takes about 500 milliseconds to compile about 8,500 lines of code at each business rule execution		Keep business rule libraries small, or consider making the library functions local to the business rule itself	
Avoid infinitive loops	B.6.5 Avoid infinitive loops	Infinitive loops will lead to severe degrade of the performance		Avoid infinitive loops in business rules	
Apply exception handling properly	B.6.6 Apply exception handling	Incorrect exception handling has a negative effect on the performance of the business rule		Use correct exception handling where exceptions are re-thrown in the “try-catch” used in business rules	
Set business logging	B.6.7 Set business rule logging	Logging many business rule details unnecessary may have a negative impact on the performance		Configure the business rules logging settings in the configuration file	
Set business logging	B.6.7 Set business rule logging	Logging many business rule details unnecessary may have a negative impact on the performance		Set a logging debug flag in the business rules code itself	

Avoid get children on many nodes	B.6.8 Avoid the function GetChildren with many nodes	Fetching a huge number of children may cause memory problems		Do not use business rules to fetch data of more than 10.000 children	
Avoid get children on many nodes	B.6.8 Avoid the function GetChildren with many nodes	Fetching a huge number of children may cause memory problems		If required to fetch selections with more than 10.000 children, then do not use “getChildren” but use “queryChildren”	
Use arrays instead of multiple read calls	B.6.9 Use arrays instead of multiple read calls	Fetching the same data again and again is not efficient and leads to performance loss		Fetch the data once from the database and use arrays instead of fetching the same data again and again from the database	
Consider using extensions as last resort	B.6.10 Consider using an extension	Some of the code in JavaScript (business rules) might run faster in Java (extensions).		Consider moving the logic to extensions as last resort	

6.7 Imports

Bulk data imports into STEP may can vary from very fast to slow, highly depending on the expectations and on following up the good practices for imports described in this paragraph. A clean, simple, optimized import without any processing may perform about a hundred records per second, whereas a cluttered, complex import with complex processing may perform one record per second or even slower.

If the STEP system suffers from perceived bad performance during import, then these recommendations should be followed up.

Please be aware that optimizing the import performance, may have an adversely affect the performance of the system as a whole.

Task	Reference	Baseline	Analysis	Recommendation	Urgency
Import errors	B.7.1 Avoid import errors	Import errors heavily influence import performance negatively		Analyze and fix any import errors	
Missing targets in import files	B.7.2 Import file - Avoid missing targets	Missing reference targets in import files trigger a second import pass		Make sure the import file doesn't contain missing targets	
Forward declarations in import file	B.7.3 Import file - Avoid forward declarations	Forward declarations of reference targets in import files trigger a second import pass		Make sure the import file doesn't contain forward declarations	
Multiple updates of same object in import file	B.7.4 Import file - Avoid multiple updates of same object in one import file	Updating same object multiple times in one import file degrades performance		Structure the import files by modified nodes, and not by attribute	
Updates of same node in multiple import files	B.7.5 Import file - Avoid updates of same object over multiple import files	Updating same node in multiple import files will cause duplicate execution of business rules, approvals and cache load and potential dramatically increase revision history		Update the same node with all relevant information in one file, rather than splitting it over multiple files or imports	

Terms List for prices on import	B.7.6 Import file - Use Term Lists for price data	Use Terms Lists for price data		Use Terms Lists for price data	
Business rules necessity on import	B.7.7 Business rules on import - necessity	Any business rule processing on imports will have a performance price		Consider if it's necessary to have all or even any actions immediately on import	
Optimistic locking due to business rules on import	B.7.8 Business rules on import - avoid optimistic locking	Optimistic locking failure will cause performance degrades. 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.		Avoid business actions that execute on tree structures or commonly referenced objects may cause optimistic locking failures	
Optimistic locking due to business rules on import	B.7.8 Business rules on import - avoid optimistic locking	Optimistic locking failure will cause performance degrades. 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.		Business action that triggers an approval inside of a try/catch block where the approval condition failure is caught and the original exception is not re-thrown may cause optimistic locking failures	
Optimistic locking due to business rules on import	B.7.8 Business rules on import - avoid optimistic locking	Optimistic locking failure will cause performance degrades. 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.		Any business action that throws an exception with the actual word "exception" in the error message will cause the importer to believe that an optimistic lock occurred, resulting in retries and eventual failure	

Business rules performance on import	B.7.9 Business rules on import - design for performance	Ensure any business rules running on import has acceptable performance		Avoid complex business rule JavaScript logic being executed on the import	
Business rules performance on import	B.7.9 Business rules on import - design for performance	Ensure any business rules running on import has acceptable performance		Avoid business rules that read or update objects other than the one being imported	
Business rules performance on import	B.7.9 Business rules on import - design for performance	Ensure any business rules running on import has acceptable performance		Avoid business conditions allowing for a logical exception or lock contention	
Business rules performance on import	B.7.9 Business rules on import - design for performance	Ensure any business rules running on import has acceptable performance		Avoid duplicate executions of logic	
Workflow initiations on import	B.7.10 Design workflow initiations on import for performance	Workflow initiation or a state transition triggered by an import results in execution of business rules configured on exit of an existing state, transition between as part of import process		Consider business rules trigger on workflow transitions	
Workflow initiations on import	B.7.10 Design workflow initiations on import for performance	Workflow initiation or a state transition triggered by an import results in execution of business rules configured on exit of an existing state, transition between as part of import process		Consider if these business rules initiate more workflows or auto submit to other states and avoid cascade executions of business rule logic	

Workflow initiations on import	B.7.10 Design workflow initiations on import for performance	Workflow initiation or a state transition triggered by an import results in execution of business rules configured on exit of an existing state, transition between as part of import process		Avoid workflow business rule conditions to fail which results in the current transaction to be rolled back	
Approvals on import	B.7.11 Design approvals on import for performance	Approvals on import may trigger additional business rules and exports which may degrade performance		Consider if approvals are necessary at import	
Approvals on import	B.7.11 Design approvals on import for performance	Approvals on import may trigger additional business rules and exports which may degrade performance		Consider which approval conditions and actions need to be executed	
Approvals on import	B.7.11 Design approvals on import for performance	Approvals on import may trigger additional business rules and exports which may degrade performance		Try to ensure that endpoints are importing externally-maintained data	
Triggering of OIEPs on import	B.7.12 Design event-driven exports for performance	Large number of Outbound Integration End Points listening on attribute groups may can lead to performance degradation		Consider which approval events are queued on Outbound Integration End Points	
Triggering of OIEPs on import	B.7.12 Design event-driven exports for performance	Large number of Outbound Integration End Points listening on attribute groups may can lead to performance degradation		Consider which event filter / generator rules will be triggered	
Triggering of OIEPs on import	B.7.12 Design event-driven exports for performance	Large number of Outbound Integration End Points listening on attribute groups may can lead to performance degradation		Ensure that the Outbound Integration End Points listen to as specific and few attributes as possible.	

Import privileges	B.7.13 Avoid complex privileges at import	When importing, all privileges are checked, for each piece of information imported		Ensure that any imports happen as a user with as broad and few privileges as possible to avoid excessive privilege-checking	
Parallel imports	B.7.14 Use parallel imports for the same Inbound Integration End Point	Parallel imports might result in performance improvements		Consider parallel imports of the same IIEP	
Migration scenarios	B.7.15 Migration scenarios	Consider migration scenarios separately		Attempt to avoid using rules that read from or write to many related objects or children	
Migration scenarios	B.7.15 Migration scenarios	Consider migration scenarios separately		Business rules should, wherever possible, only transform data on the object being imported	
Migration scenarios	B.7.15 Migration scenarios	Consider migration scenarios separately		With serial endpoints, attempt to load products in the smallest number of import files possible	
Migration scenarios	B.7.15 Migration scenarios	Consider migration scenarios separately		If necessary, parallelize the migration endpoint so that multiple concurrent background processes can be utilized	
Migration scenarios	B.7.15 Migration scenarios	Consider migration scenarios separately		Avoid optimistic locks or deadlocks and consider using two passes to rectify this	

Migration scenarios	B.7.15 Migration scenarios	Consider migration scenarios separately		Be cautious of business rules that update products other than the one being loaded and updates to execute the logic after import is complete	
System setup for imports	B.7.16 Setup STEP for maximum import performance	Optimize setup of STEP for import performance		Ensure that any imports happen as a user with as broad and few privileges as possible	
System setup for imports	B.7.16 Setup STEP for maximum import performance	Optimize setup of STEP for import performance		Ensure that there is only a limited number of Outbound Integration Endpoints are listening on change-events	
System setup for imports	B.7.16 Setup STEP for maximum import performance	Optimize setup of STEP for import performance		Make sure to set “Relaxed” locking policy for commonly referenced	
System setup for imports	B.7.16 Setup STEP for maximum import performance	Optimize setup of STEP for import performance		Try to ensure that endpoints are importing externally-maintained data as much as possible	
System setup for imports	B.7.16 Setup STEP for maximum import performance	Optimize setup of STEP for import performance		Avoid Optimistic Lock Failures by ensuring not to update the same product at the same time across multiple imports	

System setup for imports	B.7.16 Setup STEP for maximum import performance	Optimize setup of STEP for import performance		Use parallel imports as needed, but be aware of potential performance degradation across the rest of the system.	
Miscellaneous import remarks	B.7.17 Miscellaneous import related optimizations	Miscellaneous import related optimizations		Remove inactive and unused IIEPs	
Miscellaneous import remarks	B.7.17 Miscellaneous import related optimizations	Miscellaneous import related optimizations		Set IIEPs to remove files and background processes after import	
Miscellaneous import remarks	B.7.17 Miscellaneous import related optimizations	Miscellaneous import related optimizations		Use the standard asset importer	
Miscellaneous import remarks	B.7.17 Miscellaneous import related optimizations	Miscellaneous import related optimizations		Make sure assets and other objects are not imported into one huge folder	
Miscellaneous import remarks	B.7.17 Miscellaneous import related optimizations	Miscellaneous import related optimizations		Balance the import schedules of the different imports to balance the STEP system load	
Miscellaneous import remarks	B.7.17 Miscellaneous import related optimizations	Miscellaneous import related optimizations		Validate the hardware sizing and consider In-Memory to improve import performance	

6.8 Exports

One of the primary goals of most STEP implementations is to reduce the time to market requiring the flow of data from STEP to external systems such as ERP systems and e-commerce systems.

If the STEP system suffers from perceived bad performance during export, then these recommendations should be followed up.

Please be aware that optimizing the export performance, may have an adversely affect the performance of the system as a whole.

Task	Reference	Baseline	Analysis	Recommendation	Urgency
Event-based exports	B.8.1 Use event based exports over static exports	Event-based exports allow for exporting updates rather than full data sets		Use event-based exports over static selection-based exports where possible.	
Number of OIEPs	B.8.2 Limit the number of Outbound Integration End Points	An excessive amount of event based OIEPs can adversely affect the performance of the system as a whole.		Rule of thumb is having no more than 10 event-based OIEPs	
Volume of data exported	B.8.3 Limit the volume of exported data	Define to export exactly what's needed to optimize export performance		Use Advanced STEPXML exports and limit the exported data set	
Triggering definitions for object types	B.8.4 Optimize the triggering definitions with object types	Export relevant data only to optimize export performance		Tailoring an OIEP's triggering definitions on for relevant objects changes only	

Triggering definitions for attributes and references	B.8.5 Optimize the triggering definitions with triggering attributes and references	Export relevant data only to optimize export performance		Tailoring an OIEP's triggering definitions for relevant attribute and reference changes only	
Triggering definitions event filter and generator	B.8.6 Optimize the triggering definitions with event filter and event generator	Export relevant data only to optimize export performance		Consider using event-filtering and event-generation	
Multiple OIEPs	B.8.7 Consider using multiple dedicated OIEPs	Export relevant data only to optimize export performance		Avoid too much data to be exported via using multiple OIEPs with one for listening for traditional product (attribute) changes and another for listening for reference changes and outputting minimal data	
Multithreaded exports	B.8.8 Consider use multithreading	Optimize export performance via multithreading		Use multithreading when there is a large amount of data is exported, but test before use	
Export batch size	B.8.9 Optimize the batch size	Small batch-sizes invoke the export engine and surrounding framework many times incurring a significant overhead		Avoid small batch sizes	

Cross-context exports	B.8.10 Use cross-context exports	Export cross-context STEPXML on an endpoint, rather than having multiple endpoints for each context		Use cross-context exports over multiple OEIPs for each context	
Including additional data in exports	B.8.11 Limit the inclusion of additional data	Additional data can be included in the export via an event generator, a pre-processor, or via Advanced STEPXML template		Consider the impact of adding additional objects as adding many additional objects will slow the export process	
Multiple templates per OIEP	B.8.12 Limit the usage of multiple templates per OIEP	Multiple templates allow to change how data is exported for different kinds of objects		Using multiple templates in one OIEP may have a severe impact on performance as each time a new output template is required a new batch is created	

6.9 Scheduled processes

The scheduled processes might have an influence on the performance of the system.

It's recommended is to analyze the scheduled processes and see if the collections (searches) can be optimized and / or the business rules can be optimized.

Task	Reference	Baseline	Analysis	Recommendation	Urgency

Check scheduled processes	B.9 Scheduled processes	Scheduled processes might have an influence on the performance of the system		Analyze the scheduled processes and see if the collections (searches) can be optimized	
Check scheduled processes	B.9 Scheduled processes	Scheduled processes might have an influence on the performance of the system		Analyze the scheduled processes and see if the business rules can be optimized	

6.10 Revision control

All major data objects in STEP including products, assets, classification, and entities are under revision control. This means that you can go back in history and view and compare old revisions, and revive an older version of an object.

Many revisions however may have a negative impact on the performance of the system, and therefore it's key to keep the number of revisions under control and remove unnecessary revisions.

Task	Reference	Baseline	Analysis	Recommendation	Urgency
Set revision threshold	B.10.1 Setting the revision threshold	Avoid excessive creation of the revisions, which may have a negative effect on the system performance		Keep the threshold for revisions on the default setting (168 hours)	
Revision policy and baseline	B.10.2 Purge old revisions manually to create a baseline	Avoid keeping unnecessary revisions since many revisions may have a negative effect on the system performance		Define a revision policy and create a revision baseline by purging unnecessary revisions	

Automatically purge revisions	B.10.3 Automatically purging revisions periodically	Keep revision generations under control since many revisions may have a negative effect on the system performance		Define an auto-purge event processor per object type to automatically purge revisions periodically	
Revisions on Integration End Points	B.10.4 Revisions on Integration End Points	End Points may generate many revisions when these are configured wrongly which may have a negative effect on the system performance		Analyze if the Integration End Points have many revisions, and if so remove these revisions and make sure the revisions are not created automatically anymore.	

6.11 Privileges

STEP has a very granular privilege system and the privilege setups which makes the configuration of user access and rights flexible.

However, running STEP as a user with a large number of very specific privileges will influence the performance of basically any action in STEP that goes across a large number of nodes, values, or references, including export, import, bulk update, recursive approval, matching, and "multi views" like task list and multi editor.

Task	Reference	Baseline	Analysis	Recommendation	Urgency
Privilege configuration	B.11.1 Privilege configurations	The privileges of STEP are additive only, so whenever a basic action is executed, STEP		Do not overdo the privilege setup to avoid excessive privilege checking	

		looks for the first privilege that provides the permission. - Small list of privileges means that the privilege check is cheap - Large list of privileges is specific and granular and the search for the appropriate privilege takes longer - If the user tries to do something without having the privilege, is the most expensive check of all.			
Privileges at import and export	B.11.2 Privileges at imports and exports	When importing or exporting data, all privileges are checked, for each piece of information imported or exported		Ensure that any imports / exports happen as a user with as broad and few privileges as possible to avoid excessive privilege-checking	
Privileges in WebUI	B.11.3 Privileges in the WebUI	Privilege checking via the WebUI XML configuration may degrade performance due to extensive privilege checking		Avoid configure privilege checks in WebUI XML configuration file	
Privileges in WebUI	B.11.3 Privileges in the WebUI	A typical WebUI screen fetches all attribute values and then filters out the attributes based on validity of for the product type and based on user privileges.		Avoid excessive privilege checks for WebUI users	

6.12 Searches

STEP searches can lead to a degraded performance when the recommendations are not followed up.

Task	Reference	Baseline	Analysis	Recommendation	Urgency
Specific searches	B.12.1 Use specific search criteria	Optimize search criteria for optimized search performance		Use specific search criteria as much as possible	
Wildcards in searches	B.12.2 Searches with wildcards	Optimize search criteria for optimized search performance		Use as many characters as possible before using wildcards	
Wildcards in searches	B.12.2 Searches with wildcards	Optimize search criteria for optimized search performance		Avoid using wildcard as first character	
Full Text Indexable	B.12.3 Full Text Indexable	Optimize search criteria for optimized search performance		Avoid enable "Full Text Indexable" attributes unless necessary	
Main object types as search criteria	B.12.4 Main object types	Optimize search criteria for optimized search performance		Use the main object types as search criteria	
Basic searches	B.12.6 Basic searches	Optimize search criteria for optimized search performance		Specify the Attribute, ID or NAME in search criteria	

Advanced searches	B.12.7 Advanced searches	Optimize search criteria for optimized search performance		Avoid using "Include Inherited Values" where possible	
Advanced searches	B.12.7 Advanced searches	Optimize search criteria for optimized search performance		Avoid using "Regular Expression" where possible	

6.13 Matching and linking

The flexibility of matching and linking comes a high level of responsibility. The system load of this functionality may have effect on the performance of STEP.

Task	Reference	Baseline	Analysis	Recommendation	Urgency
Approval of golden record	B.13.1 Approval of the golden record	Approval of the golden record as a survivorship rule may degrade the performance of matching and linking process		Separate the approval of the golden record from the matching and linking process if possible	
Multi-context survivorship rules	B.13.2 Multi-context survivorship rules	Multi-context survivorship rules continue in all contexts		Avoid using the multi-context survivorship rules as much as possible	

A Analysis methodologies and recommendations from technical viewpoint

This appendix describes the data gathering methodologies and performance improvement recommendations from a technical infrastructure viewpoint including the client computers, internet connection and STEP application and database server setup and connections.

A.1 STEP infrastructure recommendation document

Stibo Systems has handed over the STEP infrastructure recommendation document. The STEP infrastructure recommendation document describes how the STEP environments should be setup.

[EMBED THE CLIENT INFRASTRUCTURE RECOMMENDATION DOCUMENT]

The STEP environments should be setup according to this document provided by Stibo Systems, to be able to meet the performance and scalability requirements.

The document should be updated when the load of the STEP environments has significantly increased after the STEP infrastructure recommendation has been delivered. A significant increase of the load includes for example, a significant increase of number of users using STEP, or a significant increase of the number of objects in the STEP system.

A.2 STEP client environment

This paragraph describes the data gathering methodologies from a technical infrastructure viewpoint for the STEP clients using the STEP Workbench and STEP WebUI.

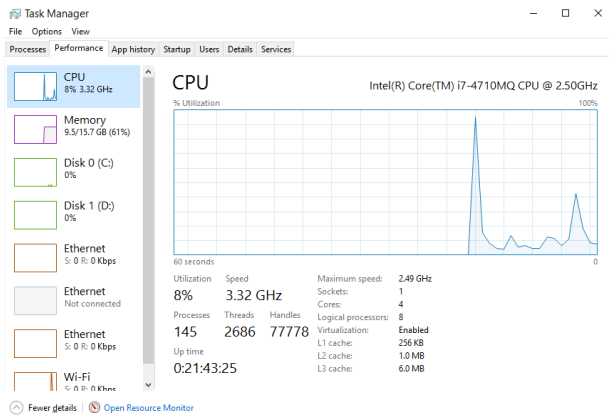
A.2.1 Client computer

It has happened before that users experienced bad performance of the STEP Workbench and STEP WebUI caused by client computers working slowly.

The client computer performance can be checked by the client himself in Windows via

- Pressing `CTRL-ALT-DEL`
- Click on `Task Manager`
- And analysing the client computer CPU and Memory

If the CPU and / or memory is high, for instance near to 100%, then the client's computer is slow in general. The STEP Workbench and WebUI will not perform well if the client's computer is slow.



A.2.2 Client internet browser

It has also happened before that users experienced bad performance of the STEP Workbench and STEP WebUI caused by bad performing internet browsers.

The internet browser performance can be checked by using the internet browser (e.g. Internet Explorer, Chrome, Firefox, Safari) and see if other websites are performing well or not.

- Client to open the internet browser such Internet Explorer, Chrome, Firefox, or Safari.
- Client to browse to websites such as www.google.com and some other websites.

If the internet browser is slow, possibly due to all kinds of plug-ins due or because of an old version of the browser, then the client can update the existing internet browser, disable browser plug-ins, or install another browser and try it again.

Also check the browser and versions against infrastructure recommendation document.

A.2.3 Client internet connection

It has also happened before that users experienced bad performance of the STEP Workbench and STEP WebUI caused by slow or unstable internet connections.

The internet connection can be checked by:

- Check the internet connection by executing a “ping” and compare the response times between a known website and the STEP server.

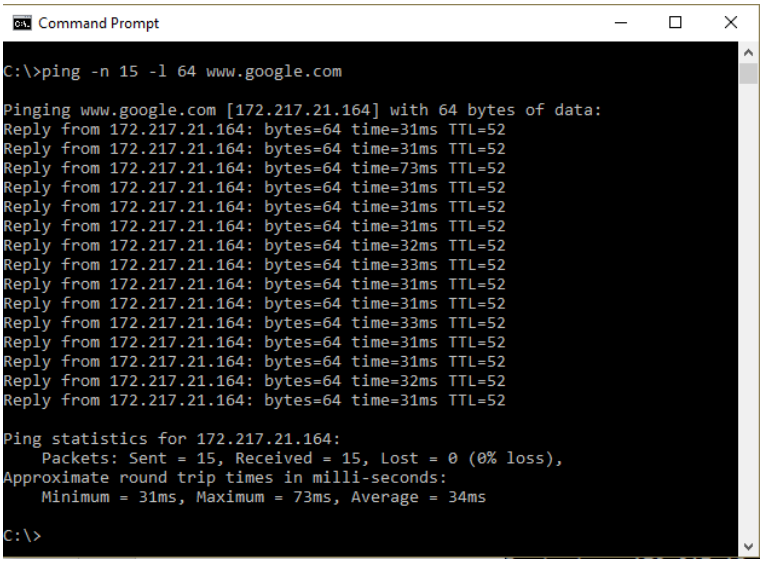
  Execute Ping in Windows via command prompt (start > cmd) and type for example:

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

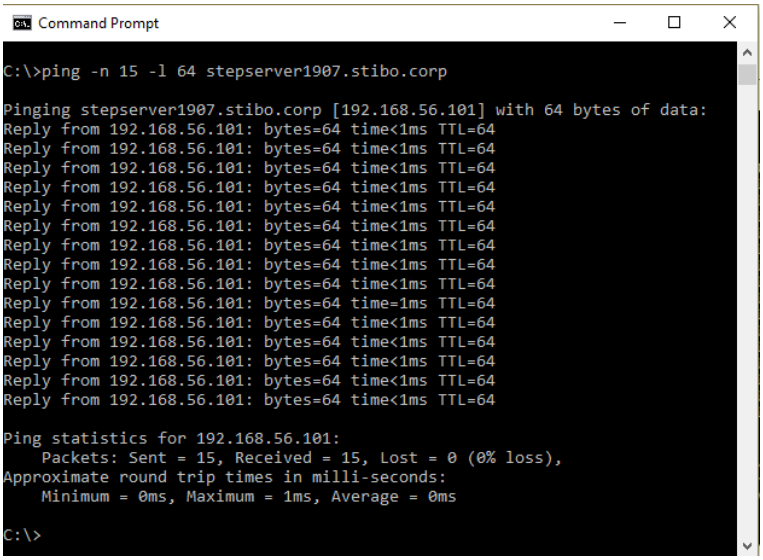
  Remarks:

  - Parameter `-n 15` means 15 echo requests instead of default 4
  - Parameter `-l 64` means 64 bytes requests instead of default 32 bytes

  Ping a website, e.g.: `ping -n 15 -l 64 www.google.com`

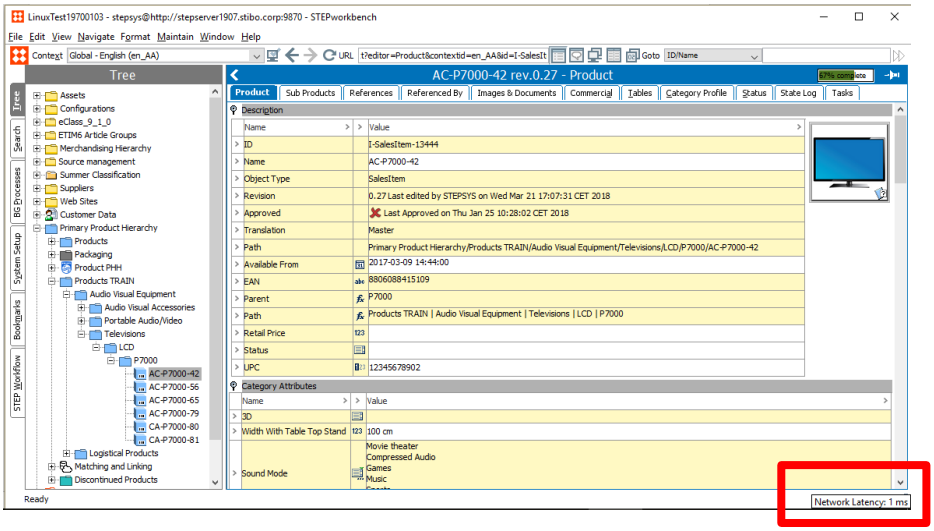


  Ping the STEP application server, e.g.: `ping -n 15 -l 64 stepserver1907.stibo.corp`



The connection response time between the client computer and the STEP application server shouldn't differ too much with the connection response time between the client computer and a random website like google.com.

- Alternatively, check the internet connection by looking up the response time of the between the client computer and the STEP application server in the STEP Workbench.
    - Open the STEP Workbench and login
    - Hoover over the lower right corner of the Workbench
    - The network latency will become visible



The internet connection determines the performance of the STEP WebUI and Workbench. Therefore, the response times of the ping and / or network latency check via the Workbench should meet normal internet connection figures.

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

Please be aware you can estimate an additional 1 millisecond in network latency for every 100km the data travels. So, if the STEP application server is in Venlo NL and the client is in Seattle USA, then a response time of about 30 milliseconds from client to STEP application server in NL may can correspond with a response time of about 110 milliseconds in from client in Seattle USA to same STEP application server in Venlo NL (distance is about 8.000 kilometres).

The easiest way to improve bad internet connection is to connect via a wired connection (cable) instead of wireless Wi-Fi. If the client still encounters bad internet connection with response times frequently above 200 milliseconds, then the company's internet connection should be improved.

A.3 STEP server environment

This paragraph describes the data gathering methodologies from a technical infrastructure viewpoint for the STEP servers.

A.3.1 STEP server configuration file

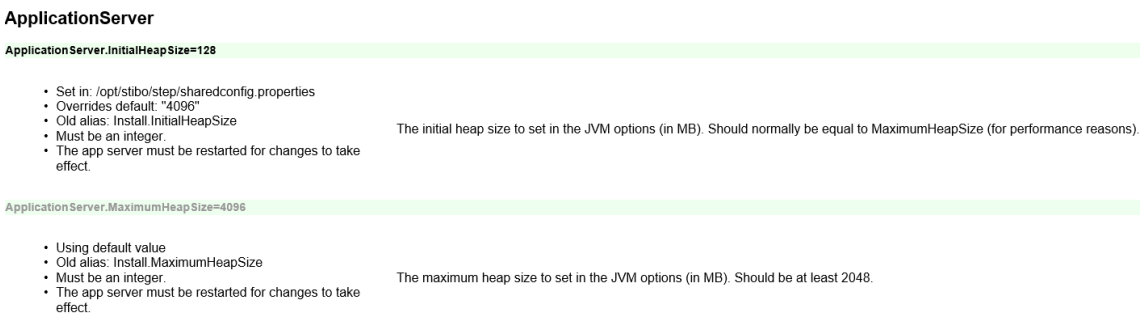
The STEP server configurations in `sharedconfig.properties` should not be changed without good reasons and without consulting Stibo Systems.

For example, the initial and maximum heap size is set by Stibo Systems and these configurations should remain exactly as defined by Stibo Systems.

Check STEP server configurations in `sharedconfig.properties` by locating the shared configuration `sharedconfig.properties` file and view the server configurations.

Alternatively, view the STEP server configurations via the STEP Admin Portal

- Open the STEP Admin Portal
- Click `Configuration`
- Select the STEP application server and click `Fetch Data`
- Go to the `ApplicationServer` and view the application server configurations. Here, the active configurations are marked in black where the inactive configurations are greyed out.



A.3.2 STEP server virtualization

Stibo Systems do support virtualization of non-production environments on VMware or Hyper-V on x86-64 based platforms.

However, virtualization of production database and application server environments is on a case-by-case basis on system load and sizing expectations, and is conditional upon a performance test conducted by

the parties and finally approved by Stibo Systems in its discretion. Peaks in performance often is caused by other systems eating up resources in a virtualized environment.

The white paper for virtualization on VMware is attached.

VMware White Paper.pdf

A.3.3 STEP folders as shared locations

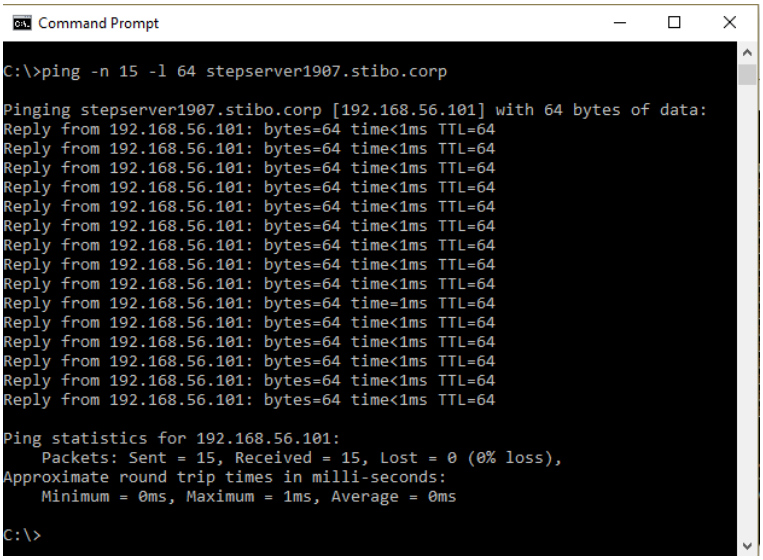
It has happened that shared locations on other servers are used as STEP folders. For instance, the hotfolders for the import files are located on a different server apart from the STEP application server.

Placing STEP files on shared locations can be very unpredictable, depending on the latency of the network. Therefore, if STEP files are on shared locations, then this infrastructure setup should be described in the infrastructure recommendation document, where the latency of the network should meet the recommendations stated in this document.

Check the network latency from the STEP application server(s) to the server where the shared file location is configured, by executing a “ping” command.

Execute Ping from the STEP application server via command prompt

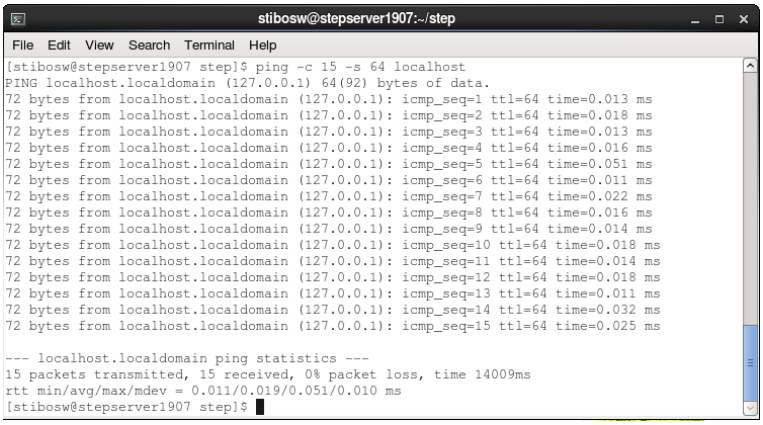
- If the STEP application server is a Windows environment, for example: `ping -n 15 -l 64 shared.file.location`
    - Parameter `-n 15` means 15 echo requests instead of default 4
    - Parameter `-l 64` means 64 bytes requests instead of default 32 bytes



- If the STEP application server is a Linux environment, for example:

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

  - Parameter `-c 15` means 15 echo requests instead of default 4
  - Parameter `-s 64` means 64 bytes requests instead of default 32 bytes



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

Please be aware you can estimate an additional 1 millisecond in network latency for every 100 km the data travels. For example, let's assume the STEP application server and STEP 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 is however located in Seattle USA, which is 8000 km

away. In this case an additional 80 milliseconds needs to be added to the network latency resulting in terrible decrease of the optimal network latency of 80,2 milliseconds instead of 0,2 milliseconds.

A.3.4 STEP server load

Via the STEP Admin Portal, the load of the systems in the cluster can be analyzed.

When the STEP server load is high, then the performance is degraded and further actions are required.

Furthermore, when the STEP environment is a cluster environment, and one of the systems in the cluster are not balanced, then performance is degraded as well and further actions are required.

The CPU and Memory load and distribution of the STEP Application Servers can be analyzed via

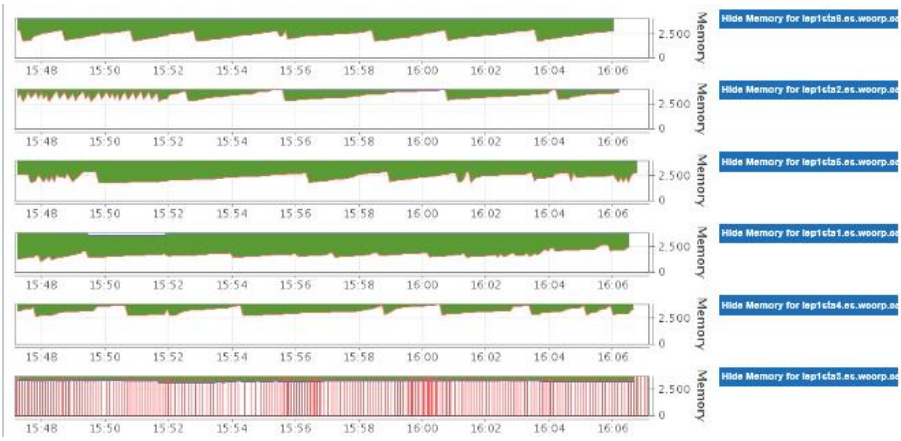
`STEP Admin Portal > Activity > Fetch Data.`

Examples:

- The CPU load of the STEP servers may look like the following graphic.

  In this example, the CPU load 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.

- The Memory load of the STEP servers may look like the following graphic.



In this example, it's noticed that the memory load is unbalanced as well 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 combined with maybe a too small heap-allocation. So, in this example it's recommended to analyze the heap size further and see if it's configured correctly. Furthermore, there seems to be incorrect load balancing between the servers in the cluster for the background processes. The system selects which of the STEP application servers should handle the background processes. Therefore, it's also recommended to analyze the load balancing more.

A.3.5 STEP server network latency

The flexibility of STEP gives rise to a potentially large number of ad-hoc requests between the STEP application server and the STEP database server. Network latency and bandwidth between the STEP application and database servers are therefore important.

Furthermore, the ability to synchronize across the STEP application server cluster is crucial. In a STEP environment with two STEP application servers, a data change done via one application server needs to be reflected as close to instantly as possible or otherwise "optimistic locking" errors might occur.

Although the network recommendations are stated in the Infrastructure Recommendation document, it's of added value to repeat these again:

- Network bandwidth requirements
    - Minimum requirement is 1 Gigabit dedicated network between all STEP servers except to file servers.
    - Recommended requirement is 10 Gigabit for enterprise systems with a heavy workload profile.
- Network latency requirements:

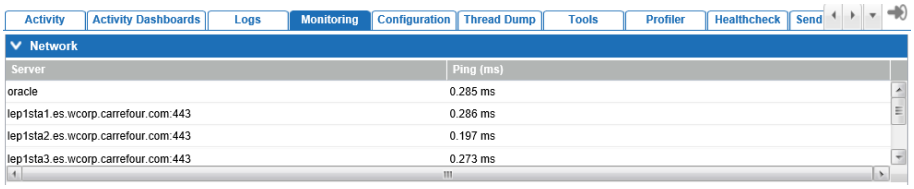
- Requirements for network latency between STEP application servers in the application cluster:
  - Maximum 0.2 milliseconds with a 64 bytes packet size
  - Maximum 1 millisecond with a 32K packet size
- Requirements for network latency between STEP application servers and database server are also:
  - Maximum 0.2 milliseconds with a 64 bytes packet size
  - Maximum 1 millisecond with a 32K packet size

The STEP network latency can be analysed as follows:

- An instant indication of the network latency between the STEP application and database servers can be seen via the Monitoring tab in the Stibo Admin Portal. This option provides a near real-time check of the latency between the servers.

  STEP Admin Portal > Monitoring.

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

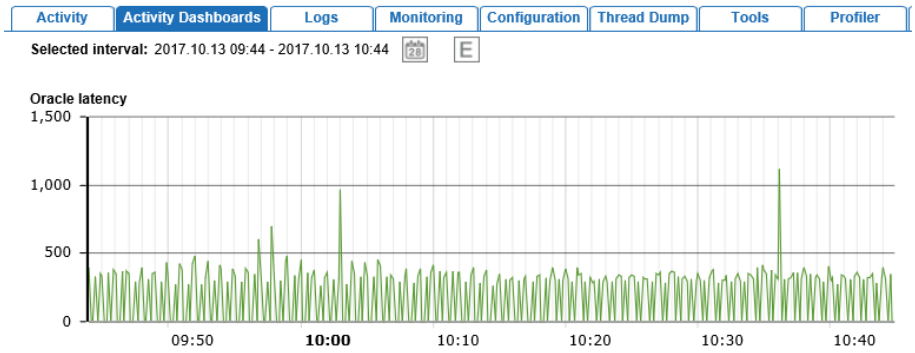


- An historic network latency analysis is also done via the STEP Admin Portal. This graphic provides the average time it takes to execute 10 very simple database requests. If the latency numbers here are high, this could be an indication of network issues, but could also be an indication of the database server having trouble keeping up with the number of requests.

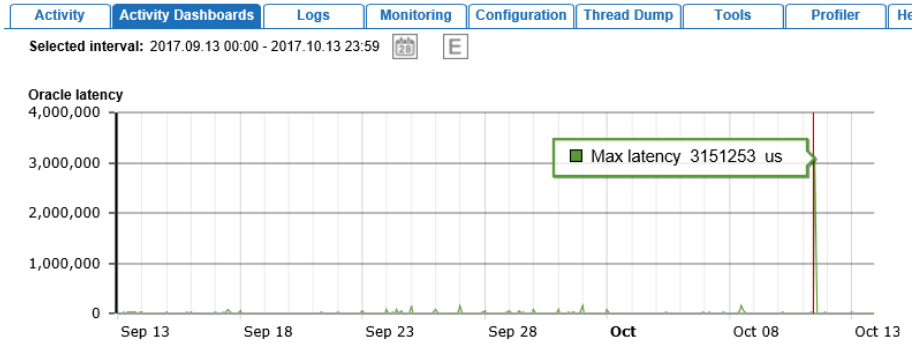
  STEP Admin Portal > Activity Dashboard > System.

  The following examples provides historic insights in the network latency

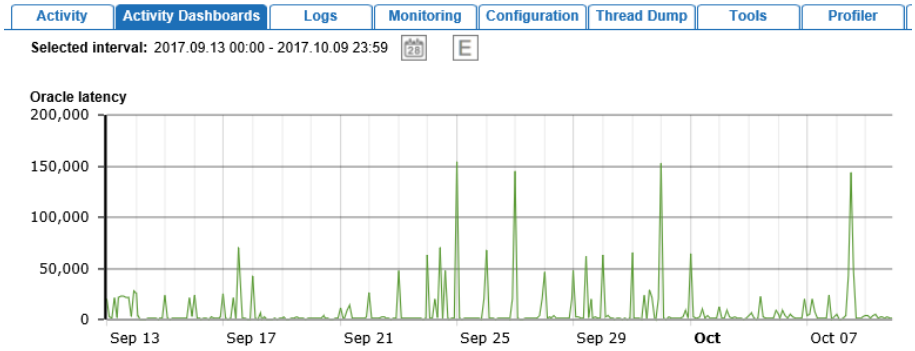
  - The 1st figure with time interval “last hour” shows that latency has a couple of spikes near to the 1 millisecond, which is something to be concerned about.



- The 2nd figure with time interval “last thirty days” shows a huge spike around October 10 of about 3 seconds which is concerning.



- And the 3rd figure with time interval before that huge spike shows many other spikes frequently above the 50 milliseconds, which is concerning as well.



STEP performance is negatively impacted when the network latency is not stable and frequently above the required 0.2 milliseconds. Also “optimistic locking” errors might occur. In those cases, the network latency should be optimized.

A.3.6 STEP database server redo log

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 whilst performing the checkpoint. Therefore, the general performance of the database is impacted if the redo log switch is performed too often.

The redo log file of the database can be analyzed via the DB ToolBox.

- Go to the DB ToolBox which is normally in "`/opt/stibo/step/admin/app-server-toolbox/`".
- Start the ToolBox via ./`db-toolbox.pl` or via `perl db-toolbox.pl`
- Execute DB ToolBox number 42: "Show redolog history"

Example:

- The redo logfile 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
```

- And the statistics of the number of redo log switches per day is 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 database performs a redo log switch about 2 to 3 times per minute in several days over the last 28 days. That impacts the general performance of the database, as every log switch triggers a database checkpoint, which again brings the database to a halt whilst performing the checkpoint. Furthermore, over the last 28 days, the redo log switch varies from about 20 times per day to about 500 times per day.

- As a rule of thumb, the database should perform a redo log switch of about 50 times per day, instead of about 500 times per day, and the redo log switches per day should be quite stable.

- In this example, it's recommended to increase the size of the redo log files, to minimize the number of redo log switches to e.g. one every half hour. This 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 file every 30 minutes.

    This will lower the number of times the database will perform a redo logfile switch, which brings the database to a halt whilst performing the checkpoint less, and therefore has a positive impact on the general performance of the database.

A.3.7 STEP database server alert log

The database alert log can be analyzed via DB ToolBox, if the alert log feature is enabled in the database.

- Go to the DB ToolBox which is normally in "`/opt/stibo/step/admin/app-server-toolbox/`".
- Start the ToolBox via ./`db-toolbox.pl` or via `perl db-toolbox.pl`
- Execute DB ToolBox number 36: "Show DB alert-log"

Example

- An example of the database alert log is attached.

DBToolBox-DBAlerts.txt

- The database alert log shows 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.

- The size of the TEMP tablespace can be found with DB ToolBox.
  - Go to the DB ToolBox which is normally in "`/opt/stibo/step/admin/app-server-toolbox/`".
  - Start the ToolBox via ./`db-toolbox.pl` or via `perl db-toolbox.pl`
  - Execute DB ToolBox number 1: "Show tablespace usage report"

```
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.
```

Current size of TEMP tablespace is one file of 32 GB.

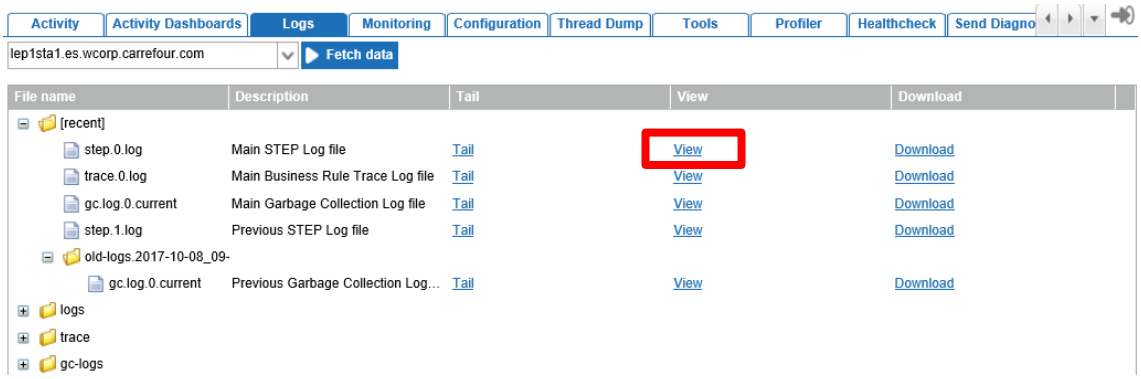
- Therefore, in this example, it's recommended to add at least one extra file of 32 GB, so the total size of the TEMP tablespace will be increased to 64 GB.
- Furthermore, in this example, other database alter log errors may include for instance
    - ORA-00060 errors: deadlock detected
    - ORA-01555 errors: snapshot too old

  Therefore, in this example, it's recommended to involve a Stibo Systems System Architect to analyze, recommend, and execute on the database alert errors, as well as generic on infrastructure and system.

A.3.8 STEP server logfile settings

If the system isn't working correctly for some reason, one of the most important tools for finding out what's going on is the logging subsystem. STEP uses powerful logging subsystem, so understanding it is important for getting the most use out of it.

The logfile is viewable via the STEP `Administration Portal > Logs > Fetch data > Main STEP Log File > View`.

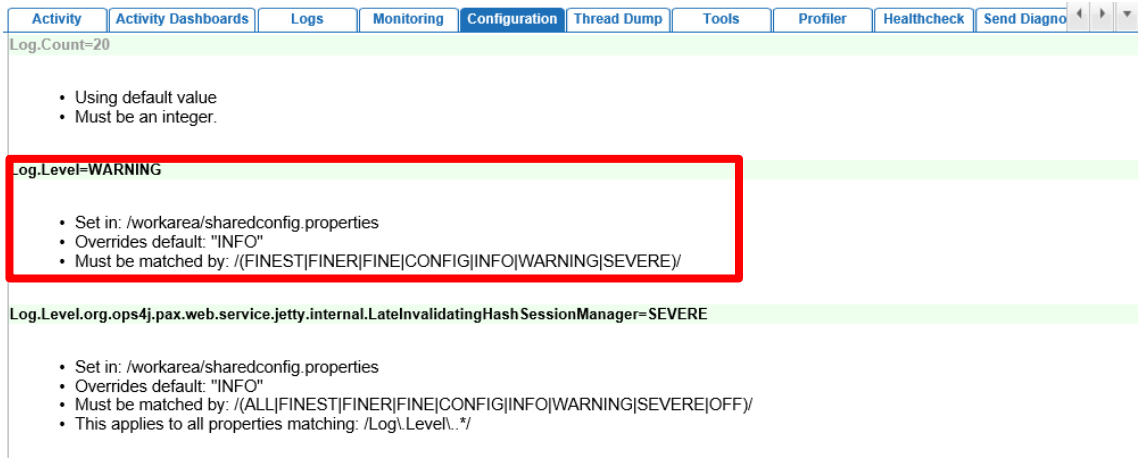


There's a STEP configuration property called `Log.Level` in the `sharedconfig.properties` or `config.properties` which controls the level of detail that logs produce. The level can be quite high so that only the most important messages are logged or it can be low to get more detail. Any level on the list implicitly logs everything above that level as well. There are several levels of logging information:

- SEVERE: Errors where the program is unable to recover and continue in a normal manner, such as a workflow looking for a property that doesn't 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 which would generally be of interest to the end user (the default).
- CONFIG: Messages which aren't generally of interest to the end user but are 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.

Log level detail FINE, FINER, and FINEST can create huge numbers of log entries in a short period of time, so they aren't typically used for the whole program.

Also, the log level configuration can also be viewed via STEP `Administration Portal` > `Configuration`.



Furthermore, it's also possible to set logging by package in a similar manner. The `Log.Level` property can be broken down into increasingly fine-grained parts of the whole by adding the name of the package to be logged to the end.

Example:

- The setting `Log.Level.com.stibo.centralhotfolder=FINE` will set the level of everything under `com.stibo.centralhotfolder` to FINE.
- In the event of an error, the log will commonly have a stack trace of some sort. The stack trace data is mostly of interest to developers, but they're good indicators of what are the most interesting log messages.
- The following is an example log entry with a stack trace. Real log entries and stack traces may look somewhat different and won't have identifying information trimmed out, but this is at least illustrative.

```
Feb 18, 2009 6:28:33 PM com.stibo.servicemanager.beans.CentralHotFolder doRun

SEVERE: Unable to handle [Trimmed] exception message Couldn't rename [Trimmed] to
[Trimmed]

java.lang.RuntimeException: Couldn't 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 data 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. In this example, the cause was one file not being able to be renamed to another file name and the error happened in a specific customer hotfolder.

The logfile settings typically differ per environment.

- The logfile detail level on a STEP development and test environment is typically set highly detailed to identify potential issues during development and testing.

- The logfile detail level on a STEP production environment is typically set far detailed to only log the severe issues during normal operation. This also avoid the risk of clogging the STEP production system with large logfiles.

Therefore, the following logfile settings are recommended

- Set the log level details on STEP DEV and STEP TEST to a debug level such as FINE to trace errors. In case there's an error and the logfile on STEP DEV doesn't provide much detail, it's recommended to log in a more granular way.

- Set the log level details on STEP QA to INFO or WARNING.

- Set the log level details on STEP PROD to SEVERE.

A.3.9 STEP components

The STEP components include

- Core functionality: The STEP version, e.g. STEP 7.5, STEP 8.2, etc.

- Hotfixes (optionally)

- Monthly Patches (optionally)

- Custom Extensions (optionally)

For example: run "./spot --components"

```
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
```

There're two things to consider

- 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, then make sure the baseline is the same on all environments.

- There might be unnecessary components on the STEP system after upgrades.

  If might well be that after an upgrade, some of the hotfixes are not needed anymore. Make sure to read the released notes carefully and be sure that all unnecessary components are removed.

A.3.10 Oracle Adaptive Features

Known bug in Oracle version 12.1.0.2. where Oracle chooses to use an "directive", which for some queries means a very bad execution plan, and response times goes from milliseconds to seconds.

This has quite an unfortunate impact on performance and is known issue also at Oracle (known bug). This will show up as Oracle using "adaptive plans", when you display execution plans for SQL queries.

The parameter should be set to FALSE in Oracle version 12.1.0.2. Once Oracle gets upgraded to 12.2.0.1 or greater, this should be reassessed.

A.3.11 Oracle Statistics

After upgrade the Oracle data statistics should be renewed for STEP in order to collect a complete new set of statistics for all indices / tables in STEPSYS schema.

Regarding 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 indices/tables in STEPSYS schema, where more than 10% 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 indices/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 are changing too frequently, and the statistics would therefore newer be up to date. In order to make Oracle choose the right path of execution, when executing queries against these tables, STEP instead utilizes 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.

To collect a complete new set of statistics for all indices/tables in STEPSYS schema, log on to database as user STEPSYS (sqlplus or the like), and execute below listed command:

```
Exec adminpck.analyze(true);
```

A.3.12 Oracle DataGuard

The Oracle database can run with DataGuard protection and DataGuard can be configured with "SYNC AFFIRM" (Maximum Availability mode).

Running with DataGuard however introduces an overhead on the STEP system.

First of all, the database will have to run in "Force Logging" mode – This to ensure that ALL writings performed on Primary database are mirrored correctly to Standby database. STEP is using 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 as much as possible. Data stored in these tables are all considered as interim and not important in a restore/recover scenario.

Secondly, DataGuard can be configured to run with different levels of protection, depending on the SLA requirements. Choosing "SYNC AFFIRM" means going for almost 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 disk on secondary database server (Standby) before the Primary database server can send back an acknowledgement to the App server.

Therefore, a test is recommended where the protection level is lowered and the results in performance are analyzed.

A.4 STEP server hygiene

This paragraph describes the data gathering methodologies from a technical infrastructure viewpoint for the hygiene of the STEP servers.

A.4.1 STEP application server disk space

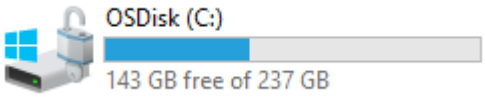
In case the system is slow in general, it could be caused by a clogged system. Every STEP application server needs to be analysed to see if the system has enough disk space left.

An easy way to check if the there's enough disk space left on the STEP application server is:

- In case of a Linux STEP application server: via 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
```

- In case of a Windows STEP application server: via the Windows Explorer.



This should be executed on every STEP application server.

If the percentage of the volume is 90% or more, then there's a good indication that the system is full.

Another indication is the analysis of the STEP logfile. If the STEP logfile shows the following error message, then a folder has reached its maximum limit of files:

```
… FileUtils.copyFile(): failed to copy file (tried multi) …
```

It's worthwhile checking both ways to see if the system is overloaded with files. If the system is overloaded, then it's usually due to import files not being removed and / or other unnecessary files, such as temporary files which have not been removed. Further analysis is required in those cases (see next).

A.4.2 Clean up unnecessary import files

When the STEP application server is cluttered, it might be caused by large number of import files in hotfolders. Furthermore, a huge number of import files themselves can also cause the system to be slow.

First it must be known where STEP keeps the import files, which is configured in the `sharedconfig.properties` file. In the following example, the location of the import files in the in hotfolder located in `/upload/hotfolders`.

```
#================================================================================#
# Filesystem Settings
#================================================================================#
# In a clustered application server setup the directories "workarea"
# and "upload" must be located on shared storage
Install.HotfolderRoot = /upload/hotfolders
Install.ImageCache = /workarea/imagecache
#Install.ProcessArea = /workarea/processarea
Install.ProcessOverviewPath = /workarea/processoverview
Install.IncidentReportFolder = /workarea/incidents
Install.BackgroundProcessArea = /workarea/background-processarea
Install.STEPWorkflowArea = /workarea/stepworkflow-area
#WebClient.AssetBaskets.Root = /workarea/assetbaskets
WebPublisher.Workarea = /workarea/webpublisher-workarea
```

The number of import files is analysed as follows:

- The number of files in a folder on the Linux STEP application server can be retrieved via

  `cd /upload`

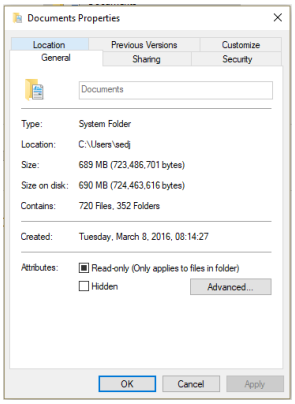
  `find hotfolders/ -type f | wc -l`

  ```
  [stibosw@euuktestepapp01p upload]$ find hotfolders/ -type f | wc -l
  find: `hotfolders/ZippedTest': Permission denied
  456580
  ```

- The number of files in a folder on the Windows STEP application server can be retrieved via

  Opening the Windows Explorer and go to /upload

  `Right-Click` the directory and select `Properties`



If the number of files import files in the hotfolder reaches the hundreds of thousands (e.g. 250k files), then it's recommended to clean these up and make sure the import files are cleaned up automatically in the future.

The size of the import files can also cause cluttering of the STEP application server. The size of the import files is analysed as follows:

- The number of files in a folder on the Linux STEP application server can be retrieved via

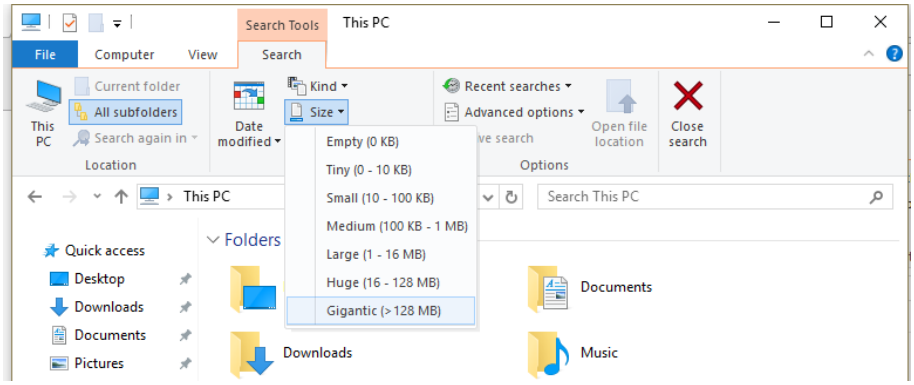
  ```
  cd /upload

  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
  ```

- The number of files in a folder on the Windows STEP application server can be retrieved via

  Opening the Windows Explorer and go to /upload

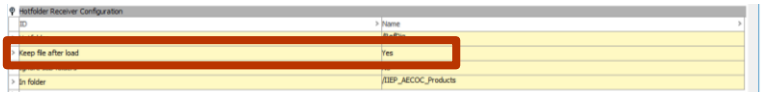
  `Right-Click` the directory and select `Properties`



After it has been identified there're too many import files and / or these import files are too big in size, its recommended to analyse why that is the case. The most common reasons are:

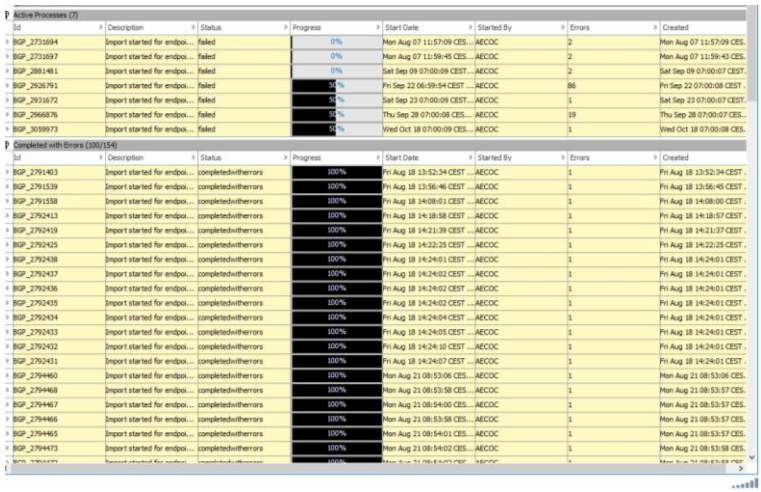
- One or more Inbound Integration End Points (IIEPs) are configured to keep the imported files after successful loading and processing. This setting obviously results in a situation where the import files are not removed and are piling up. The IIEP setting to keep the files after import is:

  `Keep files after load = Yes`.



  Simply change this setting to `Keep files after load = No` will remove the import files after successful processing.

- One or more Inbound Integration End Points (IIEPs) complete with errors or fail regularly. When an IIEP are marked as `failed` or `completed with errors`, then the import process will be saved together with the imported file for error tracing purposes. When there are errors in the import process of one or more IIEPs, then the background processes are not removed, which obviously results in a situation where the import files are not removed and are piling up.



The IIEP should successfully complete without errors to let STEP remove the background processes and therefore also the import files.

It might well be that the IIEPs suffers from `Optimistic Locking` errors. This will result in `failed` or `completed with errors` background processes and therefore a piling up of imported files. In that case, it's recommended to consider the setting `Reference Target Lock Policy` to `Relaxed` of the object types the long transaction applies to. See [!!!] for more information.

- The import directories of one or more Inbound Integration End Points (IIEPs) are configured in a non-standard in which STEP has trouble to remove the imported files after successful import and processing. When configuring an IIEP, the standard import directory structure is automatically created, which is "in", "errors", "failed", "logs", "save", although the "in" directory can be configured differently in the IIEP if required.

  For example:

  - Root folder of the import: `/upload/hotfolders/products/`
  - The "in" folder: `/upload/hotfolders/products/in`

    Here, the import files reside, and will be removed after processing.
  - The "save" folder: `/upload/hotfolders/products/save`

Here, the import files will remain when they're imported and when it's configured to keep the import files after successful processing.

- The "error" folder: `/upload/hotfolders/products/error`

  Here, the import files will remain when the import process completed with errors.

- The "failed" folder: `/upload/hotfolders/products/failed`

  Here, the import files will remain when the import process failed.

After the root cause is found why the import files are not removed automatically, and after this has been fixed, it's time to remove the unnecessary files to clean up the STEP system.

- In case the import files were not removed because of the IIEP `Keep files after load = Yes` setting, then the import files can simply be removed from the "save" directory by removing the files.

  - For a Linux STEP system, the removal of files can be done by using the "rm" command. The Linux command to remove the files is `rm -rf *` in the "save" folder where the many import files reside.

  - For a Windows STEP system, the removal of files can be done by simply using the Windows Explorer and deleting the files from the "save" folder.

- In case the import files were not removed because of any other reason than the IIEP `Keep files after load = Yes` setting, then the import files are tied to background processes in the STEP database. Removing the import files by deleting these directly from the file system will result in a situation where the background processes remain but without the corresponding import files. It's therefore recommended to remove the background processes of the IIEP rather than the import files directly. By removing the background processes, the corresponding import files will be removed automatically as well. Please see A.4.4 Clean up background processes how to remove the background processes, and thus the corresponding import files as well.

A.4.3 Clean up other unnecessary files

When the STEP application server is cluttered, and it's not caused by too many and/or too large import files, then it might be caused by large number of left-over or temporary files, e.g. from previous migrations. Furthermore, a huge number of these unnecessary files themselves can also cause the system to be slow.

There is no single location where these unnecessary files may reside on the STEP application server. However, sometimes unnecessary files are located in temporary directories such `tmp` or `temp` or `temporary`. But be very careful with temporary files the STEP system uses, such as the temporary files in the `step/diag` folder or in the `step/admin/cache` folders. These temporary files are used by the STEP system and must not be removed.

Nevertheless, it's recommended to analyse if there're many unnecessary left-over files or temporary files which can be removed in case the STEP application server is cluttered and unnecessary import files are cleaned up.

A.4.4 Clean up background processes

It is by intension in STEP that the old background process log information is not deleted automatically when a process ends in either `failed` or `completed with errors` state. The intention here is that STEP keeps the information for manual processing, so the background process owner can address the errors and fix these before deleting the background process log information manually.

However, many background processes can slow down the system. Each background processes creates a subdirectory on the file system which can clog the system. If the number of background processes in the STEP database are way too many, such as hundreds of thousands, then these background processes need to be cleaned up.

The way to check the number of background processes is done via:

- In the STEP database:
    - Go to the DB ToolBox which is normally in "`/opt/stibo/step/admin/app-server-toolbox/`".
    - Start the ToolBox via ./`db-toolbox.pl` or via `perl db-toolbox.pl`
    - Execute DB ToolBox number 38: "Show background process information"

    For example, the statistics of the number of background processes in the database are as follows.

```
Background processes status summary
+++++++++++++++++++++++++++++++++++


STATUS                COUNT_BG
--------------------- ----------
aborted                    558
completedwitherrors       1060
failed                     456
running                      2
succeeded                 8864
waiting                     25
                      ----------
sum                       10965
```

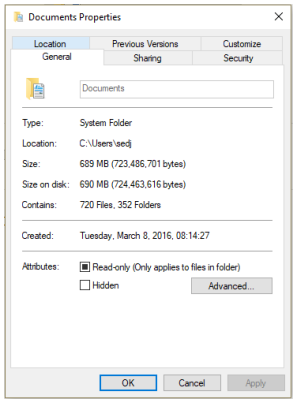
- On the STEP application server filesystem:
    - Count the number of files in directory `background-processarea` recursively on a Linux STEP application server via

      ```
      find background-processarea/ -type f | wc -l
      ```

```
[stibosw@lep1sta1 workarea]$ find background-processarea/ -type f | wc -l
1196277
```

Optionally, it's recommended to analyse the subdirectories as well and use e.g. `ls -lat` to see which files are created latest and use `du -hsx * | sort -rh | head -10` to see which files are the largest.

- Count the number of files in directory `background-processarea` recursively on a Windows STEP application server via the Windows Explorer by opening the Windows Explorer and go to `background-processarea` and `Right-Click` the directory and select `Properties`



Remark: the number of background processes in the STEP database and on the STEP application server file system may differ. It can happen that that there're way more background processes on the STEP application server (left-overs) when these background processes were not removed properly.

Once it's known there're too many background processes in the STEP database and / or on the STEP application server filesystem, it's recommended to clean up those background processes and make sure the background processes will not pill up in the future anymore.

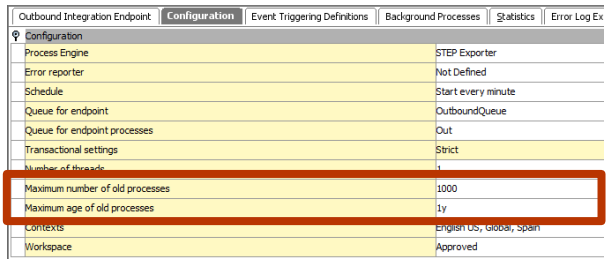
Making sure the background processes will not pill up in the future anymore is done as follows:

- Configure deletion of background processes on Integration End Points. The Integration End Points configurations to clean up background processes automatically are:
  - 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.
  - 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:

- Maximum number of old processes = 1000, which means that a maximum of 1000 succeeded and ended background processes will remain. The oldest background processes are deleted automatically when the number exceeds the specified limit of 1000.

- Maximum age of old processes = 1 y, which means that background processes are deleted automatically when these are older than 1 year.



Make sure the configurations represent a realistic situation. For example,

- Use e.g. maximum number of old processes (1000) and maximum age of old processes (1 w) in an environment with large number of small imports and exports

- Use e.g. maximum number of old processes (50) and maximum age of old processes (1 m) in an environment with small number of large imports and exports

- Configure auto-delete of background processes in configuration file. The auto-delete of the background process area in `sharedconfig.properties` can be configured with the number of hours for the background processes to be deleted after ending. This is based on the specified background process template. This is done as follows:

  - First, the location and names of the background processes need to be revealed.

    Open the `sharedconfig.properties` configuration file and lookup where the back ground processes are located, for example:

    Install.BackgroundProcessArea=/workarea/background-processarea

    - Set in: /workarea/sharedconfig.properties
    - Overrides default: "background-process-area"
    - Must be an existing directory.

  - Then go this directory on the STEP application server and see which background process templates are available, for example:

    ```
    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
drwxrwxr-x    2 stibosw stibosw    4096 Mar  2  2017 StateflowProfiling
drwxrwxr-x    2 stibosw stibosw    4096 Sep 29 09:53 SuspendBGPs
```

- Now, the auto-delete in hours of these background process templates can be configured in the `sharedconfig.properties` configuration file. This is done via `AutoDeleteBackgroundProcess.AgeInHours.[template]=[hours]`

  For example:

  AutoDeleteBackgroundProcesses

  AutoDeleteBackgroundProcesses.AgeInHours.Exporter=168

  - Set in: /workarea/sharedconfig.properties
  - This applies to all properties matching: /AutoDeleteBackgroundProcesses\.AgeInHours\..+/
  - Must be an integer.

  AutoDeleteBackgroundProcesses.AgeInHours.Importer=168

  - Set in: /workarea/sharedconfig.properties
  - This applies to all properties matching: /AutoDeleteBackgroundProcesses\.AgeInHours\..+/
  - Must be an integer.

  AutoDeleteBackgroundProcesses.AgeInHours.WebPublisher=168

  - Set in: /workarea/sharedconfig.properties
  - This applies to all properties matching: /AutoDeleteBackgroundProcesses\.AgeInHours\..+/
  - Must be an integer.

  The most important ones are usually the Exporter, Importer, Outbound, Inbound and WebPublisher background process templates.

  For example:

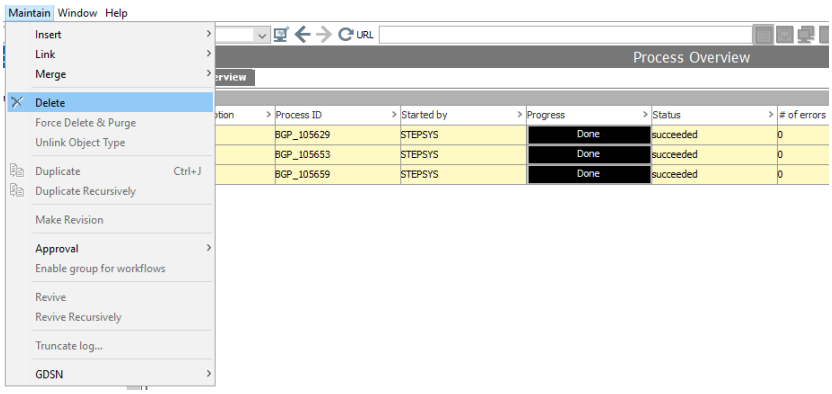
```
AutoDeleteBackgroundProcesses.AgeInHours.Outbound = 168

AutoDeleteBackgroundProcesses.AgeInHours.Inbound = 168
```

Once the configurations for automatically remove the background processes are set, then it's time to remove the background processes.

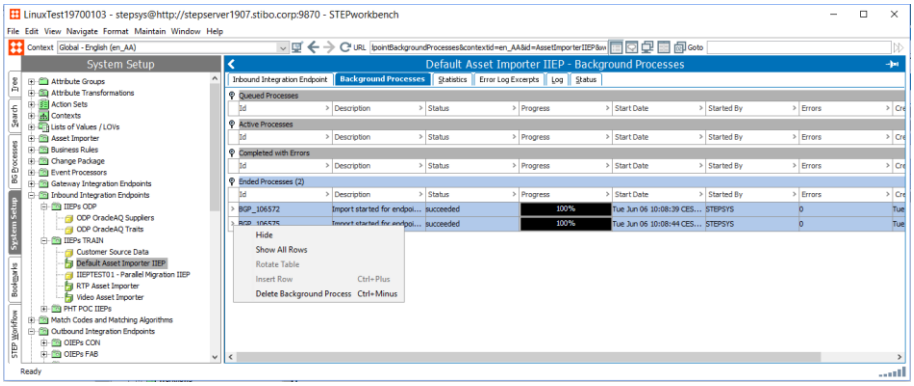
Please remind that the configurations for automatically remove the background processes only apply to the `succeeded` background processes. The `Failed` or `Completed with errors` background processes will not be auto-deleted by STEP and stay in the STEP database and on the STEP application filesystem for analysis purposes. These `Failed` or `Completed with errors` background processes should be analysed why they failed and be removed from STEP.

Cleaning up background processes is done via:

- Remove a small number of background processes via the STEP Workbench. A simple way to remove a small number of background processes is done via the STEP Workbench:
    - Go the `STEP Workbench > BG Processes`.
    - Make sure that you see all processes by clicking the appropriate icon.
    - Only the "Ended" background process can be removed. The "Queued" and "Active" background processes should be left alone.
    - The background processes can simply be removed via selecting the background processes from the tree and selecting `Maintain > Delete`.



Alternatively, it's also possible to go to the Integration End Points and select the `Background Processes` tab and remove the background processes from there.

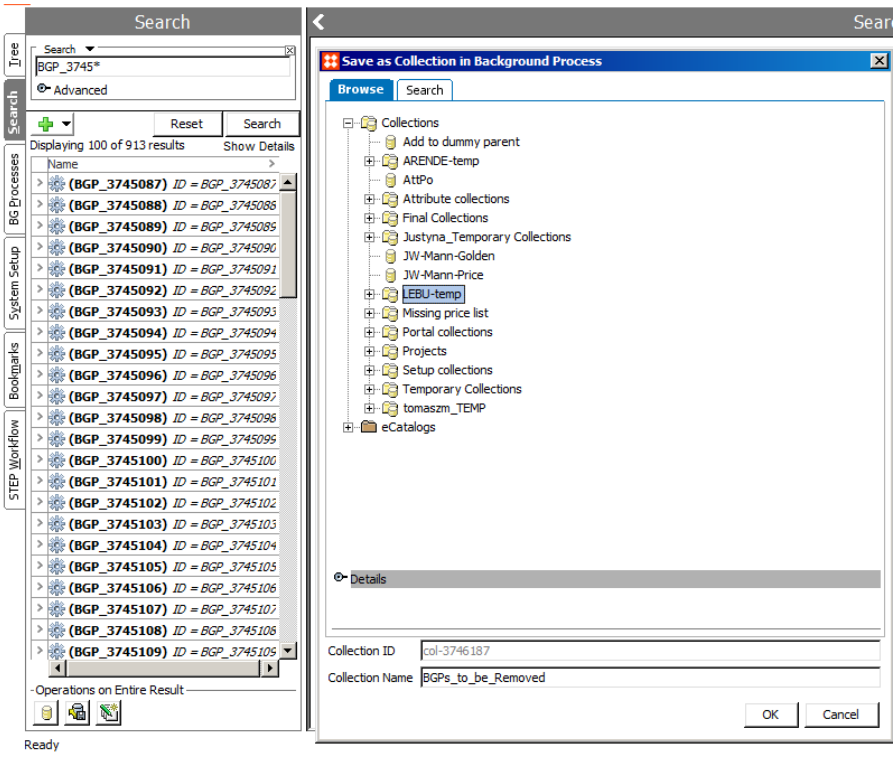


- Remove a large number of background processes via the STEP Workbench via search. A simple way to remove a larger set of background processes is done via searching the background processes in the STEP Workbench, moving these to a collection and removing the collection.
    - Go to the Integration End Point with a large amount of background processes, for example

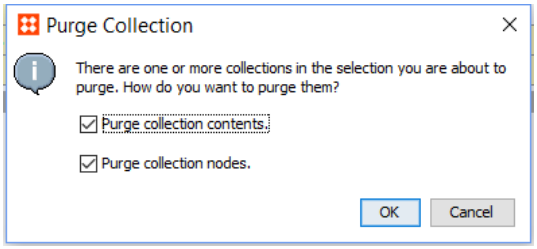
Ended Processes (100/210)

Id	Description	Status	Progress	Start Date	Started By	Errors	Created
BGP_3745223	Import started for endpoi...	succeeded	100%	Thu Jul 27 10:48:26 CEST ...	STEPSYS	0	Thu Jul 27 10:18:28 CEST ...
BGP_3745222	Import started for endpoi...	succeeded	100%	Thu Jul 27 10:49:17 CEST ...	STEPSYS	0	Thu Jul 27 10:18:28 CEST ...
BGP_3745221	Import started for endpoi...	succeeded	100%	Thu Jul 27 10:48:29 CEST ...	STEPSYS	0	Thu Jul 27 10:18:28 CEST ...
BGP_3745220	Import started for endpoi...	succeeded	100%	Thu Jul 27 10:49:42 CEST ...	STEPSYS	0	Thu Jul 27 10:18:28 CEST ...
BGP_3745219	Import started for endpoi...	succeeded	100%	Thu Jul 27 10:49:40 CEST ...	STEPSYS	0	Thu Jul 27 10:18:28 CEST ...
BGP_3745218	Import started for endpoi...	succeeded	100%	Thu Jul 27 10:48:35 CEST ...	STEPSYS	0	Thu Jul 27 10:18:28 CEST ...
BGP_3745217	Import started for endpoi...	succeeded	100%	Thu Jul 27 10:49:45 CEST ...	STEPSYS	0	Thu Jul 27 10:18:28 CEST ...
BGP_3745216	Import started for endpoi...	succeeded	100%	Thu Jul 27 10:49:24 CEST ...	STEPSYS	0	Thu Jul 27 10:18:28 CEST ...
BGP_3745215	Import started for endpoi...	succeeded	100%	Thu Jul 27 10:49:41 CEST ...	STEPSYS	0	Thu Jul 27 10:18:28 CEST ...
BGP_3745214	Import started for endpoi...	succeeded	100%	Thu Jul 27 10:49:36 CEST ...	STEPSYS	0	Thu Jul 27 10:18:28 CEST ...
BGP_3745213	Import started for endpoi...	succeeded	100%	Thu Jul 27 10:49:33 CEST ...	STEPSYS	0	Thu Jul 27 10:18:28 CEST ...
BGP_3745212	Import started for endpoi...	succeeded	100%	Thu Jul 27 10:49:26 CEST ...	STEPSYS	0	Thu Jul 27 10:18:28 CEST ...
BGP_3745211	Import started for endpoi...	succeeded	100%	Thu Jul 27 10:49:25 CEST ...	STEPSYS	0	Thu Jul 27 10:18:28 CEST ...
BGP_3745210	Import started for endpoi...	succeeded	100%	Thu Jul 27 10:49:14 CEST ...	STEPSYS	0	Thu Jul 27 10:18:28 CEST ...
BGP_3745209	Import started for endpoi...	succeeded	100%	Thu Jul 27 10:49:19 CEST ...	STEPSYS	0	Thu Jul 27 10:18:28 CEST ...
BGP_3745208	Import started for endpoi...	succeeded	100%	Thu Jul 27 10:49:38 CEST ...	STEPSYS	0	Thu Jul 27 10:18:28 CEST ...
BGP_3745207	Import started for endpoi...	succeeded	100%	Thu Jul 27 10:49:35 CEST ...	STEPSYS	0	Thu Jul 27 10:18:28 CEST ...
BGP_3745206	Import started for endpoi...	succeeded	100%	Thu Jul 27 10:49:32 CEST ...	STEPSYS	0	Thu Jul 27 10:18:28 CEST ...
BGP_3745205	Import started for endpoi...	succeeded	100%	Thu Jul 27 10:49:37 CEST ...	STEPSYS	0	Thu Jul 27 10:18:28 CEST ...
BGP_3745204	Import started for endpoi...	succeeded	100%	Thu Jul 27 10:49:12 CEST ...	STEPSYS	0	Thu Jul 27 10:18:28 CEST ...

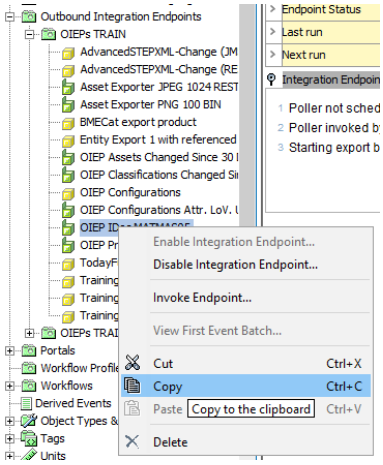
    - Select the ID column and copy and paste these IDs to a text file.
    - Go to STEP Workbench > Search
    - Search for the background processes, e.g. `BGP_3745*` or search from list and paste the IDs from the text file.
    - Add the search result to a collection, e.g. collection `BGPs_to_be_Removed`



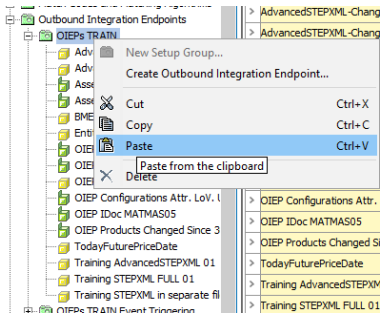
- Go to the collection e.g. `BGPs_To_Remove` and select `Maintain > Force Delete and Purge` and select `Purge collection contents`



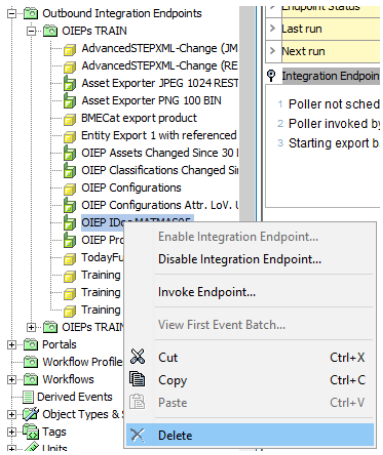
- Remove a large number of background processes via the STEP Workbench via copy. Another simple way to remove a larger set of background processes is done via copy and paste the Integration End Points and then removing the original Integration End Points.
    - Copy the Integration End Point, e.g. copy `OIEPIDOCMATMAS05-01`



- Paste the Integration End Point, e.g. paste to `OIEPIDOCMATMAS05-02`



- Once the new Integration End Point is successfully copied and running, then remove the original Integration End Point, e.g. delete `OIEPIDOCMATMAS05-01`



This will delete the Integration End Point and all related background processes as well.

- Remove a large number of background processes via a Stibo Systems script. Finally, when all of the above methods for removing background processes do not work because there’re simply too

many, then it’s also possible to remove background processes via a request to Stibo Systems to remove background processes (via a JIRA ticket).

B Analysis methodologies and recommendations from a functional viewpoint

This appendix describes the data gathering methodologies and performance improvement recommendations from a functional viewpoint.

B.1 Base setup

The base setup of the STEP contexts and dimensions, product and entity data model, assets and classifications are designed based on business requirements. Nevertheless, the base setup should be designed based on performance as well since a complex base setup can have a negative impact on the performance in general.

An analysis of the base setup is therefore worthwhile, however changes on the base setup (for performance reasons) do usually have a huge impact. For example, changing the data model have impact on attributes, references, but also can have impact on business rules, workflows, WebUI configurations, exports and imports. It's therefore highly recommended to always weight the pros and cons of such a change before changing the base setup.

Nevertheless, it's always recommended to analyse the base setup and then consider if it's worthwhile to change anything.

B.1.1 Dimensions and contexts

The setup of the STEP contexts and dimensions is flexible and is probably designed and setup based on business requirements. Nevertheless, the setup of contexts and dimensions should be designed based on avoiding unnecessary complexity and optimal performance as well, since a complex setup can have a negative impact on the performance in general.

Usually, in multi lingual setups, two dimension-points are defined

- Country
- Language

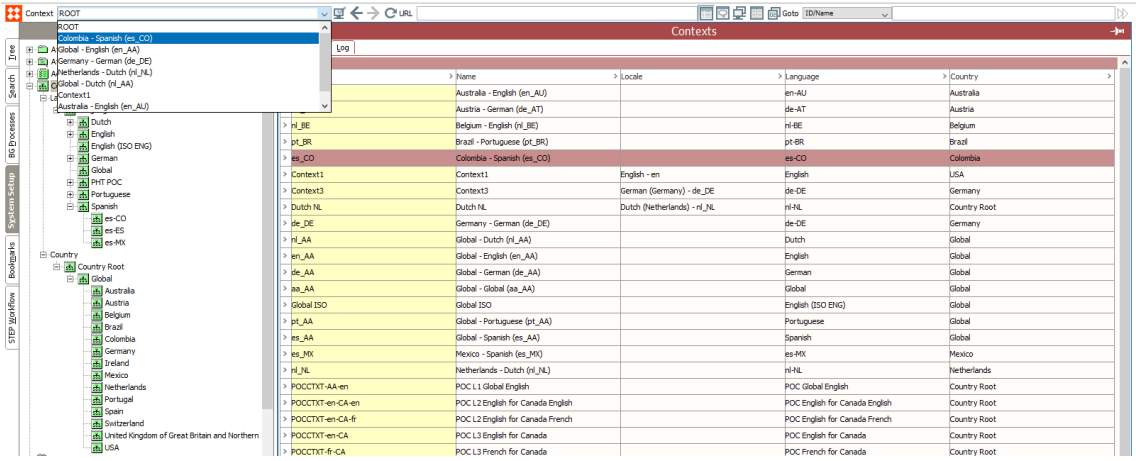
For this example, it means that every object in STEP (entity and product) can have a country and a language dimension, which allows for

- Translations in different languages (language dimension) where the same object (e.g. product AC-P7000-42) has an English, German, Spanish, etc. translation of the data.
- Differences in countries such as different accessories of the product in different countries via suppressing a reference.

The context defines the combination between the defined dimension points.

For example, the context Colombia - Spanish (es_CO) is defined with

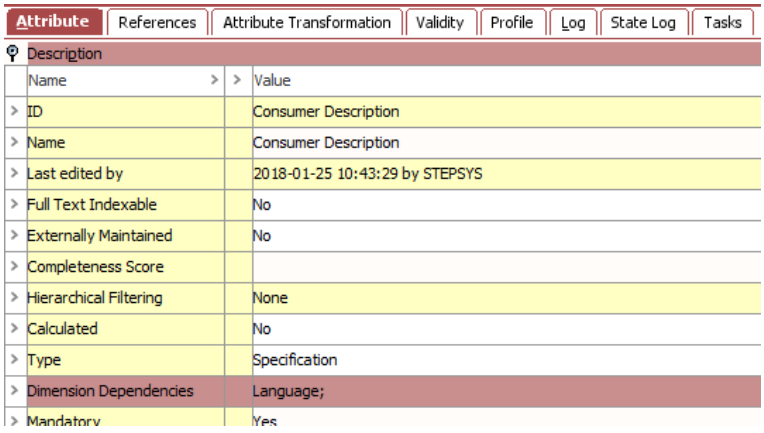
- Country = Colombia for Colombia (es-CO)
- Language = Spanish for Columbia (es-CO)



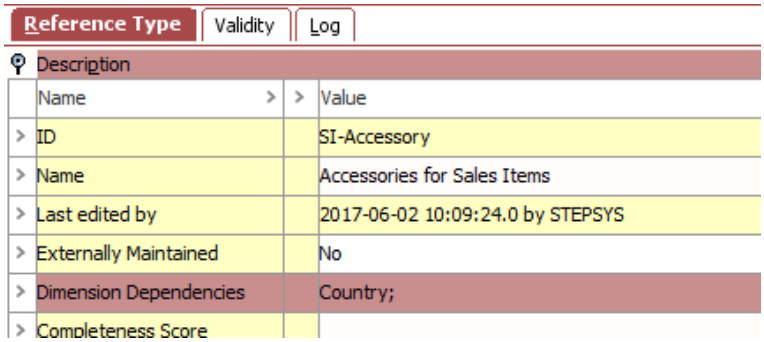
Please be aware that the dimensions allow to be setup in a hierarchy which uses inheritance. This means that the dimension point inherits the data from its parent unless it has its own data. For example, the language “Spanish for Colombia (es-CO)” is setup below the language “Spanish (es)”. This means that all data in the language “Spanish for Colombia (es-CO)” is inherited from the language “Spanish (es)” until the data in the language “Spanish for Colombia (es-CO)” is set.

When 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.



Please be aware that combining dimensions might result in a very complex situation. For instance, if metadata attribute for an asset (e.g. Asset Description) is defined to be language and country dependent, then it's possible to have a translated metadata attribute on the asset in different languages, and have a different metadata attribute value for different countries. But what does this mean in reality, if it's also considering that the metadata attribute may inherit the values from its parents (e.g. from language Spanish and from country Global)? It's hard to predict the values when making this attribute multi-dimensional. Therefore, it's recommended to avoid multi-dimensional configurations as much as possible.

Another recommendation is to be careful with the number of dimensions to configure. The more dimensions defined, the more complex the STEP solution will be. A good practice is not to define more than two dimensions (e.g. language and country). Therefore, only define a new dimension if it's absolutely required. For example, if the objects (e.g. products) do not differ much in the countries, then leave the country dimension behind and only define a language dimension.

For performance reasons it's also recommended to define as little as dimensions as possible (good practice is maximum of two, e.g. language and country) and only configure dimension dependency when required, simply because a complex dimension setup requires more extensive processing in e.g. imports, exports, bulk updates, running business rules, etc. The number of dimension points (e.g. number of languages) is however of less importance, as well as the number of contexts.

B.1.2 Data model

The data model of STEP is very flexible and allows for complex data model setups. The STEP data model is probably designed and setup based on business requirements. Nevertheless, the data model should be designed based on avoiding unnecessary complexity and optimal performance as well, since a complex data model can have a negative impact on the performance in general.

For example, if it's required to define an entity "Company" which has a visiting and distribution address with country, city, street and zip code, then the addresses can be defined as follows:

- A couple of attributes directly on the entity "Company", such as visiting address street, visiting address city, distribution address zip code, etc.
- Or, a separate entity called "Address" with type (e.g. "visiting address" and "distribution address"), street, city, zip code, etc. and a reference to reference the "Company" to the "Address".

- Or, two data containers “Visiting Address” and “Distribution Address” with country, city, street, zip code, etc.

The right data model to use is probably the usage of data containers, since

- The reuse of the same address over multiple companies is minimal. Therefore, the data model with the “Address” as a separate entity in unnecessary complex.
- Some companies might not have a “Distribution Address” making the usage of separate attributes on “Company” somewhat inflexible.

For example, if it’s required that a product belongs to one product family in most cases, but might belong in exceptional cases to another product family as well, then this can be defined as follows:

- 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 right data model to use is probably the usage of the product reference when inheritance of the second product family is not required, since it is less complex.

The STEP data model allows is very flexible and allows for different approach for the same challenge. Still, the recommendation is to not overdesign the data model and keep the data model as simple and as meaningful as possible.

For performance reasons it’s also recommended to design the data model as simple as possible, simply because a complex data models require more extensive processing in e.g. imports, exports, bulk updates, running business rules, etc.

For performance reasons it’s also recommended to use manually sorted on product objects and entities consciously.

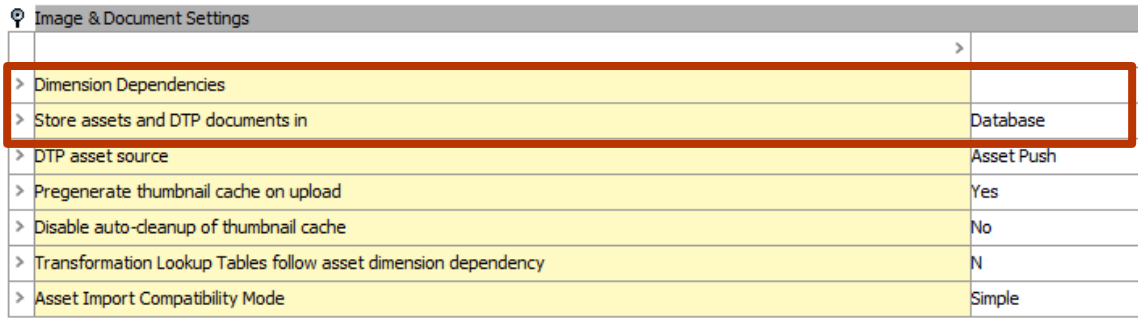
B.1.3 Assets

Digital assets are media files like images, videos documents, etc. Digital assets in STEP are asset objects which can hold attributes (metadata) and have a reference to the actual asset (binary file).

The asset binaries can be stored outside STEP in an external DAM system. The storage of the asset binary within STEP can be on the file system or in the STEP database. The method of asset binary storage in STEP (file system or database) doesn’t determine the performance of STEP. One might consider storing the asset binaries in the database for faster access, but accessing many asset binaries

in the database also results in higher database traffic. All in all, the asset binary storage method doesn't determine the performance of STEP.

Furthermore, it makes sense to check the dimension dependency of the assets. The asset dimension dependency shouldn't be defined when it's not required.



The asset configurations as displayed in the screenshot are available in the STEP Workbench > System Setup > Users & Groups.

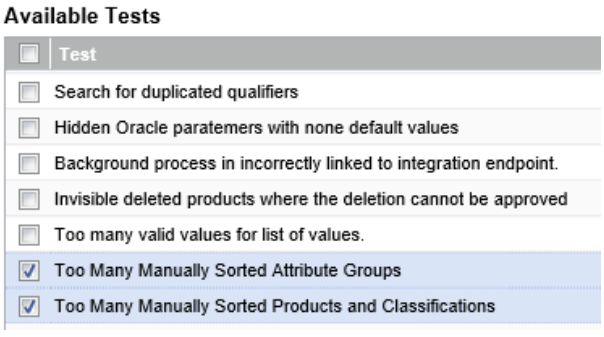
The import and export of assets should also be considered. STEP allows for asset transformations during import and export. Asset transformations obviously require system resources and therefore unnecessary asset transformations should be avoided. See the paragraphs on import and export for more information: B.7 Imports.

B.1.4 Classifications

STEP allows for setting up one primary hierarchy (blue tree) and multiple classifications (yellow trees).

However, many children in each level of a hierarchy is expensive. STEP has a name uniqueness check so many items in one node of the hierarchy degrades performance.

The easiest way to find hierarchies with too many children is via the STEP Health Check in the Admin Portal.

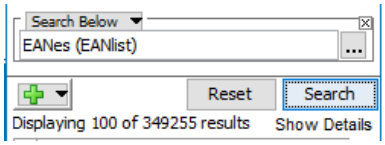


See C Analysis methodologies and recommendations via STEP Health Check for more information.

Alternatively, the hierarchies with too many children are marked in the STEP Workbench with an x and are automatically set to non-readable.

Once these hierarchies with many children are found, the search is used to analyse how many children are in the hierarchy by using the `search below` option.

For example, the following screenshot shows a hierarchy called `EANes` with about 350k children.



It's recommended to avoid hierarchies with too many children. Therefore, try organizing and categorise children in subhierarchies.

B.1.5 References

The setup of references in STEP is very flexible. References can be configured as with multiple different sources and targets (reference and reference by), are of a certain type (product, asset, classification, etc.), have metadata attributes on the reference, use a type of inheritance, and so on. All these reference configurations do not have an impact on performance as such.

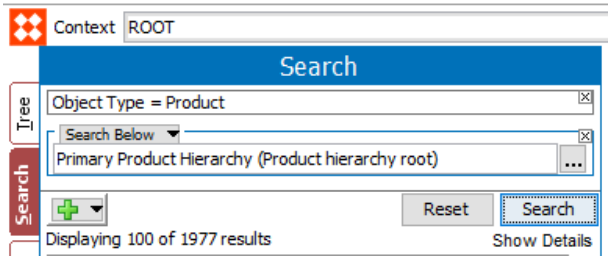
However, ad-hoc navigation across a reference to another object to fetch attribute values and / or references is expensive. This because another database access and potentially another disk access is required. For instance, navigation via references in business rules to resolve values, resolving calculated attributes or resolving references during export. It's therefore recommended to check the usage of the references in business rules to resolve values, resolving calculated attributes, or resolving references during export.

B.1.6 Global count of objects and attributes

This paragraph provides examples on how to count number of objects in STEP via search. The search displays the total number of results found.

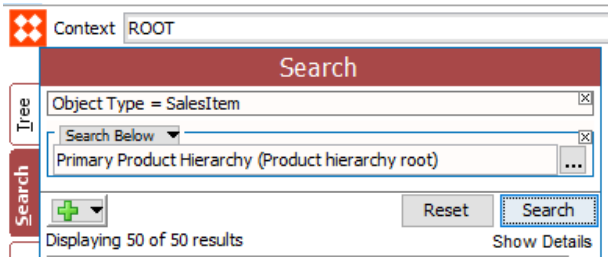
- Count number of products
    - Search for the number of all product objects via selecting
        - Object Type = Product
        - Search Below = Primary Product Hierarchy

      For example:



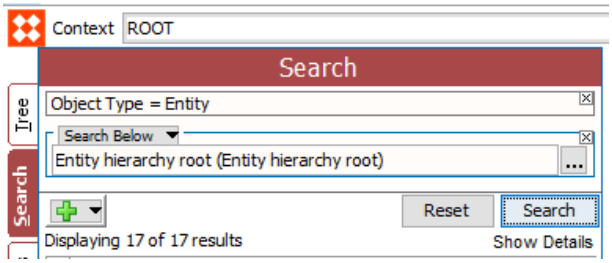
- Search for the number of each product object type via selecting
  - Object Type = [Product Object Type]
  - Search Below = Primary Product Hierarchy

  For example:



- Count number of entities
  - Search for the number of all entity objects via selecting
    - Object Type = Entity
    - Search Below = Entity hierarchy root

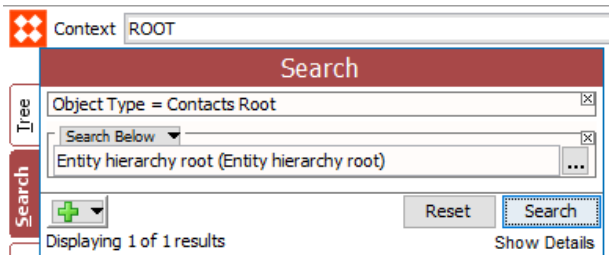
    For example:



  - Search for the number of each entity object type via selecting
    - Object Type = [Entity Object Type]

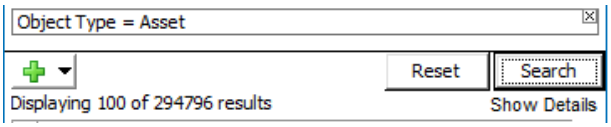
- Search Below = Entity hierarchy root

For example:



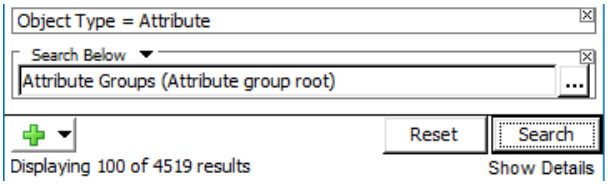
- Count total number of assets via
  - Object Type = Asset

For example:



- Count total number of attributes via
  - Object Type = Attribute
  - Search Below = Attribute Groups

For example:



B.2 Attribute and attribute group configurations

The setup of attributes and attribute groups in STEP is very flexible. Attributes can be configured as Specification Attributes, Description Attributes, have a certain value type (text, list of value, number, date, etc.), are multi-values or not, have a unit of measure, and so on. All these attribute and attribute group configurations do not have an impact on performance as such.

However, there are attribute and attribute group configuration which might have influence on the STEP performance.

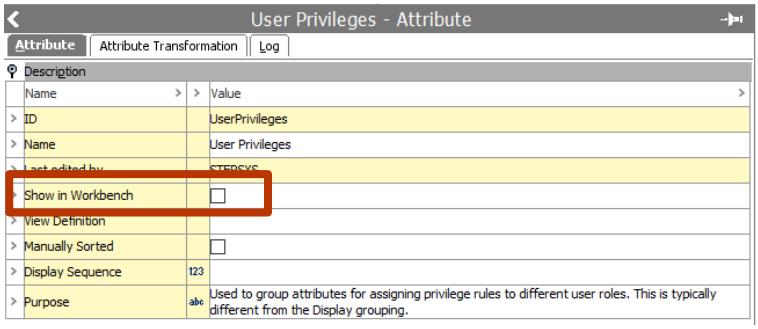
The most convenient way to analyze the attribute group and attribute configurations, is simply by exporting the attribute groups to a STEPXML file and the attributes to an Excel file.

B.2.1 Disable the display of attribute groups where possible

STEP however provides the possibility to include or exclude attribute groups from being displayed in the Workbench. However, showing many attribute values in the Workbench will require additional processing power, and as such, may may become expensive, especially the values of calculated attributes.

Setting the display of attributes is done via the configuration of the attribute groups in Workbench > System Setup > Attribute Groups. Disable the `Show in Workbench` option to prevent the attribute group from being displayed.

For example:



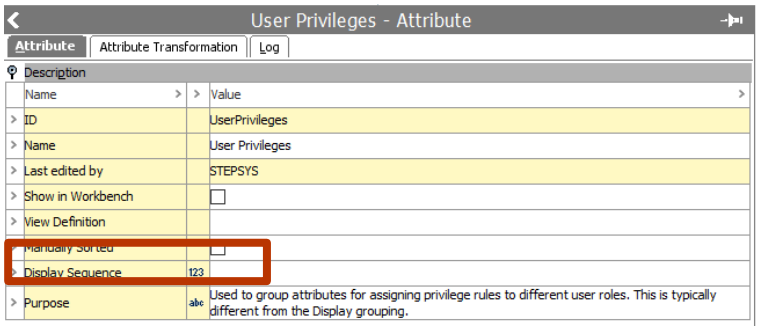
It's recommended to use the `Show in Workbench` option to configure which attribute groups need to be displayed in the Workbench and attributes do not need to be displayed in the Workbench.

B.2.2 Disable manually sorted where possible

Manually Sorted is an option in STEP Workbench to sort order attributes, assets, products, entities, etc. in their hierarchy. However, sorting attributes will require additional processing power, and as such, may have impact on performance when using extensively.

Attributes can be manually sequenced locally within an attribute group by drag-and-drop in STEP Workbench. The `Manually Sorted` option is set in the Workbench > System Setup > Attribute Groups.

For example:



More specifically the functionality contains:

- The ability to manually sequence attributes within selected attribute groups.
- Attribute groups must be explicitly marked as manually sorted in order to allow sequencing.
- An attribute must be linked directly into a sequenced attribute group in order to receive any sequencing.
- Sequence is used when a sequenced attribute group is referenced in STEP WebUI and Workbench or included in an export
- The sequence provided in the attribute group will take precedence over any other sequencing, whether specified on an attribute link or on the attribute.

The limitations are:

- Manual sequencing is only set up in STEP Workbench.
- Manual sequence can only be enabled for attribute groups that have no sub-groups.
- The manual sequencing is only supported for attributes. Sequencing of Tag Groups, LOV groups and reference Types below an attribute group is not supported.

Manually sorted is configured as follows:

- `Manually Sorted = No` means that child nodes of this object, that are holding this current object type, are ordered alphabetically in the tree hierarchy. Their sequence cannot be manually changed by a user.
- `Manually Sorted = Yes` means that child nodes of this object, that are holding this current object type, can have their sequence manually changed in the tree hierarchy. This can be done by dragging an object to another position.

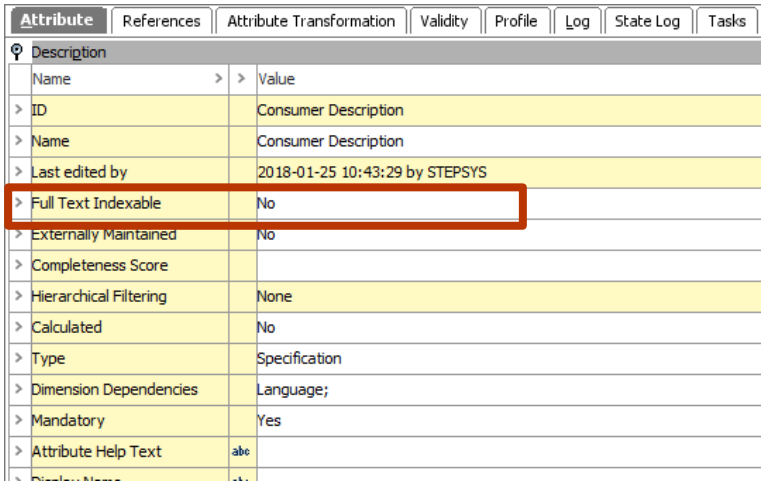
It's recommended to use the `Manually Sorted` option only when it's required to sequence the attributes in their attribute group.

B.2.3 Disable full text indexing where possible

Full text indexable is a way of indexing the values of an Attribute with validation base type 'text' in words. This will enable you to search for words within 'text' values. However, when an attribute is made as Full Text Indexable, there are some chances of having performance issues.

The `Full Text Indexable` option is set in the Workbench > System Setup > Attribute Groups > Attribute.

For example:



Without the full text indexable option you can still search for values. This option is only needed if you want to search for objects based on a word or set of words in a sentence in an attribute value.

Recommended is to use this `Full Text Indexable` option to attributes only if needed, since there are some chances of having performance issues.

B.2.4 Disable dimension dependency where possible

Previously it was mentioned that dimension dependency should only be applied when needed. This also goes for attributes. See B.1.1 Dimensions and contexts for more information.

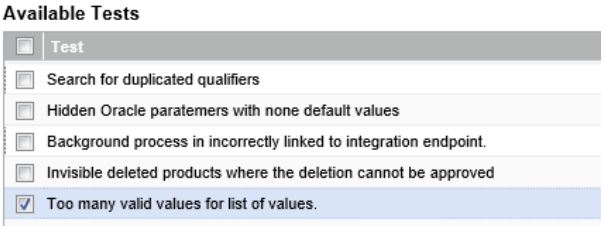
For performance reasons it's also recommended to use the dimension dependency on attributes little as possible and thus only configure dimension dependency when required.

B.2.5 Avoid using large lists of values where possible

Special attention is required for attributes with large List Of Values (LOVs). Lists of Values (LOVs) with many values can lead to performance degrades. Many LOV values are LOVs with thousands of values.

Also from a usability perspective, it's recommended to avoid large LOVs.

The easiest way to find LOVs with too many values is via the STEP Health Check in the Admin Portal.



See C Analysis methodologies and recommendations via STEP Health Check for more information.

Once the LOVs with too many values are found, it's recommended to split the LOVs up into smaller LOVs or do not use the LOV at all but a text field instead.

B.2.6 Avoid using calculated attributes where possible

Pay special attention to calculated attributes. Calculated attributes provide a lot of flexibility, but you must introduce them with great care because of the direct impact they can have on downstream deliveries. The following usages of calculated attributes can degrade performance:

- Avoid usage of calculated attributes in cross-contexts exports. The cross-context exporter does not impact the performance significantly when extracting normal attributes in many contexts compared a single context. However, the time it takes to extract calculated attributes grows close to linear with the number of contexts. This is because STEP is not able automatically to detect whether the formula that expresses the calculation is dimension dependent or not, which means that the introduction of new contexts to a downstream delivery that includes many calculated attributes can result in a significant performance degrade. Consider evaluating the need for each calculated attribute since a feed that includes a calculated attribute. Also, consider performing the calculation upfront on approval via business actions and then place the result in a normal non-calculated attribute (copy field), instead of using calculated attributes to resolve the values at export.

- Check the locality of calculated attributes. Calculated attributes that only access 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. Therefore, calculated attributes that navigate to other objects via references and / or hierarchy links must be carried out in a very controlled manner. It and should never go across transitive closures that potentially grow very large, such as all children of a large and growing entity or product hierarchy.

- Check whether the calculated attributes depend on other calculated attributes. Calculated attributes should not depend on other calculated attributes, because this kind of chain of dependencies can make performance very unpredictable.

A way to find all the calculated attributes in the STEP system is to simply export all attributes to Excel and filter on "Attribute Calculated". Then analyse these calculated attributes one by one in STEP and determine:

- If the calculated attribute is used in cross-contexts exports. If yes, consider the necessity of using the calculated attribute in the export at first. And if the calculated attribute value is required in the cross-context exports, then consider performing the calculation upfront on approval via business actions and then place the result in a normal non-calculated attribute (copy field).

- If the calculated attribute navigates to other objects via references and / or hierarchy links which go across many children. If so, then consider replacing the calculated attribute with a potential more efficient business action.

If the calculated attribute depends on other calculated attributes. If so, then consider using more efficient business actions to come to the same result.

B.3 Optimistic locking

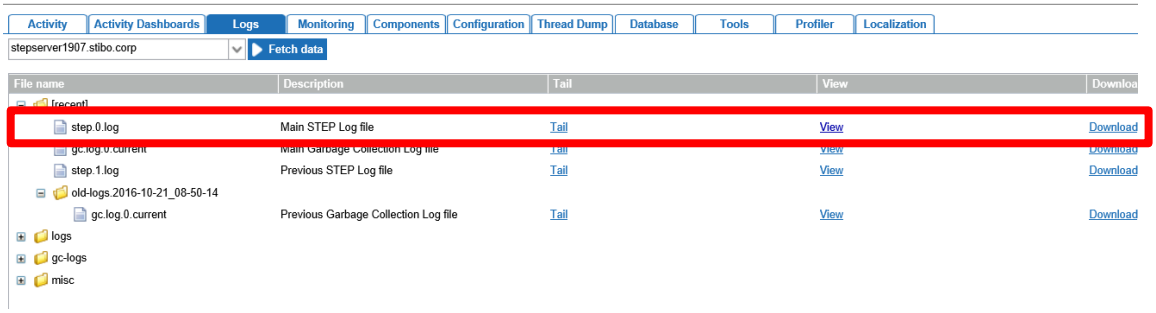
STEP runs with optimistic locking policy. Optimistic locking errors may be caused by long transactions of e.g. imports, exports, asset push, business rules. The longer a transaction, the higher the probability of introducing an optimistic locking failure when running long transactions simultaneously.

STEP will put the transaction on hold when optimistic locking occurs, and will try to process the transaction again after some time. Optimistic locking errors therefore degrade the performance of imports, exports, asset push, business rules, etc. and degrade the performance of the STEP system.

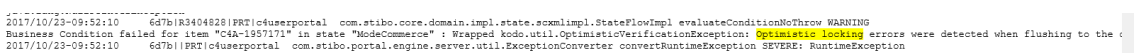
B.3.1 Analyzing optimistic locking failures

Optimistic locking errors are analysed via the STEP log file.

- Analyse the log file via the Step Admin Portal and download or view the Main STEP log file.



- Search the log file for “optimistic locking”.



Since optimistic locking errors are caused by long transactions of maybe an import, export, business rule, asset push, etc. it’s important to analyse precisely what caused the optimistic locking errors.

For example:

- 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 WebUI transaction in WebUI “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]

- Waning for optimistic locking caused by business condition transaction in state “ModeCommerce” in WebUI “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

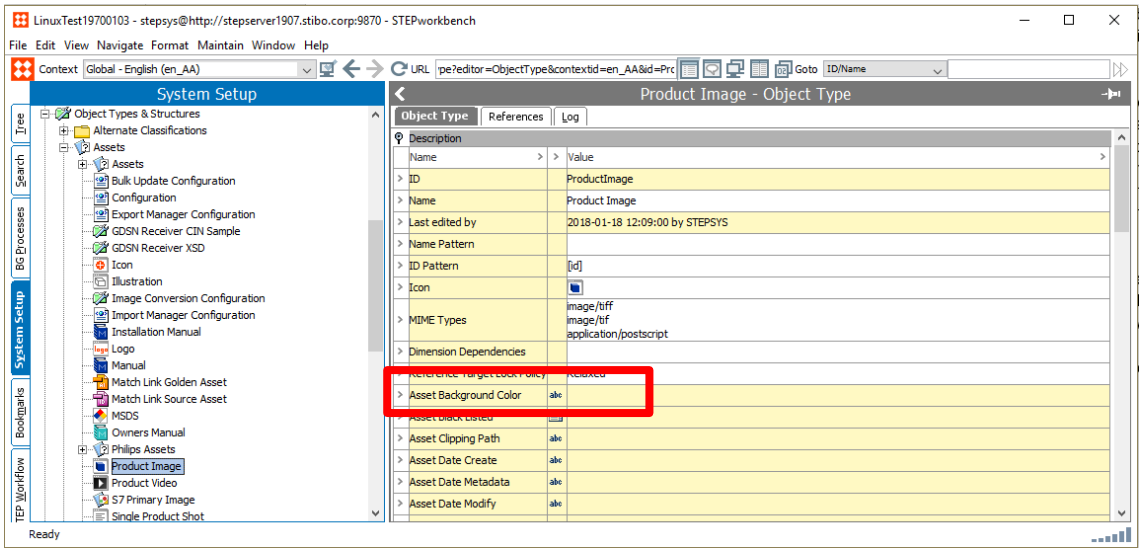
B.3.2 Recommendation: optimize large transactions

It’s recommended to keep the 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. See next paragraphs for the good practices on business rules, imports, and exports.

B.3.3 Recommendation: set reference target policy

Nevertheless, in some cases the long transactions cannot be simplified easily. In these cases, it’s worth considering setting the `Reference Target Lock Policy` to `Relaxed` of the object types the long transaction applies to.

For example: the following screenshot shows the configuration of `Reference Target Lock Policy` to `Relaxed` of object type `Foto` since the many of the optimistic locking errors in the asset push apply to this object type.



The `Reference Target Lock Policy` is used for the Assets, Entities, Classifications, and Products object types to manage how objects should be locked while they are being referenced. When a reference between two objects is being created in STEP, the target object is locked to ensure that it is not being deleted while the reference is being created, meaning only one process or user can edit it.

This can be an issue when 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. The default setting for this property is set to 'Strict' unless changed to 'Relaxed', and 'Relaxed' should NOT be used on an object type for objects that are often deleted – this could result in poor performance, and risk of locks and deadlocks.

When a reference between two objects is being created in STEP, the target object will be 'locked' to ensure that it is not being deleted while the reference is being created. In most of these cases, deletion of the object is not a deletion from the database, but the deletion that occurs due to revision control when an insert of a new history entry occurs. This can be a major issue when 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. Although STEP continues to retry the import, this can cause inbound feeds to be negatively affected.

The 'Relaxed' setting on the property puts a less restrictive lock on the reference target objects being edited so that they can be updated concurrently by more than one process and/or user, with a full lock being employed only in the case that a deletion is attempted. This setting, when used with object types that are frequently referenced but rarely deleted, will improve the performance and stability of parallel inbound imports, bulk updates, and users concurrently creating references to the same objects.

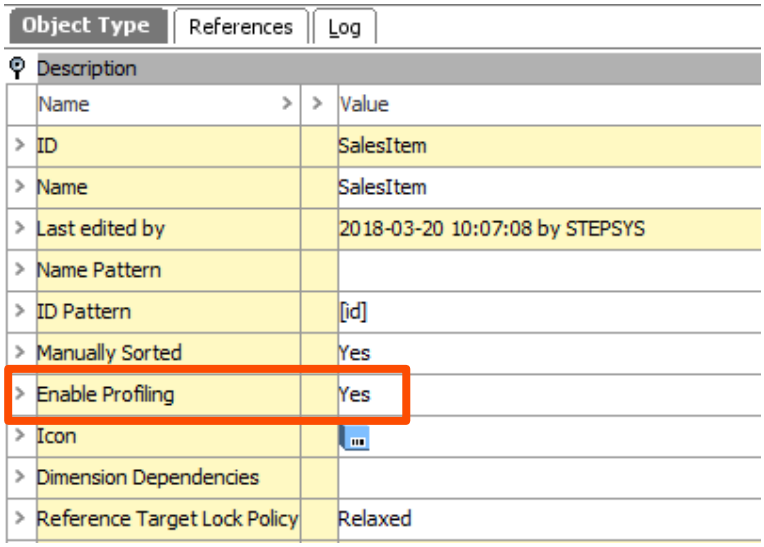
It's recommended considering changing the `Reference Target Lock Policy` to `Relaxed` when optimistic locking failures occur frequently in the logfile, the transactions cannot be optimized, and the object types that are frequently referenced but rarely deleted.

B.4 Data profiling

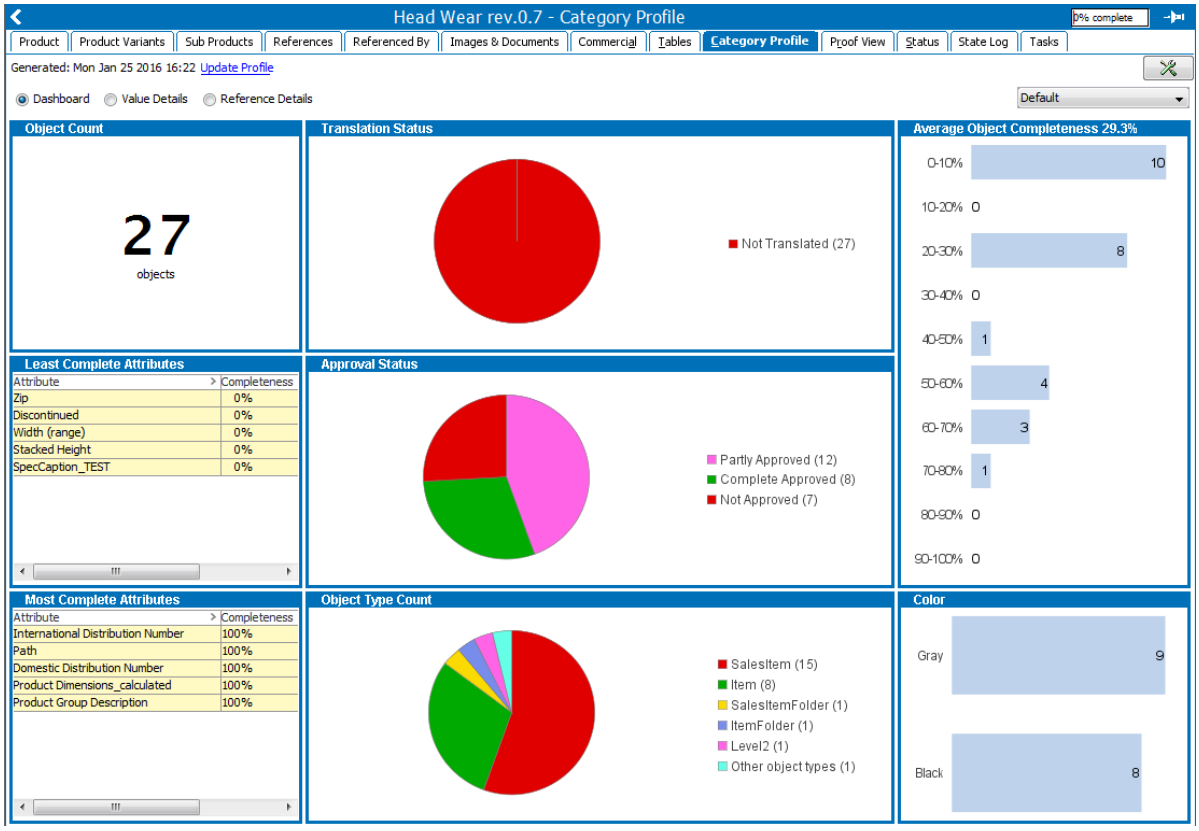
The Category / Data Profile functionality provides a detailed overview of data in a specific branch of the hierarchy in hierarchy tree. However, when large categories are profiled, the system uses a lot of memory. This may have a negative impact on the system performance.

The Category / Data Profile functionality is enabled by setting the `Enable Profiling` option on objects in Workbench > System Setup > Object Types & Structures.

For example:



Each profile contains information about the profiled data and provides easy access to correcting data errors, for example:



It's recommended to only enable profiling only when required.

Still, if it's decided to set the `Enable Profiling` option, then there're options to limit the memory usage for profiling. It's recommended to use one of the following properties to limit memory usage.

- Setting the maximum of distinct attribute values considered during profile generation per attribute in the `sharedconfig.properties` configuration file

  `DataProfile.MaxDistinctAttributeValuesConsideredDuringProfileGeneration`

  DataProfile.MaxDistinctAttributeValuesConsideredDuringProfileGeneration=100

  - Using default value
  - Must be an integer.

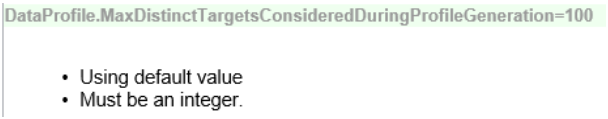
  This sets the maximum of distinct attribute values considered during profile generation per attribute. The default setting is 100. When the limit is reached the following happens:

  - Frequent value counts might 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 colour. The attribute completeness and count and the value instance counts for profiled attributes are correct.

- Setting maximum number of distinct targets for the reference or link type that is profiled in the `sharedconfig.properties` configuration file

  `DataProfile.MaxDistinctTargetsConsideredDuringProfileGeneration`

  This sets the maximum number of distinct targets for the reference or link type that is profiled. The default setting is 100.

B.5 WebUI configurations

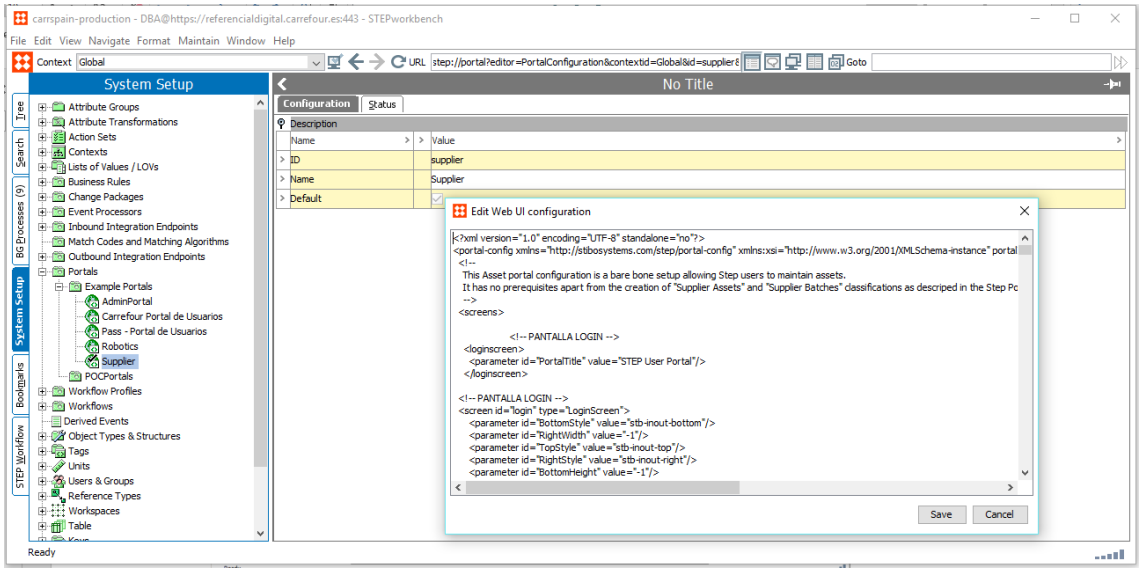
The WebUI Designer is flexible and can configure a WebUI in many ways. However, not all configurations perform equally well. Therefore, it is important to consider performance when configuring a WebUI. A screen with little data and limited functionality will load faster than a screen with lots of data and functionality.

Please also see the Health Check results (Appendix C) where the most urging WebUI issues are listed.

B.5.1 Use multiple WebUIs

The STEP WebUI configurations are files XML which are accessible in Workbench > System Setup > Portals.

For example:



When a user logs into the WebUI and opens the WebUI Homepage, the corresponding configuration XML file is loaded. So, if the WebUI configuration XML file is big, then the WebUI homepage takes quite some time to load.

A rule of thumb is to consider splitting the WebUI when the WebUI configuration is more than 25k XML lines. Therefore, it's recommended to split up the WebUI into multiple WebUIs, each with their own specific purpose.

B.5.2 Use small dedicated WebUI screens

A typical WebUI screen fetches all attribute values from the attribute group define in the WebUI screen. The screen then filters out the attributes based on validity of for the product type and based on user privileges.

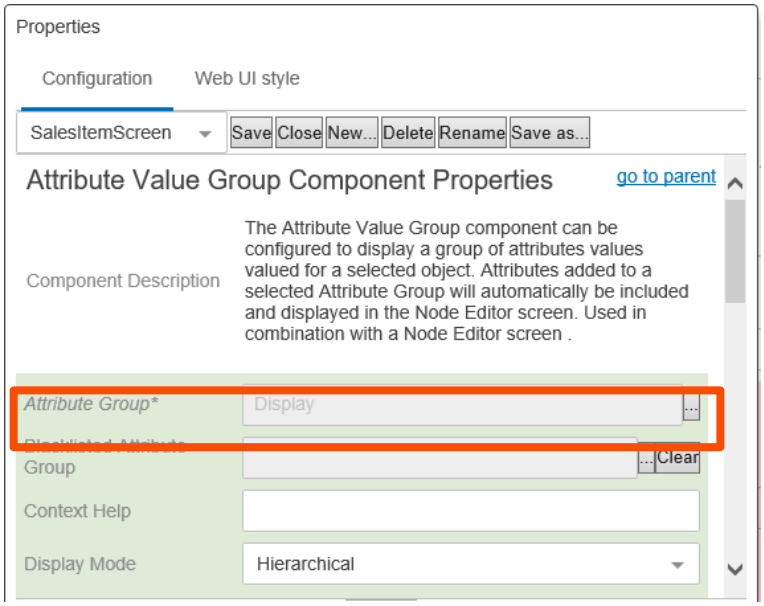
Configuring attribute groups for the WebUI screen which contain large amounts of attributes will result

- In first listing fetching all attributes from the attribute group
- And then filtering out attributes for display based on validity of for the product type and based on user privileges.

When the WebUI screen is configured to display the attributes from an attribute group with many attributes, then that may have a negative impact on the loading time of the screen.

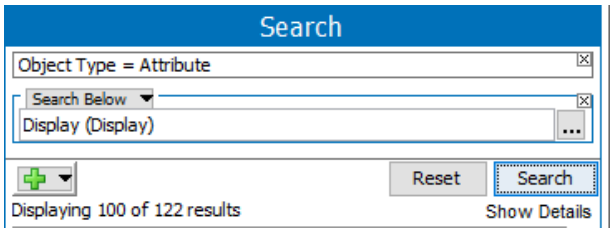
First analyse which attribute groups are used in the WebUI screen.

For example:



Then analyse how many attributes each attribute group contains.

For example:



It's recommended not use attribute groups with more than about 100 attributes. In case many configured attribute groups consequently contain more than about 100 attributes, then consider introducing attribute groups specifically for WebUI display and organize the attributes to be displayed on the WebUI screens in these WebUI specific attribute groups.

B.5.3 Avoid using images in multi-select WebUI screens

Another typical WebUI screen is a multi-select screen where a selection of the items is displayed in a table. The table view allows for thumbnails of the items to be displayed as well. 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.

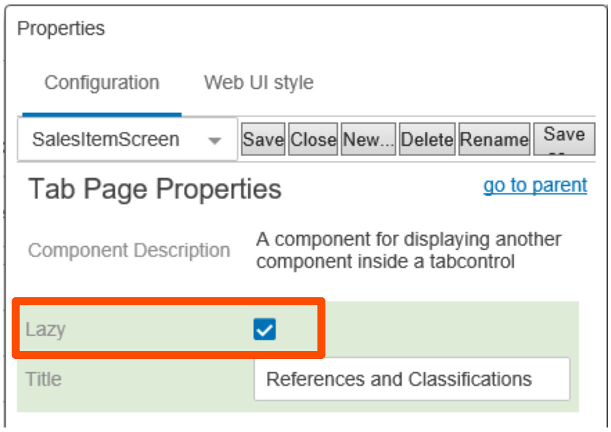
It's recommended to reconsider displaying thumbnails in multi-select screens when many items are displayed and load time of the screen is considered slow.

B.5.4 Use Lazy Loading for WebUI screens

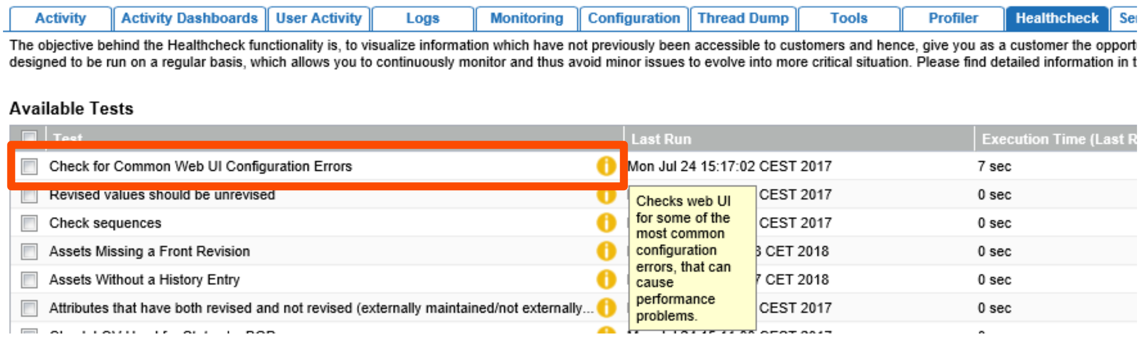
For WebUI tasks where the user needs to inspect many aspects of an item, it's recommended to split the information out into multiple tabs. However, the WebUI loads the screen including all tabs by default, which may result in a slow loading time for the WebUI screen.

There's an option in the WebUI configuration to apply `Lazy Loading` on these `Tab Page` screens. When activating, components will be 'lazy loaded' from STEP and rendered only when needed (i.e., when a tab is selected). This can reduce screen load time since only the needed components are loaded and visible to the user.

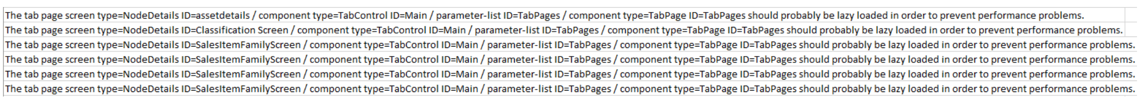
For example:



The easiest way to analyse which WebUI screen can be lazy loaded is to run the STEP Health Check – Check for Common Web UI Configuration Errors.



The result of this health check displays the screens where lazy loading is not applied, for example:


The tab page screen type=NodeDetails ID=assetdetails / component type=TabControl ID=Main / parameter-list ID=TabPages / component type=TabPage ID=TabPages should probably be lazy loaded in order to prevent performance problems.
The tab page screen type=NodeDetails ID=Classification Screen / component type=TabControl ID=Main / parameter-list ID=TabPages / component type=TabPage ID=TabPages should probably be lazy loaded in order to prevent performance problems.
The tab page screen type=NodeDetails ID=SalesItemFamilyScreen / component type=TabControl ID=Main / parameter-list ID=TabPages / component type=TabPage ID=TabPages should probably be lazy loaded in order to prevent performance problems.
The tab page screen type=NodeDetails ID=SalesItemFamilyScreen / component type=TabControl ID=Main / parameter-list ID=TabPages / component type=TabPage ID=TabPages should probably be lazy loaded in order to prevent performance problems.
The tab page screen type=NodeDetails ID=SalesItemFamilyScreen / component type=TabControl ID=Main / parameter-list ID=TabPages / component type=TabPage ID=TabPages should probably be lazy loaded in order to prevent performance problems.
The tab page screen type=NodeDetails ID=SalesItemFamilyScreen / component type=TabControl ID=Main / parameter-list ID=TabPages / component type=TabPage ID=TabPages should probably be lazy loaded in order to prevent performance problems.

It's recommended to run the health check and configure WebUI screen and tab to be lazy loaded, so that the information only loads if the user clicks it.

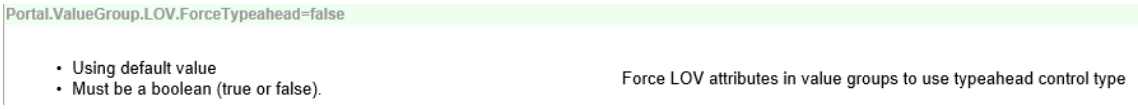
B.5.5 Use Type Ahead for LOVs in WebUI screens

For WebUI screens using attributes with lists of values (LOVs), the load time of the WebUI screen can become slower when these attributes have an LOV with large number of values.

However, it is possible to make the LOV type-ahead when using it in the WebUI. This option comes in two flavours:

- Force type-ahead for all LOVs in the WebUI. This is done via a property setting in the `sharedconfig.properties` configuration file:

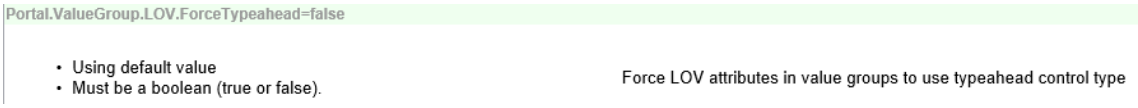
  `Portal.ValueGroup.LOV.ForceTypeahead=true`.

  Use this configuration when there're attributes used in WebUI screens and many of the LOVs used are large.

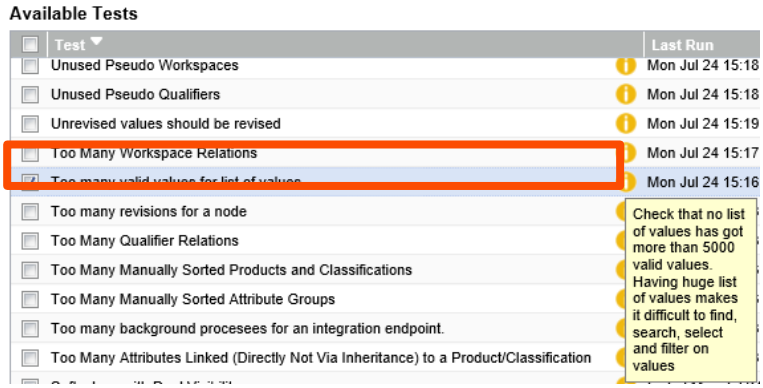
- Force type-ahead for certain LOVs in the WebUI. Certain specific LOVs can be configured not to be forced to be type-ahead based. This is done via a property setting in the `sharedconfig.properties` configuration file:

  `Portal.ValueGroup.LOV.ForceTypeahead.Exclude`

  Be aware that this property excludes the LOVs to be typed-ahead, thus use this for the LOVs with small number of values and for the LOVs which are required not to be typed-ahead.

The easiest way to analyse which LOVs have a large number of values, is to run the STEP Health Check – Too many valid values for list of values.



However, this health check only checks for LOVs with more than 5000 values. Therefore, it also makes sense to check the WebUI screens themselves and determine where large LOVs are displayed.

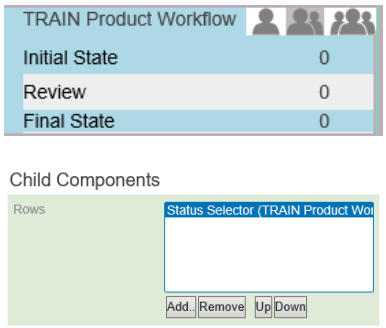
It's recommended to use to optimize the WebUI screens by using the type-ahead options when attributes have large LOVs.

B.5.6 Configure Status Selectors in WebUI correctly

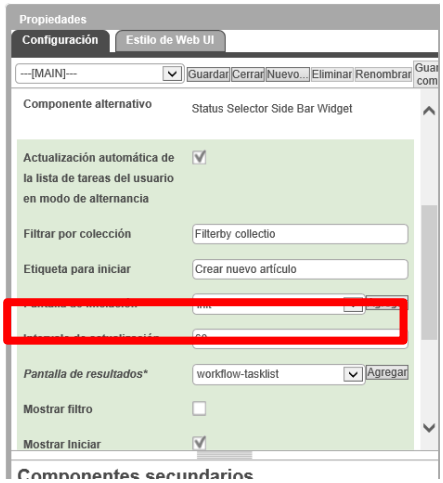
WebUI screens with workflow Status Selectors are configured to poll for updates in the background. The Status Selectors update whenever the user submits an item. However, if there're many status selectors on one WebUI screen and / or the update polling is set very fast, the WebUI screen may suffer from bad performance.

First, keep the number of status selectors in a WebUI to a minimum. A rule of thumb is not to have more than 25 status selectors in one WebUI screen for performance reasons.

For example:



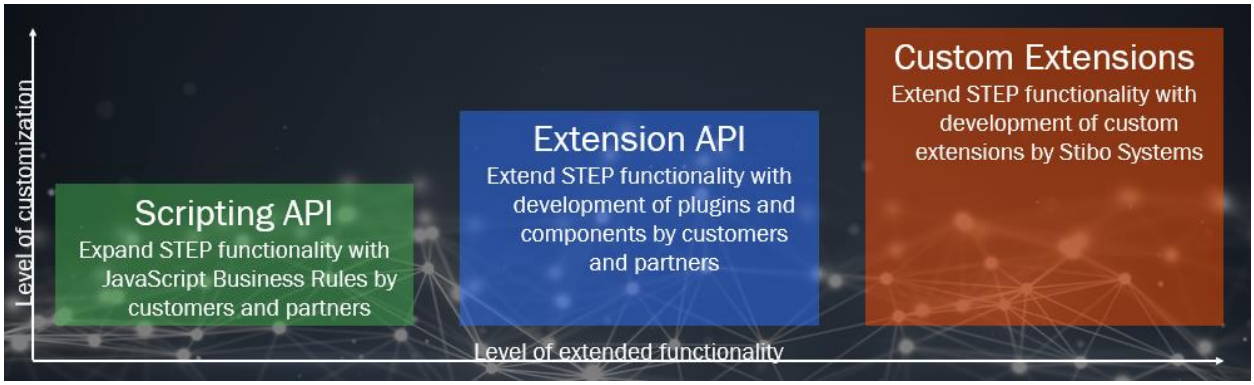
Secondly, the polling interval is set in configuration of the Status Selector Properties. Do not set the polling interval to less than a minute. The default value is 60 seconds. Disabling the refresh interval is done by setting the value to 0.



It's recommended set the refresh interval on 60 seconds (default) or more and keep the number of status selectors to a minimum (less than 25).

B.6 Business rules

Extending STEP functionality can be achieved in three ways, each with their own level of customization and ability to achieve the desired functionality.



Business rules are one of the options to extend the STEP functionality with customer specific business logic.

Business rules can be tied to e.g. imports, approval processes, workflows, WebUI screens, etc. and provide a flexible way to tailor the core functionality in a very precise manner.

It is however important to understand that with this kind of flexibility comes a high level of responsibility. The system load of the business rule execution may have effect on the performance of STEP. If there're a lot of business rules running and if the business rules are complex, then this will degrade performance simply because STEP is very busy processing these business rules.

B.6.1 Business rules analysis

Business rules come in two variants:

- Business Conditions: A Business Condition tests something in relation to an object and will, if applicable, evaluate to either true or false. Conditions should not have side effects (no actions).
- Business Actions: A Business Action does something in relation to an object such as changes data in STEP, sends emails, start Background Processes or Workflows etc.

Business rules are typically used in

- Approval

    - Conditions can be configured to be tested when an object under revision control is sought approved and can allow/prevent the approval.
    - Actions can be configured to be executed on approval and can modify data in STEP (typically data on the object being approved), send emails etc. related to the approval.

- Imports and Inbound Integration Endpoints
    - Conditions tested during imports can allow/prevent the creation or update of objects.
    - Actions executed during import can modify the objects being imported, apply actions related to them, send emails, start Workflows etc.
- Workflows
    - Conditions can be used to allow/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. The Actions can modify the object being tracked by the Workflow, other objects in STEP, modify the Workflow behaviour or send mail, start other Workflows etc.
- Bulk updates
    - Condition can be used as a pre-condition for executing the Action.
    - Actions can be executed via a Bulk Update.
    - Event filtering: Run Event Filter Conditions and Event Generator Actions.

There're several ways to analyse and monitor business rules.

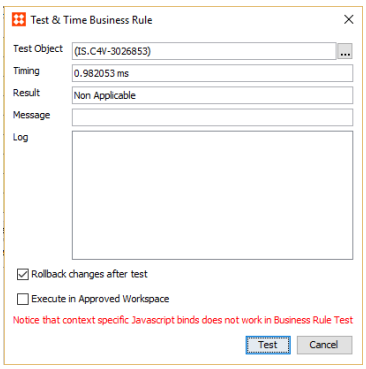
- Workbench business rule test menu

  The Workbench business rule test menu is typically used during development

    - Testing: Right click the business rule and select `Test Business Rule`.
    - Run the business rule a couple of times separate against objects (e.g. products) of which you're certain the business rule will fail or pass.
    - The also analyse the `timing` of the business rule to see its performance.

  When a long running business rule is identified, then use the test menu to test the performance of the business rule.

For example: The following screenshot shows that the business rule took about 0.98 milliseconds to complete the business rule for item `IS.C4V-3026853`. Nevertheless, be aware that it might take longer or shorter for other Items.



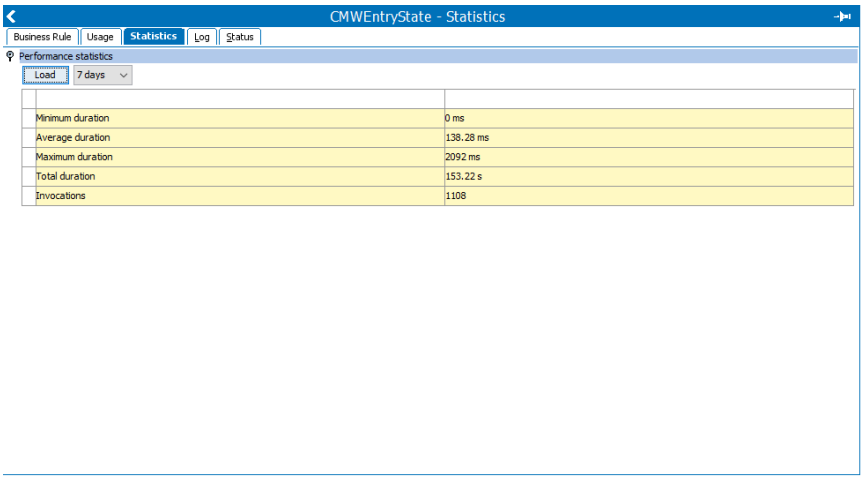
This method of business rule analysis gives a first indication of the performance of the business rule for a certain item.

- Workbench business rule statistics tab

  Secondly, the Workbench provides a business rule statistics tab to see the performance of the business rule over time.

  The business rule statistics tab displays minimum, maximum, average and total duration of the business rule as well as the number of invocations per selected period. The period can be configured to a period of an hour to a week.

  For example: The following screenshot shows the same business rule which was invoked more than 100 times during the last 7 days. That average duration was about 138 ms.



It's possible to click on the maximum duration of about 2092 milliseconds. This shows which item the business rule took longest to execute.

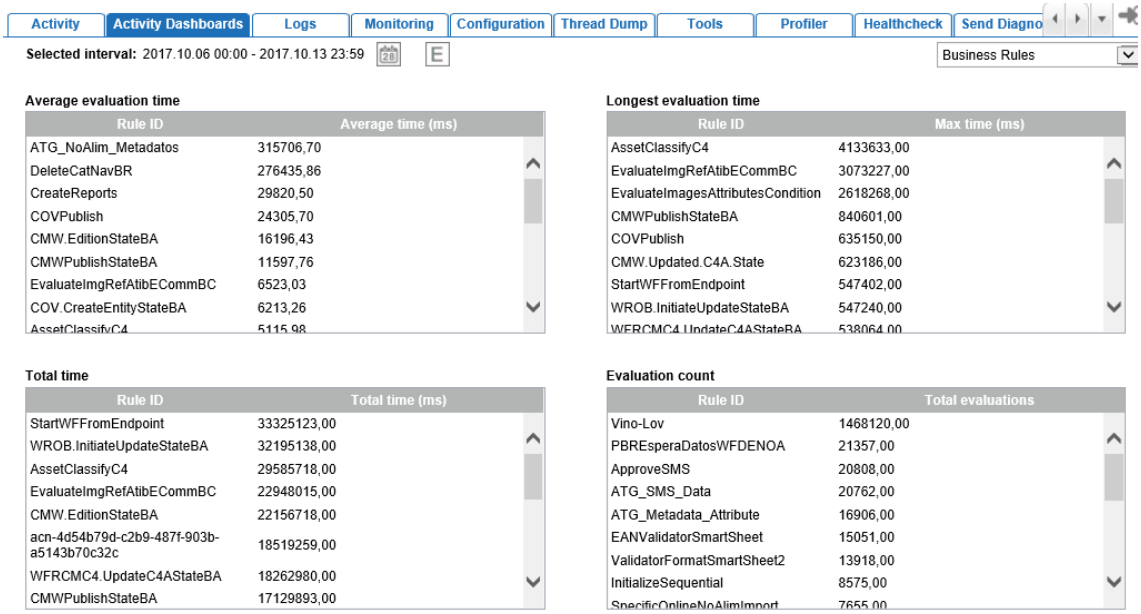
This method of business rule analysis gives an indication of the performance of the business rule over a period of time.

- Admin Portal activity dashboard for business rules

  The Admin Portal provides the possibility to track and trace business rules performance over a given period.

  The dashboard for business rules is available in STEP Admin Portal > Activity Dashboard > Business Rules.

  For example:



  The period over which the statistics are gathers can be configured. 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
  - The number of invocations

This method of business rule analysis gives an indication of the performance of the most demanding business rule over a period of time. Most important is to analyse the business rules stated under “Total time” since these are the business rules with the longest average evaluation time and the most number of invocations.

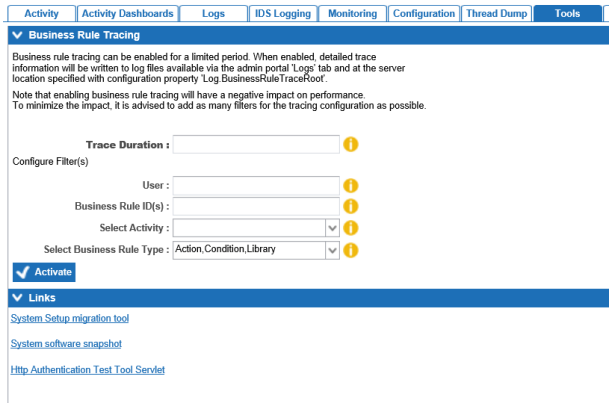
- Admin Portal business rule tracing

  There’s an option in the Admin Portal of version STEP 8.1 and higher to trace business rules. The functionality of the Business Rule Tracing section of the Tools tab is described within the interface itself.

  Business rule tracing can be enabled for a limited period. When enabled, detailed trace information will be written to log files available via the admin portal 'Logs' tab and at the server location specified with configuration property `Log.BusinessRuleTraceRoot`.

  Note that enabling business rule tracing will have a negative impact on performance. To minimize the impact, it is advised to add as many filters for the tracing configuration as possible.

  Click the yellow information icon next to each parameter for a complete description of the parameter / filter and any relevant information for populating it.



  When the necessary information has been added, click the Activate button to begin tracing.

  Note: Once tracing has been activated, the relevant business rule(s) must be triggered in STEP within the time frame defined in the Trace Duration parameter so that the rule is active for tracing. Furthermore, if the system is stopped or restarted, any tracing that was in progress will also be stopped.

  Tracing will stop automatically when the specified duration has expired. Alternatively, users can click the Stop button (available only when tracing is in progress) at any time to kill the trace prior to completion of the duration.

The detailed trace information will be written to log files available via the admin portal 'Logs' tab and at the server location specified with configuration property `Log.BusinessRuleTraceRoot`.

B.6.2 Do not update data via business conditions

Business rules come in two variants:

- Business Conditions: A Business Condition tests something in relation to an object and will, if applicable, evaluate to either true or false. Conditions should not have side effects (no actions).
- Business Actions: A Business Action does something in relation to an object such as changes data in STEP, sends emails, start Background Processes or Workflows etc.

It's highly recommended not to update data with business conditions, but with business actions.

- Use only business actions to manipulate data and use business conditions to validation conditions.
- Do not try to implement side effects in a business condition. In many cases, the STEP system assumes that business conditions have no side effects. This is also why business conditions natively do not have a transaction.

It's recommended to analyse if there are business conditions which manipulate data or have other side effects. If found, then fix these accordingly.

B.6.3 Keep business rule transactions small

Business actions have a transaction, which allows you to manipulate data in STEP.

However, business actions with long transactions will degrade the performance. Furthermore, STEP runs with optimistic locking policy. The longer the transaction, the larger is the probability of introducing an optimistic locking failure when running the business action simultaneously. See paragraph B.3 Optimistic locking for more information.

Business rule transactions therefore must be kept small. This means that it's necessary to develop business rules in full control of even the worst-case scenario.

Traversal of a substantial percentage of the complete data range should be avoided, and changes to data must be kept local (to the nearest surrounding of your main data object).

Therefore, do not introduce 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. This will cause severe performance problems. Rather, use a bulk update or a customer specific background process.

Alternatively, if the business rule transactions cannot be simplified and optimistic locking failures still frequently occur due to the business rule, then consider setting `Reference Target Lock Policy` to

`Relaxed` of these object types the business rule applies to. See paragraph B.3.3 Recommendation: set reference target policy for more information.

B.6.4 Avoid large business rule libraries

Be aware that JavaScript libraries are compiled each time on execution. Each dependency is stacked into a big script and this one is compiled before each execution of the business rule by using the `ScriptEngine.eval()` method.

For example: Script S is depending on library A and library B. Library A is depending on B. In this case the script is stacked as follows:

- A::script
- A::B::script
- B::script
- S::Bindings
- S::script

STEP caches the scripts instead of reloading them from the database. By default, 100 business rules are cached. Still, as a rule of thumb, it takes about 500 milliseconds to compile about 8,500 lines of code at each business rule execution. Therefore, be careful with huge libraries, especially if these are dependent on each other and are used in many business rules, since these libraries will be compiled every time the business rule is executed.

Even if the large library is broken down into multiple libraries, but the libraries still maintain to be dependent on each other, then still it doesn't resolve the issue.

It's recommended to analyse which performance increase can be achieved by making the library functions local to the business rule itself, in case large libraries are used and the business rule suffers from bad performance.

B.6.5 Avoid infinitive loops

Be very careful not to introduce an infinite loop in a business rule.

Infinitive loops will lead to severe degrade of the performance of the affected application server(s) and ultimately make the entire STEP installation irresponsive.

It's recommended to analyse its business rules and determine if infinitive loops are avoided.

B.6.6 Apply exception handling

If an error occurs during approval, an exception is thrown from the domain layer. If this exception is catched in business rules but not re-thrown, it will not reach the exception approval handler. In this case

you can end up with objects that can be inconsistent where some parts are approved and others are not. This behaviour also has a negative effect on the performance of the business rule.

It's recommended to pay attention to proper exception handling to avoid this behaviour when using "try-catch" in business rules.

- What is NOT recommended

  Exception is not re-thrown and therefore will not reach the exception approval handler which may cause inconsistent objects (some parts are approved and others are not)

```
try {
            node.approve();
} catch (exception) {
}
```

  - What IS recommended

    Exception is re-thrown and therefore will reach the exception approval handler avoiding inconsistent objects

```
try {
            node.approve();
} catch (exception) {
            throw exception;
}
```

For example:

```
try {
   currentObject.createReference(targetProduct, accRefType.getID());
} catch (e) {
   if (e.javaException instanceof com.stibo.core.domain.UniqueConstraintException) {
      logger.warning("Cannot create reference - reference already exists, or reference is single o
   } else {
      throw (e);
   }
}
```

It's recommended to analyse its business rules and determine if the recommended way of business rule exception handling is used. The exceptions are handled correctly (re-thrown) if the "try-catch" is used in business rules.

B.6.7 Set business rule logging

STEP provides the option to set the detail of business rules warnings and errors which should be logged in the log file. Logging many details may have a negative impact on the performance, simply because the STEP system will be by logging these details.

It's therefore recommended to configure the business rule logging properly to avoid unnecessary logging of business rule details.

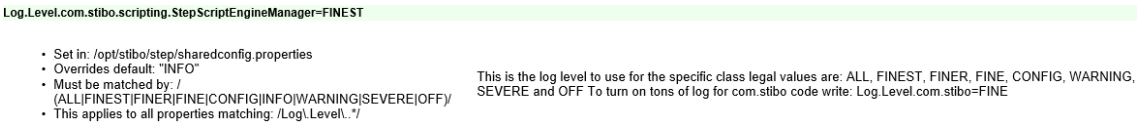
- Use the `BusinessRule.Warning.Threshold` configuration property in `sharedconfig.properties` to specify a threshold in milliseconds for business action execution. If it takes longer to execute or test a given business action, a warning is posted in the main STEP log file.

  The value is in milliseconds. So, `BusinessRule.Warning.Threshold=10000` means write warning to the STEP log file whenever a business rule execution takes longer than 10 seconds.

- Use the `Log.Level.com.stibo.scripting.StepScriptEngineManager` configuration property in `sharedconfig.properties` to specify the level of detail for logging business rules results.

  The values are `ALL|FINEST|FINER|FINE|CONFIG|INFO|WARNING|SEVERE|OFF` and use the appropriate level for each STEP server environment consciously. For example:

  - Set the log level details on STEP DEV and STEP TEST to FINE to trace errors.
  - Set the log level details on STEP QA to INFO or WARNING.
  - Set the log level details on STEP PROD to SEVERE.



- For the logging of business rules, it's good practise to log the result of business rule during development on the development server, but remove the logging when development of the business rule is successfully finished and deployed to the test, quality and production servers.

  The use of business rule logging can be analysed by analysing the STEP log file. In case the log file contains business rule remarks and results, then the business rule logs to the log file.

  An easy and transparent way to turn logging on and off, is to set a Debug Flag (y/n) in the business rule code.

  For example:

```
//Debug 'flag' DO NOT use unless you develop or test

//When doing tests DO NOT test on large amount of products

//REMEMBER to turn 'false' when you are done

var isDebug = false;

//Function to handle whetever logging of debug information should occur or not

function logDebug(message) {

                if(isDebug) {logger.info(message)}

}

...

logDebug("Here's a message for the log file")

...
```

It's recommended to avoid unnecessary business rule logging by configure the business rules logging settings in the `sharedconfig.properties` configuration file and set a logging debug flag in the business rules code itself.

B.6.8 Avoid the function GetChildren with many nodes

Business rules using calls "getChildren" on a huge number of children may cause memory problems. It should not be possible to expand a node having more than 10.000 children.

The problem that the "getChildren" uses an unsafe call that will read all children. It should be changed into using "queryChildren".

```
TreeNavigatorServiceBean

        public List getChildren(final String internalId, final Class wantedChildType,
                                final Manager manager) {
...
                        } else if (obj instanceof Product) {
                                Product product = (Product) obj;
                                children = product.getChildren(); // also works for product overrides <-- Unsafe!!!
```

It's recommended to analyse the business rules and determine if "getChildren" is not used on a selection with more than 10.000 children. Otherwise, change it to "queryChildren".

B.6.9 Use arrays instead of multiple read calls

Business rules using calls to the database for large sets of data again and again, will perform significantly worse than using one call to get the data, push it into arrays and work from there, because the number calls to the database will be minimized

Also in situations where multiple business rules are executed sequentially (e.g. as part of an approval process) and these business rule fetch the same data from the database multiple times again and again, it's worth to rewrite these business rules to fetch the data once, push the data into (multi-dimensional) arrays, and use the arrays, instead of reading the data from the database again and again.

It's recommended to analyse the business rules and determine if data is read efficiently from the database (once instead of many times) and arrays are used. Change the business rule to read data from the database preferably once and use arrays to work with the data in the business rule.

B.6.10 Consider using an extension

When all of the above doesn't bring the desired performance results and the business rules keep performing slowly, then it's worth considering using an extension instead of business rules. Some of the code using in JavaScript (business rules) might run faster in Java (extensions).

An example is the option to develop a custom event processor instead of executing many complex business rules on import of data.

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 were 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 still 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 endpoint configured with a business rule pre-processor. Simply execute import business logic in the pre-processor, then discard the event so that no exports are ever produced by the endpoint. However, endpoints do not use multi-threading while executing the pre-processor, therefore the execution time may be longer than if a custom event processor is used.

However, such a 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, please consider other options.

- The business rules running in such a processor, will have less information available than the same rule running during import, as the object cannot be inspected in both its previous and current form as easily.

- Only cases where extreme amounts of logic are executed should be considered for this solution. And in such cases, it should also be considered why it is necessary to run such complex logic.

B.7 Imports

Bulk data imports in STEP are executed manual or automated. Automated bulk data imports are via the standard Inbound Integration Endpoints (IIEPs) in STEP which is the standard tool for bulk message based data import into STEP.

Bulk data imports into STEP may can vary from very fast to slow, highly depending on the expectations and on following up the good practices for imports described in this paragraph. A clean, simple, optimized import without any processing may perform about a hundred records per second, whereas a cluttered, complex import with complex processing may perform one record per second or even slower.

Whether a fast or slow import is perceived to be a problem or not, highly depends on the situation. A 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 of a large number of records will take weeks to finalize, then this is will probably be problem. Acceptable import performance therefore highly depends on the import scenario and it's therefore important to describe the use cases for imports clearly first.

If the STEP system suffers from perceived bad performance during import, then these recommendations should be followed up.

Please be aware that optimizing the import performance, may have an adversely affect the performance of the system as a whole.

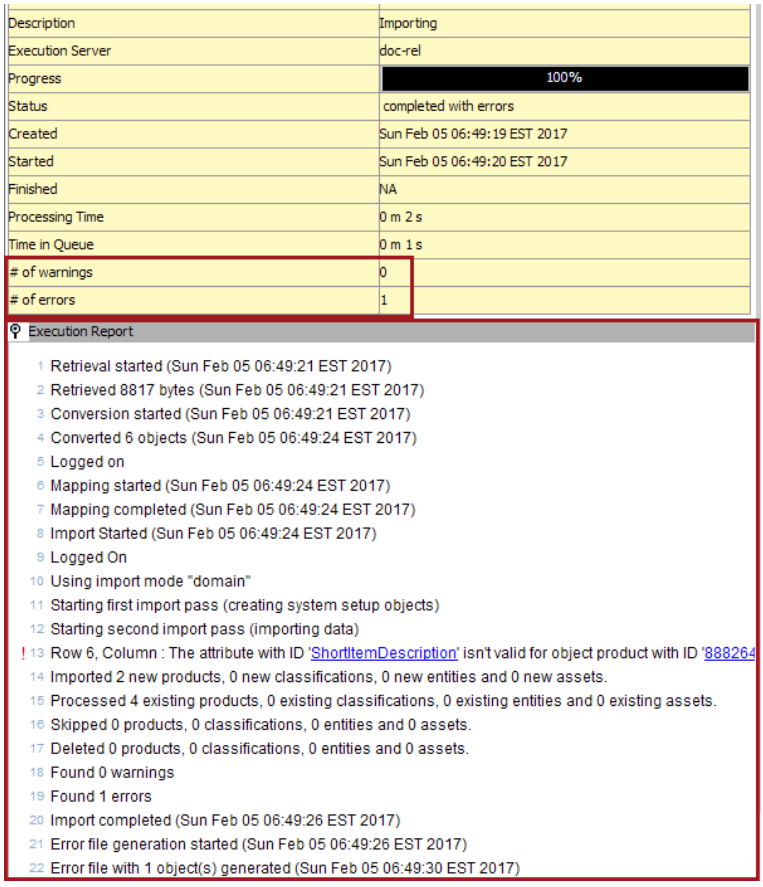
B.7.1 Avoid import errors

It's important to avoid import errors for performance reasons.

The most common types of error messages are described in the STEP Documentation

- Data Exchange > Inbound Integration Endpoints > Import Error Messages
- Data Exchange > Inbound Integration Endpoints > Import Error Message Examples

Import errors are shown in the Background processes under the Execution report and the number of errors and warning are also displayed as shown below.

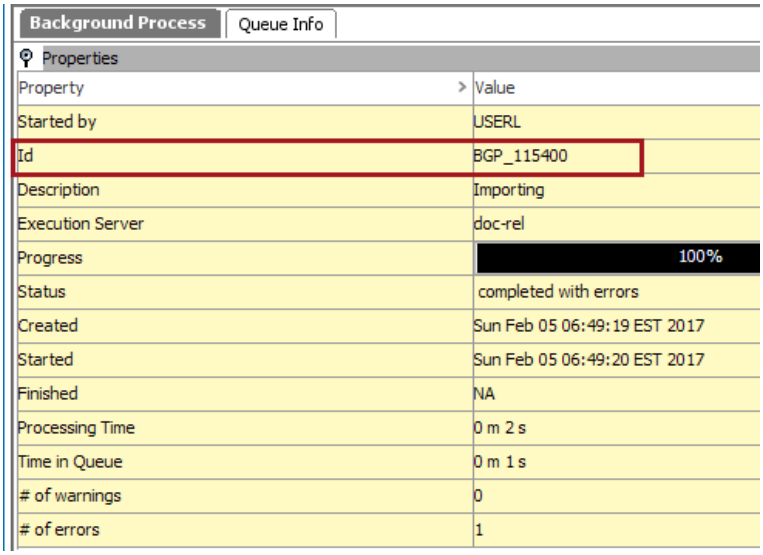


Description	Importing
Execution Server	doc-rel
Progress	100%
Status	completed with errors
Created	Sun Feb 05 06:49:19 EST 2017
Started	Sun Feb 05 06:49:20 EST 2017
Finished	NA
Processing Time	0 m 2 s
Time in Queue	0 m 1 s
# of warnings	0
# of errors	1

Execution Report

```
1  Retrieval started (Sun Feb 05 06:49:21 EST 2017)
2  Retrieved 8817 bytes (Sun Feb 05 06:49:21 EST 2017)
3  Conversion started (Sun Feb 05 06:49:21 EST 2017)
4  Converted 6 objects (Sun Feb 05 06:49:24 EST 2017)
5  Logged on
6  Mapping started (Sun Feb 05 06:49:24 EST 2017)
7  Mapping completed (Sun Feb 05 06:49:24 EST 2017)
8  Import Started (Sun Feb 05 06:49:24 EST 2017)
9  Logged On
10 Using import mode "domain"
11 Starting first import pass (creating system setup objects)
12 Starting second import pass (importing data)
! 13 Row 6, Column : The attribute with ID 'ShortItemDescription' isn't valid for object product with ID '888264
14 Imported 2 new products, 0 new classifications, 0 new entities and 0 new assets.
15 Processed 4 existing products, 0 existing classifications, 0 existing entities and 0 existing assets.
16 Skipped 0 products, 0 classifications, 0 entities and 0 assets.
17 Deleted 0 products, 0 classifications, 0 entities and 0 assets.
18 Found 0 warnings
19 Found 1 errors
20 Import completed (Sun Feb 05 06:49:26 EST 2017)
21 Error file generation started (Sun Feb 05 06:49:26 EST 2017)
22 Error file with 1 object(s) generated (Sun Feb 05 06:49:30 EST 2017)
```

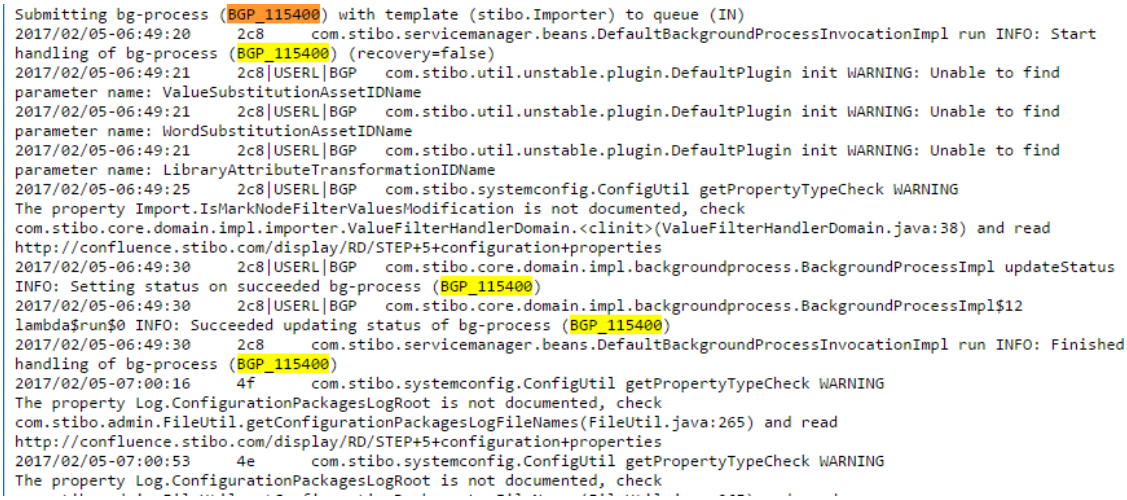
Similarly, the execution report can also be viewed in the log file by looking up the background process ID.



Background Process | Queue Info

Properties

Property	Value
Started by	USERL
Id	BGP_115400
Description	Importing
Execution Server	doc-rel
Progress	100%
Status	completed with errors
Created	Sun Feb 05 06:49:19 EST 2017
Started	Sun Feb 05 06:49:20 EST 2017
Finished	NA
Processing Time	0 m 2 s
Time in Queue	0 m 1 s
# of warnings	0
# of errors	1



```
Submitting bg-process (BGP_115400) with template (stibo.Importer) to queue (IN)
2017/02/05-06:49:20     2c8     com.stibo.servicemanager.beans.DefaultBackgroundProcessInvocationImpl run INFO: Start
handling of bg-process (BGP_115400) (recovery=false)
2017/02/05-06:49:21     2c8|USERL|BGP   com.stibo.util.unstable.plugin.DefaultPlugin init WARNING: Unable to find
parameter name: ValueSubstitutionAssetIDName
2017/02/05-06:49:21     2c8|USERL|BGP   com.stibo.util.unstable.plugin.DefaultPlugin init WARNING: Unable to find
parameter name: WordSubstitutionAssetIDName
2017/02/05-06:49:21     2c8|USERL|BGP   com.stibo.util.unstable.plugin.DefaultPlugin init WARNING: Unable to find
parameter name: LibraryAttributeTransformationIDName
2017/02/05-06:49:25     2c8|USERL|BGP   com.stibo.systemconfig.ConfigUtil getPropertyTypeCheck WARNING
The property Import.IsMarkNodeFilterValuesModification is not documented, check
com.stibo.core.domain.impl.importer.ValueFilterHandlerDomain.<clinit>(ValueFilterHandlerDomain.java:38) and read
http://confluence.stibo.com/display/RD/STEP+5+configuration+properties
2017/02/05-06:49:30     2c8|USERL|BGP   com.stibo.core.domain.impl.backgroundprocess.BackgroundProcessImpl updateStatus
INFO: Setting status on succeeded bg-process (BGP_115400)
2017/02/05-06:49:30     2c8|USERL|BGP   com.stibo.core.domain.impl.backgroundprocess.BackgroundProcessImpl$12
lambda$run$0 INFO: Succeeded updating status of bg-process (BGP_115400)
2017/02/05-06:49:30     2c8     com.stibo.servicemanager.beans.DefaultBackgroundProcessInvocationImpl run INFO: Finished
handling of bg-process (BGP_115400)
2017/02/05-07:00:16     4f      com.stibo.systemconfig.ConfigUtil getPropertyTypeCheck WARNING
The property Log.ConfigurationPackagesLogRoot is not documented, check
com.stibo.admin.FileUtil.getConfigurationPackagesLogFileNames(FileUtil.java:265) and read
http://confluence.stibo.com/display/RD/STEP+5+configuration+properties
2017/02/05-07:00:53     4e      com.stibo.systemconfig.ConfigUtil getPropertyTypeCheck WARNING
The property Log.ConfigurationPackagesLogRoot is not documented, check
```

The most common errors encountered during data import refer to an invalid attribute value. Sometimes there is a mismatch with the attribute's validation type, other times there is a mismatch with one of the other setups. For example, the attribute's constraints, such as masks, minimum and maximum values, maximum length, LOVs, the object's own object type, and so on. Therefore, when you encounter an error, look at the attribute setup and review the constraints.

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.

Make sure the statements in the STEPXML are typed correctly, for example

- If the import file is having wrong case for the property `UserTypeID` such as `UserTypeId`, then the performance will degrade.

- If the import STEPXML file has no Parent IDs given for the products, then the new products are not getting imported. If you do not want to import new products, then add “Reject new” product tag in the file, so it will not try to import those and give an error for missing Parent IDs. So, put `RejectNewProducts="true"` in the header of the STEPXML file in that case.

Recommended is to analyse the import errors and fix these accordingly.

B.7.2 Import file - Avoid missing targets

The structure of the import file can be optimized for import performance.

Ensure that your import files do not have missing reference targets, that is reference nodes which are missing in the file.

If the reference target doesn’t exist in the same import file, then the target does not exist at the time of import. The importer will skip the product and trigger a second import pass after the initial completion. The reference target doesn’t exist in the system at the time of the second import pass, the background process will log an error due to missing reference target.

A second import pass caused by missing targets bears a heavy performance impact. The sequence of the nodes in the import files is important. Therefore, analyse the import files and determine if there’re no missing reference targets.

B.7.3 Import file - Avoid forward declarations

The structure of the import file can be optimized for import performance.

Ensure that your import files do not have forward declarations, that is reference nodes being created later in the file.

If the reference target existed later in the same import file, then the target does not exist at the time of import. The importer will skip the product and trigger 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. The sequence of the nodes in the import file is important. Therefore, analyse the import file and determine if there’re no forward declarations.

B.7.4 Import file - Avoid multiple updates of same object in one import file

The structure of the import file can be optimized for import performance.

When creating import files, ensure that the same product or entity is not updated several times in the file, for maximum performance.

When creating import files, ensure to structure the files by modified nodes, and not by attribute. It is faster to update a single node with 10 attributes once, than updating each attribute individually.

B.7.5 Import file - Avoid updates of same object over multiple import files

The structure of the import file can be optimized for import performance.

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.

B.7.6 Import file - Use Term Lists for price data

The structure of the import file can be optimized for import performance.

Use Terms Lists for price data where appropriate. Especially for complex, time-limited / quantity limited price data, consider using terms lists for to speed up imports.

B.7.7 Business rules on import - necessity

Using business rules on imports is a powerful tool, but like all powerful tools comes with some risks.

Complex JavaScript business actions will add execution time to imports. Therefore, consider if any automated actions are necessary on import. The less automated actions on import, the faster the import itself will perform.

Consider the business process and consider if it's necessary to have all or even any actions immediately on import. Try to architect your solution so that imports can run 100% free of any business actions wherever possible.

The same consideration is valid to any action that happens automatically on import, such as workflows, approvals, etc. All of these are extremely useful, but try to minimize their performance impact.

See B.6 Business rules for more information on business rules.

B.7.8 Business rules on import - avoid optimistic locking

Using business rules on imports is a powerful tool, but like all powerful tools comes with some risks.

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 the reporting of 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 exception with the actual word “exception” in the error message will cause the importer to believe that an optimistic lock occurred, resulting in retries and eventual failure.

See B.6 Business rules for more information on business rules.

See B.3 Optimistic locking for more information on optimistic locking failures.

B.7.9 Business rules on import - design for performance

Using business rules on imports is a powerful tool, but like all powerful tools comes with some risks.

If there are business rules on import, then fully map the business rule execution of each import (especially endpoints) carefully 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.

- If there’s complex business rule JavaScript logic being executed on the import, then try to simplify the complexity.

- 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’s possible to ensure that no business conditions exist allowing a logical exception or lock contention. Imports should, above all else, succeed in quickly importing data with absolute reliability. 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.

- Pay attention to duplicate executions. For example, an approval triggered in a global business rule as well as in a local state transition rule. If so, then ensure no such double-checking occurs.

See B.6 Business rules for more information on business rules.

B.7.10 Design workflow initiations on import for performance

Consider whether the import results in workflow initiations and/or transitions.

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 will execute as part of the import process

- If workflow initiations and/or transitions are necessary, then consider if the business rules trigger on entry or exit. If a logic is executing on these transitions, the import performance may be heavily impacted.
- Also consider if these business rules initiate more workflows or auto submit to other states. A single workflow submit may cascade into hundreds or even thousands of lines of executed business rule logic. Ensure this does not happen.
- When a workflow business rule conditions fails, the current transaction will be rolled back which impacts performance heavily.

B.7.11 Design approvals on import for performance

Consider whether the import results in approvals.

If approvals are necessary at import, then consider which approval conditions and actions will be executed.

Try to ensure that endpoints are importing externally-maintained data. Externally maintained data requires no approvals, therefore consideration for approval conditions and actions is not required. Externally-maintained data has no revision history. Therefore, revision history growth is not a consideration.

B.7.12 Design event-driven exports for performance

When importing externally maintained data, or importing and approving, all changes must be checked against any event-driven Outbound Integration End Points, to check if an event should be generated.

Consider which approval events are queued on Outbound Integration End Points and which event filter / generator rules will be triggered. If a large number of Outbound Integration End Points exists, and if these listen on attribute groups, the system must, for each, make the check “Does this attribute exist under the given attribute group”. This can lead to performance degradation.

Ensure that the Outbound Integration End Points to as specific and few attributes as possible.

B.7.13 Avoid complex privileges at import

When importing, all privileges are checked, for each piece of information imported. In the cases where the data is not imported, all privileges are checked as well.

This 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 imports.

Therefore, ensure that any imports happen as a user with as broad and few privileges as possible to avoid excessive privilege-checking. Limiting the number of privileges for the importing user can improve performance dramatically. Only the user configured on the endpoint as the importer is relevant, this import user should generally have relatively few permissions, to ensure performance.

B.7.14 Use parallel imports for the same Inbound Integration End Point

There's a way to configure the Inbound Integration End Point (IIEP) so that the import files of the IIEP are imported in parallel. This might be convenient if you want to do an import of many separate import files which need to be imported via the same IIEP.

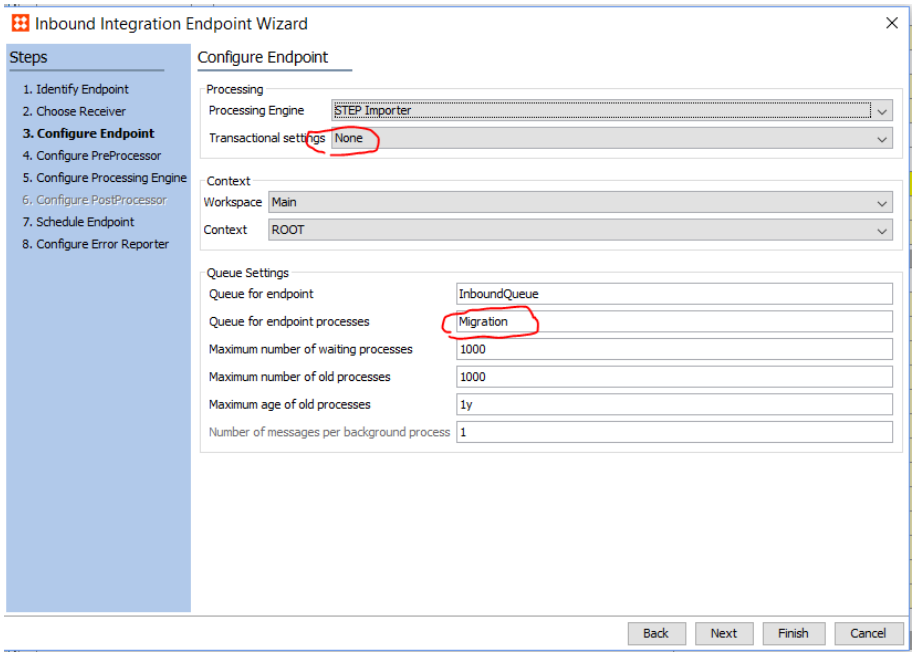
However, if the separate import files consist of the same objects, then this might result in optimistic locking errors which will degrade import performance. Parallel Imports where many products or entities reference a single shared target object (classification link targets do not apply here) will encounter lock failures, because by default STEP will take a row lock on the reference target during import.

Resolve or reduce this issue by setting the reference target lock policy to "Relaxed" on the object types that have many reference sources.

Overview of the parallel import configuration:

- Transactional setting = `None`
- Define a separate endpoint process queue, e.g. `Migration`
- Set the size of the queue in `sharedconfig.properties`

Screenshot of the IIEP configuration with Transactional Setting on `None` and Queue for endpoint process on `Migration`.

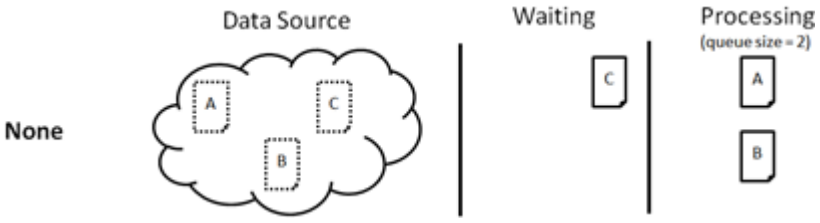


Screenshot of the sharedconfig.properties file with `BackgroundProcess.Queue.Migration.Size=10`

```
#//////////////////#
# Migrsation parallel imports
#//////////////////#
BackgroundProcess.Queue.Migration.Size = 10
```

Explanation of the Transaction Setting is None: Using the 'none' transactional setting processes data in parallel processes, without any transactional restrictions or data dependencies (useful, for example, when processing assets).

This allows the queue size to be larger than one (1). Data is not processed in a strict order and if one background process fails, the endpoint continues to process data in the next background process in the queue.



It's recommended to configure parallel imports of the same IIEP if the parallel imports are possible and if the parallel import doesn't result in optimistic locking failures.

B.7.15 Migration scenarios

The import recommendations apply to operational scenarios on production environments. Nevertheless, the recommendations can also be applied to one-off 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's a one-time operation and the volume of data is generally far greater than a typical import would be expected to process. It's also generally expected that a greater level of effort will be invested in preparing the import messages or files so that the migration can complete 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 / flushed from cache.
- If necessary, parallelize the migration endpoint so that multiple concurrent background processes can be utilized.
- 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 to rectify this. Create one set of import files that contain all information about the products except for references. Load these into a parallelized inbound endpoint. Create a second set of import files containing only references. Load these into a separate serialized endpoint. Both endpoints may be able to run at the same time, so as the data going into each can be offset to the degree that the same products are not presented on one endpoint at the same time as they load on the other.
- Be cautious of business rules that update products other than the one being loaded when using parallelized endpoints, this presents a prime opportunity for optimistic locking. If the need for these types of rules is present, consider using bulk updates to execute the logic after import is complete.

B.7.16 Setup STEP for maximum import performance

The setup of STEP can be optimized for import performance.

- Ensure that any imports happen as a user with as broad and few privileges as possible, to avoid excessive privilege-checking.

- Ensure that there is only a limited number of Outbound Integration Endpoints are listening on change-events, to avoid excessive checking of event validity.
- Make sure to set “Relaxed” locking policy for commonly referenced products to ensure performant imports.
- Try to ensure that endpoints are importing externally-maintained data as much as possible.
- Avoid Optimistic Lock Failures by ensuring not to update the same product at the same time across multiple imports.
- Use parallel imports as needed, but be aware of potential performance degradation across the rest of the system.

Recommended is to optimize STEP for import performance.

B.7.17 Miscellaneous import related optimizations

There’re other miscellaneous import related optimizations possible which are described elsewhere in this document.

- Remove inactive and unused IIEPs: It’s good practice to remove inactive and unused IIEPs on operational STEP systems. There’s no need for these IIEPs in operational production environments and therefore it’s recommended to remove all inactive IIEPs which are not used. When the inactive IIEPs are removed, then also corresponding background processes are removed.
- Set IIEPs to remove files and background processes after import
    - Configure the IIEP to remove the import files after successful import via `files after load = No`
    - Make sure STEP can recognize the import file directories to be able to remove the import files after successful import
    - Configure the IIEP not to keep the import background processes for too long via defining `Maximum number of old processes` and `Maximum age of old processes in hours`.
- Use the standard asset importer: It’s recommended to switch to the standard asset importer if the legacy asset importer is used.
- Make sure assets and other objects are not imported into one huge folder: It’s recommended to import assets and objects in a granular folder structure.
    - The standard asset importer can automatically create the folder structure of assets.

    - Furthermore, make sure to split uncategorized objects products into subfolders.
- Balance the import schedules of the different imports to balance the STEP system load
- Finally, validate the hardware sizing and consider In-Memory to improve import performance

B.8 Exports

One of the primary goals of most STEP implementations is to reduce the time to market requiring the flow of data from STEP to external systems such as ERP systems and e-commerce systems.

If the STEP system suffers from perceived bad performance during export, then these recommendations should be followed up.

Please be aware that optimizing the export performance, may have an adversely affect the performance of the system as a whole.

B.8.1 Use event based exports over static exports

Whilst this can be achieved using static exports it is usually much more efficient to utilise event messaging via an 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 STEP perspective (the time taken from the STEP side to export all data), but also from the receiving system perspective (the time to import the all data).

It's recommended to use event based exports over selection based exports where possible.

B.8.2 Limit the number of Outbound Integration End Points

Having an excessive amount of Event Based Outbound Integration End Points (OIEPs) can adversely affect the performance of the system as a whole.

Don't configure too many event-based OIEPs. Too many OIEPs adversely affect the approval / external attribute change process as each change must be checked against each OIEPs triggering definition to check whether the OIEP is interested in that specific change.

The Stibo recommendation, generally, is having no more than 10 Event Based Outbound Endpoints

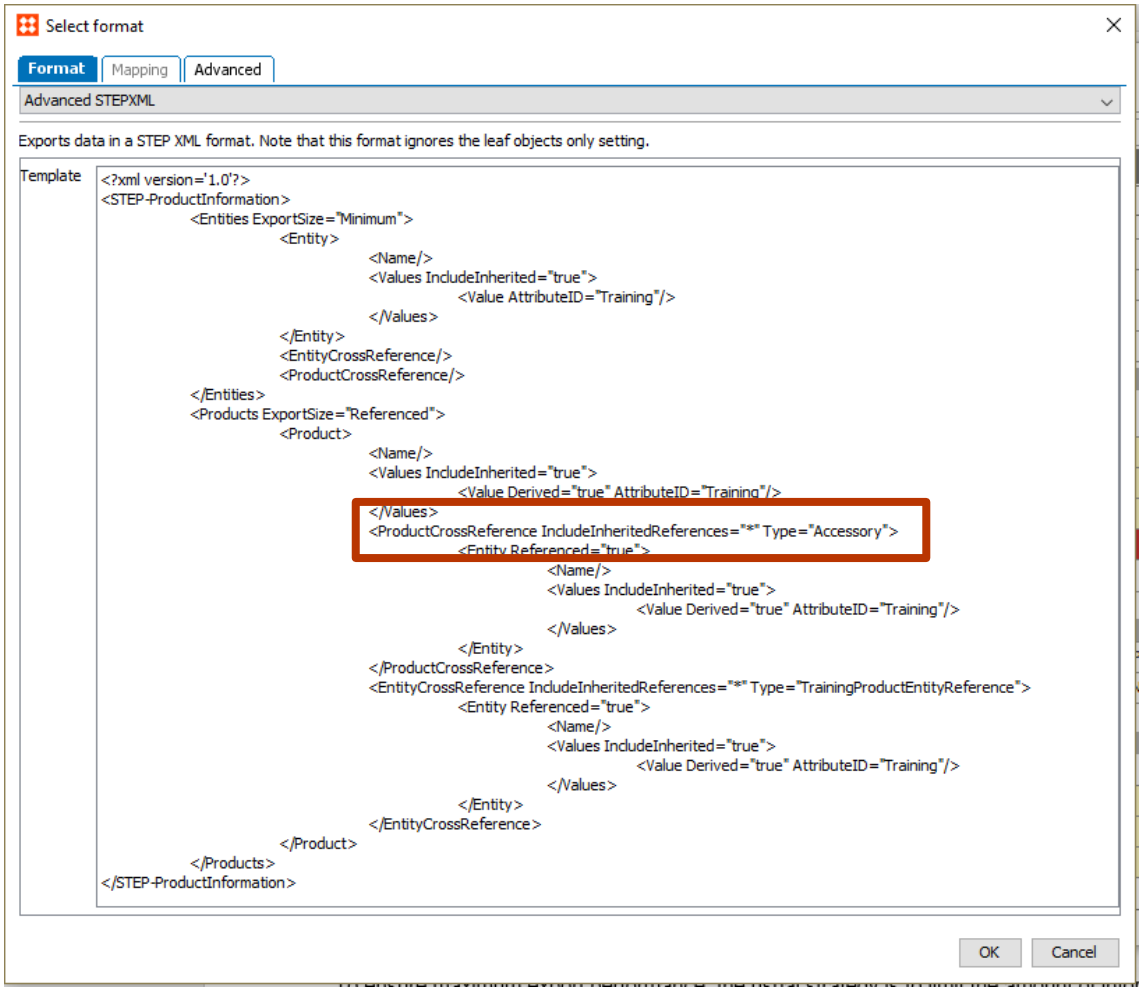
B.8.3 Limit the volume of exported data

To ensure maximum export performance, the usual strategy is to limit the amount of information exported, or limiting the amount of times the same node is exported with the same data.

With Advanced STEPXML it's possible to define what data to export and leave other data out.

For example, only include the export attribute groups instead of including all attributes. Also, the CSV export as well as Generic XML export allows you to configure the data only what's needed.

For example: the following highlighted section in the screenshot exports only the reference `Accessory`.



If's recommended to use Advanced STEPXML exports and limit the exported data set over STEPXML exports which exports the entire data set.

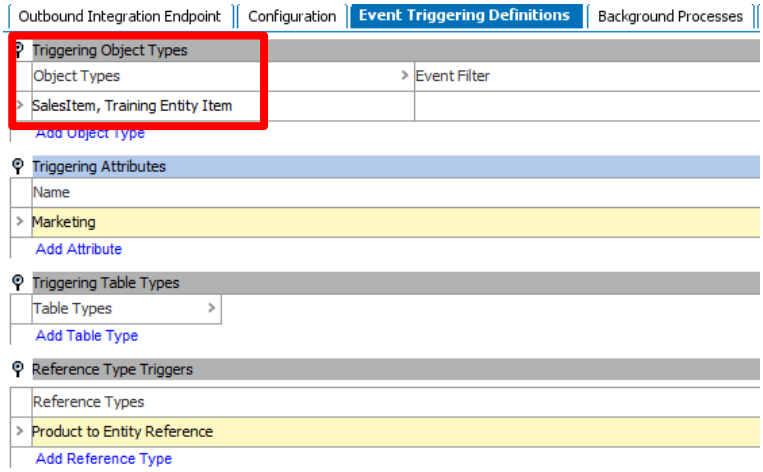
B.8.4 Optimize the triggering definitions with object types

To ensure maximum export performance, the usual strategy is to limit the amount of information exported, or limiting the amount of times the same node is exported with the same data.

The Outbound Integration End Point (OIEP) triggering definition can be optimized by tailoring an OIEP's triggering definitions to ensure that objects aren't exported that aren't required.

For instance, if 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 further downstream.

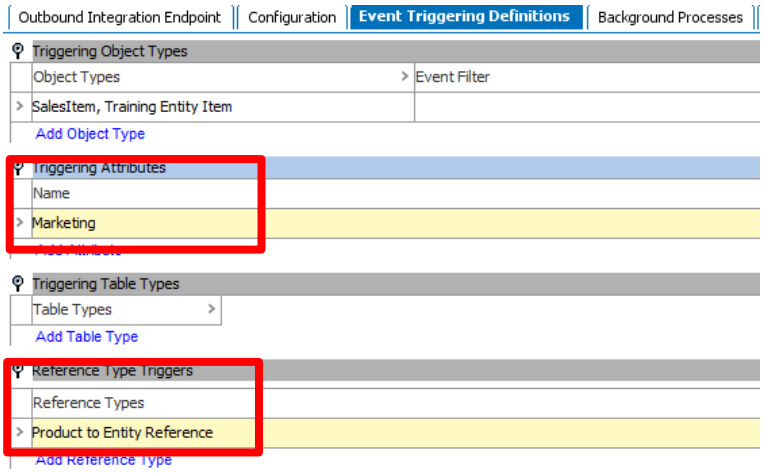


B.8.5 Optimize the triggering definitions with triggering attributes and references

To ensure maximum export performance, the usual strategy is to limit the amount of information exported, or limiting the amount of times the same node is exported with the same data.

The Outbound Integration End Point (OIEP) triggering definition can be optimized by tailoring the OIEP's attribute triggering definitions to ensure that objects are only exported when necessary. An e-commerce system for instance, is probably only interested when certain attributes of a product changes, like its description or similar. Triggering on attributes that are irrelevant for the receiving system, therefore makes little sense.

Tailor an OIEP's triggering definitions for export channels specifically. Triggering attributes can be specified by attribute group and attribute groups can be created specifically for export channels. For example, attribute "WebsiteAttributes" contain only the attribute relevant for the website export channel.



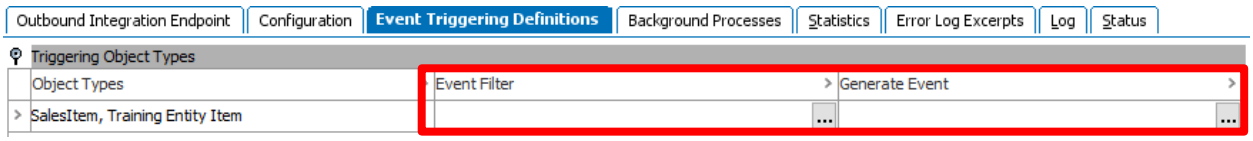
B.8.6 Optimize the triggering definitions with event filter and event generator

To ensure maximum export performance, the usual strategy is to limit the amount of information exported, or limiting the amount of times the same node is exported with the same data.

The Outbound Integration End Point (OIEP) triggering definition can be optimized to tailor the event filtering and event generation possibilities.

For example: Take care when handling inheritance as changes to attributes residing higher up the hierarchy can affect a significant number of products, if all child products are exported. In those cases, consider replacing a single high-level attribute change event with multiple child-level events.

If this is performed using the event filter / event generator mechanism the significant volume of resultant events can be handled in multiple batches which enables output files to be kept to a reasonable size, and opens up the possibility of exporting multiple batches in parallel reducing overall export times.



As we typically no longer use “approval” as a mechanism to determine whether data is suitable for publishing (we often approve much earlier within the product lifecycle) we are often asked to limit the data exported to just “published” SKUs. The event filter and pre-processor functionality can both meet these requirements although they operate in slightly different ways.

Alternatively, we could export all data downstream (related to the underlying events) and a middleware based solution could determine which data to route to which target system.

B.8.7 Consider using 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’s easy to see how the assignment of many products to a classification e.g. “Christmas Gifts” could 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 could be avoided through the use of multiple OIEPs

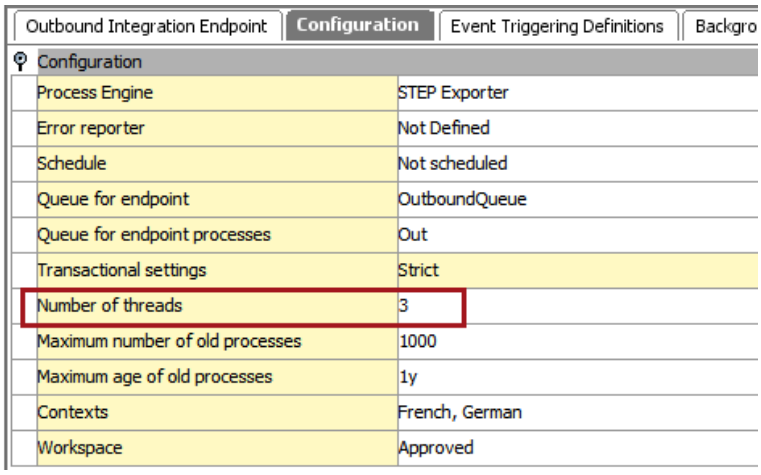
- One OIEP for listening for traditional product (attribute) changes and outputting all data for the product
- Another OIEP for listening for reference changes and outputting minimal data e.g. product ID and target ID.

B.8.8 Consider use multithreading

To increase export performance when exporting from an event-based Outbound Integration Endpoint (OIEP), it's possible to increase the number of threads for a given OIEP.

Common setup is to use multithreading when there is a large amount of data going to a downstream system on a regular basis, and when the downstream system is capable of handling it. Consider the following points before increasing the number of threads to more than one:

- The STEP system hardware should have enough resources to perform with multithreading.
- The downstream (receiving) system must be able to handle parallel events.



Although multithreading can increase export performance, it may have a negative impact on the overall system performance when a large amount of data is involved. Therefore, run the settings in a test environment, starting with a small amount of data and then increasing it, before implementing it in your production system.

It's recommended to consider using multithreading when there is a large amount of data going to a downstream system on a regular basis to improve export performance. However, multithreading may cause overall system performance degrade. Therefore, it's necessary to test it before implementing it in your production system.

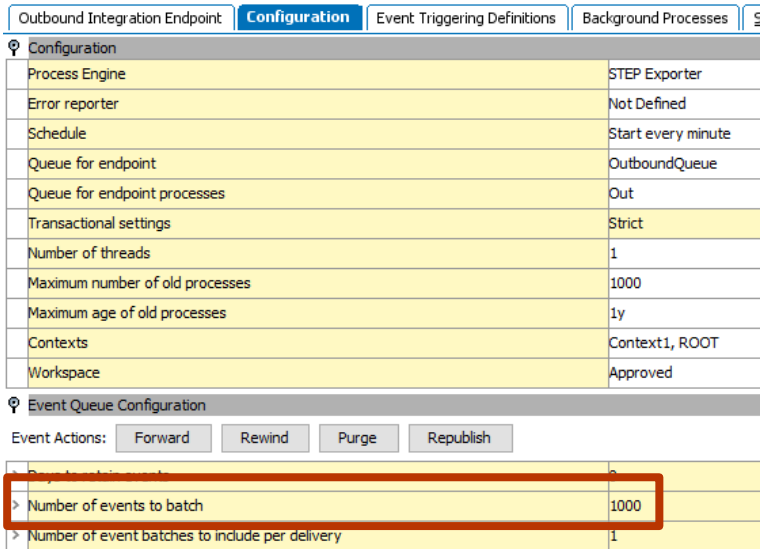
B.8.9 Optimize the batch size

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

However, a small batch-size, or no batching at all, will invoke the export engine and surrounding framework many times (once per batch) incurring a significant overhead.

In addition, a small batch-size could also cause the same product to be exported multiple times in the 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).

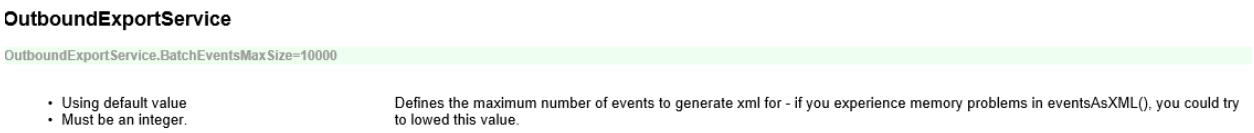


Therefore, consider using a larger batch-size (e.g. 1,000 – 10,000 events) as this will achieve two key aims:

- Larger batches will invoke the export engine and associated framework fewer times incurring less overhead
- Larger batches will reduce the amount of duplicate product data exported

Don't try and generate one product per file by setting the batch-size=1. Whilst this setting does, on the surface, appear to deliver the required results, in practice there are several problems associated with this approach.

Be aware of the batch size specified though the GUI, a system-wide setting is used to limit the number of events included within a batch ensuring that the exporter does not try and export too much data in a single invocation. This configuration in `sharedconfig.properties` is `OutboundExportService.BatchEventsMaxSize`.

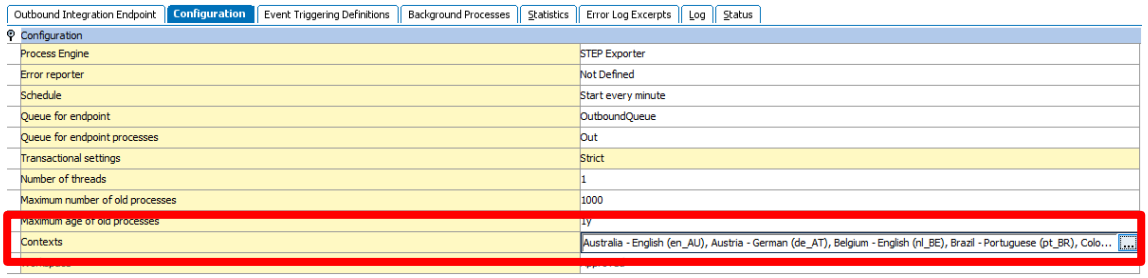


It's recommended to avoid small batch sizes.

B.8.10 Use cross-context exports

One, perhaps surprising, effect of the recommendation to limit how much data is being exported, is the recommendation to generally always 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 try to configure multiple OIEPs each outputting product details for a single context. Consider using a single OIEP configured with multiple contexts as the overall export time will be significantly faster. Subsequently, the context splitter post processor can be used, if separate context files are required.



B.8.11 Limit the inclusion of additional data

Additional data can be included in the export using several different methods

- An event generator can be used to add additional objects by using derived events.
- A pre-processor can add additional objects to the export set associated with the event batch.
- The Advanced STEPXML template can be used to include additional objects via e.g. references.

In all cases consider the impact of adding any additional objects and whether they are really needed as adding many additional objects will slow the export process.

B.8.12 Limit the usage of multiple templates per OIEP

STEP allows the usage of multiple output templates per Outbound Integration End Point (OIEP). Multiple output templates can be associated with a single OIEP enabling the OIEP to handle different types of objects/events, each having the ability to be output in a different format.

This allows the user to totally change how data is exported for different kinds of objects, which allows great flexibility. However, some considerations must be taken, when using this feature. Whilst this does seem appealing, its use can have a severe impact on performance as each time a new output template is required a new batch is created.

An alternative approach could be to have the Family and SKU changes handled by different OIEPs. However, this isn't an option if the events need to be sent in sequence. In this case it's easy to see that the SKU change could be exported, and delivered, before the Family change.

Another solution to the problem is to use STEPXML as this can easily contain the Family and SKU data in the same file.

B.9 Scheduled processes

The scheduled processes are visible in STEP > Workbench > BG Processes > Scheduled Processes.

Scheduled processes might have an influence on the performance of the system, for example if the search query to fill the collection is not optimized.

It's recommended is to analyze the scheduled processes and see if the collections (searches) can be optimized and / or the business rules can be optimized.

See B.6 Business rules for more information on business rules.

See B.12 Searches for more information on searches.

B.10 Revision control

All major data objects in STEP including products, assets, classification, and entities are under revision control. This means that you can go back in history and view and compare old revisions, and revive an older version of an object.

In STEP, historical versions of objects are stored in "Revisions". A revision represents a historical "snapshot" of an object.

New Revisions are created when:

- A revisable object is created
- The object is approved
- The object is modified after it has been approved
- The digital media file associated with an Asset object is changed (digital media files stored in STEP only)
- The object is modified by a user different from the user who triggered the previous Revision

A new revision will not be created with every change to a revised object and thus, STEP does not have traceability on the field level. This means that not all historical versions of e.g. a value are stored, but only the latest version of the value for each revision.

If the same user is making continuous edits to an object, these are not captured as a revision until the object is approved, or another user makes changes to the object.

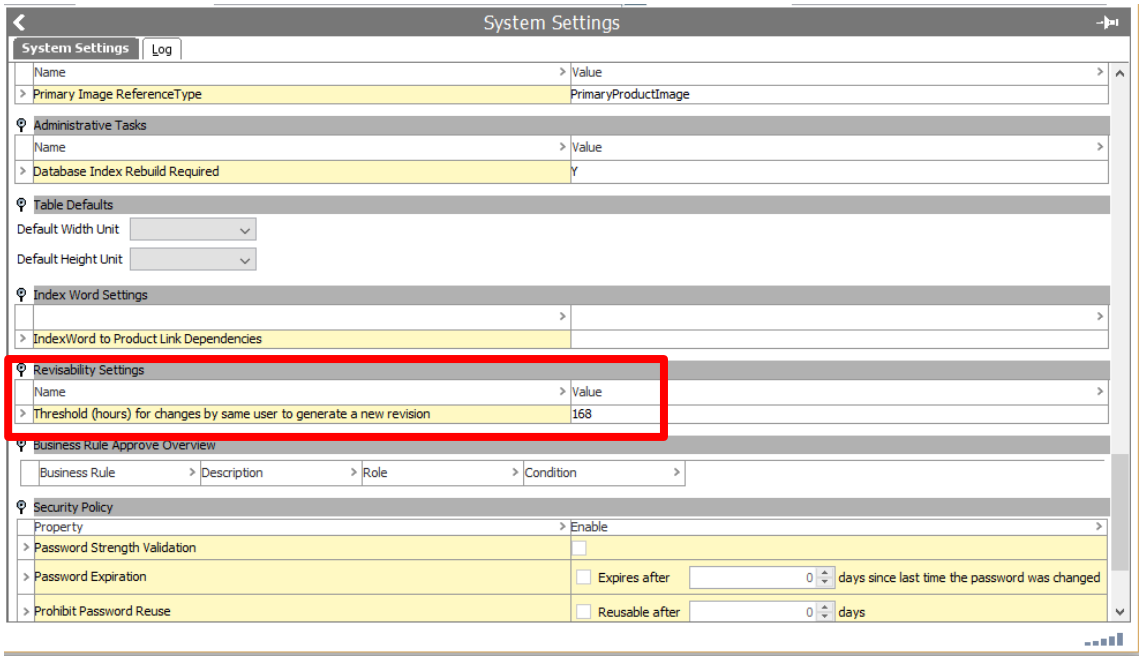
Many revisions may have a negative impact on the performance of the system, and therefore it's key to keep the number of revisions under control and remove unnecessary revisions.

B.10.1 Setting the revision threshold

By default, revisions on objects are created if a user makes a change to the object and the timing threshold (number of hours) is exceeded from 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 not otherwise have revisions made without that user choosing to do so manually. The threshold functionality ensures that changes are recorded, without creating an excessive number of revisions.

The revision threshold can be found via Workbench > System Setup > Users & Groups under Revisability Setting as follows.



The global default number of hours set for all revisable objects is 168, or one week, before a revision is created. If any user acts on the object during the timeframe specified by the threshold parameter, a revision is made when the time is exceeded. If no one acts on the object during that time, no revision is made.

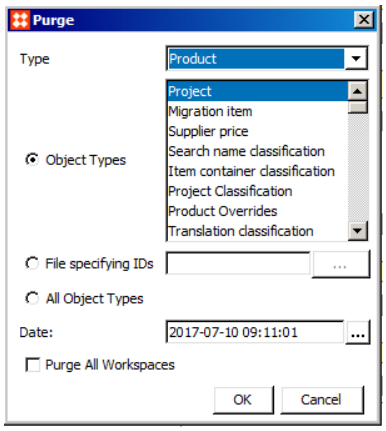
It's recommended to keep the threshold for revisions on the default setting (168 hours) to avoid excessive creation of the revisions, which may have a negative effect on the system performance.

B.10.2 Purge old revisions manually to create a baseline

Over time, the revision history grows, and this affects the time it takes to retrieve the front revision of the object itself including its name, attributes values, and references. This may have a negative effect on the system performance.

Therefore, it's recommend defining a policy for deleting old revisions that aren't used any longer. Usually, undesired revisions are first purged manually to create the baseline, and after that automatically purge of old revisions is configured to keep the number of revisions under control. Also End Points are configured in such a way that new revisions are not created every time the End Point runs.

Old revisions on Products, Classifications, Assets, and even System Configurations can be purged manually via `Workbench > File > Purge old revisions`.



A revision policy can be to

- Create a revision baseline by purging all objects in all workspaces
- Then e.g. on monthly basis purge all objects from a month ago and later

It's recommended to define a revision policy and create a revision baseline by manually purging old revisions.

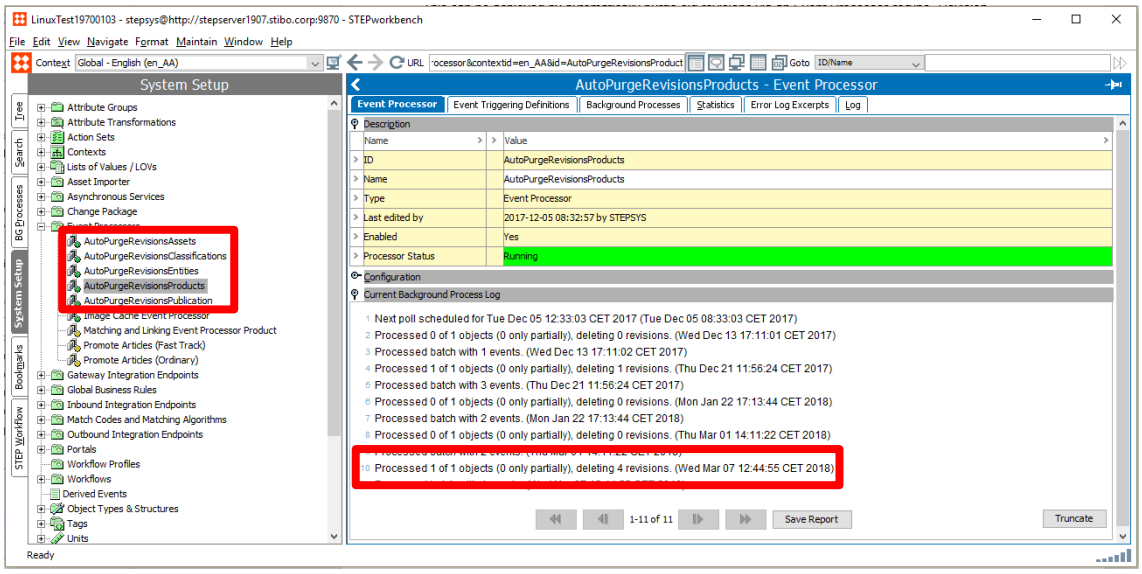
B.10.3 Automatically purging revisions periodically

Once the revision baseline is set by purging undesired revisions manually, an event-processor can be configured to automatically purge revisions periodically.

This can be achieved by automatically purge old revisions via an Event Processor of type "Revision Management". Multiple revision management event processors can be configured, which enables the number of revisions to differ based on the object type. It is recommended that all STEP systems have this event processor set up as a means of managing the STEP database size.

Please be aware that the revision management event processor does not search the STEP database and removes the revisions of all objects when the revision management event processor criteria are met. The revision management event processor is an event processor and only purges the revisions when an event occurs, and only according to the criteria set.

For example: the following screenshot shows multiple auto-purge event processors with the result or deleting 4 revisions.



Defining such an event processor is as follows:

First, the revision management event processor is created

1) Go to `System Setup` and right-click the `Event Processors` setup group.

2) Select `New Event Processor`, and the Event Processor Wizard will be displayed

3) Enter an ID and Name and select Event Processor. Recommended is to define a separate revision management event processor for each object type, thus use something as the following ID and Name “AutoPurgeRevisions”

4) The next step of the wizard allows you to configure the event processor.

   - User running event processor plugin: Here it’s recommended to create a separate user for each event processor. This user needs to have privileges which include the objects being processed. Recommended is to create a user called “EVPROCPURGEREV” or something.

   - Number of events to batch: Specify the batch size. This allows multiple events to be transmitted in a single batch. Setting this number too low can result in slower processing as more background processes will be required. Recommended is to leave this at 1000.

- Days to retain events: Specify the number of days to keep events once processed. Leave this at 0.
- Queue for event processor: Enter the name of a queue to process data from the event processor. The queue is automatically created if it does not already exist. Recommended is to name it “EVPROCPURGEREV” or something.
- 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. Recommended is to set the value to 10.
- Maximum age of old processes in hours: 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. Recommended is to leave the value to 168 hours.
- Limit of lines in execution report: Specify the maximum number of lines to store from the execution report in the log. This setting impacts storage used. Recommended is to leave the value to 1000 hours.
- Select Processor: In this case it’s the Revision Management. This event processor type allows automatic purging of object revisions to limit the total number of revisions retained.

5) The Revision Management processor plugin has the parameters as described below.

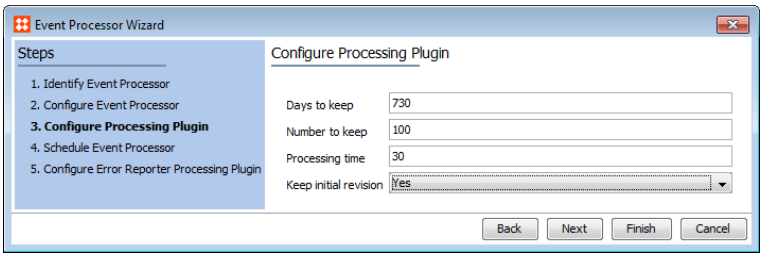
   It’s important to know that a revision must fall **outside all** conditions set in the Days to keep, Number to keep, and Processing time parameters in order to be deleted.

   Many clients only want to keep the initial revision and previous revision. In that case use the recommended values described below.

   - Days to keep: uses calendar days to determine the age of a revision, and keeps only those that are less than the number of days specified. Recommended for most client is to set this to 1.
   - Number to keep: maximum number of revisions remaining after processing. Recommended for most client is to set this to 1.
   - Processing time: determines the maximum number of seconds that the processor will run on each object. Recommended for most client is to leave this to 30.
   - Keep initial revision: when set to Yes, the first revision is kept, even if it falls outside all other conditions. Common setup is to keep the initial revision if you want to know when an object was initially created.

     So again, important to know is that a revision must fall outside all conditions set in the Days to keep, Number to keep, and Processing time parameters in order to be deleted.

For example: the following parameters



Which means:

- The revision being processed is at least 731 days old, AND
- More than 100 revisions will remain after deletion of the revision being processed, AND
- The processor has not run for 30 seconds on the revision being processed, AND
- The revision being processed is NOT the initial revision.

6) The next step is to set the schedule of the event processor. This can be done e.g. once a day.

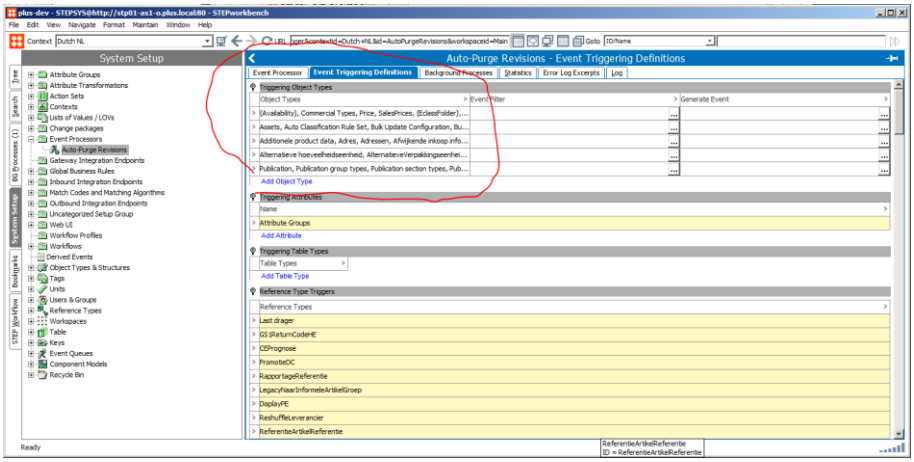
Once the revision management event processor has been created, the event trigger needs to be set in the `Event Trigger Definitions` tab.

- It's important to know that the revision management event processor works on an event of an object. Thus, if an object has been changed and meets the event definitions, then an event is generated. And if an event is generated, then the revision management event processor is executed to remove the revision of that object only when the revision management event processor criteria (as set above) are met.

  In other words, it's not the case that the revision management event processor searches the STEP database and removes the revisions of all objects when the revision management event processor criteria are met. The revision management event processor only removes the revisions according to the criteria set of a single object when an event is triggered of the single object.

  Thus, it's necessary to define which objects are subject for triggering an event and what the triggering criteria are.

- First select the object types which are subject to triggering in the `triggering object type`. Here you select the object types which are subject to trigger an event for. So, if you created a revision management event processor for products only, then you select the product objects types of which the revisions need to be removed.



Now, a common mistake is to think that's it. But STEP doesn't trigger anything if you do not specify the

- Triggering Attributes and / or
- Reference Triggering Types and / or
- Miscellaneous Triggering Types.

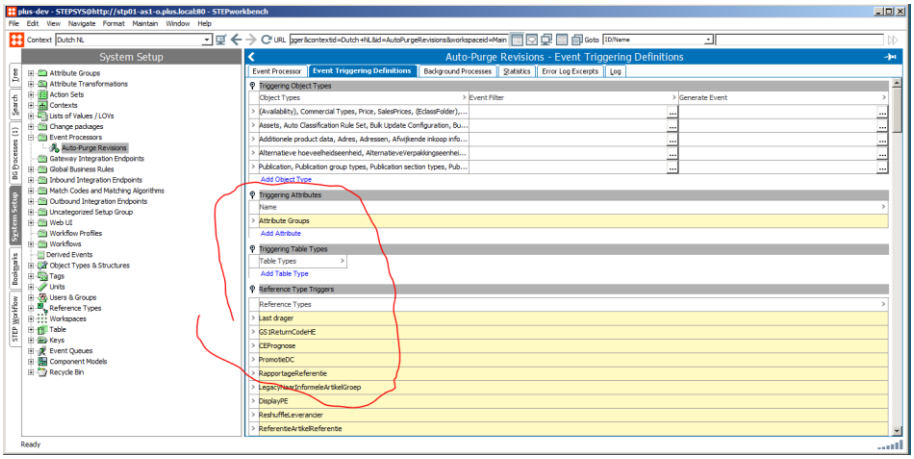
Therefore, you need to specify

- Which attribute value changes cause a triggering of an event (Triggering Attributes),
- Which reference changes cause a triggering of an event (Reference Triggering Types),
- And / or which miscellaneous triggers cause an event

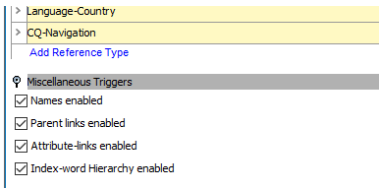
Basically, in almost all cases you want that every attribute value change and every reference change triggers an event for the revision management event processor. This means that you select

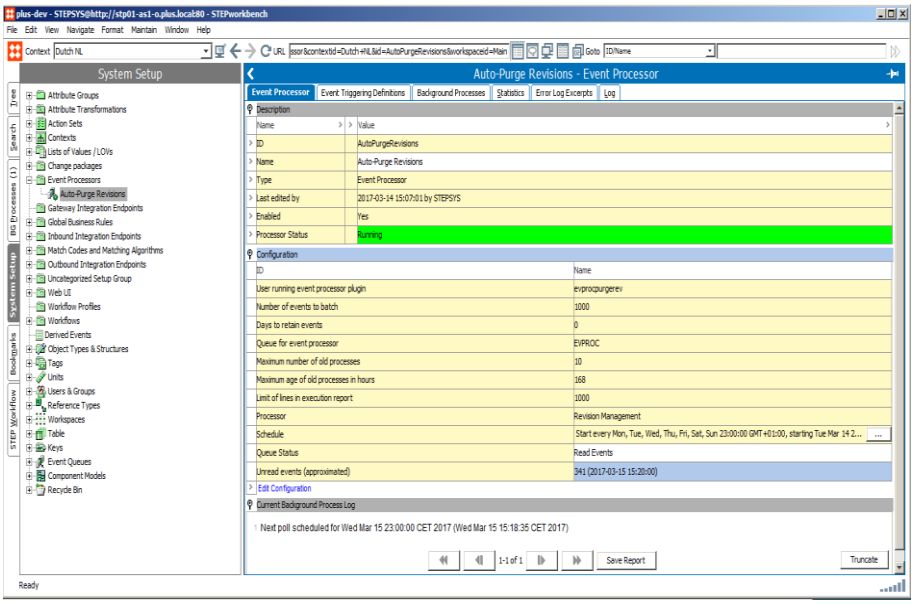
- Triggering Attributes: Select all attributes by selecting the main attribute group
- Reference Triggering Types: Select all relevant reference types. So, if you created a revision management event processor for products only, then you select all the available product reference types there are.

This results in the following.

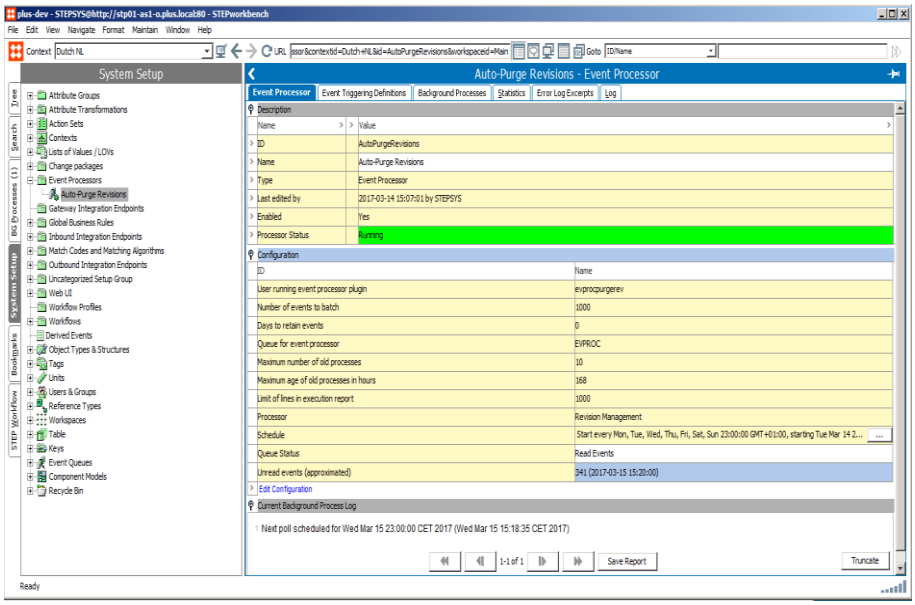


- Finally, you can also set the miscellaneous triggers at the bottom:
    - Names enabled: Triggers an event when a change is made to the STEP Name of triggering object types for a global revisable object, or when a workspace revisable object is approved. An output template must be configured with a Modify event or the endpoint will skip the event since it cannot find a corresponding template.
    - Parent links enabled: Triggers an event when changes are made to the parentage of triggering object types for a global revisable object, or when a workspace revisable object is approved. For example, when a product or asset is moved into another parent folder. An output template must be configured with a Modify event or the endpoint will skip the event since it cannot find a corresponding template.
    - Attribute-links enabled: Triggers an event when attribute hierarchy link is changed on an object of the types for a global revisable object, or when a workspace revisable object is approved. For example, when an attribute is linked to a product or classification.
    - Index-word hierarchy enabled: Triggers an event when an index word changes or is added on a triggering object type. Common setup is to use indexes and this option for print output.





Once the revision management event processor has been created and the triggering definitions are defined, you can test the revision management event processor by enabling the `Read Events` in the `Event Processor` tab. If you enable `Read Events` and then change some objects which comply to the criteria set, then you click on `Click to Estimate` button, and a number of events should pop up.



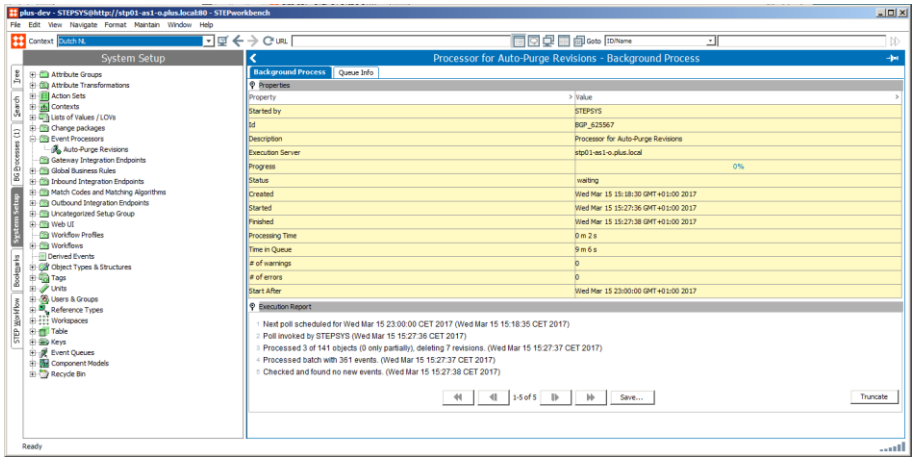
If this test is ok, then the revision management event processor can be activated. This is simply done by

- Making sure the Queue Status is on Read Event (and not on Discard Events)

- And then right-click the revision management event processor and select `Enable Event Processor`.

The result is that the revision management event processor is enabled and will be invoked on the defined schedule according to the defined event triggering definitions and will purge revisions as defined in the revision management event processor.

You can check if all works well after a while by opening the revision management event processor and check if there were events processed.



It's recommended to define an auto-purge event processor per object type to automatically purge revisions.

B.10.4 Revisions on Integration End Points

It might happen that Integration End Points (IIEP and OIEP) have a high number of revisions, in the tenths of thousands or more. Huge amount of revisions will cause the system to slow down. Furthermore, typically the Integration End Points tend to open very slowly.

The root cause of this problem is that the IEP poller is started by a different user than the one configured in the IEP.

This can be noticed by running the Health Check in the Administration Portal and run the check `Pollers started by a different user than the one configured in the IEP`

See C Analysis methodologies and recommendations via STEP Health Check for more information on the Health Check.

When there're Integration End Points (IIEP and OIEP) with tenths of thousands or more revisions, then

- First make sure new IEP revisions are not created anymore every time these IEP run. This can be done by
    - Logging into the STEP Workbench as the same user who is defined under `User`. In this example `STEPSYS`.
    - Then disable and enable the Integration End Point in the STEP Workbench.
    - The result is that the same user is shown under `Revision` and `User`.
- Secondly, revisions on IEPs can be purged from STEP 8.0 onwards by `STEP Workbench > File > Purge old revisions`. Then select `System Configuration` and `Inbound Integration Endpoint Type` and `Outbound Integration Endpoint Type`.

Please be aware that, if many revisions on IEPs occur and are purged, then the index of the database should be re-build. Re-building the indexes can be achieved via the database administrator or via a JIRA ticket.

It's recommended to analyse if the Integration End Points have many revisions, and if so remove these revisions and make sure the revisions are not created automatically anymore.

B.11 Privileges

STEP has a very granular privilege system and the privilege setups.

However, understanding the sophisticated privilege setups is important because complex privilege models may have heavily lead to degraded performance.

Running STEP as a user with a large number of very specific privileges will influence the performance of basically any action in STEP that goes across a large number of nodes, values, or references, including export, import, bulk update, recursive approval, matching, and "multi views" like task list and multi editor.

B.11.1 Privilege configurations

The privileges of STEP are additive only. This means that whenever a basic action is executed, STEP looks for the first privilege that provides the permission. Thus:

- If the user has global permission to everything, the list of privileges to consider is very small and the privilege check is cheap (and can be skipped in certain cases).
- However, if the list of privileges is very specific and granular, the search for the appropriate privilege takes longer
- Finally, if the user tries to do something without having the privilege, the conclusion that the privilege is not there, is the most expensive check of all.

It's recommended not to overdo the privilege setup to avoid excessive privilege checking.

B.11.2 Privileges at imports and exports

When importing or exporting data, all privileges are checked, for each piece of information imported or exported.

Ensure that any imports / exports happen as a user with as broad and few privileges as possible to avoid excessive privilege-checking. Limiting the number of privileges for the importing / exporting user can improve performance dramatically. Only the user configured on the endpoint as the importer / exporter is relevant, this user should generally have relatively few permissions, to ensure performance.

It's recommended to ensure that any imports / exports happen as a user with as broad and few privileges as possible to avoid excessive privilege-checking at imports / exports.

B.11.3 Privileges in the WebUI

A typical WebUI screen fetches all attribute values from the attribute group define in the WebUI screen. The screen then filters out the attributes based on validity of for the product type and based on user privileges.

Configuring attribute groups for the WebUI screen which contain large amounts of attributes will result

- In first listing fetching all attributes from the attribute group
- And then filtering out attributes for display based on validity of for the product type and based on user privileges.

When the WebUI users have complex privilege settings, then the filtering of the data based on these setting will be excessive, which may have a negative impact on the loading time of the screen.

Furthermore, privilege restrictions can be set in the WebUI configuration itself. Be careful with this and avoid excessive privilege checking in the WebUI XML configuration since it may degrade performance. The additional privilege restrictions can be recognized in the WebUI XML files by searching for "restrict="

```
<component id="Actions" restrict="Messaging,Normal user,Read Only,AssetManager,Data Stewards,Integration,Integrations,POC Country Group,PO
  <parameter id="ButtonLabel" value="i18n.stibo.DeleteAction.Label"/>
  <parameter id="DeleteFromWorkflow" value="false"/>
  <parameter id="ApproveDeletion" value="false"/>
  <parameter id="ButtonType" value="ICON_AND_TEXT"/>
  <parameter id="CssClass" value="DeleteButton"/>
</component>
```

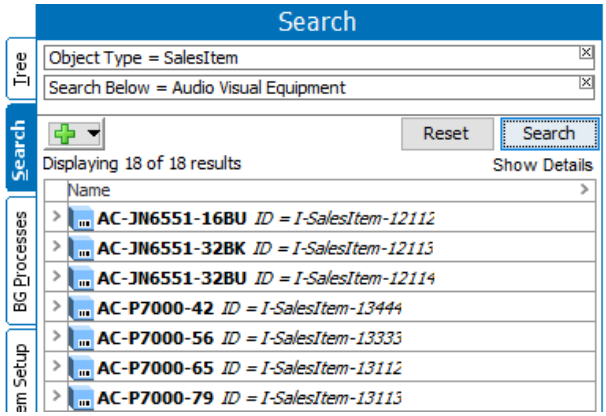
B.12 Searches

STEP searches can lead to a degraded performance when the recommendations are not followed up.

B.12.1 Use specific search criteria

The general recommendation for searching a STEP database with many objects, is to use the search criteria as specific as possible.

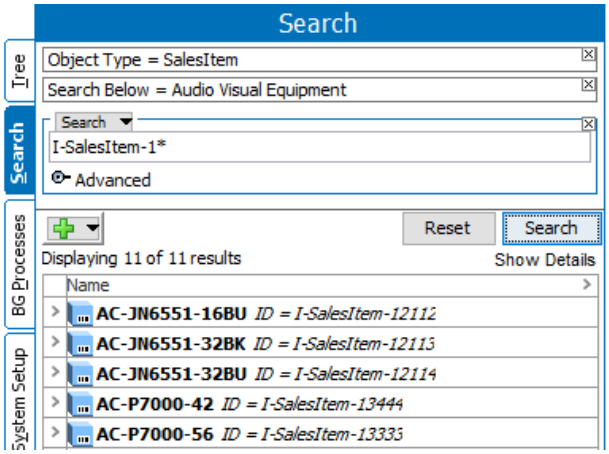
For example: specifying which object type to search for in which hierarchy is faster than searching for all object types in all hierarchies.



B.12.2 Searches with wildcards

Specify as many characters as possible before using wildcards (*). Never specify wildcard as first character because this makes the database unable to use indexing in the query and it will have to traverse all objects in the database.

For example: Use wildcard (*) after more than three characters.

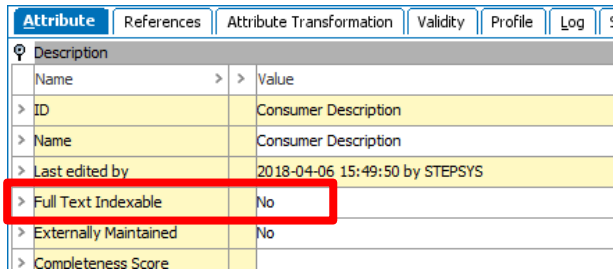


B.12.3 Full Text Indexable

The "Full Text Indexable" setting on attributes allows for searching for individual words in the text value of an attribute. When it’s required to search for words or set of words in attribute values, then "Full Text Indexable" can be considered.

However, applying full text indices on one or more attributes has a system-wide negative impact on performance because it is much more expensive to read, write and query such values.

For example: Use Full Text Indexable = No



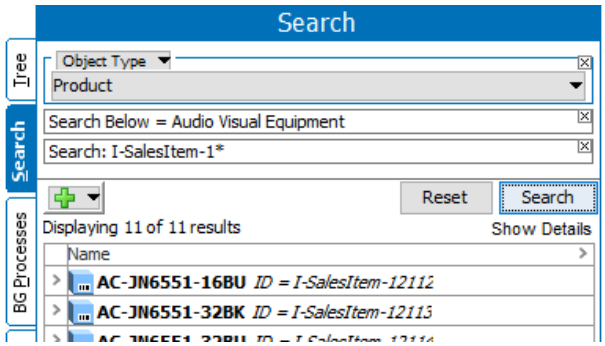
It's recommended to only enable "Full Text Indexable" on an attribute if doing full text searches on values of the attribute is absolutely necessary.

B.12.4 Main object types

Use the Object Type criteria together with other criteria, at least when you specify major object types such as Product, Entity, Asset, Classification, and so on.

However, in environments with where objects exist for this specific object type, the search can perform slight better when you specify the major object types (such as Product, Entity, Asset, Classification, and so on) instead of using the object type below the main object types.

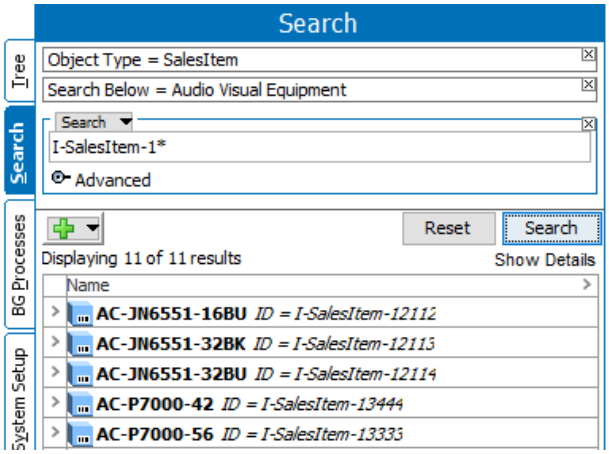
For example: Use main object type Product instead of SalesItem (which is an object type below Product).



B.12.5 Do not use the root hierarchies in Search Below

Do not use a root node high in the hierarchy with many nodes below, it unless really required to achieve the right search result.

For example: Use a granular hierarchy node in Search Below instead of using Primary Product Hierarchy



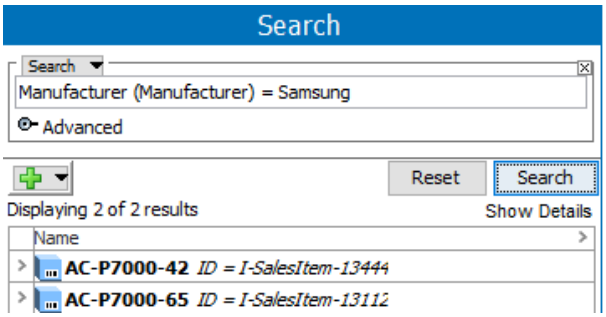
Specifically, in the WebUI, specifying many root nodes is not recommended.

It's recommended to specify the Search Below as tight as possible.

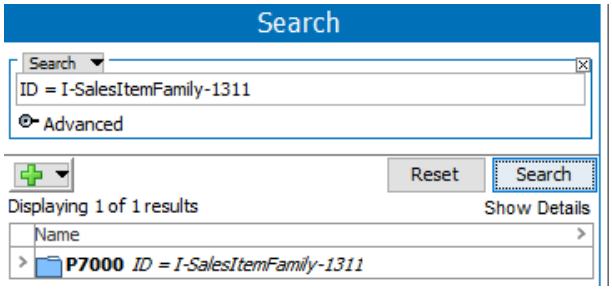
B.12.6 Basic searches

For basic searches in the STEP Workbench and STEP WebUI, use the following search syntax to achieve the best performance:

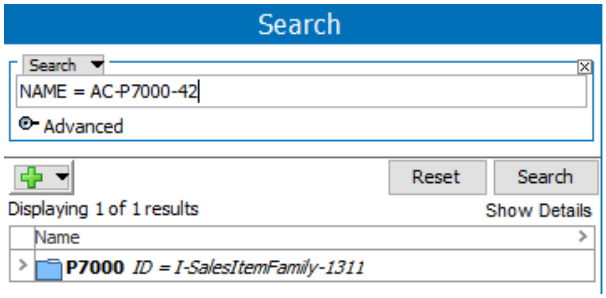
- For finding attributes use `Attribute-ID = value`. Searches in values of this one specified attribute only. For example: Manufacturer (Manufacturer) = Samsung



- For finding objects use the ID or NAME field since these fields are indexed by STEP
    - `ID = value`. Searches for objects by looking in the ID field only. For example: ID = ID = I-SalesItemFamily-1311



- `NAME = value`. Searches for objects by looking in the NAME field only. For example: NAME = AC-P7000-42



When searching just for a value without specifying the Attribute, ID or NAME, then STEP searches for the value in all attribute values, IDs, and names.

It's recommended to specify the Attribute, ID or NAME in searches.

B.12.7 Advanced searches

For advanced searches in the STEP Workbench and STEP WebUI, use the following search syntax to achieve the best performance:

- Avoid enabling "Include Inherited Values" for attribute value search unless it is really needed
- Avoid enabling "Regular Expression" as it disables database indexing in the query and leads to bad search performance.

It's recommended to avoid "Include Inherited Values" and "Regular Expression" searches where possible.

B.13 Matching and linking

The flexibility of matching and linking comes a high level of responsibility. The system load of this functionality may have effect on the performance of STEP.

If there're a lot of supplier items imported, if the matching and linking process is complex, and if the approval of the golden record is part of the process, then this will degrade performance simply because STEP is very busy processing these.

The matching and linking and golden record functionality is designed and based on business requirements. However, there're some pointers to take into account when matching and linking process suffers from bad performance.

B.13.1 Approval of the golden record

In many occasions it's possible to approve the golden record separate from the matching and linking process or not approve the golden record at all.

When approval of the golden record is required, it may be possible to approve the golden record after the matching and linking as part of a bulk update triggering an event processor scheduled to approve the golden records.

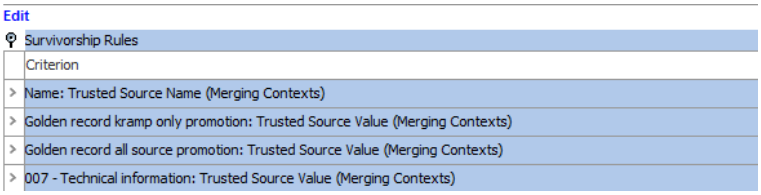
It's recommended to avoid the approval of the golden records as part of the matching and linking process.

B.13.2 Multi-context survivorship rules

In many occasions it's possible to avoid using multi-context survivorship rules for context specific names, context specific attribute values, context specific image references, and context specific web-hierarchy links.

Every time a multi-context survivorship rule is executed, then the matching and linking process doesn't end when the global value is not available, but it continues in all contexts and determined if there's an attribute value in a local context and copy that value to the golden record in that local context.

In the older STEP systems, the multi-context survivorship rules available via a custom extension and are recognized as "Merging Contexts".



In the newer STEP systems, the multi-context survivorship rules standard available and are recognized as "Multi Context".

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 will try to promote the correct value of the supplier record in the right language of golden record.

Furthermore, if the "Allow values from …" tick box is selected per default and will thereby allow sub-dimension point to be updated from less trusted sources. How this works is best illustrated by an example:

- Assume the name is language dependent and the trusted sources are “SLRV” and “CNET”, in that order.
- If the SLRV source (silver record) only has a name for “All Language” and the CNET source has a name for “English”, then the English name, which is a sub-dimension point to “All Languages”, will only be allowed to be promoted to golden record if the tick box is ticked.

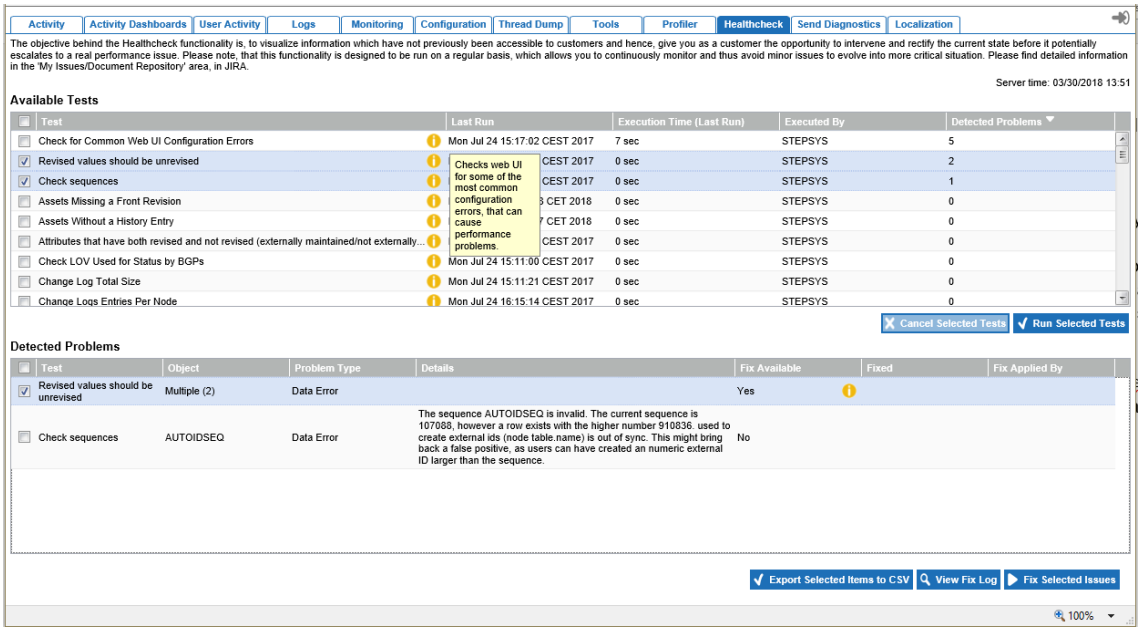
The usage of multi-context survivorship rules obviously demand way more processing from the system than simply promoting the global to the global golden record and stop there, instead of continuing in all contexts and determined if there’s a value in a local context and copy that value to the golden record in that local context.

It’s therefore recommended to avoid using the multi-context survivorship rules as much as possible.

C Analysis methodologies and recommendations via STEP Health Check

There's an option in the Admin Portal of STEP version 8.1 and higher to check the health of the STEP system.

The Health Check is intended to assist users in identification and resolution of system and performance issues. A set of standard tests are available and Stibo Systems reserves the right to add additional tests as a need for them is identified.



The upper portion of the screen allows users to select health checks to be run, while the lower portion provides results for the checks.

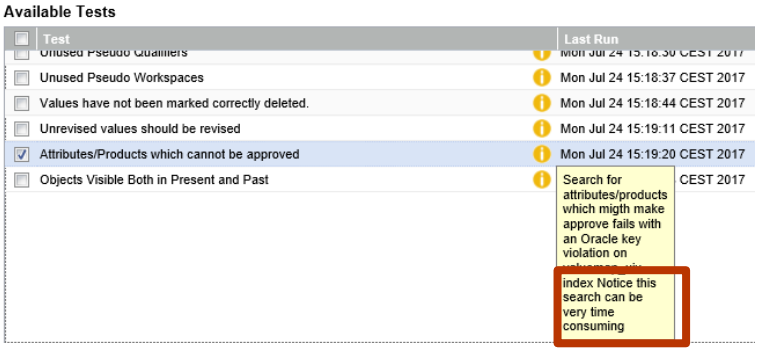
View the `Detected Problems` column after running the health check. A zero (0) value indicates no problems have been found. A non-zero value indicates the number of issues identified from the check.

Users can select the tests to be run, and view a list of issues if any are found. In some cases, fixes are made available. As fixes should only be applied only under the direction of Stibo Systems, applying a fix requires a password that must be obtained from Stibo Systems Support.

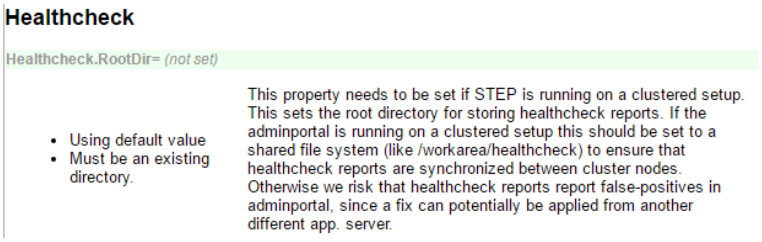
However, some issues identified may be solvable by administrators and the tool serves merely to bring them to light. In addition, it provides a way to aid Stibo Systems in resolving reported issues. If a system is having performance issues, administrators should run the available health checks and send them to Stibo Systems using the Send Diagnostics tool.

C.1 Remarks before running the health checks

Do not run all health checks at once. Some health checks may take a while to run and may degrade performance while running it. Therefore, always read the explanation of the health check before running the health check.



For cluster environments: Use of this functionality on a clustered environment requires that the `Healthcheck.RootDir` configuration property be set to a shared path to ensure that the health check reports are available, regardless of the application server node that the admin portal is accessed from.



C.2 Initial STEP Health Check results

The initial STEP Health Check results are embedded in the following document.

[EMBED THE CLIENT INITIAL STEP HEALTH CHECK RESULTS]

C.3 Final STEP Health Check results

The final STEP Health Check results are embedded in the following document.

[EMBED THE CLIENT FINAL STEP HEALTH CHECK RESULTS]

D Template for main performance problems and non-functional requirements

This appendix provides the template for the client to fill out by describing the main performance problems, non-functional performance requirements and reproduceable performance use cases.

The template is accompanied with explanations and examples.

D.1 Description of the main performance problems

Describe the main performance problems. If available, also include the JIRA ticket number(s).

D.2 Non-functional performance requirements

The non-functional requirements should indicate what is expected by the client from a STEP performance perspective. The non-functional requirements should have been defined and agreed upon in an earlier stage.

The performance related non-functional requirements are described in a table as follows

- *Nbr: Unique number of the non-functional requirement*
- *Non-functional requirement: Short description of the non-functional requirement*
- *Date: The date the non-functional requirement has been defined and agreed upon*
- *Agreed by: the list of people who agreed upon the non-functional requirement*

For example:

Nbr.	***Non-functional requirement***	***Date***	***Agreed by***
NFR1	*STEP WebUI screens should respond in 3 seconds in normal operation*	*1st January 2018*	*John Doe*
NFR2	*Product import via Excel is finished within 3 hours*	*1st January 2018*	*John Doe*

The performance related non-functional requirements are listed in the table below.

Nbr.	Non-functional requirement	Date	Agreed by
NFR1			
NFR2			
NFR3			

D.3 Use cases

The performance use cases describe the most important performance problems in a reproduceable way step-by-step. Each use case should relate to a non-functional requirement.

The use cases are described in a table as follows (for each use case):

1. *Related non-functional requirement*
2. *Short description of the use case*
3. *The environment on which the use case is executed*
4. *The date and time of execution*
5. *For each task executed in the use case:*
    a. *Nbr: Unique use case step identifier which starts with the identifier of the related non-functional requirement first (e.g. NFR1.1 which is related to nonfunctional requirement NFR1).*
    b. *Task executed to reproduce: description if the task executed*
    c. *Expected response time: the expected response time of the task*
    d. *Actual response time: the actual measured response time of the task executed*
    e. *Pass / Fail: did the task executed fail or pass*
    f. *Remarks: optional additional remarks*

For example:

6. *Non-functional requirement: STEP WebUI screens should respond in 3 seconds in normal operation*
7. *Description: The WebUI Homepage loads slowly*
8. *Environment: STEP Production*
9. *Date and time: January 10th, 2018*

Nbr.	***Task executed to reproduce***	***Expected response time***	***Actual response time***	***Pass / Fail***	***Remarks***
NFR1.1	*Start WebUI*				
NFR1.2	*Login to WebUI "Products" with "stepsys" administrator account*				
NFR1.3	*Wait until WebUI Homepage loads*	*3 s*	*30 s*	*Pass*	*WebUI homepage loads slowly*

D.3.1 Use case 1

Non-functional requirement: ….

Description use case: ….

Environment: ….

Date and time of execution: ….

Nbr.	Task executed to reproduce	Expected response time	Actual response time	Pass / Fail	Remarks
US.1.1					
US.1.2					
US.1.3					

D.3.2 Use case 2

Non-functional requirement: ….

Description use case: ….

Environment: ….

Date and time of execution: ….

Nbr.	Task executed to reproduce	Expected response time	Actual response time	Pass / Fail	Remarks
US.2.1					
US.2.2					
US.2.3					

D.3.3 Use case 3

Non-functional requirement: ….

Description use case: ….

Environment: ….

Date and time of execution: ….

Nbr.	Task executed to reproduce	Expected response time	Actual response time	Pass / Fail	Remarks
US.3.1					
US.3.2					
US.3.3					

D.3.4 Use case 4

Non-functional requirement: ….

Description use case: ….

Environment: ….

Date and time of execution: ….

Nbr.	Task executed to reproduce	Expected response time	Actual response time	Pass / Fail	Remarks
US.4.1					
US.4.2					
US.4.3					

D.3.5 Use case 5

Non-functional requirement: ….

Description use case: ….

Environment: ….

Date and time of execution: ….

Nbr.	Task executed to reproduce	Expected response time	Actual response time	Pass / Fail	Remarks
US.5.1					
US.5.2					
US.5.3					

D.4 Performance testing

If performance tests have been executed, then the most important information about these executed performance tests are described in a table.

For example:

Nbr.	***Performance test***	***Date execution***	***Results and issues found***
PT1.1	*STEP WebUI performance test with 10 concurrent users on STEP ACC environment*	*January 2nd, 2018*	*No issues found*
PT1.2	*STEP WebUI performance test with 10 concurrent users on STEP ACC environment*	*January 10th, 2018*	*Significant performance degrades*
PT2.1	*Import of 50.000 products via Excel in STEP using STEP IIEP "Import Products" on STEP ACC environment*	*February 20th, 2018*	*No issues found*

Nbr.	**Performance test**	**Date execution**	**Results and issues found**
PT1			
PT2			
PT3			

Please be aware of the following basic rules when executing performance tests.

Any largescale performance testing must respect a couple of basic and very important rules for making realistic performance testing such as:

1. A good representative and realistic data set must be in place on a STEP system observing all hardware requirements specified by Stibo Systems.
2. Each time a performance test has run the database has to be reset so that every test starts from exactly the same point for origin. For example, just manually deleting what is thought to be the test data is not good enough; there must be a restore from backup holding the agreed test data.
3. Simulating the behavior of multiple users working in parallel will require that the data these users are working with are parameterized. For example, it's usually not very realistic behavior to start 1000 users working on the same product.
4. If user scenarios are recorded to be executed by the test robot, delays between consecutive user actions must be introduced reflecting realistic user behavior. For example, a human being does not make multiple mouse clicks per millisecond.
5. Care must be taken about how to start the test. For example, it's usually not very realistic behavior when starting 1000 users within the same millisecond in bulk.
6. Running strongly parallelized 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. Be sure that expectations are well-defined before executing any tests. What is a good result and what is a bad result. What is the goal of the performance test? Usually 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.
8. Gather relevant monitoring data from the test. As a minimum, it's recommended to have a complete copy of STEPs `diag` folder for every test run. It is also useful to collect Oracle data `statspack/awr` and OS information `sar/nmon`. It is advisable to keep this data at least until a few more tests have been carried out.