STEP System Solution

Architecture Guide

StiboSystems

Version 1.5

Date: June 15th, 2015

Content

Revision Control

Version	Date	Author	Comment
1.0	2013-11-17	Flemming Andersen	First version for STEP Trailblazer
1.1	2014-02-17	Morten Jønby	Corrected minor issues and converted to new template.
1.2	2015-01-12	Morten Jønby	Minor update to header and footer.
1.3	2015-02-27	Jesper Scott	Changed 11.2.0.3+ to 11.2.0.4+
1.4	2015-06-11	Jens Ulrik Østergaard	Changed wording for client types in section 2.1
1.5	2015-06-15	Morten Jønby	Section 10: Updated support matrix.

1 Introduction

This document describes the architecture of the Stibo Enterprise Platform (STEP) and addresses the following questions vital for successful implementation of the Stibo Master Data Management (MDM) / Product Information Management (PIM) solution:

- What are the options for the STEP system to integrate with other systems in the infrastructure?
- What considerations must be made before making final decisions about fundamental topics like the hardware to use, network design, backup/recovery, clustering and security?
- What is required to maintain a STEP installation in a way so that it can run as a fail-safe scalable enterprise application that behaves in a predictable manner even in a worst case scenario?
- What are the recommended procedures for keeping the STEP software itself and any 3rd party software up to date?

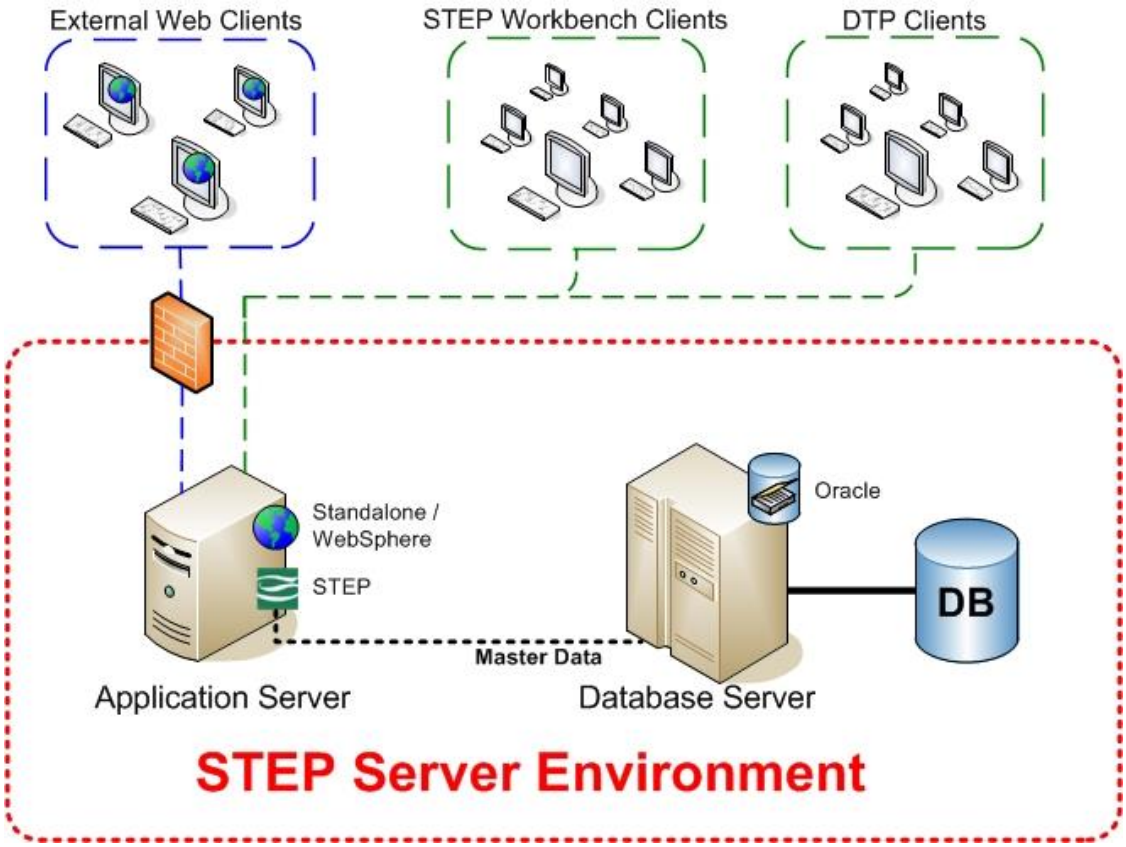
1.1 Intended audience

This document is intended for the use by experienced Information Technology Professionals. An intermediate to advanced understanding of technical architecture is assumed. The document is intended only as a guide and cannot address all situations. Please consult your implementation team or your designated Stibo Systems representative for further information.

2 Architecture overview

2.1 STEP Architecture

The illustration shows a simple STEP system setup with one application server and one database server.



The **Application Server** is a physical server running either standalone (on Oracle Java) or a WebSphere Server. Within the application server an instance of the STEP Enterprise application is running supporting three types of clients. The STEP Workflow component supporting business process management is part of the STEP application.

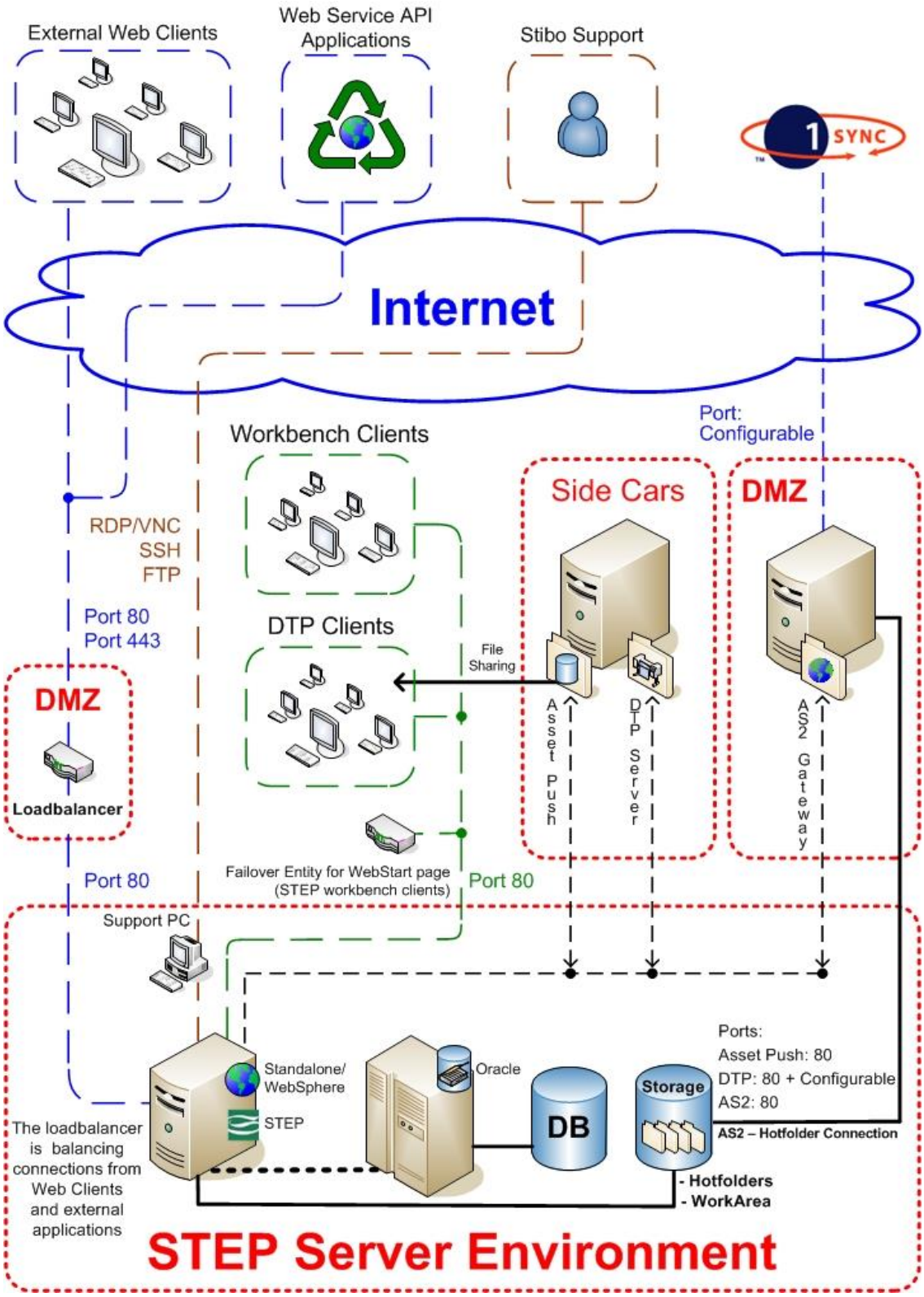
The **Oracle Database Server** provides the primary storage for all information to be stored persistently by the STEP system.

Three different types of clients are represented:

- The **STEP Workbench** is a cross platform Java client providing the full set of functionality of the STEP PIM application.
- The **Web Client** is an internet browser based client providing functionalities used by users/suppliers/vendors.
- The **Desktop Publishing (DTP) Client** represents the InDesign application extended with a STEP DTP plug-in allowing the DTP application to communicate with the STEP PIM via a dedicated Web Service API.

2.2 Enterprise Architecture

The illustration below shows how the STEP system relates to its immediate surroundings:



Further comments to the diagram above:

- External applications are able to communicate with the STEP system using the **Web Service API** or as an alternative the **REST API**. In addition there are a couple of other ways to integrate with the system, all mentioned in the integration section below.
- The STEP system itself is for simplicity only represented as a single box. Please see the diagram below for more details.
- The **Side Cars** denotes a number of applications supporting the STEP Enterprise application functionality wise. These applications are:
  - The **DTP server** (either InDesign or QuarkXPress) provides operations such as proof views and PDF renderings of the DTP documents for users without DTP clients. It is possible to install multiple, physical DTP servers for load balancing and fail over.
  - The **Asset Push Client** listens for creation and updates of assets stored in STEP and automatically extracts a configurable version and subset of these assets to the file system. To allow DTP clients and servers faster access to images, the output images can be synchronized to a file server near the DTP users if they are far away from the STEP system for further performance improvements.
- The **AS2 gateway** handles the AS2 communication with 1Sync. It can either be the Stibo supplied OpenAS2 side car or a customer-supplied and perhaps pre-existing AS2 gateway. The AS2 gateway should be placed in a separate DMZ for tighter control on the network traffic, so only incoming traffic from 1sync is allowed on the configured port.
- **DB** represents the disks necessary for storing the data of the Oracle database being part of STEP.
- The **Failover Entity** ensures that the WebStart page of the STEP Workbench does not become a single point of failure. The STEP Workbench application handles load balancing and failover directly and does not use this device once started, see also "Clustering" section. The entity can be either a dedicated piece of hardware or a computer running an Apache web server with an appropriate proxy configuration.
- The **Web Load Balancer** provides an additional point of entry for external users (through the internet) accessing the STEP system. This point of entry is either a network box or a physical server capable of performing URL filtering together with session affinity ensuring that the same session goes to the same server until the session times out. In effect, this box will perform load balancing and failover tasks for the STEP Web Client.
- For Stibo Systems Technical Support to be able to assist the customer troubleshooting problems, a remote connection from Stibo to the customer hosting the system must be configured. This can be a Citrix-based connection or a direct VPN connection to a Support PC in the customer's network.

3 Integration

Common to the components listed below is that they all contribute to the data exchange between the STEP PIM and different external sources/applications. Together these components create a suite that exposes the STEP system as an open ended and flexible system which easily integrates with an existing infrastructure.

3.1 Integration end-points

3.1.1 In-bound integration

A new concept of an "integration end-point object" is introduced, which represents an interface with one particular system and provides access to all configuration, logging and monitoring in relation to that interface. The concept is designed for both in-bound and out-bound messaging.

Previously, the only type of standard in-bound messaging in STEP was through the use of hotfolders. Other types of in-bound messaging required custom development. In-bound integration end-points now provide multiple standard plugins for retrieving data from a specific system, e.g. by reading from a queue. Once retrieved, messages are converted, mapped and imported using an Import Manager configuration directly associated with the integration end-point. While the integration end-point is active it is possible to track errors and measure throughput and latency times directly on the end-point. The end-point thus connects multiple otherwise disparate import manager background processes.

To help in data governance the integration end-point is a separately managed object with ID, Name and metadata that can tell about e.g. the system that it integrates to and its owner.

In-bound integration end-points are supported by the following user interface features:

- Wizard for creating and maintaining integration end-points
- Statistics overview, showing error counts, error queue sizes, pending message counts, throughput and latency figures
- Background process overview, showing the background processes started for the integration end-point
- Error log excerpts for failed background processes with option to drill down into details

In-bound integration end-points may pull data from:

- JMS queues
- WebSphere MQ
- Oracle AQ
- Hotfolders (existing hotfolder implementations will continue to work)

It is also possible to push data to an integration end-point using the REST API.
To avoid a huge buildup of old background processes, a cleanup procedure can be added so that only a certain number of old processes are kept.
It is possible to monitor integration end-points statistics using the REST API.

3.1.2 Out-bound integration

STEP provides separately governed integration end-points for out-bound integration, supplementing the in-bound integration end-points. It is thus possible to configure integrations with external systems in the user interface, thus reducing the time it takes to configure integration as well as reducing the need for custom development.

Each integration end-point handles interaction with one system, and it is possible to start, stop and resume integration with that system in the STEP user interface. The user interface also provides easy access to statistics about activity on the integration as well as error logs.

Each integration end-point may be assigned different administration privileges, and meta data on the object can be used to describe e.g. the owner of the external system or details on who to contact when changes to the integration are needed.

3.2 STEP Exporter

The Exporter allows the user to export data maintained within the STEP system. Basic support for a number of formats is available, such as CSV, Excel, BMECat and a STEP XML format. A general plug-in structure allows customized conversion plug-ins to be developed that can meet the exact format required by for instance an Enterprise Website. A variety of delivery options like file, FTP, SFTP and email allows the delivery of the exported data to fit the needs of any external application. Exports are run as background process managed via the internal JMS based background processing system of STEP.

3.3 Event Messaging

The event messaging functionality in STEP allows the definition of event queues in STEP that will receive events when selected data is approved or when schema changes are made. These events may then be read by integration code that delivers them to other systems. E.g. Event Messaging is well suited for doing incremental deliveries for high volume Enterprise Websites. A standard JMS adapter is made available for reading the queue from e.g. an Enterprise Service Bus. The basic message format is cross-context STEP XML, but it is possible to configure a STEPXML subset on the event queue object in addition to the events that should trigger them. It is thus possible to define an event queue that only contains events triggered by changes to a particular attribute, and where each message only contains information about attributes. It is also possible to configure that messages should contain information about which attributes have changed. If a receiving system has lost some messages it is possible to rewind the queue and re-play old messages from a given time.

3.4 STEP Importer

The STEP Importer supports import of data into the STEP system. All kinds of data maintained within STEP can be imported ranging from product data to information about STEP users. The STEP XML format can be used for these imports but other formats like CSV, UNSPSC and Excel are also supported. Like exports, imports are run as background processes. Repetitive import cycles can be automated using Hot Folders. When data is stored in a Hot Folder the STEP system will pick up the data automatically and start importing it.

3.5 STEP Web Service API

Custom applications can be built using the STEP Web Services API to read, update and delete data in the STEP PIM. The communication takes place via a standard SOAP protocol providing a maximum of decoupling between the external applications and STEP.

3.6 STEP REST API

REST stands for Representational State Transfer covering simple transfer of data over HTTP by using URLs to identify the data just as you do when you input a URL into your web browser and get back some HTML which is rendered by the web browser. With the STEP REST API you can input a URL into your web browser, which points to data in a STEP system. The STEP system will then send back the data in STEP XML form. You are thus able to view and navigate data in STEP using a web browser. The same can be done by programs that want to integrate with STEP provided they know how to issue an HTTP GET request and parse XML. It is also possible to issue an HTTP PUT request to update and approve STEP data. Finally, there is an option to apply an XSL stylesheet stored in STEP to the data before it is returned. This enables easy creation of web content proofing by storing XSL and CSS stylesheets as assets in STEP.

3.7 STEP SQL API

The STEP SQL API can be used to generate customized reports and extractions. The API consists of a series of views and stored packages. These structures can be accessed via SQL or PL/SQL or other database enabled programming languages such as Visual Basic or Java in order to extract data. In order to secure the read-only property of the API, all of the API components are installed under a separate Oracle user account. This user will be granted select rights, but not update, insert, or delete rights on the tables underlying the views.

3.8 GDSN Integration

Data can be imported from and exported to the data pool 1SYNC, a subsidiary of GS1 US, which is a Global Data Synchronization Network (GDSN)-certified Data Pool that offers a range of data synchronization solutions that eliminates costly data errors, increase supply chain efficiencies, and provide a foundation for the implementation of next-generation technologies, such as the Electronic Product Code™ (EPC) and Radio Frequency Identification (RFID).

Imports are configured as "subscriptions" in STEP with different criteria. These subscriptions are sent to 1SYNC and if products in 1SYNC match a subscription, STEP will receive an import file with the product data included. This synchronization of product data will continue until the data receiver or the data provider wishes to stop it.

Exports of STEP products can also be registered with the 1SYNC GDSN data pool. These products may subsequently be published to a specific trading partner via the GDSN network while tracking the status inside STEP. Updates to published products are tracked so that the products may be re-published.

4 Storing asset content outside the database

4.1 Storing digital assets in the file system

Rather than storing digital assets in the STEP database itself, it is possible to store the assets in a file system on the STEP server. In most cases this reduces the size of the database significantly. When using DTP tools (InDesign Server and/or Clients) this also eliminates the need for an additional copy of the assets as these can be read directly from the filesystem.

The file and directory structure of the digital assets should be considered as a strictly internal part of the STEP system that should only be manipulated via the STEP server application and never be touched manually as that could jeopardize asset look up and asset versions. Therefore an algorithm of mangling and hashing the files has been applied making it impossible to identify the digital assets in the filesystem structure from the file names.

The asset content is stored using a write once algorithm. This means that STEP will never update the content of the file even if new asset content is uploaded to the asset. Instead a new file is created with the new content and the old content file is left untouched.

Note that even though asset storage is set to be the file system, asset content is still fetched from the STEP database if no content is found for an asset in the file system. This makes it possible to migrate the assets from the database to the file system without the user suddenly missing asset content and to still have assets that should be full text searchable (e.g. PDFs) to remain in the STEP database.

4.1.1 DTP support

DTP documents can be built using assets stored in the file system in the same way as for an asset kept in the STEP database (but without using asset push), so the resulting document can be previewed and printed.

4.1.2 Migration

It may be time consuming to migrate existing digital assets from the database to the file system. Downtime while this migration process takes place is not acceptable. Therefore, when asset storage is changed from the database to the file system, a new action is enabled which makes it possible to copy an asset, all assets in a classification (recursive) or all assets in a collection to the file system. In this way the assets may be migrated in a controlled way
without downtime.

4.1.3 Integrity Check

Integrity of the file system for storing digital assets can be double checked check by selecting one or more classifications and then select "Check integrity of external assets" in the File menu. This will start a background process reporting if any assets are missing or has been changed against expectations. Should errors be found the assets with wrong or missing content are gathered in a collection and the URL of this collection is referenced directly from the execution report of the background process.

4.1.4 Backup

The backup of STEP and a full restore of STEP based on the back up will be more complicated with part of the data is stored in STEP and another part is stored in the file system. Please contact Stibo Systems to get advice on how to configure backups.

4.2 External DAM Integration

It is now possible to choose to maintain digital asset contents in an external digital asset management system (DAM) as opposed to maintaining the assets directly in STEP. STEP 6.0 provides a plugin based integration mechanism that can be used to allow the image content of a STEP asset to be maintained outside STEP while maintaining the ability to preview image thumbnails in STEP user interfaces. Please contact Stibo Systems if you are interested in such integration.

5 Business Process Management (BPM)

STEP has numerous handles providing support for modeling and integrating data management processes tailored towards specific business needs.

5.1 Background Processing

Complex and time consuming processes can be run as background processes. Examples of fundamental core features that are implemented as background processes are import, export, translation, approve recursive, and report generation. Background processes can be started from the STEPWorkbench. They are passed onto a JMS based queue system and the jobs in the queue are then processed in FIFO order allowing processes to run in parallel to the extent defined by an adjustable queue configuration. For example, a STEP System can be setup to have two separate queues for imports and exports, respectively, and the "size" of the queues can be configured so that 3 imports and 6 exports max are allowed to run in parallel.

5.2 Business Rules

Many business processes necessitate the ability to pass approved data automatically from STEP to other systems. This poses extra requirements on the validity of data - otherwise data will possibly be rejected by the receiving system. In order to ensure validity STEP provides an opportunity to add business rules to STEP that are applied at the time of approval. Such business rules may perform a check across all data defined on an object and reject or accept approval. In addition STEP has support for creating approve triggers that are also executed as an integral part of the approval process. While business rules are designed for read-only validity checks on data before and after approval, the approve triggers are able to change data as an integral part of the approval process. If any business rule fails then approval will not take place, and the user will be notified why. Customers and partners will be able to write business rules by using a subset of the Java based Core Domain API.

5.3 STEP Workflows

STEP Workflow is a component used for modelling processes where tasks are moved between people.
STEP Workflows are data-centric workflows meaning that a Workflow Instance is defined as the relation between a Workflow definition and an Object in the STEP Database.

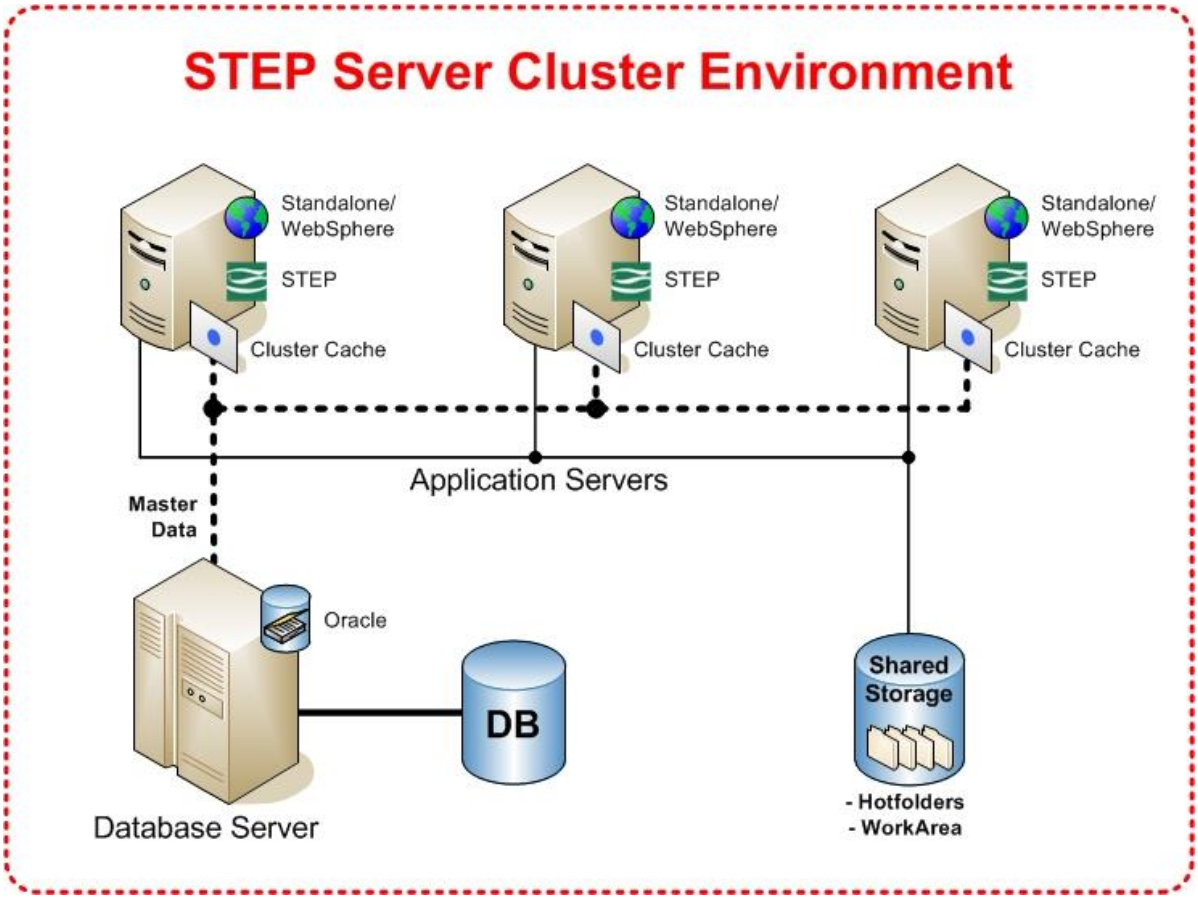
6 Clustering

Clustering creates a STEP system with high availability (HA) and scalability. In a clustered setup the load is distributed so that all members of a cluster take a fair part of the load. A successful cluster setup requires accurate configuration of many parameters: number of servers, bandwidth, network latency, etc. A fail-over strategy must be available for all of the essential components of the system.

In a cluster setup homogenous cluster nodes are recommended. The simplest type of a cluster is created by adding an additional application server that does everything the first application server does. It is recommended that clustered applications servers are set up as clones and kept on identical hardware.

6.1 Cluster architecture

The diagram below shows a STEP system running as a cluster setup with three application servers:



- The **Application Servers** are physical servers running either standalone (using Oracle Java) or a WebSphere Server. Within each of the application servers an instance of the STEP Enterprise application is running supporting all three types of clients mentioned earlier (STEP Workbench clients, Web clients and DTP clients). Each application server computer has a cache of its own that is synchronized whenever write operations take place.

- The **Oracle Database Server** provides the primary storage for all information to be stored persistently by the STEP system. Optionally this can be an Oracle Real Application Cluster setup to compensate for the risk of failure in the underlying hardware.
- **Shared storage** is a file system that is shared by all nodes in the cluster. The **WorkArea** contains e.g. an image cache where for example thumbnails of images are generated. It also stores intermediate files generated by the STEPflow component and files used by background/batch processes. The **hotfolders** are dynamic folders that automatically process files based on a hotfolder configuration. A good example is uploading of assets, where the assets are simply dropped into a folder and automatically imported into STEP. The hotfolders can be exposed directly on the Internet making it possible to place content for import directly via e.g. FTP (not shown).

IMPORTANT

Clustering a STEP system does *not* use clustering technology provided by a commercial application server. This is handled by the STEP application itself.

6.2 Application Server Roles

Any Application Servers in the cluster can be configured to serve as a server for any of the following services:

- Background services
- STEP Workbench client services
- Web client services

It is advisable to make all servers handling a specific type of service handle the same set of services. For instance, it is not recommended to have one server doing background and STEP Workbench client processing and another just STEP Workbench client processing, because the load balancer may not be able to see the differences between the two servers and may put equal amounts of STEP Workbench clients on both servers even though one of them serves background services as well.

6.3 Implementation

Running the STEP Enterprise Application in a cluster means that one instance of the application is running on each application server. Each of the application instances maintains a cache residing in memory on the corresponding application server in order to minimize the number of database requests on repetitive read operations. Whenever a client or a background batch process changes data via one clustered application instance, all of the other application instances must know about it and update their caches accordingly. The mechanism used for this cache synchronization is implemented in the JDO (Java Data Object) layer of the application and relies basically on the same implementation no matter what application server software is being used. In other words, the cluster implementation used by the STEP Enterprise Application does not use any of the application server specific clustering facilities provided by WebSphere.

6.4 Load Balancing

As shown on the architecture diagram in "Architecture Overview" the Web Client is load balanced through a hardware load balancer. Such a load balancer must support session affinity to ensure that the same session goes to the same server until the session times out. This is important, since session state is not replicated amongst servers and if the user is directed to a different server he will be asked to log on again.
In contrast the STEP Workbench client has built in load balancing capabilities building on the CPU load on each server. Once connected to a server, the client will keep using that server unless the client GUI has been idle for 15 minutes. The standard configuration for this will apply for most setups. STEP simply needs to be provided with the list of server names as known by clients in the STEP configuration file.

Notice that the STEP Workbench client has no session state on the server, thus moving from one server to the other is not a problem, except for performance reasons (the caches on the server the user last accessed is most likely to contain the data the user will request next). The STEP system will ensure that any writes to the database is viewed consistently across all application servers in the cluster.

6.5 Scalability

STEP has been designed to support both large initial deployments and growth of smaller systems through horizontal scalability. A cornerstone of the STEP design is to provide a cost effective solution that supports both. No matter if the number of users, the amount of data, or both should grow significantly STEP should not be the system putting a limit on any activities.
Serious and resource demanding tests have been and are continuously being executed to prove that the scalability potential of STEP meets these goals.

7 High Availability

The STEP system can be setup in a way that allows it quickly and automatically to recover from most types of failures, should they occur. Running multiple Application Servers in a cluster is just one of the cornerstones towards getting high availability (HA). Technically the key elements of a HA implementation are:

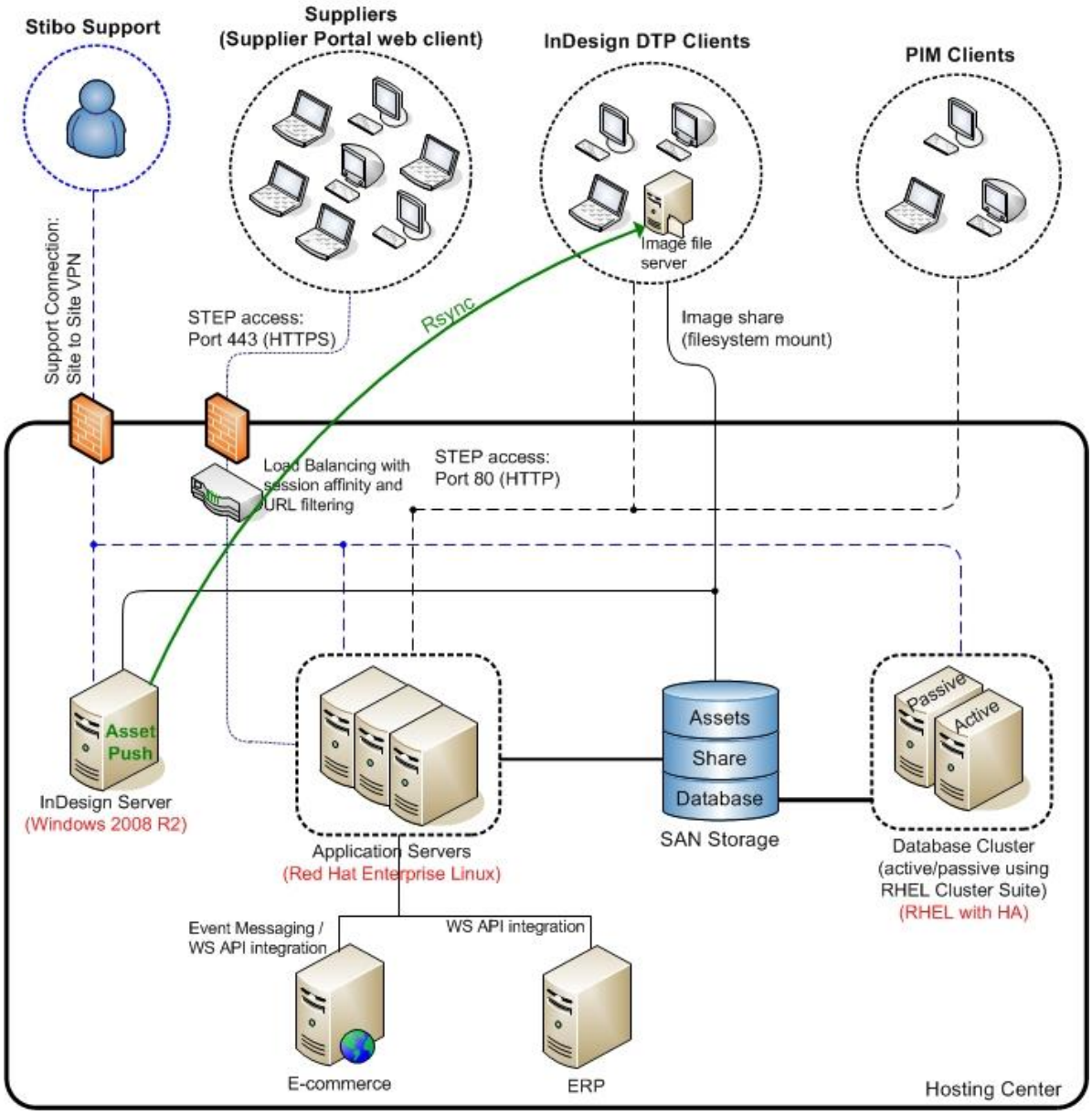
- Eliminating single points of failure
- Applying multi-pathing by creating redundant physical path components, such as adapters, cables, switches and interfaces, to create logical “paths” between the server and the storage device
- Using load balancing to ensure that servers are not overwhelmed to the point of being unable to function properly

Any high availability option may lead to higher complexity which may introduce new risks of instability and change maintenance requirements, so cost/benefit is an issue to consider when considering or planning for high availability.

Another issue to take into consideration is whether to choose a larger number of smaller servers versus a smaller number of big servers (horizontal vs. vertical scaling) and also the business need for a level of performance while one of the cluster nodes is down.
In general servers with more resources (faster CPUs, more memory etc.) are better, but larger hardware is also more expensive. Therefore it is very common to choose several smaller servers. Stibo Systems has performed a scalability test that shows that the STEP system scales almost linearly with the amount of servers (horizontal scaling). This test was performed with a 1, 2 and 4 server setup.

In the next section we will focus on high availability options for each of the main software and hardware components which contribute to the broad range of services provided by STEP. Out of scope here are two other factors impacting high availability:

1. Human failure: somebody can by mistake delete vital parts of the data stored in the system.
2. Planned downtime for system maintenance.



7.1 Application Server

If one of the application servers goes offline, the STEP system will be able to continue operation on one of the other servers without needing a server restart.

Web client users may see an HTTP error or other error if a server is unavailable, depending on the load balancer in use. The user will have to log on again, since the session state is not replicated between servers (for performance reasons). They will subsequently be able to continue working.

STEPWorkbench users may see an internal error message, if a server is unavailable. After ignoring this, they will be able to continue working. No re-logon is necessary as the STEPWorkbench client is stateless in the sense that application servers have no session state on behalf of a STEPWorkbench client.

Current failover handling does not handle the case where a server simply stops answering the client. In this case, the user will need to logon again (after possibly terminating his client). Upon logon, the system will disregard servers not responding in a timely fashion to the logon request, but such servers will of course delay the logon a little until resolved (either by taking the server down or resolving the issue).

If an Application Server running background services goes down, the background services will automatically be rerun when the other servers discovers that the server has failed (this may take a while). So no tasks are lost, but execution of the task may be delayed for a while. Notice that in general, it is hard to ensure that a server failure is noticed by the other servers, since it can simply have stopped processing the single task, but keeps answering to ping requests etc. In that case, there is no way we can be sure to automatically detect the failure.

7.2 Oracle Database Server

Oracle provides some options to support clustering and/or failover of the database:

1. Oracle Real Application Cluster (RAC)
2. Oracle Data Guard

Oracle RAC provides fault tolerance, load balancing and scalability. In an Oracle RAC environment, two or more computers (each with one instance) concurrently access a single database. This allows the application to connect to either computer and have access to a single coordinated database. In case one of the nodes in the cluster fails the database is still available through one or more other nodes.
Oracle RAC is supported on both Oracle Standard and Enterprise Edition. With Standard Edition there are some limitations though:

- The maximum number of CPUs defined by the license is for the entire cluster; it is not a per node maximum.
- Automatic Storage Management (ASM) based on RAW or block devices must be used to manage all database-related files. Third-party volume managers and file systems are not supported for this purpose.

The Enterprise Edition does not have these limitations and Stibo Systems does not provide support for ASM on raw devices. This will be the responsibility of the customer. Therefore Stibo Systems recommends using Oracle RAC only on Oracle Enterprise Edition.

The Oracle Data Guard solution provides high availability, data protection and disaster recovery. Data Guard uses a standby database which is a copy of the production database. The standby database is kept up to date by applying redo log data from the production database. If the production database fails Data Guard will switch to the standby database so this now becomes the production database.
Data Guard is only available with the Oracle Enterprise Edition.

7.2.1 Oracle and Third party software alternatives

The Oracle Enterprise Edition and the extra cost options mentioned above are very expensive and other solutions do exist. Using Oracle RAC provides close to 100% uptime, but if minimal downtime is tolerated by the business, these solutions are worth considering as they do not require the Enterprise Edition and are thus much cheaper to implement.

Possible solutions for implementing high availability are:

- Red Hat Enterprise Linux 6 with High Availability Add-On
- Windows Server 2012 R2 with Microsoft Cluster Server and Oracle Failsafe

Both of these solutions use an active/passive setup as illustrated in the figure above. If the active database node fails the clustering functionality will failover the database storage to the passive node and startup the database instance. The passive node is now the active node.

An alternative to Oracle Data Guard for providing disaster recovery using a standby database is “DBvisit standby” (www.dbvisit.com). Describing this product is out of scope of this document.

7.3 DTP Server Failover

The STEP application load balances across the DTP servers that are online. A DTP server crash will affect only those users who are in the middle of executing an operation against that particular DTP server. Such users will subsequently be able to redo the failing operation. The STEP application will dispatch to another DTP server, still online, and the operation will be able to complete.

7.4 Shared Storage

Disk crashes can be handled in a RAID setup allowing a crashed disk to be hot swapped with a new one without leading to any system breakdown. To get an even better protection against unexpected accidents an IP-Storage Area Network (IP-SAN) can establish two instances of the same storage at two different locations, thereby achieving box-to-box redundancy. One storage instance will be the active one and the other one passive. In the event of irrecoverable failure of the active system the passive instance will be become the active one. A variety of advanced techniques are available for keeping the active and passive storage in sync.

8 Network Consideration

If the barriers set up for security reasons are not aligned with the requirements of the STEP system it might lead to a non-functional system or a very poorly performing system. STEP is not particularly demanding but there are a few rules that need to be observed in order for the system to work properly.

8.1 Network boundaries and firewalls

There are several distinct network realms involved in the STEP system. It is assumed that firewalls only exist at the boundaries between the networks, so network traffic is only defined at those boundaries.

As the architecture diagram hints most application traffic happens as plain HTTP on port 80 from STEPWorkbench, web browsers and other clients, directly to the application servers. There are other requirements for network connectivity that depend on the particular configuration of the system. Examples of such connection requirements are:

- Incoming SSH or RDP (depending on system platform) to allow administrators to manage the systems in the cluster.
- Incoming FTP for files upload if hotfolders are used.
- Outgoing HTTPS to the Stibo update server, so new software can be downloaded.
- Outgoing HTTP on a configurable port to the DTP sidecar.
- File exchange with the AS2 gateway if 1SYNC integration is used.

It is important to note that the exact configuration of the network boundaries can only be determined once the specific features needed on a particular system are known.
It is recommended to work with your Stibo Systems contact for the latest port requirements.

8.2 Intra-cluster network: One switch, one VLAN

The application servers access the database using JDBC connections. The communication is a combination of small requests and larger batches that might take hours to process. The process of sending a lot of small requests will be very sensitive to latency in the network, while timeouts may kill the large batch jobs. Therefore any latency, TCP timeouts or incorrect configuration in the intra-cluster network could severely impact performance, stability and functionality of the system, and in addition might be difficult to detect and debug. In general it is important to keep the network, which binds the cluster together, as simple as possible. The Application Servers and the Database Servers must all be located on the same switch with no filtering or routing between them.
In a standard implementation it is therefore assumed that no firewalls exist between the application server(s) and the database server.

8.3 Redundancy in the network infrastructure

The principle that simple components are less prone to failure than complex ones is important when implementing network infrastructure. E.g. the network switch is not expected to fail very often and if it does it can be swapped out. If the network infrastructure needs to be made redundant the standard

tools, such as STP or OSPF with multiple NICs in each server is needed. A detailed description of such a setup is outside the scope of this document.

9 Security

In this section we briefly cover the main aspects of security in relation to STEP.

9.1 Authentication

Authentication of users in STEP can be configured in the following ways:

- Default user database maintained internally in the STEP database.
- LDAP
- LDAP with Kerberos

The LDAP authentication methods allow the STEP system to be integrated with any existing user authentication infrastructure. The advantages of these options are that users do not need to maintain an extra password for STEP and that the password policies of the central authentication mechanism are inherited to STEP.

9.2 Data Protection

The access to data in STEP is under fine-grained control by the application servers and is configured via the STEPWorkbench Client. User actions are set up and attached to user groups and the specific users then get permissions due to membership of one or more of these groups. Granting of permissions is positive in the sense that the user actions specify what the users are allowed to do, not what they are not allowed to do. If a user is a member of two different user groups with overlapping sets of privileges the resulting set of privileges will be the union of the two groups.

9.3 Network Security Considerations

The STEP cluster consists of the Oracle database server and the Application Servers which must be on the same physical network with no firewalls between them. See also Network Considerations section.

9.4 Local OS Security

As with any secure system only trusted users should be granted access to any of the servers in the cluster (Oracle database and the Application servers) in order to protect against local privilege escalation attacks, which are numerous and hard to guard against, but also to limit the possibility for interference performance wise.

10 Platform Support Matrix

Below is a list of the platforms supported on the server- and clientside.

10.1 Server Platform Support

Server Role	Platform	OS
Application	x86-64	Red Hat Enterprise Linux 6.5 64-Bit Oracle Enterprise Linux 6.5 64-Bit *(UEK3 or RHEL kernel)* MS Windows Server 2012 R2 / 2008 R2 64-Bit
Database	x86-64	Red Hat Enterprise Linux 6.5 64-Bit Oracle Enterprise Linux 6.5 64-Bit *(UEK3 or RHEL kernel)* MS Windows Server 2012 R2 / 2008 R2 64-Bit
DTP	x86-64	Mac OS X 10.8.x / 10.9.x Server 64-Bit MS Windows Server 2012 / 2008 R2 64-Bit
Asset Push	x86-64	Mac OS X 10.8.x / 10.9.x Server 64-Bit Red Hat Enterprise Linux 6.5 64-Bit Oracle Enterprise Linux 6.5 64-Bit *(UEK3 or RHEL kernel)* MS Windows Server 2012 R2 / 2008 R2 64-Bit

10.1.1 Third party software

Server Role	Software
Application	Oracle Java 7 SE (Java JDK 1.7.0_67+ 64-bit)
Database	Oracle Database 12.1.0.1 64-bit (SE/One, SE, EE) Oracle Database 11.2.0.4 64-bit (SE/One, SE, EE) *Enterprise Edition may be needed if advanced Oracle options are requested – this must be checked with Oracle*
DTP	Adobe InDesign CC Server (v10) 64-bit / CS6 64-bit
Asset Push	Oracle Java 7 SE (Java JDK 1.7.0_67+ 64-bit)
Proxy Server	Apache 2.2 (using mod_proxy) *This proxy is installed on the application server*

10.2 Client Support – Windows

Windows Client	
Processor	Intel based 2.0GHz Core i3 Ivy Bridge or newer
Number of Processor Units	1
Total System Memory	4 GB or more
Operating System	Windows 7 / 8 / 8.1
Storage	Internal 80 GB HDD or larger 100 MB of free space for STEP Client installation Additional free storage required for usage of DTP applications
Software	Java Runtime 1.7.0_xx and 1.8.0_xx (latest update) STEP Client Software Adobe® InDesign® CC / CS6 Browsers: IE 9/10/11, Firefox, Safari, Chrome

10.3 Client Support – Mac

Mac Client	
Processor Technology	Intel based 2.0GHz Core i3 Ivy Bridge or faster
Number of Processor Units	1
Total System Memory	4 GB or more
Operating System	Mac OS X 10.8.x / 10.9.x / 10.10.x
Storage	Internal 80 GB HDD or larger 100 MB of free space for STEP Client installation Additional free storage required for usage of DTP applications
Software	Java Runtime 1.7.0_xx and 1.8.0_xx (latest update) STEP Client Software Adobe® InDesign® CC / CS6 Browsers: Firefox, Safari, Chrome

About Stibo Systems

Stibo Systems provides global organisations with a leading multi-domain Master Data Management (MDM) solution. Stibo Systems enables its customers to better manage enterprise intelligence on a global scale, improve sales, and quickly adjust to changes in business requirements. Stibo Systems' STEP technology is a flexible MDM solution that provides a single trusted source of operational information for the entire enterprise. Stibo Systems offers industry-specific solutions, engineered and supported to meet the strategic information needs of global customers including: GE, Sears, Siemens, Target and Thule. Stibo Systems is a subsidiary of the privately held Stibo A/S group, originally founded in 1794 with corporate headquarters in Aarhus, Denmark.

For more information, please visit www.stibosystems.com.