EASYViSTA

White Paper Self Help - On Premise

TABLE OF CONTENTS

Contents

Download the PDF file

PRESENTATION

The purpose of this white paper is to help you understand how Self Help can be integrated into your technical environment.

Because individual constraints and technology choices make each client's infrastructure unique, every project will undergo a specific analysis during the pre-sales and/or installation phases.

PRODUCT GLOSSARY

Self Help is based on three main functions:

- A presentation function for procedures and the virtual agent.
- An edit function for procedures and the virtual agent.
- A Self Help Studio fat client installed on the procedure author workstations.

Depending on your project, the edition and presentation functions can be run on the same machine or split between machines. The target architecture and integration into your infrastructure must take this allocation into account.

GLOBAL ARCHITECTURE

Service components

Depending on the architectures and the workload, the edition and presentation servers can be run on the same machine.

The servers have identical installations and can support all Self Help functions.

Adaptability to requirements

Progressive scalability

The Self Help architecture is scalable. It can be reviewed and modified based on changes in your requirements. You can start your project with a basic architectural model and review it subsequently if the number of concurrent users increases, if your security rules change or if functionalities are added to the initial project scope.

Each tier can be scaled separately using more or less resources based on the requirements identified. Scalability primarily affects the Self Help presentation servers.

The diagram below shows a few examples of the possible presentation server architectures.

We recommend that a presentation server is not placed in a DMZ, instead use a REVERSE PROXY to publish the application.
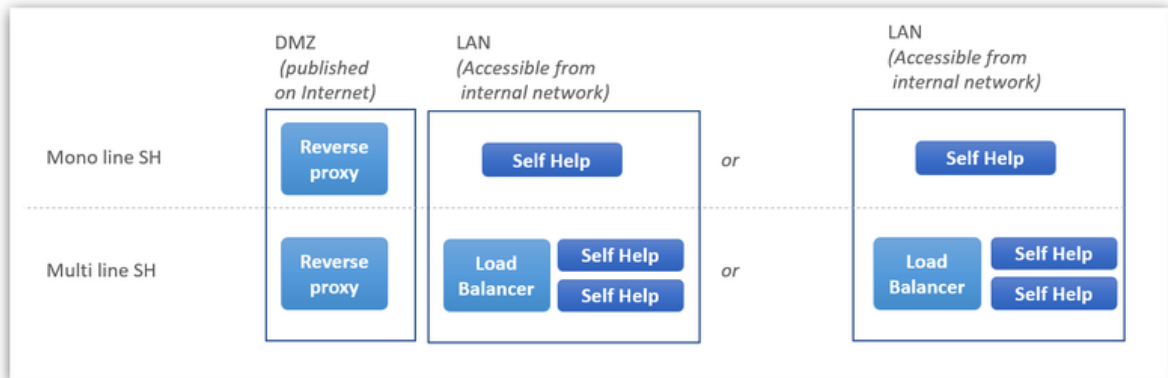
Scalability

Our services can go from simple platforms with two servers, up to configurations comprising several dozen lines, potentially split into different security zones.

The first design criterion relates to scalability, the aim being to use the two dimensions available to you:

- Scale IN: Addition of CPU or memory resources to existing machines so they can handle a greater workload. While this approach is cost-effective (fewer VMs to manage, fewer licenses, etc.), it quickly reaches its limits because certain internal system resources are not expandable (thread, etc.).
- Scale OUT: Addition of new machines to take some of the workload. This approach is more flexible, but it means the architecture is more complex due to adding a load balancer to distribute users across servers.

Whatever the architecture and without other constraints than scalability, it is always interesting to use the SCALE IN as much as possible without adding new machines.

Resilience

What is the platform's desired availability level? A high level (24/7, for example) will involve a more complex architecture that includes additional lines based on the desired scalability conditions.

These additional platforms will take on the workload in the event of failure of one of the base lines defined as necessary for handling the target workload.

The technical solution will depend on the tolerance that you desire and, therefore, the degradation in service or the complete temporary loss of this. Here are some examples:

- Temporary loss of aSelf Help line: Add an additional line
- Temporary loss of a database server: The database server must be clustered for greater availability. But if a loss of several hours is tolerable, VM re-mount and database restore can be sufficient.
- Temporary loss of the load balancer: Load balancer clustering.

We recommend that you permanently integrate these additional resources into the production chain rather than keep them offline and ready to be started. Although the cost of a running machine is greater than the cost of an offline machine, this will ensure that these machines are correctly configured / up to date when you need them.

Maintenability

If your security policy requires that you regularly update operating systems, we recommend integrating an additional line to the line which has already been sized for scalability.

This additional line, which is not included in the workload demand, will allow you to update your operating systems without impacting production.

Here's an example with three lines, A, B et C (C having been added while only A and B are required for the workload):

- A and B in production; C taken out of production + updated + returned to production.
- A and C in production; B taken out of production + updated + returned to production.
- B and C in production; A taken out of production + updated + returned to production.

Two lines are available at each step, ensuring that users are not penalized.

Load balancer

Our services can be placed behind a load balancer to evenly distribute users wishing to connect to the different lines available.

Your load balancer must allow "session persistence", in other words, a user that has already been authenticated by one of the lines must be directed to that line for as long as they are connected. If not, the user's previous authentication will not be found, and the service will ask them to log in again.

Reverse proxy

Our services can be placed behind a reverse proxy. The reverse proxy can, among other things, change the type of request (incoming https to http on the web front end), but it must not:

- Change input parameters;
- Change the URL itself (by adding folders, changing the domain, etc.).

If the reverse proxy includes content control functionality (WAF, antivirus, etc.), it must not change the content passing through it (parameters, content, etc.) apart from its own HEADERS parameters.

Security of your data in transit

To safeguard the confidentiality and integrity of data flows, we strongly recommend protecting the Self Help front-end servers with certificates.

We also advise the following:

- Use certificates provided by a trusted third party. Private certificates will work but will generate lots of errors when accessed from mobile devices outside your network (mobile phones, etc.).
- Configure your SSL so that it does not accept protocols, ciphers, etc. that are known to be vulnerable: SSL v2, SSL v3, TLS v1.0, RC4, 3DES, etc.

Here is an example to base the SSL layer on:

```
SSLProtocol all -SSLv2 -SSLv3 -TLSv1 -TLSv1.1
SSLCompression off
SSLCipherSuite ECDHE-RSA-AES128-GCM-SHA256:ECDHE-ECDSA-AES128-
GCM-SHA256:ECDHE-RSA-AES256-GCM-SHA384:ECDHE-ECDSA-AES256-
GCM-SHA384:DHE-RSA-AES128-GCM-SHA256:DHE-DSS-AES128-GCMSHA256:kEDH+AESGCM:ECDHE-
RSA-AES128-SHA256:ECDHE-ECDSA-AES128-
SHA256:ECDHE-RSA-AES128-SHA:ECDHE-ECDSA-AES128-SHA:ECDHERSA-AES256-SHA384:ECDHE-
ECDSA-AES256-SHA384:ECDHE-RSA-AES256-
SHA:ECDHE-ECDSA-AES256-SHA:DHE-RSA-AES128-SHA256:DHE-RSA-AES128-
SHA:DHE-DSS-AES128-SHA256:DHE-RSA-AES256-SHA256:DHE-DSS-AES256-
SHA:DHE-RSA-AES256-SHA:!aNULL:!eNULL:!EXPORT:!DES:!RC4:!3DES:!MD5:!PSK
```

Need for other environments

As well as the production environment, you can create additional environments to meet your organization's needs.

We recommend creating and maintaining at least one additional environment so that you can test changes before applying them to your production environment, in particular:

- Fixes or major developments on our products;
- Changes in server component configuration (Apache, Tomcat, SSL etc.);
- Version updates or fixes for operating systems or server components.

Browsers

Suppliers

For standard themes delivered with the software, the following browsers are supported:

- Google Chrome 70 minimum
- Firefox version 63.0 minimum
- Edge 42 minimum
- Internet Explorer 11 minimum

Configuration

The limit of the local cache and temp files must be adequate (> 10 MB).

If you are using the SSL protocol, you should check that the cache is authorized for the secure page.

Others

Our services do not require APPLETs or ActiveX on the client browser.

Cookies

Our services use cookies to improve website functionalities and user experience. These cookies do not contain personal or sensitive data.

Your browser must authorize our services to create cookies.
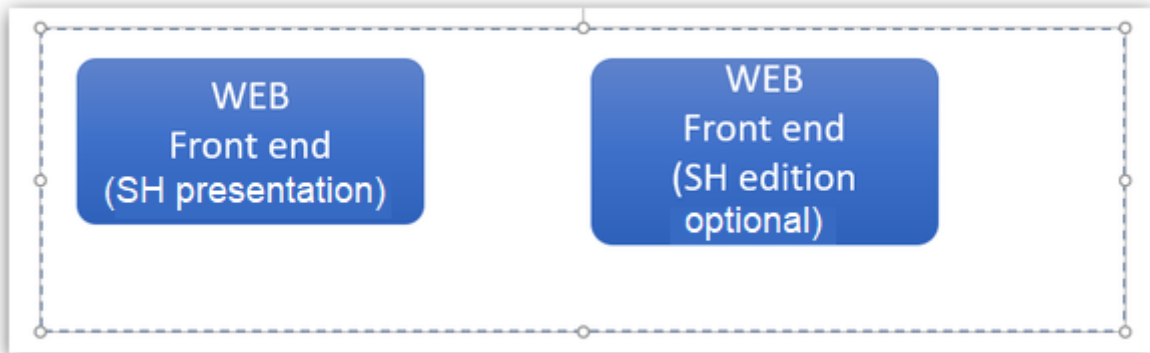
SOME ARCHITECTURE EXAMPLES

Monoline architecture over LAN

A single line reduces hosting costs but does not guarantee maximum availability.

The edition and presentation servers can be grouped together on the same machine by allocating available resources (CPU, RAM, hard disk) accordingly.

Multi-line architecture over LAN

Multiple line architecture for presentation servers offers scalability (number of base lines) and high availability (an extra line for an equivalent service, even if one line is lost).
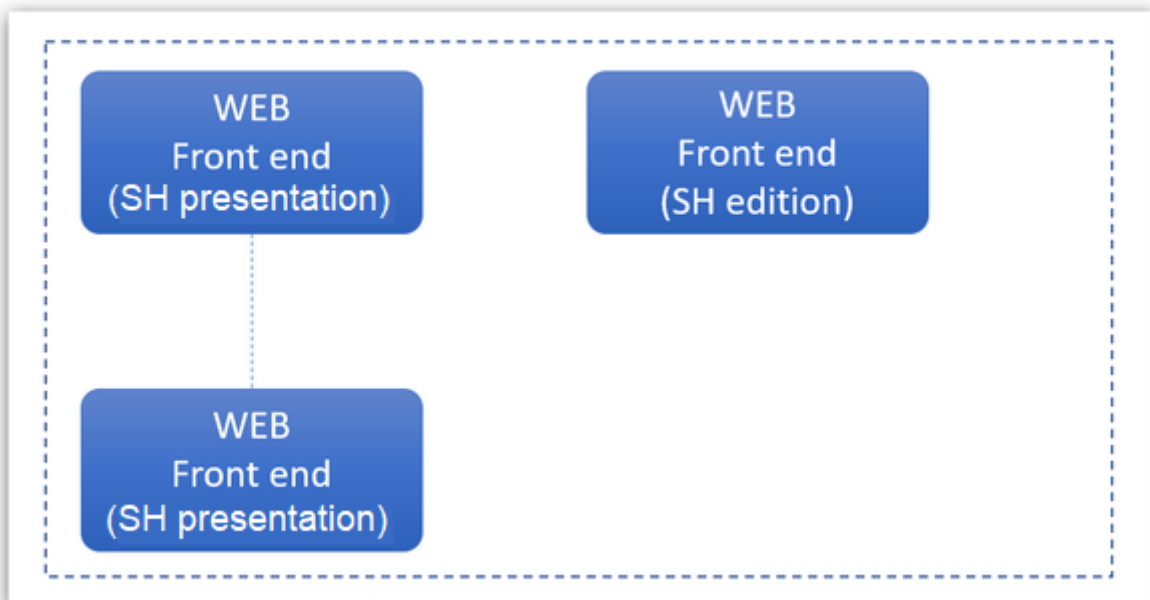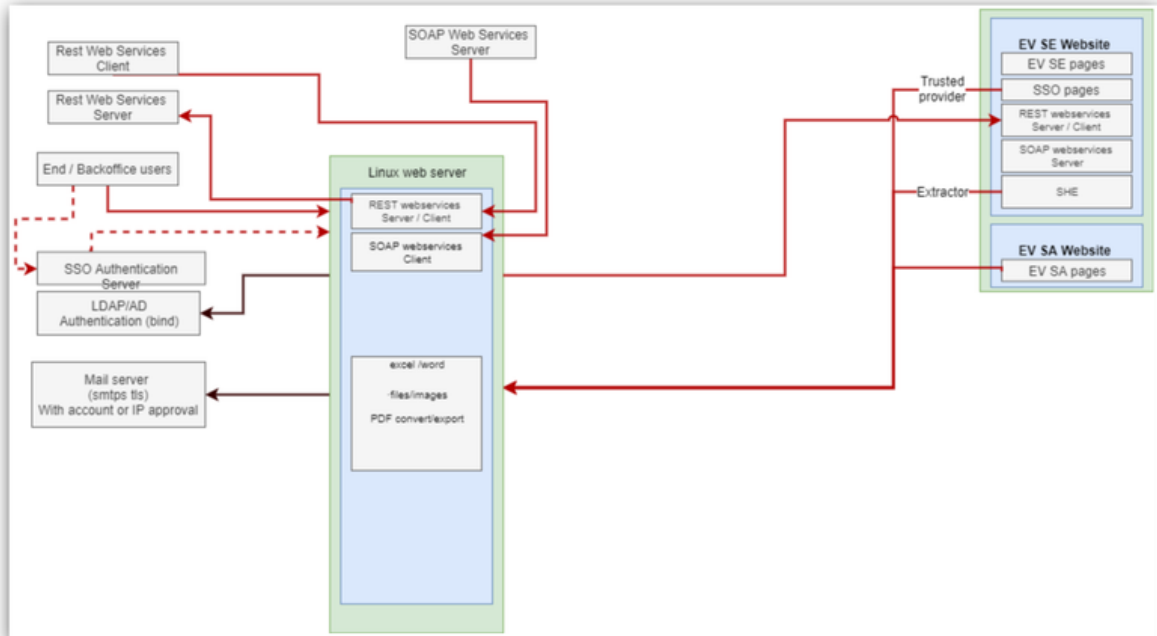

Application publishing

Self Help is published online using a proxy.

TECHNICAL INTEROPERABILITY

Schematic diagram

The following diagram presents the interoperability possibility of Self Help.

REST

REST supplier

Our services are available via RESR and SOAP 1.2.

REST client and SOAP 1.2

The services can use external REST or SOAP 1.2 services.

Email

Self Help must access your e-mail server to send e-mails to your users when handling incidents / changes.

The following protocols are supported: SMTP / SMTPS / SMTPS with TLS.

Interoperability with EasyVista Service Manager

Self Help communicates with Service Manager over https/http for the following functions:

- At the end of Self Help procedures, creation of tickets via REST or SOAP to Service Manager;
- Self Help Extractor: from Service Manager to Self Help over https to consolidate procedures in the knowledge base;
- From Service Apps, for presentation of procedures Self Help over https.

Interoperability with all other tools

Self Help communicates with your tool over https/http for the following functions:

- At the end of procedures Self Help, creation of tickets via REST or SOAP to your tool.

SERVER DESIGN

Caution: These figures are provided as a guide only because the resources required will vary depending not just on the number of users, but the number of incidents/requests created daily, web service activity and the type of use (Front Office, Back Office). *The figures provided are based on our SaaS experience.*

Remember, your target architecture will comprise one or more of the servers described below.

Server configuration

Single-server architecture

This architecture supports all Self Help functions, namely:

- Edition functions;
- Procedure presentation functions;
- The virtual agent.

Description of configuration:

Elements	Nominal configuration	Maximum configuration
Workload	• 150 users executing procedures • 15 users editing procedures	• Maximum of 300 users executing procedures • Maximum of 30 users editing procedures
OS	Linux 64-bit: • Debian, Ubuntu • Red Hat, CentOS	Linux 64-bit: • Debian, Ubuntu • Red Hat, CentOS
RAM	8 GB	16 GB
CPU	2 vCPU	4 vCPU
Disk space	150 GB dedicated to application	150 GB dedicated to application

Multiple presentation server architecture

This Self Help architecture is used to:

- Dedicate a server to edition functions;
- Dedicate other servers to procedure presentation functions and the virtual agent.

Description of edition server configuration:

Elements	Nominal configuration	Maximum configuration
Workload	15 users editing procedures	Maximum of 30 users editing procedures
OS	Linux 64-bit: • Debian, Ubuntu • Red Hat, CentOS	Linux 64-bit: • Debian, Ubuntu • Red Hat, CentOS
RAM	8 GB	16 GB
CPU	2 vCPU	4 vCPU
Disk space	60 GB dedicated to application	60 GB dedicated to application

Description of presentation server configuration:

Elements	Nominal configuration per server	Maximum configuration per server
Workload	150 users executing procedures	Maximum of 300 users executing procedures
OS	Linux 64-bit: • Debian, Ubuntu • Red Hat, CentOS	Linux 64-bit: • Debian, Ubuntu • Red Hat, CentOS
RAM	8 GB	16 GB
CPU	2 vCPU	4 vCPU
Disk space	150 GB dedicated to application	150 GB dedicated to application

Client workstation configuration for studio

Self Help Studio is a Java/Eclipse RCP application which operates in connected mode ((port 80) or https (port 443) on TCP/IP) on the central edition server. It requires a Java 1.8 virtual machine as its sole architecture, supplied and installed with the studio.

Hardware configuration:

Prerequisites	
OS	Windows 7 and later
CPU	Intel core i3 or higher (or AMD equivalent)
Web browser	For standard themes delivered with the software, the following browsers are supported: • Google Chrome 70 • Firefox version 63.0 • Edge 42 • Internet Explorer 11
Memory	4 GB
Disk space	1 GB

Studio can connect to the edition server via an enterprise proxy.

SELF HELP

Technical requirements

All servers for a Self Help platform are identical, whether they are used as presentation server or edition server.

Function	Prerequisites
Web	• Apache2 2.4.X • Tomcat 8.5 and 8.X later
Engine	• Wildfly 16.0.0 • Openjdk-8-jre-headless recommended with latest version supported by the OS

Databases	• PostgreSQL depending on machine OS:
	  • 10.X
	  • 11.X
	• Mongo:
	  • 3.6 and 3.X later
	  • 4.2

External flow matrix

Source	Destination	Ports	UDP / TCP
Your users	Presentation server	443 (https)	TCP
Your function administrators	Edition server	443 (https)	TCP
Edition server	Presentation server	443 (https)	TCP
Presentation server	WEB SE front end	443 (https)	TCP
Presentation server	WEB SA front end	443 (https)	TCP
WEB SE front end	Presentation server	443 (https)	TCP

USER AUTHENTICATION

Role distribution

For Self Help, we distinguish:

- Authentication: Confirmation of the identity of the person attempting to connect;
- Authorization: What the person identified is permitted to do in Self Help.
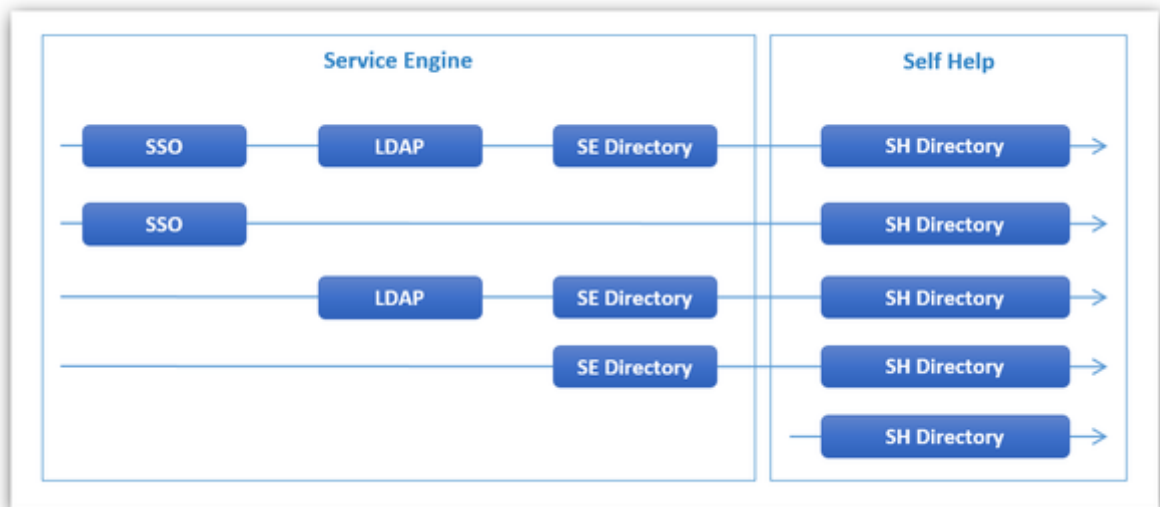
User authentication in Self Help

Self Help coupled with Service Manager

Self Help offers the following authentication methods:

- Via the application's internal employee database;
- Via Service Manager.

Authentication methods supported by Service Manager are:

- via the application's internal employee database;
- Authentication based on your LDAP/AD directories;
- Authentication based on an SSO compatible with our services.

Self Help in standalone mode

Self Help can be connected to any other tool without the Service Manager application.

Self Help offers the following native authentication methods::

- SSO SAML as a client on your SSO system;
- By bing on your LDAP servers;
- Via the application's internal employee database.

APPENDICES

Network

IPV6 is not used. You can deactivate this layer on your servers if necessary.

Antivirus

The local antivirus must be configured to not scan the following directories:

- Application log storage folder (XML format files);
- Database and log file storage folder.

Apache Configuration

List of modules to be integrated:

- rewrite
- headers
- proxy
- proxy_http
- expires
- socache_shmcb_module
- deflate
- setenvif
- ssl