EASA System Architecture

The EASA system is highly configurable to meet functional requirements specified by a **User** and implemented in an EASA application (an “EASAP”).

Consider which resources an EASAP requires. These might include:

- A Database
- A Microsoft Excel spreadsheet
- Batch computation run in a dedicated environment
- Other third party software

To a large extent, the particular resource requirements above will define the EASA architecture.

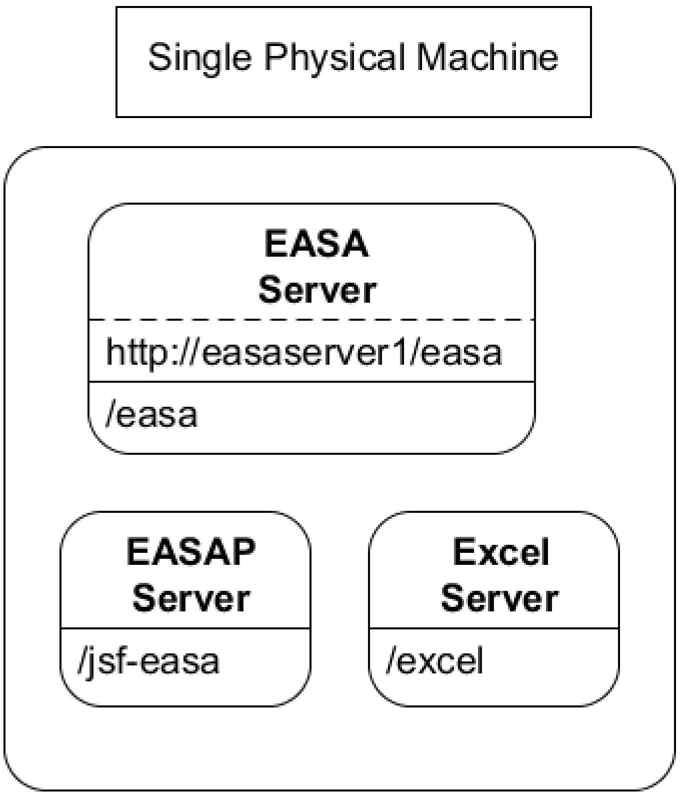
Processing and performance requirements may place constraints on the capability and the number of machines recruited for an EASA system. These include:

- Number of simultaneous **Users**
- Degree of complexity within an Excel spreadsheet
- Availability of 3rd party software licenses
- Intrinsic complexity of the EASAP

The following four architecture examples cover the most common use cases.

Default EASAP

The simplest EASA system combines several EASA computational roles within a single physical machine.



This configuration has the same capabilities of a distributed architecture. The default single EASA machine provides a simple, straightforward way to start off.

For example, after installing the 'off-the-shelf' system above, one may:

- Run an example EASAP as a **User**
- **Author** a new EASAP from the steps in a tutorial

This architecture is not generally used in production as it will likely struggle under any significant **User** load. In particular, the **Excel** resource requirements (memory and CPU) will exceed the single machine's capacity above a modest number of simultaneous **Users**.

The diagram above illustrates three EASA computational roles in an EASA system:

- The **EASA Server** is a single machine and occupies a central role in every EASA system
  - On an **EASA Server**, a **User**, an **Author** or an **Administrator** perform their various duties
    - A **User** may select a published EASAP which is then dispatched to be run on an **EASAP Server**
    - An **Author** may select an EASAP in development, modify, save or test it via the EASAP **Builder**
    - An **Administrator** may manage the operation and administration of the entire EASA system

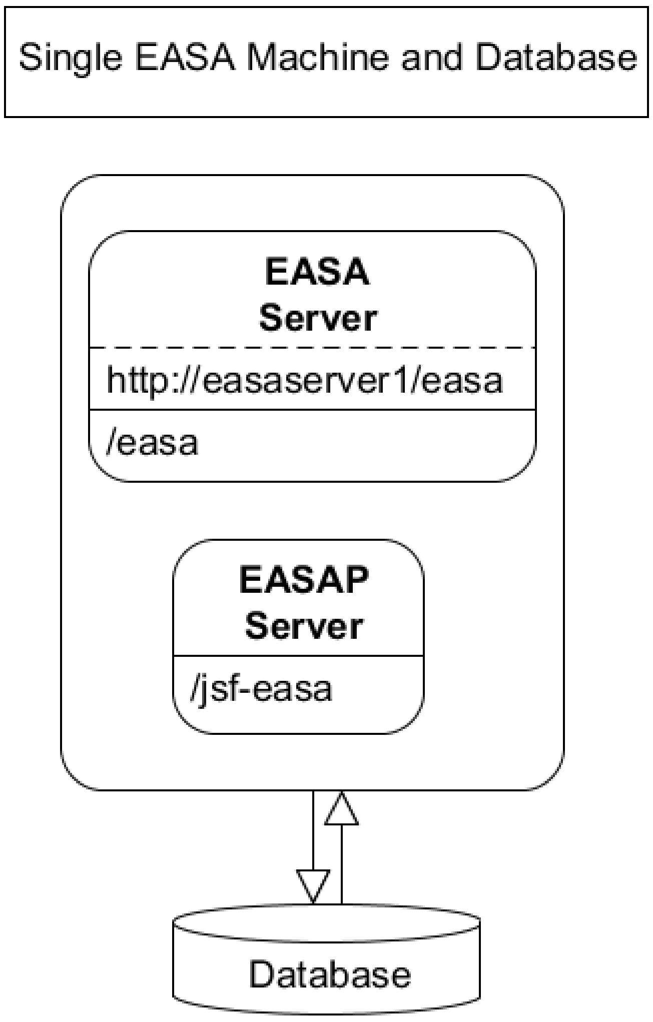
- An **EASAP Server** hosts the Java application that serves the EASAP user interface to a browser
  - A **User** is allocated a unique instance of the EASAP for the duration of the task

- An **Excel Server** hosts the spreadsheet and live Excel instance that is linked to each running EASAP
  - An **EASAP/Excel Server** pairs two server roles
  - As the number of **Users** grows, more **EASAP/Excel Servers** may be needed

Database Backend EASAP

One common configuration involves a backend database and a combined **EASA/EASAP Server**, below.

The **EASA/EASAP Server** hosts a user interface, creates database queries, and retrieves result sets.



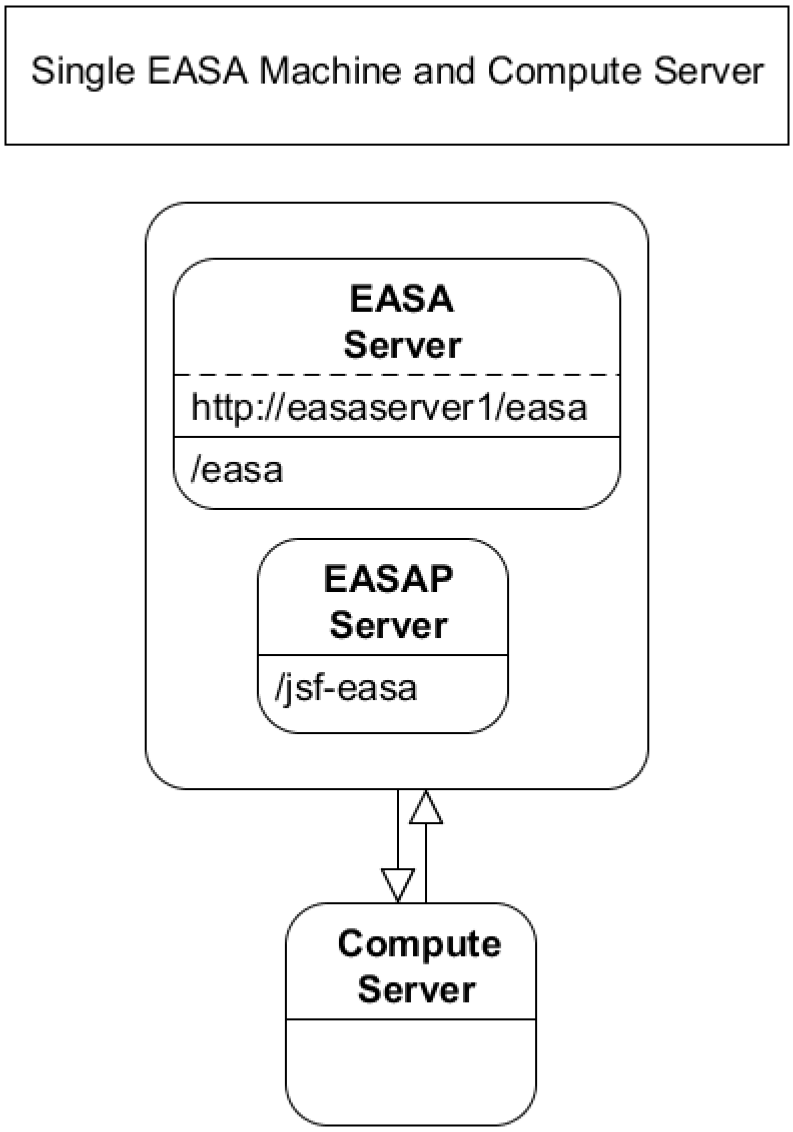
In general, this system will scale well with increasing **Users** for two reasons:

- The database is a scalable product.
- The **EASA/EASAP Server** has a lightweight role as a database client.

Specialized Compute Server EASAP

A similar architecture involves specialized computation on a machine called a **Compute Server**.

This computation may require Matlab, a C program, a batch script, or other proprietary software.



This type of system may utilize a single-user license on the **Compute Server** and allow the **EASA/EASAP Server** to manage a queue of submissions. One **User's** EASAP may be granted temporary software license access to execute a task. When the task completes, the license will be released to the next **User** EASAP instance in the queue.

Excel-intensive EASAP

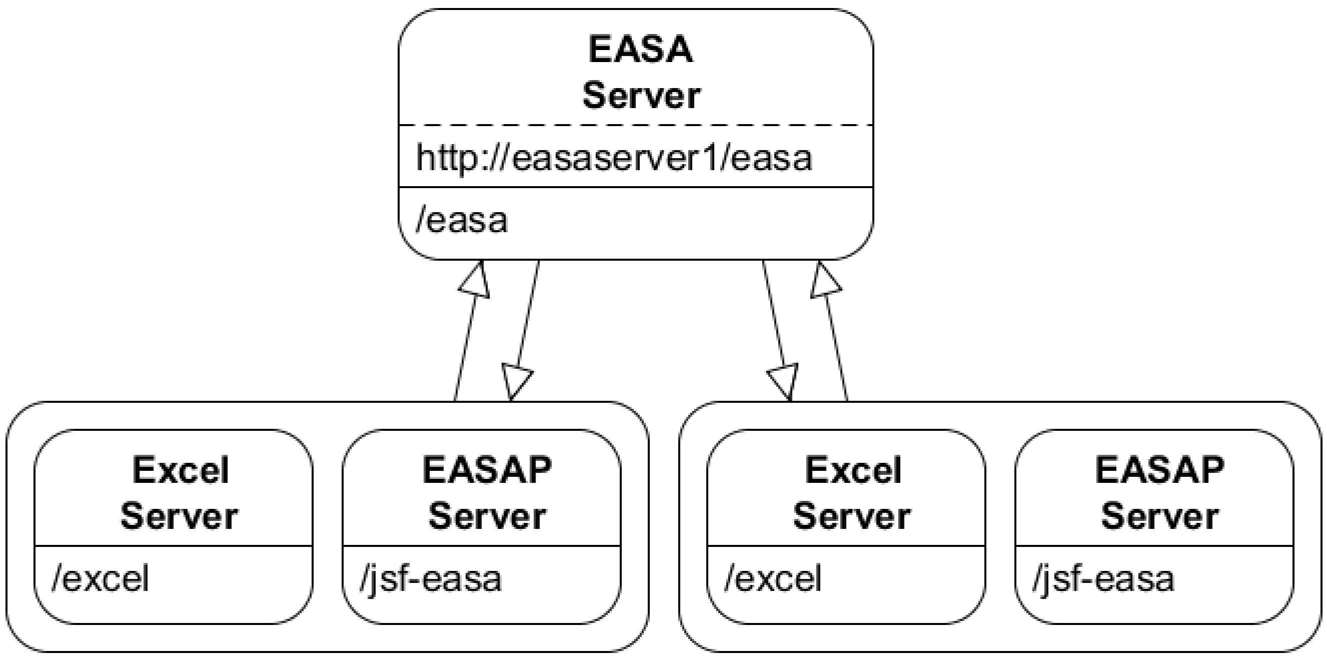
One of the most common and computationally intensive cases involves a 'live' Excel-linked EASAP that:

- May contain a large spreadsheet
- May involve lengthy computational dependencies between cells
- May need to be available for many simultaneous **Users**

This case highlights EASA's impressive ability to scale: a single **EASA Server** will balance **User** load across multiple **EASAP/Excel Servers**.

The diagram below shows two **EASAP/Excel Servers**, though typically in practice there are many more.

EASA Server and two EASAP/Excel Servers


EASA Server
http://easaserver1/easa
/easa
Excel Server
/excel
EASAP Server
/jsf-easa
Excel Server
/excel
EASAP Server
/jsf-easa