Architecture of packaged documentation server

Inhalt

[1 Inhalt 1](#_Toc171782553)

[2 Modules comprising the software 2](#_Toc171782554)

[2.1 Description of the individual module categories 2](#_Toc171782555)

[2.1.1 Main engine 2](#_Toc171782556)

[2.1.2 Server plugin 2](#_Toc171782557)

[2.1.3 Configuration plugin 2](#_Toc171782558)

[2.1.4 Documentation package plugin 2](#_Toc171782559)

[2.1.5 Frontend loader 2](#_Toc171782560)

[2.1.6 Frontend plugin 2](#_Toc171782561)

# Modules comprising the software

1. The main engine loading the modules for the distribution of the documentation
2. At least one server plugin
3. At least one documentation package plugin
4. At least one configuration plugin
5. At least one frontend loader to load the engine and a frontend plugin
   * Initially a command line loader is sufficient
6. At least one frontend plugin
   * Initially a command line frontend is sufficient

## Description of the individual module categories

### Main engine

The main engine’s purpose is to locate the available server and documentation package modules and hand them over to the frontend to present a selection to the user if no configuration is present. If an engine configuration is available the engine checks if the configured modules are existing and loads them if they exist, otherwise a notification to the user is presented.

### Server plugin

A server plugin delivers the served documentation to a client. An example of a server plugin would be an http web server.

### Configuration plugin

A configuration plugin provides a possibility to read and optionally write a server configuration. The source of the configuration can be for example the command line for a read only configuration. Another example would be one or more files holding the configuration in a dedicated format.

### Documentation package plugin

The documentation package plugin provides access to the documentation files. The simplest form of a documentation package plugin is for direct directory access on the server. A more sophisticated one would give access to documentation inside a zip file for example.

### Frontend loader

The actual executable used to start the server it loads the main engine and a frontend plugin, where the main engine represents the back end and the frontend plugin the user interface to control the server.

### Frontend plugin

The user interface of the server. This is not necessarily interactive, for a command line status output would be sufficient for a server configured through a configuration file.