Executable library

The executable library is a structured collection of development objects that users can call directly. An executable fulfills a process step function. For instance, a transaction or an SAP Fiori application can be called by a user. Accordingly, you can consider them executables. All development objects such as enhancement implementations or tables which a user cannot execute directly are not considered executables, but plain development objects.

**Definition**

The executable library is a software dependent inventory of executables which fulfill business functionalities as promised by the process steps.

The executable library contains “originals” and is supposed to have no duplicates. If an executable is used in a process step you add it as a “reference” to the usage occurrence. The reuse comes with the advantage that you document every executable only once, as the code exists only once. For example, a technical specification should be added to an executable because the technical design is shared and brings together the requirements of all business use cases. If you have documentation specific to an individual occurrence, you can add it to the references. This adds documentation to the place where it makes most sense. Generally speaking, you should document elements like technical specifications, configuration units, technical KPI document, or job documentation at the executable level.

The executable library’s first level contains a structure of logical component groups. Below the first level, every software system comes with an independent structure. Be sure to follow the software systems’ internal structures.

SAP software generally structures applications according to the application component hierarchy (SD, BC, or MM). Non-SAP applications can be represented according to their own software structure. There is no need to build the executable library manually. For ABAP-based systems, you can automatically generate and refresh the library based on usage data. Custom executables are sorted into library folders that correspond to the development packages of the executable.

**Recommendation**

The executable library is structured by logical component groups and you should leverage the automatic library generation for ABAP-based systems. Please check the appendix for naming conventions.