

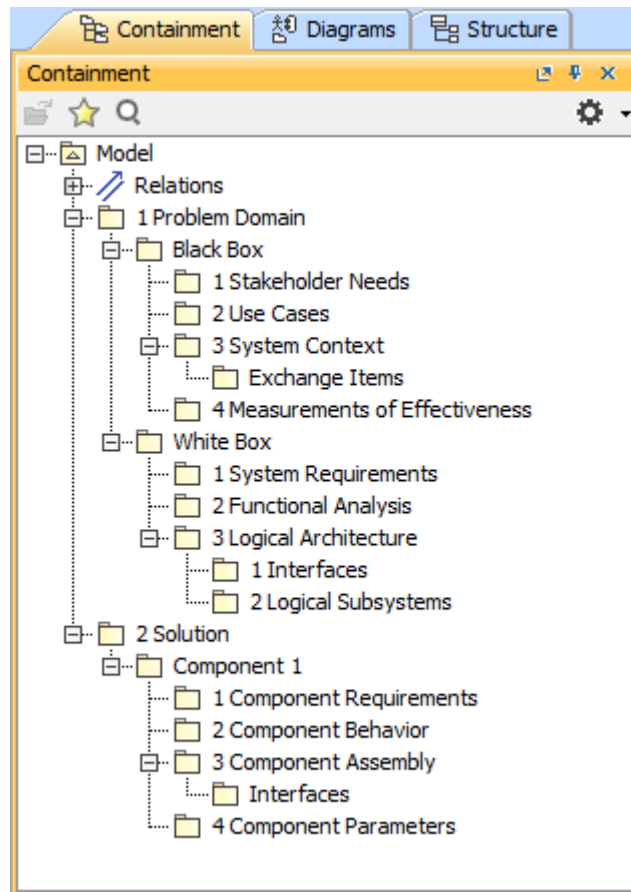
# Using packages

There are multiple ways to organize your model into [Packages](#). The S. Friedenthal in the "Practical Guide to SysML" recommends to organize your model structure in the following ways:

- By system hierarchy (e.g., system level, subsystem level, component level).
- By process life cycle where each model subpackage represents a stage in the process (e.g., requirements analysis, system design).
- By teams that are working on the model (e.g., Requirements Team, Integrated Product Team (IPT) 1, 2).
- By the type of model elements contained in it (e.g., requirements, behavior, structure).
- By model elements that are likely to change together.
- By model elements organized to support reuse (e.g., model libraries).
- By other logical or cohesive groupings of model elements based on defined model-partitioning criteria.
- A combination of the preceding.

You can also organize your model structure according to your internal methodologies. The procedures of managing Packages in the modeling tool are provided in the [Working with packages](#) and [Package import](#) pages.

In the following figure, the Packages are organized according to MagicGrid approach. You can use this predefined package structure by creating a MagicGridBlank project. The example and guidelines of how to use the MagicGrid approach, you can find by creating the MagicGridQuickStart project. [How to create MagicGrid projects >>](#)



The Package structure of MagicGrid project.

\$body