Using smart packages in your model

To better understand the usability of the smart packages, please see the following case studies.

Case study #1: Gathering use cases

The case study uses the sample project *use case diagram.mdzip*, which can be found in *<MagicDraw installation folder>\samples\diagrams*.

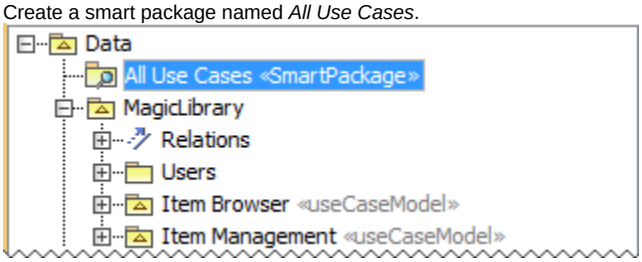
For example, we have a lot of use cases owned by different system boundaries in the project.

We need all these use cases to be in a single package, except for several particular ones. Also, we need all newly created use cases to be automatically included into this package.

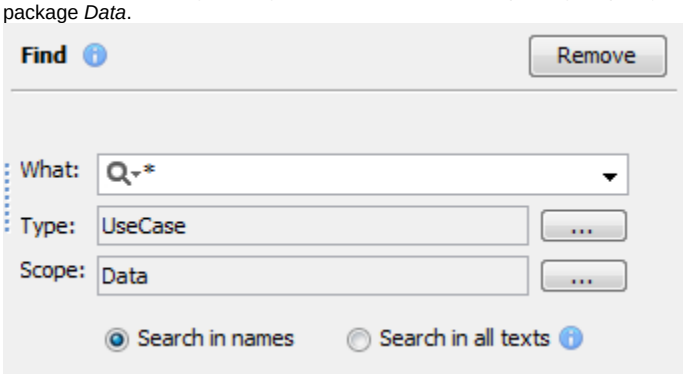
In this case, the smart packages feature is very useful. We will create a smart package with dynamic contents to gather all use cases in the model, then demonstrate how it manages further changes in the model. Finally, we will create a snapshot of the smart package to have a static list of use cases as a milestone of the model development.

To accomplish this, do the following:

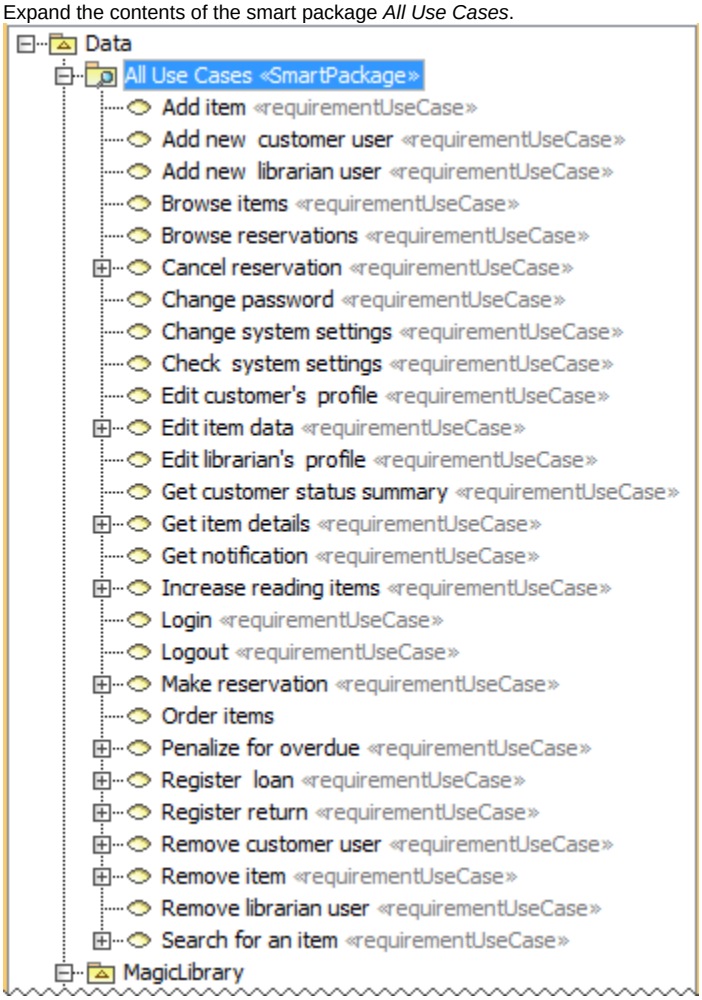
1. Create a smart package named *All Use Cases*.



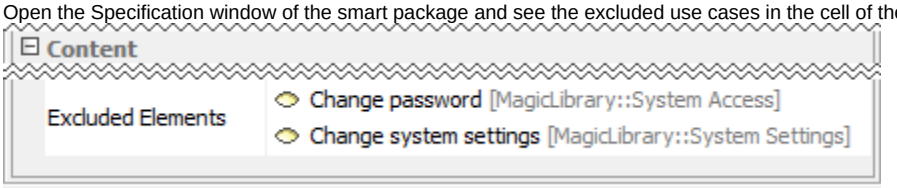
2. Define the criteria for gathering the contents of the smart package. Specify the search options to find all use case type elements from the root package *Data*.



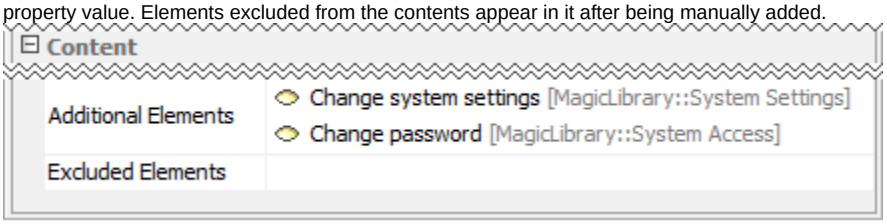
3. Expand the contents of the smart package *All Use Cases*.



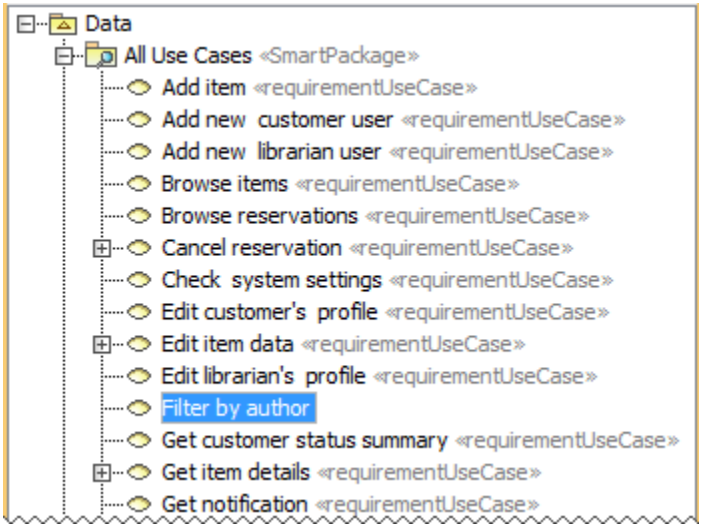
4. Exclude use cases *Change password* and *Change system settings* from the contents. See the contents of the smart package shortened.
5. Open the Specification window of the smart package and see the excluded use cases in the cell of the **Excluded Elements** property value.



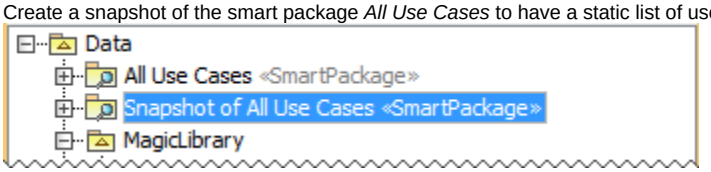
6. Drag use cases *Change password* and *Change system settings* to the smart package. See them in the contents of the smart package again.
7. Open the Specification window of the smart package again and see these manually included use cases in the cell of the **Additional Elements** property value. Elements excluded from the contents appear in it after being manually added.



8. Create a new use case named *Filter by author* under the system boundary *Item Browser*. See the new use case in the smart package *All Use Cases*.



9. Create a snapshot of the smart package *All Use Cases* to have a static list of use cases as a milestone of the model development.



10. Expand the contents of the snapshot and see that it equals the contents of the smart package *All Use Cases*.

Case study #2: Performing the requirements coverage analysis

The case study uses the sample project *hybrid sport utility vehicle.mdzip*, which can be found in <*MagicDraw installation folder*>*samples**SysML*, if the SysML plugin is installed.

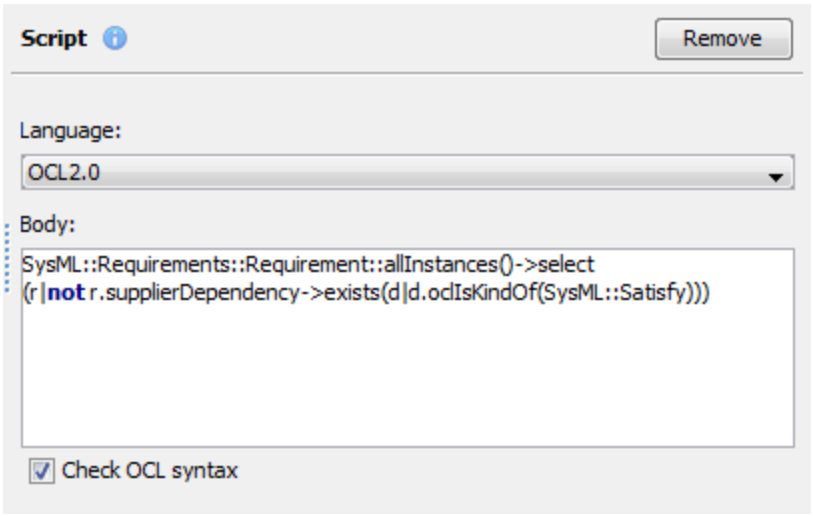
Here is another case where the smart package feature is very helpful: we need to have all unsatisfied requirements in a separate package. Also, we need requirements to automatically disappear from this package after becoming satisfied.

We will create a smart package with dynamic contents to gather all the unsatisfied requirements in the model and a dependency matrix for performing the requirements coverage analysis. Then we will demonstrate how both the smart package and the dependency matrix reflect the transition of a requirement to satisfy.

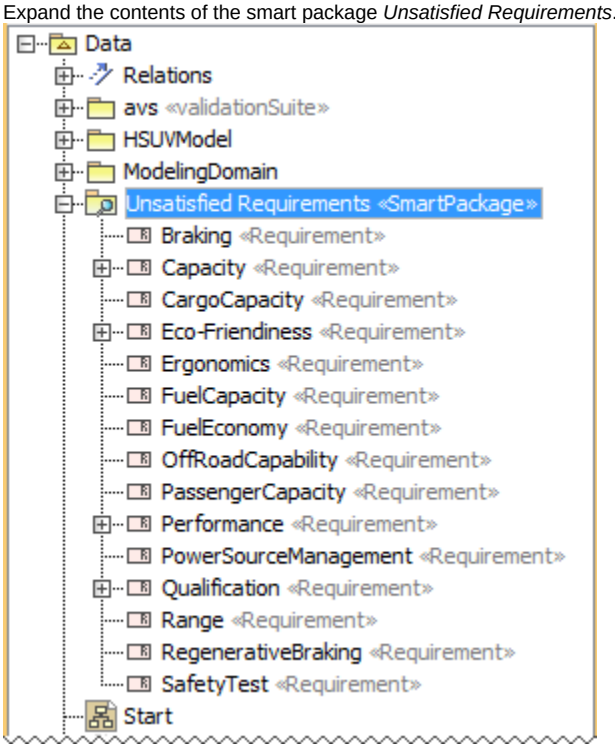
Let's do the following:

1. Create a smart package named *Unsatisfied Requirements*.
2. To gather the contents of the smart package, add a new script operation (in the **Expert** mode of the **Query** dialog) and define the following OCL 2.0 expression as the criteria:

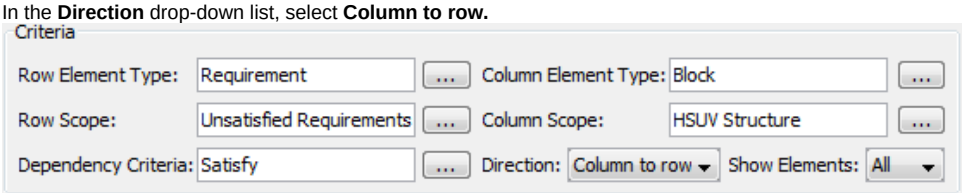
```
SysML::Requirements::Requirement::allInstances()->select(r|not r.supplierDependency->exists(d|d.
oclIsKindOf(SysML::Satisfy)))
```



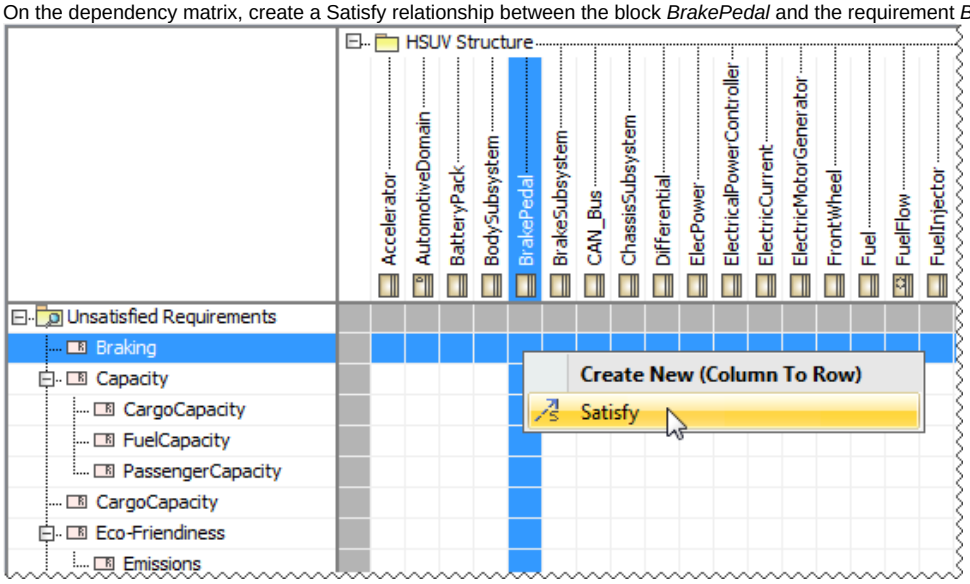
3. Expand the contents of the smart package *Unsatisfied Requirements*.



4. Create a dependency matrix and define the following criteria:
   - Specify Requirement as the row element type
   - Specify Block as the column element type
   - Specify the smart package *Unsatisfied Requirements* as the row scope
   - Specify the package *HSUV Structure* as the column scope
   - Specify the Satisfy relationship as dependency criteria
   - In the **Direction** drop-down list, select **Column to row.**



5. On the dependency matrix, create a Satisfy relationship between the block *BrakePedal* and the requirement *Braking*.

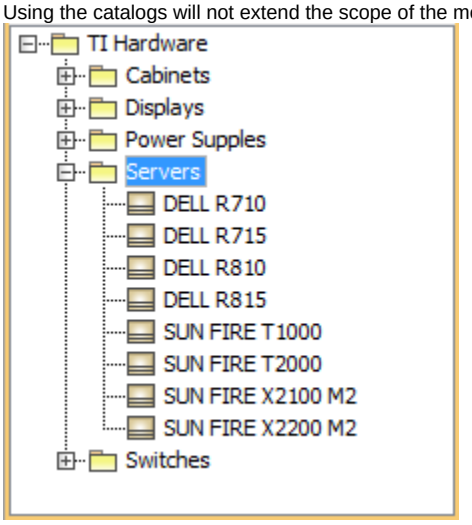


The requirement becomes satisfied and thus disappears from the contents of the smart package *Unsatisfied Requirements* and from the dependency matrix as well.

Case Study #3: Configuration management of the complex system - creating dynamic configuration catalogs

The efficient configuration management process is a challenge in the evolution of any industrial scale product family. Smart packages are a real-life out of the box solution supporting the configuration management approach.

Now study the case that illustrates the efficient management of the complex system configurations with the help of smart packages. We have a library (static package) of system components, which we need to see in several different views of the model, that is, catalogs, according to their characteristics. Using the catalogs will not extend the scope of the model, since they do not require duplication of the elements.

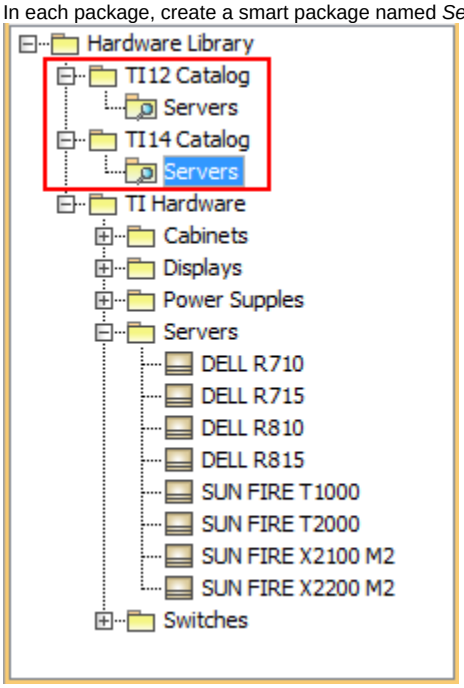


We will create two dynamic system configuration catalogs, that is, smart packages with dynamic contents, to gather the servers from the library *TI Hardware* according to the configuration version defined in a tag value of their specification.

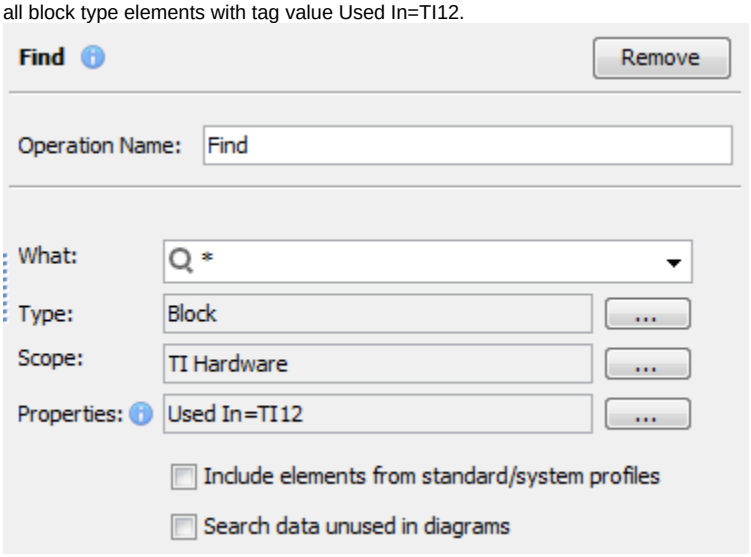
Do the following:

1. Create two packages: *TI12 Catalog* and *TI14 Catalog*.

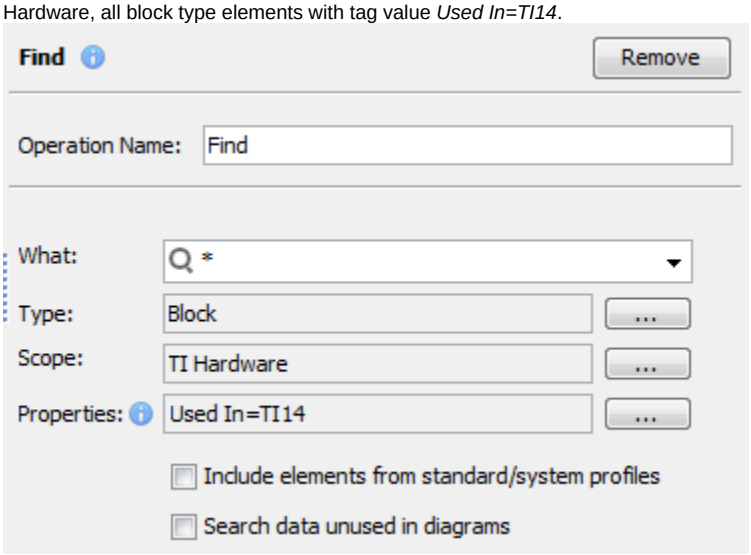
2. In each package, create a smart package named *Servers*.



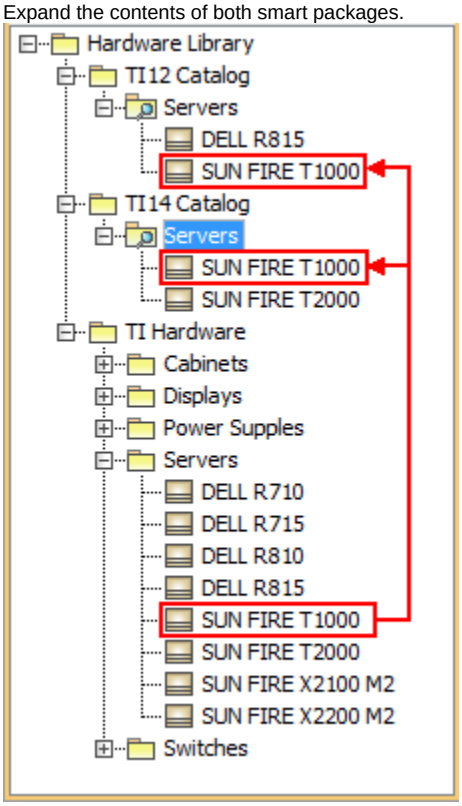
3. Define criteria for gathering the contents of the smart package *Servers* in *TI12 Catalog*. Specify search options to find in the package *TI Hardware*, all block type elements with tag value Used In=TI12.



4. Define criteria for gathering the contents of the smart package Servers in TI14 Catalog. Specify search options to find in the package TI Hardware, all block type elements with tag value *Used In=TI14*.



5. Expand the contents of both smart packages.



> The block *SUN FIRE T1000* appears in both *TI12 Catalog* and *TI14 Catalog*, since it has both tag values *Used In=TI12* and *Used In=TI14*.

6. Also, you can add a block to a smart package manually. Just drag the block to the smart package.

Related Pages

- Specifying criteria for querying model
- Model Elements
- Stereotype
- Package
- Working with Profiles