Activities Lesson 3.2 MD18

Outlining the Activity

After the second iteration the **QueryProductsService** will provide more output than just the list of products. First, you will extend the activity diagram **Query Products** of the port type operation.
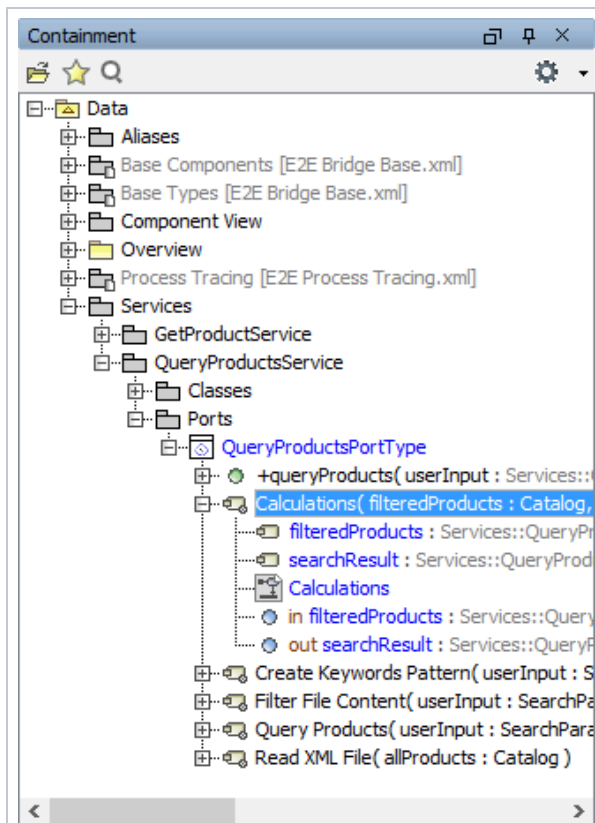

Create the new activity diagram **Calculations** in the package **Services > QueryProductsService > Ports > QueryProductsPortType**.

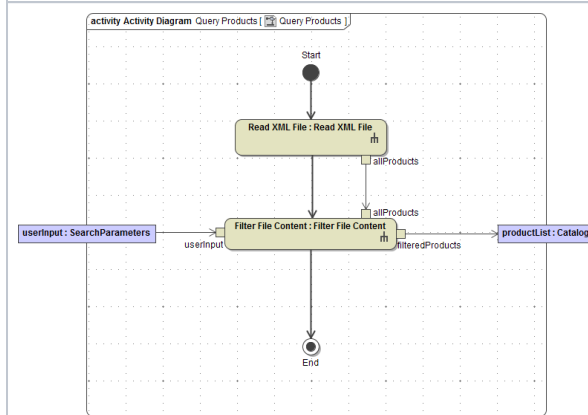Implementing the Currency Calculator

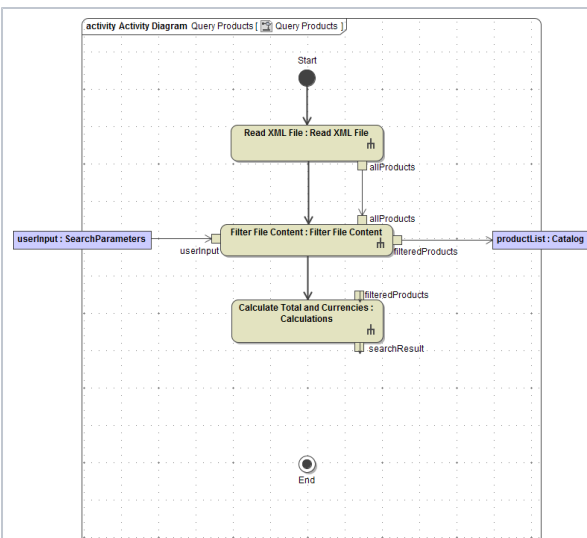**On this Page:**

- Outlining the Activity

Calculations will receive the filtered data from **Filter File Content**, add some calculations to it and deliver a new search result.

Add the necessary parameters to **Calculations**:

- **filteredProducts : Catalog**, **in**.
- **searchResult: SearchResult**, **out**.

Open the activity diagram **Query Products**.

activity Activity Diagram Query Products [Query Products]

Start

Read XML File : Read XML File

allProducts

allProducts

userInput : SearchParameters

Filter File Content : Filter File Content

userInput

filteredProducts

productList : Catalog

filteredProducts

Calculate Total and Currencies : Calculations
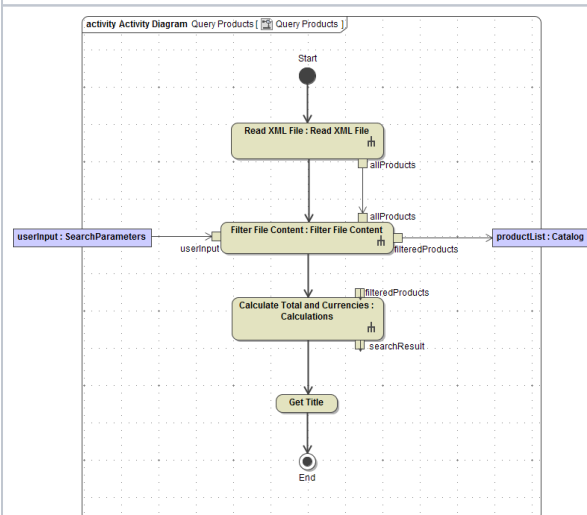
searchResult

End

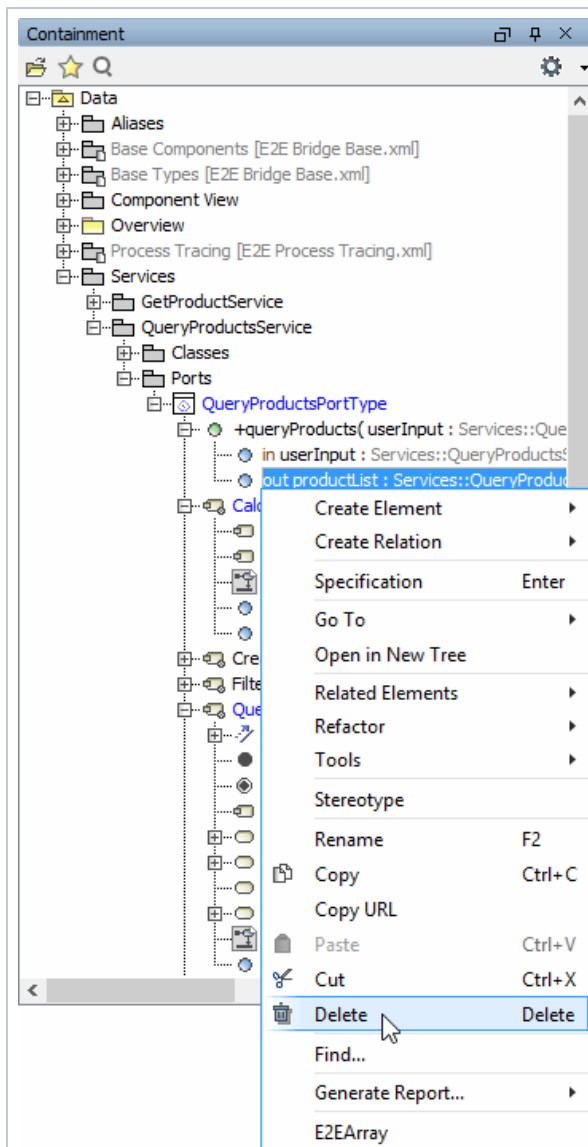Delete the control flow between the action node **Filter File Content** and the activity final **End**.

Drag and drop the activity **Calculations** on the diagram pane. Assign the name **Calculate Total and Currencies**.

Move the pins **filteredProducts** and **searchResult** on the right side of the action node.

Draw a control flow from **Filter File Content** to **Calculate Total and Currencies**.

activity Activity Diagram Query Products [Query Products]

Start

Read XML File : Read XML File

allProducts

allProducts

userInput : SearchParameters

Filter File Content : Filter File Content

userInput

filteredProducts

productList : Catalog

filteredProducts

Calculate Total and Currencies : Calculations
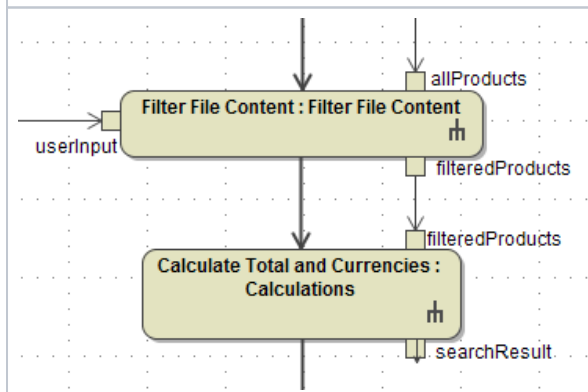
searchResult

Get Title

End

Draw the new action node **Get Title** and add the missing control flows as shown in the picture on the left.
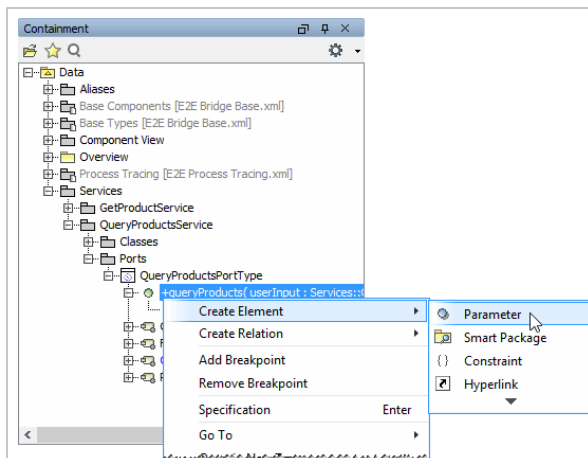
Remove the output parameter **productList** from the diagram as well as from the containment tree as it is not needed any longer.
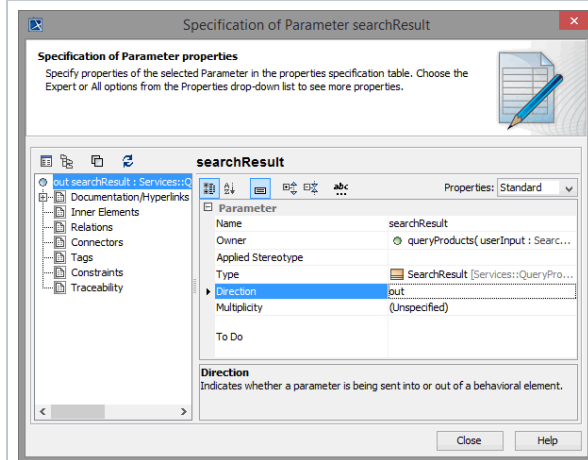
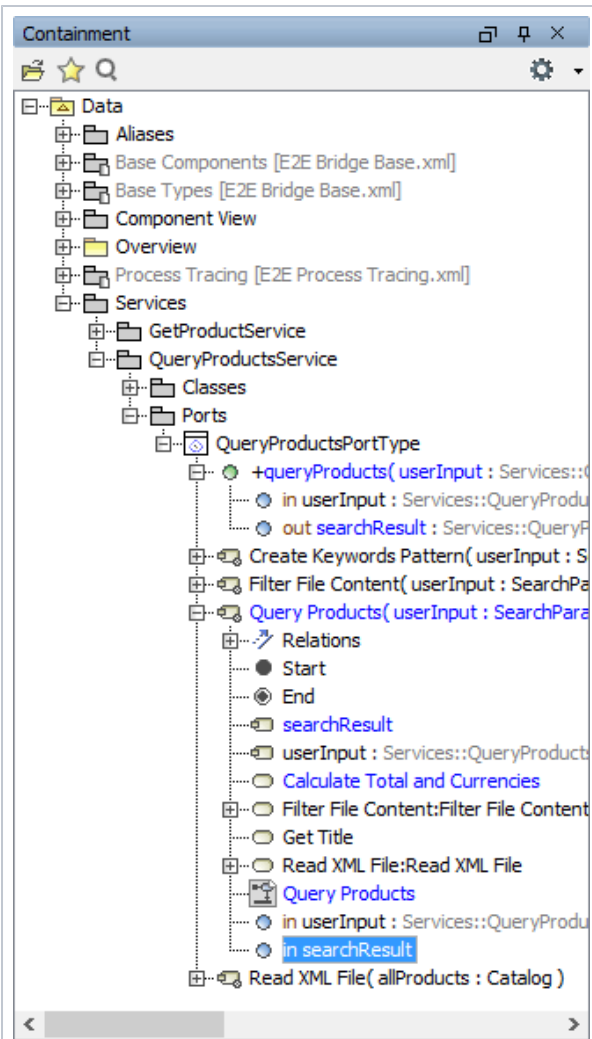Also remove it from the port type operation **queryProducts**.

Then, move the output pin **filteredProducts** as shown on the left and connect it to the corresponding input pin of **Calculate Total and Currencies**.
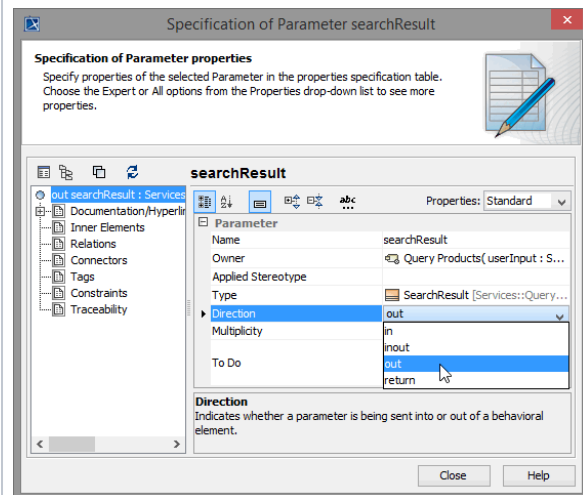
Add the new parameter **searchResult** to the port type operation **queryProducts**.
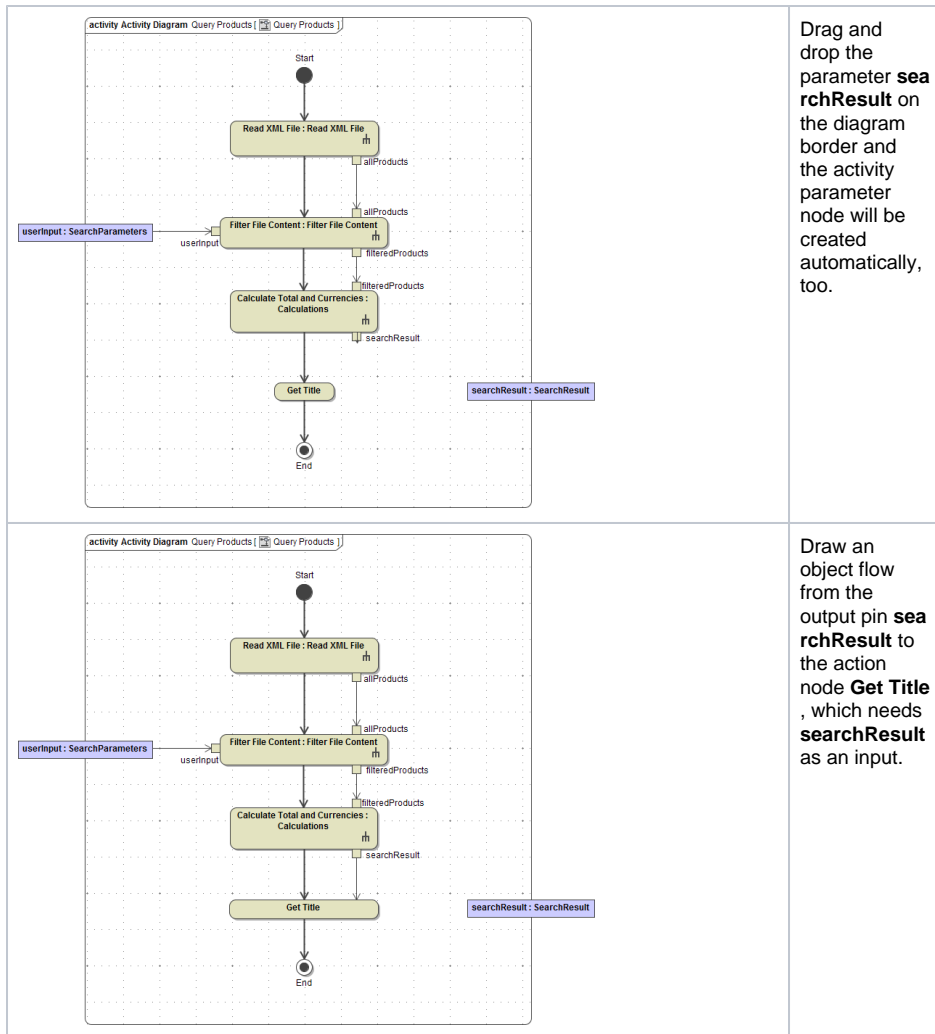

Define it as of type **SearchResult** and direction **out**.

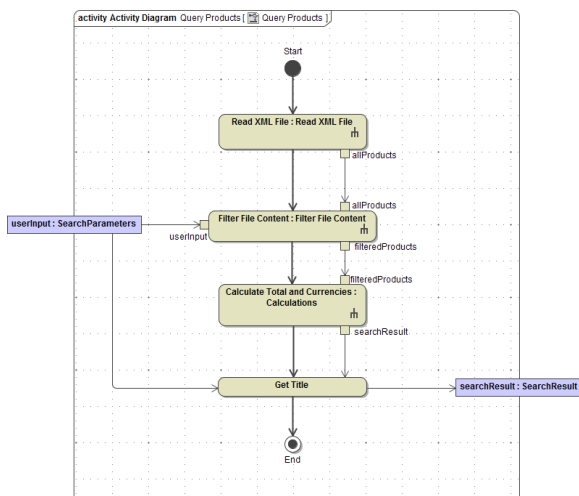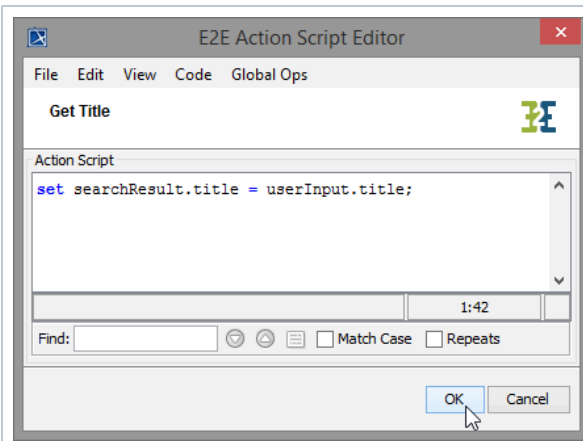Note that the parameter of the related activity was created automatically.

Check the type and the direction of this automatically created parameter and set it to **SearchResult** and **out** if necessary.

Drag and drop the parameter **searchResult** on the diagram border and the activity parameter node will be created automatically, too.

Draw an object flow from the output pin **searchResult** to the action node **Get Title**, which needs **searchResult** as an input.

In the last step, you will set the title to be returned in the action node **Get Title**. It will be assigned to the corresponding attribute in the output object **searchResult**. You need to set the value of attribute **title** of object **searchResult**. Therefore, the object is used as input of the action node **Get Title**. Connect the input parameter **userInput** to the action node **GetTitle**.
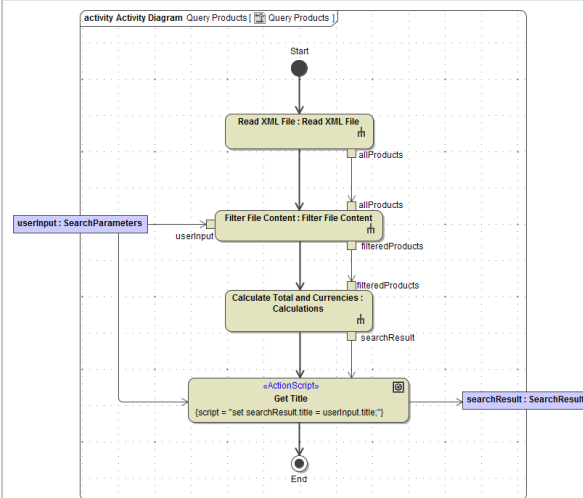
As you change an attribute of the object **searchResult**, you need to draw an object flow from the action node to the activity parameter node **searchResult**. The object will be passed to the caller, the port type operation **queryProducts**.

Finally, add the following action script to the action node **Get Title**:

```
set
searchResul
t.title =
userInput.
title;
```


The activity diagram **Query Products** is finalized.

Save the UML model.