ESI
Powered by industry, academia and TNO

Tutorial Capella Add-On

Design Space Exploration for Customer Workflows

Jozef Hooman, Koen Kanters, Alexandr Vasenev, Jacques Verriet 2023-03-11

1 INTRODUCTION

This tutorial describes the use of an open-source prototype tool [1] to support design decisions early in the process of system development by quantitative information about important system qualities. The current version concentrates on the duration of customer workflows. Our approach is based on Arcadia [2] and the prototype tool extends the corresponding open-source tool Eclipse tool Capella [3]. Assumptions about resources and durations of functions have to be added to Capella's functional chains. This is done using the PVMT add-on [4]. Next, the annotated functional chains are automatically translated to a representation of Petri nets for performance simulation. The simulation stores the results in a format that can be used by TRACE4CPS [6] to visualize the execution in different ways and perform analysis such as critical path analysis. Finally, the tool also contains the possibility to perform design space exploration by a dedicated application.

This tutorial is structured as follows. The tool installation and creation of a workspace is explained in Section 2. It is also explains how to download example projects. Section 3 provides a quick introduction of the approach by means of the 3D reconstruction example. More details can be found in subsequent sections. Section 4 describes how to create a functional chain using logical components with functions. The annotation of functional chains with value can be found in Section 5. Section 6 describes how to simulate the functional chains. Visualization of the results of a simulation and critical path analysis is explained in Section 7. Nested functional chains and reuse of chains are described in Sections 8 and 9, respectively. Section 8 also describes an alternative approach to define functional chains using function decomposition. Design space exploration can be found in Section 10. The use of EPBS configuration items as resources is illustrated in Section 11. A few details about PVMT are provided in Section 12. Notes on opening existing projects, adding pictures, simulation and known issues can be found in Section 13.

2 INSTALLATION AND CREATION OF A WORKSPACE

- Download capella-workflow-dse-win32-x64-….zip from https://github.com/TNO/capella-workflow-dse/releases, unzip it to a folder, double-click *capella.exe*, launch it with the default workspace or choose an existing folder with Capella projects. Remove the Welcome screen.
- The workspace (which contains the Capella projects) can be changed by:
  Tab File > Switch Workspace > browse to folder.
- Example projects are available using Tab File > New > Example > Capella Workflow DSE (currently there is one example project called 3D Reconstruction).

3 EXAMPLE 3D RECONSTRUCTION

A fast impression of the approach can be obtained by experimenting with the available 3D reconstruction example. Import the example using:

- Tab File > New > Example > Capella Workflow DSE; select 3D Reconstruction, click Next and Finish
- Double-click on the 3D Reconstruction.aird file to open the model

This model contains a number of ingredients of an artificial application of electron microscopy, namely the construction of a 3D model out of 2D images:

- Representation per category > Logical Architecture > Logical Function Breakdown > [LFBD} Root Logical Function describes the function hierarchy
- Representation per category > Logical Architecture > Logical Data Flow Blank contains 3 functional chain diagrams:
    - [LDFB] 0. 3D Reconstruction, which contains the following 2 chains:
    - [LDFB] 1. Acquire 2D Image
    - [LDFB] 2. Acquire 2D Image
- Representation per category > Common > Functional Chain Description contains 3 corresponding functional chain descriptions:
    - [LFCD] 00. 3D Reconstruction, which represents a customer workflow and contains the following 2 chains:
    - [LFCD] 1. Acquire 2D Image; the comments in this model provide some explanation
    - [LFCD] 2. Acquire 2D Image

    Open diagram [LFCD] 00. 3D Reconstruction and next inspect the property values using Tab Windows > Show View > Property values > Property values.

    Click on an element of the diagram to see:
        - Duration and ResourceID of functions
        - The number of repetitions of an IT node
        - The weight of outgoing arcs of a start OR node; these weights indicate the probability with which these arcs will be taken by the simulation.
          For instance, if there are three outgoing arc with weights 1, 2, and 3 then they are taken with probability 1/6, 2/6 and 3/6, respectively.
- Representation per category > EPBS architecture > Configuration Items Breakdown > [CIBD] Structure defines four configuration items.

Sections 4 and 5 describe how to construct such functional chains. The construction of configuration items can be found in Section 11.

To obtain the duration and costs of a workflow, represented by a functional chain, a separate application has been developed which also allows design space exploration (DSE).

To open the DSE application,

- Right-click on file [LFCD] 00. 3D Reconstruction or in its diagram
- Select Workflow DSE: Run

This leads to the pop-up window of Figure 1.

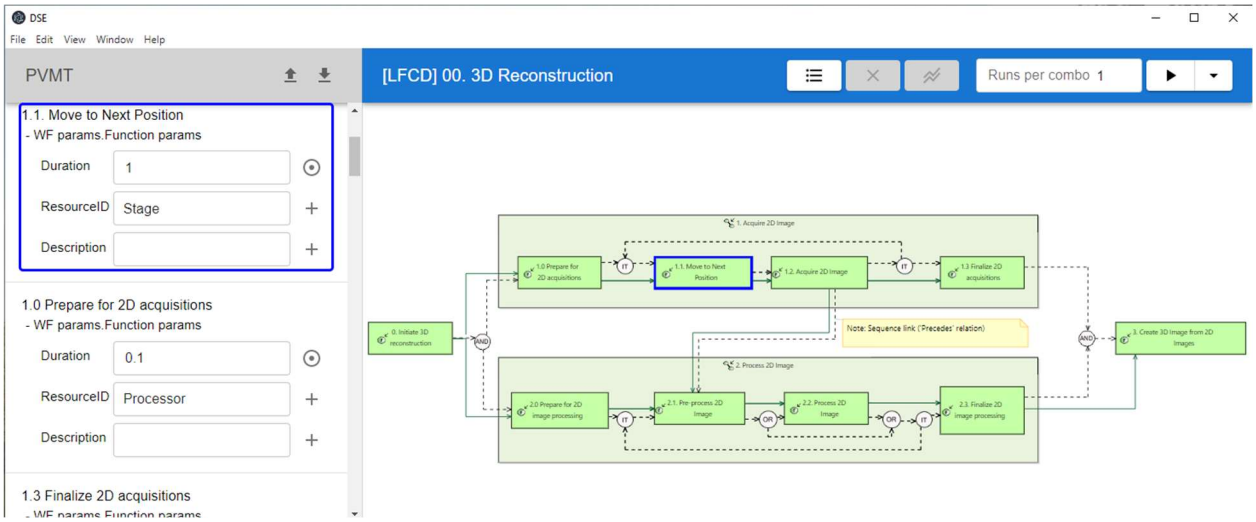


Figure 1 DSE application

Details on the use of this application can be found in Section 10. Here, a short description will be given.

On the left of the DSE application, values can be changed. For a duration, ranges can be specified using  . Similar for repetitions and weights. Alternative resources can be specified using the "+".

By  a definition editor can be opened which allows the specification of durations of functions on resources. Also the costs of resources can be specified.

By  a simulation can be started; it will simulated all possible combinations of alternatives for durations and resources.

The results can be viewed using  , previous results can be removed using  .

4 CREATE FUNCTIONAL CHAINS

First create a new project:

- Choose File > New > Project > Capella project; give a name; Finish

Next create a new architectural diagram:

- Choose Logical Architecture > Define Logical Components and Actors > [LAB] Create a new logical Architecture diagram; choose a name, e.g., default *[LAB] Structure*

In the logical architecture diagram:

- Add Logical Component(s) and Logical Actor(s)
- Add Functions to components and actors
- Connect Functions with Functional Exchanges

An example is shown in Figure 2.

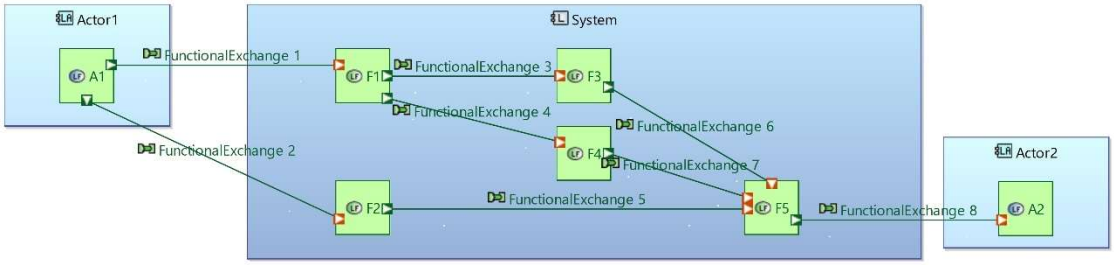


Figure 2 Example Logical Architecture Diagram

Create and fill a Functional Chain view:

- Select all Functional Exchanges as shown in Figure 2, using Ctrl + Left Mouse Button.
- Next click Right Mouse Button > Functional chain > Create Functional chain

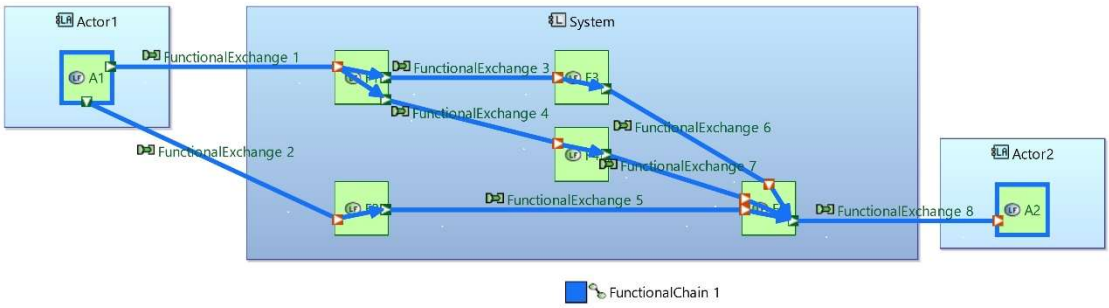


Figure 3 Functional Chain

- Press [Right Mouse Button] on the name of the Functional chain (e.g., FunctionalChain 1), choose New > Functional Chain Description; choose a name, e.g., default *[LFCD] FunctionalChain 1.*
  See Figure 4. Note that warning and errors might appear; they can be resolved later.

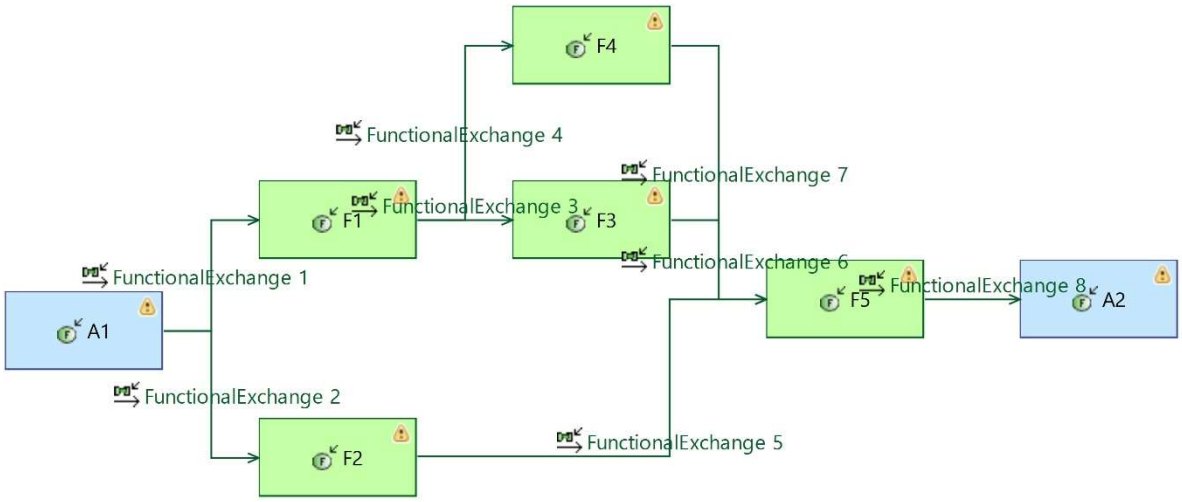


Figure 4 Functional Chain Description

Add control nodes and sequence links between the involved functions:

- First hide the functional exchanges by clicking on drop-down menu of Filters and selecting Hide Functional Chain Involvement Links, see Figure 5.

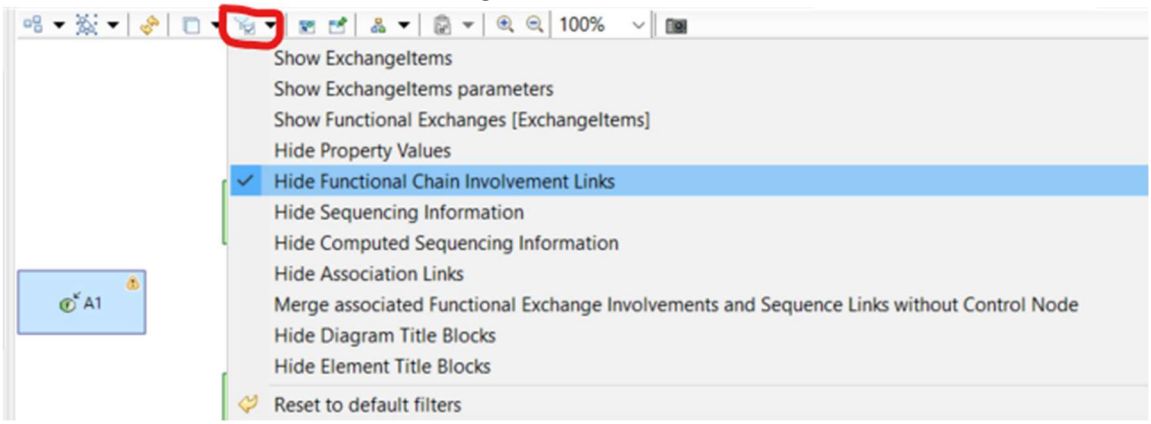


Figure 5 Hide Functional Chains

- To insert control nodes, first click on the small triangle before AND under Sequencing to show AND, OR and IT nodes as shown in Figure 6.

Figure 6 Control Nodes Palette

- Add control nodes and sequence links. An example is shown in Figure 7.

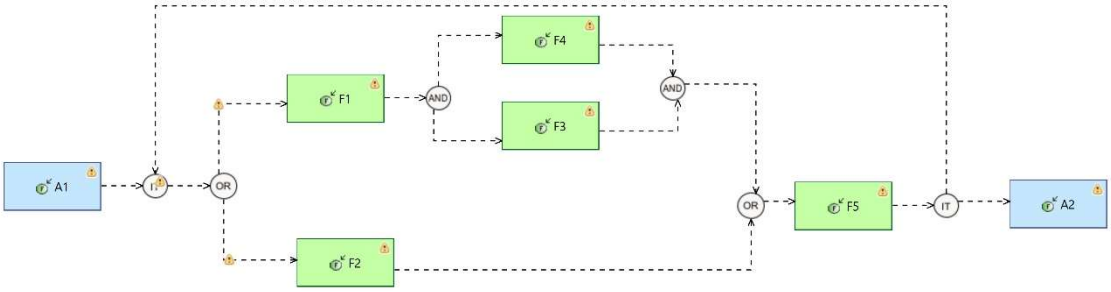


Figure 7 Functional Chain Description with Control Nodes

There are several requirements on functional chain descriptions:

- There should be exactly one start functional node, i.e., a function without incoming links
- There should be exactly one end function node, i.e., a function without outgoing links
- AND, OR, IT nodes occur in pairs, an opening node and a closing node, with the following constraints:
    - An IT opening node has two incoming links (one from the IT closing node) and one outgoing link
    - An IT closing node has one incoming link and two outgoing links (one to IT opening node)
    - The number of outgoing links of an AND opening node equals the number of incoming links of the corresponding AND closing node, similar for OR nodes

Note that the first two requirements forbid that a functional chain starts or ends with a control node. To construct such chains, dummy start or end functions have to be added.

As an example, suppose a functional chain should consist of the sequence < A1, F2 > in parallel with the sequence < F5, A2 >. Then create dummy functions, e.g., Start and End, add a functional exchange of Start with A1, and create a functional chain with this new exchange, see Figure 8.

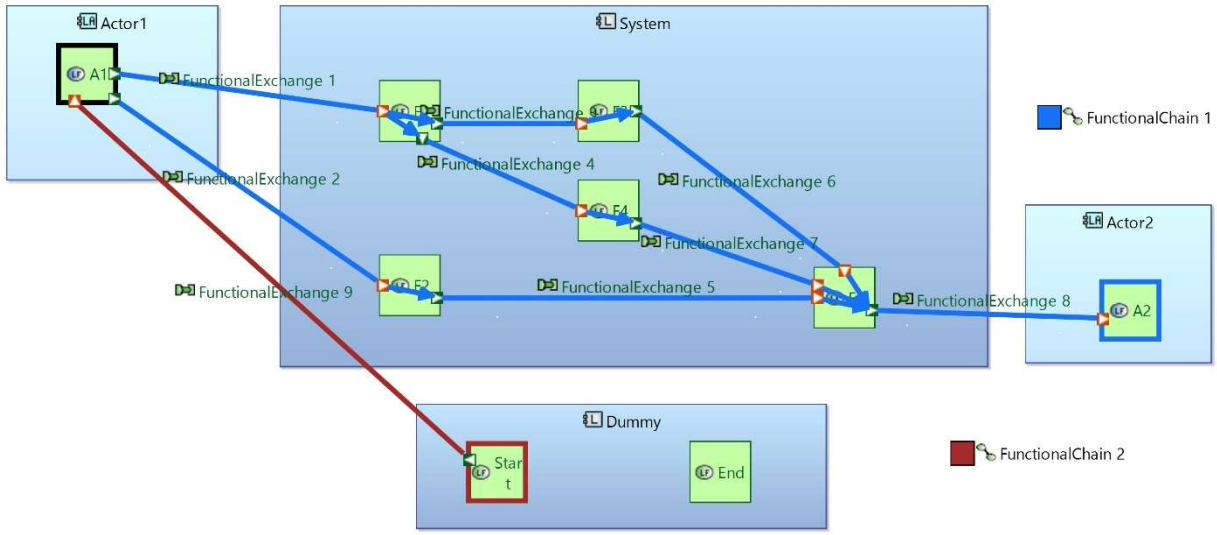


Figure 8 Adding a Dummy Start and End Function

Next create a new Functional Chain Description and Hide Functional Chain Involvement Links, as described above. Next add functions by Function under Involvements and control nodes, leading to Figure 9.

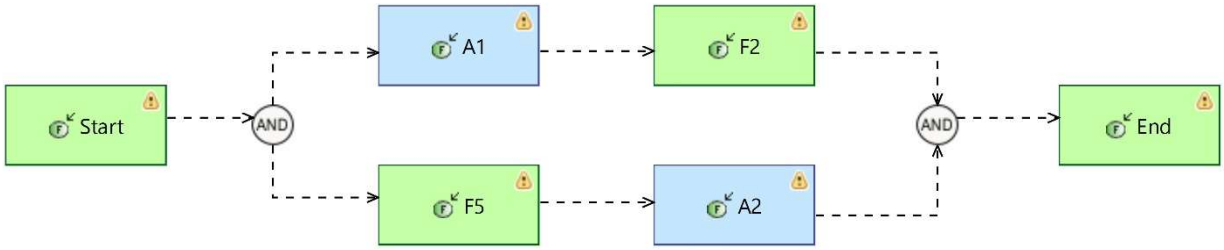


Figure 9 Functional Chain Description with Dummy Start and End Functions

5 ADD PARAMETERS TO NODES AND LINKS

The PVMT profile is defined by a .vpd file. The current profile "WF params.vpd" can be found at https://github.com/TNO/capella-workflow-dse/blob/main/tutorial/WF%20params.vpd .

Load the PVMT profile as follows. Use Property Values view:

- Tab Window > Show View > Other > Choose Property values, select Property values – Open; see Figure 10

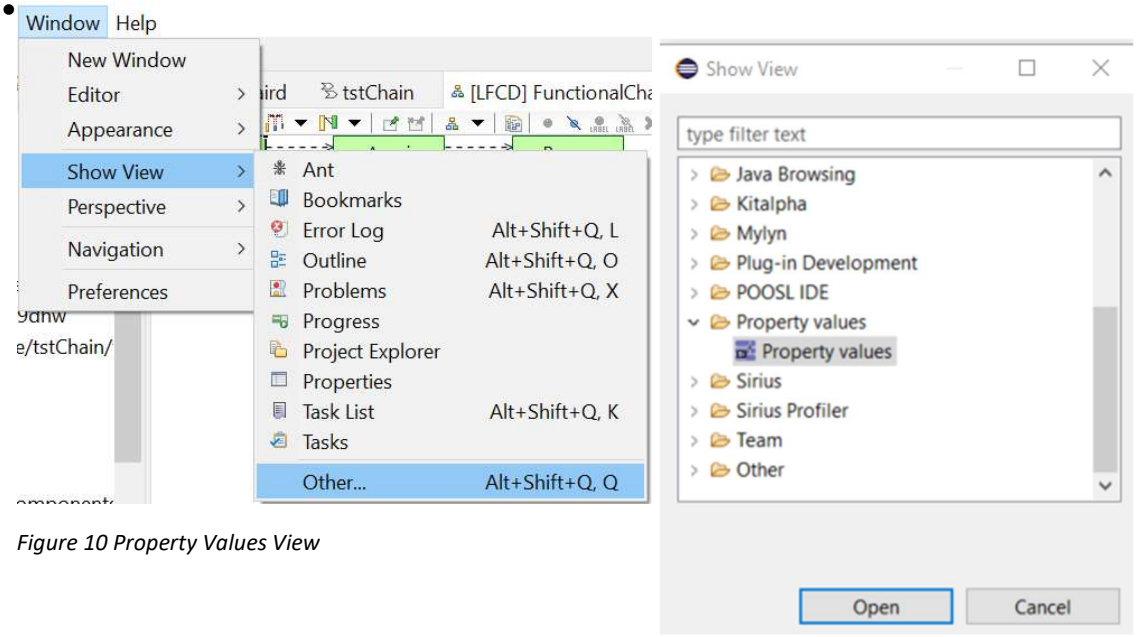


Figure 10 Property Values View

- Select a function, select Property Values tab and click on the small blue rectangle on the right (Open PV Definition Editor) as shown in Figure 11.

.

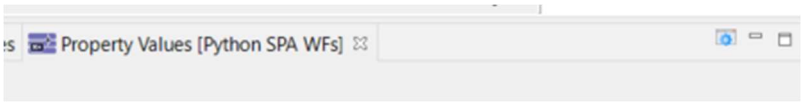


Figure 11 Open PV Definition Editor

- In Configuration PV tab, select on the right Import Property values Definition; Browse File System to locate a ".vpd" file (e.g., "WF params.vpd") and select it. Save the definition.
- Close the Configuration PV tab and observe the parameters in the Property Values tab when clicking in a function (see Figure 12), a control node, or a link in the functional chain description.

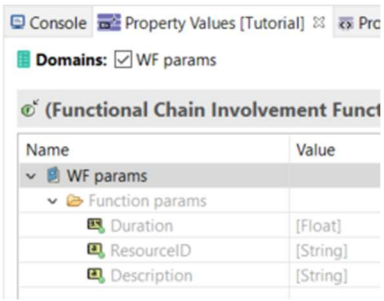


Figure 12 Property Values

Next values can be given to the elements:

- Click on an element (node or link) and provide values in the Property Values tab.
    - Duration of a function requires a Float, do not provide the "s"
    - ResourceID and Description require a string; it is not needed to type quotes
    - For the outgoing links of an OR nodes, the value determines the probability of taken the link, based on de sum of the values of all outgoing links. For other links the value is not used.
    - For IT nodes only the number of repetitions of the start IT node is used. Also, for OR and AND nodes the number of repetitions is not used.
- When a node has at least one value for a parameter, the values can be visualized:
    - In the Palette, under Common, select Applied Property Value Groups
    - Next select an element of the diagram, select >> in the pop-up window, and click OK.
    - Note that for links and control nodes there is no dashed line between the parameters block and the element.

  An example is shown in Figure 13.

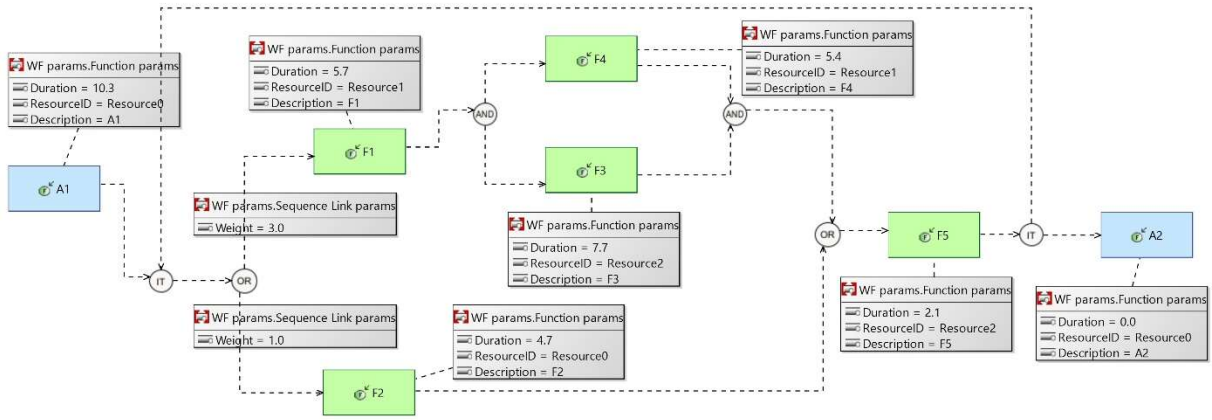


Figure 13 Functional Chain Description with parameter values

The Problems tab shows errors and warnings. If it is not present, select

Window > Show View > General > Problems

To avoid a large list with all errors and warnings in the workspace, select three vertical dots on the right in Problems tab, select Show > Errors/Warnings on Selection. Note that the warnings also indicate the default values for parameters – double clicking on the warning selects the element in the diagram. As an example, see Figure 14.

Console | Property Values [Tutorial] | Problems | Properties | Information | Semantic Browser

1 error, 13 warnings, 0 others

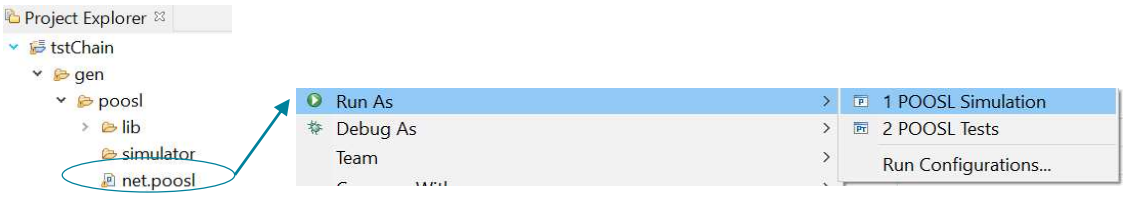
Description	Resource	P..	Location	Type
Sirius diagram editor Plugin problems (14 items)				
No or multiple end functions found, exactly 1 function with no outgoing links should exist	Tutorial.aird	/...	FunctionalChain 1	Sirius diagram ...
Function 'A1' defines no 'Duration', using default of 1.0	Tutorial.aird	/...	FunctionalChain 1	Sirius diagram ...
Function 'A1' defines no 'Duration', using default of 1.0	Tutorial.aird	/...	FunctionalChain 2	Sirius diagram ...
Function 'A2' defines no 'Duration', using default of 1.0	Tutorial.aird	/...	FunctionalChain 1	Sirius diagram ...

Figure 14 Warnings

6 SIMULATION OF WORKFLOW

The functional chain can be simulated using a POOSL model [5]:

- Right click in the diagram of a functional chain description or on the name of a file under the .aird file > Representations per category > Common > Functional Chain Description; select Workflow DSE: Export
  An alternative is to right click in the diagram.
- In case no error is detected, this leads to a pop-up message “Export finished”.
- The generated POOSL model *net.poosl* can be found in folder gen > poosl
- Execute the simulation by right-clicking on net.poosl and select
  Run As > POOSL Simulation



A pop-up will appear concerning the character setting, see Figure 15.

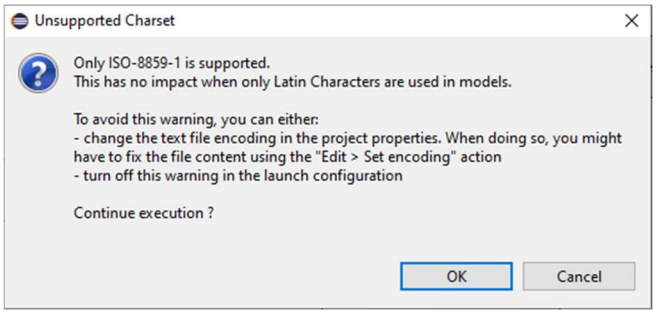


Figure 15 Pop-up about Character Setting

Click OK; the Console tab will show information about the simulation.

7 VISUALIZATION AND ANALYSIS OF SIMULATION RESULTS

Running a simulation as described in the previous section will generate file trace.etf in the gen > poosl folder. This generated file can be opened in the TRACE4CPS Gantt chart viewer by double clicking on the view. The default view of simulating the functional chain in Figure 13 is shown in Figure 16. This Gantt show the execution of the functions (vertical axis) in the functional chain over time (horizontal axis).

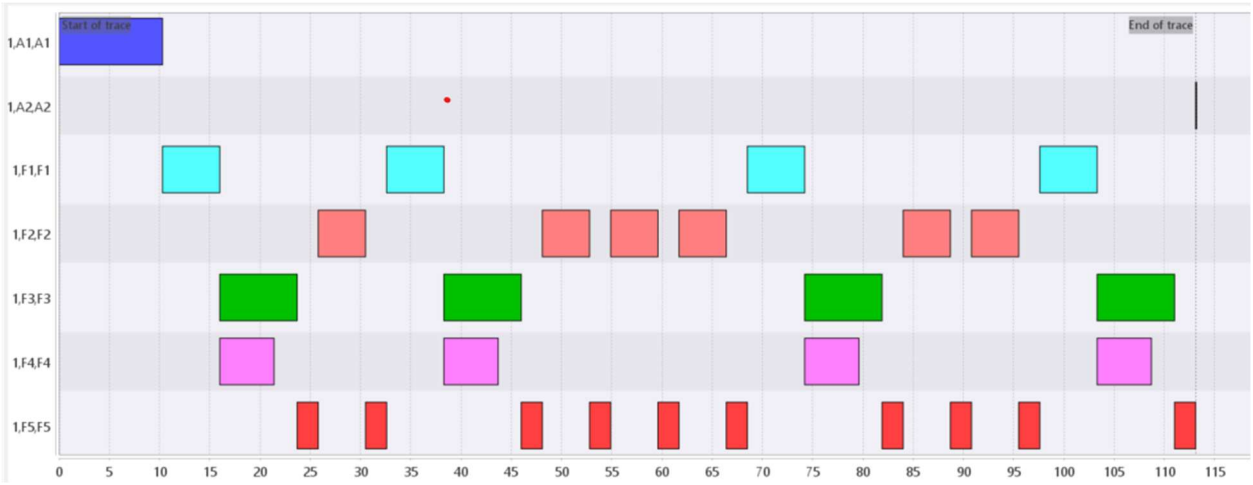


Figure 16 TRACE4CPS Gantt chart activity view

To deal with nested functional chains, as discussed in Section 8, there are a few attributed on the vertical axis. They can be removed by the claim grouping button  and selecting “name”. Note that in TRACE4CPS function execution is represented as a *claim* on a resource.

Using TRACE4CPS one can toggle between two views, an activity view (see Figure 16) and a resource view (see Figure 17). One can toggle between these views by pressing the  button. In the resource view, the grouping is done according to the resource of a function. I.e., all functions that use the same resource are in the same swim lane. In the activity view, the grouping of the functions is done according to the grouping attributes that will be explained later when dealing with nested functional chains.

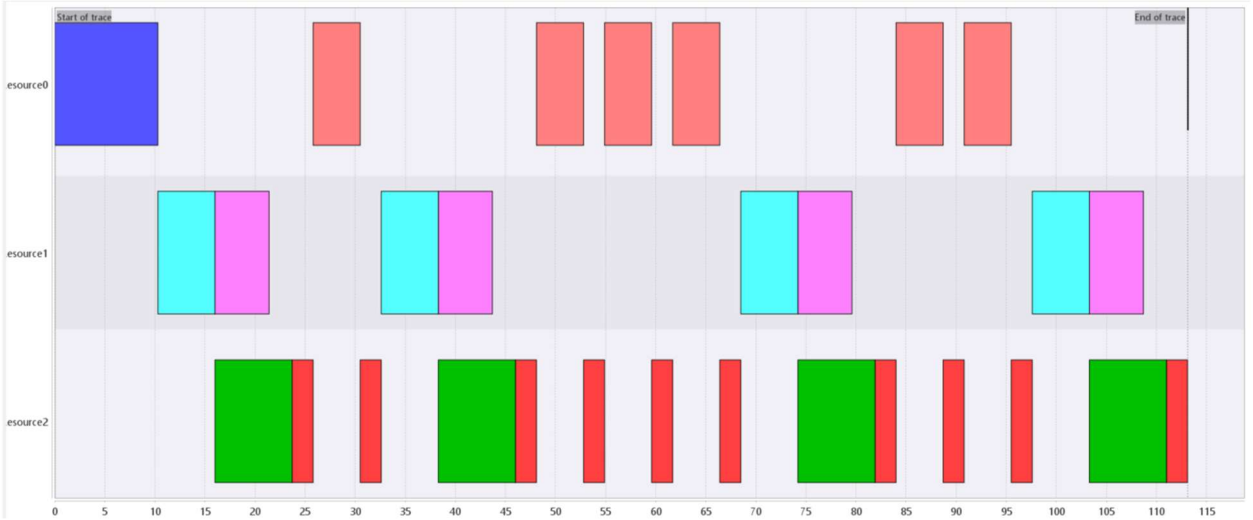


Figure 17 TRACE4CPS Gantt chart resource view

TRACE4CPS offers a large variety of analysis tools. One relevant analysis is critical path analysis, which can be used to identify the bottlenecks of a workflow. Critical path analysis can be started by pressing the button. This will open the dialog shown in Figure 18; the *NONE* option must be selected as well as the option *Use default epsilon*.

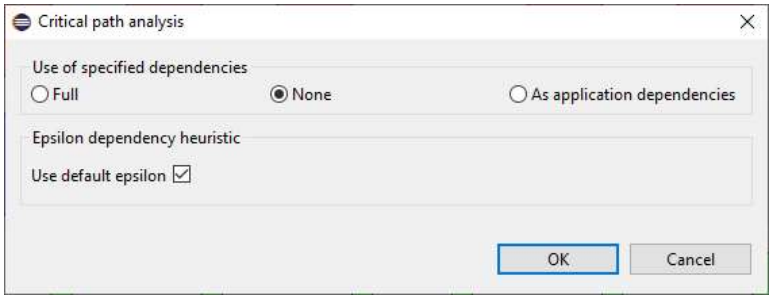


Figure 18 TRACE critical path analysis dialog

After selecting *OK*, a new Gantt chart view will be shown (see Figure 19). In the new view, functions are colored red if they lie on the critical path. If the length of the execution is to be shortened, one needs to address the functions on the critical path.

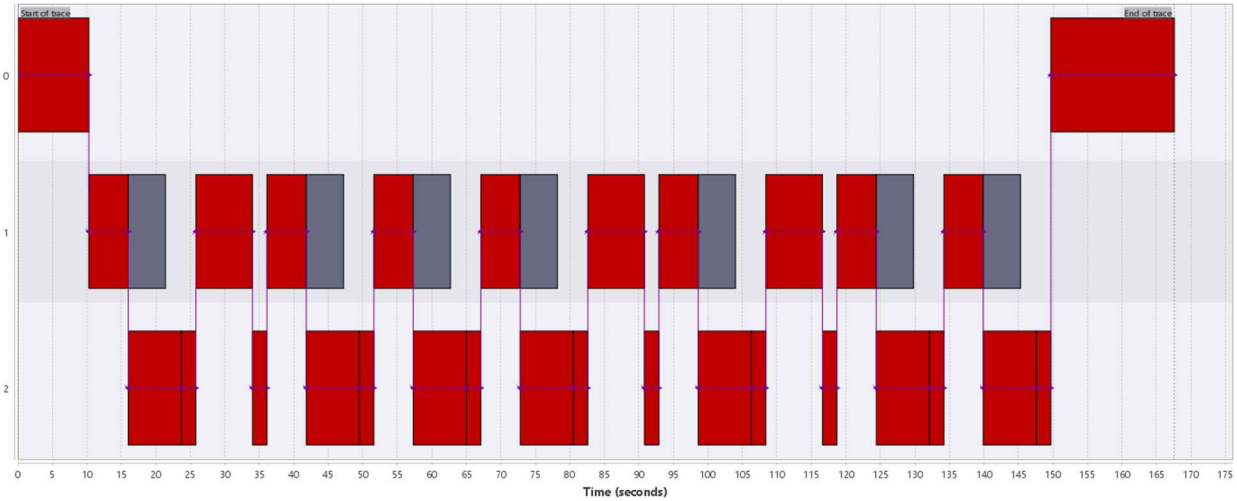


Figure 19 TRACE4CPS Gantt chart with critical path view

The critical path view cannot be stored. If the Gantt chart is closed and opened again, the default (or persisted) view with be shown.

To inspect large workflows, TRACE4CPS offers zooming, panning and scrolling functionality. Zooming can be done in various ways:

- Right-click the chart: the context menu has items to zoom one or both of the axes.
- Ctrl-down and mouse wheel zooms the horizontal axis in or out.
- Shift-Ctrl-down and mouse wheel zooms the vertical axis in or out.

- Area selection in the chart using the mouse zooms to that selection.
- A mouse dragging gesture to the left zooms out fully on both axes.

Panning is done by Alt-down and dragging the chart. The mouse wheel scrolls on the vertical axis if a vertical scrollbar is present; otherwise, it scrolls on the horizontal axis (time). The arrows on the scroll bars can be used for scrolling in smaller steps.

- By using TRACE4CPS's filtering ( ) function, one can hide functions from the Gantt chart. This can be done based on the attributes of the functions or the resources. After selecting *Filter claim* and selecting the *name* attribute and *Next*, one can select the functions to keep. After ticking the checkboxes of *F1*, *F2*, *F3*, *F4*, *F5*, selecting *Finish* and choosing the resource view, the Gantt chart in Figure 20 is shown. Resource filtering can be done in a similar manner. Filters can be removed using the function.

After filtering, one can persist the resulting view using the function. When opening the Gantt chart a next time, the persisted view will be shown.

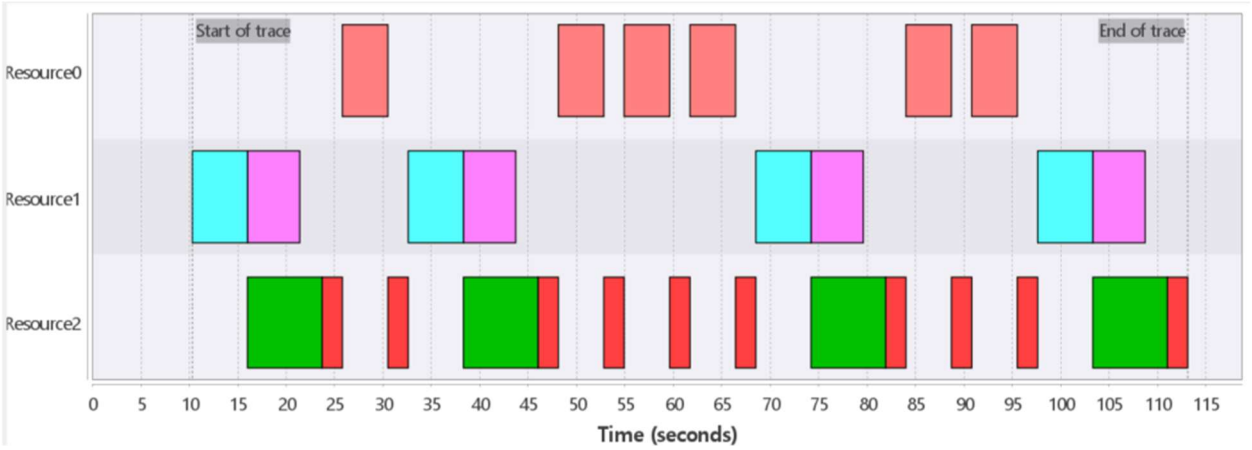
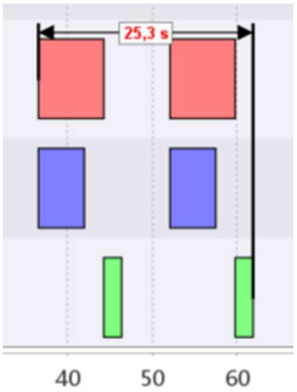


Figure 20 TRACE4CPS Gantt chart view after filtering

The distance between points in the Gantt chart can be measured by clicking *Alt* and then selecting two points in the chart, see the figure on the right. To remove the timing annotation, re-open the chart.



Information on other analyses that TRACE4CPS offers can be found in the documentation, which can be accessed via the *Help > Help Contents* menu and selecting the *TRACE4CPS user manual* section in the contents window.

8 NESTED FUNCTIONAL CHAINS

Create a new Logical Architecture diagram; if needed, first open the Activity Explorer (right click on .aird file), select Logical Architecture, Define Logical Components and Actors.

Create a number of new basic functions and add functional exchanges, see component Basic Functions in Figure 21. Also insert the Dummy component (which was created in Section 3), using Components (Components) under Components in the Palette and using ">". Next insert the End function, using Allocated Functions (Allocated Functions) under Functions in the Palette and using ">". Add a functional exchange to obtain the diagram of Figure 21.

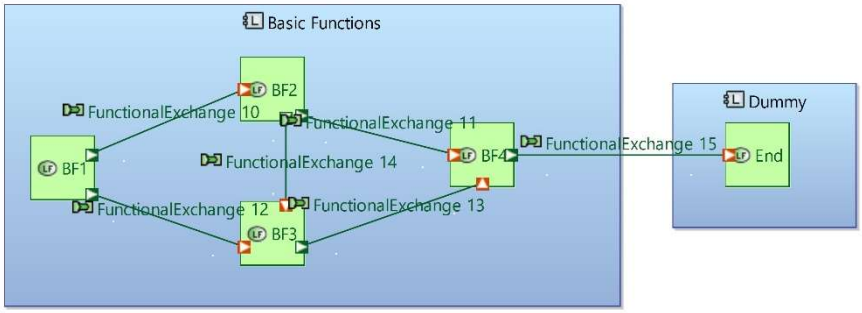
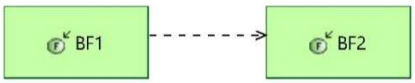


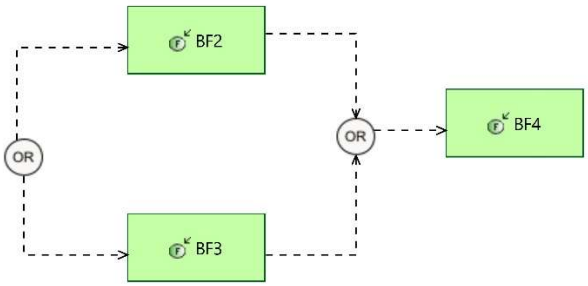
Figure 21 Basic Functions

Create functional chain 3 as shown in the next figure:



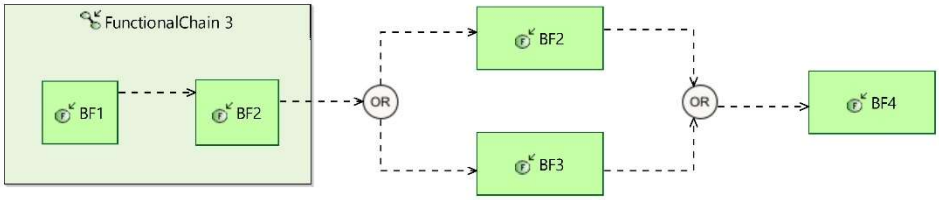
Note that the diagrams in this section often do not show warnings because values for the required PVMT attributes have been given

Create the following functional chain 4:

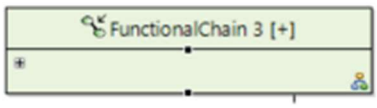


Next functional chain 3 is included in chain 4 by selecting Functional Chain under Involvements in the Palette, click in the diagram, in the pop-up select FunctionalChain2 and click OK. See the Next picture

Enlarge the box of FunctionalChain 3 to see its contents and connect BF2 to the OR node, as shown below:



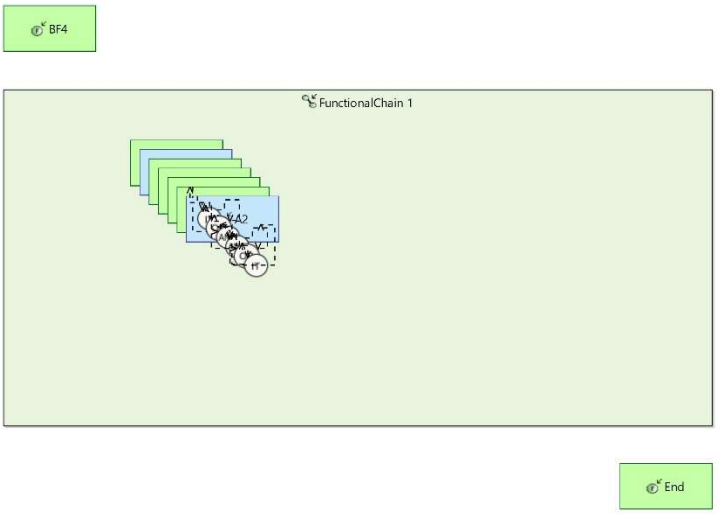
When selection FunctionalChain 3 there is a small icon in the upper left corner of the main box (see the picture below); clicking on it gives a collapsed view. It can be undone by an icon at the same place.



Note that this chain can be simulated and in the activity view of the trace it is possible to group functions based on the level. This will be explained in more detail below.

Next, we create a functional chain based on other chains. Create function chain description 5, based on the exchange between BF4 and End:

Hide the functional chain involvement links, move the End function to the lower part of the diagram, and insert FunctionalChain1 as explained above. Enlarge the box of FunctionalChain1.



Open the original diagram of FunctionalChain 1 (see Figure 13), select all elements (e.g., by Ctrl-A) and choose “Copy the format of the selected diagram elements” on menu bar:

To paste the current recorded layout, return to FunctionalChain 5, and select Paste format from menu bar:

Move the End function further down and insert FunctionalChain 4. Also here copy the format of the original diagram. Finally add sequence links and an iteration as shown in Figure 22.

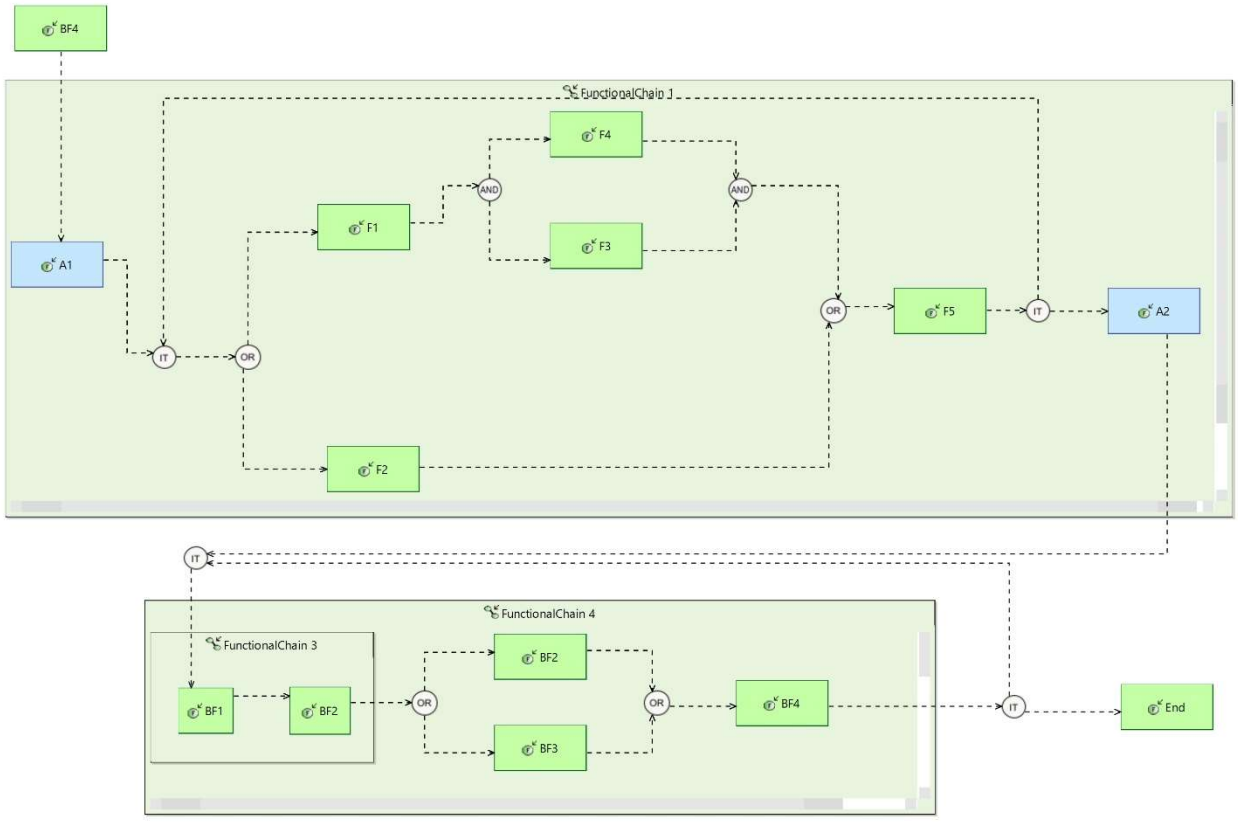


Figure 22 Nested Functional Chains

Next the model can be simulated and the resulting trace can be viewed at different levels. A few possibilities:

- By filtering claims (i.e., function executions) on name, everything except the End function can be selected to hide the artificial End.
- In the activity view, claim grouping can be used to show the activities at different levels, e.g.
  - Level1name shows BF4, chain 1 and chain 4
  - Level2name shows level 1, all functions of level 2 (A1, .. BF2, .., F5) and chain 3.
  - Level3name shows level 1, level 2, and all functions of level 3 (BF1 and BF2)
- Claim coloring provides possibilities to give the same color to functions of the same level

9 REUSE OF FUNCTIONAL CHAINS

In this section we demonstrate how to reuse functional chains, since this is slightly more complicated than the reuse of functions which was shown in FunctionalChain4 (see Figure 22). In addition, we also present and alternative way to define functional chains, namely, the use of function decompositions to model hierarchical flows.

Next create a new architectural diagram:

- Choose Logical Architecture > Refine Logical Functions, describe Functional Exchanges > [LFBD] Create a new Functional Breakdown diagram; choose a name, e.g., default *[LFBD] Root Logical Function*
- This shows all existing functions, hide them by selecting all and use right-click > Show/Hide > Hide element
- Add functions and Contained In relations to obtain Figure 23

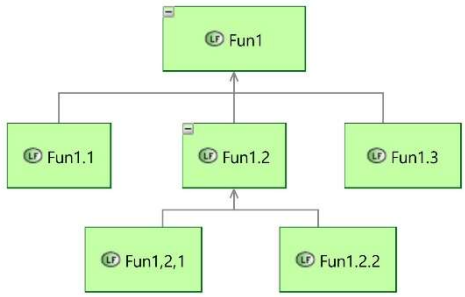
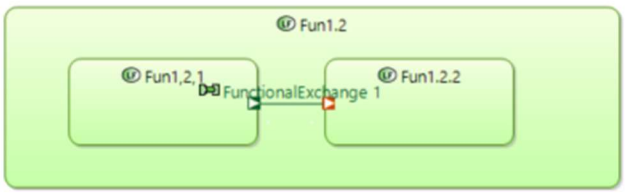


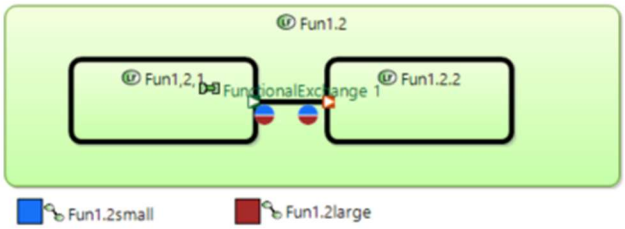
Figure 23 Hierarchy of functions

Suppose we want to create a functional chain which contains two version of Fun1.2 with different parameters, say Fun1.2small and Fun1.2large with small and large parameters respectively. This can be achieved as follows:

- Choose Logical Architecture > Refine Logical Functions, describe Functional Exchanges > [LDFB] Create a new Functional Dataflow Blank diagram; choose a name, e.g. *[LFBD] Fun1.2*
- Add existing function Fun1.2 and its sub functions. Also add a functional exchange from Fun1.2.1 to Fun1.2.2.



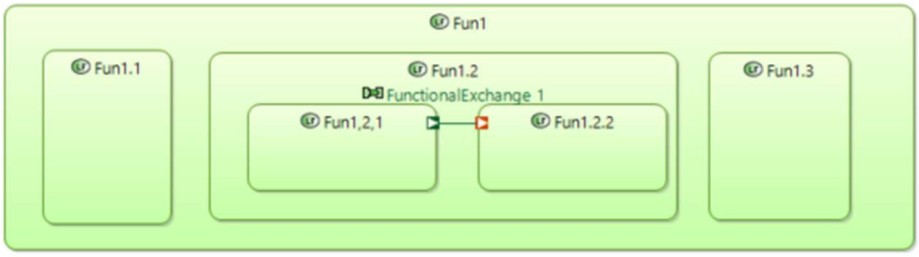
- Select the functional exchange, right-click and select Functional Chain > Create a Functional Chain. Change the name to Fun1.2small.
- Repeat the step and create another functional chain (with the same functional exchange) and call it Fun1.2large.



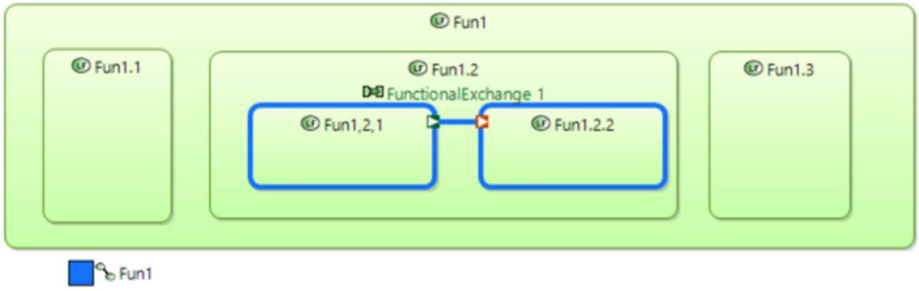
- For each chain, right-click on it, select New > Functional Chain Description; use the default names and add a sequence link from Fun1.2.1 to Fun1.2.2. Also, PVMT values can be added.

To define the intended workflow, proceed as follows:

- Create a new Functional Dataflow Blank diagram; choose a name, e.g. *[LFBD] Fun1*
- Add Fun1, all its sub functions, as shown below.



- Select the functional exchange and create a functional chain. Change the name of the chain to Fun1.



- Based on this chain, create a new Functional Chain Description Fun1.
- In the functional chain description, hide the two functions, add Fun1.3, two instances of Fun1.1, and insert the two functional chains Fun1.2small and Fun1.2large. Add sequence links to obtain Figure 24.

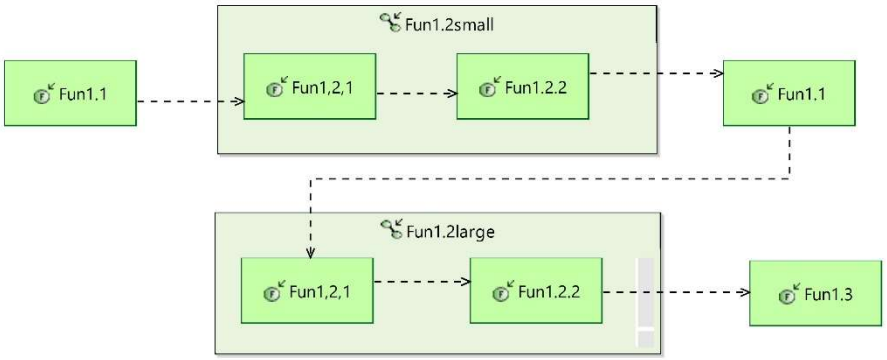
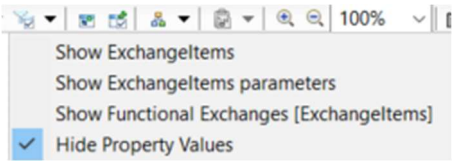


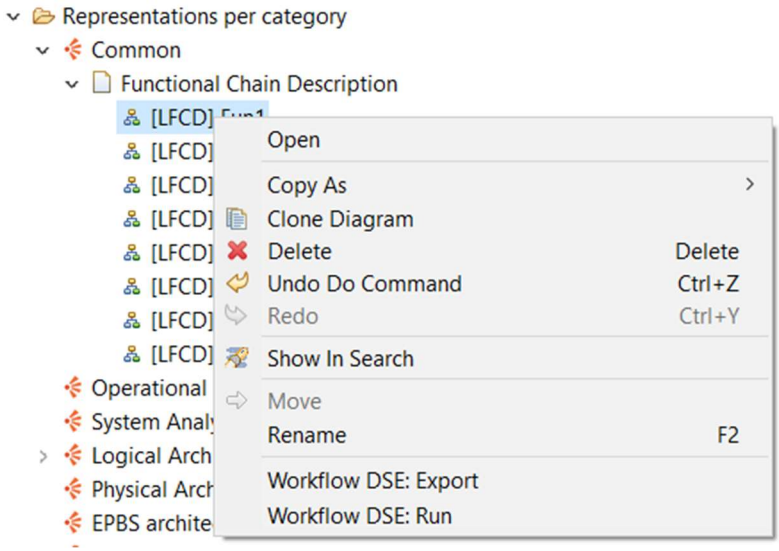
Figure 24 Workflow with reuse of functional chains

10 DESIGN SPACE EXPLORATION

To illustrate design space exploration, consider FunctionalChain1 as shown in see Figure 13. Use of the menu bar to hide the property values, see the next figure:



To start a separate design space exploration window, right-click in the diagram or on the file name in the project explorer and select Workflow DSE: Run, see the next figure.



The result is shown in Figure 25 (note that the lay-out of the chain has been adapted).

- By clicking on a function, the value specification on the left will be selected where ranges to be explored can be specified:
  - By clicking on the circle with the dot after the Duration ⊙ a minimum and maximum value and a steps size can be given.
  - By means of the + after ResourceID, alternative resources can be specified.
- Similarly, values for IT start nodes and outgoing arcs of OR start nodes can be defined.
- Note that it is also possible to click on the PVMT value specifications on the left to highlight the corresponding node in the diagram.
- The relation between resources and the duration of functions on these resources can be specified explicitly by means of button ☰. This leads to a pop-up window where these relations can be expressed. An example is shown in Figure 26. Lines can be added using + and the setting can be saved by button 💾 to file DSE-definitions.json.
- Values defined in the editor of Figure 26 overwrite values specified elsewhere. When no value is specified in the DSE tool, the PVMT values defined in Capella are used (see Section 5).

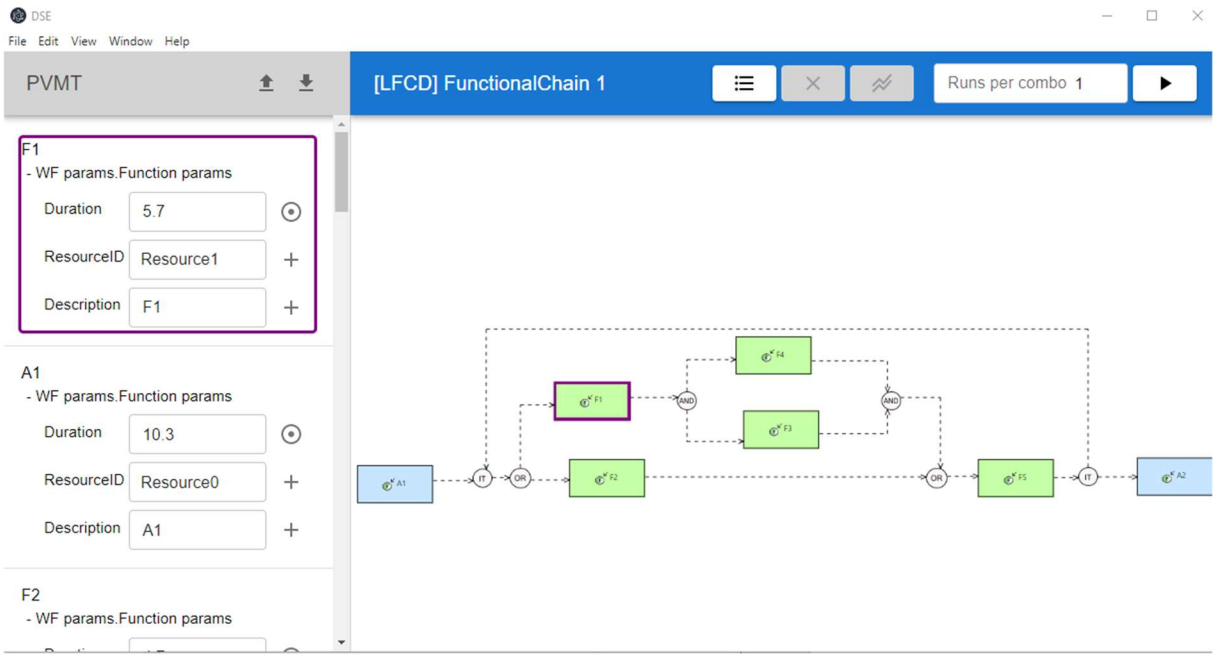


Figure 25 Design space exploration

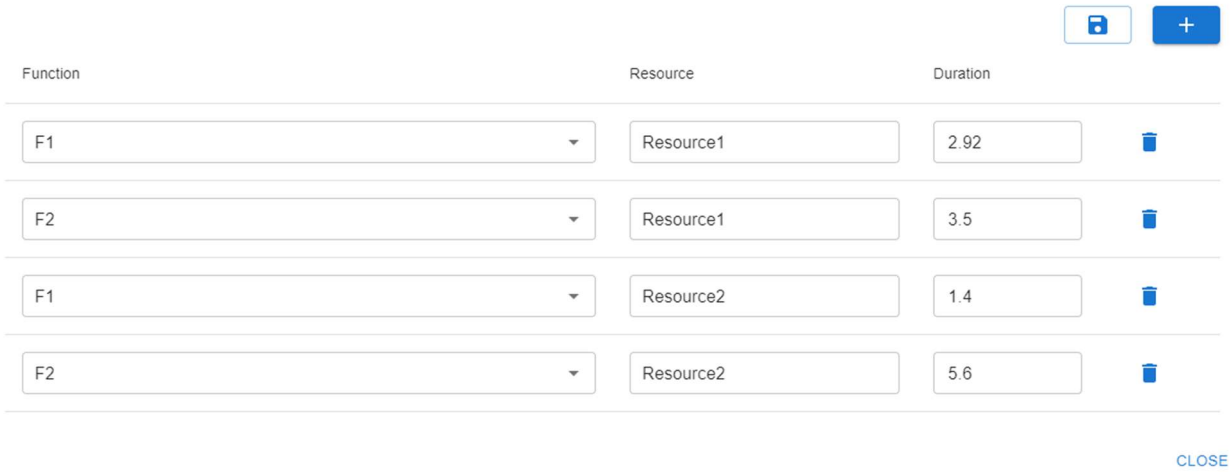


Figure 26 Definition editor

- All possible combinations of the PVMT values are simulated by the play button ▶. By the drop down button to the right of the play button, there is a choice between running a Python simulation or a POOSL simulation. By default, the faster Python simulation is used.
- When the diagram contains OR nodes, it might be useful to run multiple simulations for each combination of values (“combo” for short); this can be specified using the “Runs per combo” button. Them the average duration of all these runs will be taken.
  Note that this button is disabled when there are no OR nodes in the diagram.

- Details of all runs can be found in folder gen > dse. Previous can be deleted by button ✕.
- Button  shows a visualization of the runs; a small example is shown in Figure 27. In this case, no costs are specified, meaning costs are 0 by default.
  On a vertical axis a region can be selected to highlight the runs related to that region. For instance, in **Error! Reference source not found.** the region with the two lowest durations is selected. Note that each vertical axis can be moved manually left or right.

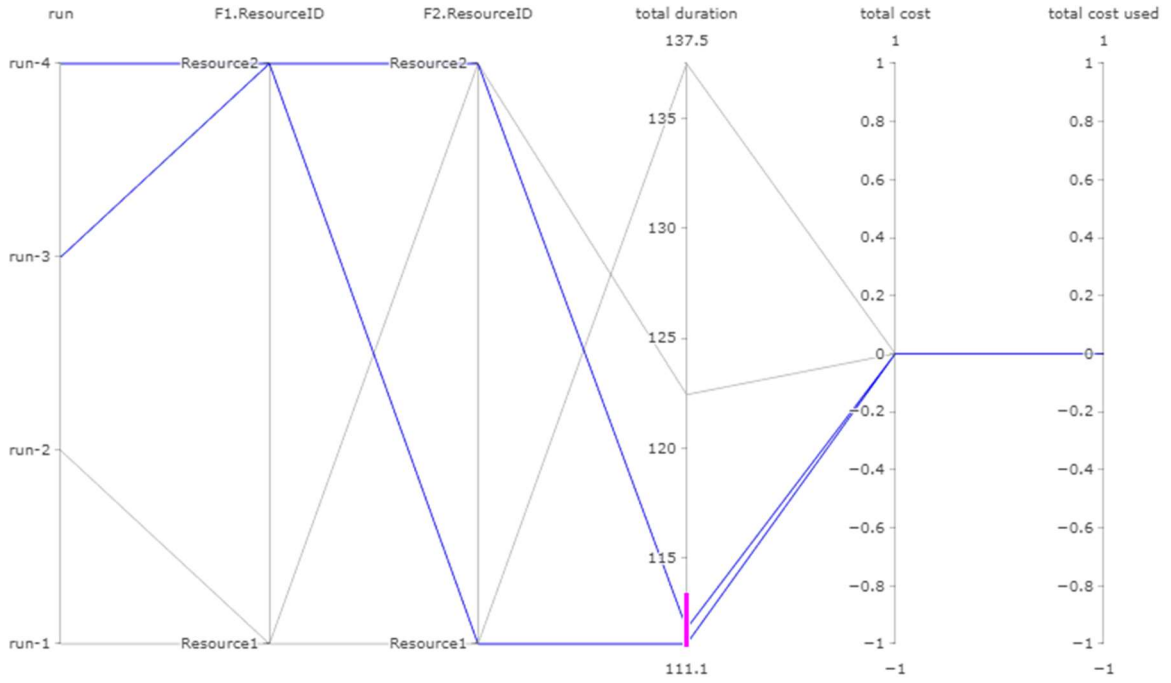


Figure 27 Visualization of results, no costs specified

The able to explore the trade-off between the duration of a workflow and the cost of resources, the definition editor also allows the specification of the costs of resources as shown in Figure 28.

The visualization of the results also shows the total costs of all resources in the chain and the costs of the resources that are actually used in the chain, see Figure 29. To investigate the trade-off between duration and costs, use the  button in the results view to see a Pareto efficiency plot. An example is shown in Figure 30, where the green line connect optimal runs in the sense that other runs have a worse performance for the same costs. Hover over the points to see the data and the number of the run. Details of runs can be found in the project explorer, folder gen > dse.

Figure 28 Editor of resource costs

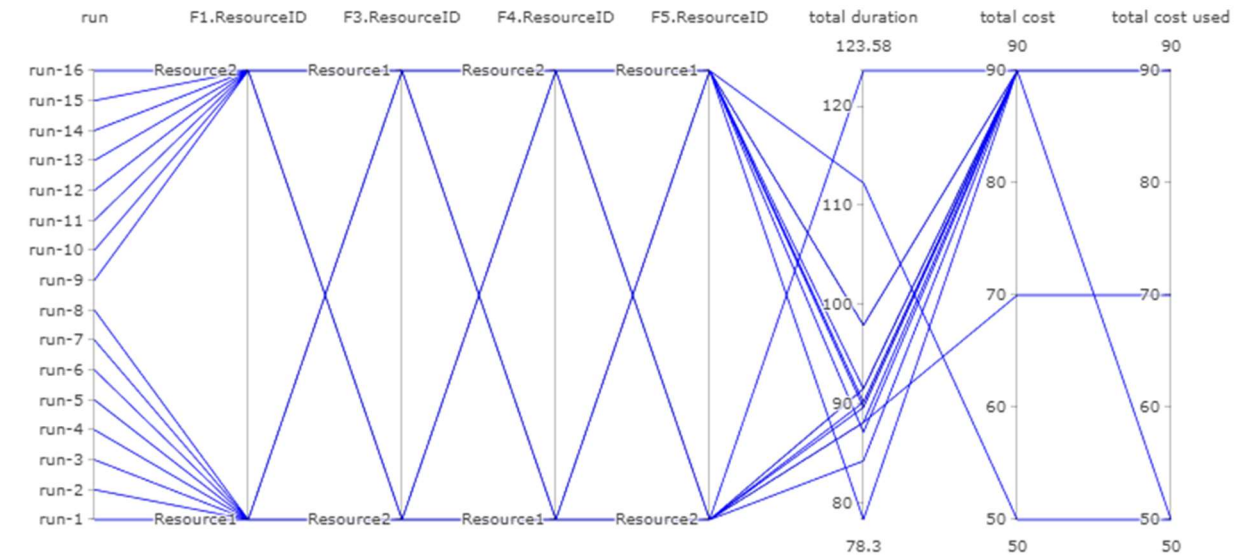


Figure 29 Visualization of results with costs

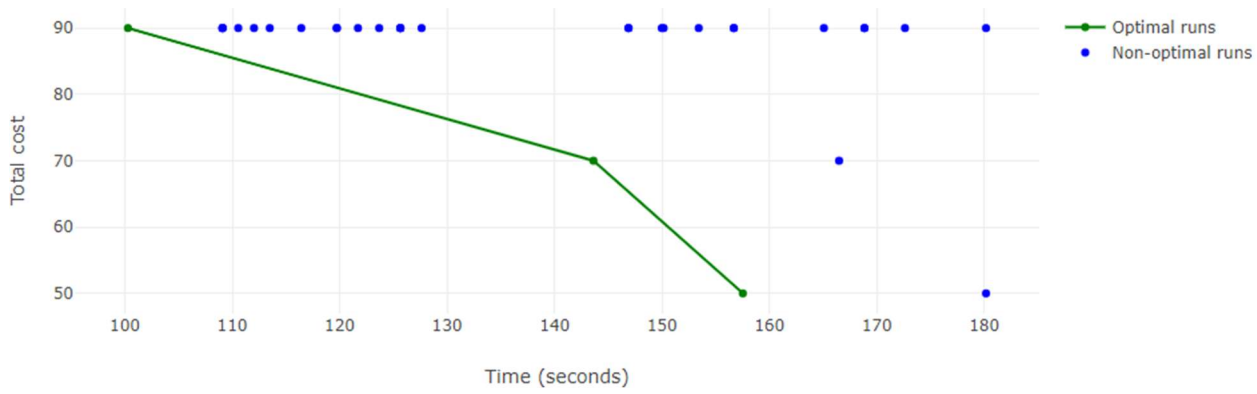
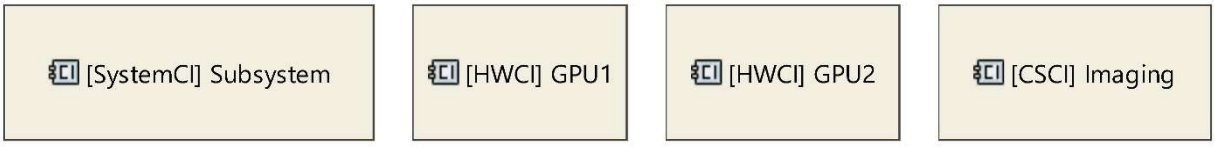


Figure 30 Pareto efficiency plot

11 USING EPBS CONFIGURATION ITEMS

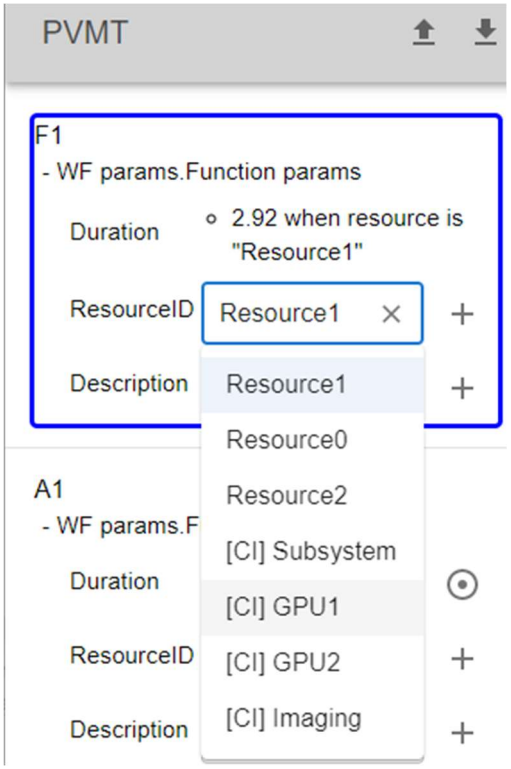
Configuration Items (CIs) that have been defined in Capella's EPBS (End-Product Breakdown Structure) perspective can be used as resources in the design space exploration. To explore how this can be done, first create a few configuration items as follows:

- Open the Activity Explorer by right-clicking on the .aird file
- Click on the EPBS step, select Define CI Components and click on [CIBD] Create a new Configuration Breakdown diagram. Choose a name, for instance, "Configuration Items".
- Create a few items, for instance as shown in the next figure:



Right-click on a functional chain, for instance [LFCD] FunctionalChain1, and select Workflow DSE: Run.

Note that it is now possible to use the configuration items as resources, as shown in the next figure:



The configuration items can also be used in the definition editor.

12 PVMT

12.1 Import and export of PVMT values

Export PVMT values:

- Open project by double-clicking on .aird file
- Right-click on the .aird file and select Export PVMT values

Similarly, PVMT values can be imported.

12.2 Change PVMT profile

To change a PVMT profile:

- Open PV Definition Editor, see figure below

- To update an existing profile, first make sure the existing PVMT values have been exported such that they can be changed and imported later. Next delete it (right-click and select Delete) and save.
- Select Import Property values Definition, see next figure

- Browse file system to select a .vpd file, Open and select OK; Save

12.3 Deleting PVMT values

Note that trying to delete a parameter value in the Property Values tab will give a null pointer exception. A solution is to delete the complete PVMT object connected to a node as follows:

- In the diagram click fn-F8, this shows the details of the diagram in the Project Explorer
- Locate the PVMT params of the function and right-click Delete, OK

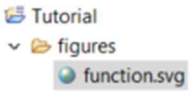
13 Other topics

13.1.1 Open existing project

- In an existing project, double click on the .aird file
- To open the Capella Workflow, right-click on .aird file and select Open Activity Explorer

13.1.2 Adding pictures

To add a picture to a diagram:



- Add the picture to the project, e.g., in folder *figures:*
- Add a Constraint to the diagram (see Common of the Palette)
- Connect the constraint to an element of the diagram, e.g., a function, using ConstraintElement of Common, as shown below.

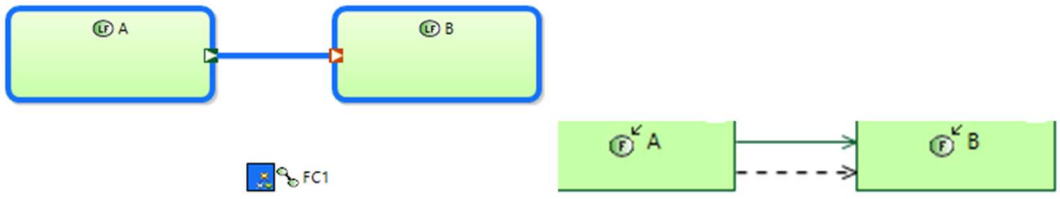
- Select the constraint and in the menu above the diagram select "Set style to workspace image", as shown in the next figure:

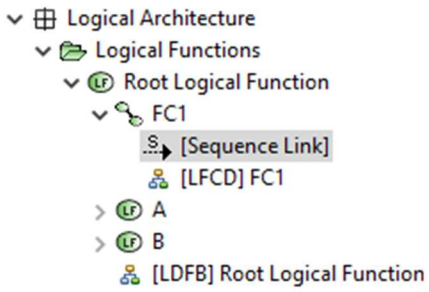
Select the image and click OK; then the constraint is replaced by the figure. Similarly, also function blocks can be replaced by pictures.

13.1.3 Deleting a function in a functional chain does remove a sequence link

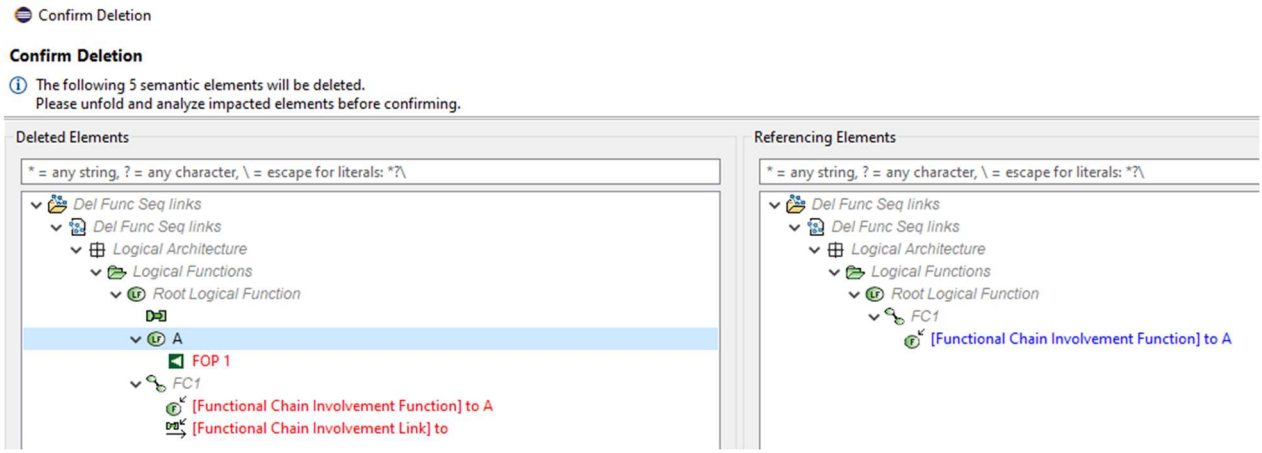
Note that deleting a function in a functional chain does not lead to the removal of a sequence link to/from the functional chain involvement function. Such a sequence link is not used in the simulation since it has no source or target. To re-use the link in the simulation, the user needs to re-specify the link's source or target.

As an example, consider functional chain FC1 which includes 2 functions, 1 functional exchange, and 1 sequence link.

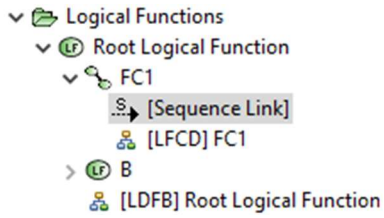




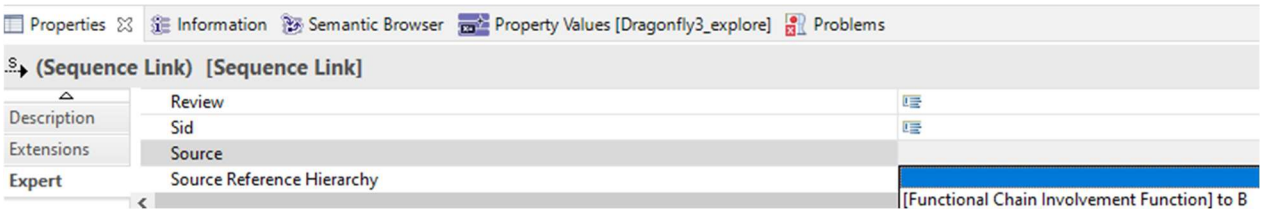
Deleting logical function A causes the ‘Confirm Deletion’ window below. Note the absence of the sequence link in the list ‘Delete Elements’.



As a result, the sequence link is not deleted.



Note that the link’s source can be respecified in:



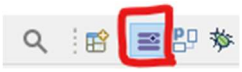
13.1.4 Simulation with POOSL

A few details about the simulation based on POOSL [5]:

- It is possible to determine the way in which POOSL resolves non-determinism, e.g., when selecting which function to execute after an AND or an OR construct:
  - Select Run Configurations in the drop-down menu right of the Run button 
  - Under POOSL Simulation, select a model and then the “Seed for resolving non-determinism” can be changed; either choose a fixed seed or select Random
  - See also Help > Help Contents > Poosl > Simulate > Running Models, see Advances features > Setting a seed for resolving non-determinism

- Details of a simulation, with the possibility to perform steps and inspect details of the POOSL, can see when running the simulation in Debug mode:
    - Right-click on the POOSL file and select Debug As > POOSL Simulation
    - For more details about debug mode, see Help > Help Contents > Poosl > Simulate > Debugging Models

13.1.5 Known issues

- After resolving a warning, the diagram might still show the yellow triangle on an element without warnings; slight move the element to remove the yellow triangle.
- TRACE4CPS: if the Gantt chart is not completely visible, e.g., the timeline is not visible, resize the window.
- When certain Papillon menu items are not available, make sure the Capella perspective is used, see the upper right corner: 
- The DSE application does not check constraints on the models. When the sum of the weights of outgoing arcs of an OR node is not positive, the simulation will block and has to be stopped manually.

14 REFERENCES

[1] Design space exploration of customer workflows in Capella, https://github.com/TNO/capella-workflow-dse
[2] Arcadia https://www.eclipse.org/capella/arcadia.html
[3] Capella https://www.eclipse.org/capella/
[4] Property Values Management Tools (PVMT); https://www.eclipse.org/capella/addons.html
[5] POOSL https://www.poosl.org/
[6] TRACE4CPS https://projects.eclipse.org/projects/technology.trace4cps