User Guide
for the STPA Add-On to Capella

Table of Contents

I. Foreword

STPA (Systems-Theoretic Process Analysis) is a risk analysis method [1][2][3]. Although it originally focuses on Safety, it can be applied to other concerns as long as they involve the notion of *control* in a broad sense, such as Cybersecurity or Performance.

The STPA Add-On is an *experimental* extension of Capella that provides model-based tool support for STPA. It can be used for standalone STPA analyses or in combination with classical Capella modelling.

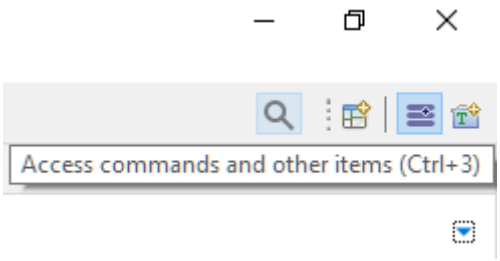
II. Activate STPA Analysis on a Capella project

As a prerequisite, it is assumed that the STPA add-on has been successfully installed in Capella. This can be easily tested: if this prerequisite is not met, the steps described in this section will fail.

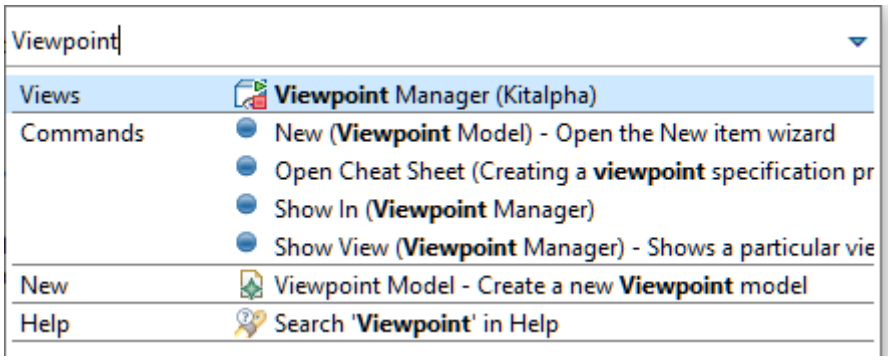
STPA-related functionality is classically available as a 'Capella viewpoint'. As such, it has to be activated once for every Capella project concerned. The activation procedure is the standard Capella viewpoint activation procedure.

First, create a new Capella project, or open an existing one.

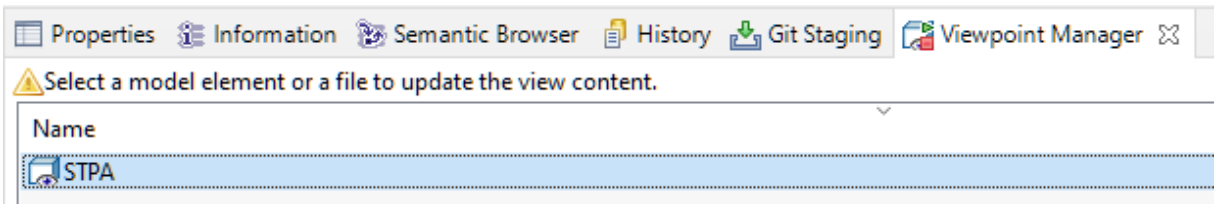
Click the button with the magnifier icon at the top right-hand corner.



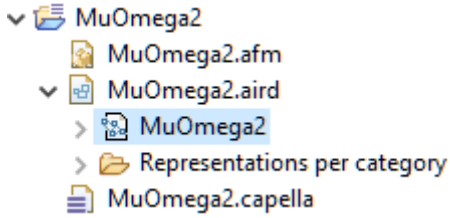
Type 'viewpoint' to see the 'Viewpoint Manager' entry and select it.



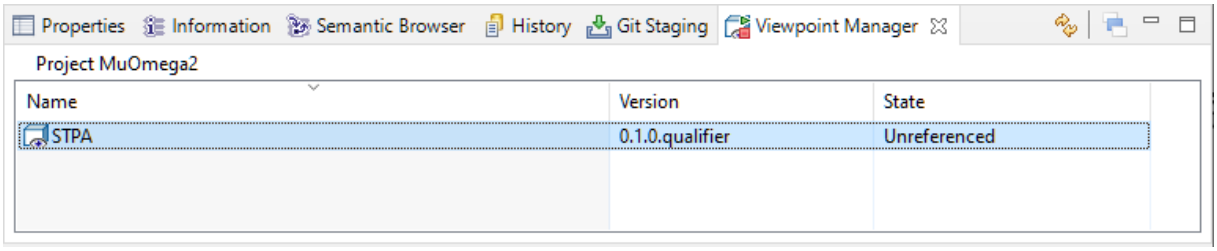
The Viewpoint Manager 'view' (sub-window) shows up at the bottom of the Capella window.



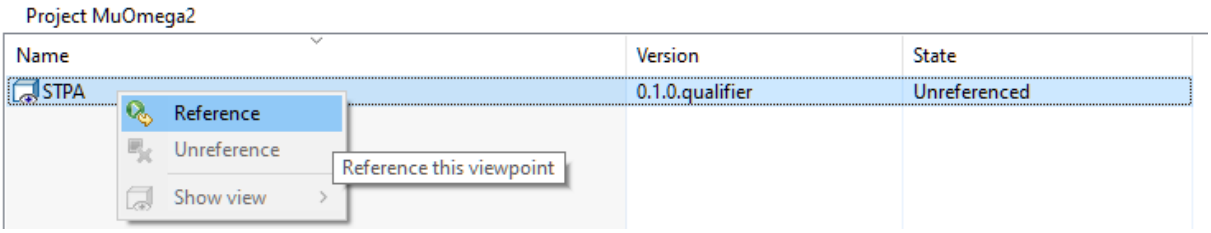
Now select a model element of the concerned project in the Project Explorer.



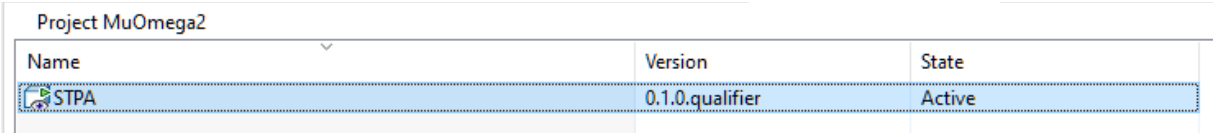
The warning in the Viewpoint Manager disappears.



Right-click the STPA row and select Reference. If the Reference menu item is greyed out, select a model element in the Project Explorer again.

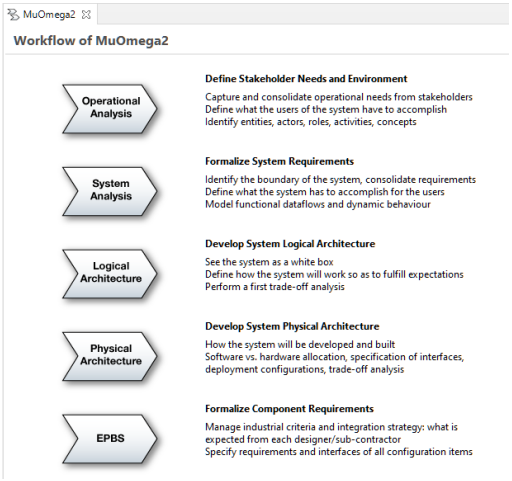


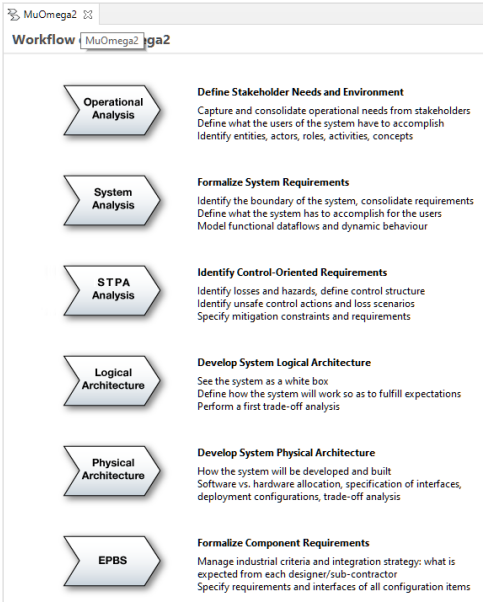
The State column shows that STPA has switched from the Unreferenced to the Active state. It means that STPA has been activated on the Capella project.



A visible consequence is that the Workflow of the project has been extended with an STPA Analysis item, located between the System Analysis and the Logical Architecture.

Before and after STPA has been activated on the project:





At this point, STPA can be applied with the help of the tool.

Note: The location of the STPA Analysis item in the Workflow view is somewhat arbitrary: while STPA requires that the purpose and perimeter of the system of interest be defined, it can be applied at very different levels of abstraction. For example, an STPA analysis can be solely based on the information contained in a high-level system analysis, while it can also rely on a precise logical or physical architecture. In the former case the analysis will focus on the interactions between the system and its environment, in the latter case it will also be able to cover, e.g., control issues that may occur within the system due to the failure of physical components or links.

The appropriate level of abstraction depends on what resources are available to carry out the analysis (people, information) and where the focus should be set to identify the most relevant issues. While relying on the knowledge of the physical architecture allows identifying precise, fine-grained controller constraints and loss scenario countermeasures, it also makes the analysis more expensive.

III. General principles of the tool

1) STPA vs. Capella modelling

An STPA analysis can be carried out independently of any other modelling work on system architecture. However, it is also possible to relate an STPA analysis with Capella modelling. The interest is twofold.

- The Capella model, as a precise source of information about the system of interest, helps carry out STPA analyses, e.g., when defining the control structure.
- The STPA analysis enriches the Capella architecture model by contributing requirements and constraints to the architecture.

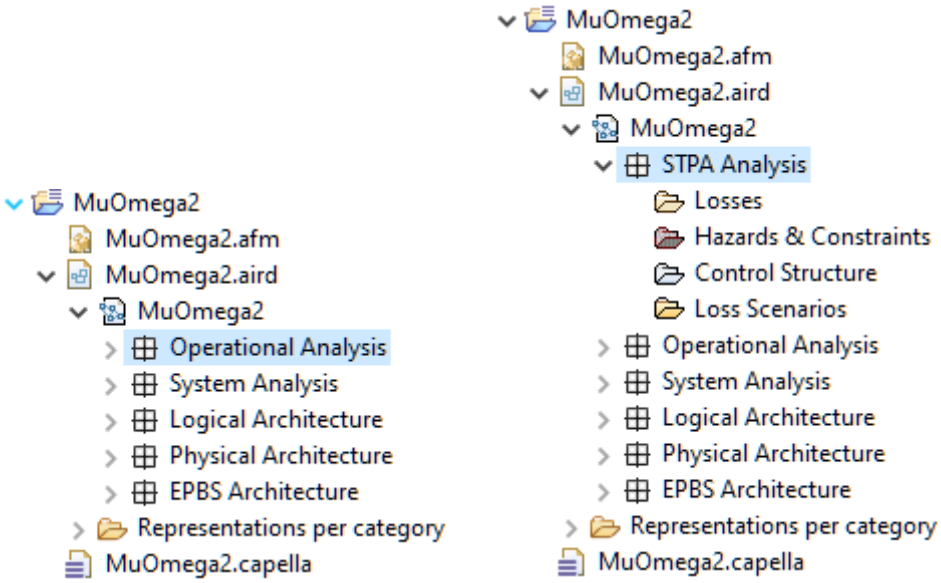
Concretely, when an STPA model element and a Capella element are related – typically, when they represent the same real-world entity –, they can be linked together[1]. Since STPA and Capella elements can also represent different facets of the same real-world entity at different levels of abstraction, an STPA element can be linked to several Capella elements and vice-versa.

Linking elements together is interesting for traceability and impact analysis. For example, an STPA control loop can correspond to a Capella functional chain. If the functional chain is modified, then the impacts on the control loop should be evaluated.

2) Data location

When applying STPA, the user edits STPA 'data' made of model elements. This data is located in a dedicated part of a Capella model that is visible in the Project Explorer view, below an 'STPA Analysis' element. This 'container' element can be seen at the same hierarchical level as System Analysis, Logical Architecture, etc. The STPA Analysis element automatically appears when the first STPA table or diagram is created.

Before (left) and after (right) STPA analysis has started:

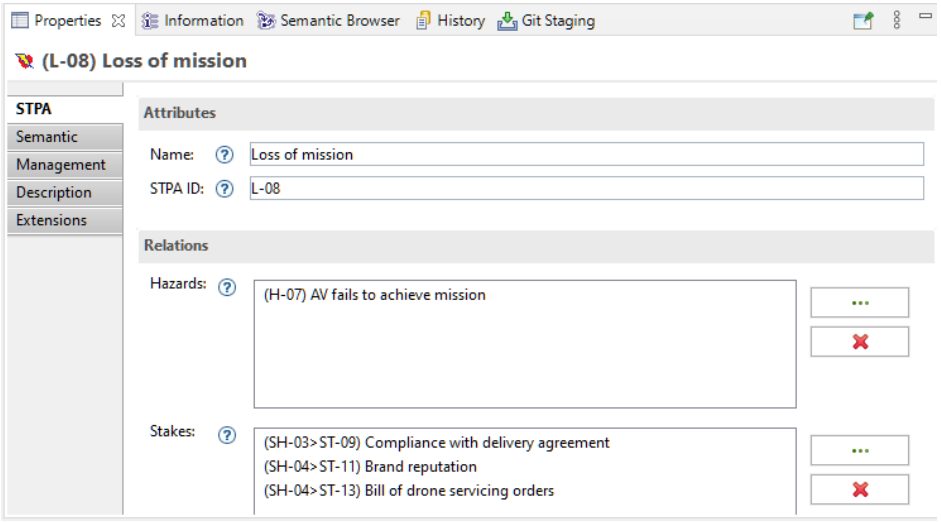


[1] Through the 'Capella Elements' property of certain STPA elements, see last section of this document.

3) Quick data visualisation and navigation

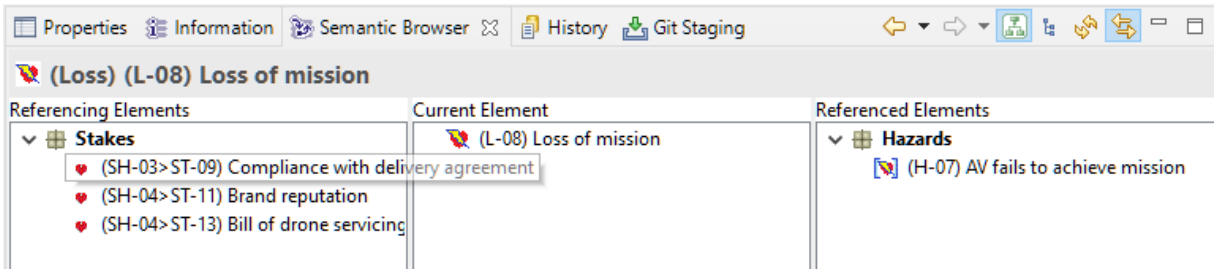
As usual in Capella, when selecting a model element in a table or diagram, its properties can be seen in the Properties and Semantic Browser views. The Properties view allows editing the element while the Semantic Browser view simplifies navigation.

The Properties view:



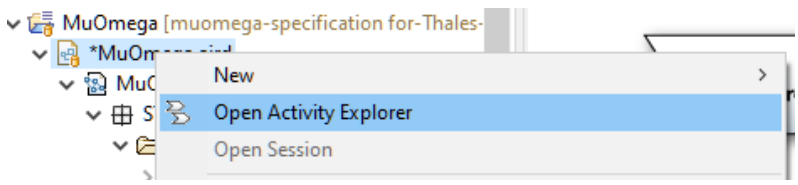
Note: The STPA tab of the Properties view only shows up if the current STPA element has been selected in a table or diagram. It does not show up if the element has been selected in the Project Explorer. This tooling issue needs further investigation.

The Semantic Browser view:

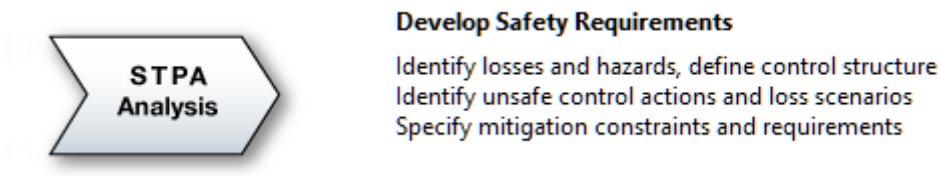


4) Workflow

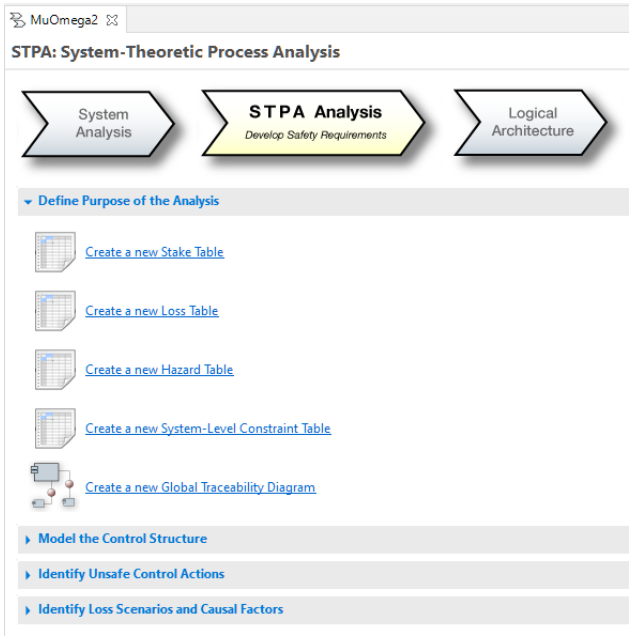
The representation of the workflow is provided by a view named Activity Explorer. It automatically shows up when a model is opened. If closed, it can be shown again by right-clicking the .aird model file and selecting 'Open Activity Explorer'.



To proceed with the STPA analysis, select the 'STPA Analysis' shape.



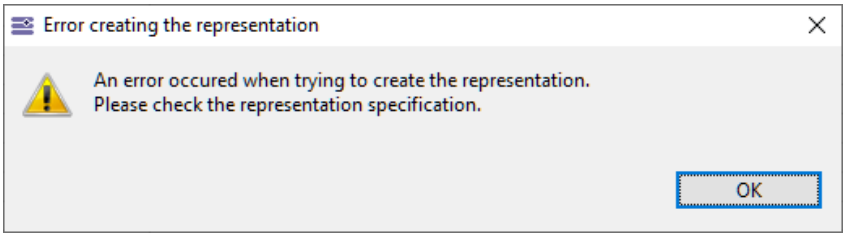
The tab dedicated to STPA opens. It is made of four expandable sections that correspond to the steps of STPA.



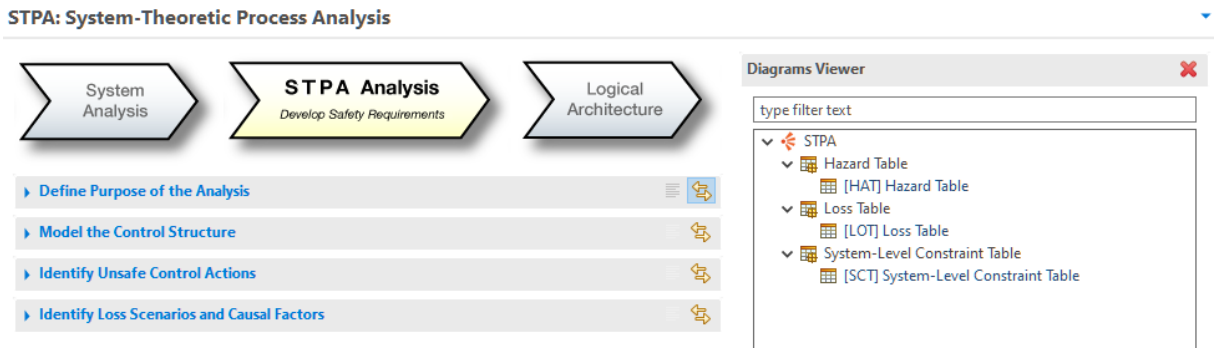
The items in each expandable section allow creating STPA diagrams or tables.

Note: Contrary to a number of Capella diagrams, STPA diagrams and tables automatically reflect the whole content of the model. It is thus generally not useful to create more than one table or diagram of the same type. For example, two Loss Tables will always have the same contents – although it can be represented slightly differently, e.g., in terms of column width or row order.

Trying to create redundant tables or diagrams may result in the following error message.

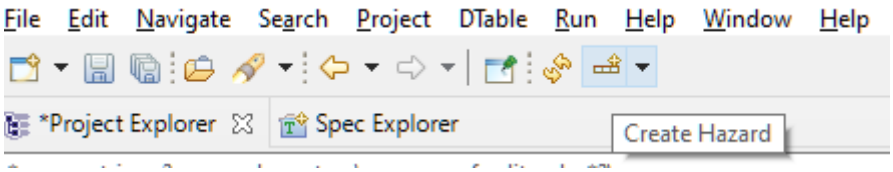


If diagrams or tables have been created, they appear on the right-hand side. Double-click the desired diagram or table to open it. Select the double-arrow ( ) button associated to a given STPA step to only see the tables and diagrams of the step.

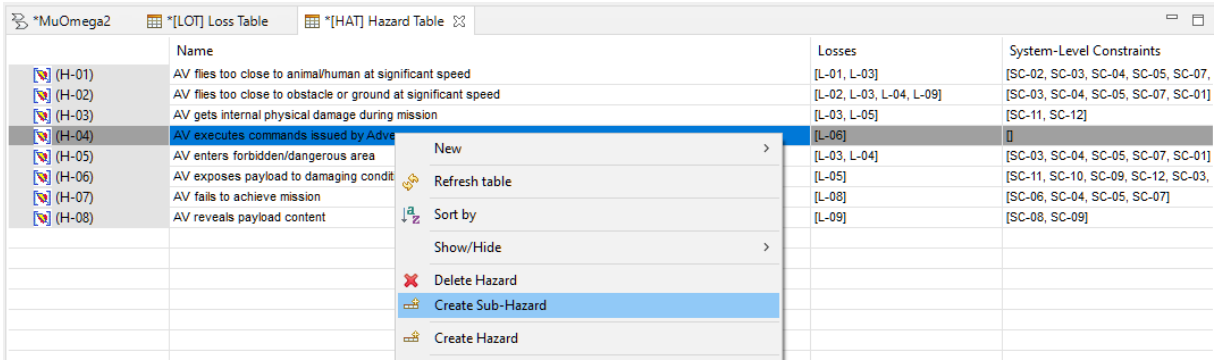


5) Edition with tables

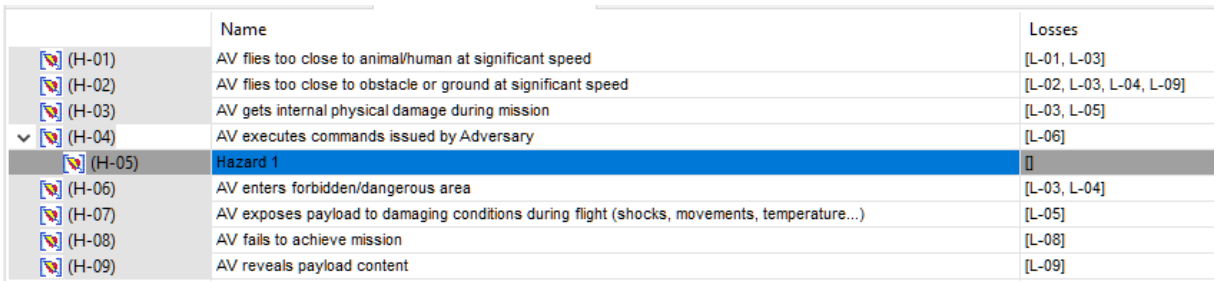
When a table is empty, a first row can be created by clicking the button on the right-hand side of the toolbar



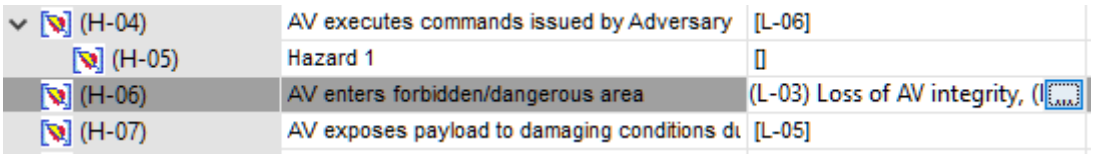
Alternatively, it is possible to right-click an existing row to create another one.



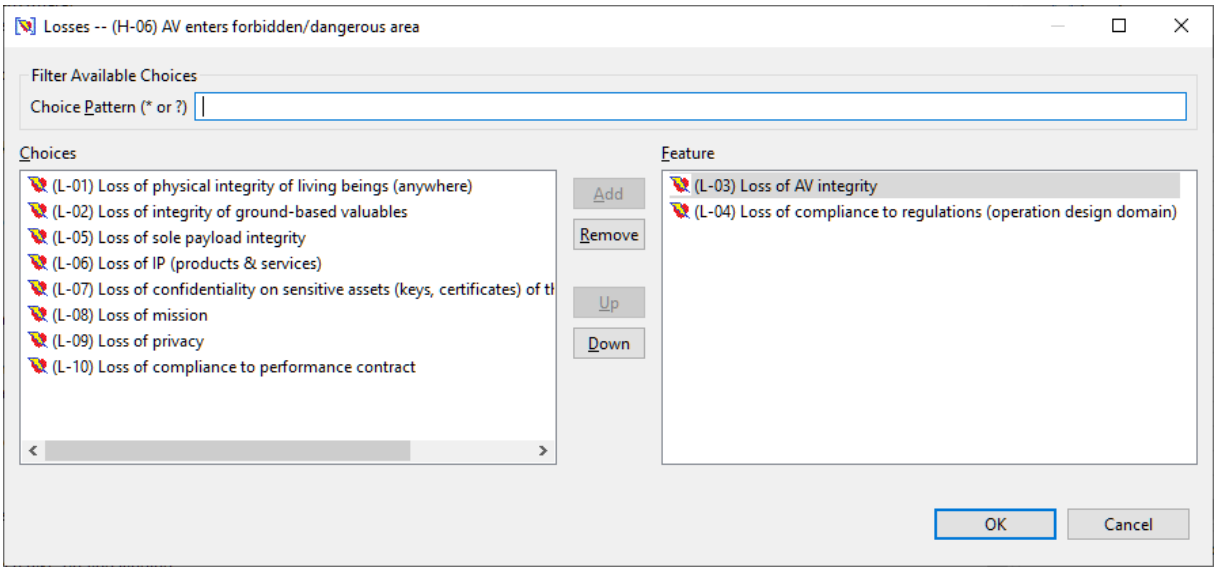
In certain tables, a right click can also lead to the creation of a row that represents a sub-element: e.g., a sub-hazard.



To edit a cell, double-click it. If the cell references model elements (e.g., the losses referenced by a hazard), a button with caption '…' shows up: click it to open a dedicated window.



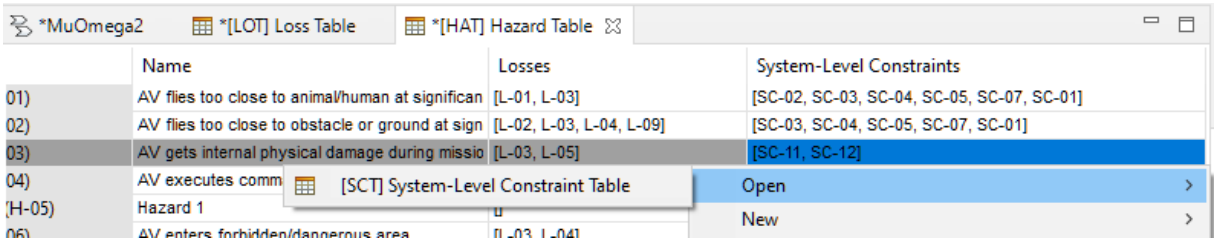
The window in question allows defining which elements among a set of possible ones (list on the left) are effectively referenced (list on the right). For example, if the list on the right is empty than no element is referenced, while the list on the left being empty means that all possible elements are referenced. Use the Add and Remove buttons to move elements from one list to the other, or double-click the elements directly.



Note: Only elements that already exist can be referenced. It may happen that no element can be referenced (both lists are empty) because no appropriate element has been created yet.

Lines in a table can be re-ordered manually. Simply drag a line and drop it wherever needed.

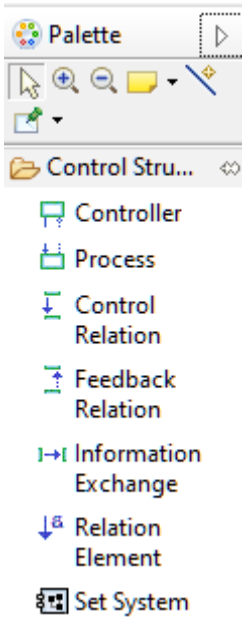
Navigation between tables (or to diagrams) can be done by right-clicking any cell in a given column and selecting Open to open an existing table or diagram, or New to create a new one. For example, selecting a cell in the System-Level Constraints column in a Hazard Table allows opening or creating a System-Level Constraint Table.



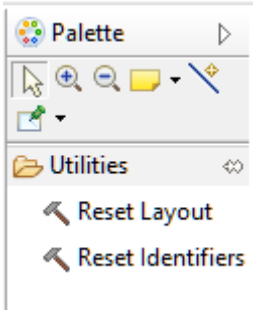
6) Edition with diagrams

STPA diagrams are similar to classical Capella diagrams.

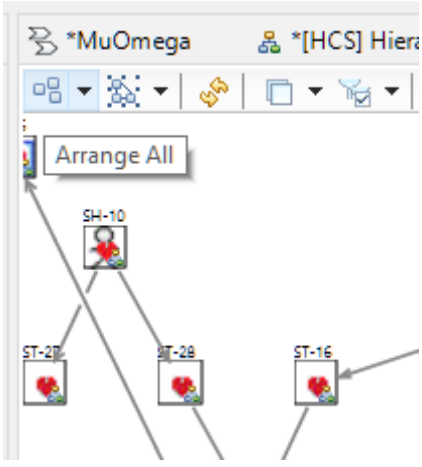
Some of them can be edited by the means of a palette located on the right of the Capella window. As an example, the palette of the Hierarchical Control Structure Diagram:



Other diagrams are mostly read-only but include a palette that allows resetting the layout or re-numbering the identifiers of STPA elements. For example, the palette of the Global Traceability Diagram:

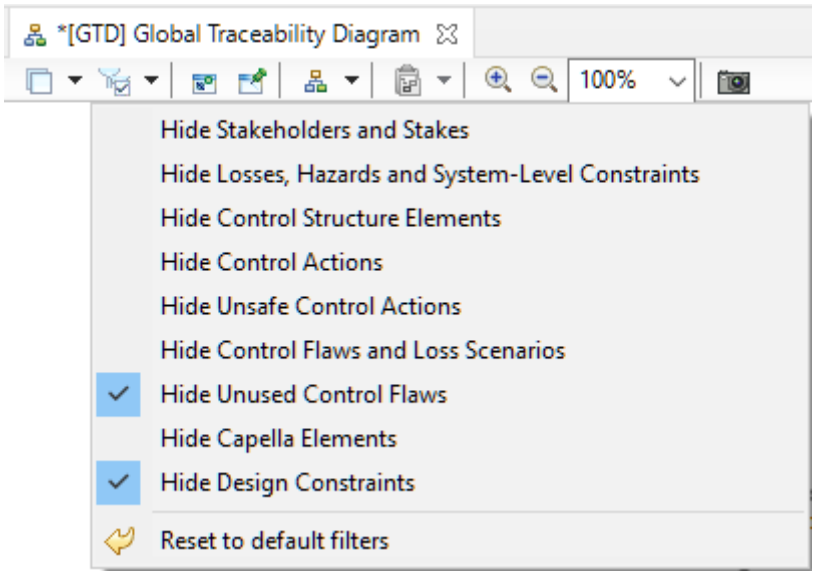


Select one of the tools in the palette then click the background of the diagram. The Reset Layout tool is often used in combination with the Arrange All button in the toolbar as the latter proposes a new graphical layout.



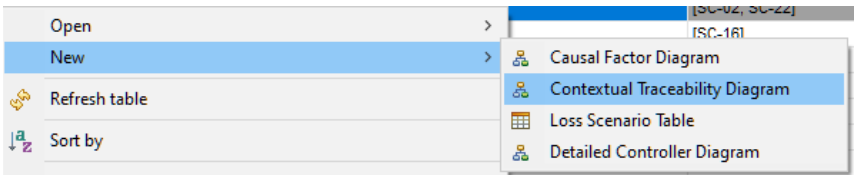
Certain diagrams have filters that allow hiding/showing certain sets of elements. They can be enabled or disabled via a button in the toolbar.

For example, in the case of the Global Traceability Diagram:

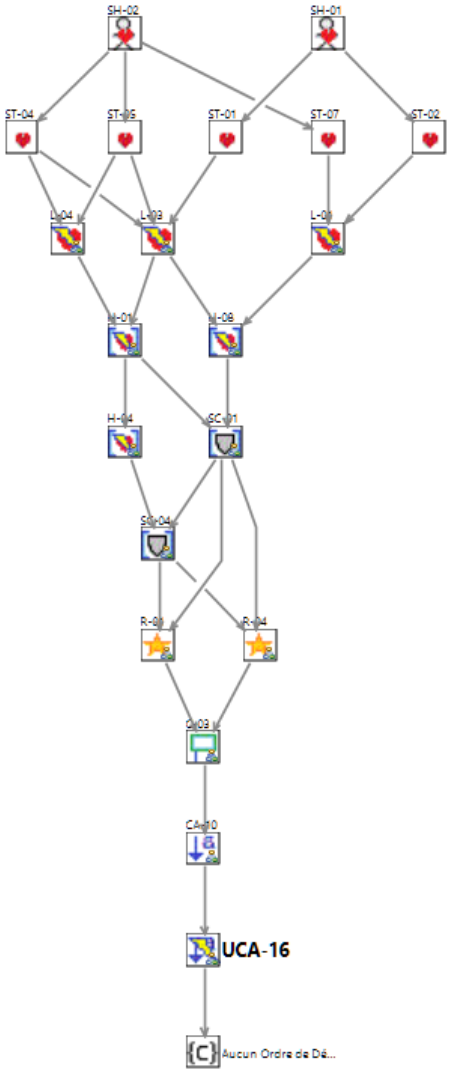


7) Traceability

Every STPA analysis element can be visualized in a Contextual Traceability Diagram. This diagram allows understanding where the element comes from in the analysis process and what part of the analysis it leads to.



For example, on an unsafe control action the resulting diagram may look like this.



The main element, here the unsafe control action, is represented in bold. Arrows represent the fact that elements are related: for example, that a given hazard is associated to a given loss.

It is possible to hover over an element to see its name and to select it to see its properties.

IV. Apply STPA

1) Define Purpose of the Analysis (cf. Handbook)

a) Identify losses (cf. Handbook)

Create a new Stake Table

Result:

*MuOmega | *[STT] Stake Table

	Name	Losses
(SH-01)	**Field Operator**	
(ST-01)	Operational secrets (keys, certificates)	[L-07]
(ST-02)	Physical integrity	[L-01]
(ST-03)	Payload integrity	[L-05]
(SH-02)	**Ground Station Operator**	
(ST-04)	Operational secrets (keys, certificates)	[L-07]
(ST-05)	Ground Station information integrity	[L-07]
(ST-06)	Ground Station physical integrity	[L-01]
(SH-03)	**Delivery Customer**	
(ST-07)	Physical integrity	[L-01]
(ST-08)	Payload content (customer good)	[L-05]
(ST-09)	Compliance with delivery agreement	[L-08, L-10]
(ST-10)	Privacy of payload transport (payload content, location, time, ...)	[L-09]

Then 

Create a new Loss Table

Result:

*MuOmega | *[LOT] Loss Table

	Name	Stakes	Hazards
(L-01)	Loss of physical integrity of living beings (anywhere)	[ST-02, ST-06, ST-07, ST-23, ST-25]	[H-01]
(L-02)	Loss of integrity of ground-based valuables	[ST-21, ST-22]	[H-02]
(L-03)	Loss of AV integrity	[ST-11, ST-12, ST-15, ST-17]	[H-03, H-06, H-01, H-02]
(L-04)	Loss of compliance to regulations (operation design domain)	[ST-14, ST-19, ST-20, ST-26]	[H-06, H-02]
(L-05)	Loss of sole payload integrity	[ST-03, ST-08, ST-11, ST-12, ST-13, ST-23]	[H-03, H-07]
(L-06)	Loss of IP (products & services)	[ST-16, ST-28]	[H-04]

b) Identify system-level hazards (cf. Handbook)

Create a new Hazard Table

Result:

*MuOmega | *[HAT] Hazard Table

	Name	Losses	System-Level Constraints
(H-01)	AV flies too close to animal/human at significan	[L-01, L-03]	[SC-02, SC-03, SC-04, SC-05, SC-07, SC-01]
(H-02)	AV flies too close to obstacle or ground at sign	[L-02, L-03, L-04, L-09]	[SC-03, SC-04, SC-05, SC-07, SC-01]
(H-03)	AV gets internal physical damage during missio	[L-03, L-05]	[SC-11, SC-12]
(H-04)	AV executes commands issued by Adversary	[L-06]	[]
(H-06)	AV enters forbidden/dangerous area	[L-03, L-04]	[SC-03, SC-04, SC-05, SC-07, SC-01]

c) Identify system-level constraints (cf. Handbook)

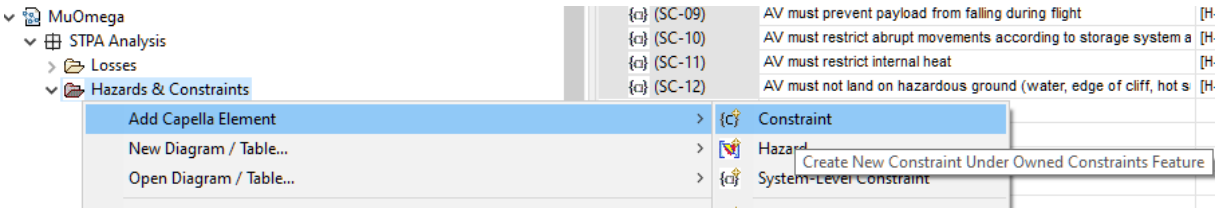
Create a new System-Level Constraint Table

Result:

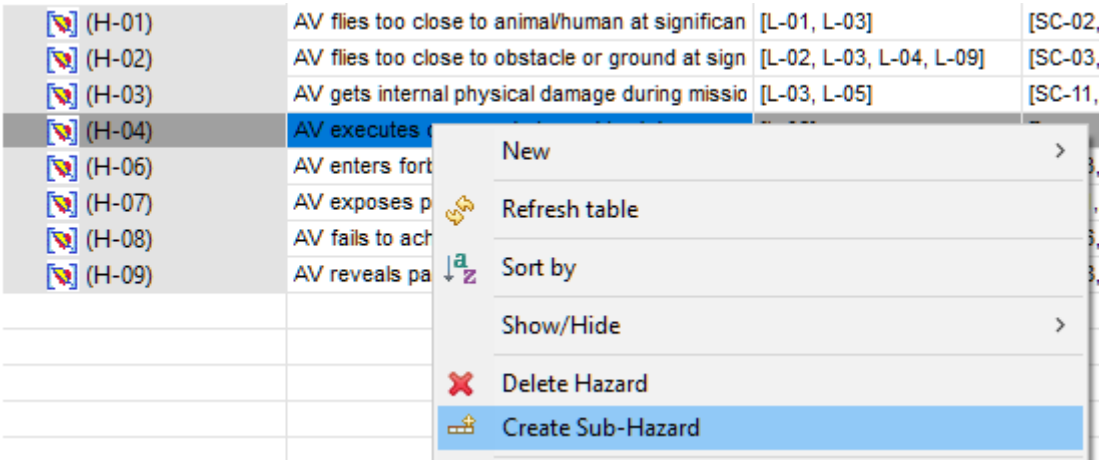
*MuOmega | *[SCT] System-Level Constraint Table

	Name	Hazards	Responsibilities	Assumptions
{α} (SC-01)	AV must not get too close to living beings, obstacles or forbidden a	[H-01, H-02, H-06]	[R-06, R-14]	[]
{α} (SC-02)	AV must allow the Field Operator to be at a safe distance at take-c	[H-01]	[R-05, R-10, R-12]	[Field Operator must not remain close t
{α} (SC-03)	AV must maintain controlled flight between take-off and landing	[H-01, H-02, H-06, H-07]	[R-04, R-08]	[Field Operator must not send take-off
{α} (SC-04)	AV must conform to flight plan as much as possible	[H-01, H-02, H-06, H-07, H-08]	[R-15, R-14, R-01, R-12, R	[]
{α} (SC-05)	AV must accept remote control and remote flight plan update	[H-01, H-02, H-06, H-07, H-08]	[R-07]	[]
{α} (SC-06)	AV must not take off if flight plan cannot be achieved	[H-08]	[R-01, R-12, R-05, R-10]	[]
{α} (SC-07)	AV must perform controlled emergency landing if flight plan cannot	[H-01, H-02, H-06, H-07, H-08]	[R-08]	[]

Elements in the Assumptions column are constraints located in the STPA Analysis. They can be created in Detailed Controller Diagrams (see last STPA step) or directly in the Project Explorer.



d) Refine hazards (cf. Handbook)



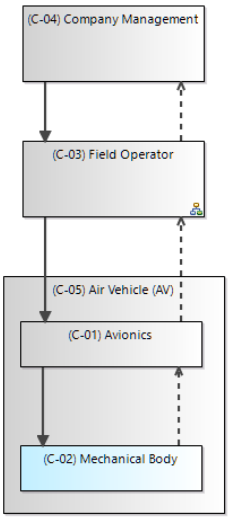
The same can be done in the System-Level Constraint Table.

2) Model the Control Structure (cf. Handbook)

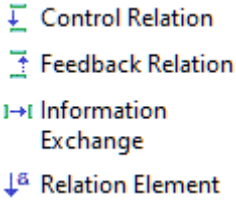
a) Identify controllers (cf. Handbook)

Create a new Hierarchical Control Structure Diagram

Result:



The relations (arrows) between controllers/processes are of type control, feedback or information exchange as proposed in the palette. Exchange elements (control actions, feedback or any piece of information) can be added to a relation by selecting Relation Element in the palette.



Alternatively, it is possible to just select a relation and press F2 to directly edit its list of exchange elements as plain text, with one exchange element per line. New lines are added by pressing Ctrl-Enter.

b) Identify responsibilities (cf. Handbook)

Create a new Responsibility Table

OR

Result:

*MuOmega | *[RET] Responsibility Table of Avionics

	Name	System-Level Constraints	Control Actions	Feedback	Information	Process Model
(R-04)	Maintain controlled flight during mission	[SC-03]	[CA-05, CA-06]	[FB-08, FB-09, FB-10]	[]	[PM-06]
(R-05)	Determine mission viability	[SC-02, SC-06, SC-09]	[CA-05]	[]	[]	[PM-06, PM-07, PM-08]
(R-06)	Execute Flight Plan	[SC-01, SC-04]	[CA-05, CA-06]	[FB-08, FB-09, FB-10]	[]	[PM-06, PM-07]
(R-07)	Execute remote commands	[SC-05]	[CA-05, CA-06]	[FB-08, FB-09, FB-10]	[]	[PM-06, PM-09, PM-10]
(R-08)	Execute emergency landing	[SC-03, SC-07, SC-10]	[CA-06]	[FB-08, FB-09, FB-10]	[]	[PM-06, PM-10]
(R-09)	Preserve payload integrity	[SC-10]	[CA-05, CA-06]	[]	[]	[PM-06]

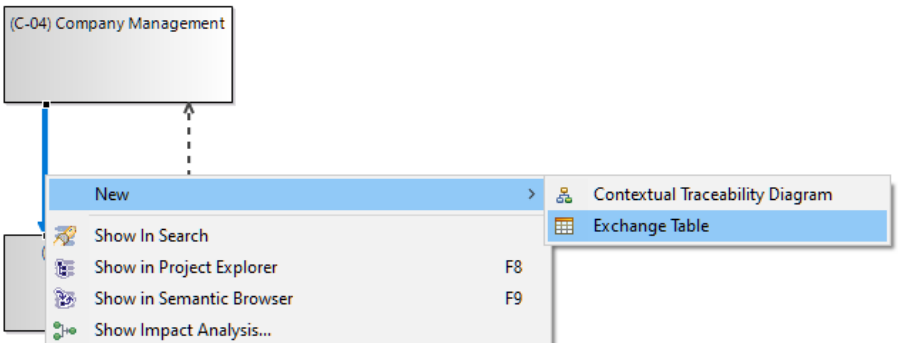
c) Identify control actions (cf. Handbook)

In the Hierarchical Structure Diagram, select Relation Element then click a Control Relation.

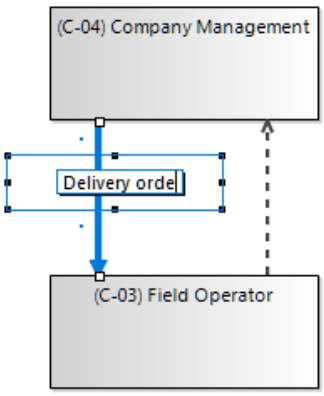
OR

Create a new Exchange Table

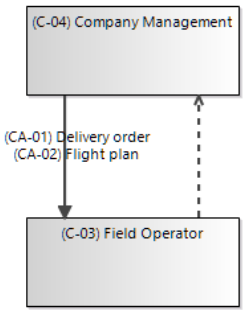
OR



OR (after pressing the F2 key)



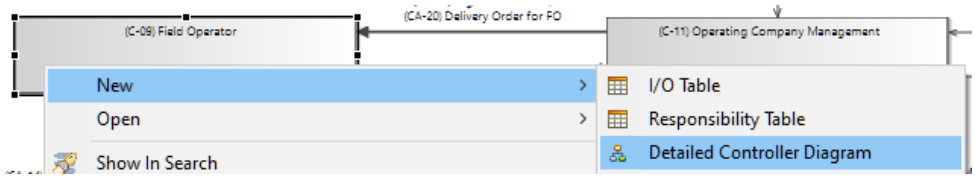
Result:



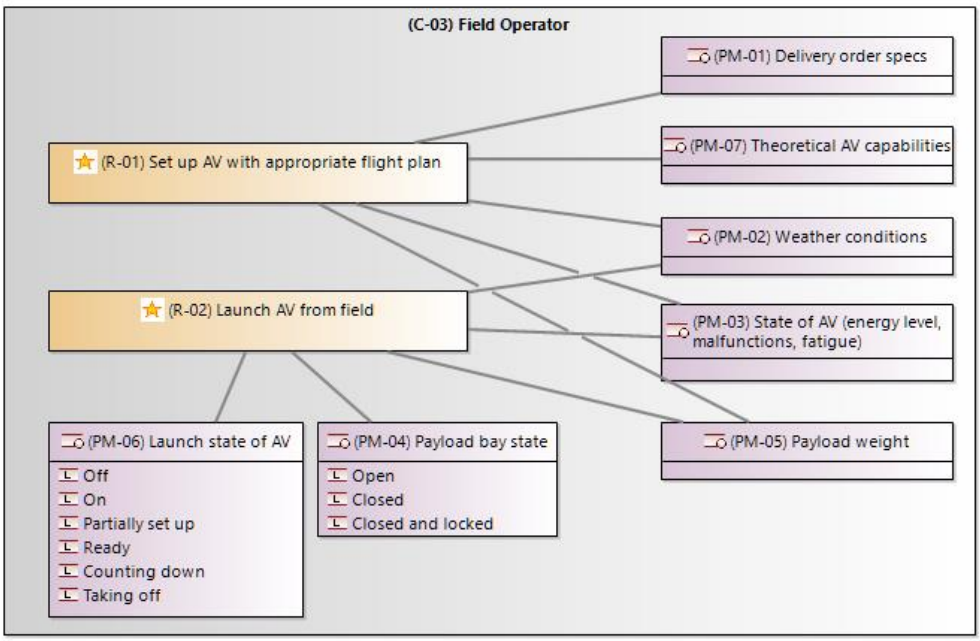
[EXT] Exchange Table of Company Management->Field Operator

	Name	Exercised Responsibilities	Supported Responsibilities
(CA-01)	Delivery order	[]	
(CA-02)	Flight plan	[]	

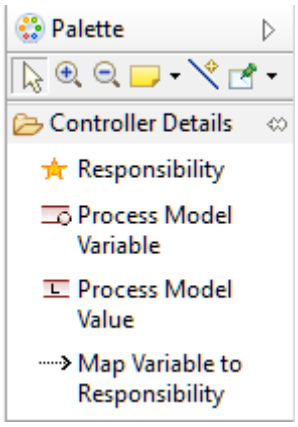
d) Identify process models (cf. Handbook)



Result:



Using:

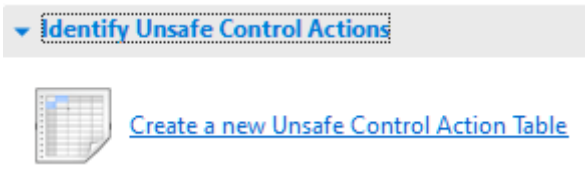


e) Identify feedback and other information (cf. Handbook)

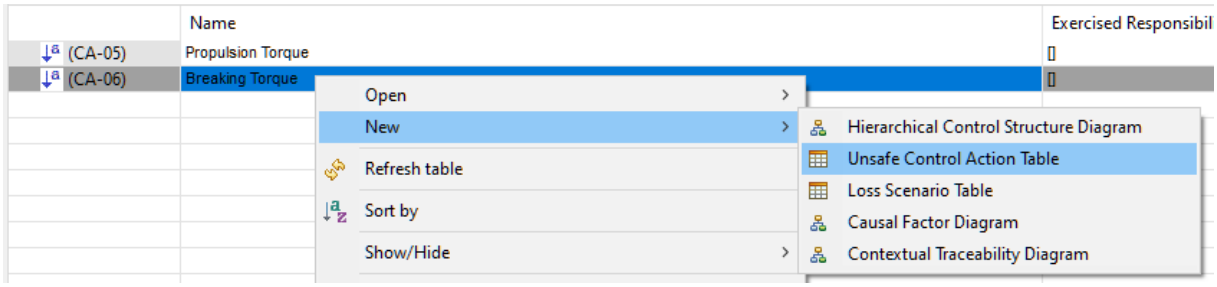
Similar to control actions, but starting from a feedback or information exchange relation.

3) Identify Unsafe Control Actions (cf. Handbook)

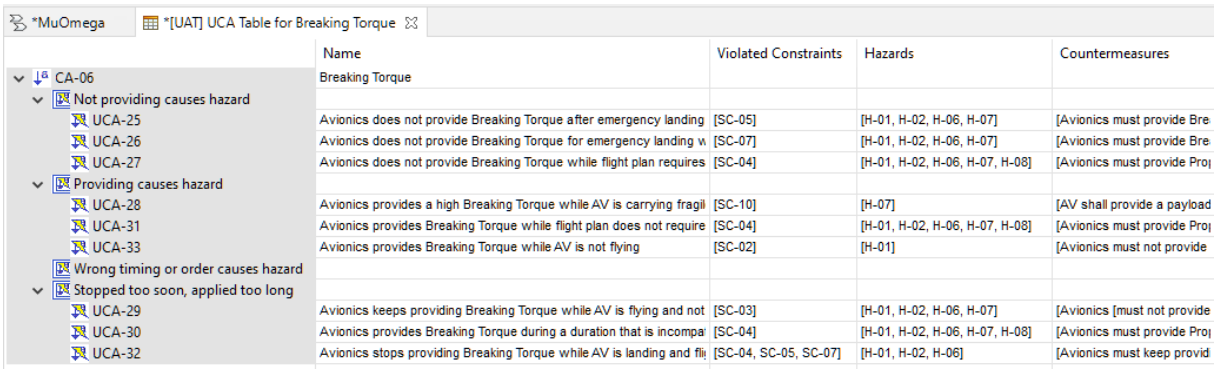
a) Identify unsafe control actions (cf. Handbook)



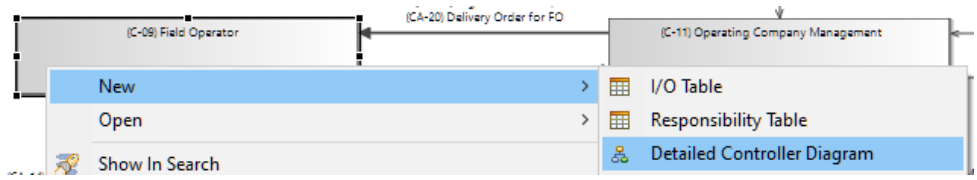
OR



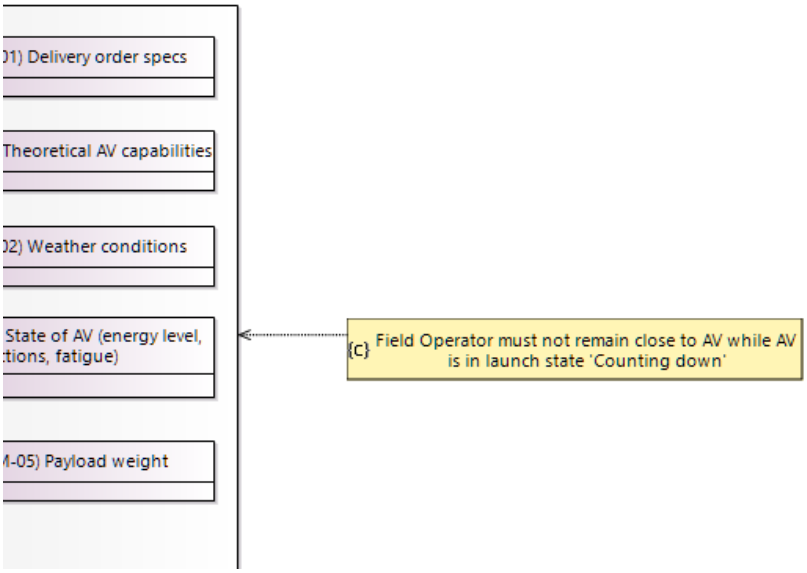
Result:



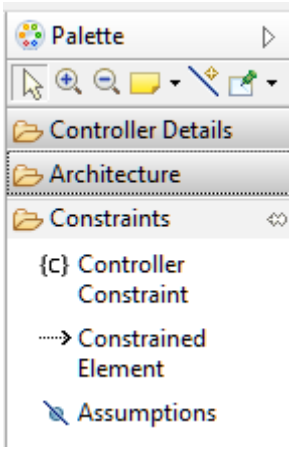
b) Define controller constraints (cf. Handbook)



Result:

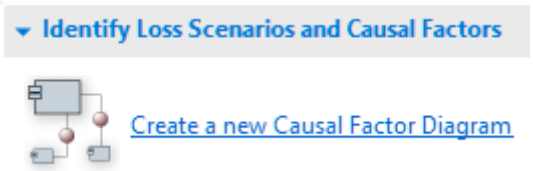

01) Delivery order specs
Theoretical AV capabilities
02) Weather conditions
State of AV (energy level,
tions, fatigue)
M-05) Payload weight
{C} Field Operator must not remain close to AV while AV is in launch state 'Counting down'

Using:

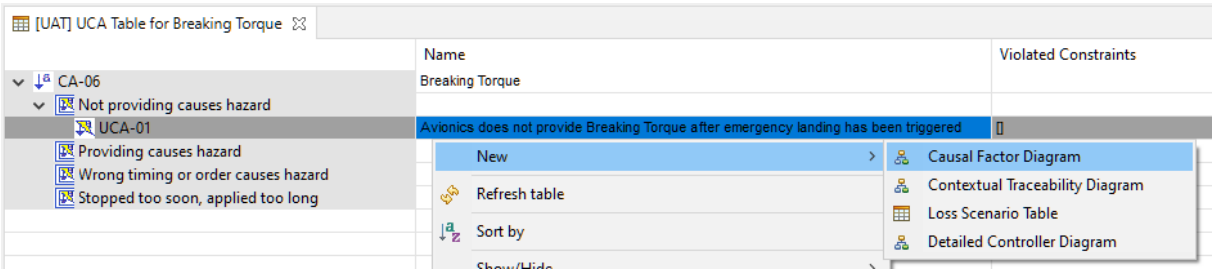

Palette
Controller Details
Architecture
Constraints
{C} Controller Constraint
Constrained Element
Assumptions

4) Identify Loss Scenarios (cf. Handbook)

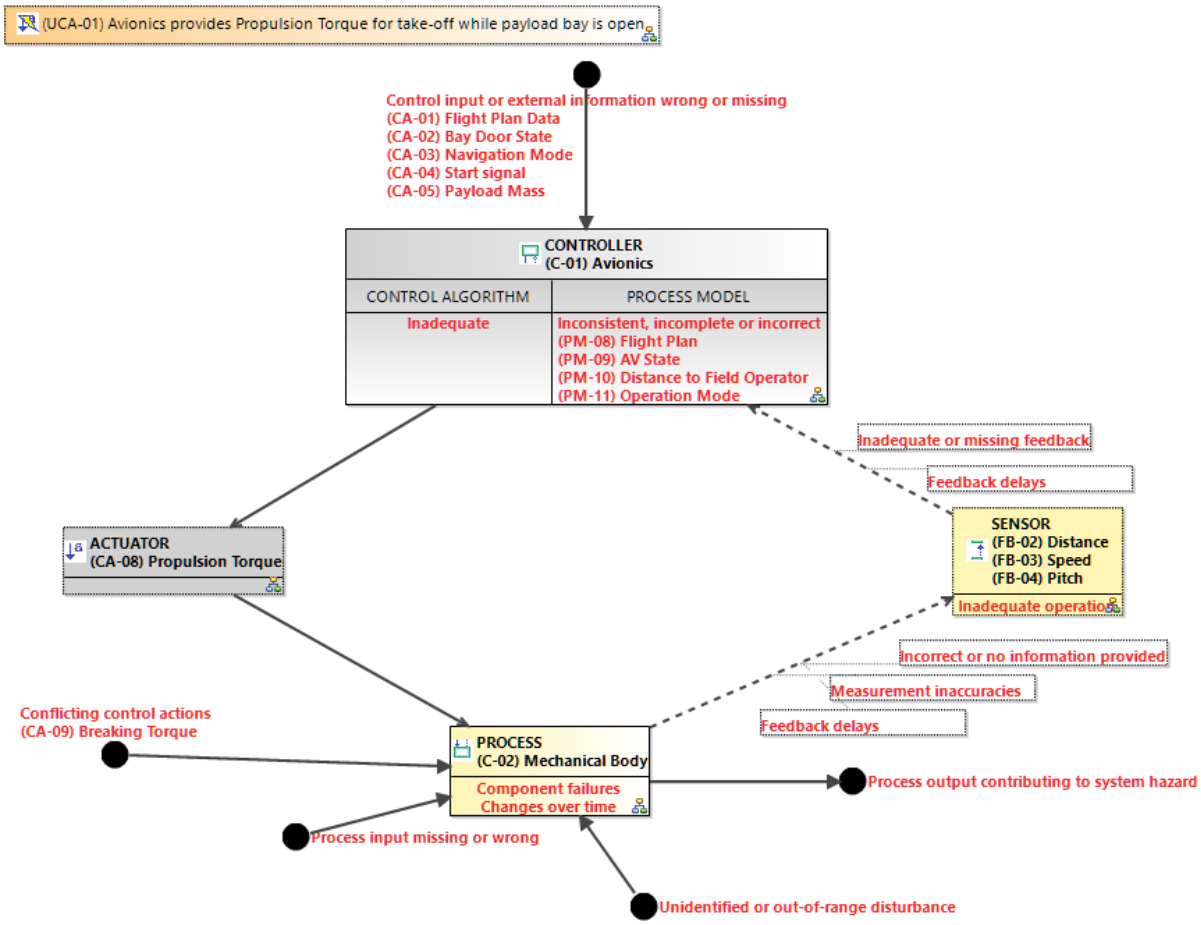
a) Identify scenarios that lead to unsafe control actions (cf. Handbook)



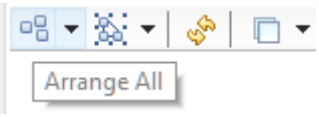
OR



Result:

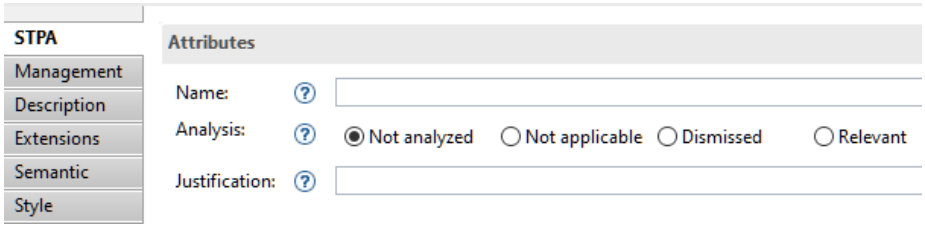


The ‘Arrange All’ button generally helps get a slightly better layout.



This diagram represents the control loop related to the unsafe control action, annotated with hint words to help reflect about causal factors. Those hint words are pretty general; they originate from Figure G.1 in [1]. More specific or precise hint words could be proposed, such as those from Figure G.2 of [1], but the choice was made to remain as simple as possible. Customization of the hint words is only possible by a manual renaming.

The idea is to go through all hint words (e.g., 'Feedback delays') representing potential causal factors, and reflect on whether they could be involved in a loss scenario. For every causal factor, the Properties view allows setting a status and justifying it (see below).



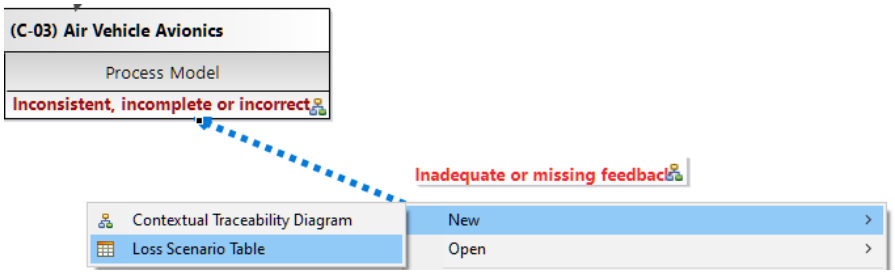
Also, the textual description of each causal factor can be edited, either via its 'Name' property or by pressing F2 in the diagram. For better assistance in the analysis, the description is pre-filled with contextual information relative to the control loop. For example, the causal factor relative to an inappropriate process model is pre-filled with process model variables that are associated to the responsibility to which the control action contributes.

The colour of the causal factors is updated according to its status.

- At the beginning, all potential causal factor are represented in **light red**. It means they have not been examined yet.
- Every causal factor being examined can be set as:
  - '**Not applicable**', i.e., irrelevant in the present context. A justification must be provided via the Properties view. The causal factor then becomes **grey**.
  - '**Dismissed**', i.e., impossible, improbable enough or without consequences. Again, a justification must be provided via the Properties view. The causal factor then becomes **green**.
  - '**Relevant**', i.e., possible with consequences. The causal factor becomes **dark red**.
- In the 'Relevant' case, corresponding loss scenarios must be defined. When the causal factor is referenced by a least one loss scenario, it becomes **orange** in order to reflect that the causal factor has been (at least partially) taken into account in the remainder of the analysis.

Note: Causal factors must not only be examined individually but also in combination. Besides, the reflection must take into account the temporal dimension, i.e., multiple successive iterations of the loop.

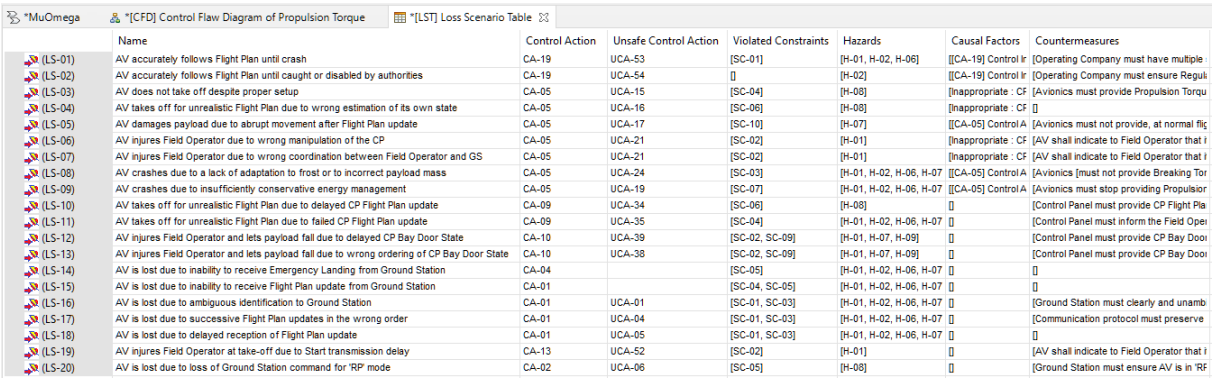
Once a causal factor or combination of causal factors leading to loss scenarios have been identified (they are in **dark red**), corresponding loss scenarios must be made explicit, i.e., they have to be created and described.



OR

Create a new Loss Scenario Table

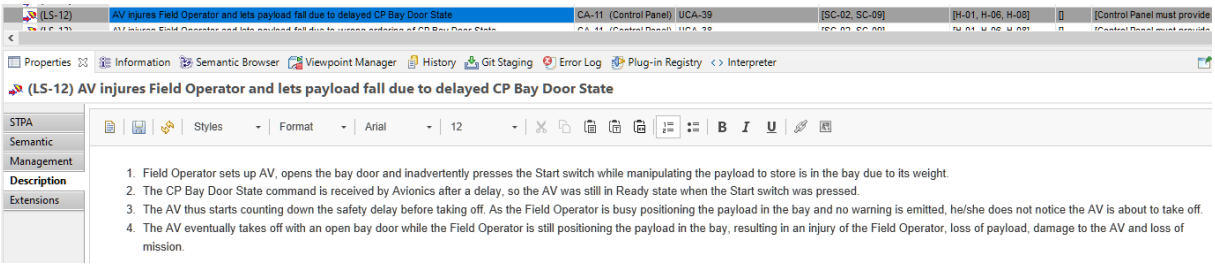
Result:



*MuOmega | *[CFD] Control Flaw Diagram of Propulsion Torque | *[LST] Loss Scenario Table

	Name	Control Action	Unsafe Control Action	Violated Constraints	Hazards	Causal Factors	Countermeasures
(LS-01)	AV accurately follows Flight Plan until crash	CA-19	UCA-53	[SC-01]	[H-01, H-02, H-06]	[[CA-19] Control Ir	[Operating Company must have multiple
(LS-02)	AV accurately follows Flight Plan until caught or disabled by authorities	CA-19	UCA-54	[]	[H-02]	[[CA-19] Control Ir	[Operating Company must ensure Regul
(LS-03)	AV does not take off despite proper setup	CA-05	UCA-15	[SC-04]	[H-08]	[Inappropriate : CF	[Avionics must provide Propulsion Torqu
(LS-04)	AV takes off for unrealistic Flight Plan due to wrong estimation of its own state	CA-05	UCA-16	[SC-06]	[H-08]	[Inappropriate : CF	[]
(LS-05)	AV damages payload due to abrupt movement after Flight Plan update	CA-05	UCA-17	[SC-10]	[H-07]	[[CA-05] Control A	[Avionics must not provide, at normal flig
(LS-06)	AV injures Field Operator due to wrong manipulation of the CP	CA-05	UCA-21	[SC-02]	[H-01]	[Inappropriate : CF	[AV shall indicate to Field Operator that I
(LS-07)	AV injures Field Operator due to wrong coordination between Field Operator and GS	CA-05	UCA-21	[SC-02]	[H-01]	[Inappropriate : CF	[AV shall indicate to Field Operator that I
(LS-08)	AV crashes due to a lack of adaptation to frost or to incorrect payload mass	CA-05	UCA-24	[SC-03]	[H-01, H-02, H-06, H-07	[[CA-05] Control A	[Avionics [must not provide Breaking Tor
(LS-09)	AV crashes due to insufficiently conservative energy management	CA-05	UCA-19	[SC-07]	[H-01, H-02, H-06, H-07	[[CA-05] Control A	[Avionics must stop providing Propulsior
(LS-10)	AV takes off for unrealistic Flight Plan due to delayed CP Flight Plan update	CA-09	UCA-34	[SC-06]	[H-08]	[]	[Control Panel must provide CP Flight Pla
(LS-11)	AV takes off for unrealistic Flight Plan due to failed CP Flight Plan update	CA-09	UCA-35	[SC-04]	[H-01, H-02, H-06, H-07	[]	[Control Panel must inform the Field Oper
(LS-12)	AV injures Field Operator and lets payload fall due to delayed CP Bay Door State	CA-10	UCA-39	[SC-02, SC-09]	[H-01, H-07, H-09]	[]	[Control Panel must provide CP Bay Door
(LS-13)	AV injures Field Operator and lets payload fall due to wrong ordering of CP Bay Door State	CA-10	UCA-38	[SC-02, SC-09]	[H-01, H-07, H-09]	[]	[Control Panel must provide CP Bay Door
(LS-14)	AV is lost due to inability to receive Emergency Landing from Ground Station	CA-04		[SC-05]	[H-01, H-02, H-06, H-07	[]	[]
(LS-15)	AV is lost due to inability to receive Flight Plan update from Ground Station	CA-01		[SC-04, SC-05]	[H-01, H-02, H-06, H-07	[]	[]
(LS-16)	AV is lost due to ambiguous identification to Ground Station	CA-01	UCA-01	[SC-01, SC-03]	[H-01, H-02, H-06, H-07	[]	[Ground Station must clearly and unamb
(LS-17)	AV is lost due to successive Flight Plan updates in the wrong order	CA-01	UCA-04	[SC-01, SC-03]	[H-01, H-02, H-06, H-07	[]	[Communication protocol must preserve
(LS-18)	AV is lost due to delayed reception of Flight Plan update	CA-01	UCA-05	[SC-01, SC-03]	[H-01, H-02, H-06, H-07	[]	[]
(LS-19)	AV injures Field Operator at take-off due to Start transmission delay	CA-13	UCA-52	[SC-02]	[H-01]	[]	[AV shall indicate to Field Operator that I
(LS-20)	AV is lost due to loss of Ground Station command for 'RP' mode	CA-02	UCA-06	[SC-05]	[H-08]	[]	[Ground Station must ensure AV is in 'RF

The details of Loss Scenarios can be written in their description.



(LS-12) AV injures Field Operator and lets payload fall due to delayed CP Bay Door State | CA-11 (Control Panel) | UCA-39 | [SC-02, SC-09] | [H-01, H-06, H-08] | [] | [Control Panel must provide

Properties | Information | Semantic Browser | Viewpoint Manager | History | Git Staging | Error Log | Plug-in Registry | Interpreter

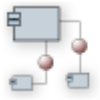
(LS-12) AV injures Field Operator and lets payload fall due to delayed CP Bay Door State

STPA | Semantic | Management | Description | Extensions

Styles | Format | Arial | 12

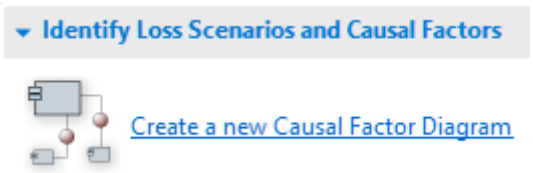
1. Field Operator sets up AV, opens the bay door and inadvertently presses the Start switch while manipulating the payload to store is in the bay due to its weight.
2. The CP Bay Door State command is received by Avionics after a delay, so the AV was still in Ready state when the Start switch was pressed.
3. The AV thus starts counting down the safety delay before taking off. As the Field Operator is busy positioning the payload in the bay and no warning is emitted, he/she does not notice the AV is about to take off.
4. The AV eventually takes off with an open bay door while the Field Operator is still positioning the payload in the bay, resulting in an injury of the Field Operator, loss of payload, damage to the AV and loss of mission.

The Countermeasures column refers to constraints. If needed, new Detailed Controller Diagrams can be created to define new constraints. STPA constraints are technically standard Capella constraints.

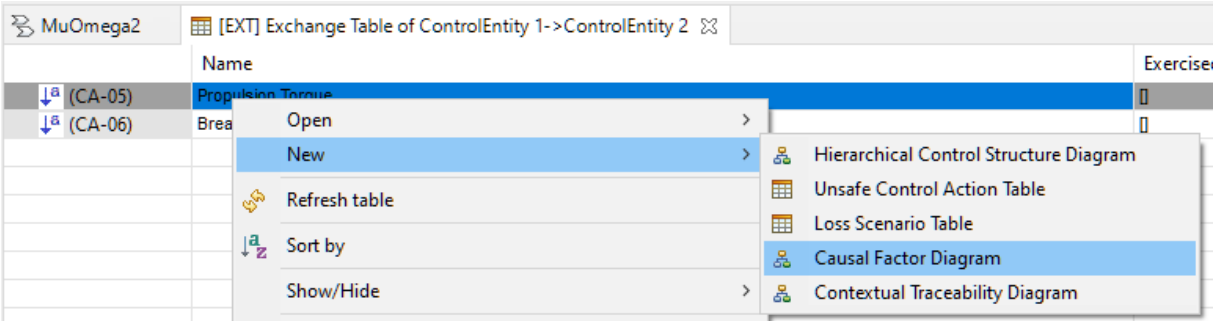


Create a new Detailed Controller Diagram

b) Identify scenarios for control actions improperly executed or not executed (cf. Handbook)

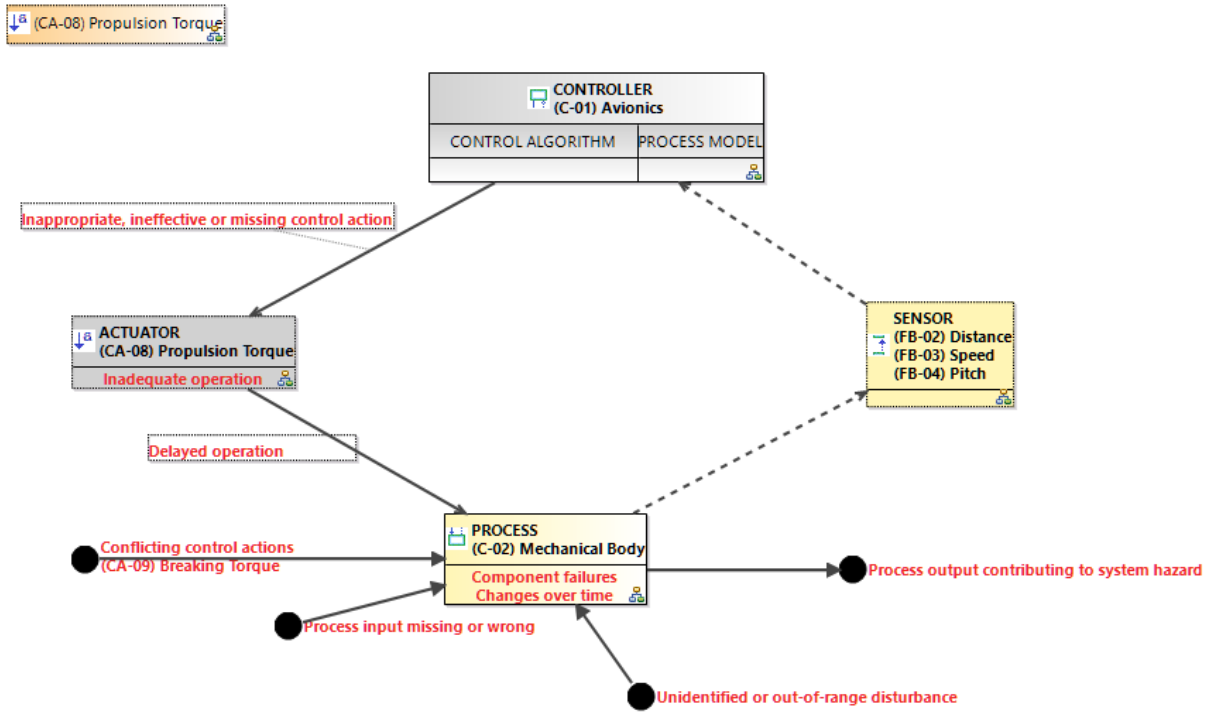


OR



Same as **a)** but the focus is on the bottom left-hand half of the control loop in the Causal Factor Diagram.

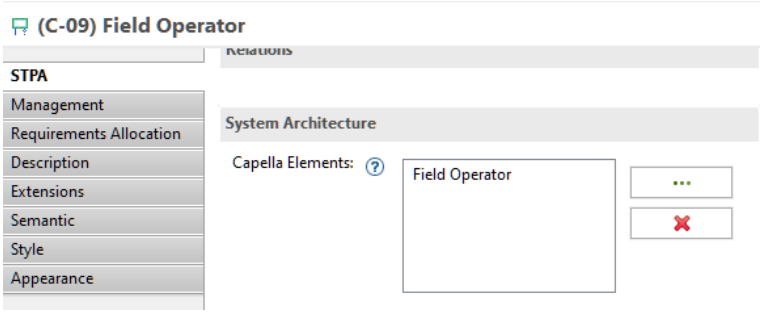
Result:



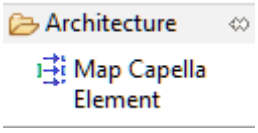
V. (optional) Define mapping to system architecture

Most of the time, a model element in an STPA analysis can be related to a Capella element, which means that both elements represent the same real-world entity. This is comparable to realisation links between Capella elements of different Arcadia perspectives (e.g., from Logical Architecture to System Analysis).

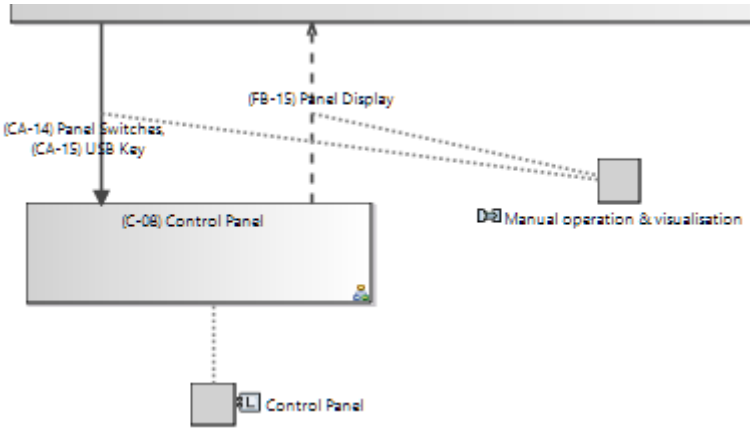
At any time during the application of STPA, it can be specified via the Capella Elements property of an STPA element.



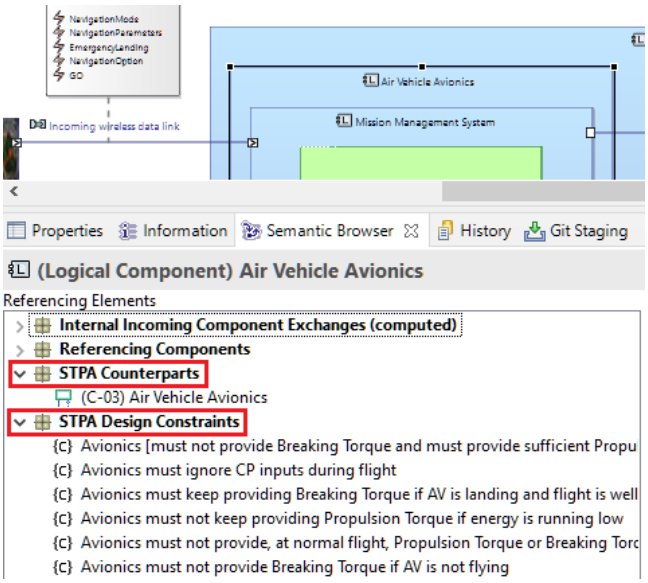
It can also be specified in a Detailed Controller Diagram via a dedicated tool in the palette.



Note that several Capella elements of the same Arcadia perspective (e.g., System Analysis) can be linked to the same STPA element and vice-versa. This is because the level of abstraction and modelling concerns usually differ between STPA and a given Arcadia perspective, leading in some cases to an N-M mapping relation.



More generally, the mapping between STPA elements and Capella elements is not constrained (e.g., in terms of element types). When an STPA element is related (mapped) to a Capella element, this relation appears in the Semantic Browser representation of the Capella element. The relation is named ‘STPA Counterparts’.



Additionally, when the STPA element is a controller, all its controller constraints that result from the STPA analysis are ‘inherited’ by the Capella element: they appear under ‘STPA Design Constraints’ in the Semantic Browser view.

VI. REFERENCES

[1] Nancy G. Leveson, John P. Thomas, *STPA Handbook* (2018). https://psas.scripts.mit.edu/home/get_file.php?name=STPA_handbook.pdf

[2] *MIT Partnership for Systems Approaches to Safety and Security*. http://psas.scripts.mit.edu/home/

[3] Nancy G. Leveson, *Engineering a Safer World*, MIT Press (2012).