



IAMP Safety Critical Systems Working Papers

MDG Profile for CAST

UML Extension for CAST (Causal Analysis based on STAMP)

Version: 1
Date: 12.09.2017
Authors: Benjamin Contreras, Sven Stefan Krauss

Copyright © 2017 The Authors. Published by ZHAW digitalcollection.

This is an open access article under the CC BY-NC-ND license (<http://creativecommons.org/licenses/by-nc-nd/4.0/>).

Peer-review under responsibility of the Institute of Applied Mathematics and Physics – Safety Critical Systems.

Authors

Benjamin Contreras benjamin.contreras@zhaw.ch

Peer Review

Sven Stefan Krauss svenstefan.krauss@zhaw.ch

Editor

Dr. Christian Hilbes christian.hilbes@zhaw.ch

Quality Assurance

Responsible for IAMP Safety Critical Systems open access publications

Sven Stefan Krauss svenstefan.krauss@zhaw.ch

Internal ID

304034

Table of Content

| | | |
|-----|--|----|
| 1 | Introduction | 4 |
| 2 | MDG Profile for CAST..... | 6 |
| 2.1 | Overview | 6 |
| 2.2 | Analysis Fundamentals Diagram | 8 |
| 2.3 | Hierarchical Control Structure Diagram | 11 |
| 2.4 | CAST Analysis Diagram..... | 17 |
| 2.5 | Proximal Event Chain Diagram | 22 |
| 2.6 | Tagged Values | 24 |
| 2.7 | Rule Sets..... | 24 |
| 3 | Using MDG Profile for CAST | 26 |
| 3.1 | Installation | 26 |
| 3.2 | Diagrams | 26 |
| 3.3 | Toolbox..... | 28 |
| 3.4 | Properties Dialog..... | 29 |
| 4 | Appendix | 31 |
| 4.1 | References | 31 |

1 Introduction

Causal Analysis based on STAMP (CAST) is a modern safety analysis technique developed by Leveson [1] which is based on the accident model Systems-Theoretic Accident Model and Processes (STAMP) [2].

We developed an extension called SAHRA (STPA based Hazard and Risk Analysis) [11] for Sparx Systems Enterprise Architect (EA) [12] (Figure 1).

EA is a popular commercial UML/SysML modeling tool which can be used for requirements engineering, system and software design. The corporate edition of EA provides multi user support with security permission system, scripting and automation API, SQL searches, configuration management integration, report generation and modeling functionality.

SAHRA includes a MDG (Model Driven Generation) profile for STPA based on EA's MDG technology [13] to provide additional diagram types, toolboxes, UML profiles, patterns and templates for STPA modeling. The SAHRA extension provides a context sensitive object browser for comfortable editing and special editors for performing STPA Step 1 and Step 2.

The SAHRA extension for Sparx Systems Enterprise Architect includes a domain specific language, also called DSL, profile which uses EA's MDG technology as a base to provide new diagram types, toolboxes, UML profiles, patterns and templates for STPA [17] - [18].

This already existing Profile called MDG Profile for STPA developed by Krauss et al. [19] bases as the foundation for the development of the MDG Profile for CAST. The also formalized concepts of safety-guided design with STPA, by Rejzek et al. [3], are used as a basis for the documentation of the CAST profile.

The goal of the MDG Profile for CAST is to integrate CAST seamlessly into SAHRA. This document describes all items of the MDG Profile for CAST.

This document formalizes the concepts of safety-guided design with CAST mentioned in [3] by providing an overview of the diagrams, elements and connectors that are defined in the MDG Profile for CAST. While the implementation of the MDG profile itself is specific to EA, the concepts and the approach to extend UML with a specific profile for CAST is generic. The purpose of this document is therefore twofold:

1. This document seeks to provide a comprehensive definition towards a domain specific modeling language for Causal Analysis based on STAMP (CAST), including the definition of terms, elements and graphical representation;
2. It aims to document best practices with CAST and software tool SAHRA.

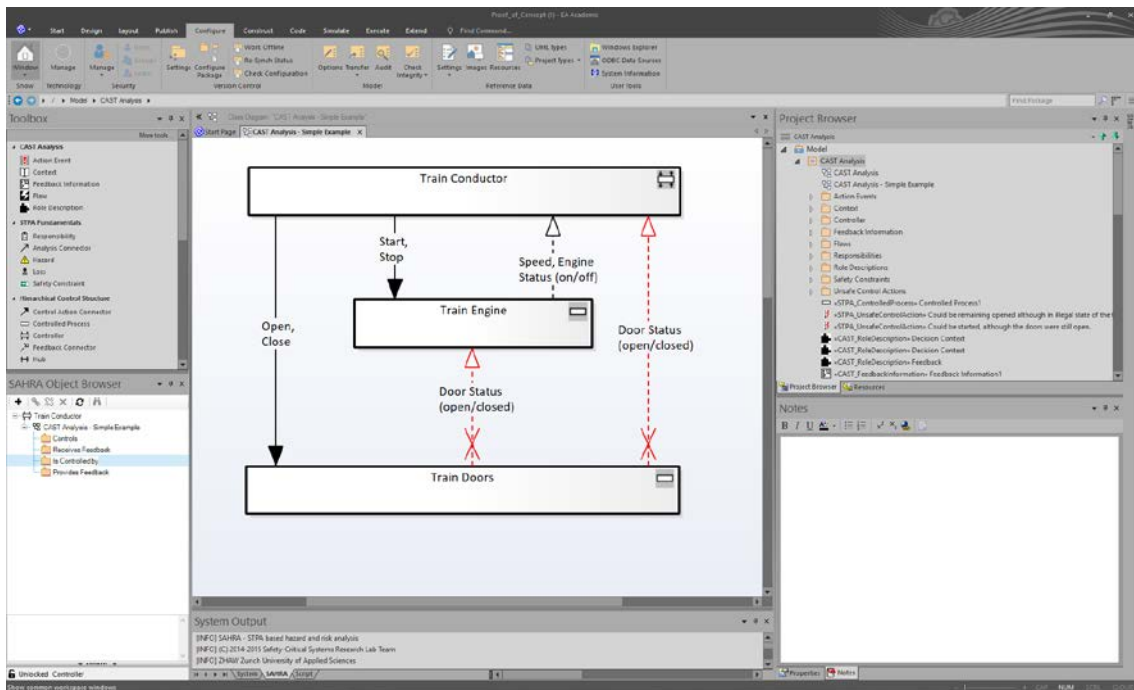


Figure 1: SAHRA – CAST based Hazard and Risk Analysis: an extension for Sparx Systems Enterprise Architect¹ to integrate CAST with a UML/SysML modeling environment.

¹ Sparx Systems, Enterprise Architect, MDG Integration, and MDG Technology are trademarks or registered trademarks of Sparx Systems Pty Ltd., Creswick, Australia.

2 MDG Profile for CAST

2.1 Overview

The MDG Profile for CAST tailors UML to CAST with new diagram types, new element types and new connector types. A model (in this context) is a set of diagrams with elements which are connected by connectors. Connectors define a relationship between two elements. A connector has a source element and a target element. The visual style of a connector defines its meaning which can be altered by applying stereotypes.

An element is a node on a diagram. Each element has at least these standard properties²:

- Name – name of the element;
- Notes – long description of the element;
- Stereotype – Type of the element.

All elements in the MDG Profile for CAST have additional properties (tagged values):

- ID – user defined text to identify the element;
- Context – user defined text to document the context of the element, for example diagram detail level;
- ParentID – user defined text to specify the parent element ID.

Some elements have special properties (tagged values):

- Date, Time – user defined to specify date and time for action events.
- IsMissing – user defined Boolean to define if a control action or feedback is missing.

To provide the possibility to extend the MDG Profile for CAST, extensions can be used. We included in this document the extensions which were useful in our case studies. To document the items which are available in the MDG Profile for CAST, a table according to table 1 is used in this document.

Table 1: Item's documentation scheme used in this document.

| Property | Description |
|-----------------------|--|
| Metatype | Name of the item |
| Purpose | Purpose of item |
| Extends | UML item on which the new item is based on |
| Stereotype | Stereotype of the item |
| Alternative Name(s) | Alternative names which can be found in STAMP/CAST related documents and presentations |
| Visual Representation | Graphical example of the new item |

Figure 2 provides an overview of the diagrams, connectors, elements and extensions which are available in the MDG Profile for CAST.

² There are more properties available, but are not further used in this document.

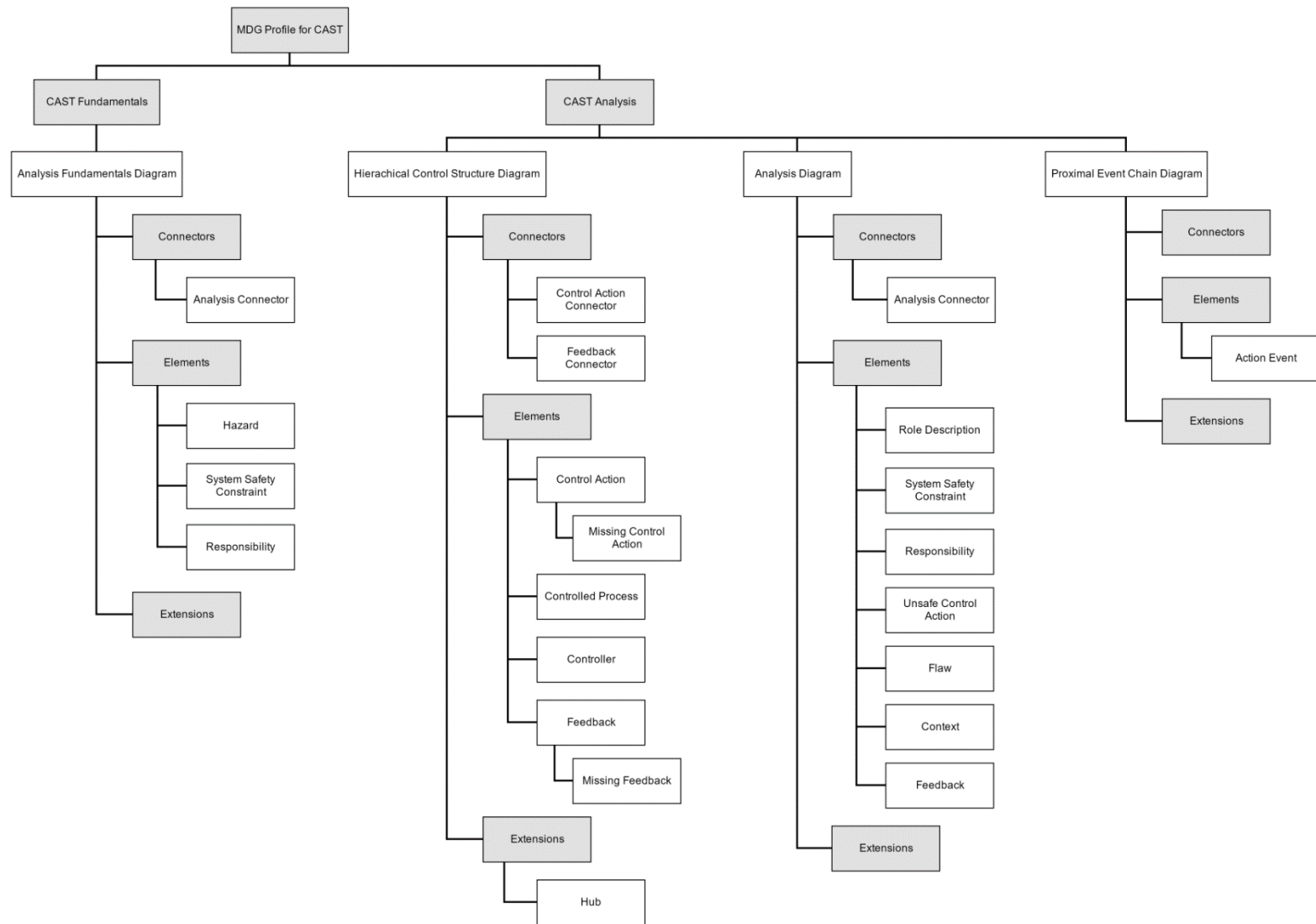


Figure 2: MDG Profile for CAST - Overview of available Diagrams, Connectors, Elements and Extensions.

2.2 Analysis Fundamentals Diagram

2.2.1 Purpose

This diagram is used to describe analysis fundamentals like: Hazards, its items and their connections (Figure 3). Valid links for Analysis Connector and related elements are defined in chapter 2.7.1.

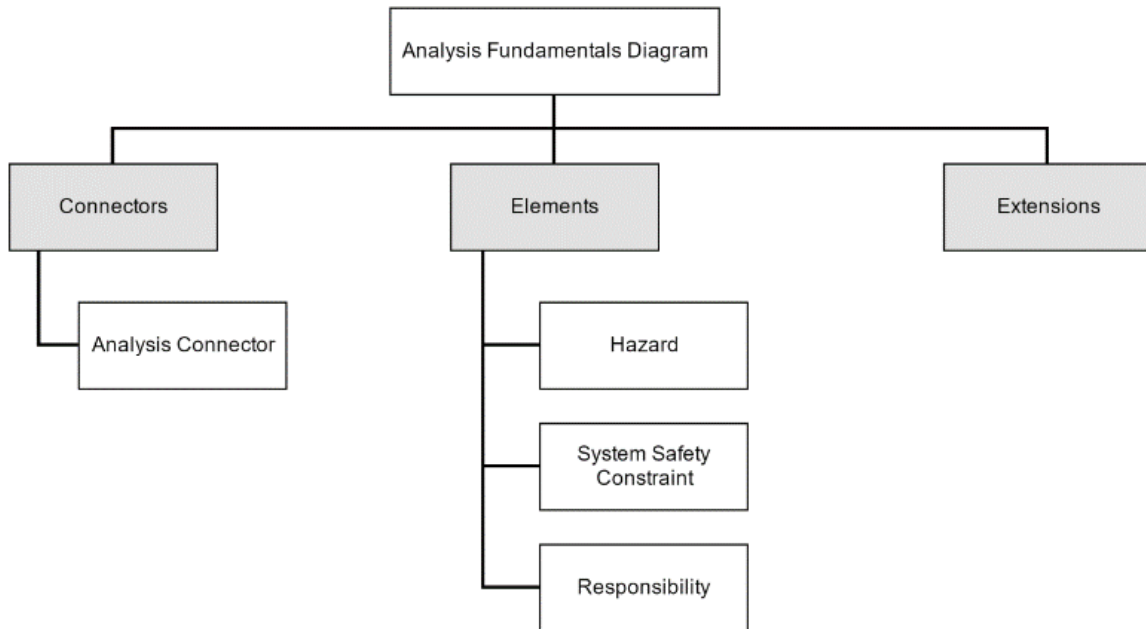


Figure 3: CAST Analysis Fundamentals Diagram Overview.


2.2.2 Example Diagram



Figure 4: Example Analysis Fundamentals.


2.2.3 Connectors

2.2.3.1 Analysis Connector

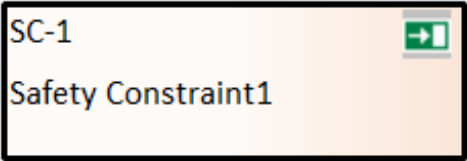
| Property | Description |
|-----------------------|---|
| Metatype | Analysis Connector |
| Purpose | Defines the relationship between two analysis elements. The direction of the connector defines the relationship, normally a potential path from cause to consequence. |
| Extends | UML::Association |
| Stereotype | STPA_AnalysisConnector |
| Alternative Name(s) | n/a |
| Allowed Connections | See 2.7.1 |
| Visual Representation |  |

2.2.4 Elements

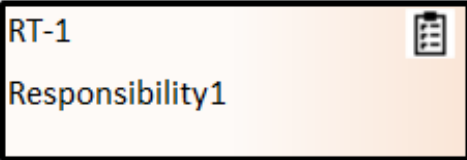
2.2.4.1 Hazard

| Property | Description |
|-----------------------|---|
| Metatype | Hazard |
| Purpose | Represents "System <u>state</u> / set of conditions that together with particular set of worst-case environmental conditions will lead to accident" [1, p. 183] |
| Extends | UML::Class |
| Stereotype | STPA_Hazard |
| Alternative Name(s) | System Level Hazard |
| Visual Representation |  |

2.2.4.2 Safety Constraint

| Property | Description |
|-----------------------|---|
| Metatype | Safety Constraint |
| Purpose | Describes a constraint that is enforced on a system or process. |
| Extends | UML::Class |
| Stereotype | STPA_SafetyConstraint |
| Alternative Name(s) | System Safety Constraint |
| Visual Representation |  |

2.2.4.3 Responsibility

| Property | Description |
|-----------------------|---|
| Metatype | Responsibility |
| Purpose | Describes a responsibility of the controller. |
| Extends | UML::Class |
| Stereotype | CAST_Responsibility |
| Alternative Name(s) | Safety Responsibility |
| Visual Representation |  |

2.3 Hierarchical Control Structure Diagram

2.3.1 Purpose

The Hierarchical Control Structure Diagram is used to create a functional, hierarchical model of the system under consideration with Controllers, Controlled Processes, (Missing) Control Actions and Feedback (Figure 4) as a foundation for the consequent analysis steps of CAST.

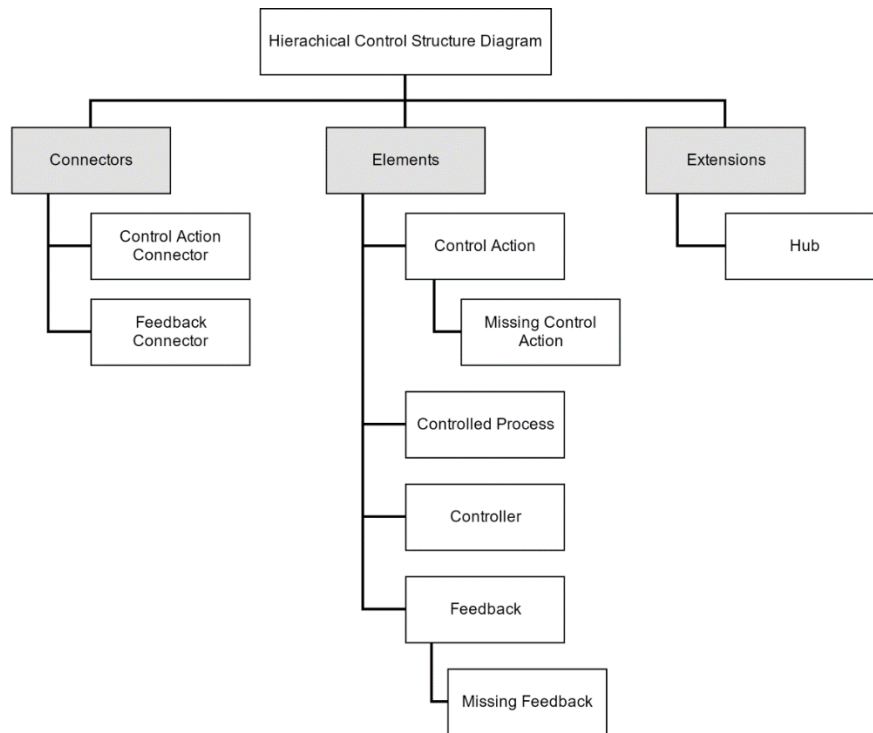


Figure 5: Hierarchical Control Structure Diagram Overview.

2.3.2 Example Diagram

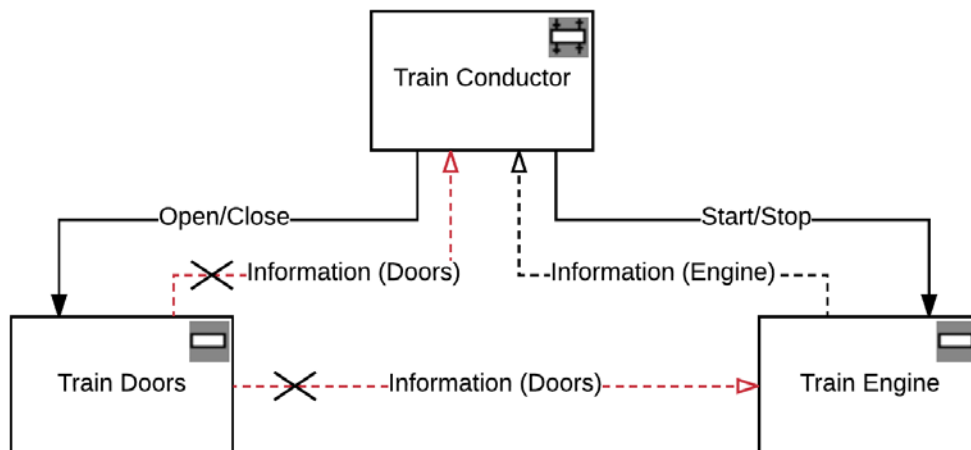




Figure 6: Example Hierarchical Control Structure.

2.3.3 Connectors

2.3.3.1 Control Action Connector

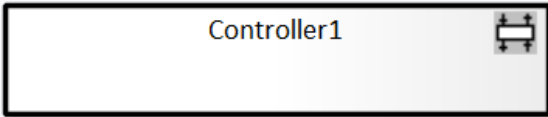
| Property | Description |
|-----------------------|---|
| Metatype | Control Action Connector |
| Purpose | Provides a route for Control Actions. A Control Action Connector can host a number of Control Actions as conveyed items. |
| Extends | UML::InformationFlow |
| Stereotype | STPA_ControlActionConnector |
| Alternative Name(s) | n/a |
| Valid connections | See 2.7.2 |
| Visual Representation |  |

2.3.3.2 Feedback Connector

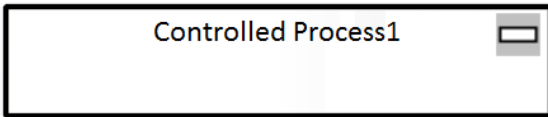
| Property | Description |
|-----------------------|---|
| Metatype | Feedback Connector |
| Purpose | Provides a route for Feedback. A Feedback Connector can host a number of Feedback items as conveyed items. |
| Extends | UML::InformationFlow |
| Stereotype | STPA_FeedbackConnector |
| Alternative Name(s) | n/a |
| Valid connections | See 2.7.3 |
| Visual Representation |  |

2.3.4 Elements

2.3.4.1 Controller

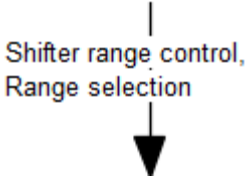
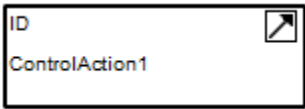
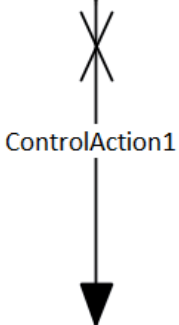
| Property | Description |
|-----------------------|--|
| Metatype | Controller |
| Purpose | A controller affects the state of the system by providing control actions based on process model and feedback. A controller can be an automated controller or a human controller. A controller can provide and receive control actions and provide and receive feedback. |
| Extends | UML::Class |
| Stereotype | STPA_Controller |
| Alternative Name(s) | n/a |
| Visual Representation |  |

2.3.4.2 Controlled Process

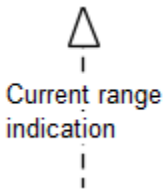

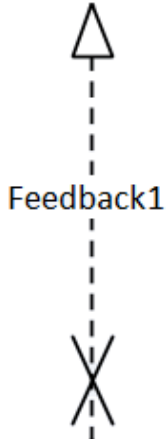
| Property | Description |
|-----------------------|--|
| Metatype | Controlled Process |
| Purpose | Represent the controlled process of the system under consideration. A controlled process can receive control actions and provide feedback. |
| Extends | UML::Class |
| Stereotype | STPA_ControlledProcess |
| Alternative Name(s) | Dynamic System State |
| Visual Representation |  |

2.3.5 Connector Conveyed Items

2.3.5.1 Control Action


| Property | Description |
|---|--|
| Metatype | Control Action |
| Purpose | Represents a control action to change the state of the system. |
| Extends | UML::Class |
| Stereotype | STPA_ControlAction |
| Alternative Name(s) | Control Action |
| Remarks | Control Actions can only be linked with a Control Action Connector. When multiple Control Actions are linked with the same Control Action Connector they are shown separated by commas. |
| Visual Representation | Hierarchical Control Structure Diagram and Step 2 Control Loop Diagram  |
| | other diagrams  |
| Tagged Value | IsMissing::Boolean |
| Visual Representation with Tagged Value |  |

2.3.5.2 Feedback

| Property | Description |
|---|--|
| Metatype | Feedback |
| Purpose | Represents a system information. |
| Extends | UML::Class |
| Stereotype | STPA_Feedback |
| Alternative Name(s) | Feedback, Control Feedback |
| Remarks | Feedback can only be linked with a Feedback Connector. When multiple Feedback items are linked with the same Feedback Connector they are shown separated by commas. |
| Visual Representation | Hierarchical Control Structure Diagram and Step 2 Control Loop Diagram  |
| | other diagrams  |
| Tagged Value | IsMissing::Boolean |
| Visual Representation with Tagged Value |  |

2.3.6 Extensions

2.3.6.1 Hub

| Property | Description |
|-----------------------|--|
| Metatype | Hub |
| Purpose | <p>The hub element is an auxiliary element to route Control Actions and Feedback. It can be used:</p> <ul style="list-style-type: none">• to split Control Action Connectors into a number of Control Action Connectors,• to split Feedback Connectors into a number of Feedback Connectors,• to join Control Action Connectors,• to join Feedback Connectors,• to maintain the consistency of Control Actions and Feedback between different diagram representations of one HCS |
| Extends | UML::Fork/Join |
| Stereotype | STPA_Hub |
| Alternative Name(s) | Bus, Node |
| Visual Representation |  |

2.4 CAST Analysis Diagram

2.4.1 Purpose

The CAST analysis diagram is used to design the CAST analysis as a model. It consists of the analysis connector, which is also used in the STPA module and all the elements needed to describe a CAST analysis.

The first level of the model is the controller which is being described. The next step is to define the Role Descriptions; these are set by default, which can be seen in the example diagram (Figure 8). Now the elements which are found in the CAST analysis can be added under the Role Descriptions.

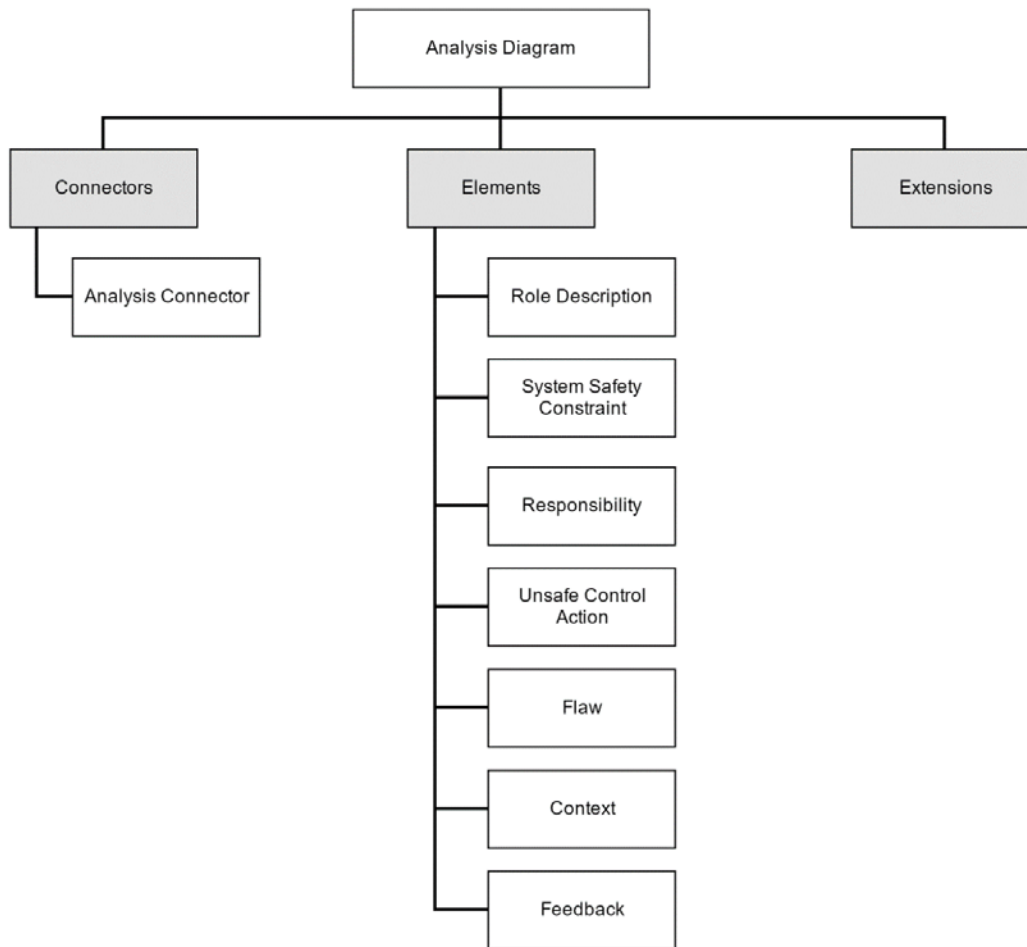


Figure 7: Step 1 Analysis Diagram Overview.

2.4.2 Example Diagram

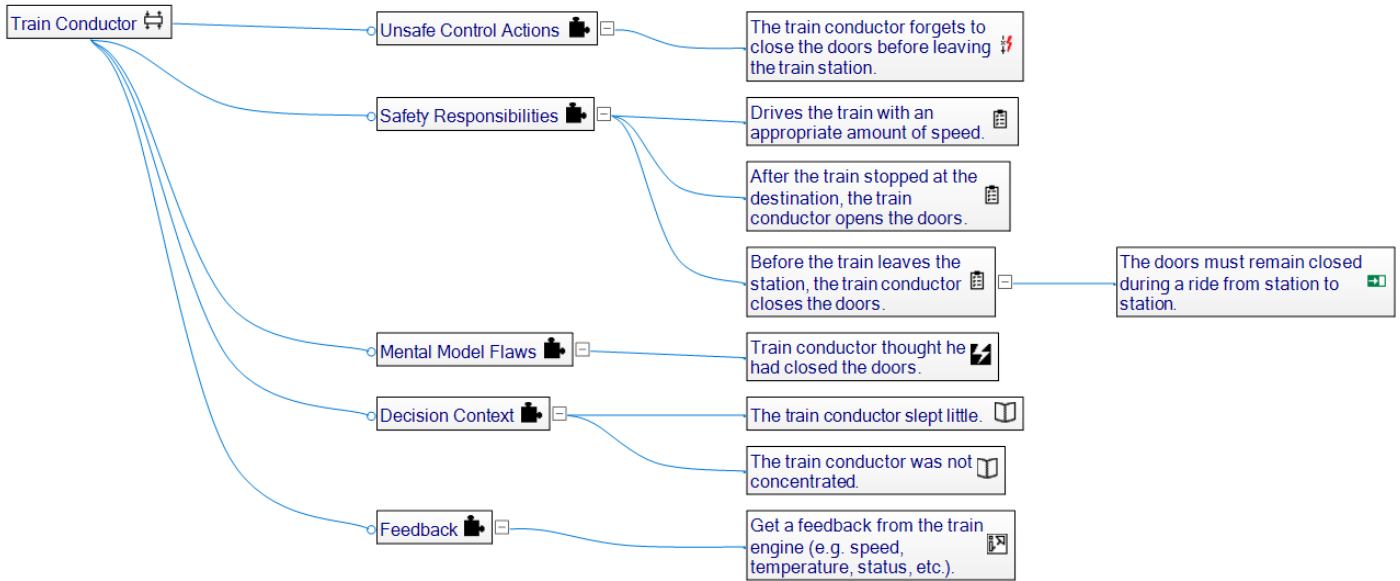


Figure 8: Example CAST Analysis Diagram as shown in SAHRA's analysis editor.

2.4.3 Connectors



The diagram uses the Analysis Connector as defined in chapter 2.2.3.

2.4.4 Elements


2.4.4.1 Role Description

| Property | Description |
|-----------------------|---|
| Metatype | Role Description |
| Purpose | It describes a role of a component in the safety control structure. |
| Extends | UML::Class |
| Stereotype | CAST_RoleDescription |
| Alternative Name(s) | Controller Description |
| Visual Representation | |

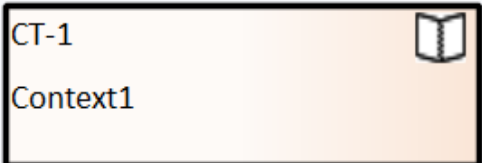
2.4.4.2 Unsafe Control Action

| Property | Description |
|-----------------------|---|
| Metatype | Unsafe Control Action |
| Purpose | Represents a (potential) Unsafe Control Action, which typically leads to Unsafe Process State, Unsafe Process Reaction or Hazard. |
| Extends | UML::Class |
| Stereotype | STPA_UnsafeControlAction |
| Alternative Name(s) | n/a |
| Visual Representation | <p>SAHRA Analysis View:</p>  <p>Other diagrams:</p>  |

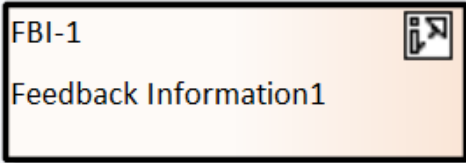
2.4.4.3 Flaw

| Property | Description |
|-----------------------|---|
| Metatype | Flaw |
| Purpose | Describes a flaw in the process/mental model. |
| Extends | UML::Class |
| Stereotype | CAST_Flaw |
| Alternative Name(s) | Model Flaw |
| Visual Representation |  |

2.4.4.4 Context

| Property | Description |
|-----------------------|---|
| Metatype | Context |
| Purpose | Describes a context of the controller. |
| Extends | UML::Class |
| Stereotype | CAST_Context |
| Alternative Name(s) | Decision Context |
| Visual Representation |  |

2.4.4.5 Feedback Information

| Property | Description |
|-----------------------|---|
| Metatype | Feedback Information |
| Purpose | Describes controller feedback information. |
| Extends | UML::Class |
| Stereotype | CAST_FeedbackInformation |
| Alternative Name(s) | n/a |
| Visual Representation |  |

2.4.4.6 Responsibility

The CAST Analysis Diagram uses the same Responsibility described in chapter 2.2.4.3.

2.4.4.7 System Safety Constraint

The CAST Analysis Diagram uses the same System Safety Constraint described in chapter 2.2.4.2.

2.5 Proximal Event Chain Diagram

2.5.1 Purpose

In the CAST analysis, a Proximal Event Chain can clarify the whole occurrence that happened. This Event Chain Diagram is another possibility designing the Event Chain as an Activity Diagram deviation.

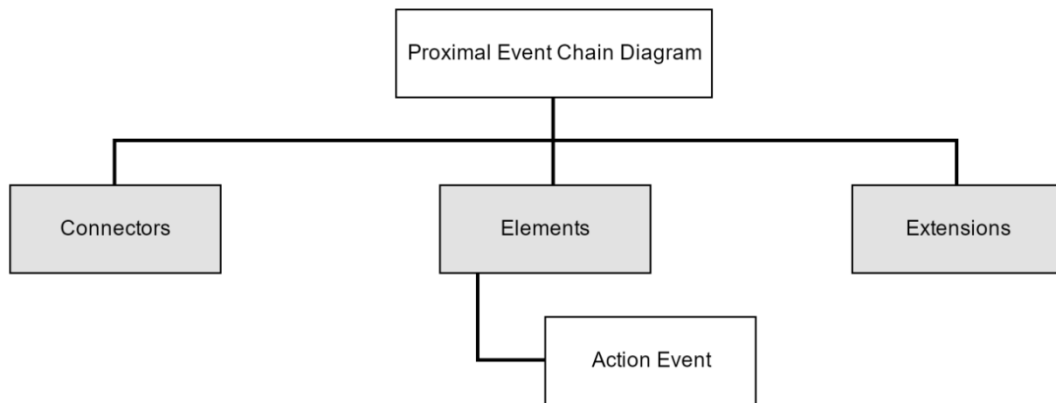


Figure 9: Proximal Event Chain Diagram Overview.

2.5.2 Example Diagram

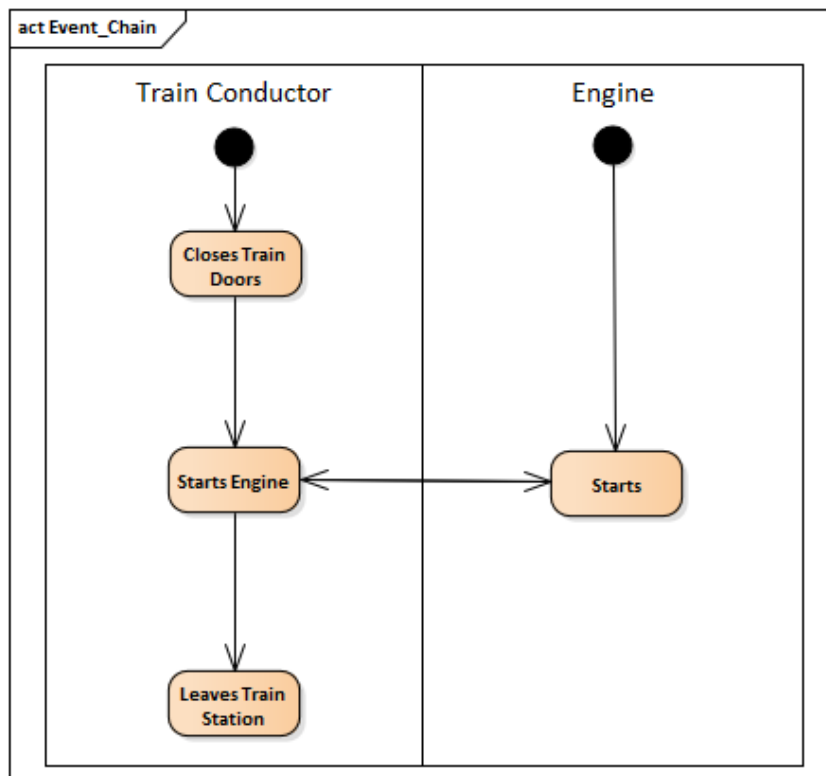


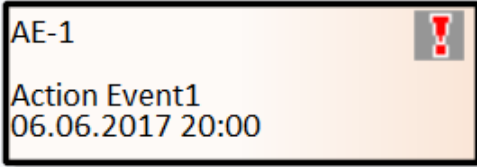
Figure 10: Example Proximal Event Chain Diagram.

2.5.3 Connectors

The Proximal Event Chain Diagram uses the same Analysis Connector described in chapter 2.2.3.1.

2.5.4 Elements

2.5.4.1 Action Event

| Property | Description |
|-----------------------|---|
| Metatype | Action Event |
| Purpose | Describes an Event/Action at a given point in time. |
| Extends | UML::Class |
| Stereotype | CAST_ActionEvent |
| Alternative Name(s) | Event |
| Tagged Value | <ul style="list-style-type: none">• Date::DateTime• Time::Custom |
| Visual Representation |  |

2.6 Tagged Values

For the new defined elements of the CAST Analysis “special” Tagged Values are needed. These can be scripted [16]. The following Tagged Values have been defined:

| Name | Type | Description | Script |
|-----------|----------|---|---|
| Time | Custom | Time is used to define the time of occurrence of an ActionEvent. | Type=Custom; Mask=DD DD; Template=__:_; |
| Date | DateTime | Date is used to define the date of occurrence of an ActionEvent. | Type=DateTime; |
| IsMissing | Boolean | Is Missing is used to declare a Control Action or Feedback as missing | Type=Boolean; Default=False; |

2.7 Rule Sets

2.7.1 Analysis Connector Rule Set

| Valid connections for Analysis Connector | | | | | | | | | | |
|--|-----------------------|------------|------------------|----------------|----------------|-----------------------|---------|----------|--------|------|
| | | Target | | | | | | | | |
| | | Controller | Role Description | Responsibility | Control Action | Unsafe Control Action | Context | Feedback | Hazard | Flaw |
| Source | Controller | ✗ | ✓ | ✗ | ✓ | ✗ | ✗ | ✗ | ✓ | ✗ |
| | Role Description | ✗ | ✗ | ✓ | ✗ | ✓ | ✓ | ✓ | ✗ | ✓ |
| | Responsibility | ✗ | ✗ | ✗ | ✗ | ✓ | ✗ | ✗ | ✗ | ✗ |
| | Control Action | ✓ | ✗ | ✗ | ✗ | ✓ | ✗ | ✗ | ✗ | ✗ |
| | Unsafe Control Action | ✗ | ✗ | ✓ | ✗ | ✗ | ✗ | ✗ | ✓ | ✗ |
| | Context | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ |
| | Feedback | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ |
| | Hazard | ✓ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✓ | ✗ |
| | Flaw | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ |

2.7.2 Control Action Connector Rule Set

| Valid connections for Control Action Connector | | | | |
|--|--------------------|------------|--------------------|-----|
| | | Target | | |
| | | Controller | Controlled Process | Hub |
| Source | Controller | ✓ | ✓ | ✓ |
| | Controlled Process | ✗ | ✗ | ✗ |
| | Hub | ✓ | ✓ | ✓ |

2.7.3 Feedback Connector Rule Set

| Valid connections for Feedback Connector | | | | |
|--|--------------------|------------|--------------------|-----|
| | | Target | | |
| | | Controller | Controlled Process | Hub |
| Source | Controller | ✓ | ✗ | ✓ |
| | Controlled Process | ✓ | ✗ | ✓ |
| | Hub | ✓ | ✓ | ✓ |

3 Using MDG Profile for CAST

3.1 Installation

The MDG profile for CAST is automatically installed when the SAHRA [11] extension is installed. For more information, please refer to the SAHRA documentation.

To check if the profile is loaded navigate to **Configure | Manage Technology...**. The MDG Technologies dialog should have an entry **STPA** (Figure 12).

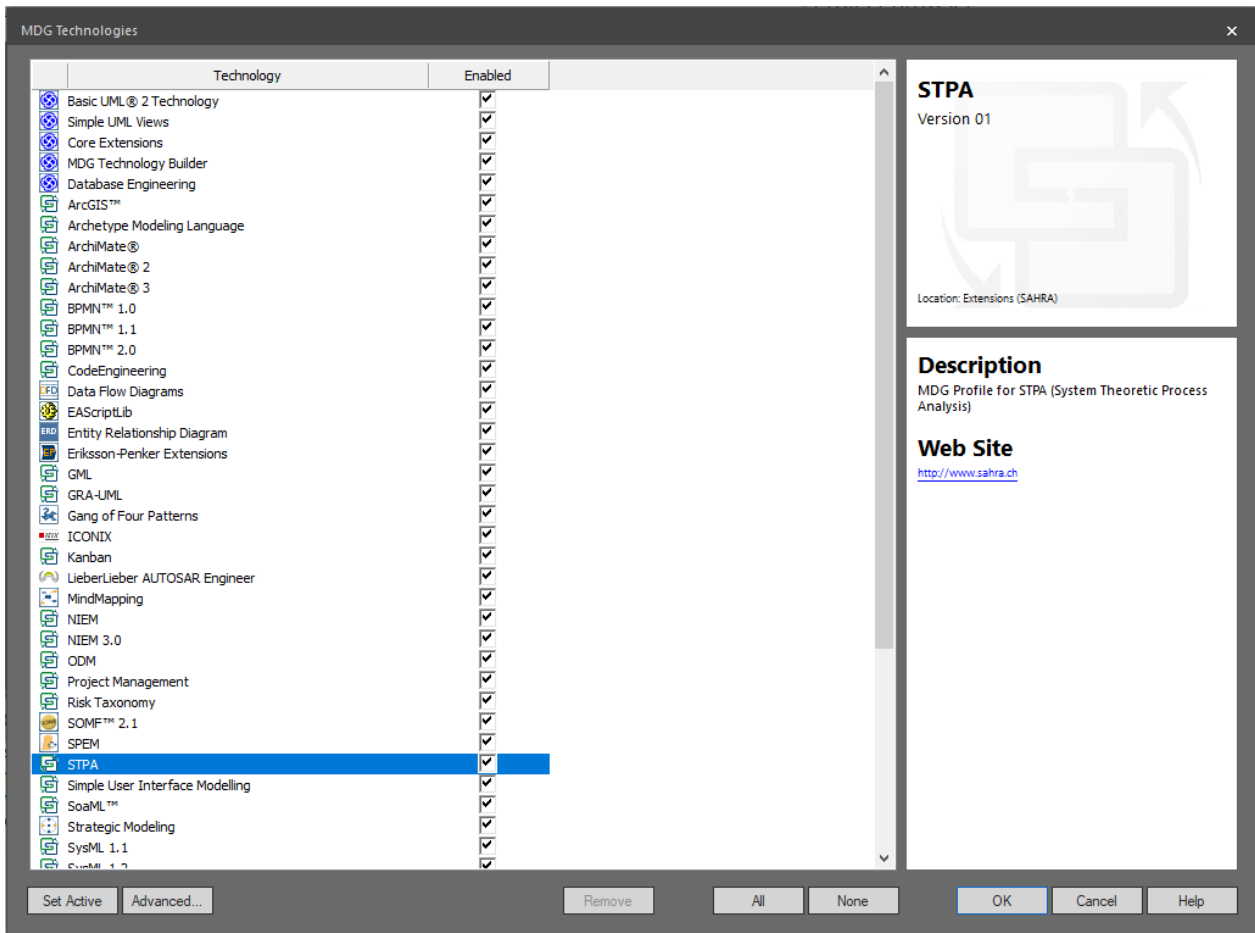


Figure 11: MDG Technologies Dialog.

3.2 Diagrams

The MDG Profile for STPA provides four new diagram types:

- CAST Analysis
- STPA Analysis Fundamentals
- STPA Hierarchical Control Structure
- STPA Step 1 Analysis^{*)}
- STPA Step 2 Analysis^{*)}

^{*)} These diagrams are only required when the MDG Profile for CAST is used without the SAHRA extension. The diagrams are not needed by the SAHRA extension editors for the CAST Analysis View.

To create a new CAST diagram, select **STPA** in the New Diagram dialog and select **STPA** under **Select From:** and the diagram type for the new diagram under **Diagram Types:**

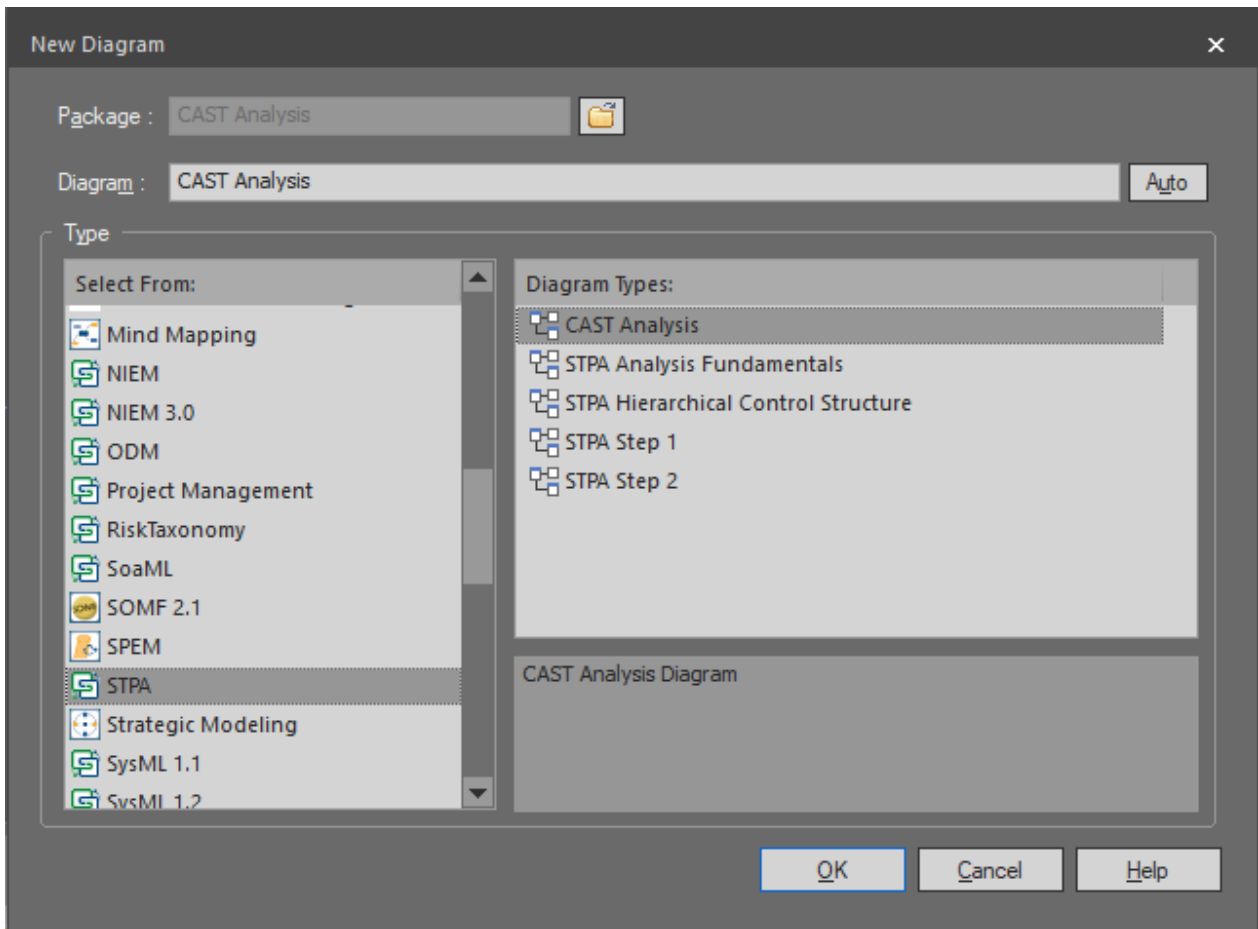


Figure 12: New Diagram Dialog.

3.3 Toolbox

When a CAST diagram is created, the STPA toolbox is shown (Figure 14). In case it is not shown, please click on **More tools...** and select **STPA** from the list.

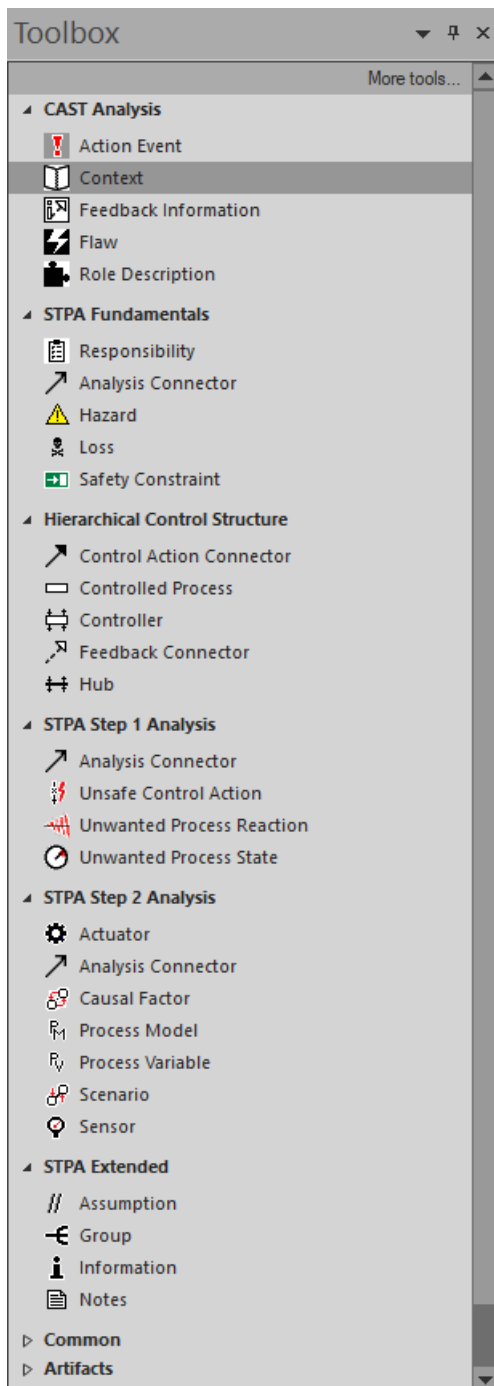


Figure 13: STPA Toolbox.

3.4 Properties Dialog

The properties dialog can be opened with a double click on an element or with **Properties...** from a context menu. The user can enter name, notes as a long description and can edit other properties (Figure 15). To show special tagged values for the element, Register **STPA** must be selected on the right hand side (Figure 16).

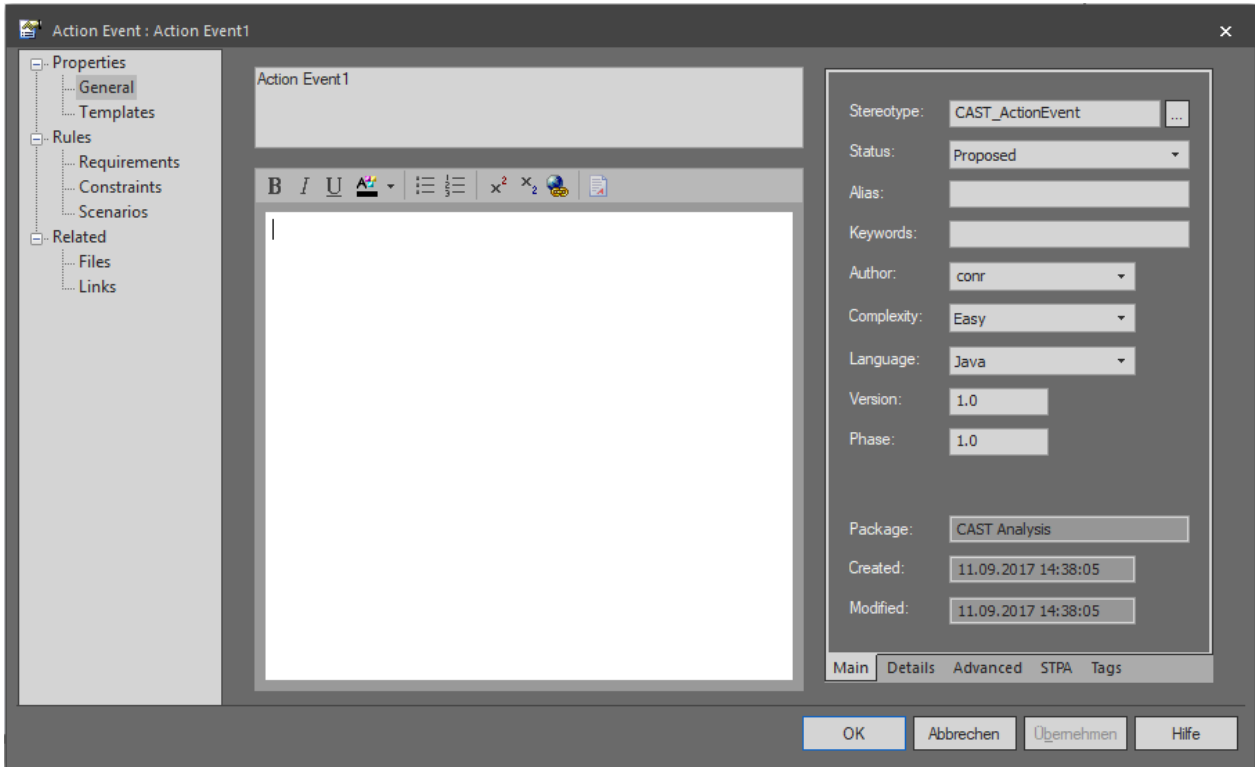


Figure 14: Properties dialog for a Action Event showing general properties like Name, Notes, Stereotype and other metadata.

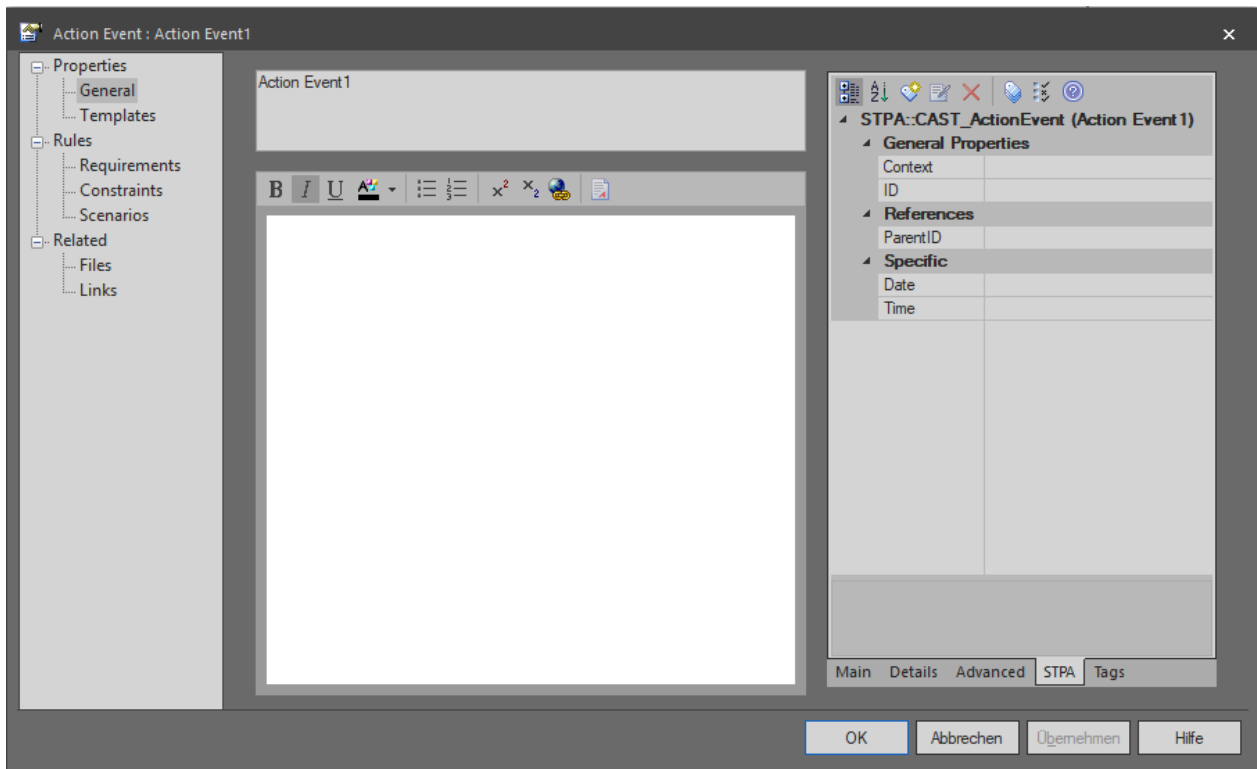


Figure 15: Properties dialog for an Action Event. Notes field is empty. Register STPA is selected to show special tagged values for STPA/CAST.

4 Appendix

4.1 References

1. Leveson, N.G., *Engineering a safer world: Systems thinking applied to safety*. 2012, Cambridge MA, USA: MIT Press.
2. Leveson, N.G., *A new accident model for engineering safer systems*. *Safety Science*, 2004. **42**(4): p. 237-270.
3. Rejzek, M., C. Hilbes, and S.S. Krauss, *Safety Driven Design with UML and STPA*, in *STAMP Workshop 2015*. 2015: MIT, Boston.
4. Object Management Group, *OMG Unified Modeling Language (OMG UML™) Version 2.5. OMG Document Number: formal/15-03-01*. 2015.
5. Object Management Group, *OMG Systems Modeling Language (OMG SysML™) Version 1.4. OMG Document Number: formal/2015-06-03*. 2015.
6. Seidl, M., et al., *UML@classroom: An introduction to object-oriented modeling*. 2015: Springer International Publishing.
7. Weilkens, T., *Systems engineering with SysML/UML: modeling, analysis, design*. 2011: Morgan Kaufmann.
8. Holt, J. and S. Perry, *SysML for Systems Engineering: A Model-Based Approach*. 2013: Institution of Engineering and Technology.
9. Wieringa, R., *Design methods for reactive systems : Yourdon, StateMate and the UML*. 2003: San Francisco : Morgan Kaufmann. 457 S.
10. Antoine, B., *Systems Theoretic Hazard Analysis (STPA) applied to the risk review of complex systems: an example from the medical device industry*. 2013, Massachusetts Institute of Technology.
11. Safety-Critical Systems Research Lab Team of ZHAW Zurich University of Applied Sciences. *SAHRA - STPA based Hazard and Risk Analysis*. [cited 2015 01.08.2015]; Available from: <http://www.sahra.ch>.
12. Sparx Systems Pty Ltd. *Enterprise Architect - UML Modeling and Lifecycle Tool Suite*. 2015 [cited 2015 01.08.2015]; Available from: <http://www.sparxsystems.com/>.
13. Sparx Systems Pty Ltd. *Model Driven Generation (MDG) Technologies*. 2015 27.07.2015]; Available from: http://www.sparxsystems.com.au/resources/mdg_tech/.
14. Sgueglia, J., *Managing Design Changes using Safety-Guided Design for a Safety-Critical Automotive System. MIT Master's Thesis, June 2015*. 2015: MIT, Boston.
15. Thomas, J., *Extending and automating a systems-theoretic hazard analysis for requirements generation and analysis*, in *PhD Thesis, Engineering Systems Division*. 2013: MIT, Boston.
16. Sparx Systems, 'Predefined Structured Types [Enterprise Architect User Guide]', 2017. [Online]. Available: http://www.sparxsystems.com/enterprise_architect_user_guide/10/extending_uml_models/predefinedtaggedvaluetypes.html. [Accessed: 07-Jun-2017].
17. Sparx Systems, 'MDG Technologies - Creating [Enterprise Architect User Guide]', 2017. [Online]. Available: http://www.sparxsystems.com/enterprise_architect_user_guide/10/extending_uml_models/mdgtechnologies_2.html. [Accessed: 05-Jun-2017].
18. Sparx Systems, 'Create UML Profiles [Enterprise Architect User Guide]', 2017. [Online]. Available: http://www.sparxsystems.com/enterprise_architect_user_guide/10/extending_uml_models/workingwithprofiles.html. [Accessed: 30-May-2017].
19. S. S. Krauss, M. Rejzek, and M. U. Reif, 'Towards a modeling language for Systems-Theoretic Process Analysis (STPA) : Proposal for a domain specific language (DSL) for model driven Systems-Theoretic Process Analysis (STPA) based on UML', Dec. 2016.