(ITEA 3 – 17003)

PANORAMA
Boosting Design Efficiency for Heterogeneous^3 Systems

Deliverable: D 6.2
Traceability Tool

Work Package: 6
Design flow and traceability

Task: 6.3
Standards-compliant traceability tools and practices

Document Type:	Deliverable (Software)	**Classification:**	Public
Document Version:	1.0	**Contract Start Date:**	2019-04-01
Document Preparation Date:	31.01.2021	**Duration:**	2022-03-31

ITEA3
INFORMATION TECHNOLOGY FOR EUROPEAN ADVANCEMENT

EUREKA

Contents

1 Introduction 1

2 Eclipse Capra 2

3 Eclipse Capra 0.8.0 3

4 Eclipse Capra 0.8.1 5

5 Impact of the Changes on Safety Practices 6

List of Figures

3.1 The Sunburst and Matrix views introduced in Eclipse Capra 0.8.0 4

5.1 A traceability matrix showing how hazards are addressed by safety goals. . . . 6

5.2 A sunburst view of a failure and its linked artifacts, showing that the failure is connected to a hazard and the hazard is connected to a safety goal. 7

List of Tables

1 Introduction

This deliverable summarizes the changes that Eclipse Capra has received since the start of the Panorama project. In particular, it describes the changes made in two releases:

Eclipse Capra 0.8.0 released on July 5, 2020

Eclipse Capra 0.8.1 released on January 4, 2021

Based on a description of Eclipse Capra in Chapter 2, this document reviews the most important changes in version 0.8.0 in Chapter 3 and the ones in version 0.8.1 in Chapter 4. Finally, an overview of the impact of these changes on safety analysis practices is presented in Chapter 5.

2 Eclipse Capra

Eclipse Capra is a traceability management tool. It allows the creation of trace links between arbitrary artefacts, provides features to edit them and keep them consistent, and visualise the relationships between them. This includes traceability matrices and graph visualisations that are helpful for reporting and change impact analysis.

In essence, Eclipse Capra allows the creation of trace links between arbitrary artefacts, as long as an adapter for these artefacts is available. Eclipse Capra currently natively supports elements from UML, SysML, AADL, EAST-ADL, as well as ReqIF models created in, e.g., Eclipse Papyrus, Eclipse EATOP, and ProR. Furthermore, adapters for test case executions managed by a continuous integration server like Hudson or Jenkins can be traced to. There is also support for source code files supported by the Eclipse Platform (e.g., Java, C, Python) and tasks from an issue tracking system supported by Eclipse Mylyn. External artefacts for which the Eclipse Platform does not offer built-in support can also be linked if a fitting adapter is provided. Built-in capabilities allow linking to Office documents and documents hosted by Google Docs. A generic file handler and a handler for EMF models also provide capabilities for formats that currently have no native support.

Once trace links are established, Eclipse Capra offers features to manage them. If a model element that is linked to is moved, e.g., Eclipse Capra will notify the user and allow changing the link accordingly. The same support is given for model elements that are deleted or renamed. Quick fixes are available to fix most isses in a semi-automatic fashion.

Eclipse Capra also offers a visualisation of the trace links that allows developers to traverse the relationships established through the links and understand how the different artefacts are connected. This is helpful when assessing the impact a change has (e.g., which design artefacts need to be adapted when a requirement has changed?) or when trying to understand how the design artefacts in a complex development project are connected. In addition, Eclipse Capra can display traceability matrices, as requested by standards like ISO 26262.

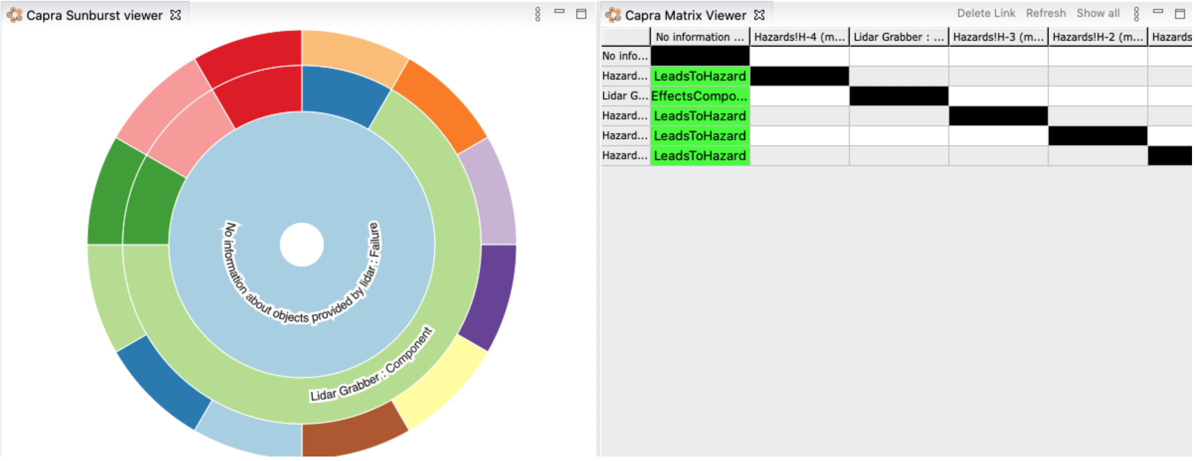
The tool is highly extensible. The meta-model used for the traceability links can easily be adapted to a specific end-user's needs. Eclipse Capra's modular architecture allows exchanging the persistence, the visualisation, and the management modules easily. New adapters for additional artefacts can easily be added without re-compilation. This allows end-users to customise almost every aspect of the tool if needed. At the same time, we provide sensible defaults that will allow the majority of users to use Capra out of the box without extensive configuration.

Eclipse Capra is available for download at `https://www.eclipse.org/capra`.

3 Eclipse Capra 0.8.0

The focus of the 0.8.0 release of Eclipse Capra was to add additional traceability link visualisation techniques and to increase operability with Eclipse APP4MC. For this purpose, the following new features were included into the tool:

- Added ability to create trace links from and to Amalthea models. While Amalthea models had been supported through the generic EMF handler before, this new feature now adds the ability to define that trace links should specifically include elements from Amalthea models. The change also makes it possible to navigate directly to the Amalthea model from within the Eclipse Capra visualisations.
- Added dedicated traceability matrix view. It is a common request to produce a traceability matrix that shows how one set of artifacts (e.g., requirements) are traced to another set of artifacts (e.g., test cases). The new traceability matrix view allows to easily create such matrices and even shows an overview of all defined traceability links. The matrix can be exported as a spreadsheet for easy inclusion into reports.
- Added sunburst trace model visualisation. A sunburst visualises all trace links that originate in a selected artifact as arcs located in a circle around it. Each arc represents an artifact that is traced to. Clicking on an arc focuses the view on that artifact and shows the traceability links that originate from it. In addition, it is possible to see chains of traceability links (i.e., transitive traceability links) up to a configurable depth. This allows easy navigation of the trace model in a compact and intuitive view.
- Other relevant changes make it easier to identify the traced artifacts from different editors and views in the Eclipse IDE and to include additional artifacts (e.g., Xtext documents) in the traceability links.


Capra Sunburst viewer
No information about objects provided by lidar : Failure
Lidar Grabber : Component
Capra Matrix Viewer
Delete Link
Refresh
Show all

	No information ...	Hazards!H-4 (m...	Lidar Grabber : ...	Hazards!H-3 (m...	Hazards!H-2 (m...	Hazards
No info...	■					
Hazard...	LeadsToHazard	■				
Lidar G...	EffectsCompo...		■			
Hazard...	LeadsToHazard			■		
Hazard...	LeadsToHazard				■	
Hazard...	LeadsToHazard					■

Figure 3.1: The Sunburst and Matrix views introduced in Eclipse Capra 0.8.0

4 Eclipse Capra 0.8.1

Eclipse Capra 0.8.1 was a maintenance release that focused on improving the user experience with a particular focus on traceability link visualisations. It included the following noteworthy changes:

- Improved visualisation of artifacts in views by removing line breaks. Certain artifacts can contain line breaks in their display name. For instance, Eclipse Capra can show the entire text of a requirement that is the origin or target of a traceability link. However, in many visualisations, the line breaks break the layout and make the artifact unreadable. The traceability matrix, e.g., does not support multi-line text as a header. Therefore, line breaks are now removed before displaying an artifact, leading to a better user experience.

- Improved selection of artifact handler. The artifact handler determines how an artifact is treated by Eclipse Capra. In this version, the component that selects which artifact handler to use for a certain artifact has been improved so that the most specific artifact handler gets the job (e.g., the handler for APP4MC instead of the generic one for EMF). This improves the user experience since, e.g., users now see more specific artifact descriptions.

- User interface improvements. A number of views included in Eclipse Capra have received updates to improve the user Experience. The view used to select elements in Microsoft Office documents now has icons in its toolbar instead of text. The user instructions for the user in the PlantUML view have been made clearer. Selected elements are now more reliably visualised in the traceability viewers.

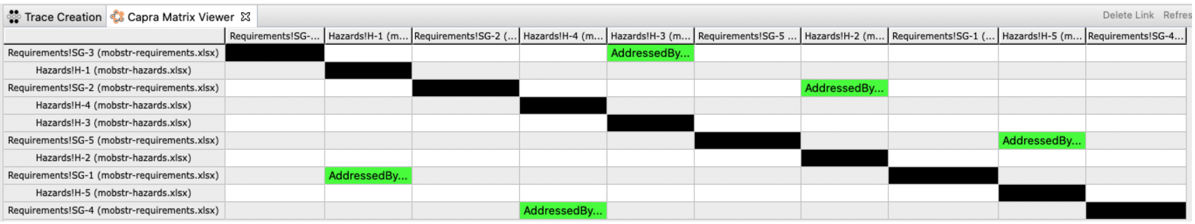
5 Impact of the Changes on Safety Practices

Eclipse Capra supports safety analysis and the construction of a safety case by providing engineers with the ability to establish and manage the relationships between the artifacts and by visualising these relationships to establish evidence that safety concerns have been addressed correctly. ISO 26262, e.g., requires engineers to provide evidence that all hazards have been addressed by safety goals which in turn have been implemented by concrete measures. Eclipse Capra allows engineers to trace between hazards, safety goals, measures, their implementation, and corresponding test cases to establish this evidence.

Once the traceability links between these artifacts have been created, engineers need to be able to access them and to include reports based on them into their safety cases. The additional functionalities provided by Eclipse Capra support this task. The traceability matrix view can be used to show, e.g., that all hazards have been addressed by a corresponding safety goal (cf. Figure 5.1). Likewise, the traceability matrix can be used to show that each safety requirements has associated test cases or that each component has a corresponding implementation. Eclipse Capra also offers the option to export the traceability matrix, which makes it easy to include the view as evidence into a safety case.

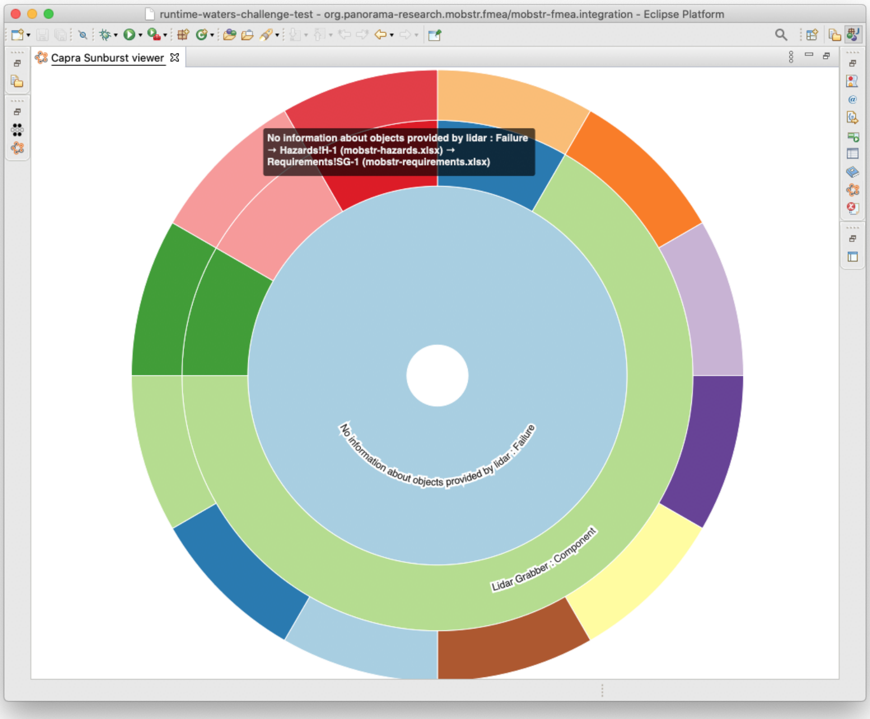
Likewise, the sunburst view is helpful in determining chains of traceability links. Figure 5.2 shows, e.g., the artifacts a failure from a failure mode and effects analysis (FMEA) is connected to. The arcs in this figure also indicate that all hazards the failure is traced to have a corresponding safety goal. Therefore, such a visualisation makes it easy for the engineer to see any gaps in the artifacts.

Overall, these changes improve the way engineers can interact with the artifacts, how they create evidence, and how they identify gaps in the set of artifacts of the system under construction. Future versions of Eclipse Capra will include further improvements, e.g., specialised tools for coverage analysis that also show which artifacts are currently not sufficiently included in the traceability model. This means that engineers will no longer have to search for gaps, but that the tool provides a way to identify them semi-automatically.



	Requirements!SG-...	Hazards!H-1 (m...	Requirements!SG-2 (...	Hazards!H-4 (m...	Hazards!H-3 (m...	Requirements!SG-5 ...	Hazards!H-2 (m...	Requirements!SG-1 (...	Hazards!H-5 (m...	Requirements!SG-4...
Requirements!SG-3 (mobstr-requirements.xlsx)	■				AddressedBy...					
Hazards!H-1 (mobstr-hazards.xlsx)		■								
Requirements!SG-2 (mobstr-requirements.xlsx)			■				AddressedBy...			
Hazards!H-4 (mobstr-hazards.xlsx)				■						
Hazards!H-3 (mobstr-hazards.xlsx)					■					
Requirements!SG-5 (mobstr-requirements.xlsx)						■			AddressedBy...	
Hazards!H-2 (mobstr-hazards.xlsx)							■			
Requirements!SG-1 (mobstr-requirements.xlsx)		AddressedBy...						■		
Hazards!H-5 (mobstr-hazards.xlsx)									■	
Requirements!SG-4 (mobstr-requirements.xlsx)				AddressedBy...						■

Figure 5.1: A traceability matrix showing how hazards are addressed by safety goals.


runtime-waters-challenge-test - org.panorama-research.mobstr.fmea/mobstr-fmea.integration - Eclipse Platform
Capra Sunburst viewer
No information about objects provided by lidar : Failure
Lidar Grabber : Component

No information about objects provided by lidar : Failure → Hazards!H-1 (mobstr-hazards.xlsx) → Requirements!SG-1 (mobstr-requirements.xlsx)

Figure 5.2: A sunburst view of a failure and its linked artifacts, showing that the failure is connected to a hazard and the hazard is connected to a safety goal.