AbRA
Architecture-based Risk Analysis

in agile developments (but not only)

Juergen Sauler, Florian Beer

BOSCH

01

Status quo

Why do we need an alternative method?

BOSCH

AbRA Architecture-based Risk Analysis
Why do we need an alternative method?

- The automotive industry is undergoing a transformative shift with the advent of Software-intensive applications like ADAS, introducing unprecedented complexity within product development
- In the context of agile methodologies, adhering to traditional risk analysis (TRA) approaches is challenging due to rapid release cycles and the continuous evolution of products
- Significant resources are currently invested in CI/CV to accelerate SW deployment. These benefits are significantly diminished if we must wait weeks or months for safety argumentation due to the manual nature of the process

AbRA: Created by developers,
optimized for developer needs.

Jürgen Sauler, Florian Beer | 2024-07-04

02

AbRA Architecture-based Risk Analysis

Implementation of the concept

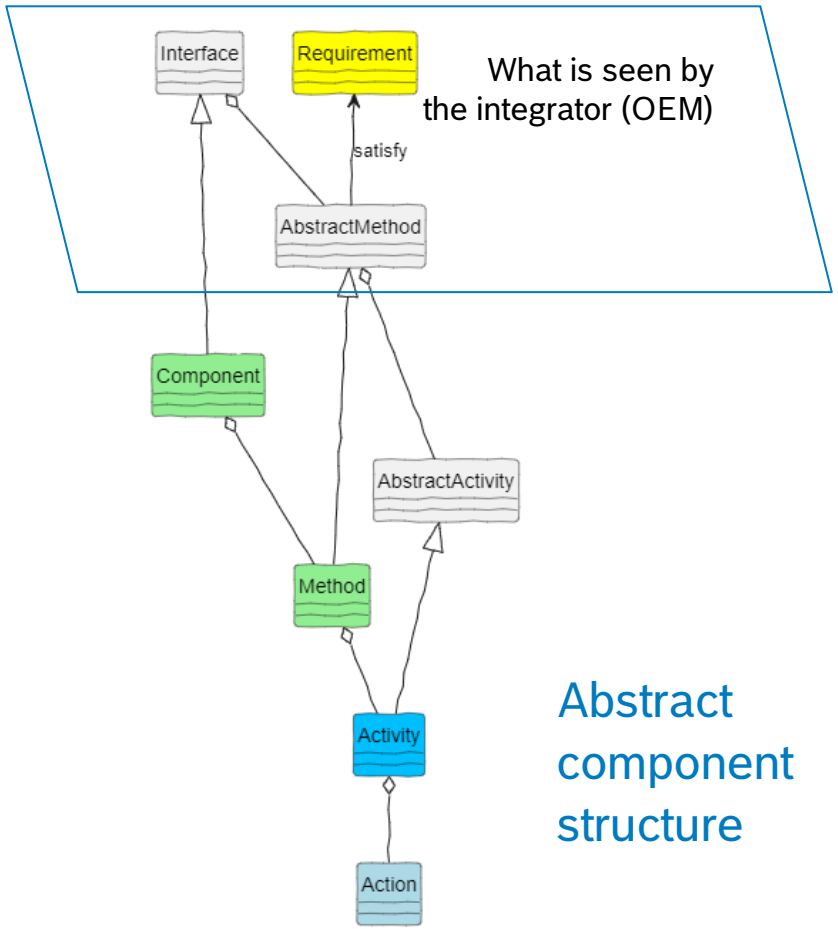
BOSCH

Extension of architectural model

- Regular architectural elements
  - Interfaces define the signature and visible behavior
  - Components implement the defined behavior
  - Activities define how the defined behavior is reached by combining actions and decisions

- Integrator (OEM) sees interface and interface behavior

Abstract component structure



Abstract component structure

Jürgen Sauler, Florian Beer | 2024-07-04

AbRA Architecture-based Risk Analysis

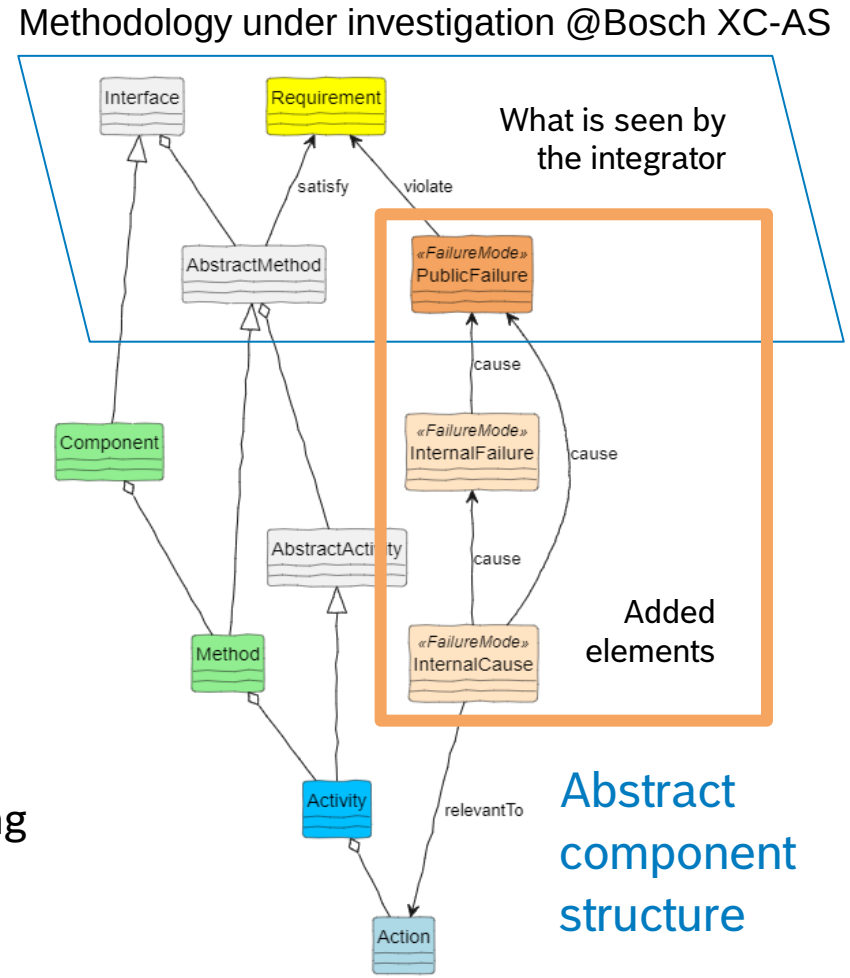
Extension of architectural model

Extension of architectural model for AbRA

- New architectural element:
  - The ***FailureMode*** is the **only** new item for TRA
  - The ***ThreatMode*** would be the **only** new item to support TARA

- Regular architectural elements
  - Interfaces define the signature and visible behavior
  - Components implement the defined behavior
  - Activities define how the defined behavior is reached by combining actions and decisions

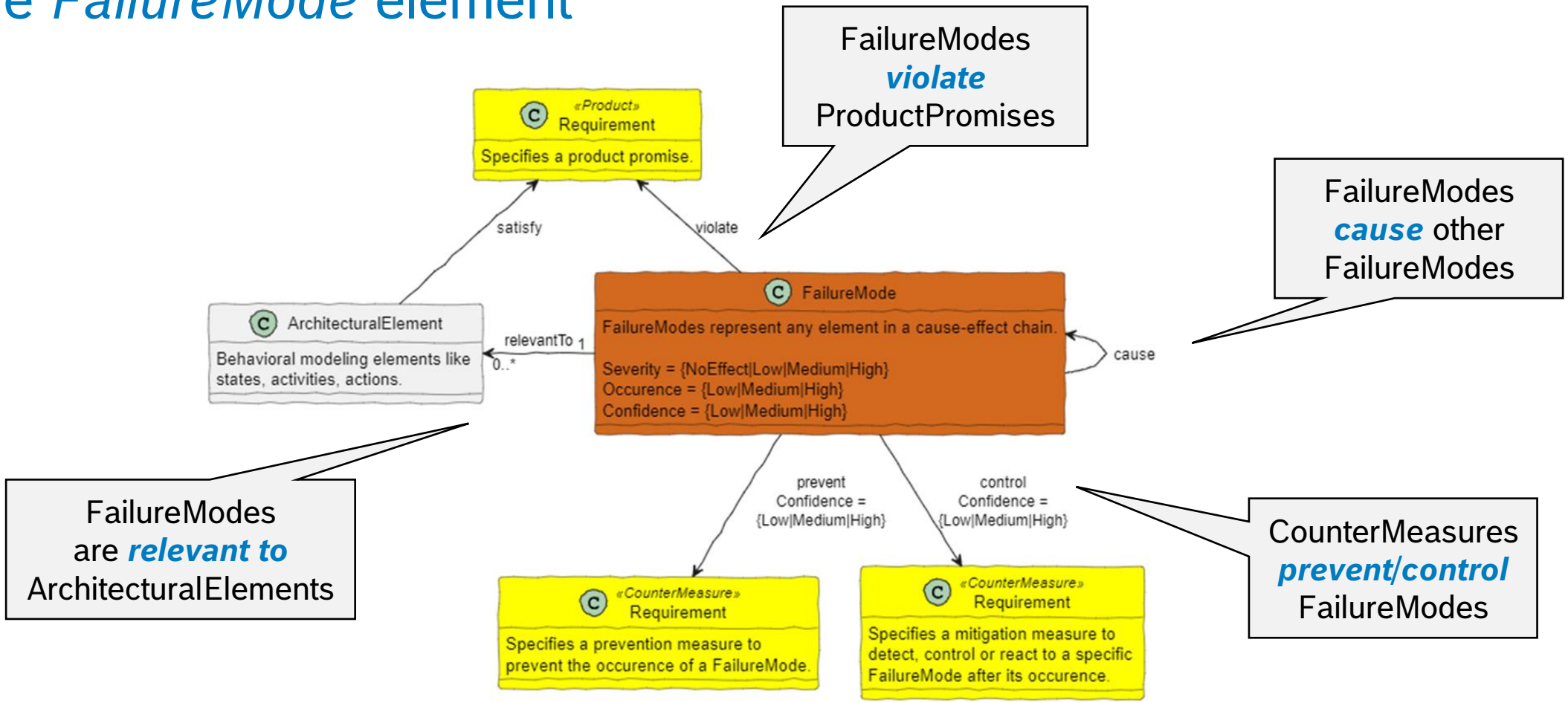
- Integrator sees interface, interface behavior and ***Public Failures***



AbRA can be used with all architectural models based on UML/SysML

AbRA Architecture-based Risk Analysis

The *FailureMode* element

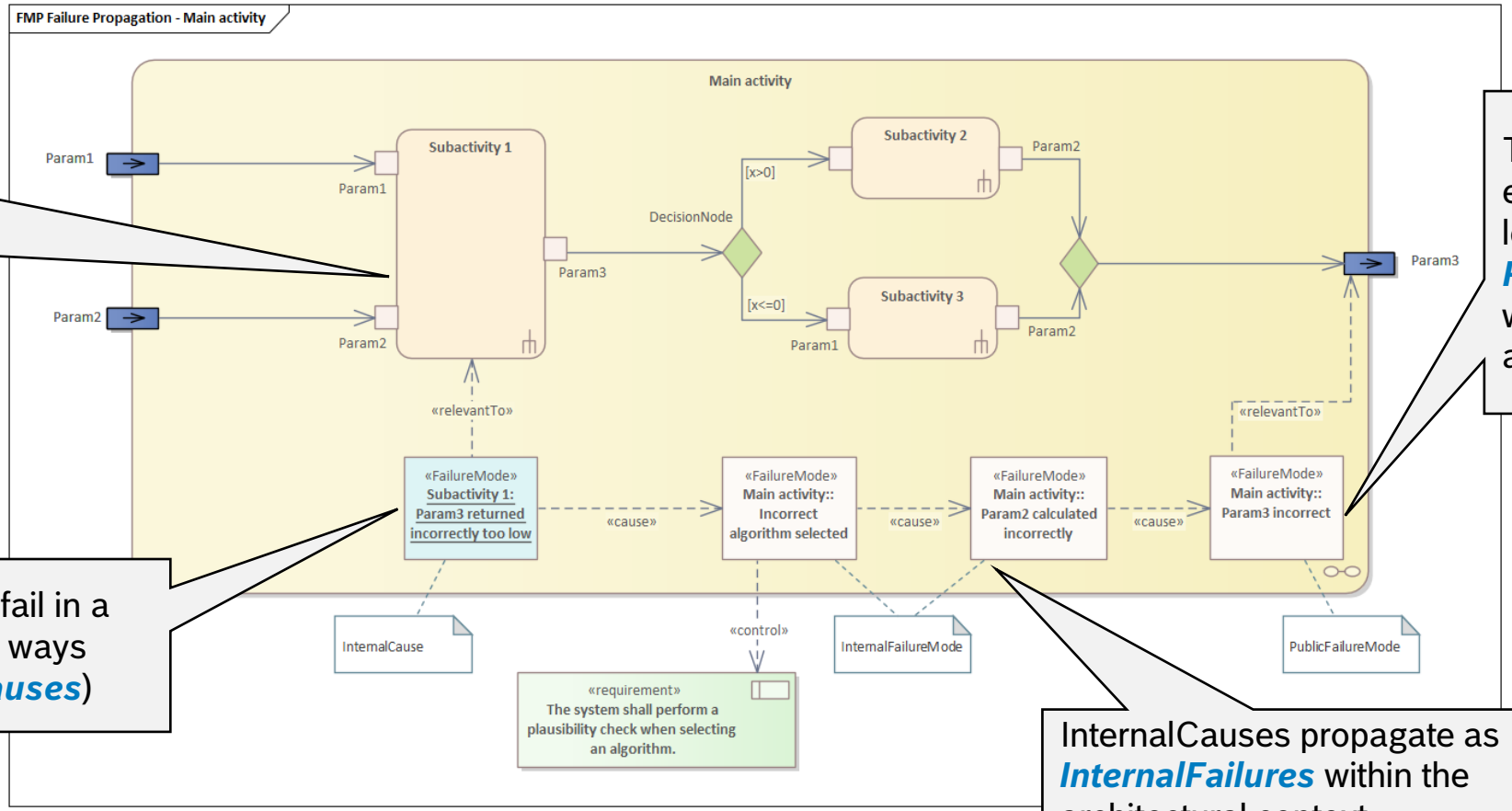


Jürgen Sauler, Florian Beer | 2024-07-04

BOSCH

How does it look alike in real – one simple example

Subactivity 1 is a behavioral architectural element ...



... and it can fail in a multitude of ways (***Internal Causes***)

The propagation eventually leads to ***PublicFailures***, which are visible at the interface

InternalCauses propagate as ***InternalFailures*** within the architectural context

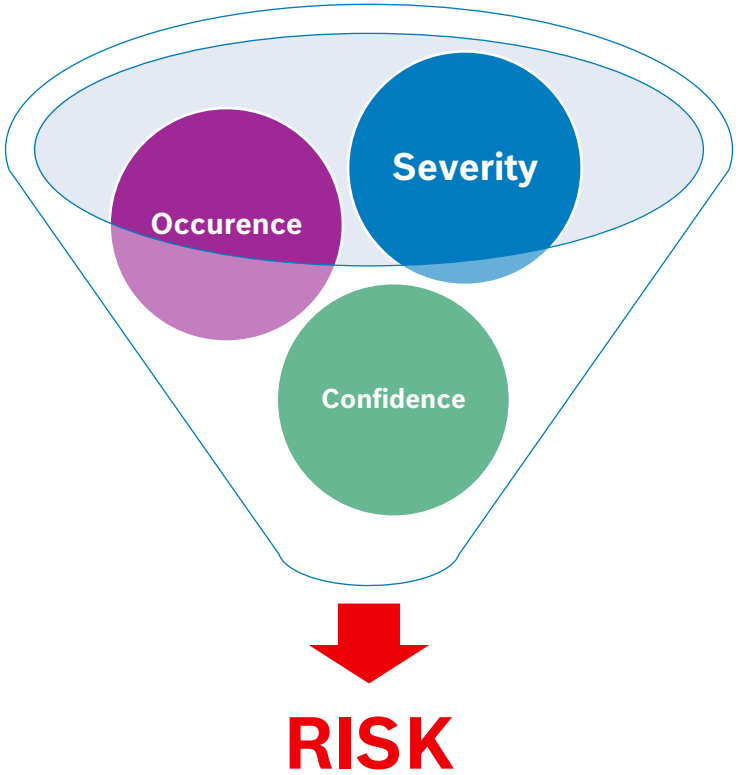
AbRA Architecture-based Risk Analysis
Risk evaluation

- Just like in classical approaches, **Risk** is determined by

- **Occurence**
  - e.g. probability of systematic failures

- **Severity**
  - e.g. severeness of a requirement violation

- **Confidence**
  - e.g. confidence in defined countermeasure(s)

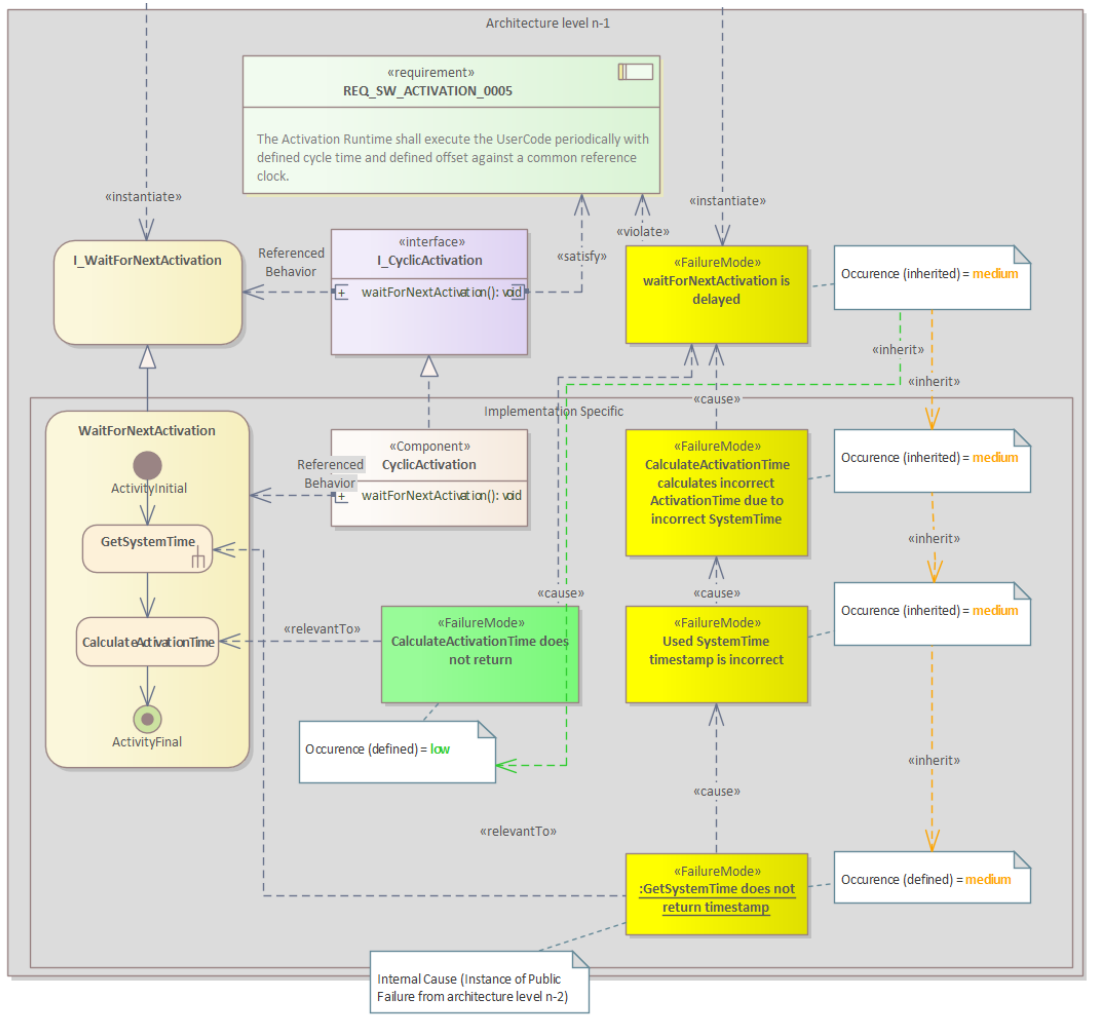


AbRA risk evaluation is comparable to FMEA risk evaluation and results can be used in the same way

AbRA Architecture-based Risk Analysis

Risk evaluation

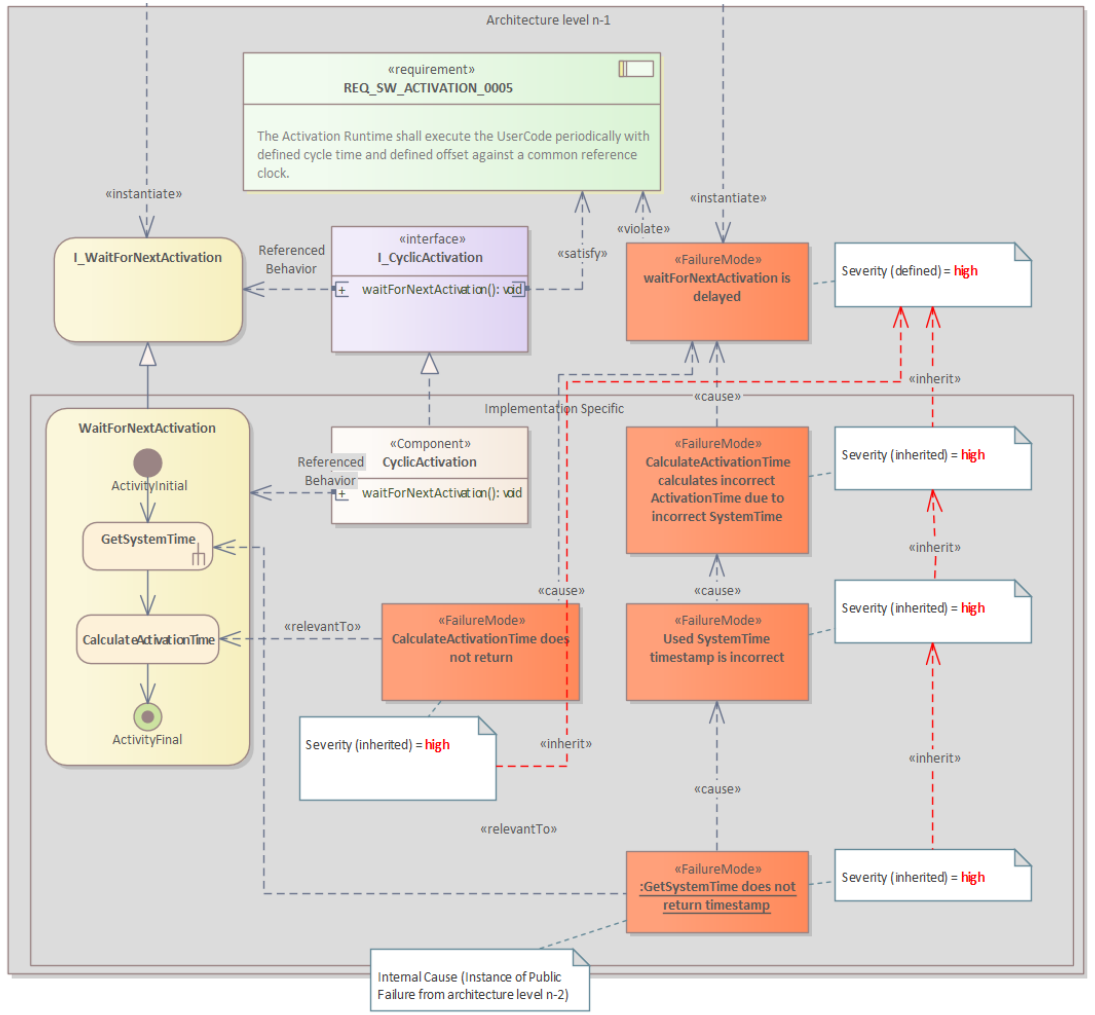
- **Occurence**
  - The developer estimates the probability of systematic failures at the architectural element to be analyzed
  - is inherited bottom-up through the cause-effect-chain within one architectural level
  - Occurence = { Low | Medium | High }



Jürgen Sauler, Florian Beer | 2024-07-04

Risk evaluation

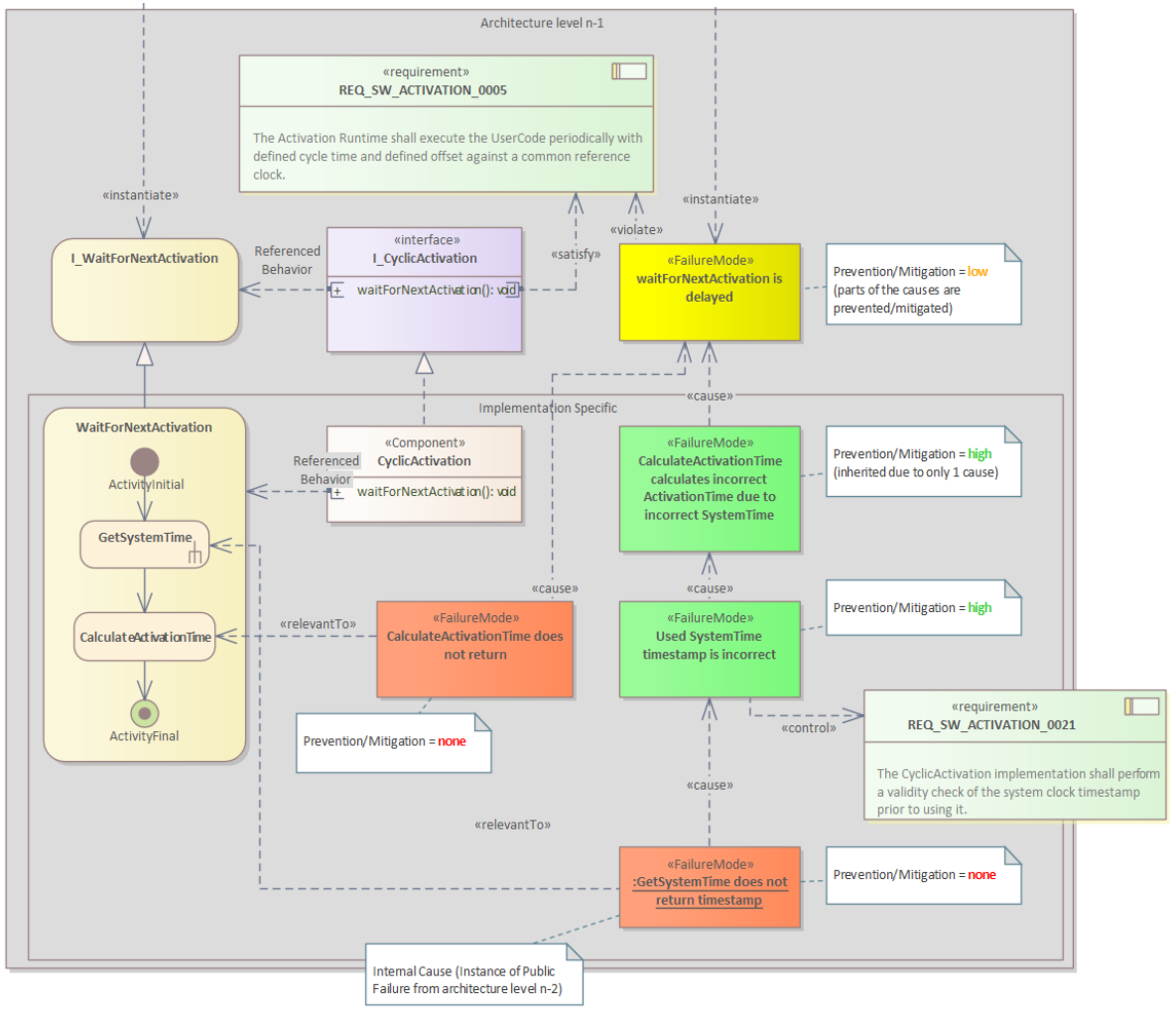
- **Severity**
  - is evaluated at the ***PublicFailure*** level of the architectural element that is analyzed
    - Does a PublicFailure ***violate*** a product requirement?
    - How severe is the violation?
  - is propagated top-down through the cause-effect-chain within one architectural level
  - Severity = { Low | Medium | High }



Jürgen Sauler, Florian Beer | 2024-07-04

Risk evaluation

- **Confidence**
  - is evaluated at linking of ***CounterMeasure*** Requirements
    - Does the CounterMeasure prevent or mitigate the linked FailureMode?
    - How confident is the CounterMeasure?
  - Prevention measures may cut a cause-effects-chain
  - Mitigation measures cannot cut a cause-effects-chain, but only add coverage
  - Confidence = { Low | Medium | High }



Jürgen Sauler, Florian Beer | 2024-07-04

AbRA Architecture-based Risk Analysis
Risk evaluation

Risk (Action Priority)

- From **Occurence**, **Severity** and **Confidence** in Prevention/Mitigation an overall **Risk** or **Action Priority** rating can be established

- **Action Priority** can be used for further Risk management activities

Severity	Occurence			Confidence in Prevention/Detection
	low	med	high	
low				high
				med
				low
				no Countermeasure
med				high
				med
				low
				no Countermeasure
high				high
				med
				low
				no Countermeasure

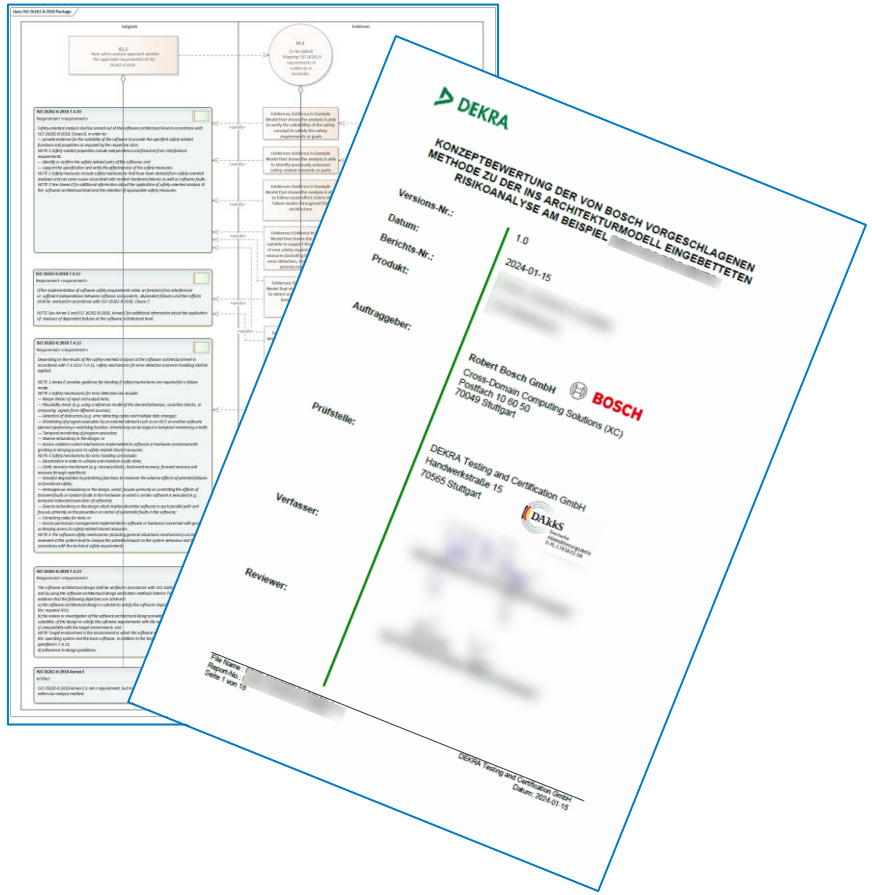
*The displayed matrix is a draft. The actual risk rating matrix is not finally defined.

AbRA Architecture-based Risk Analysis
Use in ISO 26262 context

- The proposed Architecture-based Risk Analysis method was justified against the analysis requirements of ISO 26262: 2018 part 4, 6, 9

- DEKRA Assessor confirmed method as suitable to fulfil ISO 26262, additional evidence documents were provided from BOSCH to DEKRA

- A **formalized concept evaluation report** was provided to BOSCH by DEKRA



AbRA Architecture-based Risk Analysis
Synergizing AbRA and MBSE

- AbRA itself is intentionelly simple

- Due to ist simplicity, AbRA can fully inherit all the qualities of the architectural model
- And leverages significant advancements in the MBSE environment over recent decades, e.g.
  - Collaborative Support (Continuous Architecture)
  - Continuous Validation (Model checker, Compliance check, ...)
  - Integration into Pipeline (automated quality gates)
- And will capitalize on all future improvements

AbRA's greatest potential benefit is increasing motivation to invest in a high-quality architectural model

03

How to make AbRA powerful

A good architectural model is the base for a good risk analysis
Agile Team Integration (Collaborative Architecture)

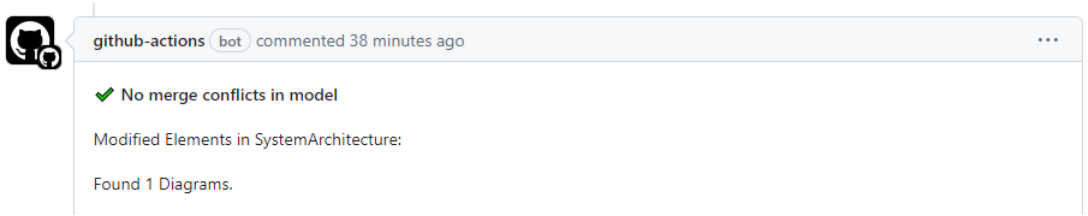
- Treat architecture like code, incorporating continuous Integration and Verification
  - Leverage Git‘s versioning and collaboration capabilities
- Empowers the entire team to collaboratively work on architecture and AbRA

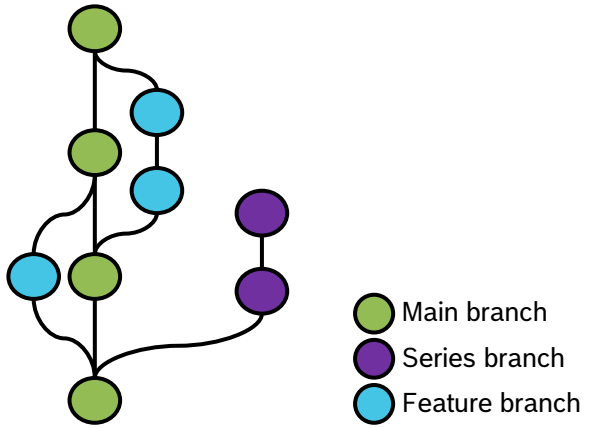
From Architecture as role to Architecture as competence

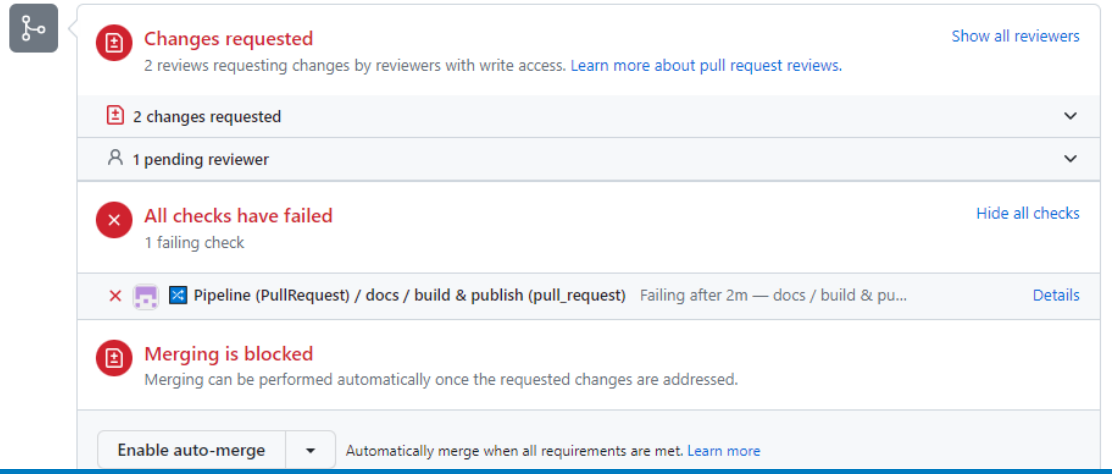
From Quality as role to Quality as competence

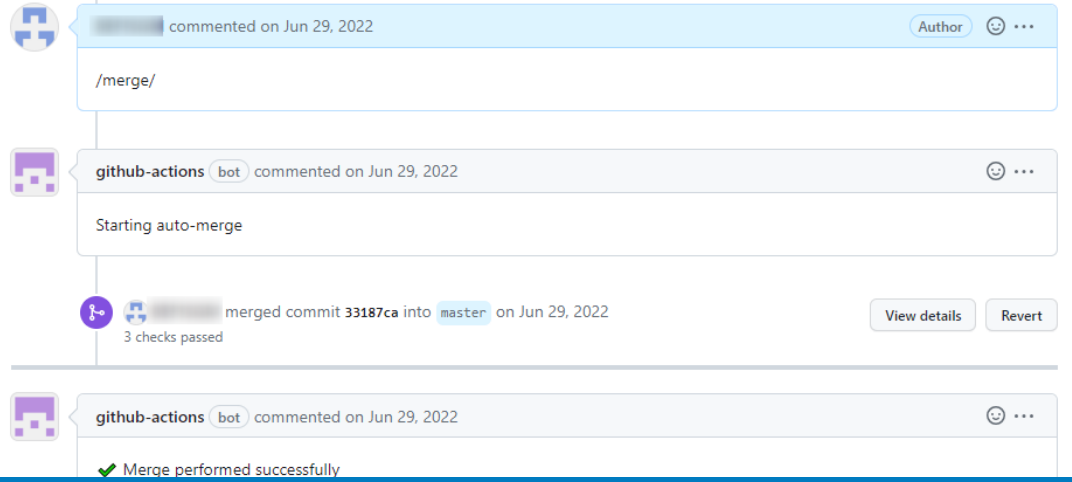
A good architectural model is the base for a good risk analysis

Collaborative Architecture







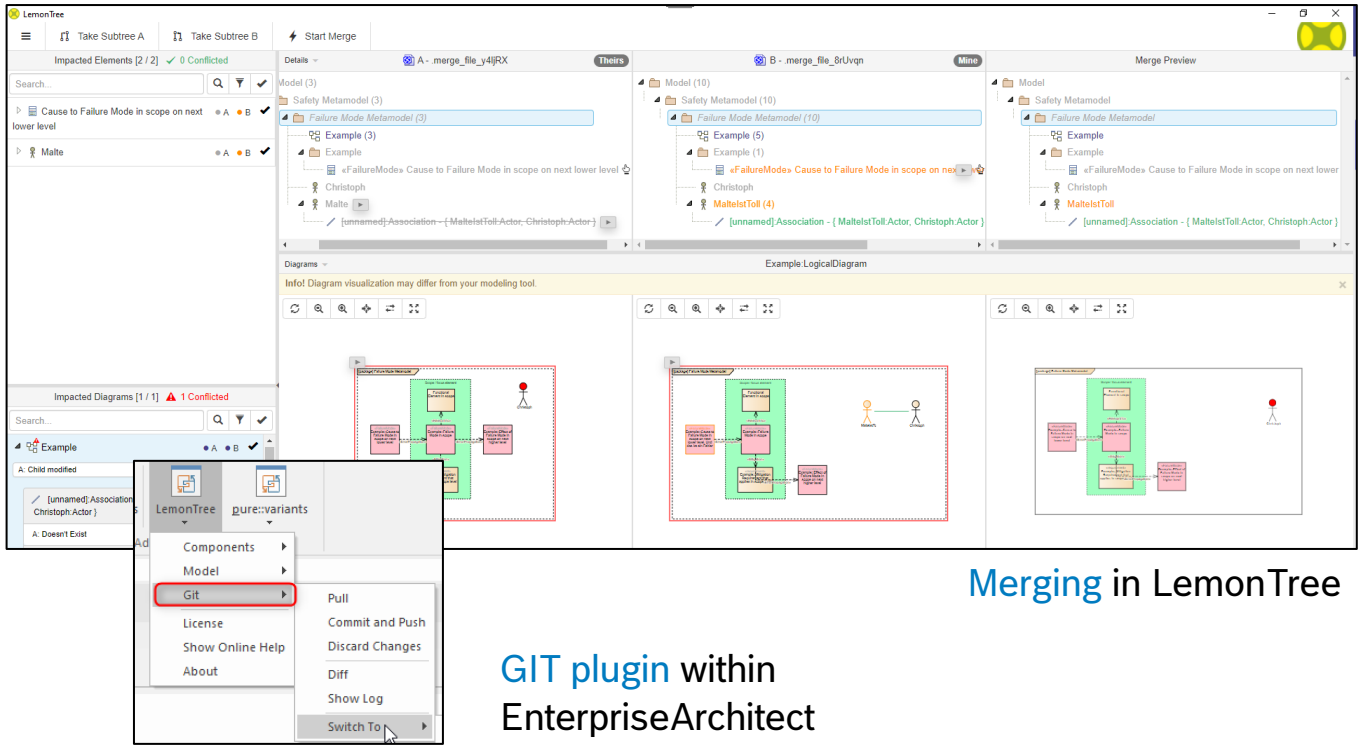


Enables distributed Risk Analysis: Locally and in time

Collaborative Architecture:
Multiplying the benefits of AbRA: Collaborative Architecture

- Use of LieberLieber LemonTree® for MBSE GIT workflow with Sparx Systems Enterprise Architect®
  - Diff-based reviews of the models
  - Automated merge of non-conflicting changes
  - Supporting manual resolving of merge conflicts based on model properties
  - Package deployment to other models

→ Enabling collaborative architecture



Merging in LemonTree

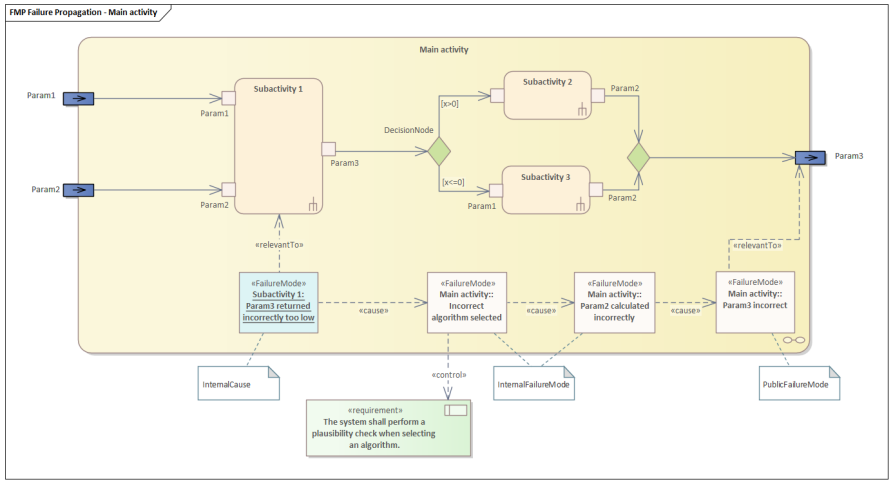
GIT plugin within EnterpriseArchitect

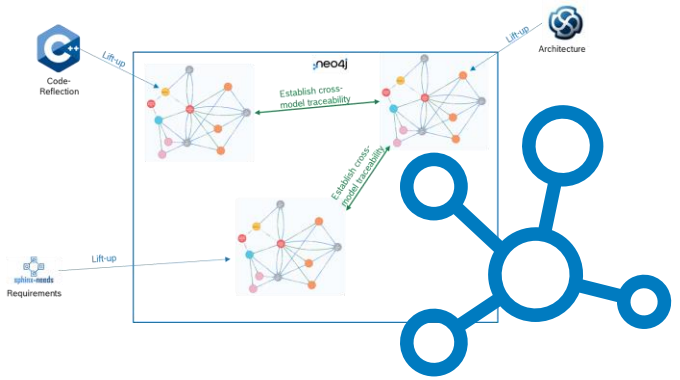
Jürgen Sauler, Florian Beer | 2024-07-04

Collaborative Architecture:
Support by automated Verification & Validation

- The goal of **consistency-checks** between architecture, analysis and implementation is only possible by the use of **advanced automation**

- AbRA benefits from the **continuous validation** of the model, e.g. each pull-request triggers the
  - Model checker to
    - ensure that modelling guidelines are fulfilled
    - AbRA method is applied correctly
  - Consistency check between architectural model and source code
  - ...





We can always prove that we have analyzed what we have deployed!

04

Summary

BOSCH

AbRA Architecture-based Risk Analysis
Summary

AbRA is intended to create an alternative for classic methods like FMEA in ADAS system projects with the primary goals:

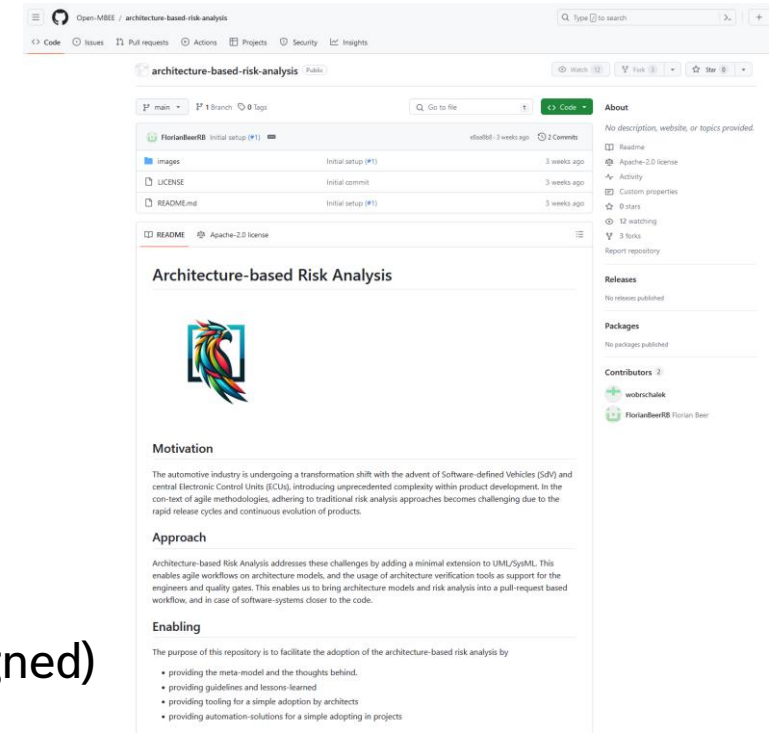
- Ensure rapid alignment between design and risk analysis
- Enabling distributed analysis (locally and in time)
- Ensure Risk Analysis is accurate and up-to-date
- Maximize the use of automation for efficiency
- Integrate with existing methods to enhance effectiveness
- Minimize **resource** consumption and manual effort

Currently being piloted in highly complex agile development projects (early phase)

AbRA Architecture-based Risk Analysis

AbRA will become Open Source

Establish an open-source community to promote AbRA as a standard approach for risk analysis in the industry



Encourage contributions from the community

- Collaborative handling of architecture (contribution with partner aligned)
- Automated architecture verification (potential partners identified)
- Improvement potentials for AbRA identified from broad application

Neutral home as enabler for industry-wide adoption:
https://github.com/Open-MBEE/architecture-based-risk-analysis

Thank you!

...Questions?

All pictures are created by Dall-E