ITEA 2

INFORMATION TECHNOLOGY FOR EUROPEAN ADVANCEMENT

openETCS

Model Based Safety Analysis with the Safety Architect Approach

Frédérique Vallée – All4tec

supported by:

Federal Ministry of Education and Research

Région de Bruxelles-Capitale

GOBIERNO DE ESPAÑA MINISTERIO DE CIENCIA E INNOVACIÓN

openETCS@ITEA2 Project

John Doe

Paris, 03.07.2013

Content

1. Introduction
2. System Engineering and Safety
3. ALL4TEC Risk Analysis
4. Safety Architect Tool
5. Application in OpenETCS

1. Introduction

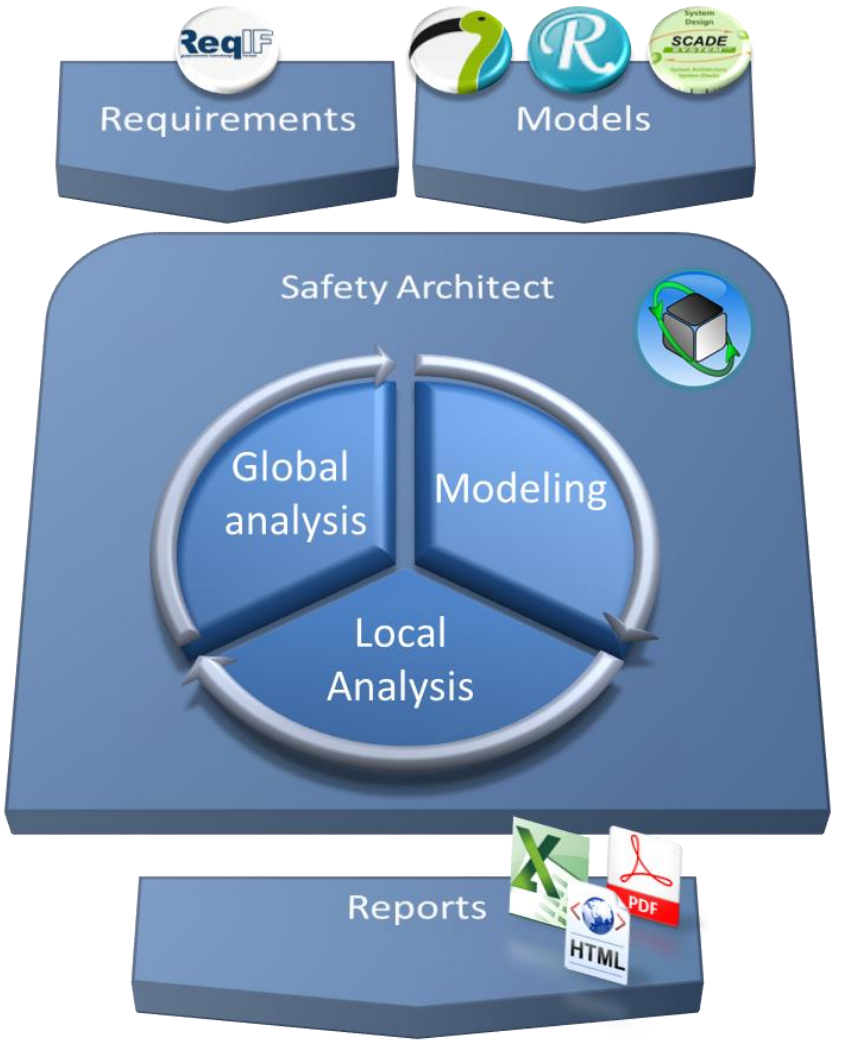
What is he Safety Architect approach?

- ✓ **Risk analysis based on Fault Tree automatic generation**
  - **… issued from the ALL4TEC specific FMEA approach**
- ✓ **Independent from any engineering tool**
  - **… but capable to interface with many engineering tools**

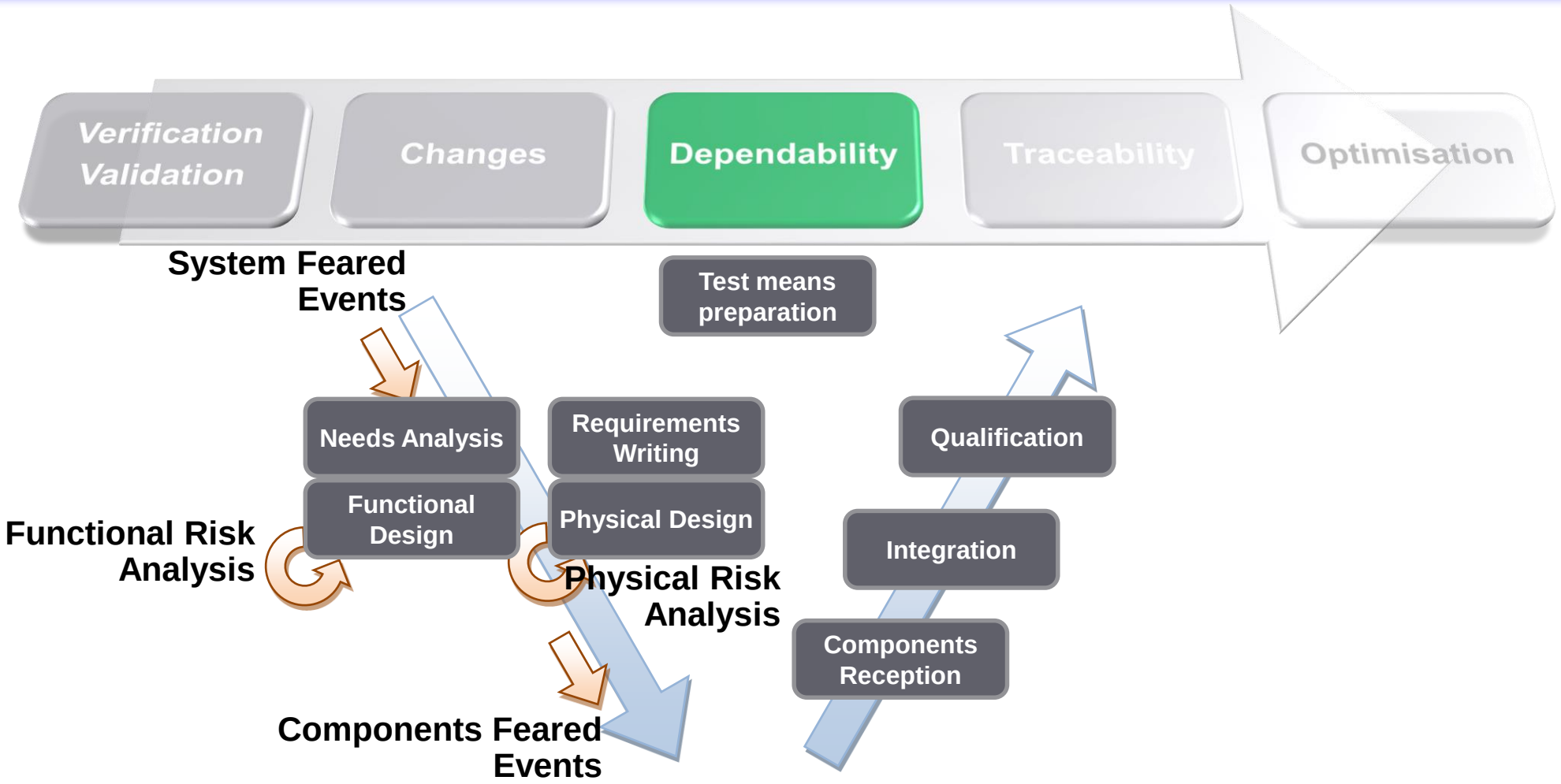
All4tec at a Glance

Safety Architect - Principles

- Integrate safety analysis and design
- Decrease risk of error
- Focus on added value tasks
- Reduce analysis costs

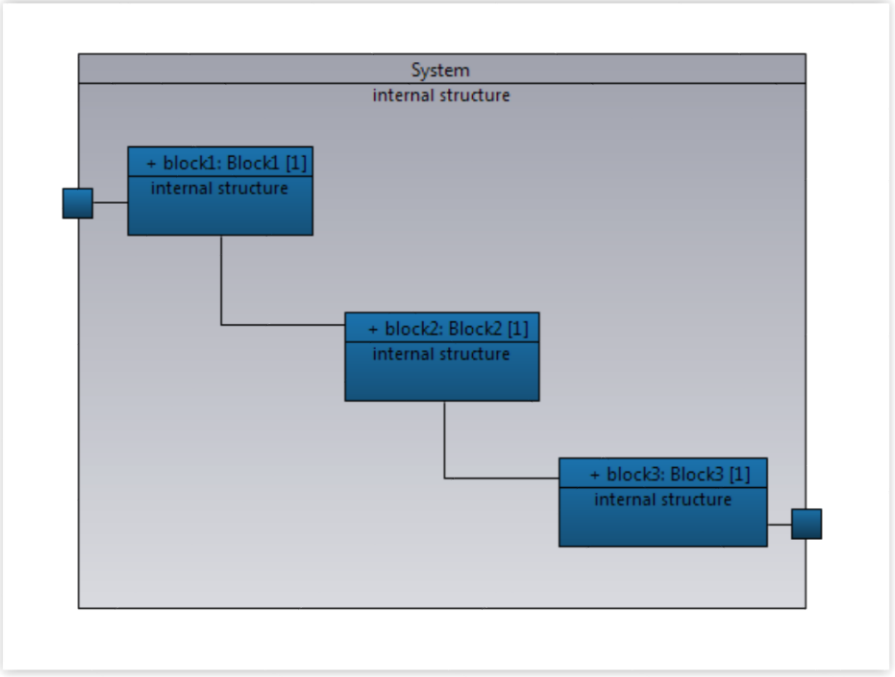


2. System Engineering and Safety



Model Driven System Engineering

- ✓ **Control the complexity**
- ✓ **Keep flexibility**
- ✓ **Ensure consistency**



3. ALL4TEC Risk Analysis

Hierarchical modelling

Local analysis (manual)

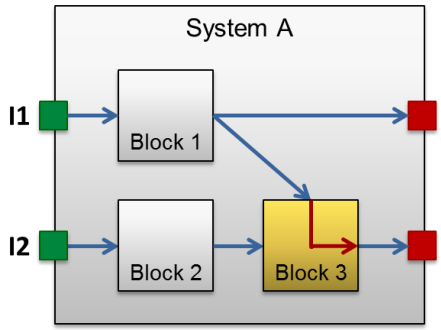
- ✓ **Takes barriers into account**

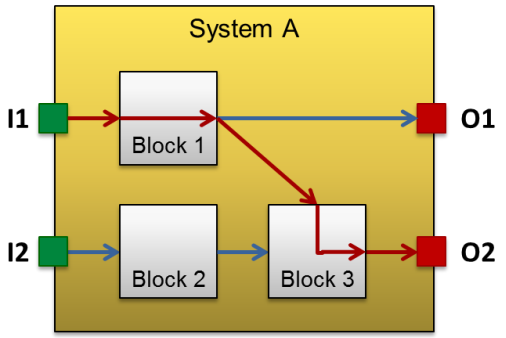
Global analysis (automatic)

- ✓ **Propagation tree of failure modes until feared event**

FMEA table and fault tree

- **If analysis is not satisfactory, model is revisited with additional barriers**

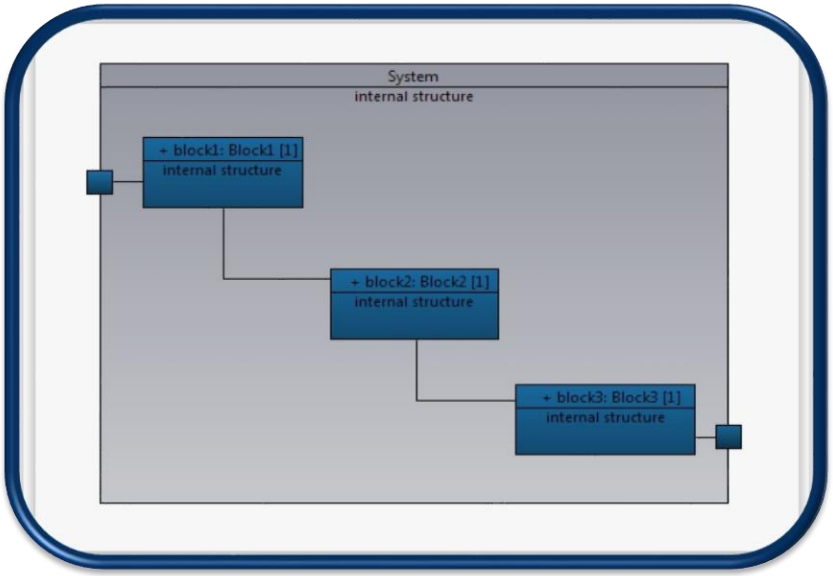


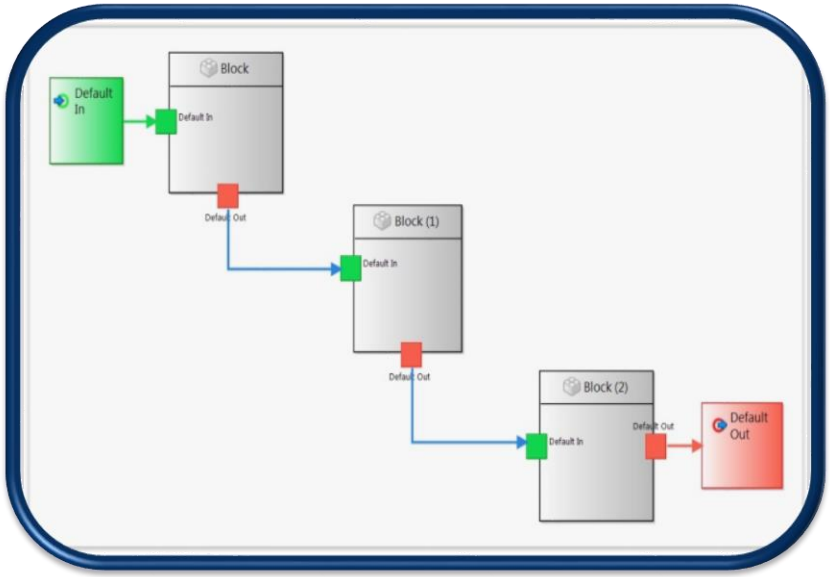


4. Safety Architect Tool

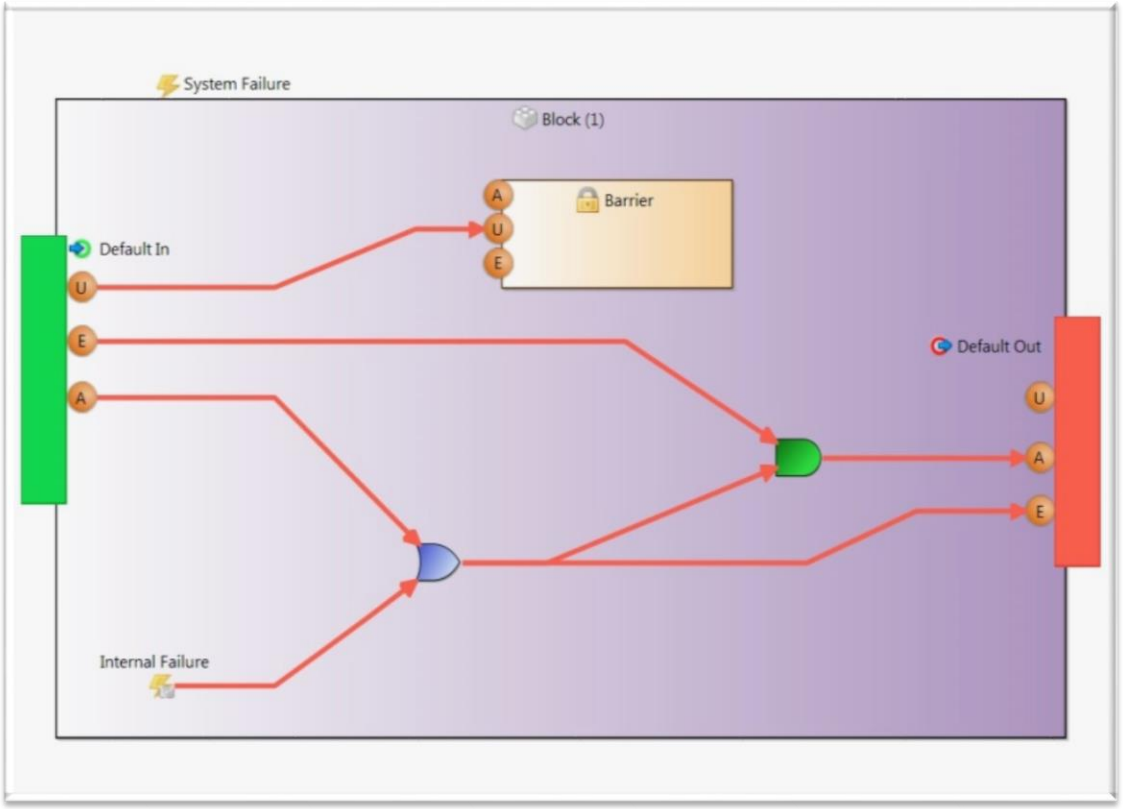
Safety Architect Tool

Safety Architect Tool





Safety Architect tool



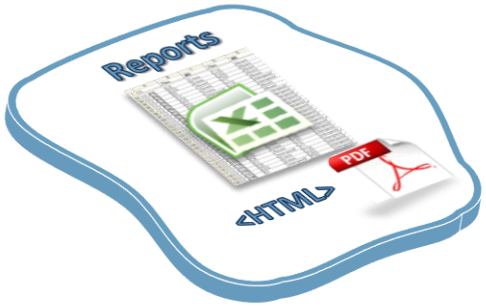
- ✓ **Each block is analyzed independently**
- ✓ **Input failure modes are linked to output failure modes**
- ✓ **Fear events are identified**

Safety Architect tool

- ✓ **Propagation of failure until meeting a feared event**
- ✓ **Identification of ALL critical paths for EACH single feared event**

Name
Non broken circuit
{A - CircuitBreaker::A4 - Opening Command::A41 - Quick Opening}->[F412](A)
{A - CircuitBreaker::A4 - Opening Command::A41 - Quick Opening}->[E26](U)
{A - CircuitBreaker::A4 - Opening Command::A41 - Quick Opening}->[F23](A)
{A - CircuitBreaker::A2 - Check Short Circuit::A23 - ShortCircuitDetect}->[F23 - ShortCircuit detected](A)
{A - CircuitBreaker::A4 - Opening Command::A43 - Uncharge relay}->[F43](A)
{A - CircuitBreaker::A4 - Opening Command::A43 - Uncharge relay}->[F411](A)
{A - CircuitBreaker::A4 - Opening Command::A41 - Quick Opening}->[F411](A)
{A - CircuitBreaker::A4 - Opening Command::A41 - Quick Opening}->[E26](U)
{A - CircuitBreaker::A4 - Opening Command::A41 - Quick Opening}->[F23](A)
{A - CircuitBreaker::A2 - Check Short Circuit::A23 - ShortCircuitDetect}->[F23 - ShortCircuit detected](A)

Safety Architect tool



SAFETY ARCHITECT

PROPAGATION TREE REPORT

1. Feared event

lame : Non broken circuit
)ate : Sep 12, 2014, 4:06 PM

2. Propagation tree

- Non broken circuit
  - {A - CircuitBreaker::A4 - Opening Command::A41 - Quick Opening}->[F412](A)
    - {A - CircuitBreaker::A4 - Opening Command::A41 - Quick Opening}->[E26](U)
    - {A - CircuitBreaker::A4 - Opening Command::A41 - Quick Opening}->[F23](A)
      - {A - CircuitBreaker::A2 - Check Short Circuit::A23 - ShortCircuitDetect}->[F23 - ShortCircuit detected](A)
  - {A - CircuitBreaker::A4 - Opening Command::A43 - Uncharge relay}->[F43](A)
    - {A - CircuitBreaker::A4 - Opening Command::A43 - Uncharge relay}->[F411](A)

SAFETY ARCHITECT

Global report

Table of content

1. Properties
2. System Events
3. Blocks
   - 3.1. A11 - Wait
   - 3.2. A12 - Plug-in_Relay
   - 3.3. A13 - Check_CAN_and_Tension

FMEA Report

	ID	System function	Function	Failure mode	RRF	Mode
1	⊟{CircuitBreaker}	CircuitBreaker				Standard mode
2	System Failure			System Failure	NONE	Standard mode
3	⊟{A - CircuitBreaker}	A - CircuitBreaker				
4	⊟{A - CircuitBreaker::A5 - UI management}		A5 - UI management			
5	Internal Failure			Internal Failure	NONE	Standard mode
6	⊟{A - CircuitBreaker::A5 - UI management} -> [E1]					
7	{A - CircuitBreaker::A5 - UI management}->[E1](U)			U	NONE	Standard mode
8	{A - CircuitBreaker::A5 - UI management}->[E1](A)			A	NONE	Standard mode
9	{A - CircuitBreaker::A5 - UI management}->[E1](E)			E	NONE	Standard mode
10	⊞{A - CircuitBreaker::A5 - UI management} -> [F23]					

SAFETY ARCHITECT

CRITICAL PATHS REPORT

Table of content

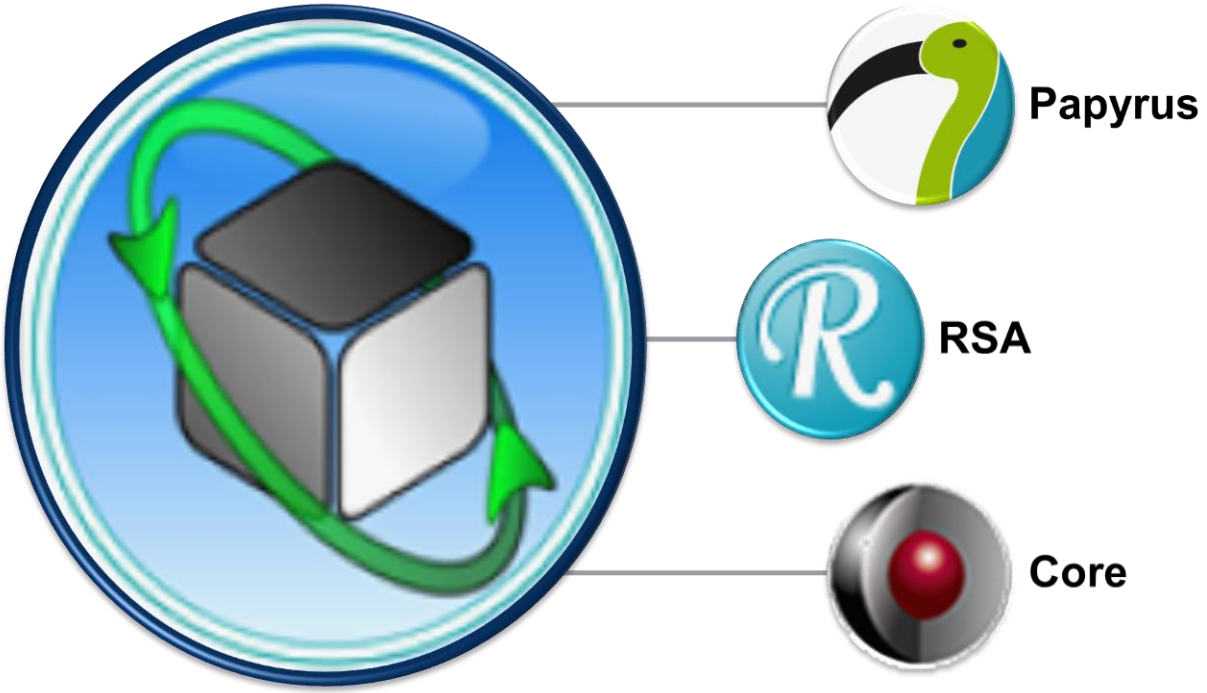
1. Feared events
   - 1.1. Erroneous vote
   - 1.2. Missing vote
   - 1.3. Untrue votes

1. Feared events

1.1. Erroneous vote

1 {Recovery::Standby} -> [Barrier (1)](A) AND Internal Failure => {Recovery::Sensor} -> [SensorDF](A) => {Recovery::Prim
{Recovery::Primary} -> [PrimaryDF](A) => {Recovery::Standby} -> [PrimaryDF](A) => {Recovery::Standby} -> [StandbyD

Safety Architect Tool

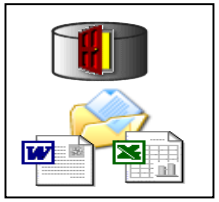


5. Application in OpenETCS

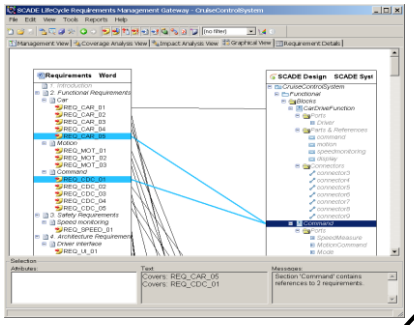
- ✓ **Open Source Licence**
- ✓ **New name : ESF – Eclipse Safety Framework**
- ✓ **Integration in the secondary tool chain**
  - ✓ **Safety Analysis of the system model (SCADE System)**
  - ✓ **Safety verification requirements**

Workflow

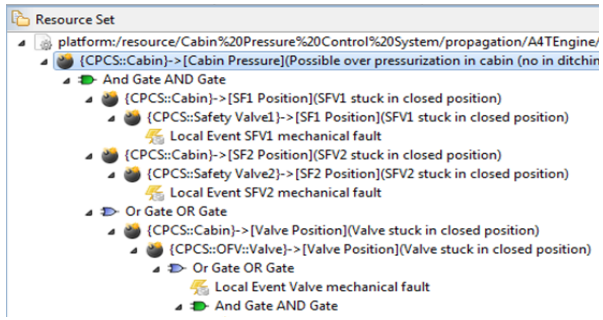
System Requirements



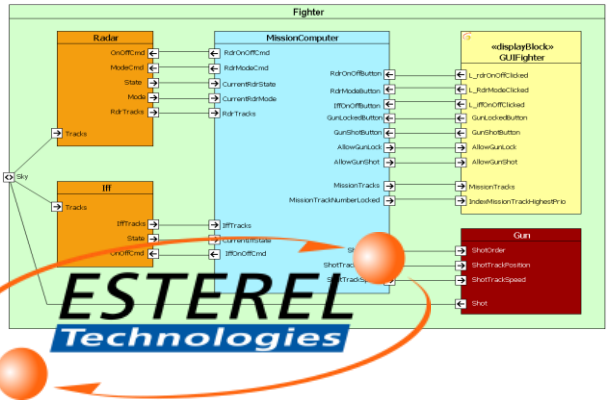
Traceability



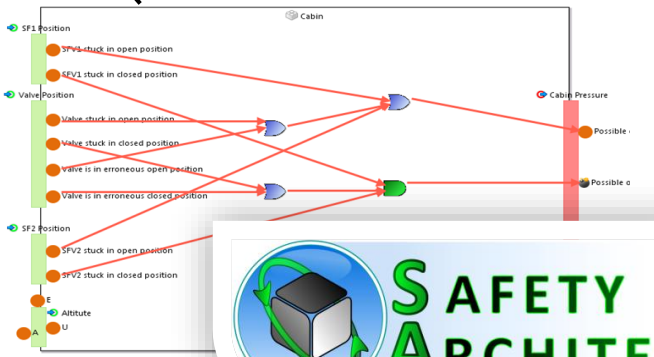
Safety Analysis



System Design



Translation&Merge



THANK YOU !