Traceable Engineering of Fault-Tolerant SoSs

Zoe Andrews, Claire Ingram, Richard Payne and Alexander Romanovsky
Newcastle University, UK

Jon Holt and Simon Perry
Scarecrow Consultants Limited, UK

SoSECIE webinar: 09/15/2015

Outline

- Introduction
- Engineering Fault-Tolerant SoSs
- Conclusions

Outline

- **Introduction**
  - **Systems of Systems**
  - **Systems of Systems and Dependability**
  - **The Fault Modelling and Analysis Method**
- Engineering Fault-Tolerant SoSs
- Conclusions

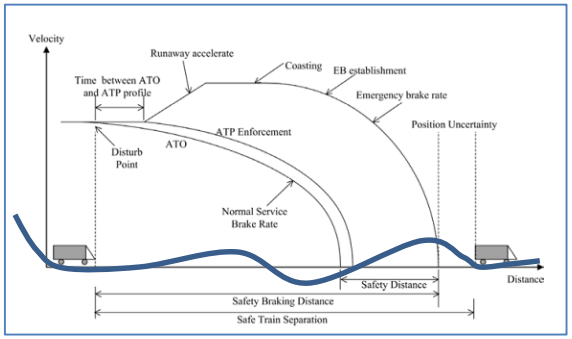
Our Research …

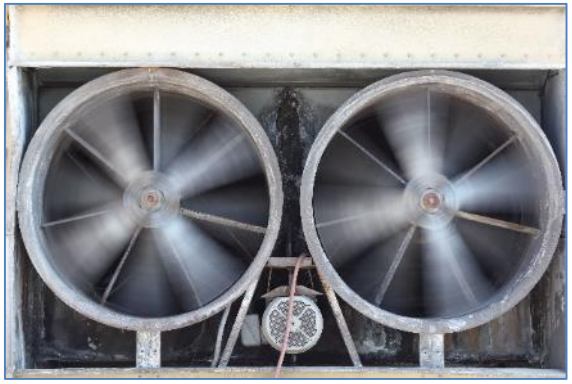
Design technology (foundations, methods, tools) for:

- Systems of Systems (SoS)
- Cyber-Physical Systems (CPS)

We focus on model-based design:

- Models as a basis for collaborative development
- Machine-assisted analysis of models as a means of managing development risk





"HVAC Ventilation Exhaust" by PictorialEvidence - Own work. Licensed under CC BY-SA 3.0 via Wikimedia Commons - http://commons.wikimedia.org/wiki/File:HVAC_Ventilation_Exhaust.jpg#/media/File:HVAC_Ventilation_Exhaust.jpg

Current Projects

- Strongly supported by European Union
- Current Priorities:
  - *Research*: development of sound toolchains for verifying global properties of CPSs (INTO-CPS)
  - *Innovation Support*: creating ecosystems in model-based engineering for CPS (CPSE Labs)
  - *Strategic Actions*: Roadmaps and Transatlantic Research Agenda (Road2CPS, TAMS4CPS)

www.into-cps.au.dk
@IntoCps

www.cpse-labs.eu
@CPSE_Labs

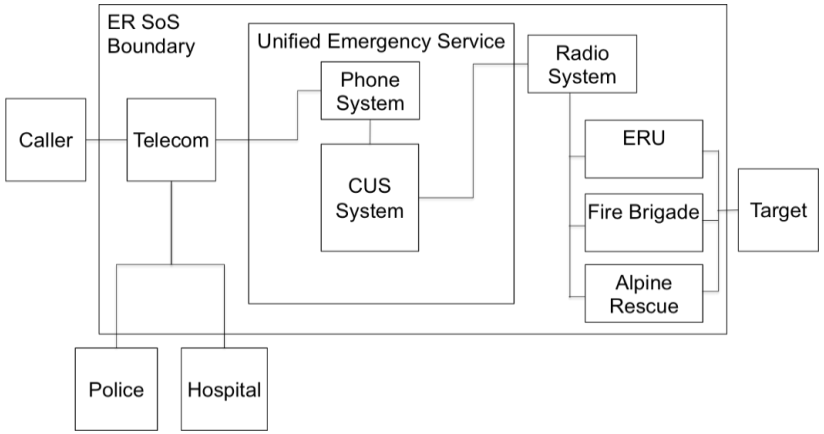
www.road2cps.eu
@ROAD2CPS

www.tams4cps.eu
@TAMS4CPS

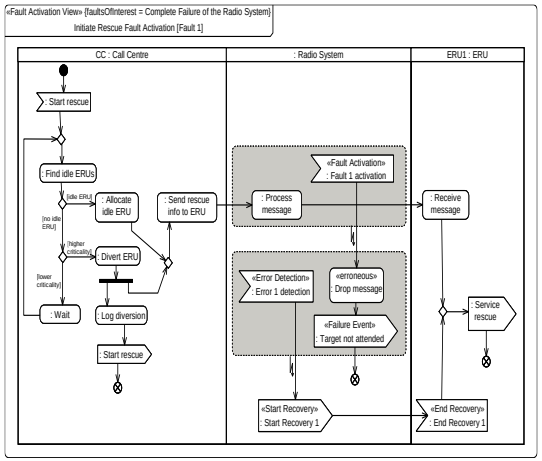
Systems of Systems

- Assembly and integration of independent “constituent systems” interacting via a networked infrastructure
- Collectively offer a new global (“emergent”) service on which value and reliance is placed.
- Independence
- Distribution
- Evolution
- Emergence

- Model-based SoS Engineering as a way of mastering complexity



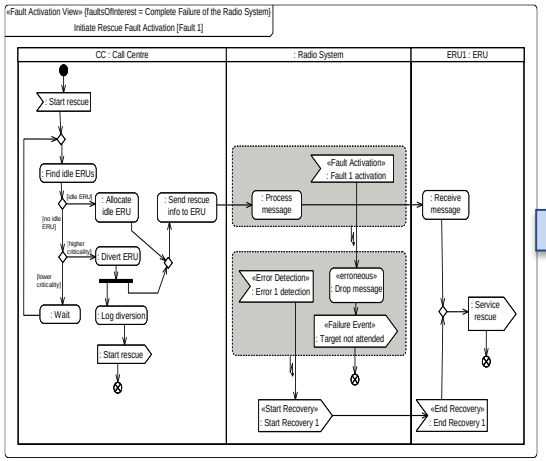
COMPASS Technology



Semi-formal Modelling

- SysML used for SoS modelling
- Guidelines for Requirements, Architecture, Integration
- SoS modelling profiles
- Architectural patterns and extensible architectural frameworks (AFs)

COMPASS Technology




```
process CallCentreProc = begin
...
actions
 MERGE1(r) =
  (dcl e: set of ERUId @ e := findIdleERUs();
   if  e = {}
   then DECISION2(r)
   else
     (dcl e1: ERUId @ e1 :=
       allocateIdleERU(e, r); MERGE2(e1, r)))
    ...


process InitiateRescue = CallCentreProc
          [| SEND_CHANNELS |] RadioSystemProc
          [| RCV_CHANNELS |] ERUsProc
```

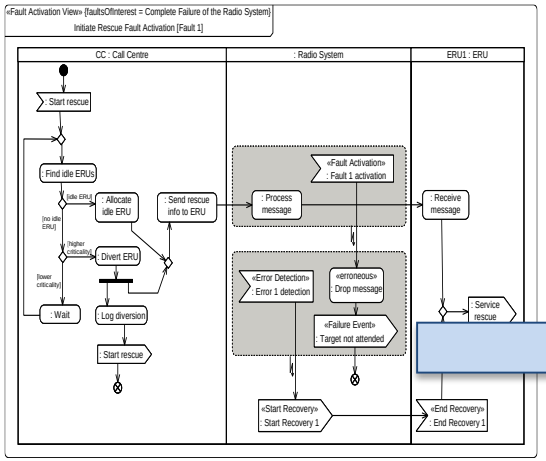
Semi-formal Modelling

- SysML used for SoS modelling
- Guidelines for Requirements, Architecture, Integration
- SoS modelling profiles
- Architectural patterns and extensible architectural frameworks (AFs)

Formal Modelling

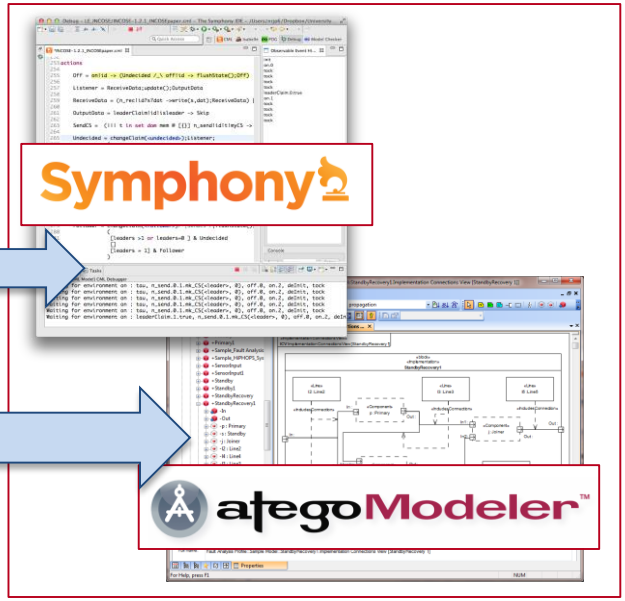
- CML allows representation of behavioural semantics of the SoS
- Supports contract specification
- Describes functionality, object-orientation, concurrency, real-time, mobility.
- Can be extended to new paradigms

COMPASS Technology




```
process CallCentreProc = begin
...
actions
 MERGE1(r) =
  (dcl e: set of ERUId @ e := findIdleERUs();
   if  e = {}
   then DECISION2(r)
   else
     (dcl e1: ERUId @ e1 :=
       allocateIdleERU(e, r); MERGE2(e1, r)))
   ...

process InitiateRescue = CallCentreProc
          [| SEND_CHANNELS |] RadioSystemProc
          [| RCV_CHANNELS |] ERUsProc
```



Semi-formal Modelling

- SysML used for SoS modelling
- Guidelines for Requirements, Architecture, Integration
- SoS modelling profiles
- Architectural patterns and extensible architectural frameworks (AFs)

Formal Modelling

- CML allows representation of behavioural semantics of the SoS
- Supports contract specification
- Describes functionality, object-orientation, concurrency, real-time, mobility.
- Can be extended to new paradigms

Tool-supported Analysis

- Model-checker
- Automated proof
- Static Fault Analysis
- Test generation
- Simulation
- Model-in-Loop Test
- Exploration of design space

SoS and Dependability

Independence

Distribution

Evolution

Emergence

Challenge:

CSs are *autonomous* and *independent*, responsibility for preventing SoS failure when a CS fails or withdraws from the SoS is unclear

Approach:

Record explicitly where responsibility for fault tolerance lies

SoS and Dependability

Independence

Distribution

Evolution

Emergence

Challenge:

Networking technology that SoS-level functionality relies on may introduce faults

Approach:

Model-based support for exploration of recovery strategies that can be applied when networking failures occur

SoS and Dependability

Independence

Distribution

Evolution

Emergence

Challenge:

CSs evolve independently, need to preserve dependability under evolutions of the SoS

Approach:

Model-based traceability facilitates impact analysis to understand the impact of unexpected changes

SoS and Dependability

Independence

Distribution

Evolution

Emergence

Challenge:

Need to analyse the emergent dependability of the SoS, but constituents systems may be black boxes

Approach:

Document fault assumptions for constituents and provide tool support for analysis of emergent dependability properties

SoS and Dependability

SoS failure *"a deviation of the service provided by the SoS from expected (correct) behaviour"*

SoS error *"the part of the SoS state that can lead to its subsequent service failure"*

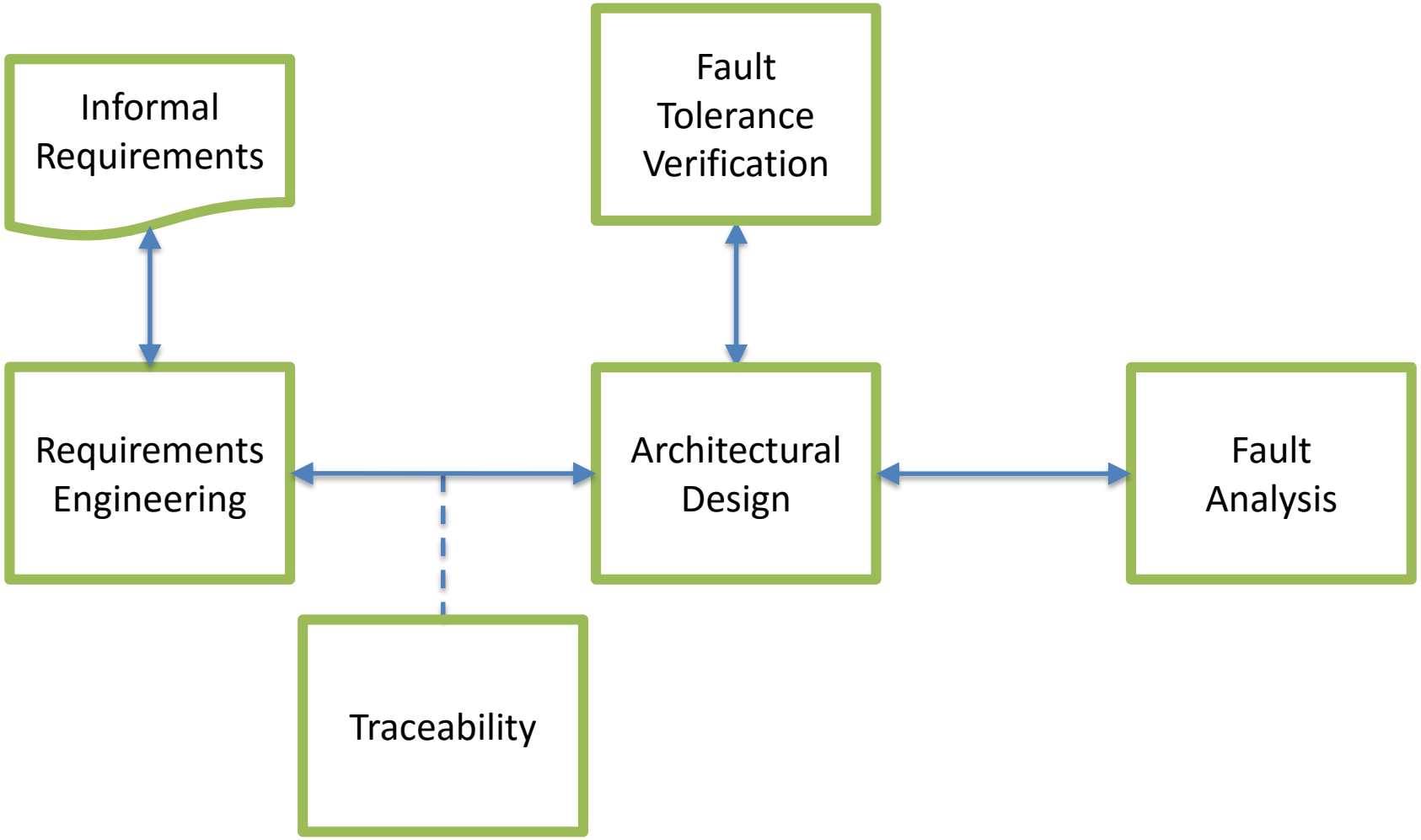
SoS fault *"the adjudged or hypothesised cause of an error"*

The focus of our work is on **fault-tolerant SoSs**

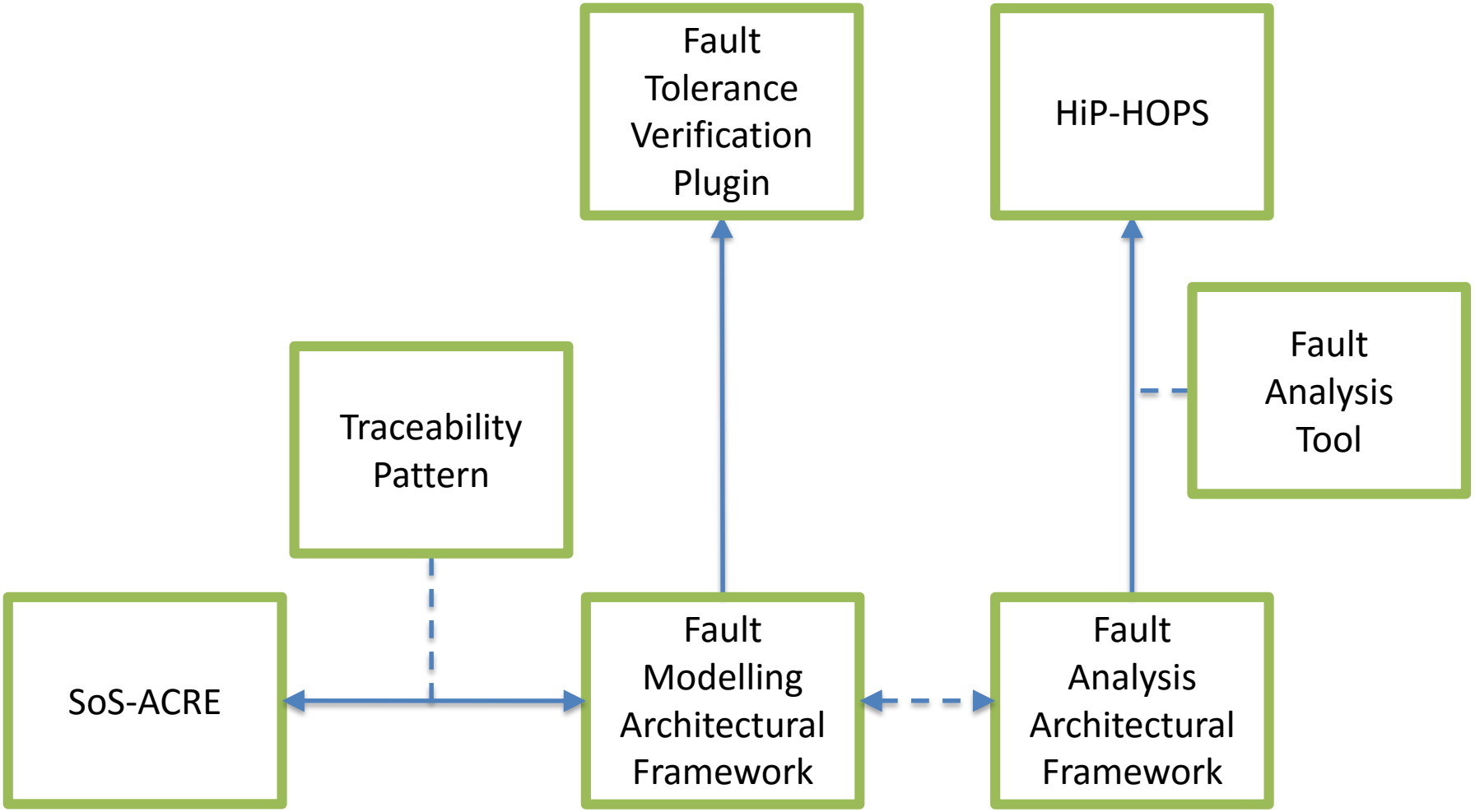
- preventing SoS failures from arising in the presence of faults
- achieved by detecting errors and conducting system recovery

Avizienis, A.; Laprie, J.-C.; Randell, B. & Landwehr, C., "Basic concepts and taxonomy of dependable and secure computing," *Dependable and Secure Computing, IEEE Transactions on* , vol.1, no.1, pp.11,33, Jan.-March 2004.

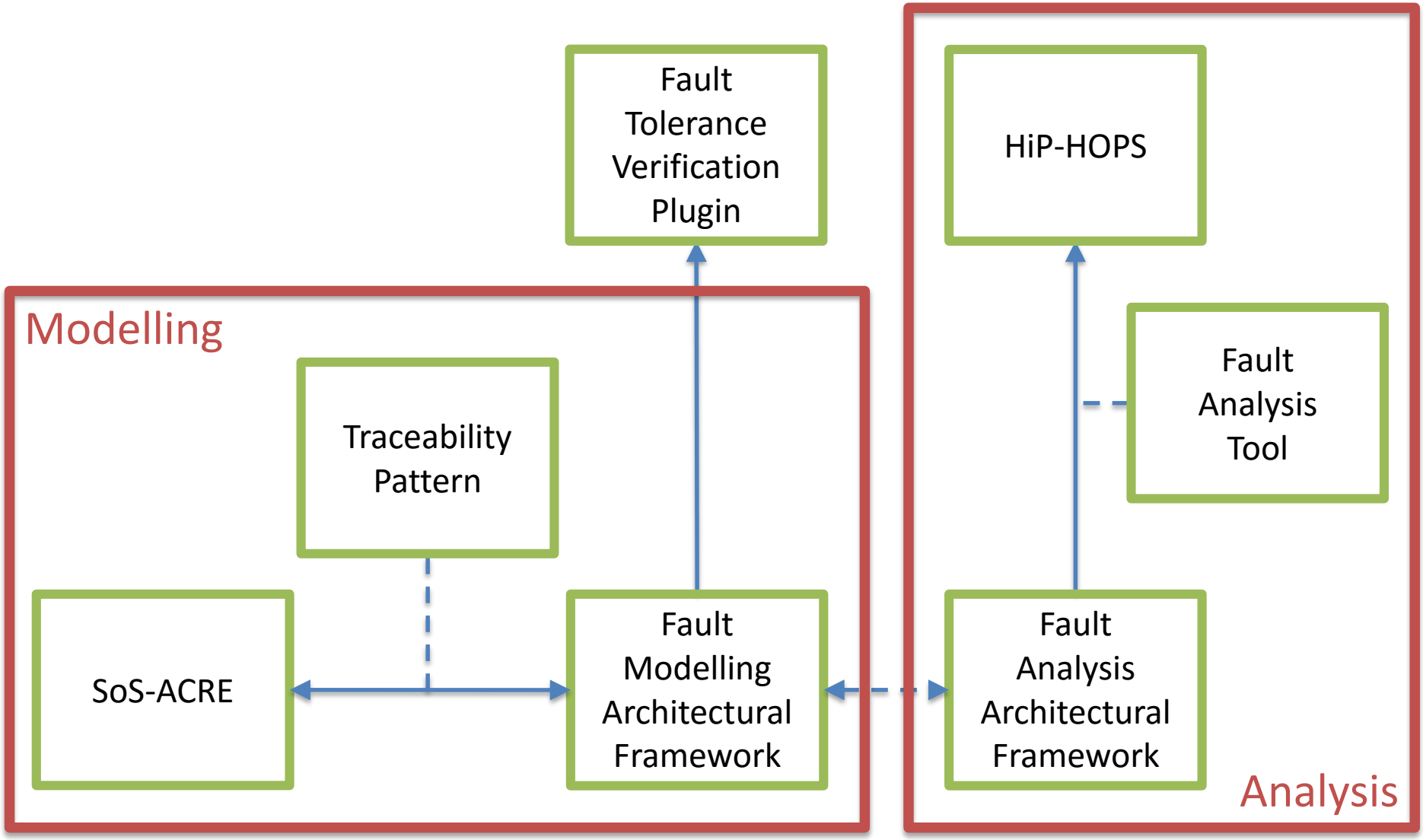
Methodology



Technologies

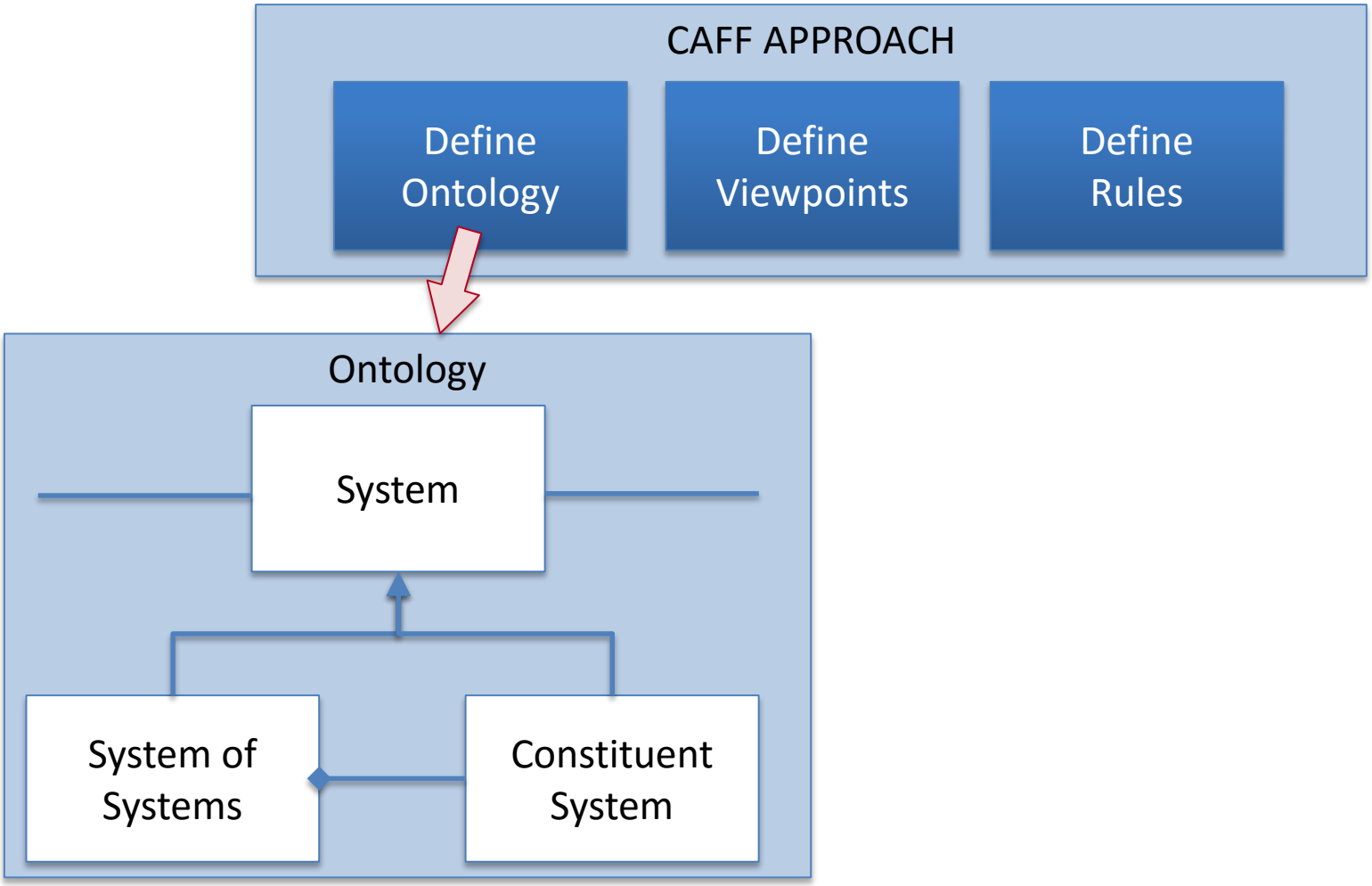


Technologies

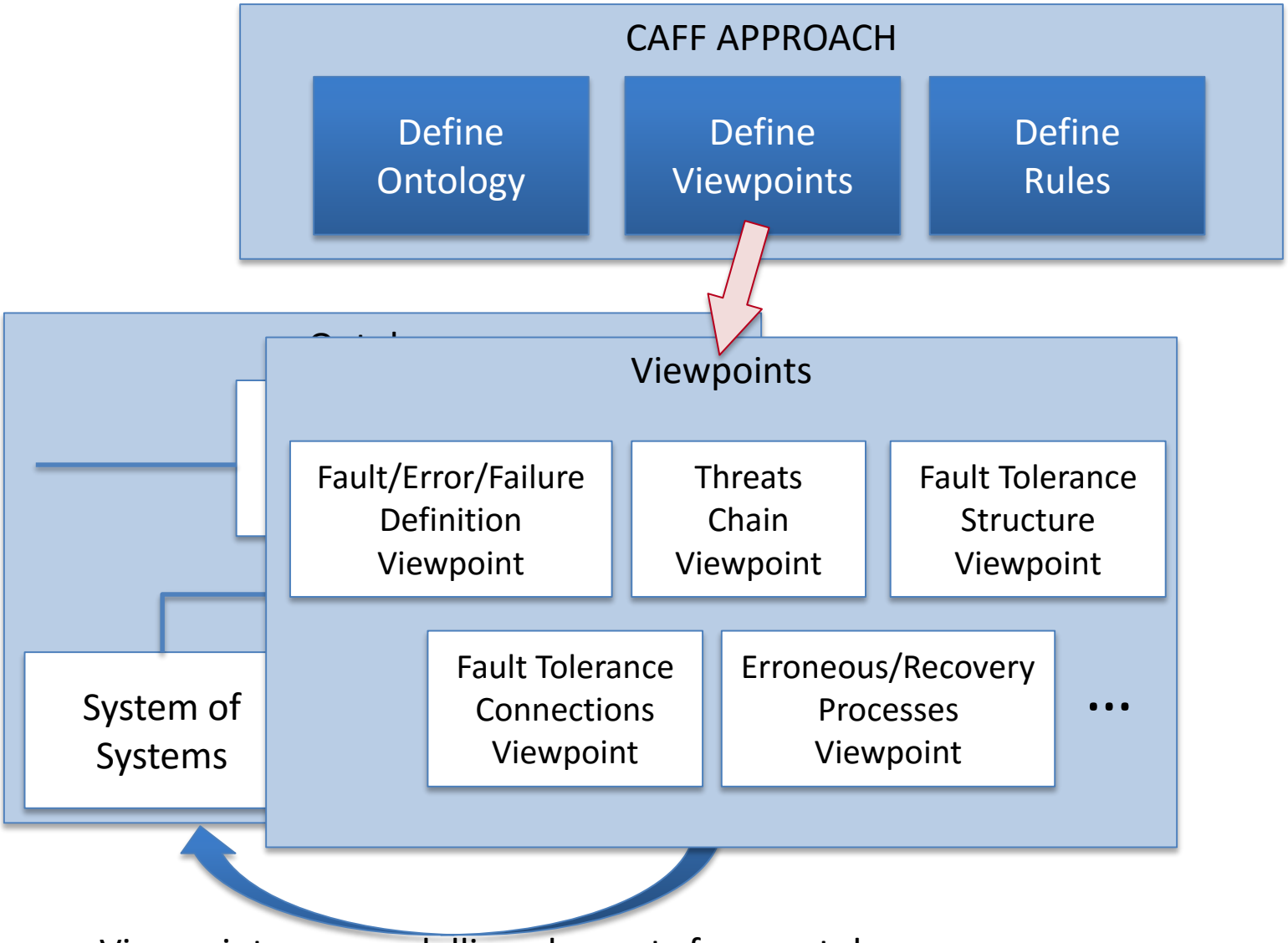


COMPASS AF Framework (CAFF) Approach

COMPASS AF Framework (CAFF) Approach

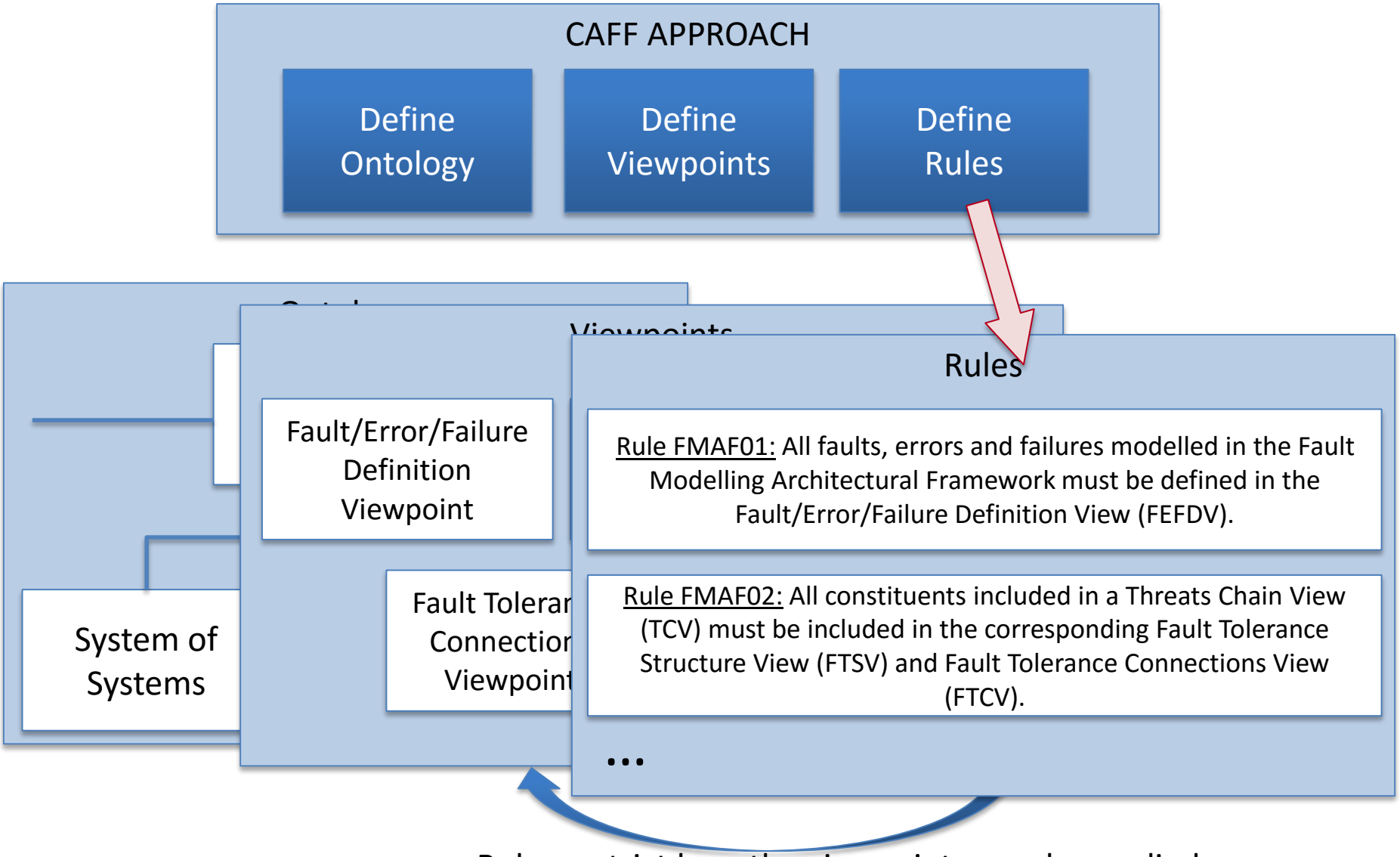


COMPASS AF Framework (CAFF) Approach



Viewpoints use modelling elements from ontology

COMPASS AF Framework (CAFF) Approach



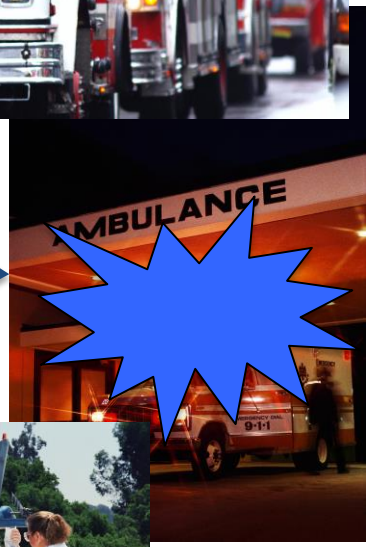
Outline

- Introduction
- **Engineering Fault-Tolerant SoSs**
  - **Case Study**
  - **Requirements Engineering**
  - **Fault Modelling**
  - **Requirements Tracing**
  - **Fault Analysis**
- Conclusions

Emergency Response SoS

Emergency Response SoS

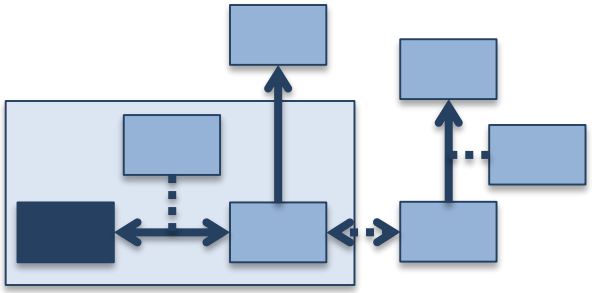
Emergency Response SoS



Emergency Response SoS

Requirements Engineering

- Apply SoS-ACRE: a structured way of engineering and managing the requirements of an SoS

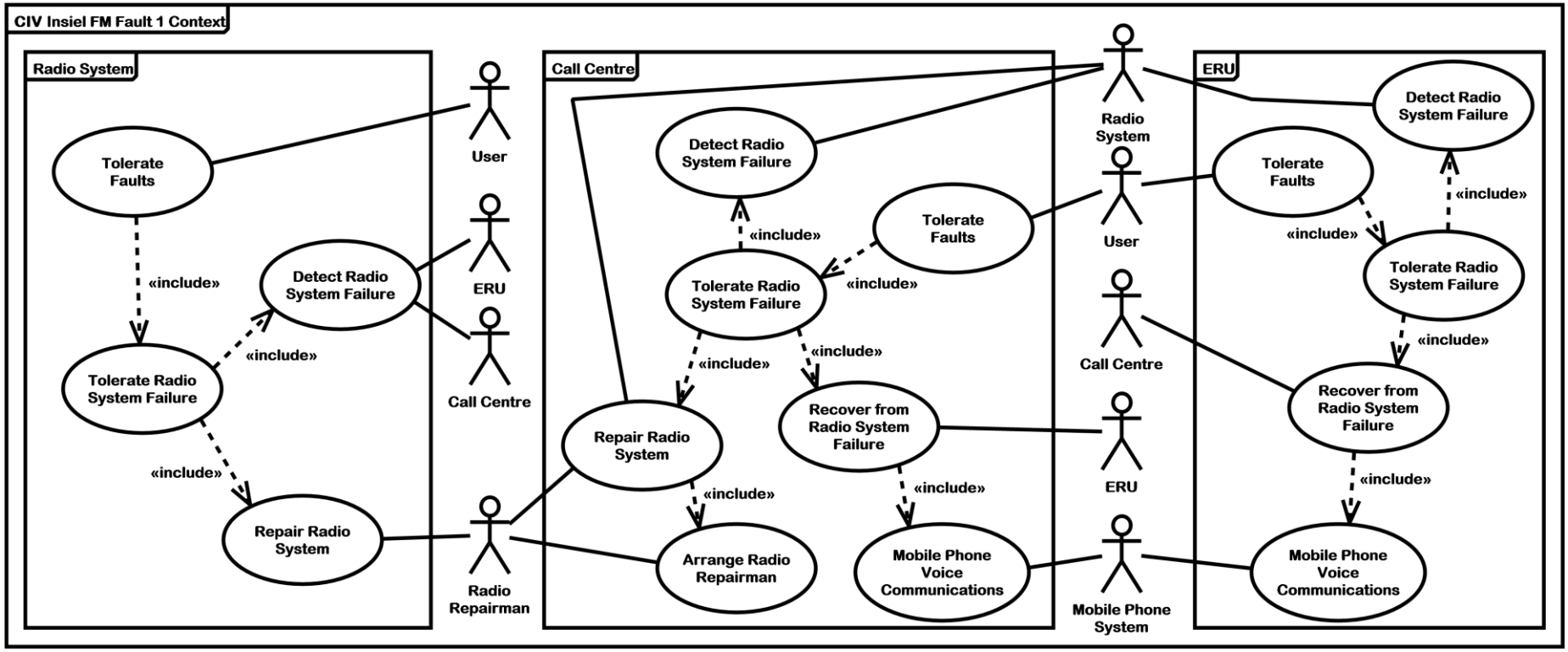


We focus on the requirements *engineering* processes

- Key elements of SoS-ACRE for fault tolerance:
  - define the fault tolerance requirements of the SoS (what faults)
  - examine the fault tolerance requirements in the context of the CSs
  - identify error detection and recovery scenarios

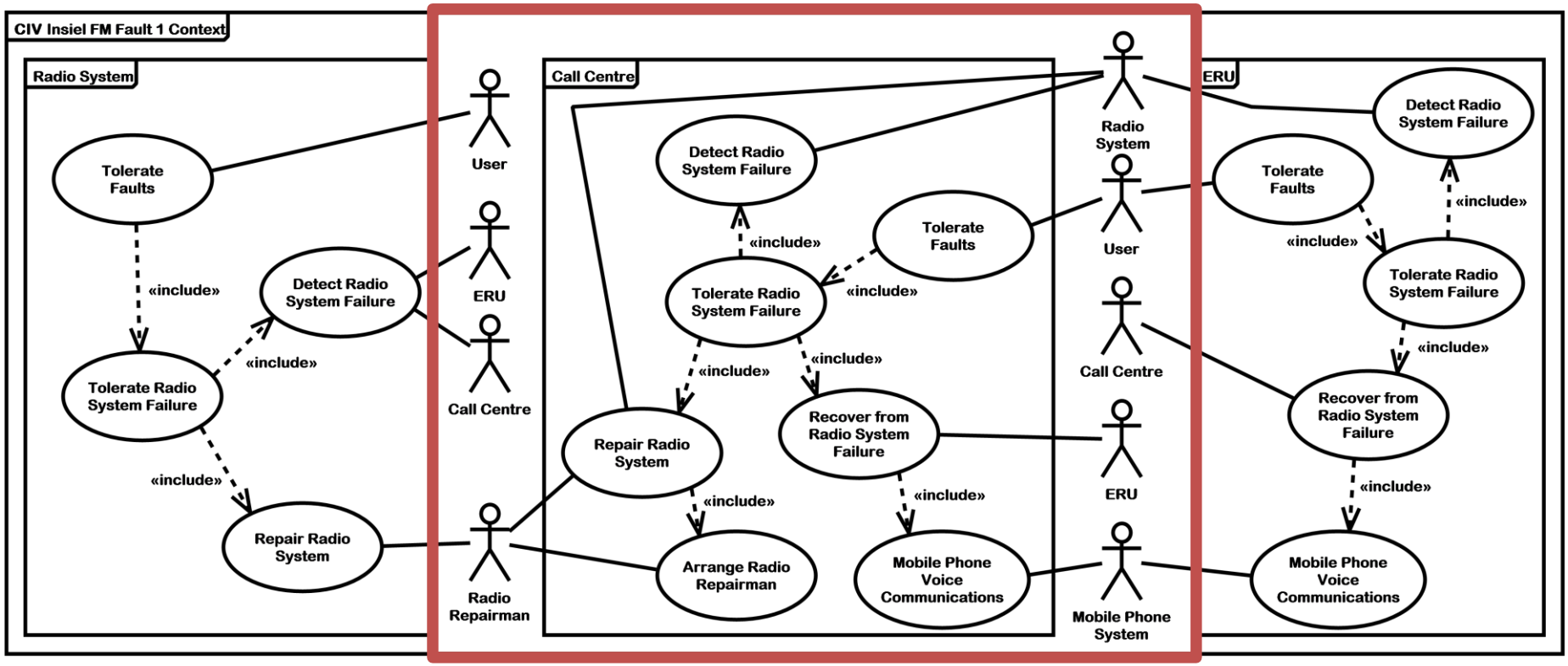
ER SoS: Context Interaction

Requirement: Tolerate Radio System Failure

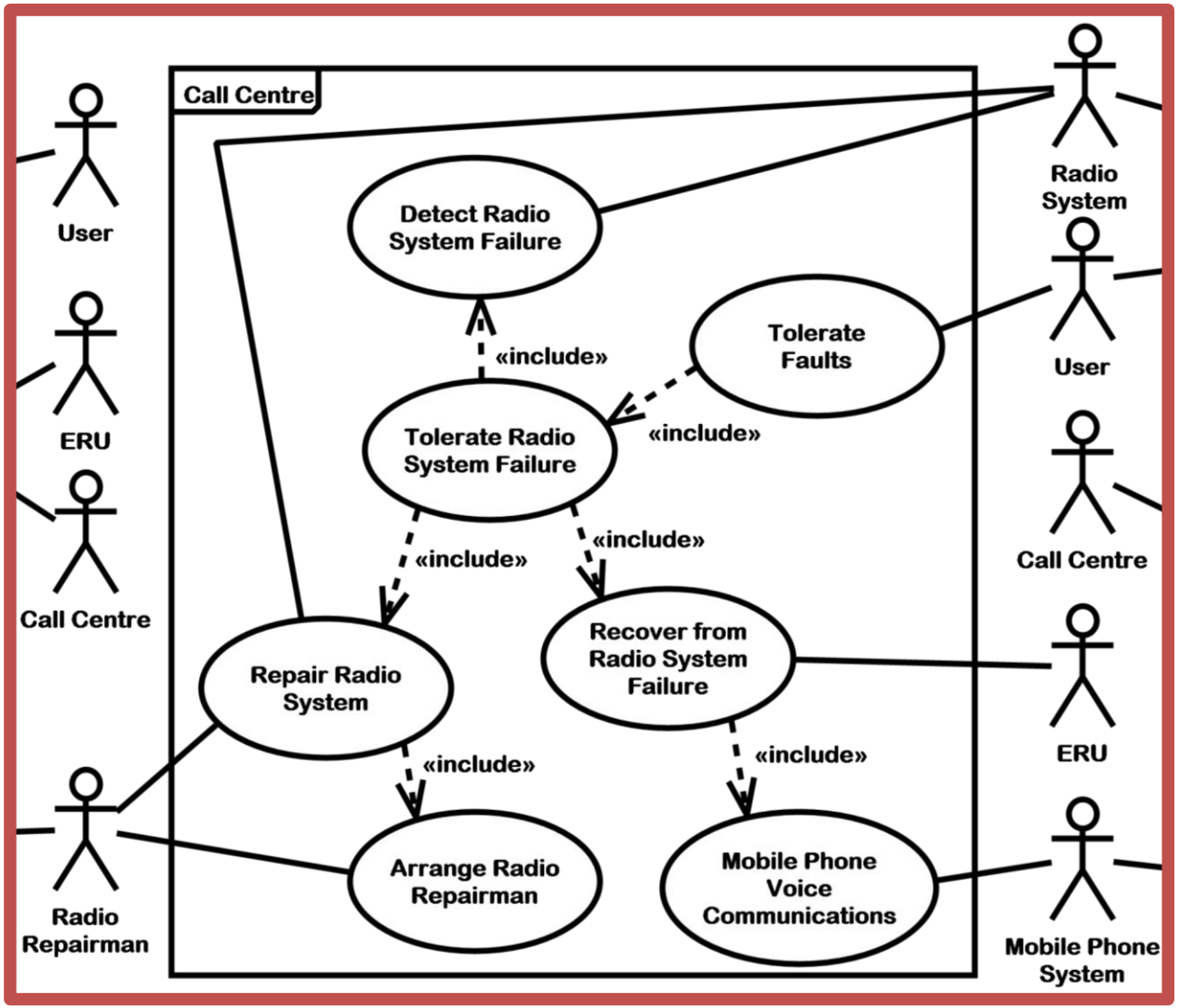


ER SoS: Context Interaction

Requirement: Tolerate Radio System Failure

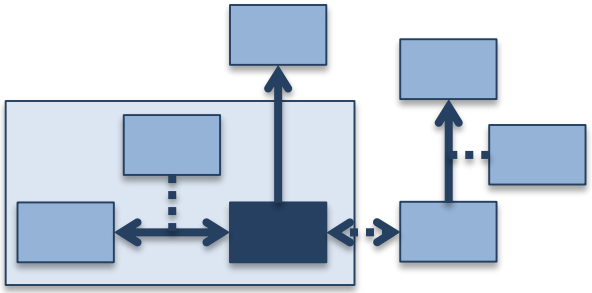


ER SoS: Context Interaction

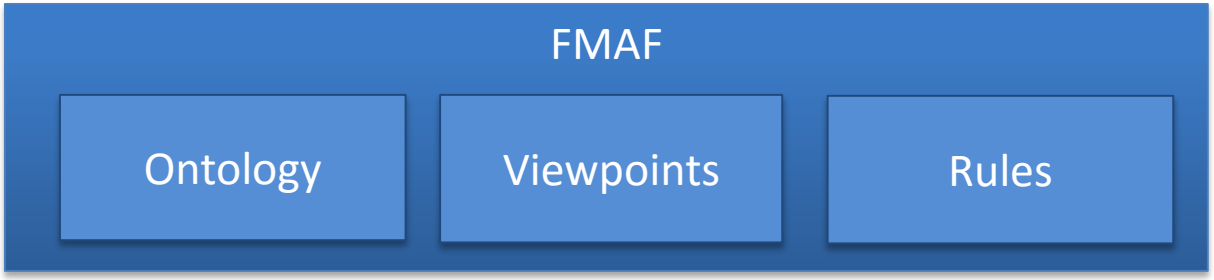


Fault Modelling

- Defined a Fault Modelling Architectural Framework (FMAF) + SysML profile

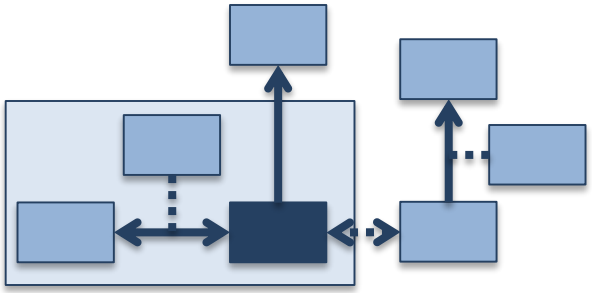


- Prompts an SoS engineer to consider the impact of faults at the early stages of design
- A coherent set of views & concepts for designing fault-tolerant SoSs

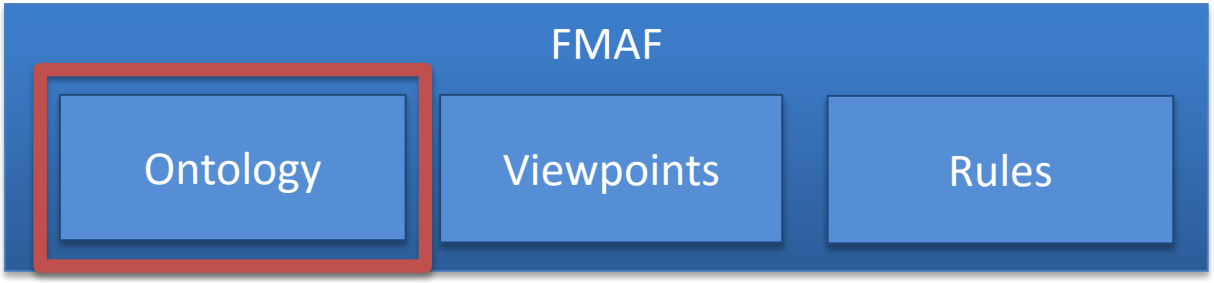


Fault Modelling

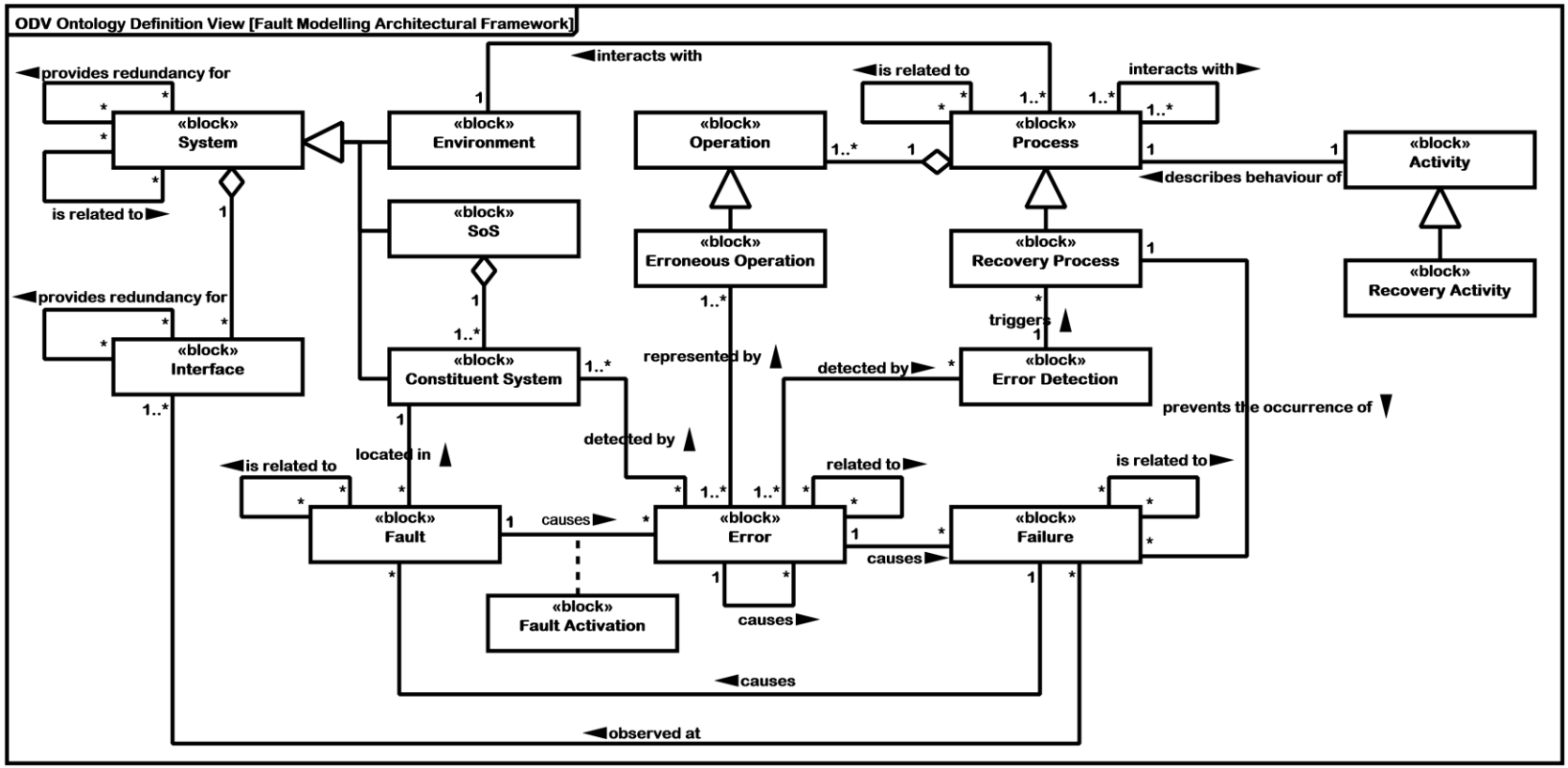
- Defined a Fault Modelling Architectural Framework (FMAF) + SysML profile



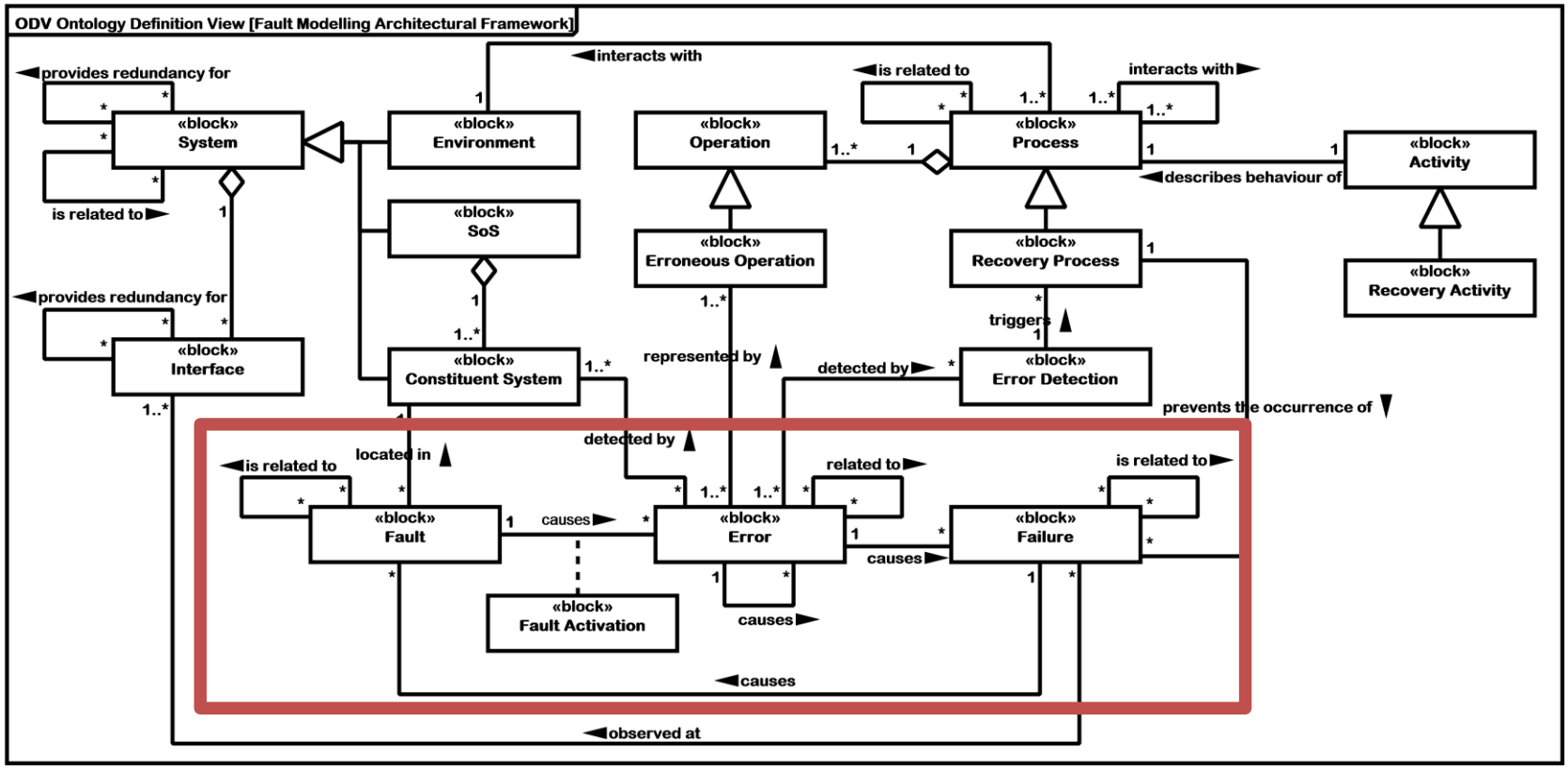
- Prompts an SoS engineer to consider the impact of faults at the early stages of design
- A coherent set of views & concepts for designing fault-tolerant SoSs



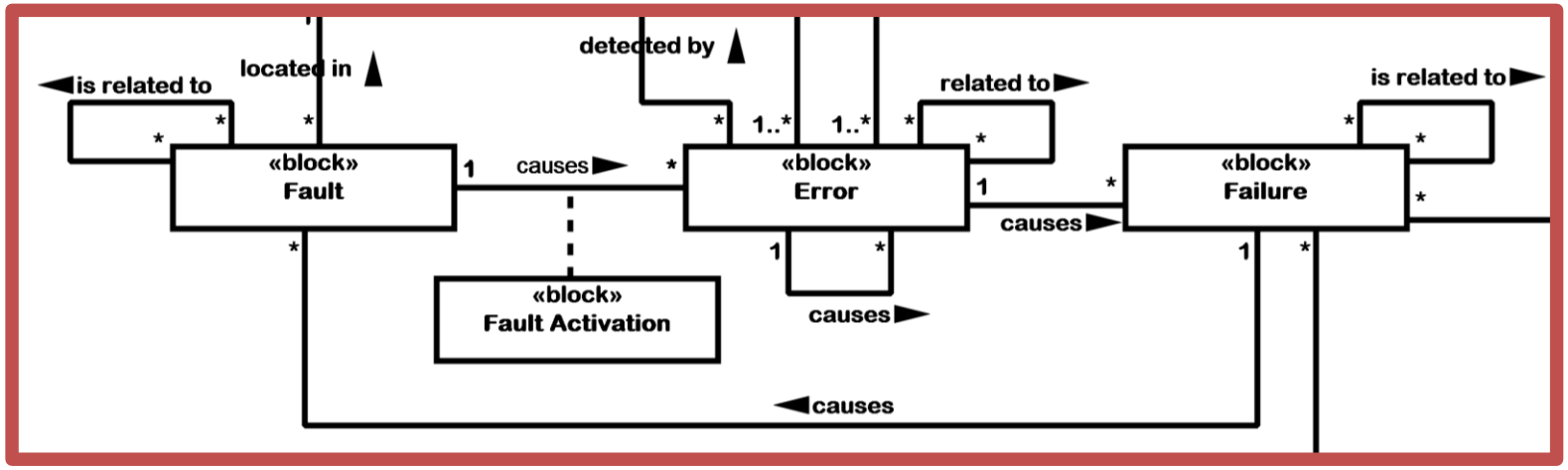
FMAF: Ontology Definition



FMAF: Ontology Definition

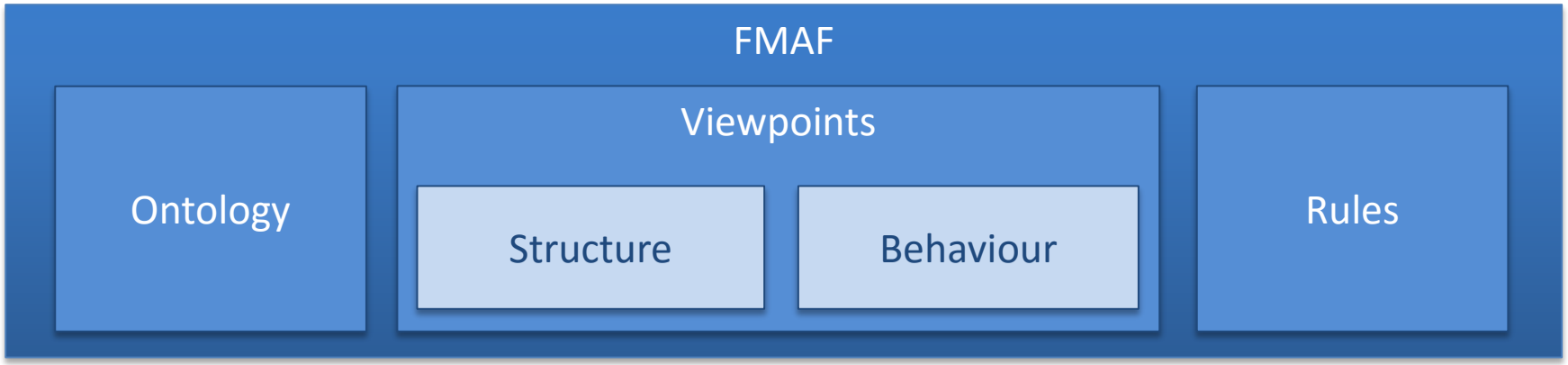


FMAF: Ontology Definition

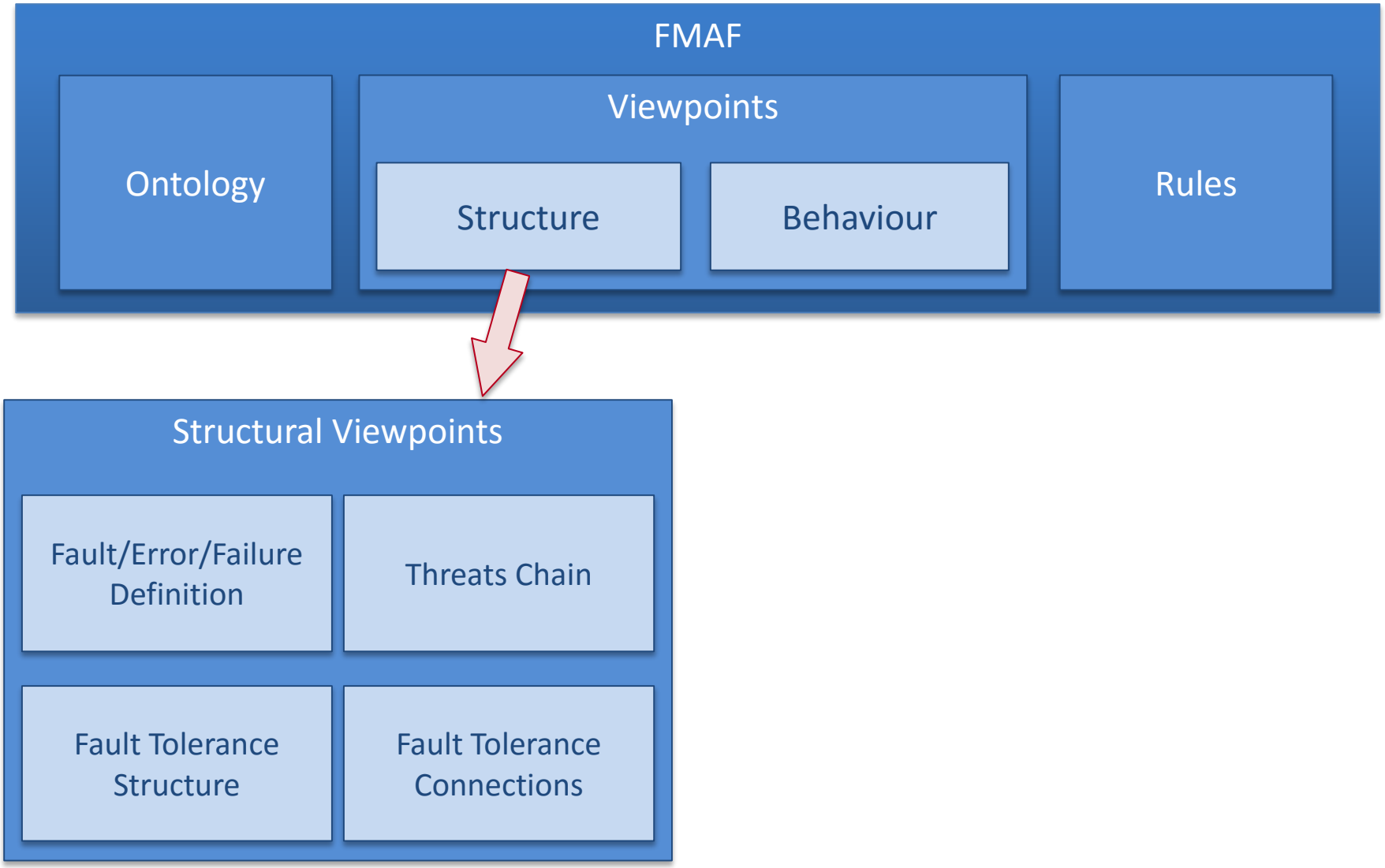


- **SoS failure**: *"a deviation of the service provided by the SoS from expected (correct) behaviour"*
- **SoS error**: *"the part of the SoS state that can lead to its subsequent service failure"*
- **SoS fault**: *"the adjudged or hypothesised cause of an error"*

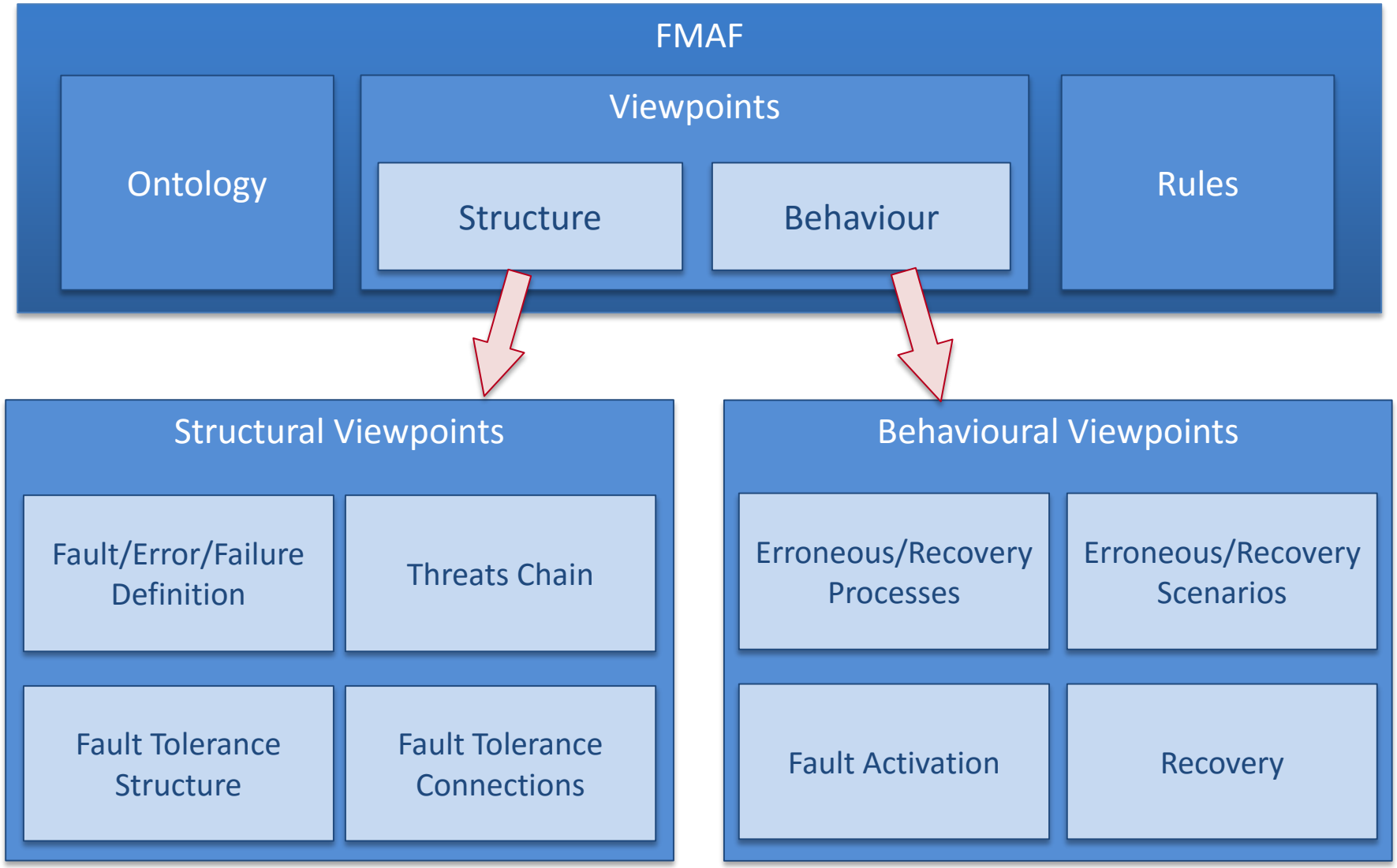
FMAF: Viewpoints



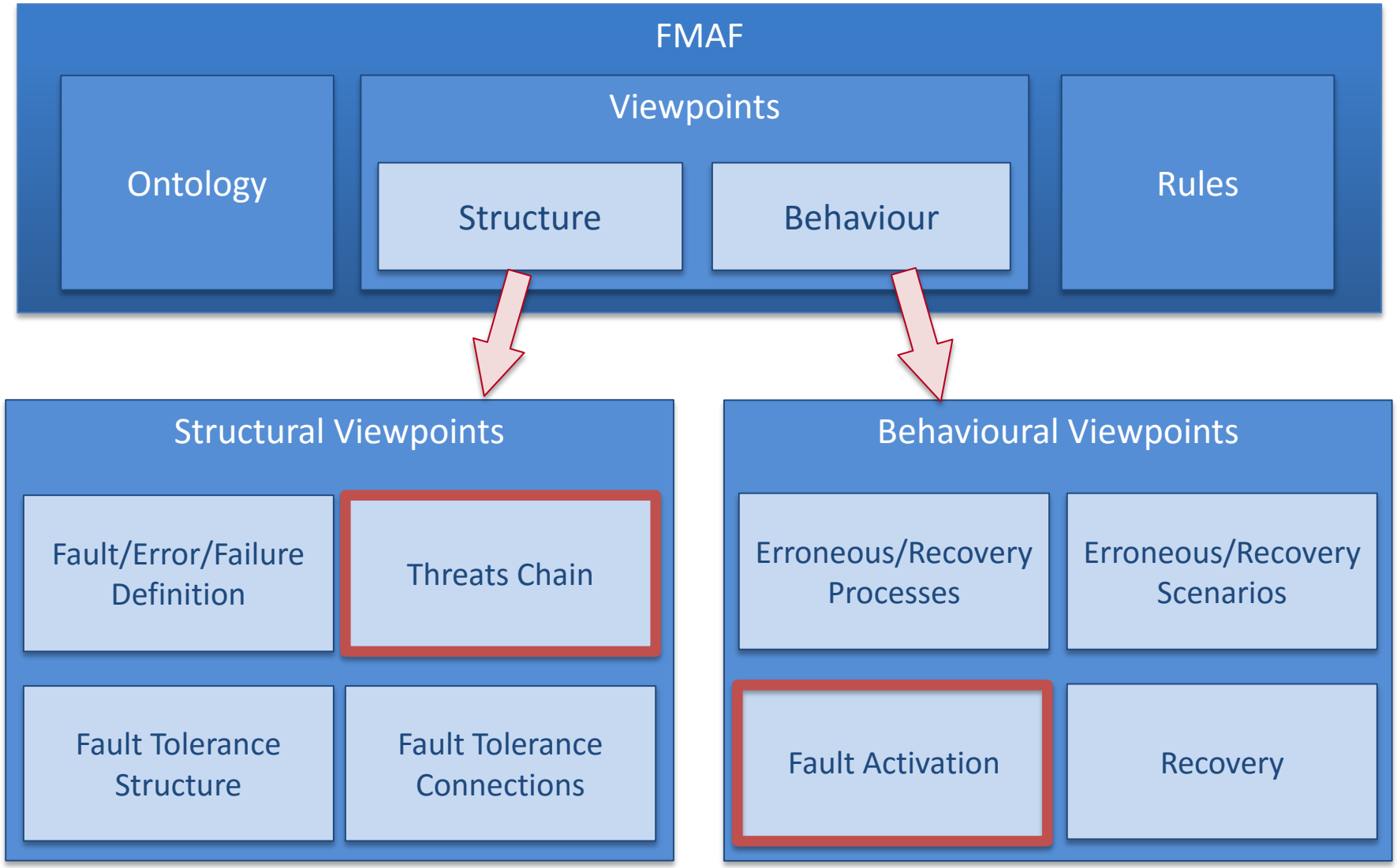
FMAF: Viewpoints



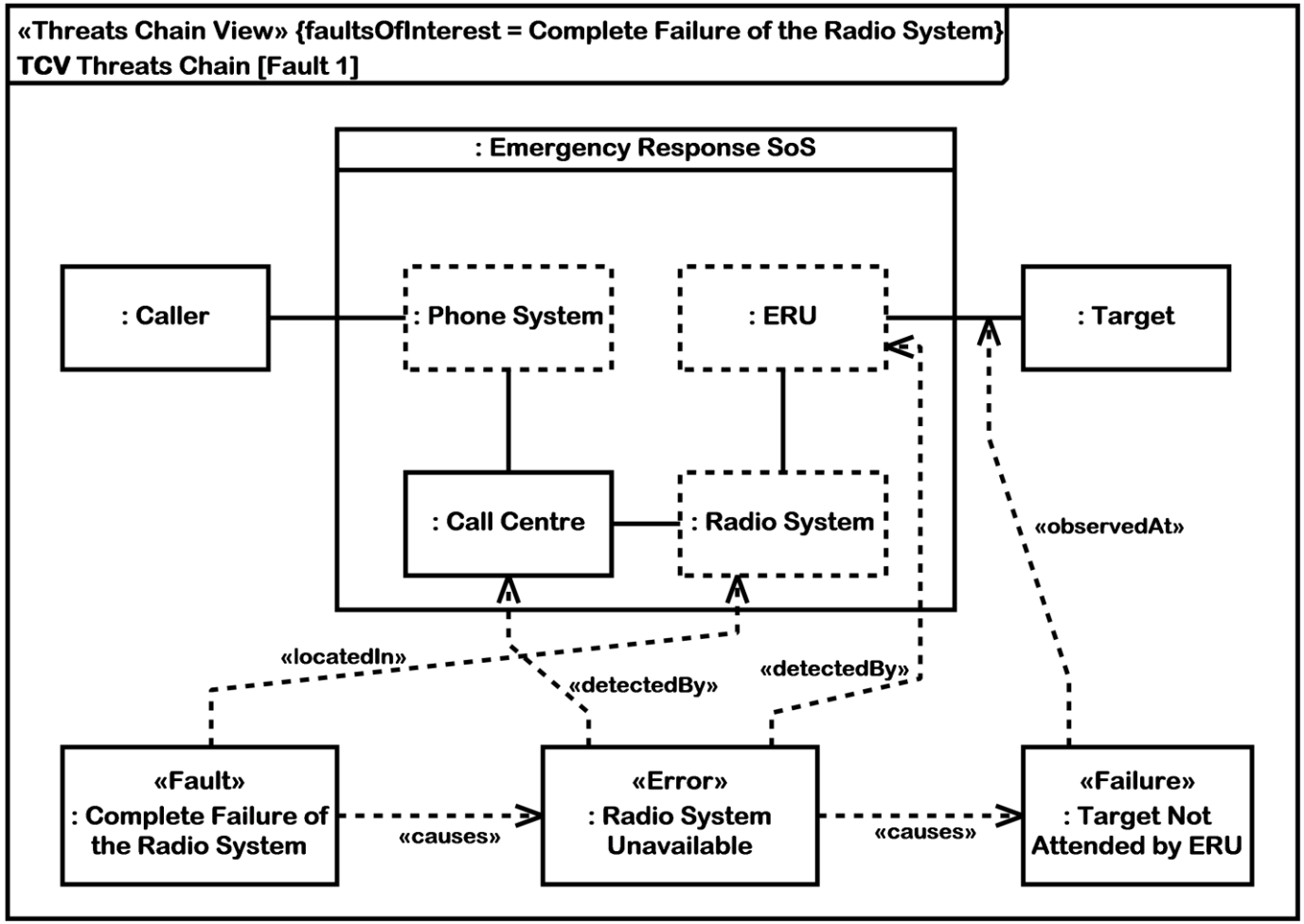
FMAF: Viewpoints



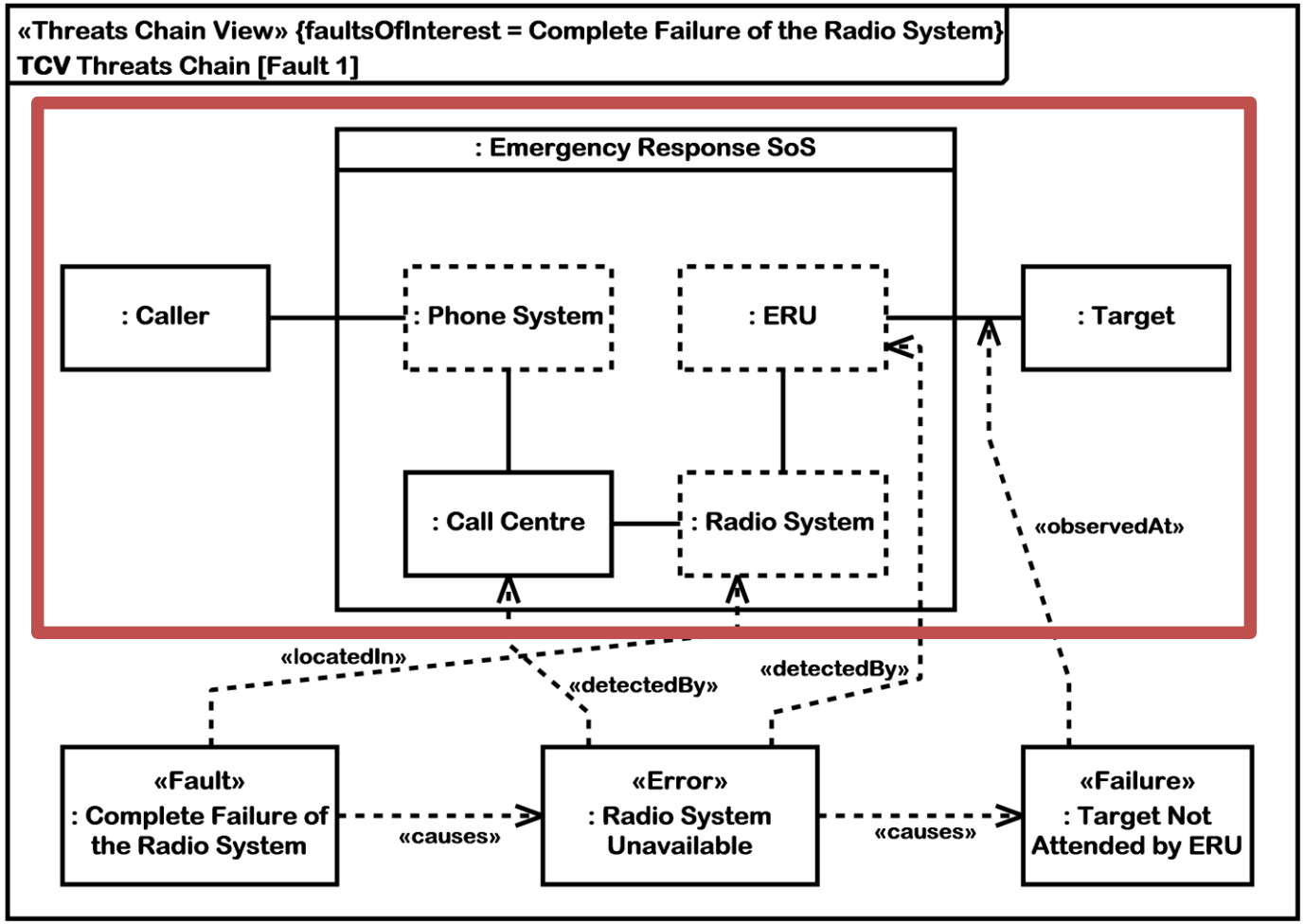
FMAF: Viewpoints



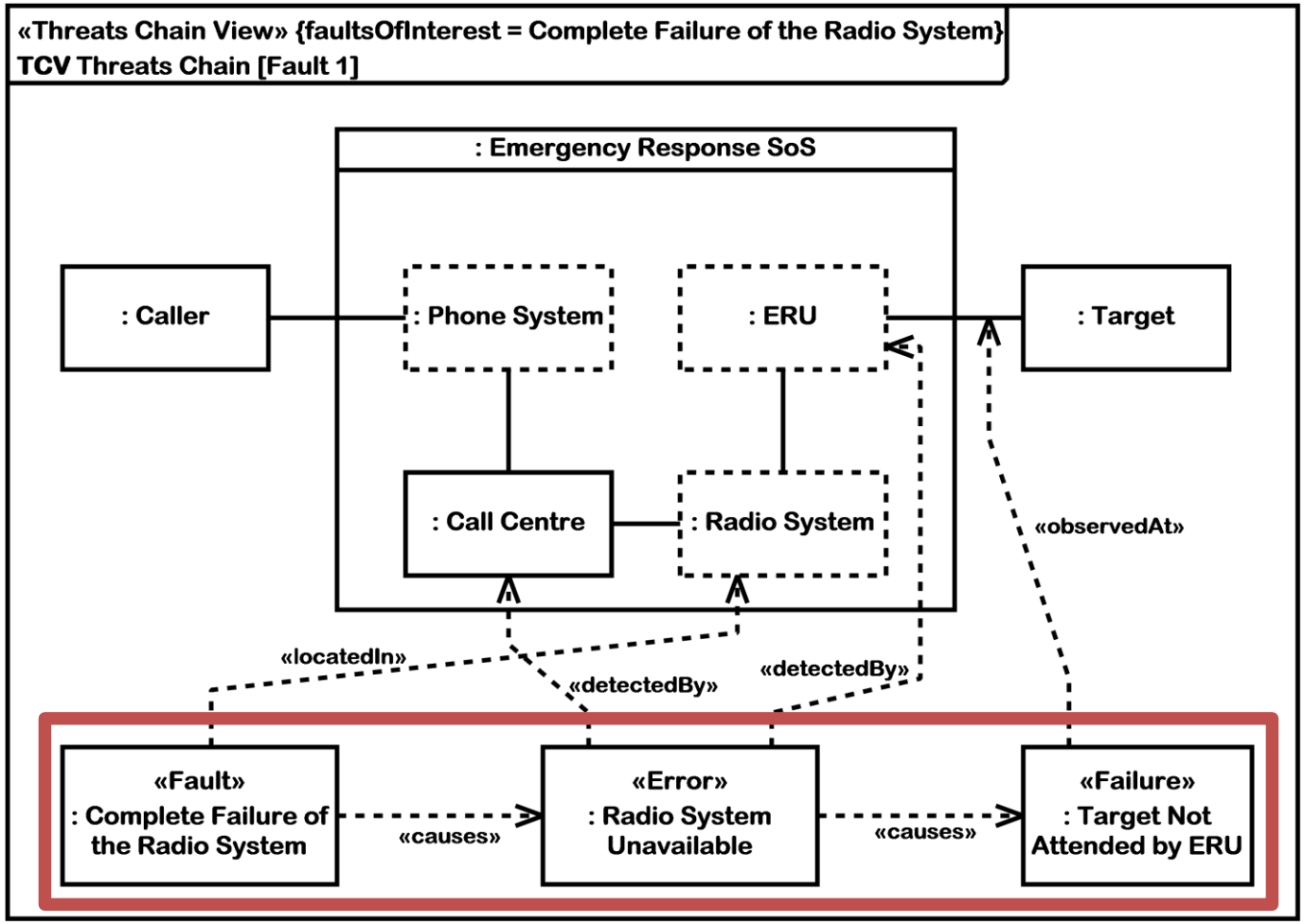
ER SoS: Threats Chain



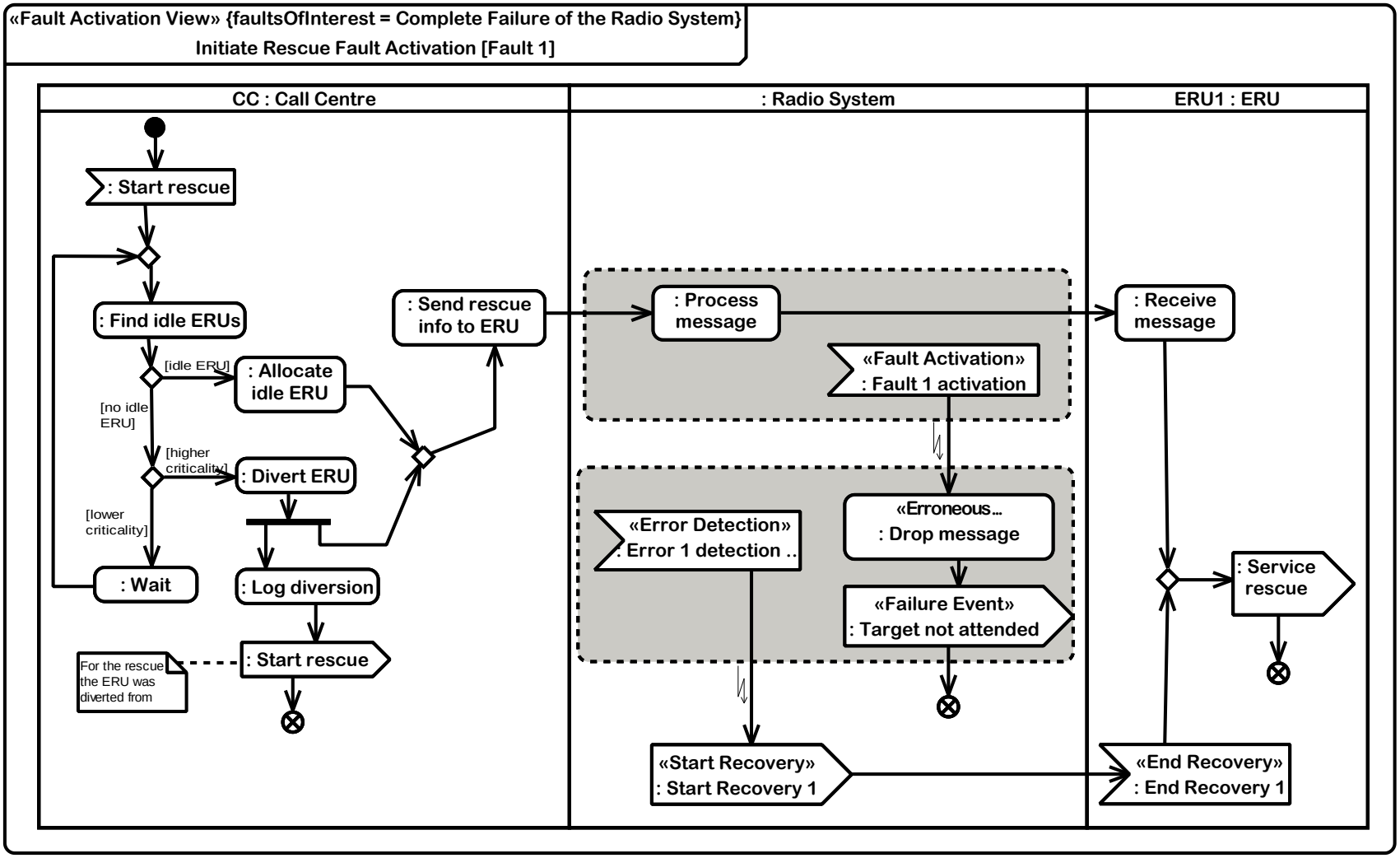
ER SoS: Threats Chain



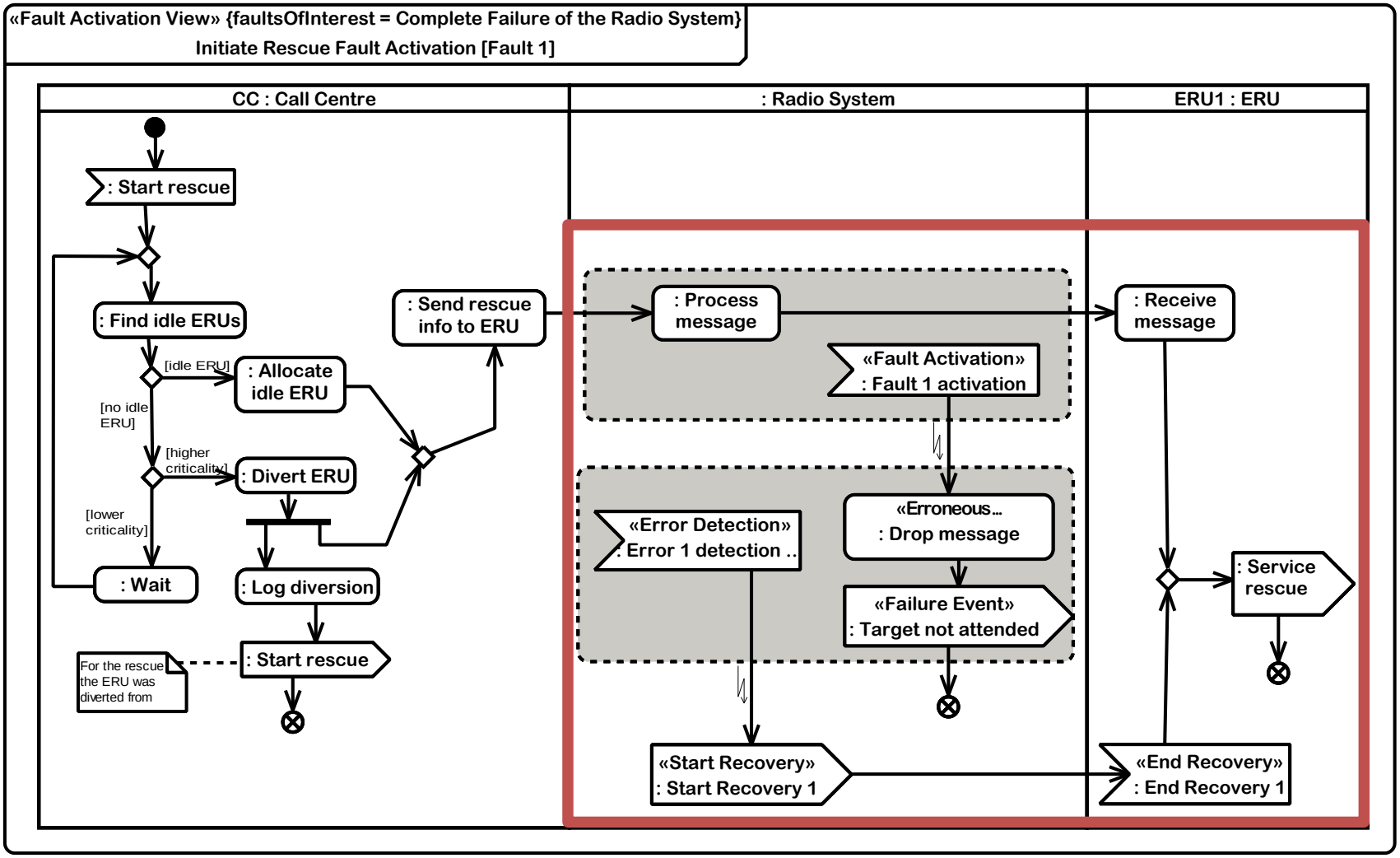
ER SoS: Threats Chain



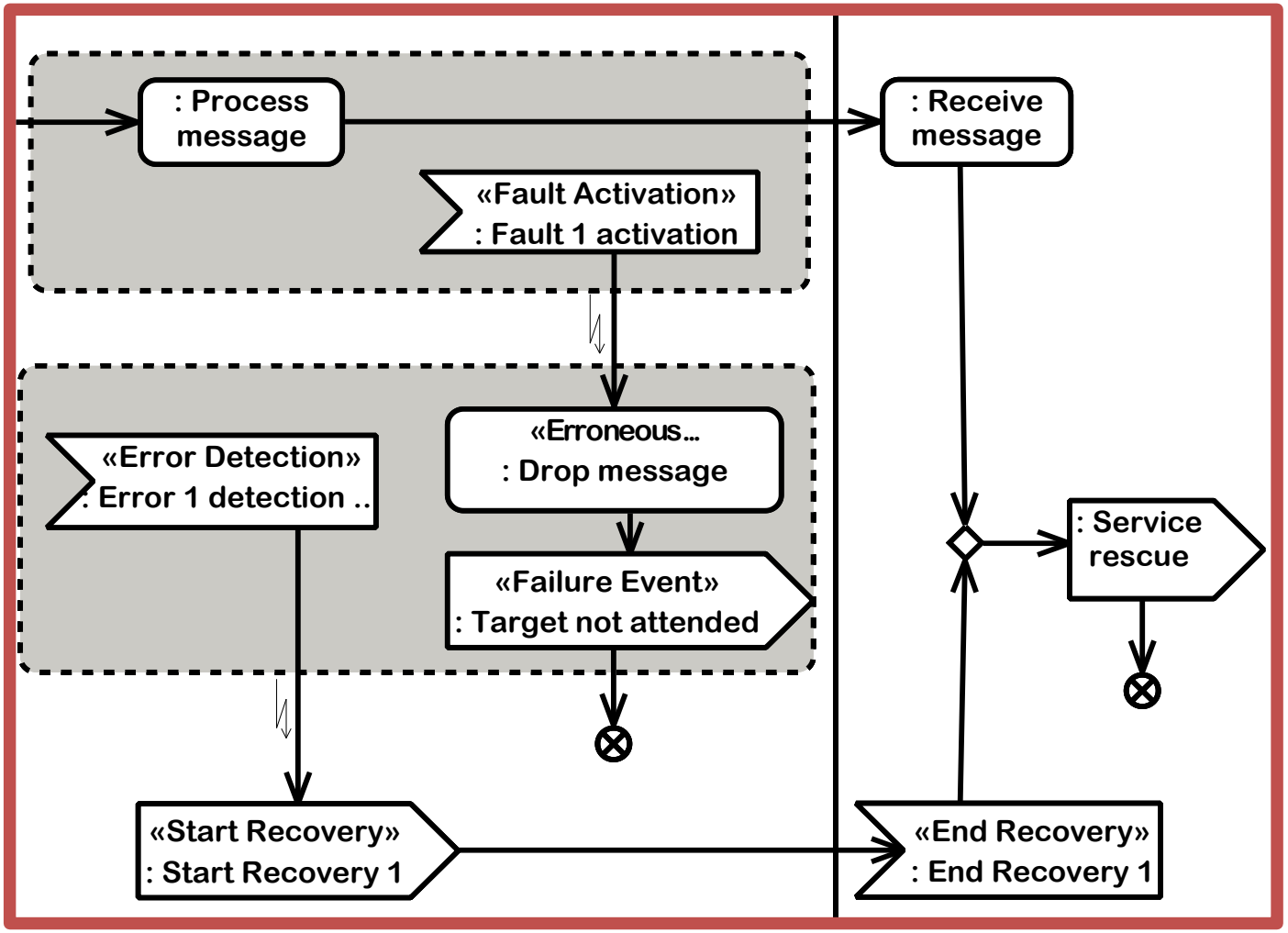
ER SoS: Fault Activation



ER SoS: Fault Activation

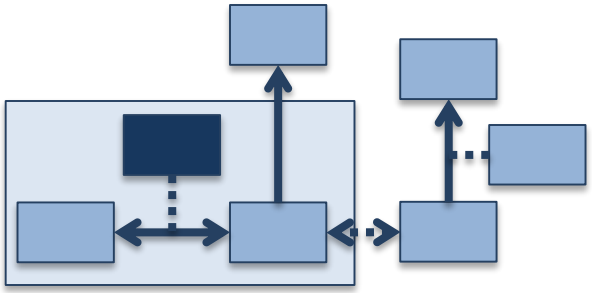


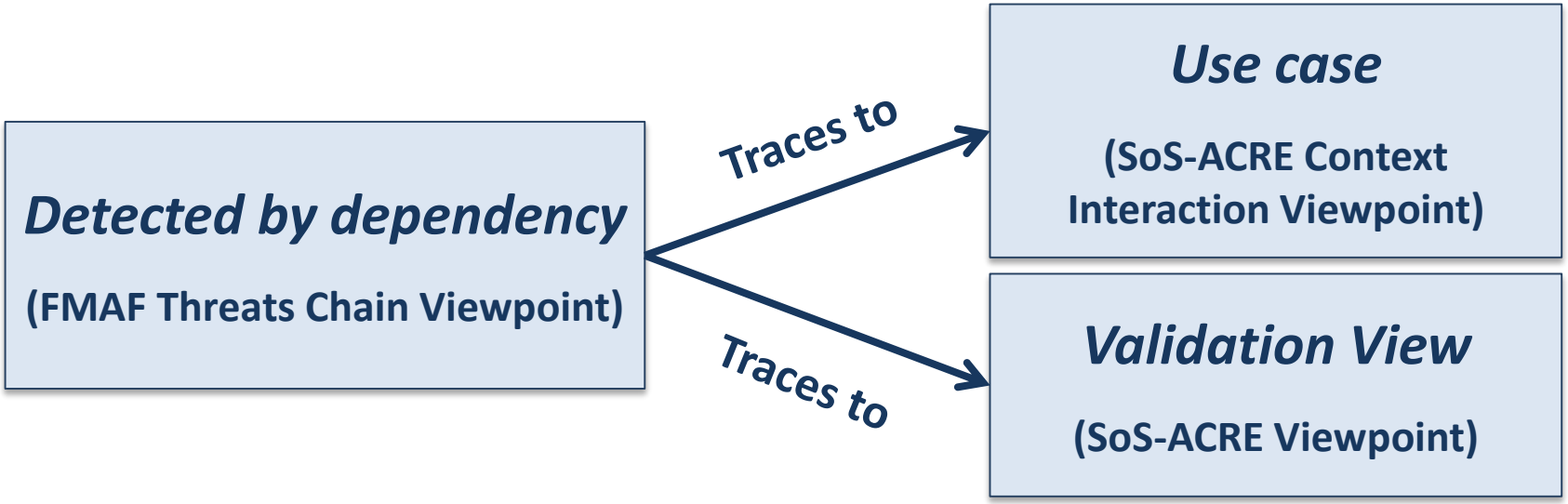
ER SoS: Fault Activation



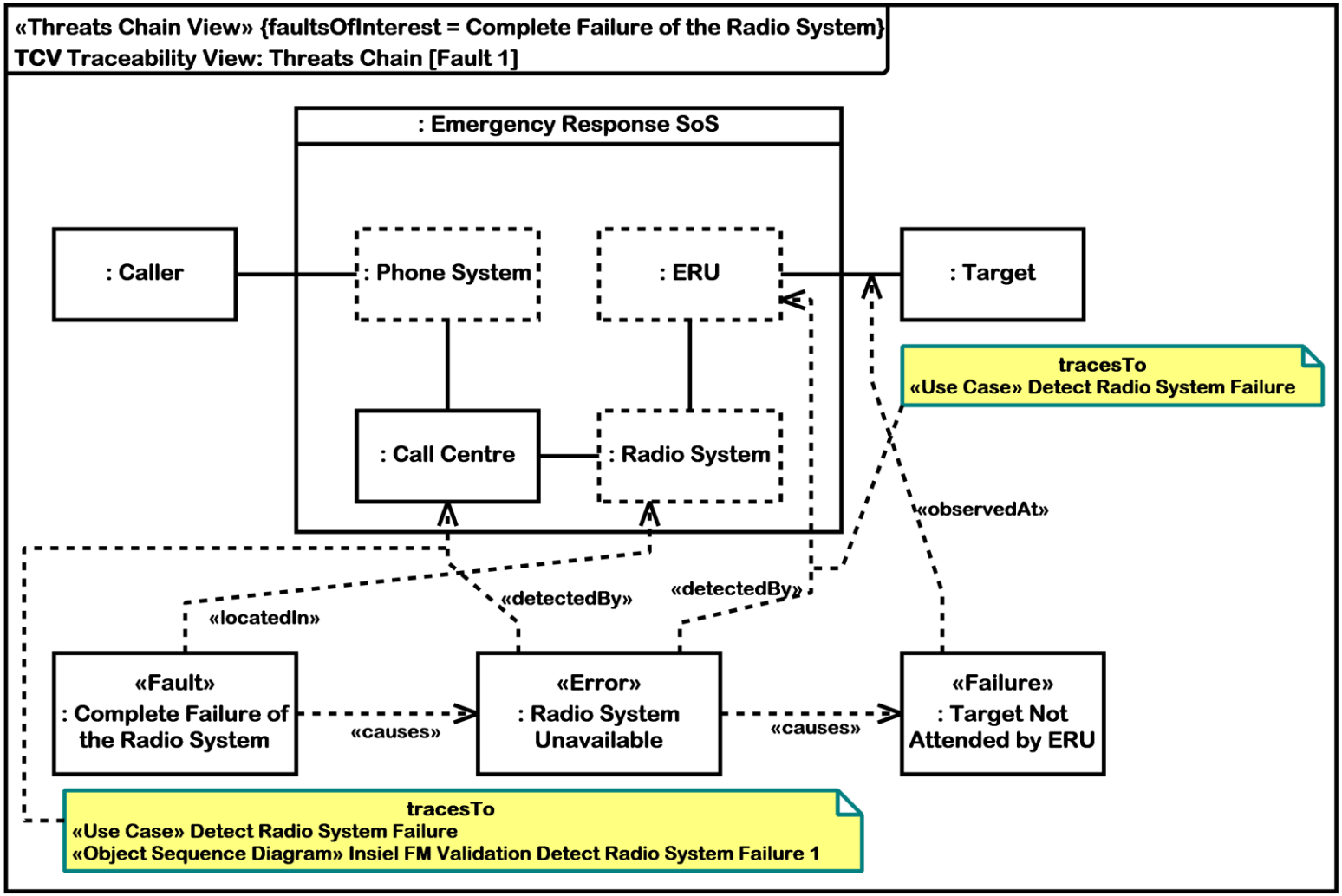
Requirements Tracing

- Traceability Pattern
  - a structured way of defining traceability
  - in this case we will use it for fault tolerance requirements traceability



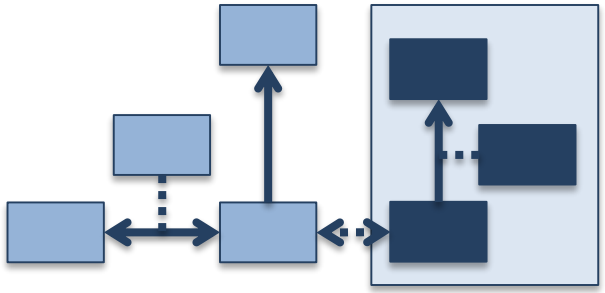


Requirements Tracing (ER SoS)



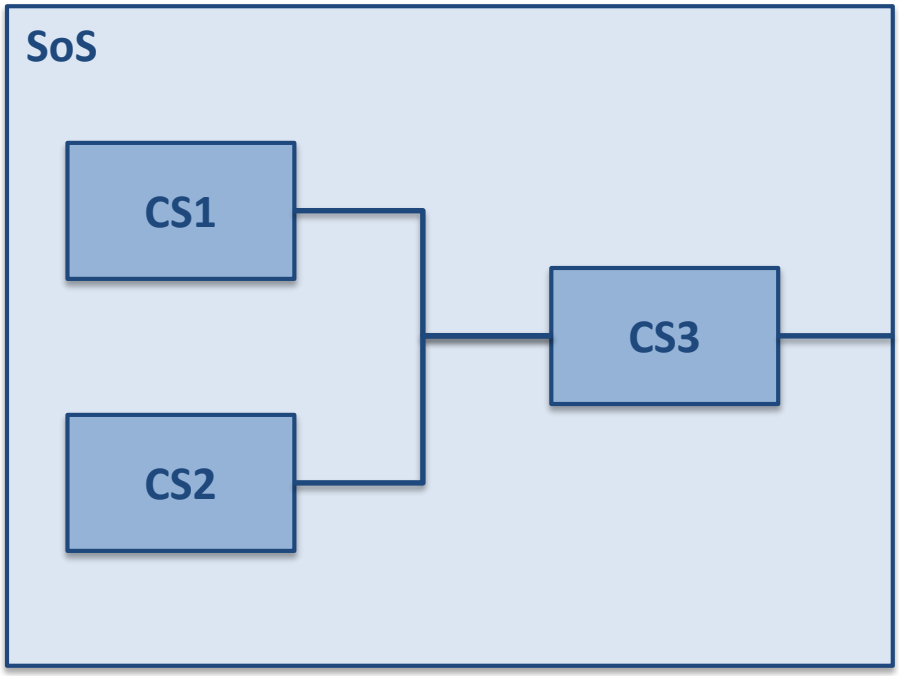
Fault Analysis

- Assists an SoS engineer in analysing the impact of CS/component level faults on SoS functionalities

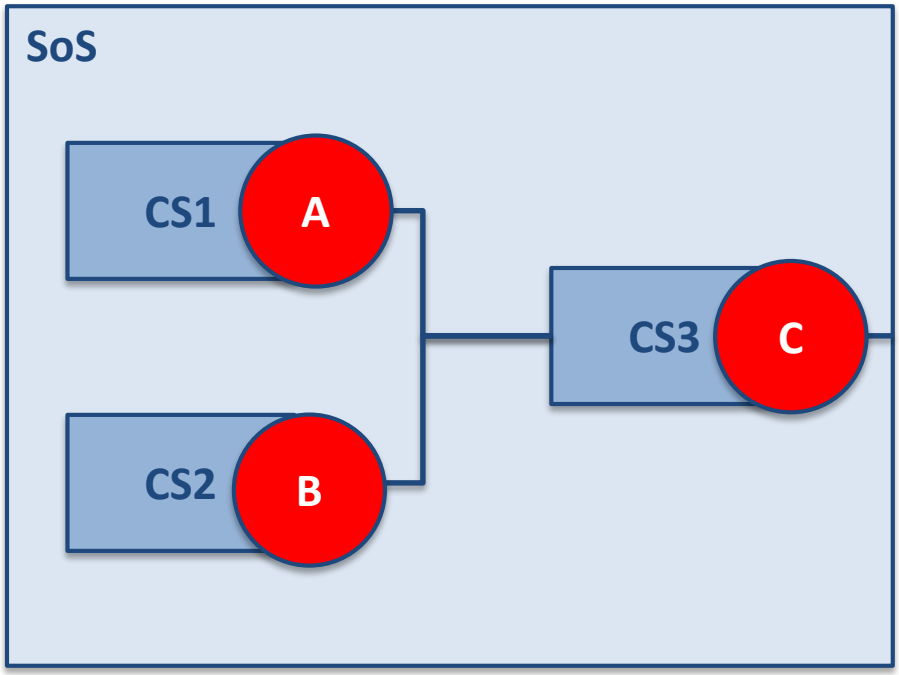


- Provide modelling and tool support for static fault analysis
  - Fault Analysis Architectural Framework (FAAF) + SysML profile
  - 'Ergonomic profile' and translation tool for Atego Modeler
- Fault analysis capabilities provided by HiP-HOPS tool
  - Fault Tree Analysis
  - Failure Modes and Effects Analysis

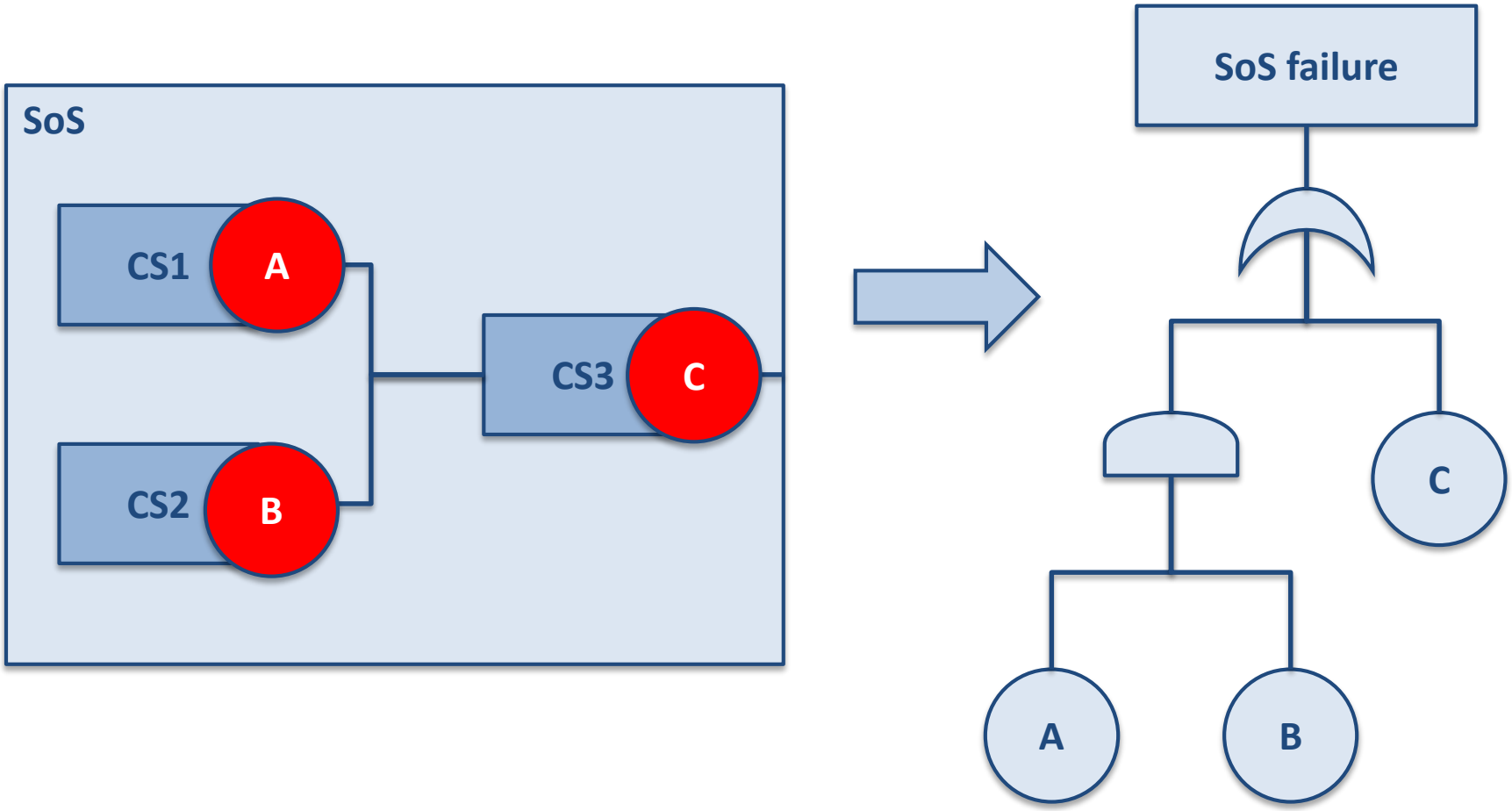
Fault Analysis: Example



Fault Analysis: Example

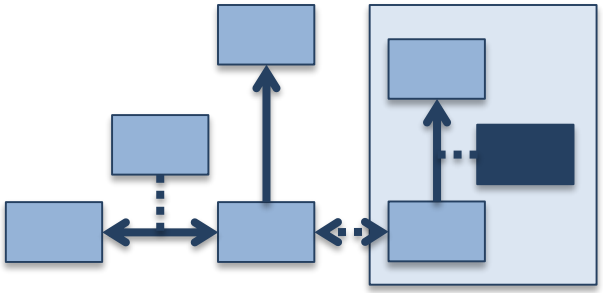


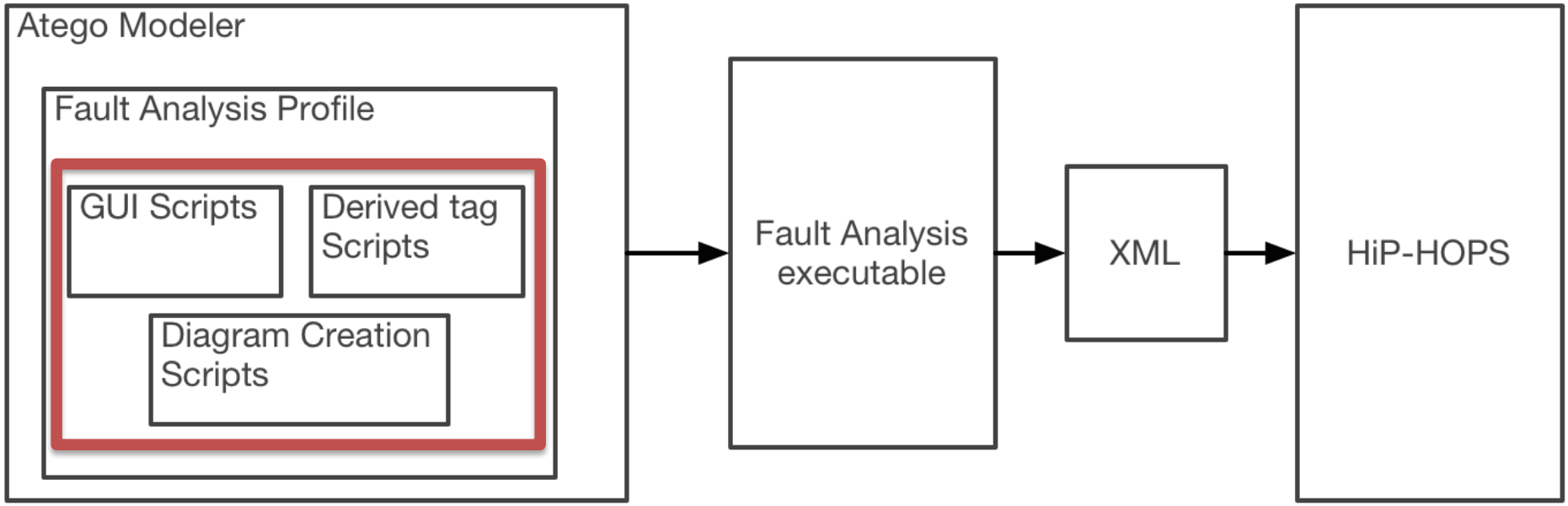
Fault Analysis: Example



Tool Overview

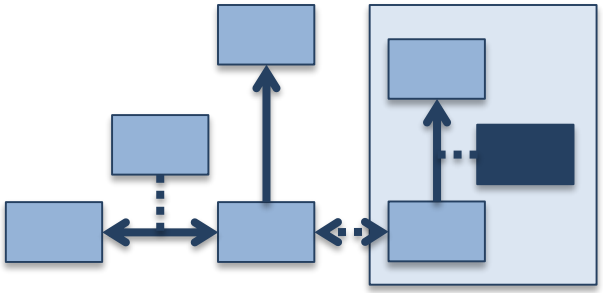
1. Ergonomic profiling

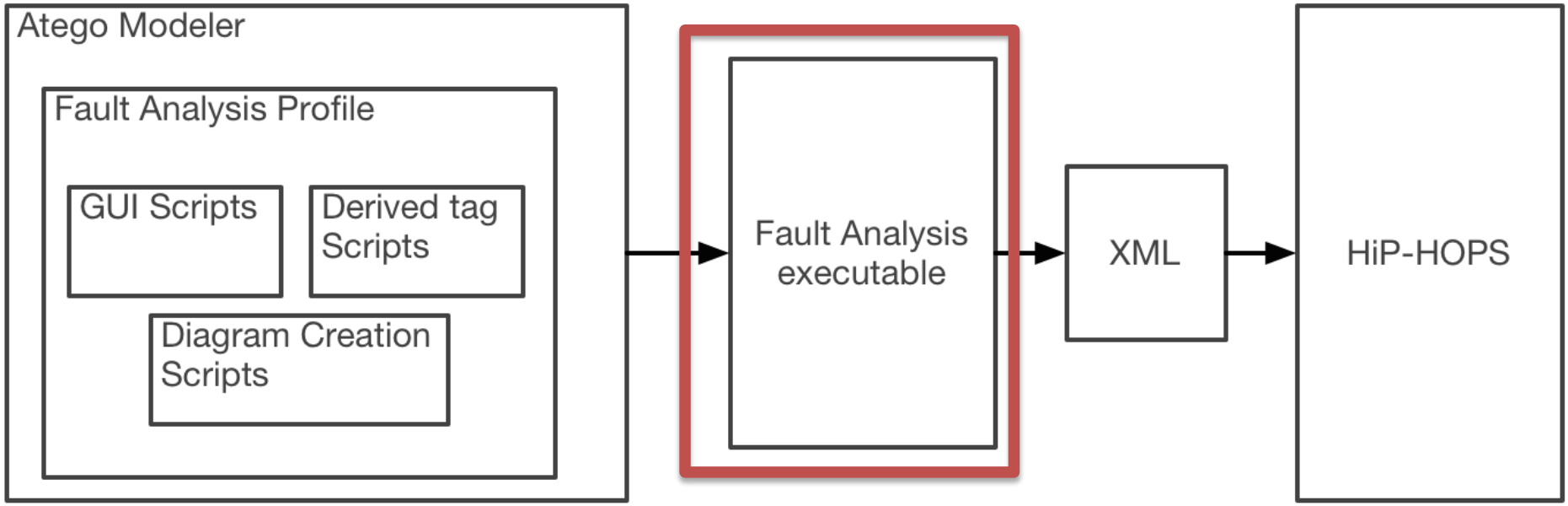




Tool Overview

1. Ergonomic profiling
2. **Exporting models for analysis**





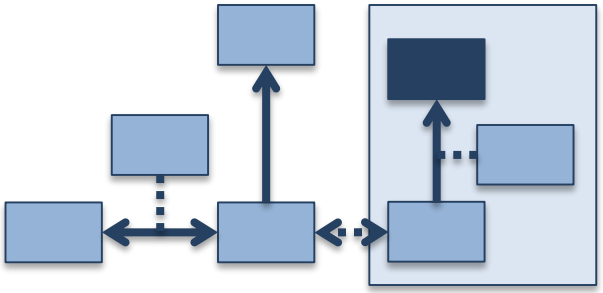
Fault Tree View: Value-Country_A_TMS.actOutA

FaultTrees | **FMEA**

Top Event (Effect)	Value-Country_A_TMS.actOutA
Description	N/A
System Unavailability	0.263319
Severity	0

Fault Trees | **Cut Sets**

- FaultTrees
  - Value-Country_A_TMS.actOutA
    - Value-Country_A_TMS.actOutA ()
      - Value-Country_A_TMS.actOutA (69)
        - Country_A_TMS.CountryA_CommsFailure (66)
        - Country_A_TMS.Country_A_Actuator_1.Actuator_Failure_A (91)
        - Country_A_TMS.Country_A_TCC.TCC_Failure_A (77)
        - Country_A_TMS.Country_A_Sensor_2.Sensor2_Failure_A (98)
        - Country_A_TMS.Country_A_Sensor_1.Sensor1_Failure_A (74)
        - Country_B_TMS.CountryB_CommsFailure (1)
        - Country_B_TMS.Country_B_TCC.TCC_Failure_B (11)
        - Country_B_TMS.Country_B_Sensor_1.Sensor1_Failure_B (5)
        - Country_A_TMS.Country_A_Actuator_2.Actuator_Failure_A (101)
        - Country_A_TMS.Country_A_Actuator_3.Actuator_Failure_A (108)
        - Country_A_TMS.Country_A_Actuator_4.Actuator_Failure_A (115)



Outline

- Introduction
- Engineering Fault-Tolerant SoSs
- **Conclusions**

Conclusions

Systems of systems have characteristics that motivate a structured approach to fault modelling and analysis of SoSs:

- Modelling support for requirements engineering, architectural design, requirements tracing of fault-tolerant SoSs
- Modelling and tool support for fault tolerance verification and static fault analysis

Future directions:

- Tool-supported integration of the fault modelling and fault analysis architectural frameworks and SysML profiles
- Support for modes of operation, either functional modes or degraded modes (e.g. partial functionality in presence of errors)
- Multi-objective optimisation, trading off attributes such as dependability, cost and energy consumption

zoe.andrews@ncl.ac.uk @ZoeAndrewsNcl
claire.ingram@ncl.ac.uk @_Claire_Ingram
richard.payne@ncl.ac.uk @riffio

This work is part of the COMPASS project: research into model-based techniques for developing, maintaining and analysing SoSs

COMPASS

thecompassclub.org

http://sourceforge.net/projects/compassresearch/files/StaticFaultAnalysis/

Further Reading

- Andrews, Z.; Fitzgerald, J.; Payne, R. and Romanovsky, A. "Fault Modelling for Systems of Systems," in *Proceedings of the 11th International Symposium on Autonomous Decentralised Systems (ISADS 2013)*, March 2013, pp. 59–66.
- Andrews, Z.; Payne, R.; Romanovsky, A.; Didier, A. L. and Mota, A. 2013. "Model-based development of fault tolerant systems of systems" *in Proceedings of the* 2013 IEEE International Systems Conference (SysCon 2013).
- Ingram, C.; Riddle, S.; Fitzgerald J.; Al-Lawati, S.A.H.J.; Alrbaiyan, A., "SoS Fault Modelling at the Architectural Level in an Emergency Response Case Study" in Proceedings of the Workshop on Engineering Dependable Systems of Systems, 2014 (EDSoS 2014)
- Andrews, Z.; Ingram, C.; Payne, R.; Plat, N., SysML Fault Modelling in a Traffic Management System of Systems" in Proceedings of the 2014 IEEE International Systems of Systems Engineering Conference (SoSE 2014).
- Andrews, Z.; Ingram, C.; Payne, R.; Romanovsky, A.; Perry, S. & Holt, J., Traceable Engineering of Fault-Tolerant SoSs, *in* "International INCOSE Symposium, Las Vegas, USA", June-July 2014.

- COMPASS Deliverables D24.2, D33.3 and D43.3: http://www.compass-research.eu/
- Fault Analysis Tool: http://sourceforge.net/projects/compassresearch/files/StaticFaultAnalysis/

- Atego Modeler: http://www.atego.com/products/atego-modeler/
- HiP-HOPS: http://hip-hops.eu/