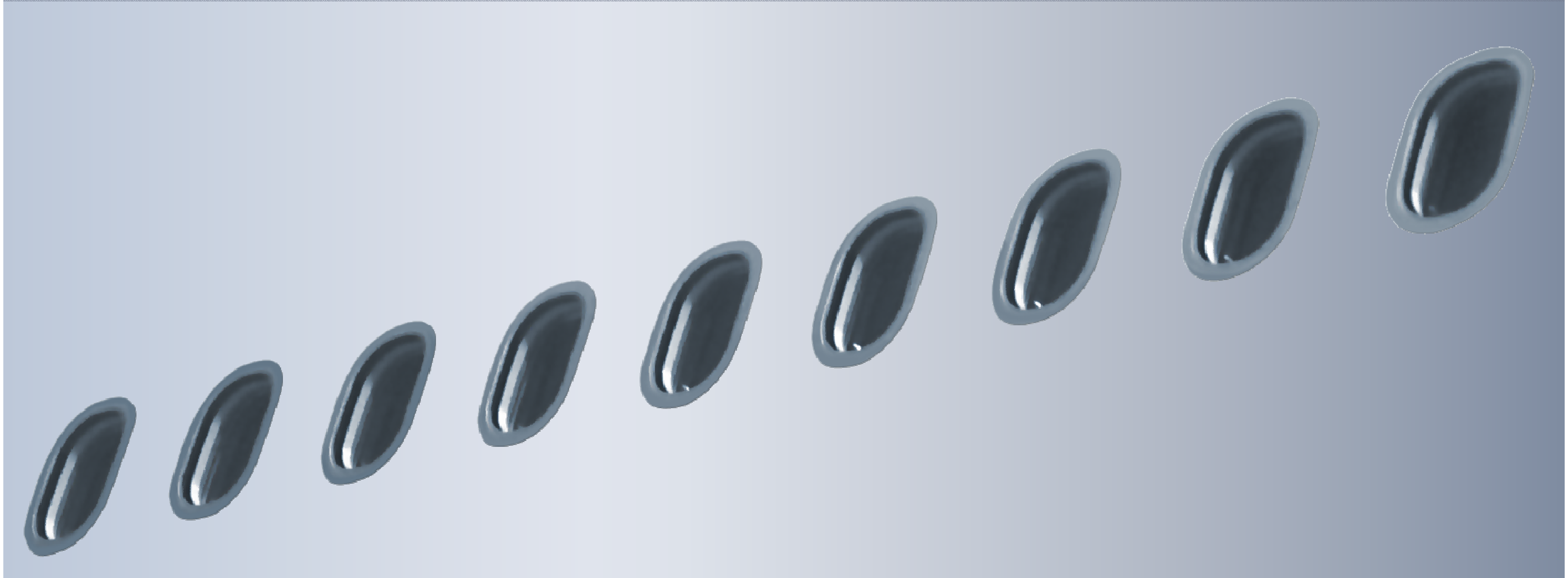


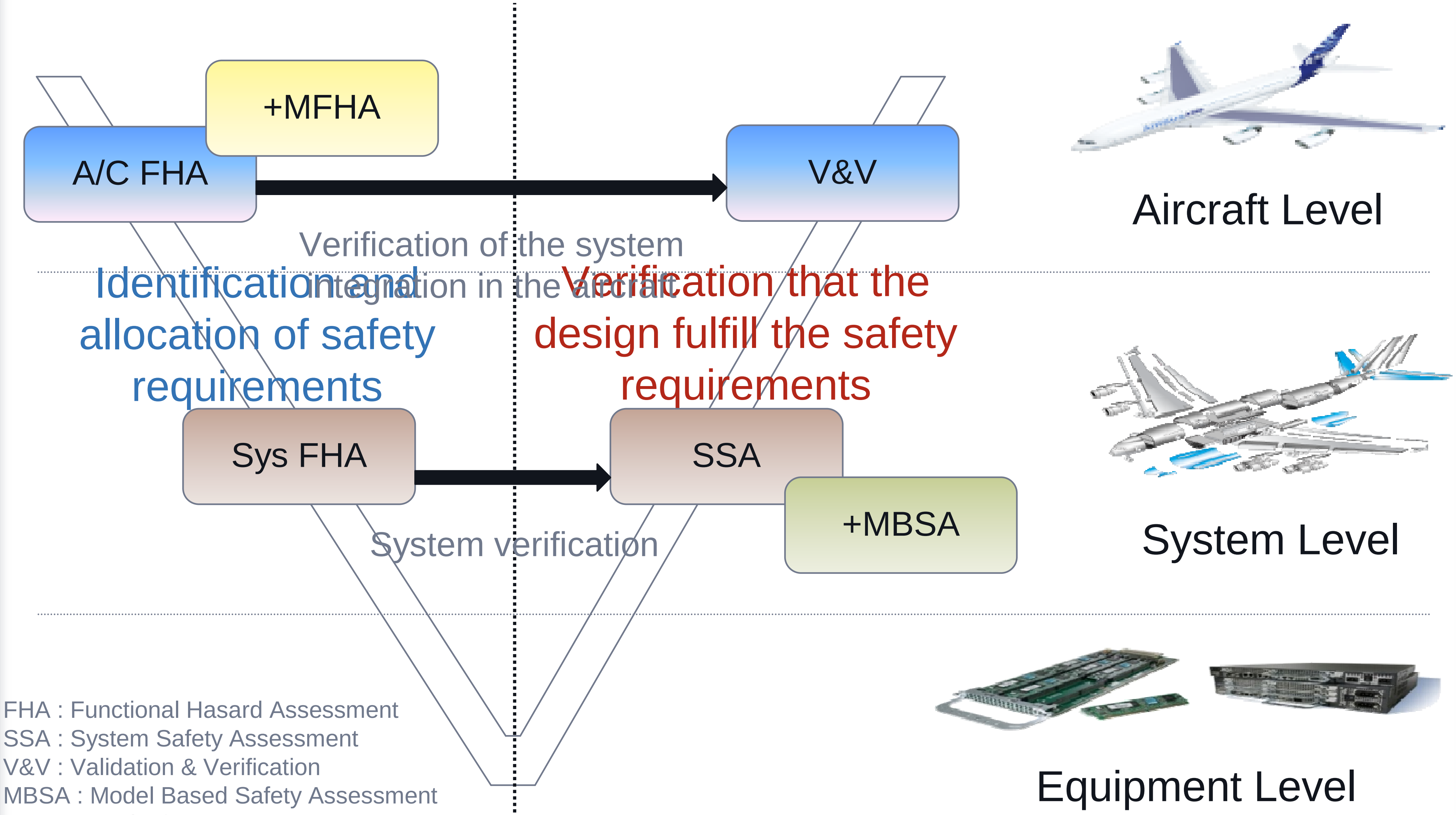
MBSAW 2012

Safety models generation to support Functional Hazard Assessment

Presented by
Sébastien Maitrehenry / PhD student Airbus/Onera/ENSMA

Plan

Context : Safety in aircraft design



FHA : Functional Hasard Assessment
SSA : System Safety Assessment
V&V : Validation & Verification
MBSA : Model Based Safety Assessment
MFHA : Model for FHA

AltaRica models to support the A/C FHA

Modeling of the functional failures propagation:

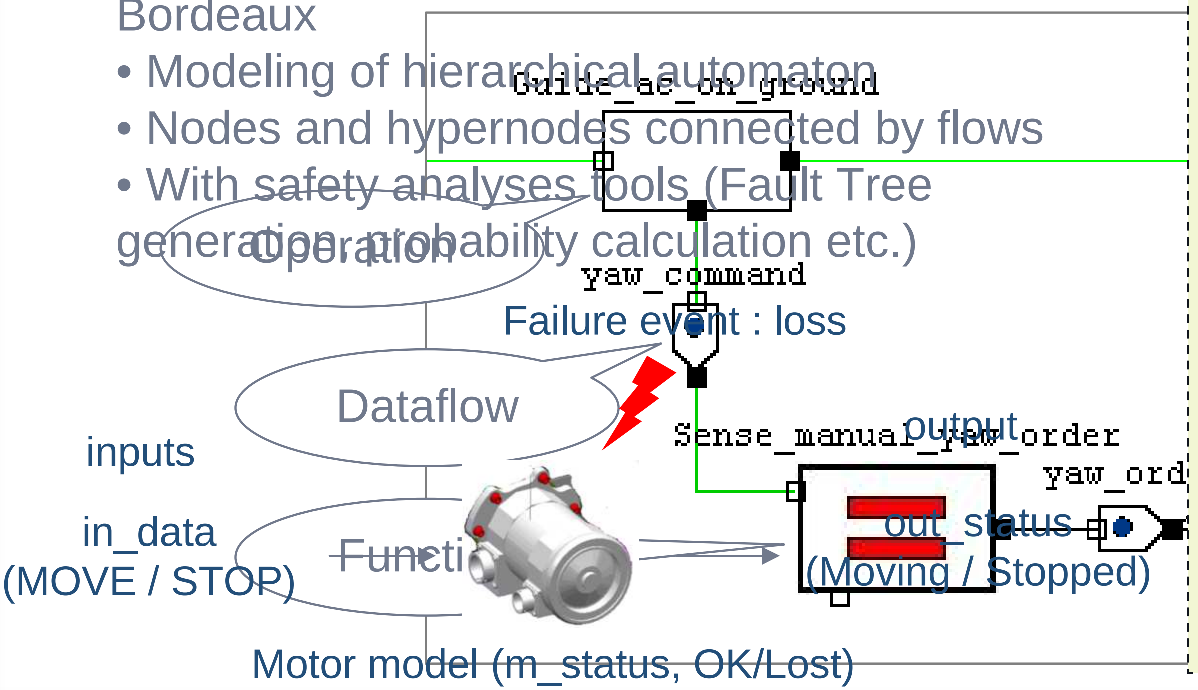
- Assessment of the contribution of a function to the flight (functional effectiveness)
- Modeling of the functional failures (total loss, partial loss, erroneous).
- Distinction of various functional data (needed, useful, etc.).

With the AltaRica language [Arnold & al. 2000]

- Formal modeling language developed at Bordeaux
- Modeling of hierarchical automaton
- Nodes and hypernodes connected by flows
- With safety analyses tools (Fault Tree generation, probability calculation etc.)

Example: AltaRica model of Guidan…

(with AltaRica equiped with a graphical t…



Node Motor_model
state
m_status : {OK, Lost};

flow
in_data : {MOVE, STOP};
out_status : {Moving, Stopped};

event
loss;

trans
m_status = OK |- loss -> m_status := Lost;

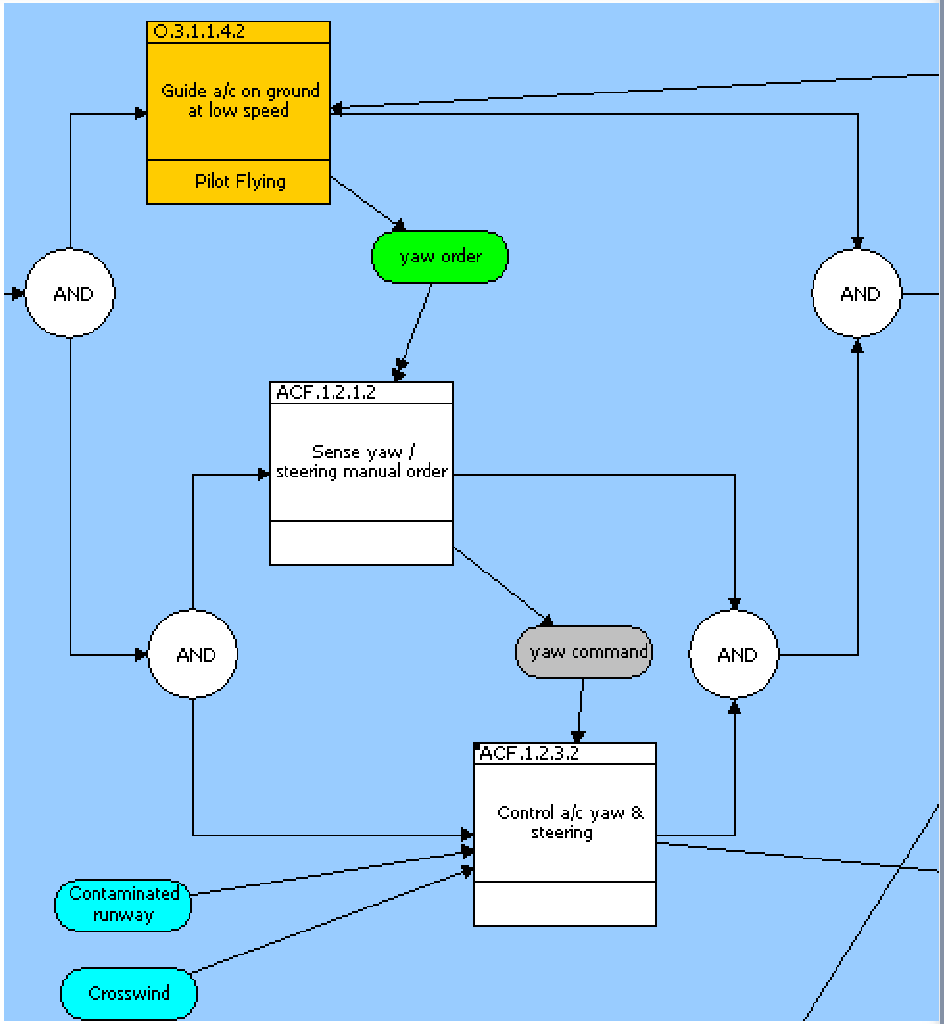
assert
out_status = case{in_data = MOVE
and m_status = OK : Moving,
else Stopped};
Edon

ALFA : Operational and functional models

ALFA (Aircraft Level Functional Approach)

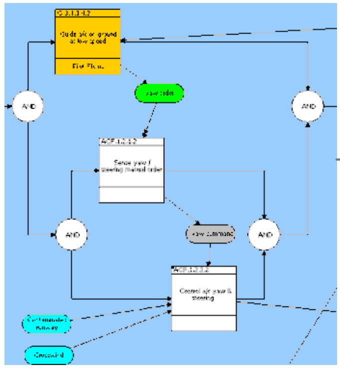
- Reference models to describe the fulfilment of flight scenarios in terms of:
  - human actions (**operations**)
  - aircraft actions (**functions**).

- Tool suported: CORE (Vitech Corporation) with the formalism EFFBD (Enhanced Functional Flow Bloc Diagram). [Seidner 2009]



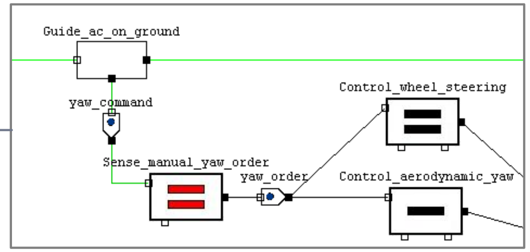
Designers are not safety specialists & CORE is not adapted for safety analyses

Bridge between design and safety



CORE model

AltaRica model



1. Manipulation of heterogeneous CORE models and heterogeneous AltaRica models → Meta-modeling
2. Need for an automatic generation of the AltaRica models from the CORE models, with traceability. → Model transformation
3. Taking into account the specificities of each domain, design and safety (complete separation of these two domains) → With annotations

Plan

Requirements for the model transformation

1. Independent of any modeling languages

- For the safety models
- For the design operational and functional models

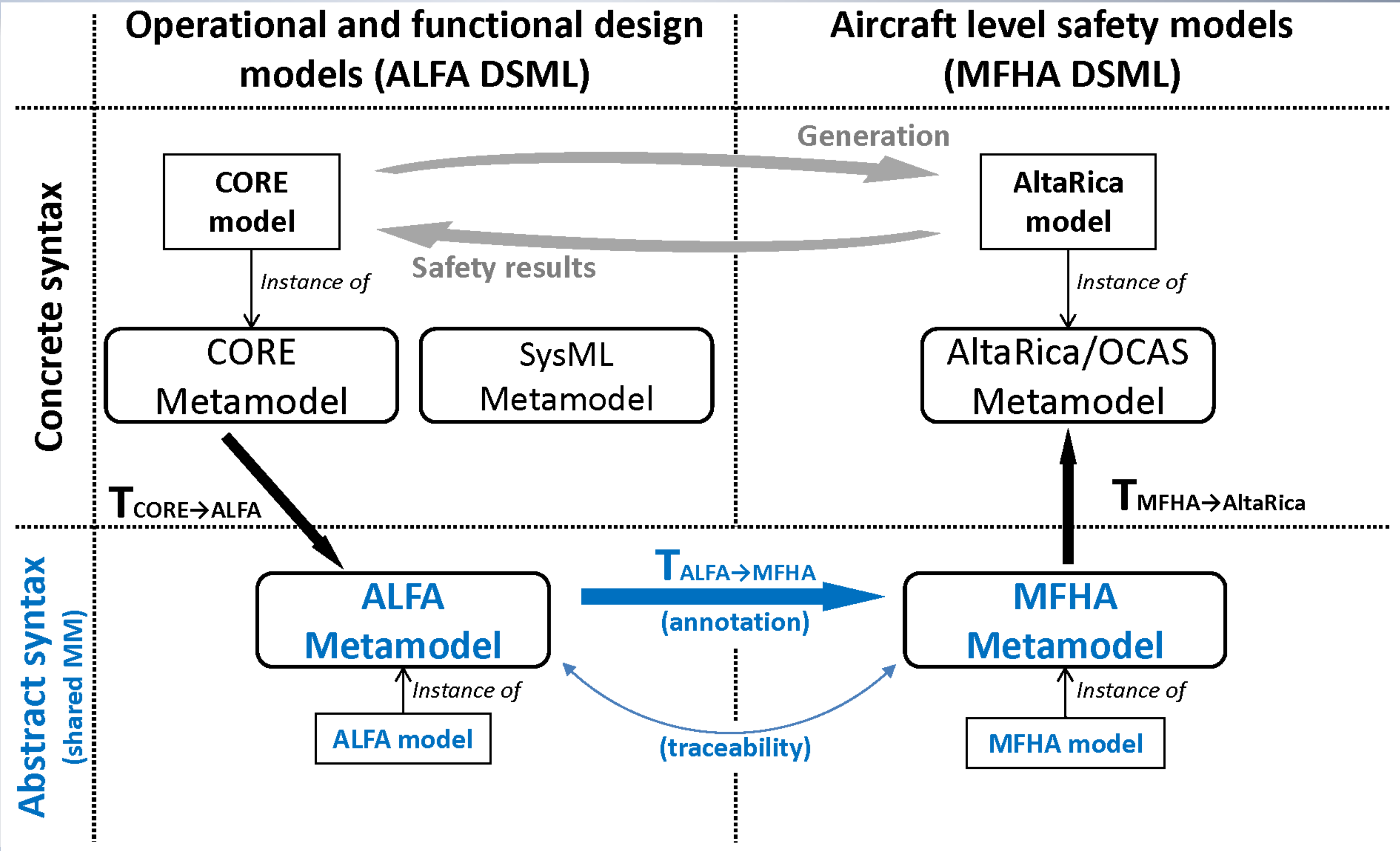
2. Preservation of the existing design and safety processes

3. Open infrastructure

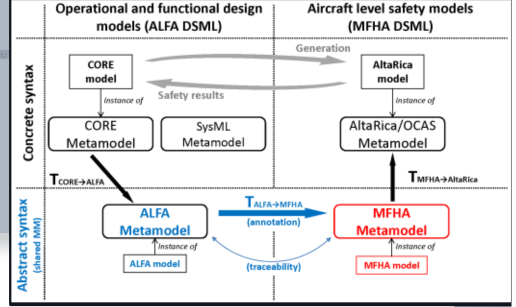
4. Tracing safety results

5. Handling annotations

Our proposal



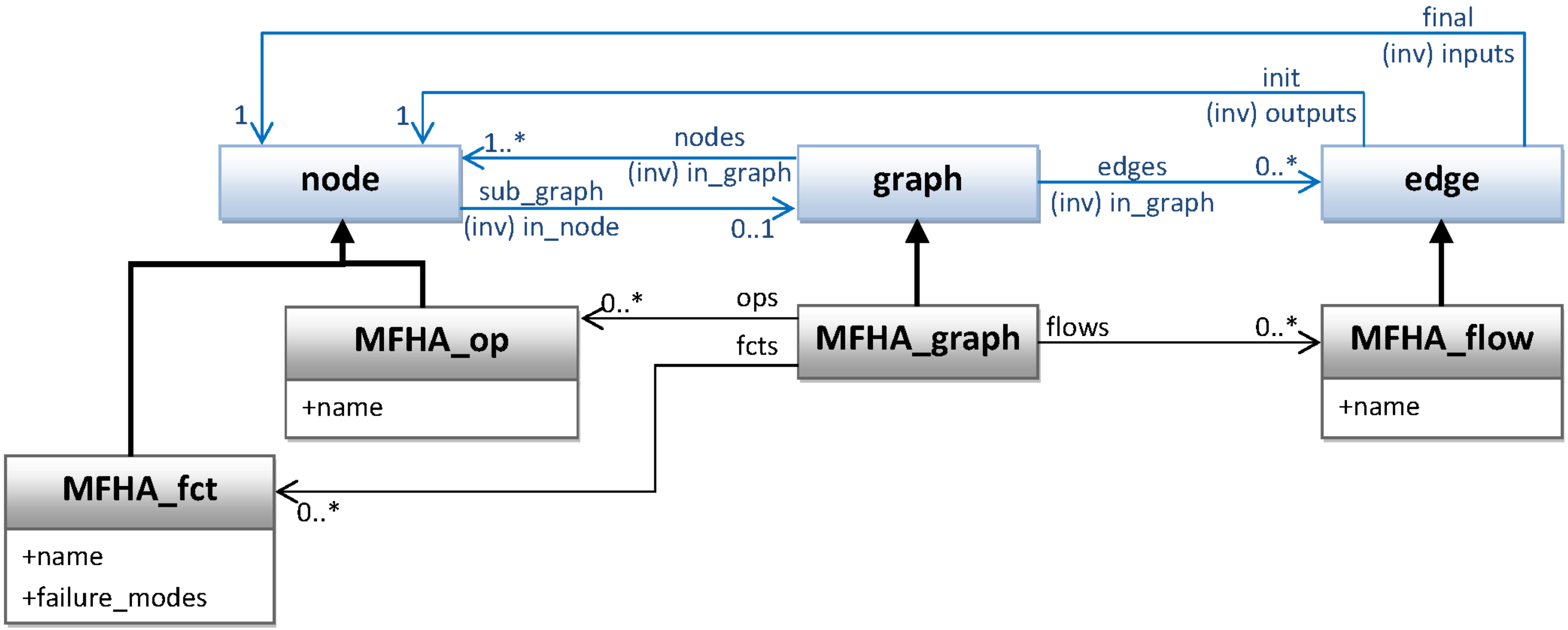
Plan

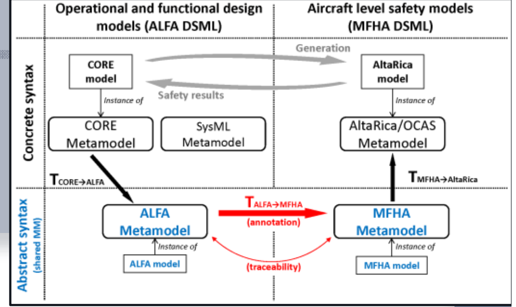


ALFA and MFHA metamodels

ALFA and MFHA models are hierarchical graphs

- Use of classical modeling techniques
- Definition of a generic model of graphs (with constraint rules)
- Specialization to ALFA and MFHA models by addition of DSML specific properties





Transformation

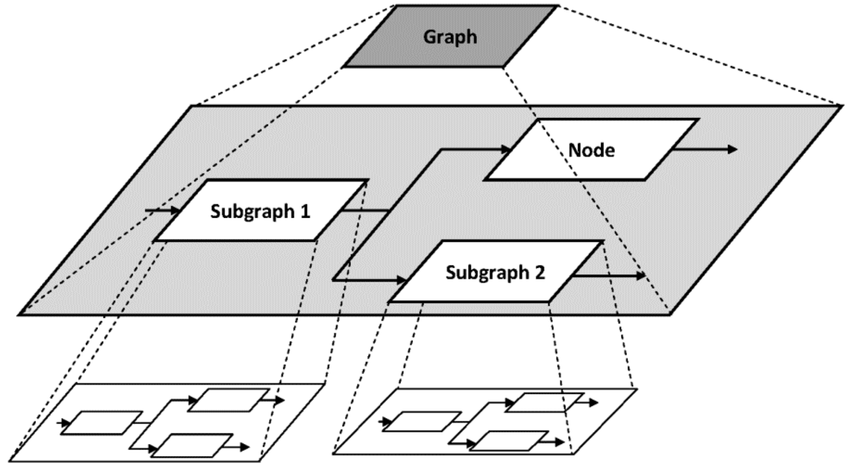
➔ Modeling language: EXPRESS

A data modeling language, with constraints declaration and procedural specifying.

➔Top down hierarchical transformation:

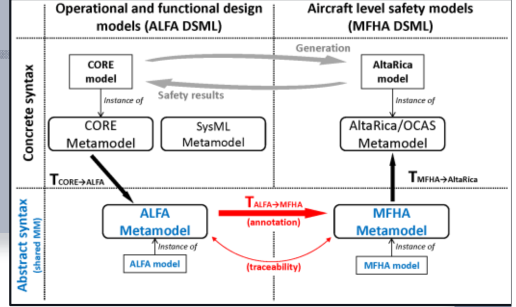
Graph, level N, is transformed (all contained nodes and links).

All subgraphs, level N-1, are transformed with the same procedure (recursive transformation).



➔Each ALFA node transformed into a MFHA node.

➔ **New nodes** added (ex: model activation)

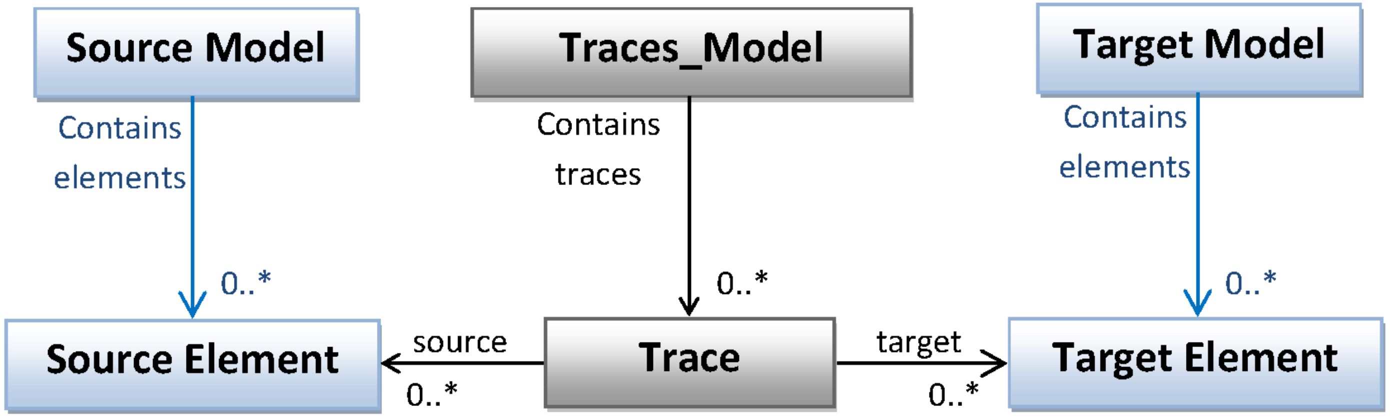


Traceability in the model transformation

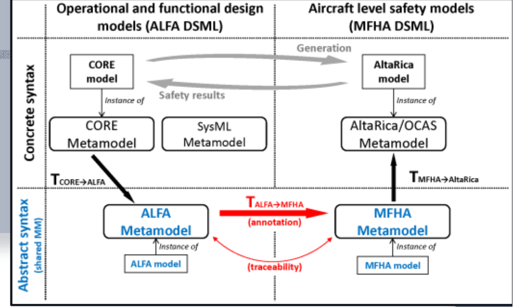
Why? → perform model management, evolution and validation (in particular for safety requirements).

What? → traceability of all elements (node, links) transformation.

How? → with a separate model of traceability



Exemple of trace:

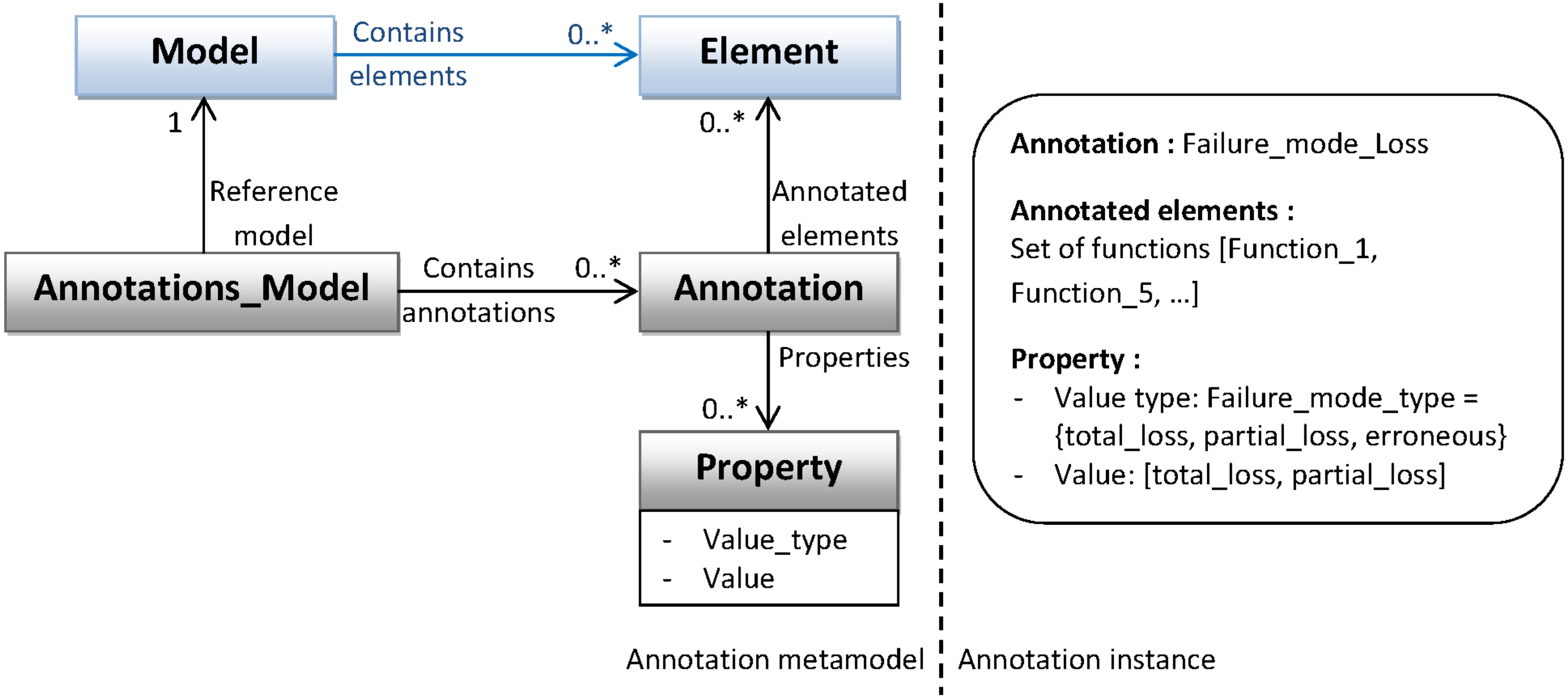


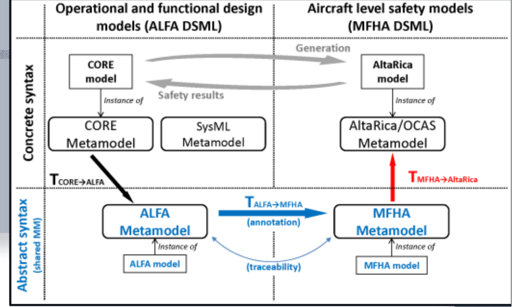
ALFA models annotation

Why? → enriching the operational and functional models with safety information

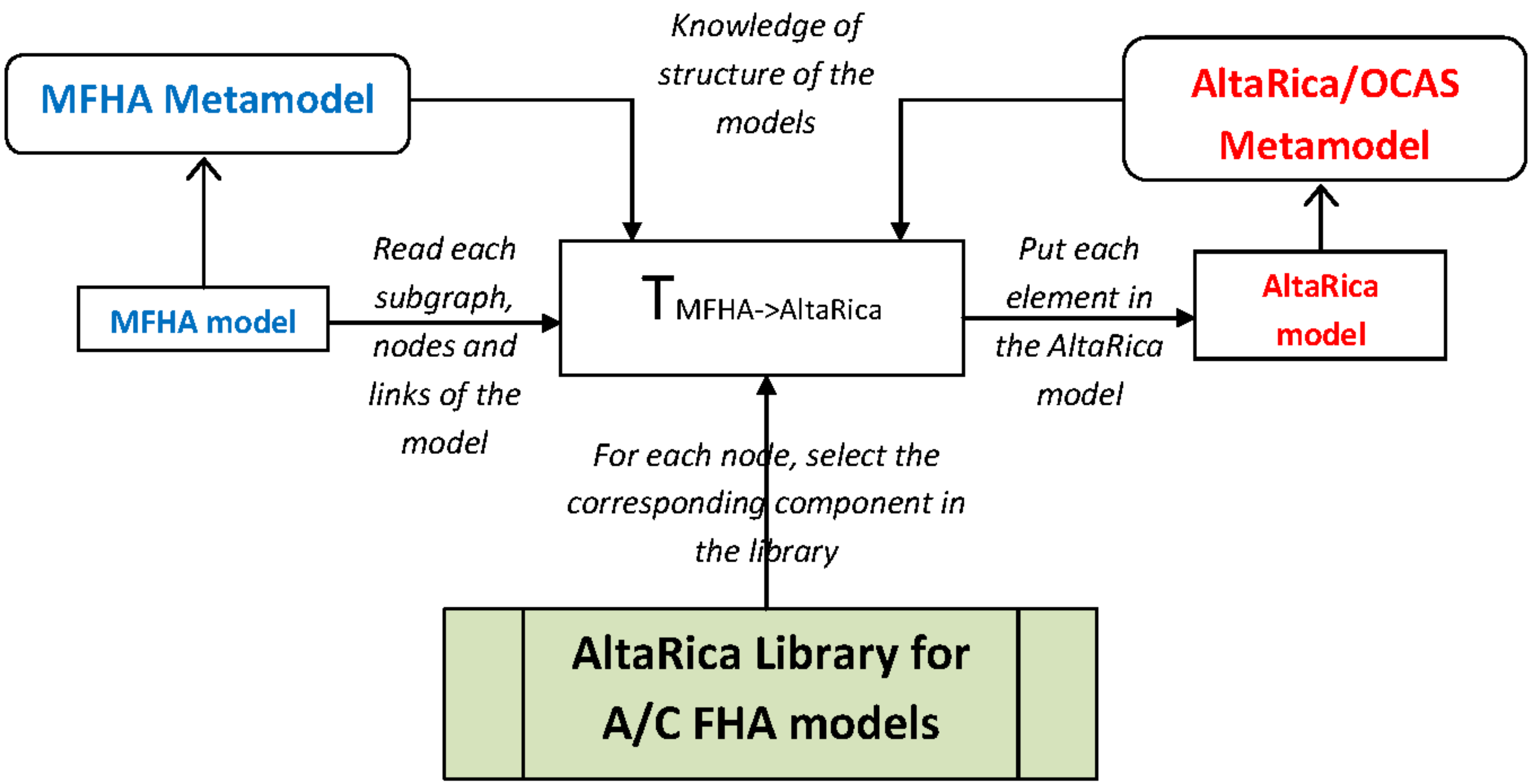
What? → safety domain knowledge: functional failure modes, functional effectiveness, etc.

How? → with a separate model of annotation (no modification on the design models)



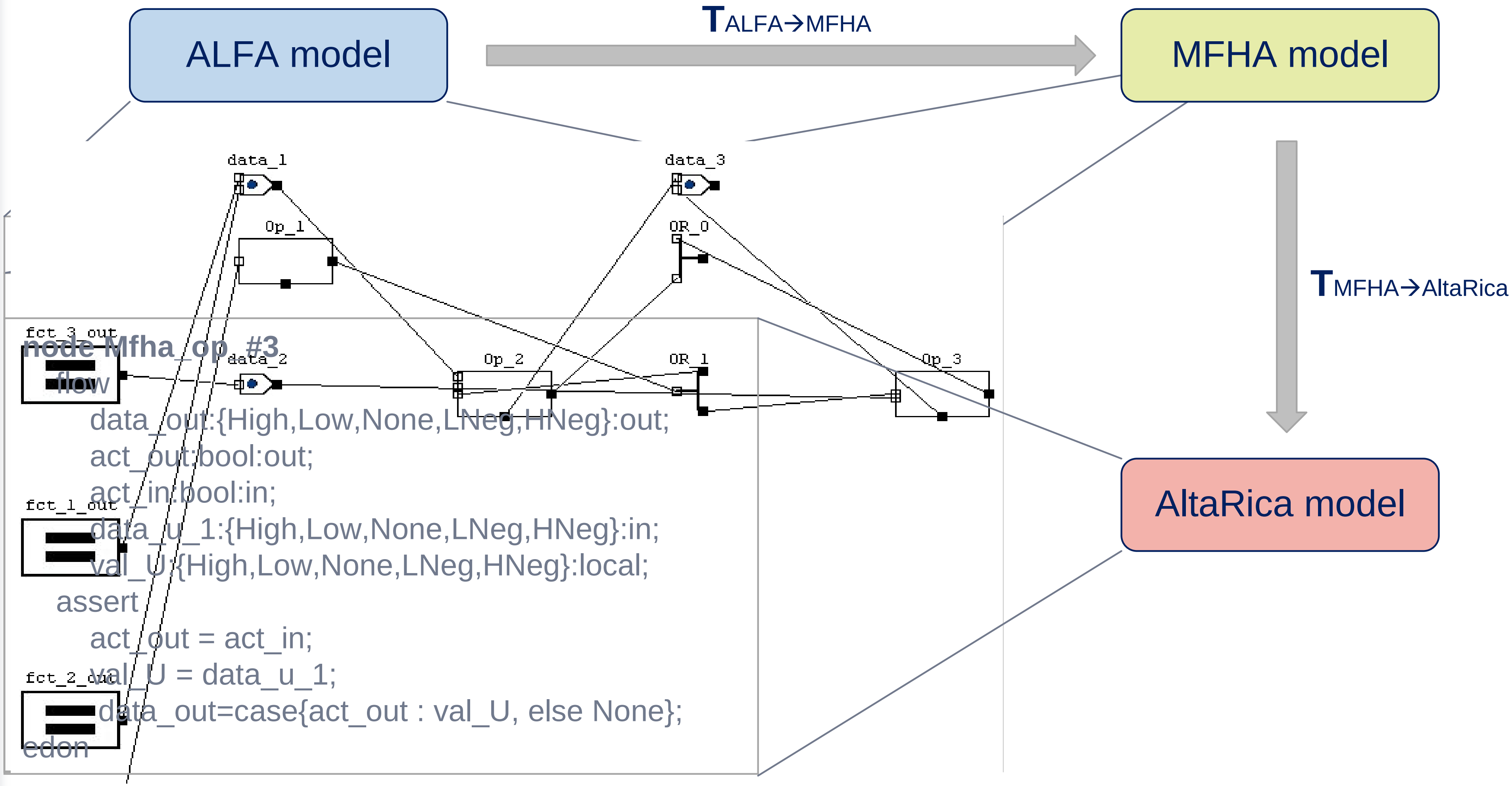


MFHA models to AltaRica models



Plan

Prototyping and experimentations



Experiments: Take-Off models, ~30 nodes, 2 levels of hierarchy

Open questions and improvements

- ➔ **Improve the global modeling process**
  - ➢ **Who should define annotated data?**
    - *Failure Modes* are defined by the safety specialists
    - *Functional Effectiveness (performance)* should be defined by the designers
  - ➢ **Are the annotation sufficient to analyse all functional dependencies?**
- ➔ **Formalize the design and safety domain knowledge**
  - ➢ **Use of Ontologies?**
- ➔ **Scalability**
  - ➢ **Composition/Decomposition in graphical views?**

Plan

Conclusion

1. Reduce the gap between the design and the safety

2. Formalised models for safety analyses

3. By a generic approach

4. Preserving existing modeling activities

5. Including traceability

6. Handling domain annotation