

DE LA RECHERCHE À L'INDUSTRIE

cea



Model-based system engineering for safety analysis of complex systems

MBSAW'12 Nataliya YAKYMETS, Hadi JABER, Agnès LANUSSE
CEA, LIST, Laboratory of Model-Driven Engineering for Embedded Systems

www.cea.fr



digiteo

list

11 Septembre 2012

❖ Classical SA techniques: Fault trees, FMEA, etc...

- Performed mostly manually
- Time consuming, costly, high probability of errors...
- Little tool support
- No strong links between system engineering and SA

❖ Achieved results in formal approaches

- Industrial toolsets (Isograph, Item, Relex, etc) : well-elaborated but costly
- Academic tools (FSAP-NUSMV, ARC/AltaRica, etc) : gap in graphical representation of SA information

❖ Goals

- To provide a support for SA engineers by integrating SA techniques into model-driven engineering
- To leverage features of UML/SysML to develop a toolset for model checking and fault tree generation

CONTENTS



Motivation



Model-Based Systems Engineering & Safety Analysis



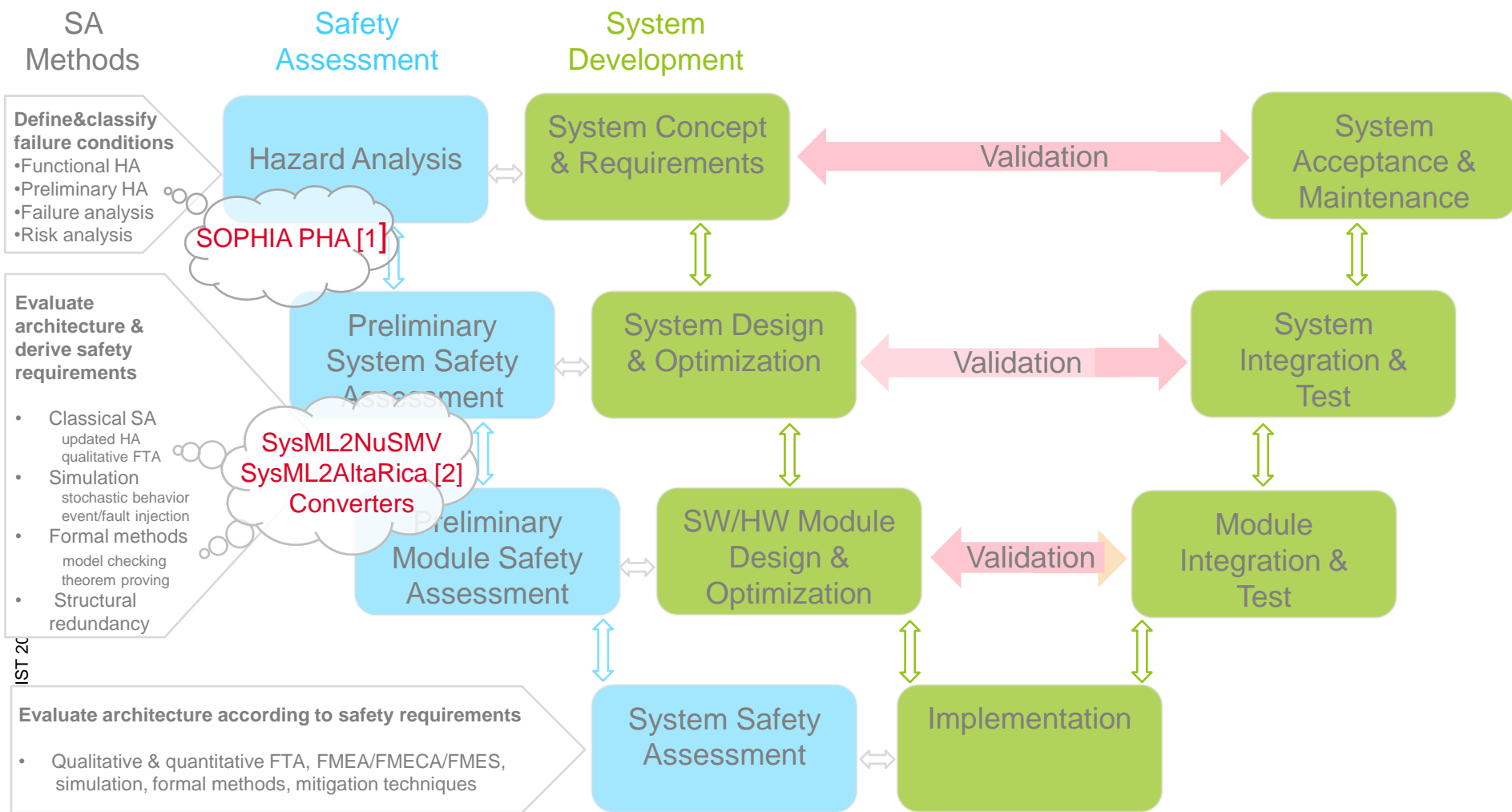
Safety Analysis Toolset



Example



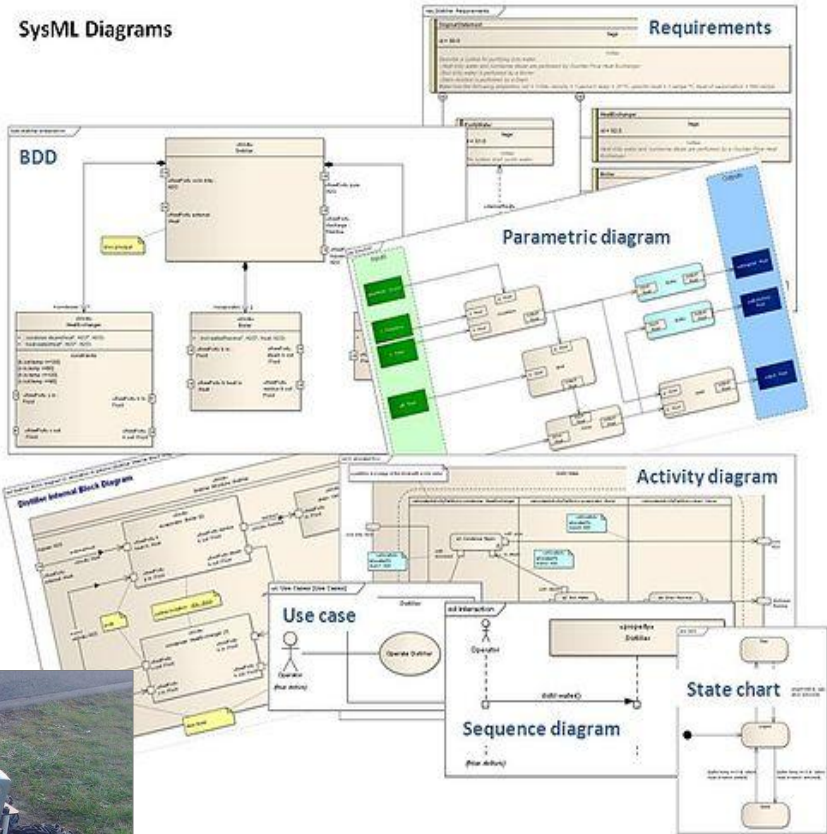
Conclusion & Further work



[1] Cancila, D.; Terrier, F.; Belmonte, F.; Dubois, H.; Espinoza, H.; Gérard, S. & Cucurru, A. SOPHIA: a Modeling Language for Model-Based Safety Engineering 2nd Int.l Workshop On Model Based Architecting And Construction Of Embedded Systems, 2009, 11-26;

[2] H. Jaber, Analyse de sûreté à partir de modèles de systèmes, M.Sc. Thesis, CEA

Engineers are finding that they can save significant time by modeling their system before building a physical prototype



© CEA LIST 2012

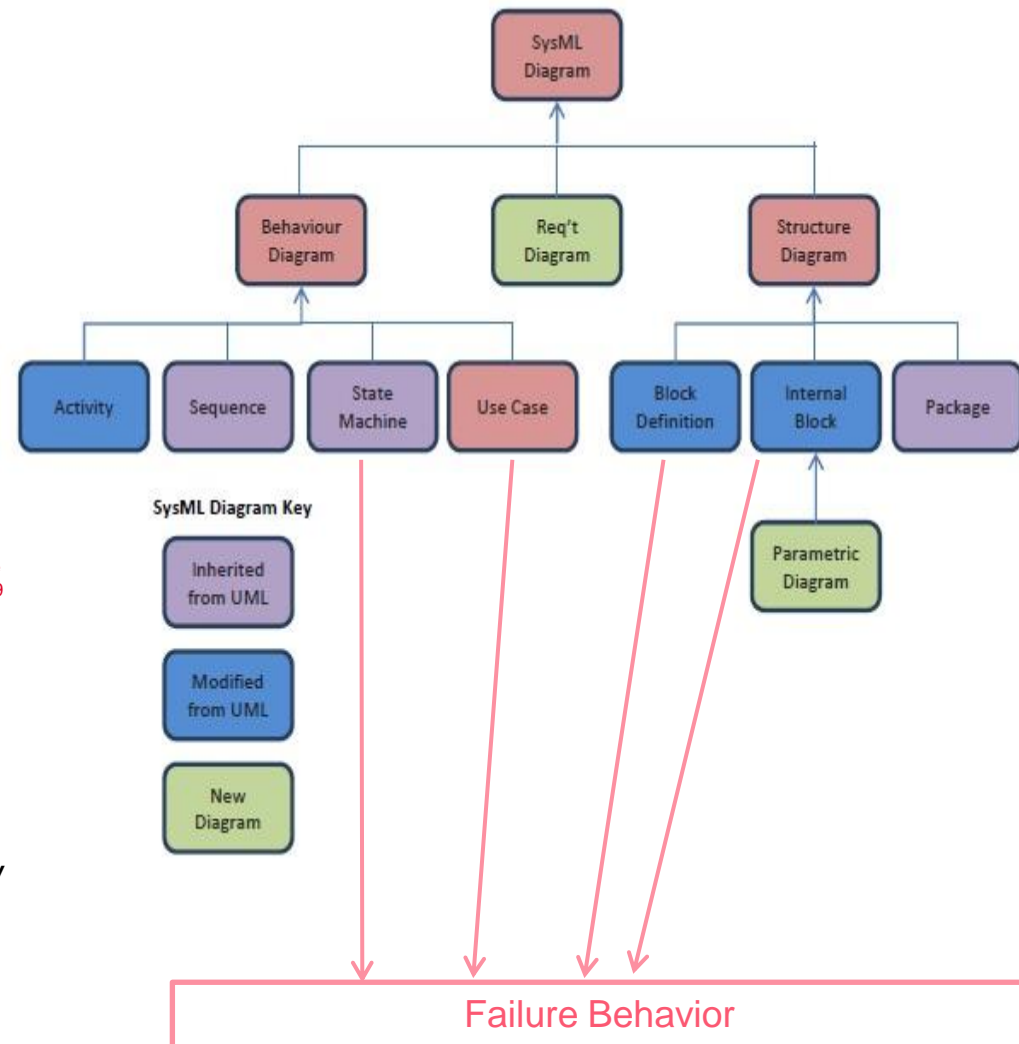
Engineers are finding that they can save significant time by modeling their system before building a physical prototype

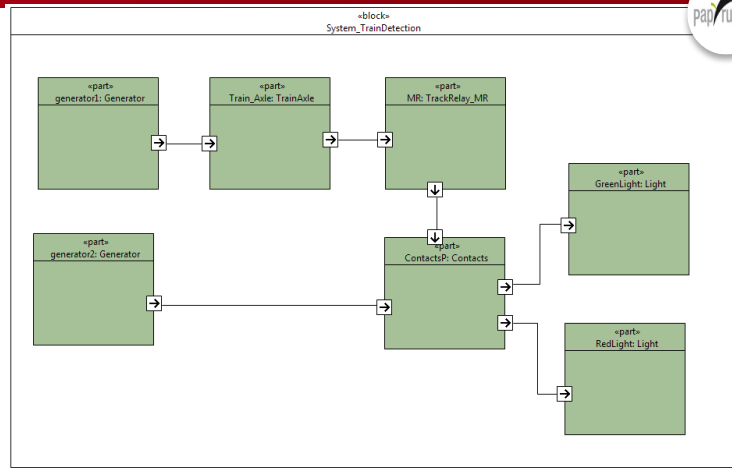
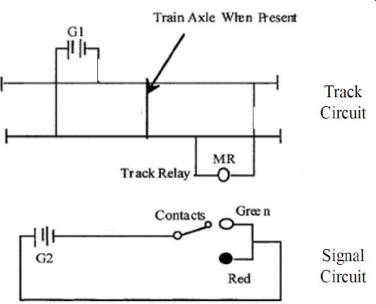
Why SysML ?

P. David, V. Idasiak & F.Kratz (2009a). Use and improvements of SysML in reliability study. Proc. of the 55th Annual Reliability and Maintainability Symposium, RAMS 2009, Fort Worth, Texas, USA, Jan. 2009

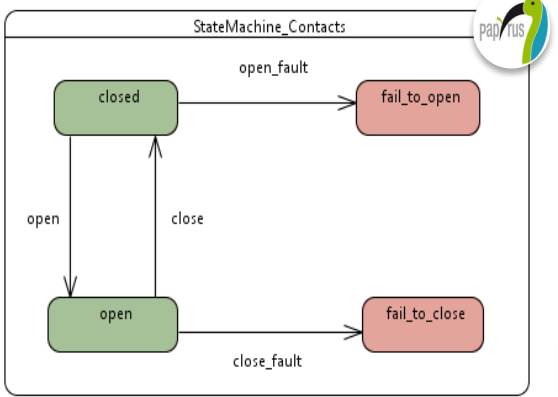
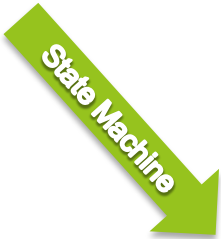
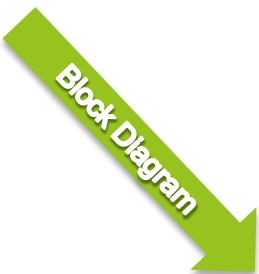
- General-purpose modeling language
- Semantics are flexible and expressive
- Global overview of architecture
- Combines HW & SW
- Integrated requirements and life cycle traceability support
- SCADE integrates system view with SysML

© CEA LIST 2012



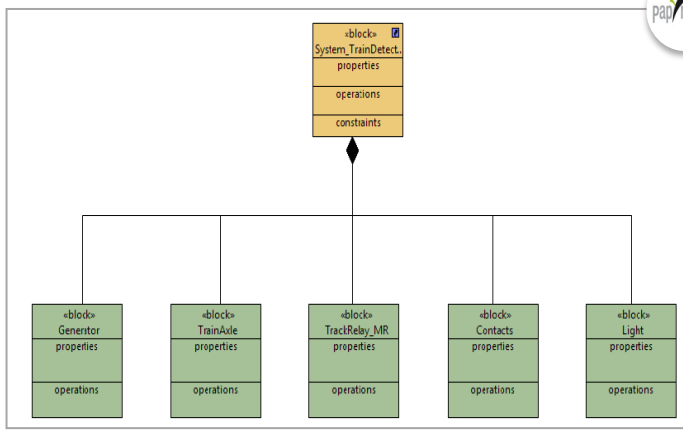


J. D. Andrews and J. J. Henry, "A computerized fault tree construction methodology," in Proc. of the Institution of Mechanical Engineers, 1997; 211(E), pp. 171-183



© CEA LIST 2012

list



Block Links & Failure Propagation



SA Tools?

Tool	NuSMV [1]	ARC, AltaRica [2]	KB3 Figaro	XFTA [3]	HIP-HOPS [4]	Safety Architect [5]	Isograph tools	Item Tools	Relex Tools
Safety Analysis									
Open source	✓	✓	✓	✓	Partners, MAENAD project	Partners, BUILD-IT-Safe project	30 days	30 days	30 days
Hazard Analysis					Failure analysis		✓	Risk analysis	Risk analysis
FT generation	✓	✓			✓	✓	✓	✓	✓
FTA & optimization			✓	✓	✓	✓	✓	✓	✓
Event ordering analysis	✓	✓					✓	✓	✓
Model checking	BDD/SAT, Plain, CTL, LTL, PSL	BDD/SAT, Plain, μ -calculus, CTL*							
FMEA/FMECA/FMEDA					FMEA	FMEA	FMEA FMECA	✓	FMEA FMECA
Common Cause Analysis			✓	✓			✓	✓	✓

© CEA LIST 2

[1] A. Cimatti et al. NuSMV2: An OpenSource Tool for Symbolic Model Checking. CAV'2002, Copenhagen, Denmark, July 27-31, 2002

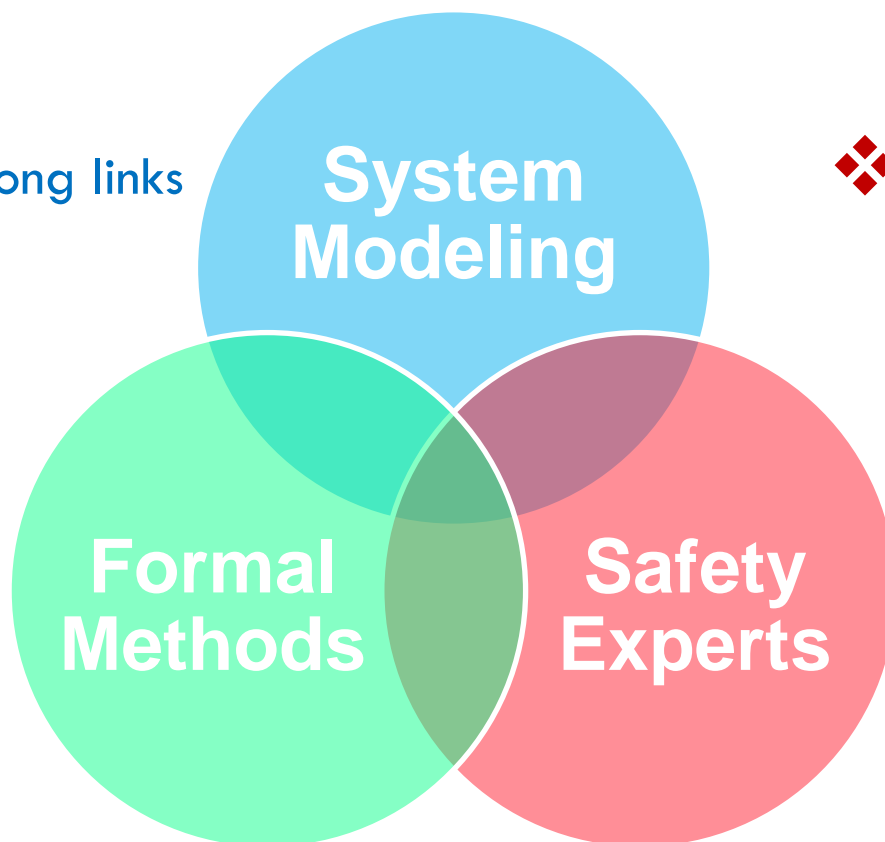
[2] A. Rauzy. Modes automata and their compilation into fault trees. Reliability Engineering and System Safety, 78 :1–12, 2002

[3] <http://www.lix.polytechnique.fr/~rauzy/xfta/xfta.htm>

[4] Y. Papadopoulos et al. Engineering Failure Analysis & Design Optimisation with HiP-HOPS, Journal of Engineering Failure Analysis, 2011

[5] F. Vallée SAFETY ARCHITECT© : un outil d'AMDE compatible avec les concepts et outils d'Ingénierie des Systèmes Complexes, MBSAW'11






❖ Establishing strong links
and collaboration
between different
communities

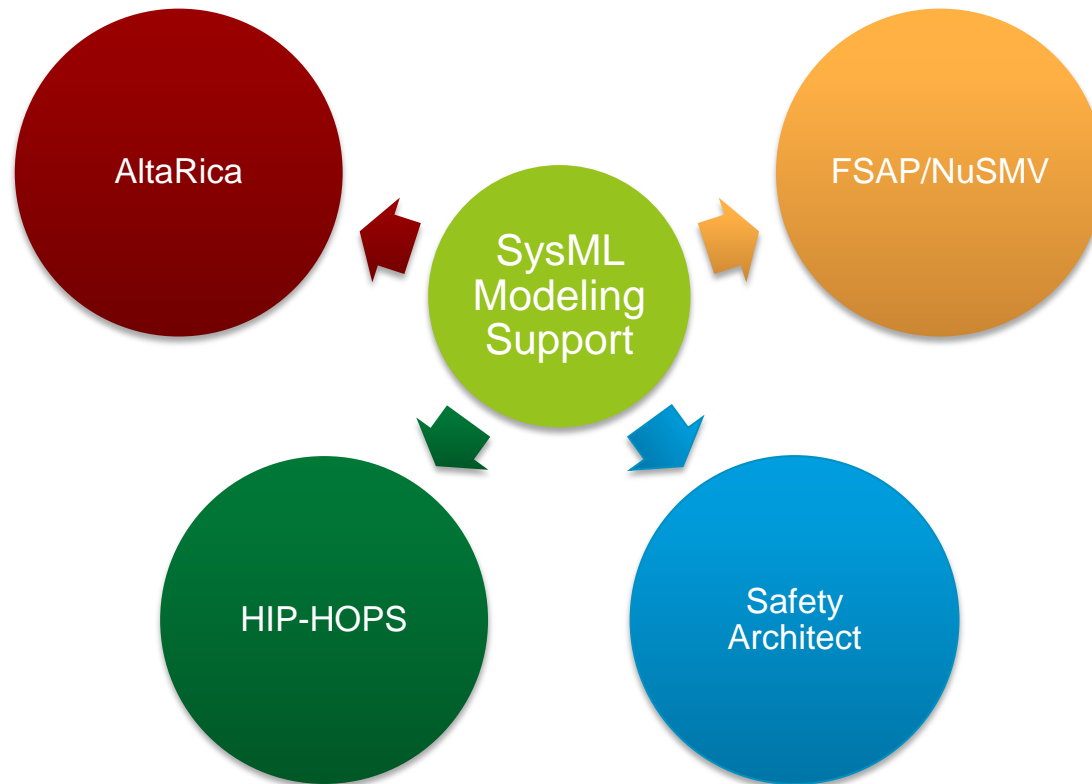


❖ Benefits

- Integrate safety analysis techniques into system modeling environment
- Make available different tools for representing both qualitative and quantitative analysis

CONTENTS

-  Motivation
-  Model Based Systems Engineering & Safety Analysis
-  **Safety Analysis Toolset**
-  Example
-  Conclusion & Further work

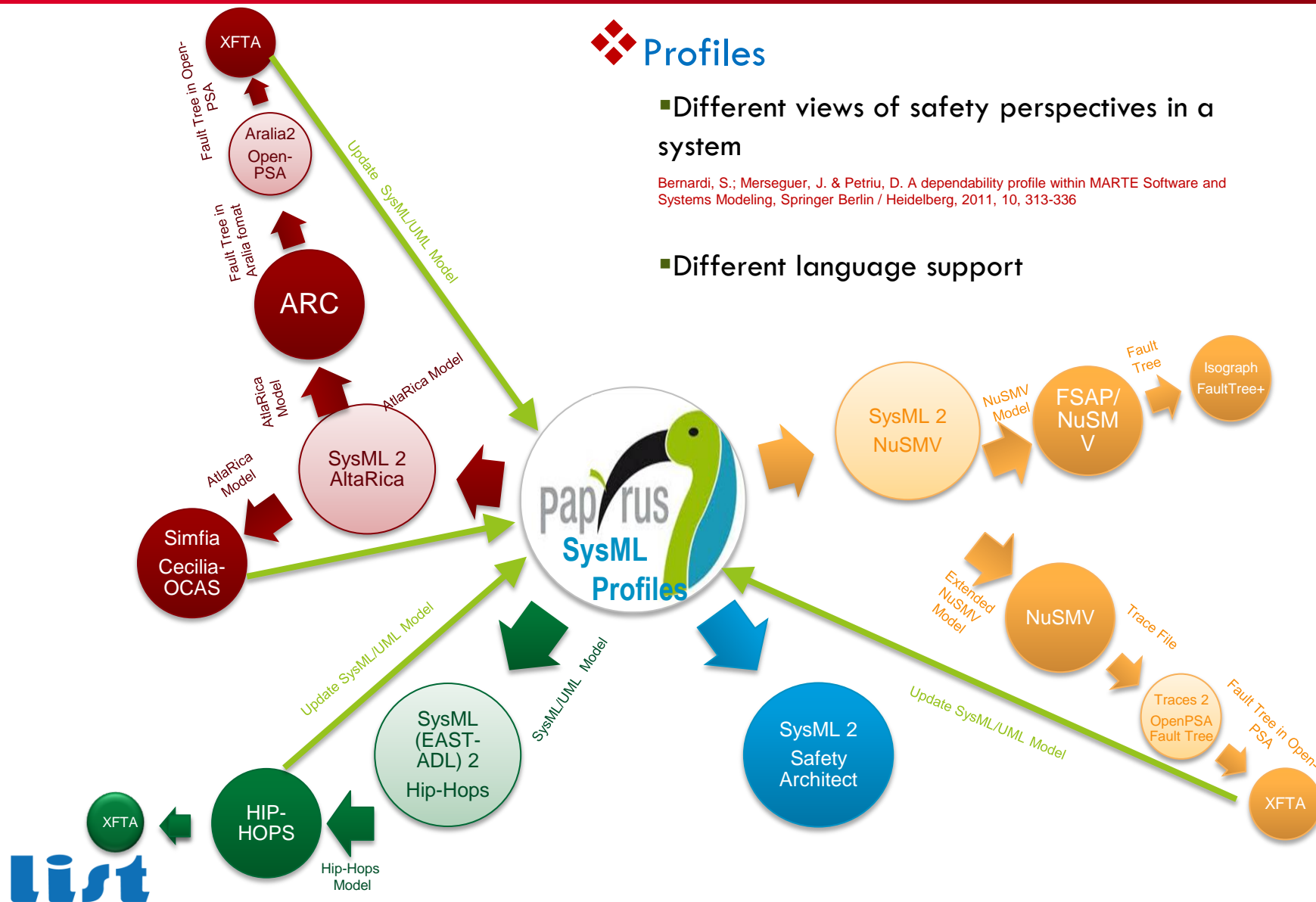


Profiles

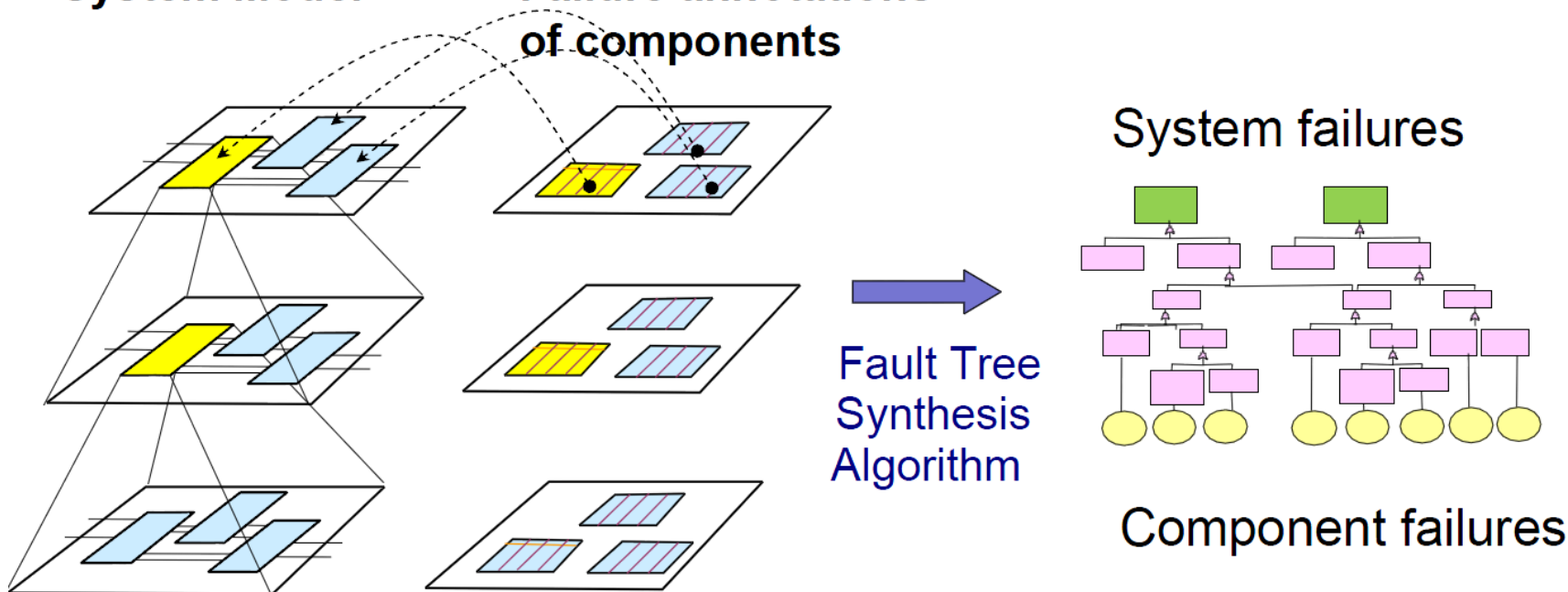
- Different views of safety perspectives in a system

Bernardi, S.; Merseguer, J. & Petriu, D. A dependability profile within MARTE Software and Systems Modeling, Springer Berlin / Heidelberg, 2011, 10, 313-336

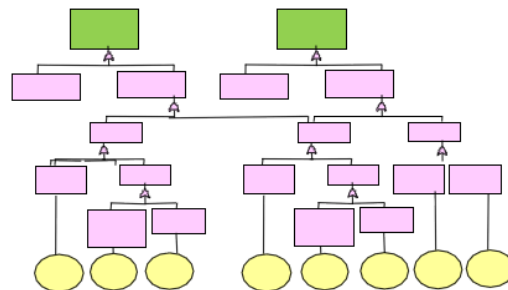
- Different language support



System Model + Failure annotations of components = Global view of failure:



System failures

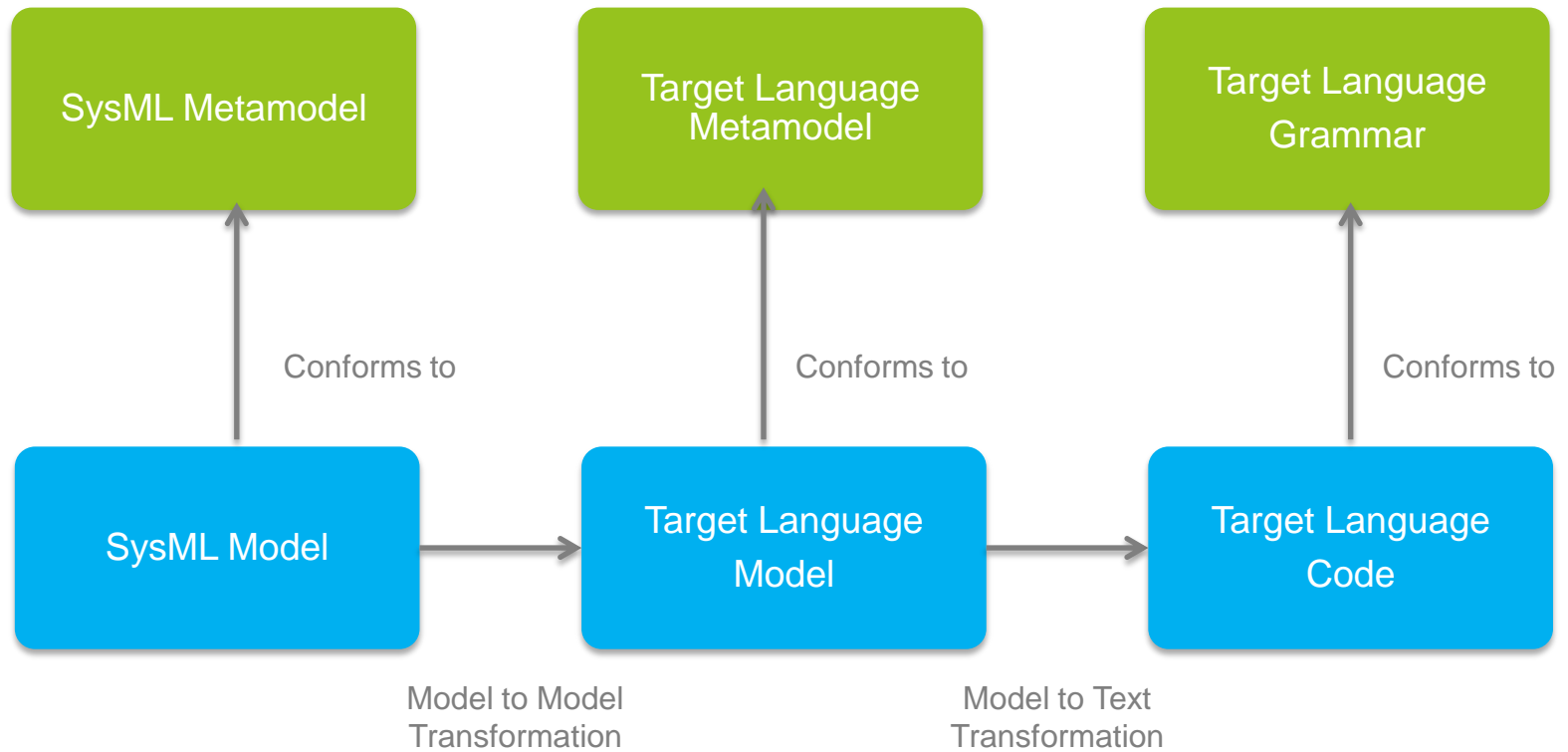


Component failures

- ❖ Profiles (EAST-ADL, etc)
- ❖ Analytical way (Safety Architect, HIP-HOPS)
- ❖ System modeling diagrams (State Machine SysML diagrams, comments, etc)

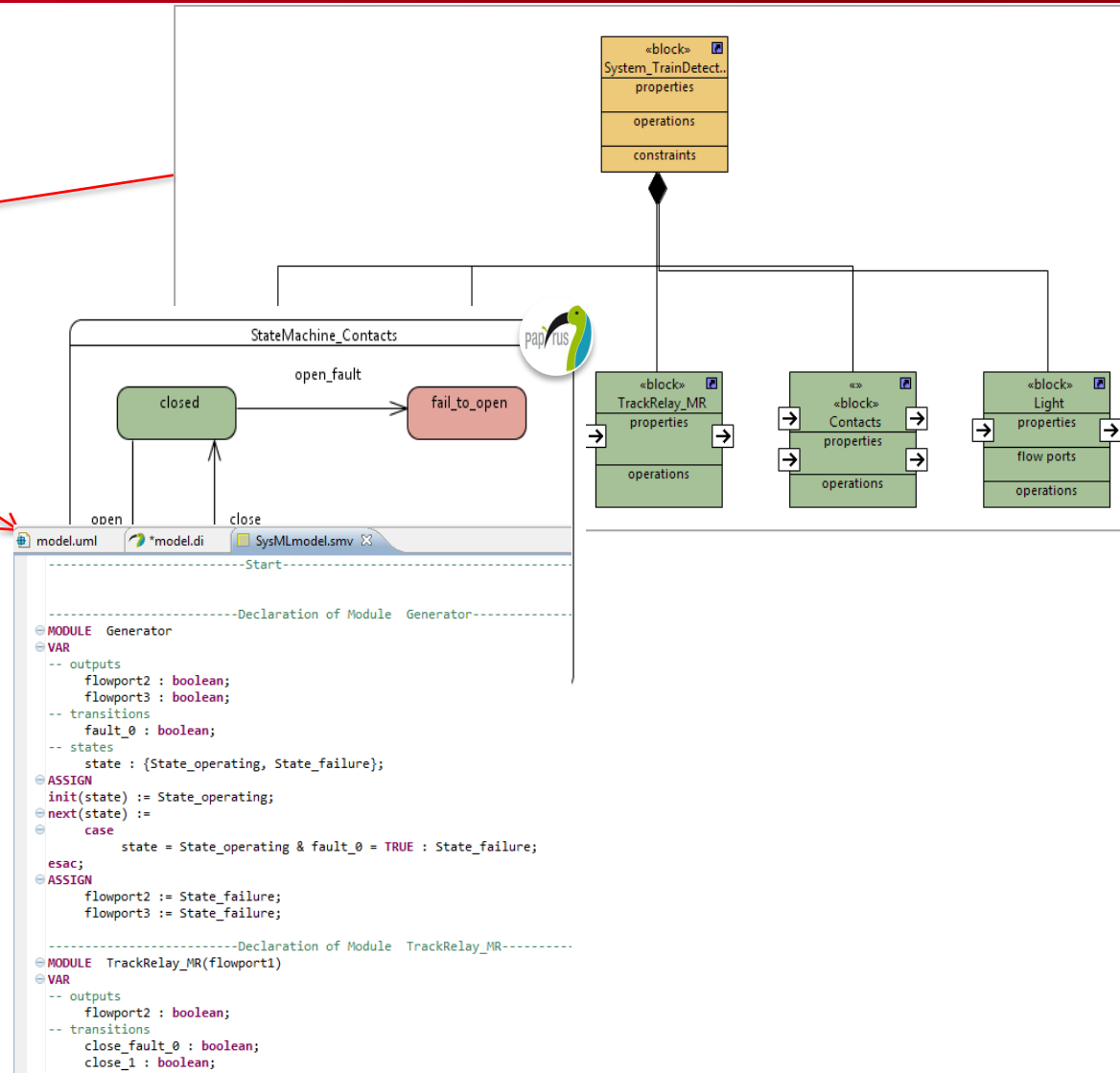
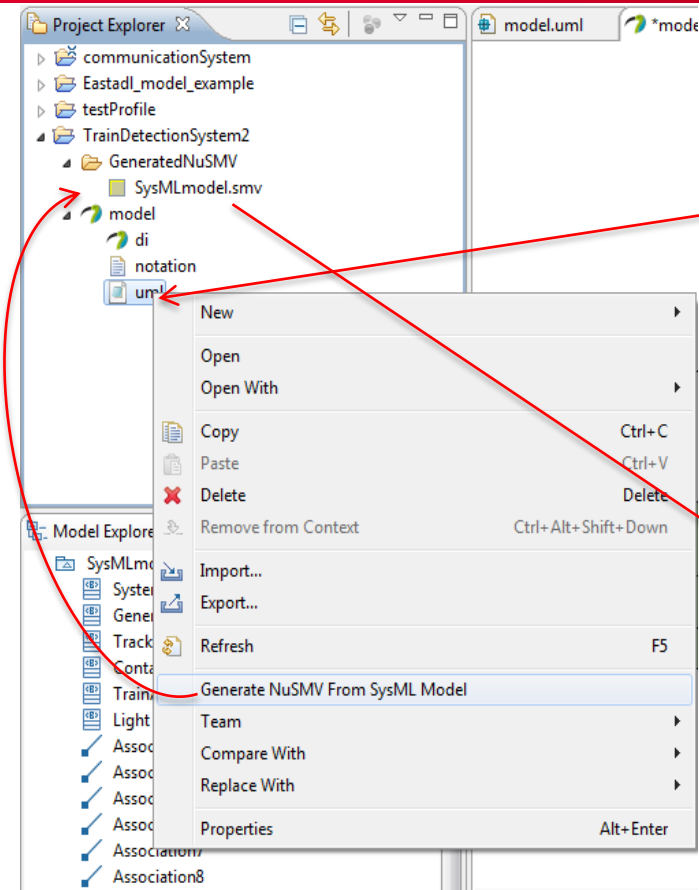


- ❖ Profiles (EAST-ADL, etc)
- ❖ Analytical way (Safety Architect, HIP-HOPS)
- ❖ System modeling diagrams (State Machine SysML diagrams, comments, etc)








The screenshot displays the Papyrus IDE interface for the SysML2AltaRica converter. The main workspace is divided into several panes:

- Project Explorer:** Shows the project structure, including 'ExempleIntroductif' and 'TrainDetectionSystem'. Annotations point to 'AltaRica généré', 'Commandes d'ARC', 'Résultat d'ARC en format texte Et en format XML (openPSA)', and 'Modèle SysML'.
- Diagram View:** Displays a SysML diagram with yellow blocks and their interconnections. A red arrow points to a specific output point labeled 'Sortie à analyser'.
- Code Editor (out_System.txt):** Shows the generated AltaRica code, including event declarations and state transitions. A red arrow points to this pane with the label 'Résultats d'ARC'.
- Code Editor (ExempleIntroductifModel.alt):** Shows the generated SysML block definition code for the 'Motor' block. A red arrow points to this pane with the label 'Code AltaRica généré'.
- Model Explorer:** Lists the model elements, such as '«Block» System', '«Block» Motor', and '«Block» Battery'.
- Properties View:** Shows the properties of the selected 'IBD' (Internal Block Definition) element, including its name, type, and appearance.



CONTENTS

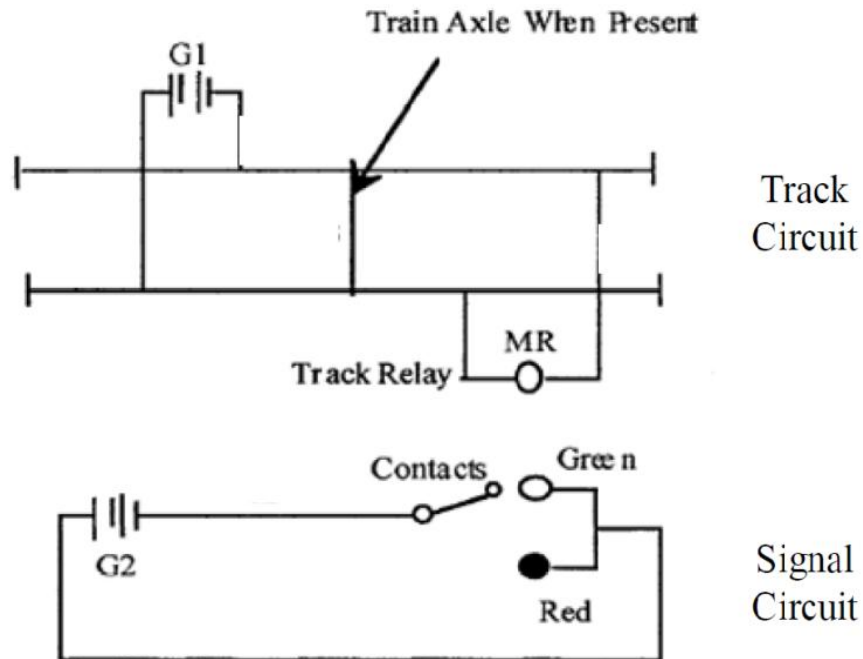
-  Motivation
-  Model Based Systems Engineering & Safety Analysis
-  Safety Analysis Toolset
-  **Example**
-  Conclusion & Further work

❖ No train

- G1 excites Relay core, which in turn attracts the Contact, so that Signal Circuit for Green light is closed (Green light is on, Red light is off)

❖ Train arrives

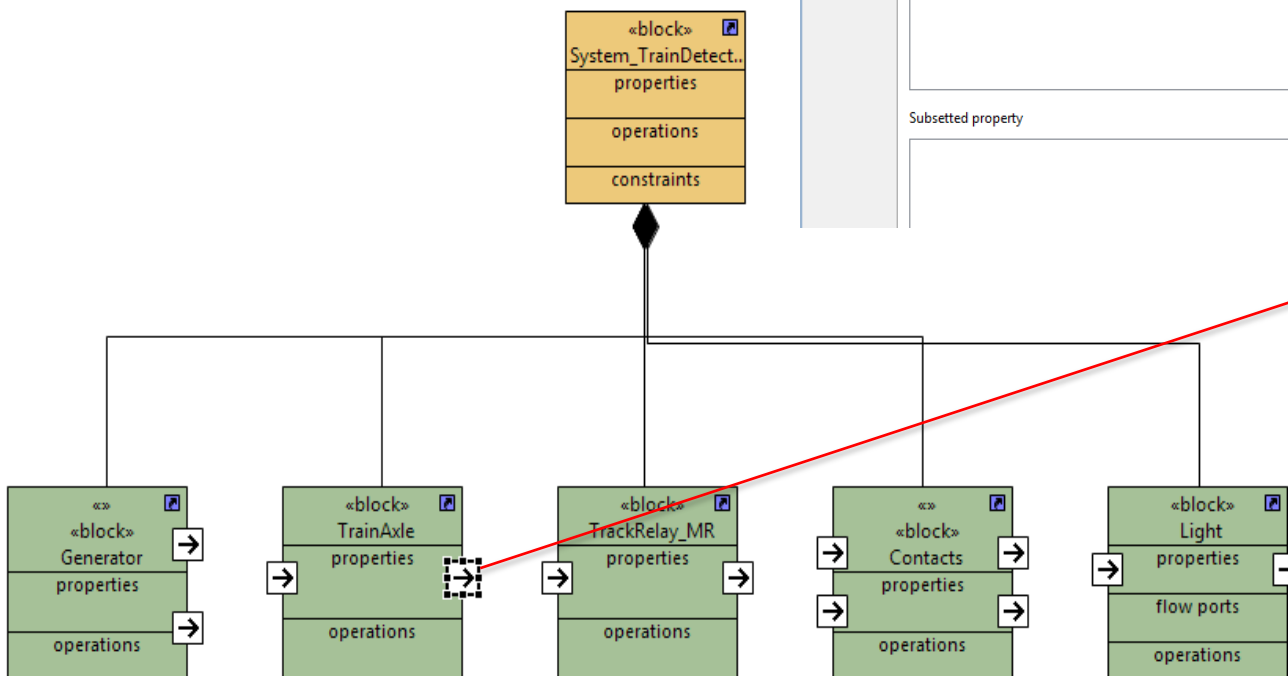
- Track Circuit is short-cut through the Train Axle. Therefore, the Relay is not excited, and Red light is on while Green light is off



J. D. Andrews and J. J. Henry, "A computerized fault tree construction methodology," in Proc. of the Institution of Mechanical Engineers, 1997; 211(E), pp. 171-183

Block annotation

Block output deviations



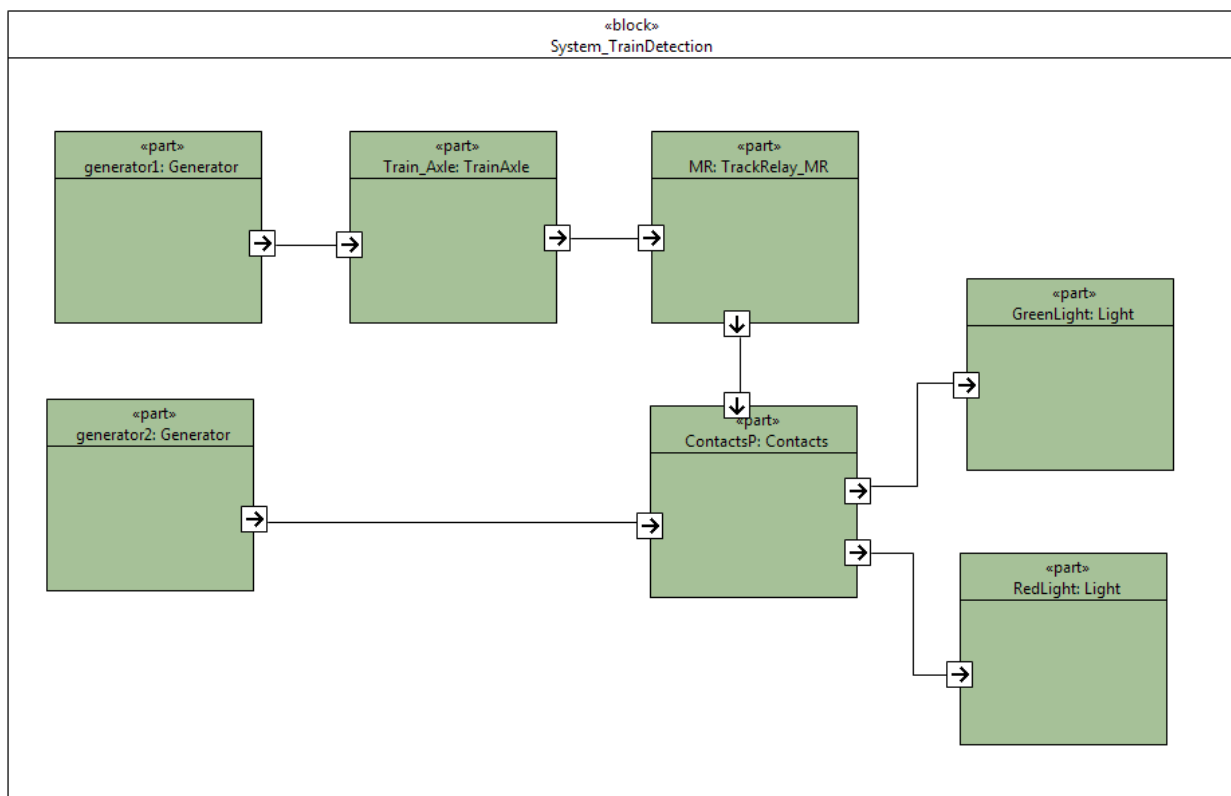
Properties window for **flowport2**:

- Name: flowport2
- UML: Name
- SysML: Is behavior: true false (Is derived)
- Profile: Is derived union: true false (Is leaf)
- Appearance: Is ordered: true false (Is service)
- Advanced: Visibility: public
- Default value: `OutputDeviation=flowport1 OR broken`
- Type: Boolean
- Provided: (empty)
- Subsetted property: (empty)

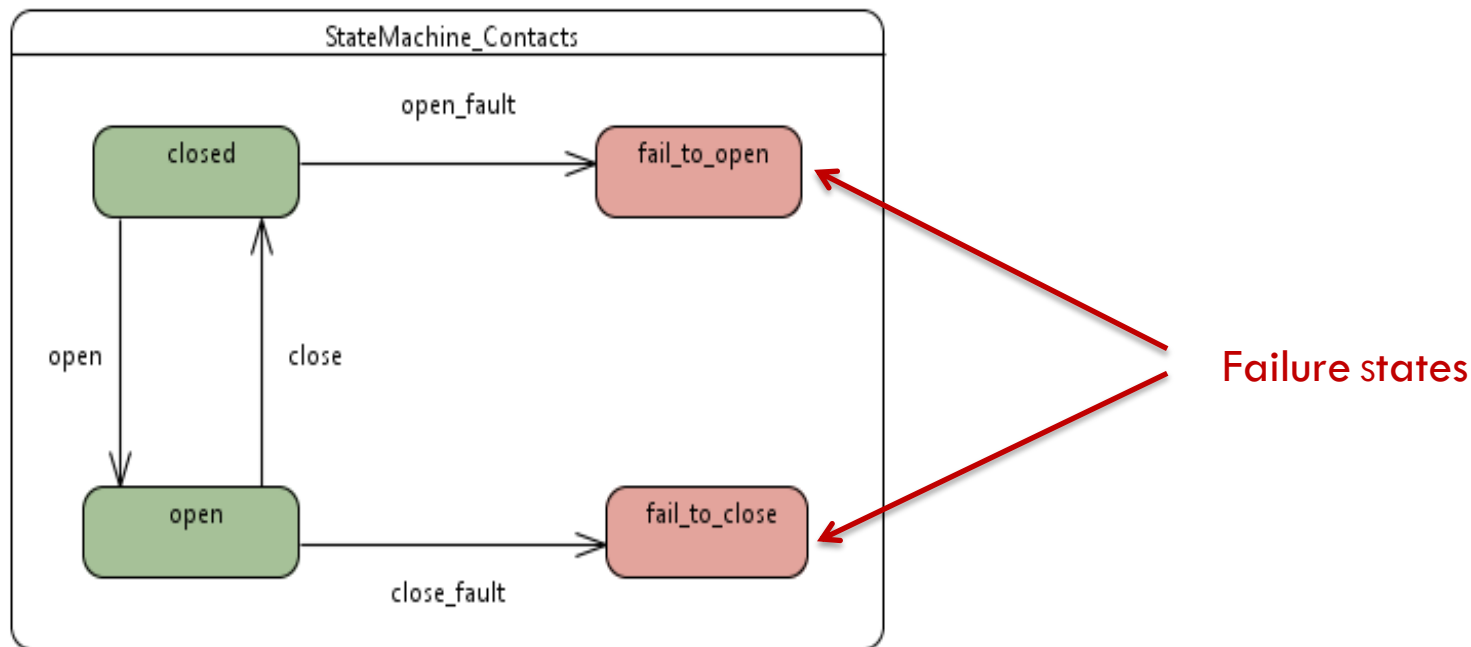
Context menu options:

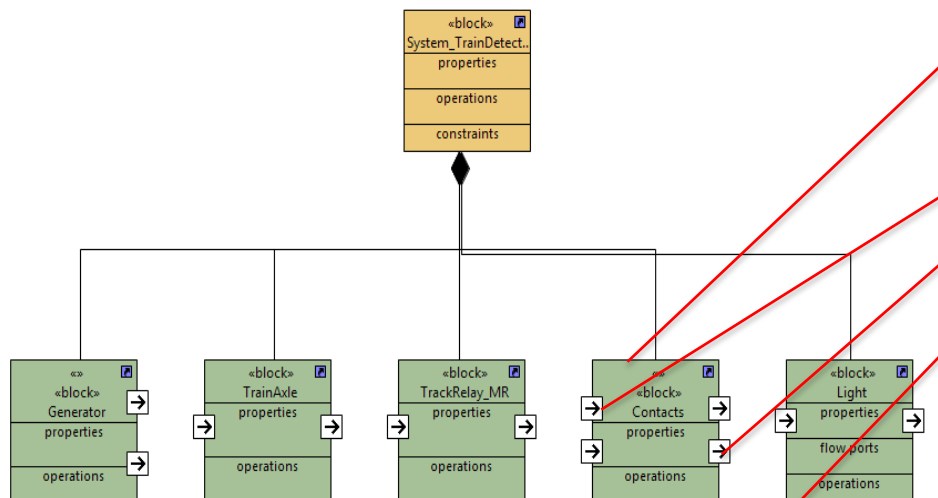
- Duration
- DurationInterval
- Expression
- InstanceValue
- Interval
- LiteralBoolean
- LiteralInteger
- LiteralNull
- LiteralString
- LiteralUnlimitedNatural
- OpaqueExpression**
- StringExpression
- TimeExpression

- ❖ Block interconnections
- ❖ Propagation of faults through the system



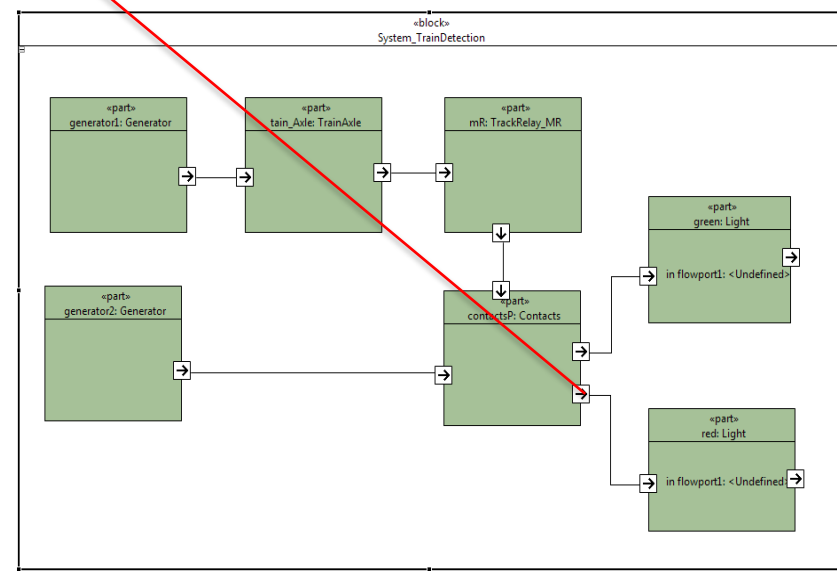
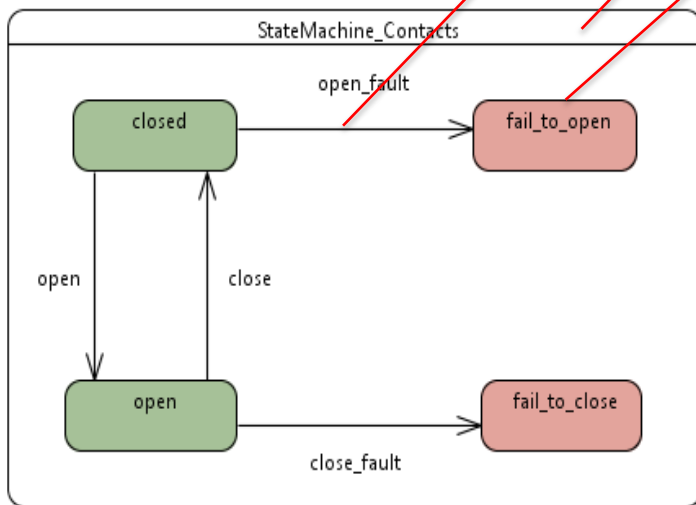
- ❖ Each block has an associated State Machine Diagram
- ❖ Annotation of the failure modes for each block





```

-----Declaration of Module Contacts-----
MODULE Contacts(flowport1, flowport2)
VAR
-- outputs
  flowport3 : boolean;
  flowport4 : boolean;
-- transitions
  open_fault_0 : boolean;
  open_1 : boolean;
  close_fault_2 : boolean;
  close_3 : boolean;
-- states
  state : {closed, fail_to_open, fail_to_close, open};
ASSIGN
  init(state) := closed;
  next(state) :=
  case
    state = closed & open_fault_0 = TRUE : fail_to_open;
    state = closed & open_1 = TRUE : open;
    state = open & close_fault_2 = TRUE : fail_to_close;
    state = open & close_3 = TRUE : closed;
  esac;
ASSIGN
  flowport3 := flowport1 | flowport2 | fail_to_open | fail_to_close;
  flowport4 := flowport1 | flowport2 | fail_to_open | fail_to_close;
  
```



© CEA LIST 2012



TrainDetectionSystem2
GeneratedNuSMV
TrainDetectionSystem2 Model.smv
TrainDetectionSystem

Model Explorer
No Model Available

Outline
SMV Model

- Contacts
- PowerSupply
- Control
- Transmitter
- Pedestal
- main

```
-----Start-----
-----Declaration of Module Contacts-----
MODULE Contacts(in2_rcv, in1_rcv)
...

```

Property	Value
BDD Stats	
Arc violation threshold	0
Cache hit threshold for resizing	30%
Dead nodes counted in triggering reordering	no
Default BDD reordering method	4
Default ZDD reordering method	4
Dynamic reordering of BDDs enabled	no
Dynamic reordering of ZDDs enabled	no
GA population size	0
Garbage collection enabled	yes
Garbage collections so far	0
Group checking criterion	7
Hard limit for cache size	1398101
Limit for fast unique table growth	838860
Maximum growth while sifting a variable	1,2
Maximum number of variables sifted per reordering	1000
Maximum number of variable swaps per reordering	2000000
Memory in use	4737740
Next reordering threshold	4004
Number of BDD and ADD nodes	4
Number of BDD variables	0
Number of buckets in unique table	256
Number of cache collisions	0
Number of cache deletions	0
Number of cache entries	262144
Number of cache hits	0
Number of cache insertions	0
Number of cache look-ups	0
Number of crossovers for GA	0
Number of dead BDD and ADD nodes	0
Number of dead ZDD nodes	0
Number of LIVE BDD and ADD nodes	4
Number of LIVE ZDD nodes	0
Number of ZDD nodes	0
Number of ZDD variables	0
Peak number of live nodes	4
Peak number of nodes	1022
Realignment of BDDs to ZDDs enabled	no
Realignment of ZDDs to BDDs enabled	no
Recombination threshold	0
Reorderings so far	0
Soft limit for cache size	1024
Symmetry violation threshold	0
Time for garbage collection	0,00 sec
Time for reordering	0,00 sec

*.smv

Statistics

TrainDetectionSystem2
 GeneratedNuSMV
 TrainDetectionSystem2 Model.smv
 TrainDetectionSystem

Start
 Declara
 MODULE Contacts(in2_rcv, in1_rcv
 Properties Problems Error Log
 Property
 BDD Stats

```
NuSMU > check_ctlspec
-- specification AG rcv.state = stuck_at_0 is false
-- as demonstrated by the following execution sequence
Trace Description: CTL Counterexample
Trace Type: Counterexample
-> State: 1.1 <-
control.out_control = TRUE
control.fault_0 = FALSE
control.state = operating
ps.out_ps = TRUE
```

```
***** Simulation Starting From State 3.1 *****
NuSMU > show_traces -v
<!-- ##### Trace number: 3 ##### -->
Trace Description: Simulation Trace
Trace Type: Simulation
-> State: 3.1 <-
control.out_control = TRUE
control.fault_0 = TRUE
control.state = operating
ps.out_ps = TRUE
ps.fault_0 = FALSE
ps.state = operating
pedestal.out_pedestal = TRUE
pedestal.fault_0 = TRUE
pedestal.state = operating
trans1.out_trans = TRUE
trans1.fault_0 = FALSE
trans1.state = operating
rcv.out_rcv = TRUE
rcv.fault_0 = FALSE
rcv.state = operating
trans2.out_trans = TRUE
trans2.fault_0 = TRUE
trans2.state = operating
-> State: 3.2 <-
control.out_control = FALSE
control.fault_0 = FALSE
control.state = internal_failure
ps.out_ps = FALSE
ps.fault_0 = FALSE
ps.state = operating
pedestal.out_pedestal = FALSE
pedestal.fault_0 = FALSE
pedestal.state = internal_failure
trans1.out_trans = FALSE
trans1.fault_0 = FALSE
trans1.state = operating
rcv.out_rcv = FALSE
rcv.fault_0 = FALSE
rcv.state = operating
trans2.out_trans = FALSE
trans2.fault_0 = FALSE
trans2.state = internal_failure
-> State: 3.3 <-
control.out_control = FALSE
control.fault_0 = FALSE
control.state = internal_failure
```

```
te = internal_failure ! trans2.state = internal_f
...
ing execution sequence
...
ample
```

no
no
0
0
1024
0
0,00 sec
0,00 sec

Model Explorer No Model Available

Outline SMV Model
 Contacts
 PowerSupply
 Control
 Transmitter
 Pedestal
 main

The screenshot shows an IDE with a project explorer on the left and a main editor window on the right. The project explorer shows a project named 'TrainDetectionSystem' with a sub-project 'GeneratedAltaRica'. The main editor window displays the content of 'out1_System.arc', which is an Open PSA file. The file content is as follows:

Node	Content
?-? xml	version="1.0" encoding="UTF-8" standalone="no"
open-psa	
define-fault-tree	
name	FT
define-gate	
name	TOP Event
or	
gate	
name	GateAND1
gate	
name	GateAND2
gate	
name	GateAND3
gate	
name	GateAND4
define-gate	
name	GateAND1
and	
basic-event	
name	in2_System_isAbsent
define-gate	
name	GateAND2
and	
basic-event	
name	concats.concatsFailToOpen_occurs
define-gate	
name	GateAND3
and	
basic-event	
name	generator2.internalFailure_occurs
define-gate	
name	GateAND4
and	
basic-event	
name	greenLight.internalFailure_occurs

Fault tree in
Open PSA format

CONTENTS

-  Motivation
-  Model Based Systems Engineering & Safety Analysis
-  Safety Analysis Toolset
-  Example
-  **Conclusion & Further work**

❖ System Modelling Environment for Safety Analysis

- The use of UML/SysML and Papyrus platform for further model checking and fault tree generation
 - Graphical and accessible language
 - Modelling of system architecture, behaviour and failure logic
- The support of two transformation methods and associated SA flows based on ARC(AltaRica) and NuSMV
- Fault tree generation, optimization and analysis
- Graphical representation of failure data using State Machine Diagrams

❖ Integrate different SA tools, plug-ins, profiles and libraries

❖ FMEA generation and analysis

❖ Develop SA profiles for SysML models

- Reflect the results of quantitative and qualitative safety analysis in SysML models
- Graphical representation of Fault Trees
- Distinguish failure modes in State Machines

Commissariat à l'énergie atomique et aux énergies alternatives
Institut Carnot CEA LIST
Centre de Saclay | 91191 Gif-sur-Yvette Cedex
T. +33 (0)1 XX XX XX XX | F. +33 (0)1 XX XX XX XX

Direction DRT
Département DILS
Laboratoire LISE