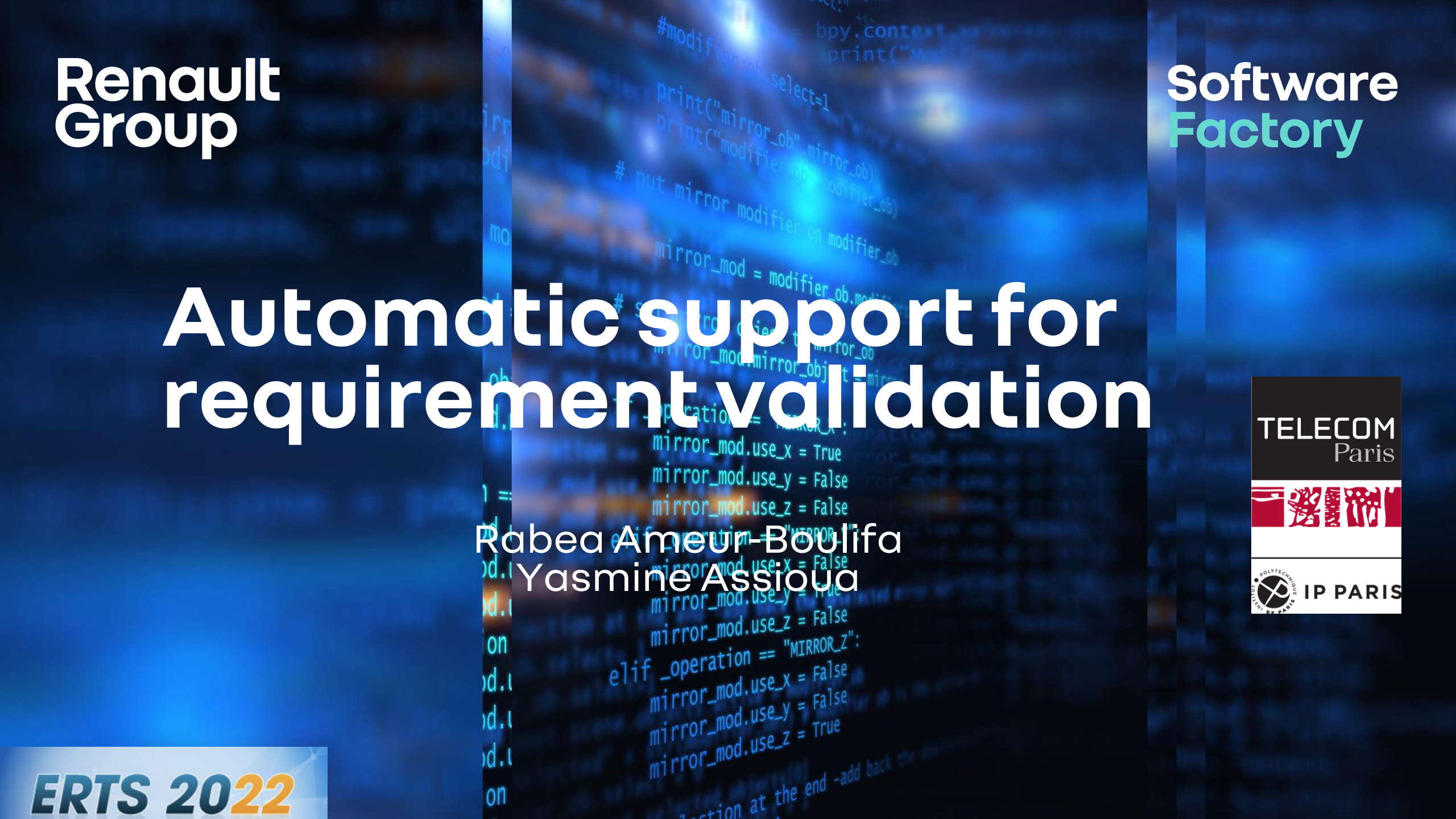


Renault Group

Software Factory

Automatic support for requirement validation

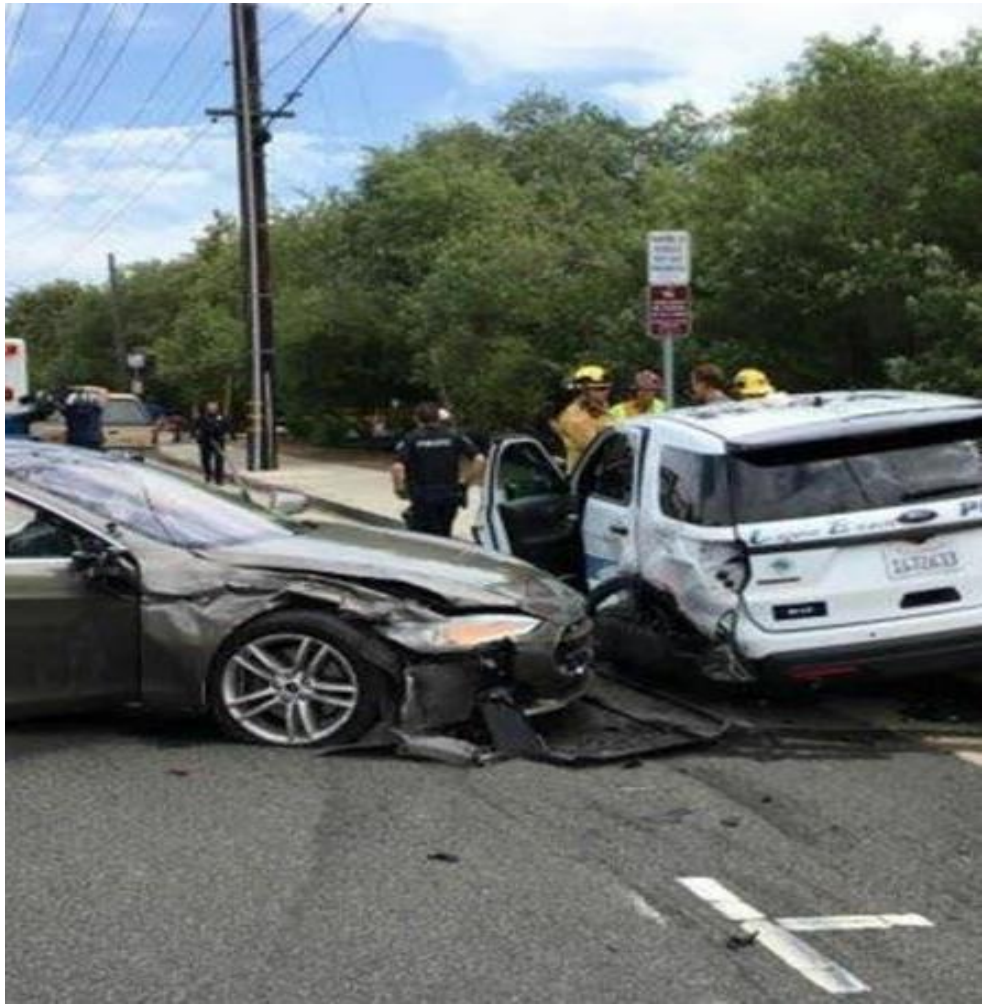
Rabea Ameur-Boulifa
Yasmine Assioua

TELECOM Paris

IP PARIS

ERTS 2022

Context



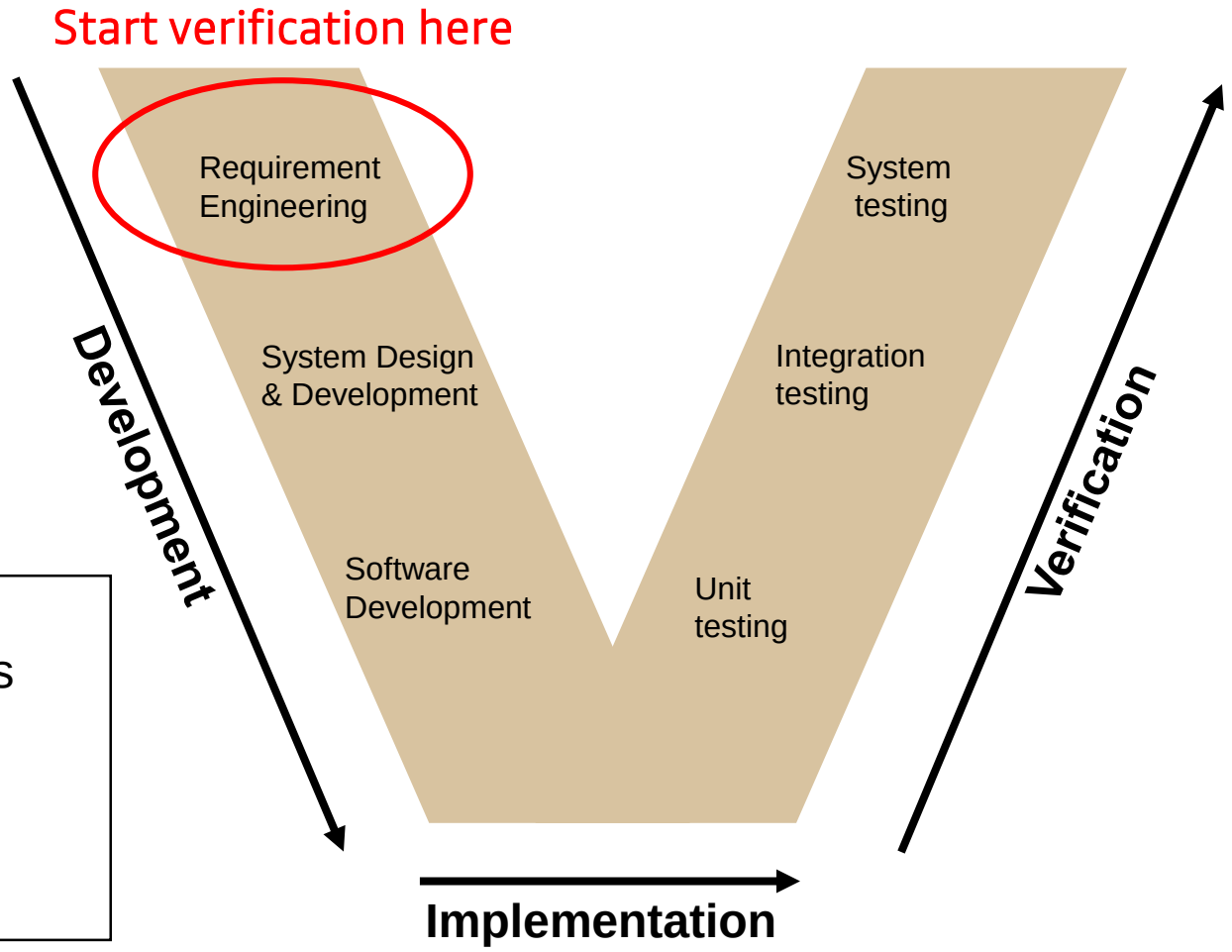
AUTONOMOUS VEHICLES

- Rapid development of new types of vehicles (more connected, more autonomous, …)
- Liability of automobile manufacturers in the event of failure

⋙ Method to **develop reliable and robust** vehicles and cover **highly critical verification**

How to proceed ?

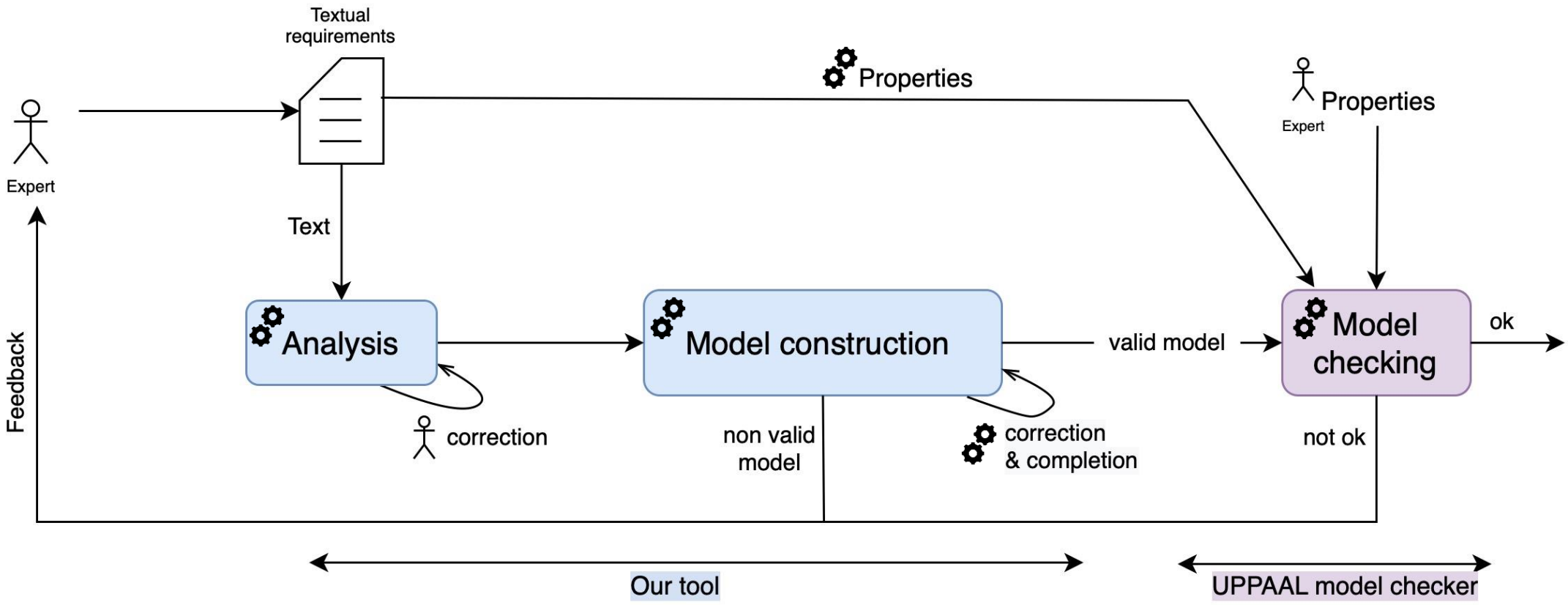
Standard software development process :



- Early validation of requirements
  - Identify ambiguities
  - Identify inconsistencies
  - Identify contradictions

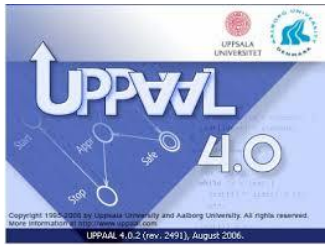
➤➤➤ Offer a **rigourous method** based on **formal methods** for **requirements early validation**

Our approach



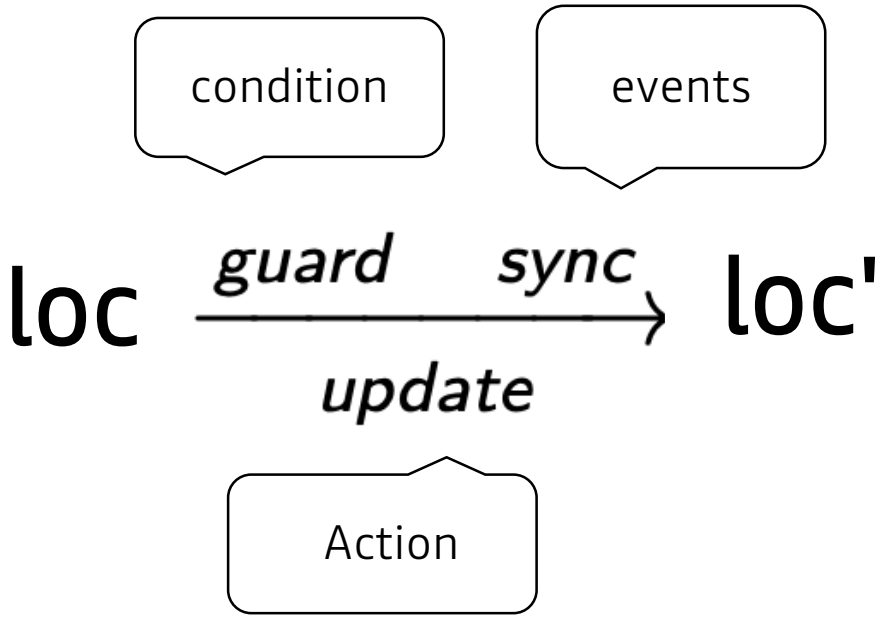
Model

- **Build a model** of the system under design
- Adopted **UPPAAL model => Model checking**



UPPAAL model :

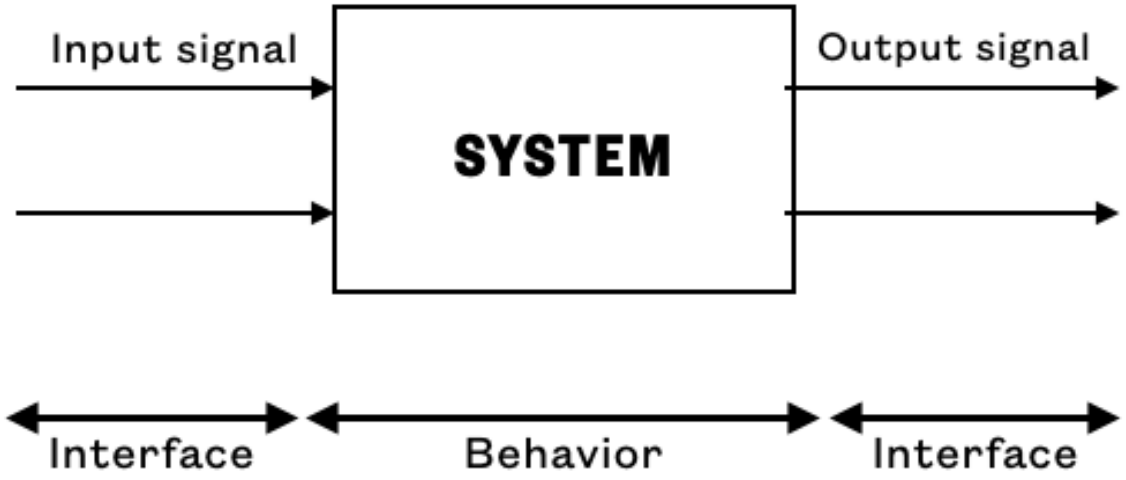
- Concurrents processes
- Process = Timed automata

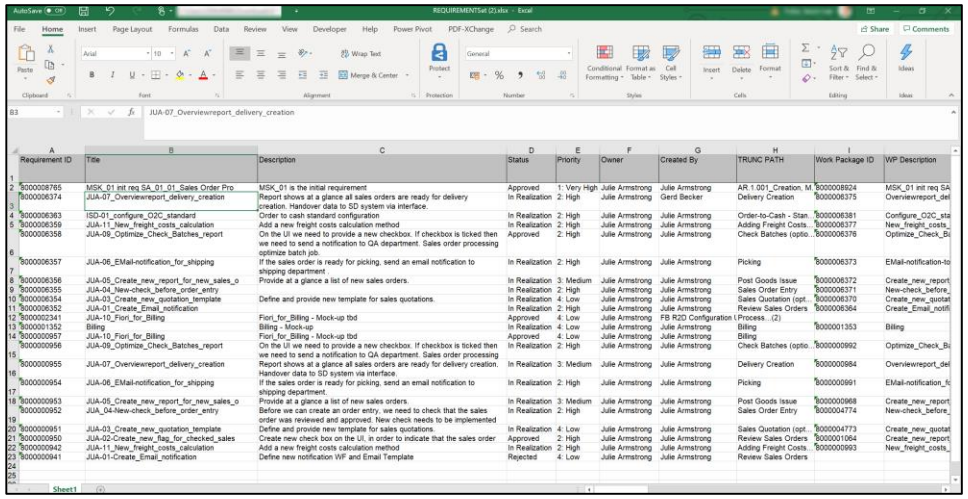


➤➤➤ **Systematic generation** of model from textual requirements

Textual requirements

- Each **system** is described by a set of requirements
- Requirements are in textual form, written in **natural langage**
- Requirements are of **various form** with a **specific syntax**





Requirement ID	Title	Description	Status	Priority	Owner	Created By	TRUNC PATH	Work Package ID	WP Description
8000008765	MSK_01 init req SA_01_01_Sales Order Pro	MSK_01 is the initial requirement	Approved	1: Very High	Julie Armstrong	Julie Armstrong	AR.1.001_Creation, M	8000008924	MSK_01 init req SA
8000006374	JUA-07_Overviewreport_delivery_creation	Report shows at a glance all sales orders are ready for delivery creation. Handover data to SD system via interface.	In Realization	2: High	Julie Armstrong	Gerrd Becker	Delivery Creation	8000006375	Overviewreport_del
8000006363	ISD-01_configure_O2C_standard	Order to cash standard configuration	In Realization	2: High	Julie Armstrong	Julie Armstrong	Order-to-Cash - Stan	8000006381	Configure_O2C_sta
8000006359	JUA-11_New_freight_costs_calculation	Add a new freight costs calculation method	In Realization	2: High	Julie Armstrong	Julie Armstrong	Adding Freight Costs	8000006377	New_freight_costs_
8000006358	JUA-09_Optimize_Check_Batches_report	On the UI we need to provide a new checkbox. If checkbox is ticked then we need to send a notification to QA department. Sales order processing optimize batch job.	Approved	2: High	Julie Armstrong	Julie Armstrong	Check Batches (optio	8000006376	Optimize_Check_Ba
8000006357	JUA-06_EMail-notification_for_shipping	If the sales order is ready for picking, send an email notification to shipping department	In Realization	2: High	Julie Armstrong	Julie Armstrong	Picking	8000006373	EMail-notification-to
8000006356	JUA-05_Create_new_report_for_new_sales_o	Provide at a glance a list of new sales orders.	In Realization	3: Medium	Julie Armstrong	Julie Armstrong	Post Goods Issue	8000006372	Create_new_report
8000006355	JUA-04_New-check_before_order_entry		In Realization	2: High	Julie Armstrong	Julie Armstrong	Sales Order Entry	8000006371	New-check_before_
8000006354	JUA-03_Create_new_quotation_template	Define and provide new template for sales quotations.	In Realization	4: Low	Julie Armstrong	Julie Armstrong	Sales Quotation (opt.	8000006370	Create_new_quotat
8000006352	JUA-01_Create_Email_notification		In Realization	2: High	Julie Armstrong	Julie Armstrong	Review Sales Orders	8000006364	Create_Email_notifi
8000002341	JUA-10_Fiori_for_Billing	Fiori_for_Billing - Mock-up tbd	Approved	4: Low	Julie Armstrong	FB R2D Configuration	Process...(2)		
8000001362	Billing	Billing - Mock-up	In Realization	4: Low	Julie Armstrong	Julie Armstrong	Billing	8000001353	Billing
8000000957	JUA-10_Fiori_for_Billing	Fiori_for_Billing - Mock-up tbd	Approved	4: Low	Julie Armstrong	Julie Armstrong	Billing		
8000000956	JUA-09_Optimize_Check_Batches_report	On the UI we need to provide a new checkbox. If checkbox is ticked then we need to send a notification to QA department. Sales order processing	In Realization	2: High	Julie Armstrong	Julie Armstrong	Check Batches (optio	8000000992	Optimize_Check_Ba
8000000955	JUA-07_Overviewreport_delivery_creation	Report shows at a glance all sales orders are ready for delivery creation. Handover data to SD system via interface.	In Realization	3: Medium	Julie Armstrong	Julie Armstrong	Delivery Creation	8000000984	Overviewreport_del
8000000954	JUA-06_EMail-notification_for_shipping	If the sales order is ready for picking, send an email notification to shipping department.	In Realization	2: High	Julie Armstrong	Julie Armstrong	Picking	8000000991	EMail-notification_fo
8000000953	JUA-05_Create_new_report_for_new_sales_o	Provide at a glance a list of new sales orders.	In Realization	3: Medium	Julie Armstrong	Julie Armstrong	Post Goods Issue	8000000968	Create_new_report
8000000952	JUA_04-New-check_before_order_entry	Before we can create an order entry, we need to check that the sales order was reviewed and approved. New check needs to be implemented	In Realization	2: High	Julie Armstrong	Julie Armstrong	Sales Order Entry	8000004774	New-check_before_
8000000951	JUA-03_Create_new_quotation_template	Define and provide new template for sales quotations.	In Realization	4: Low	Julie Armstrong	Julie Armstrong	Sales Quotation (opt.	8000004773	Create_new_quotat
8000000950	JUA-02-Create_new_flag_for_checked_sales	Create new check box on the UI, in order to indicate that the sales order	Approved	2: High	Julie Armstrong	Julie Armstrong	Review Sales Orders	8000001064	Create_new_report
8000000942	JUA-11_New_freight_costs_calculation	Add a new freight costs calculation method	In Realization	2: High	Julie Armstrong	Julie Armstrong	Adding Freight Costs	8000000993	New_freight_costs_
8000000941	JUA-01-Create_Email_notification	Define new notification WF and Email Template	Rejected	4: Low	Julie Armstrong	Julie Armstrong	Review Sales Orders		

- STRComp (system technical requirement component)
- 400 rows
- 300 textual requirements
- schemas, tables,

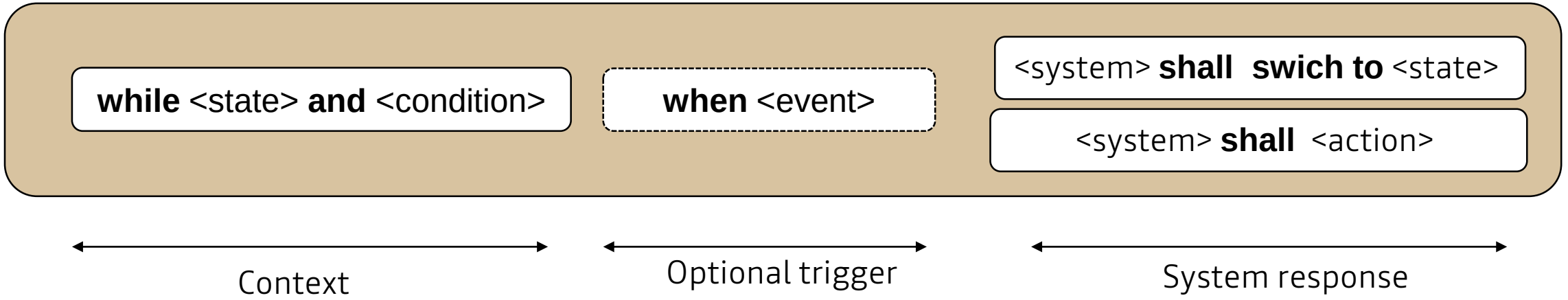
Language for system specification

- **Language** for requirements specification **as templates**
- EARS-like language
- **Each template** build a part of the model

Type	Pattern
Interface	⟨*system*⟩ **shall receive and process from** ⟨*actuator*⟩ **the signal** ⟨signal-name⟩ **(with the following values:** [-⟨values⟩]+**)?.**
	⟨system⟩ **shall send and process to** ⟨actuator⟩ **the signal** ⟨signal-name⟩**(with the following values:** [-⟨values⟩]+ **)?.**
State-driven	**while** ⟨*state*⟩ **and** ⟨condition⟩ **(when** ⟨trigger⟩**)?,** ⟨system⟩ **shall** ⟨action⟩**.**
Event-driven	**when** ⟨trigger⟩**,** ⟨system⟩ **shall** ⟨action⟩**.**
Action-driven	**when entering** ⟨state⟩**,** ⟨system⟩ **shall** [-⟨action⟩]+**.**
Event	⟨system⟩ **shall detect** ⟨trigger⟩**, if**⟨signal-name⟩ = ⟨value ⟩ **for more than** ⟨delay⟩**.**
Constraint	**if**⟨system⟩ **is in** ⟨state⟩ **and entrance conditions to**⟨state⟩ **are satisfied,** ⟨system⟩ **shall switch to** ⟨state⟩**.**

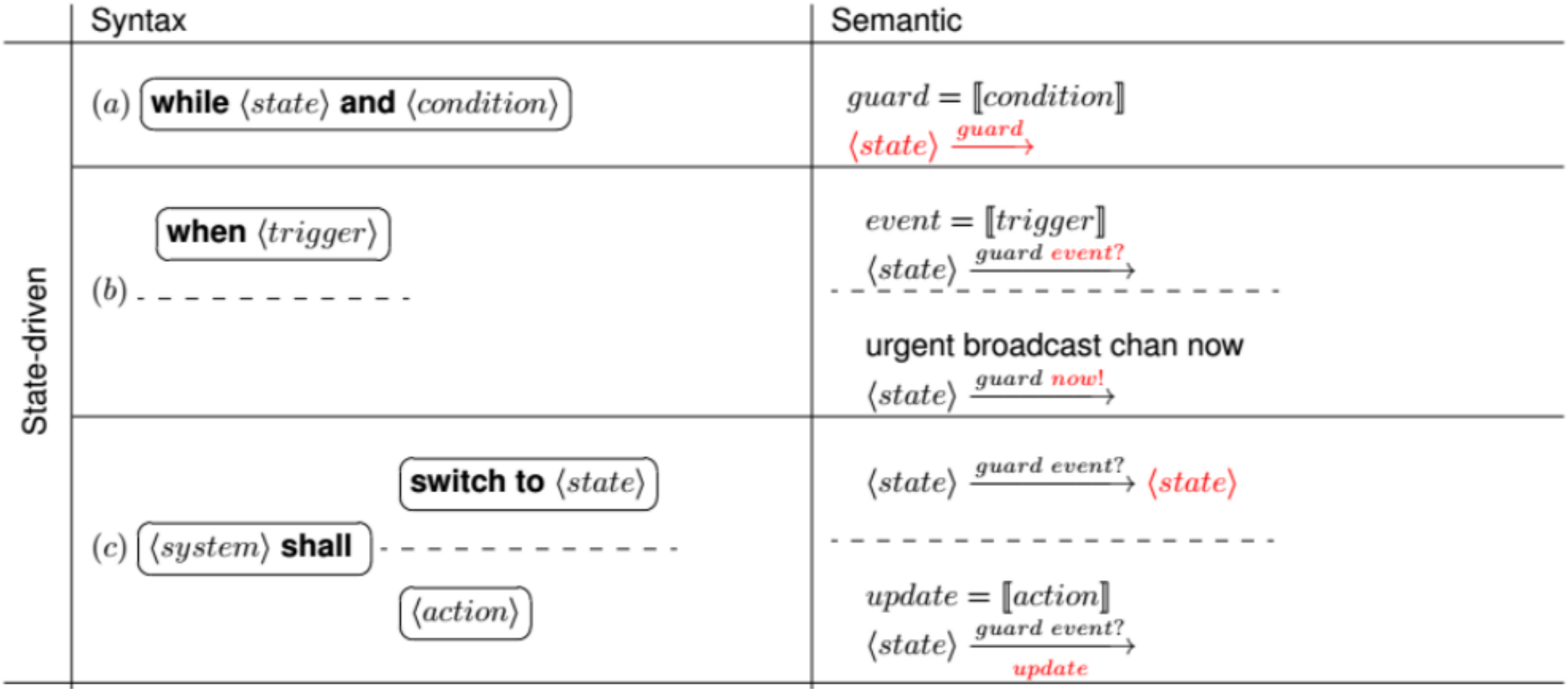
From textual requirements to model

- **State driven** requirement template



From textual requirements to model (continue)

- **Translation** algorithm



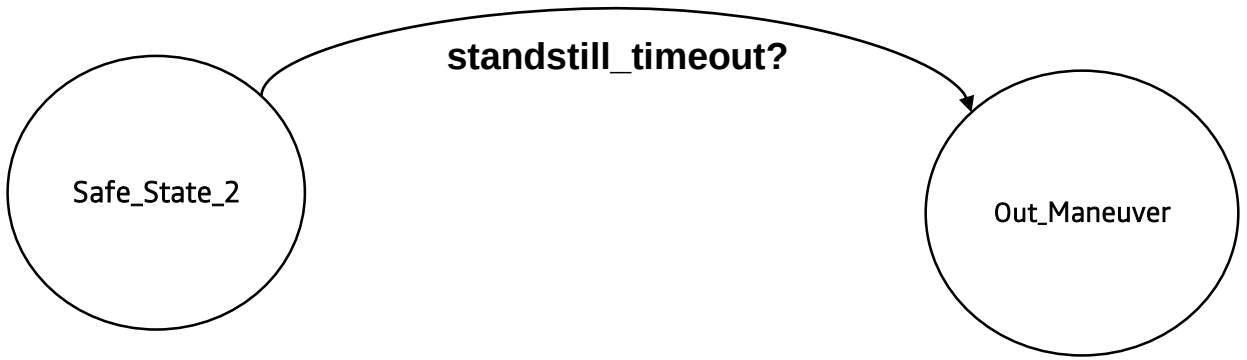
Illustration

Textual requirement

Translation algorithm

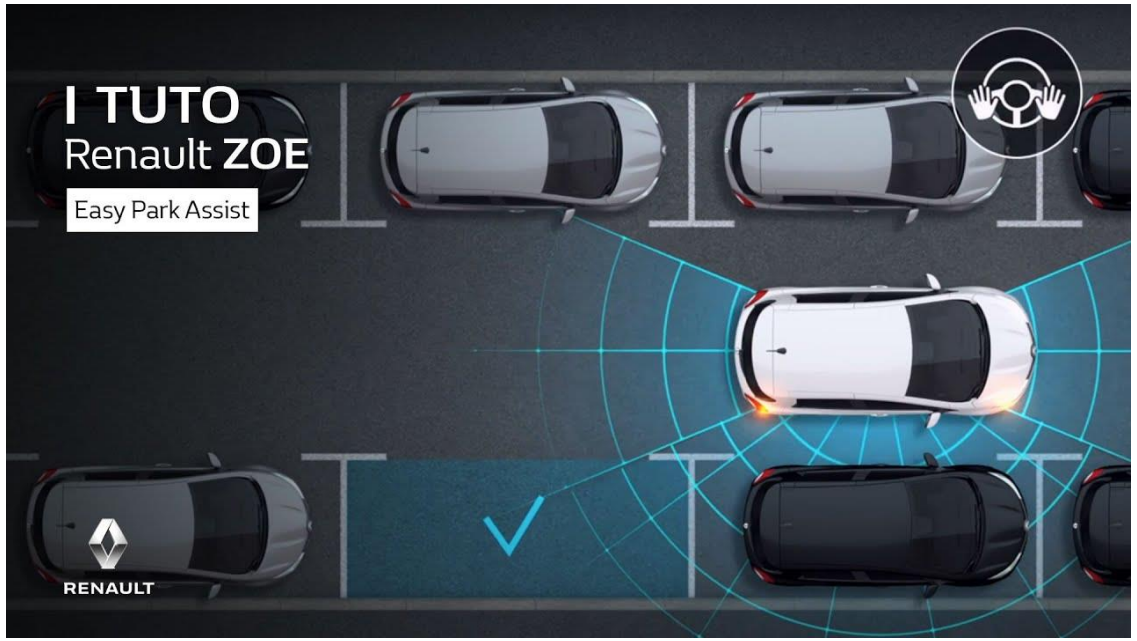
Formal model

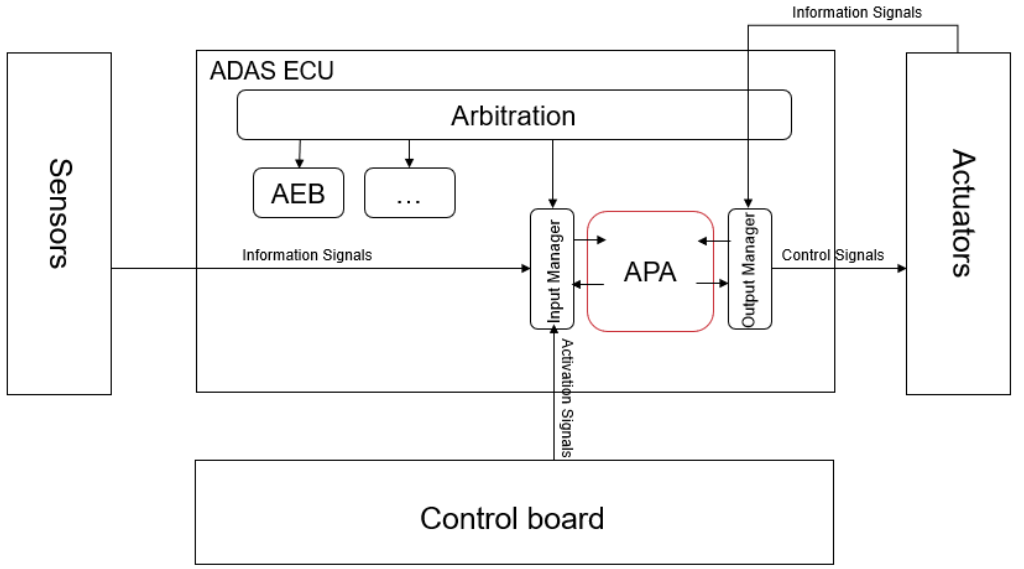
while Safe_State_2 **when** standstill_timeout **system shall switch to** Out_Maneuver



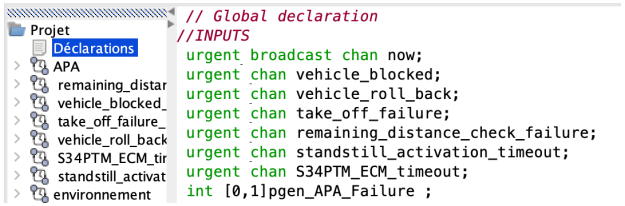
Use case: Automatic Park Assist (APA)

- Advanced driver assistance system (ADAS)
- Helps the driver during the parking maneuver

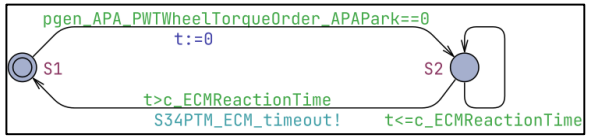




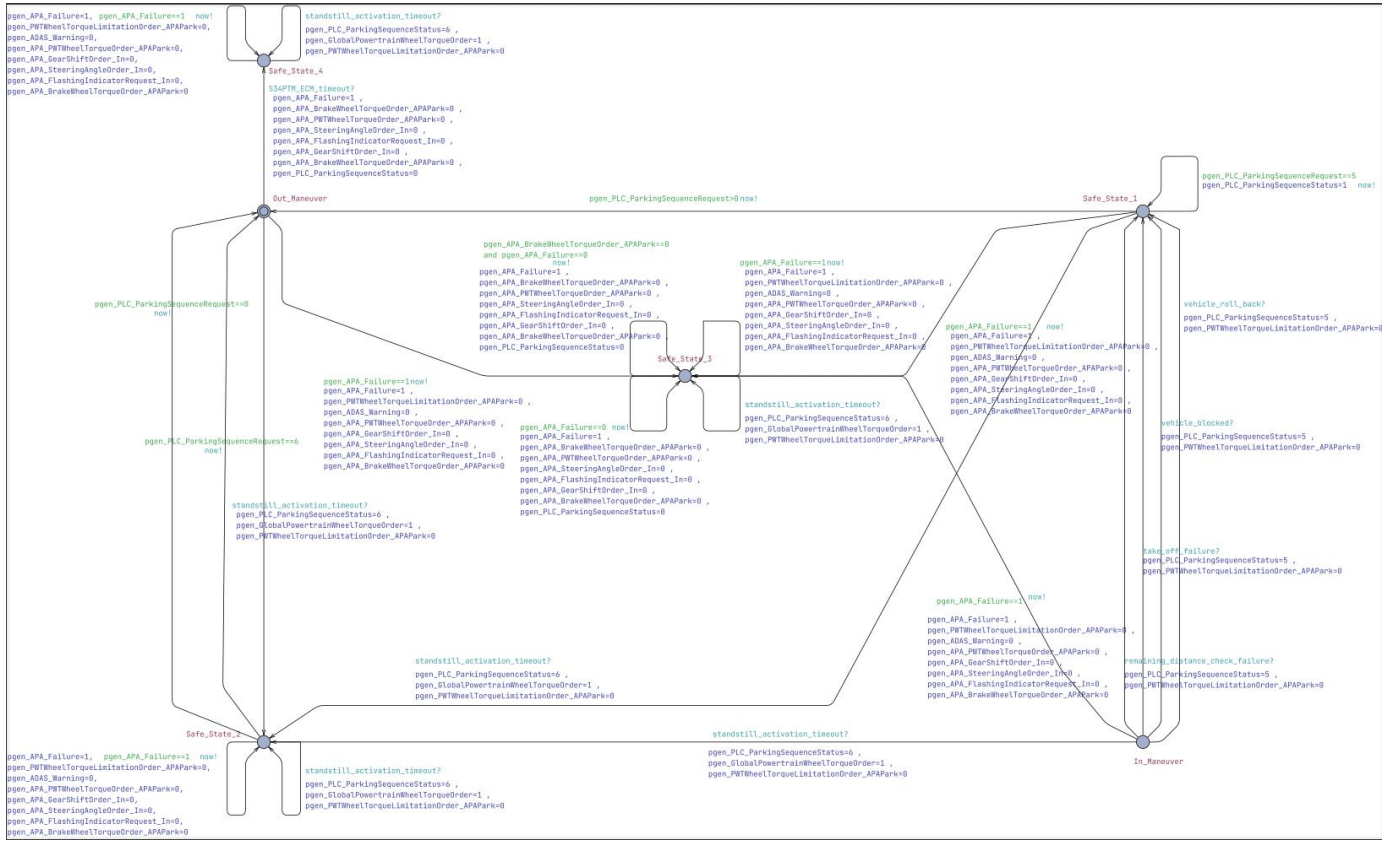
Use case: Automatic Park Assist (APA) (continue)



Interface declaration

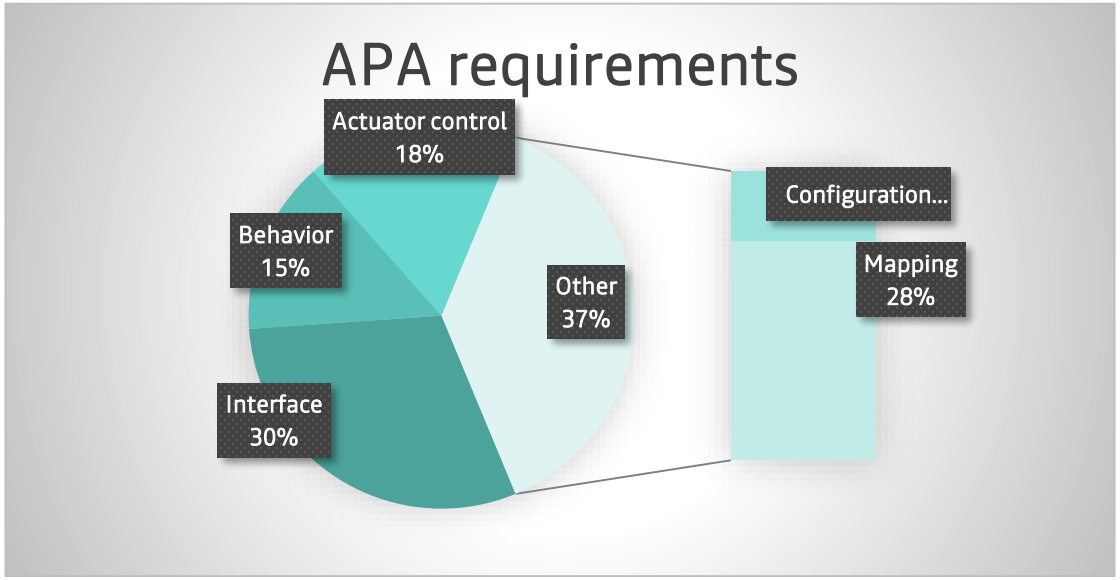


Event emission templates X7



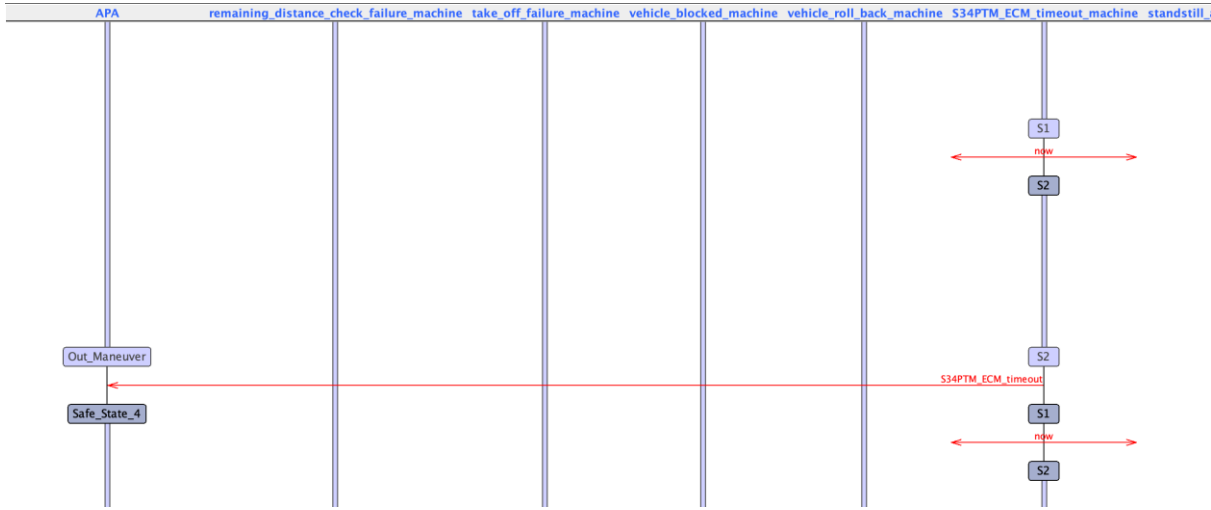
Main Template

Model checker application



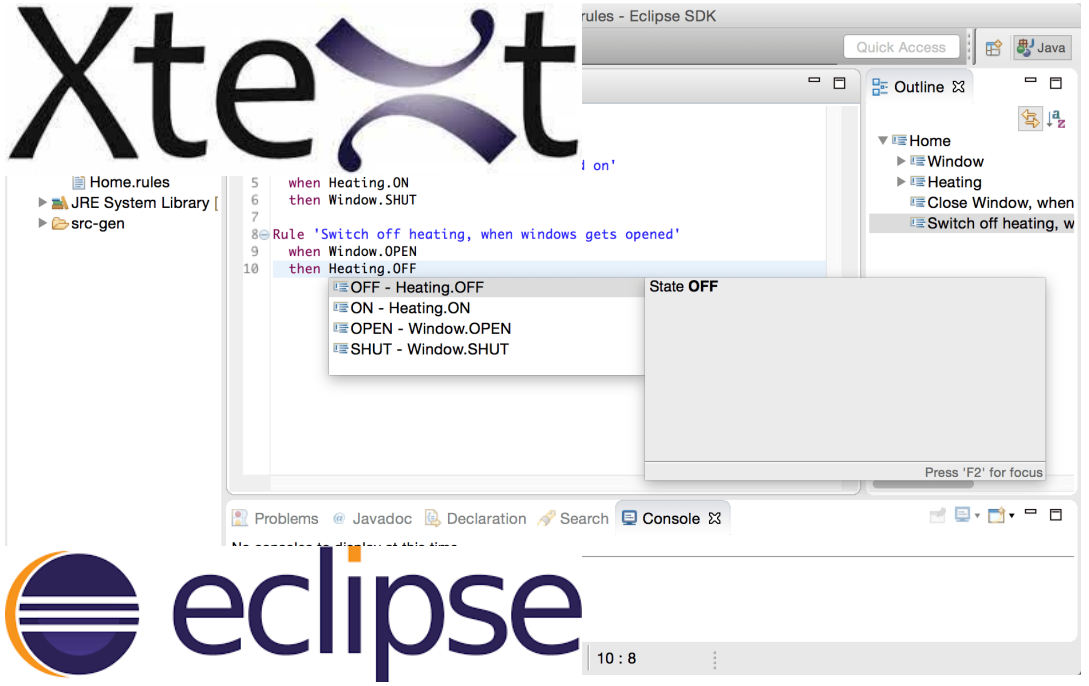
- **Simulation**
  - Sequence diagram
  - Gantt diagram

- **Properties verification**
  - Generic properties (deadlock, state reachability)
  - Specific properties (requirements, external expert)



Conclusion

- **Proof of work** of the method
  - Two uses cases : Autonomous driving & Automatic park assist
- Requirements **language patterns**
  - Proposition of language
  - Scalable regarding engineers needs
- Translation **algorithm**
- **Implementation** using Eclipse
  - Tool for systematic model generation from textual requirelents
- **Validation** on real case studies:
  - APA safety module
  - Powertrain Activation function



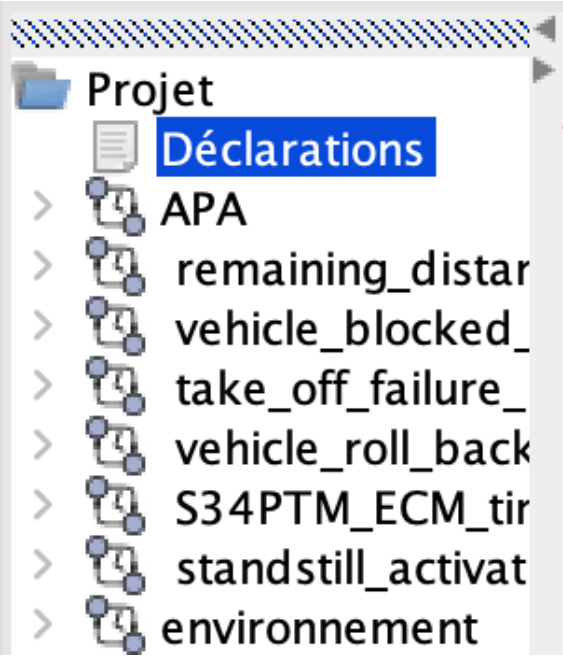
Any questions ?

Thank you

#1 Interface rule

REQ 01.

APA system shall receive and process the signal **pgen_Failure** with the following values : - **vgen_NoFailure** - **vgen_Failure**

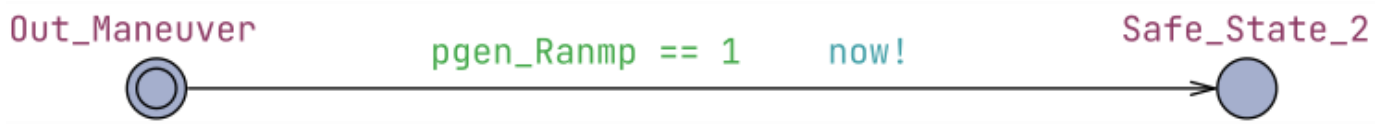



```
// Global declaration
//INPUTS
urgent broadcast chan now;
urgent chan vehicle_blocked;
urgent chan vehicle_roll_back;
urgent chan take_off_failure;
urgent chan remaining_distance_check_failure;
urgent chan standstill_activation_timeout;
urgent chan S34PTM_ECM_timeout;
int [0,1]pgen_APA_Failure ;
```

#2 State driven rule

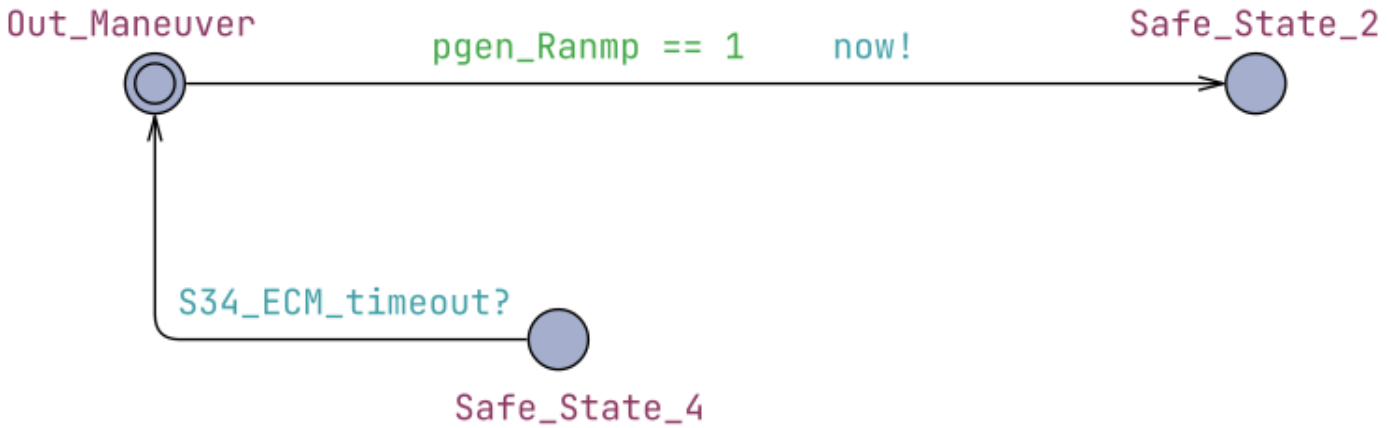
REQ 02.

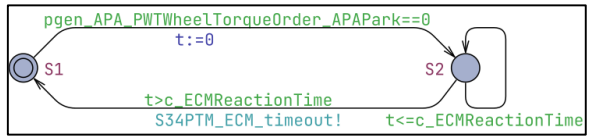
While APA **is in** Safe_State_2 **and** pgen_Ramp =vgen_on, APA **system shall switch to** Out_Maneuver



REQ 03

While APA **is in** Out_Maneuver **when** S34_ECM_Timeout, APA **system shall switch to** Safe_State_4.

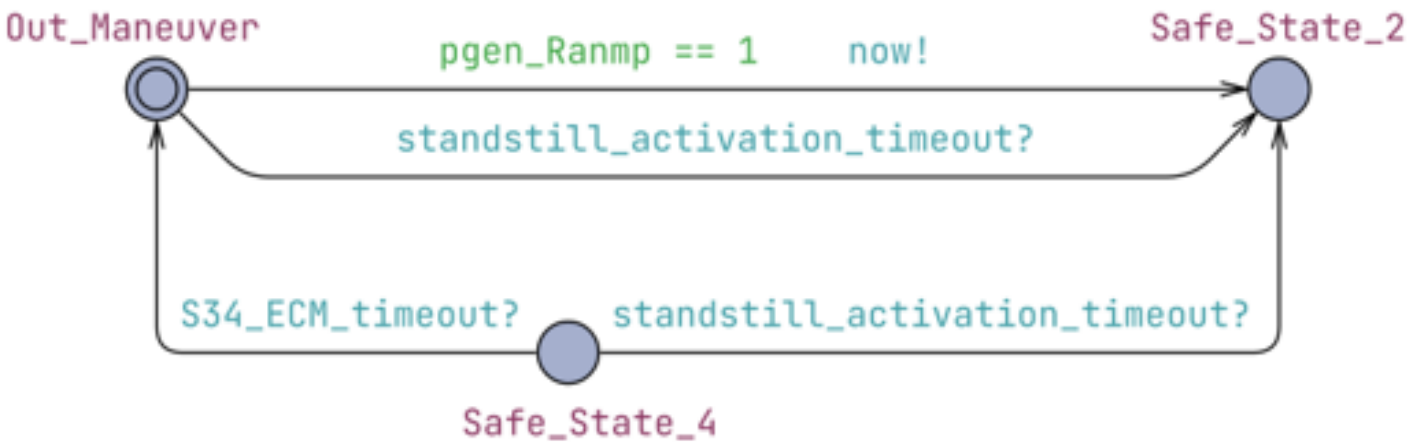


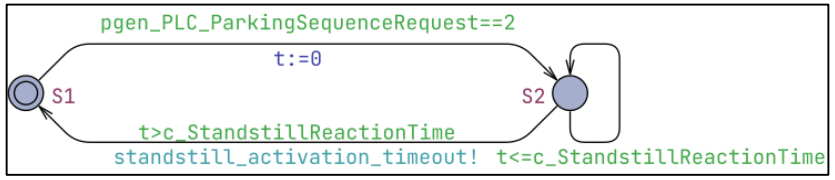


#3 Event driven rule

REQ 04.

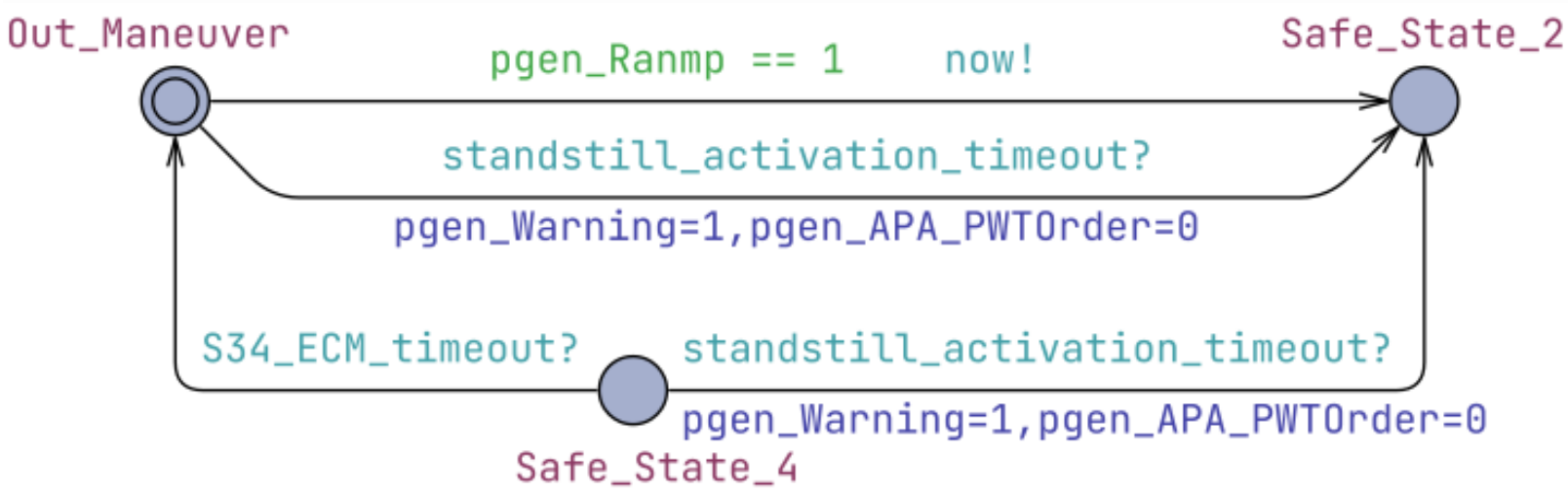
When standstill activation timeout APA system **shall switch to** Safe_State_2.





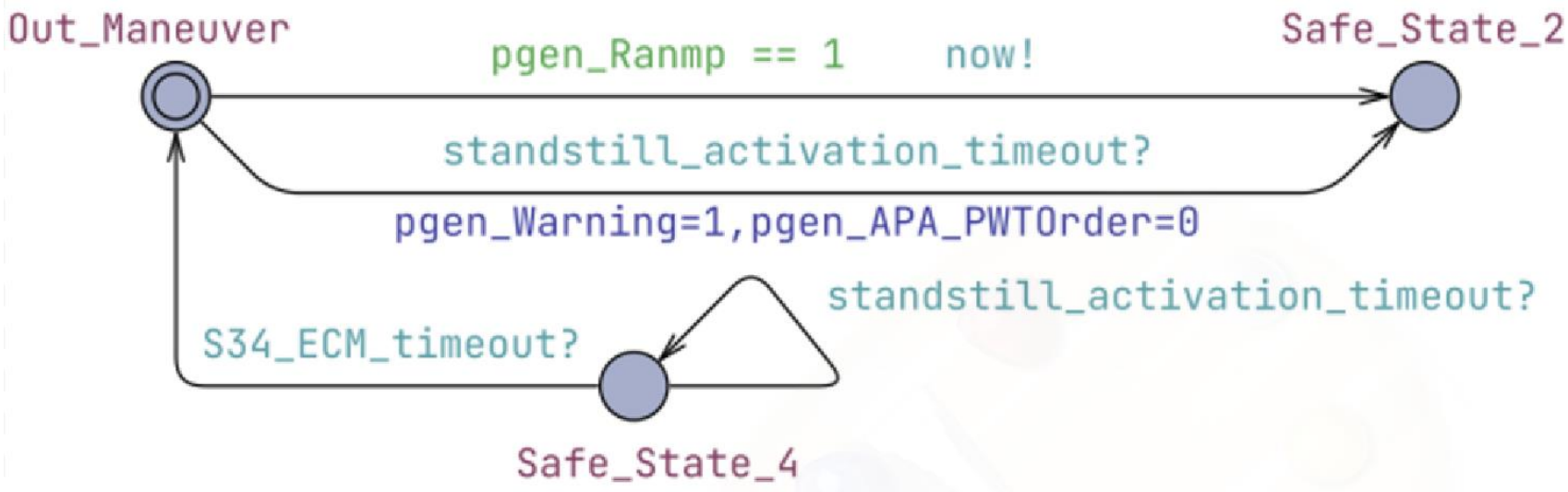
REQ 06.

When entering Safe_State_2 **APA system shall : set** pgen_Warning **to** vgen_Alert – **release** powertrain **control**

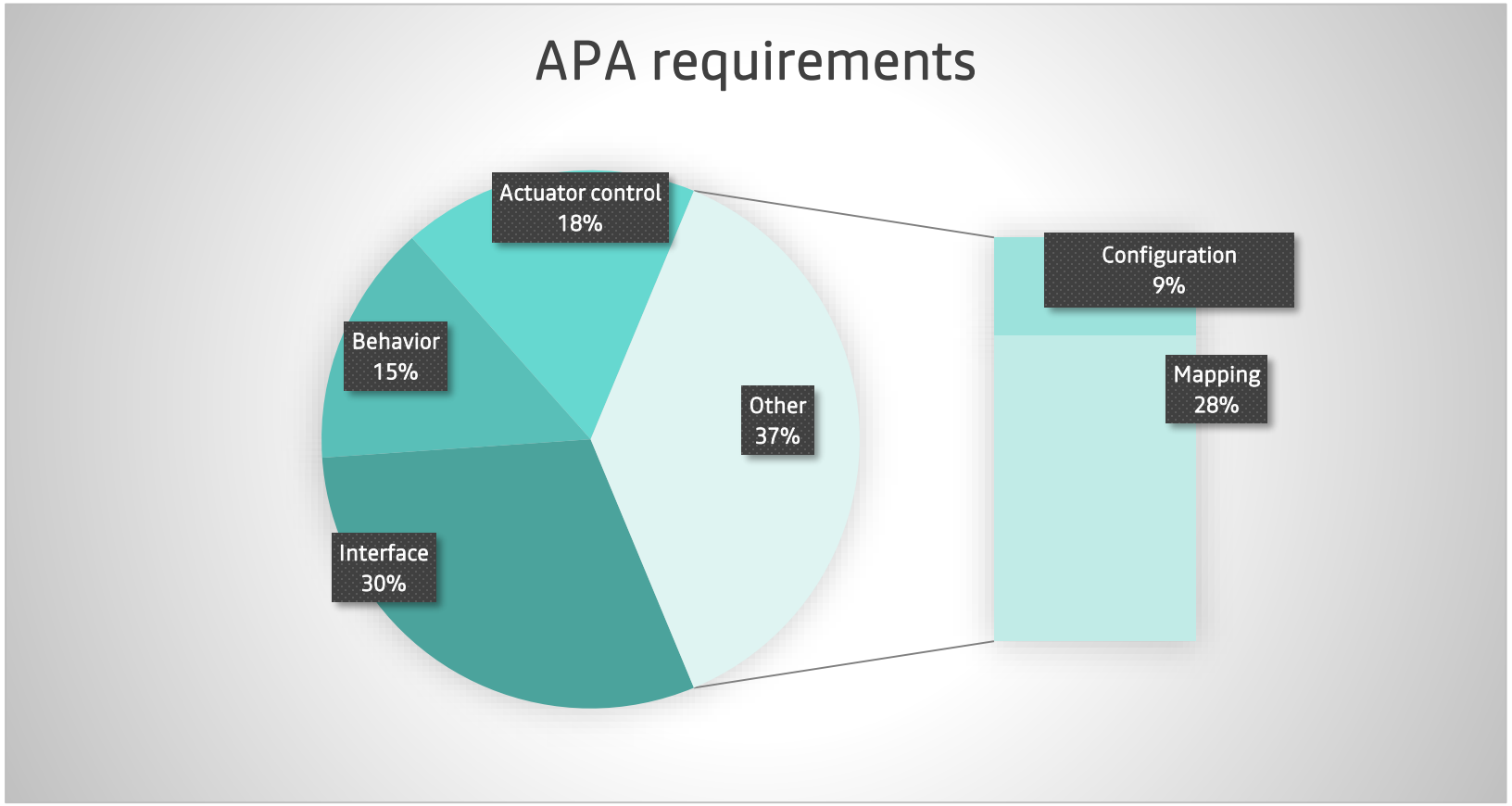


REQ 08.

If APA system **is in** Safe_State_4 **and entrance conditions to** Safe_State_2 **are satisfied,** APA system **shall switch to** Safe_State_4**.**



USE CASE : Automatic Park Assist (APA)



Requirement analysis

- Multiple definition of same object (state, events..)
- spelling mistakes
- Multiple interpretation for the same terms
- Ambigous terms
- Use of negation

⇒ Standardize terms and define key words
⇒ ~70-80 % of rewrite to apply the systematic translation

WHY MODELISE REQUIREMENTS ?

- **Visualise** all requirements
- Have an **abstraction** of the system
- Apply **model checking** method (automatic **formal method**)