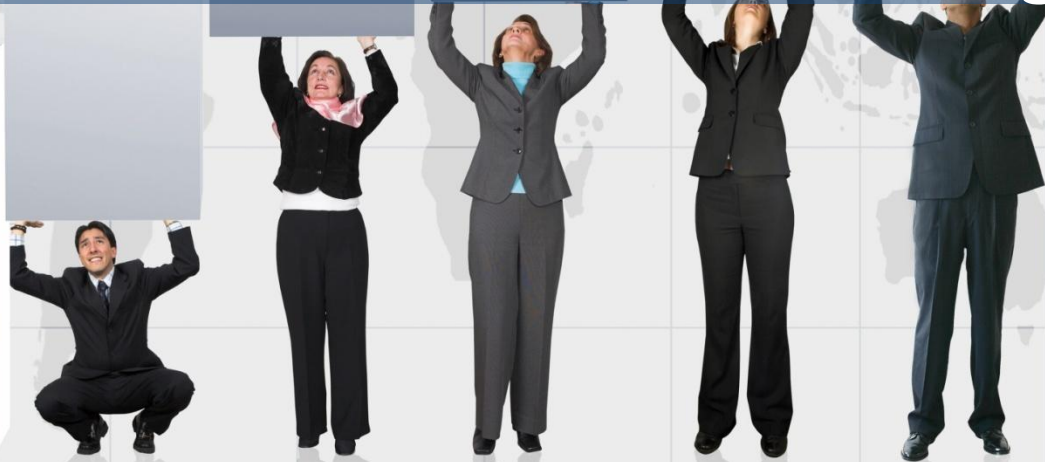# OpenETCS WP7

Benchmark Subset_026 : SysML session – Modeling with Papyrus Mode Switching case

SUMMARY

1. **Benchmark**
2. **Modeling in SysML with Papyrus**
3. **Mode Switching case**
4. **Model Based Testing and SysML**
5. **Conclusion**

1 - Benchmark

- **Chapter 5.9 : Procedure On-Sight :**
  - **Present Mode Switching**
  - **Use a timer**
- **Our model only consider transitions from FS to OS, and from OS to FS for an easier understanding**
- **Use a simplified system context for benchmark purpose**
- **Ongoing work with the CEA**
- **Transitions are fully described, but not yet implemented (mainly text)**
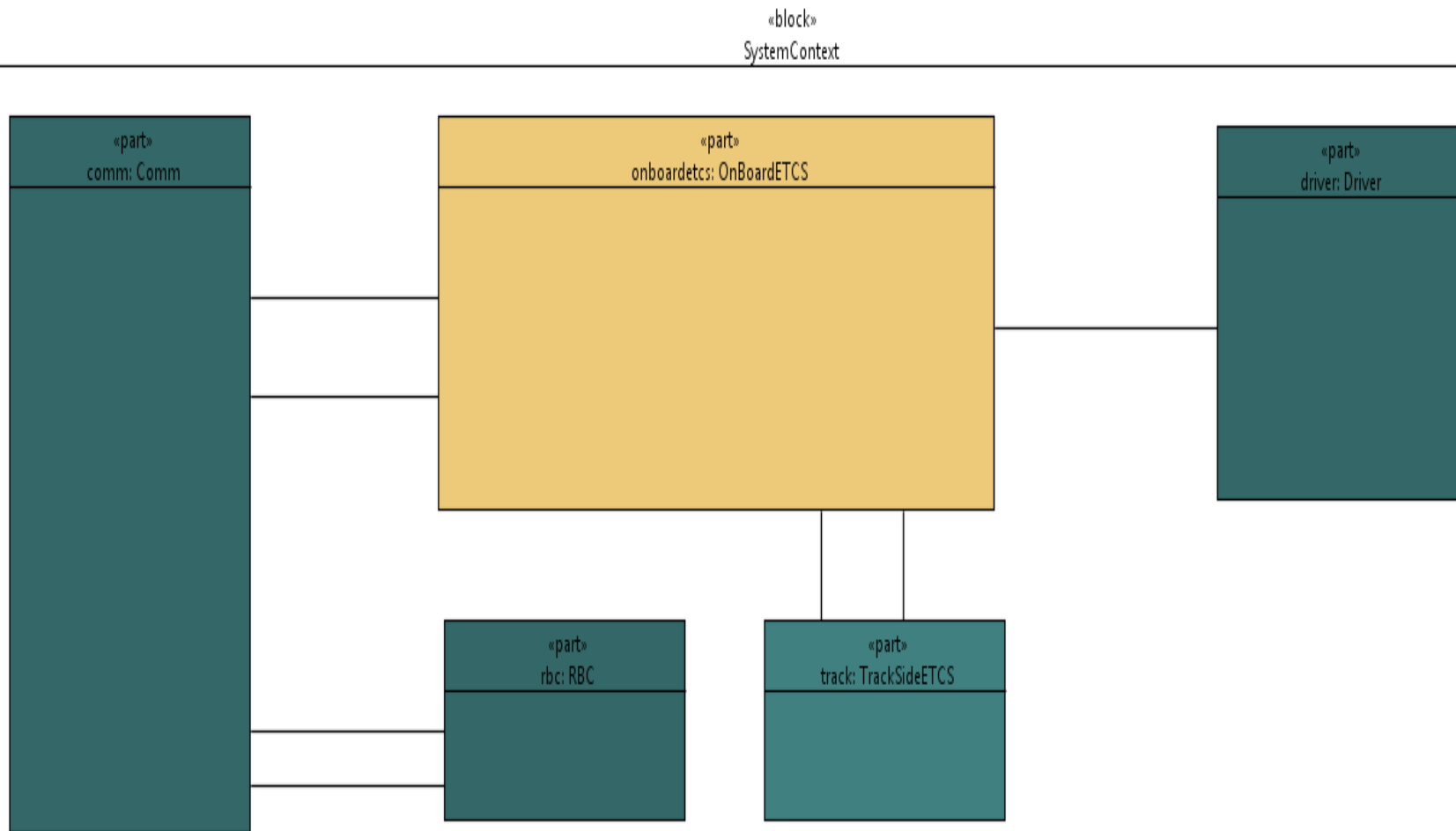
ALL4TEC
ETUDES & CONSEIL

2 – Modeling in SysML with Papyrus

- **Use of IBD, StateMachines and Activity diagrams**

- **Plan to set up requirements traceability and sequence diagrams**

- **Can be extended to model the whole transitions, using statemachines**

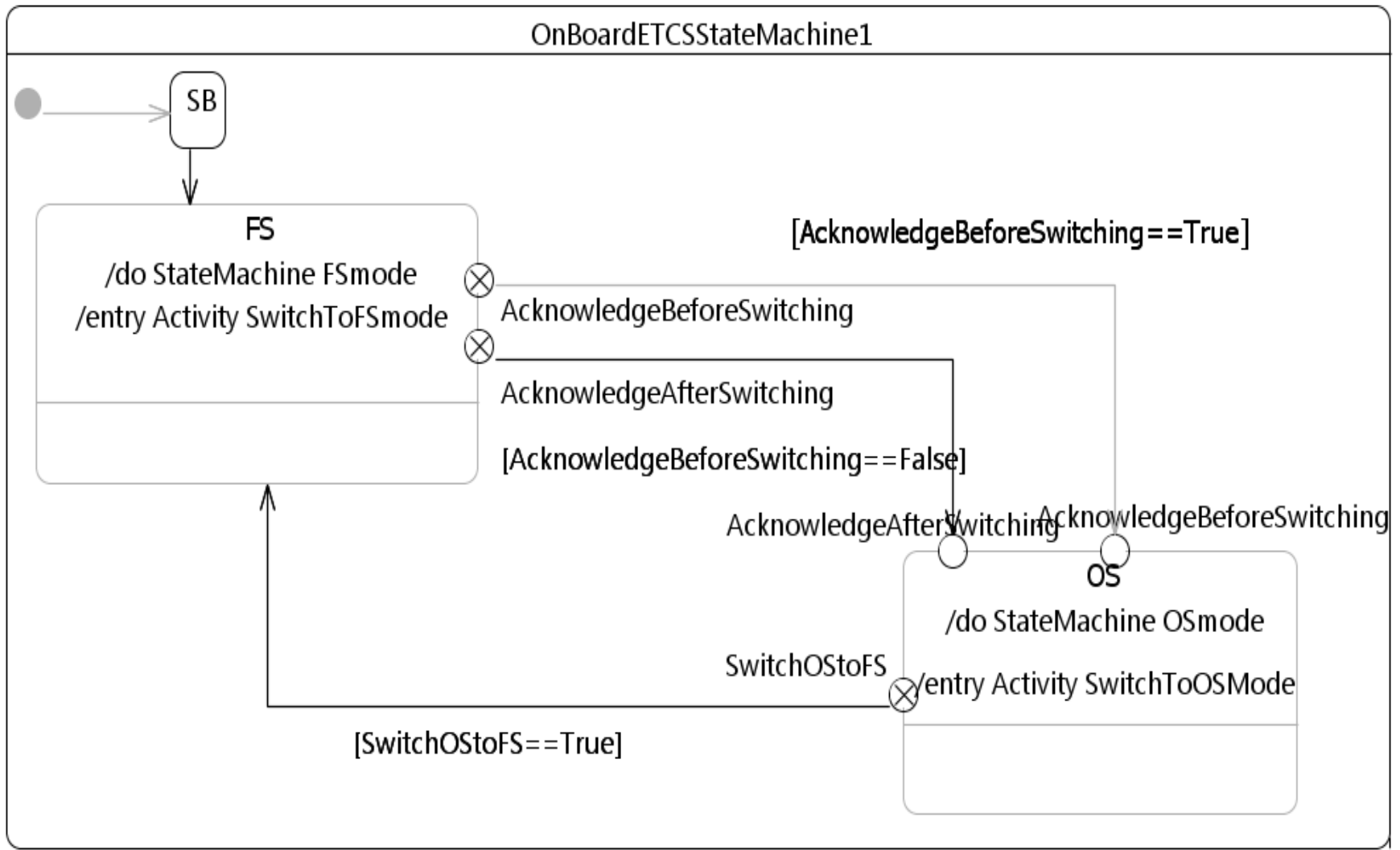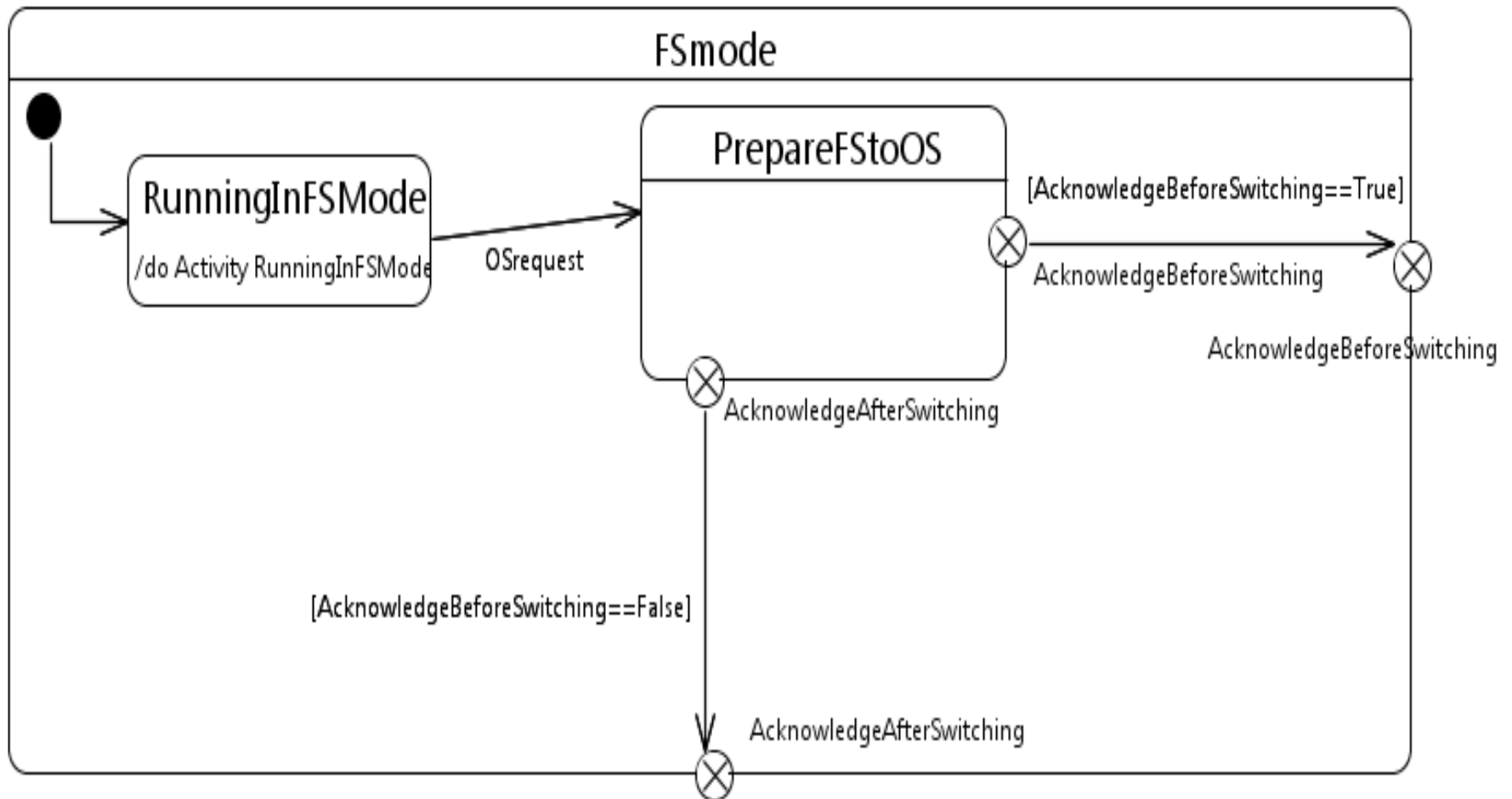- **Many solutions to model mode switching (see part 3)**

Full system context
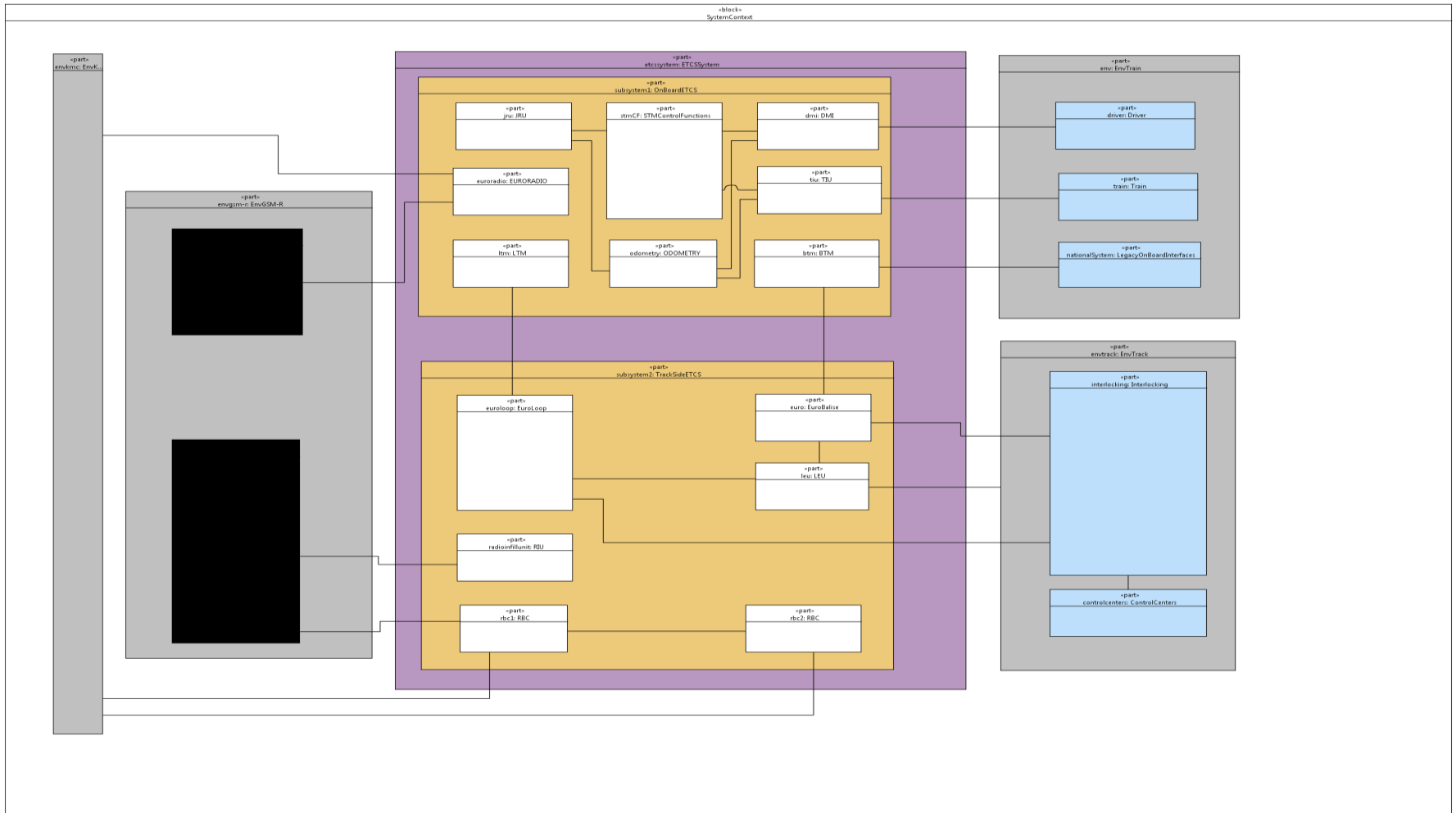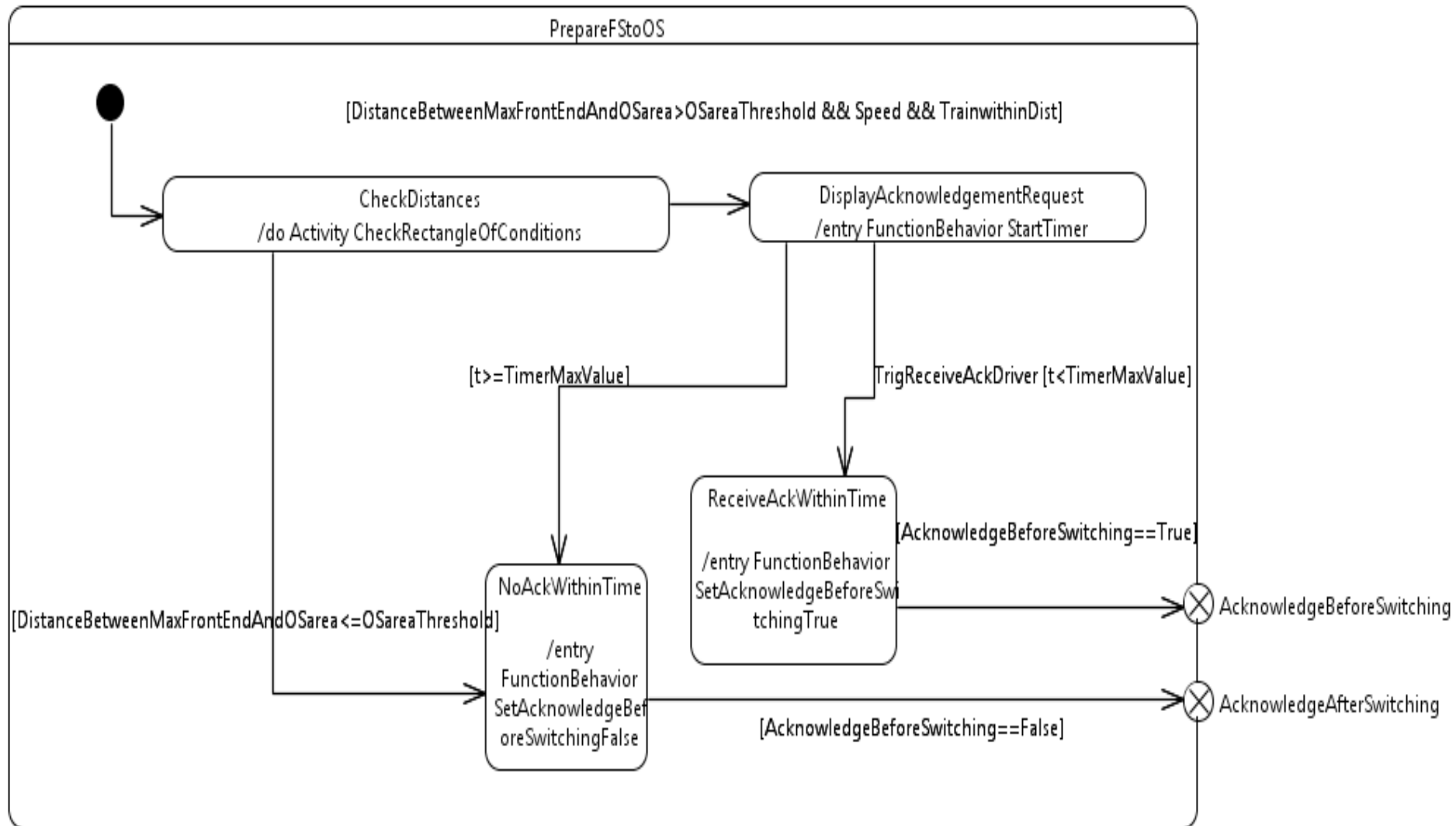
Simplified system context
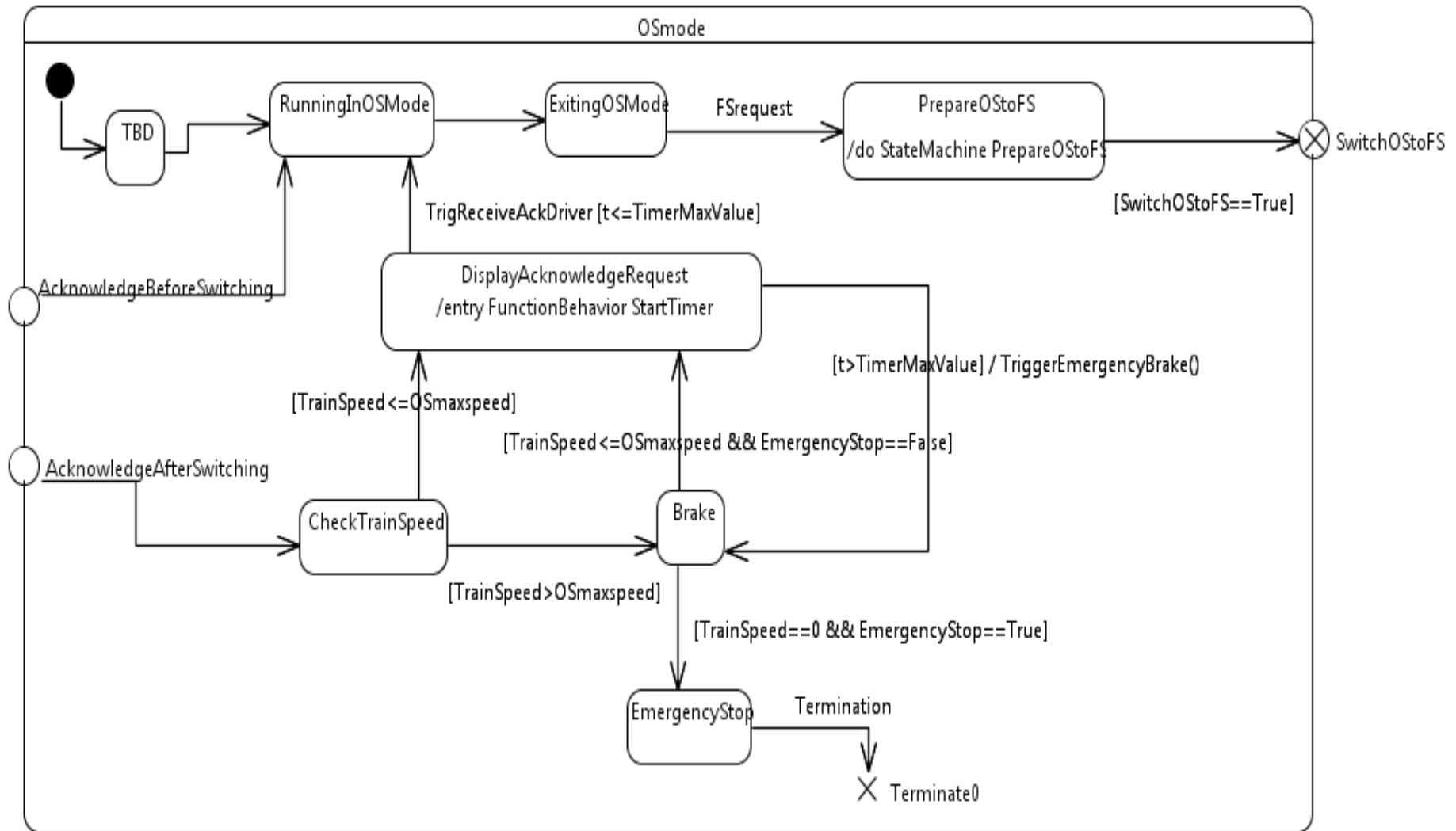
OnBoard State Machine

Beginning of the procedure

Inside the procedure – FS side

Inside the procedure – OS side

3 – Mode Switching case

- **Validation of the norm trough studies of scenarios:**
  - **Critical because dangerous**
  - **And/or lightly/bad defined configurations in the norm**

- **Definitions scattered through the norm :**
  - **Mode definitions: § 4.4.9, § 4.4.12 and § 5.9**
  - **Mode transition: § 4.6**
  - **Condition of transition: § 4.6, § 4.8 and § 5.9**
  - **Acceptation of information : § 4.8**

- **Specification model should help designer precise their thought**

ALL4TEC
ETUDES & CONSEIL

First model proposed

FS → OS

Change Mode Trigger
[Change Mode Guard]
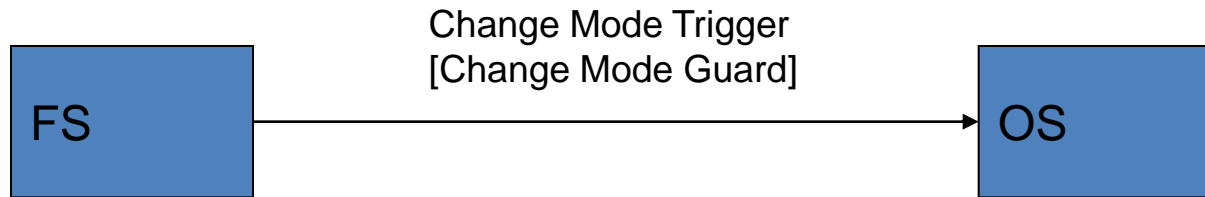
Pros :

- Readable in term of Mode-switching

Cons :

- Do not explicit the tests and activities related to the mode switch
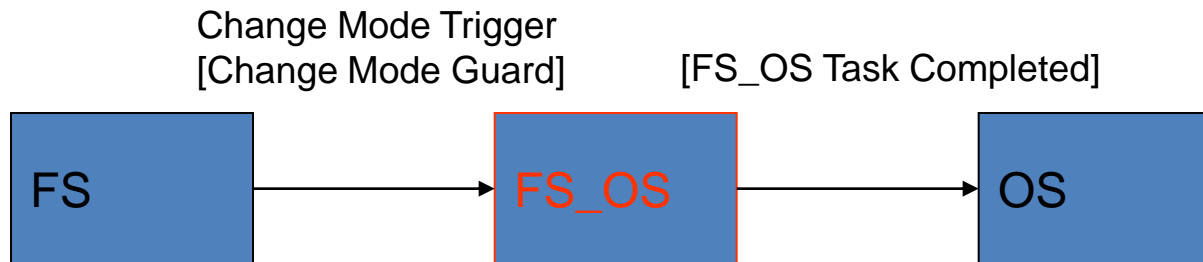
Add an intermediate state

Change Mode Trigger
[Change Mode Guard]   [FS_OS Task Completed]

```
┌──────────┐          ┌──────────┐          ┌──────────┐
│          │          │          │          │          │
│   FS     │─────────▶│  FS_OS   │─────────▶│   OS     │
│          │          │          │          │          │
└──────────┘          └──────────┘          └──────────┘
```
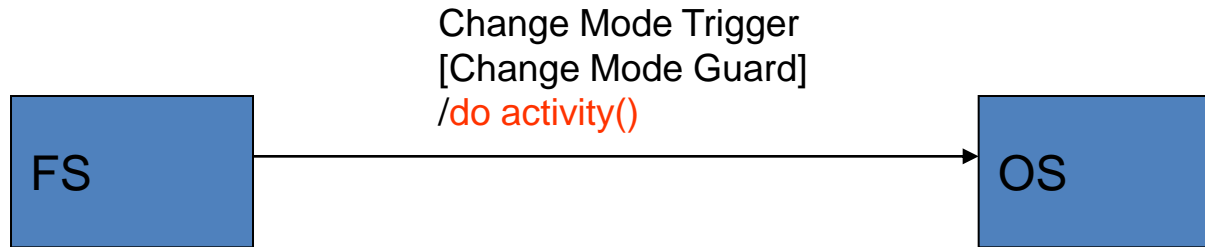
Pros :

- Mode switching state is explicitly shown in diagram

- Tests and specific behavior may be directly described in the newly created state

Cons :

- Added states may be unstable for some mode switching

- If generalized, need to add a lot of state that will probably mess up the mode diagram

ALL4TEC
ETUDES & CONSEIL

Insert tests and activities in transitions activity

Change Mode Trigger
[Change Mode Guard]
/do activity()

FS → OS

Pros :

- Few modifications
- Keep a good readability of the mode diagram

Cons :

- This design can lead to serious safety issues when the activity failed
- Complex or network-related activities may lead to heavy time consumption during which the mode diagram cannot react to mode change requests

ALL4TEC
ETUDES & CONSEIL

# Make a « switch mode » statemachine
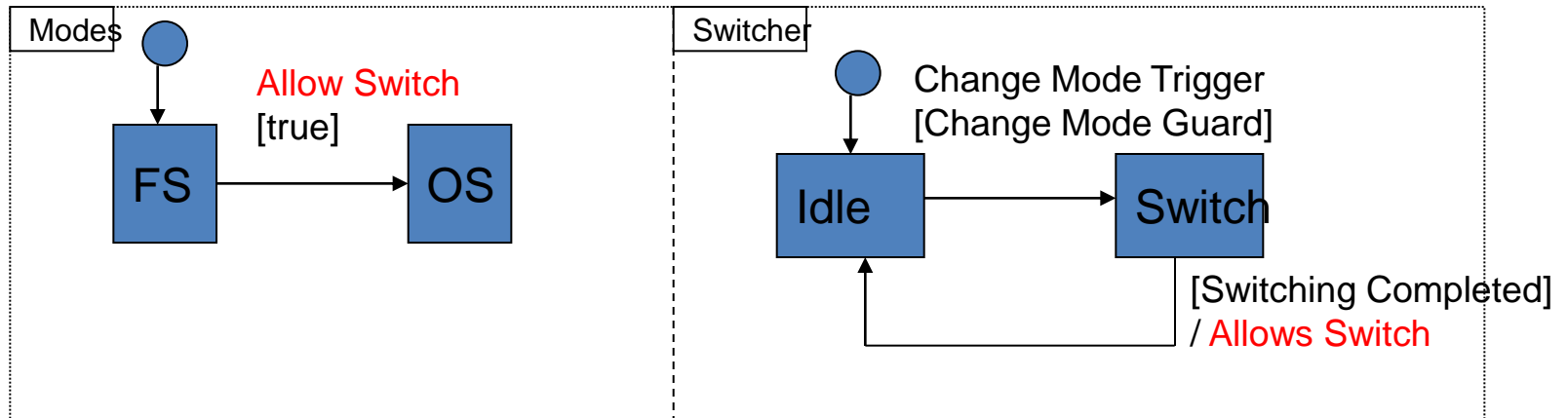
Pros :

- Few modifications and clear division between modes and mode-switching

- Coherent with the mode transition table described in the norm

- Can be extended to statemachines such as train behavior (brake/accelerate, etc..)

Cons :

- (VERY) Complex statemachine to build

ALL4TEC
ETUDES & CONSEIL

Distribute test and activities between source and target modes

FS | [FS Related Tests Completed] → | OS

(FS - OS)
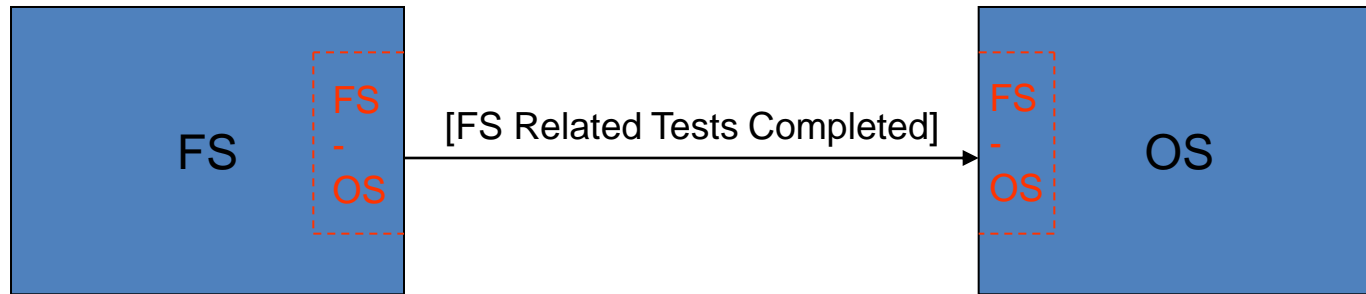
Pros :

- Mode remains unchanged until configuration to switch is safe
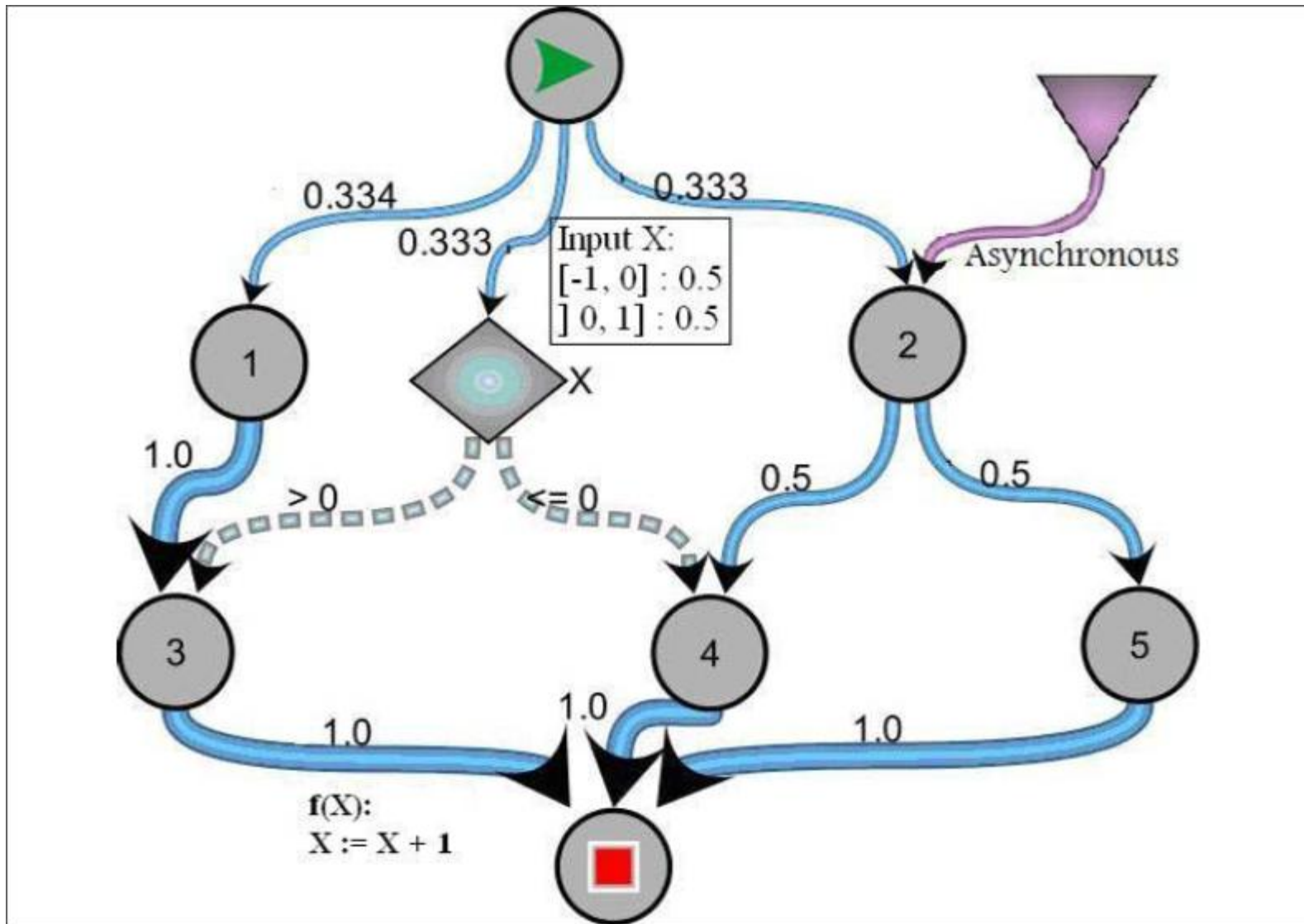- Behavior "related" to each mods are within their modes

Cons :

- Need heavy modifications of modes
- Define behavior "relation" to modes may be handy
- Initial request to change mode may not appear

ALL4TEC
ETUDES & CONSEIL

4 – Model Based Testing and SysML

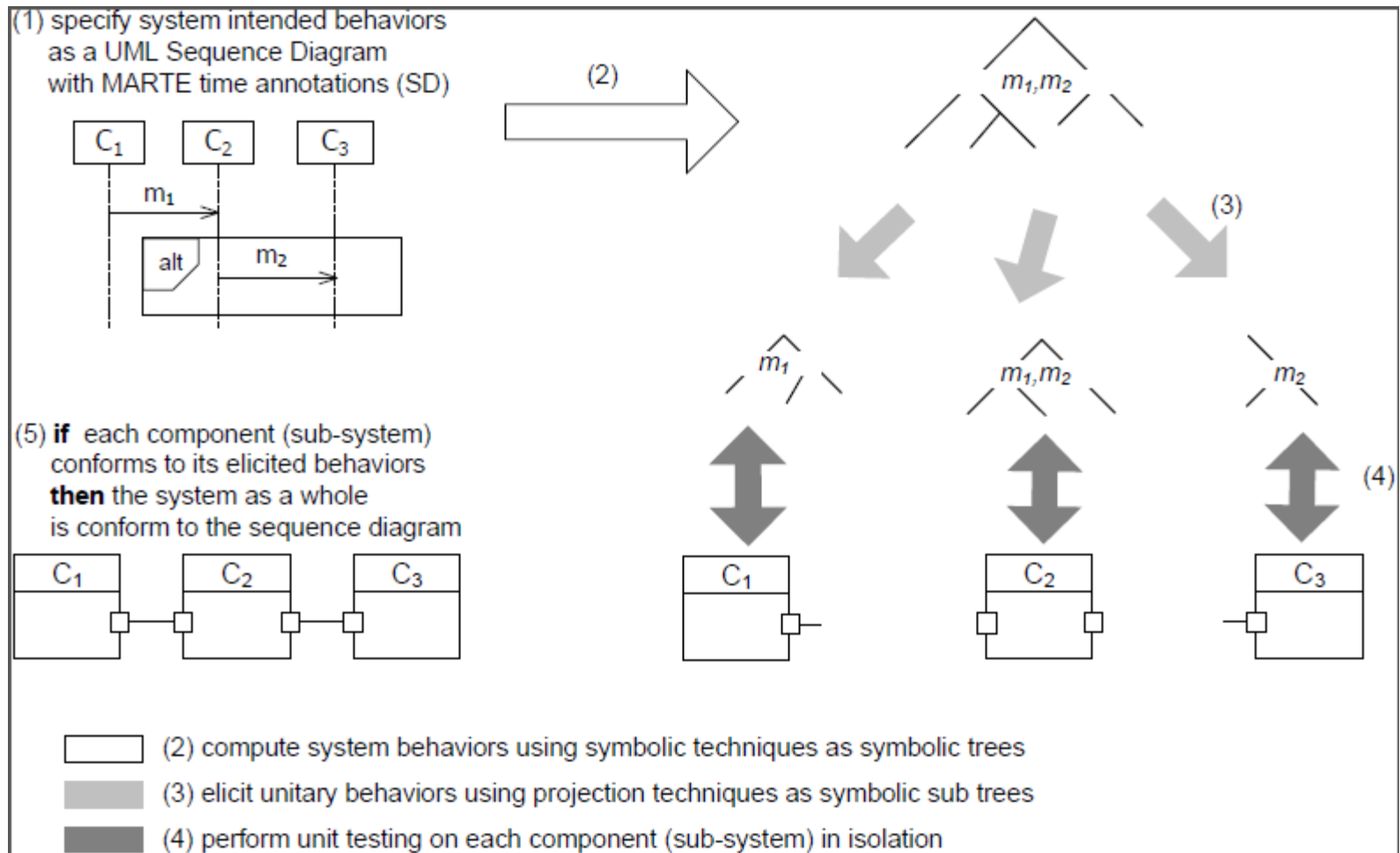- **Purpose : Generate sequence of inputs to stimulate SUT**

- **Scenarios can be described as sequence diagrams in SysML**

- **Currently, another model specially designed for testing is needed, and other tools are needed.**

- **Within our partnership with CEA, we can use 2 tools to insure MBT (Diversity for symbolic execution, and MaTeLo for black-box testing)**

ALL4TEC
ETUDES & CONSEIL

4 – Model Based Testing and SysML - Diversity

(1) specify system intended behaviors as a UML Sequence Diagram with MARTE time annotations (SD)

(2)

(3)

(5) **if** each component (sub-system) conforms to its elicited behaviors **then** the system as a whole is conform to the sequence diagram

(4)

(2) compute system behaviors using symbolic techniques as symbolic trees

(3) elicit unitary behaviors using projection techniques as symbolic sub trees

(4) perform unit testing on each component (sub-system) in isolation

5 - Conclusion

- **Papyrus is an open-source tool, supported by the CEA, and often updated**

- **This methodology of modeling can be used with any tool, and need to be discussed within SysML modelers group**

- **SysML with Papyrus can be a good mean to facilitate MBT or safety analysis because of its good synergy with other existing tools**

- **But its HMI is a little difficult to use at this moment**

Thank you for your attention

Feel free to ask some questions