



VTS-based Specification and Verification of Behavioral Properties of AADL Models

D. Monteverde InTEC, UADE, Argentina

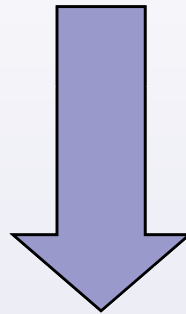
A. Olivero InTEC, UADE, Argentina

S. Yovine VERIMAG-CNRS, France

V. Braberman, FCEyN, UBA, Argentina

Motivation

- Visual Timed Scenarios (VTS)



Specification and Verification of Behavioral
Properties on AADL models



Outline

- Introducing VTS
- Linking VTS with AADL
- Translating VTS into TPN
- Case Studies and Applications
- Conclusions



Visual Timed Scenarios (VTS) [ICSE 2004]

- High-level *graphical notation*
- Allows to describe *event patterns* that represent a set of runs of the system
- Defines a *partial-order* of relevant events
- Express *causality dependencies* and *temporal restrictions* between events of the system
- Used for verification over timed automata models
- Extended to Conditional Scenarios [TSE 2005]

VTS Scenarios – Basic Graphical Notation

event label

●
△
point name
(optional)

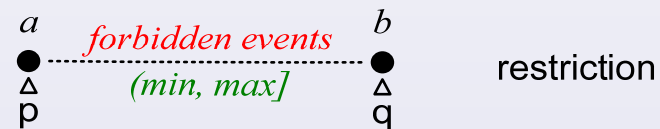
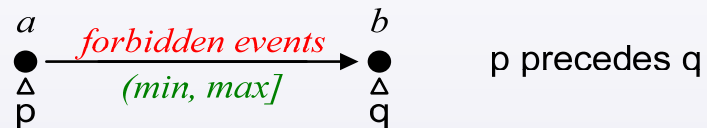
point



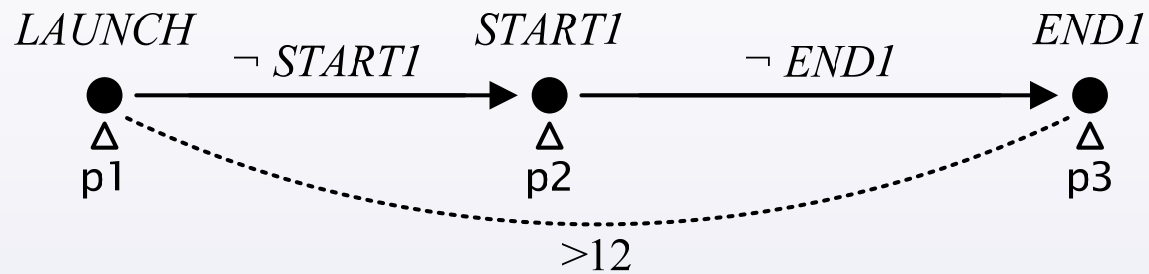
begin



end



VTS Scenarios – Example



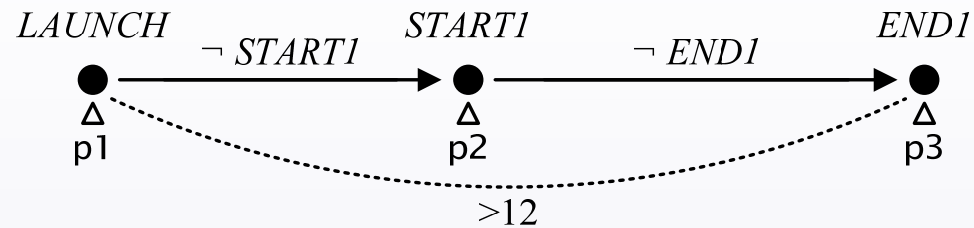


VTS Scenarios – Semantics

- VTS scenarios are interpreted by **existential semantics**
- VTS is used to state questions of the form:

"Is there a potential run of the system that can match this generic scenario?"
- A run satisfies a VTS scenario iff there exists a **match** between them.

VTS Scenarios - Matching



Runs (sequences of events with time stamps)

match?

s1: ... **p2** START1, e1, e3, **p1** LAUNCH, e5, **p3** END1, ...
 (10.1) (12.4) (15.1) (17.3) (19.2) (20.8)

NO, p1 does not precede p2

s2: ... **p1** LAUNCH, e1, **p2** START1, e2, **p3** END1, END1, ...
 (1.4) (2.2) (5.0) (10.2) (11.4) (14.3)

NO, time ≤ 12 between p1 and p3

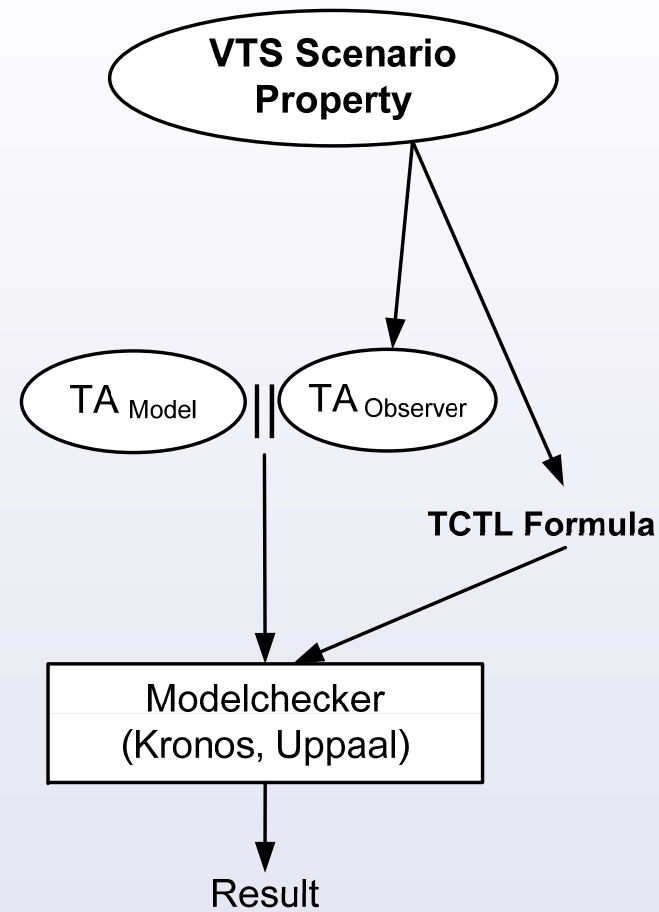
s3: ... **p1** LAUNCH, e1, **p2** START1, e2, END1, **p3** END1, ...
 (1.4) (2.2) (5.0) (10.2) (11.4) (14.3)

NO, END1 occurs between p2 and p3

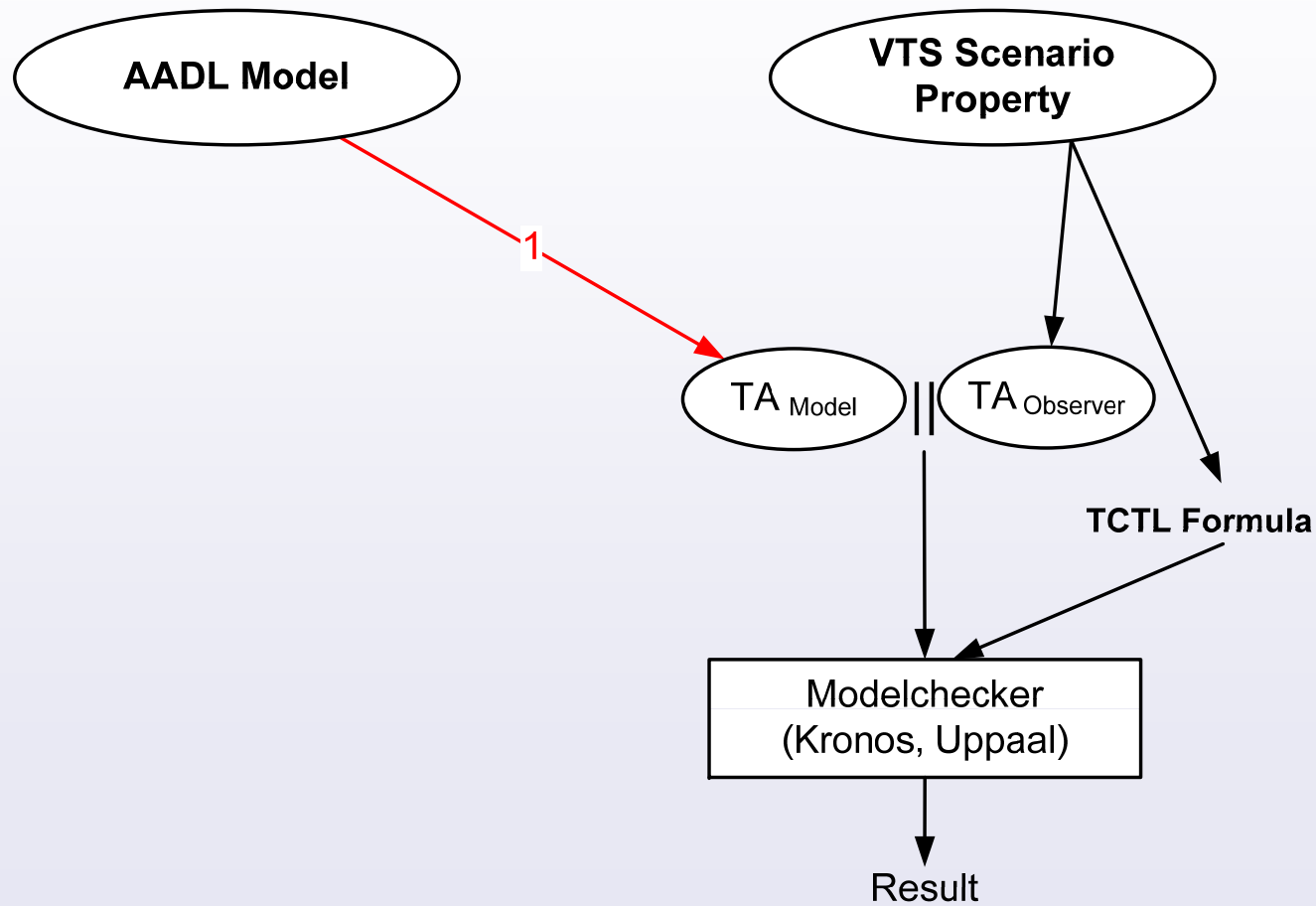
s4: ... **p1** LAUNCH, e1, **p2** START1, e5, **p3** END1, ...
 (1.1) (2.1) (5.2) (10.1) (14.5)

YES

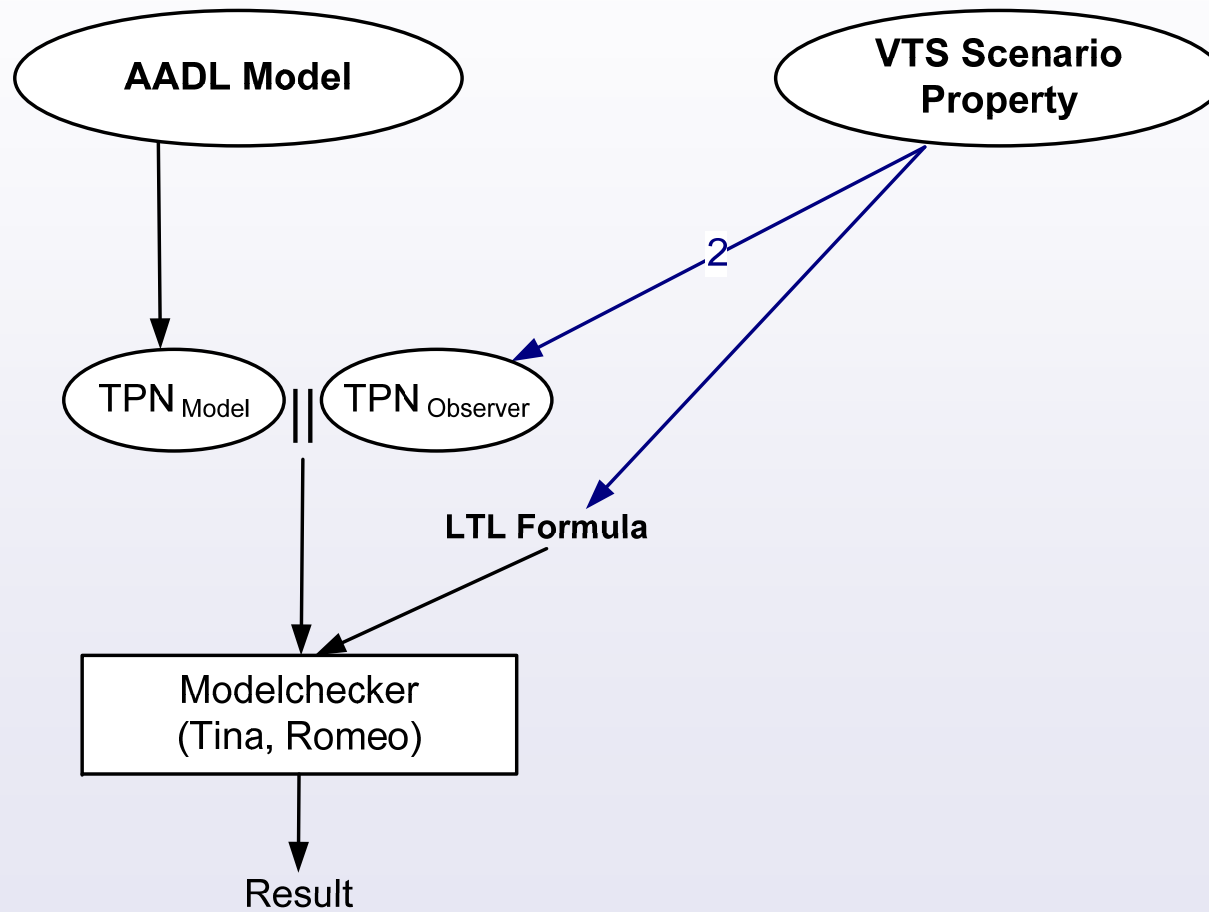
Model-checking VTS via Timed Automata



Linking VTS with AADL

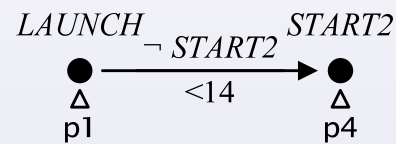


Linking VTS with AADL



Translating VTS into TPN

- Procedure
 - **Split** the scenario into different parts,
 - **Translate** each part into a TPN component,
 - **Combine** all TPN components.

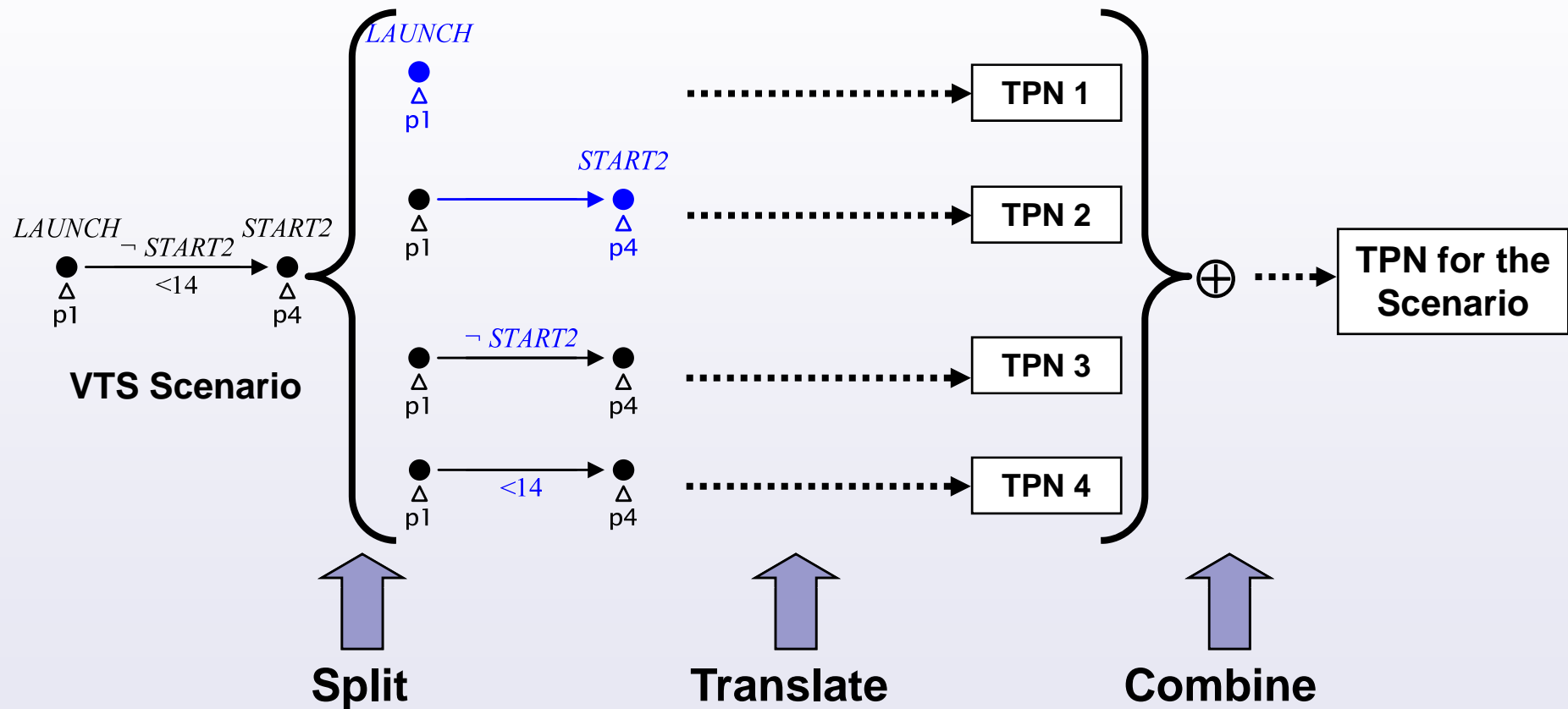


VTS Scenario

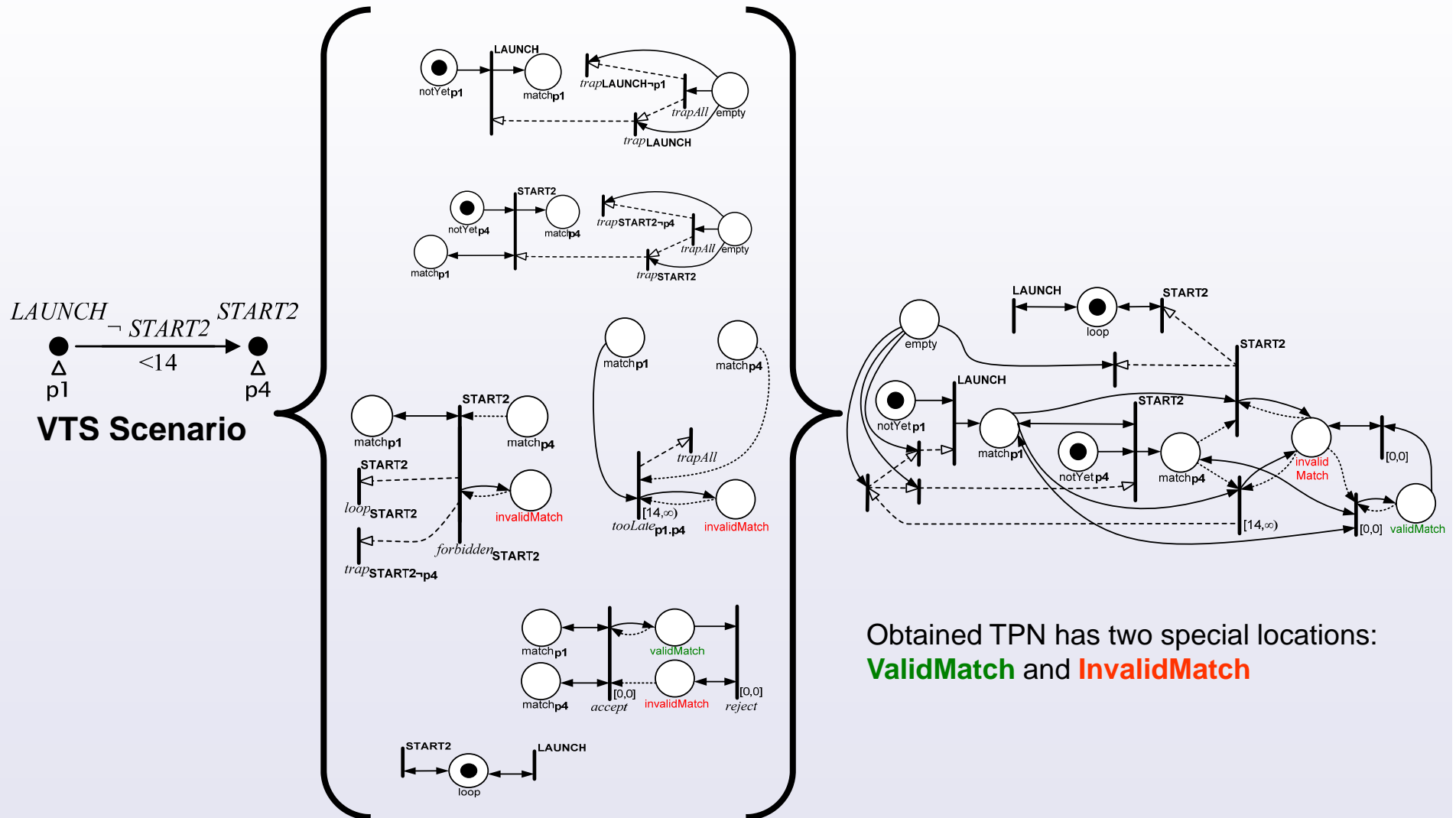


TPN for the Scenario

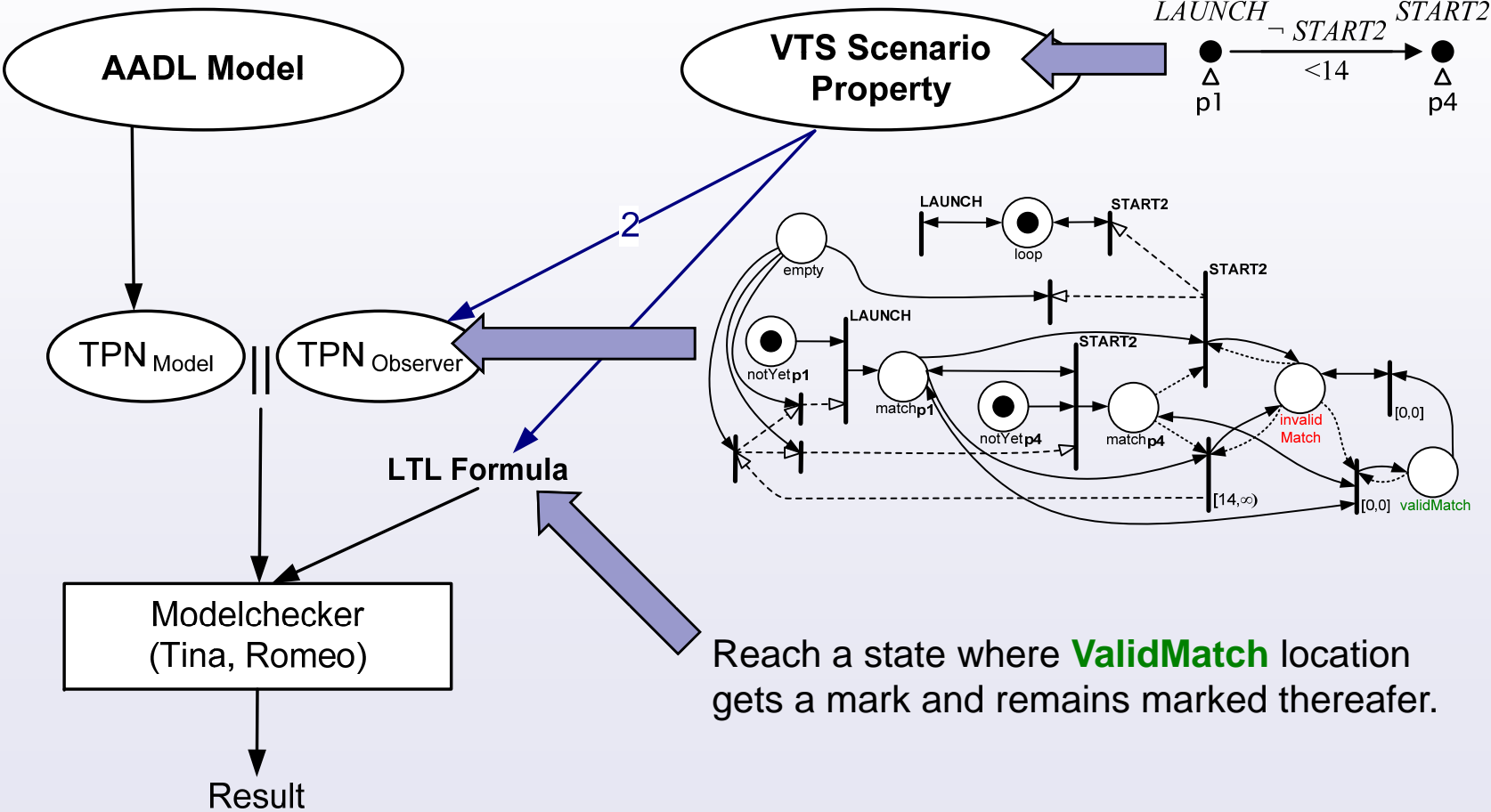
Translating VTS into TPN



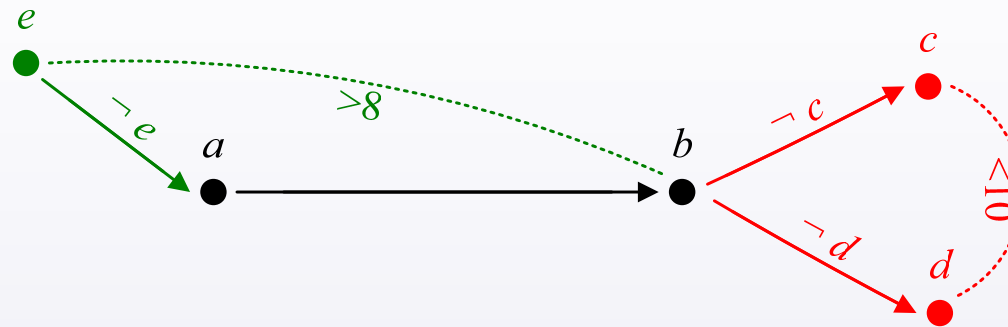
Translating VTS into TPN



Model-checking VTS via TPN

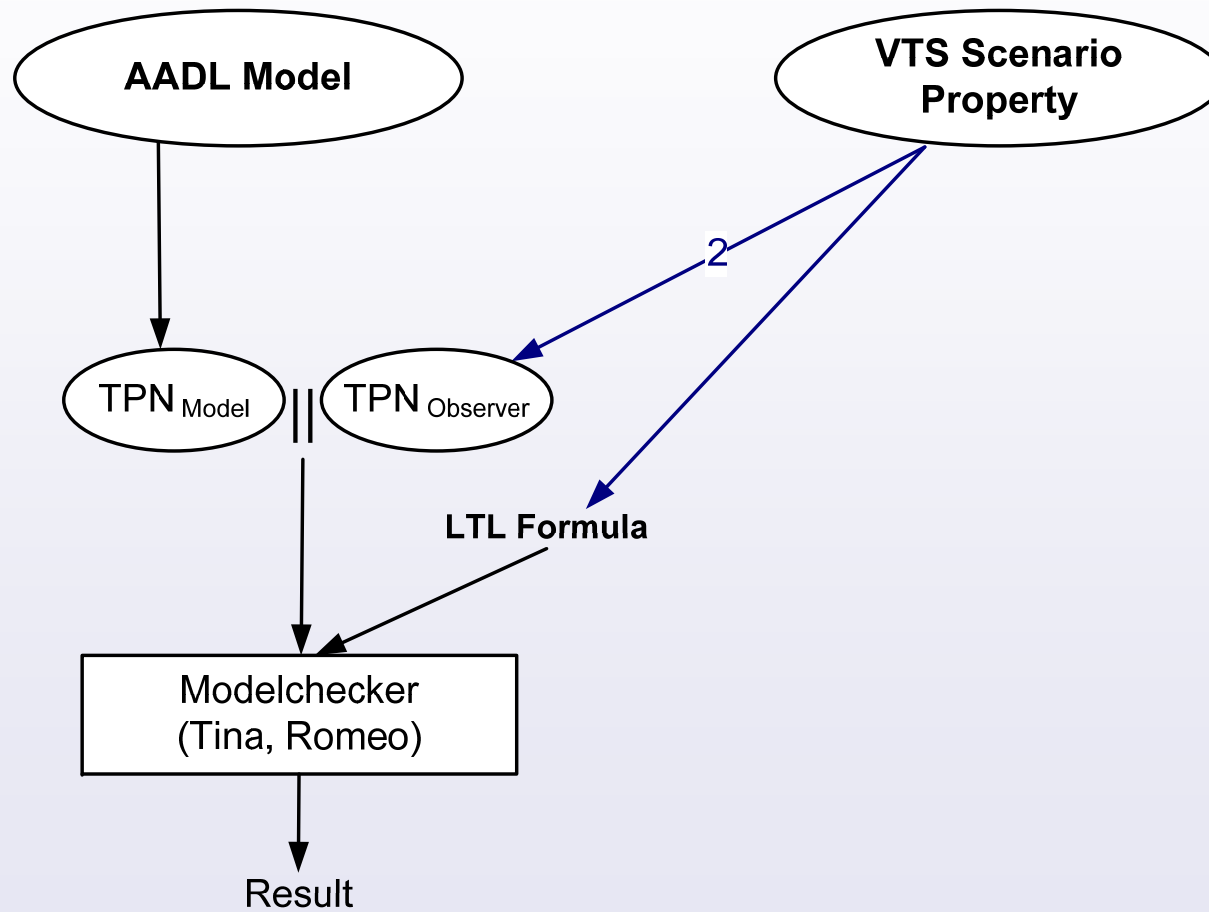


VTS Conditional Scenarios – General Idea



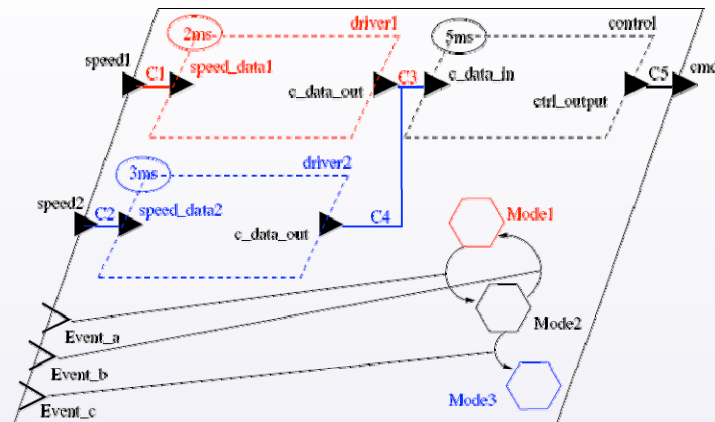
- **FOR ALL** run
 - if run \models **scenario** then (run \models **scenario** or run \models **scenario**)
- For verifying a CS we generate the set of all ESs that violated it.

Linking VTS with AADL

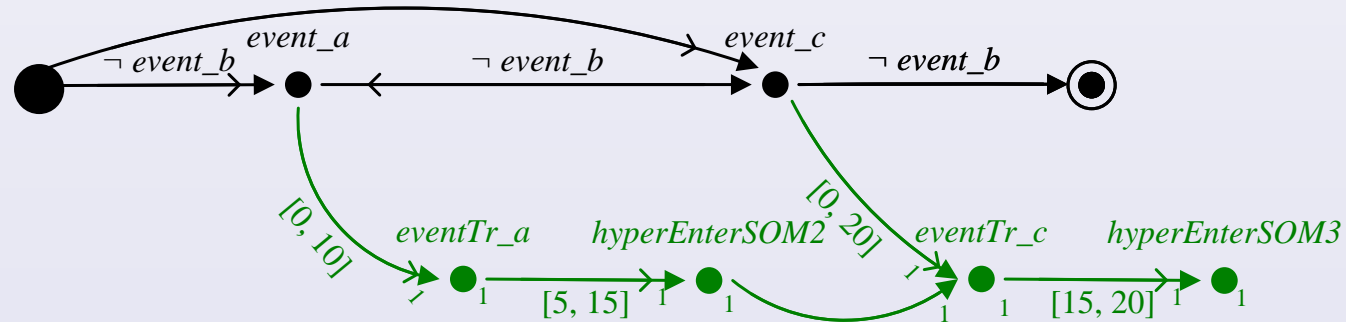


AADL Mode Change Protocol

[Source: D. Bertrand et al, "A Study of the AADL Mode Change..." 2008]



AADL Specification



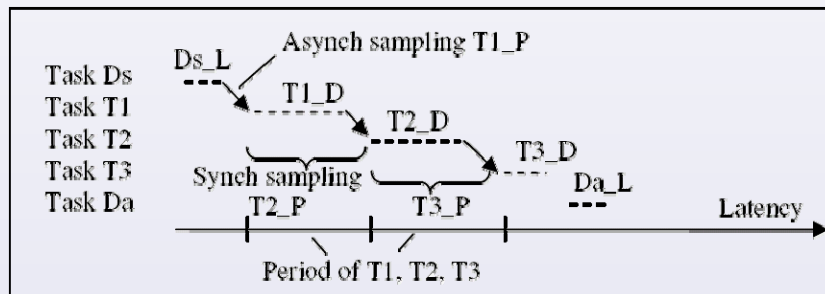
Property (CS)

AADL Flows specification

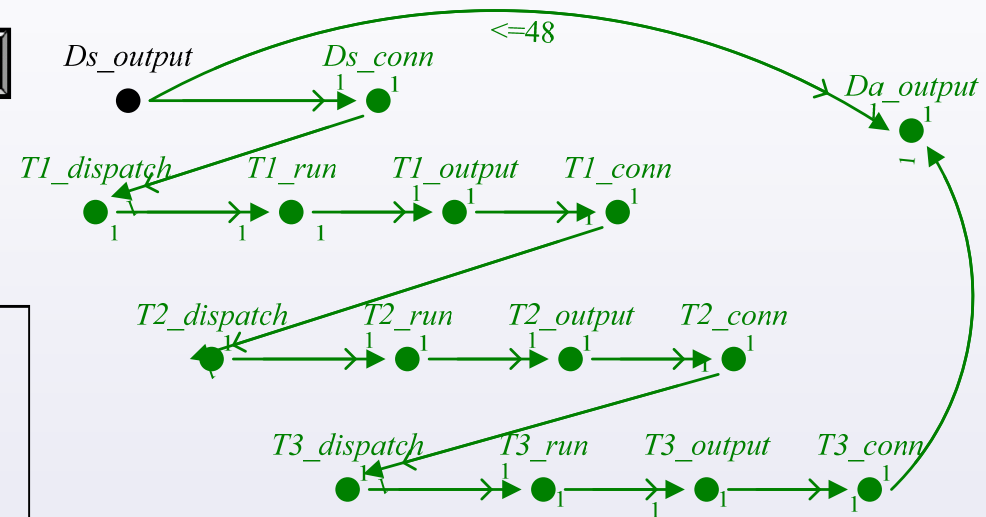
[Source: P. Feiler et al, "Flow latency analysis with the AADL" 2007]



Flow from a Sensor through 3 Task to an Actuator



Data-Driven Flow Processing Chain



Property (CS): whenever sensor produces an output, then:

- (1) the flow is realized, and
- (2) latency ≤ 48 .



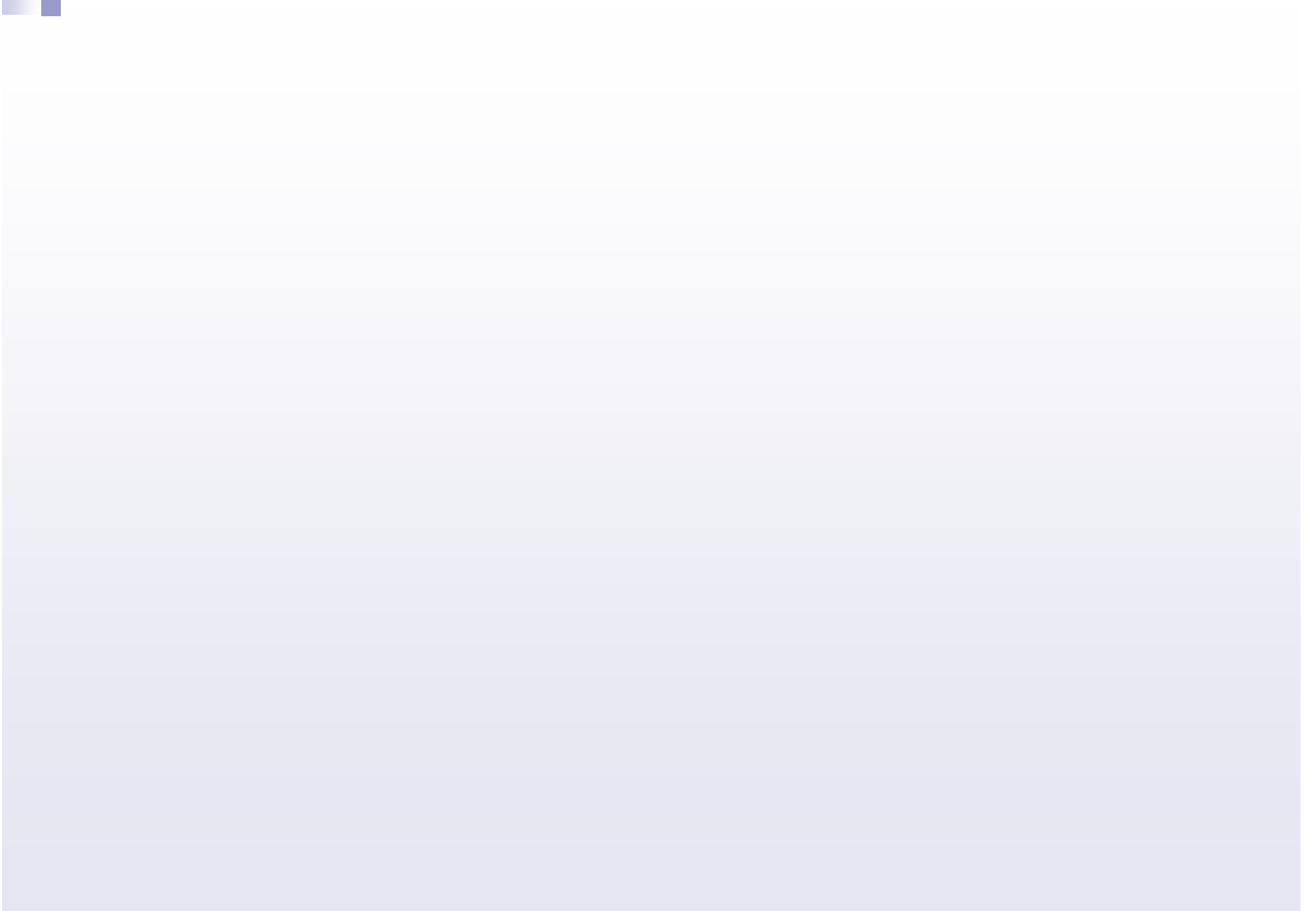
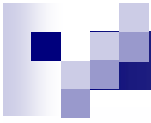
Conclusions & Future work

■ Conclusions

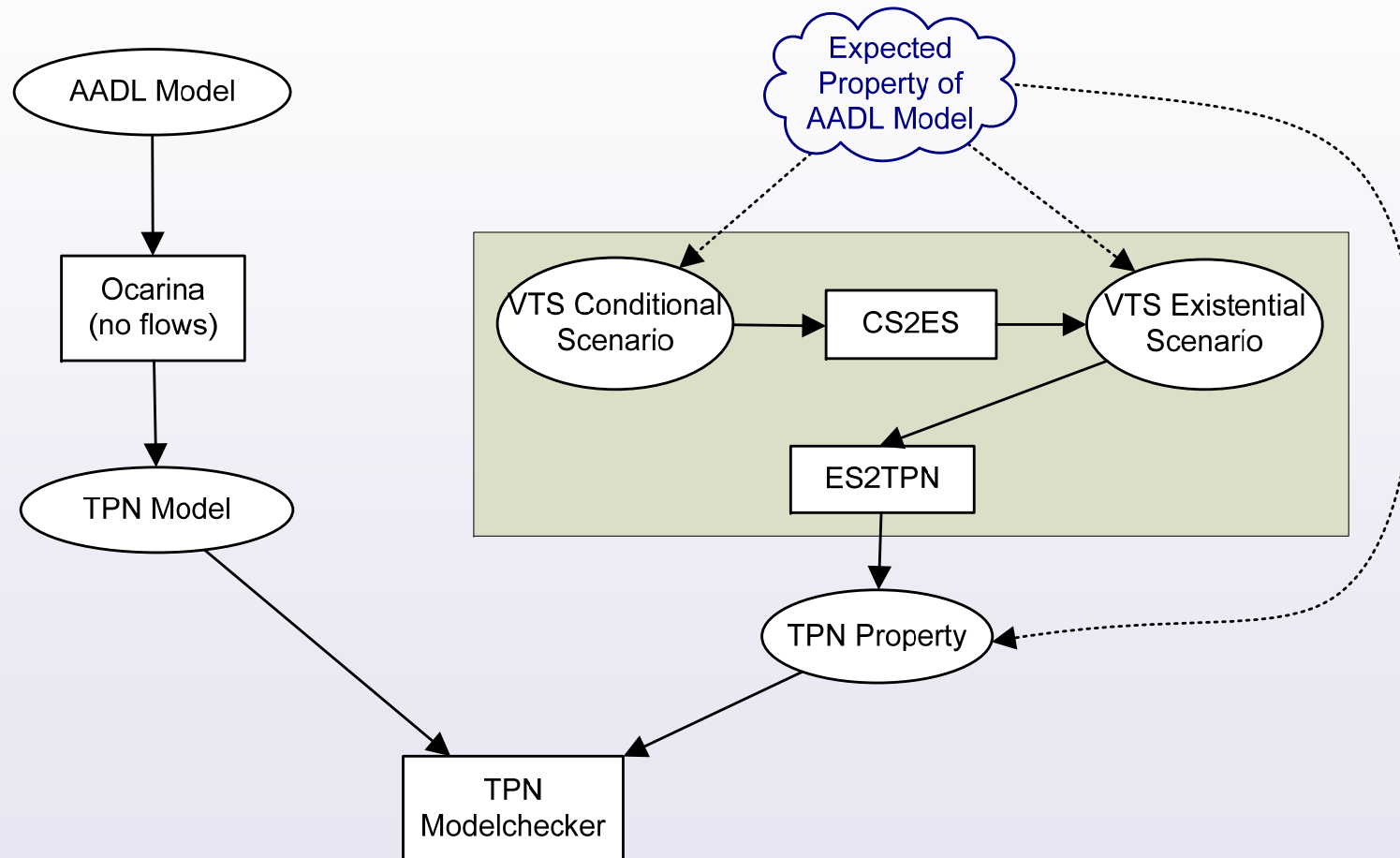
- We implemented a tool that automatically translates VTS into TPN. This translation also comprises VTS CSs.
- VTS Scenarios appeared to be adequate to express properties of AADL models.

■ Future work

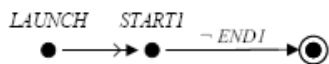
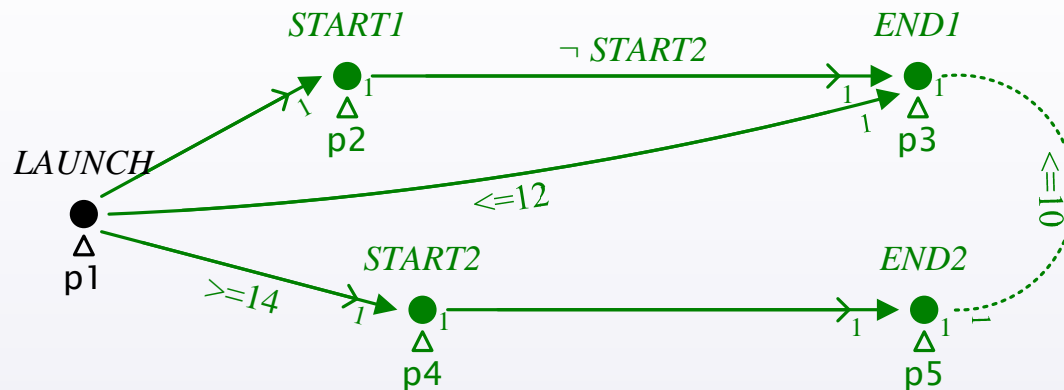
- Investigate the possibility of taking this translation to intermediate modeling language FIACRE.
- Explore the connection with VTS and AADL Flows to express more complex flows.



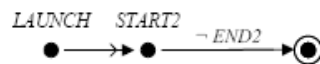
Tool chain integrating AADL and VTS



Example CS 2Jobs – Antisceanarios (ESs)



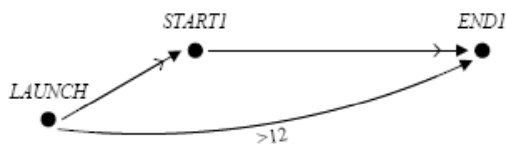
(a) Job_1 starts, but does not terminate (1)



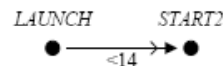
(b) Job_2 starts, but does not terminate (2)



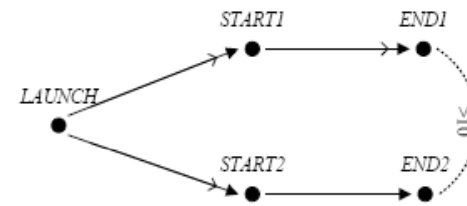
(c) Job_2 starts while Job_1 is in execution (3)



(d) Job_1 terminates after 12. (4)



(e) Job_2 starts before 14. (5)



(f) More than 10 passes between jobs ends. (6)



Modelchecking formula

- The scenario property is not satisfy iif:
 - LTL formula:
 - $\square (\text{ValidMatch} \Rightarrow \langle \rangle (-\text{div} \vee \text{InvalidMatch}))$
 - In words:
 - For every run, when run reaches a **ValidMatch**,
then (is Not Time Divergence or reaches **InvalidMatch**)

