Comparing Model Checking and Model-based Simulation

Davide Basile, **Franco Mazzanti**

(RSSRAIL 2025)

Istituto di Scienza e Tecnologie dell'Informazione "A. Faedo"
Consiglio Nazionale delle Ricerche

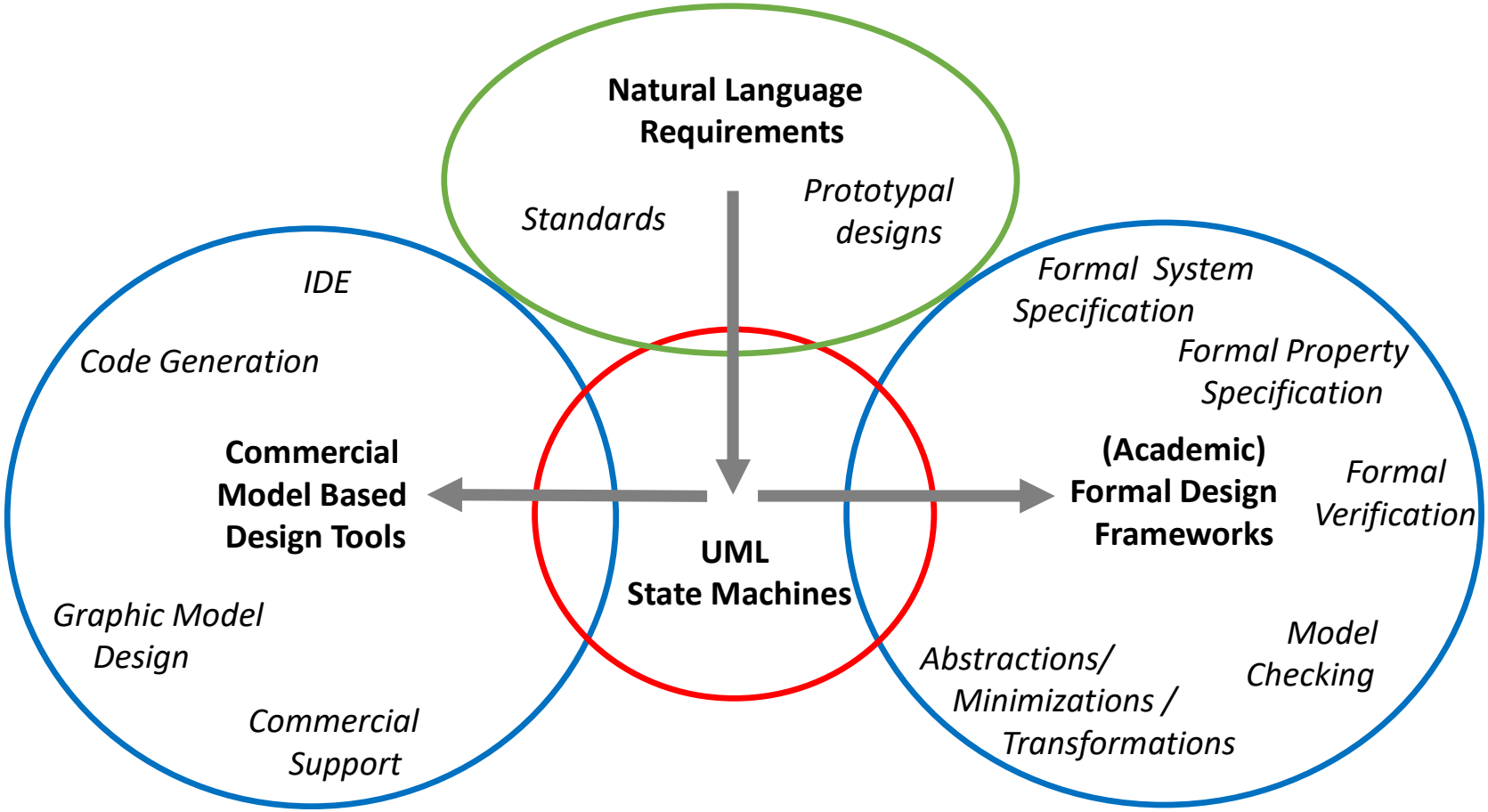
Formal Methods and Tools Lab

Finanziato dall'Unione europea NextGenerationEU

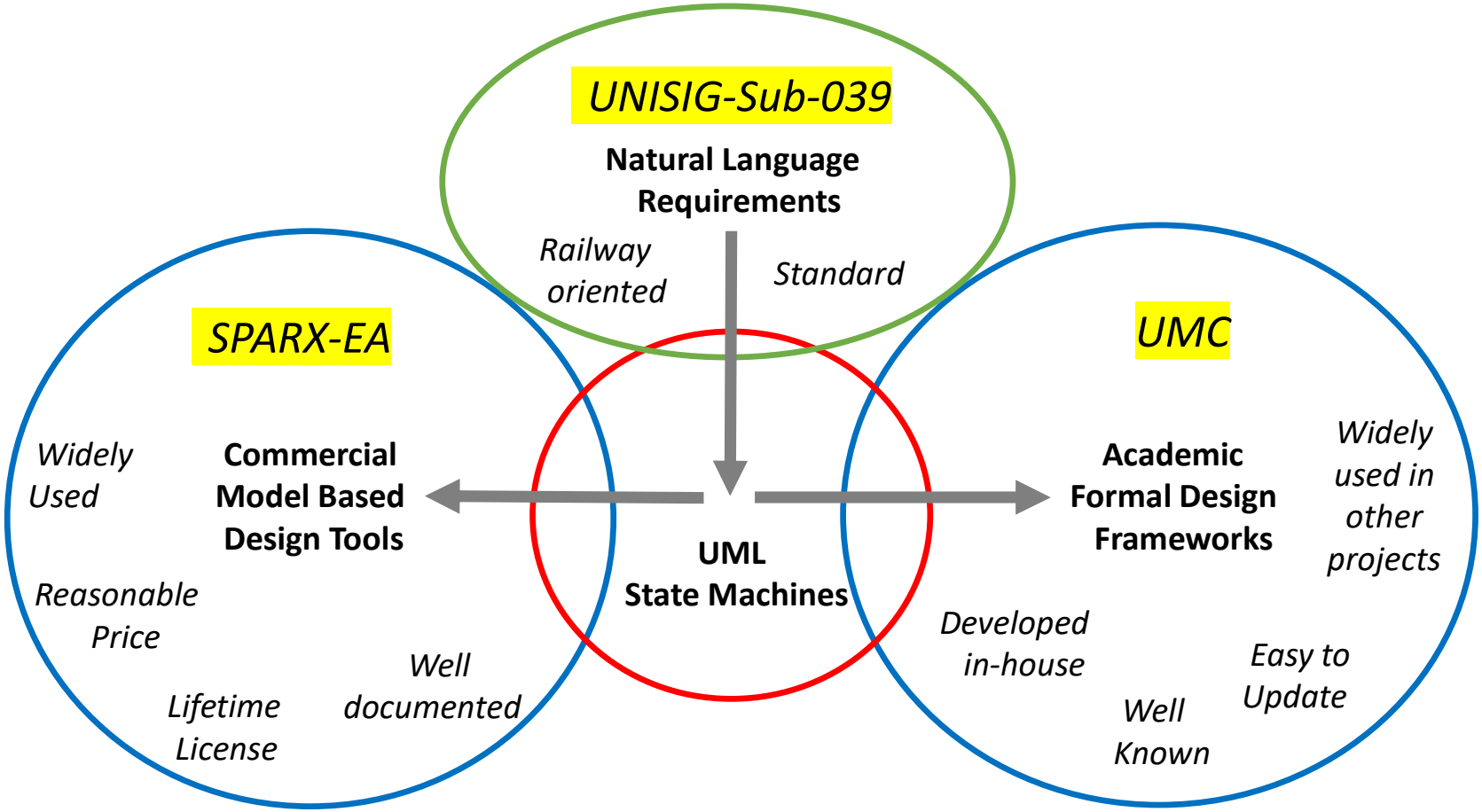
Ministero dell'Università e della Ricerca

Italiadomani

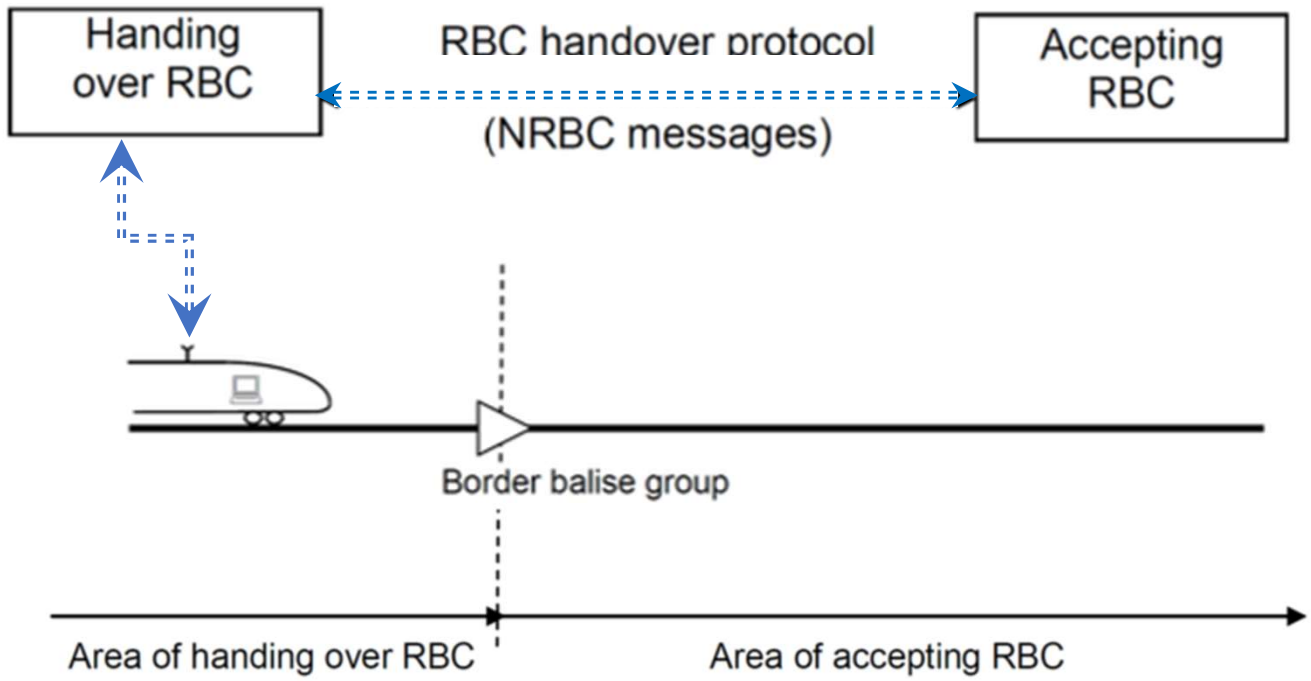
… The Experiment … (Idea)



… The Experiment … (Context)

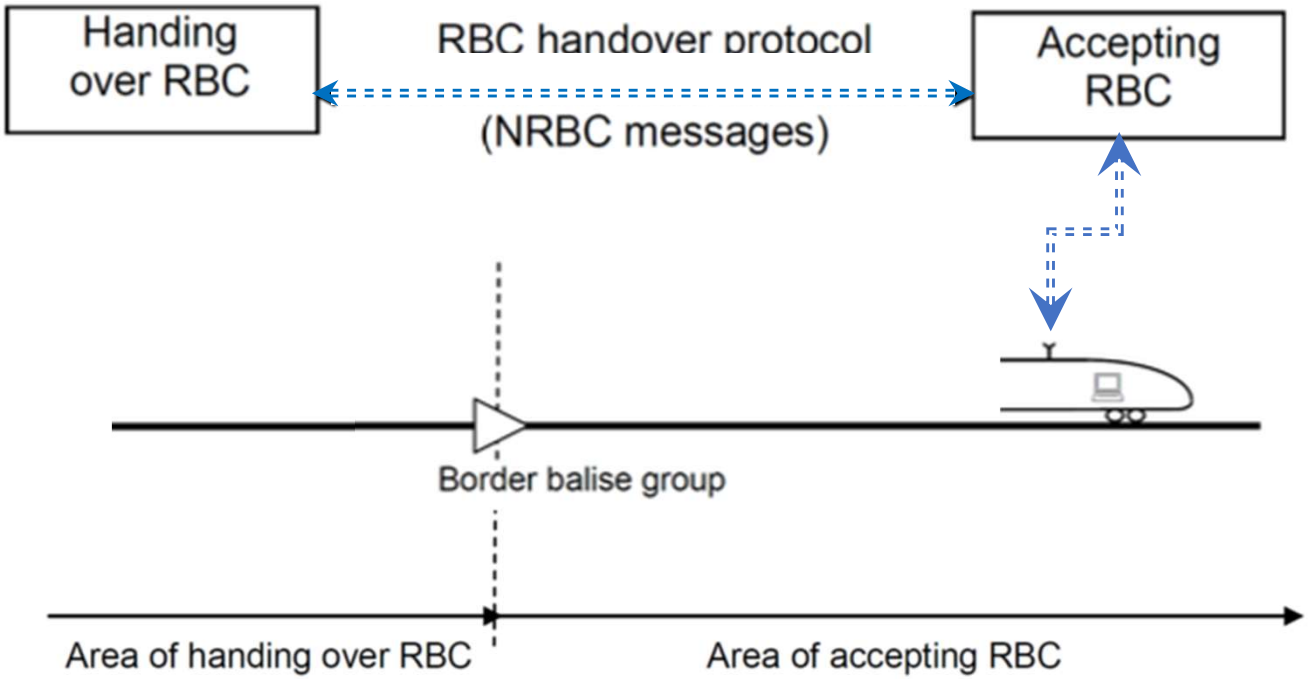


... The Case Study ... (Overview)



RBC: Radio Block Centre

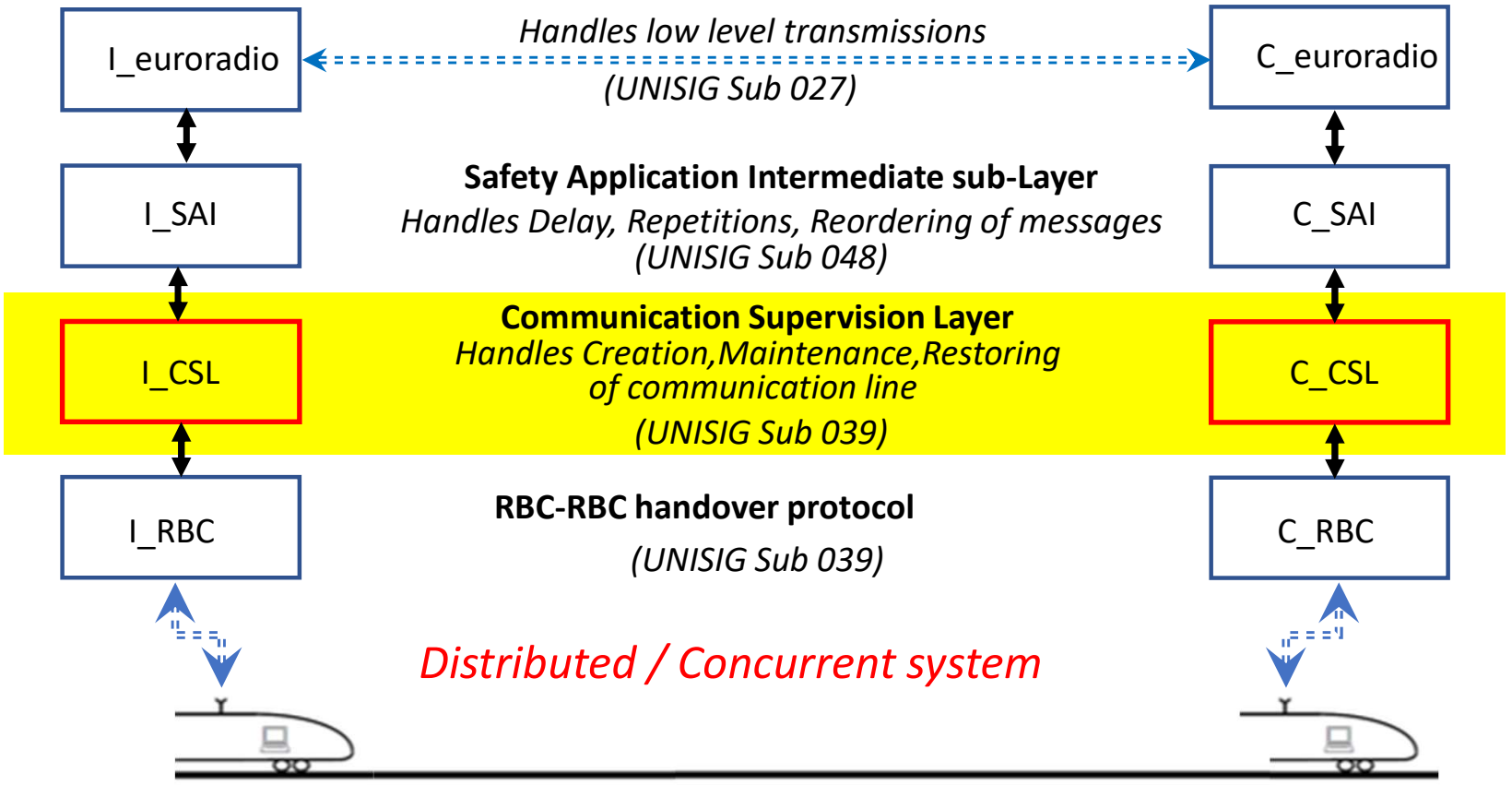
... The Case Study ... (Context)



RBC: Radio Block Centre

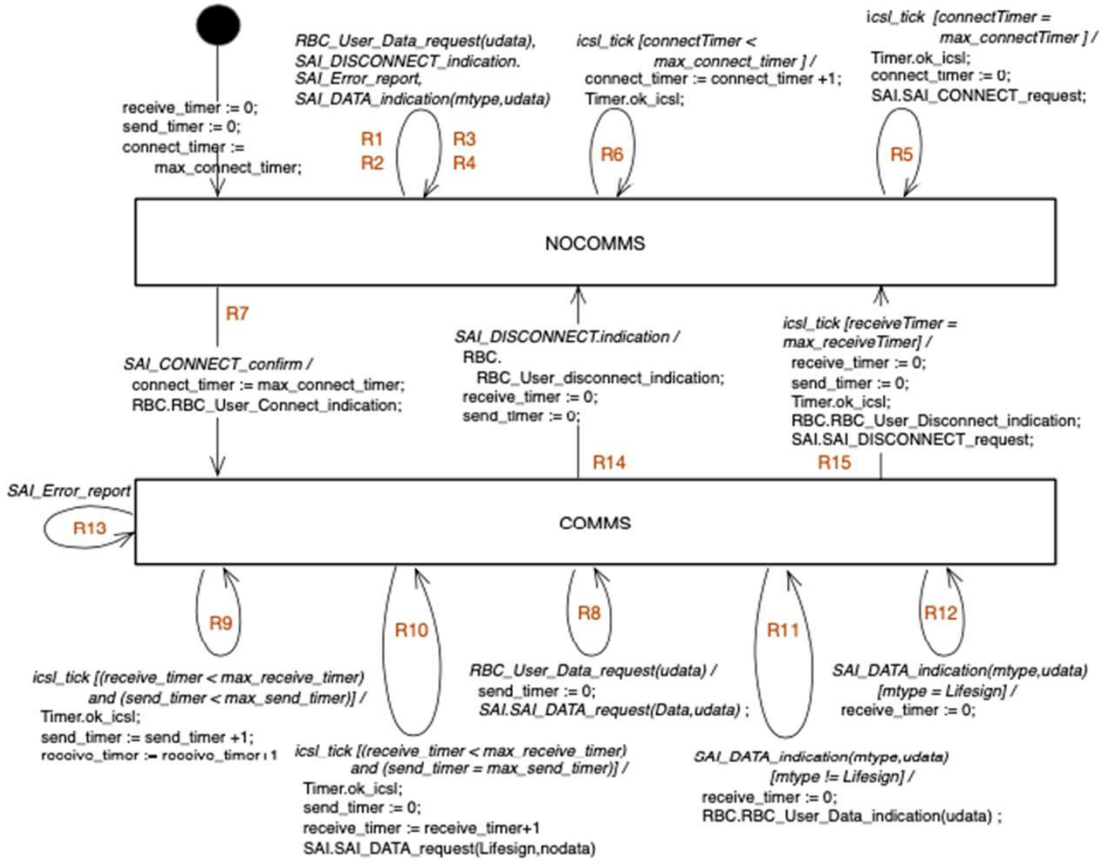
I_RBC

... The RBC – RBC Communications ...

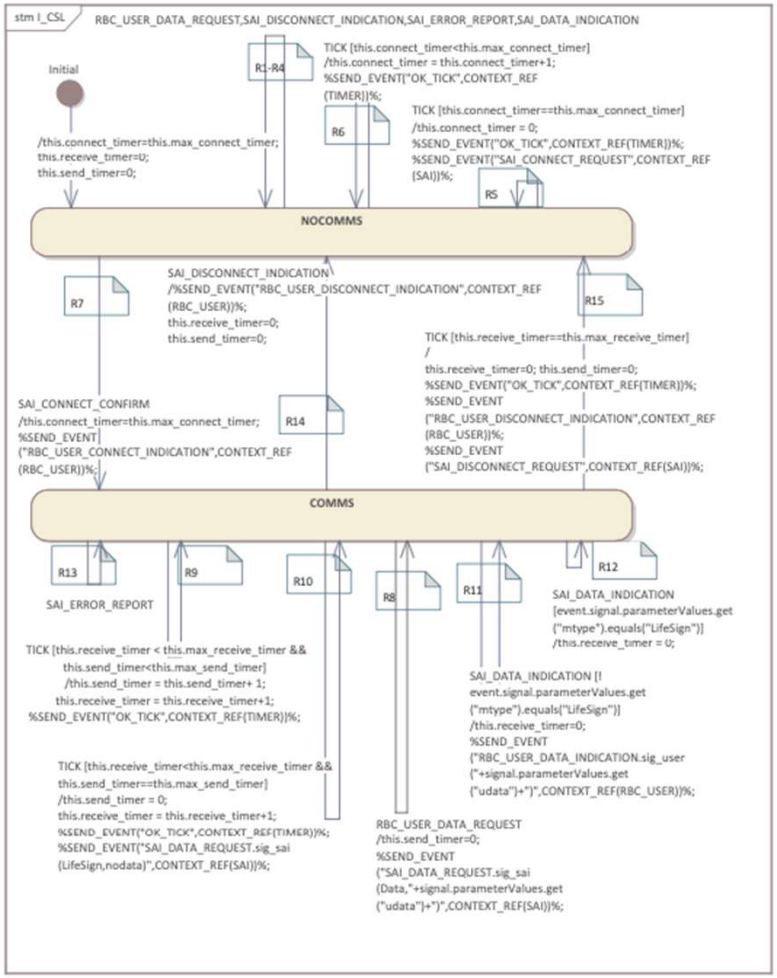


... The Communications Supervision Layer ...

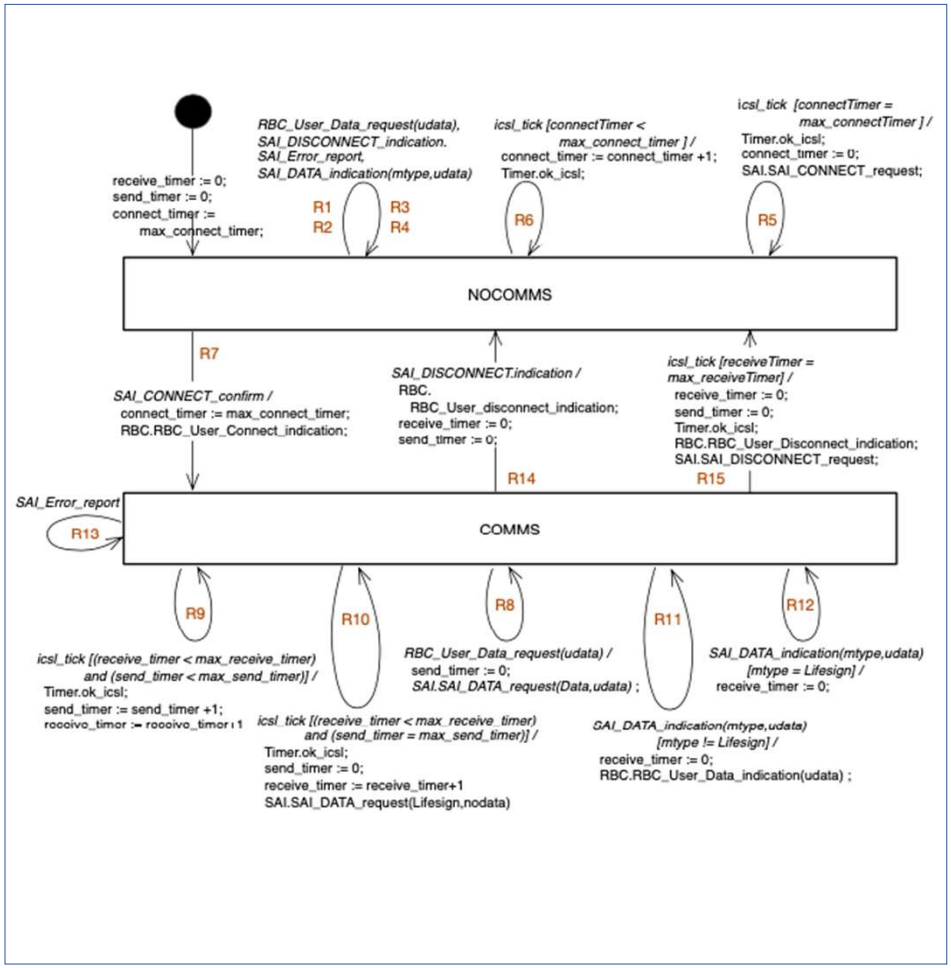
- Requests creation of a connection
- Sends life-signes to keep it alive
- In case of silence restarts the line
- Forwards RBC messages



… The Communications Supervision Layer …



SPARX-EA



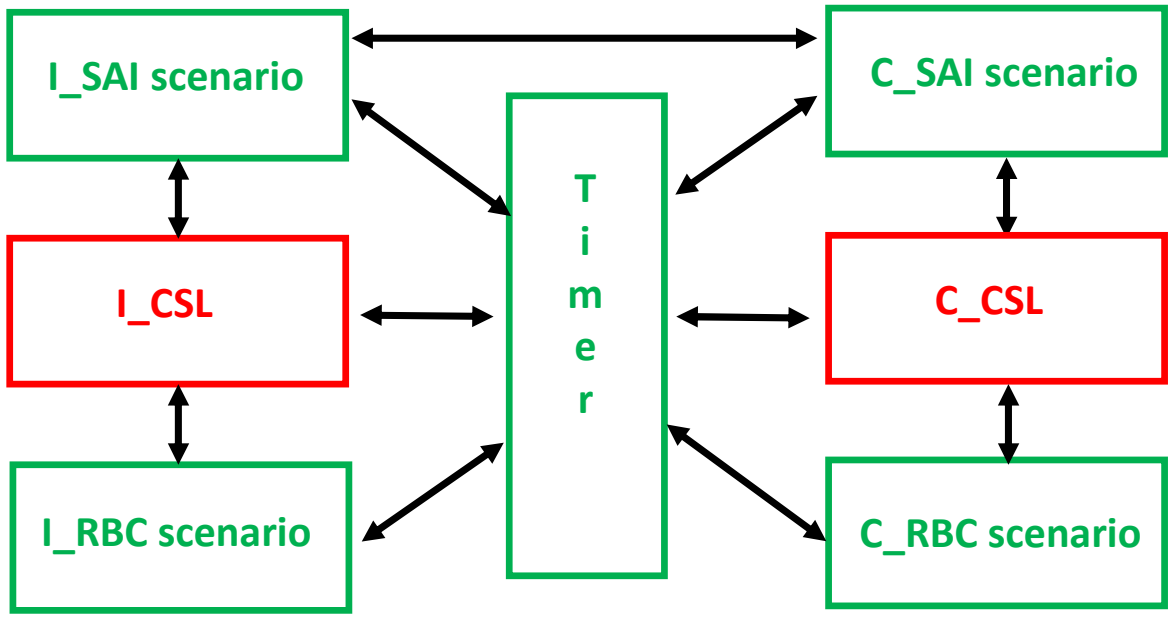
UMC

... Modeling System Behavior ... (UMC)

UMC requires the system to be a complete closed system

The actual **CSL code** ..
... must be composed with **scenario based** components modelling the environment

All components evolve in **parallel**, potentially in an highly **nondeterministic** way



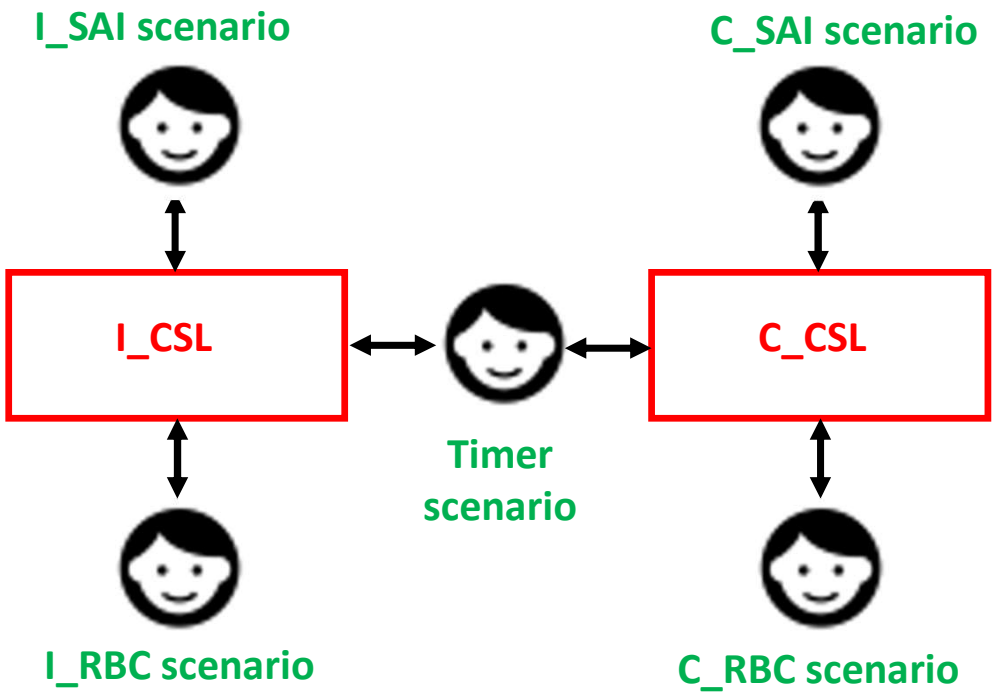
The structure of a UMC scenario involving the full system

... Modeling System Behavior ... (SPARX-EA)

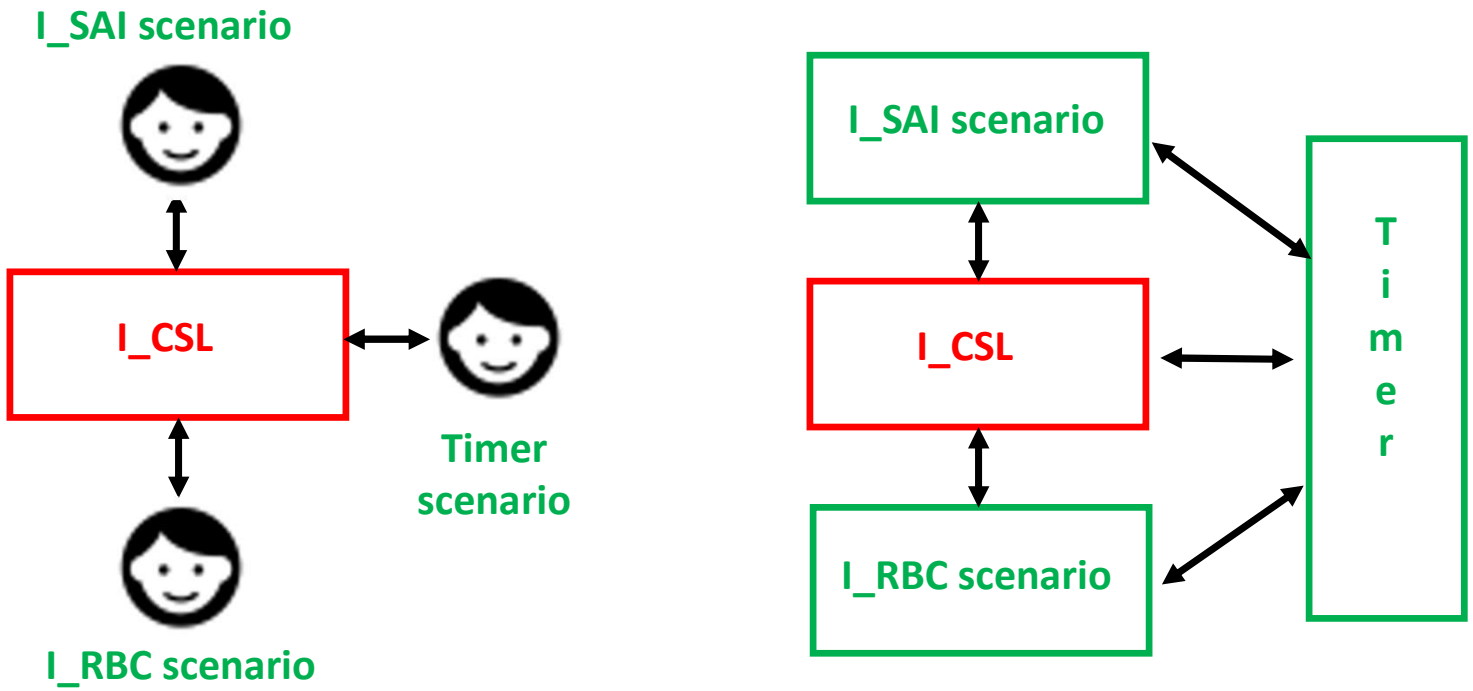
SPARX execution/simulation engine is **deterministic**

if we compose the CSL code with all the scenario based missing components, the result would be a single (deterministic **sequential** system)

The SPARX **user** must take the role of the environment



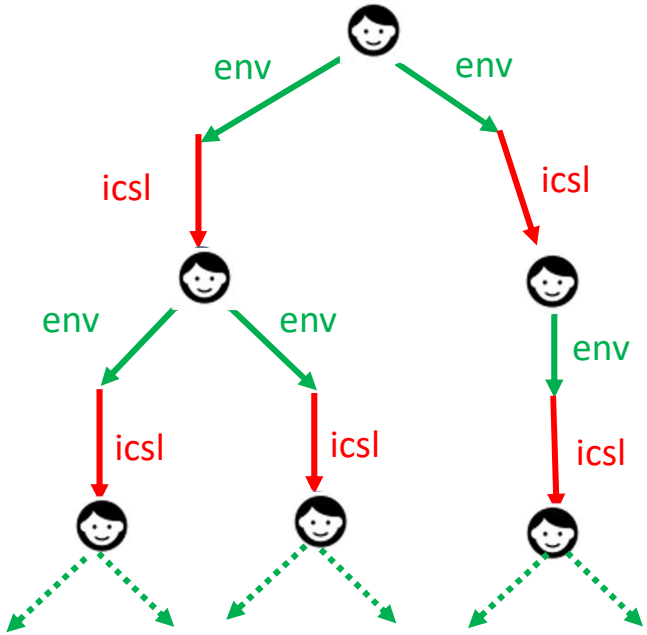
... A simple scenario...



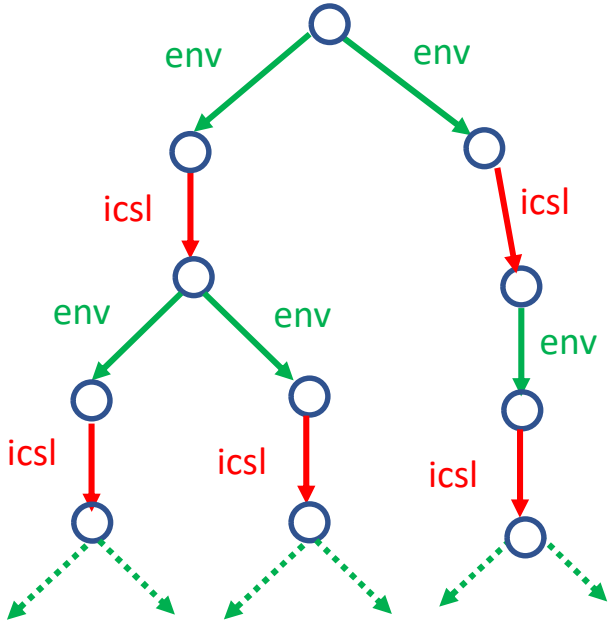
SPARX-EA

UMC

… Evolution Steps …

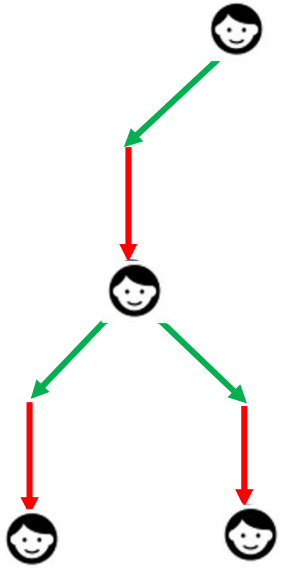


SPARX-EA



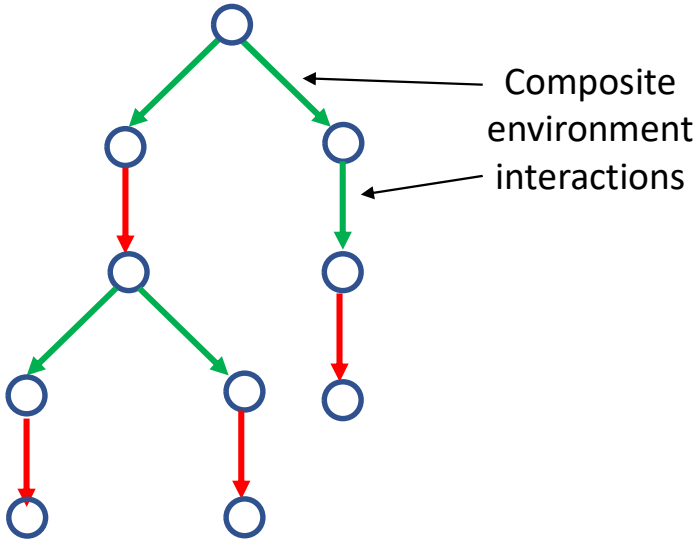
UMC

… Comparison … (1)



SPARX-EA:
As soon as the User provides an input, the system consumes it

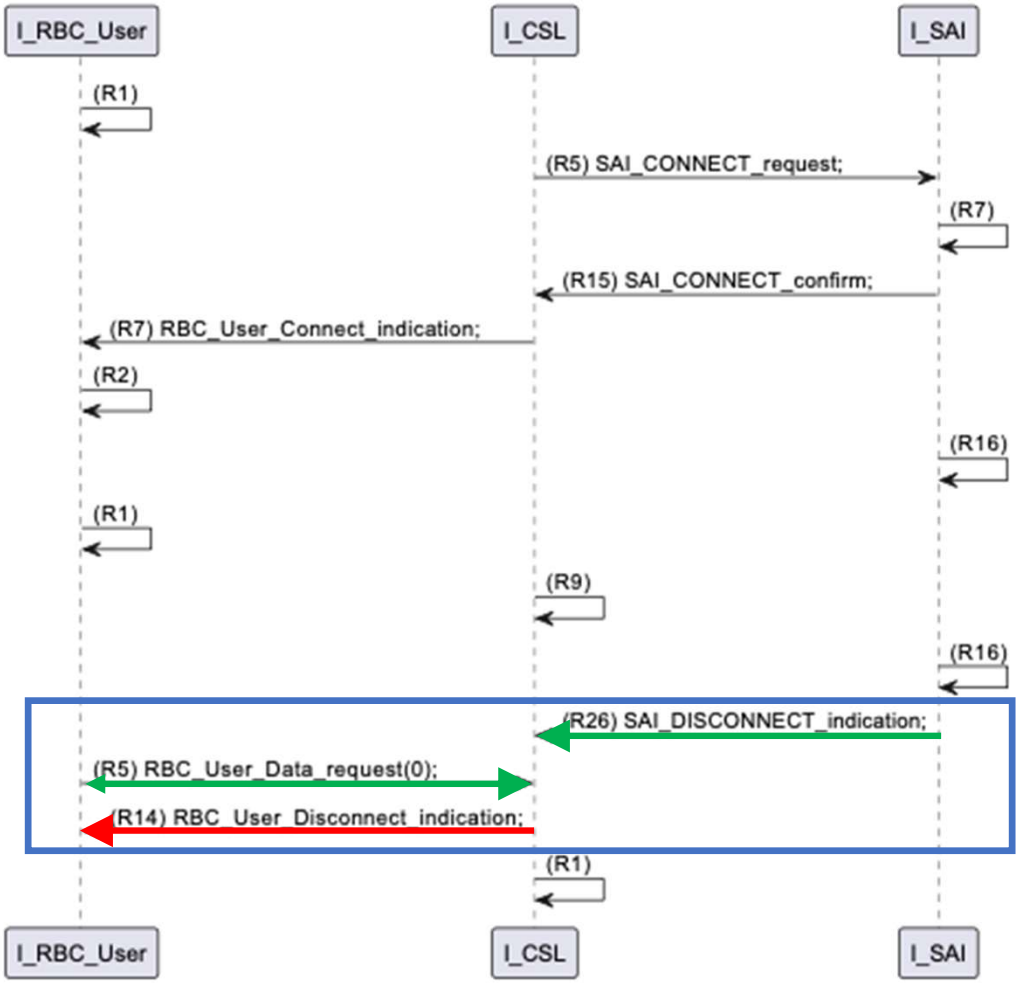
Composite environment interactions cannot be simulated



There are execution traces in UMC not replicable in SPARX-EA

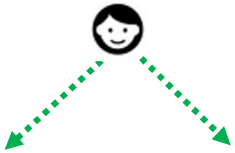
SPARX-EA

UMC


I_RBC_User
I_CSL
I_SAI
(R1)
(R5) SAI_CONNECT_request;
(R7)
(R15) SAI_CONNECT_confirm;
(R7) RBC_User_Connect_indication;
(R2)
(R16)
(R1)
(R9)
(R16)
(R26) SAI_DISCONNECT_indication;
(R5) RBC_User_Data_request(0);
(R14) RBC_User_Disconnect_indication;
(R1)
I_RBC_User
I_CSL
I_SAI

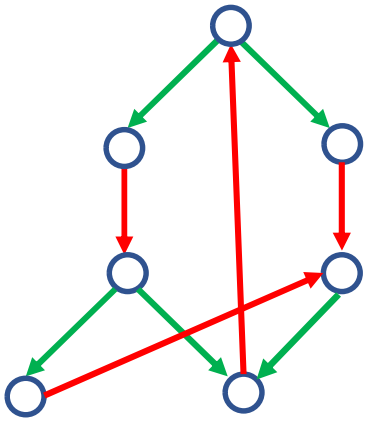
... Comparison ... (2)

Succesfull univerasl properties cannot be fully reproduced in SPARX-EA



It is always true that, after a first connect indication, IRBC cannot receive a second connect indication, unless it receives a disconnect indication in the meanwhile

AG [IRBC User Connect indication] (
A[{not IRBC User Connect indication} W
{IRBC User Disconnect indication}])

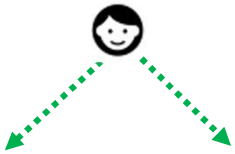


Failing reachability properties cannot be reproduced in SPARX-EA

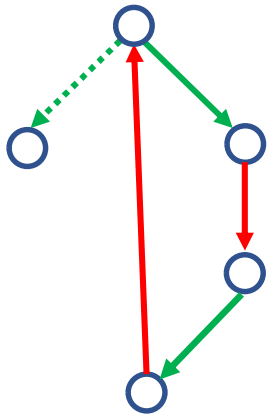
SPARX-EA

UMC

... Comparison ... (3)



Property:
There is an execution sequence such that, after the having established a connection, that connection is never lost



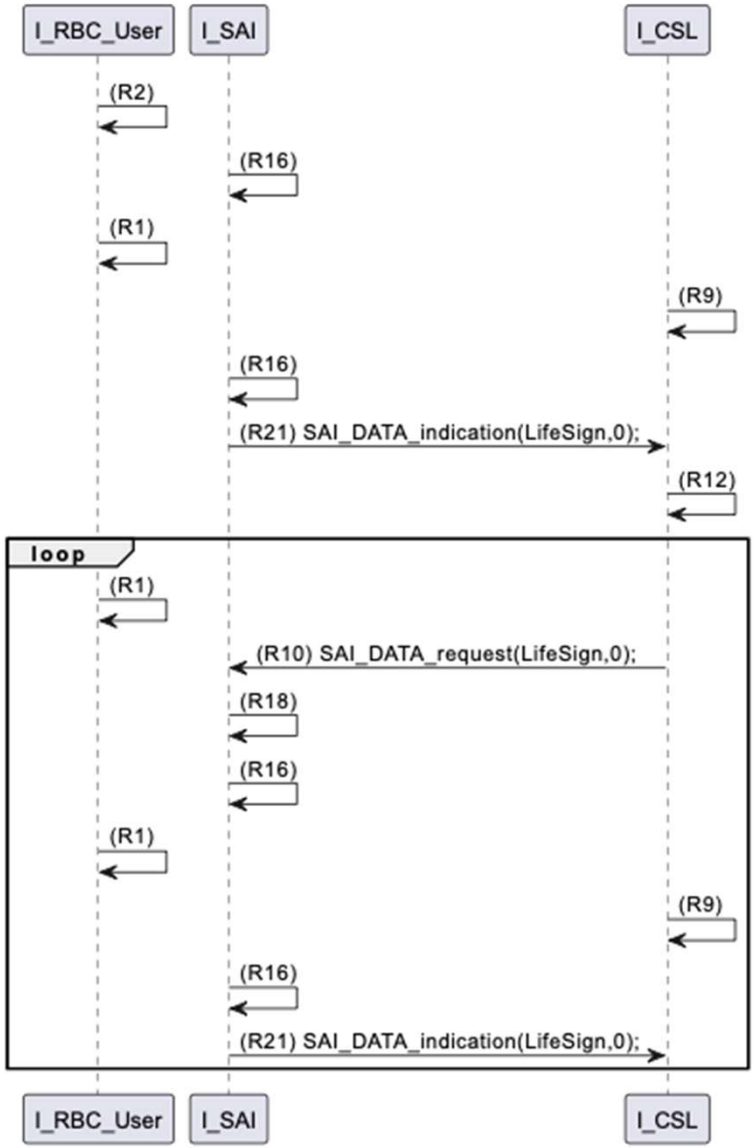
Infinite traces witnessing a property cannot be reproduced in SPARX-EA

SPARX-EA

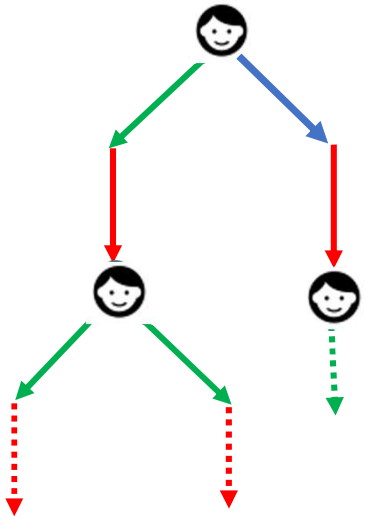
UMC

There is an execution sequence such that,
after the having established a connection,
that connection is never lost

EF {IRBC User Connect indication} (
EG {not IRBC User Disconnect indication})



… still on Environments …

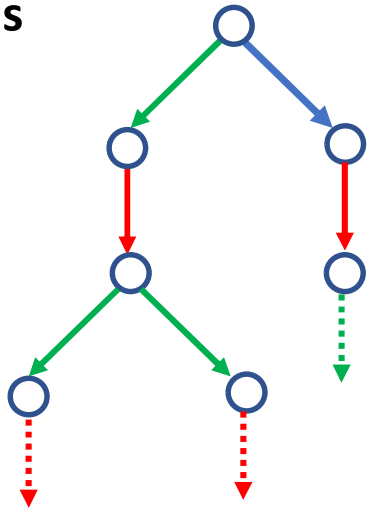


Property:

Something bad eventually happens

In SPARX-EA the user has the freedom to model also apparently meaningless environment interactions

In UMC the designer needs to explicilty devise a (meaningless) scenario for modeling the same kind of behaviour

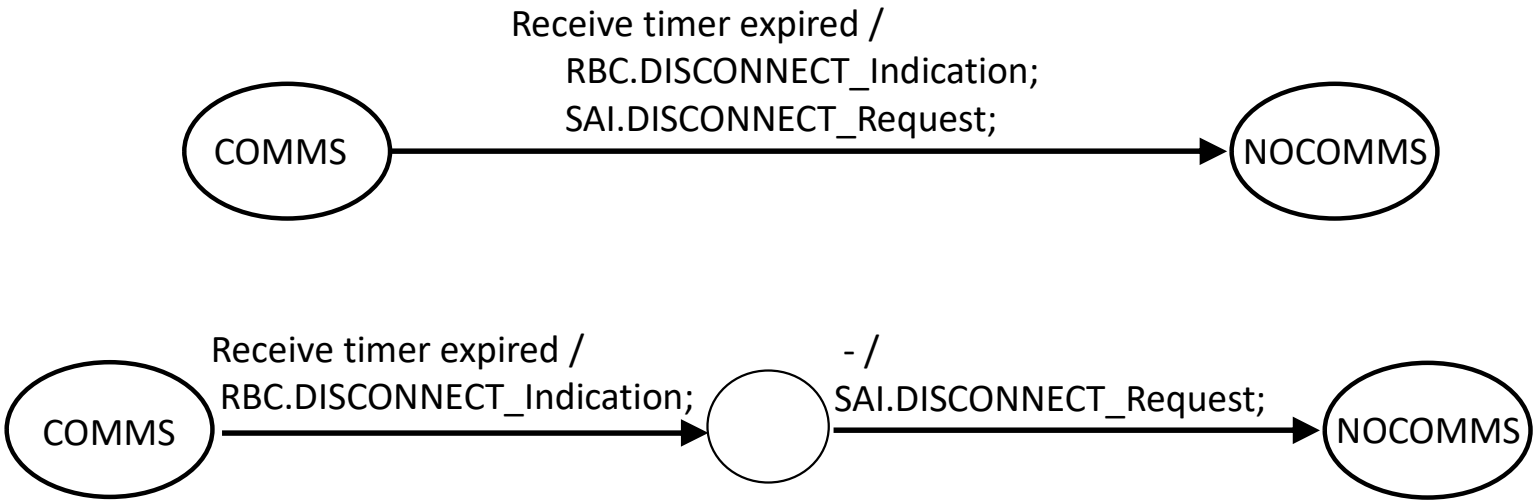


False positives more difficult to control in SPARX-EA

SPARX-EA **UMC**

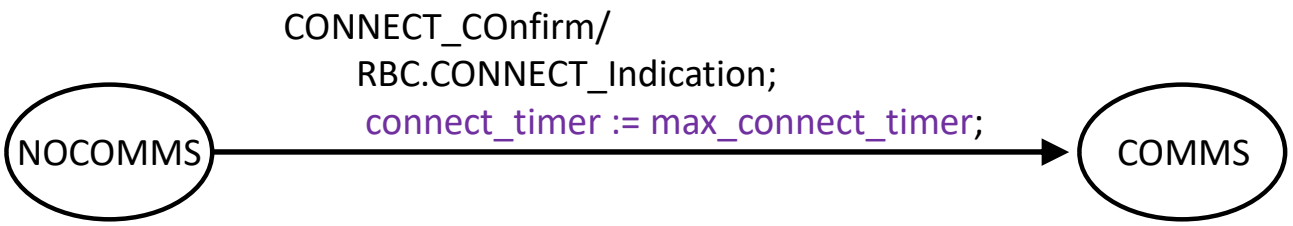
... differences in UML designs... (atomicity)

R15: “when ICSL is in the COMMS state, and the receive timer expires, a disconnect_indication is sent to the user, a disconnect_request is sent to the SAI, and ICSL moves to the NOCOMMS state”



... differences in UML designs... (variables resetting)

R7: “when ICSL is in the NOCOMMS state and receives a CONNECT_confirm from SAI, it sends a CONNECT_Indication to the RBC and moves in the COMMS state”



... further works ...

- It would be interesting to be able to generate the UMC model directly from the XMI encoding of the SPARX-EA model
- It would be interesting to implement a SPARX-EA execution engine with nondeterministic behaviour to help testing concurrent systems
- It would be interesting if it were possible to select in UMC a SPARX-EA profile, reducing nondeterminism to match the sequential SPARX-EA behaviour

... Thanks! ...