Integration of the **FMVWG** (Formal Methods and Verification WG) and the **RASWG** (Reliability and Availability Studies WG)

Borja Fernandez Adiego
Jean-Charles Tournier
Enrique Blanco Viñuela
Ignacio David López Miguel
(BE-ICS)

Context - Failure categories

RAS WG

1. **Random Hardware** Failures
   - From degradation mechanism

Stochastic methods

Measures to combat the hardware random failures (e.g. RBD, FTA, etc.)

2. **Systematic** Failures
   - Incorrect **specification/design**
   - Human errors
   - **Software** errors
   - Maintenance and modifications
   - ….

Deterministic methods

Measures to combat the systematic failures (e.g. **formal specification**, **formal verification**, (functional) testing, etc.)

FMV WG

All types of failures have an impact on the **reliability** and **availability** of the global system

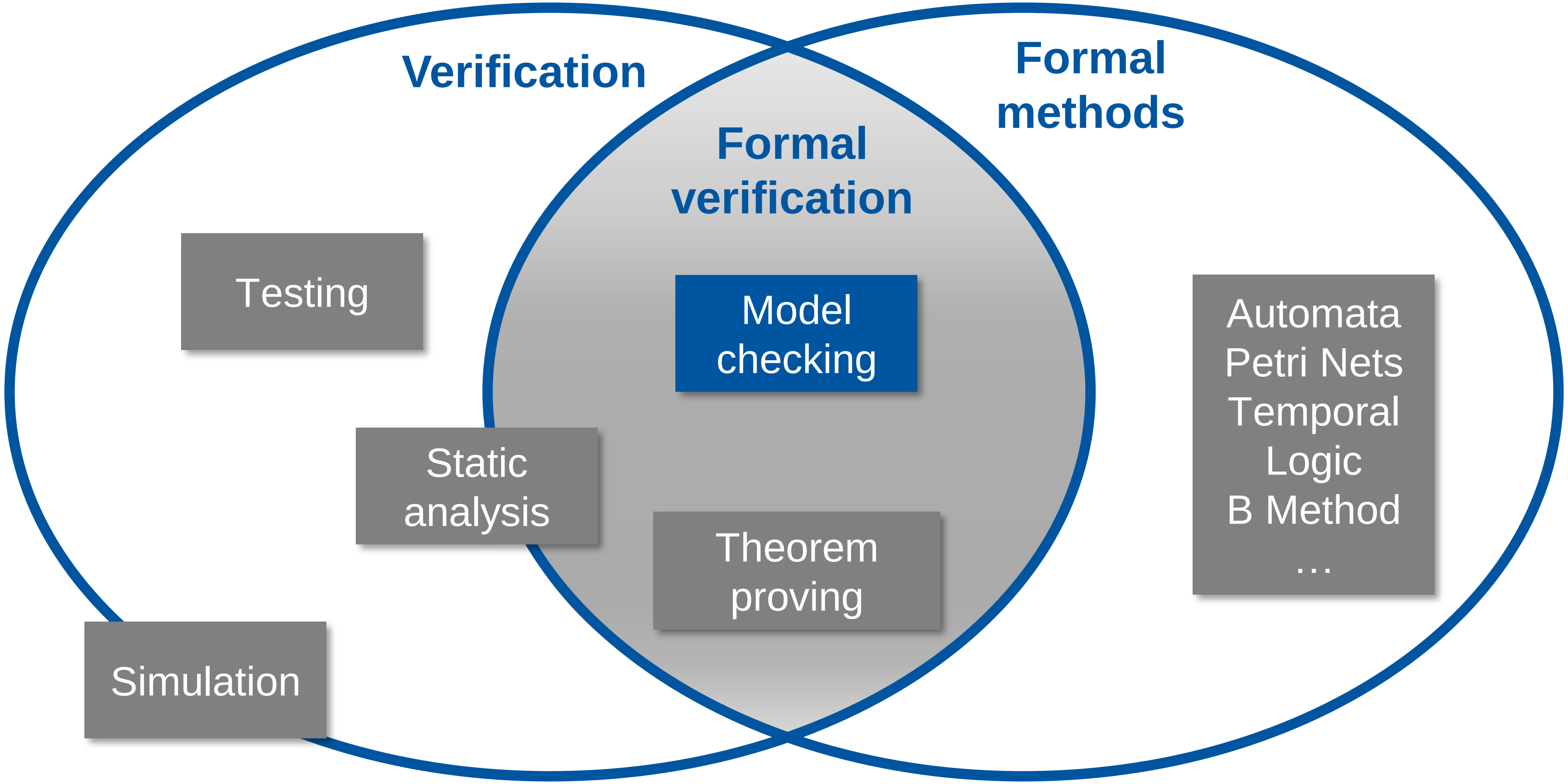
FMVWG potential scope

- ❑ Focus on **some** of the **systematic failures** (i.e. **software, design/specification**, etc.)

- ❑ By applying **formal methods** techniques to **specification** and **verification**

- ❑ https://readthedocs.web.cern.ch/pages/viewpage.action?spaceKey=FMVWG&title=Formal+methods+and+verification+working+group+Home

FMVWG potential scope



FMVWG topics (some examples)

❑ **PLC** programs formal verification

❑ **SystemVerilog** formal verification

❑ **C++** formal verification (e.g. FESA user code)

Formal verification

❑ Formal specification for **PLC** programs

❑ **SystemVerilog** assertions

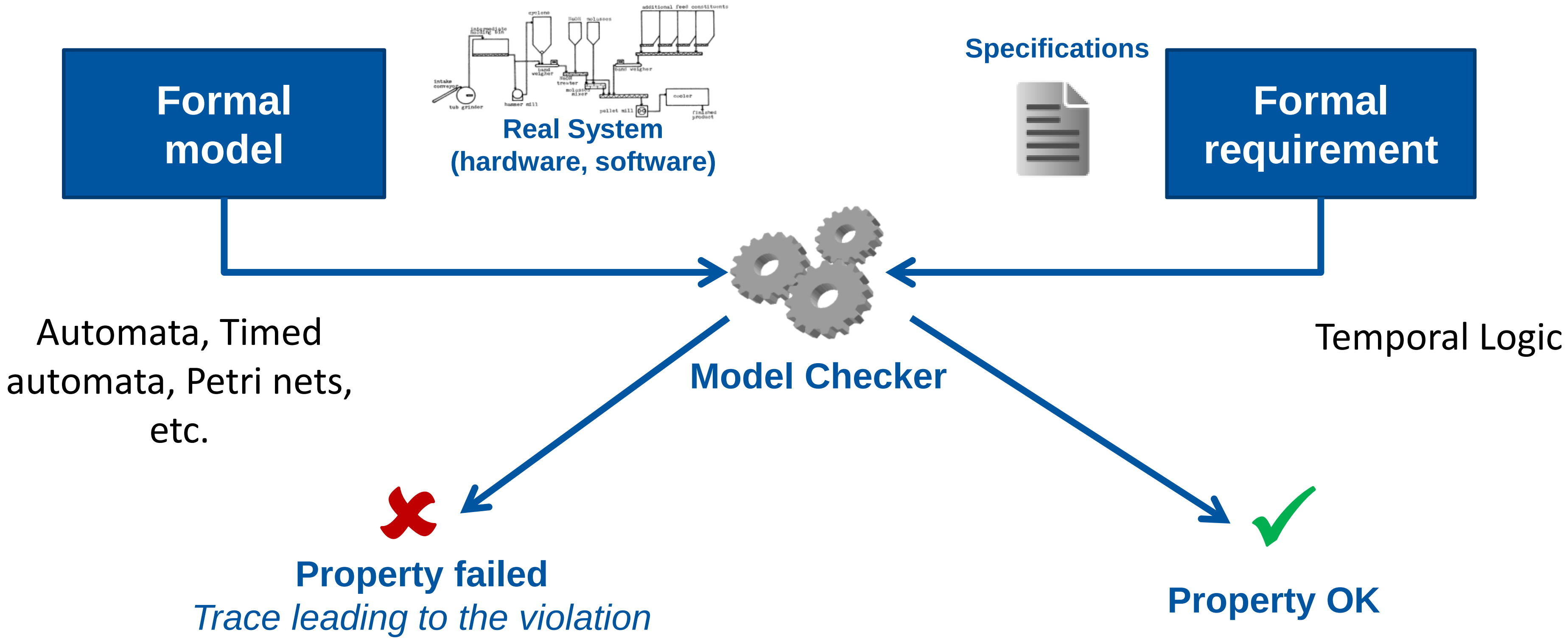
❑ …

Formal specification

Introduction to **model checking**

Given a **global model** of the system and a **formal property**, the **model checking algorithm checks exhaustively** that the model meets the property

Clarke and Emerson (1982) and Queille and Sifakis (1982)



Formal model

Real System (hardware, software)

Specifications

Formal requirement

Model Checker

Automata, Timed automata, Petri nets, etc.

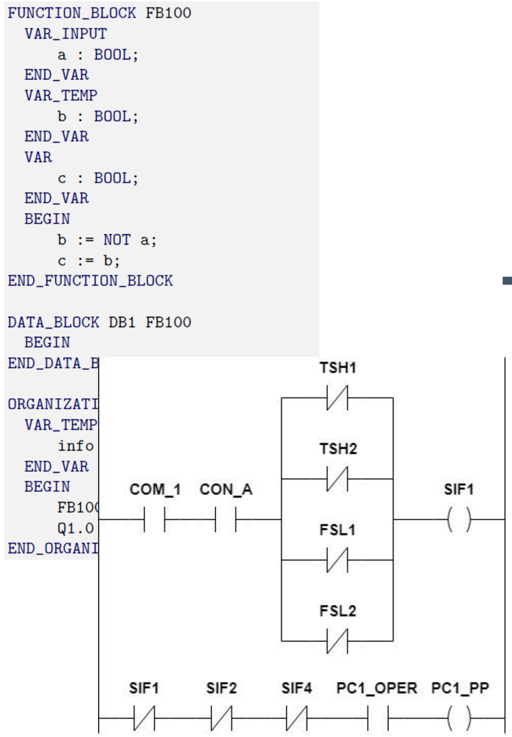
Temporal Logic

Property failed
Trace leading to the violation

Property OK

PLCverif methodology

PLC programs



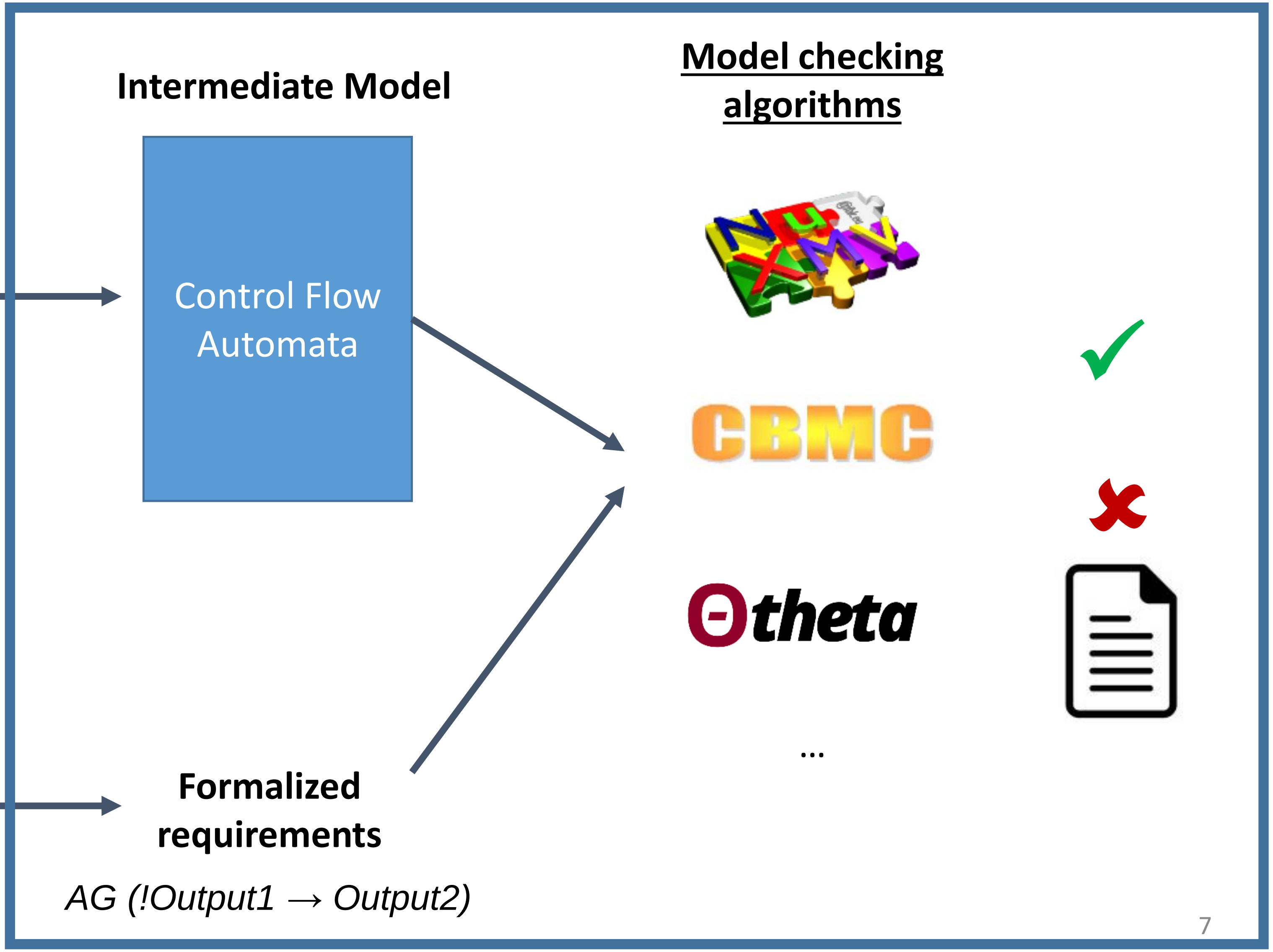
```
FUNCTION_BLOCK FB100
  VAR_INPUT
      a : BOOL;
  END_VAR
  VAR_TEMP
      b : BOOL;
  END_VAR
  VAR
      c : BOOL;
  END_VAR
  BEGIN
      b := NOT a;
      c := b;
END_FUNCTION_BLOCK

DATA_BLOCK DB1 FB100
  BEGIN
END_DATA_B

ORGANIZATI
  VAR_TEMP
      info
  END_VAR
  BEGIN
      FB10(
      Q1.0
END_ORGANI
```

Requirements

*If **Output1** is FALSE*
*then **Output2** is TRUE*



Intermediate Model

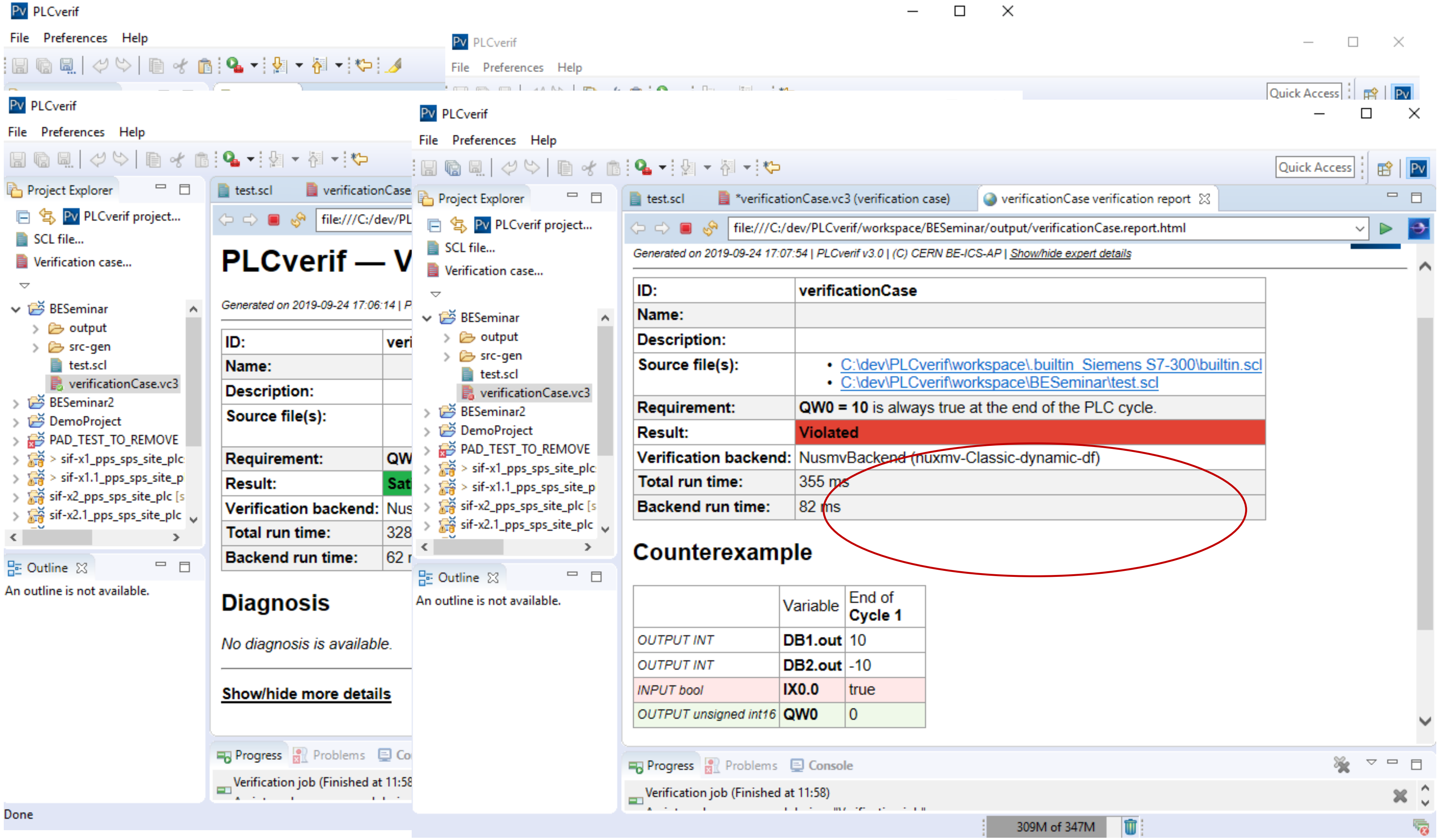
Control Flow Automata

Model checking algorithms

Formalized requirements

AG (!Output1 → Output2)

PLCverif usage



file:///C:/dev/PLCverif/workspace/BESeminar/output/verificationCase.report.html

Generated on 2019-09-24 17:07:54 | PLCverif v3.0 | (C) CERN BE-ICS-AP | Show/hide expert details

ID:	**verificationCase**
Name:	
Description:	
Source file(s):	• C:\dev\PLCverif\workspace\.builtin_Siemens S7-300\builtin.scl • C:\dev\PLCverif\workspace\BESeminar\test.scl
Requirement:	**QW0 = 10** is always true at the end of the PLC cycle.
Result:	**Violated**
Verification backend:	NusmvBackend (nuxmv-Classic-dynamic-df)
Total run time:	355 ms
Backend run time:	82 ms

Counterexample

	Variable	End of Cycle 1
OUTPUT INT	**DB1.out**	10
OUTPUT INT	**DB2.out**	-10
INPUT bool	**IX0.0**	true
OUTPUT unsigned int16	**QW0**	0

Verification job (Finished at 11:58)

file:///C:/dev/PL

PLCverif — V

Generated on 2019-09-24 17:06:14 | P

ID:	**veri**
Name:	
Description:	
Source file(s):	
Requirement:	**QW**
Result:	**Sat**
Verification backend:	Nus
Total run time:	328
Backend run time:	62 r

Diagnosis

No diagnosis is available.

Show/hide more details

Verification job (Finished at 11:58

FMVWG and RASWG integration

- **Common goal – improve reliability and availability** of our systems

- Many **benefits**:
  - Maximum **visibility**
  - Join forces and avoid duplication of efforts
  - Collaboration between groups
  - Etc.

- Challenges:
  - RASWG scope is (already) very large
  - Formal methods domain is also very large
  - Different methods
  - Different target (systematic vs hardware random failure detection)

- Concerns:
  - Dilution of formal methods topics in the RASWG agenda (ideally we would like to have a significant number of dedicated meetings/presentations)