VALU3S

VVML: Specifying Workflows for V&V Methods

– *Formalisation* –

VALU3S Summer School 2023, Genoa, Italy

José Proença (ISEP)

19 July 2023

ECSEL Joint Undertaking
Electronic Components and Systems for European Leadership

This project has received funding from the ECSEL Joint Undertaking (JU) under grant agreement No 876852. The JU receives support from the European Union's Horizon 2020 research and innovation programme and Austria, Czech Republic, Germany, Ireland, Italy, Portugal, Spain, Sweden, Turkey. Disclaimer: The ECSEL JU and the European Commission are not responsible for the content on this presentation or any use that may be made of the information it contains.

FCT Fundação para a Ciência e a Tecnologia

REPÚBLICA PORTUGUESA

Photo (of the Iberian Ibex) by Arturo de Frías. This work is financed by National Funds through FCT - Fundação para a Ciência e a Tecnologia, I.P. (Portuguese Foundation for Science and Technology) within the project IBEX, with reference PTDC/CCI-COM/4280/2021.

Who am I

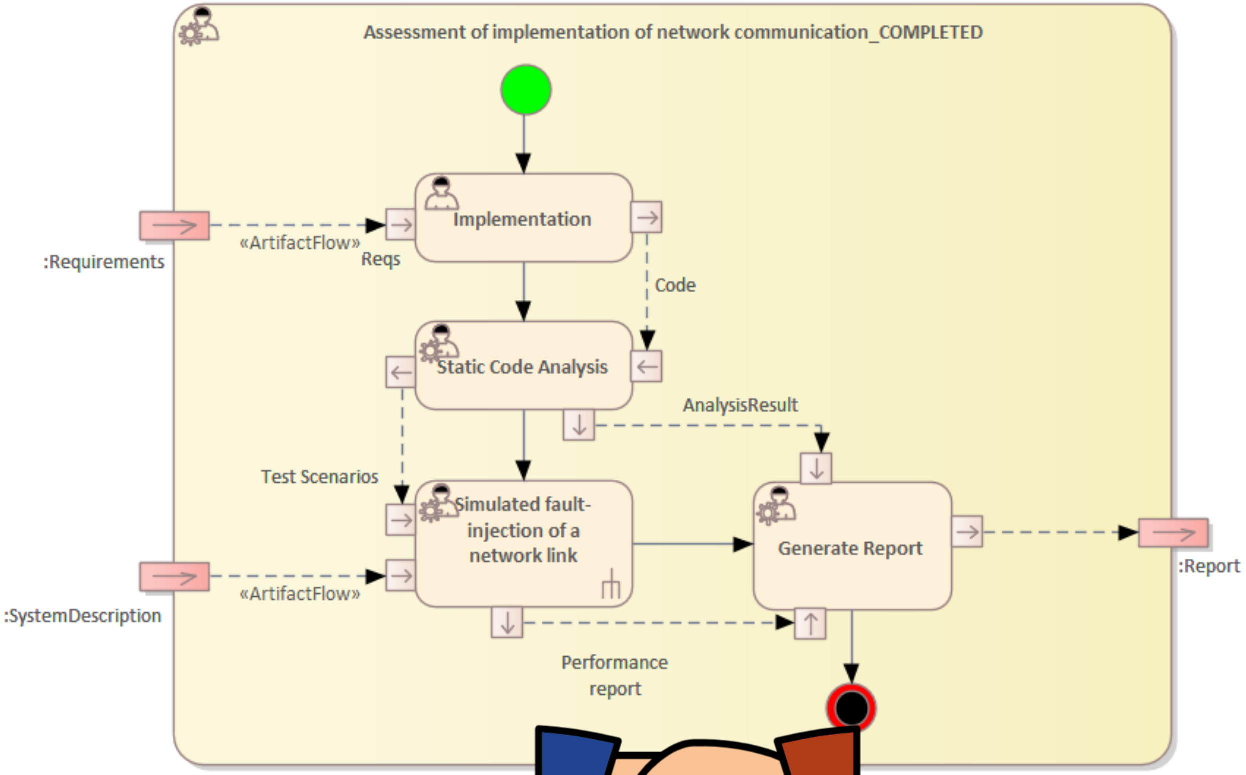
- José Proença
- Polytechnic of Porto, Portugal
  - CISTER – Real-Time & embedded computing systems
- Ph.D. since 2011 from Leiden University, the Netherlands

- Research Interests
  - Formal methods/verification
  - Distributed and concurrent systems
  - Programming languages

https://jose.proenca.org

	Tuesday - 18th	Wednesday - 19th	Thursday - 20th
9h00-10h30	Introduction to V&V of dependable CPS	VVML: Specifying Workflows for V&V Methods	Symbolic Model Checking of Hybrid Systems
10h30-11h00	Break	Break	Break
11h00-12h30	An overview to testing of safety-critical CPS	Formal requirements engineering	Deductive Verification in a Nutshell
12h30-14h00	Lunch & Poster Presentation	Lunch & Poster Presentation	Lunch & Poster Presentation
14h00-15h30	Software-implemented fault injection	Introduction to Model Checking	An overview of relevant safety and cybersecurity standards
15h30-16h00	Break	Break	Break
16h00-17h30	Simulation-based fault injection	A V&V framework for storing elements of V&V activities	An overview of relevant safety and cybersecurity standards

VVML's goal

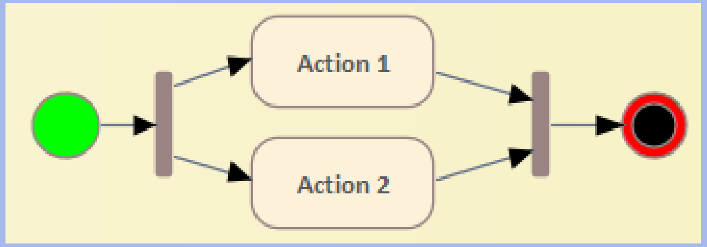


Users + Development team

Verification team

(Informal) Guidelines for correct workflows

Structure



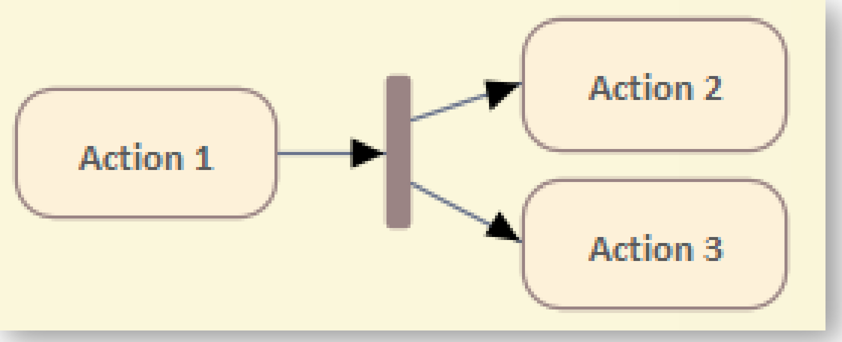
- Exactly **1 start** & **1 stop**
- Mandatory: input/output sequence flow
- Mandatory: >1 output artifact
- …

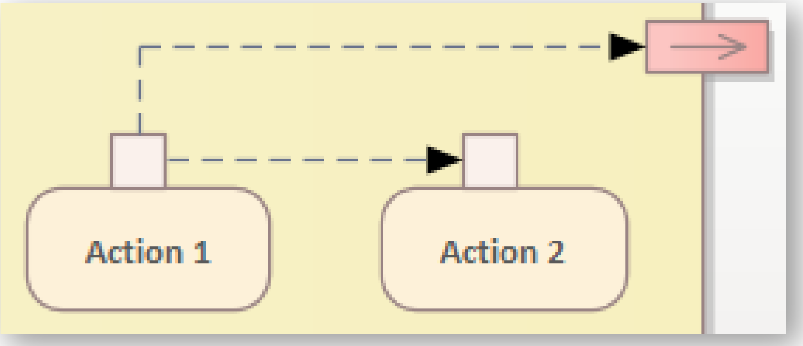
Behaviour

- ***Act*** is executed when any previous ***Act'*** is finished
- Nested calls are **atomic**
- …

(Informal) Guidelines for correct workflows

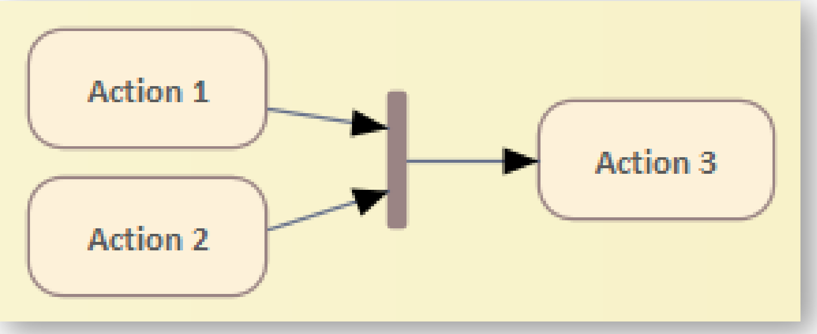
More behavioural guidelines

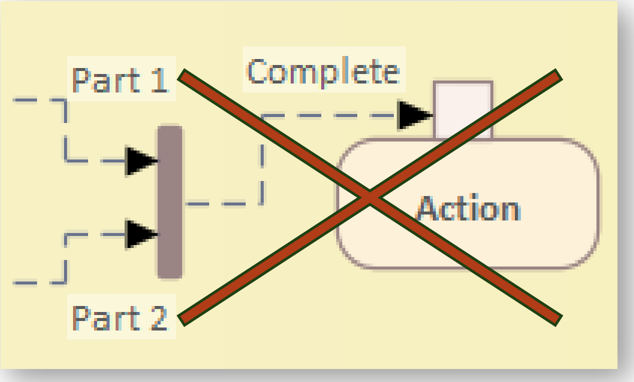




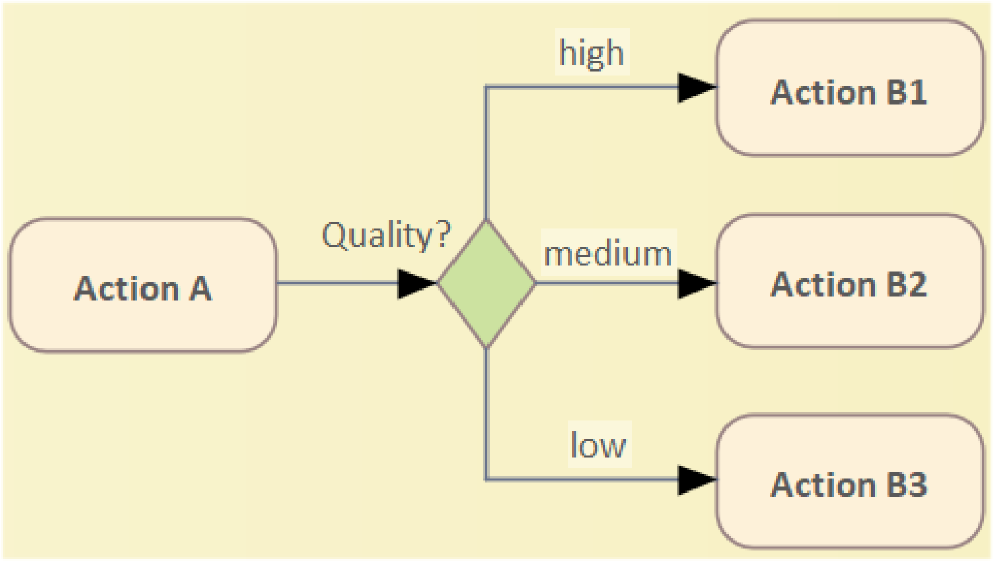
- Fork to start parallel
- Join to merge parallel

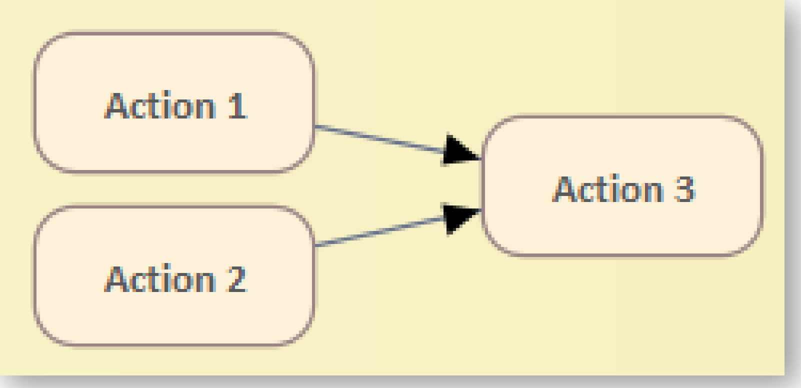
- Copy artefacts
- Do NOT join artefacts





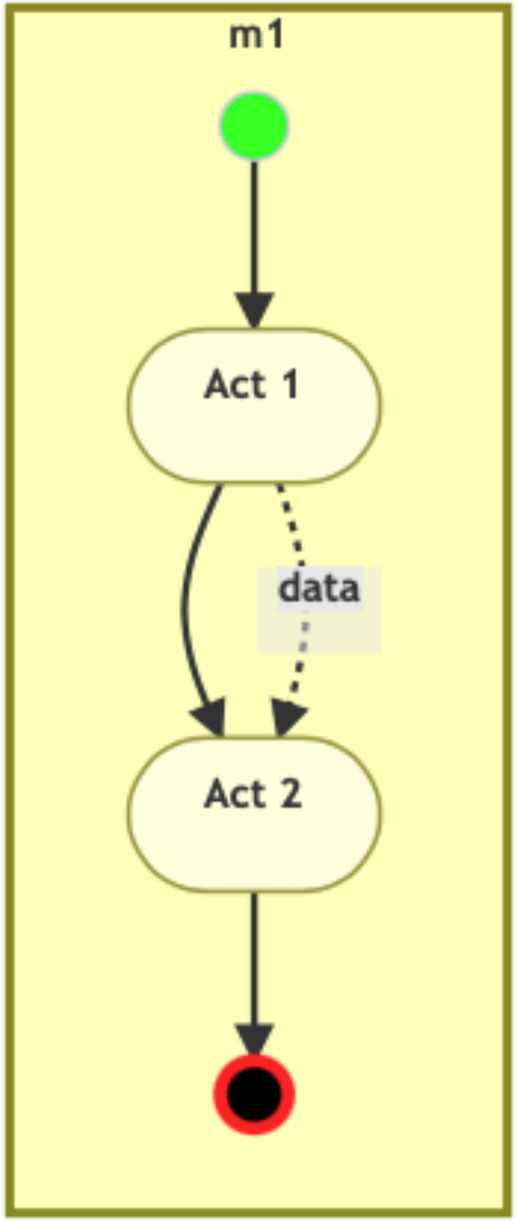
More behavioural guidelines



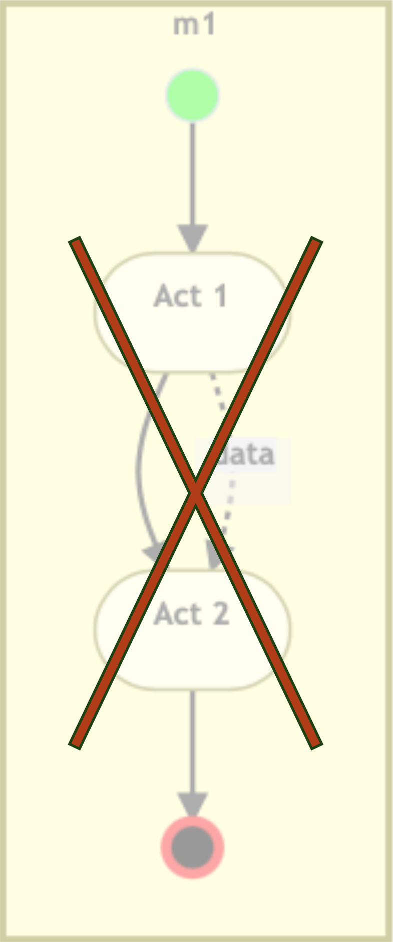


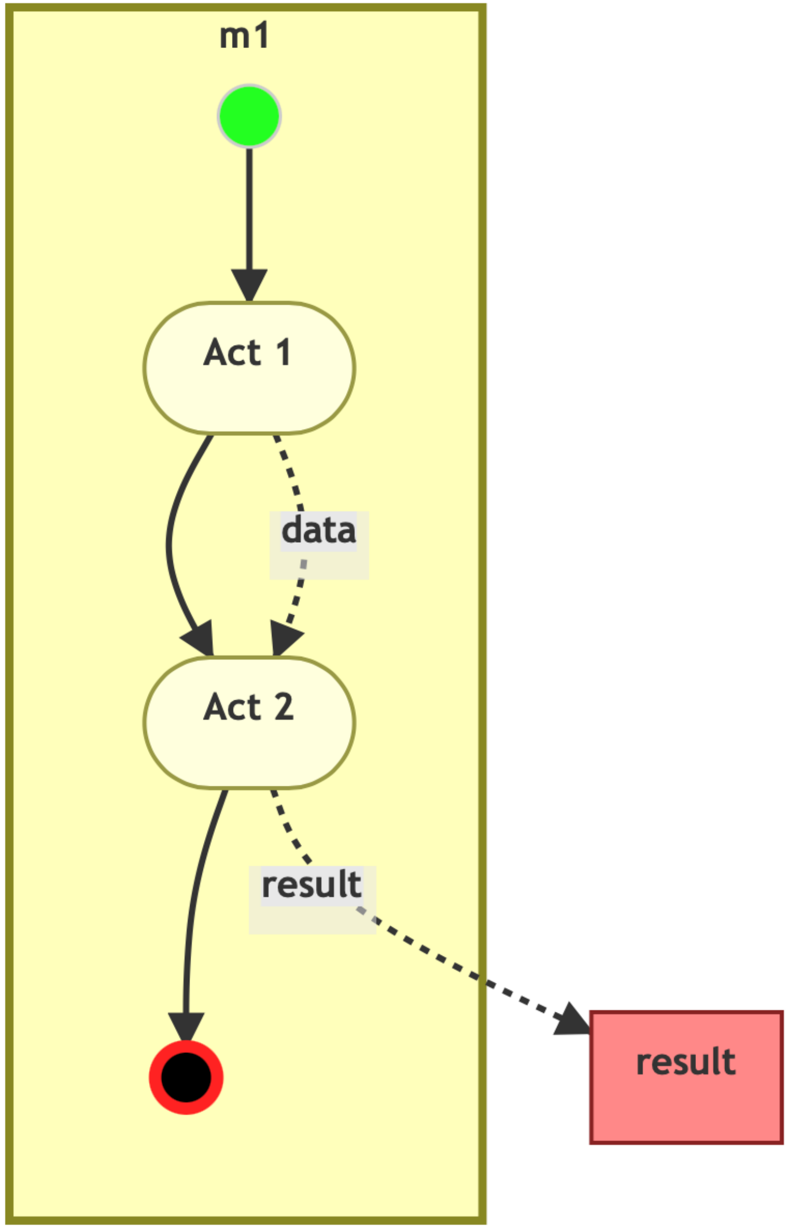
- Start alternative flows
- Join alternative flows

Is it correct? 1/10

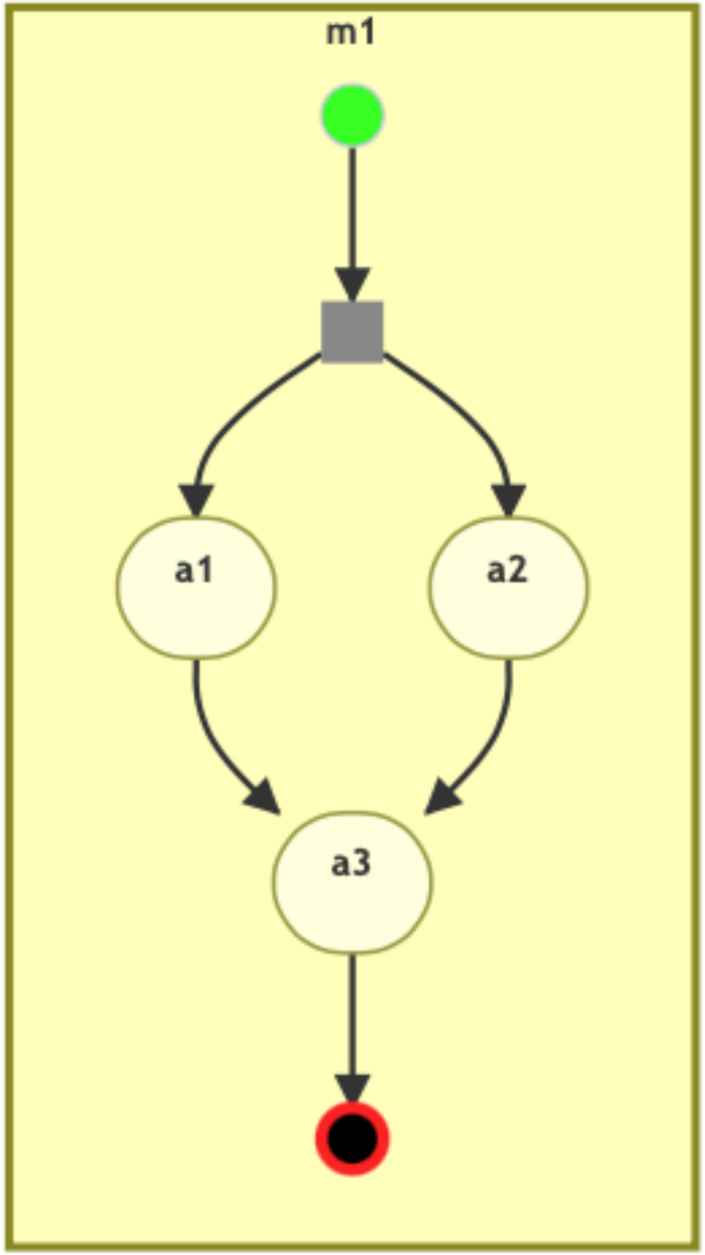


Is it correct? 1/10

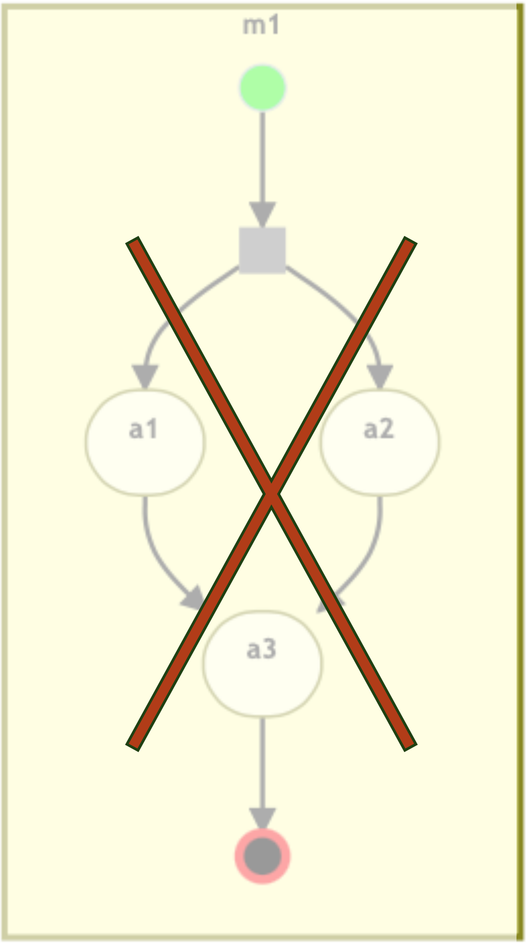


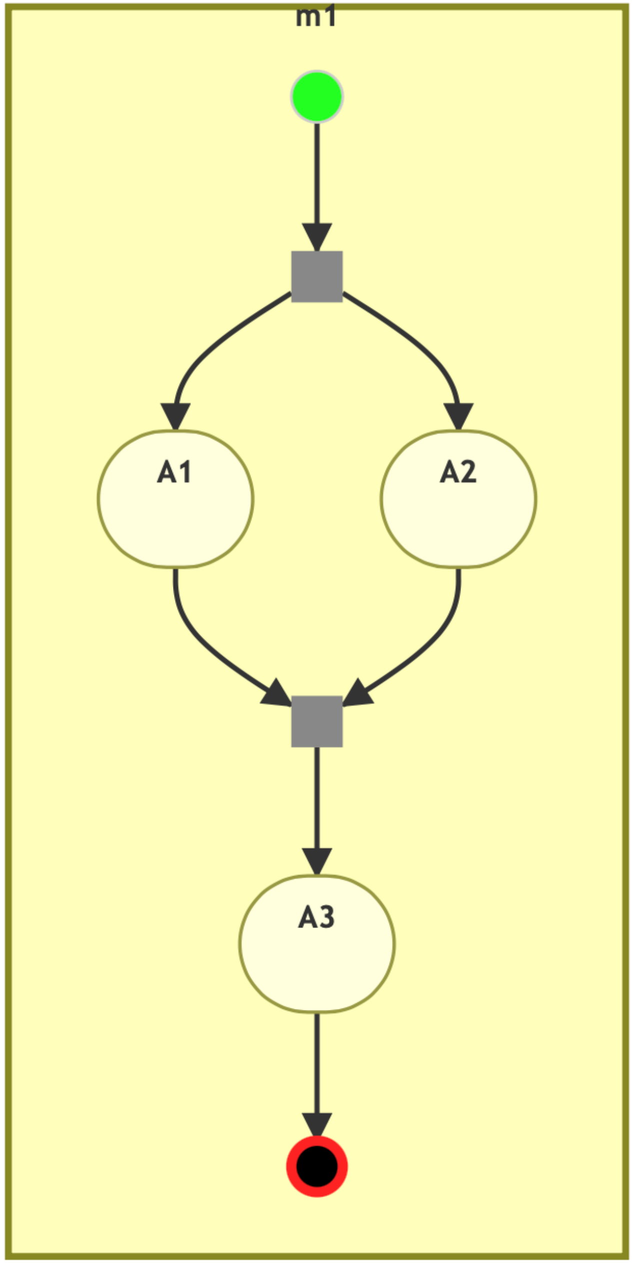


Is it correct? 2/10

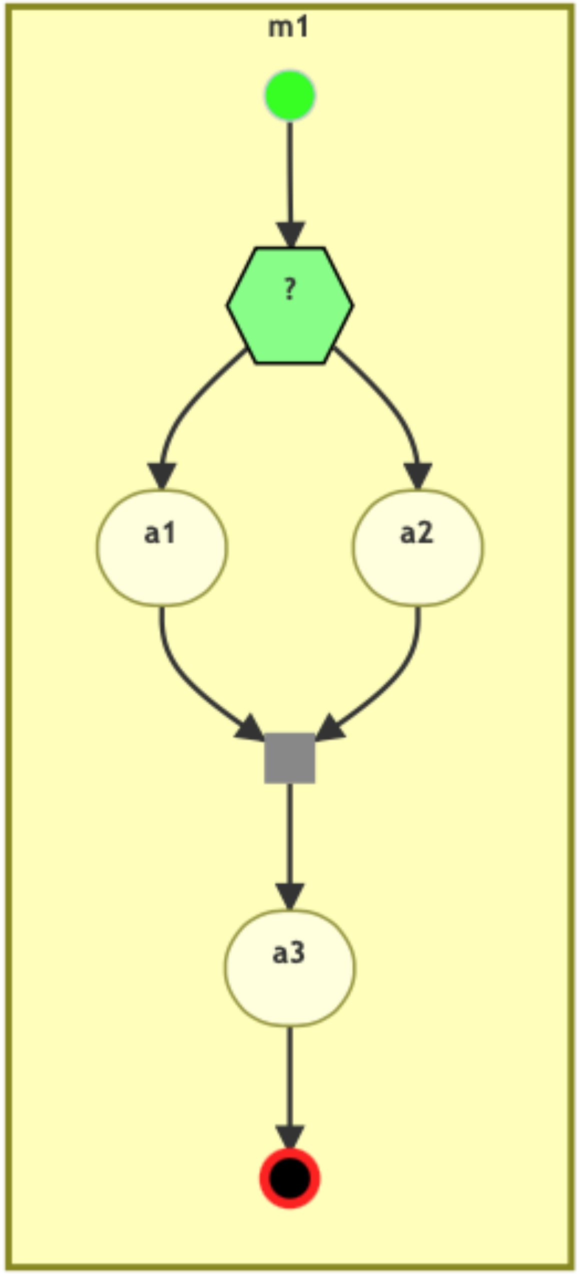


Is it correct? 2/10

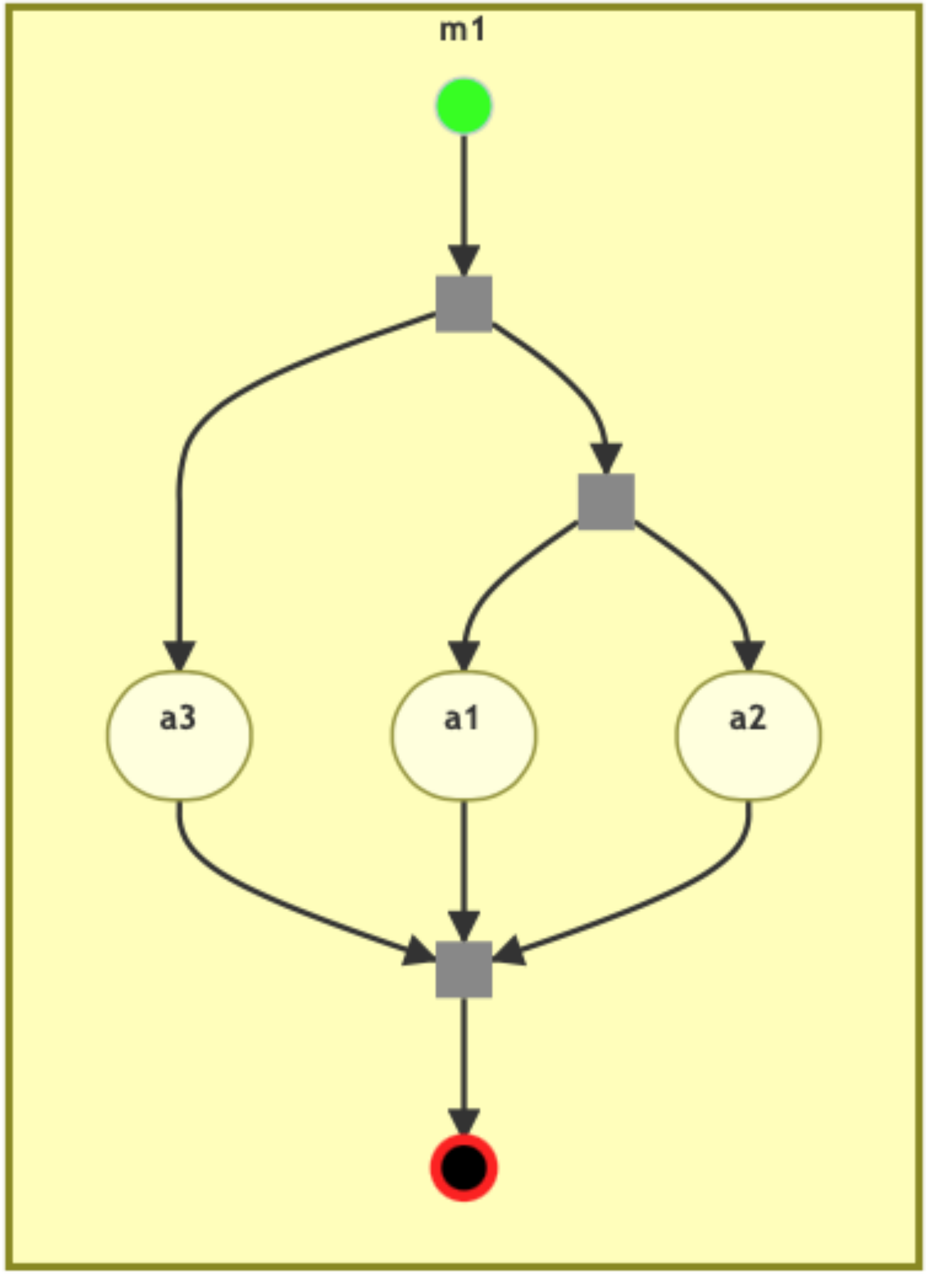




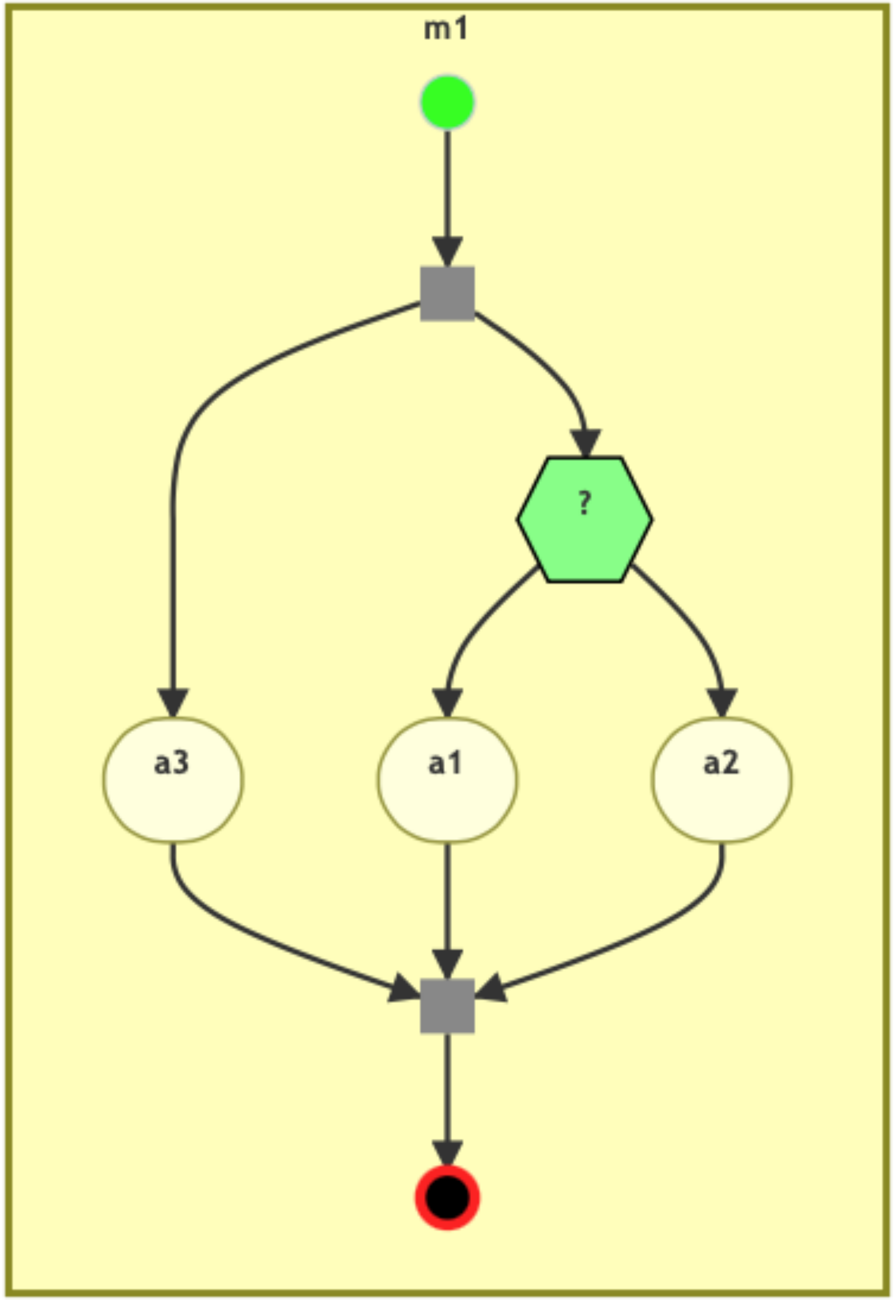
Is it correct? 3/10



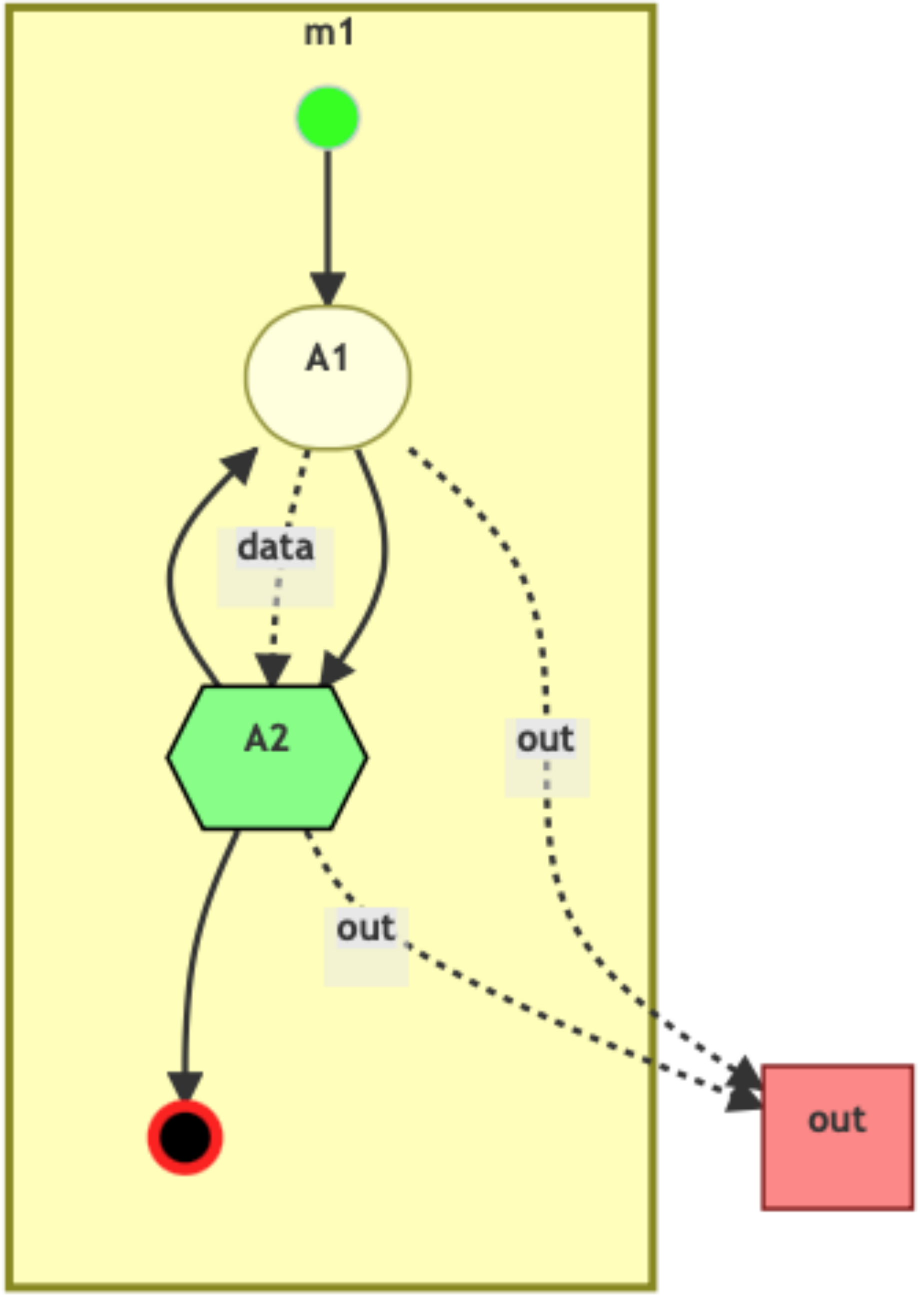
Is it correct? 4/10



Is it correct? 5/10



Is it correct? 6/10



Continue online…

https://
cister-labs.github.io/
coreVVML/
?#6

Correct VVML workflow?

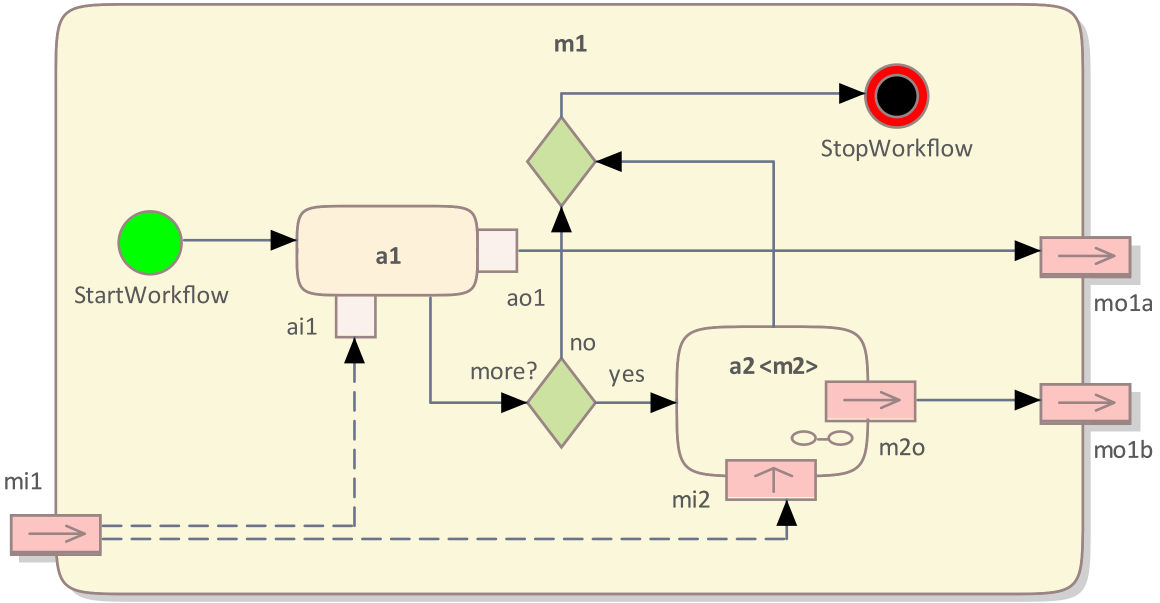
- (**good structure**)
- **never blocks** before reaching the stopping node
- never reaches the **stopping** node while some activity **is still running**
- can **always** reach the **stopping** node
- never **re-enters** a **running** activity
- is able to **start all** of its activities

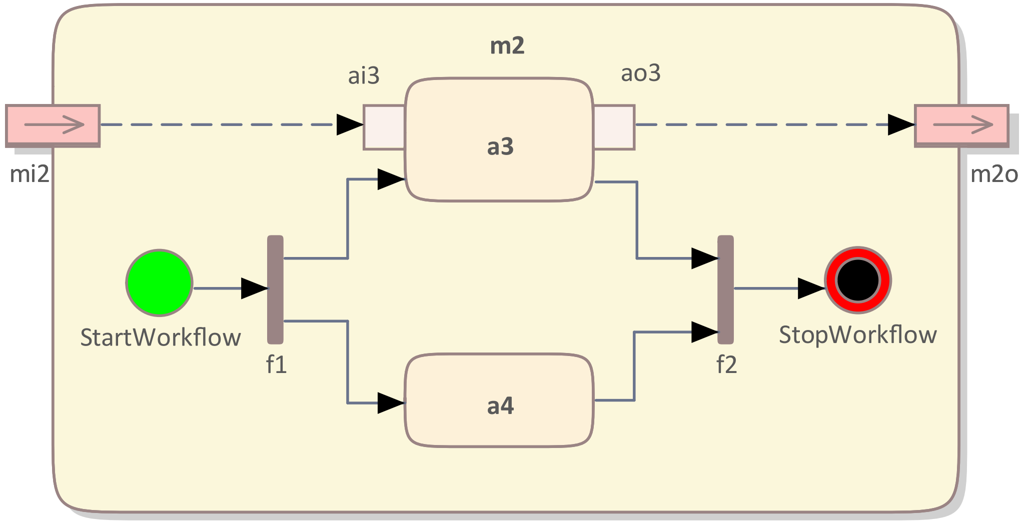
Correct VVML workflow?

- (**good structure**)
- **never blocks** before reaching the stopping node
- never reaches the **stopping** node while some activity **is still running**
- can **always** reach the stopping node
- never **re-enters** a **running** activity
- is able to **start all** of its activities

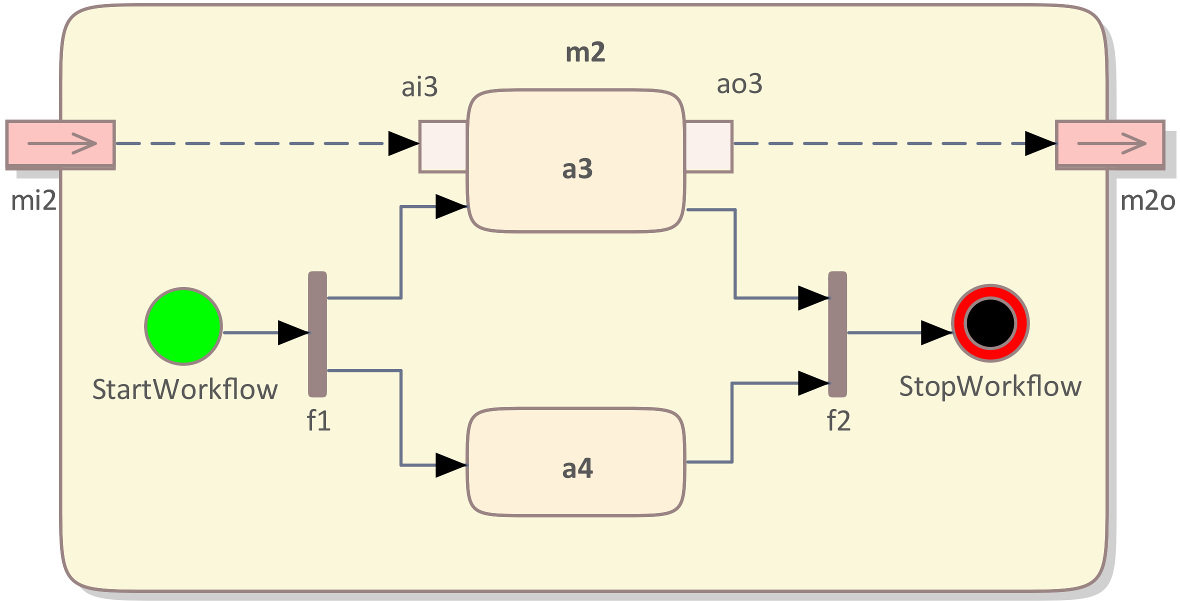
What does it mean <u>PRECISELY</u> to <u>RUN</u>?

Core VVML – Syntax





Core VVML – Syntax



$$\langle A, I, \downarrow, F, Sr, Sk, \rightarrow, \dashrightarrow, \alpha, \gamma \rangle$$

$$A = \{a_3, a_4\}$$

$$I = \{f_1\}$$

$$\downarrow = \{f_2\}$$

$$F = \{f_1, f_2\}$$

$$Sr = \{mi_2, ao_3\}$$

$$Sk = \{ai_3, mo_2\}$$

$$\rightarrow = \{\langle f_1, a_3 \rangle, \langle f_1, a_4 \rangle, \langle a_3, f_2 \rangle, \langle a_4, f_2 \rangle\}$$

$$\dashrightarrow = \{\langle mi_2, ai_3 \rangle, \langle ao_3, mo_2 \rangle\}$$

$$\alpha = \{ai_3 \mapsto a_3, ao_3 \mapsto a_3\}$$

$$\gamma = \{\}$$

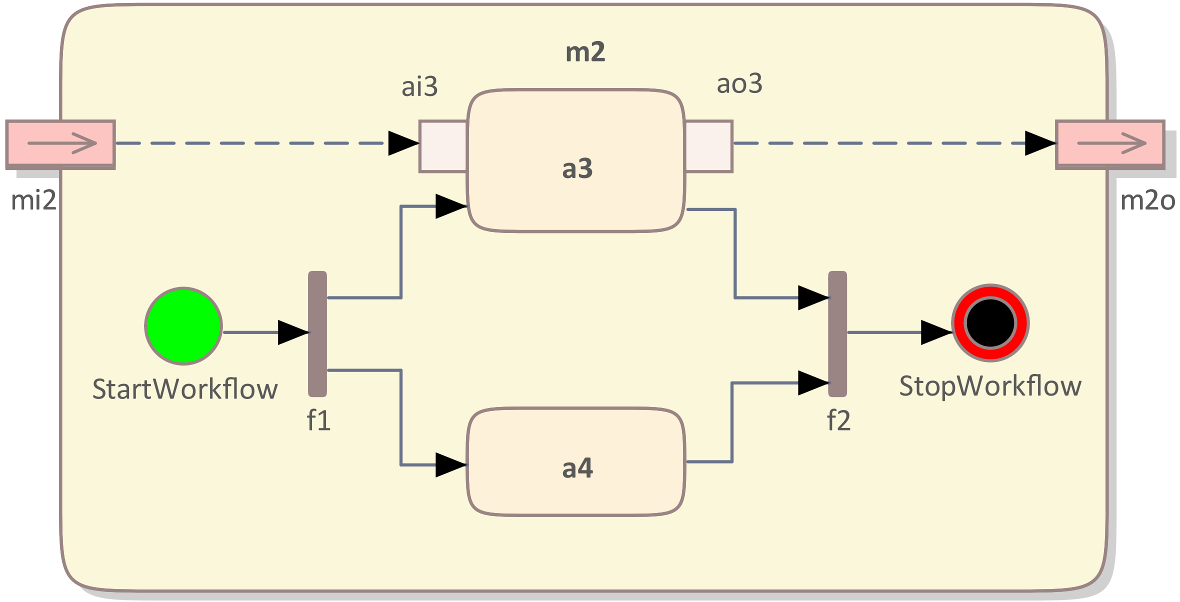
Core VVML – Semantics (without artefacts)

Method State

$$\langle AS, FS \rangle$$

$$a_3 \mapsto \text{Idle} \qquad f_1 \mapsto 1$$

$$a_4 \mapsto \text{Idle} \qquad f_2 \mapsto 0$$



Activity State

Idle

Ready

Run

Done

Core VVML – Semantics (without artefacts)

$$\frac{}{\langle AS[a \mapsto \textsf{Ready}], FS\rangle \longrightarrow \langle AS[a \mapsto \textsf{Run}], FS\rangle}\quad(\text{start})$$

$$\frac{\gamma(a) = \bot}{\langle AS[a \mapsto \textsf{Run}], FS\rangle \longrightarrow \langle AS[a \mapsto \textsf{Done}], FS\rangle}\quad(\text{end})$$

$$\frac{\gamma(a) = m_2 \qquad m_2 \textsf{ executes}}{\langle AS[a \mapsto \textsf{Run}], FS\rangle \longrightarrow \langle AS[a \mapsto \textsf{Done}], FS\rangle}\quad(\text{call})$$

$$\frac{\begin{array}{c} AS = \{a \mapsto \textsf{Ready} \mid a \in m.I \cap m.A\} \\ FS = \{f \mapsto 1 \mid f \in m.I \cap m.F\} \\ \langle AS, FS\rangle \Longrightarrow \langle \_, \_\rangle \end{array}}{m \textsf{ executes}}\quad(\text{init})$$

Core VVML – Semantics (WITH artefacts)

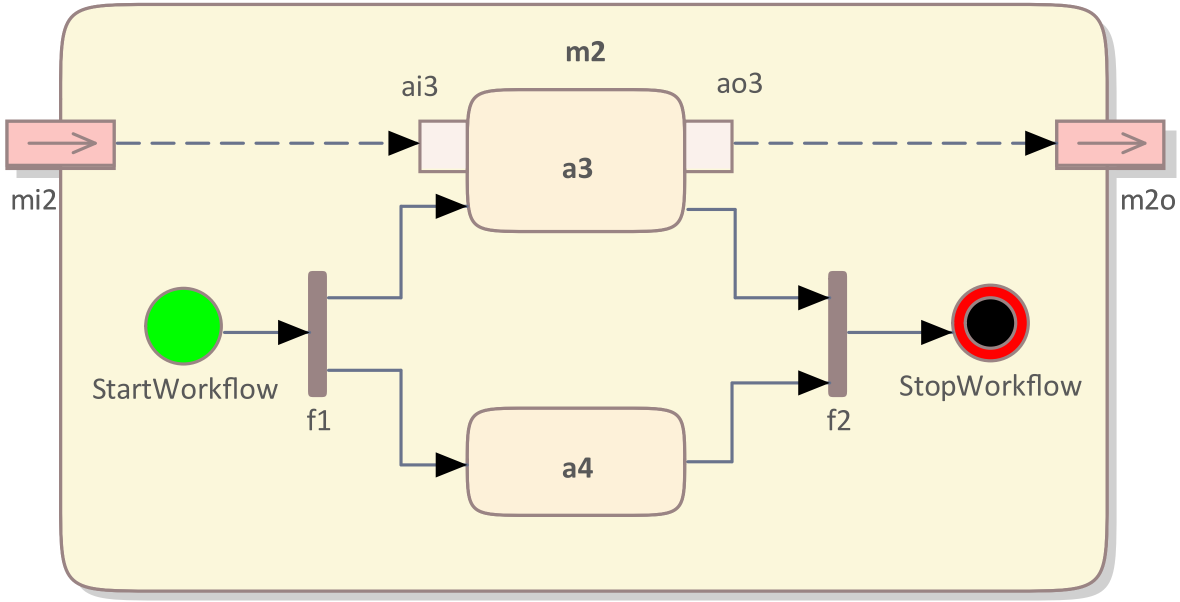
Method State

$$\langle AS, FS, \boxed{PT\rangle}$$

$a_3 \mapsto$ Idle $\quad f_1 \mapsto 1$

$a_4 \mapsto$ Idle $\quad f_2 \mapsto 0$

$$\boxed{\begin{array}{l} ai_3 \mapsto t_\perp \\ ao_3 \mapsto t_{a_3} \end{array}}$$



Activity State

Idle

Ready

$\boxed{\text{Run}(PT)}$

Done

Core VVML – Semantics (WITH artefacts)

$$\frac{}{\langle AS[a \mapsto \mathsf{Ready}], FS\boxed{, PT}\rangle \longrightarrow \langle AS[a \mapsto \boxed{\mathsf{Run}(PT(\mathsf{inputs}(a)))}], FS\boxed{, PT}\rangle} \qquad \text{(start)}$$

$$\frac{\gamma(a) = \bot \qquad PT_a \subseteq \{p \mapsto \mathtt{t}_a \mid p \in \mathsf{outpupt}(a)\}}{\langle AS[a \mapsto \mathsf{Run}(\_)], FS, PT\rangle \longrightarrow \langle AS[a \mapsto \mathsf{Done}], FS, PT[PT_a]\rangle} \qquad \text{(end)}$$

$$\frac{\begin{array}{c}\gamma(a) = m_2 \qquad \langle m_2, PT\rangle \rightsquigarrow PT_2 \\ PT_a = \{p \mapsto \mathtt{t} \mid (p \mapsto \mathtt{t}) \in PT_2, p \in \mathsf{outputs}(a)\}\end{array}}{\langle AS[a \mapsto \mathsf{Run}(\_)], FS, PT\rangle \longrightarrow \langle AS[a \mapsto \mathsf{Done}], FS, PT[PT_a]\rangle} \qquad \text{(call)}$$

Core VVML Tools

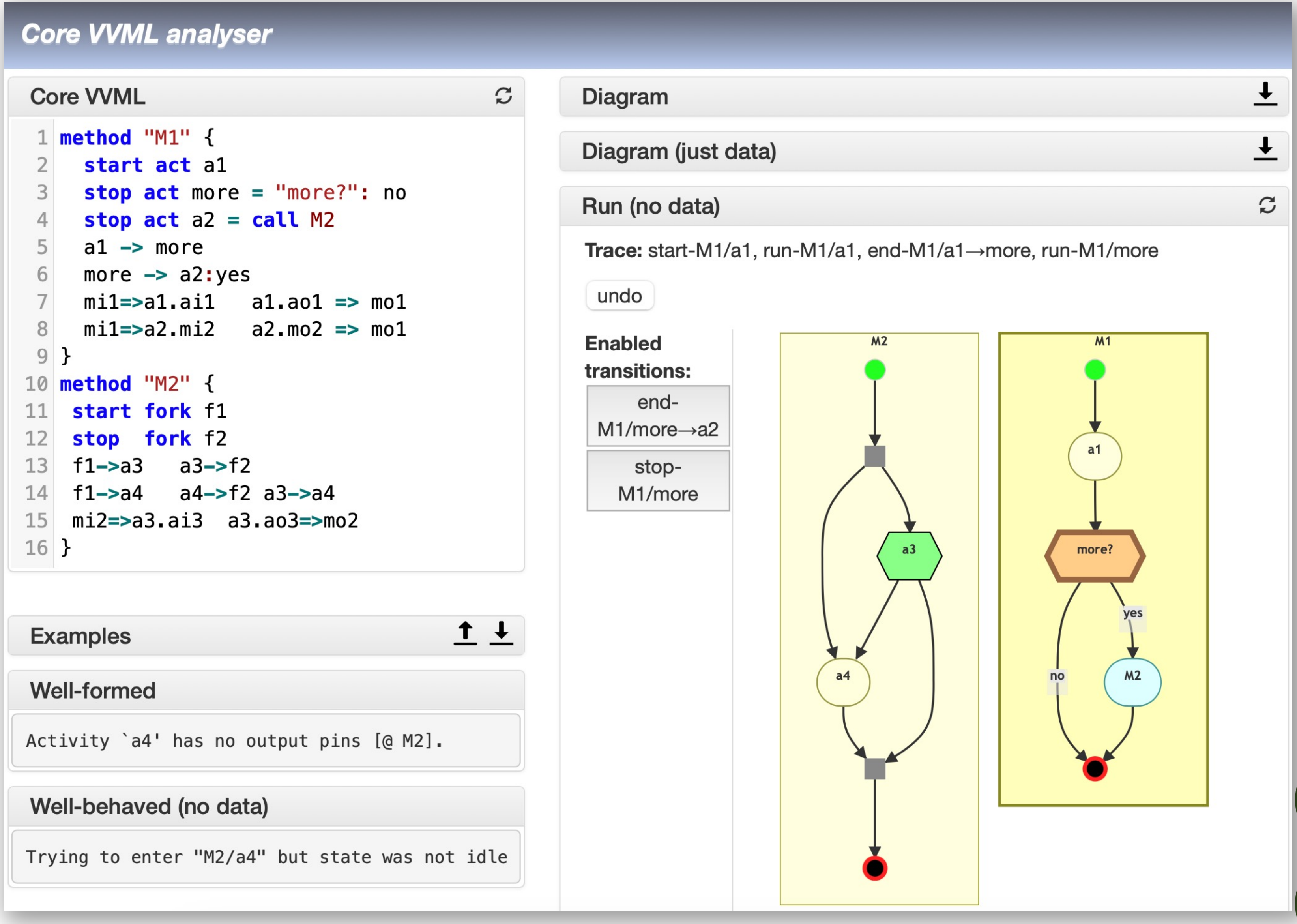
https://cister-labs.github.io/coreVVML/

Simulate & Automatically check:

- **Structure** (well-formed)
- **Behaviour** (well-behaved)

No artefacts yet:

- Not useful yet
- Need *contracts*



Core VVML analyser

Core VVML

```
method "M1" {
  start act a1
  stop act more = "more?": no
  stop act a2 = call M2
  a1 -> more
  more -> a2:yes
  mi1=>a1.ai1   a1.ao1 => mo1
  mi1=>a2.mi2   a2.mo2 => mo1
}
method "M2" {
 start fork f1
 stop  fork f2
 f1->a3    a3->f2
 f1->a4    a4->f2 a3->a4
 mi2=>a3.ai3  a3.ao3=>mo2
}
```

Examples

Well-formed

```
Activity `a4' has no output pins [@ M2].
```

Well-behaved (no data)

```
Trying to enter "M2/a4" but state was not idle
```

Diagram

Diagram (just data)

Run (no data)

Trace: start-M1/a1, run-M1/a1, end-M1/a1→more, run-M1/more

undo

Enabled transitions:

end-M1/more→a2

stop-M1/more

VALU3S

Verification and Validation of Automated Systems' Safety and Security

www.valu3s.eu

Quantitative methods for cyber-physical programming

lmf.di.uminho.pt/Ibex

ECSEL Joint Undertaking
Electronic Components and Systems for European Leadership

This project has received funding from the ECSEL Joint Undertaking (JU) under grant agreement No 876852. The JU receives support from the European Union's Horizon 2020 research and innovation programme and Austria, Czech Republic, Germany, Ireland, Italy, Portugal, Spain, Sweden, Turkey. Disclaimer: The ECSEL JU and the European Commission are not responsible for the content on this presentation or any use that may be made of the information it contains.

FCT Fundação para a Ciência e a Tecnologia

REPÚBLICA PORTUGUESA

Photo (of the Iberian Ibex) by Arturo de Frías. This work is financed by National Funds through FCT - Fundação para a Ciência e a Tecnologia, I.P. (Portuguese Foundation for Science and Technology) within the project IBEX, with reference PTDC/CCI-COM/4280/2021.