



LUND UNIVERSITY

On Extending JGrafchart with Support for FMI for Co-Simulation

Theorin, Alfred; Johnsson, Charlotta

2014

[Link to publication](#)

Citation for published version (APA):

Theorin, A., & Johnsson, C. (2014). *On Extending JGrafchart with Support for FMI for Co-Simulation*. Paper presented at 10th International Modelica Conference, Lund, Sweden.

Total number of authors:

2

General rights

Unless other specific re-use rights are stated the following general rights apply:

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal

Read more about Creative commons licenses: <https://creativecommons.org/licenses/>

Take down policy

If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.

LUND UNIVERSITY

PO Box 117
221 00 Lund
+46 46-222 00 00

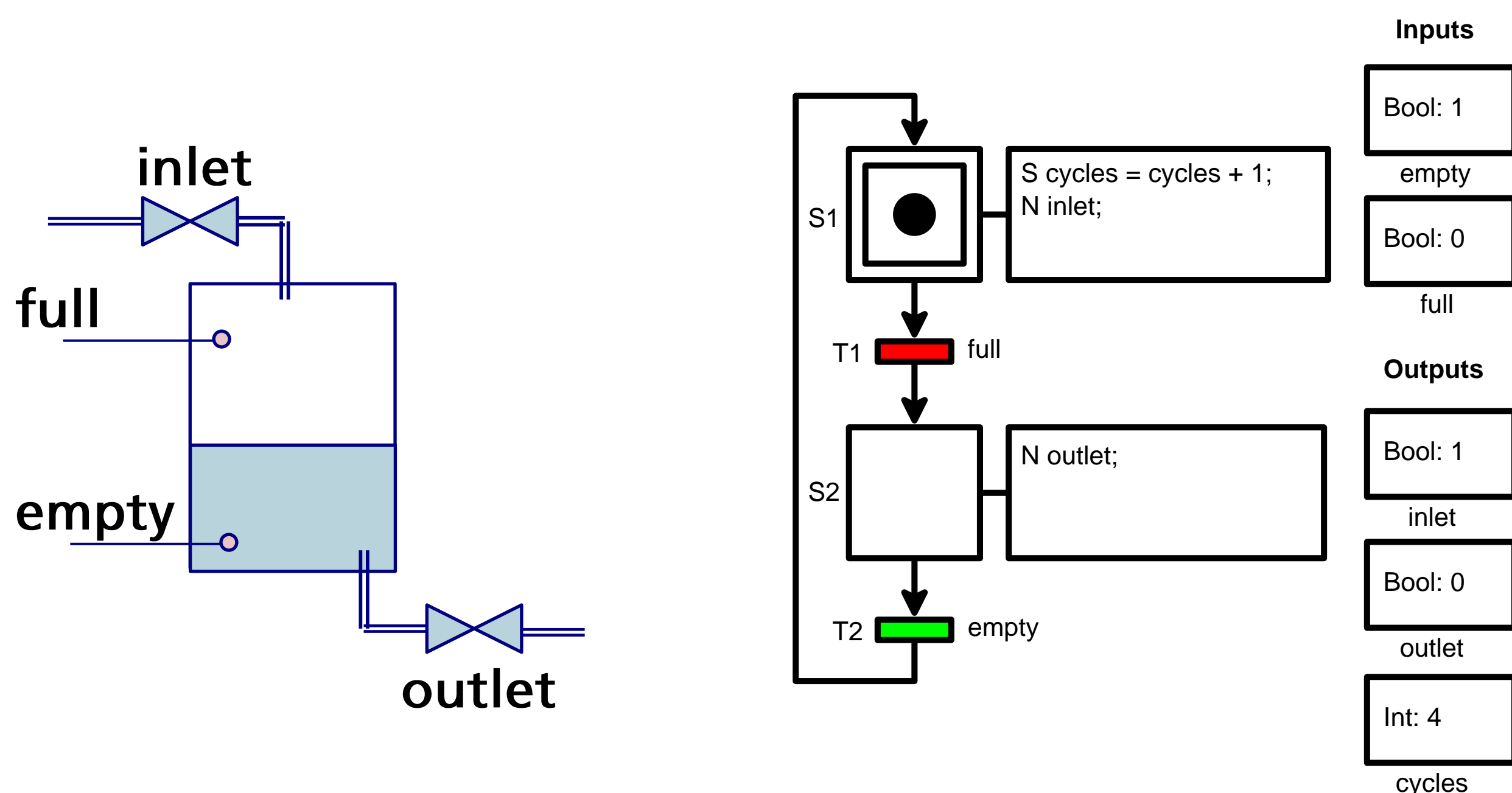
On Extending JGrafchart with Support for FMI for Co-Simulation

Alfred Theorin and Charlotta Johnsson
Department of Automatic Control, Lund University, Lund, Sweden

Goal

JGrafchart applications are often simulated before controlling the real system. The simulation models are often very simple. It is desirable to connect JGrafchart to a simulation environment to get better tool integration and thus gain access to better modeling capabilities.

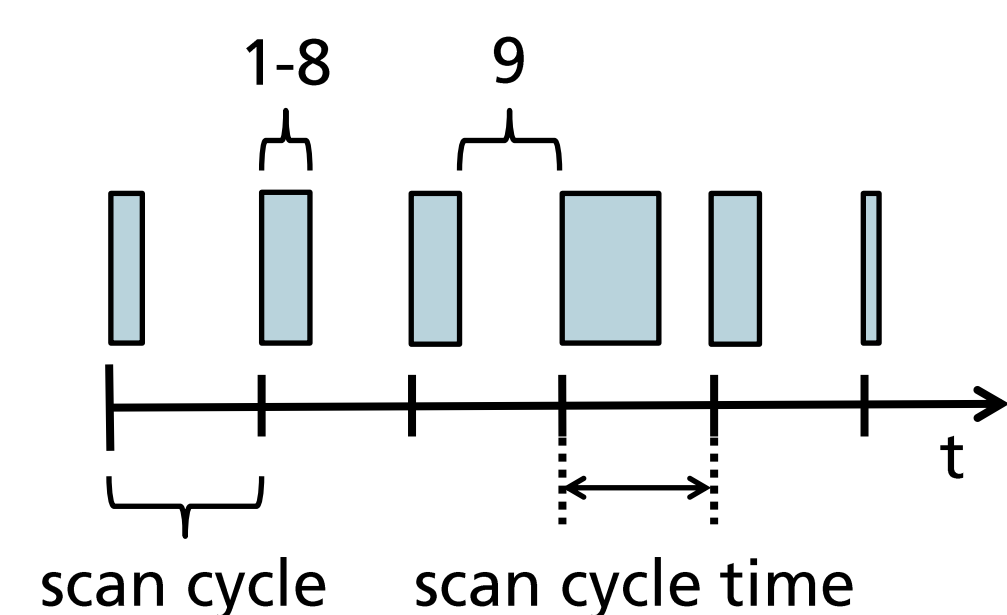
The Grafchart Language



A tank with digital level sensors and inlet/outlet valves.

A Grafchart application for filling and emptying the tank, counting the number of cycles.

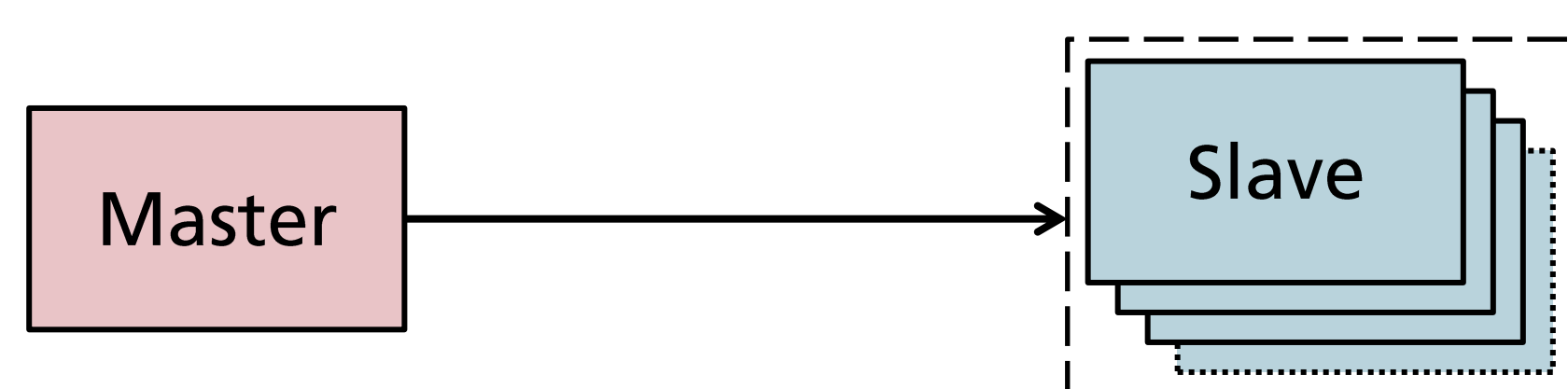
1. Read inputs
2. Mark fireable transitions
3. Unmark conflicting transitions of lower priority
4. Fire marked transitions
5. Update step properties t and s
6. Execute P actions
7. Mark variables subject to N actions
8. Update marked variables
9. Sleep until next scan cycle



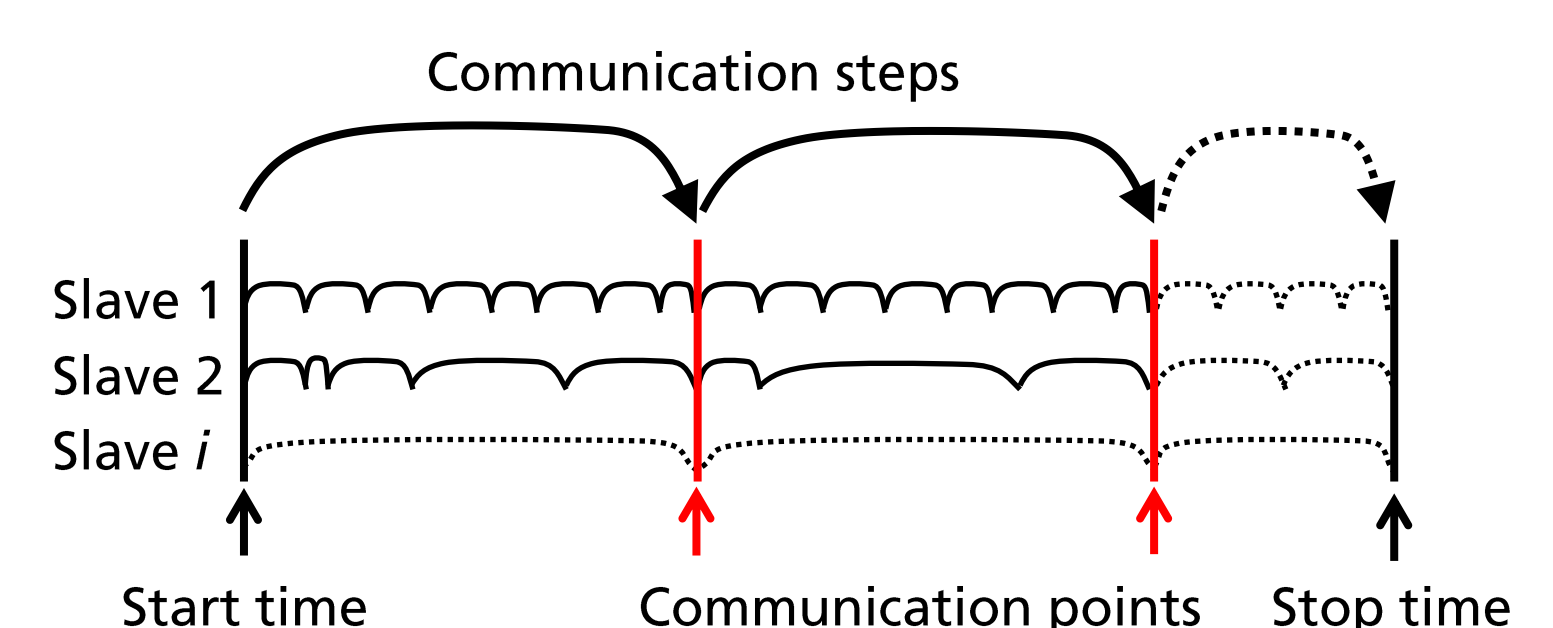
Grafchart Execution model [1]

FMI for Co-Simulation

FMI for Co-Simulation is a standard for simulation of coupled technical systems with focus on time-dependent problems [2].

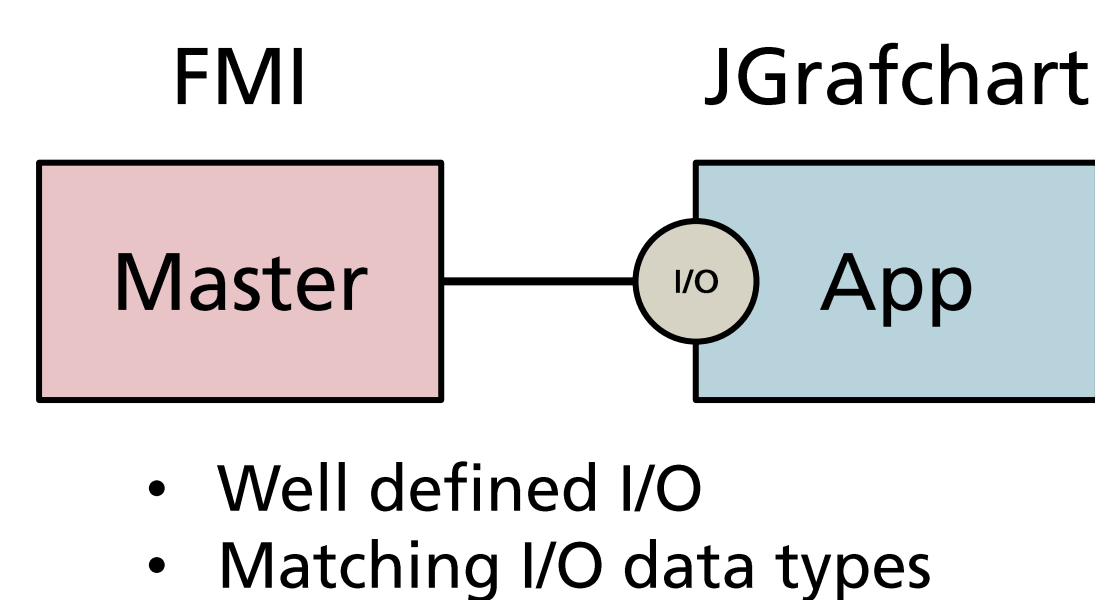


- Coordinates the co-simulation
- Decides step sizes
- Slave communication
- Executes step when requested
- Capabilities: redo step, variable step size, ...
- Requests desired step size

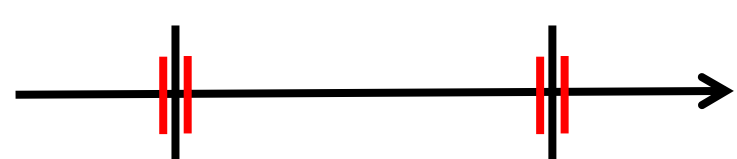


FMI co-simulation

JGrafchart with FMI Support



- Well defined I/O
- Matching I/O data types

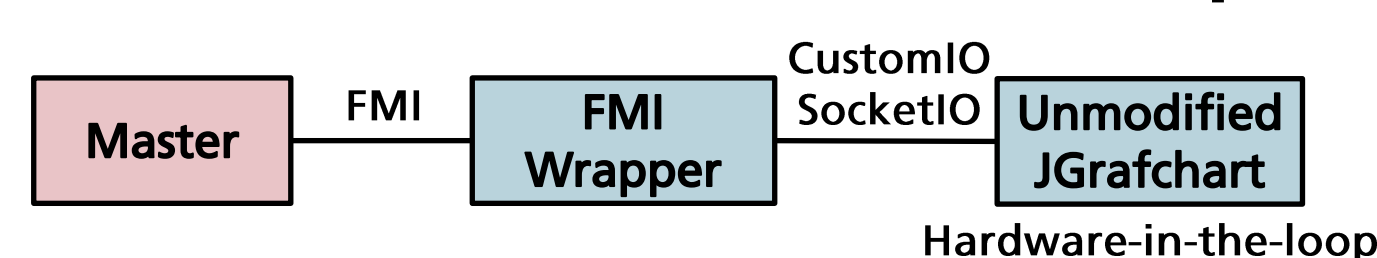


- Ideally, scan cycle execution takes no time
- Ideally, communication points right before and after each scan cycle execution
- Small step size should always be good enough

Current Limitations

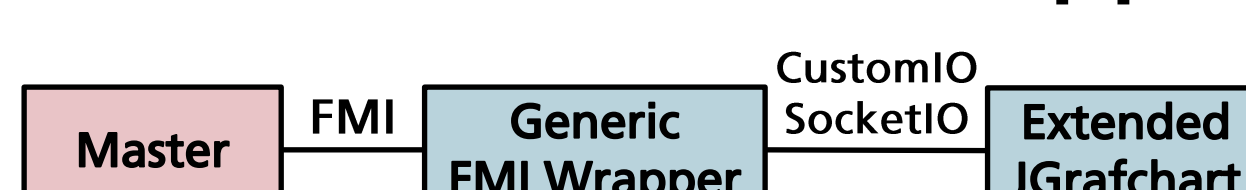
- Wall clock time only
- Cannot get/set App state ⇒ Cannot redo step

1. Hardware-in-the-loop



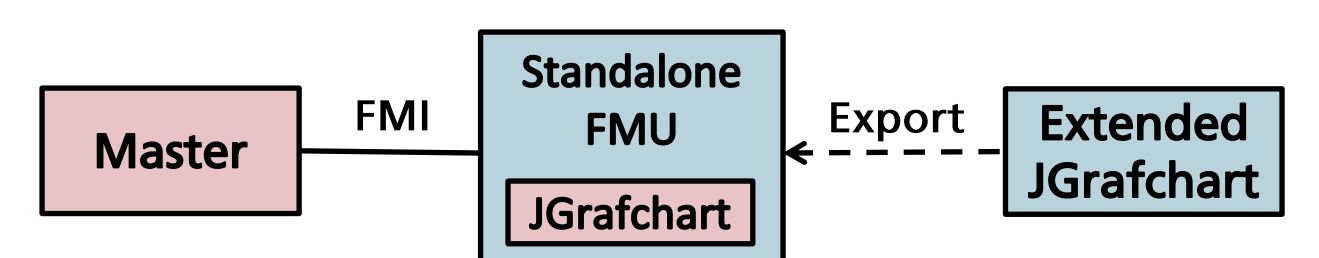
- + Unmodified JGrafchart
- Run in wall clock time
- Slow dynamics ⇒ lengthy simulation

2. Generic FMI Wrapper



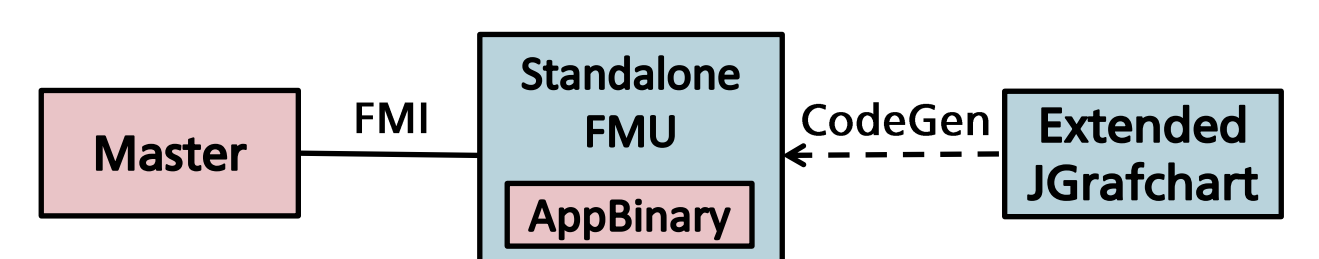
- Extended with support for external clocks
- + Not in wall clock time
- + Can add playback to inspect scan cycles
- Moderate effort

3. FMU Resource



- Extended with FMU packaging support
- + Self-contained
- Large FMUs

4. Code Generation



- Extended with support for FMU generation
- + Self-contained
- + Small FMU footprint
- Large effort
- Currently fragile

Future Work

So far only conceptual. Next step is prototyping, starting with hardware-in-the-loop and then improving to (at least) Generic FMI Wrapper.

References

- [1] A. Theorin, *Adapting Grafchart for Industrial Automation*, Licentiate Thesis ISRN LUTFD2/TFRT-3260-SE, Department of Automatic Control, Lund University, Sweden, 2013-05
- [2] FMI Development Group, *Functional Mockup Interface for Model Exchange and Co-Simulation – 2.0 Release Candidate 1*, Tech. rep. Modelica Association, 2013-10

Acknowledgments

Financial support from the VINNOVA-FFI project LISA is gratefully acknowledged. The authors are members of the LCCC Linnaeus Center and the eLLIIT Excellence Center at Lund University.