Industrial Cyber-Physical Systems

A Tool Integration Approach for Architectural Exploration of Aircraft Electric Power Systems

Hokeun Kim, Liangpeng Guo, Edward A. Lee
and Alberto Sangiovanni-Vincentelli

EECS, University of California, Berkeley

The 1st IEEE International Conference on
Cyber-Physical Systems, Networks, and Applications

Taipei, Taiwan August 19 - 20, 2013

Sponsored by the iCyPhy Research Consortium, a collaboration of
UC Berkeley, Caltech, IBM, and United Technologies.

Introduction

- Aircraft electric power system (EPS)
  - Generation, conversion and distribution of power for aircraft utilities
  - Safety-critical cyber-physical system
  - Consists of power generators, buses, contactors, loads and sensors
  - Becoming increasingly more complex
- References

  - K. Emadi and M. Ehsani, “Aircraft power systems: technology, state of the art, and future trends,” Aerospace and Electronic Systems Magazine, IEEE, vol. 15, no. 1, pp. 28–32, 2000.
  - L. Guo, M. Maasoumy, M. Mozumdar, P. Nuzzo, N. Ozay, U. Topcu, H. Xu, R. Murray and A.Sangiovanni-Vincentelli, "Aircraft Electric Power System: Description, Specifications and Design Challenges", MuSyC DSCS internal report, March 2012, unpublished

Introduction (cont'd)

- Characteristics of modern safety-critical cyber-physical systems
  - Consist of heterogeneous components
  - Complex systems both in functionalities and underlying architectures
  - Timing behavior is part of correctness
  - Validation of reliability is critical

Introduction (cont'd)

- Design challenges
  - How can we model heterogeneous components in safety-critical cyber-physical systems together?
  - How can we cope with architectural exploration problem with complex functionalities?
  - How can we validate timing behavior in advance?

Introduction (cont'd)

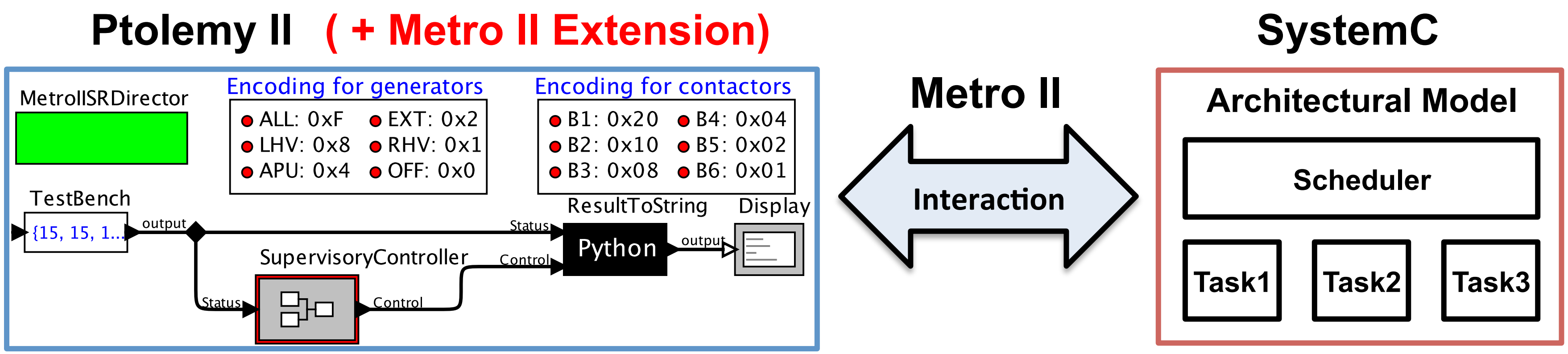
- Tool integration approach
  - Creating a platform for architectural exploration of safety-critical cyber-physical systems <u>by integrating Ptolemy II and Metro II</u>
- Ptolemy II
  - A system design framework supporting experimentation with multiple heterogeneous models of computation (e.g. DE, SDF, SR, etc.)
- Metro II
  - Design environment for platform based design where the mapping can be easily changed and thus suitable for architectural exploration

Introduction (cont'd)

- Design challenges revisited
  - How can we model heterogeneous components in safety-critical cyber physical systems together?
    - By using Ptolemy II that supports multiple models of computation
  - How can we cope with architectural exploration problem with complex functionalities?
    - By decoupling functional aspects from architectural aspects using Metro II
  - How can we validate timing behavior in advance?
    - By running co-simulation on the integrated platform of Ptolemy II and Metro II

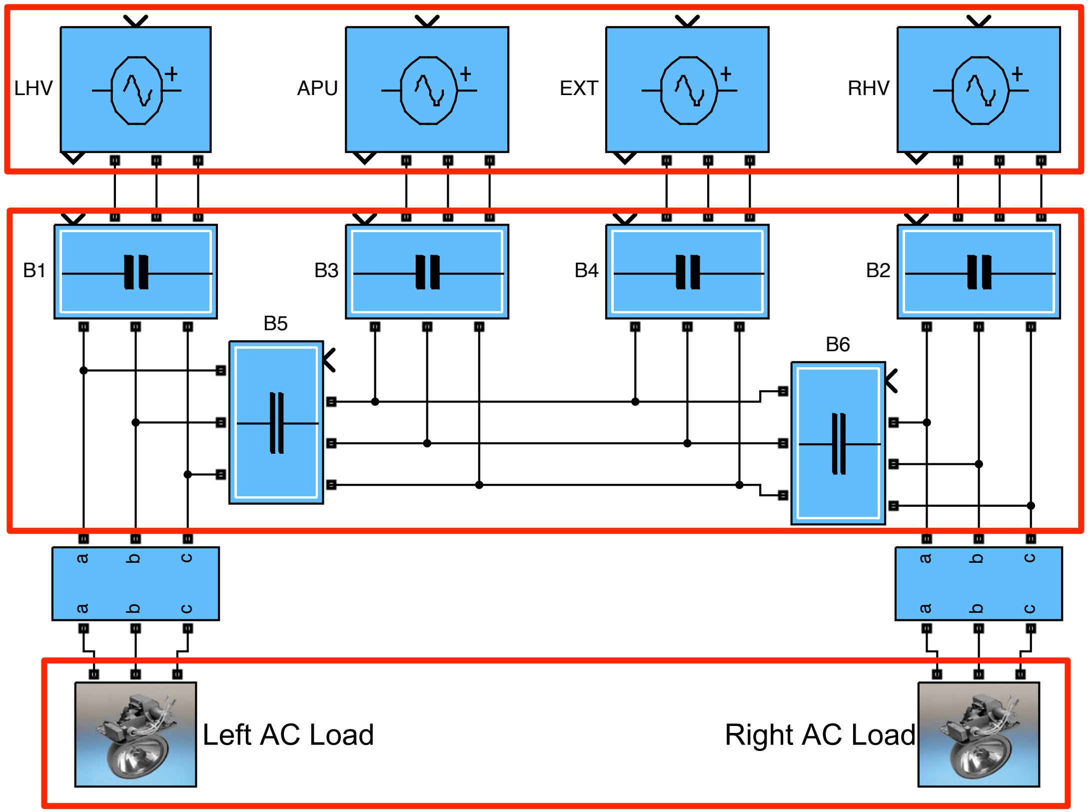
Approach

- Approach overview



Functional Model

- Aircraft EPS specification



Generators
- **Generate AC power**
- **May have faulty behaviors**
- **Main & backup generators**

Contactors
- **Transfer power from generators to loads**
- **Set up control paths**

AC Loads
- **Always need to be powered by exactly one generator**
- **Can be powered off while generators are replaced**

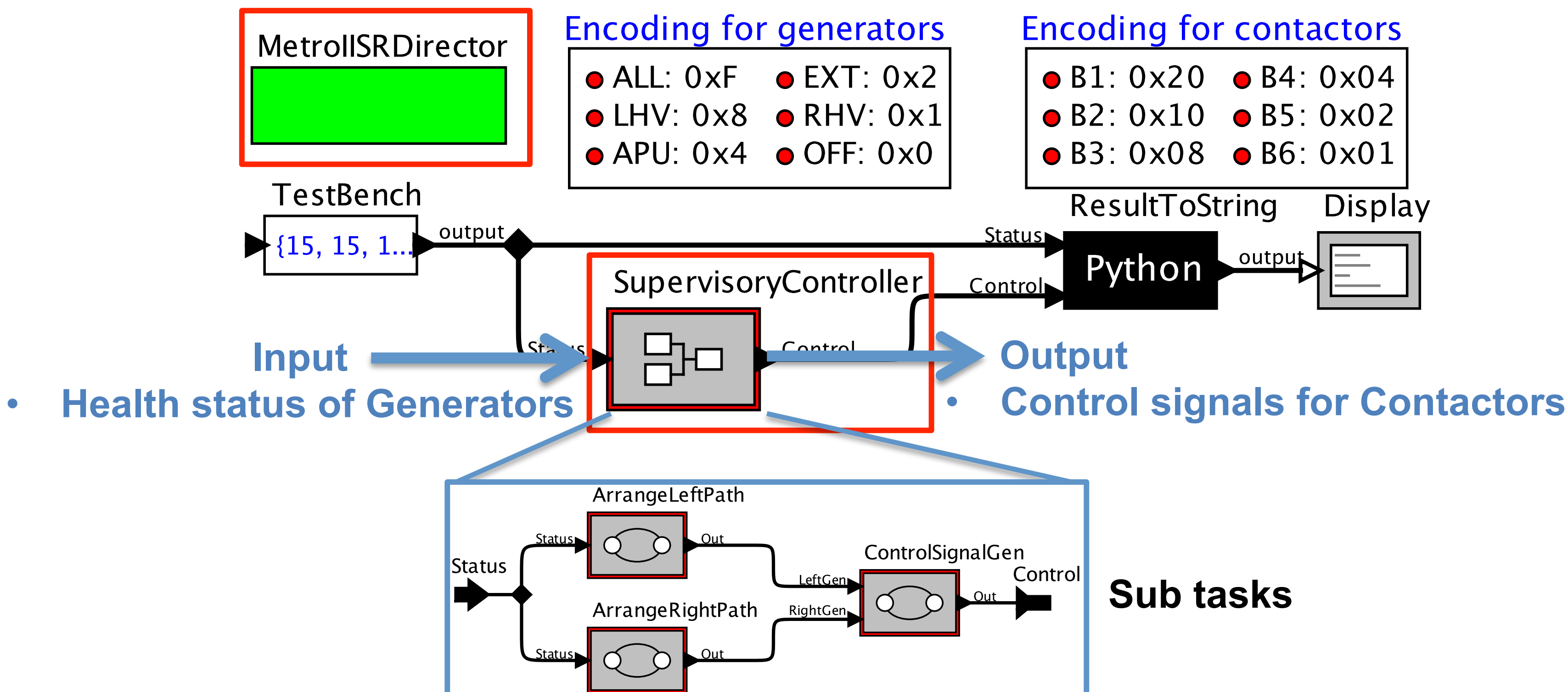
Courtesy : P. Nuzzo

Functional Model (cont'd)

- A supervisory controller for aircraft EPS (Ptolemy II)

Director

- **Implements <u>Synchronous / Reactive</u>**
- **Metro II extension**



Sub tasks

Functional Model (cont'd)

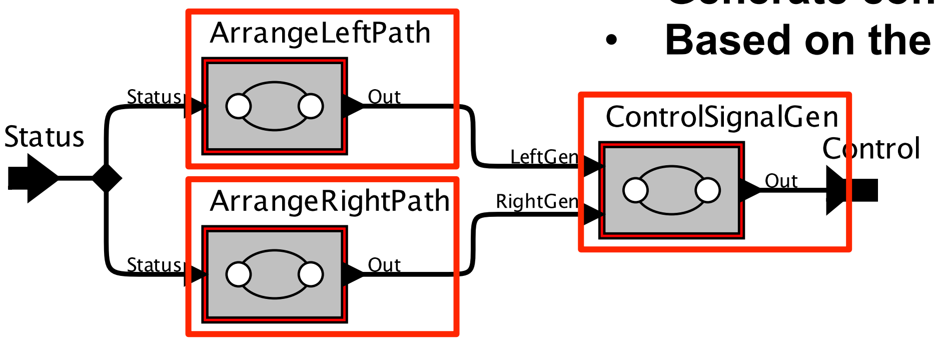
- Tasks inside of the supervisory controller

ArrangeLeftPath (ALP)

- **Selects a generator for Left AC Load**
- **Based on health status**

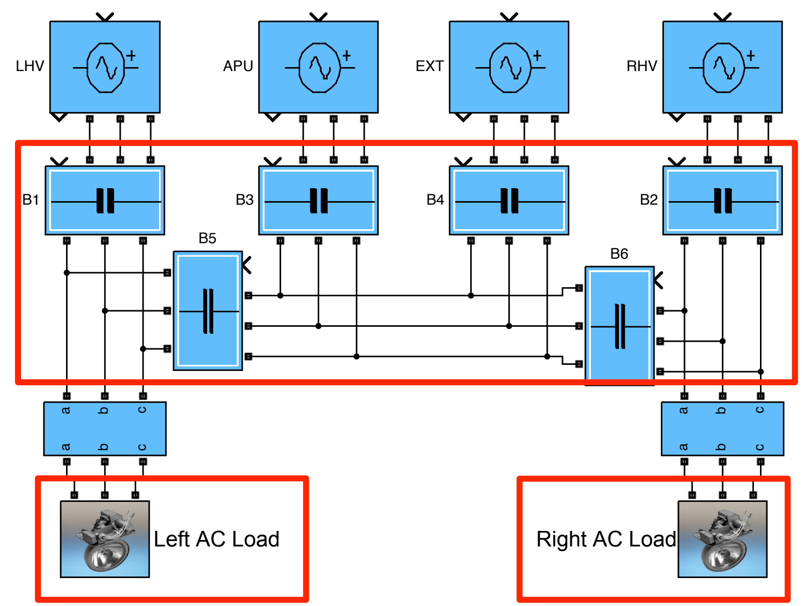
ControlSignalGen (CSG)

- **Generate control signals for contactors (B1 ~ B6)**
- **Based on the generator selections of ALP and ARP**



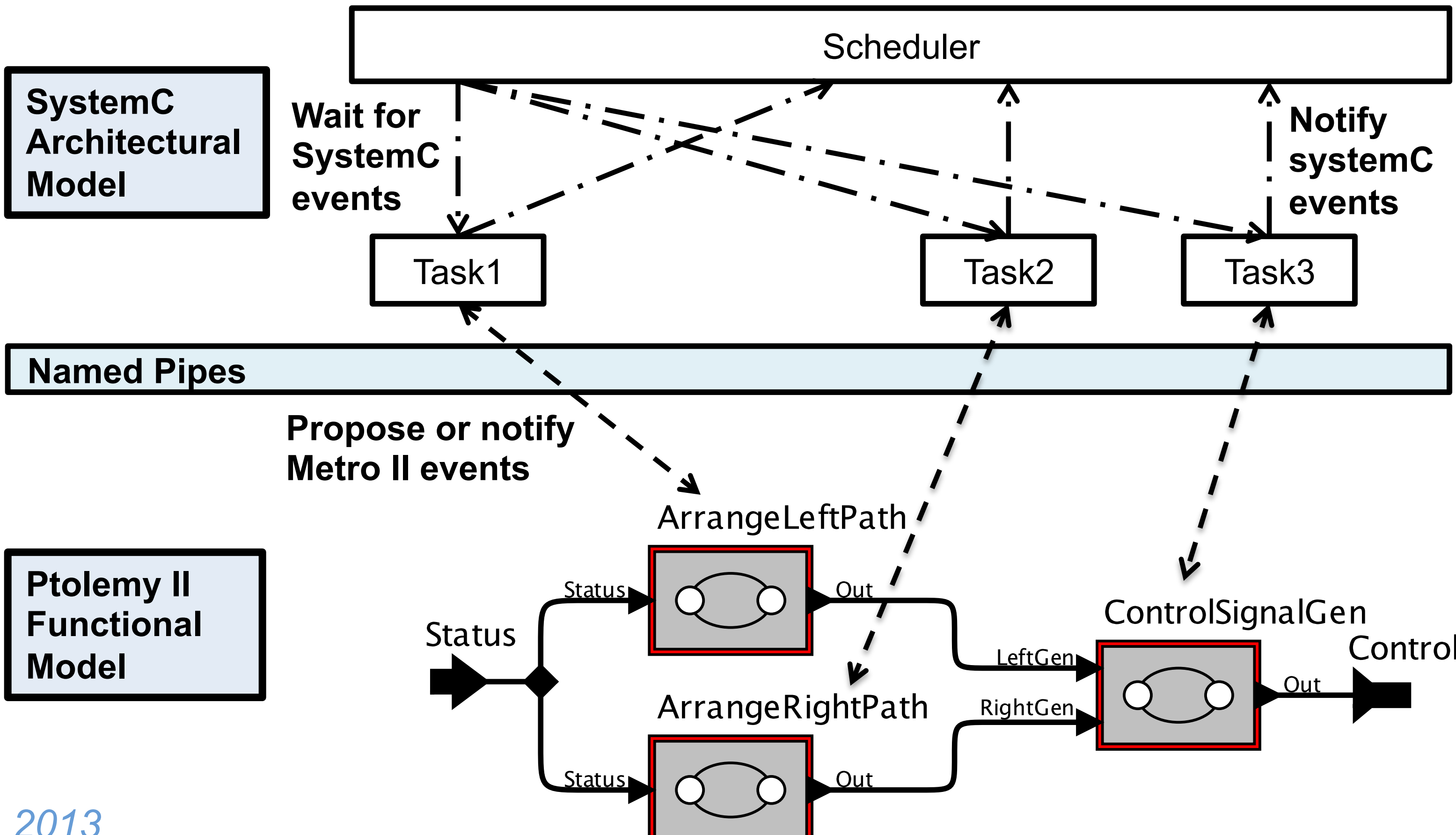
ArrangeRightPath (ARP)

- **Selects a generator for Right AC Load**

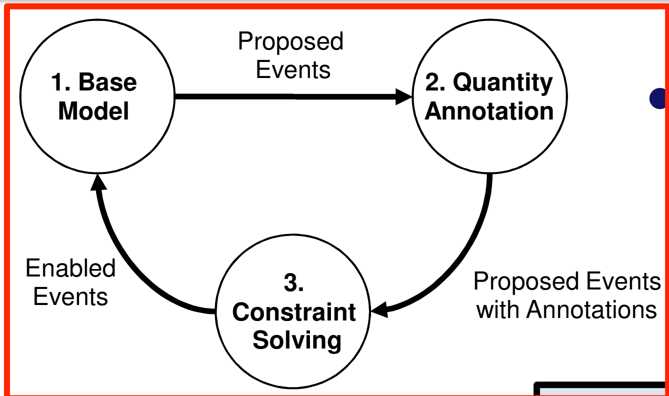


Architectural Model

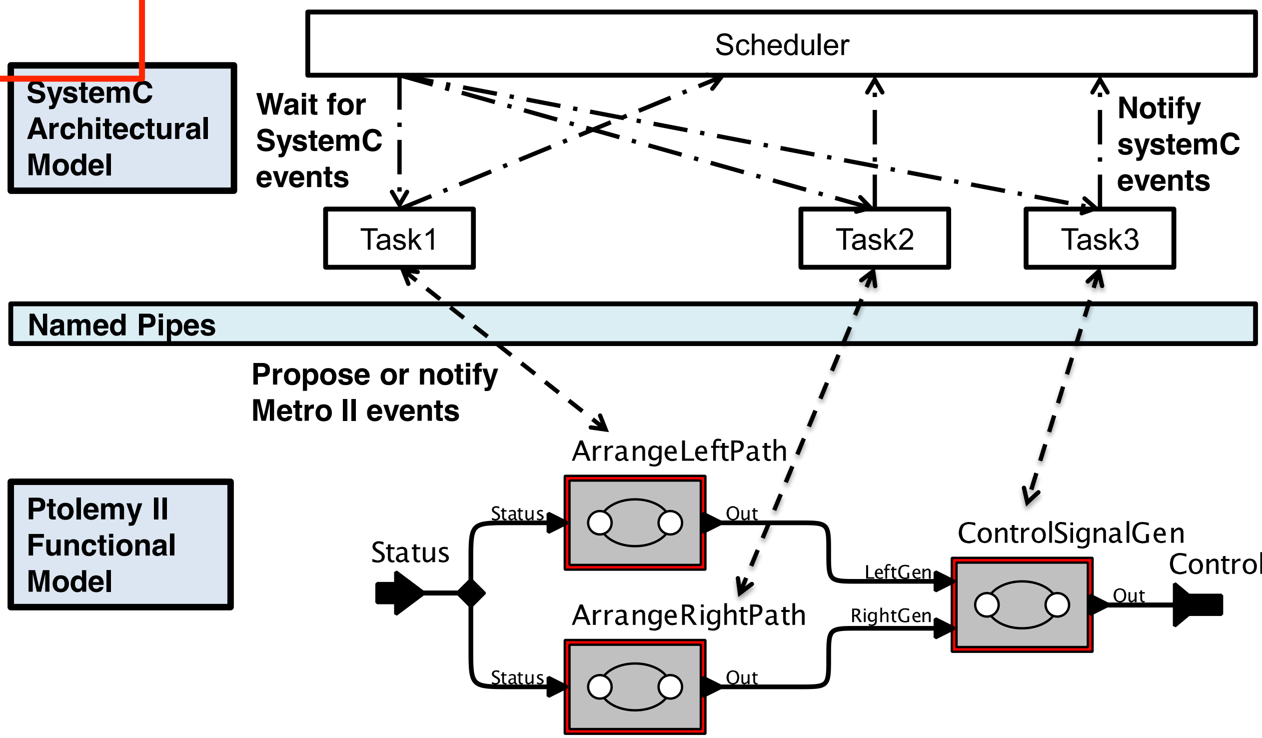
- Architectural model overview
  & interaction with the functional model



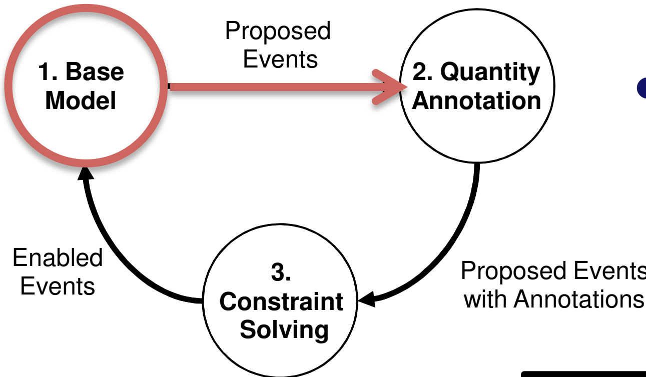
Architectural Model (cont'd)



- Metro II execution semantics & Co-simulation flow

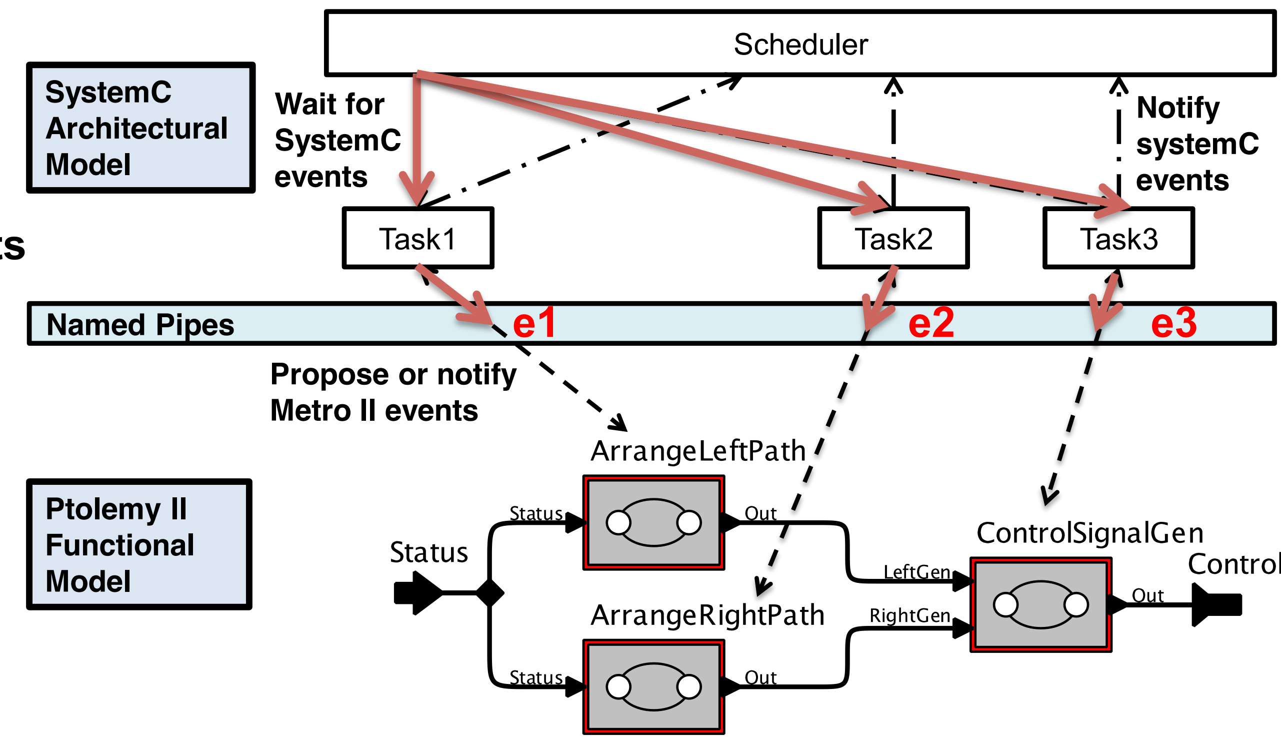


Architectural Model (cont'd)

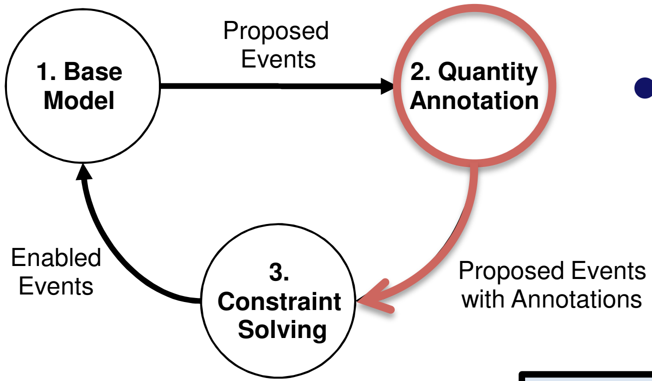


- Metro II execution semantics & Co-simulation flow

e1, e2, e3 – Metro II events

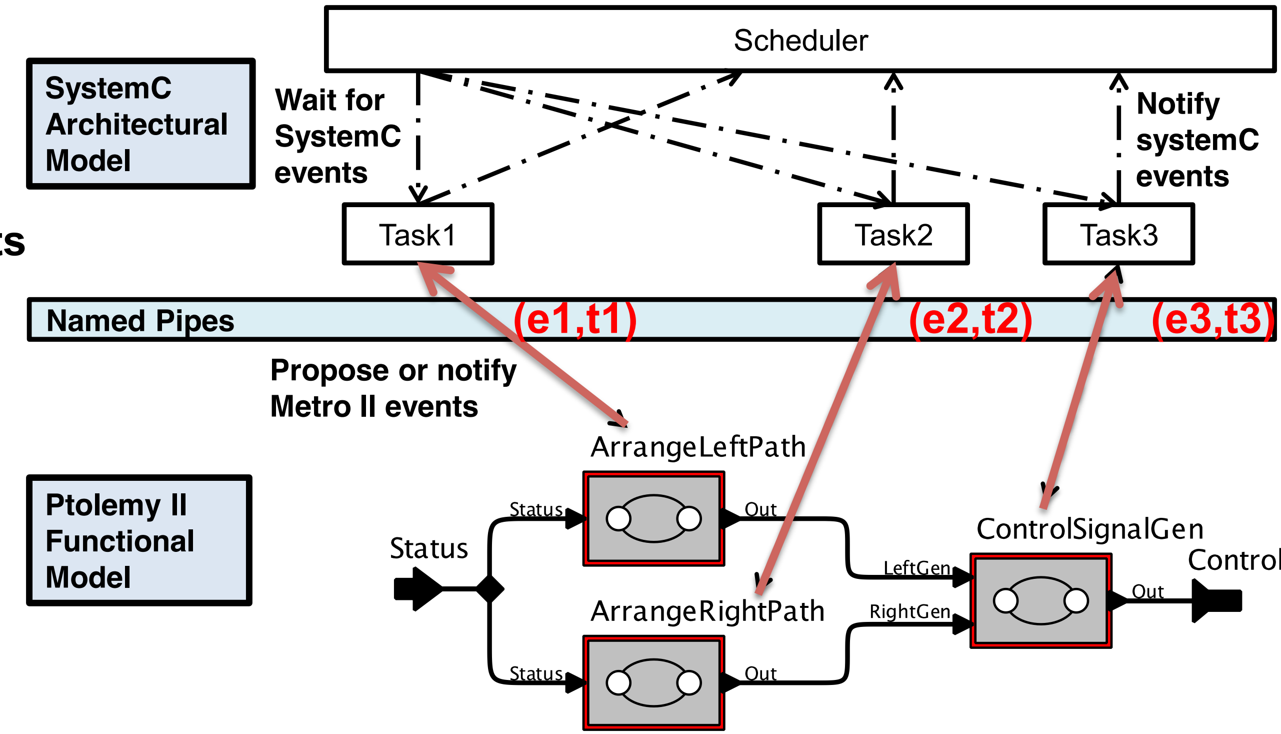


Architectural Model (cont'd)

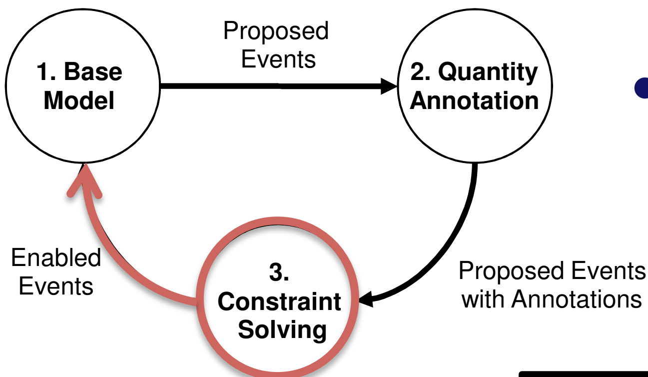


- Metro II execution semantics & Co-simulation flow

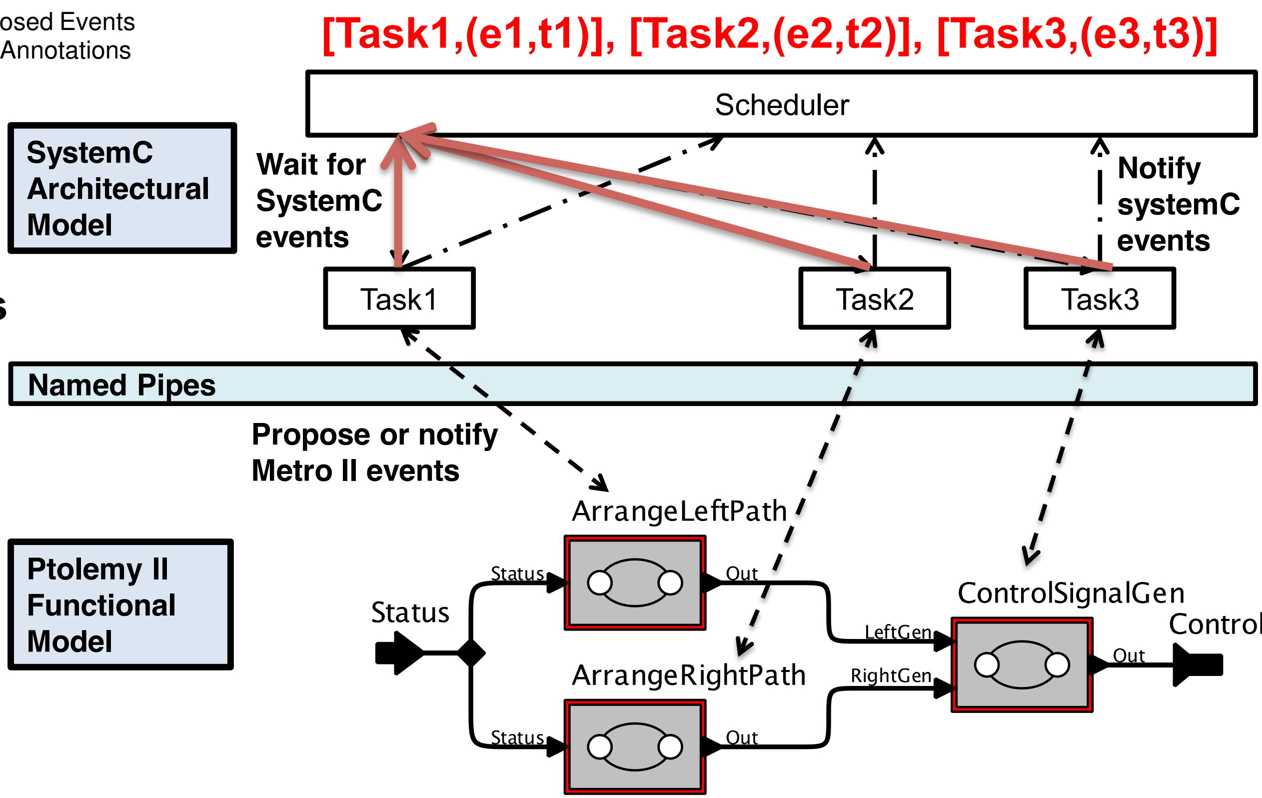
e1, e2, e3 – Metro II events
t1, t2, t3 – Global clock



Architectural Model (cont'd)

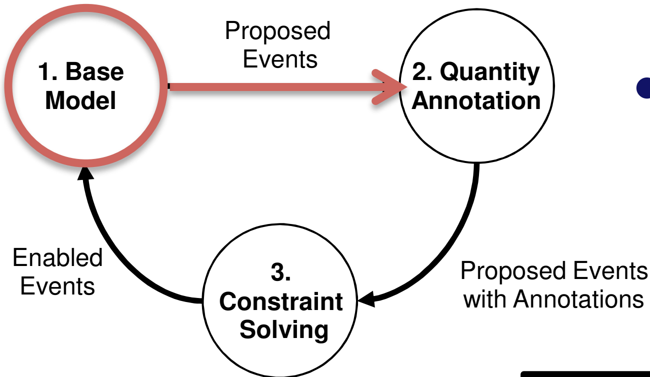


- Metro II execution semantics & Co-simulation flow



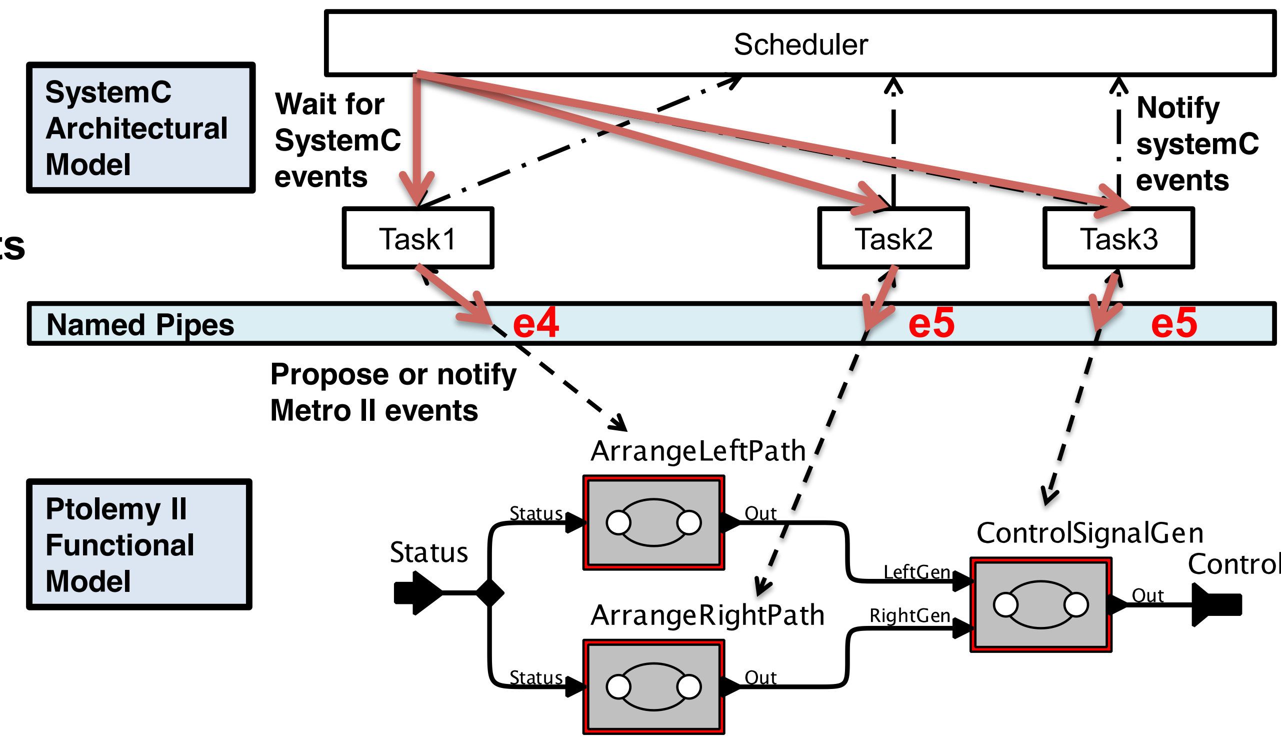
e1, e2, e3 – Metro II events
t1, t2, t3 – Global clock
Global clock += (scheduling overhead +execution time of tasks +synch overhead + etc.)

Architectural Model (cont'd)



- Metro II execution semantics & Co-simulation flow

e4, e5, e6 – Metro II events



Architectural Model (cont'd)

- Architectural parameters
  - Scheduling overhead
  - Priority of tasks
  - Speed of processing elements (or execution times of tasks)
  - Parallelization of independent tasks
  - Synchronization overhead for parallelized tasks

Experiments and results

- Example architectural alternatives

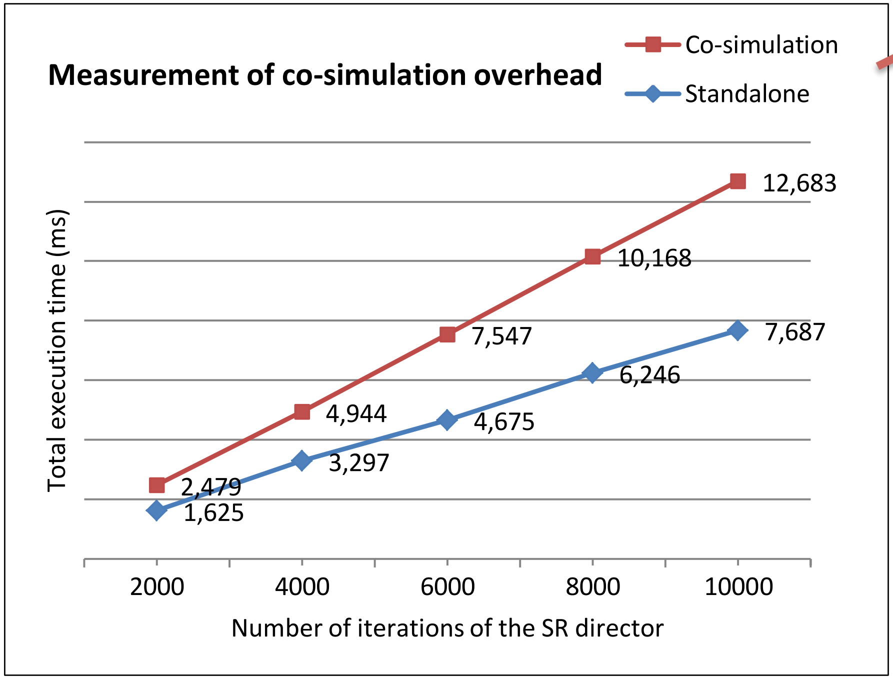
Candidate #	Scheduling Overhead (ns)	Execution Time (ns) ALP/ARP/CSG	Parallelization of ALP & ARP	Synch Overhead (ns)
1	10	40/45/20 **Shortest (=Fastest)**	No **No Parallelism**	-
2	10	65/70/40 **Slower Than #3**	Yes **Parallel Processing**	5 **Less Overhead Than #3**
3	10	50/55/30	Yes	15

- Results
  – Ten iterations of functional model with a given test bench

Candidate #	Total Execution Time (ns)
1	1150
2	1250
3	1100 **Least Total Execution Time**

Experiments and results (cont'd)

- Measuring co-simulation overhead
  - Total simulation time of Ptolemy II model
    - Co-simulation VS Standalone (Ptolemy II only)



- **Linear overhead**
- **1.58x execution time**

➔Potential for Scalability

Conclusion

- Summary
  - Co-simulation environment supporting performance prediction and comparison of architectural candidates for safety-critical cyber physical systems
  - Through a tool integration approach with
    - Ptolemy II – Supports heterogeneous MoCs
    - Metro II – Decouples the modeling of functional aspects and architectural aspects

Conclusion (cont'd)

- Future work
  - Functional model
    - More complex safety-critical system examples
    - Examples with heterogeneous directors (MoCs)
  - Architectural model
    - Creating general architectural models
    - Considering more architectural parameters (e.g. memory access overhead, I/O operation overhead)

Q & A

Thank you!