

Models of Computation

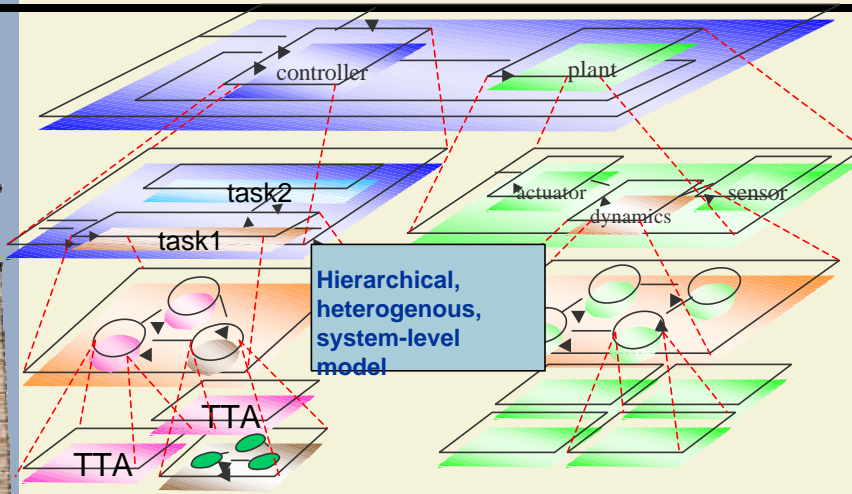


Edward A. Lee
UC Berkeley

With thanks to the entire Ptolemy Team.

Ptolemy Miniconference
Berkeley, CA, March 22-23, 2001

Components and Composition



Ptolemy Miniconference, Berkeley, 2

Thomas Kuhn, originator of the paradigm paradigm



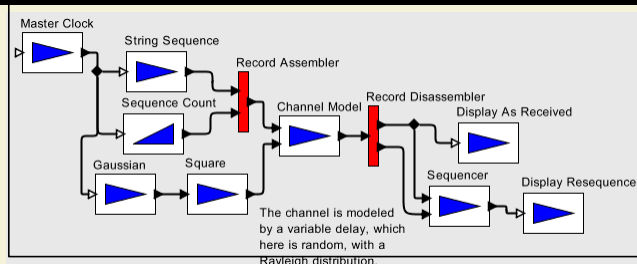
Models of Computation



- **What is a component? (ontology)**
 - States? Processes? Threads? Differential equations? Constraints? Objects (data + methods)?
- **What knowledge do components share? (epistemology)**
 - Time? Name spaces? Signals? State?
- **How do components communicate? (protocols)**
 - Rendezvous? Message passing? Continuous-time signals? Streams? Method calls? Events in time?
- **What do components communicate? (lexicon)**
 - Objects? Transfer of control? Data structures? ASCII text?

Ptolemy Miniconference, Berkeley, 3

A Laboratory for Exploring Models of Computation

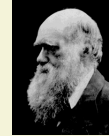


Ptolemy II – Java based, network integrated

- A realization of a model of computation is called a “domain.” Multiple domains can be mixed hierarchically in the same model.

Ptolemy Miniconference, Berkeley, 4

Survival of the fittest is the *only* reasonable way to choose among these.



Domains

- CSP – concurrent threads with rendezvous
- CT – continuous-time modeling
- DE – discrete-event systems
- DDE – distributed discrete-event systems
- DT – discrete time (cycle driven)
- FSM – finite state machines
- Giotto – time driven cyclic models
- GR – graphics
- PN – process networks
- SDF – synchronous dataflow
- xDF – other dataflow

Each is realized as a director and a receiver class in Ptolemy II

Each of these defines a component ontology and an interaction semantics between components. There are many more possibilities!

Ptolemy Miniconference, Berkeley, 5

Heterogeneity

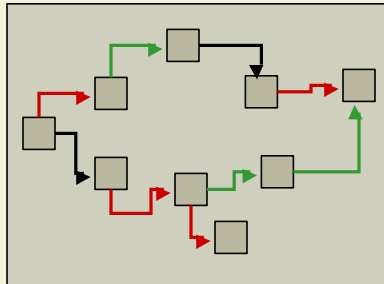
- Mixed-signal systems
 - DE + CT
 - DT + CT
- Hybrid systems
 - FSM + CT
- *Charts – Modal Model
 - FSM + anything
- Complex systems
 - Resource management
 - Signal processing
 - Real time

Ptolemy Miniconference, Berkeley, 6

Hierarchical Heterogeneity vs. Amorphous Heterogeneity

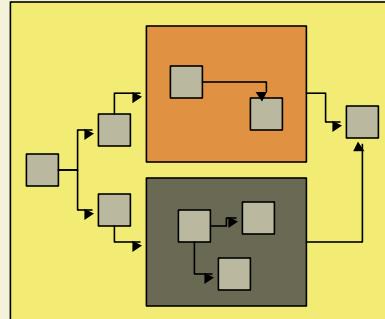


Amorphous



Color is a communication protocol only, which interacts in unpredictable ways with the flow of control.

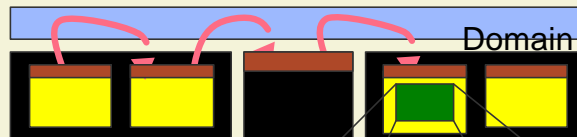
Hierarchical



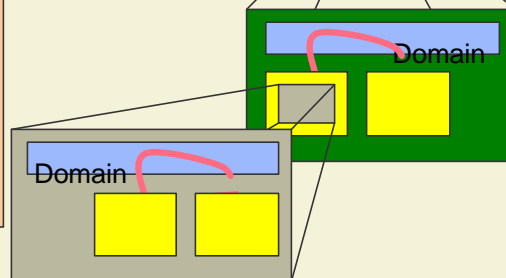
Color is a domain, which defines both the flow of control and interaction protocols.

Ptolemy Miniconference, Berkeley, 7

Hierarchical, Compositional Models

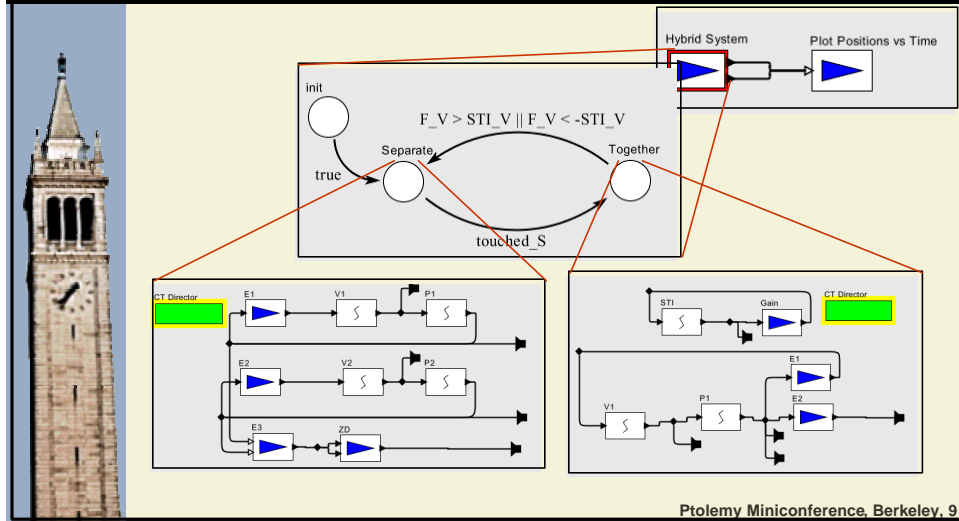


Schedulers (Directors) are nested hierarchically, each interacting with components through the Executable interface. Directors themselves implement this same interface, so the model is compositional.



Ptolemy Miniconference, Berkeley, 8

Example: Hybrid System Models CT + FSM



Key Advantages

- Domains are specialized
 - lean
 - targeted
 - optimizable
 - understandable
- Domains are mixable (hierarchically)
 - structured
 - disciplined interaction
 - understandable interaction

Ptolemy Miniconference, Berkeley, 10

What we are trying to avoid:



Software architecture may end up like this as it scales up.

Poor common infrastructure.
Weak specialization.
Poor resource management and sharing.
Poor planning.

Ptolemy Miniconference, Berkeley, 11

Also to Avoid



Elegant, unified, and beautiful, but rigid, inflexible, and difficult to adapt. Plus, it takes 100 years to build.



Ptolemy Miniconference, Berkeley, 12

Elegant Federation



Source: Kaplan McLaughlin Diaz, R. Rappaport, Rockport, 1998

Elegant federation of heterogeneous models.

Two Rodeo Drive, Kaplan, McLaughlin, Diaz

Ptolemy Miniconference, Berkeley, 13