Manufacturing Engineering

Ontological Behavior Modeling for UML:

Behavior as Composite Structure

Conrad Bock, U.S. National Institute of Standards and Technology
James Odell, Computer Sciences Corporation
Tim Weilkiens, oose Innovative Informatik
James Baker, BAE Sytems
Antoine Lonjon, MEGA

September 13, 2009 (revised July 16, 2010 for DM2 WG)

NIST

Overview

- Semantics generally.
- Ontological approaches.
- Behaviors as composites
  - Generalization
  - Composition
  - Events
  - Participants
  - Flows (object flow and messaging)
    - External Participants
    - Flow Ordering
    - Composition with Flows and Participants (Bindings)

“Semantics” Defined

- **When using a language:**
  - **How systems will behave (M0) when they are built according to a user model (M1).**
- **When specifying a language:**
  - **How to tell which M0 system behaviors conform to which M1 models.**

Standardized Semantics: Goal

- **Multiple users of the same model …**
  - **… should have the same expectation of how systems will behave when built according to model.**
- **Multiple implementers of same modeling language tools …**
  - **… should have the same expectations as the users.**

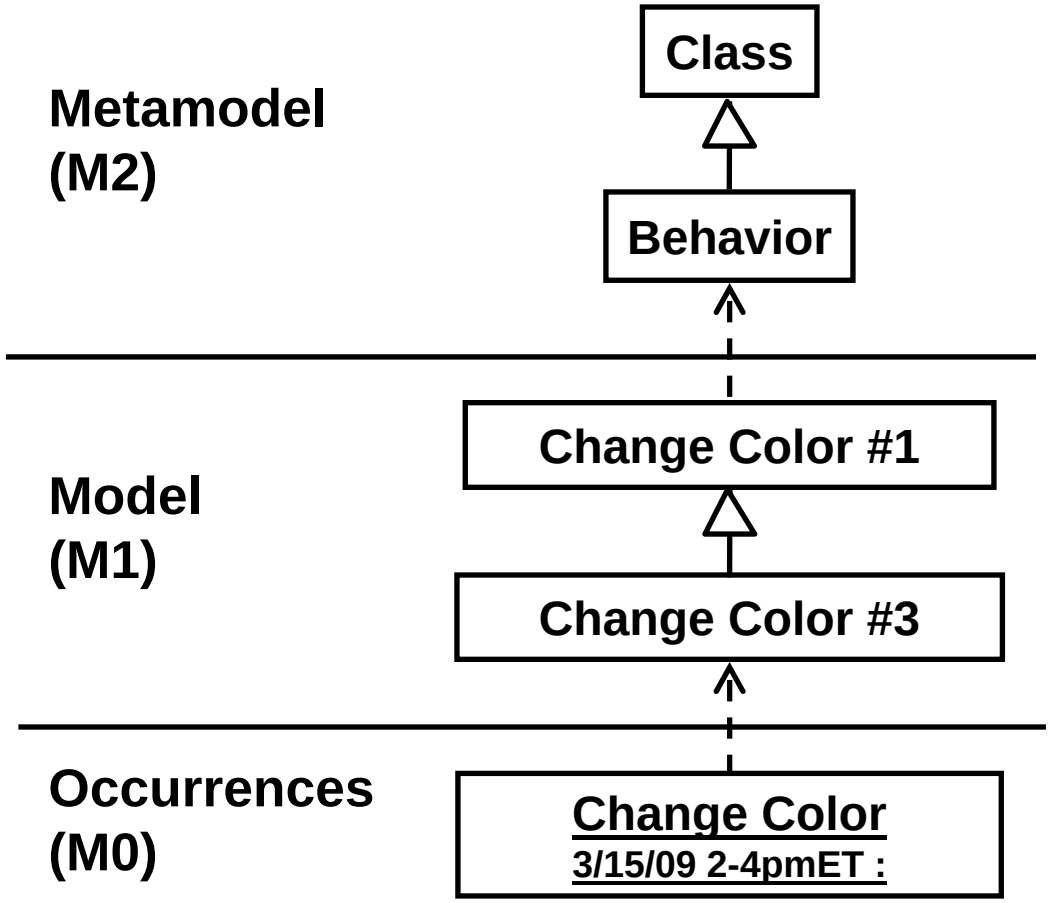
Standardized Semantics: How?

- **Common practice and communication.**
- **Informal documentation with examples.**
- **Reference implementations.**
- **Compliance testing.**
- **Formal documentation, sometimes mathematical.**

Ontological Specifications

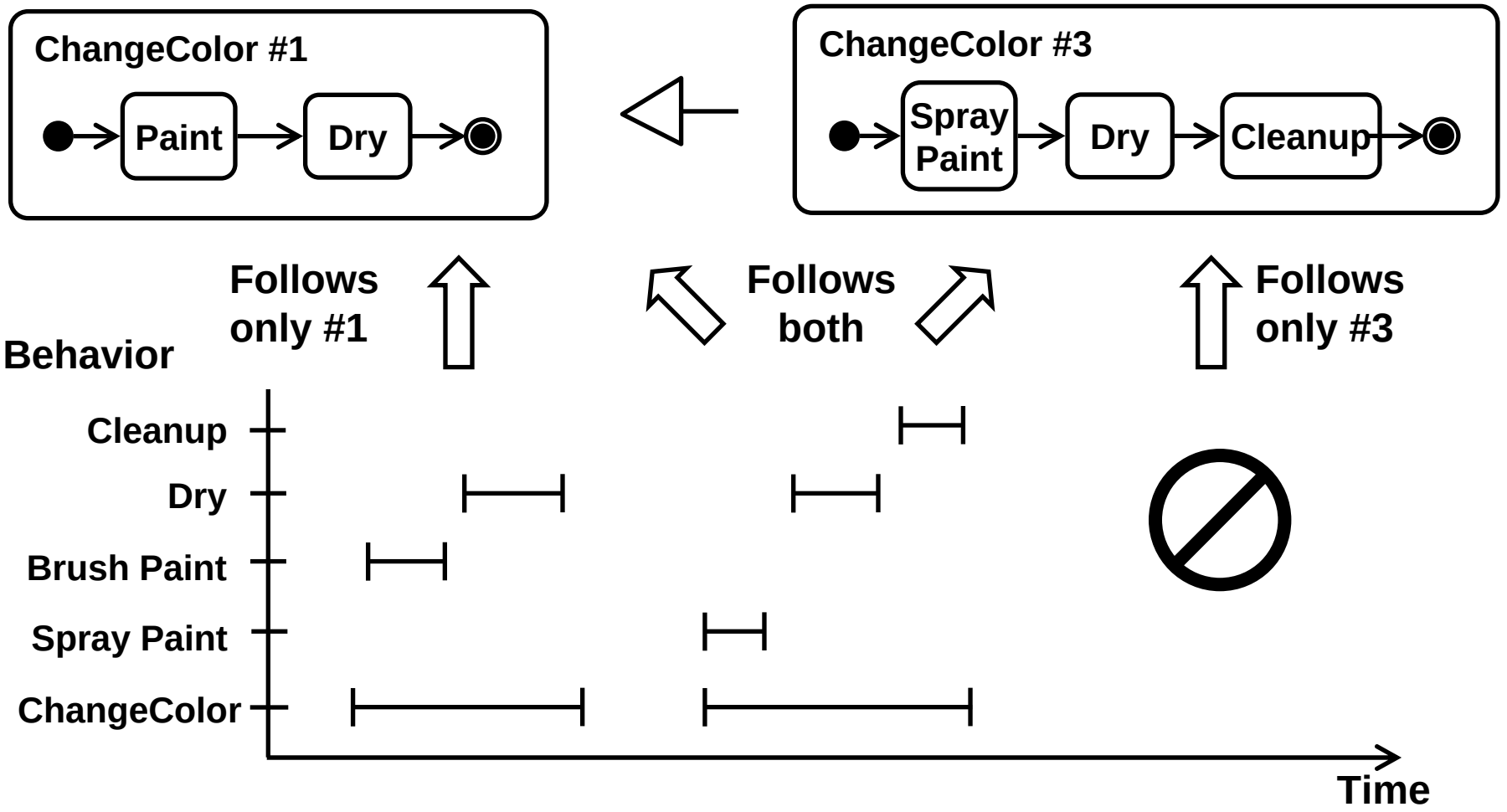
- **Formality a matter of degree:**
  - **Ratio of structure to informal text.**
- **Ontological approach:**
  - **Build up thin layers of structural elements that have simple definitions.**
- **Advantages:**
  - **Reference implementations and compliance tests not needed, but easier to construct.**
  - **Helps consolidate common practice and communication.**
  - **Already used at OMG: specializing metamodels, and profiles.**

Example: UML Behaviors



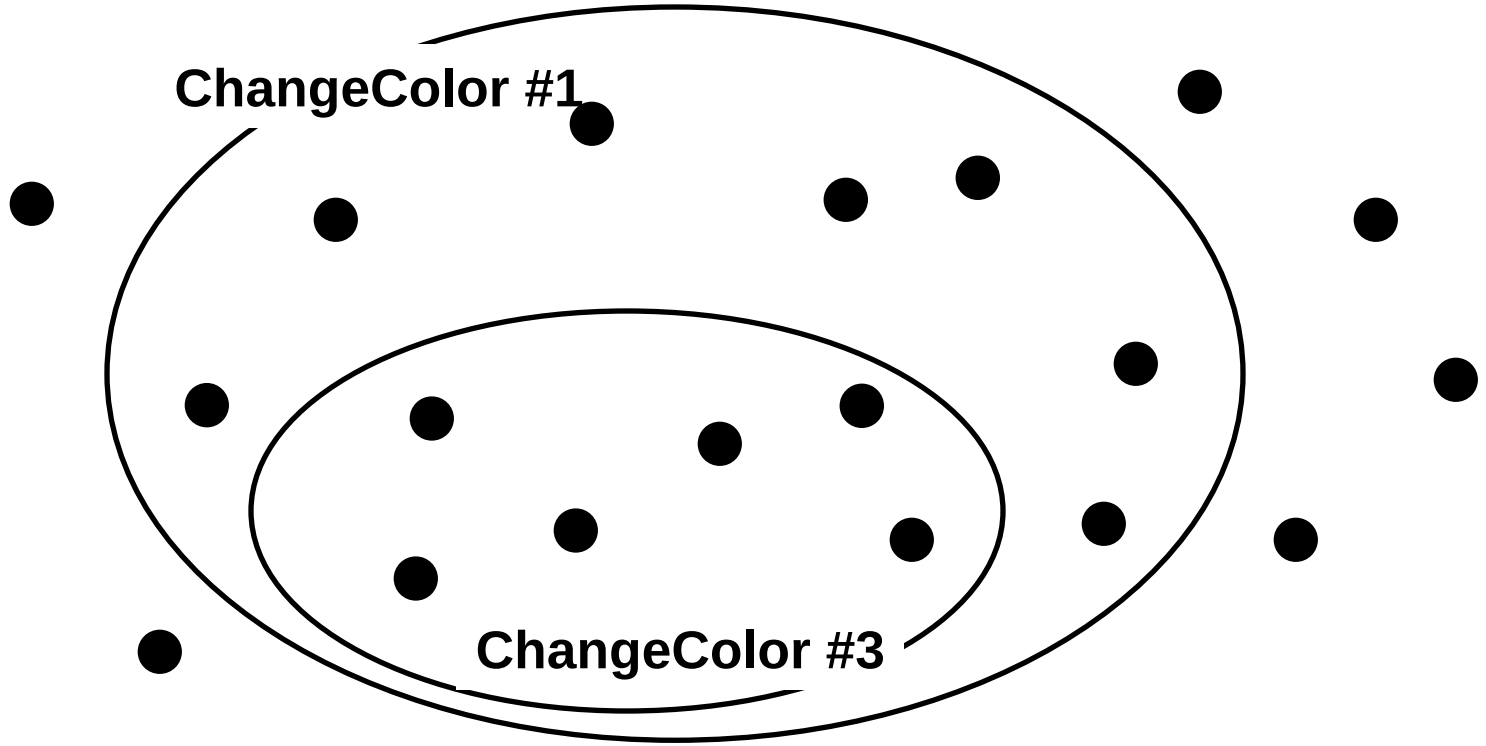
- **Behaviors are**
  - **layered on Classes at M2.**
  - **specialized at M1.**
  - ***occur* (execute, are performed) at M0.**

Behavior Generalization



- **UML definition of generalization:**
  - **Every occurrence (instance) of the specialized behavior (class) is an occurrence of the general behavior.**

Behavior Generalization



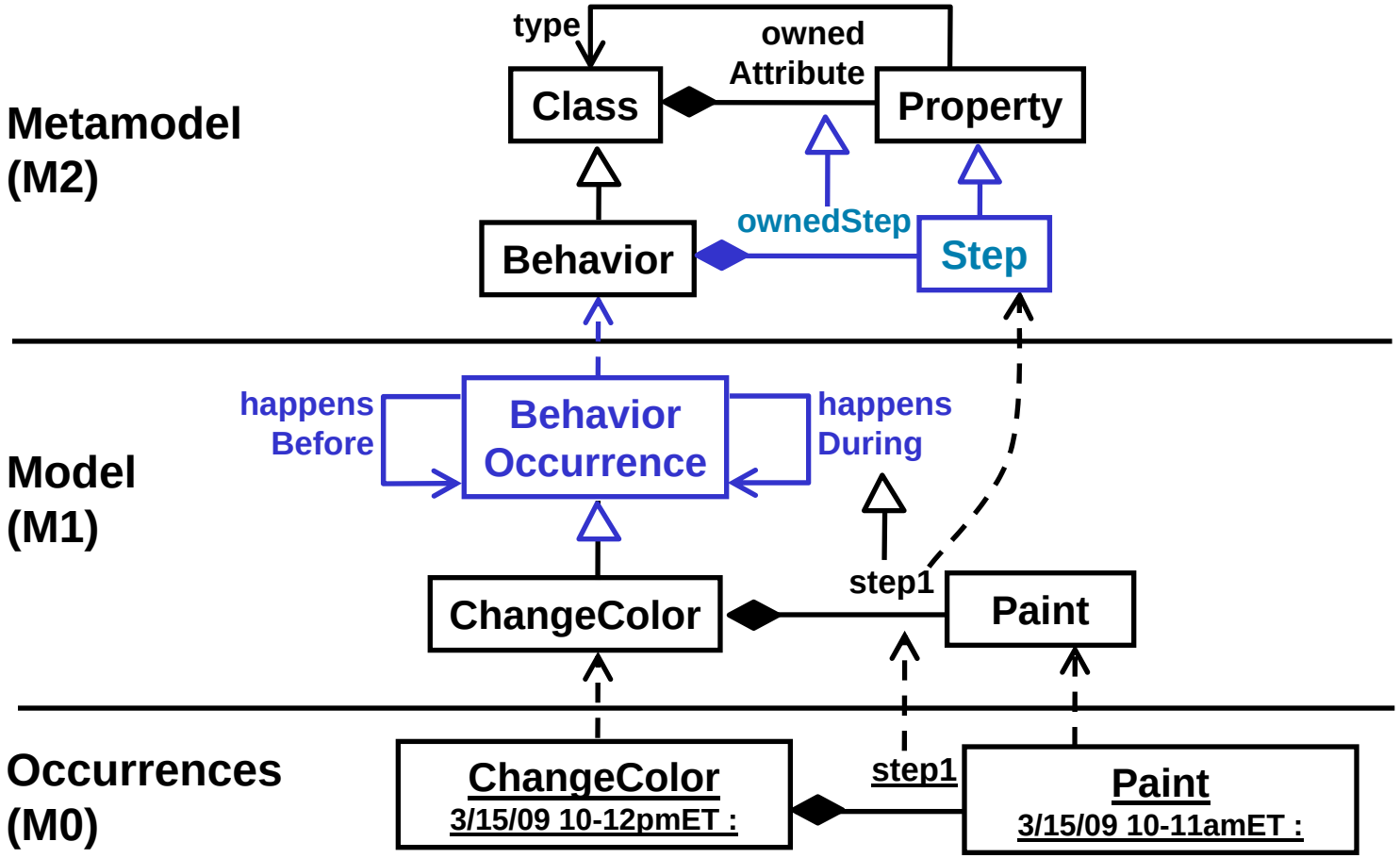
● = occurrence

- Venn diagram illustration of previous example.

Behavior Composition

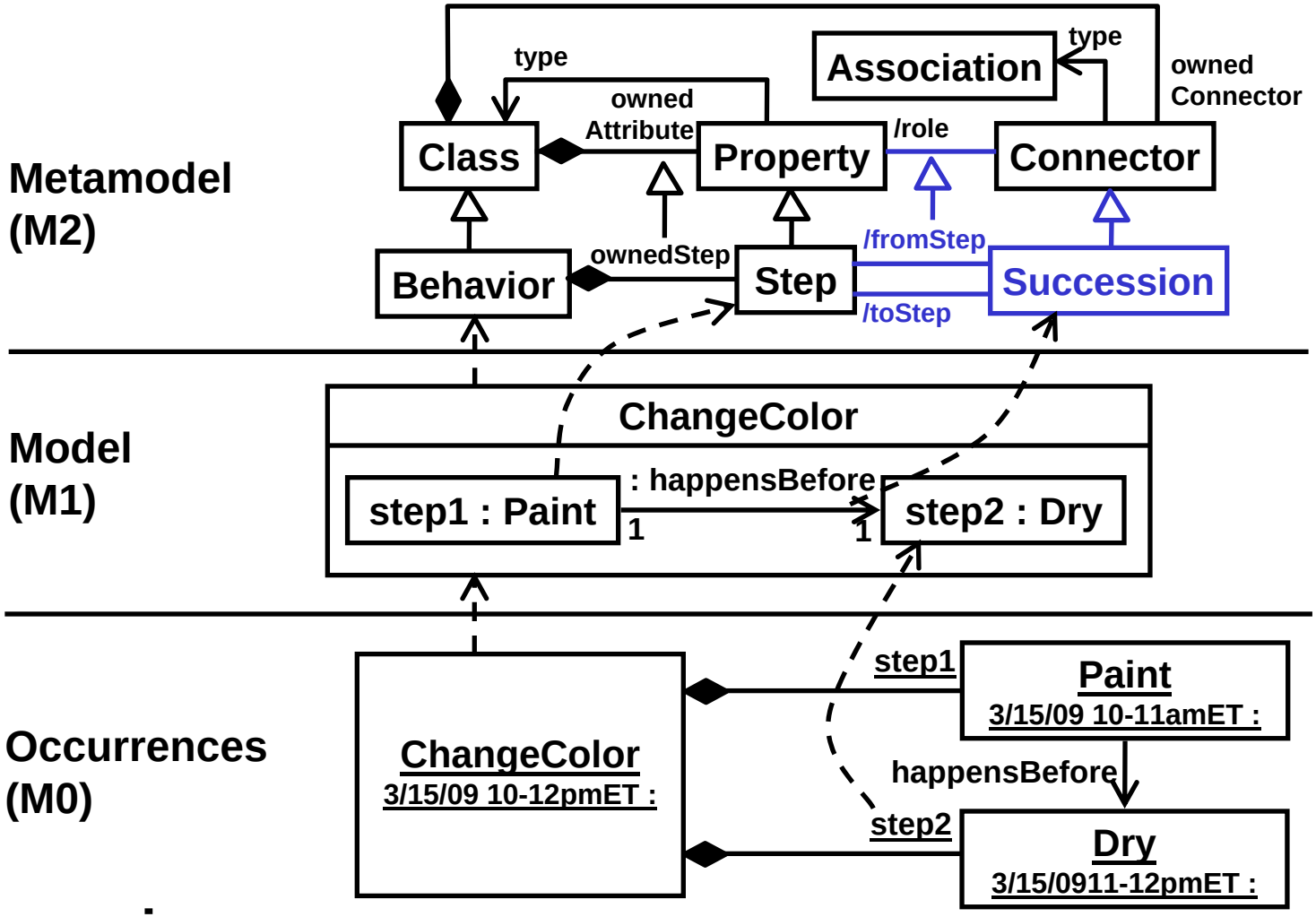
- **Whole-part**
  - **Activities have actions.**
  - **State machines have state behaviors and submachines.**
  - **Interactions have interaction uses, messages and actions.**
- **Part-part**
  - **Activities have control flow between actions.**
  - **State Machines have transitions between states.**
  - **Interactions have general orderings between messages.**

Whole-part for Behaviors



- **Steps:**
  - layered on Properties at M2.
  - typed by behaviors at M1, and specialized from general temporal relation.
  - have “suboccurrences” as values at M0.

Part-part for Behaviors

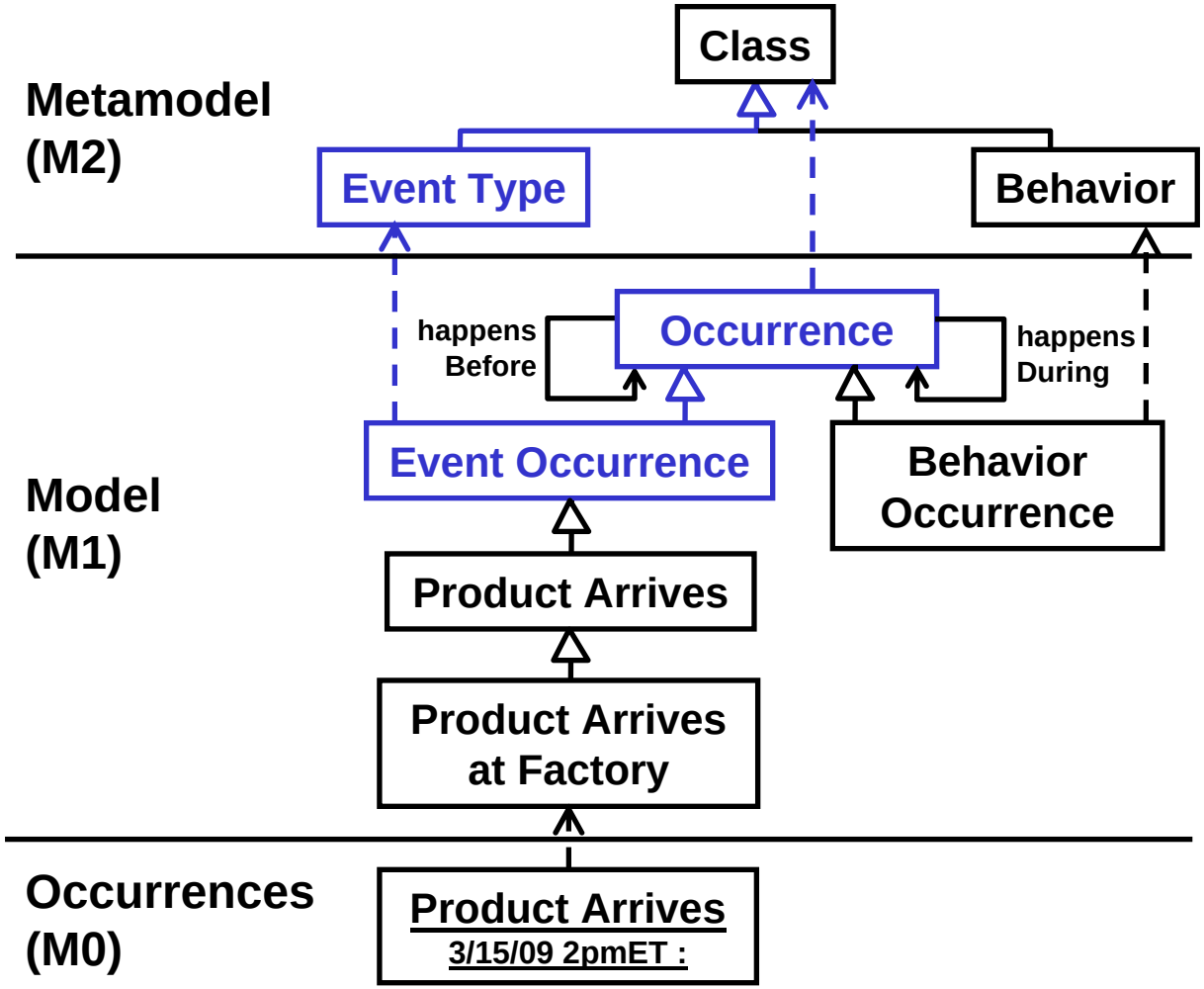


- **Successions:**
  - **layered on Connectors at M2.**
  - **typed by general temporal relation at M1.**
  - **result in links between suboccurrences at M0.**

Events

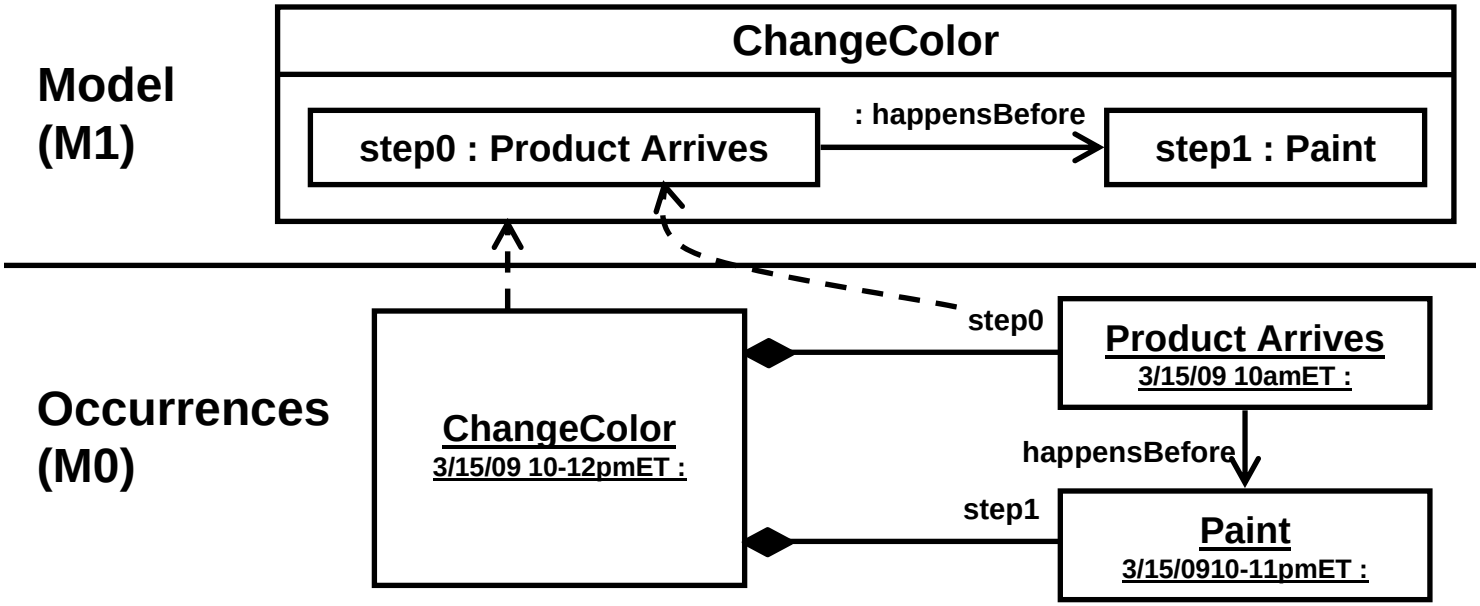
- **Events are alot like behaviors:**
  - **They occur at particular times at M0.**
  - **Can be specified by types at M1, which can be subtyped.**
  - **Can be parts of behaviors.**
  - **Can be specified to happen in a certain order under those behaviors.**
- **UML has:**
  - **A common abstract syntax for events, but they are not types.**
  - **Some semantics in Common Behavior not based on classes …**
  - **… used to varying degrees in the semantics of the other kinds of behaviors.**

Event Types and Occurrences



- **Event types are:**
  - **layered on Classes at M2.**
  - **specialized at M1 (temporal relations promoted).**
  - **occur at M0.**

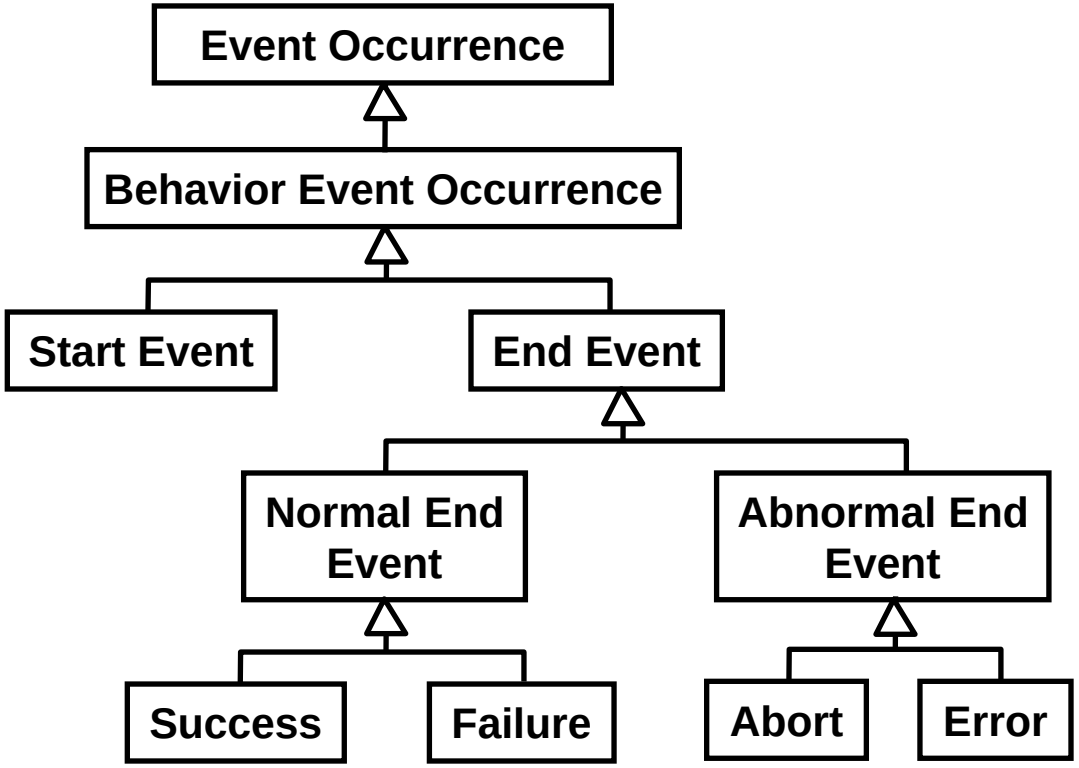
Events in Behavior



- **Event types can be types of properties …**
- **… which are ordered by successions.**

Behavior Events

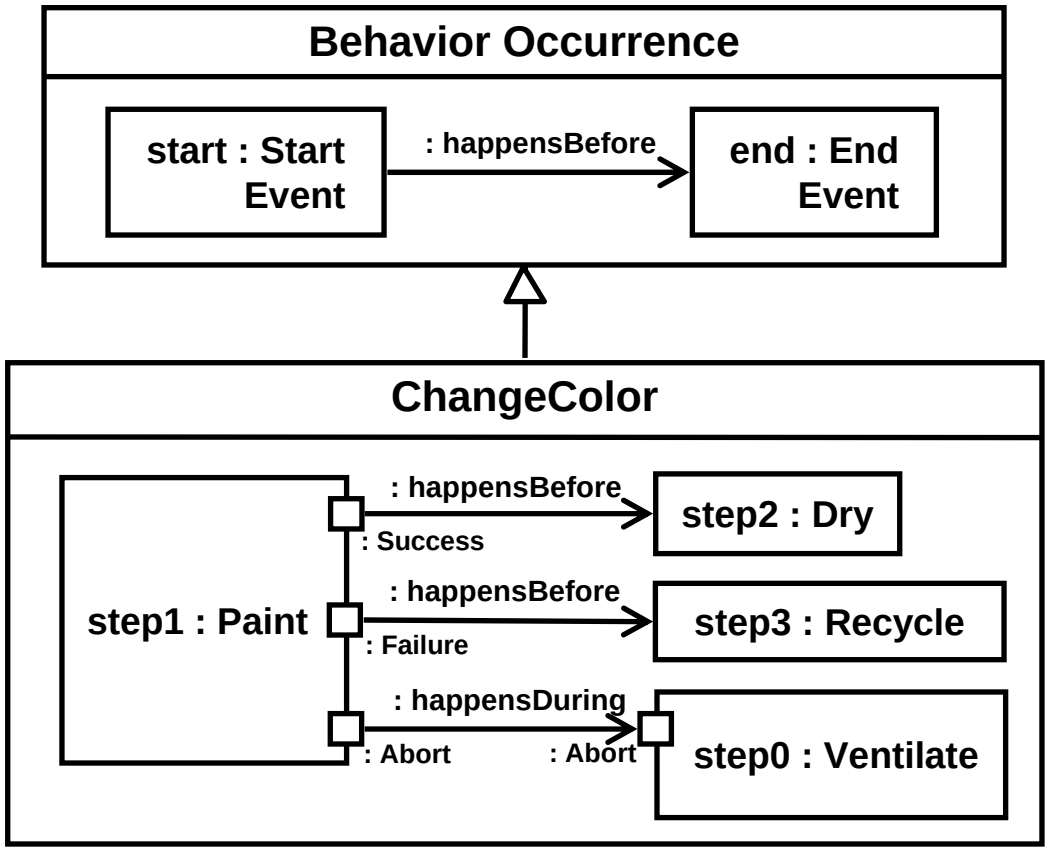
Model (M1)



- **Behaviors have specialized events for their lifecycle ...**

Behavior Events as Parts

Model (M1)

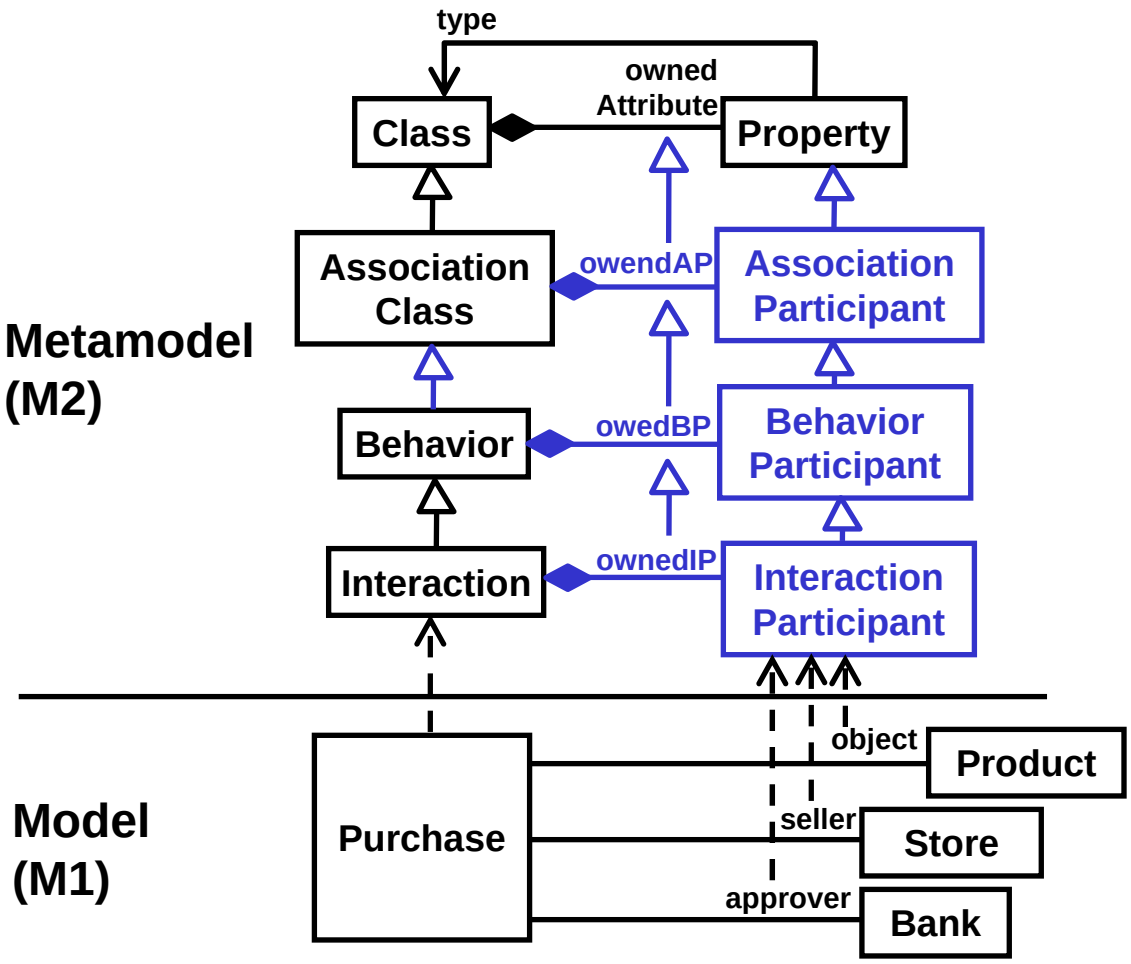


- … which can by the types of “port” properties ...
- … that are ordered by successions.

Participants

- **Behaviors involve objects that behave.**
  - **Interactions have lifelines.**
  - **Activities have object nodes, variables, and partitions.**
  - **Behaviors have parameters.**
- **Association involve objects that are linked.**
- **Behaviors are associations between their participants.**

Participant Properties

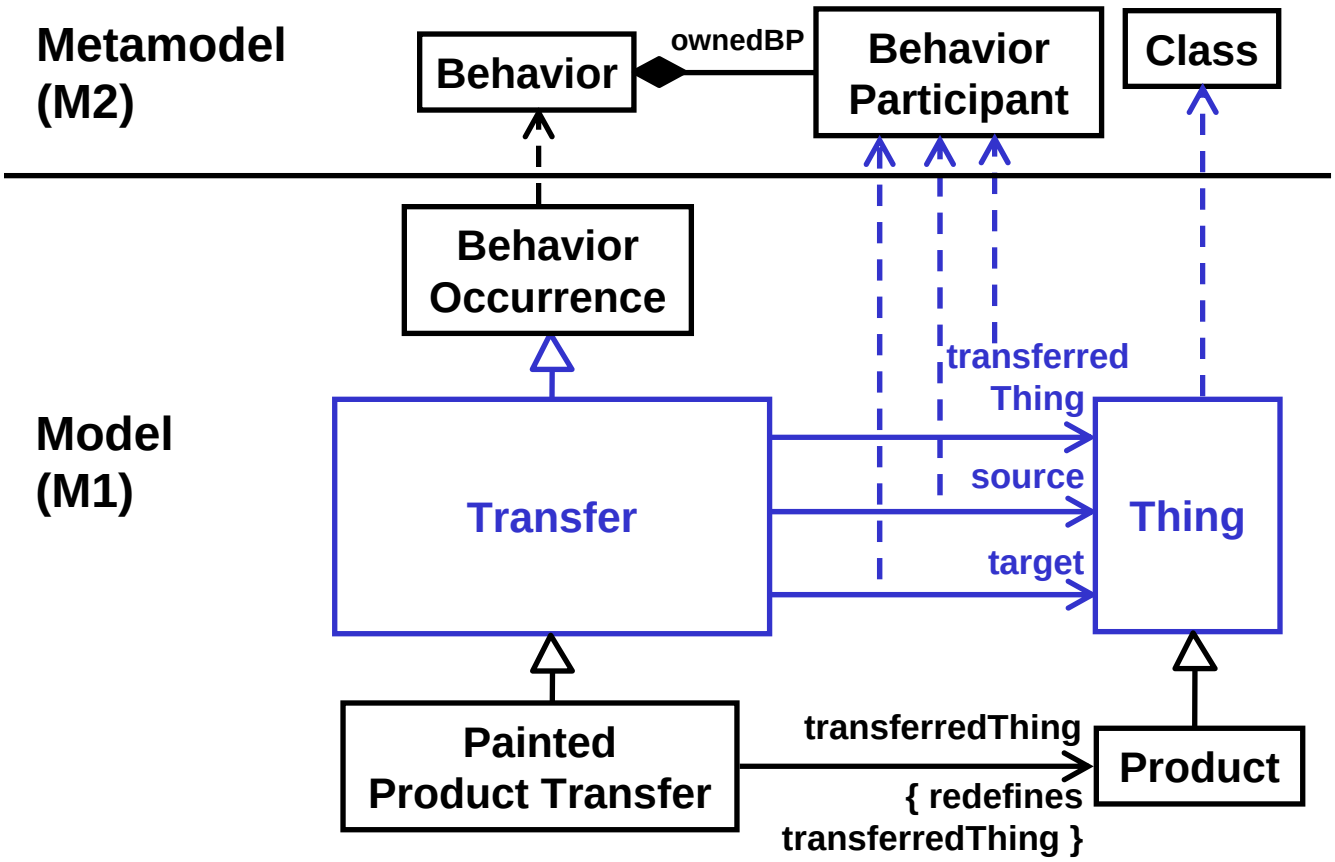


- **Participants:**
  - **layered on Properties at M2.**
  - **assigned participant types at M1.**
  - **have individual values at M0 on occurrences / links.**

Object Flows and Messaging

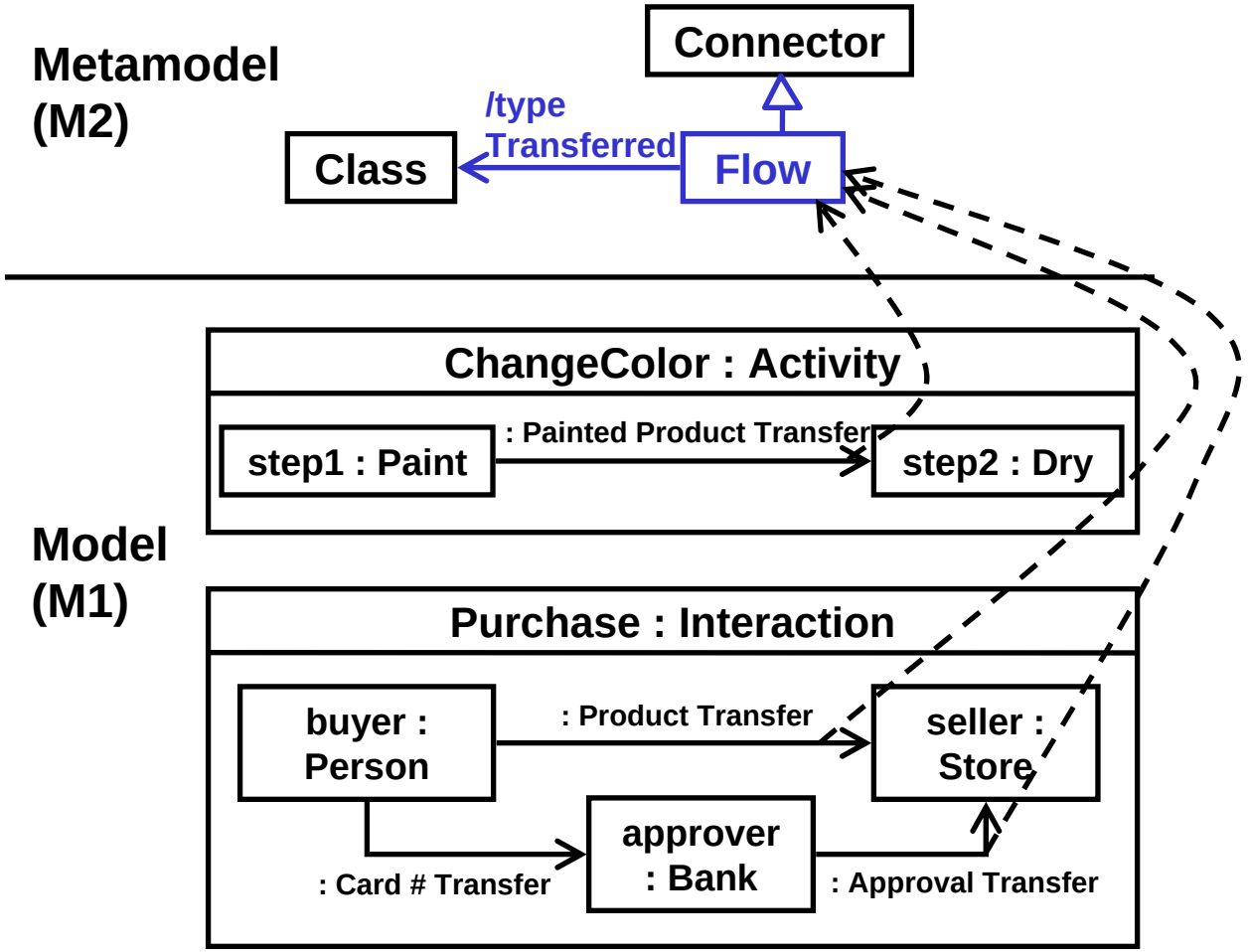
- **Specify transfer of entities at M0.**
  - **Activities have object flows linking pins on actions.**
  - **Interactions have messages linking lifelines.**
- **Transfers take time, they are behavior occurrences.**
  - **Start when entity begins flowing, or message leaves the sender.**
  - **End when entity stops flowing, or message arrives at receiver.**

Transfers



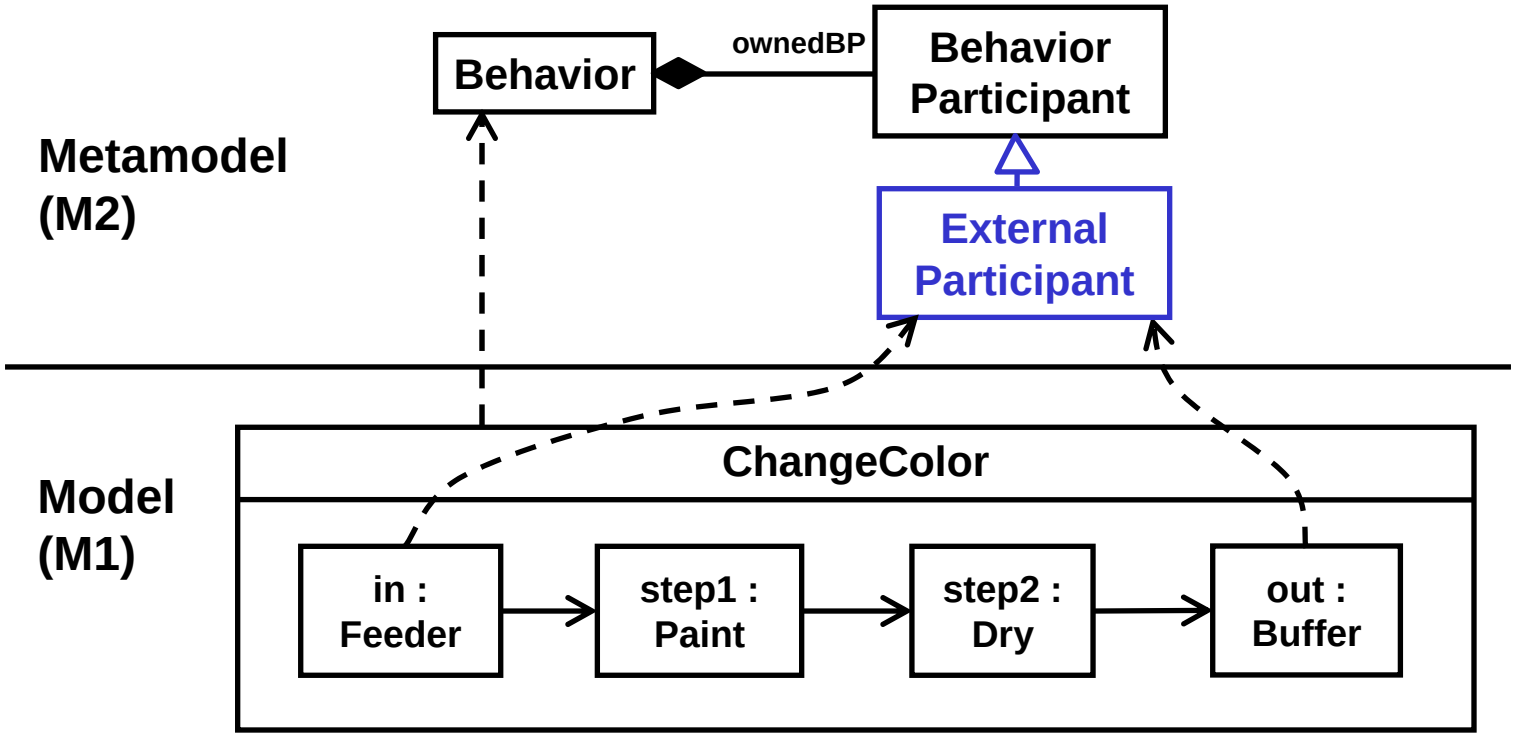
- **Transfers:**
  - a special kind of Behavior Occurrence at M1.
  - have participant properties, and are specialized at M1.
  - occur at M0 involving individuals that are values of participant properties.

Flows



- **Flows:**
  - **layered on Connectors at M2.**
  - **typed by transfers at M1.**
  - **have transfer occurrences as values at M0.**

External Participants

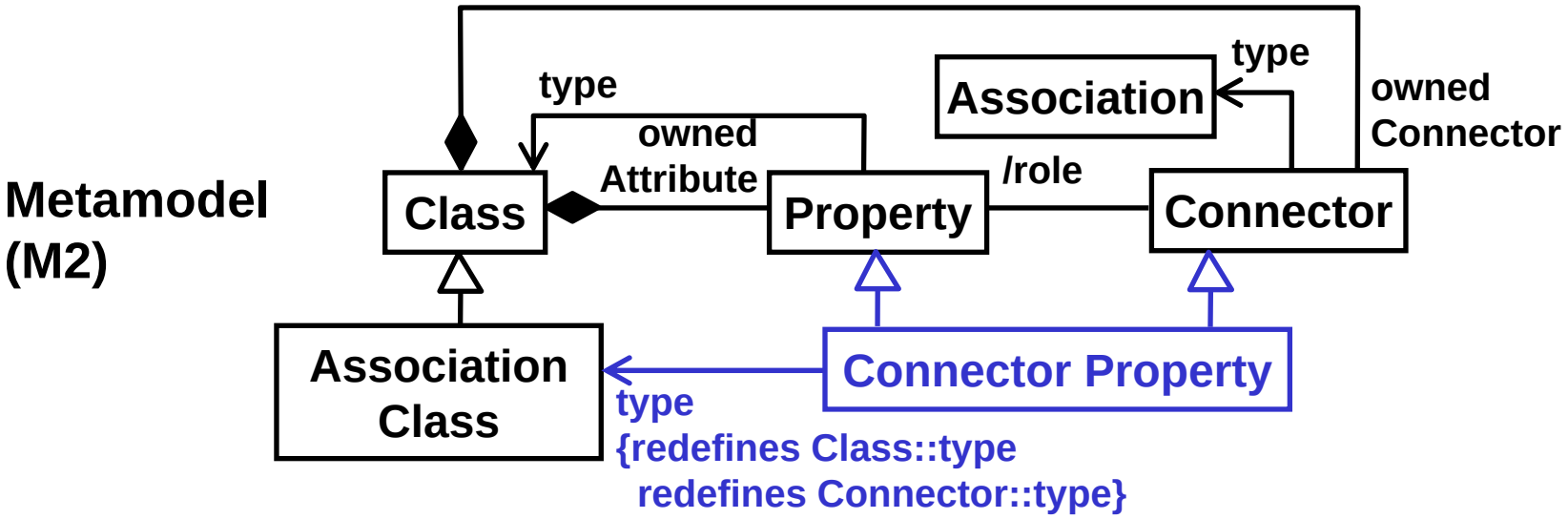


- **External participants:**
  - **layered on Behavior Participants at M2.**
  - **can be linked by flows at M1 for inputs and outputs.**
  - **result in occurrences of transferring at M0.**

Flow Ordering

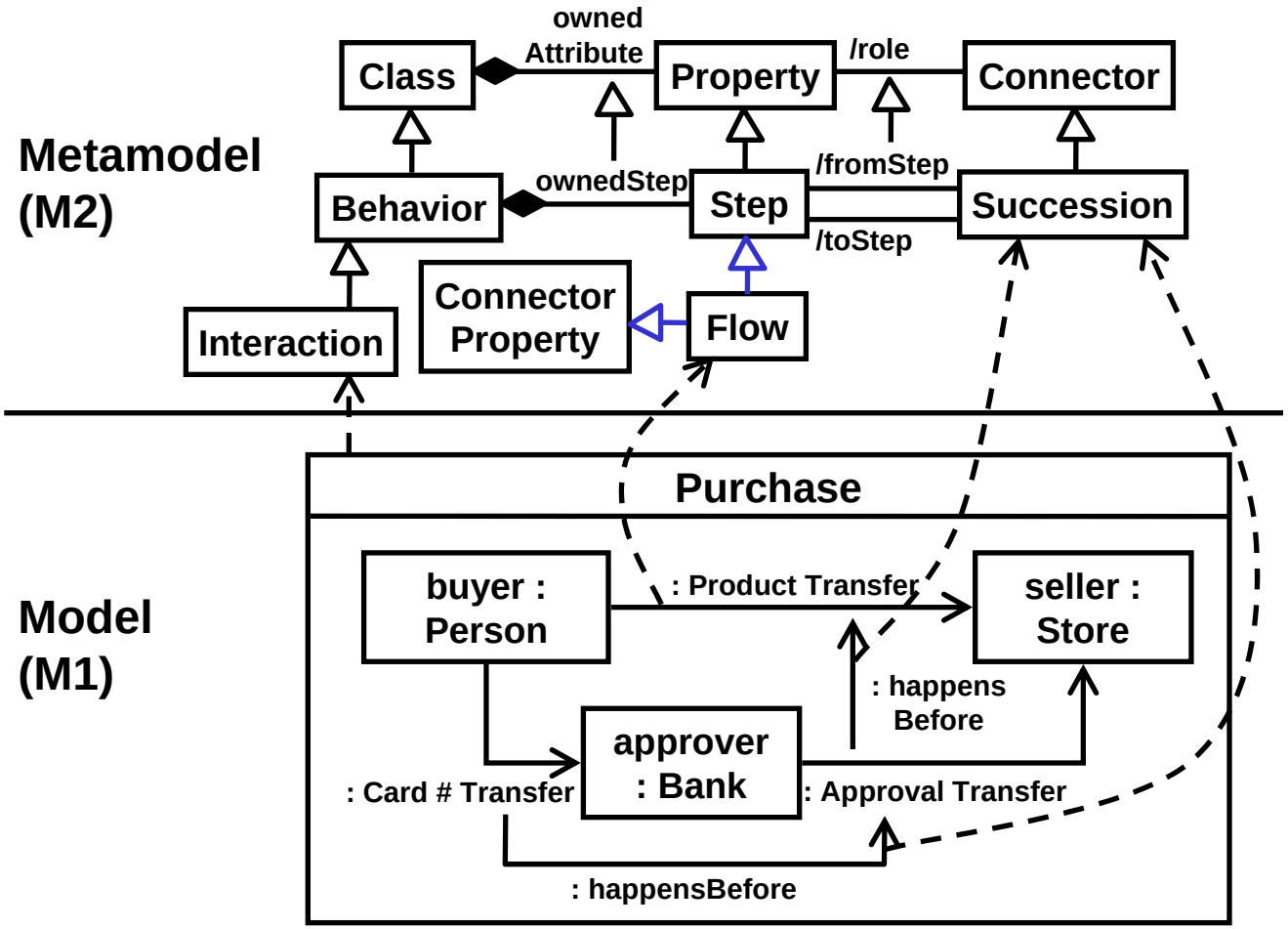
- **Some flows happen before others**
  - **Interactions order messages and interaction uses.**
  - **Protocol state machines specify allowed orders of operation calls and other protocols.**
  - **Activities order actions for sending and receiving messages.**
- **Requires successions between flows (connectors between connectors).**

Connector Properties



- **Connector Properties:**
  - **layered on Connectors and Properties at M2.**
  - **have association classes as types at M1.**
  - **have links as values at M0.**
- **M0 values are the links specified by the connector.**

Flow Properties



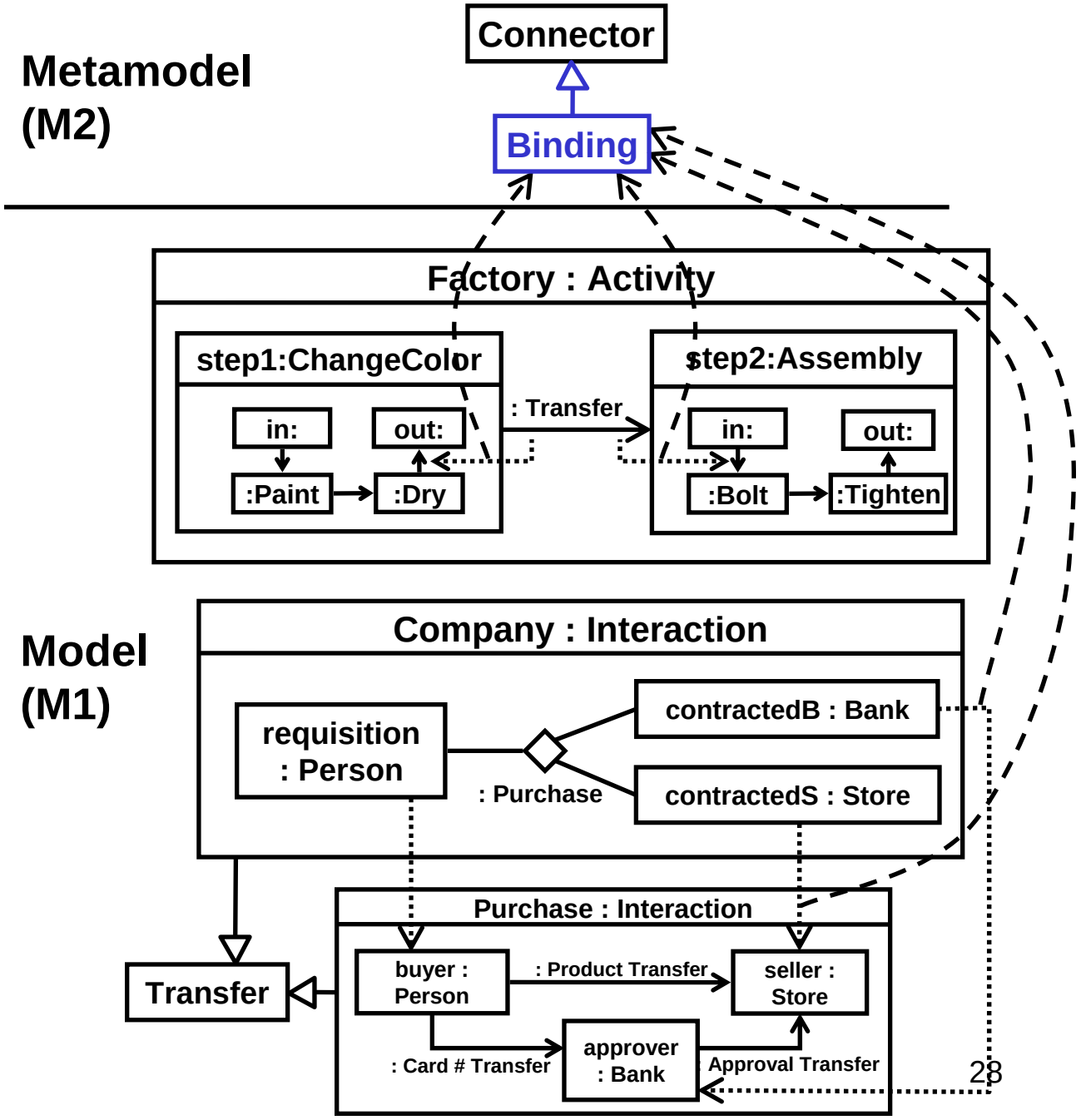
- **Flows:**
  - **layered on Connector Properties and Steps at M2.**
  - **connected by successions at M1.**
  - **have transfer occurrences as values at M0.**

Composition with Flows and Participants

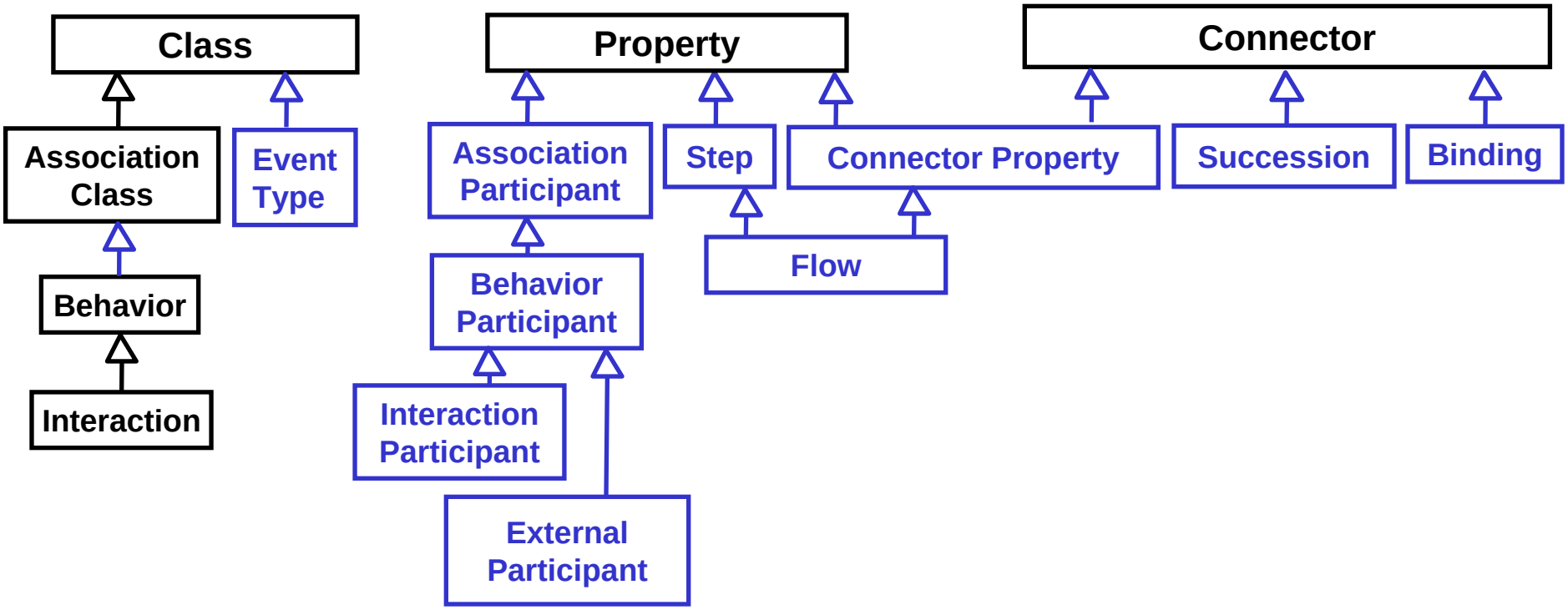
- **Flows are part of behavior composition.**
  - **Activities have pins matching behavior parameters.**
  - **Interactions have arguments matching behavior parameters, used with collaboration, and collaboration role bindings.**
- **Requires specifying equivalence between transfers.**

Bindings

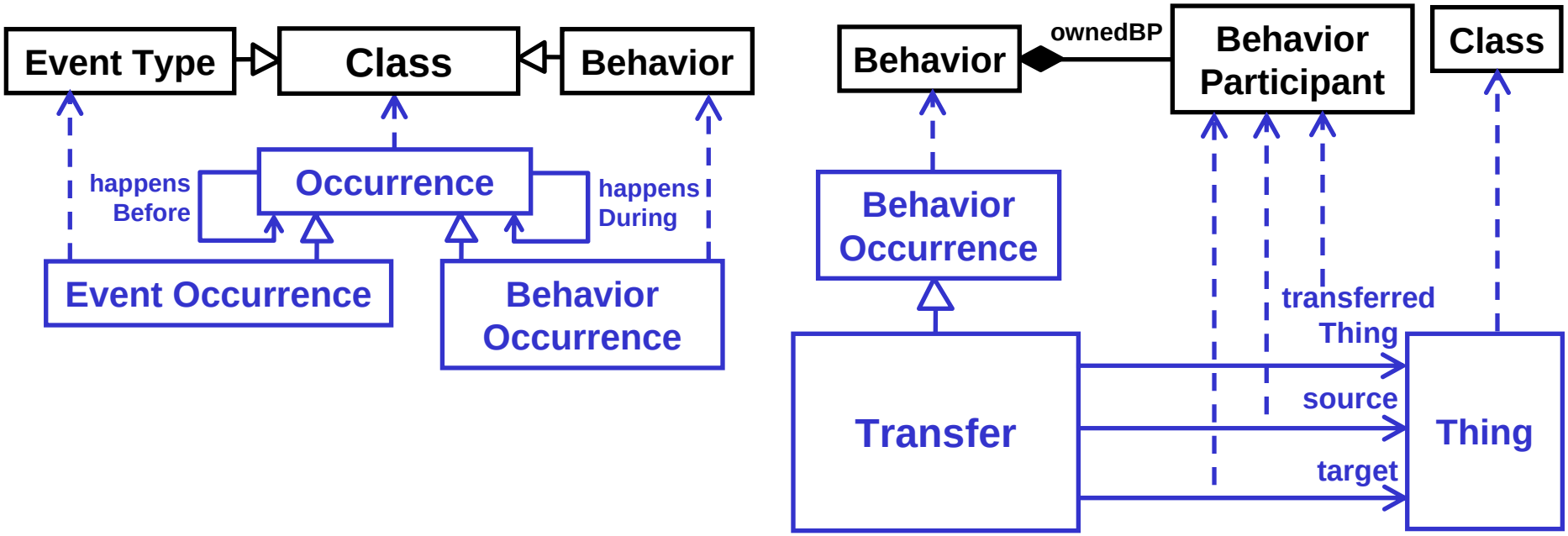
- layered on Connectors at M2.
- link flows and participants at M1.
- specify equivalence of transfers at M0.



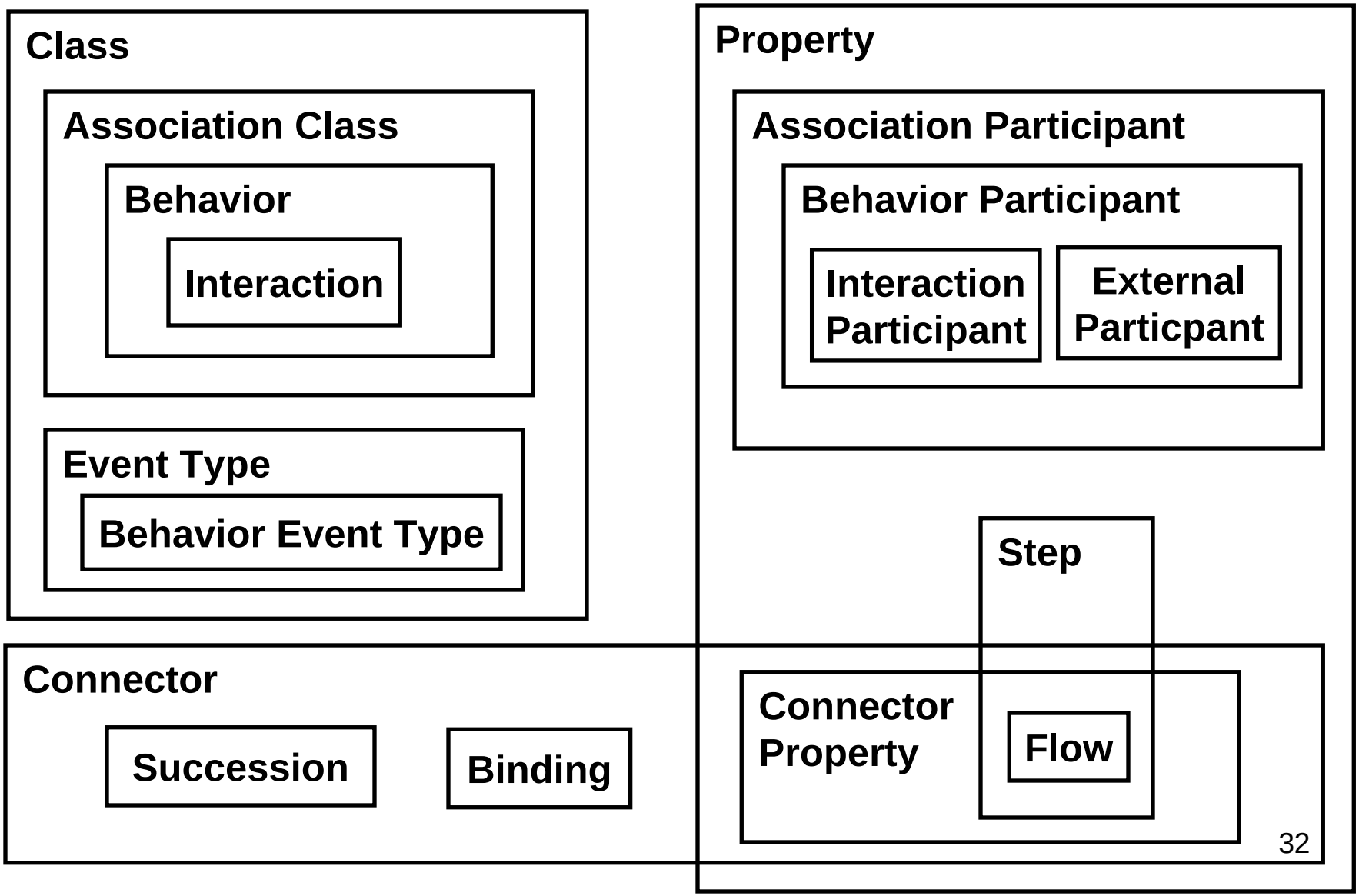
Metaclass Taxonomy



Model Library



Metaclass Taxonomy (CDGN)



Summary

- **Semantics determines when M0 elements conform to M1 models.**
- **Metamodels should**
  - **reflect common semantics among M1 model elements.**
  - **have thin layers of clearly defined abstractions.**
  - **be augmented with M1 libraries to capture the relationship to M0.**
- **Behavior as example:**
  - **M1 behaviors and events specify M0 occurrences.**
  - **Specialize from Kernel at M2, including from Class, Property, Connector, and Association Class.**
  - **Capture occurrences and temporal relations at M1.**