State Machines as Composite Structure:

(Onto)Logical State Machines

Part 1

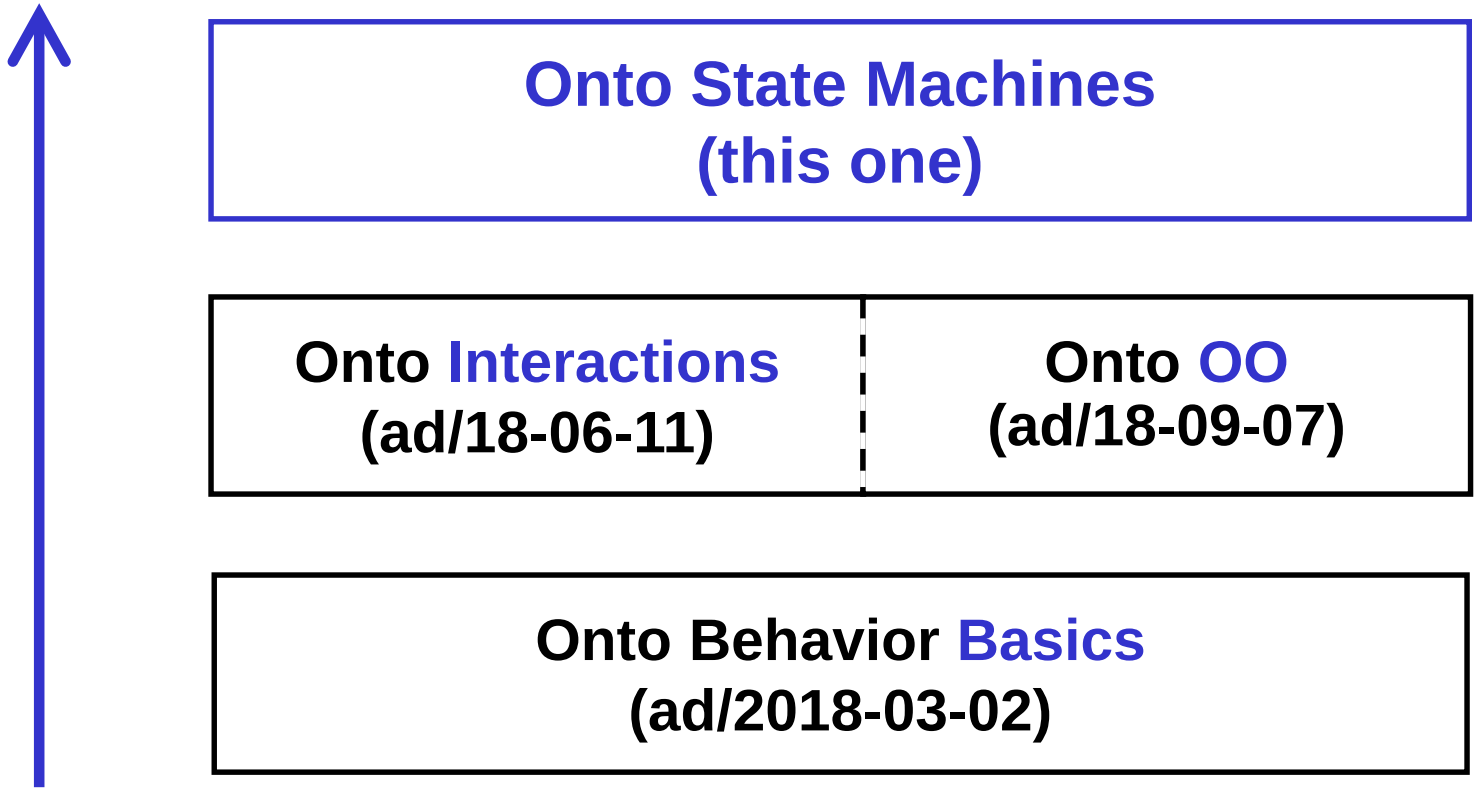
Bjorn Cole
Georgia Institute of Technology

Conrad Bock,
U.S. National Institute of Standards and Technology

Overview

- RoadMap
- Motivation
  - Behavior, review
  - Interactions, review
  - State machines, requirements
- State Machines Solution
  1. Stimuli = end of transfers (events)
  2. State and transition behaviors
  3. Matching past events to transitions
- Summary

Behavior as Composite Structure Presentation Stack



Overview

- RoadMap
- **Motivation**
  - **Behavior, review**
  - Interactions, review
  - State machines, requirements
- State Machines Solution
  1. Stimuli = end of transfers (events)
  2. State and transition behaviors
  3. Matching past events to transitions
- Summary

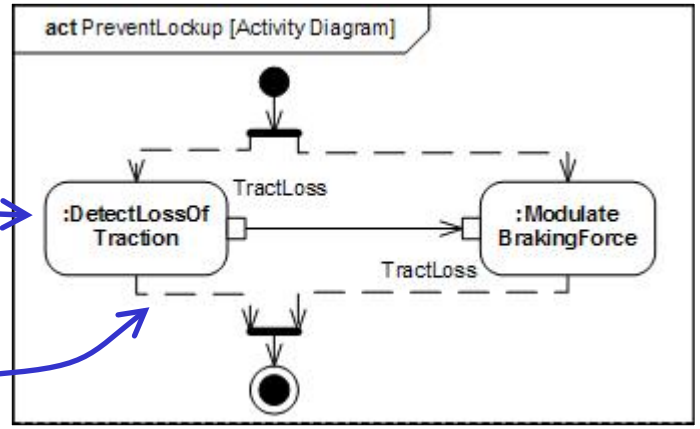
General Problem

- **UML has three behavior diagrams.**
  - **Activity, state, interaction.**
- **Very little integration or reuse between them.**
  - **Three underlying metamodels.**
  - **Three representations of temporal order.**
- **Triples the effort of learning UML and building analysis tools for it.**

General Solution

- **Treat behaviors as assemblies of other behaviors.**
  - **Like objects are assemblies of other objects.**
- **Assembly = UML internal structure**
  - **Pieces represented by properties.**
  - **Put together by connectors.**
- **Put all behavior diagrams on the same underlying behavior assembly model.**

Behaviors as Composite Structure

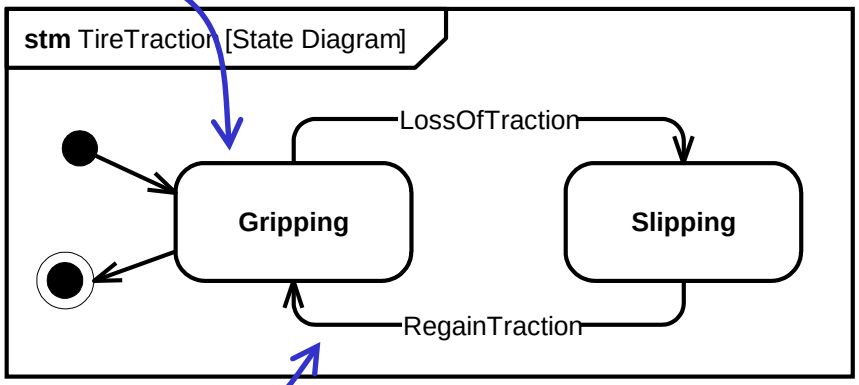


Property

Connector

Activity

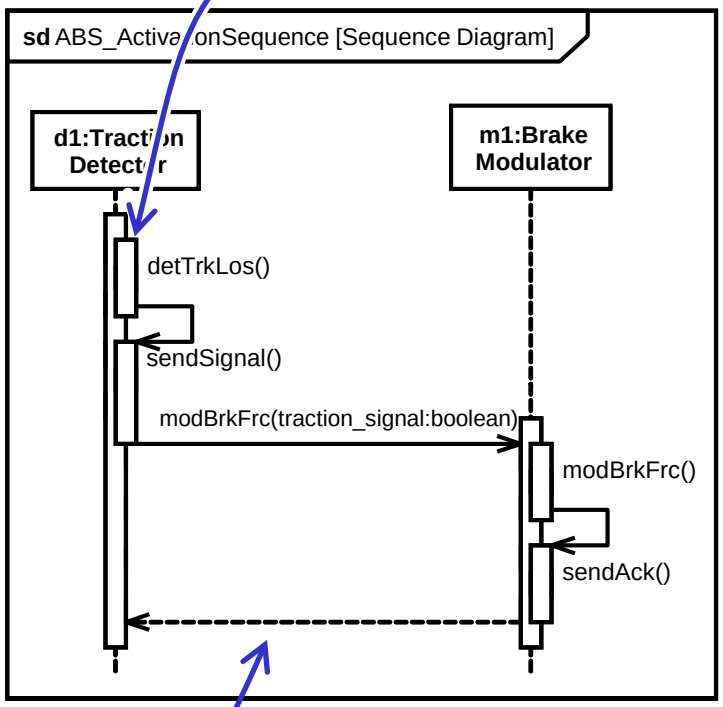
Property



Connector

State Machine

Property



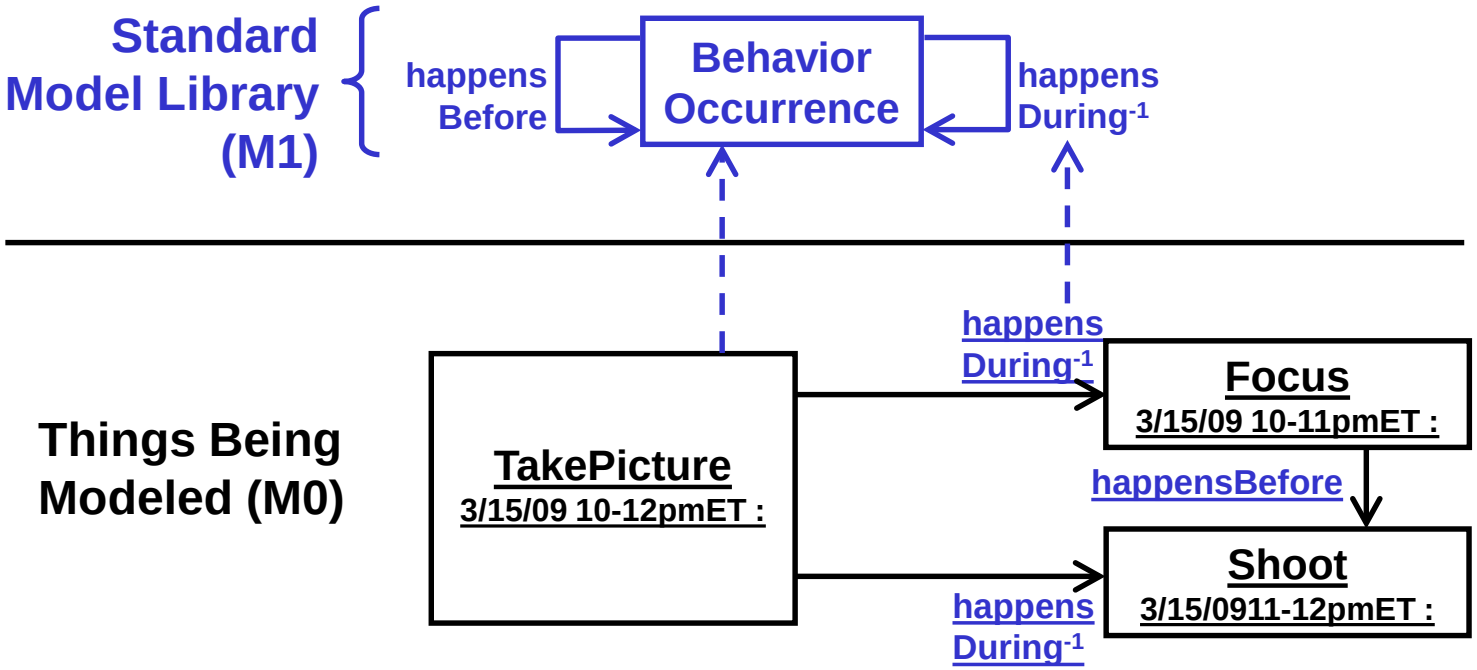
Connector

Interaction

Behavior: What's Being Modeled?

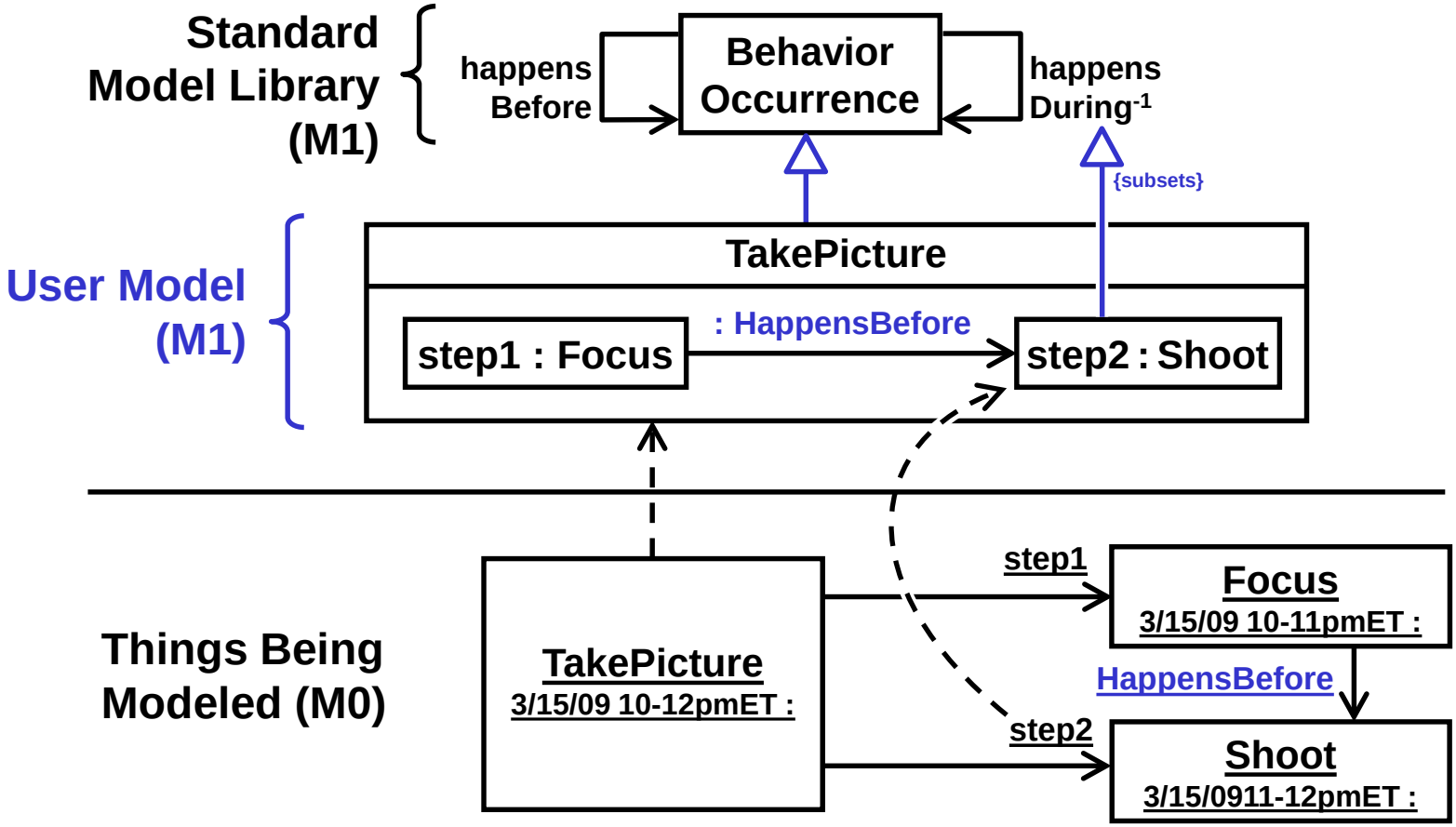
- "Things" that occur in time
  - Eg, taking a picture, focusing, etc.
  - Not "behaviors", "actions", etc.

Behavior: What's in Common?



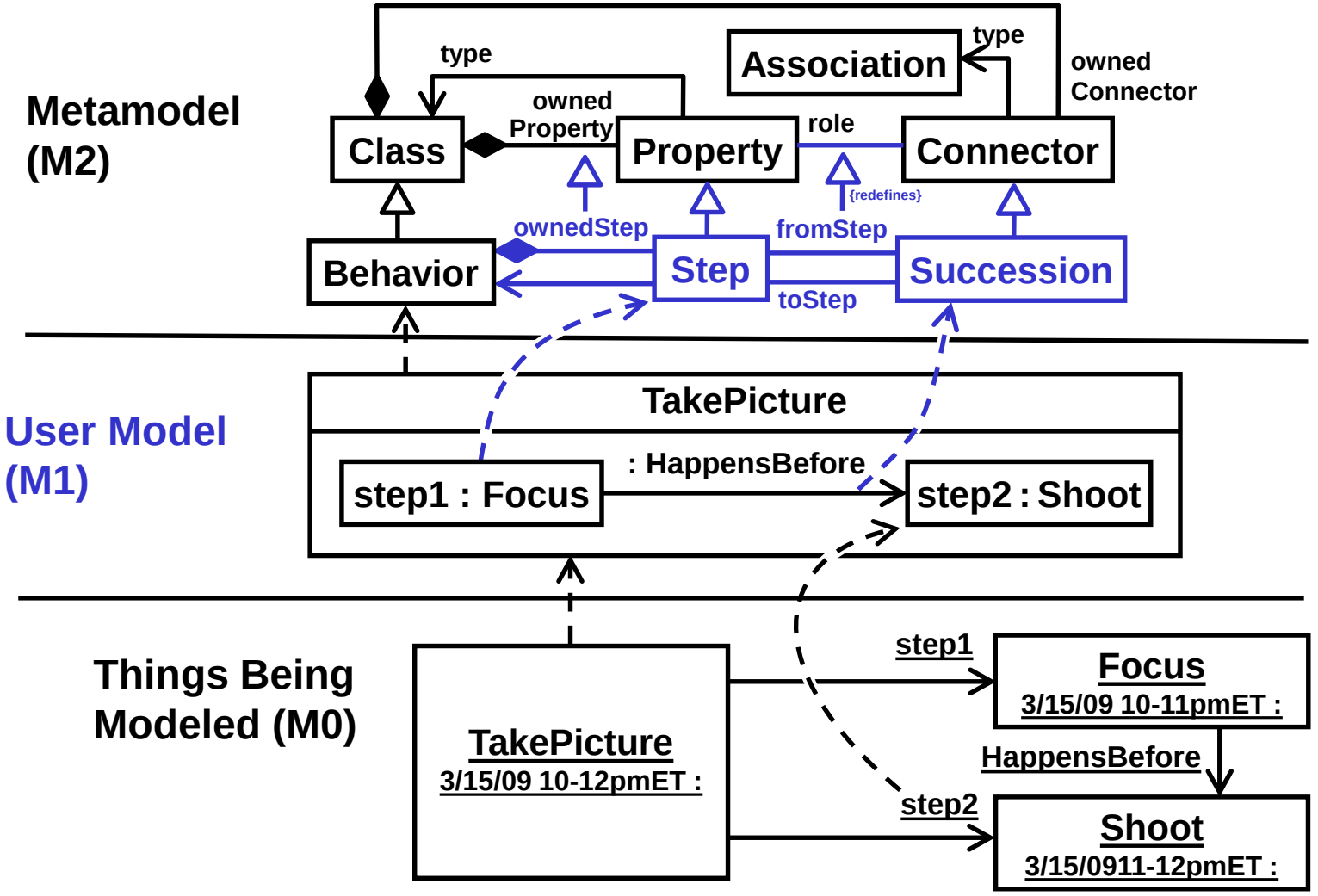
- **They happen before or during each other.**
  - **Construct M1 library for this.**
  - **Use it to classify things being modeled.**

Behavior: Use Library



- **Specialize library classes and subset/redefine library properties.**

Behavior: Too repetitive at M1?

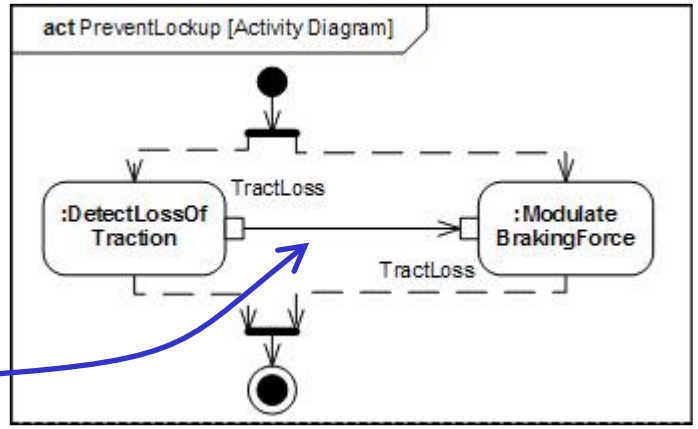


- **Capture M1 patterns in M2 elements.**
  - **Tools apply patterns automatically.**

Overview

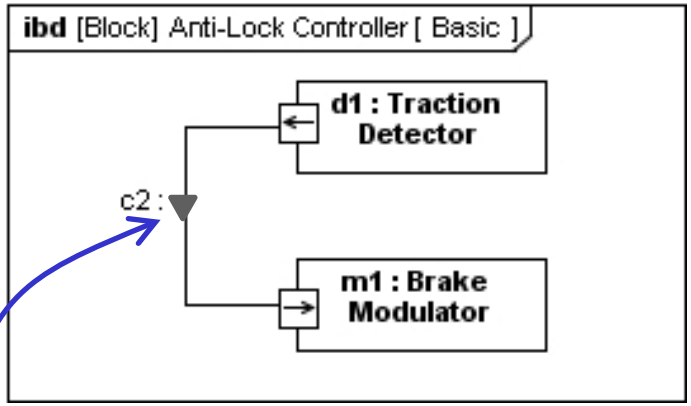
- RoadMap
- Motivation
  - Behavior, review
  - **Interactions, review**
  - State machines, requirements
- State Machines Solution
  1. Stimuli = end of transfers (events)
  2. State and transition behaviors
  3. Matching past events to transitions
- Summary

Interactions Problem



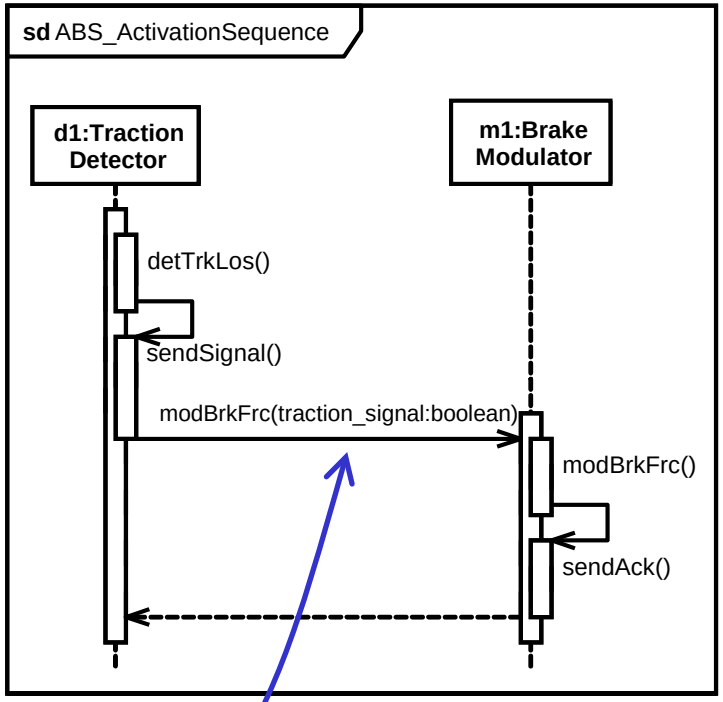
Object Flow

Activity



SysML Internal Block Diagram

Item Flow



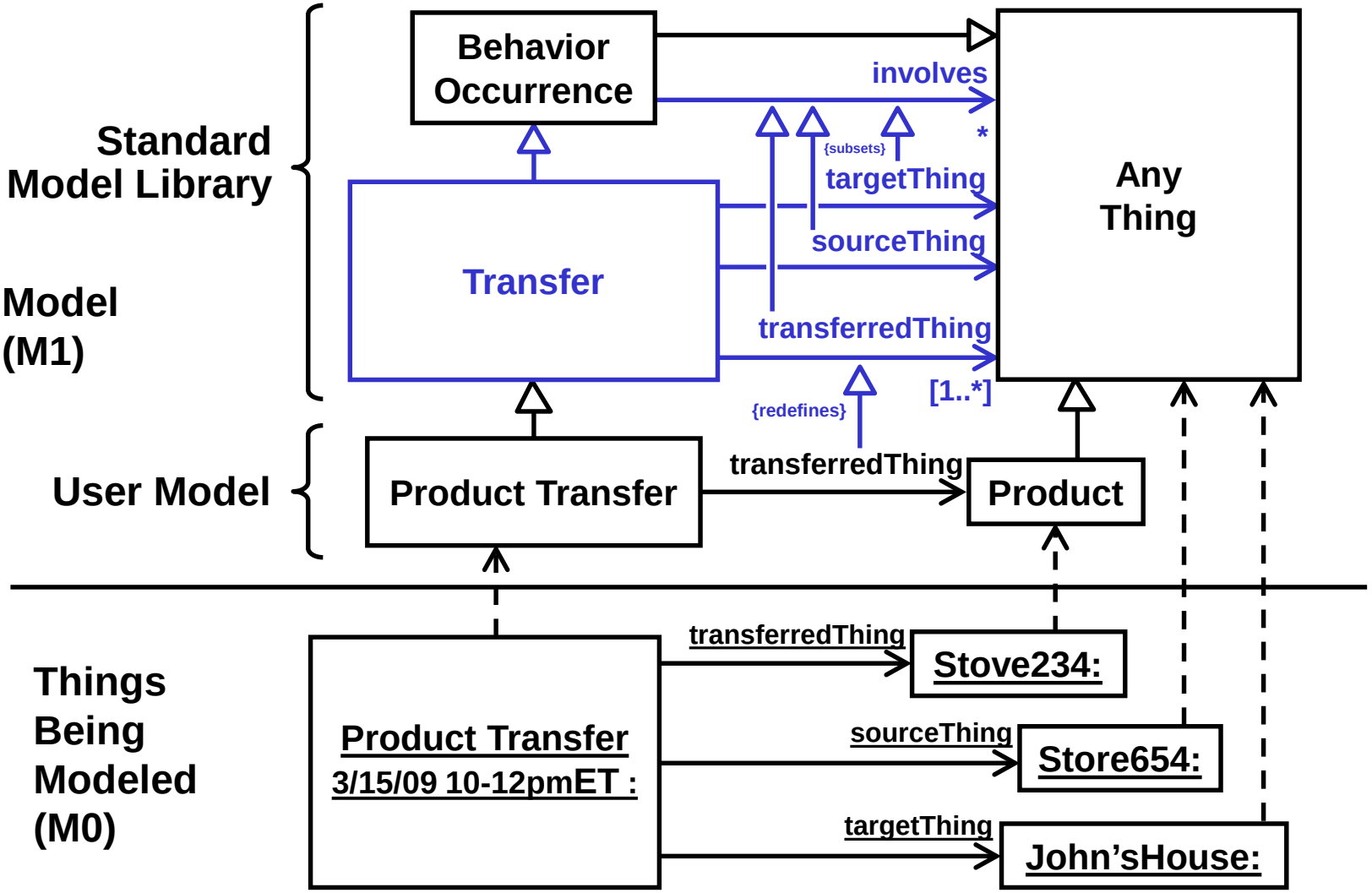
Interaction

Message

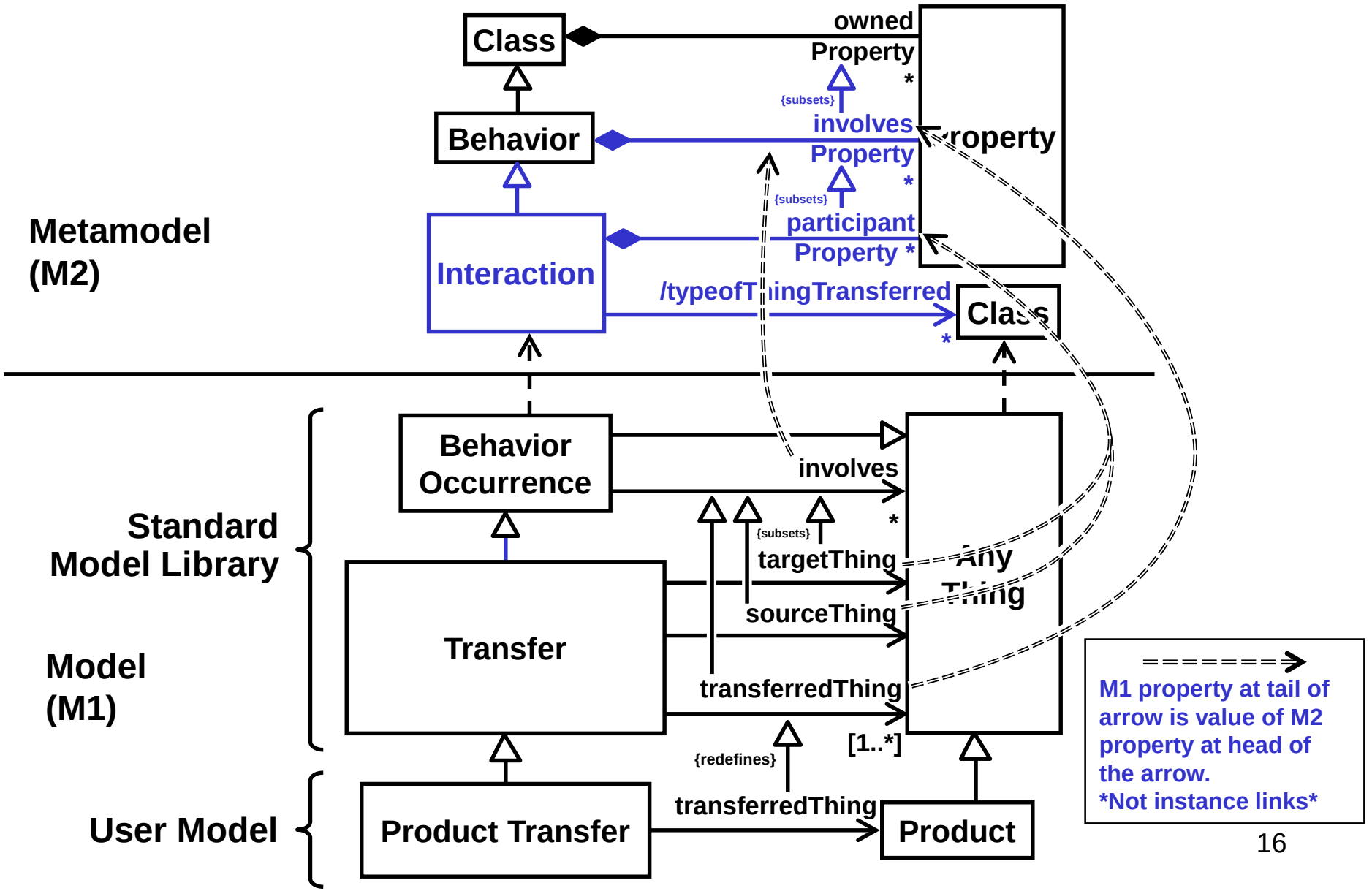
Interactions Requirements

1. **Between things that outlive interactions.**
   - **Objects have many interactions over time.**
   - **Not just between steps in an activity.**
2. **Interactions are reusable and composable.**
   - **The same kind of interaction might be used in many user models and**
   - **contain many other interactions ordered in time.**
3. **Interacting objects have "mailboxes".**
   - **Things being exchanged leave and arrive at specified places in the interacting objects.**
   - **Aka, output/inputs.**

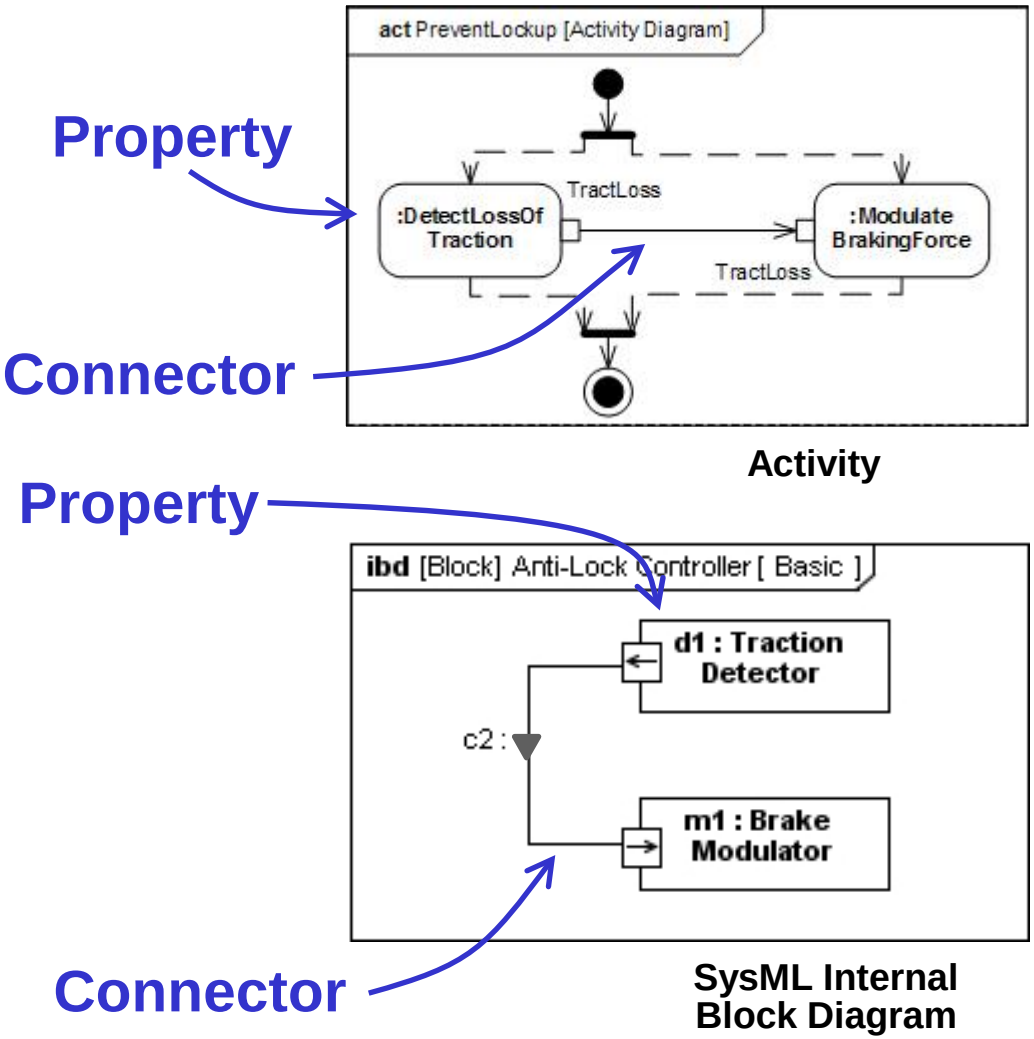
Transfers (M1)

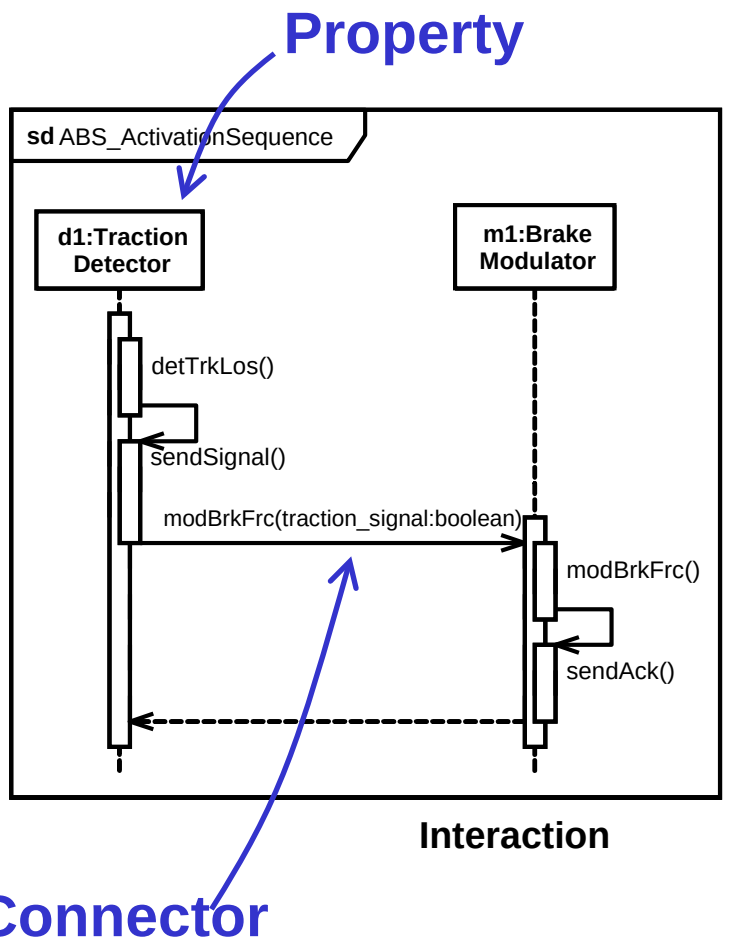


Interactions (M2)



Transfers Along Connectors?



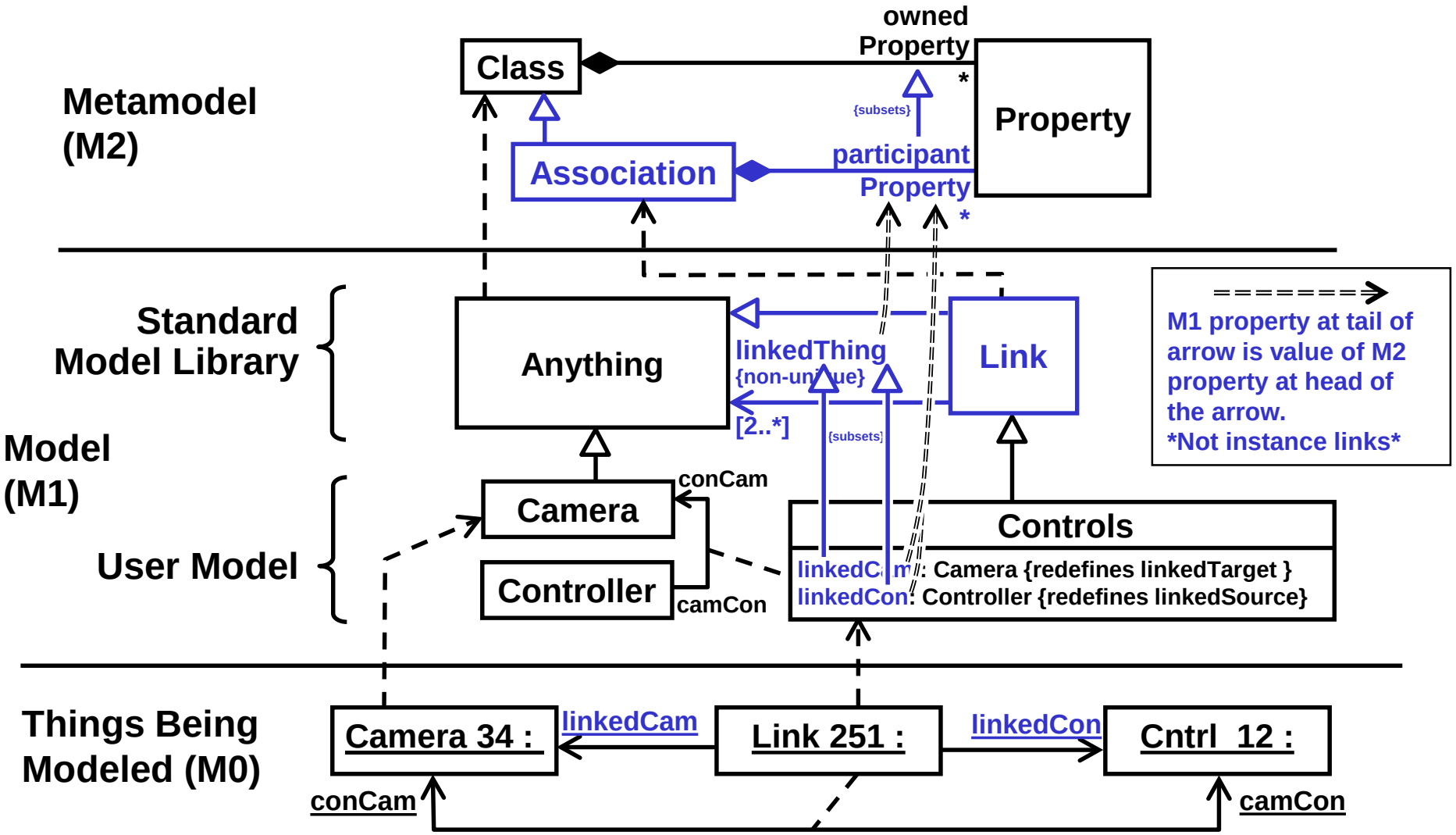


- **Connectors are typed by associations.**
  - **But transfers are behaviors.**

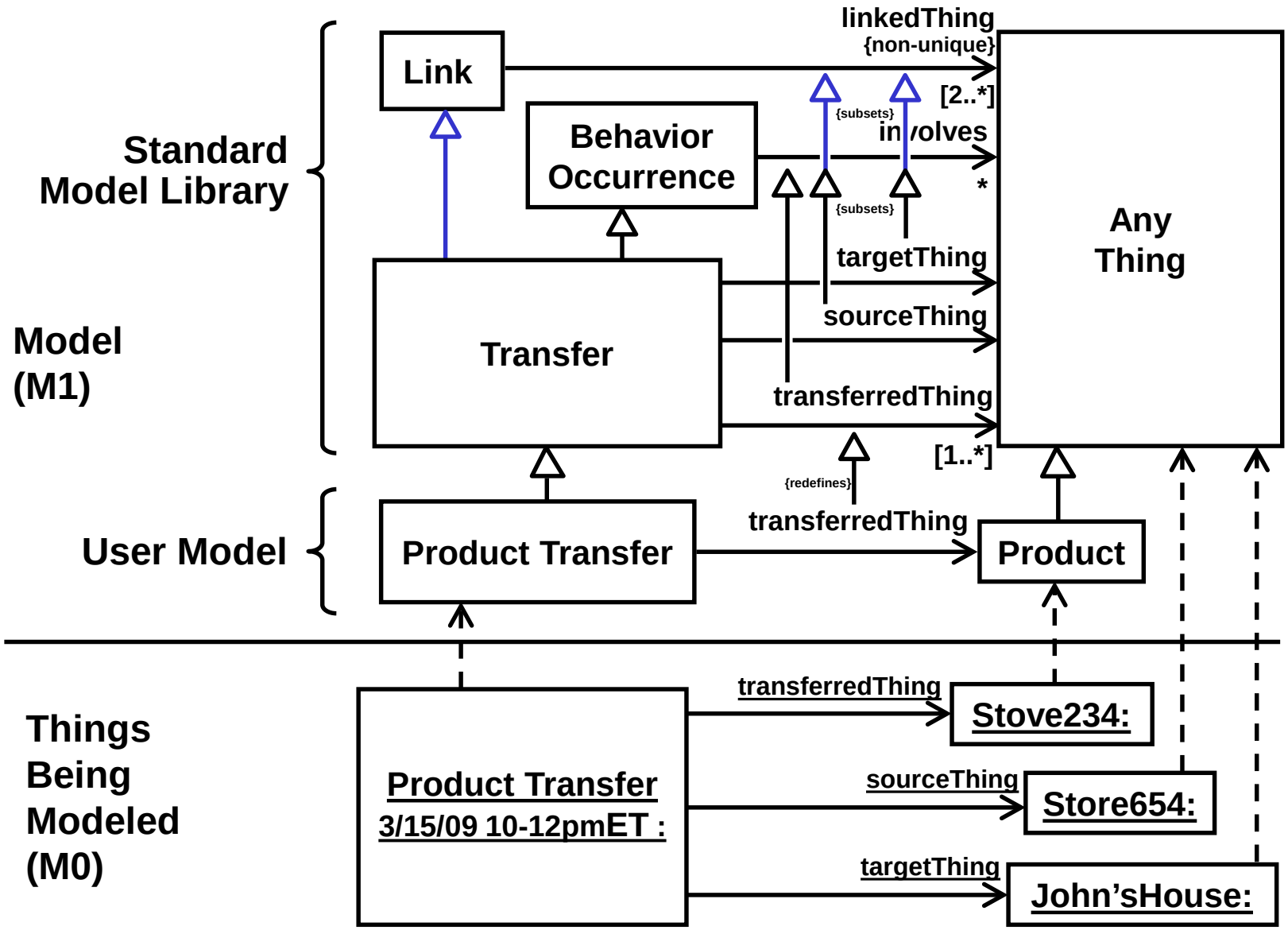
Interaction = Behavior & Association

- **Associations and behaviors both have objects that participate in them.**
  - **Associations link their participants.**
  - **Behaviors involve their objects.**
    - **Interactions have lifelines.**
    - **Activities have object nodes, partitions, etc.**
    - **Behaviors have parameters.**
- **Interactions are behaviors that are also associations between their participants.**

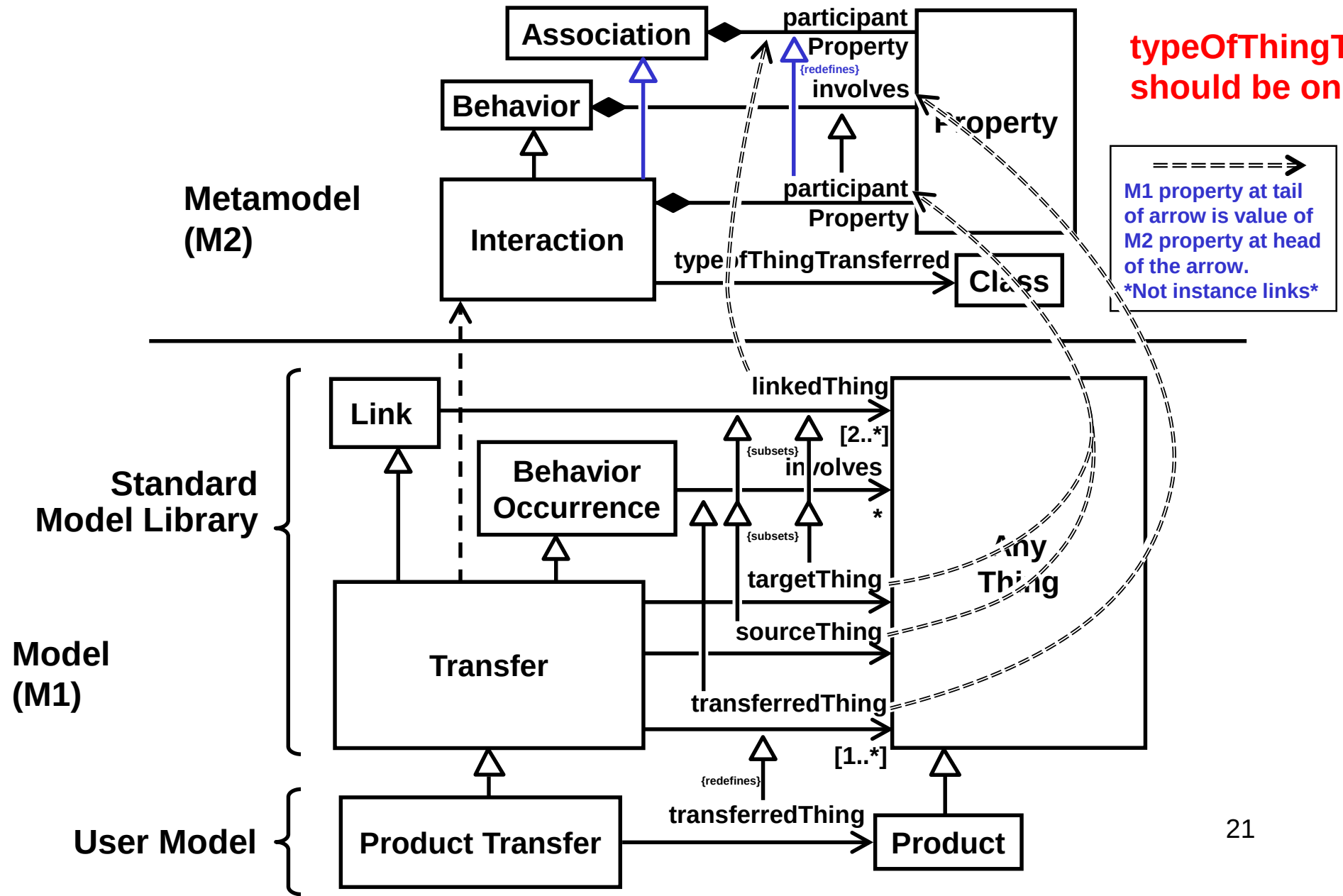
Links (M1) & Associations (M2)



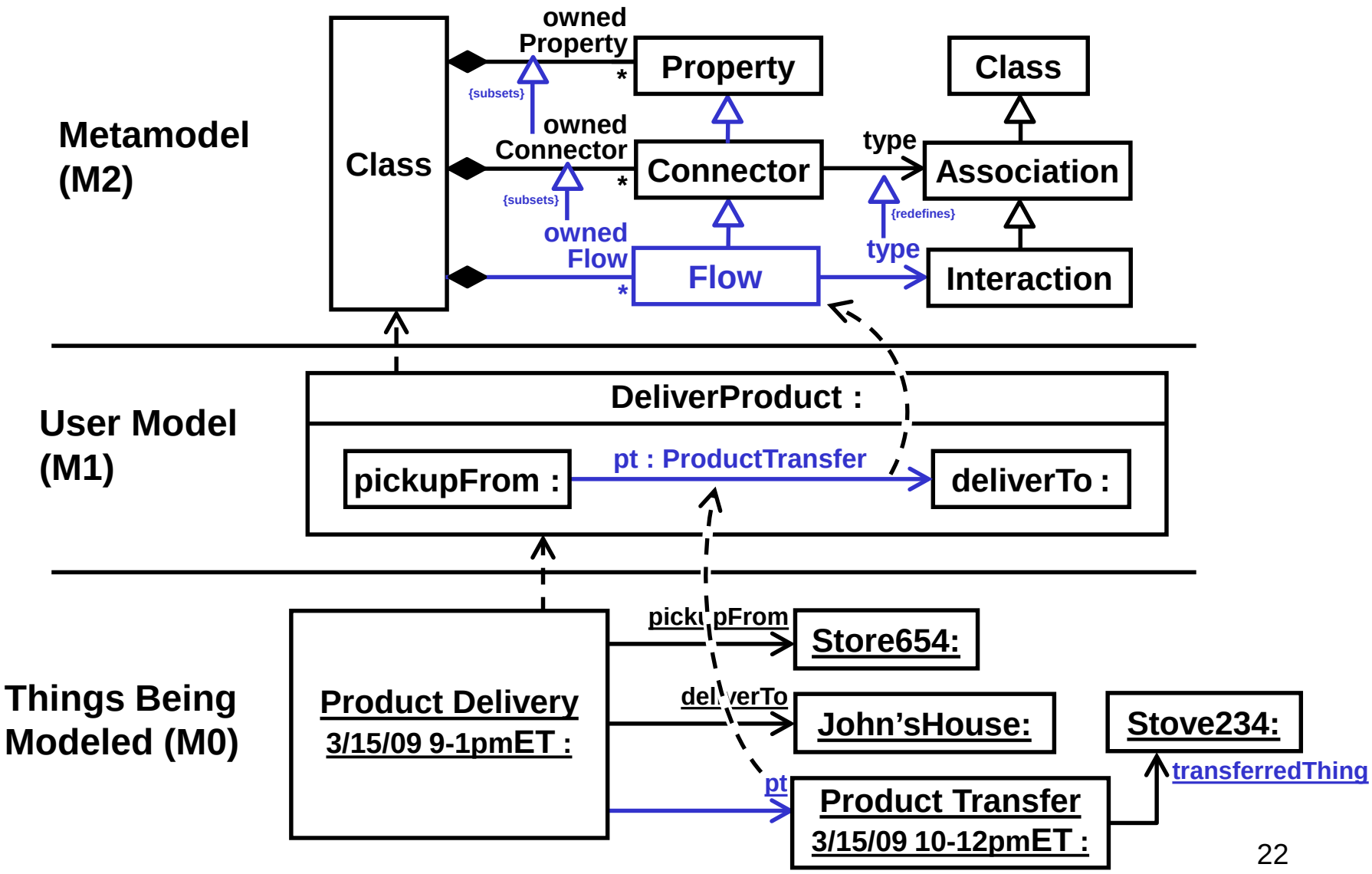
Transfers as Links (M1)



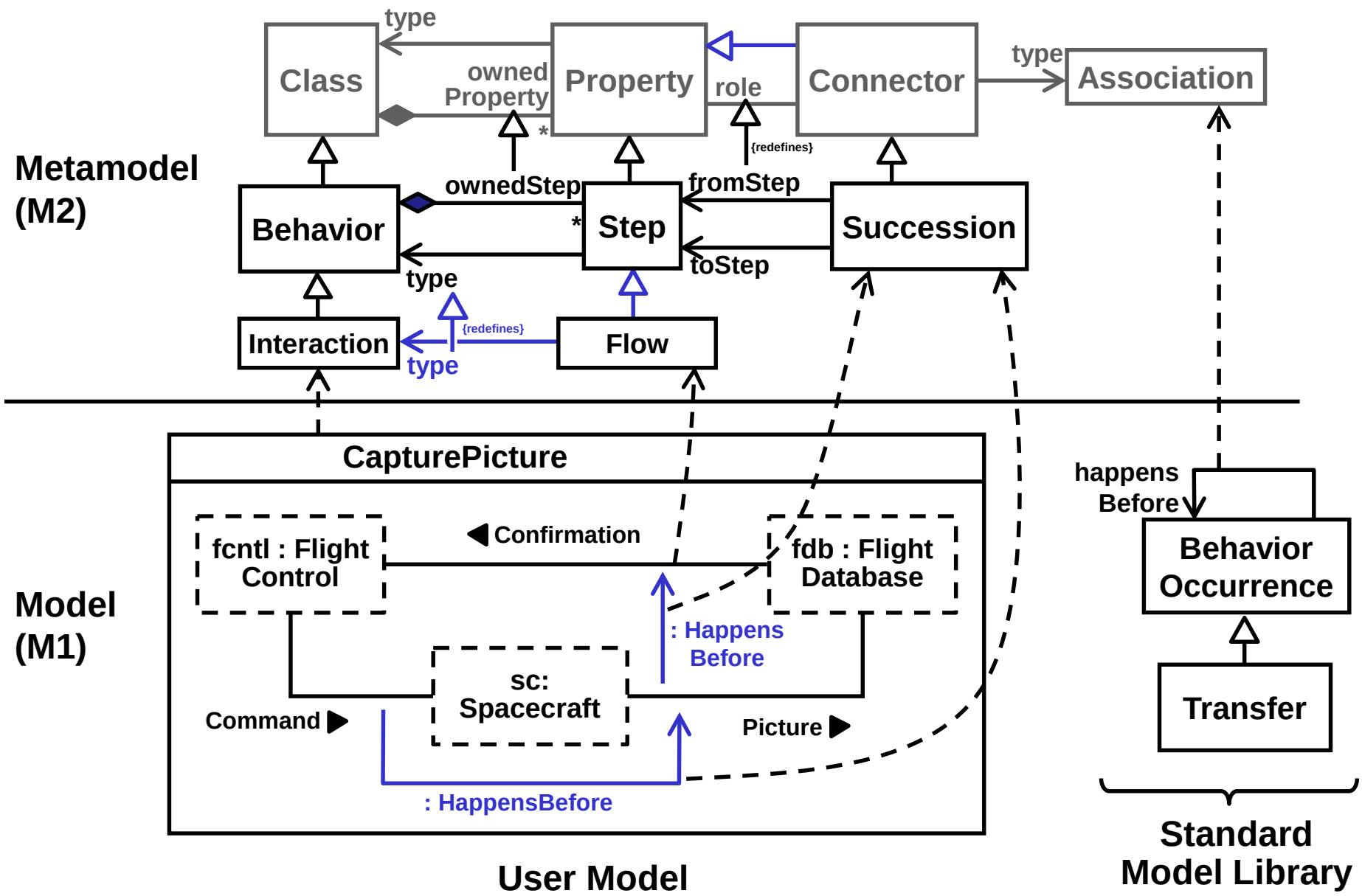
Interaction Participants (M2)



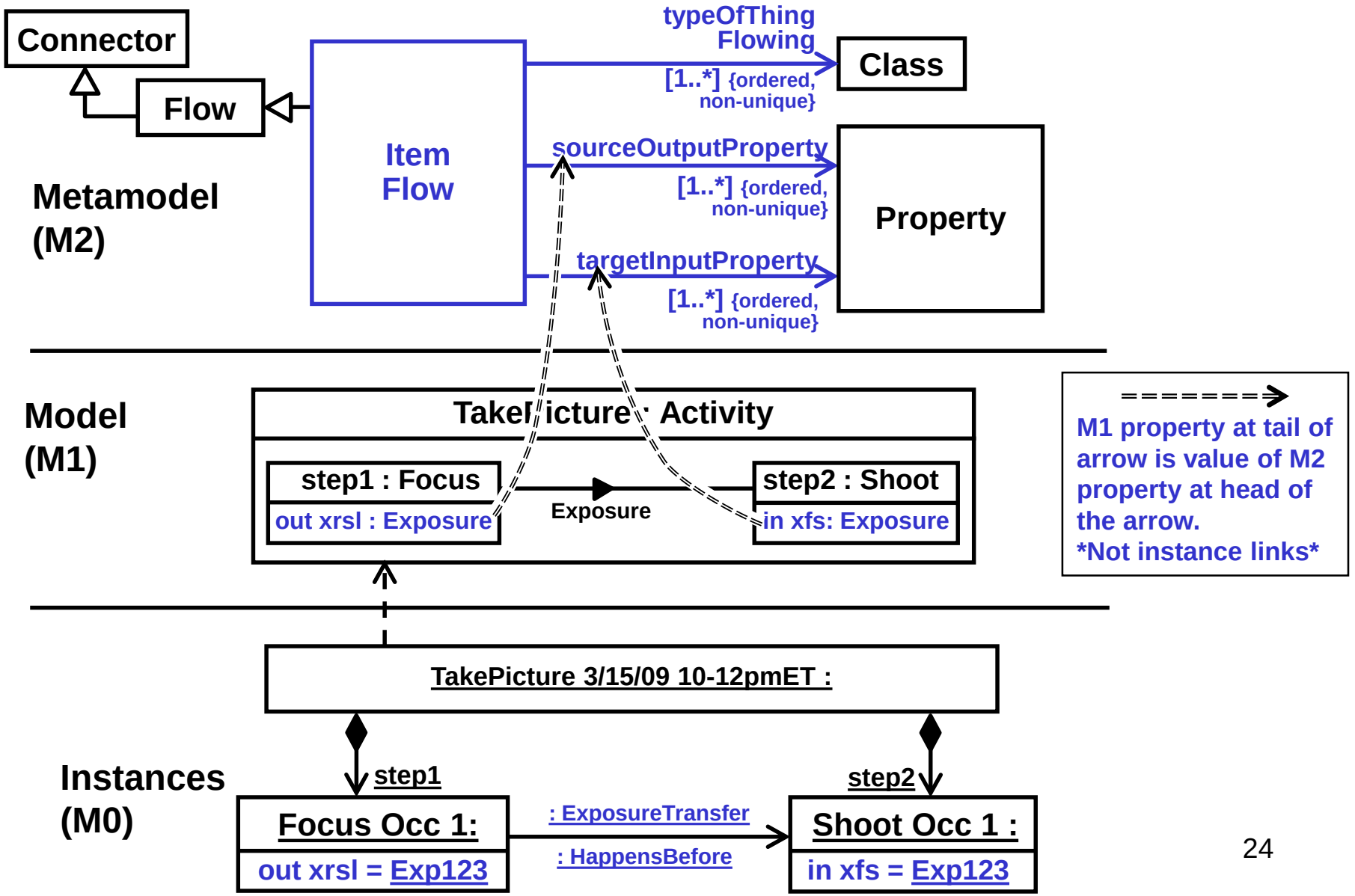
Connectors Reusing Interactions



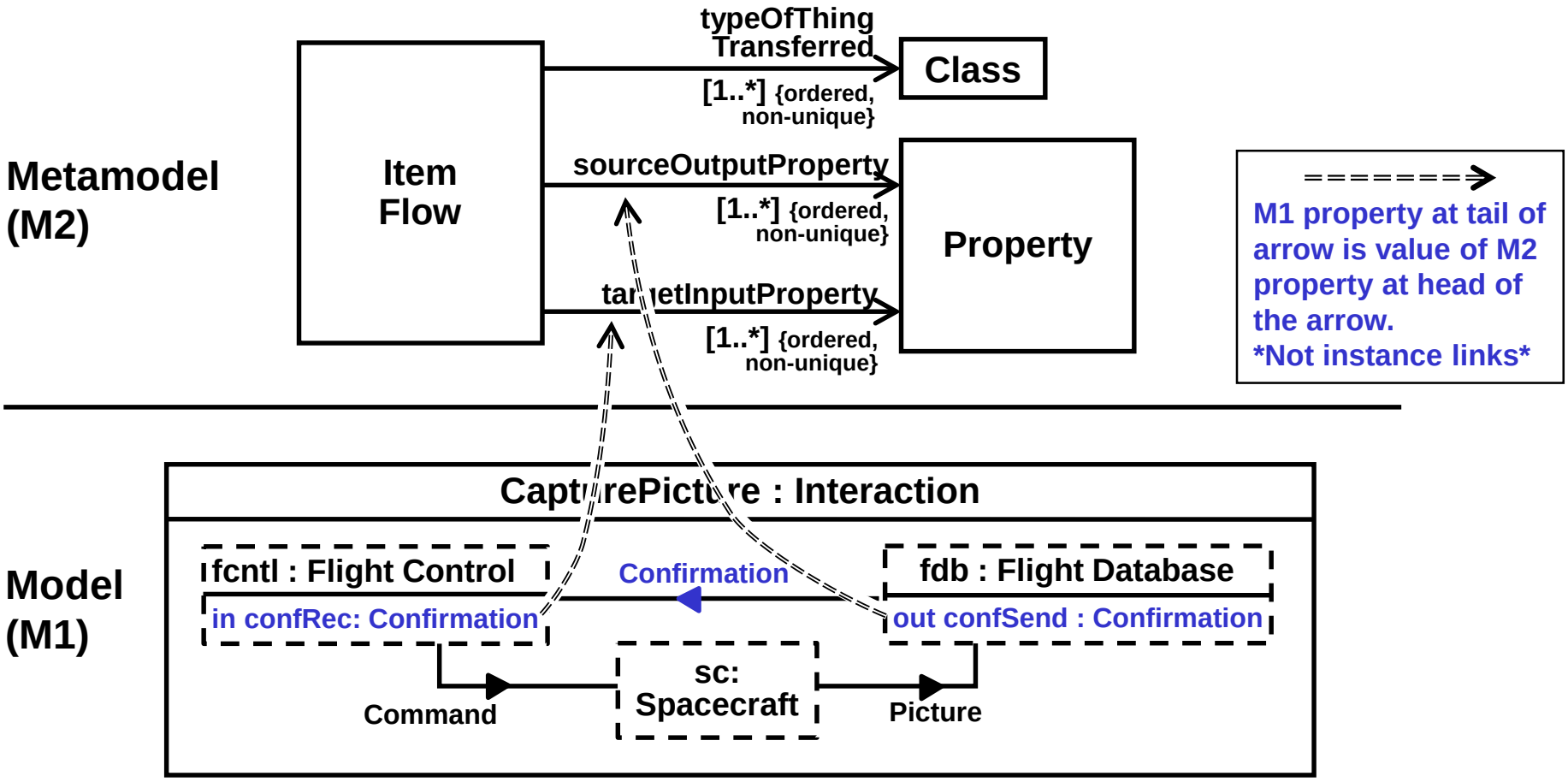
Flow Steps


Metamodel
(M2)
Class
type
owned
Property
*
Property
role
{redefines}
Connector
type
Association
Behavior
ownedStep
*
Step
fromStep
toStep
Succession
type
Interaction
{redefines}
type
Flow
Model
(M1)
CapturePicture
fcntl : Flight
Control
Confirmation
fdb : Flight
Database
: Happens
Before
sc:
Spacecraft
Command
Picture
: HappensBefore
User Model
happens
Before
Behavior
Occurrence
Transfer
Standard
Model Library

Flows & Out/Inputs (OF)



Flows & Out/Inputs (FP)



Overview

- RoadMap
- Motivation
  - Behavior, review
  - Interactions, review
  - **State machines, requirements**
- State Machines Solution
  1. Stimuli = end of transfers (events)
  2. State and transition behaviors
  3. Matching past events to transitions
- Summary

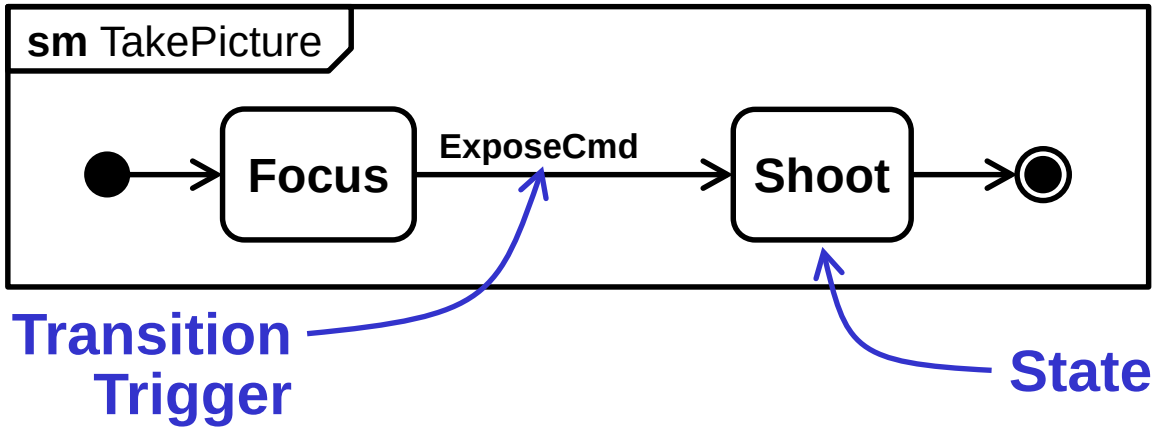
States of What?

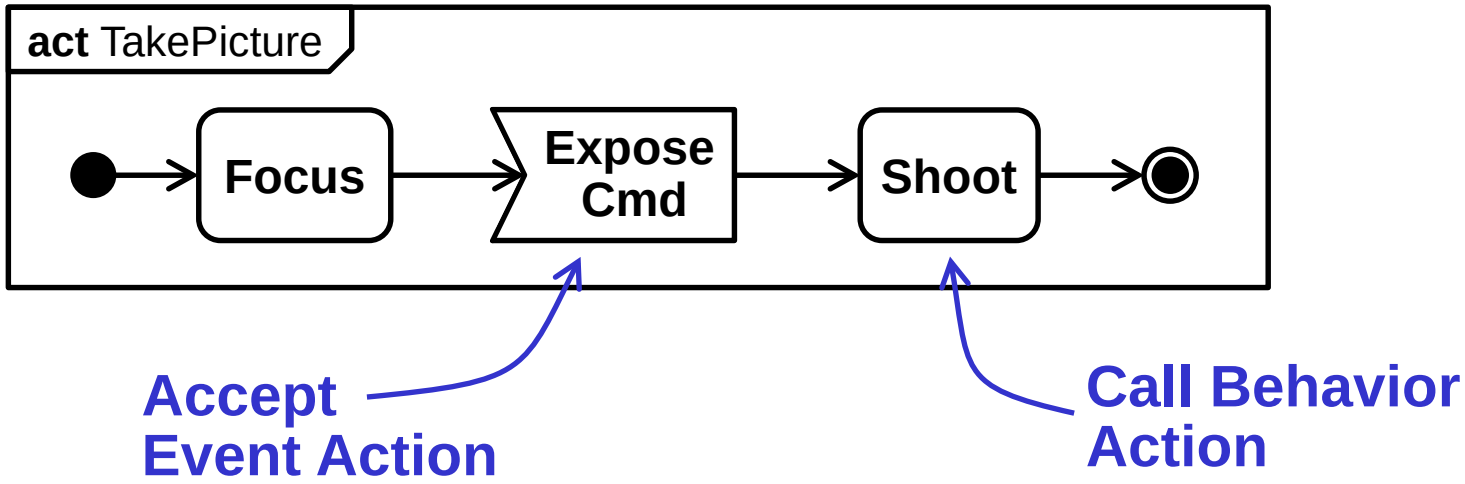
- **Objects, based on properties**
  - **Person in married state = has a spouse.**
- **Behaviors, based on past behavior**
  - **Vending machine in dispensing state after receiving selection and money states.**
- **UML states are mostly behaviors ...**
  - **… tied to objects.**
  - **Weakly include object state invariants.**
- **Both kinds can be in "machines" that react to external stimuli.**

State Machine Problem

- **UML has two ways things can react to external stimuli:**
  - **State Machines have transitions.**
  - **Activities have accept event actions.**
- **Very little integration or reuse.**
  - **Two underlying metamodels/profiles.**
  - **Two representations of reactions.**
  - **Slightly different temporal semantics.**
- **Doubles the effort of learning UML and building analysis tools for them.**

State Machine Problem





State Machine Requirements

1. **Must selectively react to stimuli ("events").**
   - **Based on kind of stimulus and …**
   - **… current & previous stimuli/reactions ("states")**
2. **Must simplify reaction behaviors, splitting them up …**
   - **by state and between states (transitions).**
   - **within states.**
3. **Must react to past events**
   - **Can have complicated reaction rules to events in the past.**

Overview

- RoadMap
- Motivation
  - Behavior, review
  - Interactions, review
  - State machines, requirements
- **State Machines Solution**
  1. **Stimuli = end of transfers (events)**
  2. State and transition behaviors
  3. Matching past events to transitions
- Summary

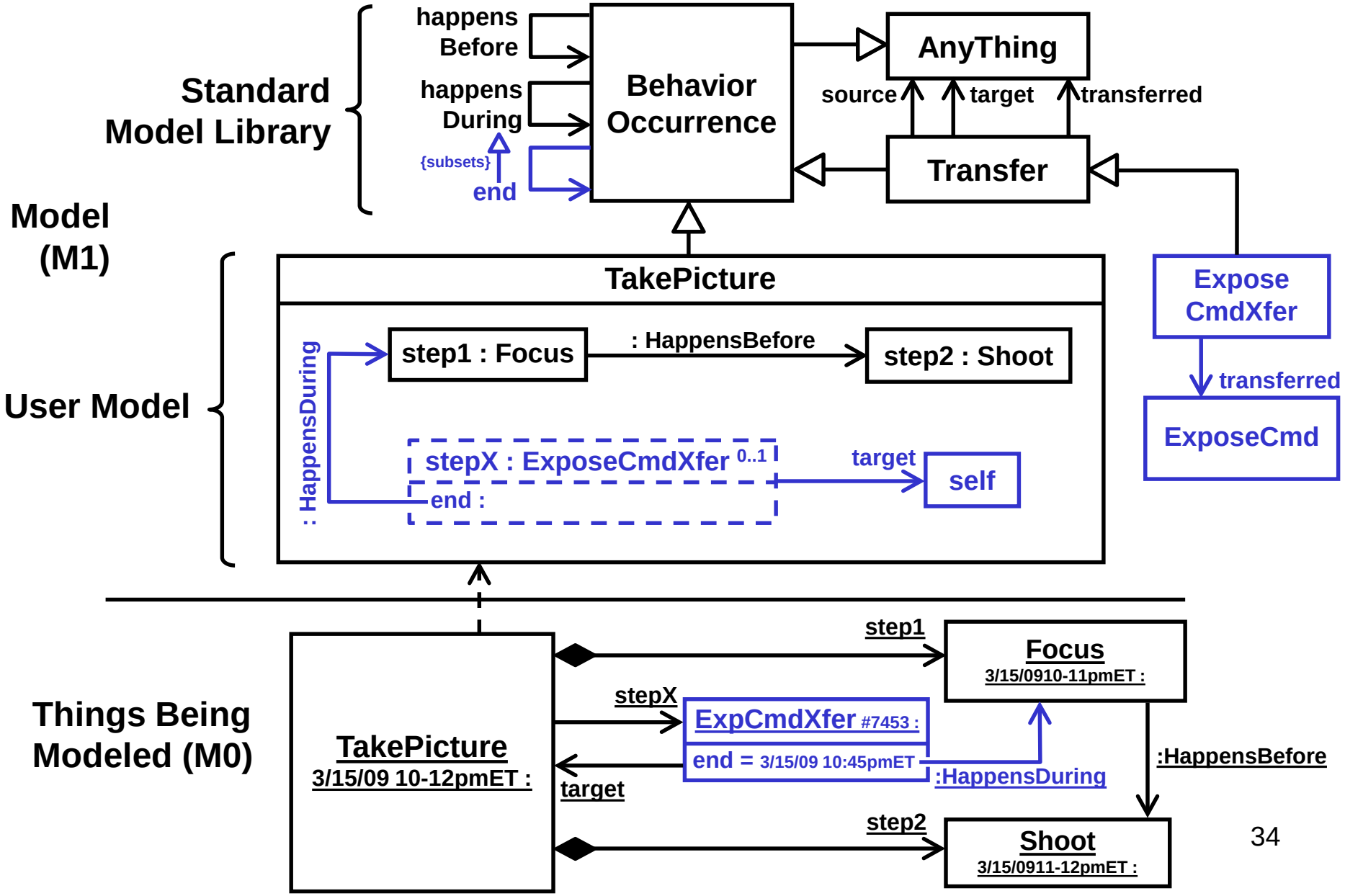
State Machine Solution (Part 1)
(Reacting to stimuli)

- **Reaction depends on current state.**
  - **Change states (leave current one, enter another).**
  - **Re-enter current state.**
  - **Do nothing.**
- **Events can arrive during or before states expecting them.**
  - **Addressed separately.**

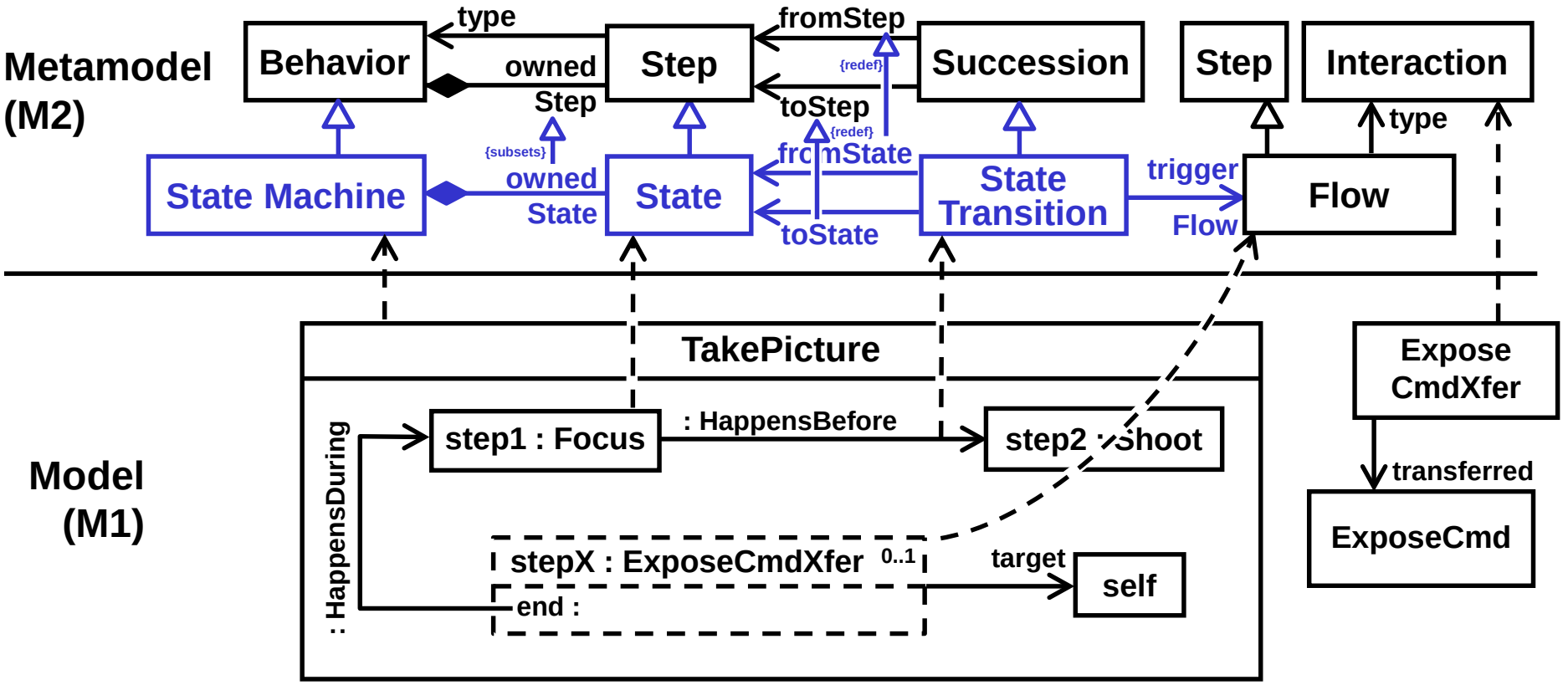
State Machine Solution (Part 1)
(Reacting to stimuli)

- **UML events = things "arriving" at objects**
  - **Signals, operation calls**
  - **Not events happening externally**
    - **Except unmodeled "changes" to anything.**
- **Treat as ends of transfers targeting objects.**
  - **Receiver doesn't specify sender.**

UML Events = Ends of Transfers



State Machines (M2)



- **Transitions are successions that …**
  - **go out of steps …**
  - **that interactions (triggers) end during …**
  - **that target the machine.**

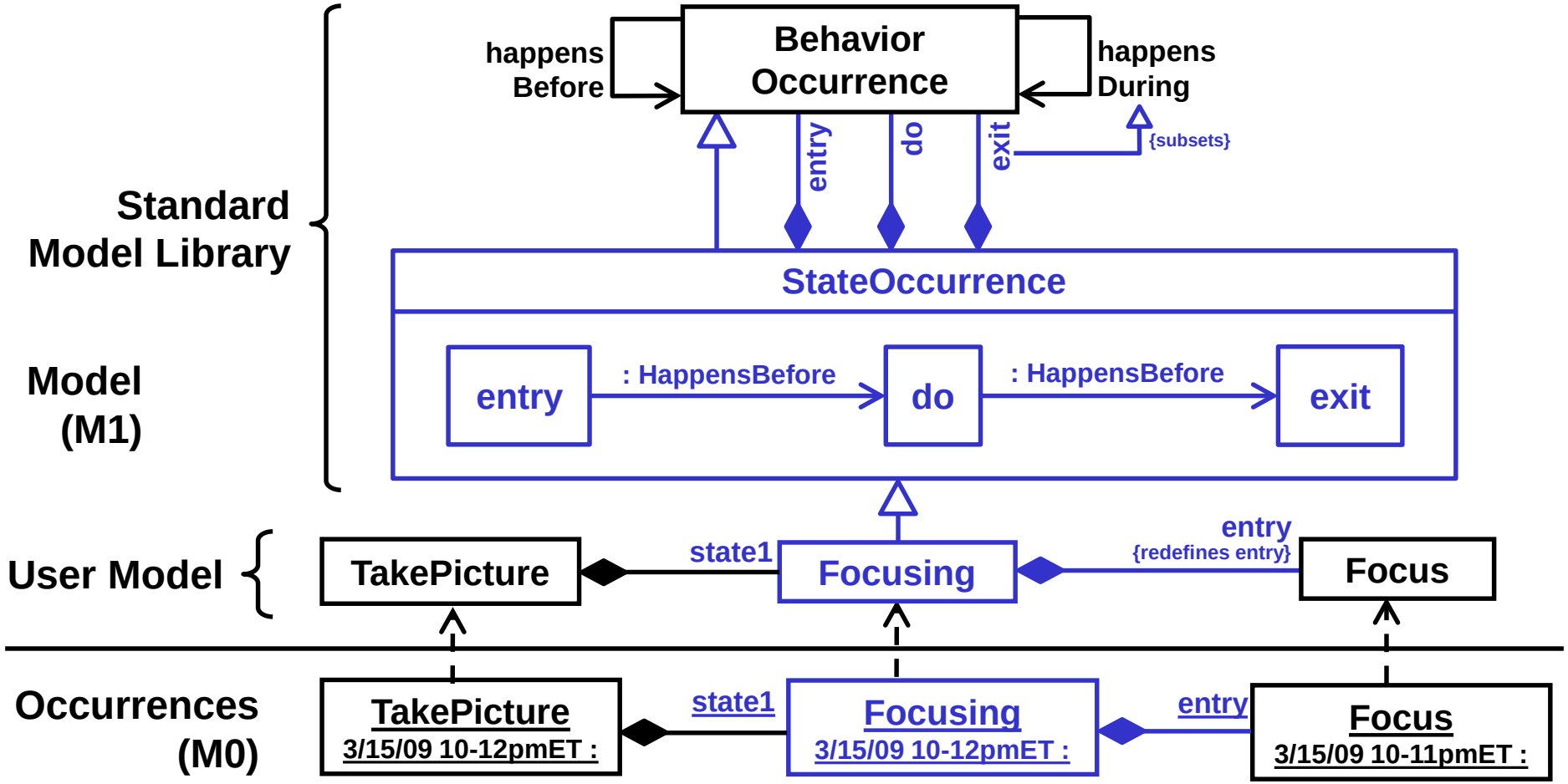
Overview

- RoadMap
- Motivation
  - Behavior, review
  - Interactions, review
  - State machines, requirements
- State Machines Solution
  1. Stimuli = end of transfers (events)
  2. **State and transition behaviors**
  3. Matching past events to transitions
- Summary

State Machine Solution (Part 2)
(Simplifying reaction behaviors)

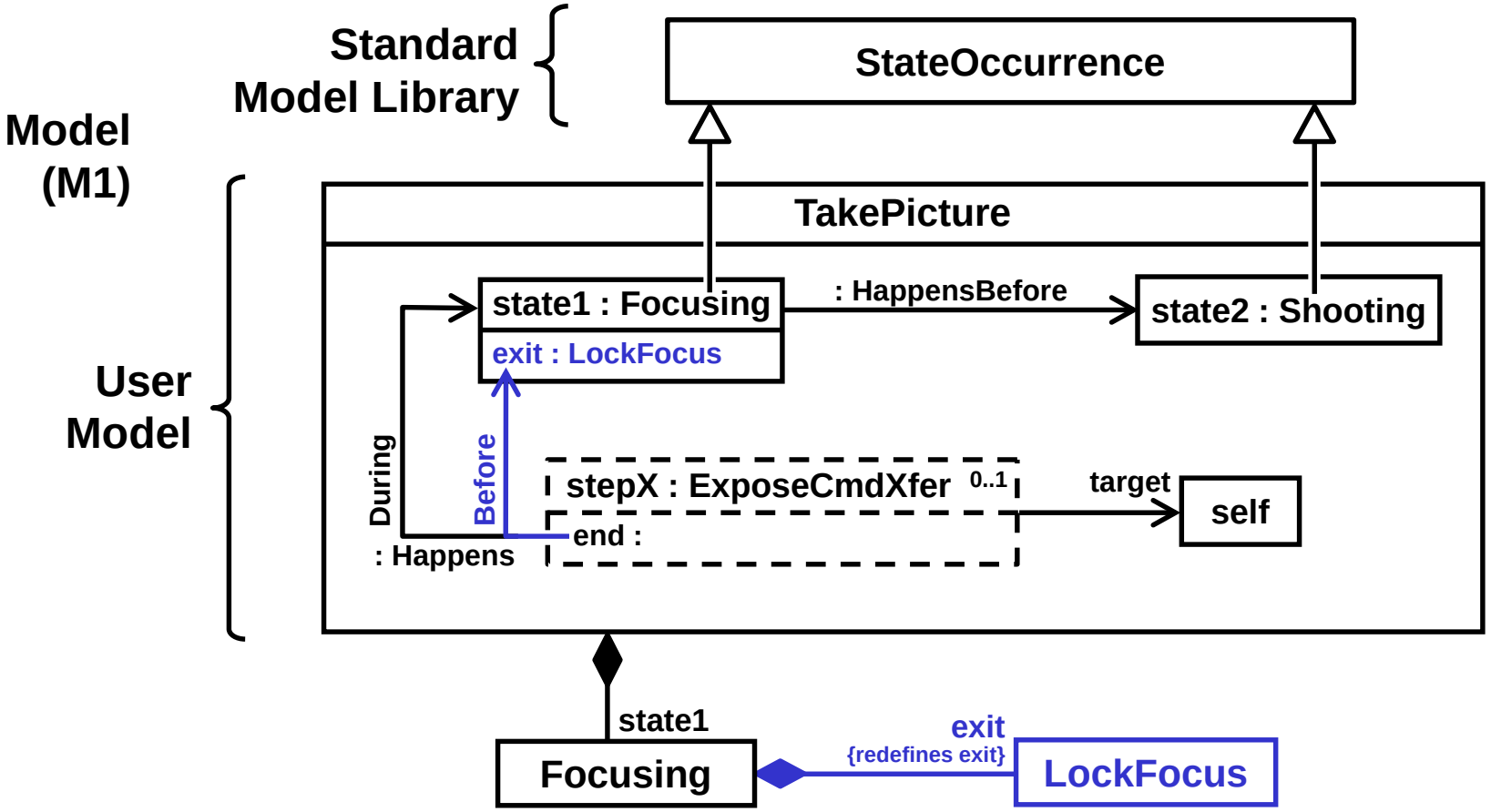
- States have entry, do, and exit behaviors
  - Happen going into, being in, going out of states.
  - No other state behaviors, simplifies behavior modeling.
  - Only do behaviors can be stopped by events.
- Transitions have effect behaviors
  - Happen after source state exit and before target state entry.

State Behaviors (M1)



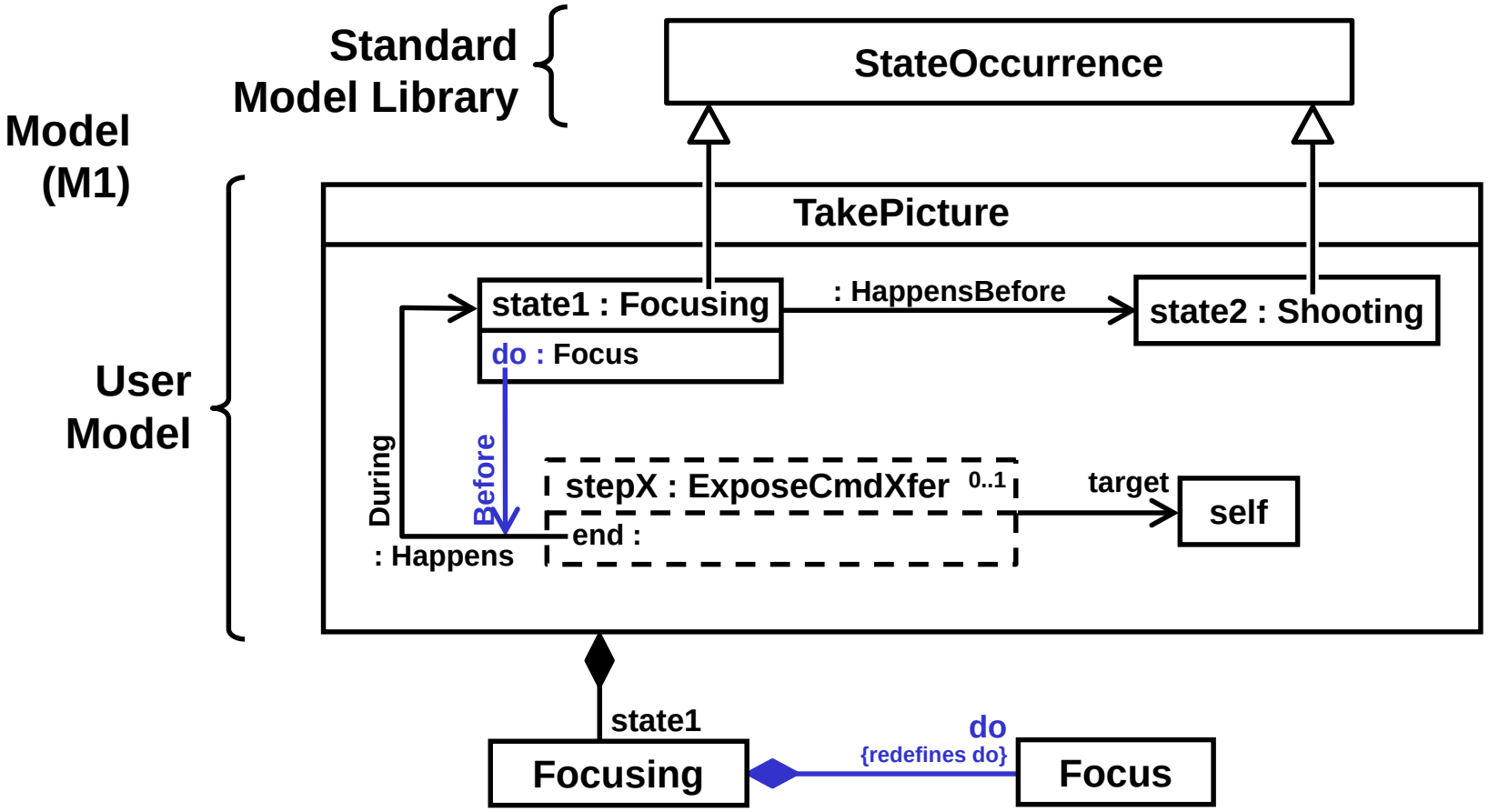
- **State occurrences:**
  - **Are behavior occurrences typing state properties...**
  - **with exactly three step properties ordered in time**

Triggering Exit Behaviors



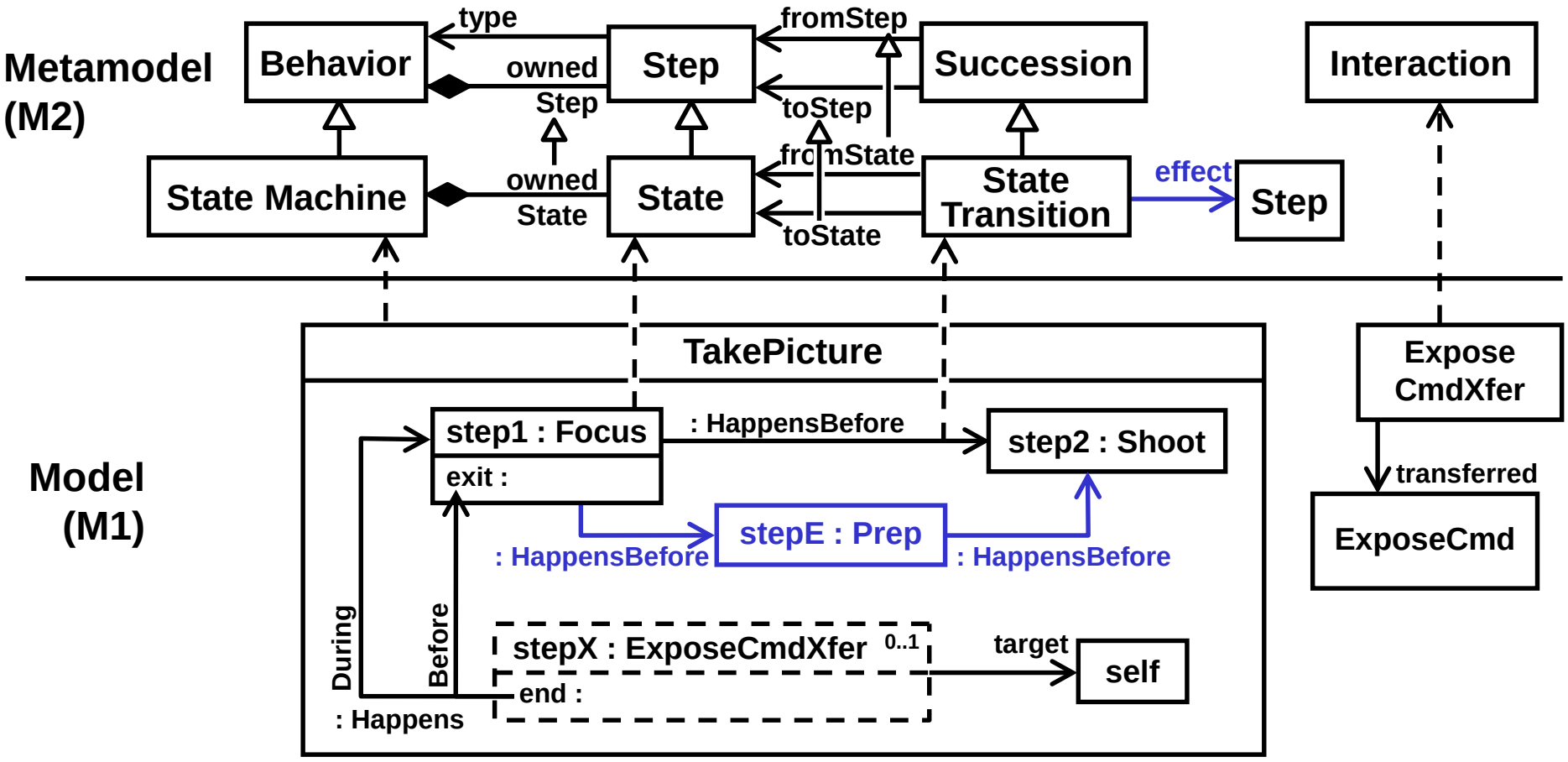
- **Exit behaviors happen after triggers end.**

Aborting Do Behaviors



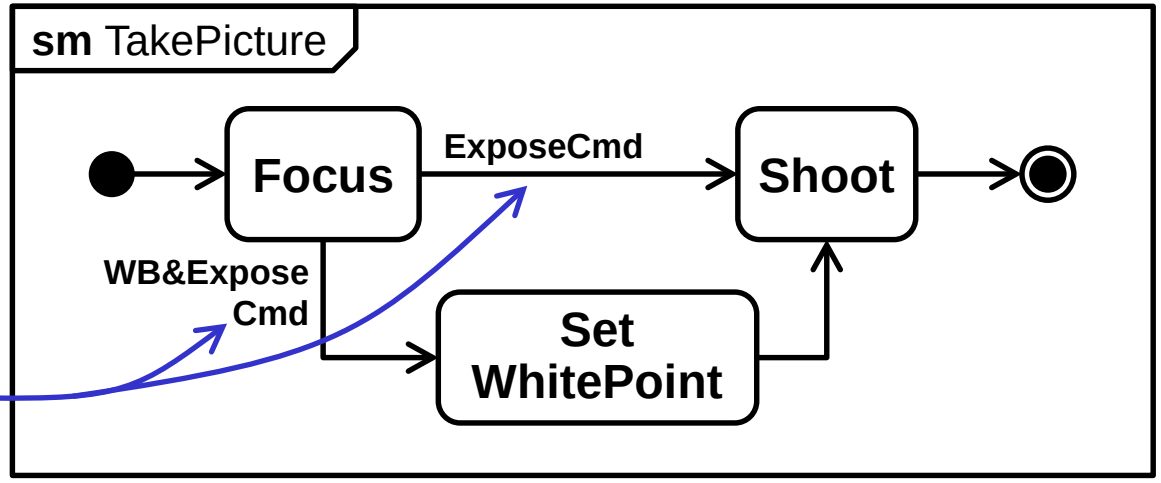
- Do behaviors stop before event arrives
  - Even if they aren't finished.
  - Assumes do behaviors are abortable.

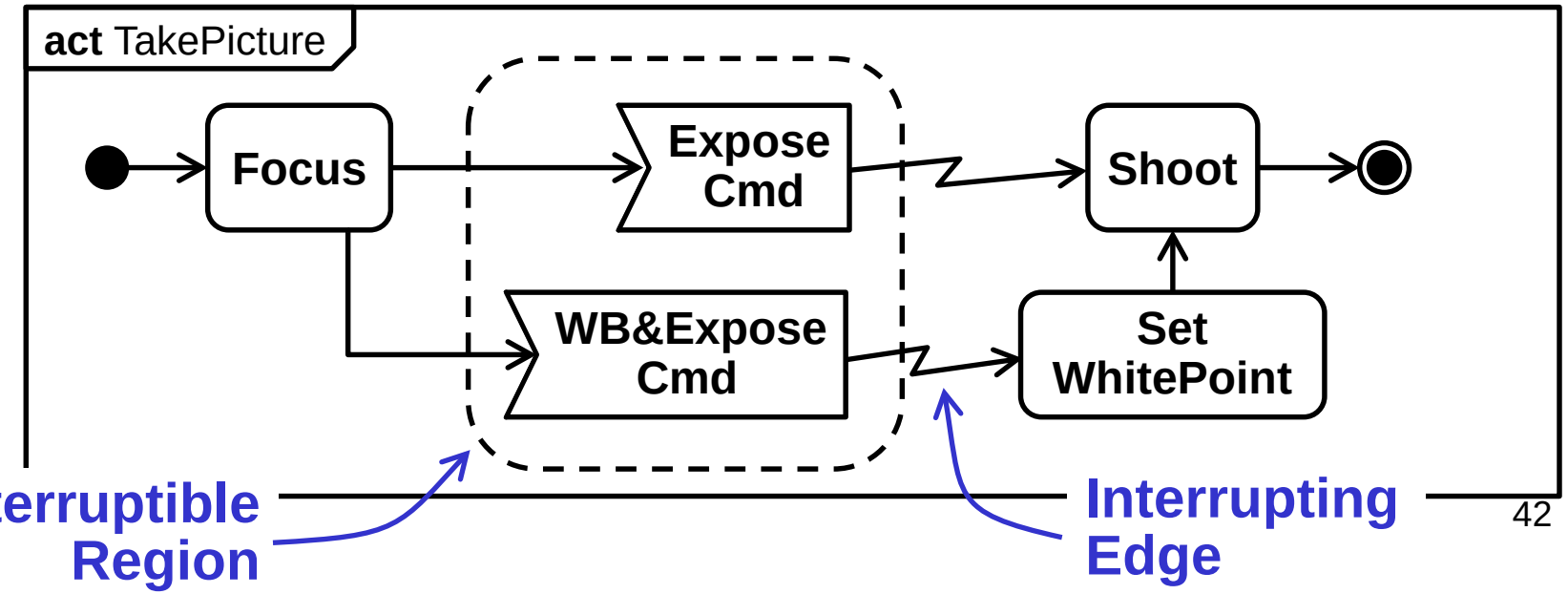
Transition Behaviors



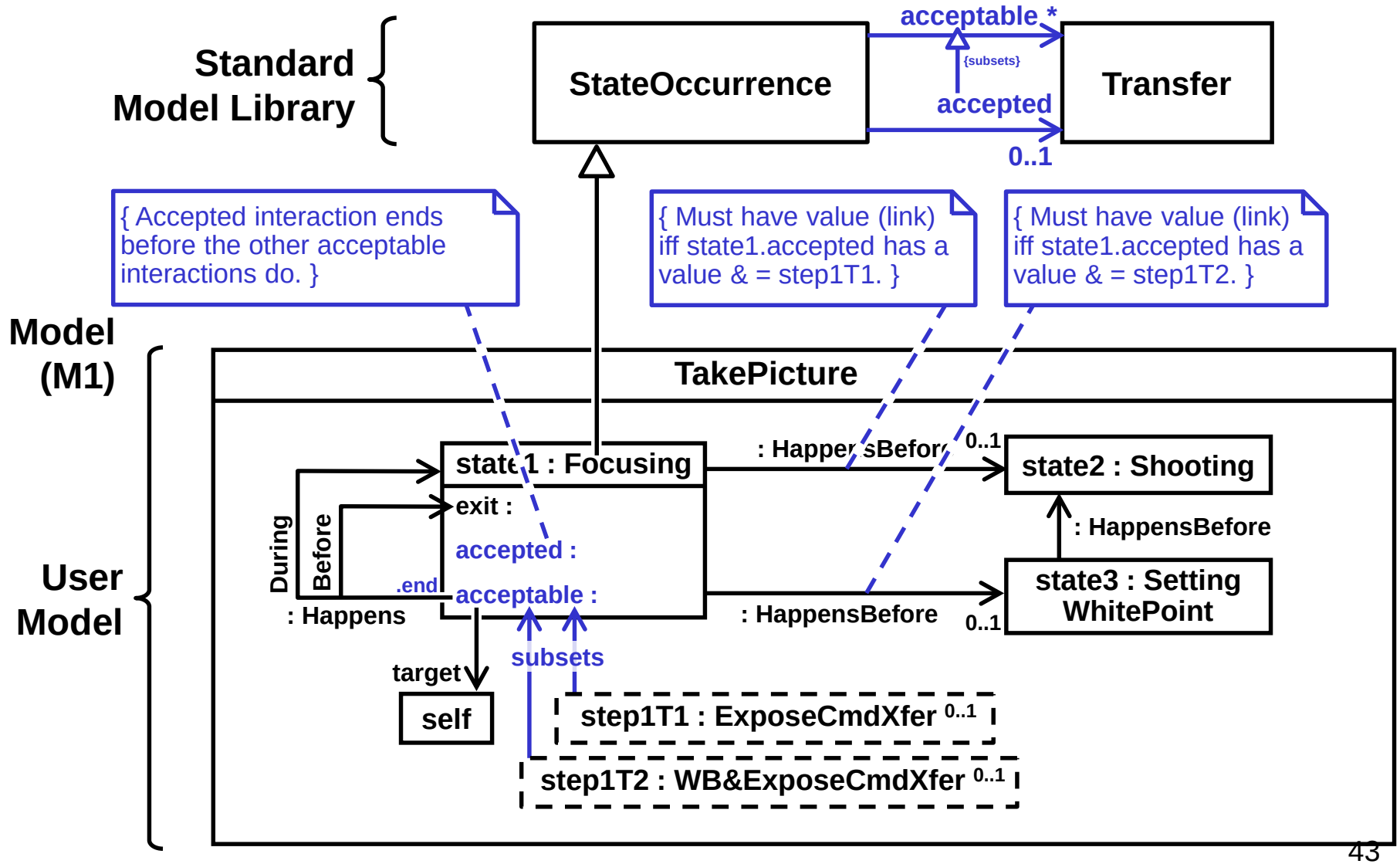
- **Transitions can specify behaviors to happen in between states.**

State Machine Problem (#2)

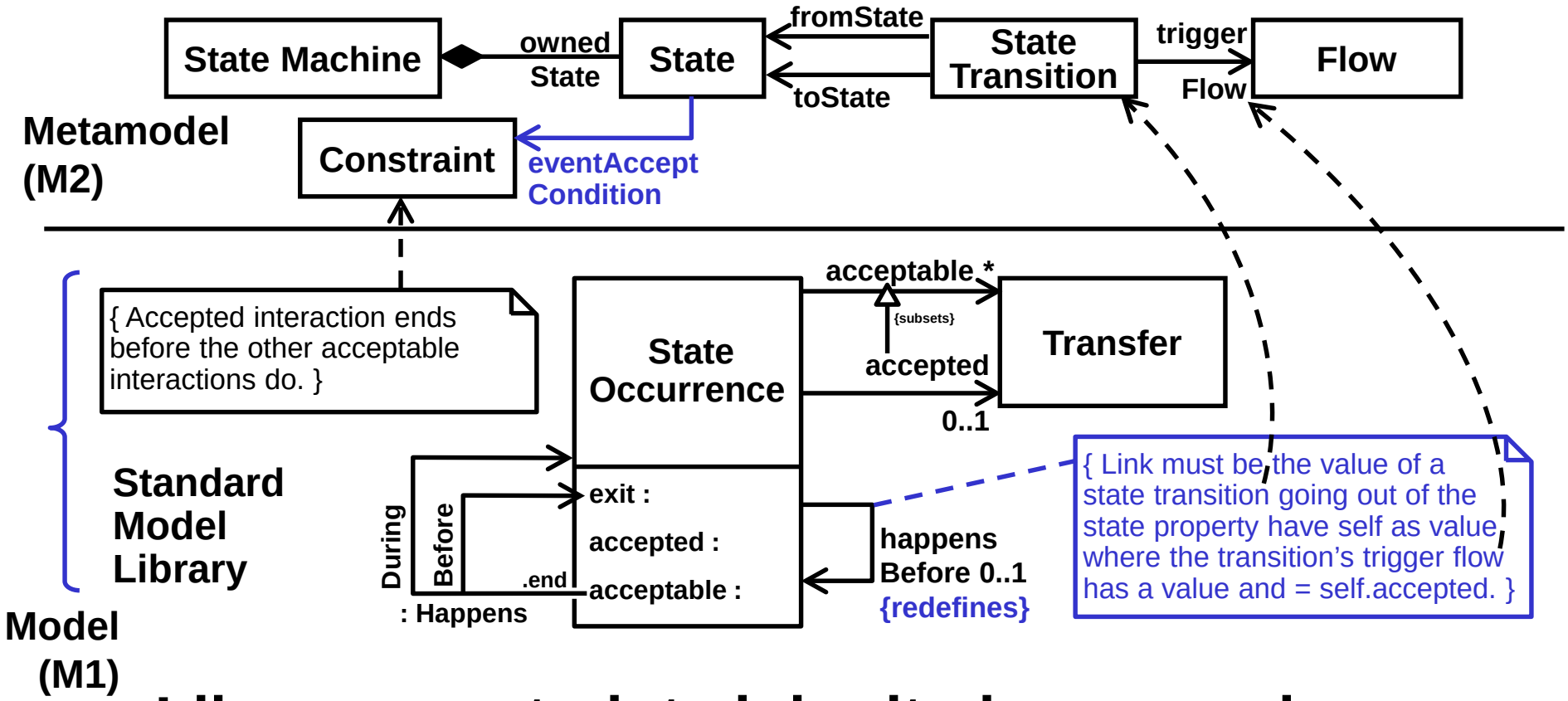




Competing Transitions (M1)



Competing Transitions (M1Lib/M2)



- **Library constraints inherited or reused**
  - **Acceptable/exit timing moved to library.**
  - **Transition constraints use M2.**
  - **Commonly used acceptance constraints.**

For all models

For models to use as needed

Overview

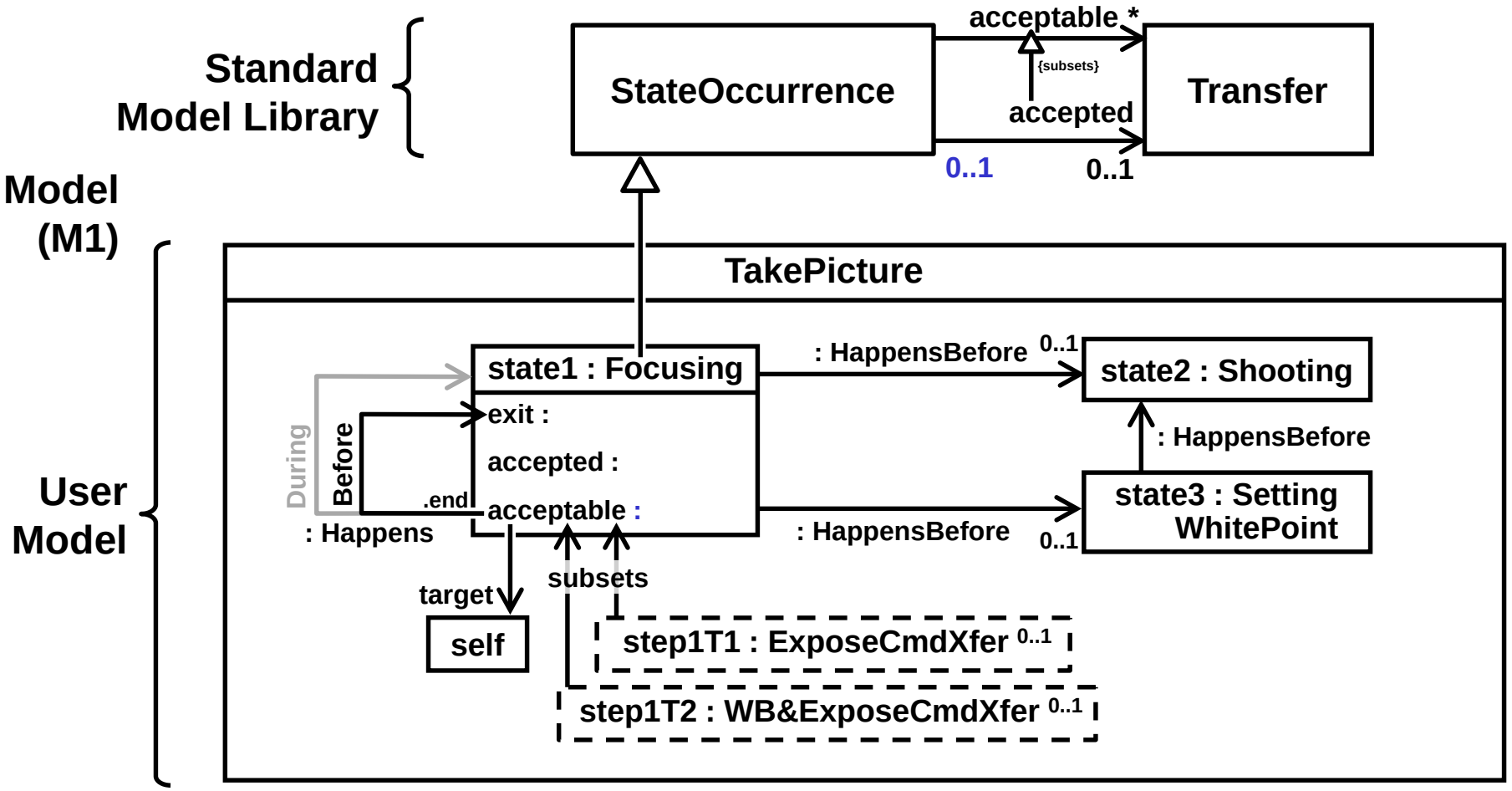
- RoadMap
- Motivation
  - Behavior, review
  - Interactions, review
  - State machines, requirements
- **State Machines Solution**
  1. Stimuli = end of transfers (events)
  2. State and transition behaviors
  3. **Matching past events to transitions**
- Summary

State Machine Problem (Part 3)

(Reacting to past events)

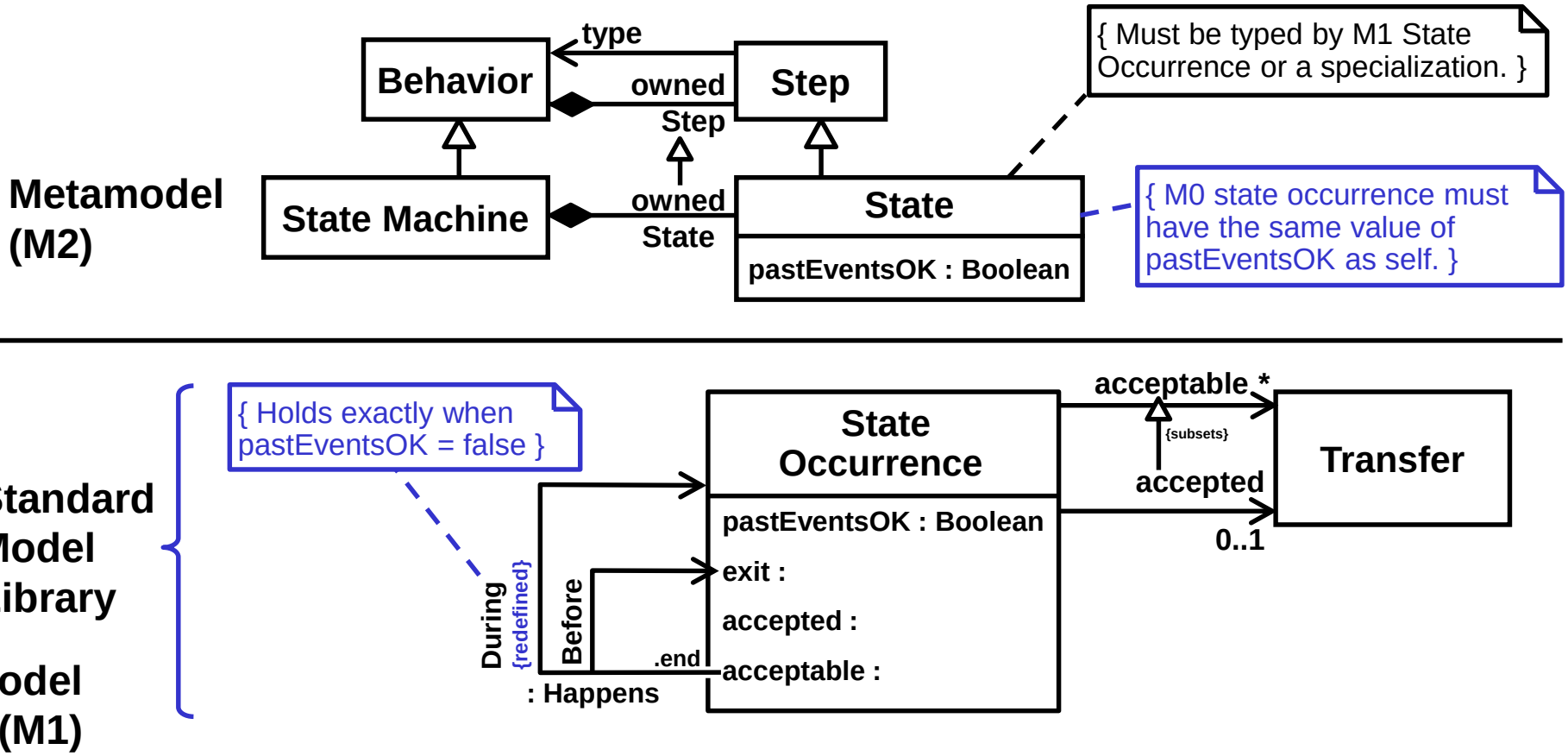
- **So far, states are only triggered by events that arrive during the state.**
- **Want to enable states to be triggered by events that arrive before the state.**

Past Events (M1)



- **Events arriving before state are acceptable.**
  - **But can only be accepted once.**

Past Events (M1 Library / M2)



- **HappensDuring redefined to apply as indicated by boolean.**

State Machine TBD

- **Concurrent regions.**
- **Multiple machines and activities using the same events.**
  - **Objects with multiple behaviors.**
- **More complex event handling.**
- **Pulling from buffer, rather than matching (maybe).**

Overview

- RoadMap
- Motivation
  - Behavior, review
  - Interactions, review
  - State machines, requirements
- State Machines Solution
  1. Stimuli = end of transfers (events)
  2. State and transition behaviors
  3. Matching past events to transitions
- **Summary**

Summary

- **Unify reacting to events using**
  - **Transfer ends as events**
  - **Properties for state behaviors.**
- **Model of event processing**
  - **Matching events by constraints …**
  - **… easier for end user than event handling procedures.**
- **Speeds learning and analysis integration.**

More Information

- **Intro to Behavior as Composite Structure**
  - http://doc.omg.org/ad/2018-03-02
- **Interaction as Composite Structure**
  - http://doc.omg.org/ad/18-06-11
- **Object-orientation as Composite Structure**
  - http://doc.omg.org/ad/18-09-07
- **Earlier slides (more onto, includes interactions)**
  - http://conradbock.org/bock-ontological-behavior-modeling-jpl-slides.pdf
- **Paper:** http://dx.doi.org/10.5381/jot.2011.10.1.a3
- **Application to BPMN:** http://conradbock.org/#BPDM
- **KerML: Contact Chas Galey** charles.e.galey@lmco.com