Position Statement

3rd Bellairs Workshop

Thomas Kühne
Darmstadt University of Technology

Who am I?

- Where have I been?
  - » TH Darmstadt
  - » Staffordshire University
  - » University of Kaiserslautern
  - » University of Mannheim
  - » TU Darmstadt
- What are my interests?
  - » metamodeling
  - » language extensions
  - » domain-specific languages/modeling
  - » model-driven development
  - » transformations

Who am I?

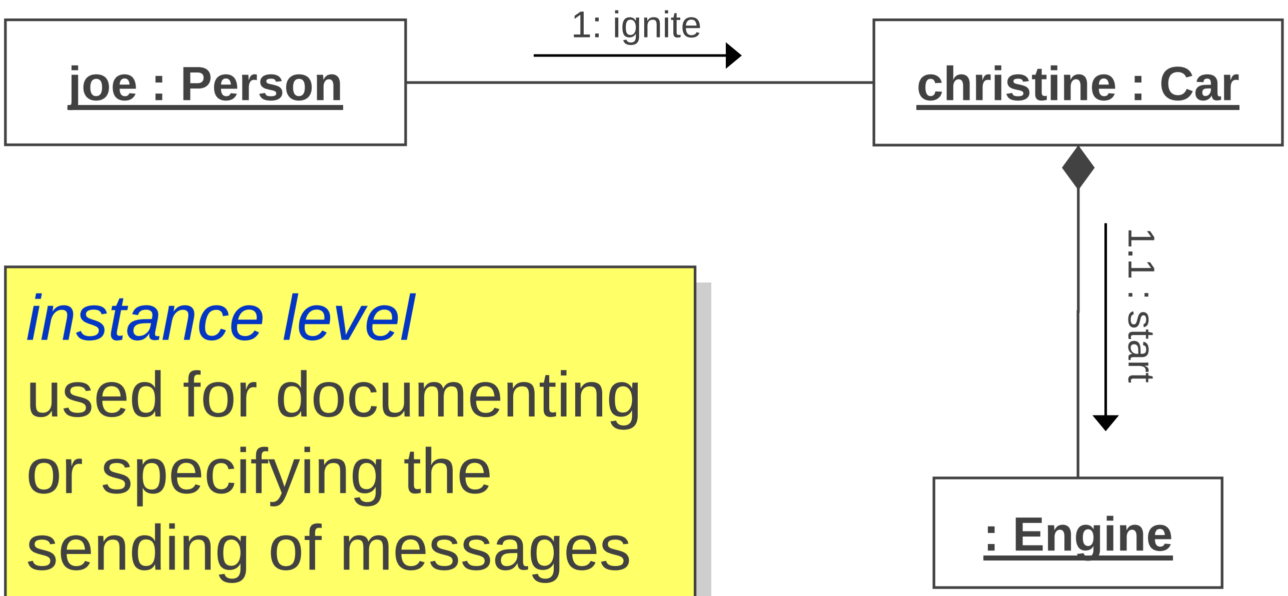
- What have I done?
  - » foundations of metamodeling
  - » metamodeling architectures (tools)
  - » domain metamodeling (deep characterization)
  - » stereotype usage
  - » profile definition
  - » architecture stratification

Interesting Problems

- What is an “Abstract Object”?
  - semantics
  - difference to objects / classes
- How to avoid “Accidental Complexity”?
  - language / paradigm limitations
  - better support

Abstract Objects: Example

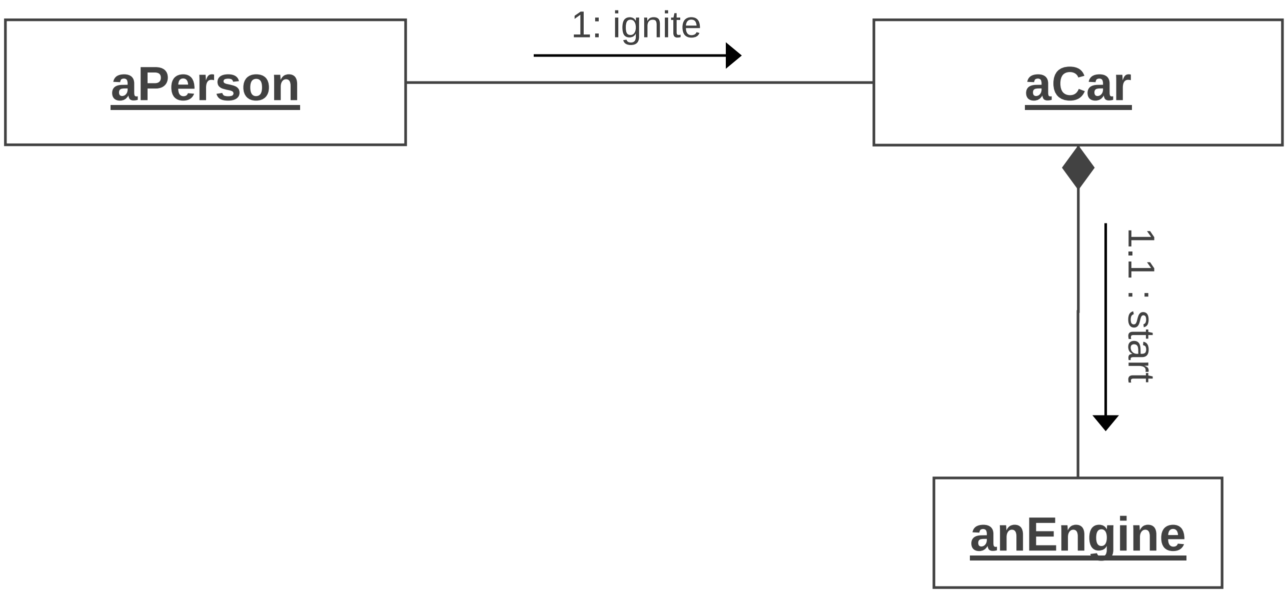
Collaboration Instance



instance level
used for documenting or specifying the sending of messages within an object scenario

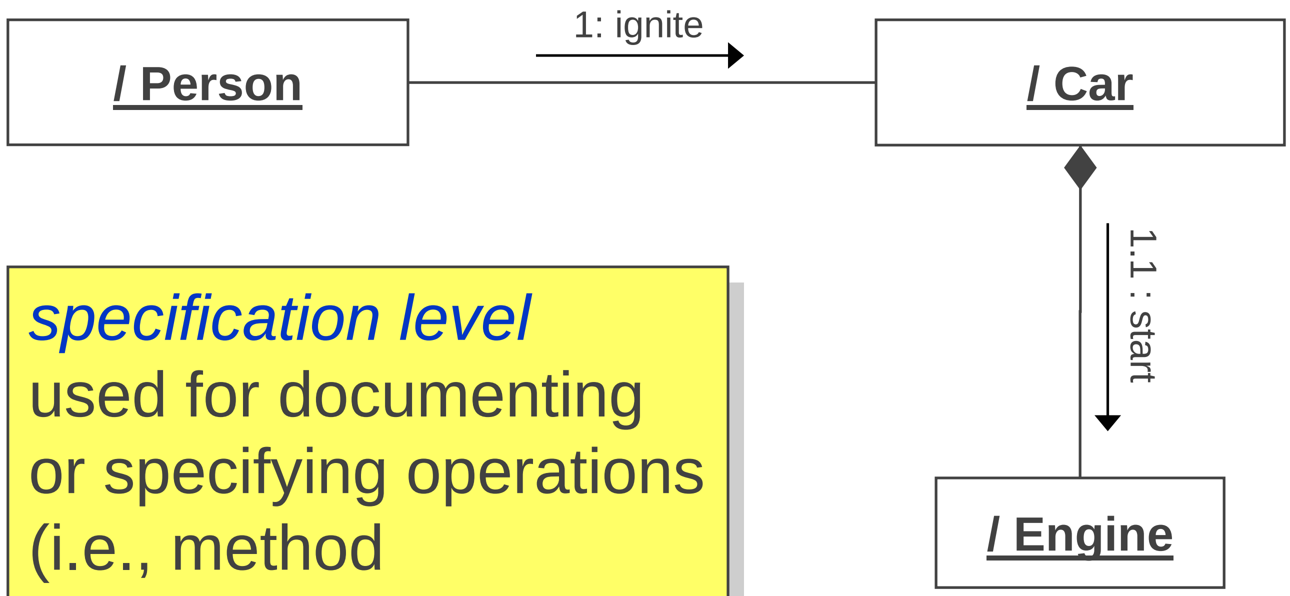
Abstract Objects: Example

Collaboration ~~Instance~~



Abstract Objects: Example

Collaboration



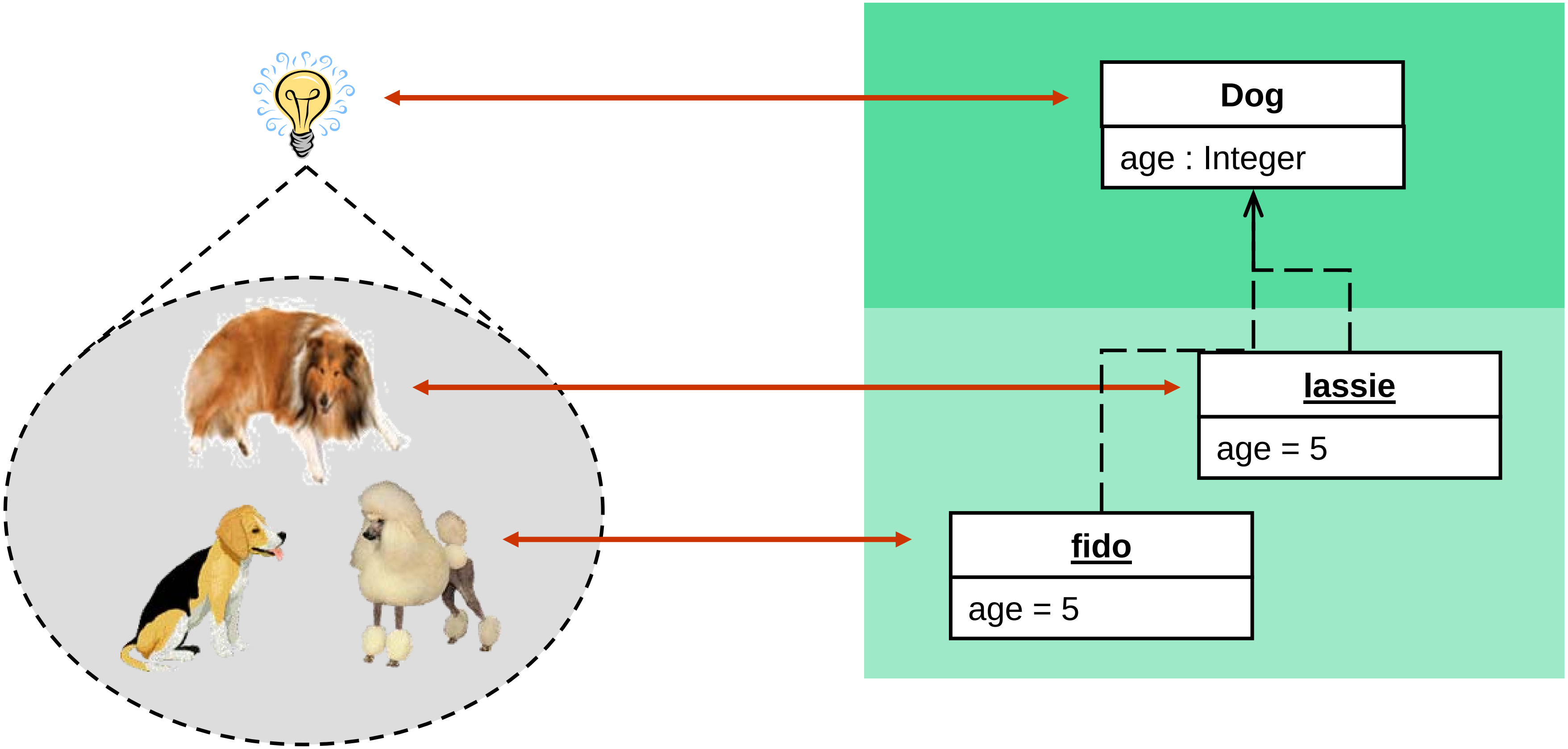
specification level used for documenting or specifying operations (i.e., method implementations)

What Are They?

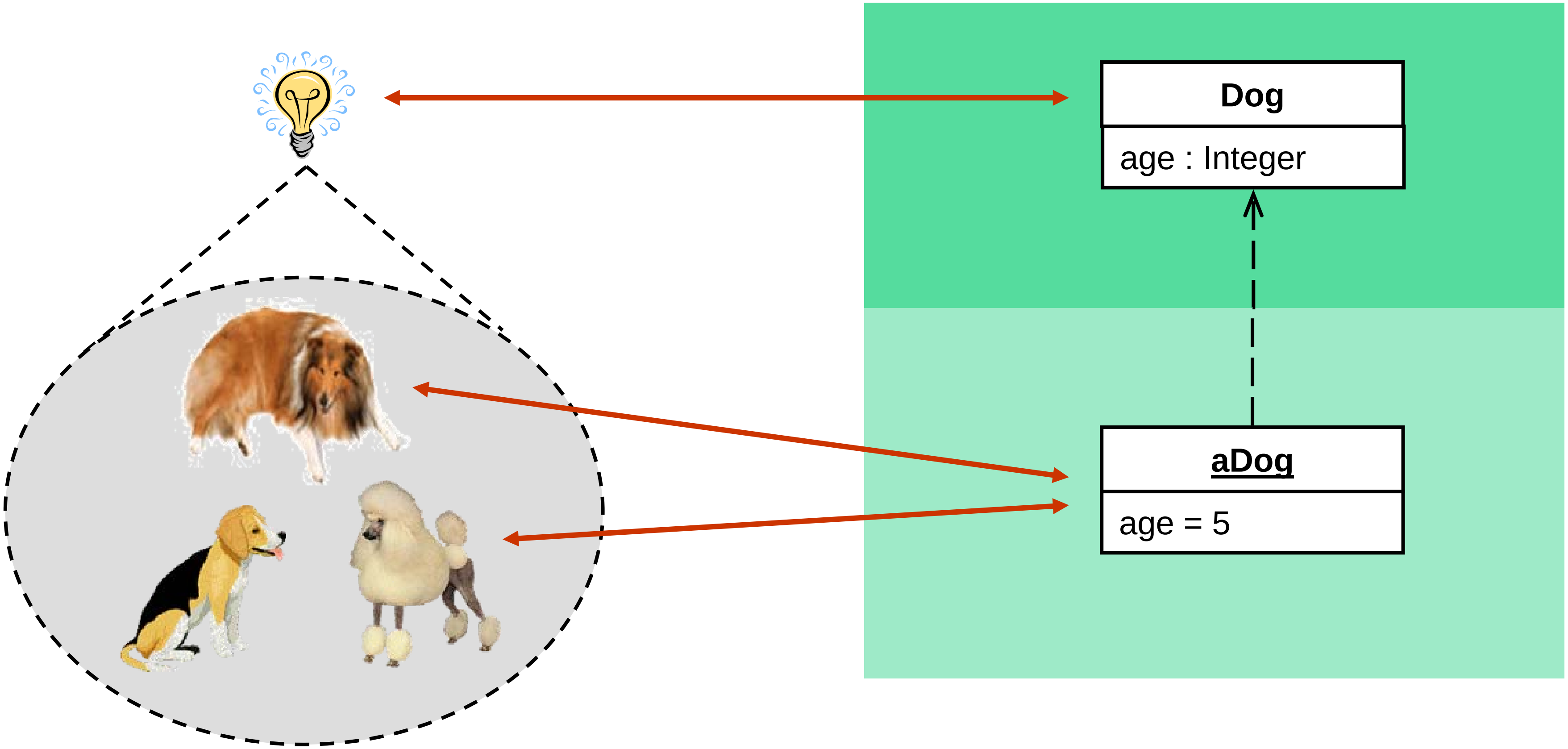
An “Abstract Object” is…

- an instance of an abstract class
- something “in between” classes and objects
  - » → level $M_{1/2}$?
  - » similar to a classifier but not a class
- a placeholder for a pure object
  - » (typed) variable
  - » “hole” in a schema

Class / Object Semantics



Class / Object Semantics



What Are They?

"Abstract Objects" are…

- objects which represent more than one element of the universe of discourse

Object

» age is a (injective) function

» object structure is isomorphic to (a subset of) the UoD

Abstract Object

» age is a relation

» object structure is homomorphic to (a subset of) the UoD

Accidental Complexity

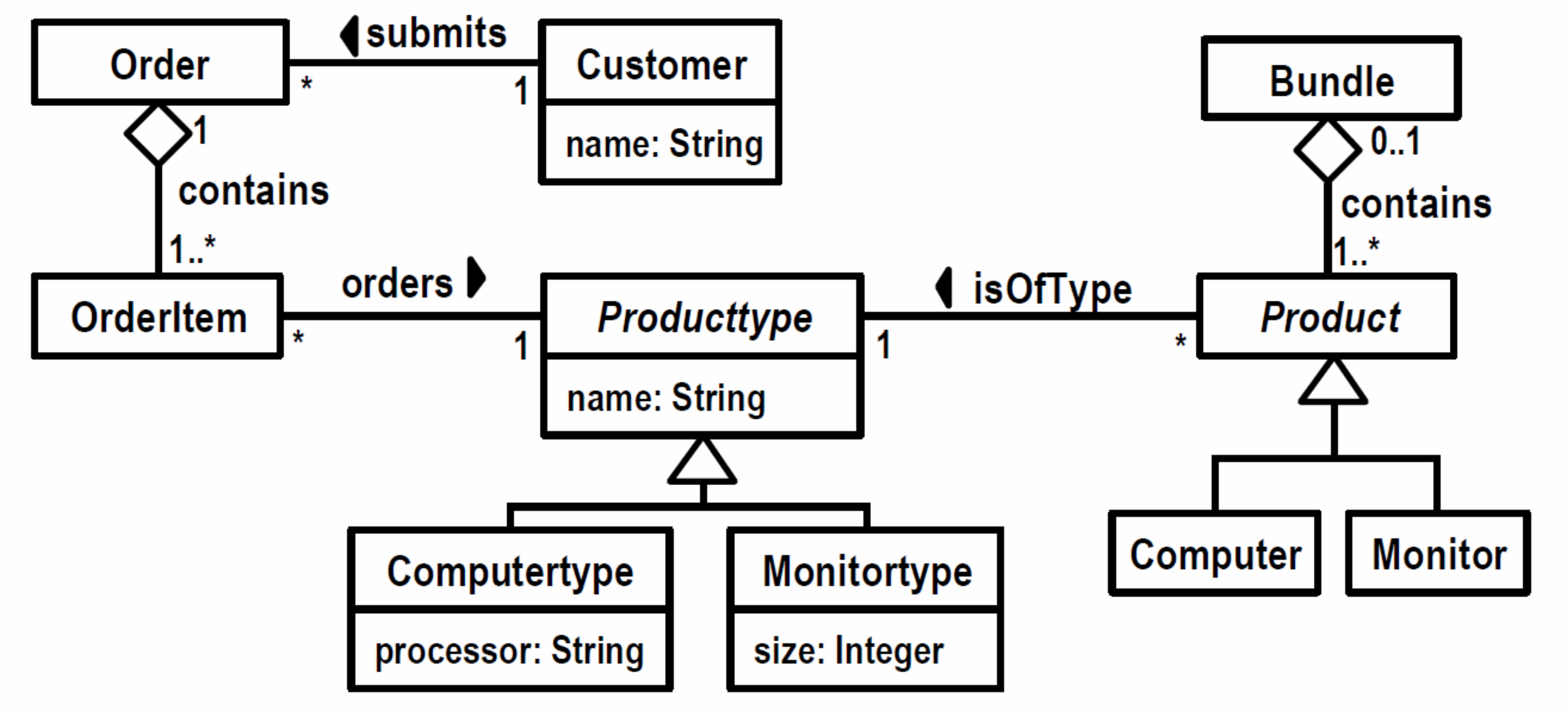
Complexity of a model may come from two sources

- the modeled original is inherently complex
  → inherent complexity

- the modeling paradigm / notation is adding complexity
  → accidental complexity
  - mismatch of problem & solution
  - structural complexity worse than notational complexity

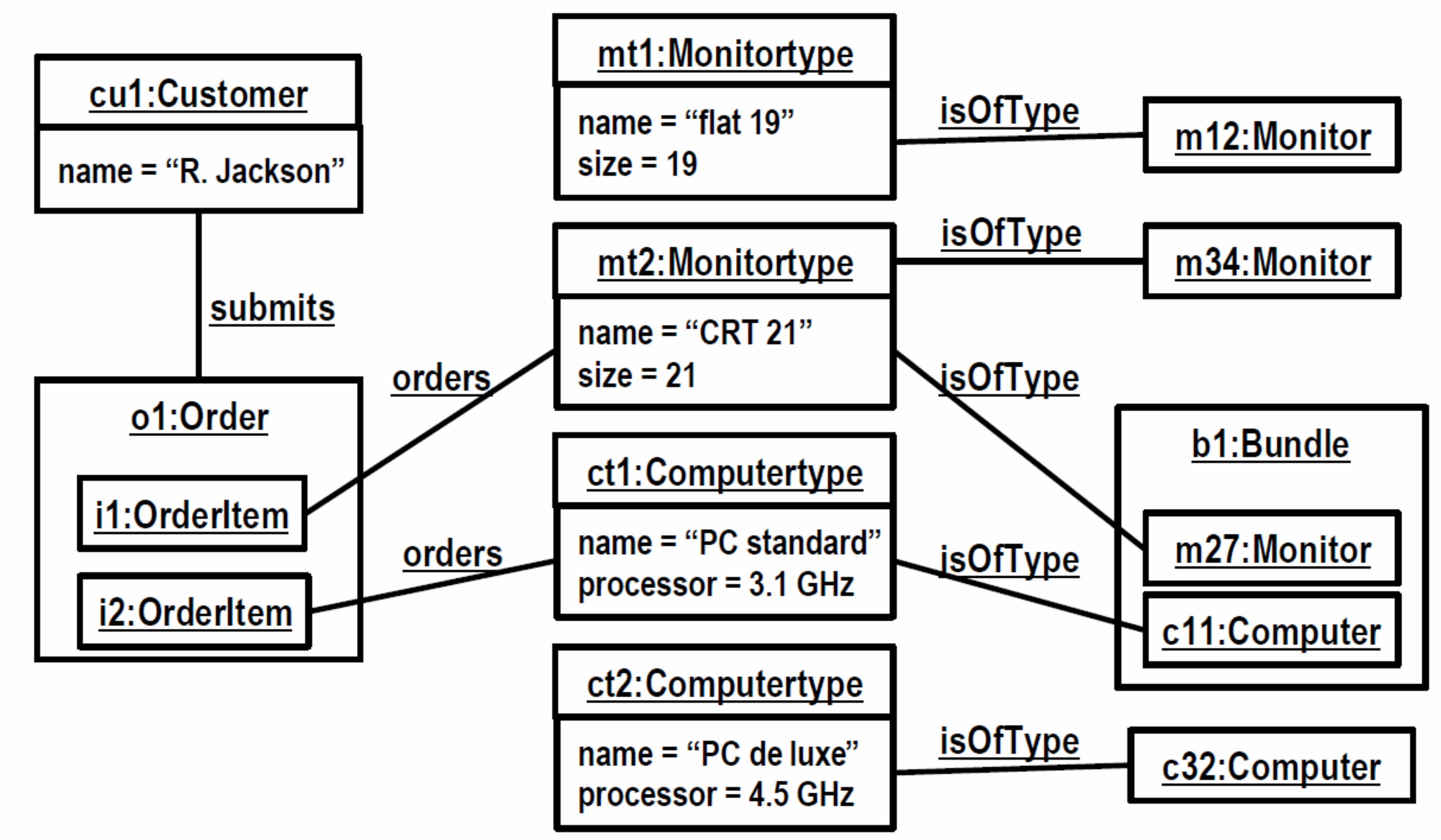
Example (Types)

- Computer hardware sales ([Engels et al. 2005])

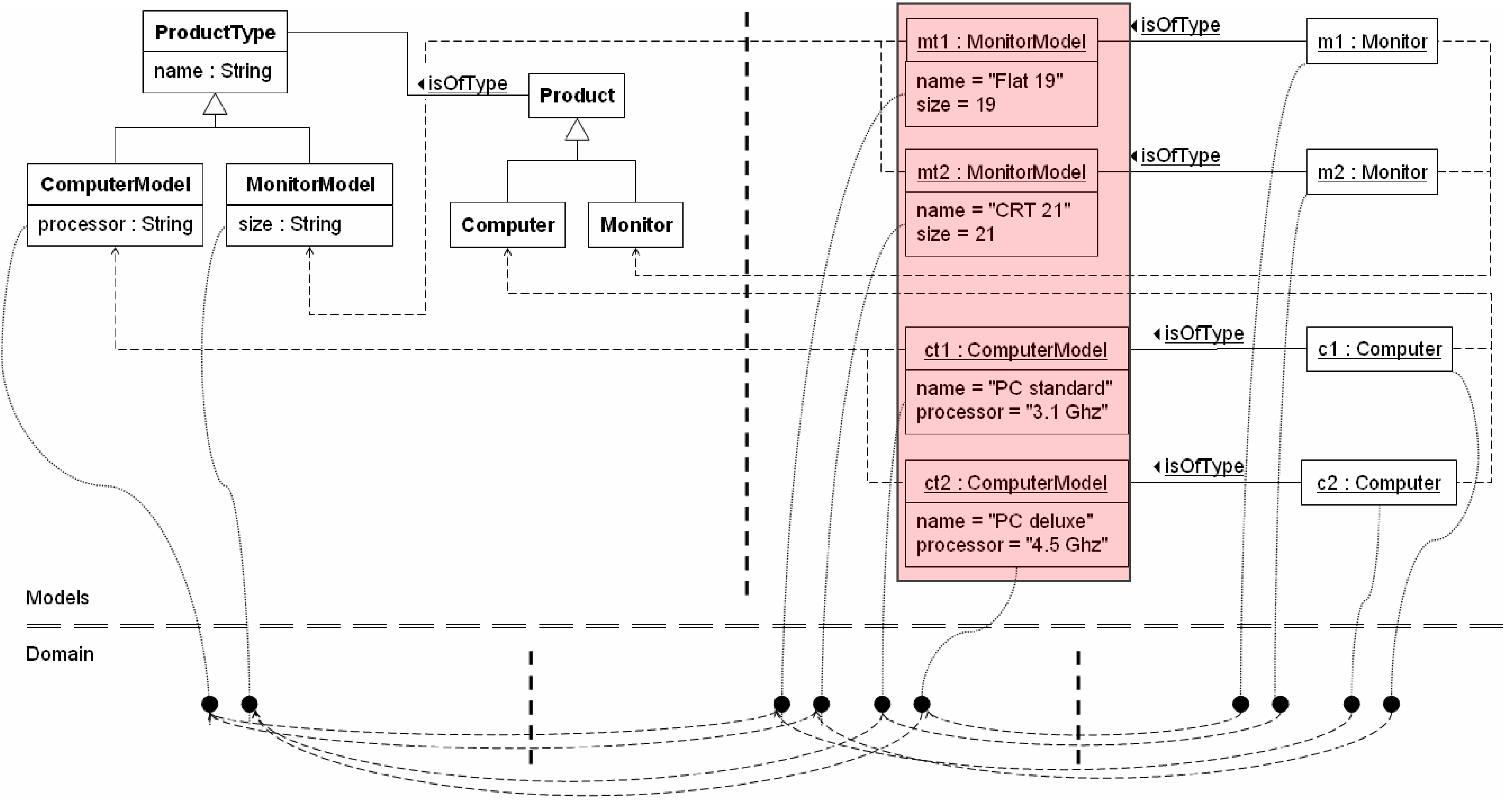


Example (Instances)

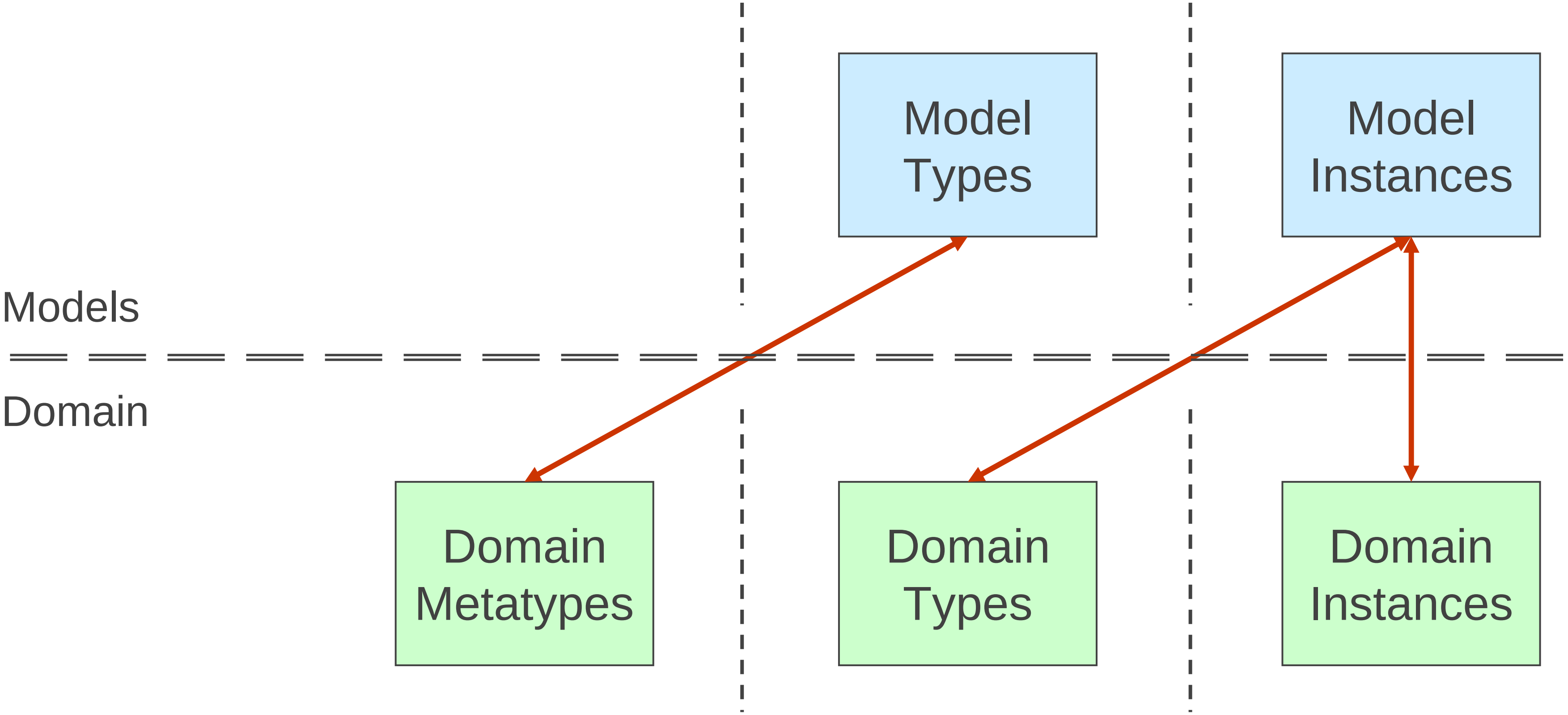
- Computer hardware sales ([Engels et al. 2005])



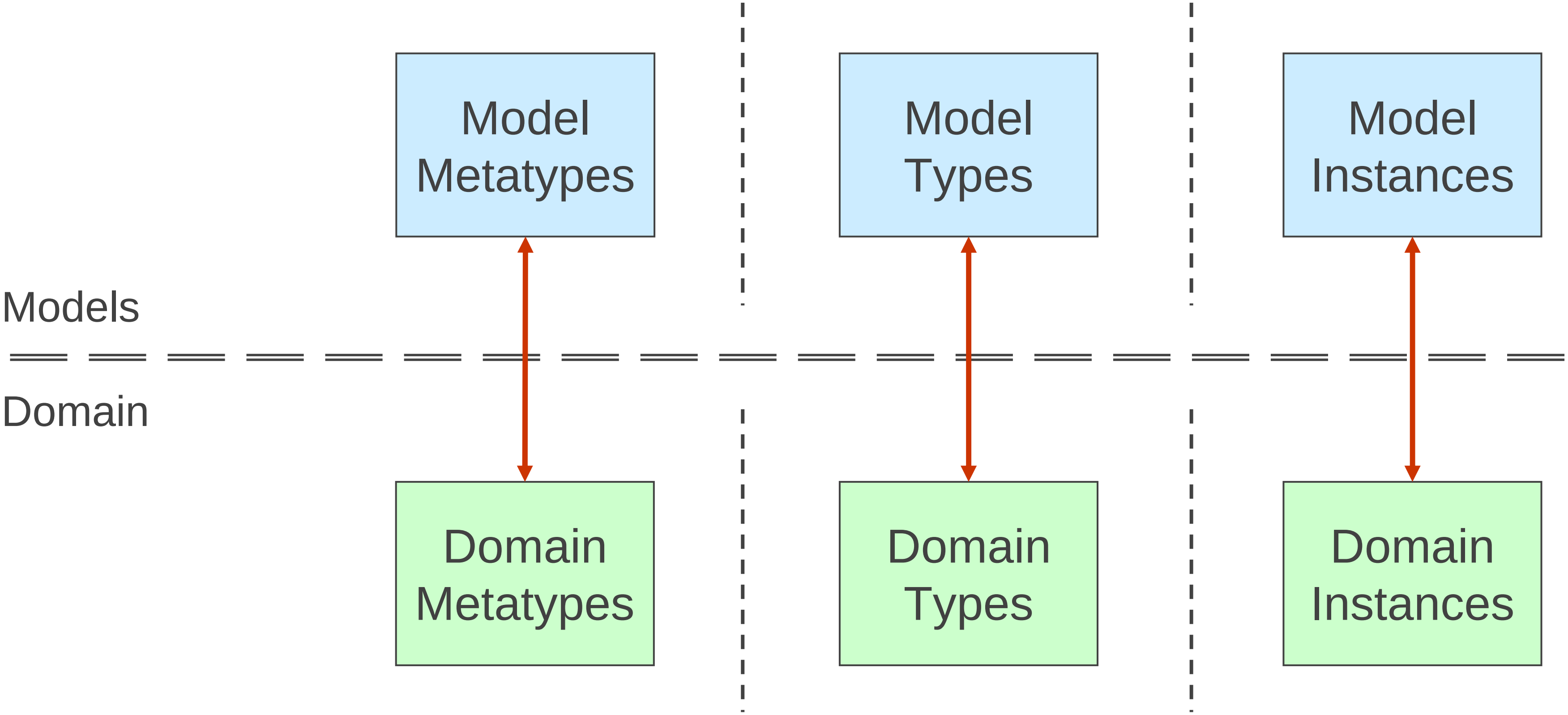
Example (Types & Instances)



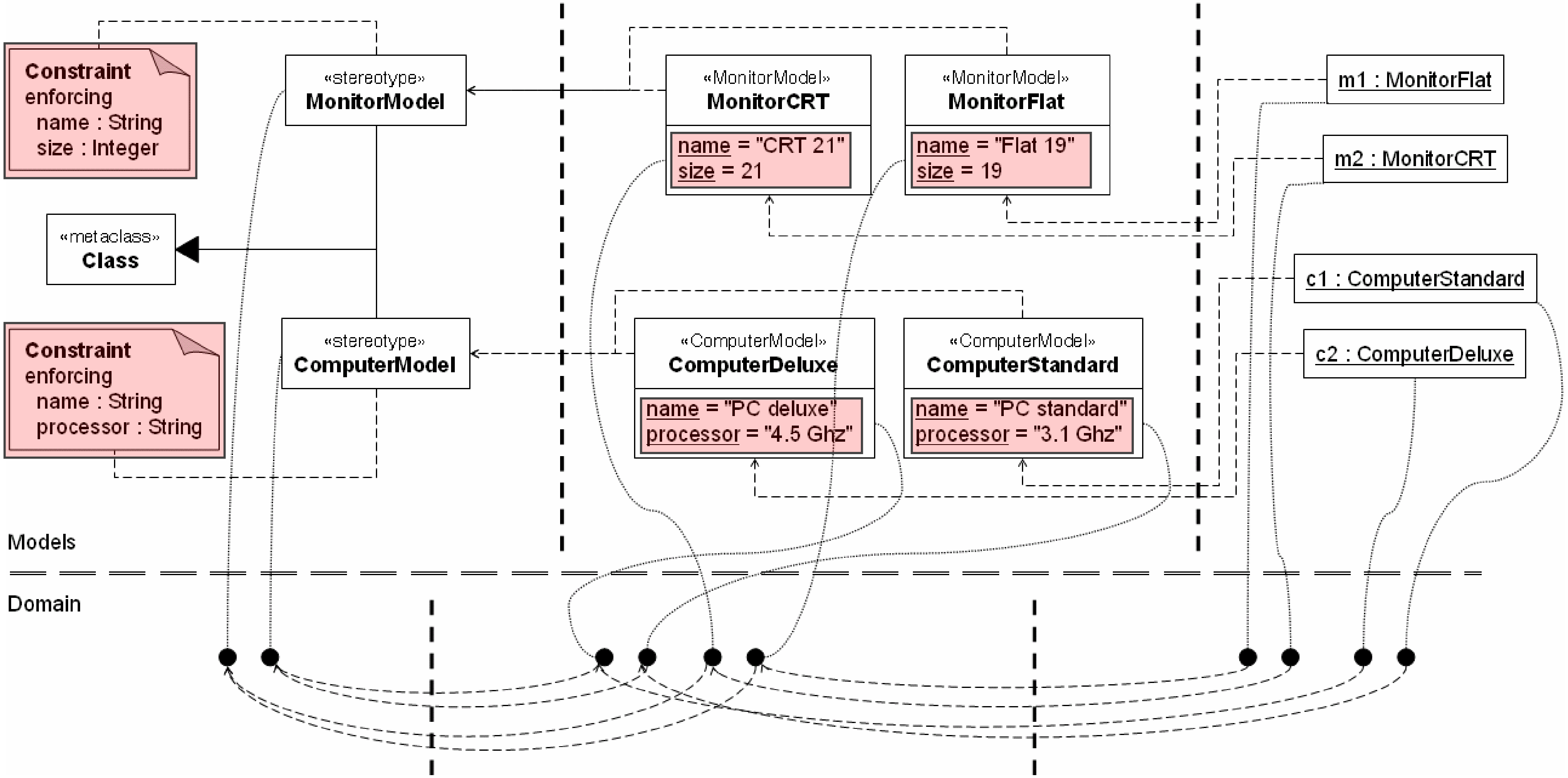
Level Compression



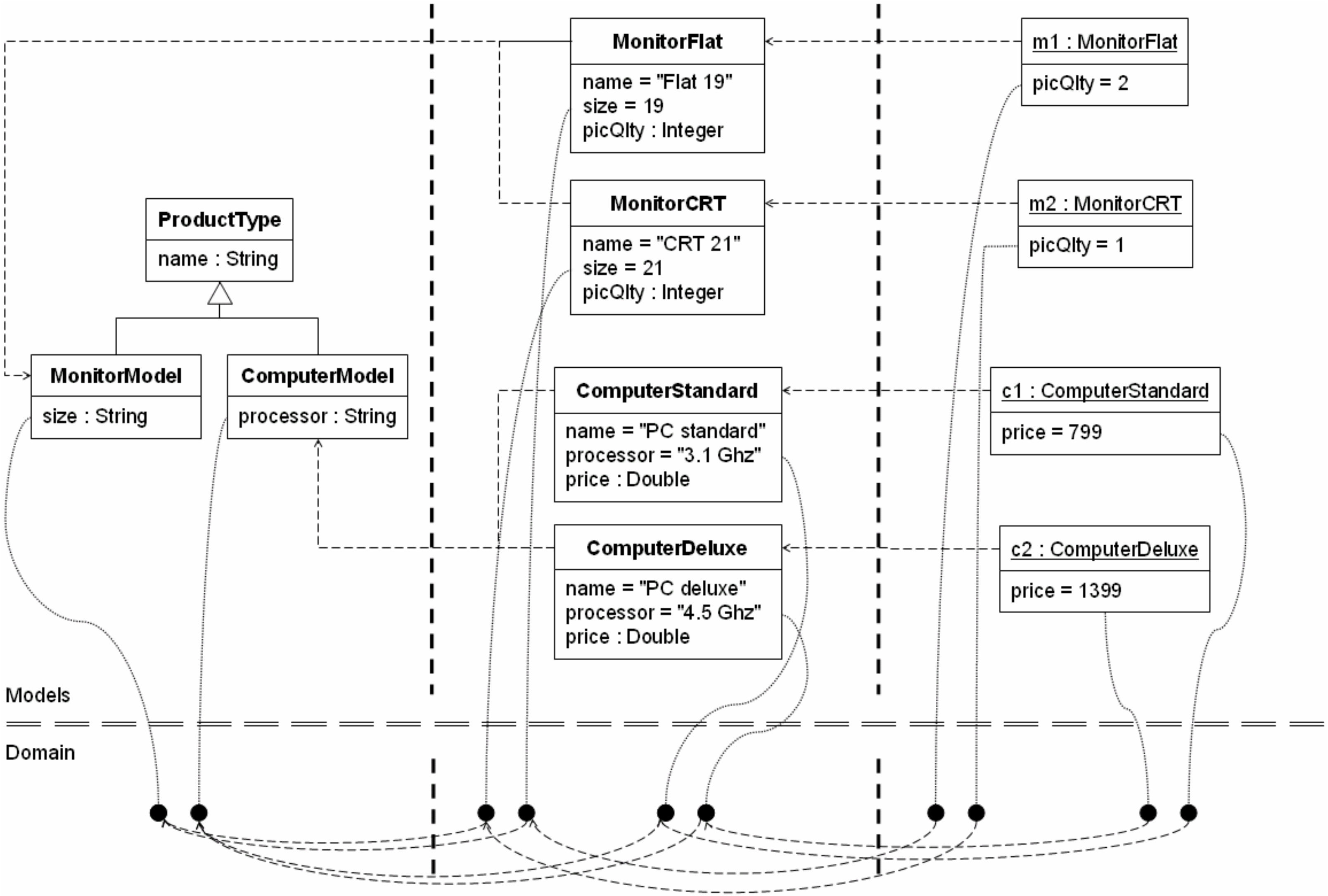
Direct Mapping



Levels Match (Stereotypes)



Levels Match (Metatypes)



ProductType
name : String

MonitorModel
size : String

ComputerModel
processor : String

MonitorFlat
name = "Flat 19"
size = 19
picQlty : Integer

MonitorCRT
name = "CRT 21"
size = 21
picQlty : Integer

ComputerStandard
name = "PC standard"
processor = "3.1 Ghz"
price : Double

ComputerDeluxe
name = "PC deluxe"
processor = "4.5 Ghz"
price : Double

m1 : MonitorFlat
picQlty = 2

m2 : MonitorCRT
picQlty = 1

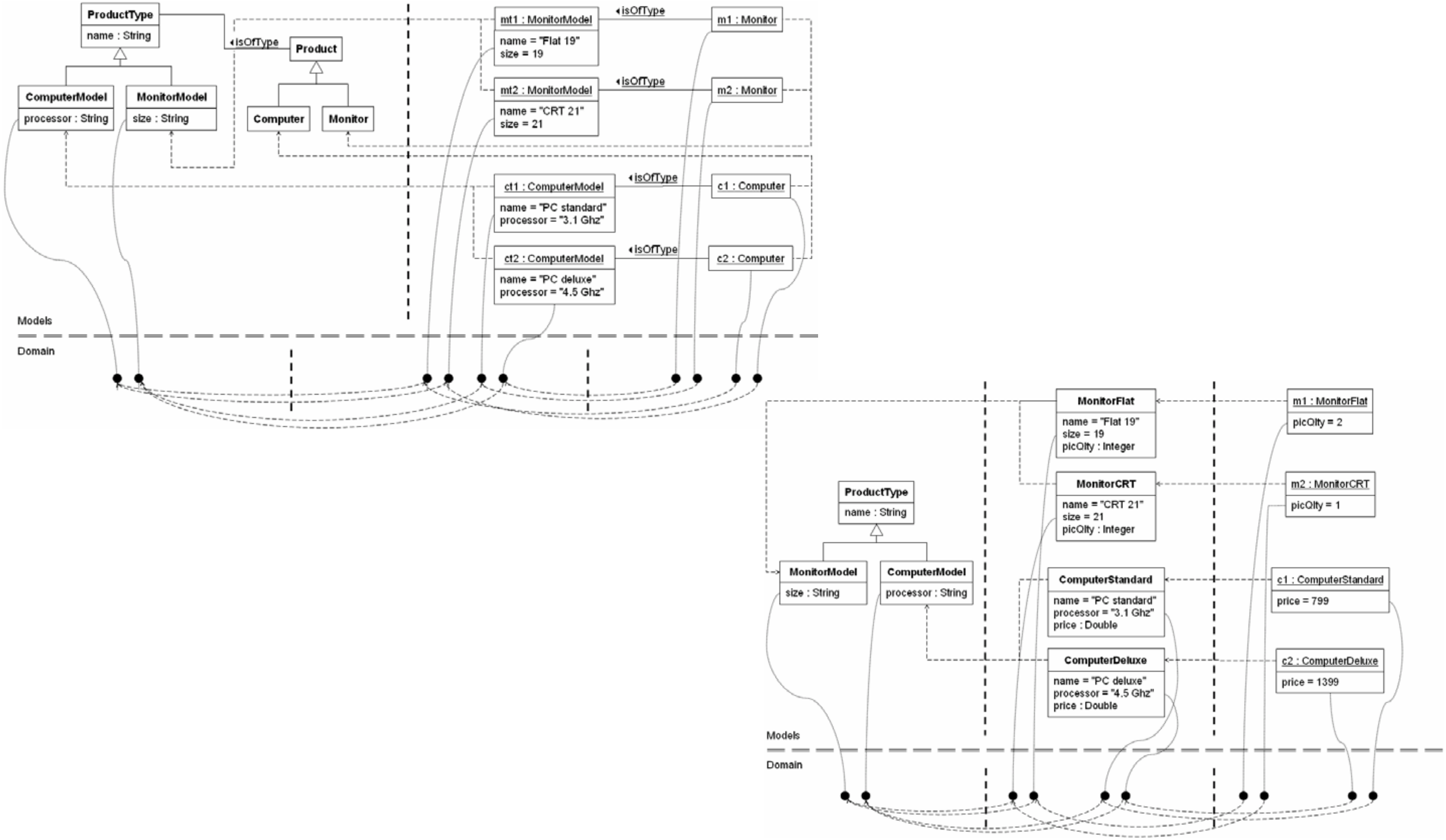
c1 : ComputerStandard
price = 799

c2 : ComputerDeluxe
price = 1399

Models

Domain

Comparison


ProductType
name : String
isOfType
Product
ComputerModel
processor : String
MonitorModel
size : String
Computer
Monitor
mt1 : MonitorModel
name = "Flat 19"
size = 19
isOfType
m1 : Monitor
mt2 : MonitorModel
name = "CRT 21"
size = 21
isOfType
m2 : Monitor
ct1 : ComputerModel
name = "PC standard"
processor = "3.1 Ghz"
isOfType
c1 : Computer
ct2 : ComputerModel
name = "PC deluxe"
processor = "4.5 Ghz"
isOfType
c2 : Computer
Models
Domain
MonitorFlat
name = "Flat 19"
size = 19
picQlty : Integer
m1 : MonitorFlat
picQlty = 2
MonitorCRT
name = "CRT 21"
size = 21
picQlty : Integer
m2 : MonitorCRT
picQlty = 1
ProductType
name : String
MonitorModel
size : String
ComputerModel
processor : String
ComputerStandard
name = "PC standard"
processor = "3.1 Ghz"
price : Double
c1 : ComputerStandard
price = 799
ComputerDeluxe
name = "PC deluxe"
processor = "4.5 Ghz"
price : Double
c2 : ComputerDeluxe
price = 1399
Models
Domain

Goals

When do I consider the workshop a success?

- is there a problem to my “abstract object” solution? (→ “Abstract Object” = “Role” only?)
- examples for accidental complexity
- interesting discussions
- direct influence on publication activities