



Defense Information Systems Agency
Joint Interoperability Test Command

System Entity Structure Framework and Methodology

Team NGIT

Bernard P. Zeigler, Ph. D

Arizona Center for Integrative Modeling and Simulation
(ACIMS)

Doohwan Kim, Ph. D.

RTSync.com



Motivation – Current Problems in Ontology Development for Data Engineering

- Need a process that goes from model to Schema in a streamlined manner
- Modeling framework should rest on rigorous formal foundation
- Framework should support a flexible specialization process to generate conforming submodels (alternative architectures, profiles)
- Model structure should be hierarchical and scalable
- Framework should support concepts for restructuring and harmonization of existing standards
- Framework should in both structure and behavior
- Approach should include pragmatic considerations

Chapter 4

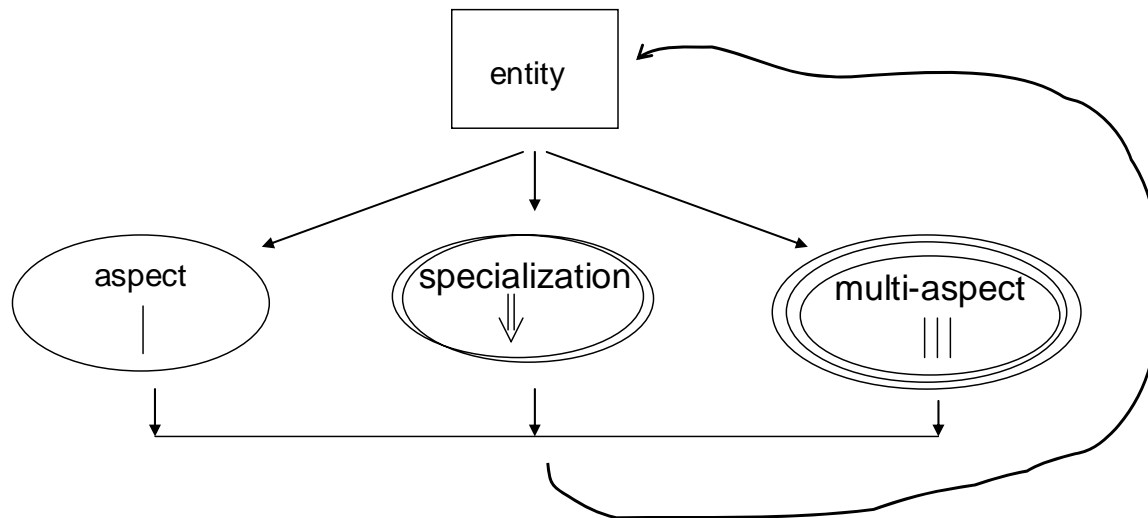
SES Items and Uses

See Chap. 4
P 68

| item | denotes | when to use |
|----------------|--|---|
| entity | a thing in the real or modeled world | Use to represent a thing that stands alone or is a part or variant of another thing. |
| aspect | the relationship between a thing and its components when decomposed from a certain perspective | Use when you want to represent an “and” connective among sub-things of a thing—where the “and” denotes the necessity that all of the sub-things must appear together to comprise the thing. |
| multi-aspect | a special kind of aspect in which all the components are homogeneous in nature | Use for the same objective as an aspect except that the components are all from the same class. |
| specialization | the relationship between a thing and its variants from a given family | Use when you want to represent an “or” connective among sub-things of a thing—where the “or” denotes the fact that a choice of one of the variants can replace the original. |
| variable | a property, quality, or attribute of an entity to which it is attached | Use to distinguish and characterize different instances, in space, time, or other dimensions. Can provide range specifications for data types and value constraints. |

Overview of SES items and relationships

See Chap. 4
P 54

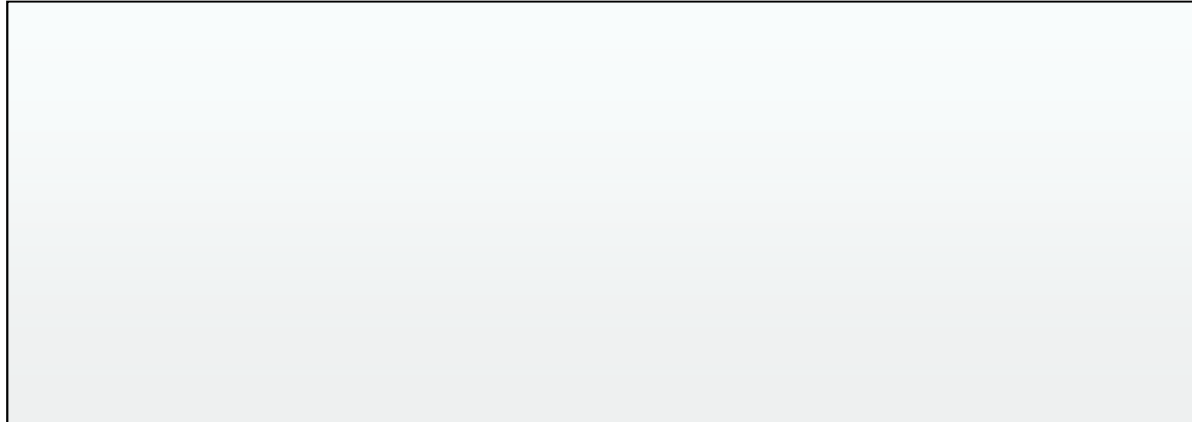


Example: Aspects for Physical Decomposition and Content Decomposition

See Chap. 4
P 55

Aspects may be related to Pragmatic Frames

In pruning, aspects are “downselected”



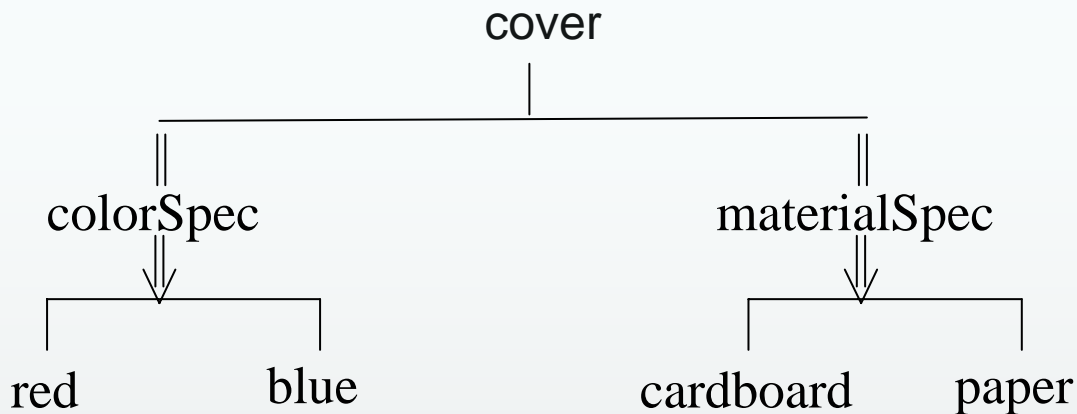
Natural Language Specification:

From the physical perspective, a book is made of a front cover and a back cover

From the content perspective, a book is made of a preface and a main body

Example: Specializations for Color and Material

See Chap. 4
P 59



Specializations may be related to Pragmatic Frames

In pruning, specializations form compounds

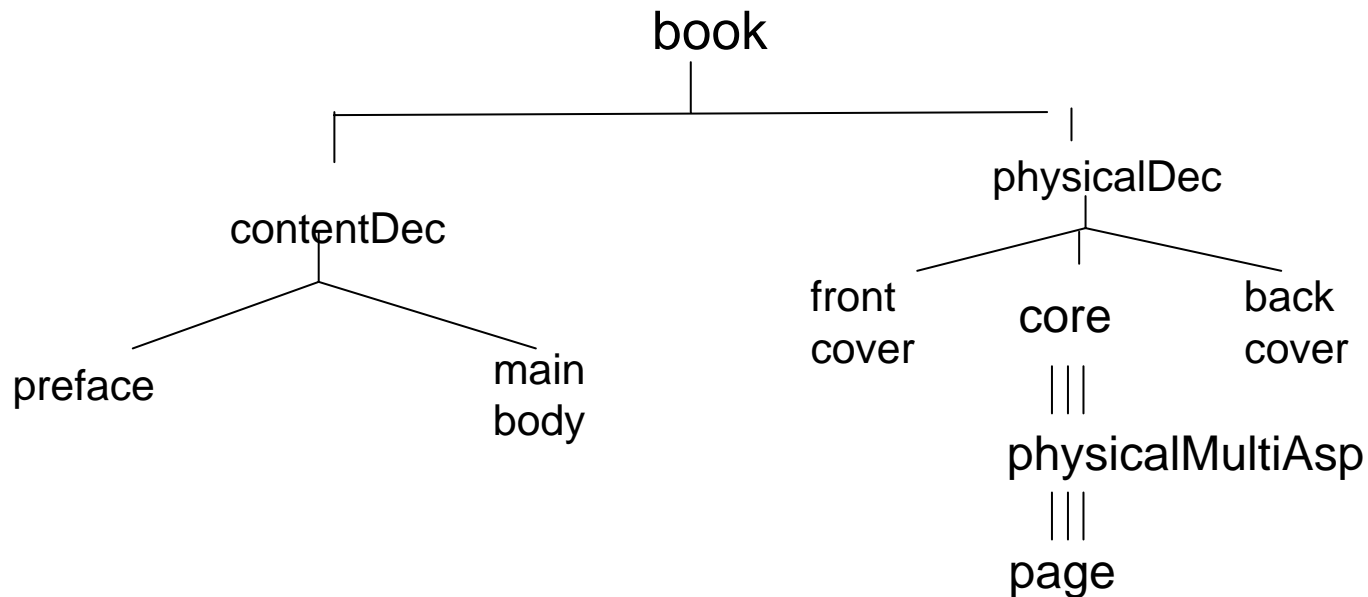
Rules can constrain possible selections

Natural Language Specification:

The cover can be red or blue in color

The cover can be cardboard or paper in material

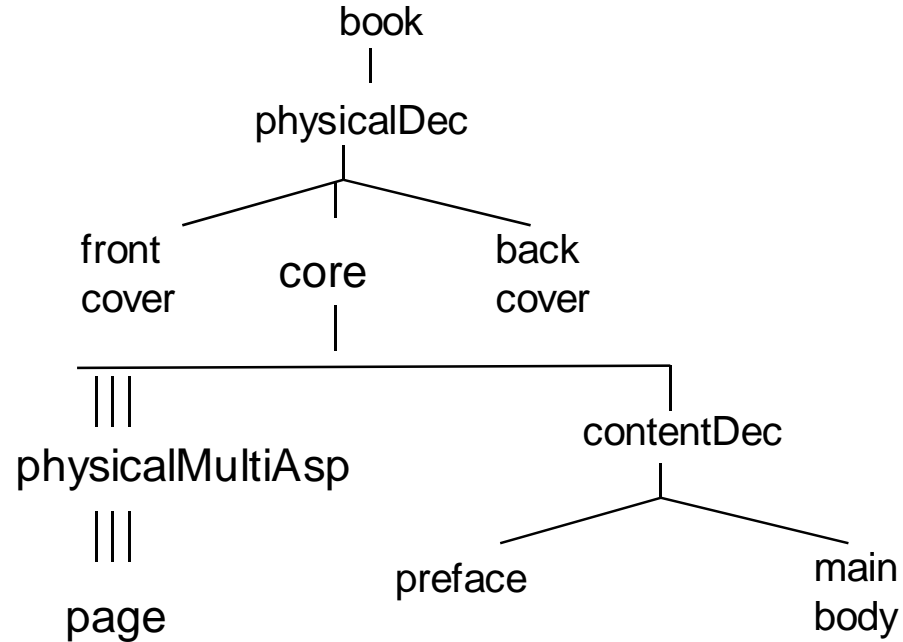
SES for Book with a Core



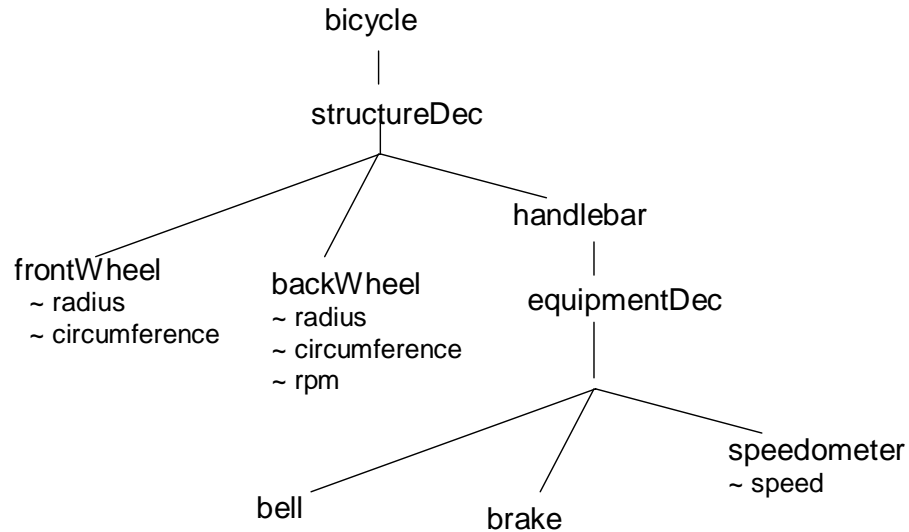
From the physical perspective, a book is made of a front cover, core, and a back cover

From the physical perspective, the core is made of more than one page

An entity can have both multiAspects and aspects



SES for Bicycle with Attached Variables



From the structure perspective, a bicycle is made of a front wheel, a back wheel, and a handlebar

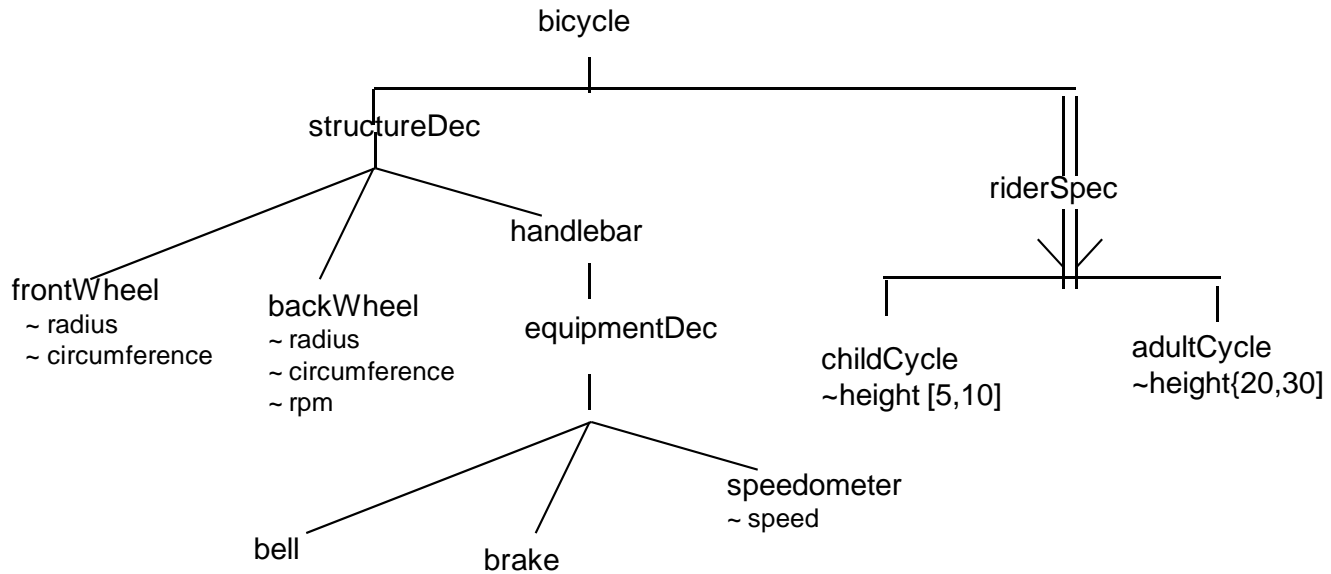
From the equipment perspective, the handlebar is made of a bell, a brake, and speedometer

The back wheel has a radius, a circumference, and an rpm

The speedometer has a speed

The range of backwheel's radius is double with values [10,30]

SES for Bicycle with Rider Specialization



A bicycle can be childBike or adultBike in rider

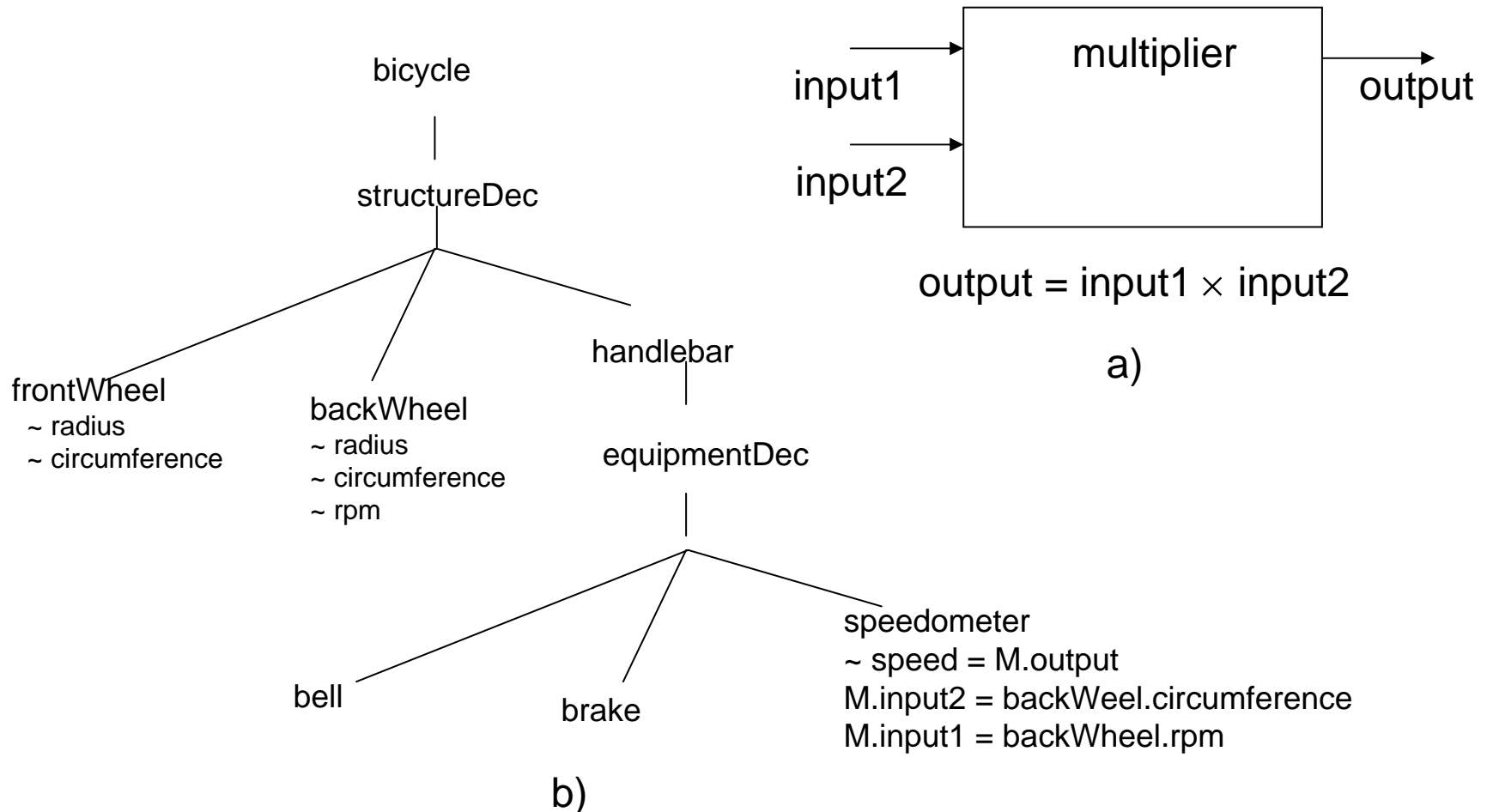
A childBike has a height

The range of a childBike's height is double with values [5,10]

An adultBike has a height

The range of an adultBike's height is double with values [20,30]

Representation of Computed Values



The speedometer's speed comes from the multiplier's output where
 The multiplier's input1 comes from the back wheel's circumference where
 The multiplier's input2 comes from the back wheel's rpm

Chapter 5

SES Axioms

uniformity: Any two nodes that have the same labels have identical attached variable types and isomorphic subtrees.

strict hierarchy: No label appears more than once down any path of the tree.

alternating mode: Each node has a mode that is either entity, aspect, or specialization; if the mode of a node is entity, then the modes of its successors are aspect or specialization; if the mode of a node is aspect or specialization, then the modes of its children are entity. The mode of the root is entity.

valid brothers: No two brothers have the same label.

attached variables: No two variables attached to the same item have the same name.

inheritance: the parent and any child of a specialization combine their individual variables, aspects and specializations when pruning is activated

Mathematical Representation

$SES = \langle$

$Entities,$

$Aspects,$

$Specializations,$

$rootEntity,$

$entityHasAspect,$

$entityHasSpecialization,$

$aspectHasEntity,$

$specializationHasEntity,$

$entityHasVariable,$

$variableHasRange$

\rangle

$Entities, Aspects, Specializations, Variables$

are finite mutually disjoint sets

$rootEntity \in Entities$

$entityHasAspect \subseteq Entities \times Aspects$

$entityHasSpecialization \subseteq Entities \times Specializations$

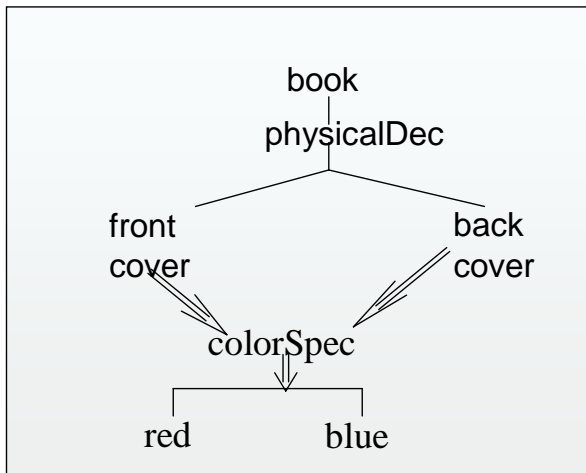
$aspectHasEntity \subseteq Aspects \times Entities$

$specializationHasEntity \subseteq Specializations \times Entities$

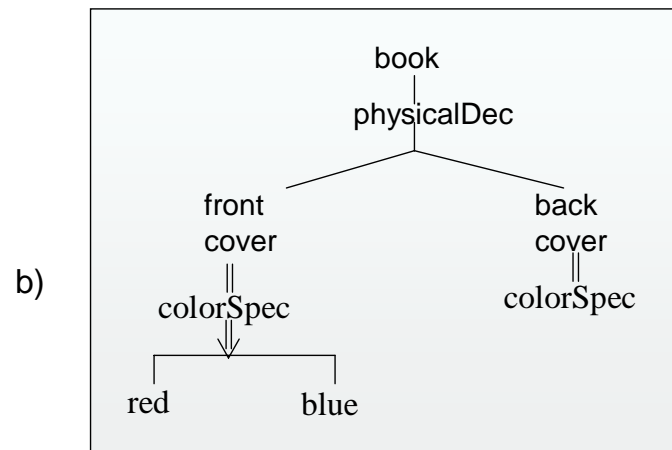
$entityHasVariable \subseteq Entities \times Variables$

$variableHasRange \subseteq Variables \times RangeSpec$

Uniformity Axiom in Graph and Tree Depictions

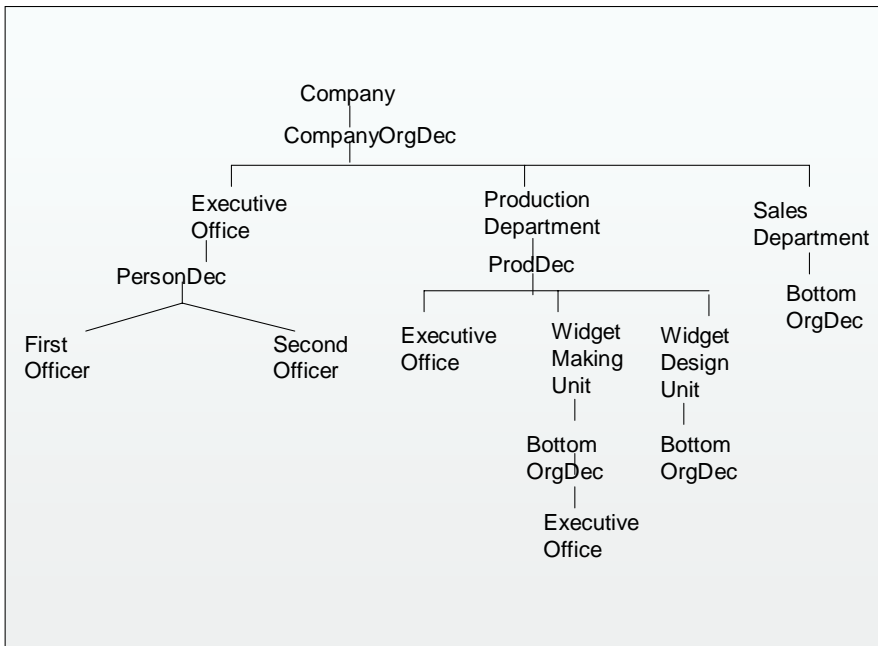


a)



b)

Illustrating the Uniformity Axiom



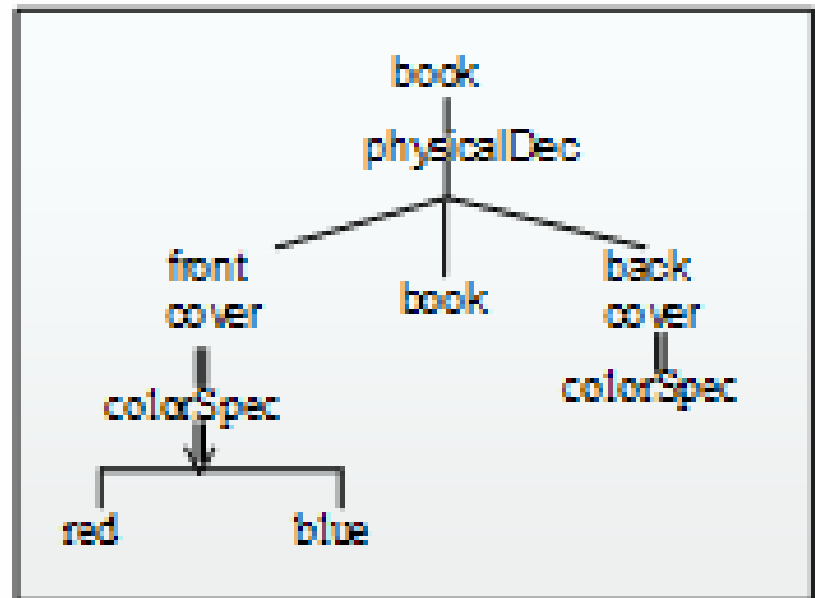
- From the organization perspective, a Company is made of an Executive Office, a Production Department, and a Sales Department
- From the personnel perspective, the Executive Office is made of a First Officer and a Second Officer
- From the production perspective, the Production Department is made of an Executive Office, a Widget Making Unit, and a Widget Design Unit
- From the bottom organization perspective, the Widget Making Unit is made of an Executive Office
- From the bottom organization perspective, the Widget Design Unit is like the Widget Making Unit
- From the bottom organization perspective, the Sales Department is like the Widget Making Unit

An Example Where Strict Hierarchy is Violated

Algorithm 5.2: Check for strict hierarchy in an SES specification:

```
//initialize
checkHierEntity("rootEntity")
//
checkHierEntity(path){
  entity = lastItemOnPath(path)
  newAspects = entityHasAspect(entity)
  For each aspect ∈ newAspects
  if isAnywhereOn(aspect, path), break "repeat aspect found";
  else checkHierAspect(path + "aspect")
}
checkHierAspect(path){
  aspect = lastItemOnPath(path)
  newEntities = aspectHasEntity(aspect)
  For each entity ∈ newEntities,
  if isAnywhereOn(entity, path), break "repeat entity found";
  else
  checkHierEntity(path + "entity")
}
//similarly for specializations
```

From the physical perspective, a book is made of a front cover, a back cover, and a book.



Unique Path Labeling

Strict hierarchy applied to the SES gives the following property:

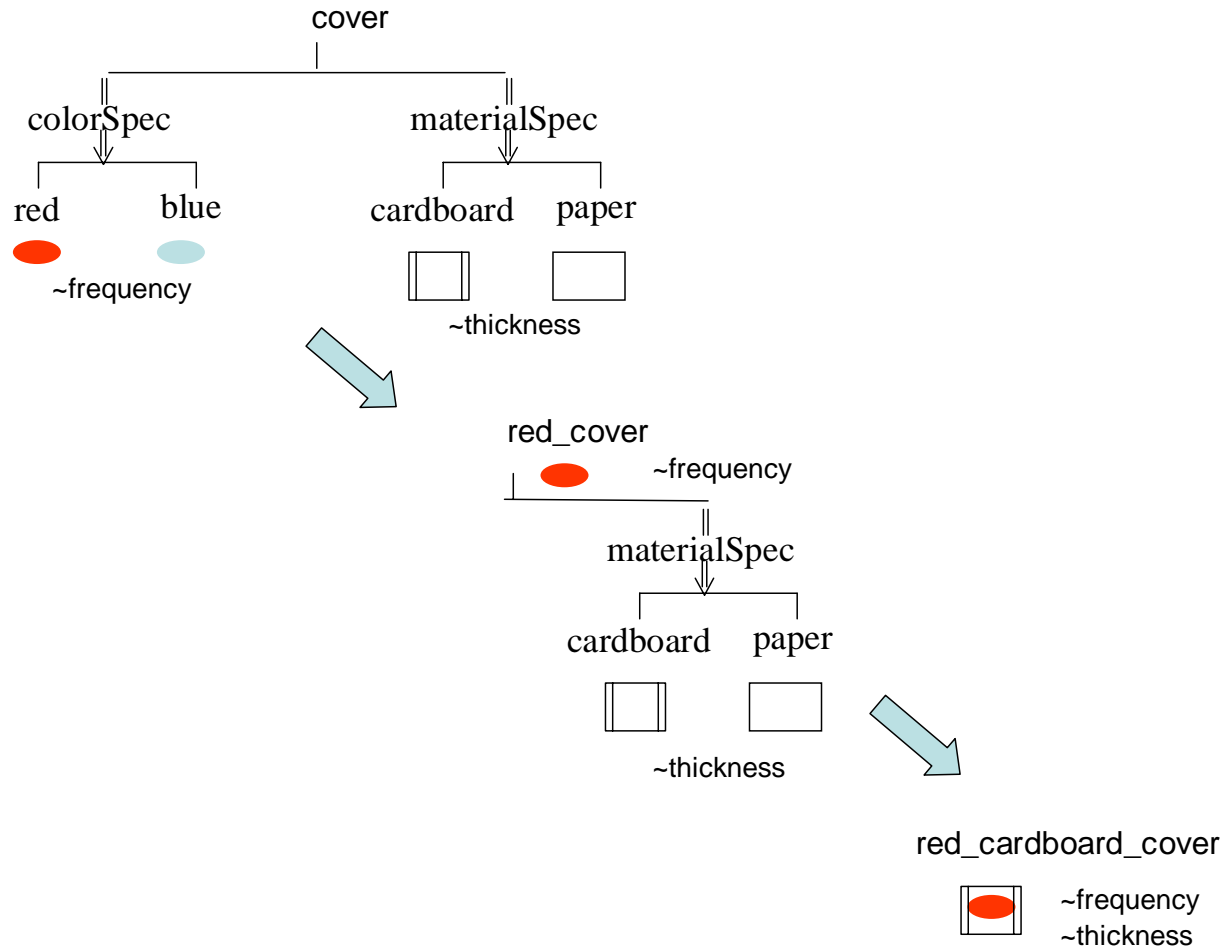
Unique path labeling property: there is a uniquely labeled path from the root to every node in the expanded graph.

“Company, CompanyOrgDec, ExecutiveOffice, PersonDec” is the path from root to the first occurrence of FirstOfficer.

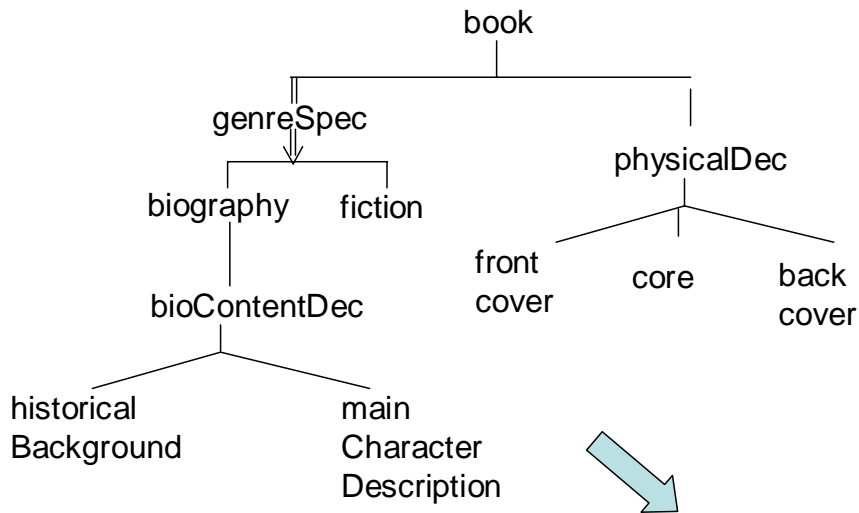
“Company, CompanyOrgDec, ProductionDepartment, ProdDec, ExecutiveOffice, PersonDec” is the path from root to the second occurrence of FirstOfficer

Something less than the full path from the root to a label may be sufficient to disambiguate it.

Illustrating Multiple Inheritance

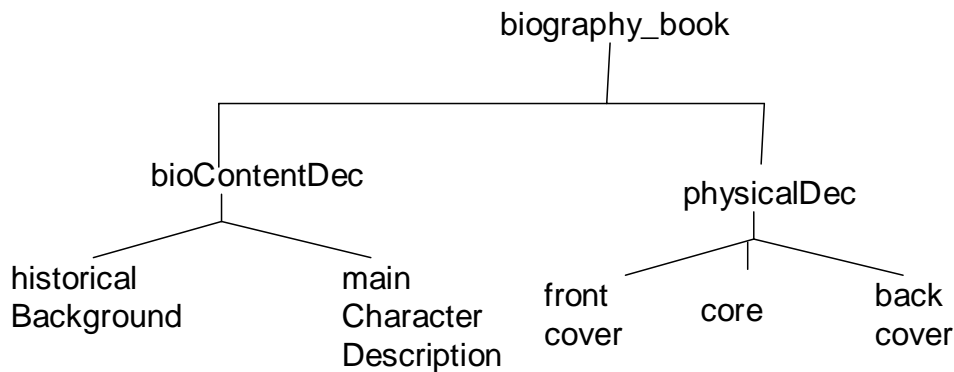


Inheritance of Aspects

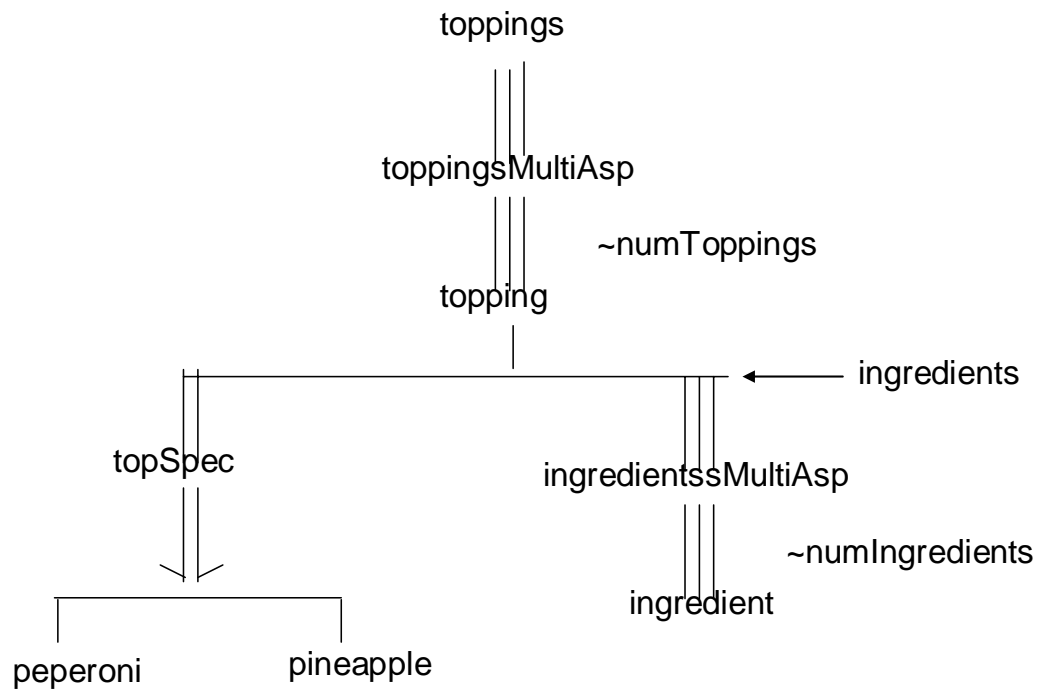


From the physical perspective, a book is made of a frontCover, a core, and a backCover

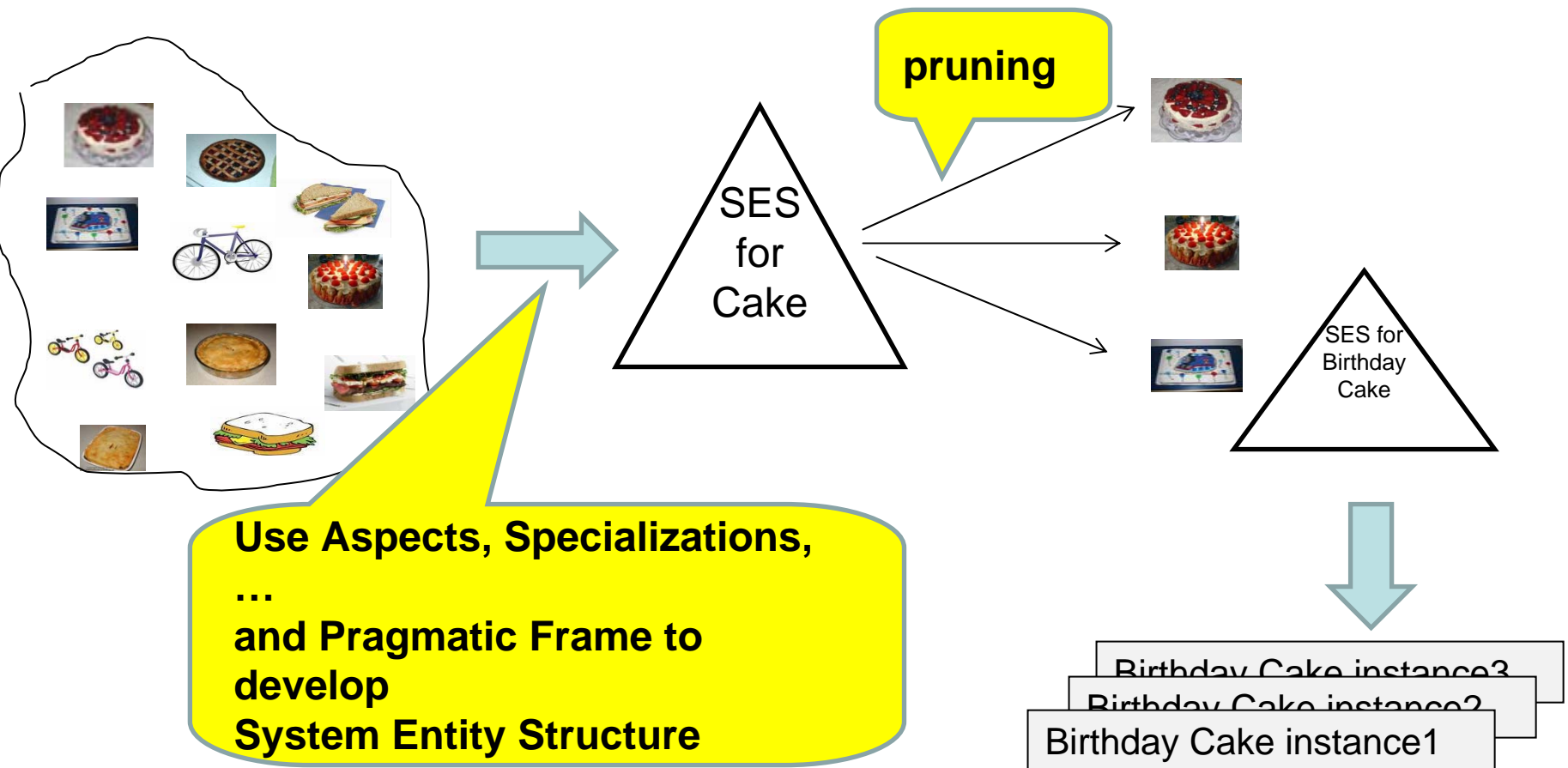
A book can be biography or fiction in genre



Inheritance of multiAspects



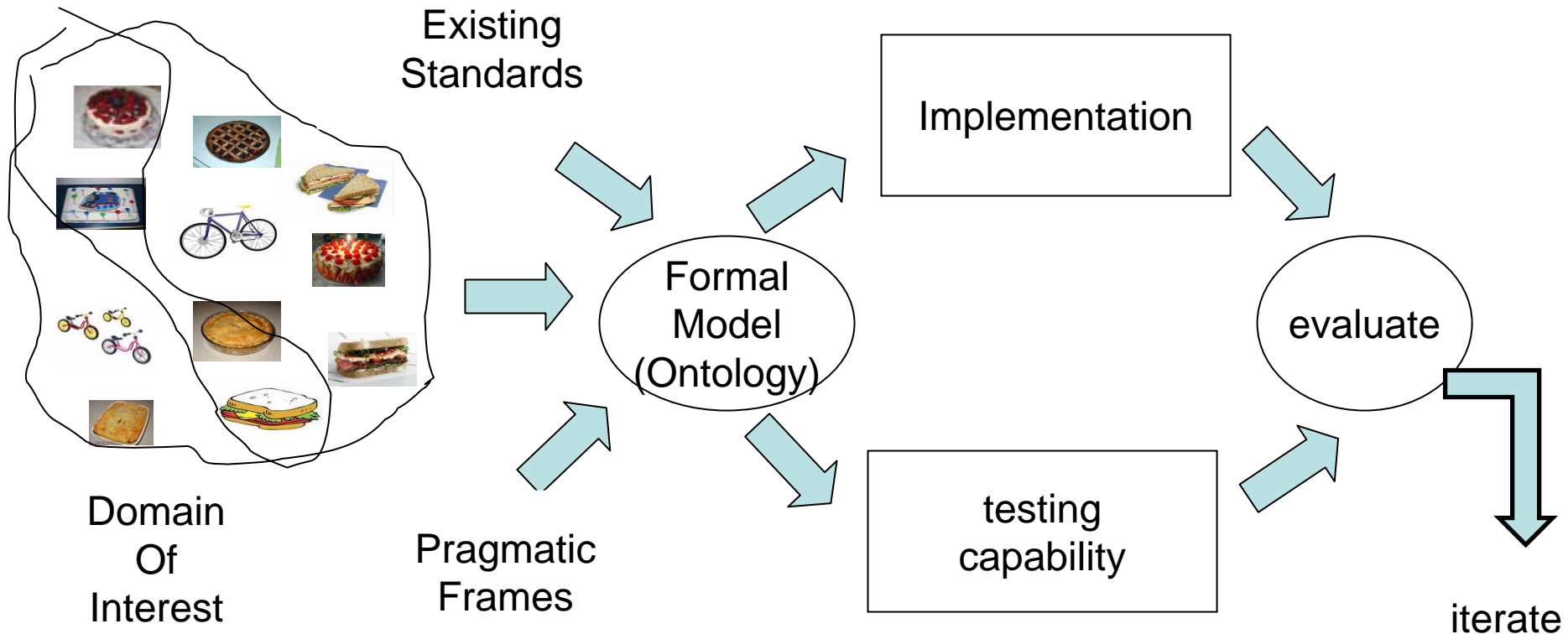
Pruning Creates Submodels and Instances



Problems

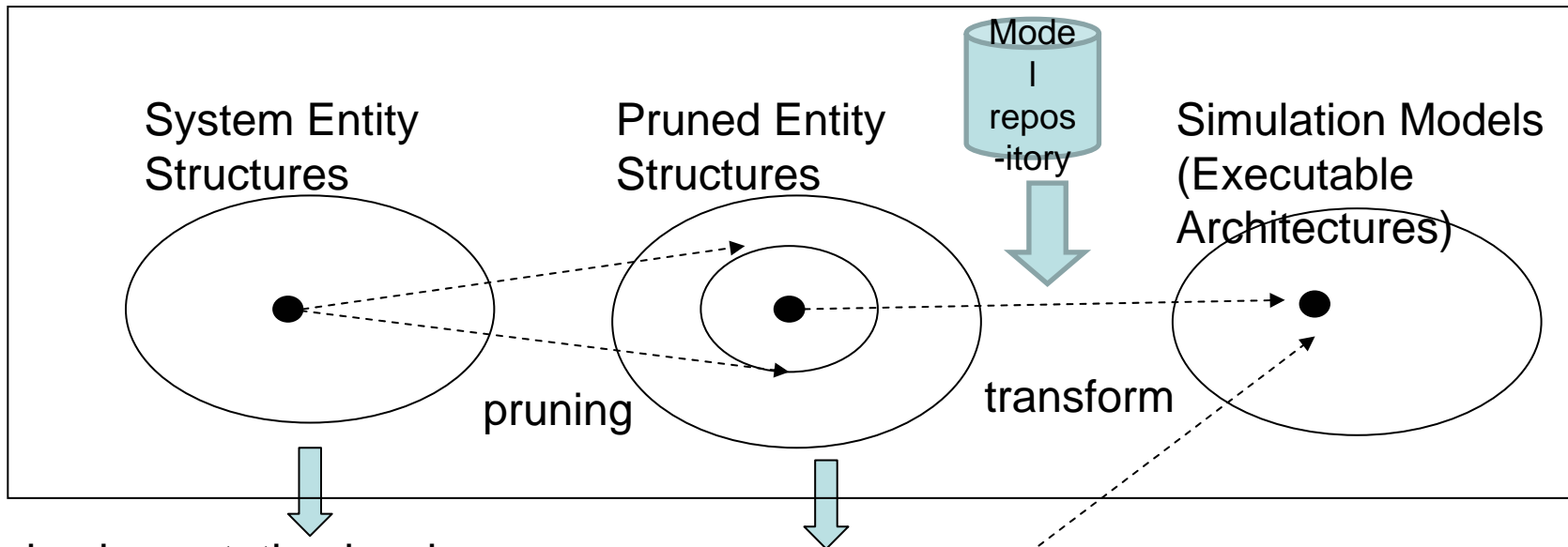
1. Use the SES to describe the class of all sandwiches that you like eating. Hint: From a physical perspective, consider what makes a sandwich different from other kinds of food preparations? Then, starting with some sandwiches that come to mind, ask yourself what ingredients can be replaced by others (e.g., peanut butter by margarine?) or co-exist together. Are there constraints on combinations of ingredients?
2. A song or other music composition has two types of decompositions, one based on the arrangements of notes within measures, and the other on melodic, harmonic and rhythmic themes. Develop an SES for the class of simple nursery rhymes. Can you restrict its prunings to just those that are likely to be acceptable to children?
3. Describe the players on the field of a baseball game. Rather than treating each player individually, use aspects, multiAspects and specializations to capture natural categories such as infielders. Develop similar representations for football and basketball games. Write an SES for “game” that has baseball, football and basketball explicitly within a specialization. Can you come up with a generic SES that can be pruned to describe any one of the games without using the explicit approach? In other words, are there commonalities that can be factored out of the individual SESs to provide a deeper representation from which each can be pruned?

Model-based Approach to Data Engineering Development and Testing

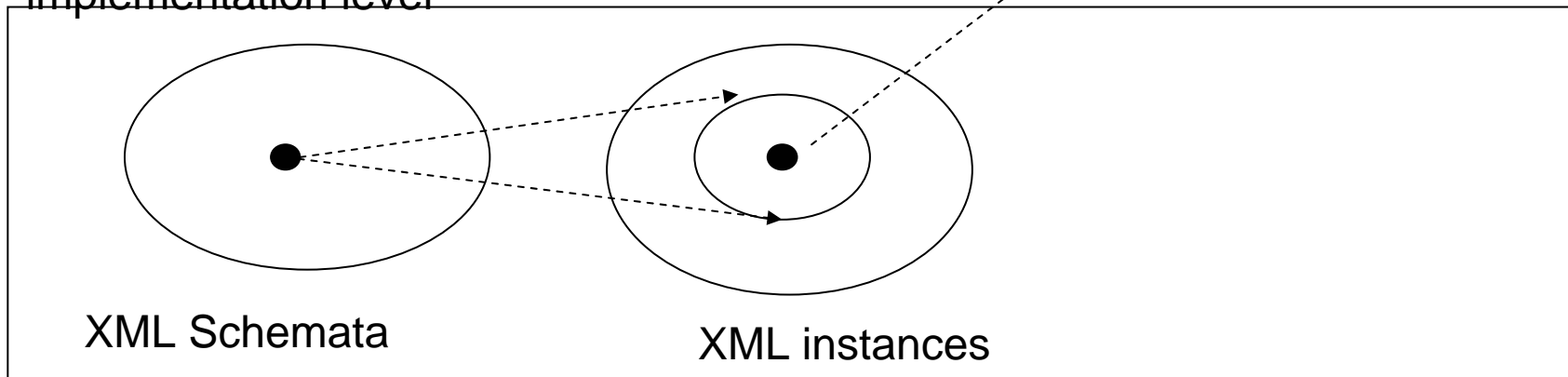


System Entity Structure Framework

ontology level



implementation level



SES Basic Methodology

At the ontology level, the modeler develops one or more SESs that are merged to create an ontology to satisfy the pragmatic frames of interest in a given application domain.

An SES can be specified in various ways e.g., via restricted natural language, or in XML.

It is then automatically encoded to an XML schema/document type definition (XSD or DTD) at the implementation level.

Integral to the SES concept is the operation of pruning to create pruned entity structures – with a model repository, they transform to simulation models

The XML instance documents specified by a schema are formally represented by the family of pruned entity structures at the ontology level.

Rule-based Pruning also supports generation of sub-models (profiles