Metamodeling

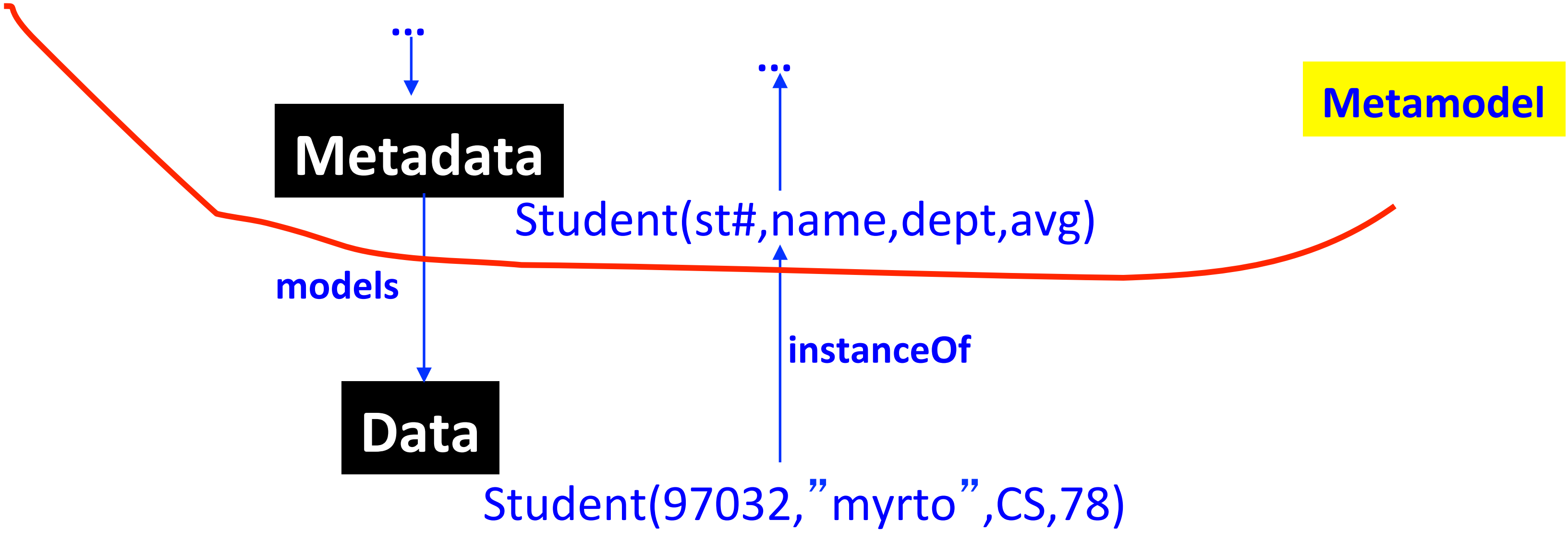
What is Metamodeling?
Dimensions on Metamodeling
The Information Resource Dictionary Standard (IRDS)
Repositories

What is Metamodeling?

- "Meta" means literally "after" in Greek.
- Meta-related themes have fascinated people throughout the centuries, e.g., [Hofstadter79] [Gaarder94]
- In Computer Science, the term is used heavily and with several different meanings:
  - ✓ In Databases, metadata means "data about data" and refer to data dictionaries, repositories, etc.;
  - ✓ In Programming Languages, meta-interpreters are interpreters of a (program) interpreter [Smith84];
  - ✓ In Conceptual Modeling, metamodel is a model of a data model, e.g., an E-R model of the relational model, or an ER model of the ER model.

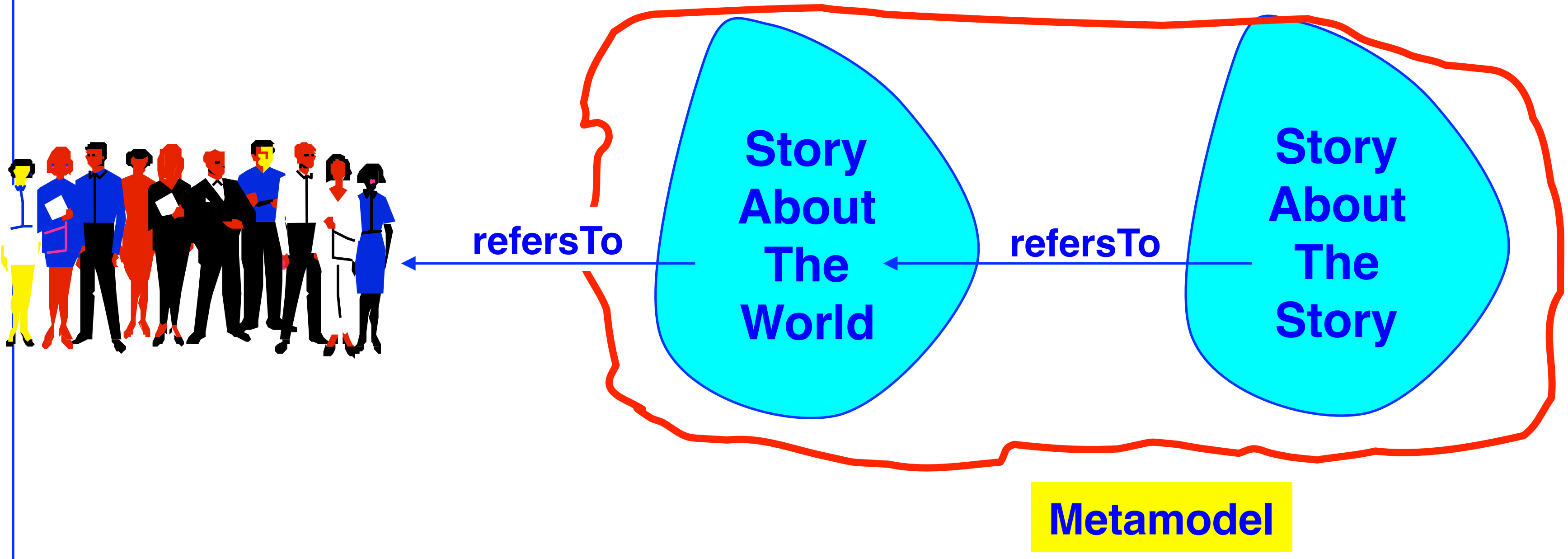
Metamodeling

- Data is modelled by metadata ("schemas", "classes",…) which are parts of the metamodel; these units are instances of meta^{2}data which are parts of a metametamodel, etc.
- We'd like to have metamodels which are self-descriptive to an arbitrary level of self-description.



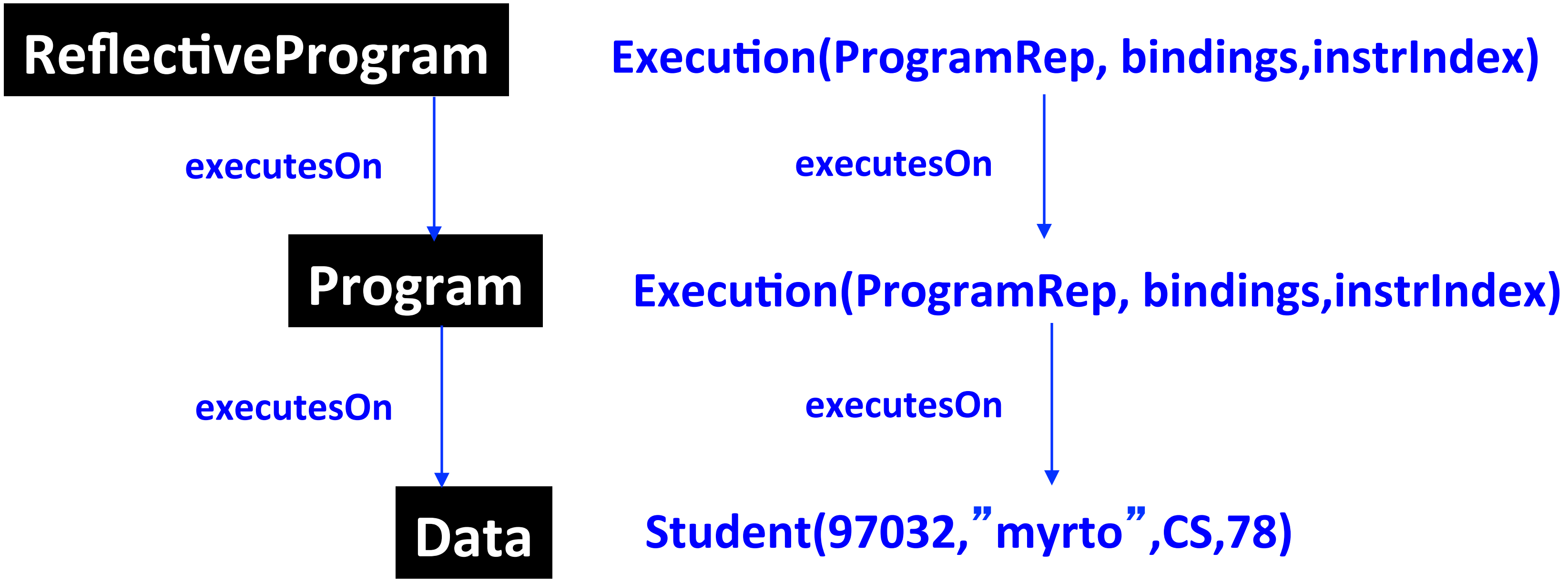
Another Dimension of Metamodeling

- The world is modelled by a story; the story is modelled by a metastory,…[Gaarter94]



…and Another...

- A program execution operates on data; a meta-execution operates on a program execution,….[Smith84]



ReflectiveProgram

executesOn

Program

executesOn

Data

Execution(ProgramRep, bindings,instrIndex)

executesOn

Execution(ProgramRep, bindings,instrIndex)

executesOn

Student(97032,”myrto”,CS,78)

Metamodeling in Art

Maurits Cornelis Escher

What's Interesting about Metamodeling?

- Ability to talk about any part of another model.
- Self-description, and all the complications that entails …
- Integration of several models into one metamodel description, leading to inconsistencies.

Requirements on Metamodeling Notations

- Should be capable of describing other conceptual models, e.g., the ER model, or SADT.
- Support facilities for defining primitive concepts, such as entity, activity, goal within the metamodel.
- Offer support for modeling multiple -- possibly contradictory -- perspectives, e.g., Maria at different times, from different viewpoints;
- Support variable granularity descriptions, as with geographic information;
- Support a variety of referential relationships, such as defines, denotes, mentions, includes, etc.

… Not a new Idea …

→The Backus-Naur Form (BNF) is a language for defining the syntax of other languages (through a grammar).

→For example

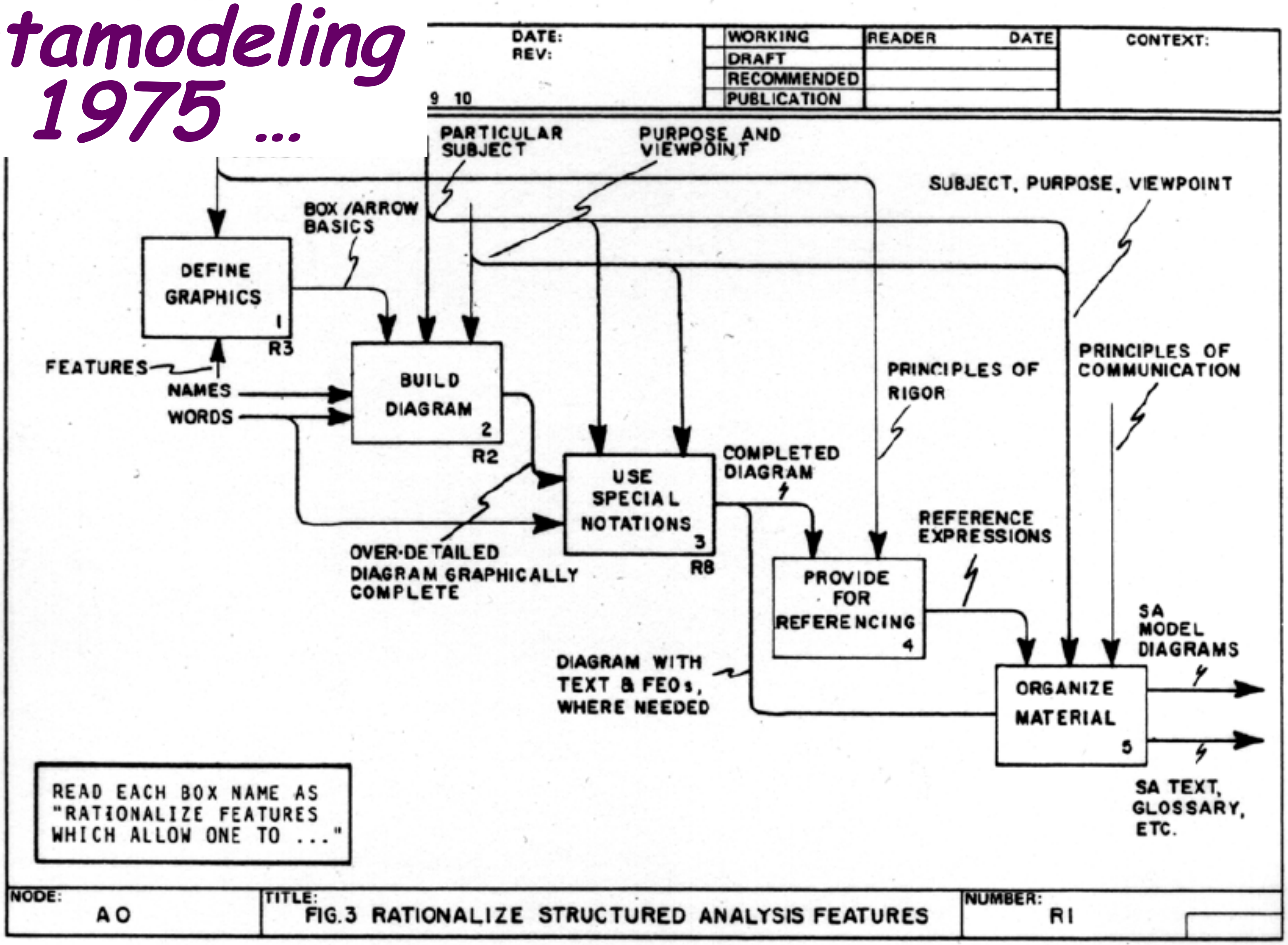
✓A simple grammar: NP ::= Noun | Adj NP
N ::= person | tree
Adj ::= tall | old | young

✓A grammar for BNF:

```
BNF ::= BNF-Rule | BNF-Rule BNF
BNF-Rule ::= LHS '::=' RHS
LHS ::= Non-Terminal
RHS ::= Symbol | Symbol RHS | RHS '|' RHS
Symbol ::= Terminal | Non-Terminal
```

…Metamodeling in 1975 …

SADT® DIAGRAM FORM ST098 9/75
Form © 1975 SofTech, Inc., 460 Totten Pond Road, Waltham, Mass. 02154, USA



DATE:
REV:

9 10

□ WORKING
□ DRAFT
□ RECOMMENDED
□ PUBLICATION

READER DATE

CONTEXT:

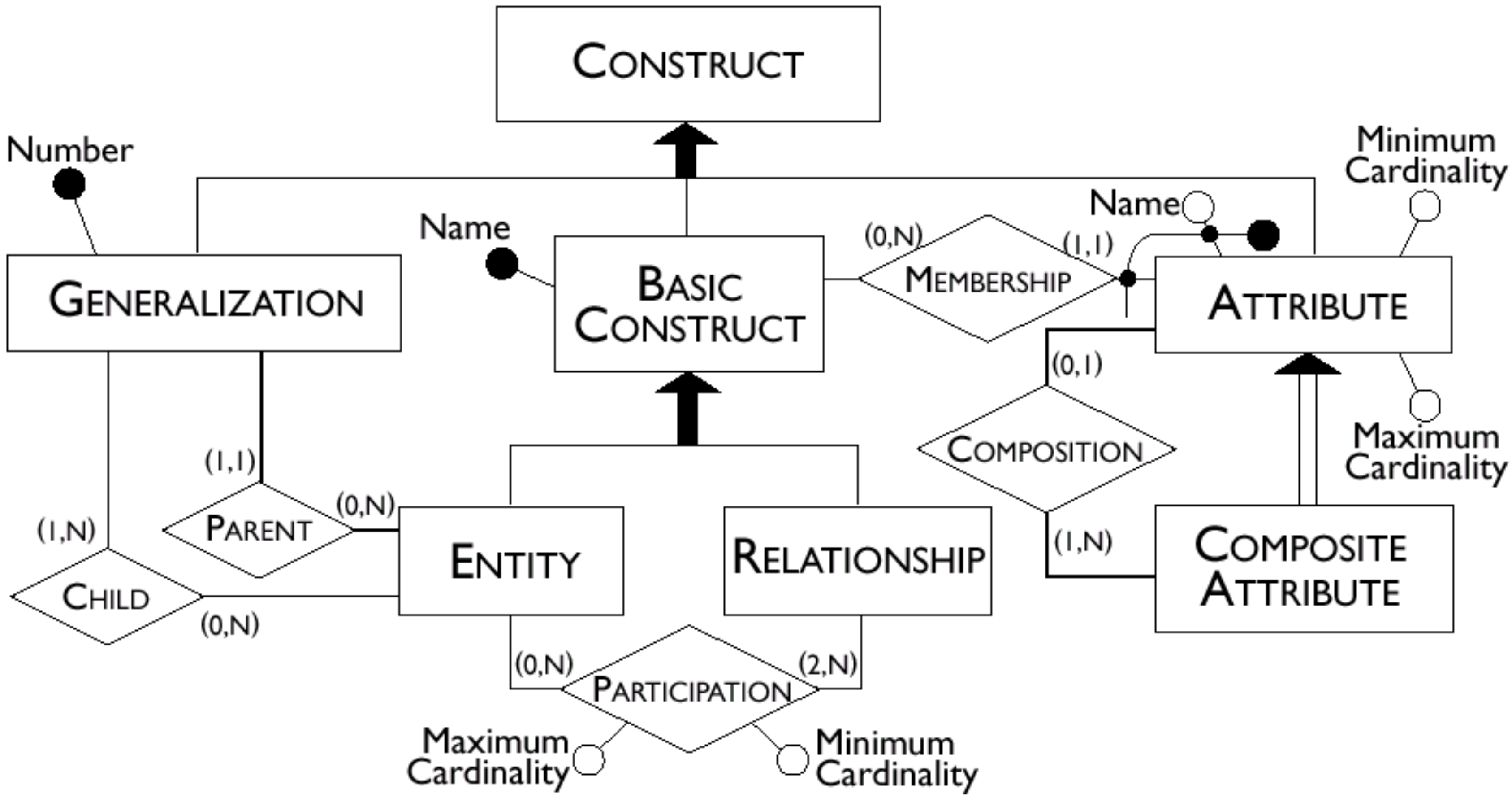
PARTICULAR SUBJECT
PURPOSE AND VIEWPOINT
SUBJECT, PURPOSE, VIEWPOINT
BOX /ARROW BASICS
DEFINE GRAPHICS 1
R3
FEATURES
NAMES
WORDS
BUILD DIAGRAM 2
R2
USE SPECIAL NOTATIONS 3
R8
COMPLETED DIAGRAM
OVER-DETAILED DIAGRAM GRAPHICALLY COMPLETE
PRINCIPLES OF RIGOR
PRINCIPLES OF COMMUNICATION
PROVIDE FOR REFERENCING 4
REFERENCE EXPRESSIONS
DIAGRAM WITH TEXT & FEOs, WHERE NEEDED
ORGANIZE MATERIAL 5
SA MODEL DIAGRAMS
SA TEXT, GLOSSARY, ETC.

READ EACH BOX NAME AS "RATIONALIZE FEATURES WHICH ALLOW ONE TO . . ."

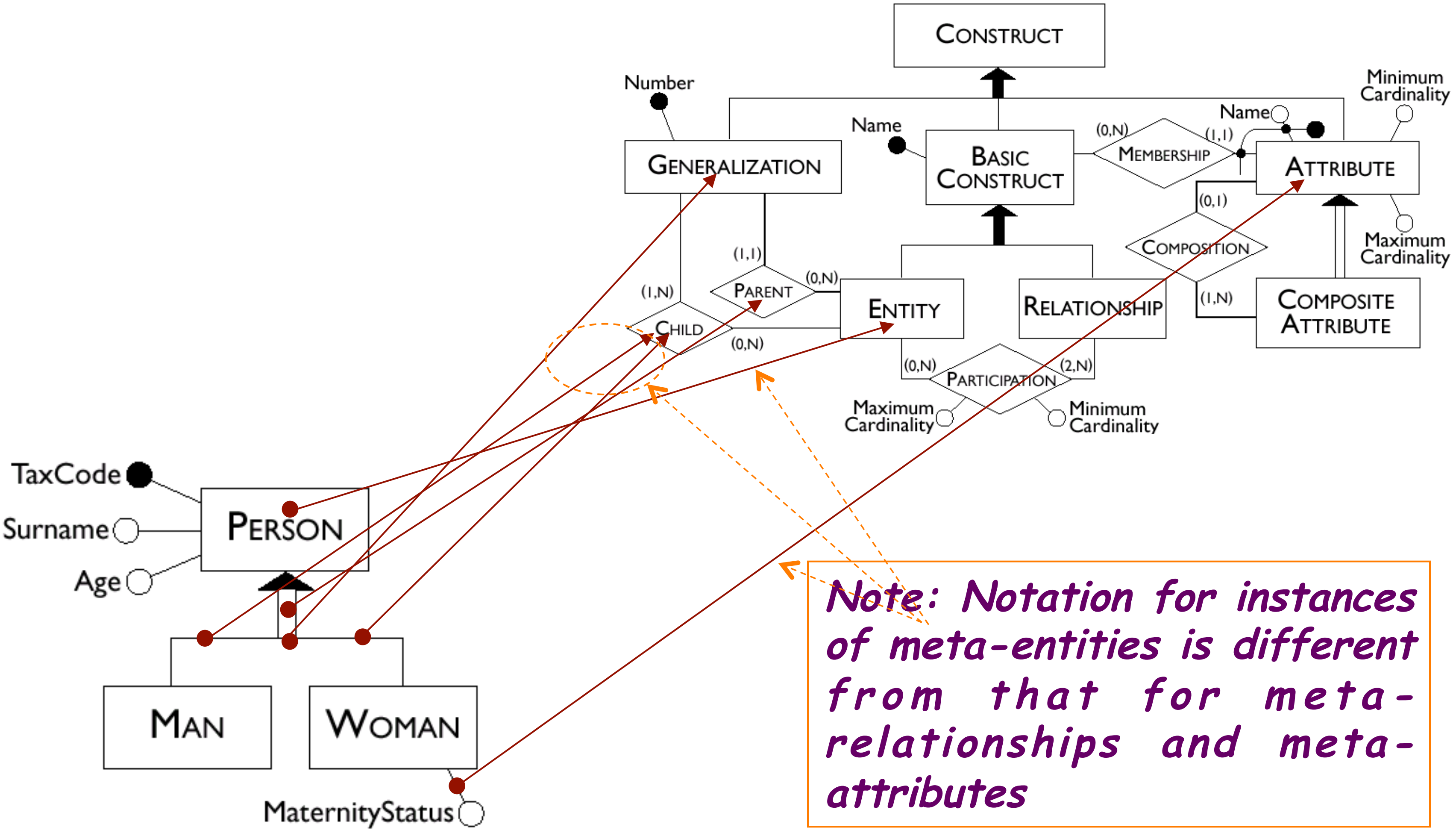
NODE: A0
TITLE: FIG.3 RATIONALIZE STRUCTURED ANALYSIS FEATURES
NUMBER: R1

Fig. 3. Rationalize SA features.

The EER Metamodel as an EER



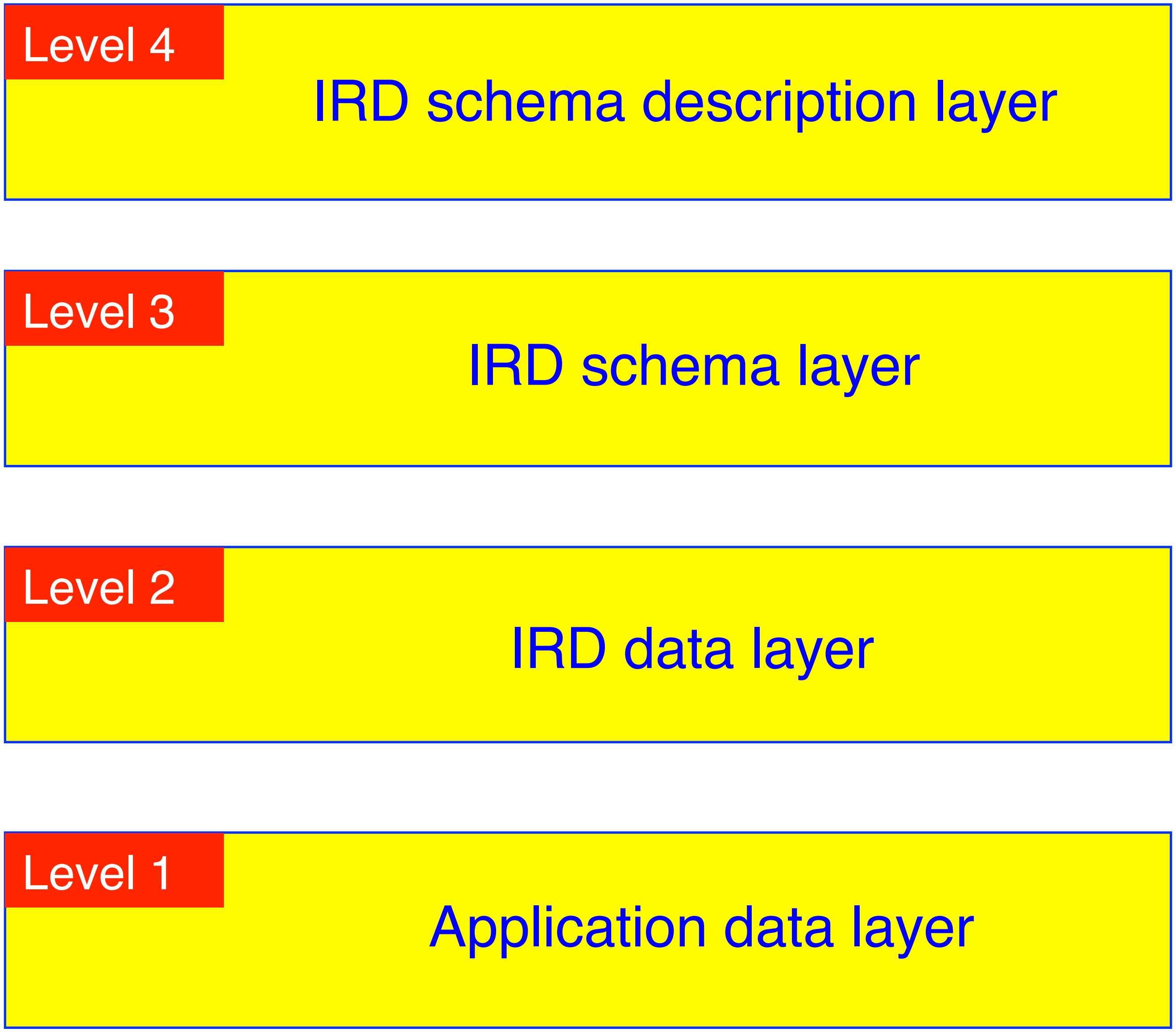
Instantiating the EER Metamodel



Note: Notation for instances of meta-entities is different from that for meta-relationships and meta-attributes

IRDS - Information Resource Dictionary Standard

- Data dictionary standard, since 1988 (ANSI X3.138)
- Technology-independent standard, akin to ER model.
- Proposes 4 different levels of data:
  - ✓ Bottom level -- application data, e.g., software code;
  - ✓ Level 2 -- data dictionary for application data, e.g., procedures, variables, data types, etc.
  - ✓ Level 3 -- schema for the data dictionary, e.g., what is a procedure (in the programming language the code is written in), what is a variable,…
  - ✓ Level 4 -- different types of IRDS schemas, e.g., programming language schemas vs requirements modeling ones.



Level 4

IRD schema description layer

Level 3

IRD schema layer

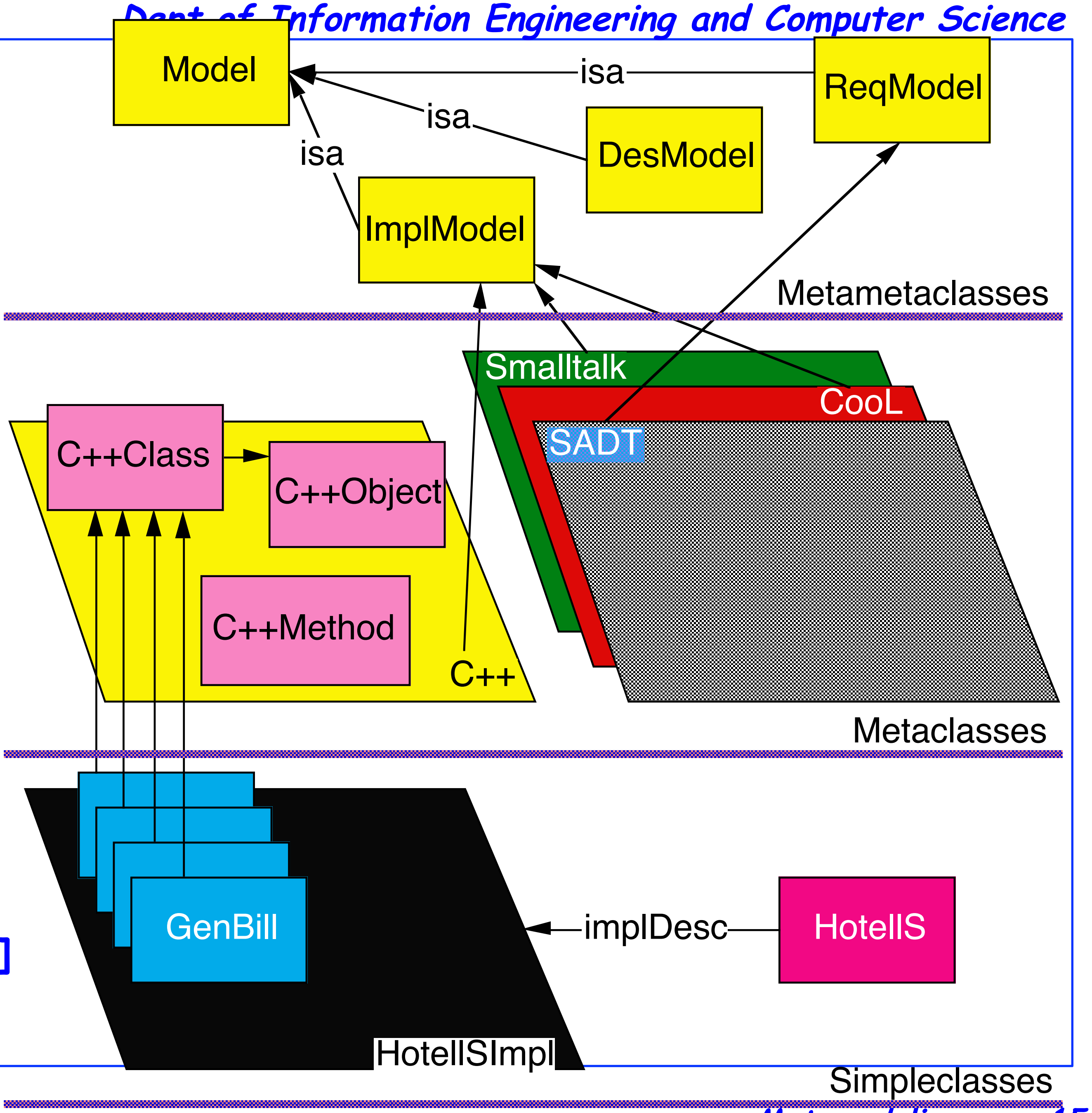
Level 2

IRD data layer

Level 1

Application data layer

Telos version of levels 2-4



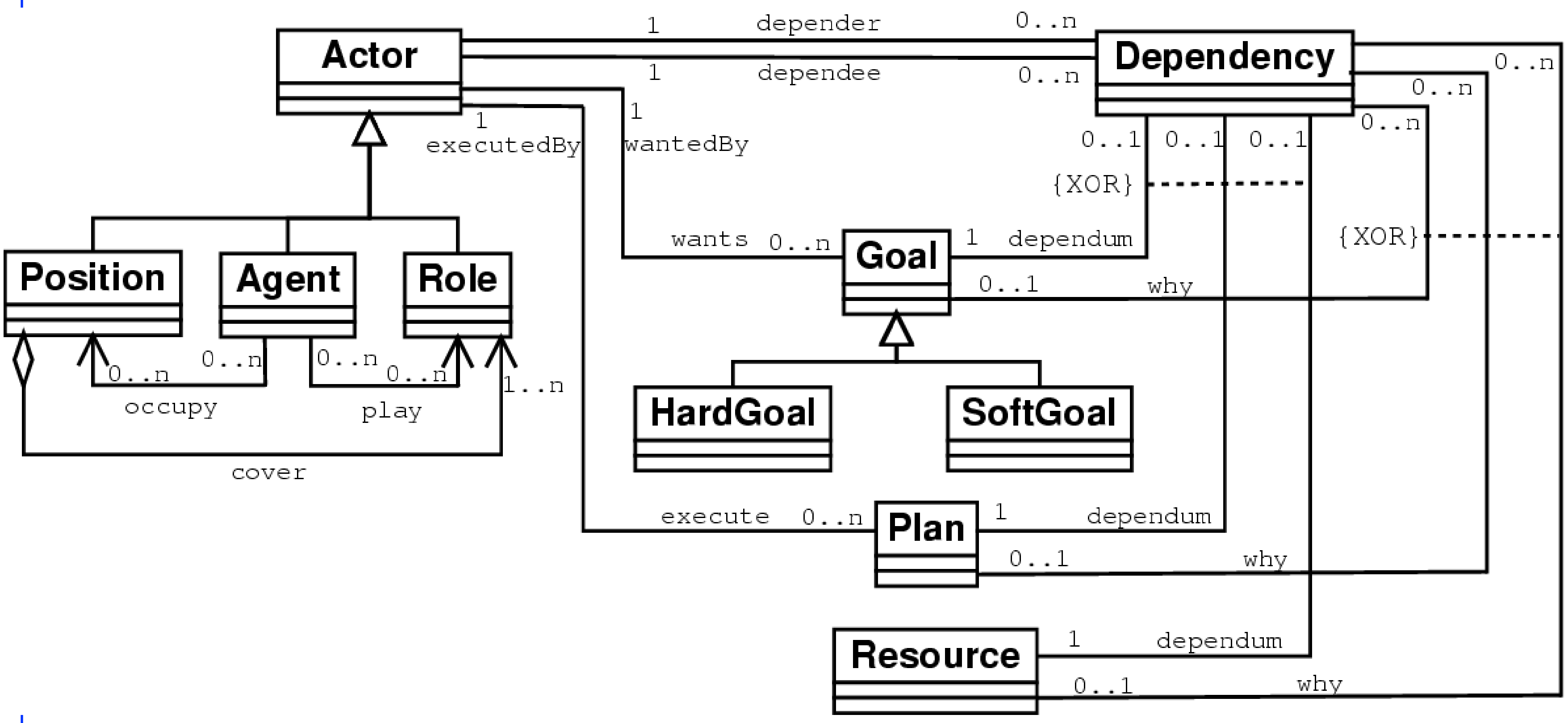
[Constantopoulos94]

Metadata in SQL

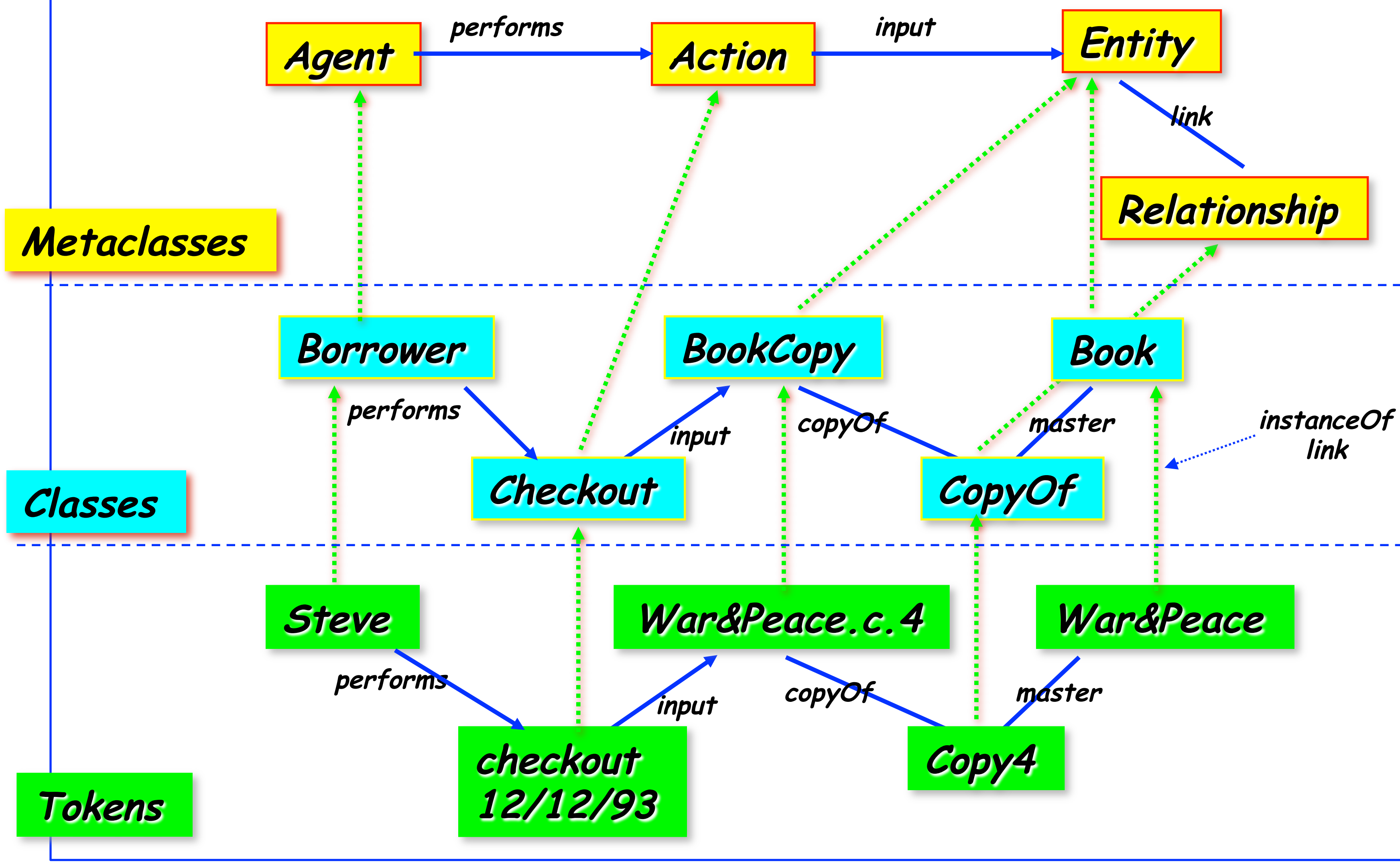
- A *relational catalogue* contains the data dictionary, i.e., a description of the relational schema $\mathcal{D}$ of the database.
- It is based on a relational schema $\mathcal{MD}$ whose relations describe the relations, columns, domains in $\mathcal{D}$ but also $\mathcal{MD}$ (reflectivity).
- The SQL-2 standard describes a Definition_Schema (composed of tables) and an Information_Schema (composed of views).

Rel	Attr	Dom	Default
Employee	name	String	null

i/Tropos*


Actor
1
depender
0..n
Dependency
1
dependee
0..n
1
executedBy
1
wantedBy
0..1
0..1
0..1
{XOR}
0..n
0..n
0..n
{XOR}
Position
Agent
Role
wants
0..n
Goal
1
dependum
0..1
why
0..n
0..n
0..n
0..n
1..n
occupy
play
cover
HardGoal
SoftGoal
execute
0..n
Plan
1
dependum
0..1
why
Resource
1
dependum
0..1
why

KAOS


Agent
performs
Action
input
Entity
link
Relationship
Metaclasses
Borrower
BookCopy
Book
performs
input
copyOf
master
instanceOf
link
Classes
Checkout
CopyOf
Steve
War&Peace.c.4
War&Peace
performs
input
copyOf
master
Tokens
checkout
12/12/93
Copy4

OMG's Meta Object Facility - MOF

- Unlike programming languages, a lot of modeling languages are not textual – so we use a different meta-language instead of BNF, called the MOF
- MOF is an OMG standard for modeling languages
  - ✓ It is a kind of model of metamodels (a meta-metamodel)
  - ✓ UML infrastructure, UML superstructure, the OCL, relational database models, specializations of UML (i.e.., almost everything) can all be represented within the MOF
  - ✓ Modelling concepts are defined as "metaclasses"
  - ✓ Metaclasses themselves are instance objects of MOF classes
- The MOF involves a 4-layer architecture too.

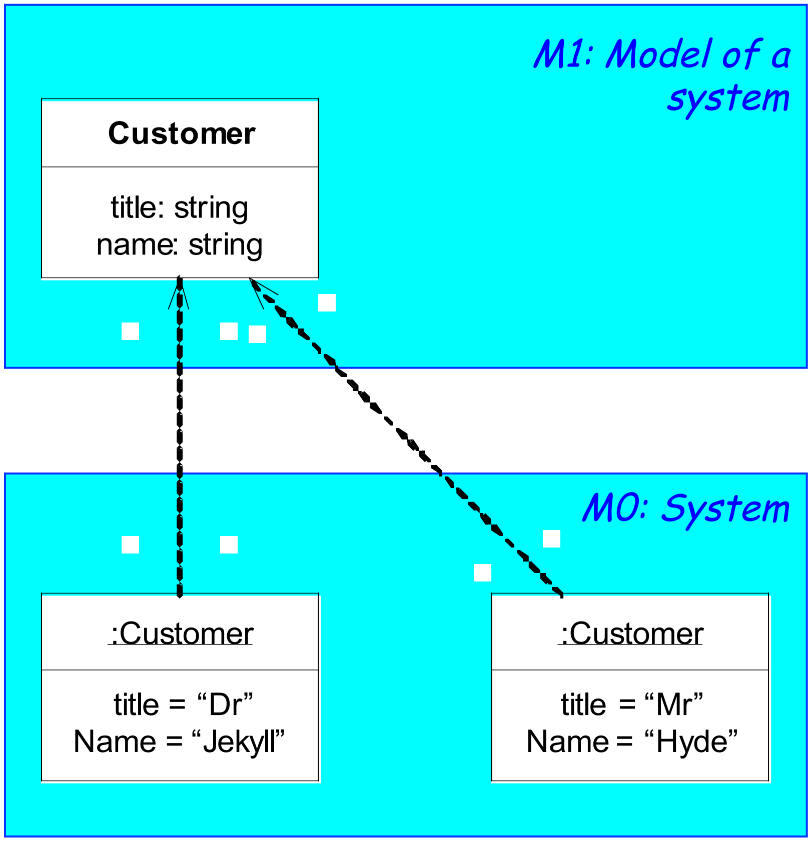
Layers M0 and M1

→You are familiar with M0 and M1

→Layer M0 defines an actual system
- ✓Instances and/or executing instances
- ✓E.g., component instances, customer objects, representing actual customers accessing an e-Commerce system

→Layer M1 is a system model
- ✓Defines the types of entities and relationships that make up a system
- ✓E.g., component specifications, UML class model defining a Customer class

→Every element of M0 is an instance of an element from M1



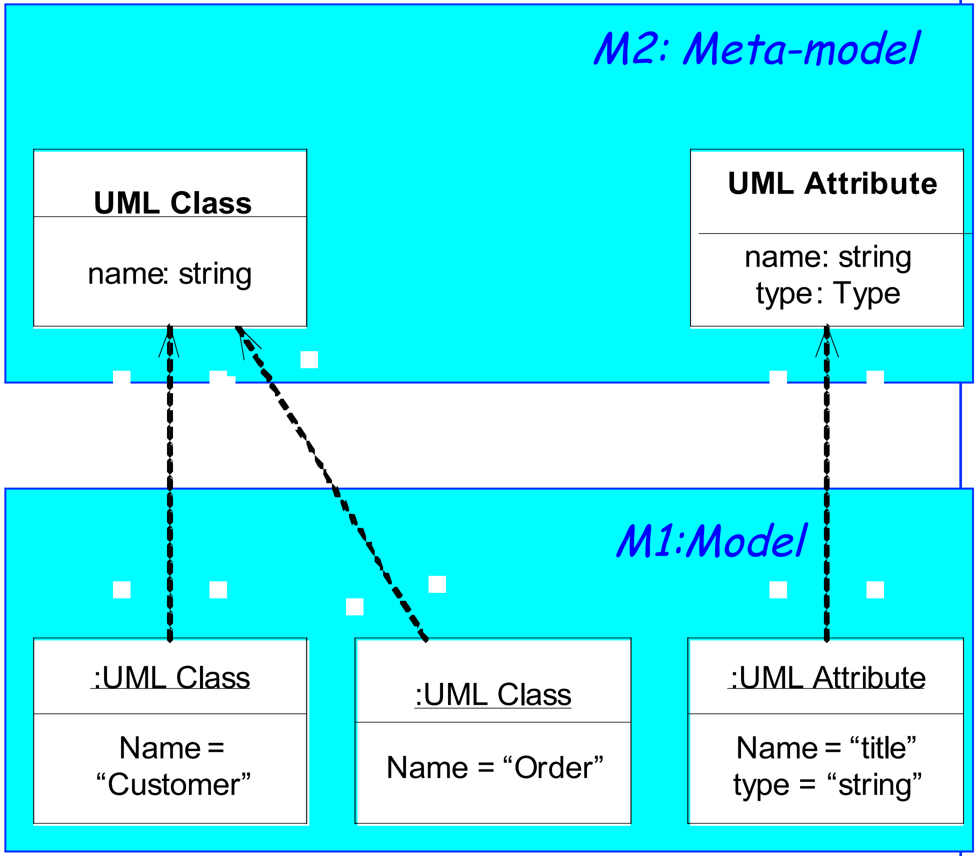
Layer M2

→ Language used to make models in M1 defined by a model in M2.

→ M1 models instances at M0, M2 models concepts at M1

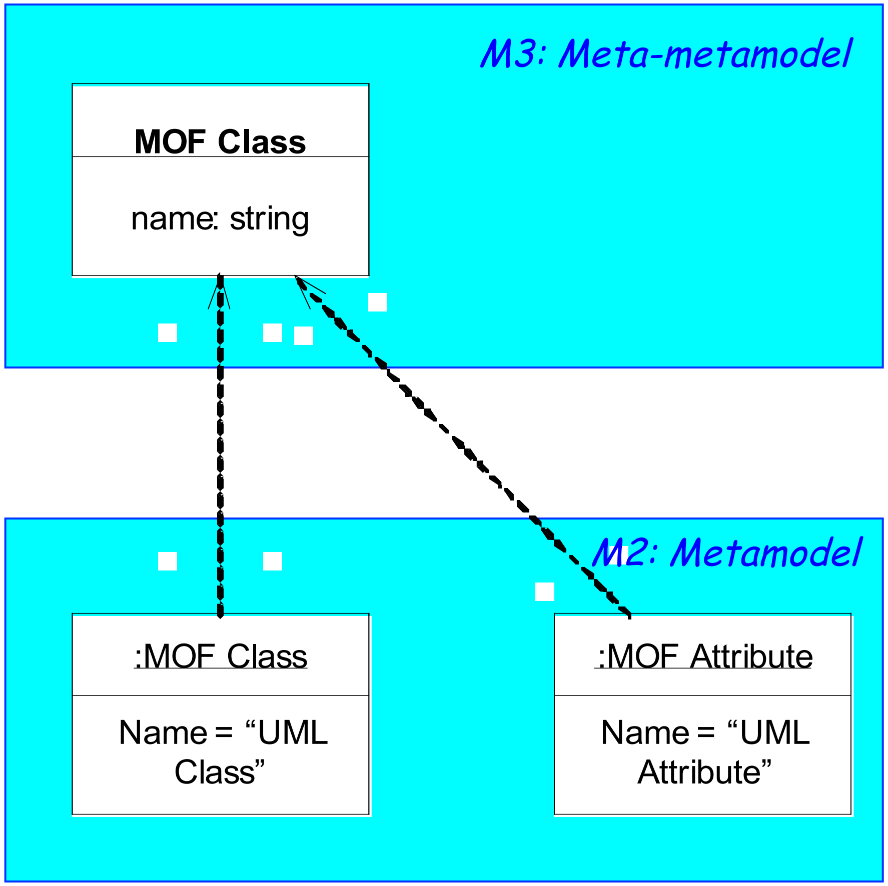
→ For example Class, Association, Component are defined as M2 classes

→ Every element of M1 is an instance of M2



Layer M3

- → Layer M3 defines the model of metamodels in M2 – the meta-metamodel
- → These concepts are defined through class definitions (meta-metaclasses)
- → The metaclasses of M2 are themselves instances of M3 classes
- → The OMG standard for defining M3 models is the MOF – M3 classes are called **MOF classes.**



Why All These Layers?

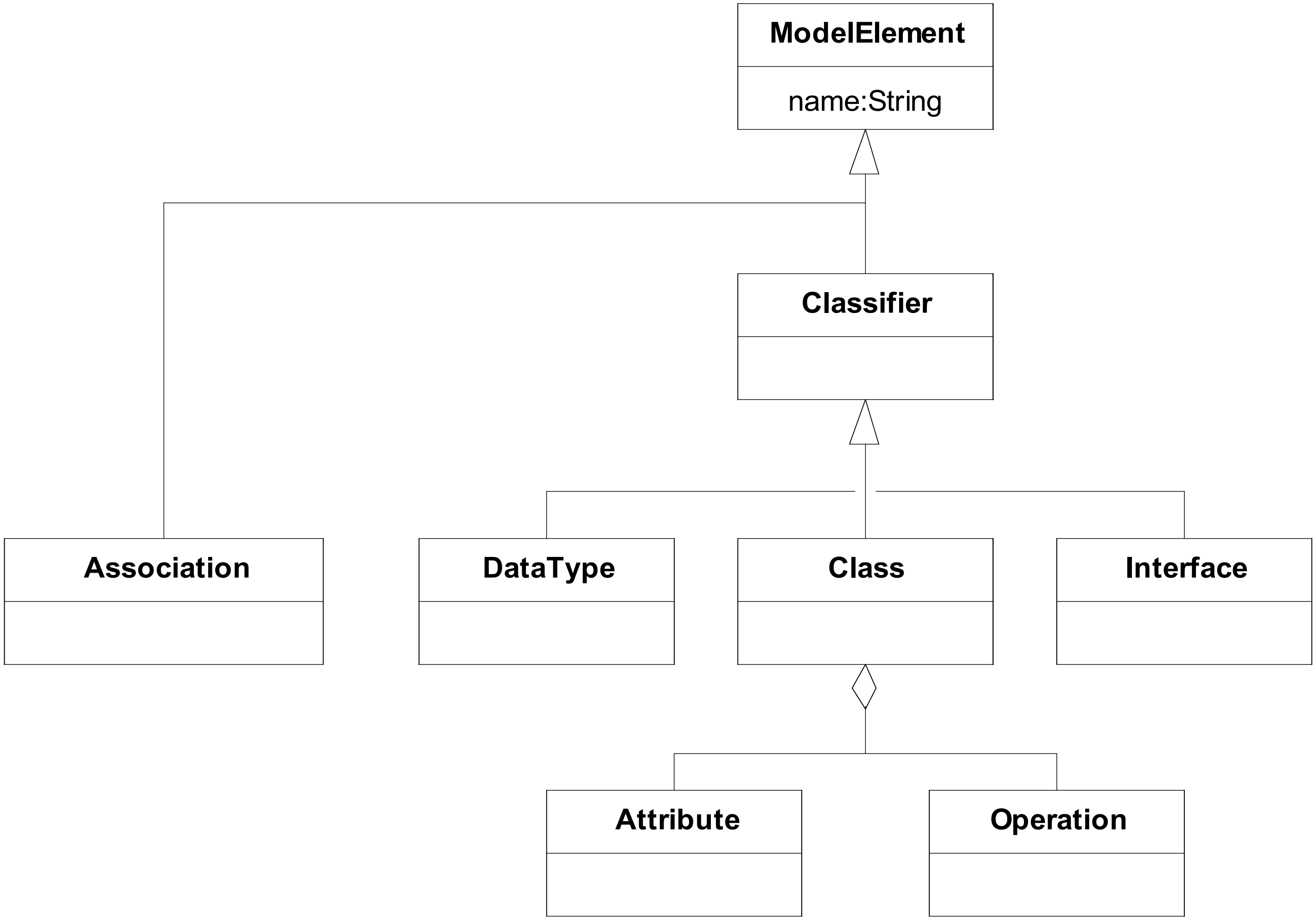
→The usefulness of M0 and M1 should be clear – writing good models is essential to sound software development

→M2 is important so we can define modelling languages

- ✓As we have seen, it is important to define different modelling languages for different contexts
- ✓E.g., a modelling language for architectures COM+ architectures

→M3 is important to manipulate and transform models.

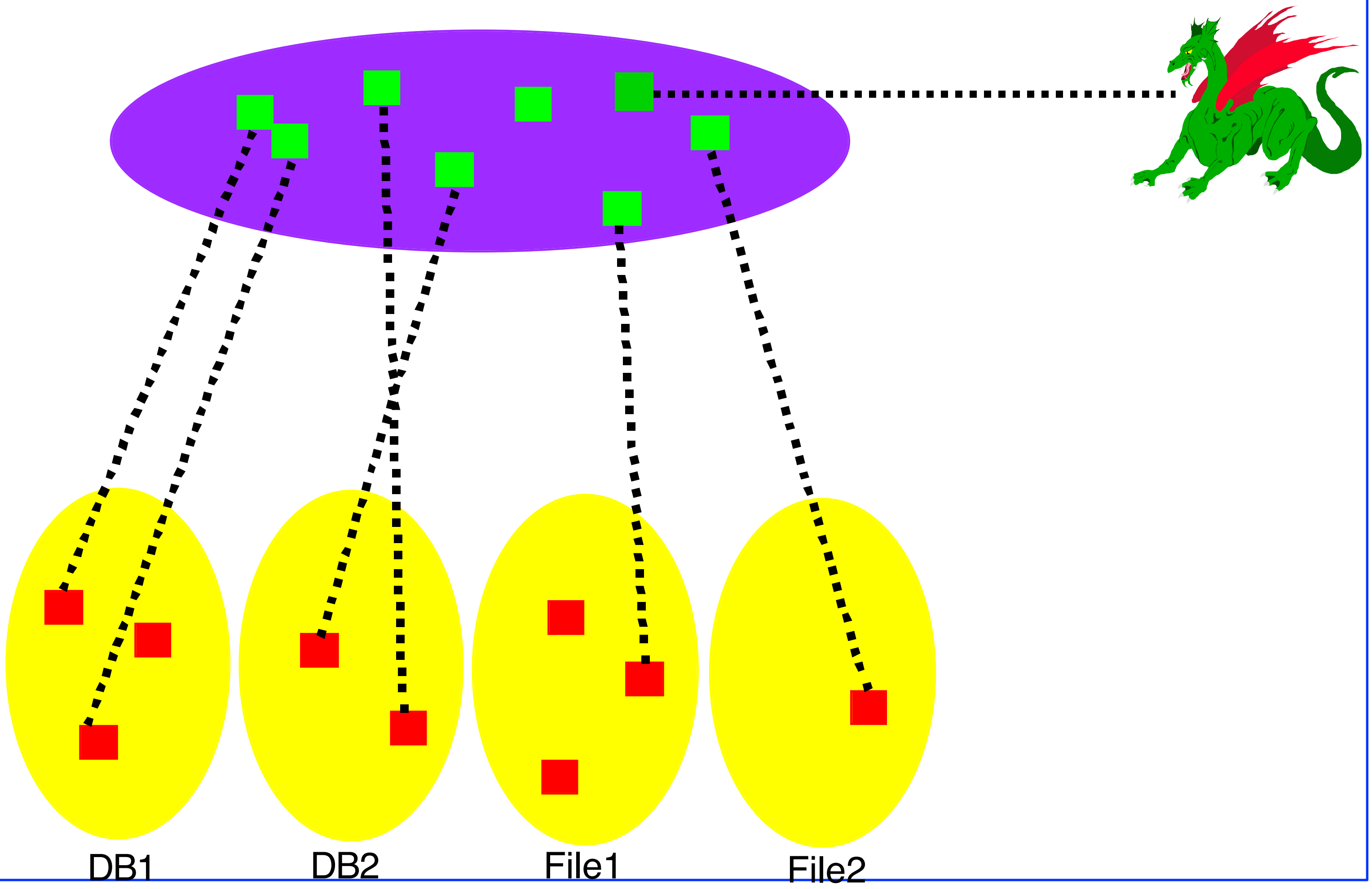
(Part of) UML 2.0 Class Metamodel



Repositories

- A (data) repository stores and manages information about one or more data sources.
- A repository system consists of a conceptual model (often akin to ER model), a model base (information/data/ knowledge base, operations for doing retrievals, updates, check-in/check-out, etc.
- There are many commercial repository products,
  - ✓ Many are hard-coded meta-models (commodity tools)
  - ✓ Most run on RDBMSs (Platinum, SAP, Oracle, MS, …)
  - ✓ Some based on proprietary DBMS (Softlab, Viasoft)
  - ✓ A few run on OODBs (IBM, Unisys)

Repositories as Metadata Managers



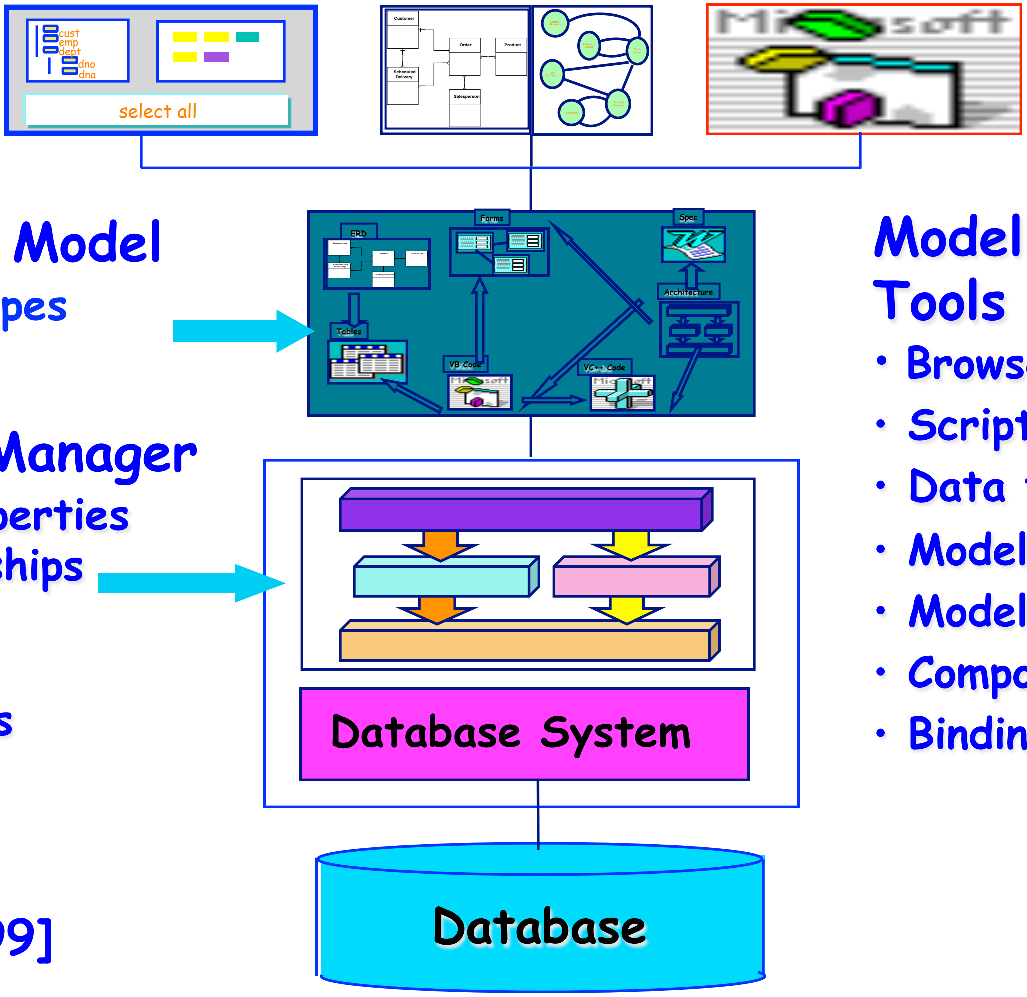
A Repository Product

Information Model

- Predefined types

Repository Manager

- Objects, properties
- Rich relationships
- Extensibility
- Versioning
- Configurations



Database System

Database

Model-Driven Tools

- Browser
- Scripting language
- Data translators
- Model editor
- Model merge
- Component mgr
- Binding/renaming

[Bernstein99]

References

- [Bernstein99] Bernstein, P., "Using Meta-Data to Conquer Database Complexity", Colloquium presentation, University of Toronto, October 1999; http://www.research.microsoft.com/~philbe.
- [Gaarder94] Gaarder, J., *Sophie's World*, Farrar, Straus and Giroux Inc., 1994.
- [Hofstadter79] Hofstadter, D., *Godel, Escher, Bach: An Eternal Golden Braid*, Vintage Books, 1979.
- [Smith84] Smith, B. C., "Reflection and the Semantics of Lisp", Proceedings of the Eleventh Annual Conference on Principles of Programming Languages (POPL), Salt Lake City, 23-35, 1984.