MANCHESTER 1824

The University of Manchester

Binding Ontologies and Coding Systems to Electronic Health Records and Message

Alan Rector[1], Rahil Qamar[1] & Tom Marley[2]

[1]University of Manchester

[2]University of Salford

with acknowledgements to the Semantic Mining Network of Excellence, the UK Connecting for Health programme, and the Terminfo Group

rector@cs.man.ac.uk

www.co-ode.org

www.semanticmining.org/

NIBHI

Plan of the paper

- The use case and requirements
- Theoretical background
- Engineering background
- Issues in using OWL
- Summary

Medical IT's odd organisational structure

- **Separate / independent development**
  - **Medical Ontologies / Terminologies**
    - SNOMED, GALEN, NCI thesaurus, potentiual OBO Disease Ontology, etc.
  - **Medical information models**
    - HL7 messages
    - OpenEHR Archetypes

- **A Common Manifestation of the Oddity**
  - **The "value set" problem**
    - Even after the ontology is designed, the EHR/Message still has to define the valid value sets in terms of it

- **Related problems**
  - **Relation of Simple Knowledge Organisation System (SKOS) and other thesauri with ontologies**
  - **"Classes as Values"**
    - Semantic Web Best Practice Working Group (SWBP) (Noy et al)

What this paper is not about

- Whether any specific coding system, ontology, or information model is appropriate, true, good, bad, or otherwise
  - The methods presented here are pertinent even with a “perfect” ontology and a fully adequate and well aligned information model
    - Although they are more obvious with ill formed ontologies and information systems
    - ...and do allow us to cope with imperfect ontologies and suboptimal record structures
- There is another paper to be written on how best to construct ontologies and information models to fit together
  - This is not it
- We have tested with HL7 and SNOMED becaue that gave us independently developed test cases & NHS paid
  - This implies neither endorsement nor criticism of either

Requirements summary

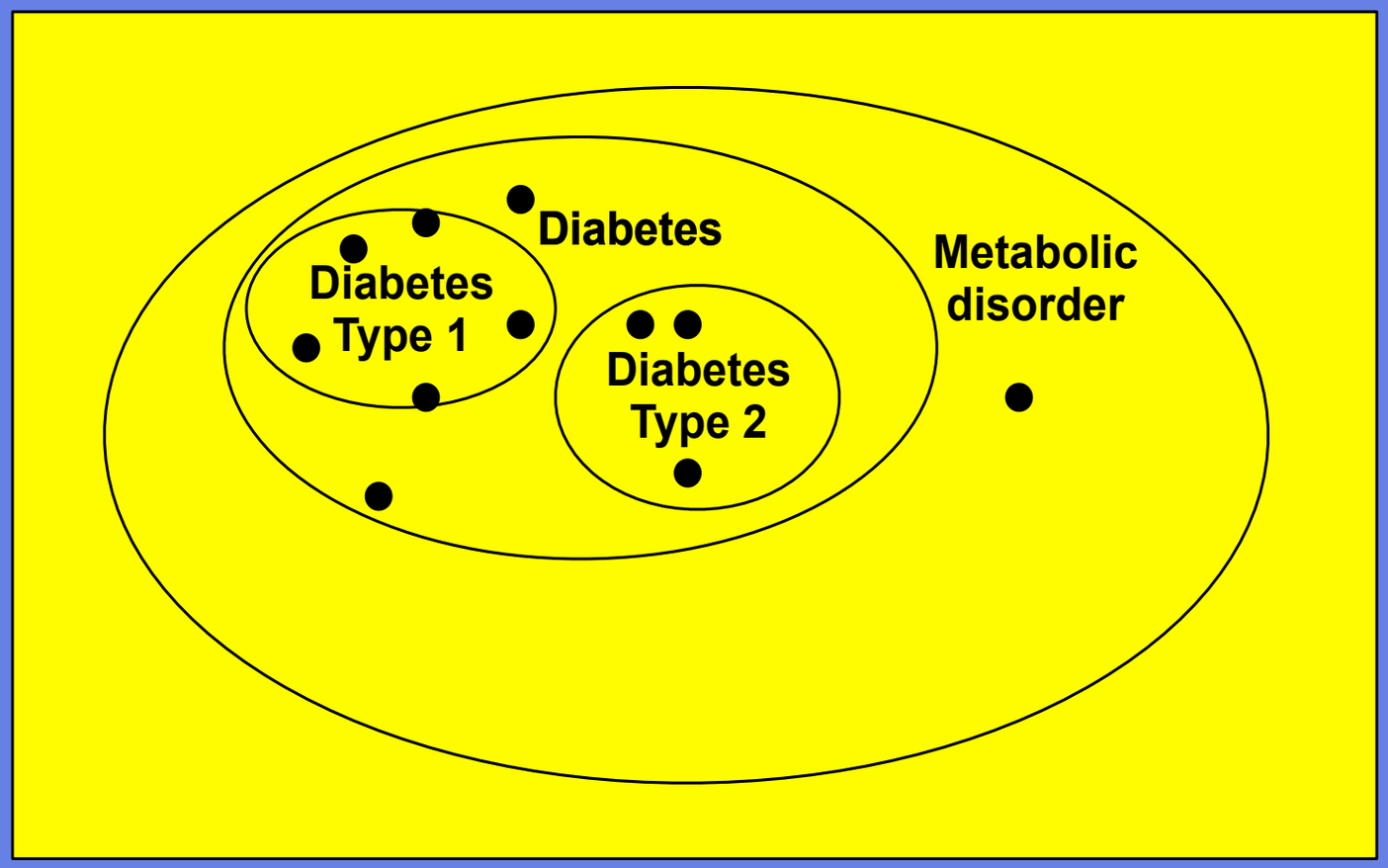
- Explicit “Code Binding Interface” (“CBI”) interface
  - What codes can be used when and where
- Expressivity
  - Meets requirements on next slide
- Compositional coding systems (SNOMED-CT)
  - Deal with compositional coding systems
- Part of general methodology for
  - progressively constraining information models
  - Factoring information models into re-usable submodels
- Use of standard languages with well defined semantics
  - Formally specified independently testable constraints

Expressivity for the “Code Binding Interface”

- Any enumerated list of codes (without their subcodes)
- A code and all its subcodes
- All the subcodes of a code but not the parent code
- Any boolean combination of the above

There is no first order solution to these requirements

- You cannot use a first order expression about the *members* of classes to select a class and exclude its subclasses
  - Unless you really mean to exclude the individual in the subclasses
    - which in general you do not



MANCHESTER 1824

The University of Manchester

Pointed out the obvious

Data structures and what they carry information about are different

- **Information models and "ontologies" are at different levels**
- **Information structures are "meta" to ontologies**
  - **The purpose of modelling an ontology is to represent our conceptualisation of the world**
    - The question is does allow us to make correct predictions about our observations of the world
  - **The purpose of modelling an information structure is to specify valid data structures structures to carry information about that world**
    - To constrain the data structures to just those which a given software system can process

NIBHI

Data structures and what they carry information about have different characteristics

- **Example:**
  - **All persons have a sex**
  - **However not all data structures about people have a field for sex**
- **Information structures are intrinsically closed**
  - **Valid structures can be exhaustively and completely described (up to recursion)**
- **Ontologies are intrinsically open**
  - **We can never describe the world completely**

MANCHESTER 1824

The University of Manchester

Fundamental task for EHRs and Messages

- Begin with logical statements about the world
  - Our best efforts to represent what we believe is true
- Encode those statements in data structures for transmission
  - Provably valid for the software in hand
- Decode data structures faithfully back to logical statements about the world
- With only well defined loss of information

NIBHI

From logical statements about the world to information structures and back again (possibly with loss)

Logical statements about the world:

"*All diabetes are metabolic diseases*"

"John has diabetes and it is brittle"

Valid Specifications for data structures

"Valid diabetic data structures have:

a topic of code for diabetes,

a diagnosis code that is diabetes or one of its subcodes, and

a brittleness code that is one of the subcodes for diabetic brittlenes and nothing else"

MANCHESTER 1824

The University of Manchester

Caveat: *Do not confuse higher order knowledge about the world and meta knowledge about the representation*

- **Higher order knowledge about the classes themselves**
  - **"Endangered species"**
  - **"Darwin described Galapagos Finches"**
- **Information about the software artifact**
  - **Editorial**
    - The representation of "species" in this ontology was authored by Alan Rector on the basis of defintions in the OED, Wordnet, and UMLS
  - **Structural**
    - The identifier for the class "Diabetes"in this representation is "12345"
    - In this representation system
      - *the immediate primitive superclass of "Diabetes" is "metabolic disorder"*
      - *the inferred immediate superclasses are "Chronic disease" , "Cardiovascular risk factor",*
      - *what we are concerned with in this paper*

NIBHI

Representing Information Models and Codes: Basic approach

- An information model can the thought of of as

  A logical theory of classes of information structures

  - The instances of the classes are concrete data structures - EHRs, messages, etc - carrying data about specific patients, tests, organisations, cases of disease, ...

Simple example of a class of information structures for ‘Diabetes’

- Three fields - one example of each case

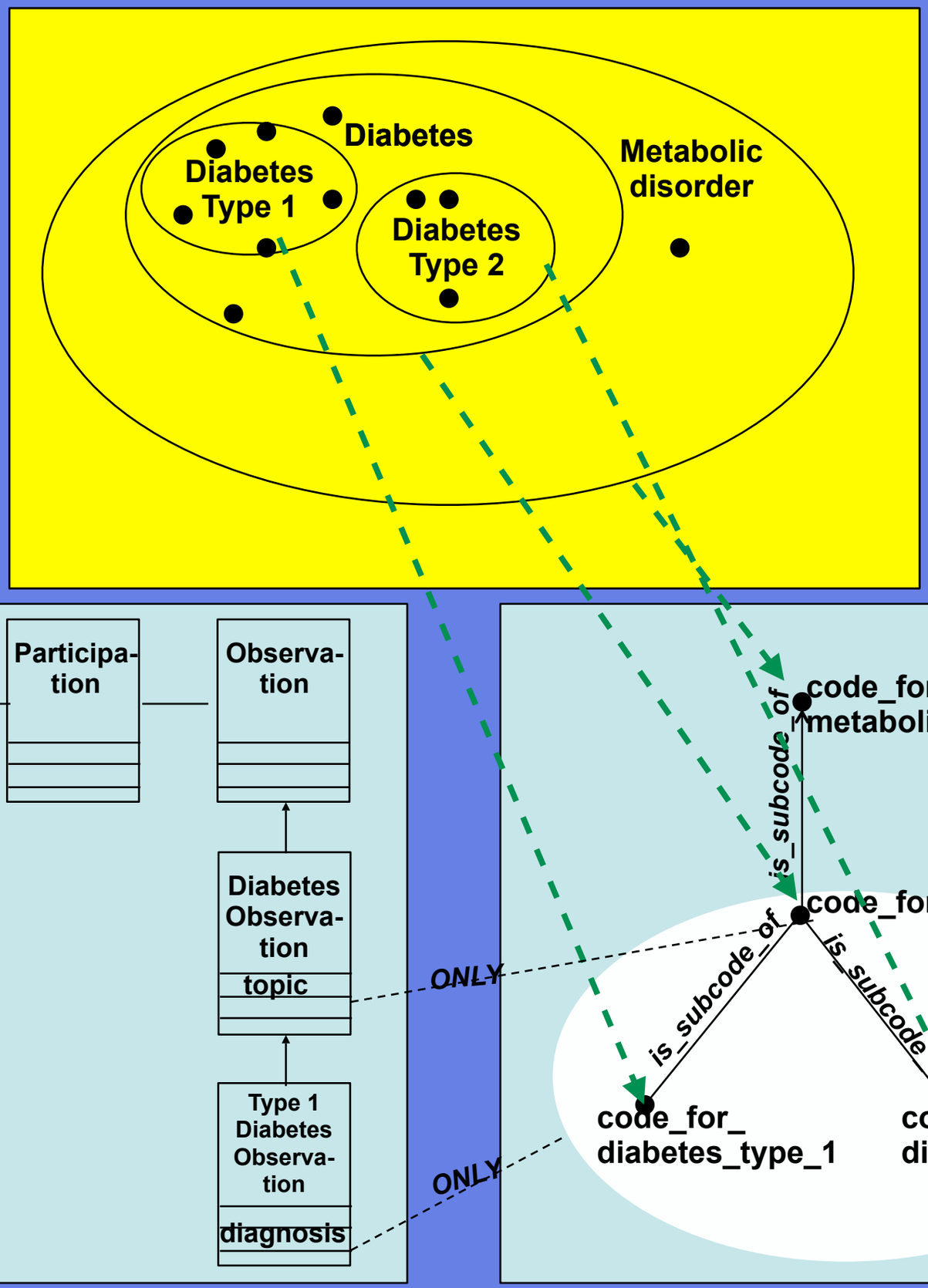
Field	Valid Code Set
Topic	Exact code for diabetes
Diagnosis	The code for diabetes or any kind of diabetes
Brittleness	Any of the subcodes of brittleness but not brittleness itself

“ontology” representations and “codes”

- The instances of “ontology” represent (conceptualisations of) things in the world
  - Cases of diabetes
  - Patients
  - Insulin metabolism
  - islet cells
- The instances in data structures are data items in human artefacts
  - Information structures of associations and attributes, elements, etc.
  - Individual codes represent representations of classes *in given representation of an “ontology”*
    - *What is true of a given logical / mathematical / software* ***artefact***

Model of Meaning & Model of Data structure

Model of Meaning/ "ontology"



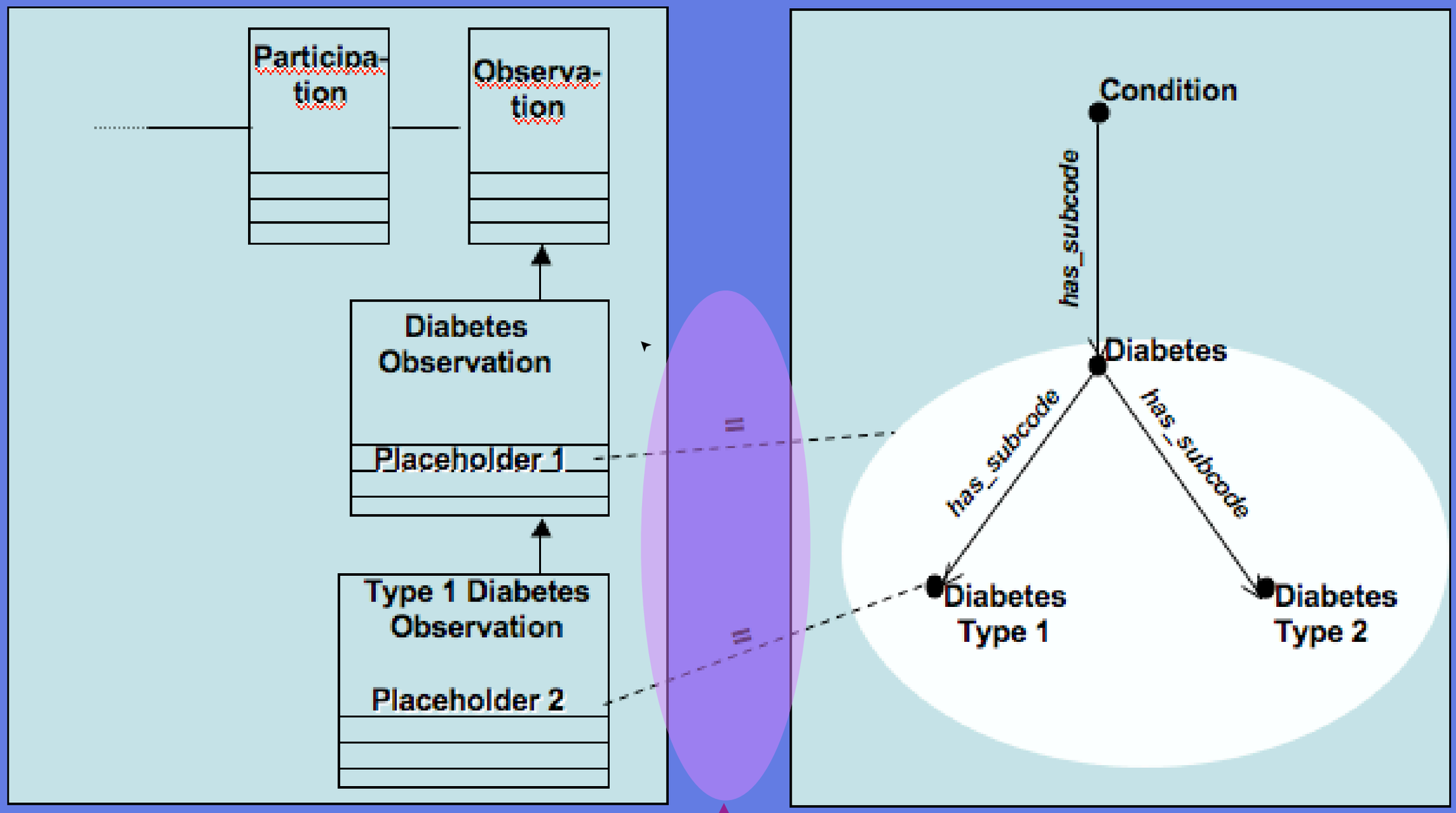
Model of data structures in Information Model

Model of codes in Information Model

Software Engineering Issues
Clean testable, maintainable
"The Code Binding Interfaced"

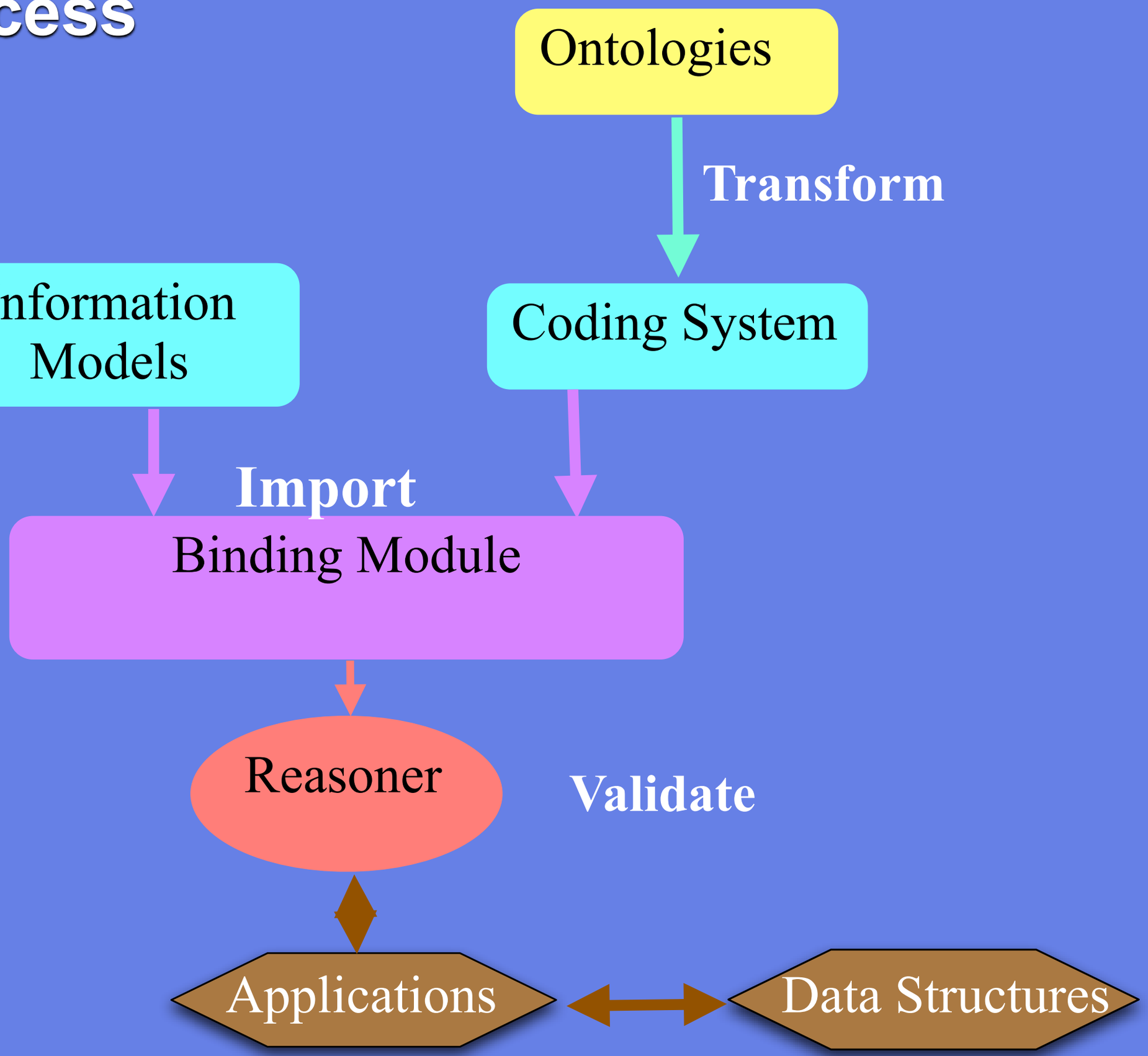
- **Placholder codes**
  - **Classses of codes used in the the information model to express constraints**
    - Only a member of this class of placeholder codes can be used here
  - **But not defined in the information model**
    - *Defer binding*
- **A notion partly taken from Beale's Archetype Definition Language's "Ontology " section,**
  - **which would better be called a "Code Binding Section"**
  - **And whose semantics are weak - we propose here a methodology for a stronger semantics**

Binding using placeholders



Code Binding Interface

Outline of Process



Doing it in OWL

- Why use OWL?
  - Treat OWL as a logic constraint language with tools and community
    - Can be used to represent ontologies
    - Can also be used to represent constraints data structures
  - Generic with well defined semantics:
    - It means what it means

- Forces distinctions often left implicit to be made explicit
  - The good news:
    - *Can* be explicit about which expressions are open or closed
    - *Can* be explicit about layering and metamodels
  - The bad news:
    - *Must* be explicit about which expressions are open or closed
    - *Must* be explicit about layering and meta models

- NB using some constructs from OWL 1.1
  - but included in all current OWL reasoners

Key issues in OWL

- **Open World Reasoning**
  - **Negation means provably false**
    - Expressions not present are merely “under-specified”
      - OWL, Descriptin Logics, FoL
  - **False means cannot be proved true**
    - What is missing is false
      - Databases, logic programming, most rules systems, SQL, etc.
- **Absence of the Unique Name Assumption**
  - **Two entities are different *only* if we say they are different**
    - Means that it is easy to say that two things are logically equivalent
    - In most database, logic programming, systems, etc. if two things have different names they are different
- **Therefore: Owl often requires additional explicit axioms (since you have the freedom to do either, you must specify which)**
  - ***Closure* axioms when closed world reasoning is required**
  - ***Distinguishing* axioms when things are known to be different**

Example definition of a class of data structures

CLASS Diabetic_data_structure →

has_attr EXACTLY 1 Topic,

has_attr EXACTLY 1 Diagnosis,

has_attr EXACTLY 1 Brittleness,

Example definition of a class of data structures with placeholders

CLASS Diabetic_data_structure →

has_attr EXACTLY 1 Topic,
has_attr SOME (Topic & has_code SOME Placeholder_diabetes_only_code),

has_attr EXACTLY 1 Diagnosis,
has_attr SOME (Diagnosis & has_code SOME Placeholder_diabetes_or_sub),

has_attr EXACTLY 1 Brittleness,
has_attr SOME (Brittleness & has_code ONLY Placeholder_diab_brittleness_sub).

Example definition of a class of data structures with placeholders and closure axiom

CLASS Diabetic_data_structure →

has_attr EXACTLY 1 Topic,
has_attr SOME (Topic & has_code SOME Placeholder_diabetes_only_code),

has_attr EXACTLY 1 Diagnosis,
has_attr SOME (Diagnosis & has_code SOME Placeholder_diabetes_or_sub),

has_attr EXACTLY 1 Brittleness,
has_attr SOME (Brittleness & has_code ONLY Placeholder_diab_brittleness_sub),

has_attr ONLY (Topic OR Diagnosis OR Brittleness).

The Code Binding Interface
A set of equivalence axioms

- *The code for diabetes (only)*

  Placeholder_diabetes_only_code ↔
  {code_for_diabetes}

- *The code for diabetes or any of its subcodes*

  Placeholder_diabetes_or_sub ↔
  {code_for_diabetes} OR is_sub_of VALUE code_for_diabetes

- *The code for any subcode of diabetic_brittleness*

  Placeholder_diab_brittleness_sub ↔
  is_sub_of VALUE code_for_diabetic_brittleness

A fragment of the ontology and the derived coding system

- CLASS Diabetes →
  Metabolic_disorder,
  has_quality EXACTLY 1 Brittleness.

  CLASS Diabetes_type_1 →
  Diabetes,
  is_caused_by_some (Damage AND
  has_locus SOME Pancreatic_islet_cells).

  ...

- INDIVIDUAL code_for_diabetes ∈

  CODE,
  has_sub VALUE code_for_diabetes_type_1,
  has_sub VALUE code_for_diabetes_type_2,
  has_sub ONLY {code_for_diabetes_type_1,
  code_for_diabetes_type_2}.

A fragment of the ontologym and the derived coding system

- **Note:**
  - **Not everything in the ontology is necessarily represented in the coding system**
    - In this coding system causation by pancreatic islet cell defect is ignored in
  - **We could introduce additional features in the coding system not from the ontology**
    - Whether or not this would be a good idea
  - **We could deliberately create a more complex relationship for has sub**
    - A reasonable way to account systematicaly for “broader than/narrower than”

Summary

- "Ontologies" represent (our conceptualisation of) the world
  - The criteria for correctness is our predictions of observations of the world
- Data Structures are used to convey information about (our conceptualisation of) the world
  - They criteria for correctness is validity for use in multiple information systems
- Codes are data structures which represent the representations in ontological artefacts
  - They are "meta" to the ontology
- Ontologies & coding systems will continue to be developed independently
  - Formal interfaces and deferred binding are therefore required between them
    EVEN WERE THEY PERFECT
  - *The mechanism of placeholder codes is presented (in part) here is sufficient to form such a "Code Binding Interface"*