COMP62342
Using Ontologies

Sean Bechhofer
sean.bechhofer@manchester.ac.uk

Uli Sattler
uli.sattler@manchester.ac.uk

Today

- ✓ SKOS
- ✓ Linked Data
- Some clarifications of misunderstandings I saw in your essays
- More on Profiles
- Using Ontologies
  - for MCQ generation
  - in an information system
- Wrap Up

Clarifications

OWL, DL, semantics

- Check out this example
- Does this ontology entail

  Furniture SubClassOf
  hasShape exactly 1 Shape

  ?
- Can we improve this ontology?

```
Class: Square SubClassOf Shape
Class: Circle SubClassOf Shape
Class: Rectangle SubClassOf Shape

DisjointClasses: Square, Circle, Rectangle

Class: Shape SubClassOf
              (Square or Circle or Rectangle)

Property hasShape Range: Shape
                  Domain: Furniture

Class: Furniture SubClassOf
              hasShape some Shape

Class: Chair SubClassOf Furniture and
              hasShape only Rectangle
```

MANCHESTER 1824

The University of Manchester

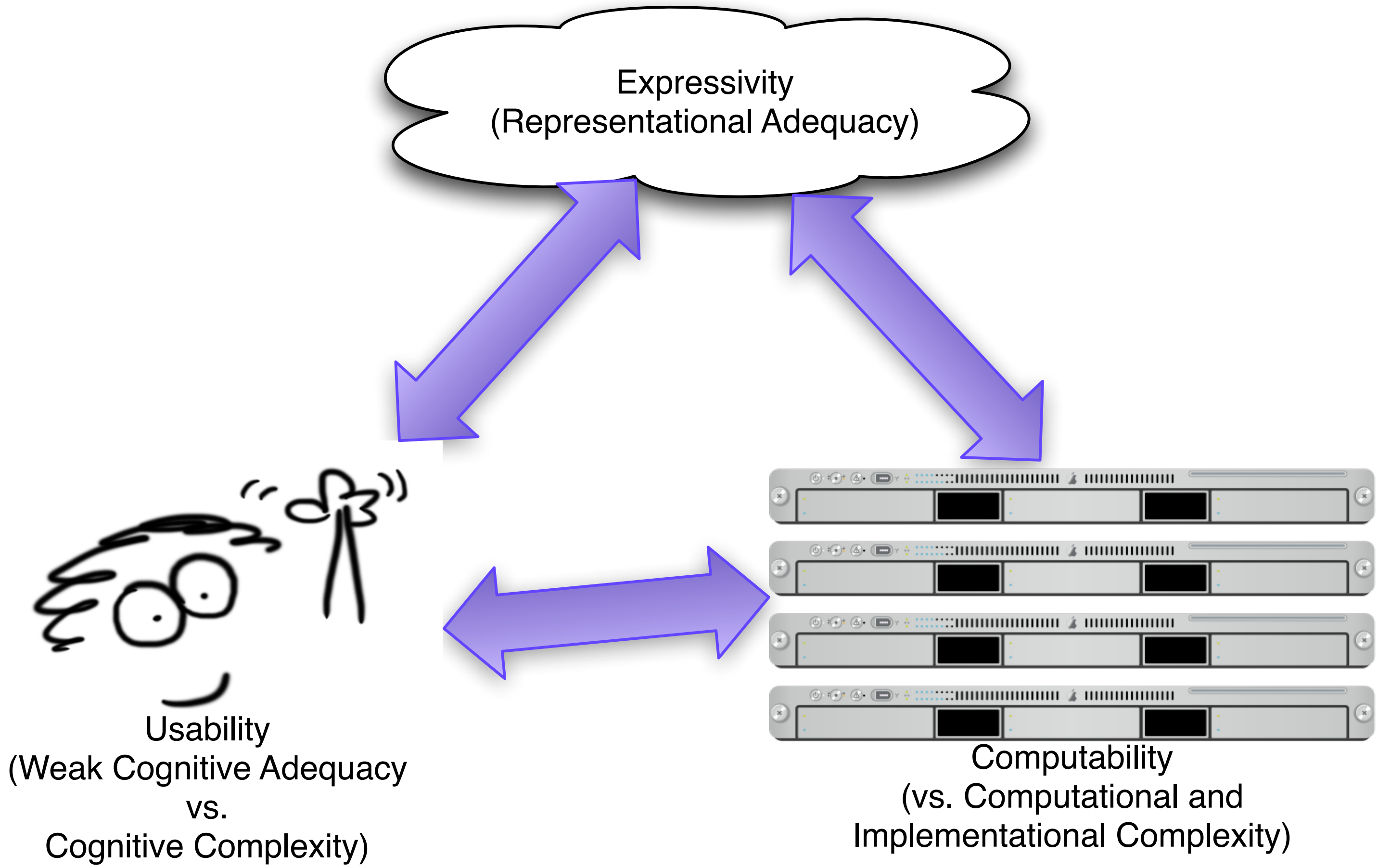
Part-Whole Relation

- hasPart and isLocatedIn are 2 different properties.
- Which one relates
  - your lungs and your chest?
  - a bed and its bedroom
  - an apple and its tree
- How do they interact?

ObjectProperty: hasPartOf InverseProperty isPartOf
objectPropertyCharacteristic Transitive
ObjectProperty isLocatedIn SubPropertyChain isLocatedIn o isPartOf

More on Profiles

The Design Triangle



MANCHESTER 1824

The University of Manchester

OWL Expressivity

- OWL is an expressive ontology language providing a number of class forming operators and axiom types
  - full Booleans
    - and, or, not
  - Property Restrictions
    - some, only, min, max, exact
  - Enumerations
    - Explicit classes formed from individuals
  - Subclass and Equivalence
  - Property
    - Hierarchies
    - Chains
    - Characteristics: functional, inverse
- Expressivity comes with a (computational and cognitive) cost
  - Do we always need all this expressivity?

OWL Profiles

- …are trimmed down sublanguages/fragments that trade

  expressive power for *efficiency of reasoning*

- Restrictions are placed on the
  - operators, e.g., no or, no not
  - axiom types supported, e.g., no InverseObjectProperties(p q)

- Three profiles, EL, QL and RL are defined in the OWL Profiles Recommendation

  http://www.w3.org/TR/owl2-profiles/

- …each of them is maximal for that profile's computation complexity, i.e., weakening any restriction results in increased computational complexity
- Other profiles could be defined

Profiles (from last week)

- OWL 2 EL:
  - only ‘and’, ‘some’, SubProperty, transitive, SubPropertyChain
  - it’s a *Horn* logic
    - no reasoning by case required,
    - no disjunction, not even hidden
  - designed for big class hierarchies, e.g. SNOMED,
- OWL 2 QL:
  - only restricted ‘some’, restricted ‘and’, inverseOf, SubProperty
  - designed for querying data in a database through a class-level ontology
- OWL 2 RL:
  - no ‘some’ on RHS of SubClassOf, …
  - designed to be implemented via a classic rule engine
- For details, see OWL 2 specification!

The University of Manchester

MANCHESTER 1824

Ontologies
and
(Knowledge) Graphs

Ontologies and Graphs?!

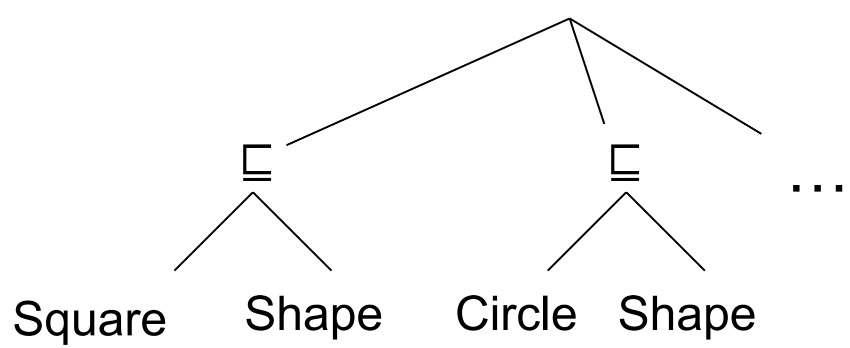
- An OWL ontology O is a **set of axioms** that
  - can be (inconsistent)
  - entails other axioms
  - can be the result of parsing an OWL file
    - in one of the many OWL syntaxes
  - can be viewed as a **graph:**
    - e.g., the parse tree of its axioms

Class: Square SubClassOf Shape
Class: Circle SubClassOf Shape
Class: Rectangle SubClassOf Shape

DisjointClasses: Square, Circle, Rectangle

Class: Shape SubClassOf
(Square or Circle or Rectangle)

Property hasShape Range: Shape



Ontologies and Graphs?!

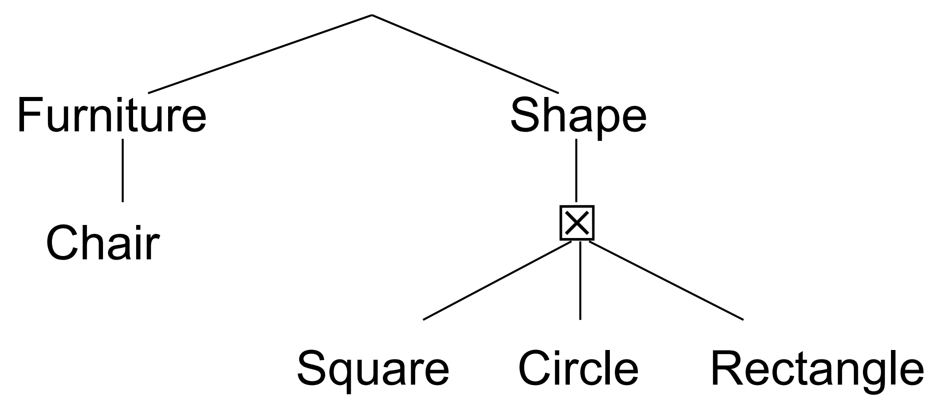
- An OWL ontology O is a **set of axioms** that
  - can be (inconsistent)
  - entails other axioms
  - can be the result of parsing an OWL file
    - in one of the many OWL syntaxes
  - can be viewed as a **graph:**
    - e.g., the **asserted class hierarchy** (see Protégé)

Class: Square SubClassOf Shape
Class: Circle SubClassOf Shape
Class: Rectangle SubClassOf Shape

DisjointClasses: Square, Circle, Rectangle

Class: Shape SubClassOf
(Square or Circle or Rectangle)

Property hasShape Range: Shape



Ontologies and Graphs?!

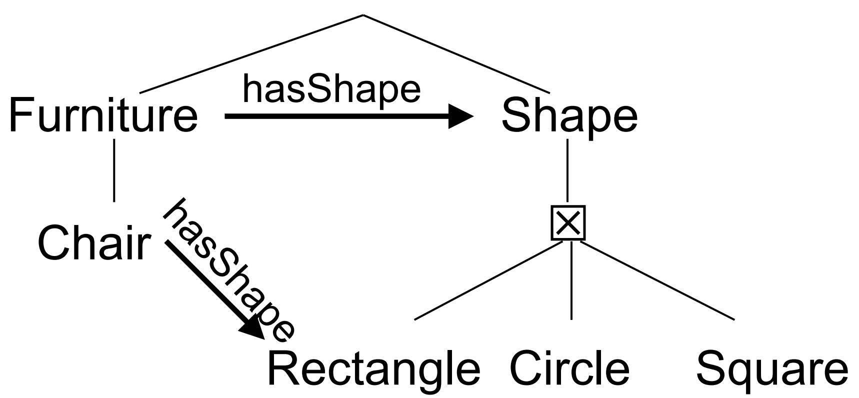
- An OWL ontology O is a **set of axioms** that
  - can be (inconsistent)
  - entails other axioms
  - can be the result of parsing an OWL file
    - in one of the many OWL syntaxes
  - can be viewed as a **graph:**
    - e.g., some **adorned inferred class hierarchy**

```
Class: Square SubClassOf Shape
Class: Circle SubClassOf Shape
Class: Rectangle SubClassOf Shape

DisjointClasses: Square, Circle, Rectangle

Class: Shape SubClassOf
              (Square or Circle or Rectangle)

Property hasShape Range: Shape
```

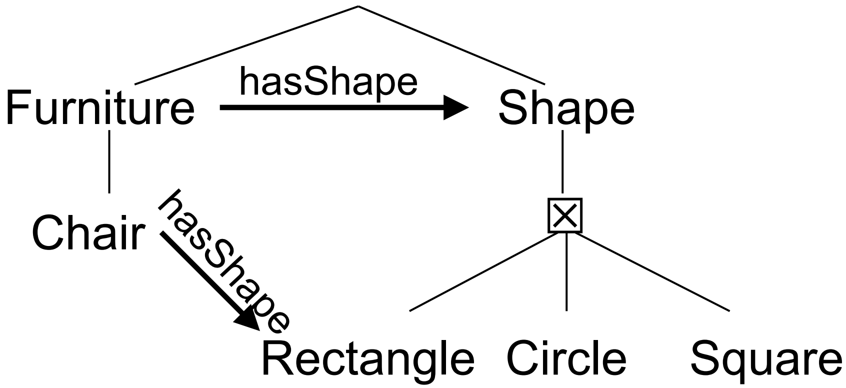


Which adorned graphs to build?

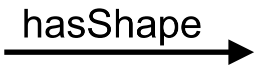
Property hasShape Range: Shape
Domain: Furniture

Class: Furniture SubClassOf
hasShape some Shape

Class: Chair SubClassOf Furniture and
hasShape only Rectangle



How many arrows do we need?
And where do we put them?

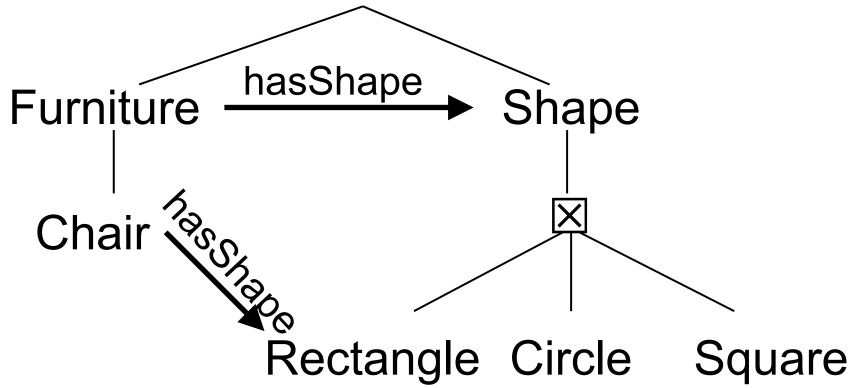


Which adorned graphs to build?

Property hasShape Range: Shape
Domain: Furniture

Class: Furniture SubClassOf
hasShape some Shape

Class: Chair SubClassOf Furniture and
hasShape only Rectangle



What is **the graph of an ontology?**
Ask - different people mean different things!

Why Ontologies?
What do we use them for?

Remember from last week:

- An OWL ontology O is a **document:**
  - therefor, it cannot **do** anything - as it isn't a piece of software!
  - in particular, an ontology cannot **infer** anything (a reasoner may infer something!)
- An OWL ontology O i
  - with 'impo
  - correspon
  - the OWL A
    - parse a
    - access it
  - a **reasoner** is only interested in this set of axioms
    - **not** in annotation axioms
    - see https://www.w3.org/TR/owl2-primer/#Document_Information_and_Annotations
    - https://www.w3.org/TR/2012/REC-owl2-syntax-20121211/#Annotations

So, what to do with these documents/ ontologies?

E.g., let's create MCQs!

- Given that some
  - ontology captures rich domain knowledge
  - assessment/MCQ generation is costly & relevant
  - in particular for healthcare & medicine

➡ why not auto-generate MCQs from ontologies?

- Building on research we have done so far,
  - in particular on how to make **good** MCQs, e.g., control difficulty
- we have been exploring this with **Elsevier**
  - towards more complex MCQs, e.g., patient cases
- interesting new app with **new reasoning problems**
  - similarity of concepts and cases

…over to Ghader!

the next slides are for fall-back

Anatomy of an MCQ

Which of these is **not a** mammal? — Stem

1. Dolphin
2. Whale
3. Tuna
4. Chimpanzee

Distractors (1, 2, 4)
Key (3)
Options
MCQ

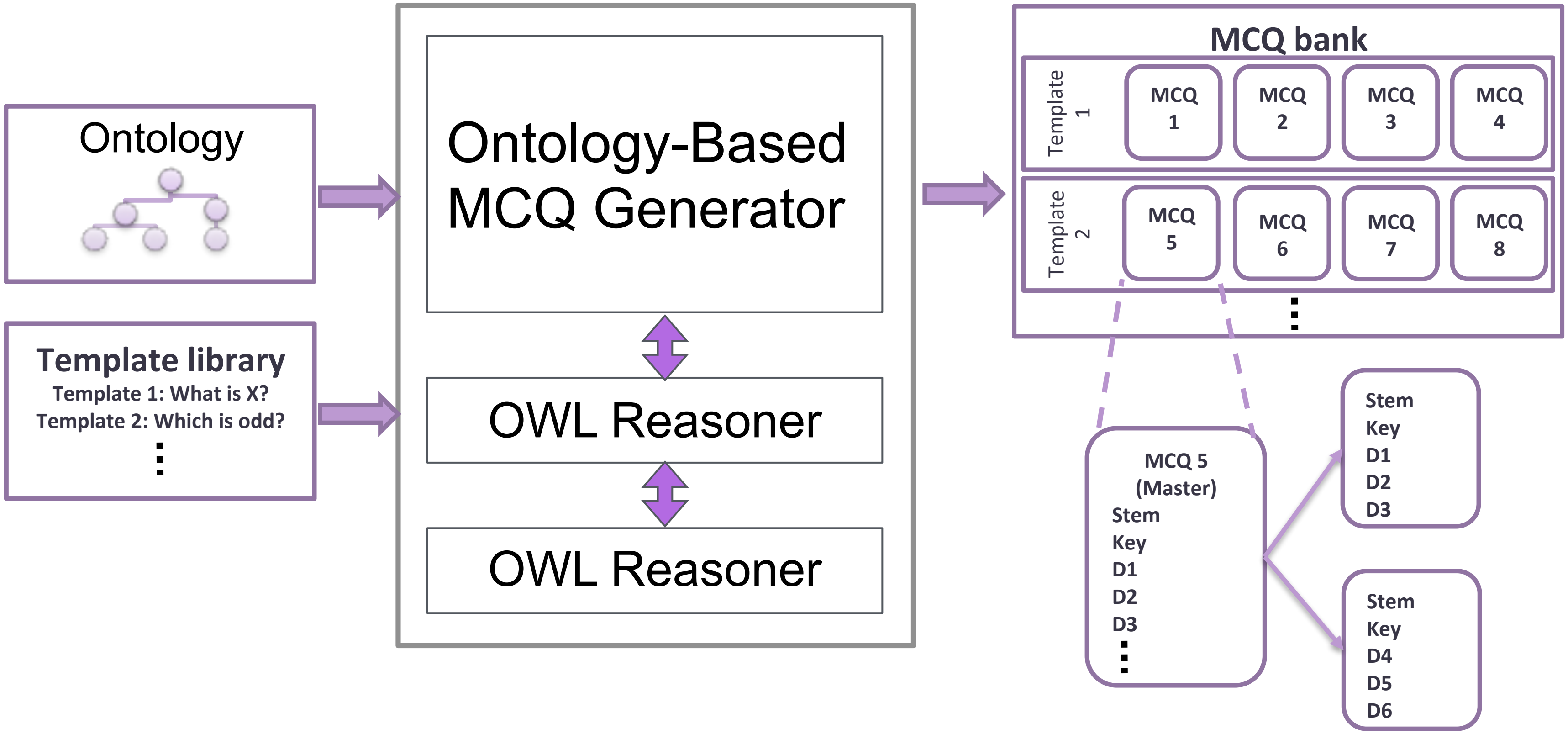
Follows a **template:**

Stem: Which of these is **not a** (Class) X**?**

Distractors: Y with $O \models Y \sqsubseteq X$

Key: Y with $O \not\models Y \sqsubseteq X$

Ontology-based MCQ generation



The more similar D is to K,
the more difficult is Q.

Anatomy of an MCQ

Which of these is **not a** mammal?

1. Dolphin
2. Whale
3. Tuna
4. Chimpanzee

1. Zebra
2. Giraffe
3. Tuna
4. Chimpanzee

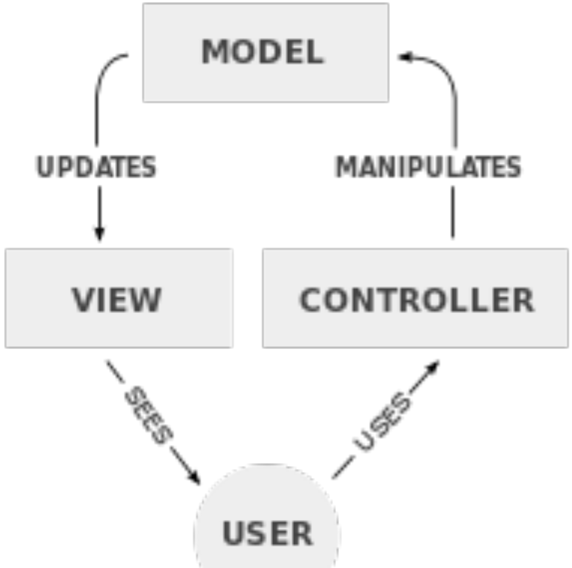
(Why) Is Whale more similar to Tuna than Giraffe?

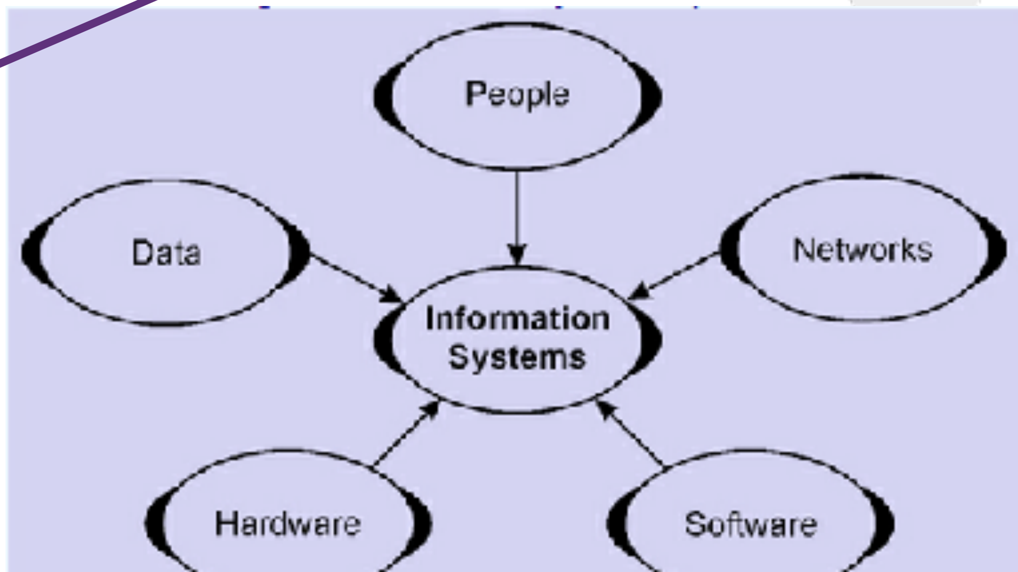
How to measure similarity of classes in ontologies?

What else do we do with ontologies?

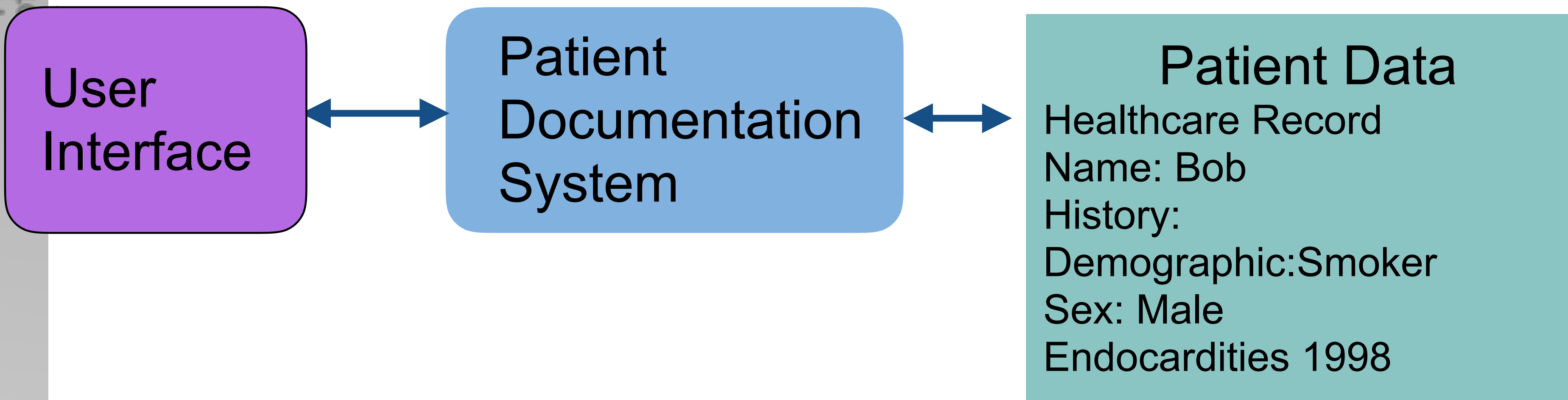
- OBIS: Ontology-Based Information Systems
- Think MVC/Front-End Back-End
- IS needs to store some data, in:
  - relational database
  - no-SQL database
  - files
  - XML docs
  - …
  - Ontology

Which?



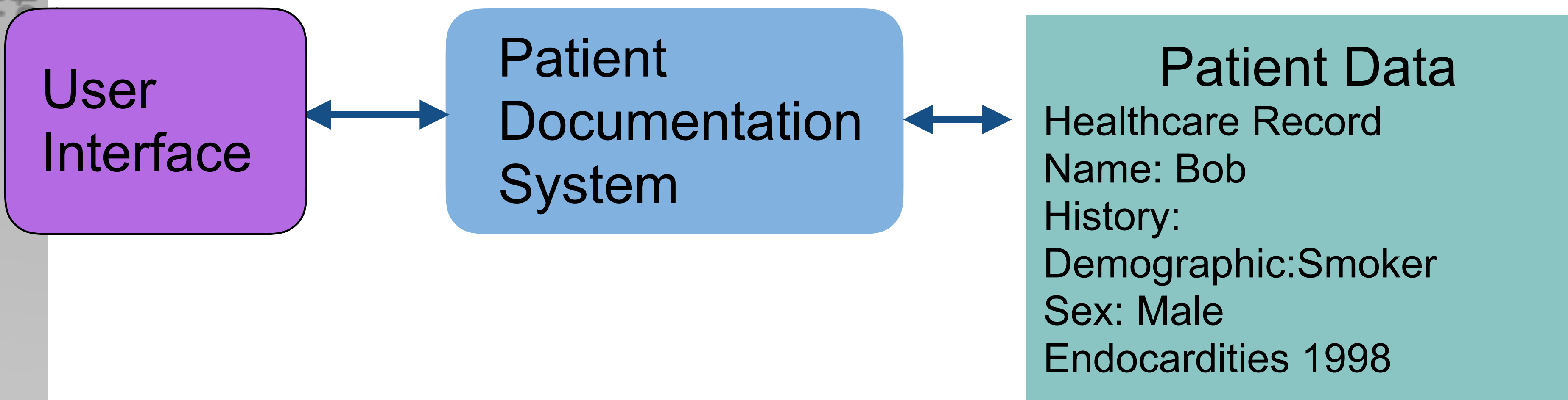


E.g.: Patient Documentation System



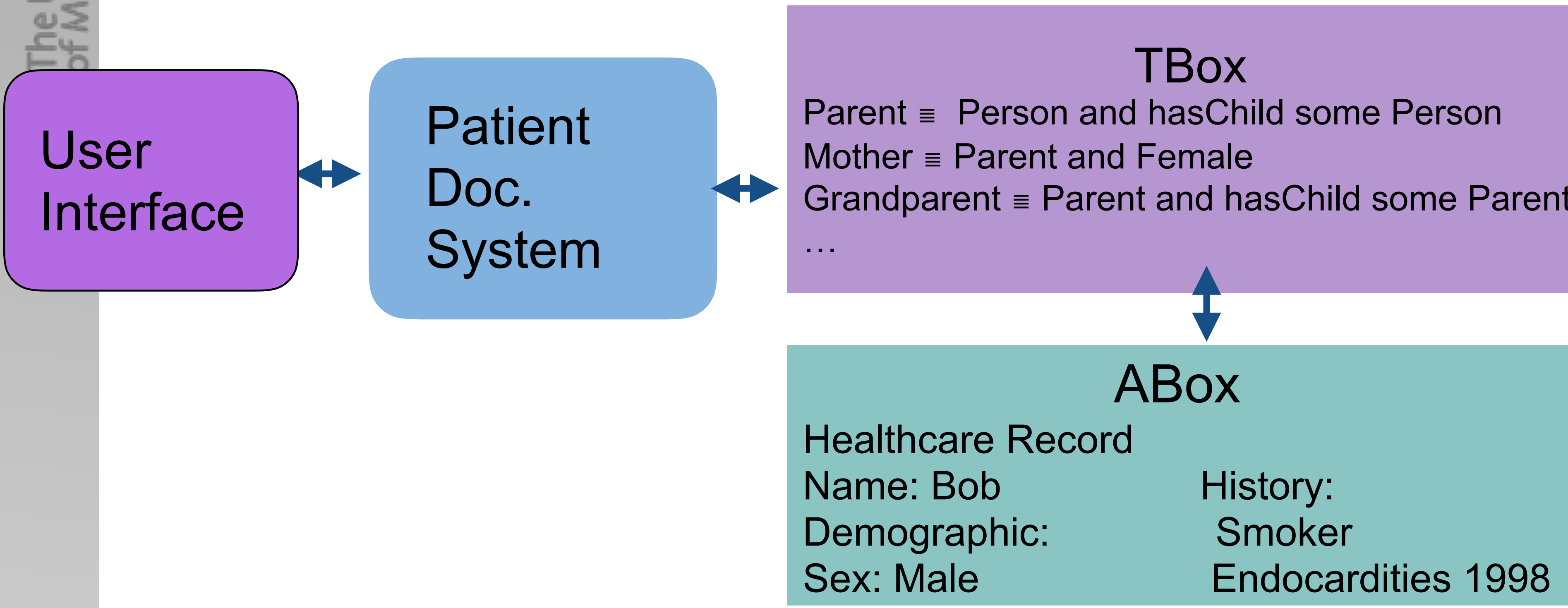
- Information System relies on Patient Data
  - recorded in different systems with possibly different structures
  - recorded by different clinicians with different styles
- Holding Data in DB:
  - many complex queries that need to change with changes in medicin

E.g.: Patient Documentation System



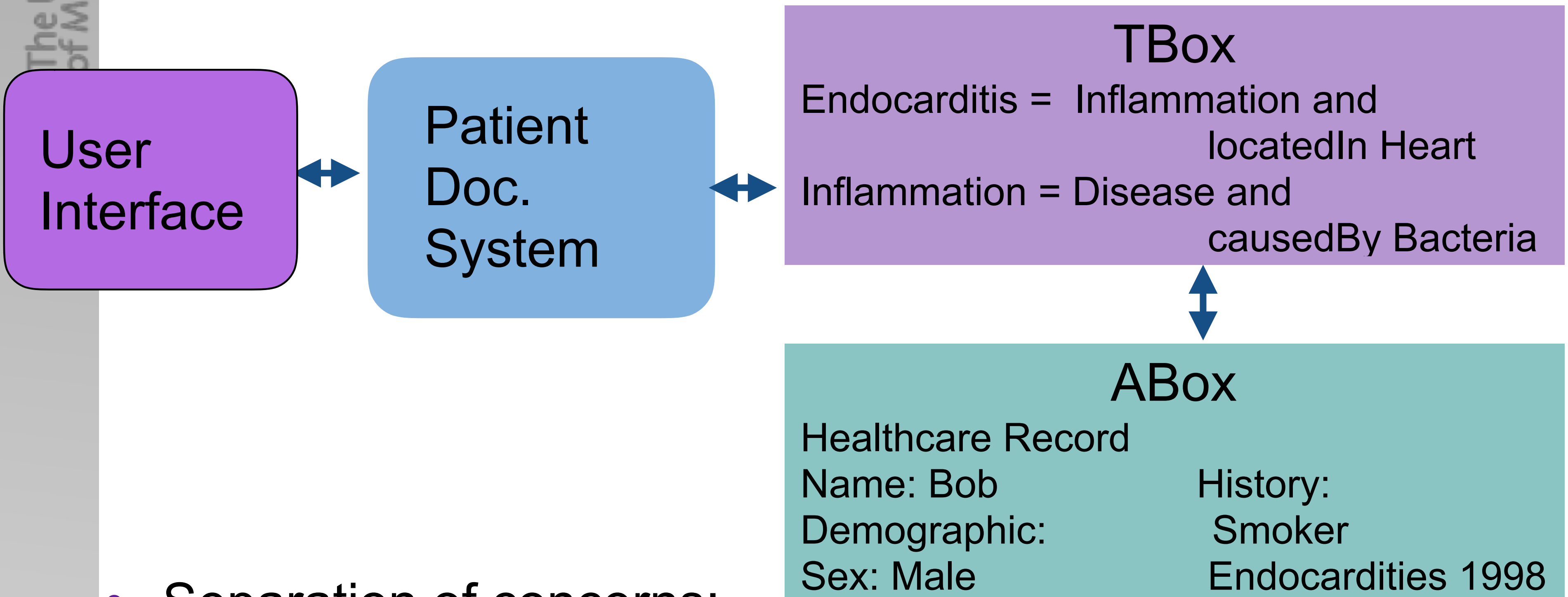
- Toy example: get all *Parents* from database - get
  - those who have a *known child*
  - those described as *Mother* or *Father*
  - those described as *Grandmother* or *Grandfather*
  - …

Why basing ISs on Ontologies?



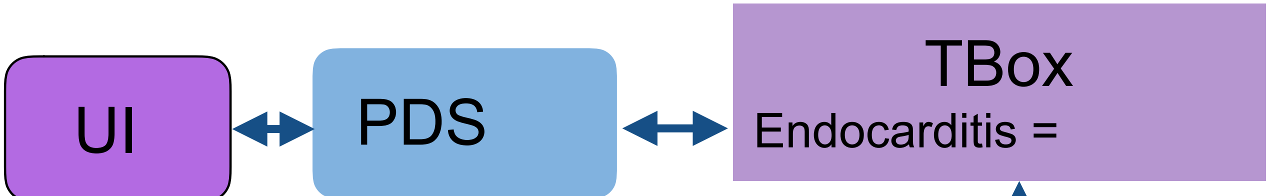
- Toy example: get all *Parents* from ontology:
  - use suitable TBox and
  - retrieve all those who are **entailed** to be an instance of *Parent*
  - …

Why basing ISs on Ontologies?



- Separation of concerns:
  - – background knowledge & terminology into ontology
  - – data into DB or ABox
  - • suitably linked/mapped
  - – behaviour into program code

Why basing ISs on Ontologies?



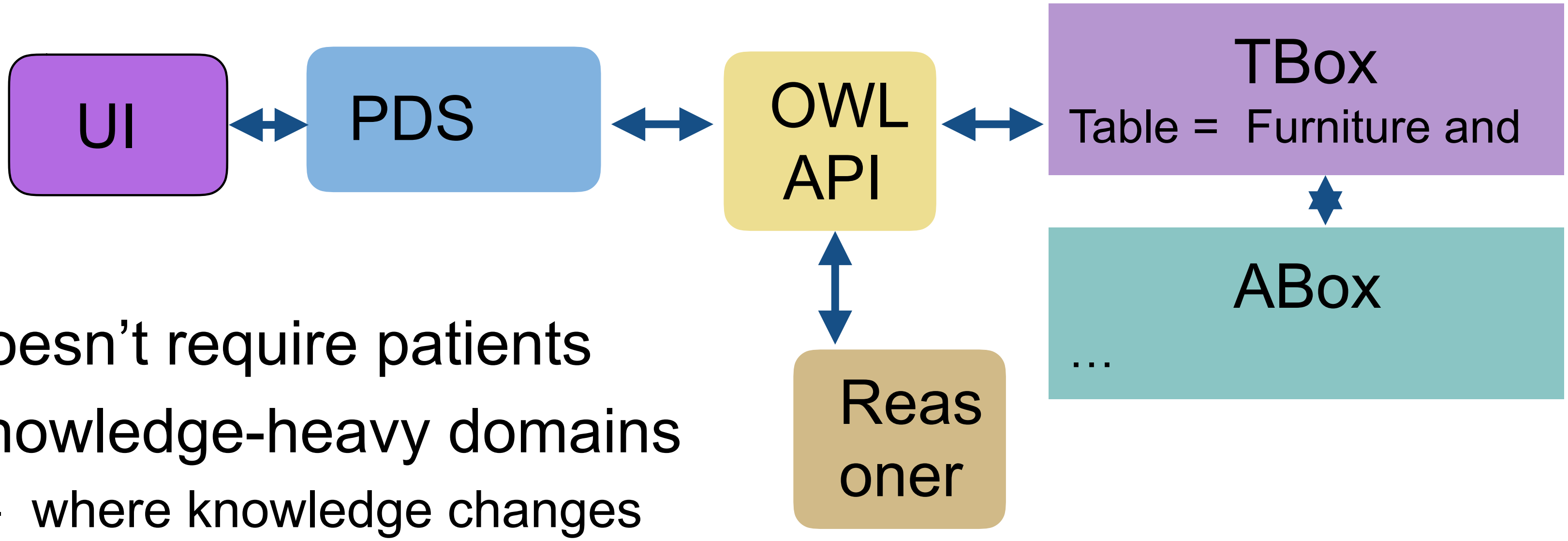
- Separation of concerns

✓ flexible access to data can deal with
  - **incomplete** knowledge
  - data coded in different ways
  - complex expressions: post-coordination!
  - data coded & queries on varying levels of granularity

✓ via terms as appropriate to IS
  - same data can be linked to different ontologies

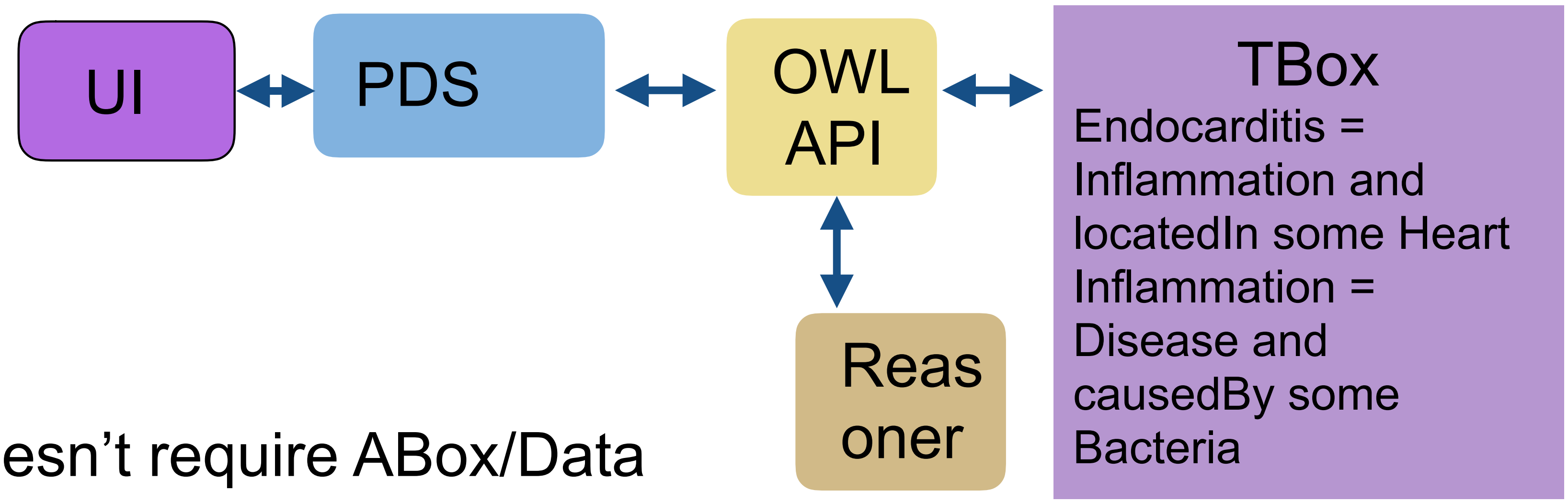
✓ maintainable
  - changes in background knowledge reflected in updated ontology

Ontology-Based ISs



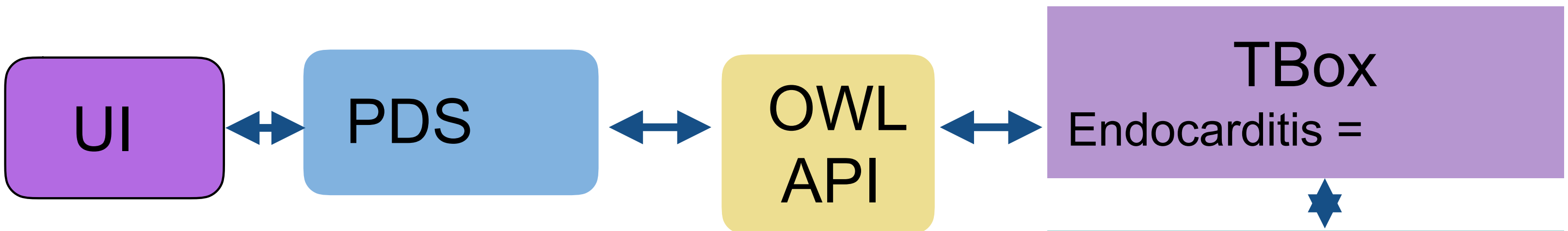
- doesn't require patients
- knowledge-heavy domains
  - where knowledge changes
- Example:
  - furniture
  - restaurants & food properties: allergies, ethical,…
  - biochemistry
  - defence, intelligence
  - (nano) engineering
  - recruitment/skills management

Ontology-Based ISs



- doesn't require ABox/Data
- sometimes only TBox
  - e.g., NCI Thesaurus, where a large medical thesaurus & its hierarchy is maintained as the Inferred Class Hierarchy of rich OWL ontology

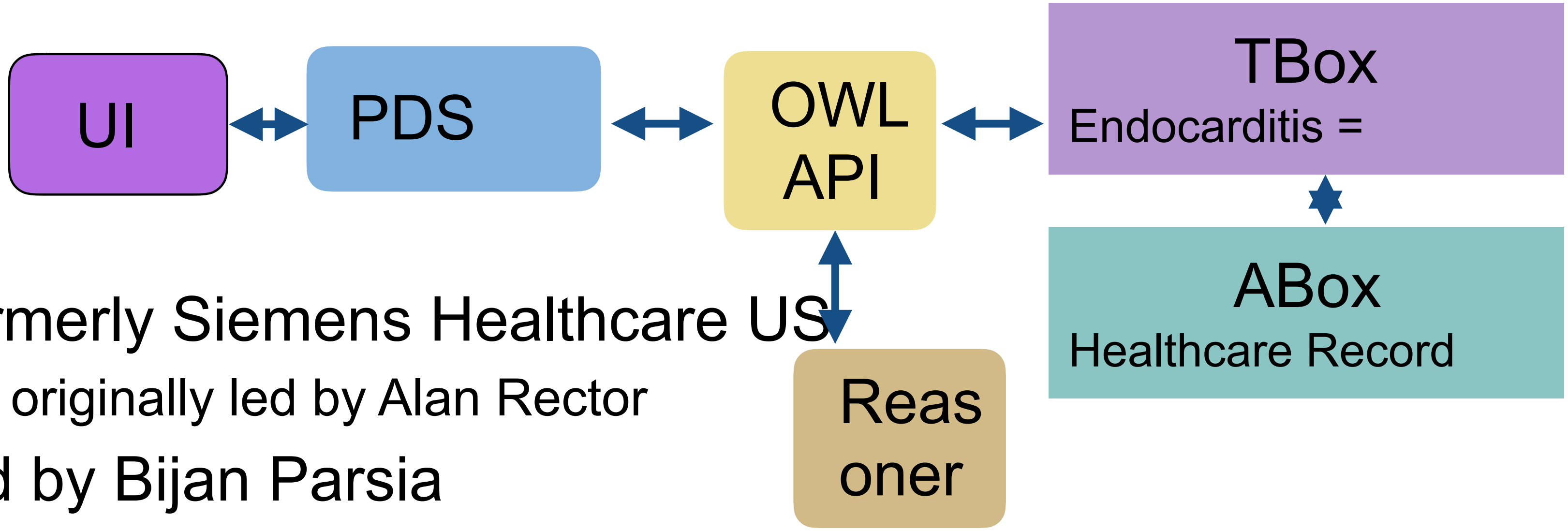
Building Ontology-Based ISs



- involves difficult design choices
  - which ontologies?
    - build own?
    - reuse/extend/combine others?
  - how to map?
  - what to put in OWL classes or Java classes?
  - how to make it scale?
  - which tools to use?
    - OWL API
    - reasoner

We tried to give you knowledge & understanding to answer these questions

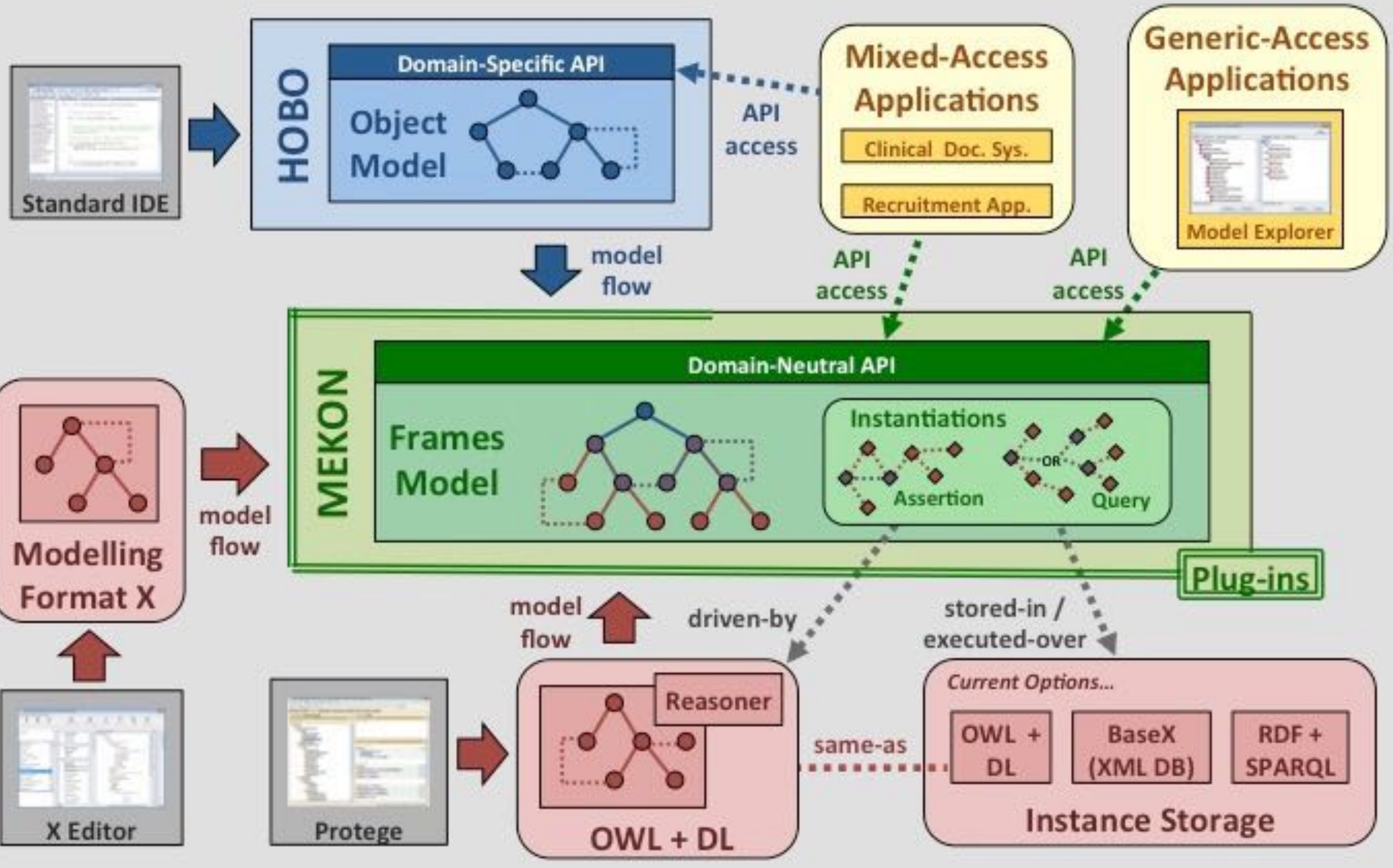
E.g., Cerner Collaboration



- formerly Siemens Healthcare US
  - originally led by Alan Rector
- led by Bijan Parsia
- concerned with patient documentation systems:
  - given the information about patient we have so far
  - what should we ask/document next?
- fine example where
  - **behaviour** depends on but differs from
  - static knowledge captured in ontology
- led to development of Chiron, Hobo, Mekon,…

MEKON & HOBO

Java frameworks for building ontology-driven applications



Colin Puleston, University of Manchester (puleston@manchester.ac.uk)

Challenges of Building an OBIS

- Reasoner Performance/Scalability
  - if your usage scenario doesn't fit reasoner performance, consider
    - other reasoner; see ORE
    - suitable profile
    - your scenario
- New (reasoning) problems crop up
  - entailment explanation (see Protégé's "?")
  - modularity (in OWL API *tools*!)
  - similarity (see MCQ generation)
- Training, maintenance
  - who's building/maintaining the ontology?
  - who's writing the code?
- Tool support
  - many OWL tools around, but few stable/commercial

That’s it!

MANCHESTER 1824

The University of Manchester

What have we learnt?

- Intro to Knowledge Representation
  - Why do this?
- Knowledge Acquisition
  - What & how do we model?
- Formalisation, Ontology Patterns
  - How to represent things (in OWL) in actionable way?
- Semantics and Reasoning
  - Models, entailments, tableau, classification, …
  - What exactly is it we are saying and what are the consequences?
- OWL API: actions with ontologies
- SKOS
  - An alternative to OWL using OWL
- Linked Data
  - Using OWL or RDF(S) for data on the Web
- Usage of ontologies

MANCHESTER 1824

The University of Manchester

Coursework this Week

- Core Task: Furniture Ontology (50% of your coursework mark)
  - Submit your **ontology** (group) by Monday, May 13
  - Submit your **report** (individual) by Thursday, May 16 (65% of CT mark)
  - **Peer assess** your ontologies, by Thursday, May 16 (35% of CT mark)
- W5 Query application
  - use the OWL API to query an ontology
  - Monday, May 13
- W5 Post-coordination
  - a short essay

MANCHESTER 1824

The University of Manchester

Your furniture Ontology

- An ontology of furniture
- Classes that enable us to represent furniture & answer competency questions like
  - Which pieces of furniture are found in the greatest number of rooms?
  - Which items of furniture are available in different sizes?
  - What are those sizes?
  - …see BB for more CQs: we've added some more!
- Class hierarchy organised using the PIMPS upper ontology.
- Peer assessed

- Plus a reflective report on how you built it, interesting aspects of the model

The University of Manchester

MANCHESTER 1824

Exam

- Online Exam via Blackboard
- Two hours
- Multiple Choice Questions
- Short Essays
- Answer **all** questions

- …use Forum for questions!