Models of Uses & Models of Meaning

Alan Rector
Information Management Group / Bio Health Informatics Forum
Department of Computer Science, University of Manchester

rector@cs.man.ac.uk
co-ode-admin@cs.man.ac.uk

www.co-ode.org
protege.stanford.org
www.opengalen.org
www.clinical-escience.org
Meaning, Use & Implementation

• Models of use
  – When it is needed – what should be “handy”
    • To say? to Ask? To Find? To use?
    • The “how” as well as the “What”

• Model of meaning
  – What can be be said – what it implies
    What can be asked – what it includes
    What do the ‘chunks’ mean?

• Model of implementation
  – How it is to be used
    • Ontology nested in EHRs
      – HL7 Terminfo
        » HL7 + S-CT
      – CEN 13606 + Archetypes + Compositional terminology
    • Interface specifications, regression testing, conformance testing
Uses –

- Many systems embody one use
  - Data entry
    - ORCA, MedCin
  - Decision support
    - QMR vocabulary
  - Epidemiology
    - ICD
  - Information indexing
    - MeSH

- A few systems make split explicit
  - GALEN/PEN&PAD Perspectives
  - Read “Entry terminology”
  - S-CT “Close-to-user” forms
Lessons from GALEN/PEN&PAD

• Architecture – orthogonal
  – Meaning
  Language
  Classification
  Entry
  Indexing – “Reference Knowledge Resources”

• Use follows practice rather than meaning
  – Use requires additional information
    • Some additional information can be indexed on the ontology
  – Use does not interoperate in detail
    • Formalisms for specifying model of use can interoperate
Separation incorporated in the GALEN Server

Client

Server

Multilingual Module

Concept Module

Code Conversion Module

Indexing Module

A single point of access for language, classification, code conversion, and indexing - well separated internally
Data entry

- In a setting for a task by a user about a condition
- What is common & must be fast?
  What is necessary & must be safe
- Any data entry protocol is in part a guideline
Classification

• Model of use for epidemiology and recording
  – Strongly built in and difficult to disentangle
  – Volume II of ICD most explicit

• Index to starting points and rules
  – Simplification
    • But leaves much to do
Classification
Map rather than Model for Legacy Systems

Hypertension excluding in pregnancy
Hypertension during Pregnancy

No Match
One Match
More than one Match
Exclusions
Integrating rather than Cross Mapping

- Vascular surgery
  - Arterial repair
  - Venous repair
    - REA X00yA Repair of pulmonary vein
    - REA 790F1 Repair of partial anomalous pulmonary venous connection
    - REA 79050 Correction of total anomalous pulmonary venous connection
    - NCSP FFG Transponering av totalt anomalt munk
    - NCSP FFG00 Reparation vid suprakardiell TAP
    - NCSP FFG20 Reparation vid infrakardiell TAP
    - NCSP FFG30 Reparation vid blandad TAPVD
    - REA X00yX Repair of systemic venous pathway
    - REA XE0Fj Repair of acquired arteriovenous fistula
    - NCSP PHC Sutur av ven
    - NCSP FFJ Interatrial korrektion av systemvenöst blodflöde
    - USE V201 Hémostase d’un accès vasculaire artério-veineux par suture ou par patch
    - NCSP PWA Reoperation p g a av sårruptur
    - V432 Suture latérale de la veine cave inférieure sus-hépatique ou des veines sus-hépatiques
Authoring

- Intermediate representation
  - Split domain expert, ontology expert, and implementor tasks
  - Separate information collection from implementation
Intermediate Representation

Expertise to be conserved

OpenGALEN
The ontology nested in the EHR

the ehr (hl7 rim)

[moodCode="Event"
subject="Relative"
code={
  diabetes (subject person_in_family)
}]

the ontology (snomed-ct)

→

<family_hx (assoc_find Diabetes)>

the combined meaning

What is legal? Required?
Mandatory?...
Model of Use in HL7 Message or EHR
An operational example

**Shortform:** “Family history must have mood code EVN”;

**Paraphrase:** “If the code has a context of ‘person in family’ then transform it to ‘family_history_of’ subject to error if mood code not EVN and subj not subkind of Relative”;

**transform:**

\[
\text{[moodCode=M subj=R code=\{disease::D (subj_rel_ctx person_in_family::P)\}]} \rightarrow \\
<\text{family_hx (assoc_find D)}>;
\]

**bindings:**  

**error_conditions:**

- NOT EVN::M -- ‘M “illegal mood. Only EVN mood with family history”’ :: E3.5.1,

- NOT (Relative::R OR R=nil) -- “HL7 subj ” R “not compatible with SCT sub_rel_ctx ” F’ :: E3.5.2.
Decision Support / Applications

• What do applications query?
  – Who owns the complexity?
    • Does every application have to know everything?
  – “Conservation of complexity”
    • Ignoring it just pushes the problem onto somebody else

• Beware – meaning of queries vs meaning of statements
  – Meaning of queries dependent on use
    • Does the patient has “Asthma”?
      – For triggering a warning about drug contraindications
      – For establishing a diagnosis
      – For entry into a clinical trial
Testing and Migration

• Building a life cycle – is it fit for use?
  – What is the migration path?

  – Use/test cases & exemplars
  – Identifying problems – alternative solutions - exploring consequences – deciding amongst alternatives
  – Specifying solutions
    • Human and machine readable form
  – Setting conformance tests for specifications
    • Building reference implementations
  – Monitoring for problems
    • Recording of problems and changes
Key Research Questions

- **Formalising models of use**
  - UML has >6 formalisms
    Ontologies currently only have one. Why?
  - Formalising transformations

- **Formalising views**
  - A short statement analogous to DB view:
    "A view is a reified query, persistent or transcient"

- **Formalising a decision support API**
  - Formalising semantics of nested representations