openEHR An open e-health computing platform

Thomas Beale
CTO, Ocean Informatics

Forum 2013
Tromsø

What is openEHR?

Some simple questions:

Why can't existing systems keep up with changing requirements?

Why do we REPLACE clinical information systems every few years?

Why are clinical users usually left out of system design?

- Most current (clinical) information systems are not adaptive
- Their capabilities diverge continually from needs

Cost of System Replacement

- Cost of new system O(€10 – 200m)
- Costs of retraining
- Costs of data migration
  - Cost of data loss
  - Cost of data migration errors

Cost of Inability to adapt

- Docs create secret departmental Access DBs
- Other ‘peripheral’ systems added to compensate
- Free-text soaks up more content
- User frustration

Cost of not engaging clinical users

- System doesn't do what they want
- Users develop 'workarounds'

Some simple questions:

Why is there no decision support industry in health?

Why do computerised clinical guidelines remain a largely academic activity?

Historically…

- No standard way to talk to the EHR
- No standard way to query the EHR

Cost of CDS queries today

- 500 clinical guidelines
- Average of 5 queries each, some complex
  - e.g. find any BP over 160 not related to exercise
- Average of 8h professional work to design & validate a query
- ➔ 500 x 5 x 8h = 20,000h = 8 person years
- OK if we do it once…
- Not OK if we do it for each vendor product
- And that's assuming standard EHR data!

The openEHR approach

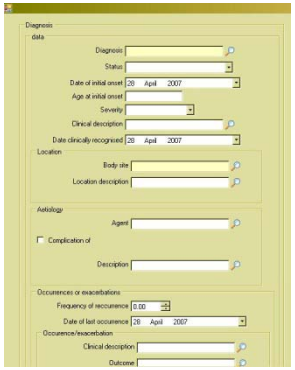
- A semantic architecture designed to flexibly standardise:
  - Data, Content structures, Terminology use, APIs
- Content developed by clinicians
- Build adaptive systems – software not dependent on content models
- Make querying based on content, not physical databases

The openEHR approach

- A Services-oriented Architecture (SOA) designed to standardise how applications talk to systems
  - APIs
- And how systems talk to each other
  - EHR Extracts
  - Generated message schemas
  - Generated document schemas

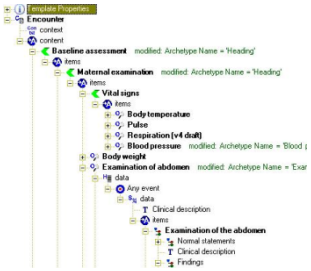
Levels of Information Semantics

Discharge summary UI form



Concrete: GUI, messages, documents

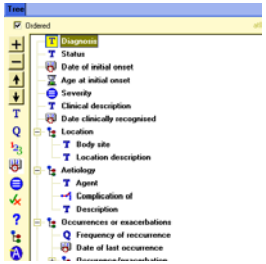
Discharge summary content model



Use-case specific data sets - Templates

Terminology Interface

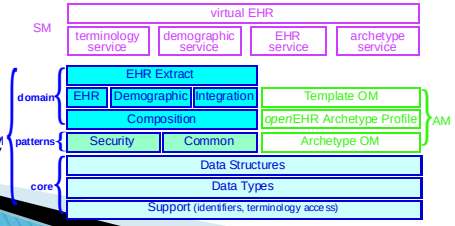
HbA1C, phys. exam, meds list, vital signs etc



Theme-based models of content – Archetypes (library of content definitions)

Querying

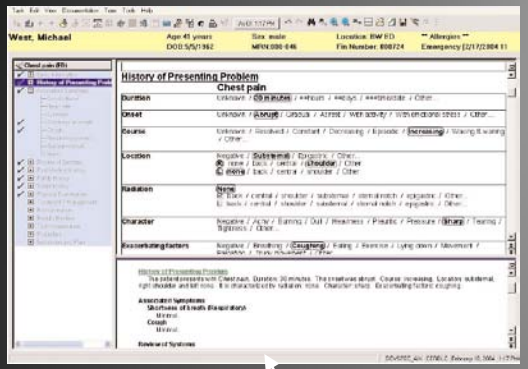
Observation, Quantity, coded text etc

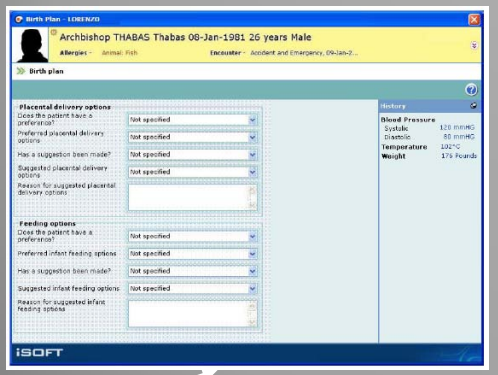


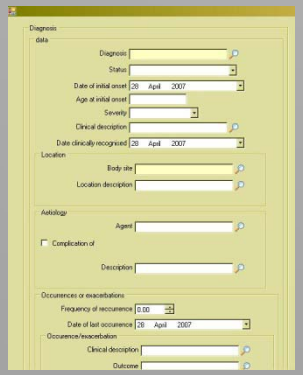
Data Representation and sharing - Reference Model

openEHR

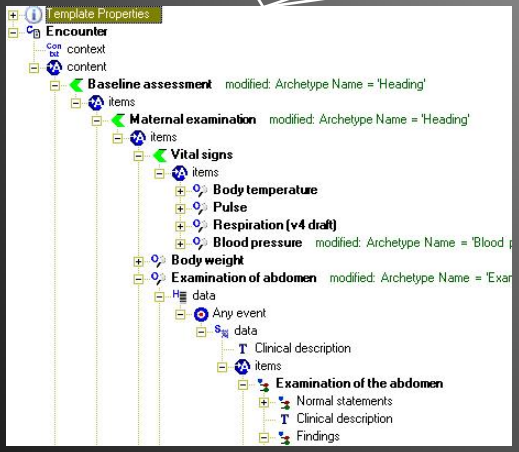
GUI & Templates







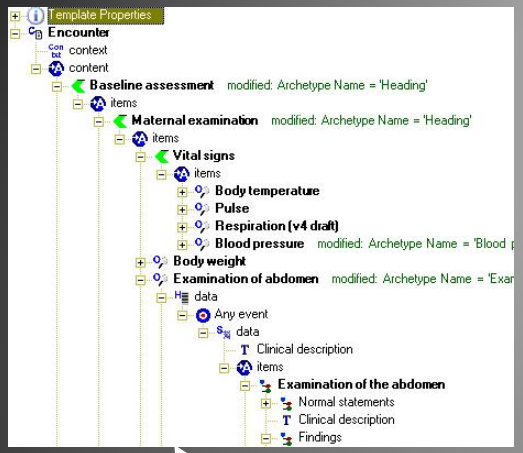
The cognitive User interface: Different ways of Presenting & Capturing the Same information

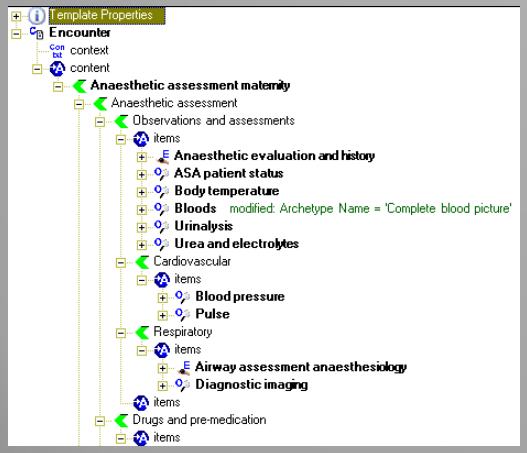


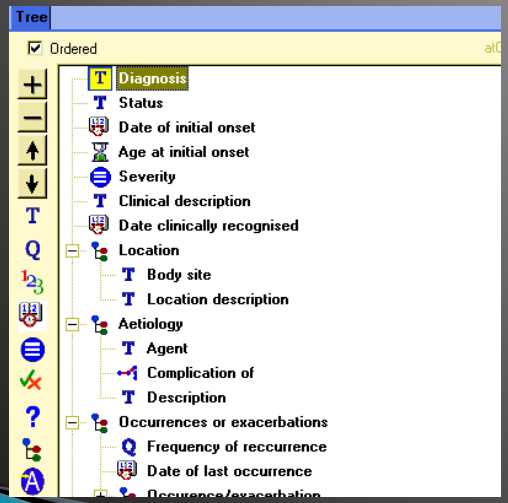
Logical data-sets: Achieved by templates That re-use and Organise underlying Standardised data Points *according to Business process event*

openEHR

Templates & Archetypes





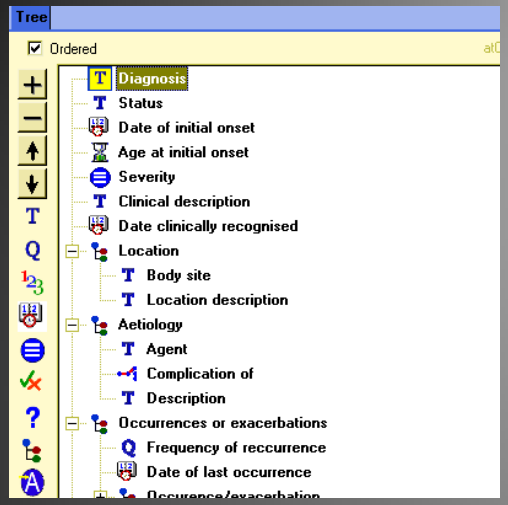


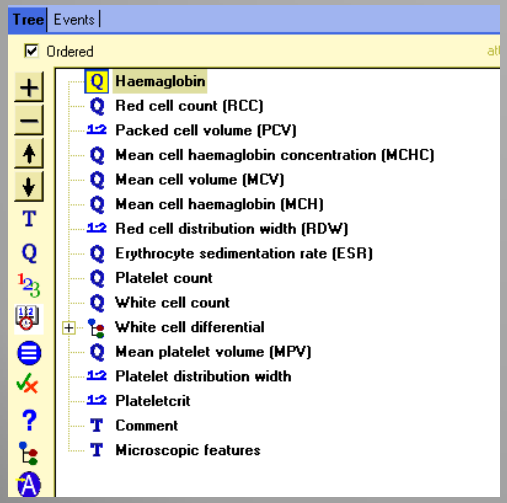
Logical data sets:
Templates – using Selected items from a Number of archetypes

Standardised models of The data:
Achieved by archetypes Organised by topic,
Independent of use

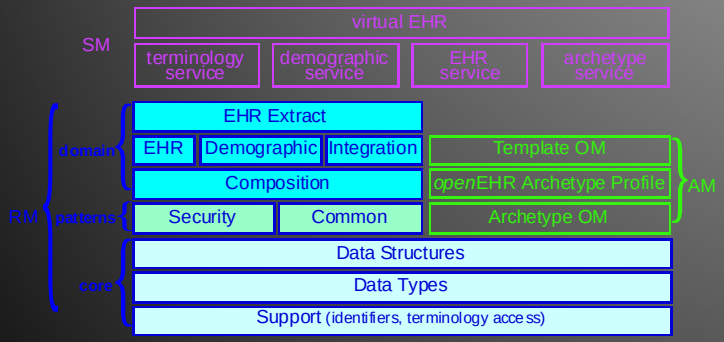
openEHR

Archetypes and Reference Model





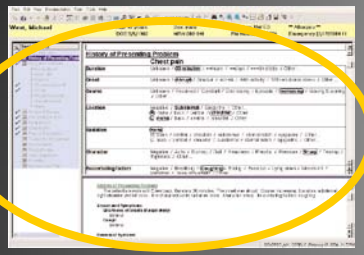
Standardised clinical models of the data:
Archetypes – all based On same reference model



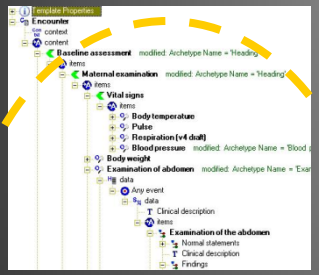
Standardised technical representation of the data:
The reference model – Enables interoperability

openEHR

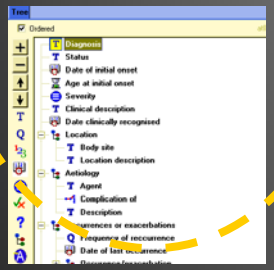
Why Current Health Information Systems don't solve the problem



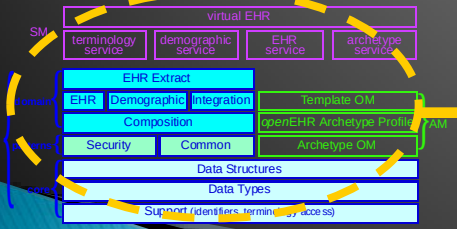
They have a form-builder



Possibly a library of 'elements'

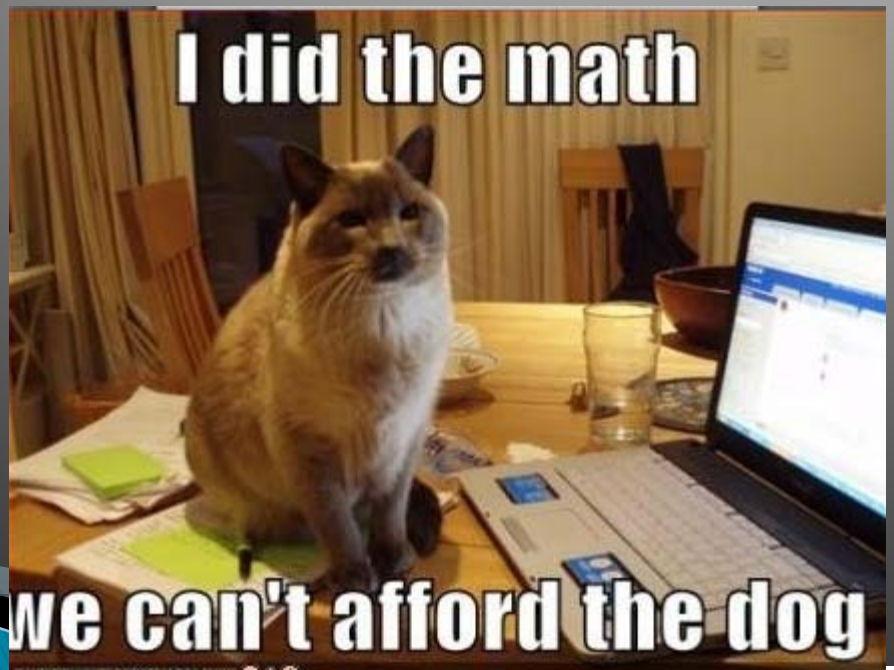


And only SQL, against
The proprietary database

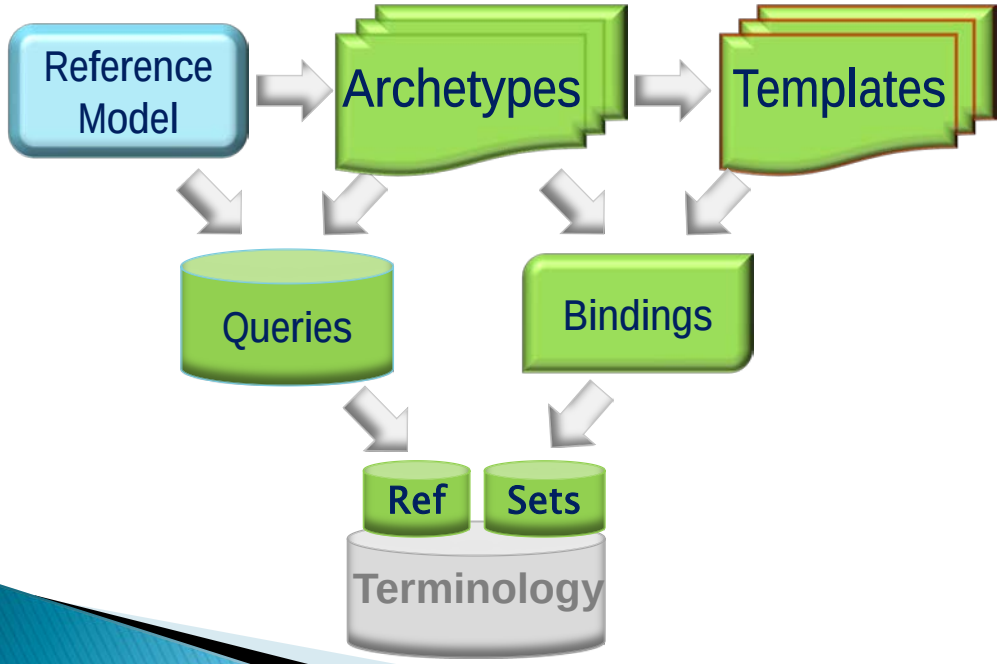


And a proprietary database

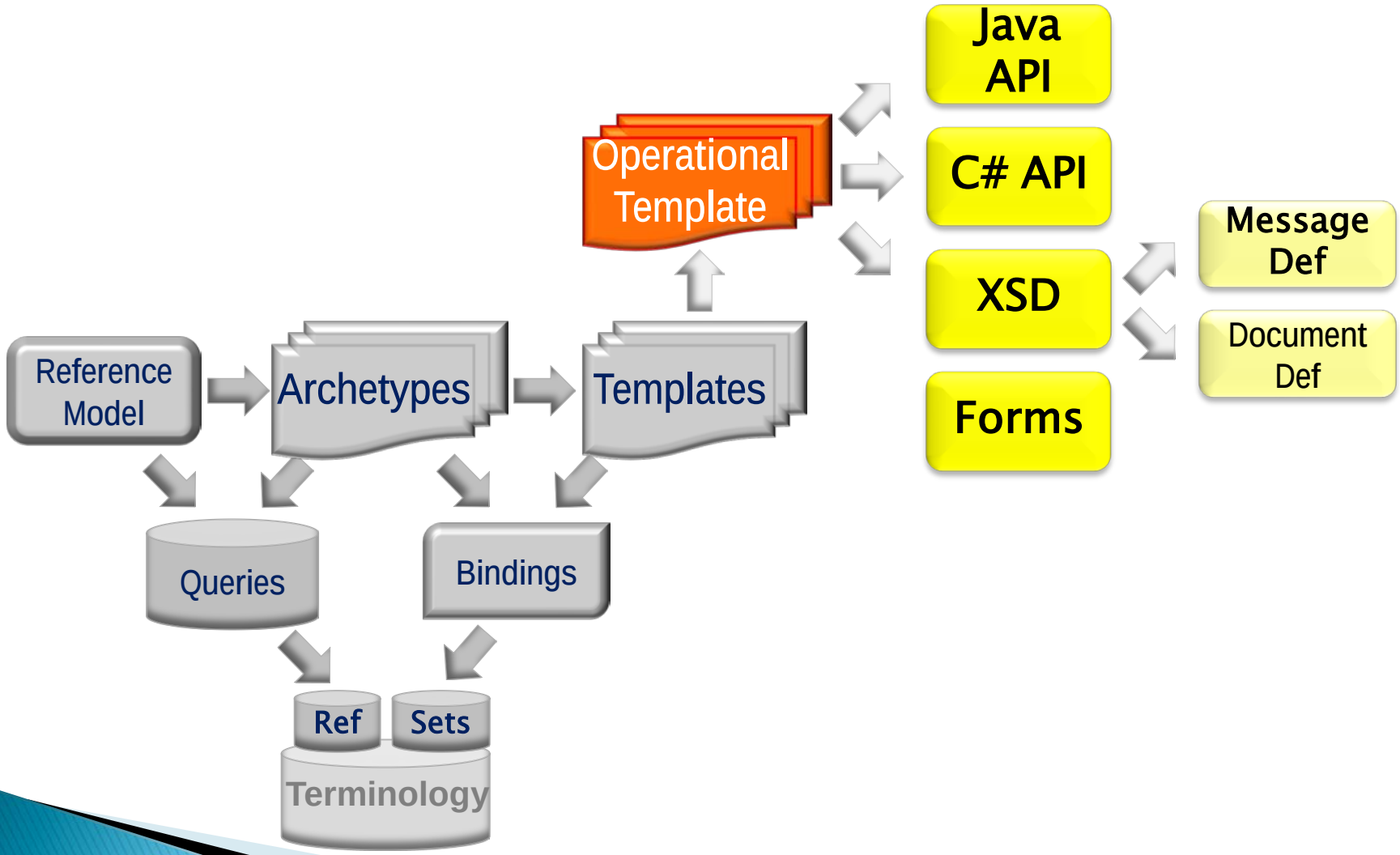
Now the seamless part...



Models...



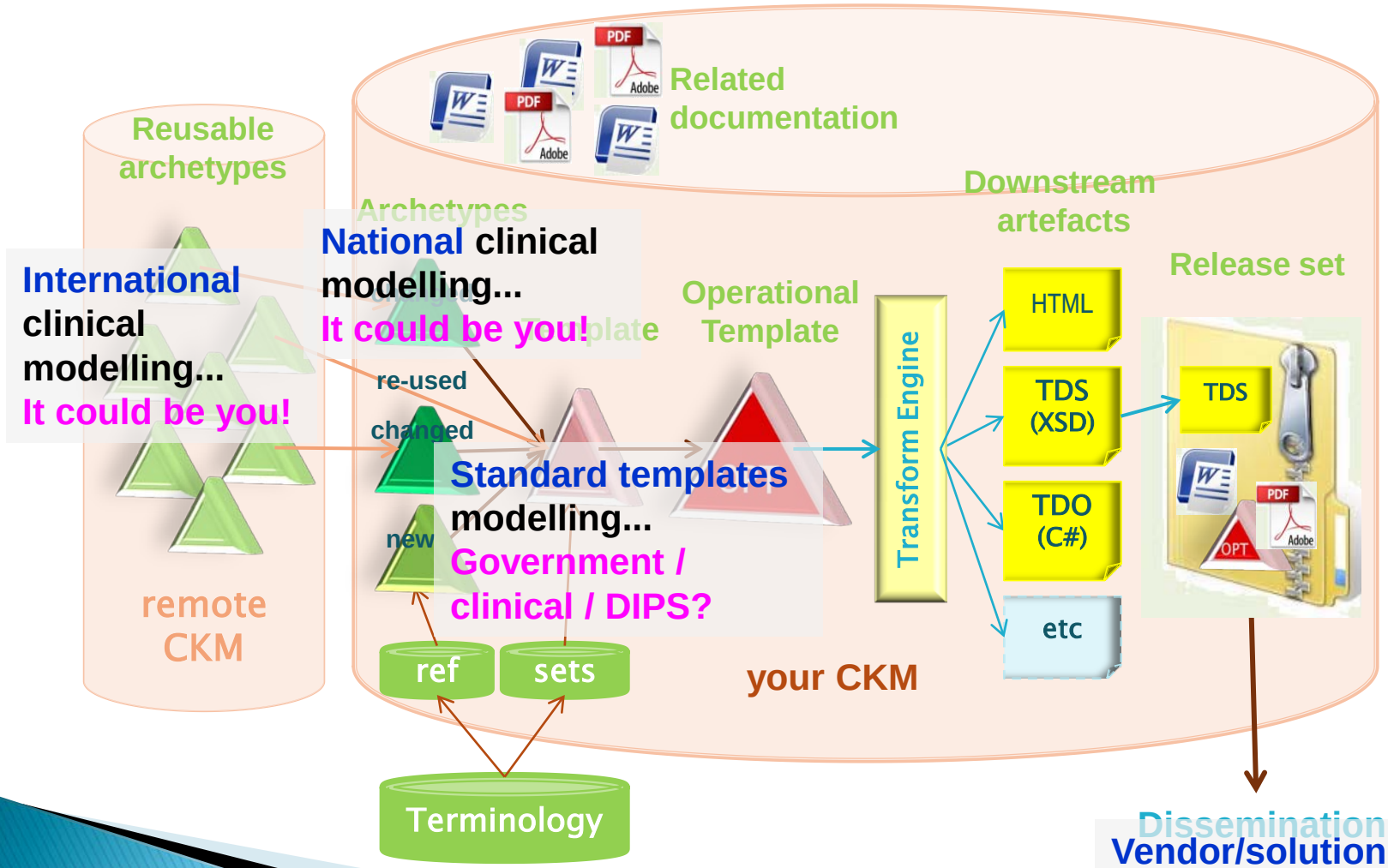
And downstream artefacts



Engaging software developers

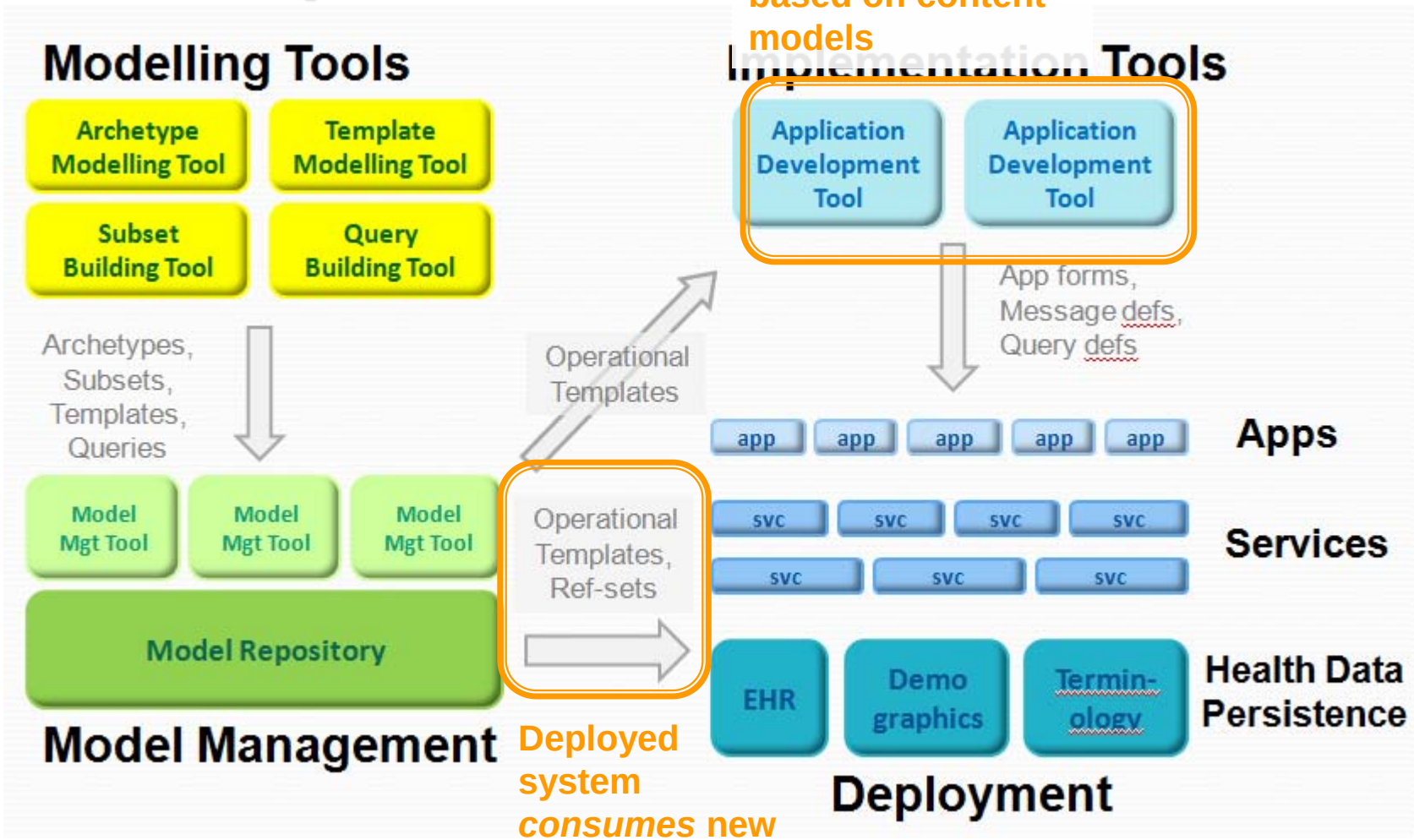
- We generate ‘normal’ downstream artefacts
  - XSDs, facade classes, HTML, GUI XML
- These can be used by typical developers
- The data they create can always be converted back to canonical form

Who does what?

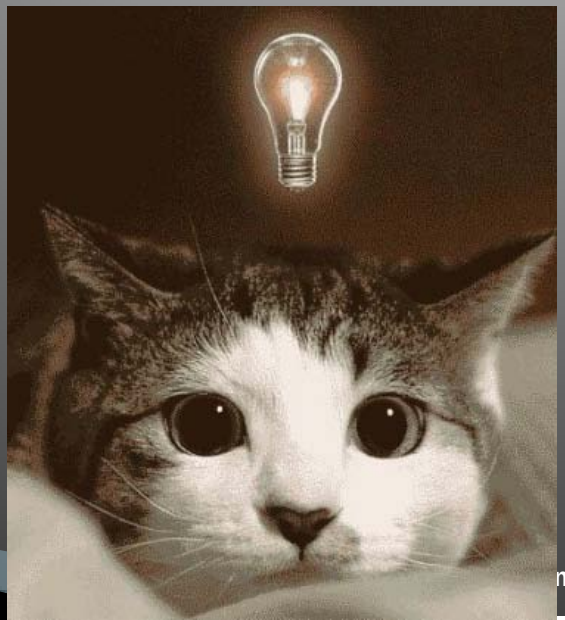


openEHR

Development Ecosystem



Query architecture

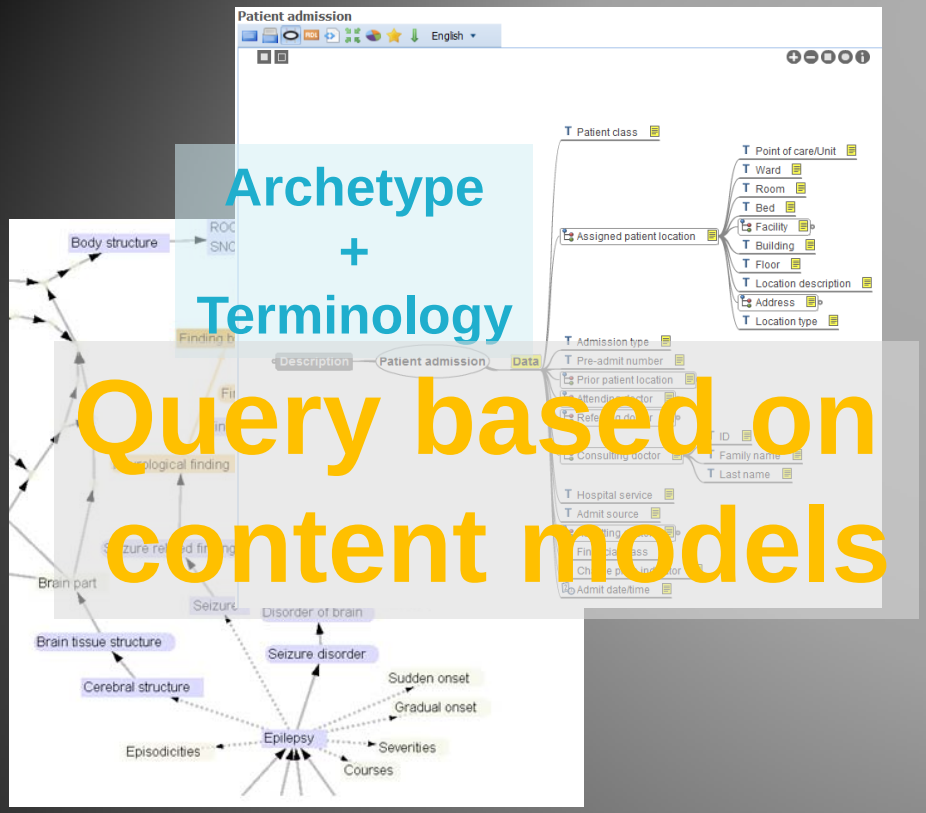


Why do we care?

- You only put data in once...
- You get it out a hundred times, in a dozen different ways

- → huge economic value in the data

- → querying matters...



Sales ID	Customer ID	Sales Date	Sales Amount
1	101	12/09/2008	10000
2	101	01/09/2008	23789
3	102	02/07/2008	45000
4	103	11/06/2008	25345

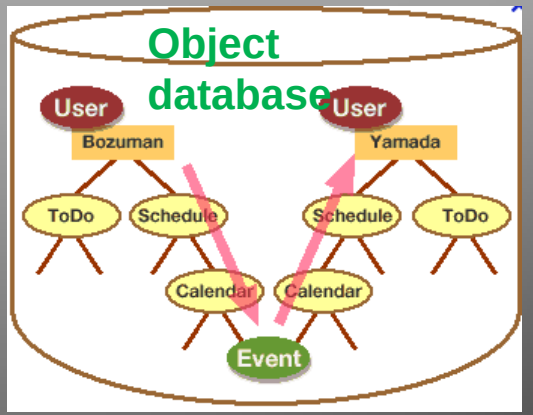
Relational database

Customer ID	Customer Code	Customer Name	Customer Address	Customer Phone
101	C00101	All sec Corp	Houston, Texas	001-325-789-321
102	C00102	John S	Chennai	0091-44-273910
103	C00103	Bridge Inc.	Delhi	0091-11-456801
104	C00104	Symphony Org	Bombay	0091-22-568902

Vs

Query based on storage model

SQL query based on physical schema of relational database DB is different!

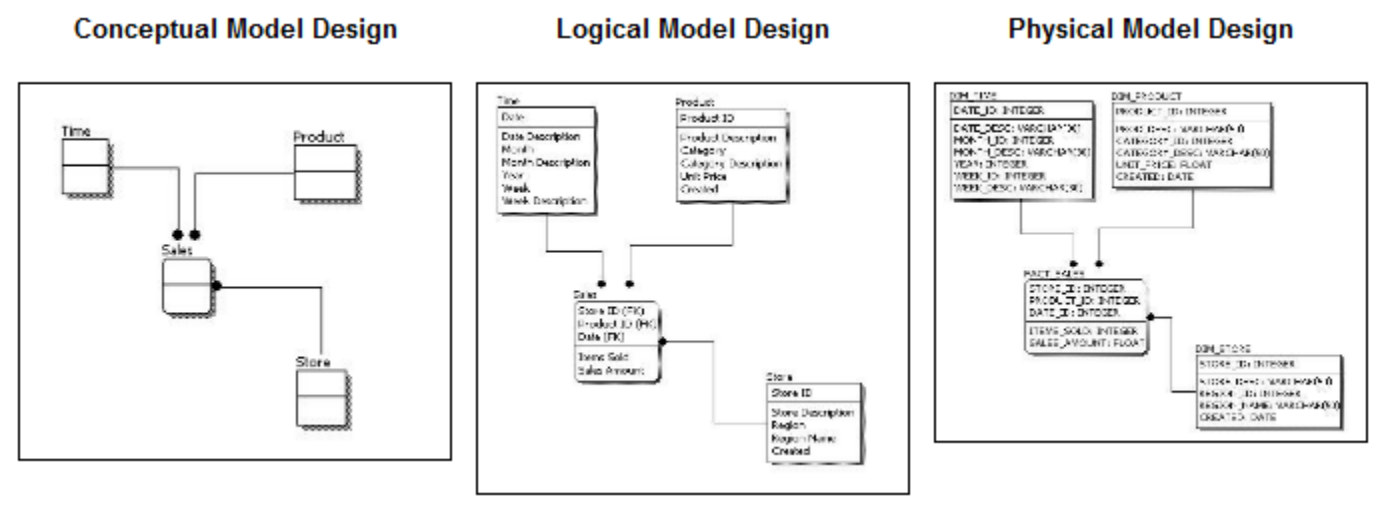


Portable query mentioning ‘data/consulting doctor/last name’

OQL query based on physical object storage model

openEHR

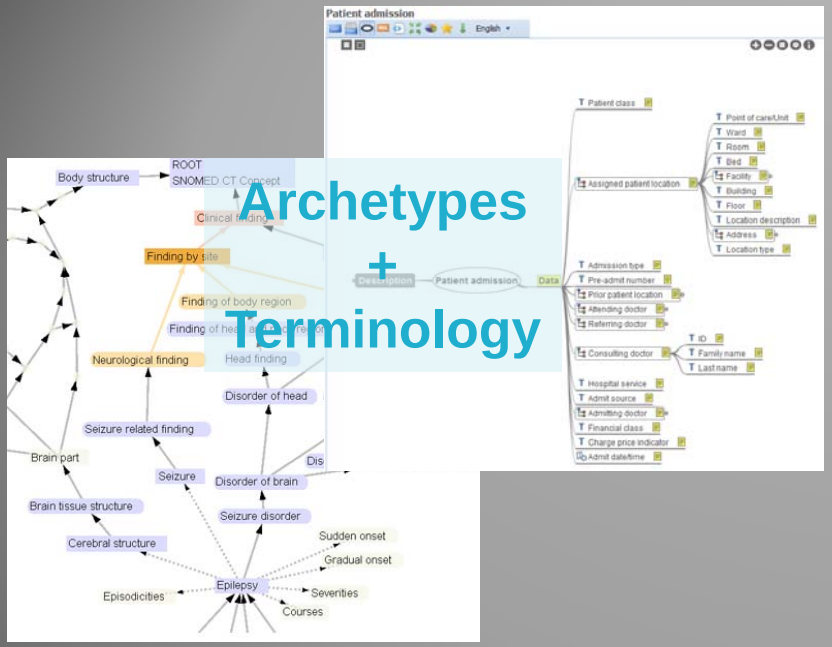
Not to be confused with:
Logical / Physical levels of relational modelling...
These are ALL models of storage



Not quite for free...

- It means the need for a *query interpreter* to process portable queries, performing any necessary conversions between the logical information model form and the physical storage form

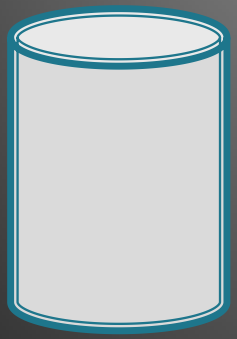
Logical



Portable query based on content model

Query Interpreter

Physical



Sales ID	Customer ID	Sales Date	Sales Amount
1	101	12/09/2008	10000
2	101	01/09/2008	23789
3	102	02/07/2008	45000
4	103	11/06/2008	25345

Relational database

Customer ID	Customer Code	Customer Name	Customer Address	Customer Phone
101	C00101	All sec Corp	Houston, Texas	001-325-789-321
102	C00102	John S	Chennai	0091-44-273910
103	C00103	Bridge Inc.	Delhi	0091-11-456801
104	C00104	Symphony Org	Bombay	0091-22-568902

SQL query based on physical schema + object in-memory data access

openEHR

Advantages

- Currently, there are ~no portable queries in health, nor in most other industries
- Preventing any meaningful business analytics or decision support industry
- Why? Because BI, CDS etc queries are complex and expensive to design. Having to replicate the work for each site database and/or vendor proprietary database is not economically feasible

The Result

It should not matter if the patient BP was captured in a GP visit, via a nurse form used in a hospital, or at home – a longitudinal query for BPs should return all the results

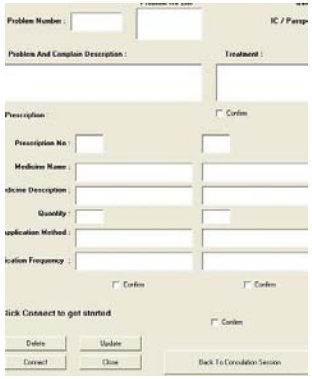
HIS ED screen

BP = 120/80

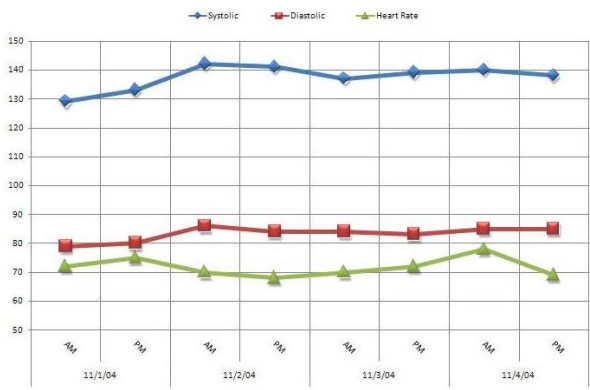
Specialist app

BP = 140/95

GP app



BP = 170/108

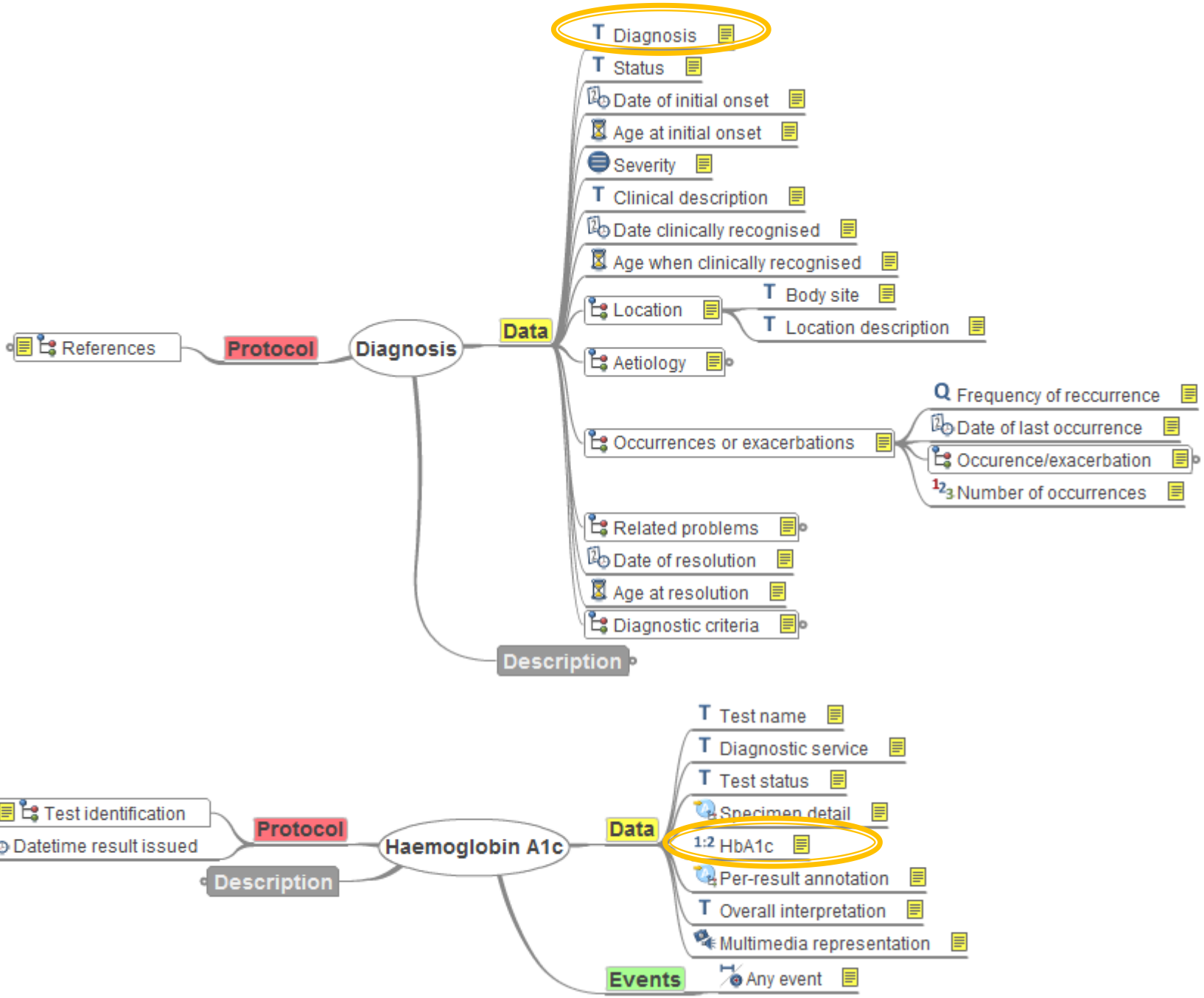


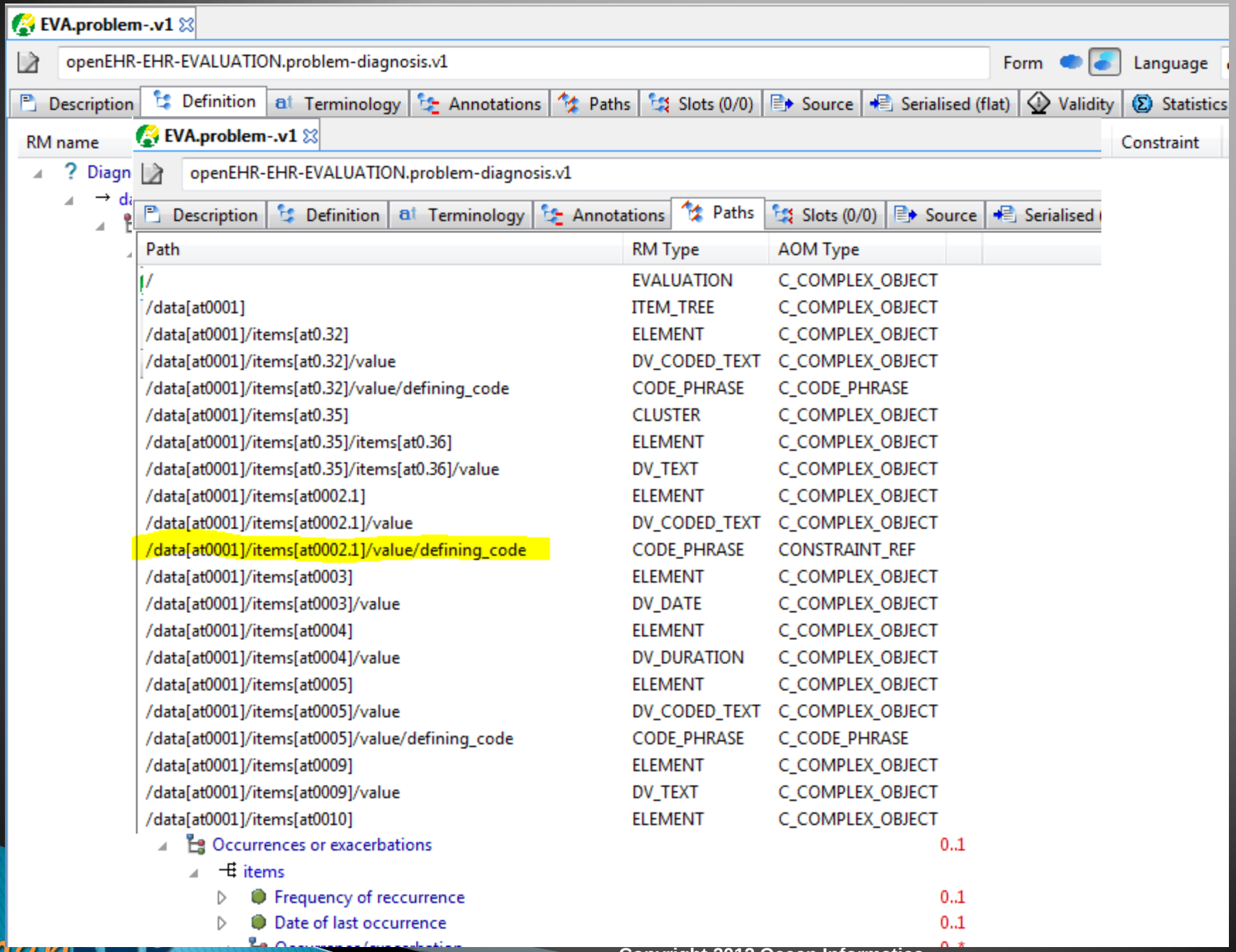
openEHR

How do we do it?

Scenario

- Get the number of patients with diabetes who have HbA1c results greater than 7.0 in last 12 months.


Diagnosis
Status
Date of initial onset
Age at initial onset
Severity
Clinical description
Date clinically recognised
Age when clinically recognised
Location
Body site
Location description
Aetiology
Occurrences or exacerbations
Frequency of reccurrence
Date of last occurrence
Occurence/exacerbation
Number of occurrences
Related problems
Date of resolution
Age at resolution
Diagnostic criteria
References
Protocol
Diagnosis
Data
Description
Test name
Diagnostic service
Test status
Specimen detail
HbA1c
Per-result annotation
Overall interpretation
Multimedia representation
Test identification
Datetime result issued
Protocol
Haemoglobin A1c
Data
Description
Events
Any event

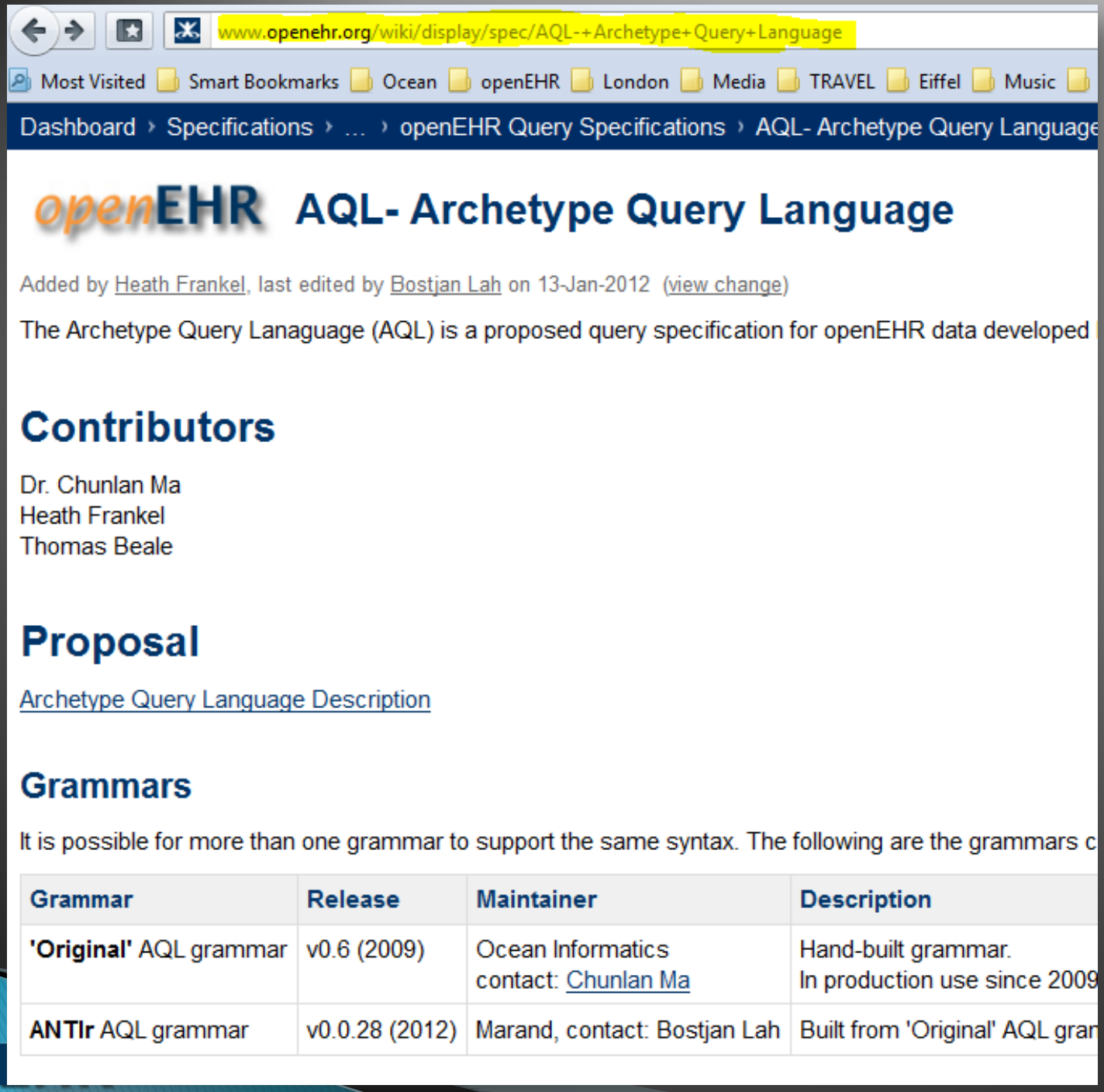


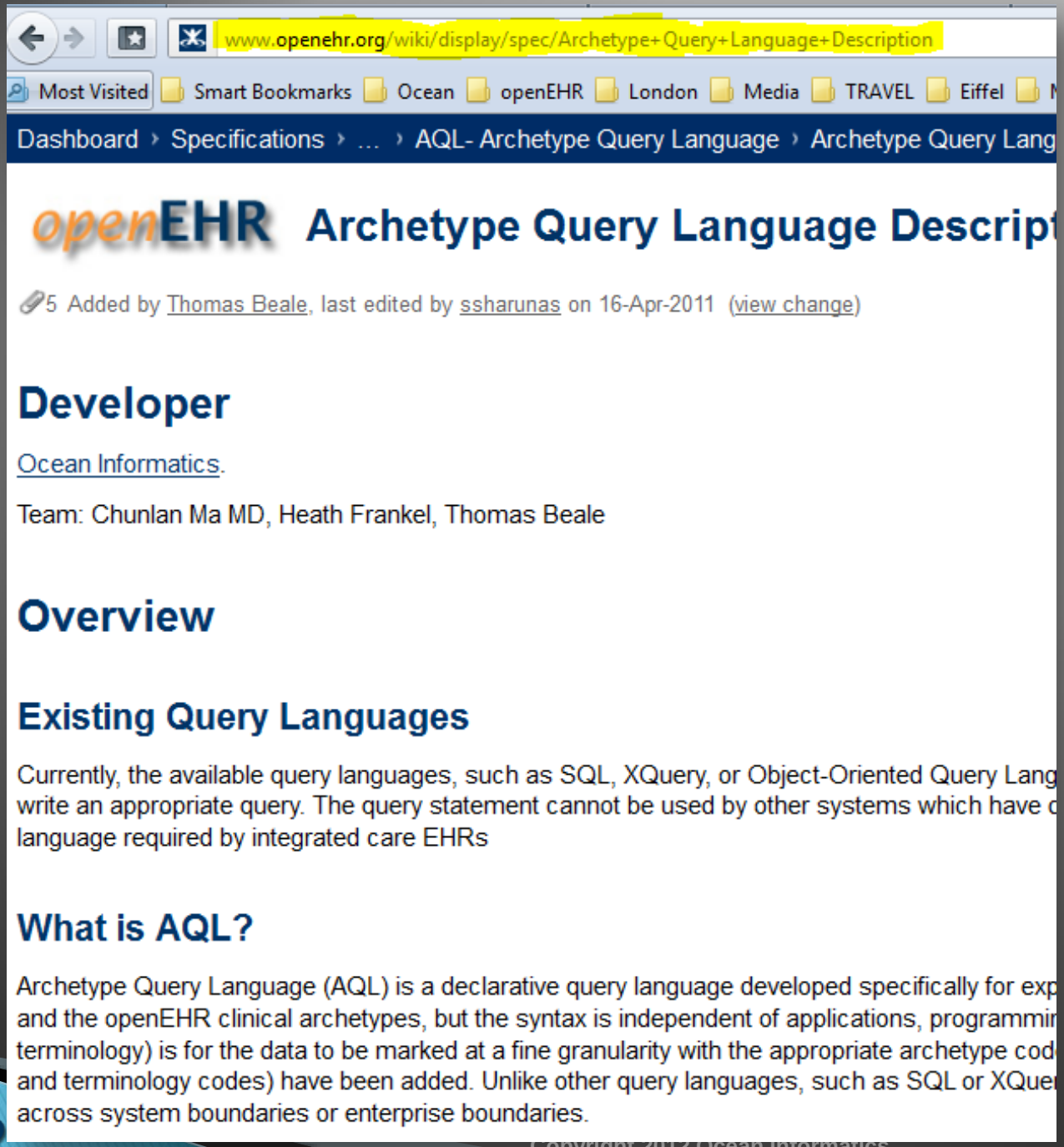
Scenario

```
SELECT COUNT(e/ehr_id)
FROM EHR e
   CONTAINS
   (COMPOSITION probs [openEHR-EHR-COMPOSITION.problem_list.v1]
   CONTAINS EVALUATION dx
         [openEHR-EHR-EVALUATION.problem-diagnosis.v1] AND
   COMPOSITION lab_rpt [openEHR-EHR-COMPOSITION.report.v1]
   CONTAINS OBSERVATION hba1c
         [openEHR-EHR-OBSERVATION.lab_test-hba1c.v1])
WHERE
   dx/data/items[at0002.1]/value/defining_code/code_string matches
         {terminology:SNOMEDCT?refset_id=1234567} AND
    lab_rpt /context/other_context/items[at0006]/items[at0013]/value >
         current-date() - P1Y AND
    hba1c /data/events[at0002]/data/items[at0013.1]/value/numerator > 7
```

- This query is based solely on the information model and the content models....
- No knowledge of the physical persistence, which can differ across vendors and even single vendor in different locations.

Querying requires a query language


www.openehr.org/wiki/display/spec/AQL-+Archetype+Query+Language
Most Visited
Smart Bookmarks
Ocean
openEHR
London
Media
TRAVEL
Eiffel
Music
Dashboard › Specifications › ... › openEHR Query Specifications › AQL- Archetype Query Language
openEHR AQL- Archetype Query Language
Added by Heath Frankel, last edited by Bostjan Lah on 13-Jan-2012 (view change)
The Archetype Query Lanaguage (AQL) is a proposed query specification for openEHR data developed
Contributors
Dr. Chunlan Ma
Heath Frankel
Thomas Beale
Proposal
Archetype Query Language Description
Grammars
It is possible for more than one grammar to support the same syntax. The following are the grammars c
Grammar
Release
Maintainer
Description
'Original' AQL grammar
v0.6 (2009)
Ocean Informatics
contact: Chunlan Ma
Hand-built grammar.
In production use since 2009
ANTlr AQL grammar
v0.0.28 (2012)
Marand, contact: Bostjan Lah
Built from 'Original' AQL gran



The big picture

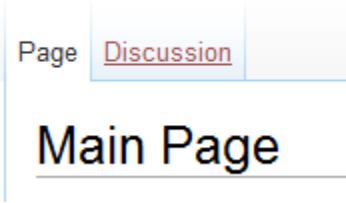
openEHR

What openEHR provides...

- A standard logical information model, describing basic data types and structural relations
  - On which quality software can be based
- Ability to build trustworthy, open content models
- Reliable, known use of terminology
- Language in which to write portable queries

Archetype formalism – ADL 1.5

- There is growing acceptance of standardised content-modelling in health using archetypes
  - CIMI taskforce, led by Dr Stan Huff (Intermountain)
  - openEHR.org archetypes
  - Other national programmes
  - New OMG Archetype Modelling Language (AML) RfP
  - VA's Model-driven Health Tools (MDHT) now moving to incorporate archetype semantics



Where is openEHR used?

Who is using openEHR?

- Healthcare Providers and Authorities
- Funded Research Projects
- Non-profit and Open Source Organisations
- Governments
- Academic Research

Large / national programmes

- Australia
- New Zealand
- UK now starting to use archetypes again
- Moscow health

What can Norway do?

- DIPS will be formally involved in openEHR Foundation as a vendor
- Norwegian clinical modelling is already happening
  - Get involved in international effort
  - Share your models
  - Some leadership probably needed going forward
- Various academic efforts

- The future is very exciting

Resources

- http://www.openEHR.org