From semantics to services – the openEHR experience
OMG SOA in HEALTHCARE CONFERENCE
13–15 JULY 2011

Thomas Beale, Washington, 13 July 2011
What are we trying to do today?
Ultimate ICT goals

• To **Compute** with health information
  • Cross-enterprise
  • Patient-centric
  • Over time & technology changes
Ultimate ICT goals

• In particular:
  • X-enterprise patient care pathway tracking
  • Decision support for doctors
  • Business process analysis for provider orgs
  • Business intelligence for payers & public health
  • Medical research ‘study’ analysis
  • Person-centred data analysis, ethically targetted marketing etc
  • Integrate health data with social & educational media streams in patient-centred portals
Ultimate ICT goals

• While dealing with relentless change in
  • Medicine, esp. drugs, interactions, procedures...
  • Information
  • Care processes
  • Patient needs
  • Legislation
Getting there requires...

- A ~change-immune **semantic architecture**, allowing
  - **meaning** of information and healthcare process steps etc to be safely and reliably defined
  - **convertability** of information from proprietary / legacy sources to common formats
Getting there requires...

- A systems and **services architecture** defining groupings and access protocols enabling:
  - **Aggregation** of information from source systems
  - Varying **levels of conformance**, esp. for existing systems
  - **Incremental deployment**
  - Satisfying **changing business** needs
Assumptions

- A services architecture with no semantic underpinning can deliver:
  - Human – human information transmission
  - Very basic search facilities
  - Limited computability, where information is already widely standardised, e.g. HL7v2 lab, ADT
  - Some security / privacy support

- But not generalised patient-centric computability – can’t access the main economic potential
The clock is ticking...

- Today we are still creating peta-bytes of non-interoperable, non-computable health data
- Post-hoc ‘re-engineering’ of the data to make it computable is too expensive to be realistic
- We know this because medical research projects regularly burn their entire budget on data re-engineering
The semantic part
Historical Industry Structure

- **Stdgs orgs + Professional bodies**: Write docs, create message specs
- **GOVs / MoHs**: ... Write data specs, minimum data sets, schemas for DS, referral etc
- **VENDOR / INTEGRATOR**: ...consume documents and msg specs
- **DOCs & Patients**: ...use systems

Developers make up what they don’t understand.
Historical Industry Structure

- Standards organisations + Professional bodies
- GOVs / MoHs
- VENDOR / INTEGRATOR
- DOCs & Patients

**Lock-in**
- Ad hoc
- Poor interoperability
- Expensive, low reuse

**Chaotic, non-computable**, Expensive

- Write docs, create message specs
- Write data specs, minimum data sets, schemas for DS, referral etc
- Consume documents and msg specs
- Make up what they don’t understand

- Proprietary form definitions
- XSD
- Manual

- PROTOCOLS
- Buy (poor) Solutions

© Thomas Beale 2011
**openEHR approach**

- **Stds orgs + Professional bodies**
  - terminology
  - archetypes
  - collaborative knowledge repository

- **GOVs / MoHs**
  - templates
  - Operational template

- **VENDOR / INTEGRATOR**
  - TOOL
  - TDO
  - XSD

- **DOCs & Patients**
  - GUI
  - APIs
  - Present’n & Services

**build archetypes & terminology that define their Information – e.g. via IHTSDO**

**...build templates and issue as standards e.g. Discharge Summary**

**...consume std templates and create their own, making OPTs**

**developers**

**build SOLUTIONS based on the platform**

**PROVIDERS**

**buy Solutions**
openEHR approach

Stds orgs + Professional bodies

GUARANTEED SEMANTICS

Collaborative knowledge repository

VENDOR / INTEGRATOR

GOVs / MoHs

DOCs & Patients

...use systems

Open, reusable

Standard tools

Interoperable

Easy Development

Cheaper

Knowledge engineering

Software engineering

build archetypes and terminology that define Information – e.g. via IHTSDO

...build templates and issue as standards e.g. Discharge Summary

...consume std templates and developers build SOLUTIONS based on the platform

PROVIDERS buy Solutions

TOOL

TOOL

TOOL

TOOL

GUI

TDO

XSD

present’n

Platform

TERMS

ARCH

... Archetypes

archetypes
templates

archetypes
templates

archetypes
templates

archetypes
templates

build archetypes And terminology that define Information - e.g. via IHTSDO

© Thomas Beale 2011
The basic plan

- A general theoretical paradigm or **framework**
- An **architecture** specific to the domain, including
  - Actual specifications for formalisms, models etc
- Actual **models**
The openEHR framework

Templates

1:N

Archetypes

1:N

Reference Model

Use-case specific data-set definitions

All possible item definitions for health

Defined connection to terminology

Terminology interface

Querying

Portable, model-based queries

Terminologies

ICPC

ICDx

Snomed CT

Defines all data

Use-case specific data-set definitions

All possible item definitions for health

 Defines all data
AQL query

- SELECT com2/context/start_time/value as START_DATE, 
  obs1/data[@at0001]/events[@at0006]/data[@at0003]/items[@at0004]/value/magnitude as SYSTOLIC, 
  obs1/data[@at0001]/events[@at0006]/data[@at0003]/items[@at0005]/value/magnitude as DIASTOLIC, 
  obs3/data[@at0002]/events[@at0003]/data[@at0001]/items[@at0004]/value/magnitude as PULSE_RATE, 
  obs4/data[@at0001]/events[@at0002]/data[@at0003]/items[@at0004]/value/magnitude as RESPIRATORY_RATE.

- FROM EHR e[ehr_id/value='f271cd26-23fc-43a1-b411-34cdadaea067'] CONTAINS 
  COMPOSITION com2 [openEHR-EHR-COMPOSITION.encounter.v1] 
  CONTAINS (OBSERVATION obs3 [openEHR-EHR-OBSERVATION.heart_rate-pulse-zn.v1] OR OBSERVATION obs1 [openEHR-EHR-OBSERVATION.blood_pressure-zn.v1] 
  OR OBSERVATION obs4 [openEHR-EHR-OBSERVATION.respiration.v1] .... 

- WHERE com2/name/value matches {'Vital functions', 'Respiratory assessment', 'Assessment scales'} AND obs9/name/value = 'PEF before' AND obs10/name/value = 'PEF after' AND com2/context/start_time >= '20110406T000000.000+0200' AND 
  com2/context/start_time < '30000101T000000.000+0100'
Properties - software

Generated software
- Templates
- Archetypes
- Reference Model

Developed software
- Terminology interface
- Terminologies
  - ICPC
  - ICDx
  - Snomed CT

Generate

Consume

Software core

© Thomas Beale 2011
The architecture

Reference Model

Int'l archetypes  Nat'l / local archetypes  Nat'l / local templates

terminology

All data = same information model

canonical openEHR

Querying

java  C#  etc

Template-based artefacts

Msg XSD  Doc XSD  GUI XML

data

© Thomas Beale 2011
The key...

- Is the operational template (OPT) – this is the joining point between the semantic specifications and deployable software artefacts that can be used by normal developers.

Evaluate inclusions, flatten constraint inheritance
Key Outcomes

- Normal developers can engage – openEHR + Snomed become economic and ~quick
- Semantic connection exists between definitions and implementations
  - now we know what the meaning of data are, and DS and BI can work...
The openEHR framework

Templates

Archetypes

Reference Model

1:N

Terminology interface

Querying

Terminologies

ICPC

ICDx

SNOMED CT
The reference model

Will continue to grow, to accommodate process, workflow etc

- EHR
- EHR Extract
- Demographics
- Composition
- Party
- Security
- Participations
- Audit
- Versioning
- Data Structures
- Data Types
- Primitive types, Ids

http://www.openehr.org/releases/1.0.2/roadmap.html

© Thomas Beale 2011
The openEHR framework

- Templates
- Archetypes
- Reference Model

Terminologies:
- SNOMED CT
- ICDx
- ICPC
The archetype architecture

- Downstream software artefact transformations
- Template model & serialisations
  - Archetype Query Language (AQL)
  - Archetype Object Model (AOM)
  - Archetype Def. Language (ADL)
- Data Types
- Primitive types, Ids

http://www.openehr.org/releases/1.0.2/roadmap.html
Managing knowledge artefacts

- Content models & terminology ref sets managed outside of software process & people
- Needs:
  - Governance
  - Methodology
  - Identification
  - Sharing and release rules
  - etc
Clinical Knowledge Manager

- A tool for involving clinicians in defining clinical content based on archetypes, templates, and termsets
CKM Core Principles

Separation of technical and clinical aspects to successfully involve clinicians in

- Informal Discussions
- Formal Reviewing (content, terminology binding, translations)
- Sharing
- Publishing
- Revision/Version Management
- Release and Dependency Management
CKM Approach

- Web 2.0 design
  - Easier to engage clinicians: Can now use 5 mins or 1 hour of an expensive specialised clinician's time; before, they lost hours on physical meetings
- Implementation is growing as we learn
  - Can respond quickly to changing needs, evolving methodology
- More than a tool
  - Engage and manage the community
CKM Users

- International openEHR CKM instance
  - > 630 users
  - From 64 countries
- National programs with an instance of CKM
  - Australia: Nehta
  - Sweden: SKL
Key Messages

- Knowledge Management is crucial
  - High-quality archetypes with high-quality clinical content
  - Semantically interchangeable between clinical systems; also the basis for decision support
  - Key to success: how to engage with clinicians and capture their knowledge

- CKM - [http://www.openehr.org/knowledge](http://www.openehr.org/knowledge)
Key Outcomes

- We can now start to situate existing standards in the framework
  - Concrete content-specific standards like HL7 message definitions, CDAs, CCRs etc are DOWNSTREAM generations and/or mappings of operational templates
  - Meaning we can connect them into a semantic framework and potentially guarantee their semantics
  - Rather than manually building them in a standalone fashion
Tool-based standards

Reference Model

OPT

java
C#
etc

Msg XSD
Doc XSD

13606 XSD

13606 XSD

data

Querying

HL7v2
CDA XSD
epSOS XSD

© Thomas Beale 2011
The Services Part
General approach

- Build from bottom up – ‘Native’ services
  - Needed services, consistent with information and model artefacts
- And from top-down – ‘Standard’ services
  - Connect IHE, HSSP etc to native services
Services

Querying data

Reference Model

terminology

13606 XSD

OPT

java
C#
etc

Msg XSD

Doc XSD

GUI XML

HL7v2

CDA XSD

epSOS XSD

© Thomas Beale 2011
On Services...

- Some architectural inspiration from Microsoft Connected Health Framework (CHF)
- Native layer
  - Take as much of IHE, HSSP etc interfaces & adapt for native compatibility, making archetype-aware
- Standards layer
  - Service – service connection
Microsoft CHF

- ProcessModels
- Service Models
- Information Models
EHR Service

Workflow & Pathway Services

Service Interface Components

- Patient Identity & Health Status
- Patient Consents
- Patient Records
- Patient Management
- Assessments & Care Plans
- Population Screening Groups
- Orders & Results
- Waiting Lists & Schedules
- Clinical Processes
- Health Care Providers
- Professional Roles & Teams
- Professional Permissions
- Medication
- Care Pathways
- Healthcare Classifications
- Clinical Data Management
- Rules Engine
- Healthcare Knowledge Management
- Clinical Decision Support

Business Components

demographics

Security & privacy

© Thomas Beale 2011
The openEHR services architecture

http://www.openehr.org/wiki/display/spec/openEHR+Service+Model

© Thomas Beale 2011
Native services

- All services archetype/template aware
- Query service based on AQL / a-path
  - No SQL queries against physical database!!!
Key data services - EHR

- ARCHETYPE-AWARE
- Virtual EHR – fine-grained creation, modification, retrieval, querying
- EHR back-end – coarse-grained DVCS-like interface – ‘change-set’ based
- EHR audit log
Key data services - patient

- Demographics – ARCHETYPE-AWARE
  - Authentication info
  - Patient relationships
  - HCP relationships – teams etc

- EHR subject X-ref service
  - openEHR EHRs are identified by EHR id only
  - Deals with merged & split EHRs, i.e. 2 subject ids → 1 EHR, 2 EHR ids → 1 subject
  - Enables dynamic distribution of EHRs
  - Becoming the EHR meta-data service
Key knowledge services

- Archetypes & templates
- Terminology ref-sets
- Terminology service
  - access
  - Inferencing
  - Terminology administration
- Medications, devices
- Allergies & interactions database
Key process services

- Event-based notifications
- Care pathway
  - Based on archetyped openEHR structures
- Booking / appointments
  - Requires access to patient requests & doctor’s diary & other resource availability data
- Doctor’s diary
  - Forces syncing of appointments to filler
- Patient diary
  - Allows multi-function visits
Key elements that MUST WORK

- Standardised **querying** of data, based on knowledge artefacts, not physical DB
  - standardised knowledge artefact identification, including versions
  - standardised ability to designate finest grain items in the data
- Enabling URI to any data item
Key elements that MUST WORK

- Everything in openEHR relies on **archetype paths**, which are X-path compatible
- The two tests are being able to:
  - Write portable queries
  - Create URIs to finest grain of data
AQL query

- SELECT com2/context/start_time/value as START_DATE,
  obs1/data[at0001]/events[at0006]/data[at0003]/items[at0004]/value/magnitude as SYSTOLIC,
  obs1/data[at0001]/events[at0006]/data[at0003]/items[at0005]/value/magnitude as DIASTOLIC,
  obs3/data[at0002]/events[at0003]/data[at0001]/items[at0004]/value/magnitude as PULSE_RATE,
  obs4/data[at0001]/events[at0002]/data[at0003]/items[at0004]/value/magnitude as RESPIRATORY_RATE....

- FROM EHR e[ehr_id/value='f271cd26-23fc-43a1-b411-34cdadadaea067'] CONTAINS COMPOSITION com2 [openEHR-EHR-COMPOSITION.encounter.v1]


- WHERE com2/name/value matches {'Vital functions', 'Respiratory assessment', 'Assessment scales'} AND obs9/name/value = 'PEF before' AND obs10/name/value = 'PEF after' AND com2/context/start_time >= '20110406T000000.000+0200' AND com2/context/start_time < '20110406T000000.000+0200'
openEHR URI

- ehr:1234567/87284370-2D4B-4e3d-A3F3-F303D2F4F34B@latest_trunk_version/content[openEHR-EHR-SECTION.vital_signs.v1]/items[openEHR-EHR-OBSERVATION.heart_rate-pulse.v1]/data/events[at0006, 'any event']/data/items[at0004]
Where paths come from
Conclusions
Basic premise

- If we want to share and compute with health data at any level of detail, we need a knowledge-based architecture
Architecture

Semantic underpinning

Connect to standards

Querying

Model-based querying

Knowledge-enabled services

openEHR

© Thomas Beale 2011
Services lessons

- Need native and standard layers, i.e.
  - Inside-out, outside-in
- Knowledge-based architecture brings new needs:
  - New knowledge services – archetypes, ref-sets
  - Other services must be knowledge-aware
- Business services need to be small, with changeable interfaces
Knowledge awareness means...

- Service layer understands:
  - knowledge artefact identification system
  - Fine-grained data item identification
- Which means we need standardised knowledge models
- Aka DCMs
The ultimate test

- If you can create a URI to ‘my instantaneous resting, lying down systolic BP, recorded 7/Jan/2011’ then you can communicate at any level of detail about health information