

# openEHR Specifications Strategic Directions 2008

---

Having established a solid set of baseline specifications, the *openEHR* Foundation and community aims to build on these in order to provide more direct value to implementers, clinicians and users of health information systems. Community and vendor involvement is encouraged in all the activities described below. This will be enabled by the presence of the recently added *openEHR* [wiki](#)<sup>1</sup> and [issue tracking](#)<sup>2</sup> systems. The following describes the goals proposed for 2008. See [here](#)<sup>3</sup> for the specifications delivery calendar.

## New and Enhanced Specifications

### Template Model

*openEHR* Templates have now been in use in the NHS for over a year, providing valuable implementation experience. This will be used to inform the development of open specifications and schemas for templates, including additions to the ADL language supporting specialisation and templating; the Template Definition Model (TDM; an object model of template definitions) and a Template Definition Model Schema (.xsd of the TDM).

### *openEHR* / Snomed CT integration

Active work is going on in *openEHR* and within UK NHS Connecting for Health to define technical binding approaches for Snomed CT and *openEHR* archetypes and templates. This work is aimed at solving a number of challenges, including how to determine the correct set of attributes when designing coded data points in an archetype and how to map a post-coordinated code expression to a number of data points in an archetype.

A terminology subsetting language created by Ocean Informatics for its terminology server is being offered to IHTSDO as a basis for an open language used to define dynamic subsets. *openEHR* will adopt the result of this collaboration with IHTSDO as its terminology subset language, which will enable the creation of reusable subsets of Snomed CT and other terminologies.

### Virtual EHR (vEHR) and EHR Service Interfaces

A number of interface definitions are available from industry for two important EHR-related interfaces. The vEHR API is a virtual EHR interface as might be used in a middleware component, and offers an integrated programming interface to back-end services for a clinical application. It provides access to the EHR, demographics, archetypes, templates, terminology, querying and identity and security features.

The EHR Service Interface provides a coarse-grained interface to an EHR service, allowing access to EHR Compositions and other top-level EHR objects, as well as querying.

### Care Pathway Support

With many base specifications in place and stable in *openEHR*, new work in 2008 will concentrate on

- 
1. <http://www.openehr.org/wiki/dashboard.action>
  2. <http://www.openehr.org/issues/browse/SPEC>
  3. daisy:233-OE (*openEHR* Specifications Roadmap 2008)

supporting distributed care pathways. This will provide a standardised way of finding and merging distributed medication orders and statuses, as well as enable the generation of a map of referrals, admissions, discharges and other transfer events.

## Data Retrieval and Querying

There is a growing emphasis on querying and data retrieval. *openEHR* has been designed from the outset to ensure that data is queryable based on archetype paths. Archetype paths act like dependable identifiers for data values, while also providing direct implementation support, due to being Xpath-compatible. A new querying language, Archetype Query Language (AQL) has already been [described](#)<sup>4</sup> in MedInfo 2007 (under its working name of EQL), and is already in extensive use in one *openEHR* implementation. Further development is planned in order to add the capability of terminology subset-based querying, so as to support proper inferential queries.

## Implementation

### Operational Templates

An Operational Template in *openEHR* is generated by evaluating a Template Definition along with the archetypes and terminology it references, to produce a single resulting 'template', which is the equivalent of a single large archetype. This is used directly at runtime in *openEHR* systems as well as being the precursor for data capture forms (including using various XML formalisms such as XAML, XForms), and as the input of Template Data Schemas (see below).

### Template Data Schema (TDS)

The concept of the Template Data Schema is new in *openEHR*, and provides a greatly improved capability for integration. The standard approach to *openEHR* templates is to express them in a generic formalism, which has an associated generic XML schema. However with a single transform, a TDS can be generated for each template. The resulting schema is hard-wired to the template's contents, and suited for XML data transformation and communication, for example as a message. This allows data sources such as pathology laboratories to generate their content according to schemas directly describing their result types, without having to understand *openEHR*. The same logic follows for any kind of content. The TDS approach holds great promise for integration because any data that conforms to TDS .xsd in the standard XML fashion is guaranteed to be convertible to standard *openEHR* content format. The capability of producing TDSs from templates is effectively a facility for machine-generation of message definitions, replacing previous manual message definition approaches.

### Template and Schema for the ASTM Continuity of Care Record (CCR) and HL7 CCD

A template definition and schema will be created for the ASTM CCR, showing how convenient this is to model and use in *openEHR* systems.

---

4. daisy:61-OE (Publications on Archetypes)