Virtual Medical Record (VMR)

The virtual medical record is an object model of healthcare data suitable for use in standards based clinical decision support. It is designed to be used with any source data that contains the relevant data items. It is inspired by existing HL7 V3 standards but is designed to be usable with source data drawn from non HL7 based sources.

The VMR can be represented in a number of formats and here is represented by both UML Models and GELLO class definitions. These two representations are interchangeable at a semantic level and are simply different representations of the same object models. The models produced by this process can then be used by any execution language.

There are four broad groups of classes in the VMR Project:

1. Core Context classes
2. Simplified Patient Summary Classes
3. Access classes for existing HL7 Models and other templating formalisms
4. Data type classes (based on ISO 21090 standard and standard UML/OCL/GELLO types)

Understanding these four groups of classes is critical to understanding the VMR structure and the characteristic are explained below:

1. Core Context Classes
   1. When patient data is evaluated for the purposes of clinical decision support it may be done in a number of contexts. The data may relate to a single patient, it could be about all the patients in a practice or even all patients in a state. The evaluation may be occurring as part of a decision formalism (such as GLIF) or as part of an eMeasure evaluation. It is also possible that the evaluation is occurring in the process of creating a patient summary or lab test.
   2. These contexts of evaluation have specific properties that are represented by a “Context Class” and the evaluation occurs in the scope of this class. The default context is that of a single patient where only data relating to the focus patient is available. Context classes also have methods that are relevant to evaluation context. eg during evaluation of an eMeasure named local variable exist and an access method is provided to retrieve this data. During the creation of a single lab test the context class would provide access to the relevant order information for that test. While several context classes are defined, the VMR can be extended locally to provide new context classes, as long as the extended context descends from an existing context class.
   3. The context class also provides important environment information such as what terminology the patient data uses to encode values and what date time range of data is available during the evaluation. It also provides for mechanisms to access the other classes in the VMR and add new model definitions to the dynamic model system.
2. Simplified Patient Summary Classes
   1. These classes were modelled on the HL7 Green CCD specification and provide a simple patient data model suitable for a large percentage of decision support needs. They do not try and represent every requirement but provide a simplified model that can be created from data residing in existing systems and instances of a CCD document.
3. Access classes for existing Models
   1. When the simple Patient Summary classes do not provide enough detail these classes provide access to the full HL7 RMIM models, or any other templating formalism that can be transformed into a class model. These classes can be plugged into the VMR dynamically depending on the availability of data locally. The VMR defines a grammar for defining these access class definitions. This grammar is also used for defining other parts of the VMR and can be transformed into xmi for use in UML models.
   2. The HL7 Pedigree Model is the prime example of a model that may be available for evaluation and this model would be available in its entirety by providing the class definitions derived from the standard xsd definition.
4. Data type classes
   1. The VMR uses a simplified profile of the ISO 21090 datatypes and the base classes present in GELLO, which is a superset of the OCL/UML datatypes.
   2. Model specific datatypes may be present in access classes for existing models, but where possible they are mapped to ISO Datatypes.

Work in progress…..