

# eVIP Profile

## Introduction

The eVIP profile is an application profile of the MedBiquitous Virtual Patient (MVP) data specification. An application profile accepts the formats and rules of the base specification and augments them with additional requirements such as controlled lists and vocabularies, file formats and metadata elements.

The eVIP profile consists of:

- Schemas that defines how to encode Virtual Patients using xml.
- A specification of how to package the Virtual Patient.
- Requirements for conformance with the eVIP profile.

Review of the eVIP profile will be performed on a regular basis during the whole duration of the project.

## Schemas

The schemas are based on the Medbiquitous Virtual Patient specification version 0.47 (2008-03-19)

A set of UML class diagrams summarize the VPD, DAM, and AM schemas and their relations (umlvpd\_dam\_am\_v3.pdf).

See enclosed xsd files for details regarding how to encode the Virtual Patient content.

- Virtual Patient Data: virtualpatientdata.xsd
- Data Availability Model: dataavailabilitymodel.xsd
- Activity Model: activitymodel.xsd

## Package

The Virtual Patient is packaged as a folder and distributed as a file-archive in a standard zip-format.

Folder structure:ROOT

*MVP+* - activitymodel.xsd

+ - activitymodel.xml

+ - dataavailabilitymodel.xsd

+ - dataavailabilitymodel.xml

+ - virtualpatientdata.xsd

+ - virtualpatientdata.xml

+ /media

+ - image1.jpg

+ - image2.jpg

*Scorm+* index.html

+ player.swf

+ adlcp\_v1p3.xsd

+ imscp\_v1p1.xsd

+ imsmanifest.xml

+ datatypes.dtd

+ xml.xsd

+ XMLSchema.dtd

+ /address

+ address.xsd

+ /common

+ anyElement.xsd

+ dataTypes.xsd

+ elementNames.xsd

+ elementTypes.xsd

+ rootElement.xsd

+ vocabTypes.xsd

+ vocabValues.xsd

+ /extend

+ custom.xsd

+ /unique

+ strict.xsd

+ /vocab

+ custom.xsd

*Metadata+* metadata.xml

+ healthcarelom.xsd

+ /healthcare

+ healthcaremetadata.xsd

+ healthcarevocabularies.xsd

XHTML+ xhtml1-strict.xsd

## Conformance Metrics

Conformance testing, sometimes also called compliance testing, is the process of verifying whether a VP system meets the eVIP profile standard specifications. In the process, conformance testing can identify bugs in the import/export tool, and once those bugs are eliminated, conformance testing can verify that the fixes were successful and within the applicable standards. Level 3 conformance is the aimed level for the eVIP project. Experimental efforts to target level 4 conformance are also the scope of the project.

## **Levels of conformance**

### **Level 1 - Package validation**

The archive structure and content conforms with the eViP profile specifications

### **Level 2 - XML\*/XSD \*validation**

The XML files are valid and well-formed relative to their schemas

### **Level 3 - Import validation**

Import of a VP will unpack relatively meaningfully in a level 3 compliant player/authoring system.

### **Level 4 - Runtime validation**

An imported VP will run as intended and in a meaningful way.