Model Driven Health Tools

Design and Implementation of CDA Templates

https://mdht.projects.openhealthtools.org/cda

Dave Carlson
Contractor to CHIO
David.Carlson@va.gov

October 29, 2010

Agenda

- **MDHT tooling for designing and publishing specifications (30 minutes)**
  - Dave Carlson, VHA
- **MDHT generated runtime APIs for reference implementation (15 minutes)**
  - John Timm, IBM Research
- **MDHT in Mirth Results CDAPI (15 minutes)**
  - Jon Bartels, Mirth Corp.
- **MDHT in CMS C-HIEP and COSS Projects (15 minutes)**
  - Nitin Jain, IBM GBS
- **Open Discussion and Q&A (15 minutes)**

MDHT Project Benefits for CDA

- **Specification Designers (Standards organizations)**
  - Formalize representation of CDA implementation guide conformance rules
    - Replace current practice of using MS Word for specification development
    - UML specification is testable for consistency and use of best practices
    - Enables automated model-driven development and code-generation
  - Automate publication of implementation guides in multiple formats (PDF and XHTML) and alternative content structure (ballot document vs. implementer view)
  - Automate generation of CDA instance validator from specification model (using Java and OCL)
- **Business Analysts**
  - Consistent format of published implementation guide between different standards organizations (HL7, IHE, and HITSP)
  - Cross-referenced, hyperlinked reference material accelerates analysis and EHR mapping
  - Publish a “developer view” of implementation guide that combines conformance rules from all inherited templates and base CDA type
- **Java Implementers**
  - Reduce Development Cost: Time and resources for analysis and implementation of CDA content and conformance rules
  - Reduce Maintenance Cost: High quality, domain-specific API for programmatic access to CDA content, and validating conformance with standard implementation guide rules (e.g. CCD and HITSP C32/C83)
  - Future support possible for non-Java implementation languages

Full Development Lifecycle

- **Design new specifications**
  - as PIM
  - Reuse/reference spec items (e.g. HL7 CCD)
  - TODO: add model analysis to support harmonization
- **Publish implementation guides**
  - For SDO specification
  - For analyst and developer
  - TODO: expand publishing for service interfaces/interactions
- **Validate CDA instances**
  - Conformance testing generated from model
  - Can only automate computationally testable rules
- **Generate reference implementation for adopters**
  - Transform specification model to PSM
  - Domain-specific Java APIs

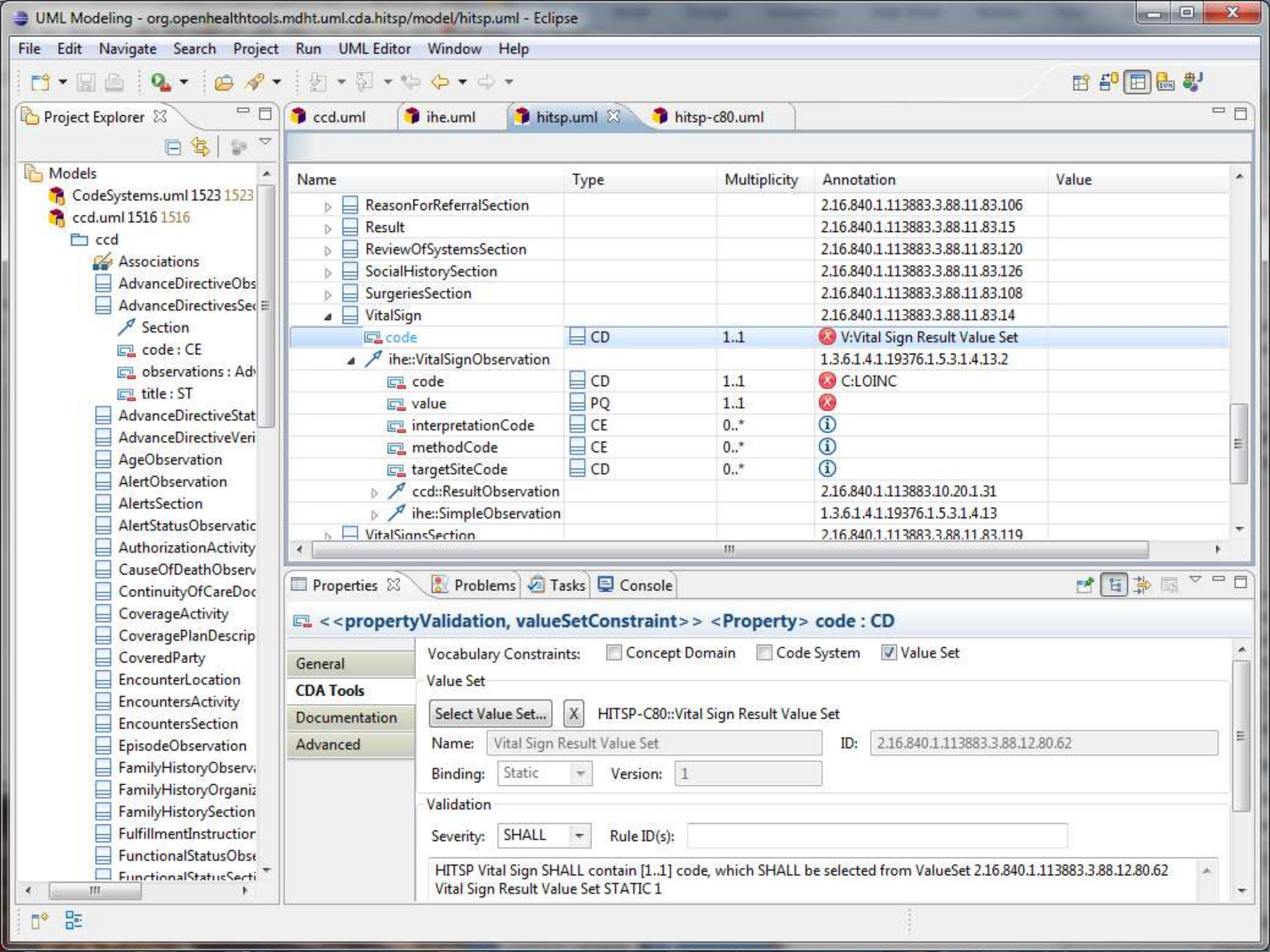
CDA Template Models

- **We are currently working on UML models and Java implementations of the following CDA-based document types:**
  - HL7 Continuity of Care Document (CCD)
  - IHE Patient Care Coordination (PCC) Profiles
  - HITSP C83 and C32 Patient Summary
  - HL7 Public Health Case Report (PHCR)
  - IHE Lab Report Document
  - HITSP C74 Personal Health Monitoring Report
  - Essential Hypertension

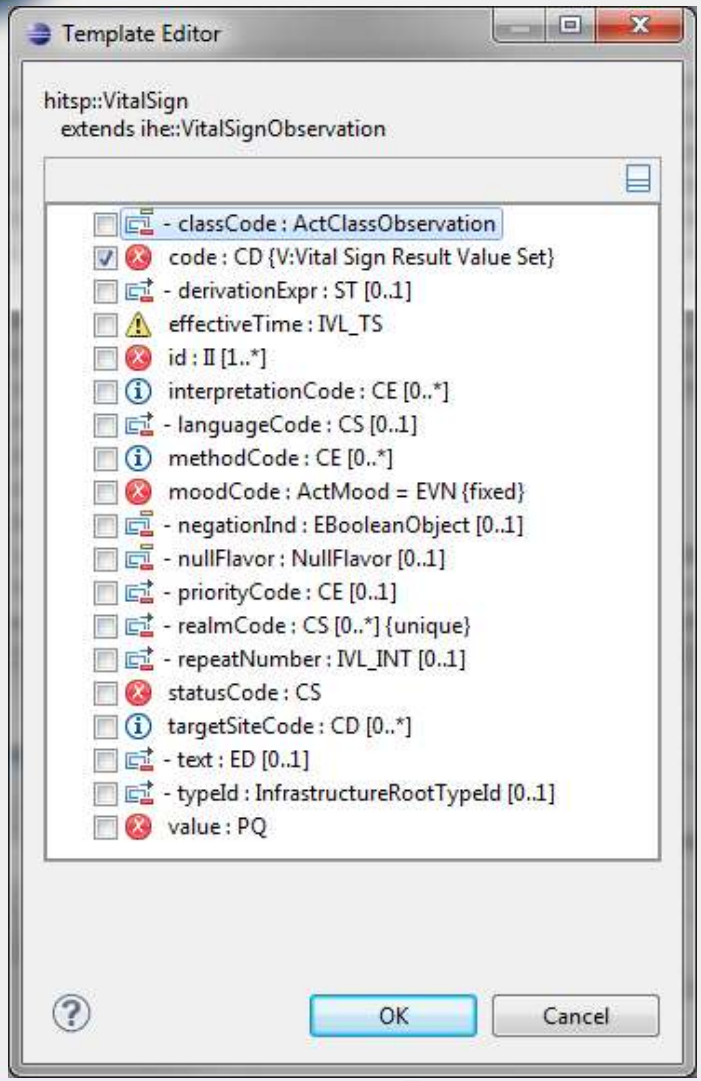
Authoring CDA Templates

- Create a UML class for each template and specify all conformance rules using property redefinitions, directed associations, or using OCL expressions for complex rules.
- We found that the most intuitive and efficient editor for template definitions is a spreadsheet-style table editor. This editor directly modifies the underlying UML model, but with a different interface from the typical class diagram.
- UML class diagrams may also be created as views of the model, or used as an alternative design interface.

- Separate models were created for CDA, CCD, IHE PCC, and HITSP C32/C83 with all conformance rules inherited from base template classses.


UML Modeling - org.openhealthtools.mdht.uml.cda.hitsp/model/hitsp.uml - Eclipse
File Edit Navigate Search Project Run UML Editor Window Help
Project Explorer
ccd.uml
ihe.uml
hitsp.uml
hitsp-c80.uml
Models
CodeSystems.uml 1523 1523
ccd.uml 1516 1516
ccd
Associations
AdvanceDirectiveObs
AdvanceDirectivesSec
Section
code : CE
observations : Adv
title : ST
AdvanceDirectiveStat
AdvanceDirectiveVeri
AgeObservation
AlertObservation
AlertsSection
AlertStatusObservatic
AuthorizationActivity
CauseOfDeathObserv
ContinuityOfCareDoc
CoverageActivity
CoveragePlanDescrip
CoveredParty
EncounterLocation
EncountersActivity
EncountersSection
EpisodeObservation
FamilyHistoryObserv
FamilyHistoryOrganiz
FamilyHistorySection
FulfillmentInstructior
FunctionalStatusObse
Name
Type
Multiplicity
Annotation
Value
ReasonForReferralSection
2.16.840.1.113883.3.88.11.83.106
Result
2.16.840.1.113883.3.88.11.83.15
ReviewOfSystemsSection
2.16.840.1.113883.3.88.11.83.120
SocialHistorySection
2.16.840.1.113883.3.88.11.83.126
SurgeriesSection
2.16.840.1.113883.3.88.11.83.108
VitalSign
2.16.840.1.113883.3.88.11.83.14
code
CD
1..1
V:Vital Sign Result Value Set
ihe::VitalSignObservation
1.3.6.1.4.1.19376.1.5.3.1.4.13.2
code
CD
1..1
C:LOINC
value
PQ
1..1
interpretationCode
CE
0..*
methodCode
CE
0..*
targetSiteCode
CD
0..*
ccd::ResultObservation
2.16.840.1.113883.10.20.1.31
ihe::SimpleObservation
1.3.6.1.4.1.19376.1.5.3.1.4.13
VitalSignsSection
2.16.840.1.113883.3.88.11.83.119
Properties
Problems
Tasks
Console
<<propertyValidation, valueSetConstraint>> <Property> code : CD
General
CDA Tools
Documentation
Advanced
Vocabulary Constraints:
Concept Domain
Code System
Value Set
Value Set
Select Value Set...
X
HITSP-C80::Vital Sign Result Value Set
Name:
Vital Sign Result Value Set
ID:
2.16.840.1.113883.3.88.12.80.62
Binding:
Static
Version:
1
Validation
Severity:
SHALL
Rule ID(s):
HITSP Vital Sign SHALL contain [1..1] code, which SHALL be selected from ValueSet 2.16.840.1.113883.3.88.12.80.62 Vital Sign Result Value Set STATIC 1

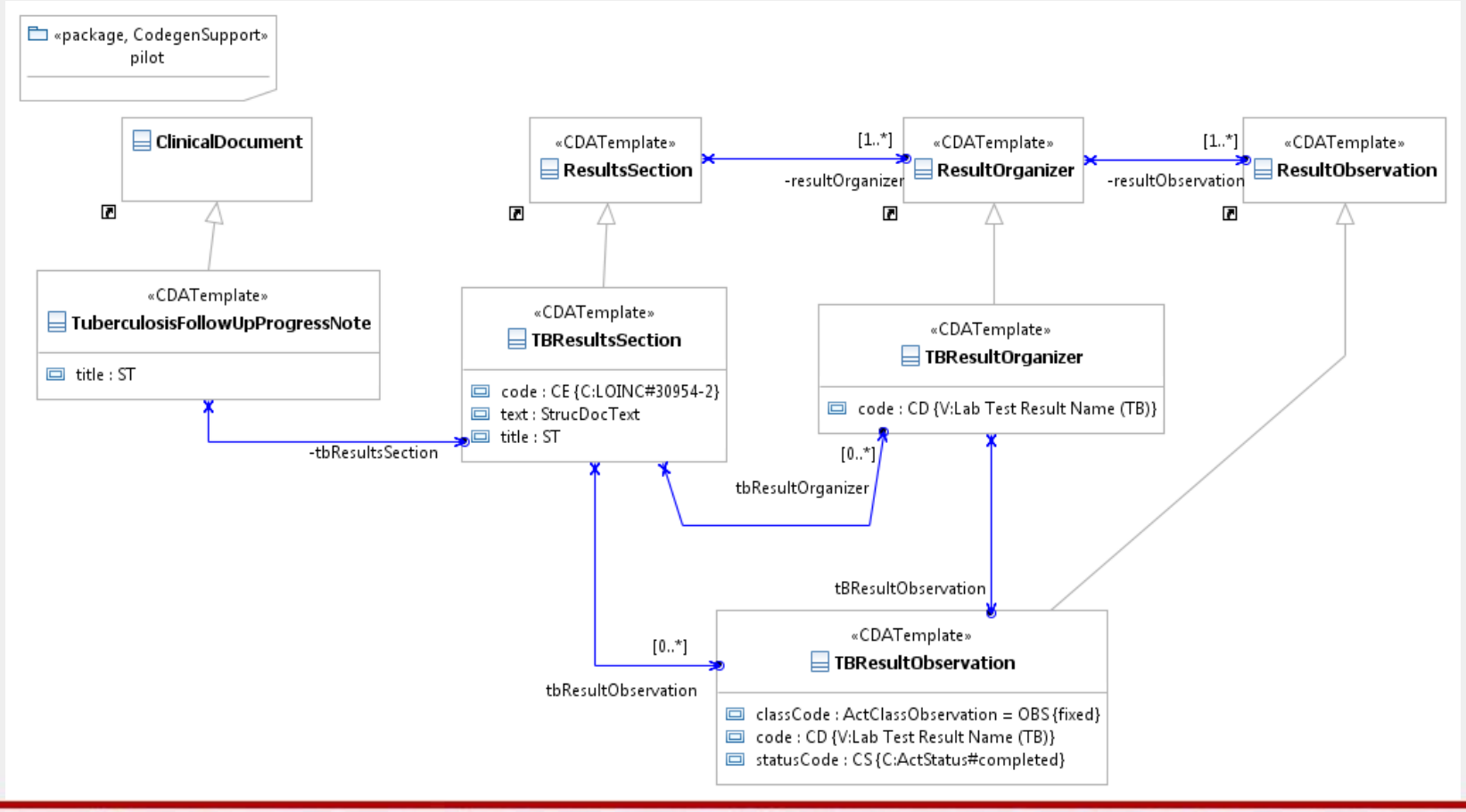
Add a new Template



- Easy reuse and reference to templates in other IG models.
- Dialog wizard to create a new template that conforms to another base template.
- For example:
  - Add: TB Encounter
  - Select base: CCD Encounters Activity
  - Check off inherited attributes that will be restricted
  - Then use Table editor to refine the new constraints.

Class Diagrams

We are working on integrating an open source UML class diagram editor provided by the Eclipse UML2Tools project, but it is not yet ready for end-users. When templates are created using the table editor, one or more class diagrams may be created as views of the model.



Publishing IGs

- The UML model created with template definitions is automatically transformed to DITA XML (OASIS standard), which is then published to PDF and Eclipse Help HTML format using the open source DITA-OT toolkit.
- Automatic generation of example XML instance snippets for each template, included in the published IG.

- Separate developer documentation: Includes the complete aggregate list of all inherited elements and conformance rules. Thus, a developer does not need to "follow the breadcrumbs" of template conformance references. Example provided in PDF output.

Development Only. The Normative content for these specifications may be found on the HL7, IHE, and HITSP web sites.

Vital Sign Observation

[Observation: templateId `1.3.6.1.4.1.19376.1.5.3.1.4.13.2`]

TODO: add class description

1. Conforms to CCD Result Observation template (templateId: `2.16.840.1.113883.10.20.1.31`)
2. Conforms to Simple Observation template (templateId: `1.3.6.1.4.1.19376.1.5.3.1.4.13`)
3. **SHALL** contain [1..1] `code` (CodeSystem: 2.16.840.1.113883.6.1 LOINC STATIC 2.26), where its data type is CE
4. **SHALL** contain [1..1] `value`, where its data type is PQ
5. **MAY** contain [0..*] `interpretationCode`
6. **MAY** contain [0..*] `methodCode`
7. **MAY** contain [0..*] `targetSiteCode`

Figure 1. Vital Sign Observation example

```
<?xml version="1.0" encoding="UTF-8"?>
<ClinicalDocument xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns="urn:hl
  <component>
    <structuredBody>
      <component>
        <section>
          <entry>
            <observation moodCode="EVN">
              <templateId root="2.16.840.1.113883.10.20.1.31"/>
              <templateId root="1.3.6.1.4.1.19376.1.5.3.1.4.13"/>
              <templateId root="1.3.6.1.4.1.19376.1.5.3.1.4.13.2"/>
              <id root="c205dd33-6ced-4ab3-8abb-3fbf3e7259f1"/>
              <code codeSystem="2.16.840.1.113883.6.1" codeSystemName="LOINC"/>
              <statusCode code="completed"/>
```

Generate XML Example

Figure 1. Condition example

```
<?xml version="1.0" encoding="UTF-8"?>
<ClinicalDocument xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns="urn:hl7-org:v3" xsi:schemaLocation="urr
  <component>
    <structuredBody>
      <component>
        <section>
          <entry>
            <act classCode="ACT" moodCode="EVN">
              <templateId root="2.16.840.1.113883.10.20.1.27" assigningAuthorityName="CCD Problem Act"/>
              <templateId root="1.3.6.1.4.1.19376.1.5.3.1.4.5.1" assigningAuthorityName="IHE Concern Entry"/>
              <templateId root="1.3.6.1.4.1.19376.1.5.3.1.4.5.2" assigningAuthorityName="IHE Problem Concern Entry"/>
              <templateId root="2.16.840.1.113883.3.88.11.83.7" assigningAuthorityName="HITSP Condition"/>
              <id root="da560483-e64f-40ae-8f6c-39a4722c27ac"/>
              <code nullFlavor="NA"/>
              <effectiveTime>
                <low value="1972"/>
                <high value="2008"/>
              </effectiveTime>
              <entryRelationship>
                <observation classCode="OBS" moodCode="EVN">
                  <templateId root="2.16.840.1.113883.10.20.1.28" assigningAuthorityName="CCD Problem Observation"/>
                  <templateId root="1.3.6.1.4.1.19376.1.5.3.1.4.5" assigningAuthorityName="IHE Problem Entry"/>
                  <templateId assigningAuthorityName="HITSP Condition Entry"/>
                  <code/>
                  <text/>
                  <statusCode code="completed"/>
                  <effectiveTime>
                    <low value="1972"/>
                    <high value="2008"/>
                  </effectiveTime>
                  <value xsi:type="CD"/>
                </observation>
              </entryRelationship>
            </act>
          </entry>
        </section>
      </component>
    </structuredBody>
  </component>
</ClinicalDocument>
```

Developer Documentation (PDF)

Vital Sign

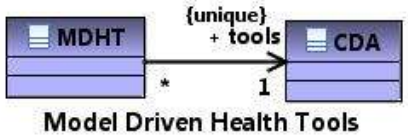
[Observation: templateId `2.16.840.1.113883.3.88.11.83.14`]

1. Conforms to *IHE Simple Observation* template (templateId: `1.3.6.1.4.1.19376.1.5.3.1.4.13`)
2. Conforms to *RIM Infrastructure Root*
3. Conforms to *RIM Act*
4. Conforms to *CDA Clinical Statement*
5. Conforms to *CDA Observation*
6. Conforms to *CCD Result Observation* template (templateId: `2.16.840.1.113883.10.20.1.31`)
7. Conforms to *IHE Vital Sign Observation* template (templateId: `1.3.6.1.4.1.19376.1.5.3.1.4.13.2`)
8. [CCD] **SHALL** contain [1..1] `@moodCode` = "EVN"
9. [CCD] **SHALL** contain [1..*] `id`
10. [CCD] **SHALL** contain [1..1] `statusCode`
11. [CCD] **SHOULD** contain [1..1] `effectiveTime`
    - Represents the biologically relevant time (e.g. time the specimen was obtained from the patient).
12. [HITSP] **SHALL** contain [1..1] `code`, which **SHALL** be selected from ValueSet 2.16.840.1.113883.3.88.12.80.62 Vital Sign Result Value Set STATIC 1
13. [IHE] **MAY** contain [0..*] `methodCode`
14. [IHE] **MAY** contain [0..*] `interpretationCode`
15. [IHE] **SHALL** contain [1..1] `value`, where its data type is PQ
16. [IHE] **MAY** contain [0..*] `targetSiteCode`
17. [CCD] **SHOULD** satisfy: The value for 'code' SHOULD be selected from LOINC (codeSystem 2.16.840.1.113883.6.1) or SNOMED CT (codeSystem 2.16.840.1.113883.6.96), and MAY be selected from CPT-4 (codeSystem 2.16.840.1.113883.6.12).
18. [CCD] **SHALL** satisfy: The methodCode SHALL NOT conflict with the method inherent in code
19. [CCD] **SHALL** satisfy: Where value is a physical quantity, the unit of measure SHALL be expressed using a valid Unified Code for Units of Measure (UCUM) expression.
20. [CCD] **SHOULD** satisfy: Contain one or more referenceRange to show the normal range of values for the observation result

Validating CDA Instances

- The template conformance rules represented in UML are transformed to OCL as part of the automated code generation.
- The resulting Java classes encapsulate all validation rules and may be used to parse and validate a CDA document instance.
- We have created an example web application that may be used to validate CDA documents for implementation guides that we have modeled. See http://cdatools.org
- We are integrating validation from the generated Java libraries into the Eclipse open source XML instance editor. This editor already includes good support for "content assist" and validation based on the CDA.xsd schema. But we have used the Eclipse extension points to also validate CDA instances using the conformance rules that go beyond schema structure.

Validation on a Web Application



Model Driven Health Tools

MDHT Clinical Document Architecture (CDA) Document Validation Services

The site provides validation services for CDA documents using the MDHT CDA Tooling solution and the complete source code is available on the MDHT Subversion. The web page uses a web service to provide XML validation content which is then rendered using XSLT.

MDHT CDA Diagnostics Service

CDA Document	
	Browse...
CDA Diagnostic Level	Only Error and Warnings Diagnostics
	Validate

Validation Results on Web

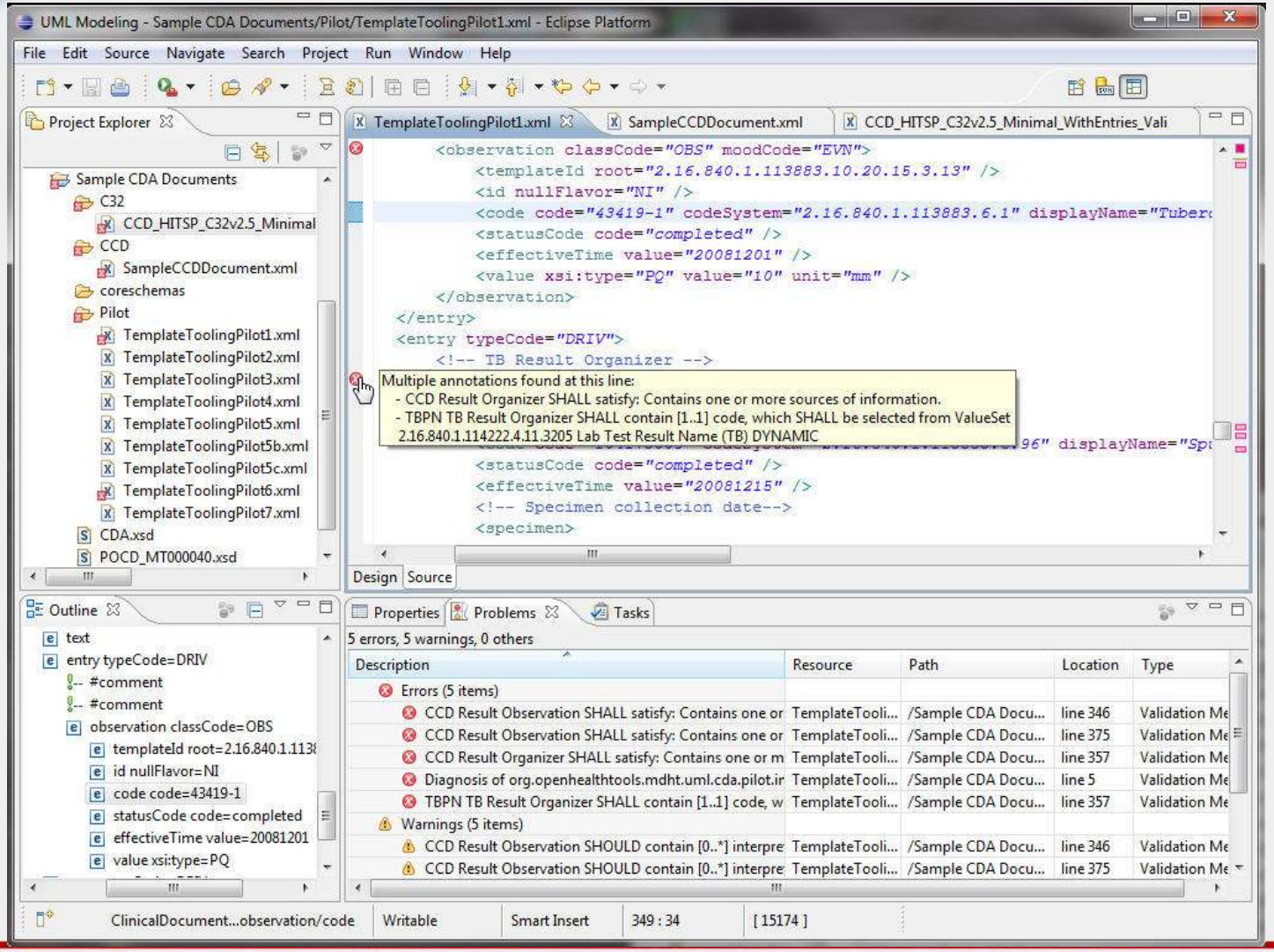
MDHT Clinical Document Architecture (CDA) Document Validation Services

Diagnostics For CDA Document xxx

Total	Errors	Warnings	Information
9	5	4	0

Specification	Severity	Message
/clinicalDocument[1]		
org.eclipse.emf.ecore	error	Diagnosis of org.openhealthtools.mdht.uml.cda.pilot.impl.TuberculosisFollowUpProgressNoteImpl@1098594{urn:hl7-org:v3#//@clinicalDocument}
/clinicalDocument[1]/component[1]/structuredBody[1]/component[2]/section[1]/entry[1]/observation[1]		
org.openhealthtools.mdht.uml.cda.ccd	warning	CCD Result Observation SHOULD satisfy: Contain one or more referenceRange to show the normal range of values for the observation result
org.openhealthtools.mdht.uml.cda.ccd	error	CCD Result Observation SHALL satisfy: Contains one or more sources of information.
org.openhealthtools.mdht.uml.cda.ccd	warning	CCD Result Observation SHOULD contain [0..*] interpretationCode
/clinicalDocument[1]/component[1]/structuredBody[1]/component[2]/section[1]/entry[2]/organizer[1]		
org.openhealthtools.mdht.uml.cda.ccd	error	CCD Result Organizer SHALL satisfy: Contains one or more sources of information.
org.openhealthtools.mdht.uml.cda.pilot	error	TBPN TB Result Organizer SHALL contain [1..1] code, which SHALL be selected from ValueSet 2.16.840.1.114222.4.11.3205 Lab Test Result Name (TB) DYNAMIC
/clinicalDocument[1]/component[1]/structuredBody[1]/component[2]/section[1]/entry[2]/organizer[1]/component[1]/observation[1]		
org.openhealthtools.mdht.uml.cda.ccd	warning	CCD Result Observation SHOULD satisfy: Contain one or more referenceRange to show the normal range of values for the observation result
org.openhealthtools.mdht.uml.cda.ccd	error	CCD Result Observation SHALL satisfy: Contains one or more sources of information.
org.openhealthtools.mdht.uml.cda.ccd	warning	CCD Result Observation SHOULD contain [0..*] interpretationCode

CDA IG Validation in XML Editor



Domain Specific Java API