11073 Tooling
HL7 F2F May 2018
Köln
goo.gl/7SkSyQ

Michael Faughn
Prometheus Computing

TL;DR

- The DIM is expressed as UML. The UML governs everything.
- Device Profile Editor web app, printed standard, and XSD are programmatically generated.
- Web app is nominally usable now (prove me wrong).

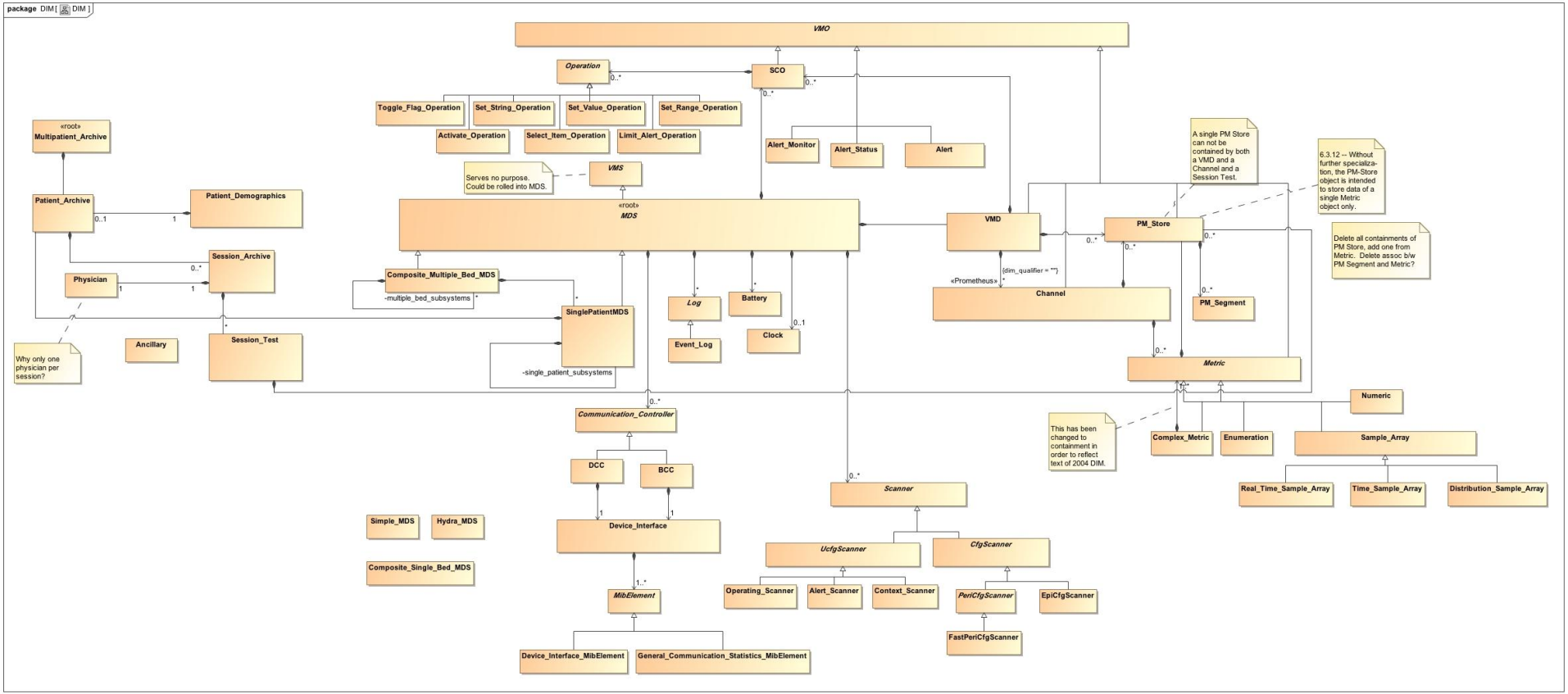
UML Development

- Initial UML model of the DIM programmatically derived from 11073-10201:2004 in early 2012
- Work on applications begun FY 2013
- Manual revisions performed
- Ongoing development of additional UML

Tool Genesis from UML Models

- IEEE11073:10101 Nomenclature (RTMMS)
- IEEE11073:10201 Point of Care Device DIM
- IEEE11073:20601 Personal Health Device DIM
- IEEE11073:10207 BICEPS
- IEEE11073:20101 (ASN.1 Simple Types)
- Device Profiles
- UML Metamodel (represents 10201 UML in web applications)
- Printed Standard

PoCD DIM Classes



The Model is the Standard

Why? Because it is computable / automatable.

Programmatically deriving artifacts from a common source of truth help to ensure harmonization.

- Printed Standard
- Software tools (Device Profiling, Validation, ...)
- XML Schema
- Inter-Standard Interoperability

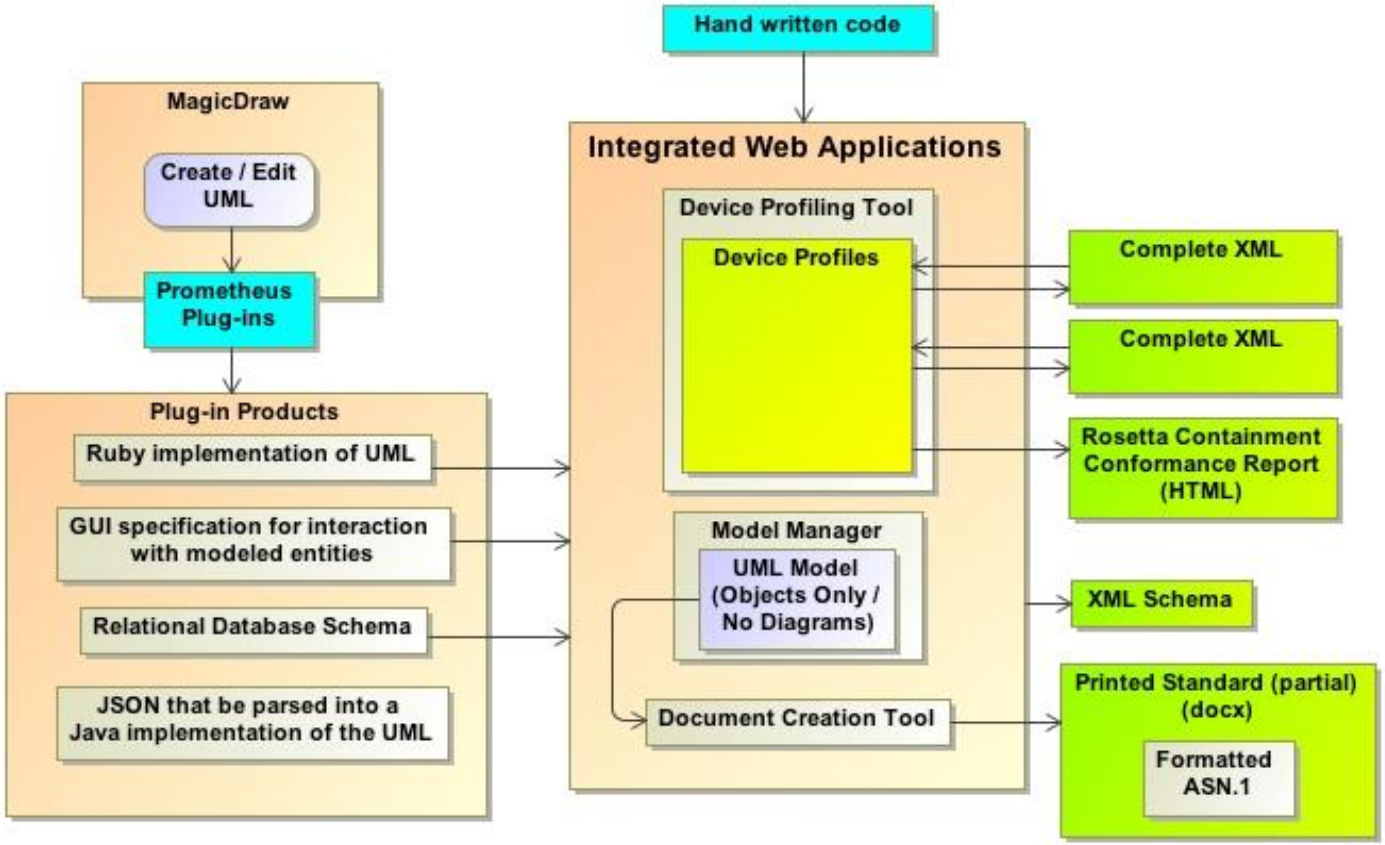
UML Challenges

- UML (or UML tools) has trouble expressing some constructs in a convenient way
  - Class instance variables are unsupported
  - BNF (i.e. ASN.1)
  - BIT STRING (syntactic representation of multiple boolean attributes)
- Keep the standard ‘pure’ vs. supporting the functionality that artifacts require.

UML Challenges

- Each UML element type used has to be implemented for code module that produces an end product (web application, printed standard, etc.). Lots of work the first time you do it and every time you build a new module to produce a new type of artifact.
- UML and XML have some differences that are difficult to reconcile.
- Available UML tools aren't perfect.

From UML to Everything Else



Generated from UML

- Web Application Specification
  - Database Schema
  - Class definitions
  - Default User Interface Specification
- ~1000 classifiers
- ~ 3,000 lines of in-memory code per classifier*
- UML!

* once fully processed by the application generation toolchain

Generated from Application Code

Web Application

- Device XML
  - Concise “Rosetta Containment Hierarchy”
  - Comprehensive, complete representation
- HTML “dotted” notation tables for devices

Command Line

- XML Schema
- ASN.1
- DIM pseudocode
- .docx for inclusion in IEEE document
- ad-hoc, custom reports

Web Application Features

- Create and Manage Device Infomation Model Profiles
  - Containment + Nomenclature
- RTMMS mirror
- Manage UML model(s) (*with caution)
- Build standards document content (.docx)

Generating an IEEE Standard

From the Model to the Document

- 99+% of Section 7: DIM Object Definitions
  - ~⅔ of the content of the entire document
- Structure and content of generated material is managed via web application

iMeetCentral (goo.gl/CKk8FG)

11073 Working Group* -> Drafts -> Drafts in Progress

- 2010 Draft
- Current, Generated Draft
- Diff b/w 2010 Draft and Current Draft

*__Not__ 11073 PoCD Working Group Workspace

Comments & Revisions

Google Sheet: goo.gl/NE8FjJ

- Try to understand and resolve all items
- Assume that the new draft represents a best effort to address items found in the spreadsheet
- Clean the slate and start Comments & Revisions document going forward

dim.prometheuscomputing.com

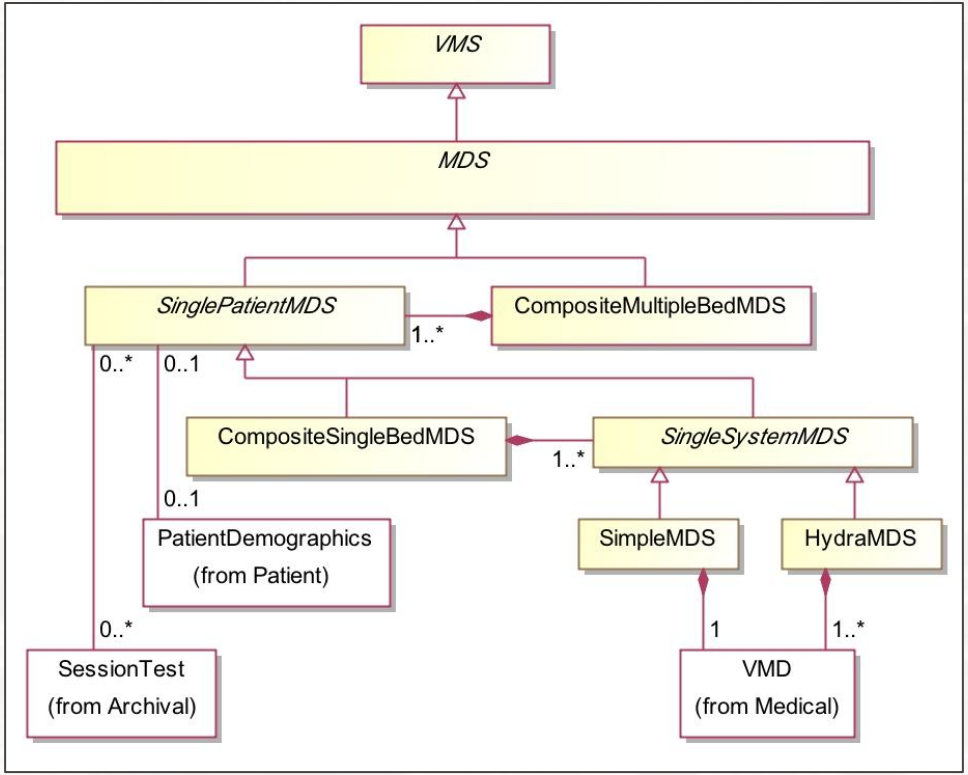
Ask for an invitation code:
michael@prometheuscomputing.com

Since New Orleans WG Meeting (Winter 2018)

New Orleans Slide Deck: goo.gl/1bujhK

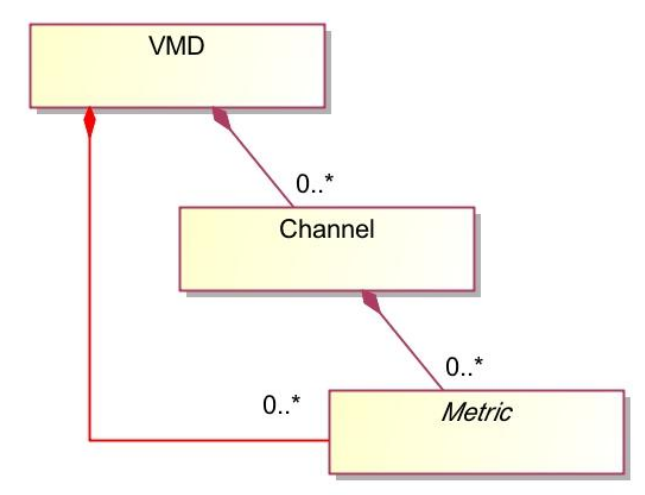
MDS Types

- More clearly conveys the intent of the 2004 version.
- Should not require any changes on the part of implementers (backwards compatible).



Channel Not Mandatory

2004: “The Channel object is not mandatory for representation of Metric objects in a VMD.”



Channel Now Mandatory

2004: “The Channel object is mandatory for representation of Metric objects in a VMD. In some cases a VMD may contain a single Channel and that Channel may be functionally superfluous. This may be likely if the attribution of the Channel object consists only of the mandatory *type* and *handle* attributes. In this case it is recommended that the value for the *type* attribute be initialized using the values corresponding to the term having the reference ID *MDC_DEV_CHAN*, which is defined in IEEE11073:10101. Usage of this term, in this context, may serve as a signal that the Channel object in question may not require representation in other messaging formats (e.g. IHE PCD-01).”

MDS Question

- Can a composite MDS have multiple MDIBs?
  - Implications for CommunicationController

“A medical device MDIB contains either no Communication Controller object or one BCC object or one DCC object”

Alert Question (Section 6.4)

“If a device contains an Alert object, it shall not contain any Alert Status or the Alert Monitor objects. A single Alert object is needed for each alarm condition that the device is able to detect.”

“If a device contains an Alert Status object, it shall not contain any Alert or the Alert Monitor objects. Each VMD or MDS in the MDIB is able to contain at most one instance of the Alert Status class.”

“If a device contains an Alert Monitor object, it shall not contain any Alert or Alert Status objects. An MDS shall not contain more than one Alert Monitor object.”

IEEE Standards Document

- Refined document generation
  - Formatting changes
  - Bug fixes
- Fragment generation (individual clauses, tables, etc.)
- Now avoiding generation of Section 7 as a whole
- Figures fixed

Getting to Ballot

- Address IEEE's comments in the unofficial pre-ballot draft
- Include UDI in the standard
- Review / resolve a few other comments

Next Steps

- UI improvements
- Refactoring code to facilitate the maintenance of multiple profile editors (PoCD, PHD, BICEPS).

...

- You tell me.

dim.prometheuscomputing.com

Ask for an invitation code:
michael@prometheuscomputing.com