Implementation of the International Defence Enterprise Architecture Specification (IDEAS) Foundation in DoD Architecture Framework 2.0

9 MARCH 2010
DAVE MCDANIEL
Contractor, Enterprise Architecture & Standards
Office of the DoD Deputy Chief Information Officer
+1 (619) 253-9040  davem@silverbulletinc.com
Outline of Presentation

- IDEAS Recap
- Why we used IDEAS – benefits
  1. Re-use of common patterns saved a lot of work
  2. Reconciliation and analysis tool
  3. Information pedigree model
  4. Design reification and requirements traceability
  5. Services description
  6. Semantic precision
  7. Mathematical precision
- How we implemented IDEAS
- Implementation challenges
IDEAS Recap
Top-Level Foundation

- Four dimensionalist -- x,y,z,t
- Extensional -- physical existence is the criterion for identity
- Signs and representations are separated from referents
- Mathematics:
  - Type theory ~ Set theory
  - Mereology (wholes and parts)
  - 4D Mereotopology (spatio-temporal relations)

Commutative and anti-commutative, e.g., \( A \cap B = B \cap A \)
Reflexive and irreflexive, e.g., \( A \subset A, A \nsubseteq A \)
Associative, e.g., \( A \cup (B \cup C) = (A \cup B) \cup C; \quad A \cap (B \cap C) = (A \cap B) \cap C \)
Transitive, e.g., \( A \subset B \land B \subset C \Rightarrow A \subset C \)
others:
\( a \in A \land A \subset B \Rightarrow a \in B \)
if \( \{A_i\} \) forms a partition of \( A \) then \( a \in A_j \Rightarrow a \notin A_k \forall j \neq k \)
Mereotopologic Math Examples

• Overlaps, spatial relationships (mereotopology)
  Parthood \( xPy \equiv x \text{ is a part of } y \)
  Proper part \( x \) is a proper part of \( y \) \( x\langle P\rangle y \equiv xPy \land \neg yPx \)
  \( P \) and \( \langle P \rangle \) are transitive: \( xPy \land yPz \Rightarrow xPz \)
  \( aPb \land a \neq b \Rightarrow \neg bPa; \)
  \( P \) is antisymmetric: \( xPy \land yPx \Leftrightarrow x = y \)
  Overlap proposition \( xOy \Leftrightarrow \exists z \in zPx \land zPy \)
  Overlap operator: \( x \bigcap y = z \in z_oPx \land z_oPy \land \forall z_i \neq z_o, z_iPx \land z_iPy \Rightarrow z_iPPz_o \)
  Underlap \( xUy = \exists z \in xPz \land yPz \)
  \( xOy \) and \( xUy \) are reflexive, symmetric, and intransitive
  Overlap Associative \( aO(bOc) = (aOb)Oc \)

• Behaviors -- Sequences, before-after (4D mereotopology)
  Before \( xBy \) is transitive: \( xBy \land yBz \Rightarrow xBz \)
  Proper before is irreflexive \( \neg u \langle B \rangle u \)
  Proper before is anti-commutative \( a \langle B \rangle b \Rightarrow \neg b \langle B \rangle a \)
Some Math Sources

- National Center for Ontologic Research (NCOR),
  http://ontology.buffalo.edu/smith/
- Direct Model-Theoretic Semantics for OWL 2,
  http://www.w3.org/TR/2009/REC-owl2-direct-semantics-20091027/
  - Vocabulary
  - Interpretations
    - Object Property Expressions
    - Data Ranges
    - Class Expressions
  - Satisfaction in an Interpretation
    - Class Expression Axioms
    - Object Property Expression Axioms
    - Data Property Expression Axioms
    - Datatype Definitions
    - Keys
    - Assertions
    - Ontologies
  - Models
Benefits of IDEAS for DoDAF 2
1. Rigorously worked-out common patterns are reused

- Saved a lot of repetitive work – “ontologic free lunch”
- Concentration of rigor on common patterns results in higher quality and consistency throughout
- Model compactness -- DM2 is tiny compared to its predecessor by \textbf{two} orders of magnitude!
- Easier to learn -- a few hard concepts are easier to learn than thousands of conceptually intractable ones.
- Implementations get reuse too – same code, queries, … work for many datasets
2. Reconciliation and analysis tool (slide 1 of 4)

- State of practice in data modeling:
  - Noun and adjective analysis
  - Similar to natural language written in a diagram
  - Often laden with entrenched but obsolete technology considerations

The fundamental concepts of Entity-Relationship and Class Models:

- **predicate**
- **subject**
- **object**

Implicit, built-in, language features:
- predicate “has” (for attributes)
- Plural, singular notions (cardinality)
- Sufficiency and completeness notions (e.g., no-nulls)
One Result of this practice -- data model “wars”

Users of these different models believe their model is the best for many purposes, in many cases overlapping purposes.

- Anti-Submarine Warfare (ASW) COI
- Blue Force Tracking (BFT)
- C2 Interoperability Group
- CBRN
- Coalition C2 Interoperability (Coal C2)
- Common Sensor
- GEOINT Standards COI (GWG COI)
- Global Force Management (GFM)
- GPS Based Positioning Navigation Timing Service
- Integrated Fires
- Joint Air and Missile Defense
- Joint Air Track (JAT)
- Joint Electronic Warfare Data Standardization
- Joint Targeting Intelligence (JTI)
- Maritime Domain Awareness
- Meteorology-Oceanography (METOC)
- Mine Warfare
- Symbology (SYM)
- Undersea Warfare XML (usw-xml)

Like diverse languages, there is a high cost to learn
Some real-world and costly results of this practice

• Cost and project risk
  – Developers and integrators must learn multiple proprietary “languages”
  – Need to build many *translators*
  – Over promised ability of “translation hubs”
  – Context, interdependent, and value-dependent translations

• Operational impact
  – E.g., from “lossy” translations, mis-translations, …
  – Difficulty in transitioning new technologies, e.g., automated processing tools
  – Prohibits or impedes scaling and cross-domain integration and data sharing
  – Impedes Net-Centricity / OA / SoA due to need for much human interaction, e.g., no automated unanticipated users

The costs and risks – both project and operational -- are usually underestimated
Reconciling Using IDEAS
Analysis Technique: BORO

- Agreed-upon principles that provide a principled basis for issue analysis

Example decision process:

1. Select a data element or instance
2. Does it represent something with spatial & temporal extent?
   - NO
   - Does it represent something with members?
     - NO
     - Add to ontology
     - YES
     - Identify exemplar members and feed back into process
     - YES
     - Identify Things related by the tuple and feed back into process

Example BORO analysis diagram:

Dave’s Document, Original
Copy and Send
Copy
DD, copy 1
Send part
Flow process
Receive part

Dave’s Document, Copy 1, in flow state

3. Information Pedigree Model

- Workflow model, e.g., Open Provenance Model (provenance = linked together pedigrees)
- = Activity model (OV-5 + 6c) ➞ got nearly for free!
4. Design Reification and Requirements Traceability
5. Service Descriptions (1 of 2)

From OASIS SoA RAF, Figure 27, “Service Description”

Service Description is an artifact and requires components to store, find, access, and manage the artifact.

Service Description identifies available metrics and how to access; requires components to gather, store, and provide access to metrics.

Service Description:
- Identifies known policies and contracts; requires mechanisms to create and maintain policies (may be outside SCA), components to store, find, access, and manage PSCs.

**Service Description**
- Service Reachability
  - Protocols
  - Endpoint
- Service Interface
  - Behavior Model
  - Information Model
- Service Functionality
  - Functions
  - Technical Assumptions
    - must be consistent with
  - Service Level Real World Effect
    - Temporal sequence of actions
  - Service Level Interaction Policies
    - All actions of this service
- Policies & Contracts
  - Contracts
- Metrics
  - All actions of this service
  - Temporal sequence of actions
Service Descriptions as Modeled in DM2

This means a Service Description can have all the structure of an Architectural Description, e.g.,

- Activities
- Before-After Rules
- Conditions
- Data structures
- Locations
- Dependencies
- Etc.

Got this one for free too!
6. Semantic Precision for Heterogeneous Data Integration

**Free-text**
- Human-interpretable only

**Structured document**
- Human-interpretable but with a predictable organized arrangement
- More structure than structured text
  - Named records (or tables or classes) that are some sort of container for named fields (or attributes or columns).
  - Associations and relationships, containers can point to information in other containers
  - Because of the labeling, you can tie the information together and query them. A SQL query is just fundamentally a selection of the information.
  - Referential integrity, data validation, cardinality rules, etc.

**Database**
- Mathematically structured
  - Applicable mathematics:
    - Set or type theory
    - Mereology
    - Mereotopology
    - 4 dimensionalism
    - Predicate calculus
    - Logics: modal, Kripke, …
  - Rules, operators:
    - Commutative, reflexive, transitive, …
    - Member-of, subset-of, part-of, …

**A spectrum of information sharing**

**Depends on near-universal mathematics and science that all learn very similarly**
Heterogeneous Data and EA

• For example:
  – Interoperability assessment
  – Capability gaps and overlaps
  – Capability evolution measures
  – SoS, FoS
  – Portfolio optimization
  – Joint, multi-agency, coalition operations
  – Analysis of alternatives

The very reason for EA implies a need to look at data from multiple sources
7. Mathematical precision

Create architectural descriptions

Submit for core process event
For example:
1. Capability solution proposal
2. Acquisition milestone review
3. Interoperability and supportability assessment checkpoints
4. Budget cycle (PPBE, IRB, CPM)
5. Ops Plan (contingency update cycle, actual)

Get and integrate relevant datasets

Analyze and assess
For example:
1. Queries for disconnects, inconsistencies, …
2. Specialized tools (e.g., cost / risk / performance / sustainment models, interoperability assessment)
3. Process simulators (e.g., comms flow, workflow, Petri nets, state machines)
4. Campaign, mission, engagement, etc. simulators

All have high-sensitivity to misinterpreted, erroneous, incomplete, incompatible, … data

Present Results for core process decisions
How did we implement IDEAS in DM2?
The DM2 Has Three Levels

- DIV-1
- DIV-2
  (This is where almost all the design and analysis work is done)
- DIV-3
  (Auto-generated from the LDM)

- Conceptual Data Model (CDM)
  Concepts and concept relationships

- Logical Data Model (LDM)
  Reified and Formalized relationships

- Physical Exchange Schema (PES)
  XML encoding of LDM
Conceptual Phase

DoDAF 2.0 “Core” Process Workshops
1. Joint Capabilities Integration and Development System (JCIDS)
2. Program, Planning, and Budgeting Environment (PPBE)
3. Defense Acquisition System (DAS)
5. Systems Engineering
6. Capabilities Portfolio Management

Existing Models and Databases
(many)

DoDAF 1.5 “Parking Lot” Issues

Data WG
1. Collect terms
2. Make a pass on “core” terms
3. Group related terms
4. Gather authoritative definitions for “Core” terms
5. Proposed definitions (+rationale, examples, and aliases)

Authoritative Documents (e.g., DODI, CJCSI, …)

EA Presentation WG

Terms with rough consensus definitions, sources, aliases, rational, examples

Process EA information needs

Design information collection template

Conduct and facilitate

Compile process information needs

Data dictionaries & models

DoD Methods WG
Logical Phase

Data WG

Using a UML class modeling tool:

- **Add relationships**
  - During this activity, repeating association patterns became apparent – IDEAS!

- **Add attributes**
  - During this activity, normalization led the WG to see that attributes are just relationships – IDEAS!

- **Refine detail**
  - During this activity, it became apparent:
    - Details are just specializations – IDEAS!
    - Term reconciliation could be done using BORO – IDEAS!

Initial thinking about relationship types. (IDEF 5)

Ontology Relationship Types

- Meronymic
- Influence
- Dependency
- Classification

- Case
- Temporal
- Spatial

CDM

EA Methods WG

EA Presentation WG

1. Data Dictionary
2. UML Ontology Model
Mechanization

- Add DoDAF concepts and concept relationships as extensions (subtypes) to IDEAS
  1. Start with words and definitions
  2. Use BORO analysis to figure out the IDEAS type
  3. Identify and include in data dictionary aliases and composites (concepts that are modeled as a structure, e.g., Role, Goal.)
Independent Entities
Specialization

DoDAF 2 Domain Concepts
So their mathematical meaning is known
Physical Level

- Auto-generated from UML-ish file – no additional semantics added or changed
- Because the native XSD generator in the UML tool did not understand IDEAS Profile, an XSD generator had to be developed (UK and US)
- Four XSD’s:
  1. IDEAS Foundation, version 1.0
  2. DM2 additional foundation
  3. Classification marking (externally controlled)
  4. DM2 exchange data
- Very simple structure

\[\text{never instantiated, metadata reference only}\]
Challenges
Frameworks

- IDEAS precision reveals ambiguities in framework models which requires revisions of the descriptions, deeper analysis of purposes, …
- The mathematics of some associations are ambiguous and take work to figure out, e.g., maps-to, depends-on, has-authority-over
Socialization Challenges

- Ontology education
  - Computer Science education unwittingly emphasizes human interpretations of names and descriptions
  - Ontologic experience is so everyday, conscious dialog about it is difficult
  - Marketing claims about ontology, semantics, interoperability, … have, and continue to, confuse the user community

- Educating the business value of precision
  - Makes work harder for architectural description producers
  - Integration and analysis needs have often been forgotten
DM2 Collaboration Helped

- DM2 WG open to all
- Collaboration Site
- Business rules, e.g.,
  - Aggregation and subtype rules
- Coordination with many other groups, e.g.,
  - Controlled vocabulary
  - Data models
  - Vendors and implementers
  - Software and systems organizations

1. Current baseline CDM, LDM, and PES files and documentation
2. Working copy
3. IDEAS model and profile
4. Folders with:
   - WG information
   - References and research
   - Tutorials and briefings
5. Next meeting info
6. Links to IDEAS & BORO
Adoption Challenges

**Adopter Types**

- Database or repository implementers – how to
- Software and systems engineering tool vendors – mapping semantics
- Modeling and Simulation and Executable architecture tool vendors and developers – scenario, C&P, … representation
- Custom analysis tool vendors and developers, e.g., portfolio analysis or interoperability assessment tools – relevant parameter representation

**Mitigators**

1. Pilot, early adopter, and vendor support
2. Sample database
3. Education and communication program on wide range of EA data assets
4. Semantic interoperability layers definition
5. Exemplars and corresponding education
The Wide Range of EA Data Assets

DM2 is the neutral format for Interchange

<table>
<thead>
<tr>
<th>IDEAS, OWL, SUMO, …</th>
<th>Users Props</th>
</tr>
</thead>
<tbody>
<tr>
<td>DM2 DM2</td>
<td>PES XMI w/DMM</td>
</tr>
<tr>
<td>XML</td>
<td>XML ODBC</td>
</tr>
<tr>
<td>etc.</td>
<td>etc.</td>
</tr>
</tbody>
</table>
DoDAF 2 Exemplars

- They are:
  - Collections of architectural views and their corresponding DM2 PES XML document examples
  - From coherent datasets, e.g., UPDM S&R, NCES ISP

- How they are being developed:

  1. Discuss Candidate Datasets with Core Process Stakeholders
  2. Conform Diagram to DoDAF 2 and Add Legends
  3. Add Additional Markups for DM2
  4. Enter Into DM2 Database
  5. DM2 DB
  6. DM2 PES XML Document
  7. Review With DM2 WG

  - 1. DoDAF Journal
  - 2. DoDAF Outreach Brief - Views
  - 1. DoDAF Outreach Brief – DM2 Developers / Analyst / Integrator
  - 2. DM2 Description Document – PES
  - 3. DoD MDR
  - 4. DM2 Collaboration Site
1. DoDAF 2 Exemplars:
   View Diagrams
   View DM2 PES Datasets

   Tool DB
   (or data structure)

   Develop views in tool

   DoDAF View
   Diagram Publisher

   DoDAF 2 View
   Diagrams and
   Descriptions

   Review for
   DoDAF 2 Conformance

   DoDAF WG

   DM2 PES
   XML Generator / Exporter

   DM2 PES
   XML Document

   DM2 PES
   XML Document
   Validator

   Data Browsers

   Review for
   DM2 Conformance

   DM2 WG

   DM2 DB

   DoDAF 2 View
   Diagrams and
   Descriptions
Summary

- The IDEAS project started as a data sharing project.
  - It produced fruit that was not originally anticipated, e.g.,
    - A formal foundation based on solid mathematics
    - A methodology for analysis of domain concepts
- The adoption by DoD (DoDAF) is the beginning of being able to integrate, cross-walk, and analyze heterogeneous federated architectural description data sources
  - This is critical in achieving DoD’s EA goals
- To introduce this level of rigor takes care, patience, and a good communications team
Questions and Comments?