System Lifecycle Handler for enabling a digital thread for smart manufacturing

NIST MBE Summit 2018
April 4, 2018

Manas Bajaj, PhD & Dirk Zwemer, PhD - Intercax
Tom Hedberg - NIST
Contents

• System Lifecycle Handler - Foundations
  • Project goals
  • Use Cases
  • Graph foundations

• System Lifecycle Handler – Proof-of-concept demonstrations
  • Querying the digital thread
  • Building the digital thread
  • Maintaining the digital thread
  • API for the digital thread
  • Tools, APIs, and open standards used in the PoC

• Next steps

• Questions and Comments
NIST Digital Thread Project

• Developing methods & open standards to support validating, certifying, and connecting engineering models across lifecycle

• Goals
  • Seamless traceability:
    System -> Design -> Manufacturing -> Operations -> Maintenance
  • High-quality manufacturing
  • Enterprise knowledge reuse

• Learn more at: http://www.nist.gov/el/msid/syseng/dtsm.cfm
System Lifecycle Handler (SLH) – Use Cases

• **Build** digital thread for systems
  • **Connect** to data across the enterprise (Systems, PLM, CAD, ALM, Project management, Manufacturing, Operations) and spin a digital thread
  • **Generate** models as information moves across disciplines

• **Query and search** the digital thread

• **Manage** the digital thread
  • **Track changes** in versioned models connected in the digital thread
  • **Compare, synchronize, repair** connections and models

• **Visualize** the digital thread
• Organizations deal with a diverse, multi-vendor engineering toolset.

• They create and store product/system data in a variety of tools, models and repositories: PLM, ALM, CAD, spreadsheets, databases, SysML models...
• Federating data and models across disciplines
• Connecting model elements and data at different levels of abstractions
Digital Thread – A simple example showing artifacts and connections

If I change this requirement, what is the downstream impact, e.g. to CAD and CAM models?

Trace the CAD and CAM models for this part and compare attributes against test results

Requirements (e.g. DOORS-NG)  Mechanical Design (CAD, e.g. NX, CREO, STEP)  Manufacturing (CAM, e.g. MTConnect)  Quality & Inspection (e.g. QIF)

Inter-model connections  Intra-model connections
• What is the purpose of model-based connections?

  Reference Connections
  Track/compare/sync versions of connected elements

  Data Map Connections
  + Track/compare/sync element attributes

  Function Wrap Connections
  + Track/execute connection elements

  Model Transform Connections
  + Track/compare/sync element structure (multi-level)
Digital Thread is a *Graph*

- **Graph – Nodes and Edges**
- **Nodes and Edges may have**
  - Name
  - Type (Typed Graph)
  - Properties (Property Graph)
- **Edges may have**
  - Direction (Directed vs. Undirected Graph)
- **Graphs can be**
  - Stored
  - Queried (Pattern matching)
  - Traversed (e.g. Breadth-first, Depth-first)
  - Generated and Transformed
  - Analyzed
Information models can be abstracted as graphs

**System Model** (SysML)

**Part/Item Structure** (Teamcenter)

**Requirements** (DOORS NG)

**Project Structure** (JIRA)

**Simulink model**
Contents

• System Lifecycle Handler - Foundations
  • Project goals
  • Use Cases
  • Graph foundations

• System Lifecycle Handler – Proof-of-concept demonstration
  • Querying the digital thread
  • Building the digital thread
  • Maintaining the digital thread
  • API for the digital thread
  • Technology stack and open standards

• Next steps

• Questions and Comments
Testbed model for proof-of-concept – Configurable UAV

```
<p>| | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
```
Enclosure Box for an Avionics assembly used in the Configurable UAV
Available data and models for the Enclosure Box

- **SysML** model of the UAV and the payload
- CAD models for multiple variants and revisions of the Box (SolidWorks files on GitHub)
- Design flow management in **JIRA**
- 20 instances of each part are manufactured. For each instance:
  - Machine sensor data streams for each part instance (**MTConnect 1.3 XML on GitHub**)
  - NC code data - ISO 6983 (**G-code files on GitHub**)
  - First article inspection reports (**QIF 2.1 XML on GitHub**)
  - Receiving inspection reports (**QIF 2.1 XML on GitHub**)
Approach for organizing the digital thread – Schema and Linked Models

Product Concept Level
- Jama: Box Reqs
- NIST MTC CRADA ASSBLY
  - NIST MTC CRADA BOX
  - NIST MTC CRADA COVER
    - Jama: Cover Reqs

Design Variant Level
- JIRA: Box Design Status
- GitHub: Box Design CAD
- NMC ASSBLY Rev D
  - NMC BOX Rev D
  - NMC COVER Rev B
    - JIRA: Cover Design Status
    - GitHub: Cover Design CAD

Part Instance Level
- GitHub: Box Mfg Data
- GitHub: Box Quality Data
- GitHub: Box Incoming Data
- NMC ASSBLY SN D01
  - NMC BOX SN D01
  - NMC COVER SN B01
    - GitHub: Cover Mfg Data
    - GitHub: Cover Quality Data
    - GitHub: Cover Incoming Data
Demo video #1 – Querying the digital thread for the simple avionics box assembly
Demo video #2 - Building the digital thread for the simple avionics box assembly
Model Transformations – Automatically spinning the digital thread

Generate Models (System <-> PLM)

Switch repos

**Note:**
- Options to generate SysML dependencies and trace links are not selected in Syndea settings.
- Finished generating block structure and creating connections for part DDS Middleware.
Digital Thread as a conduit for information flow

Requirements (SE -> ME)

Mass properties (ME -> SE)
Tracking changes in connected models in the digital thread

<table>
<thead>
<tr>
<th>Conn ID</th>
<th>Source</th>
<th>Target</th>
<th>Latest Target</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>e3f03...</td>
<td>Unmanned Aerial Vehicle</td>
<td>000464/C;2-Unmanned Aerial Vehicle</td>
<td>000464/C;2-Unmanned Aerial Vehicle</td>
<td>The block Unmanned Aerial Vehicle...</td>
</tr>
<tr>
<td></td>
<td>wimax module : WiMax Module</td>
<td>wimax module : 000472/A;1-WiMa...</td>
<td>wimax module : 000472/A;1-WiMax M...</td>
<td>Part property wimax module and...</td>
</tr>
<tr>
<td></td>
<td>visual camera : Visual Camera</td>
<td>visual camera : 000475/A;1-Visual ...</td>
<td>visual camera : 000475/A;1-Visual Ca...</td>
<td>Part property visual camera and...</td>
</tr>
<tr>
<td></td>
<td>trackers : Sensor</td>
<td>thermal camera : 000476/A;1-Ther...</td>
<td>thermal camera : 000476/A;1-Thermal...</td>
<td>Part property tracker has no co...</td>
</tr>
<tr>
<td></td>
<td>thermal camera : Thermal Camera</td>
<td>software : 000487/B;1-Software S...</td>
<td>software : 000487/B;1-Software Syst...</td>
<td>Part property thermal camera an...</td>
</tr>
<tr>
<td></td>
<td>software : Software System</td>
<td>payload controller : Payload Controller</td>
<td>payload controller : 000470/A;1-P...</td>
<td>Part property payload controller...</td>
</tr>
<tr>
<td></td>
<td>payload controller : Payload Controller</td>
<td>modem : 000474/A;1-Spread Spec...</td>
<td>modem : 000474/A;1-Spread Spectrum...</td>
<td>Part property modem and part o...</td>
</tr>
<tr>
<td></td>
<td>ir detector : Wide Angle IR Detector</td>
<td>gprs : 000466/A;1-GPS</td>
<td>gprs : 000466/A;1-GPS</td>
<td>Part property gprs and part occu...</td>
</tr>
<tr>
<td></td>
<td>gprs module : GPRS UMTS Module</td>
<td>flight controller : Flight Controller</td>
<td>flight controller : 000469/A;1-Flight C...</td>
<td>Part property flight controller an...</td>
</tr>
</tbody>
</table>


Ready 11:56:55 AM 549M of 735M

20
Total System Model – A snapshot of the digital thread

2012-07-12, 1000h US ET

Connections based on Reference, Data Map, Function Wrap, Model Transform, and Composite patterns

Connect architecture model (SysML) with domain-specific models

Total System Model (TSM) as a digital blueprint of the system connecting models across disciplines, tools, and version-management systems

Goal: Seamless traceability between disciplines across the system lifecycle
Total System Model maturing though the lifecycle

TSM evolves as each of the version-managed models evolve
Handler System – A registry for all participants in the digital thread

• Provides a unique registry for all artifacts in the digital thread (similar to DOI)
• Global ID system to identify artifacts (part, sensor, data, machines, ...)
  • chain of addresses {A1, A2, A3, ...} similar to postal address. Addresses may be URIs.
• Basic meta-data for each artifact (full artifact data in native repository – PLM, ALM,...)
• Uses Handle.Net system

MFG.IO

PEOPLE

MACHINES

FEDERATED DATA

THINGS

SYSTEM INTEGRATION
Example entry in the Handle System for a Heat Sink part

Handler system
https://hdl.mfg.io

Entry for the Heat Sink part
API for the digital thread

- REST/HTTP API to access data in the digital thread, such as:
  - Repositories, projects, model elements, and connections
  - Query capabilities to search for connections given type, source, target, etc. (basic graph navigation)
  - Foundation for new apps that can be built to access, analyze, enhance the digital thread
Tools used in this proof-of-concept (POC)

System Lifecycle Handler (POC Implementation)

Syndeia

Handle.net

uses

connects to

Neo4j

Jama

GitHub

JIRA

NX, SolidWorks

See www.syndeia.com for a complete list of capabilities
Open Standards and APIs used in this POC

- SysML
- MTConnect
- QIF
- JSON
- REST/HTTP
- OSLC
- OAuth
- JDBC
- Other relevant open standards – STEP, FMI
- Native APIs, and Multiple open source Apache and Google libraries
Next Steps

• Common schema(s) for the artifacts and relationships in the digital thread
• Library of queries (FAQs) for the digital thread
• Tracking active lifecycle states – design, make, ops, service
• Explore multi-level change management scenarios, e.g. replay cascading changes that may happen if one artifact changes
• Test suites for V&V of the digital thread, automated testing and release builds (Technical Data Packages) of the digital thread
Questions and Comments

Manas Bajaj, PhD -- manas.bajaj@intercax.com
Dirk Zwemer, PhD -- dirk.zwemer@intercax.com
Intercax

Thomas Hedberg -- thomas.hedberg@nist.gov
NIST