Systems of Systems Engineering and the Internet of Things

Or

“Your IoT App is only as good as it’s least stable interface”

GJ Bleakley (graham.bleakley@uk.ibm.com)
M Hause (mhause@ptc.com)
• Introduction
  – What are systems of systems
  – The IOT as a special class of Systems of System
  – Perceived approach to developing IOT Systems

• Architecture and Engineering the I/IoT
  – Challenges of I/IoT
  – IIC Reference Architecture
  – UAF
• A SoS is an integration of a finite number of constituent systems which are independent and operatable, and which are networked together for a period of time to achieve a certain higher goal. (Jamshidi 2009)
• System Engineering is like specifying a building, e.g. a swimming pool

• Enterprise Architecture is like urban planning

  – Need for the swimming pool driven by council/government health objectives
    • The Swimming Pool is only part of the solution
    • Other means to reach the objective, e.g., running tracks, cycle lanes
    • Need to integrate these solutions into existing services
    • Provide service infrastructure to support and maintain them

• EA is about managing and developing Systems of Systems
• Trip Preparations
• City traffic
  – Interactions with other autonomous cars
  – Interactions with pedestrians
  – Overtaking Situations
• Autonomous Parking
• Public Driverless Taxi
• PDL back to the airport
• Private Car comes home
• Billing
• **Operational independence**
  – The constituent systems can operate independently from the SoS and other systems.

• **Managerial independence**
  – The constituent devices and the infrastructure are managed by different entities

• **Evolutionary development**
  – The SoS evolves over time.
  – With every change the overall behaviour of the SoS will change
  – Interfaces need to be maintained and kept consistent

• **Emergent behaviour**
  – When the SoS changes, new emergent behaviour can be observed (e.g., changes in the traffic flow when new streets are opened).

• **Geographic distribution**
  – The degree to which interaction and behaviour of the system is widely spread or localized
Classification of SoS

- **Directed**
  - The SoS is created and managed to fulfill specific purposes. The constituent systems are subordinated to the SoS.

- **Acknowledged**
  - The SoS has recognized objectives, a designated manager, and resources for the SoS.
  - The constituent systems retain their independence.
  - Changes in the systems are based on cooperative agreements

- **Collaborative**
  - The component systems interact voluntarily to fulfill agreed upon central purposes.

- **Virtual** *(Basically, the internet)*
  - The SoS lacks a central management authority and a centrally agreed upon purpose for the SoS.
  - The SoS must rely on invisible mechanisms to maintain it.
Types of SOS and focus of the EU DANSE project

¹Taken from the DANSE EU project
US DoD System of Systems

- SoS are typically composed of existing systems which are identified to address new or emerging user needs, often while these systems continue to support original users.
- SoS evolution is based on changes in these constituent systems, making these systems the essential building blocks of SoS.

Diagram: Diagram illustrates the process of translating capability objectives, monitoring and assessing changes, understanding systems and relationships, developing and evolving SoS architecture, assessing SoS performance, addressing requirements and solution options, and orchestrating upgrades to SoS. Systems are SoS building blocks.
"The changing nature of products is disrupting value chains, forcing companies to rethink and retool nearly everything they do internally."
Common perception of the Internet of Things

Households connect to the internet

Collections of things

Connect to the cloud

RFID

Digital Information

Security

Large Collection bulk of info

Everything to do with the internet

All communications

Source: IDC’s 2014 Internet of Things Survey
What the IOT really is?
Connected car example

• Designing and connecting devices
• Adding intelligence at the appropriate levels
• Design the back end systems
  – Analytics
  – Decision Making
• Optimize the overall systems architecture
• Understanding and using the information gathered

Monetizing the information that you have gathered
Example “Connected Washing Machine”  
Perspective of white goods manufacturer

- Reduce energy costs by connecting your washing machine to the internet
- Let your washing machine decide when to go through its cycle based upon cheapest energy tariff!

- Works for a few devices but not scalable
  - Think of a city of 20 Million People (Seoul)
- Applications like this can have powerful socio-economic effects
  - Go from Consumer to Industrial/Commercial domain
“Connected Washing Machine” realistic example

- Much more complex architecture
- No single owner or beneficiary of the application
- Complex set of partnership agreements
- Highly federated and distributed
- Need for the system to be engineered

Washing Machines

1-Millions

White Goods companies
11 Unique challenges for the IIOT

1. Ownership - No one owns "all of it" ...

2. Social, Pervasive and has to add value - no one is going to pay for a subscription to a service if it doesn’t
   – work very well,
   – offer value, or
   – fit into their lives without them having to really think about it

3. Complexity and dependency magnified

4. Security, safety and industry standards magnified
11 Unique challenges for the IIOT

5. Impact on development and implementation and testing –
   – Working with other partners to deliver complete solution.
   – Increases burden on development of (devices, infrastructure and services, etc.).
   – Device, services, etc. need to be upgradable (needs to service based).
   – Work either connected or not connected, be up 24/7, never fail, have redundancy (when safety implications), etc....

6. Time to market.

7. Always on, always connected, has to always work without user interaction,
   – Consumers, users, and buyers of such systems will demand it
   – ... e.g. turn the 311,000 traffic signals in the USA into smart traffic lights. We don’t want to have to manually maintain these?
11 Unique challenges for the IIOT

8. Highly diverse use cases
   – We can only begin to imagine what’s possible

9. Traditional development practices don’t scale to these types of systems

10. Foray of new business relationships -- that will need to be properly managed

11. Volume of data that will need to be;
   – Managed
   – Analyzed,
   – Processed, etc.
So, what is the solution?
Picture worth a thousand words?
IIC Framework

Biz View
- A Model
- Usage View
- A Model
- Another Model

Usage View
- B Model
- Component
- Interaction
- Sequence

Functional View
- B Model
- Component
- Interaction
- Sequence

Business Viewpoint
- Biz values, objectives & capabilities
- Usage scenario/activities

Usage Viewpoint
- Usage scenario/activities

Functional Viewpoint
- Functional decomp & structure
- Interfaces & interactions

Implementation Viewpoint
- Functional entities to technologies mapping

Contributors:
- Biz decision makers
- System Engineers
- Product Managers
- System Architects
- Architects
- Engineers
- Developers
- Integrators
- Deployment
- Operations
IIC Reference Views

• Business View
  – Enterprise Architecture
  – Systems of systems Architecting

• Usage and Functional View
  – Systems and SoS engineering

• Implementation View
  – Analytics
  – Software development
    • Embedded Software
    • Services
  – Cloud
MBSE is a model-based approach to Systems Engineering, typically applying the SysML modeling language to deal with system complexity and enabling unambiguous communication amongst interested parties.

What is Model Based System Engineering?

- **Software** (Device, Cloud, Mobile)
- **System**
- **Electrical**
- **Mechanical**
- **Manufacturing**
- **Service**
- **Marketing**
- **Management**
- **Analysis**
What is the UAF?

• The Unified Architecture Framework® (UAF®)
  – Is a generic and commercially orientated architecture framework.
  – Flexible framework that can be customized to different domains
  – Defines Enterprise Architecture (Systems of Systems)

• A standard framework for defining many different aspects of complex architectures

• Supports an MBSE approach based on SysML

• Same pattern applied across different stakeholder domains
What is the UAF?

• Rationale for the UAF
  – No standardized frameworks for MBSE
  – Integration with existing OMG standards, e.g. SysML, UML
  – Industry driven, Tool vendor supported
  – Implemented in most popular modeling tools: IBM Rhapsody, No Magic MagicDraw, PTC Integrity Modeler

• Supports BPMN 2.0

• Support for Additional Viewpoints beyond those defined in DoDAF, MODAF, NAF
  – Human Systems Integration (HSI)
  – Security views
  – Ongoing work for Threat/Risk Analysis
<table>
<thead>
<tr>
<th>Category</th>
<th>Taxonomy</th>
<th>Structure</th>
<th>Connectivity</th>
<th>Processes</th>
<th>States</th>
<th>Interaction Scenarios</th>
<th>Information If</th>
<th>Parameters Pm</th>
<th>Constraints Ct</th>
<th>Roadmap Rm</th>
<th>Traceability Tr</th>
</tr>
</thead>
<tbody>
<tr>
<td>Metadata Md</td>
<td>Metadata Taxonomy Md-Tx</td>
<td>Architecture Viewpoints Md-Sr</td>
<td>Metadata Connectivity Md-Cn</td>
<td>Metadata Processes Md-Pr</td>
<td>-</td>
<td>-</td>
<td></td>
<td>Metadata Constraints Md-Ct</td>
<td>Strategic Deployment St-Rm</td>
<td>Strategic Traceability St-Tr</td>
<td></td>
</tr>
<tr>
<td>Strategic St</td>
<td>Strategic Taxonomy St-Tx</td>
<td>Strategic Structure St-Sr</td>
<td>Strategic Connectivity St-Cn</td>
<td>-</td>
<td>Strategic States St-St</td>
<td>-</td>
<td></td>
<td>Strategic Constraints St-Ct</td>
<td>Strategic Phasing St-Rm</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Services Sv</td>
<td>Service Taxonomy Sv-Tx</td>
<td>Service Structure Sv-Sr</td>
<td>Service Connectivity Sv-Cn</td>
<td>Service Processes Sv-Pr</td>
<td>Service States Sv-St</td>
<td>Service Interaction Scenarios Sv-Is</td>
<td></td>
<td>Service Constraints Sv-Ct</td>
<td>Service Roadmap Sv-Rm</td>
<td>Service Traceability Sv-Tr</td>
<td></td>
</tr>
<tr>
<td>Personnel Pr</td>
<td>Personnel Taxonomy Pr-Tx</td>
<td>Personnel Structure Pr-Sr</td>
<td>Personnel Connectivity Pr-Cn</td>
<td>Personnel Processes Pr-Pr</td>
<td>Personnel States Pr-St</td>
<td>Personnel Interaction Scenarios Pr-Is</td>
<td></td>
<td>Personnel Constraints Pr Ct</td>
<td>Personnel Availability Pr-Rm</td>
<td>Personnel Traceability Pr-Tr</td>
<td></td>
</tr>
<tr>
<td>Security Sc</td>
<td>Security Taxonomy Sc-Tx</td>
<td>Security Structure Sc-Sr</td>
<td>Security Connectivity Sc-Cn</td>
<td>Security Processes Sc-Pr</td>
<td>-</td>
<td>-</td>
<td></td>
<td>Security Constraints Sc-Ct</td>
<td>-</td>
<td>-</td>
<td></td>
</tr>
<tr>
<td>Projects PJ</td>
<td>Project Taxonomy PJ-Tx</td>
<td>Project Structure PJ-Sr</td>
<td>Project Connectivity PJ-Cn</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td></td>
<td>Project Roadmap PJ-Rm</td>
<td>Project Traceability PJ-Tr</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Standards Sd</td>
<td>Standard Taxonomy Sd-Tx</td>
<td>Standards Structure Sd-Sr</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td></td>
<td>Standards Roadmap Sr-Rm</td>
<td>Standards Traceability Sr-Tr</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Actuals Resources Ar</td>
<td>Actual Resources Structure Ar-Sr</td>
<td>Actual Resources Connectivity Ar-Cn</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td></td>
<td>Parametric Execution/Evaluation Ar-Cn</td>
<td>-</td>
<td>-</td>
<td></td>
</tr>
</tbody>
</table>

**Conceptual Data Model, Environment Pm-En**

**Logical Data Model, Physical Schema, Real World Results**

**Measurement Pm-Me**

**Dictionary * Dc**

**Summary & Overview SmOv**

**Requirements Rq**
Benefits of UAF

• Executable Architectures
  – Dynamic simulation/execution
    • Verify behavior,
      – State, Activity level, message sequences
    • Verify interfaces
  – Computational Analysis
    • Parametric analysis
      – Trade studies and Architecture optimization

• Customizable approach
  – Business motivation (monetizing the information)
  – Cloud and edge Architecture
  – Dev-Ops
  – Compliance
• Defense:
  – Used by DOD and its contractors on various MBSE and IT projects
  – Being picked up outside of the US
    • Used in Europe, Australia, Asia, S. America

• Industry and Government (external to Defense):
  – European research projects (DANSE, Compass)
  – Starting to be looked at by European industrial companies familiar with MBSE
    – NASA, CACI, etc.
  – Also used for
    – modeling business processes,
    – information systems architectures
UAF Support for the IIC

What does UAF provide?
IIC Framework

**Business Viewpoint**
- Biz values, objectives & capabilities

**Usage Viewpoint**
- Usage scenario/activities

**Functional Viewpoint**
- Functional decomp & structure
- Interfaces & interactions

**Implementation Viewpoint**
- Functional entities to technologies mapping

**Biz View**
- A Model

**Usage View**
- A Model
- Another Model

**Functional View**
- Component
- Interaction
- Sequence

**Implementation View**
- Map
- Map

**Roles**
- Biz decision makers
- System Engineers
- Product Managers
- System Architects
- Architects
- Engineers
- Developers
- Integrators
- Deployment
- Operations
Business, Usage and Functional Viewpoint

Key Objectives -> Key Capabilities

Business Viewpoint (Strategic)

Measures

Key Capabilities -> Aspect

Functional Viewpoint

Roles

Decompose

Activities

Sub-Roles

Decompose

Activities

Usage Viewpoint (Operational)

Interact with

Decompose

Group and carry out

Allocated to

Implementation Viewpoint (Systems and Services)

Functions

Function

Function

Function

Function

HW technology

SW technology

Analytics

Services

Security

Control
Implementing the IIC-AF with UAF

Behavioral Analysis

Business Views -> Strategic Views

Usage Views -> Operational Views

Functional Views -> Functional Aspects of Systems View

Implementation Views -> Structural aspects of Systems Views

IoT cloud

HW-SW design

Analysis & Exploitation

Behavior Specifications

IoT cloud
UAF Support for the IIC

What is missing?
How should it evolve?
• **UAF** has the potential to improve communication, collaboration and interoperability for the IOT

• **Grid approach** allows customization of framework to the IIC Architecture Framework
  – Already Complete
  – Tool implementations under way

• **Provides**
  – Support for SoS
  – Support for the IoT
  – Support for the IIC

• **Call industrial partners to support**
  – Use Cases
  – Test Cases
  – Examples
Thank You!

Graham Bleakley, PhD
IBM
Matthew Hause
PTC

E-mail:
graham.bleakley@uk.ibm.com
mhause@ptc.com