Web Service Development Framework and API Management

Strategy and Best Practices

Yong Cao
The Boeing Company
Vision: Service and Web APIs
Vision: Service Deployment

Global Product Data Interoperability Summit | 2017

Clients

Cloud (PCF, Azure, AWS)

API Gateway

Service Farm

App Farm

API Gateway

PLM

MOM

ERP

Others...

COTS (On-Prem)
Motivation / Requirement

Application (COTS or Boeing)

Web API

Web API Service

“GET http://api.web.boeing.com/airplanes/airplaneID”

Loosely Coupled Integration

Application (COTS or Boeing)

Independent
- Build, Test, Deployment (Agile)

Persistent
- “One Stop Shopping”
- Life Cycle Management

Universal
- Platform, OS, Device Agnostic
- Simple, User Friendly

Fault Tolerant
- Scalable
- Universal
# Boeing.TearDownBRT.Api

<table>
<thead>
<tr>
<th>Method</th>
<th>Endpoint</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GET</td>
<td>/api/tdl/assembly</td>
<td>Gets all the AssemblyItem objects.</td>
</tr>
<tr>
<td>GET</td>
<td>/api/tdl/assembly/{assemblyId}</td>
<td>Gets an AssemblyItem object.</td>
</tr>
<tr>
<td>GET</td>
<td>/api/tdl/assembly/{assemblyId}/photos</td>
<td>Gets all the PhotoItem objects for an Assembly.</td>
</tr>
<tr>
<td>GET</td>
<td>/api/tdl/assembly/{assemblyId}/children</td>
<td>Gets child AssemblyItem objects for an Assembly.</td>
</tr>
<tr>
<td>POST</td>
<td>/api/tdl/photos</td>
<td>Creates a PhotoItem</td>
</tr>
<tr>
<td>GET</td>
<td>/api/tdl/photos/{photoId}</td>
<td>Gets a PhotoItem object.</td>
</tr>
<tr>
<td>GET</td>
<td>/api/tdl/teardown</td>
<td>Gets all TeardownItem objects.</td>
</tr>
<tr>
<td>GET</td>
<td>/api/tdl/teardown/{teardownId}</td>
<td>Gets a TeardownItem objects.</td>
</tr>
</tbody>
</table>

# Boeing.SLATE

<table>
<thead>
<tr>
<th>Method</th>
<th>Endpoint</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GET</td>
<td>/api/model</td>
<td>Gets all the models available in the database.</td>
</tr>
<tr>
<td>GET</td>
<td>/api/model/{model}</td>
<td>Gets all the top-level nodes for the specified model.</td>
</tr>
<tr>
<td>GET</td>
<td>/api/model/{model}/node/{nodeId}</td>
<td>Gets all the child nodes for the specified parent node.</td>
</tr>
<tr>
<td>GET</td>
<td>/api/model/{model}/node/{nodeId}/details</td>
<td>Get all the detailed information for the specified node id.</td>
</tr>
<tr>
<td>GET</td>
<td>/api/model/{model}/node/{nodeId}/properties</td>
<td>Gets all the properties for a specified node id.</td>
</tr>
<tr>
<td>GET</td>
<td>/api/model/{model}/node/{nodeId}/references</td>
<td>Gets all the references for a specified node id.</td>
</tr>
<tr>
<td>GET</td>
<td>/api/model/{model}/node/{nodeId}/links</td>
<td>Gets all the links for a specified node id.</td>
</tr>
</tbody>
</table>
Wrap COTS application around with **Service APIs**.

Addressing a functional gap
SOA Patterns: On-Demand Data Migration

Global Product Data Interoperability Summit | 2017

1. Data Request

2. On-Demand Data Request

3. Send Data Back

Existing PLM

Data Lake

New PLM

User Clients

Downstream Applications

Data Service (Push)

Data Service (Pull)
SOA Pattern: Transitional Architecture
SOA Pattern: Transitional Architecture

Current Implementation

Existing App

Legacy Interface

Transition

Web APIs

Temp Data Store

Existing App

Legacy Interface

Legacy Interface

Web APIs

Legacy Interface
SOA Pattern: Transitional Architecture

Current Implementation

Legacy Interface

Existing App

Existing App

Legacy Interface

Legacy Interface

Web APIs

Web APIs

Transition

Temp Data Store

Web APIs

2CES Final

PLM/MOM

Web APIs

(For Functional Gap)

Existing App

Web APIs

Existing App

Legacy Interface

Existing App

Legacy Interface

Legacy Interface
Hybrid Integration / Deployment

- Security
- Monitoring/logging
- SLA

- Queriable API
- Protocol Translation

PaaS (Cloud) Web Service
IaaS Web Service
On_Prem Web Service
SaaS Web Service

API Gateway
API Management

Service Consumer
Service Consumer
Service Consumer
Service Consumer
Demo: IBM API Connect

Innovate with our APIs

Welcome to our API portal where you will find a great selection of APIs for your awesome innovative apps.
Service Oriented Architecture (.Net)

Global Product Data Interoperability Summit | 2017

- Service Consumer
- Web Service API
- Business logic
- Data Access Layer
- Contract Objects
- DTOs and Entities
- Data Sources
Service Oriented Architecture (.Net)

Microservices Architecture

- Web API
  - Data Microservice
    - SQL
  - Data Microservice
    - MongoDB
  - Data Microservice
    - Oracle

Service Consumer

- Web Service API
- Client Proxy
- Contract Objects

WCF Service Host

Business logic

Data Access Layer

Data Sources

DTOs and Entities
Service Oriented Architecture (.Net)

Microservices Architecture

- Web API
- Business Logic
- Data Microservice
  - SQL
- Data Microservice
  - MongoDB
- Data Microservice
  - Oracle
Product System SOA Integration Team

Global Product Data Interoperability Summit | 2017

• Motivation: “Jump Start” for every new integration in 2CES
  • Enterprise SOA Standard
  • Software Development Best Practices
  • DevOps
**Product System SOA Team SharePoint Site**

**Global Product Data Interoperability Summit | 2017**

**Link:** [https://collab2.web.boeing.com/sites/PS_SOA/SitePages/Home.aspx](https://collab2.web.boeing.com/sites/PS_SOA/SitePages/Home.aspx)

**Product Systems SOA**

**Product Systems - Service Oriented Architecture**

**Vision:**

- Legacy Apps
- COTS
- Web
- APIs
- Mobile
- IoT
- Analytics/BI

**Goals:**

Implement a SOA approach to address application/system integration issue for 2CES effort.

**Tasks and Steps:**

1. Identify architecture patterns for system integration in 2CES to-be architecture.
2. Develop SOA use-cases established in step 1.
3. Develop software architecture patterns using SOA approach to address the use cases.
4. Implement general SOA integration solutions with key technical stacks as a reusable library, source code packages, and deployable services.
5. Test the integration solution using a prototyping approach on several typical and important 2CES projects (PLM and MOM).
6. Create support and governance processes to promote these reusable solution for all (new) 2CES projects.

**Deliverables:**

1. SOA architecture patterns for system integrations to support 2CES/BC2020 (Collaborate with BC2020/2CES Team)

**Current Projects**

- MSCI - SkyLight
- Teardown Lab
- WCM Plant centric
- Performance Testing
- Customer Config

**The Team**

- Kumar, Hemanth
  Program Sponsor

- Cao, Yong
  Lead Architect

- Davis, Lana S
  Project Manager

- Suyam Prakash
  Technical Architect
DevOp: .Net

Global Product Data Interoperability Summit | 2017

[Diagram of DevOps process involving Visual Studio, Git, TFS Build, NuGet, SonarQube, and Microsoft Release Management]
SOA Template Source Code

Global Product Data Interoperability Summit | 2017

GITLAB Sources

• **Template Project**

• **Reuse Package Projects**
  - NuGet Namespace Prefix: Boeing.Reuse.SOA

• **SOA Gen**
SOA Gen Demo

Global Product Data Interoperability Summit | 2017

SOA Generator

https://soagen.apps.pcfpre-phx.cloud.boeing.com/home
Programming Revolution:

- Procedural (e.g. C, Fortran)
- Object Oriented
- Component
- Service
Technical Advantages of SOA

Programming Revolution:
- Procedural (e.g. C, Fortran)
- Object Oriented
- Component
- Service
Technical Advantages of SOA

Programming Revolution:
- Procedural
- **Object Oriented (C++)**
- Component
- Service

Pros: Programmability
Cons: Object Dependency
Hard to maintain
Programming Revolution:
- Procedural
- Object Oriented
- **Component (Java, C#)**
- Service

Pros: Reduced Dependency
Cons: Build Together  Hard to build, test and deploy
Programming Revolution:
• Procedural
• Object Oriented
• Component
• Service (Cobra, Web)

Pros: *Decoupled* service component (Swap any service component independently)