Using Web Services for MES Integration

Chris Anderson – ATK  
Sr. Manufacturing Analyst

Phil Zerr – Callisto Integration  
Sr. Systems Engineer

October 16, 2013
Agenda

• About ATK

• Evolution of Shop Floor Communication

• Web Service Environment

• ArchestrA Integration with Web Services
They Made Me Do it

- Headquartered in Arlington, VA, with approximately 14,000 employees
- $4.8B in sales
- $7.8B in backlog
- Company/customer funded R&D $603.0M
- More than 60 facilities in 21 states, Puerto Rico and Internationally
- Representatives in more than 50 countries throughout the world

Sales by Customer
- Commercial/International 33%
- U.S. ARMY 29%
- NASA 10%
- U.S. Navy 13%
- Other U.S. Government 9%
- U.S. Air Force 6%
- U.S. Army 29%
- Other U.S. Government 9%
- NASA 10%
- U.S. Navy 13%
- Commercial/International 33%
- U.S. Air Force 6%

Sales by Contract Type
- Prime Contractor 66%
- Sub-Contractor 34%

Aerospace Group

Defense Group

Sporting Group
About ATK – Our Place

COMMERCIAL
- Engine Cases – Over 200 Produced
- Nacelle Structures
- Airframe Structures
- 757 Cargo Bay Stanchions – Over 10,000 Produced (from 1985 to 1994)
- 767 Springs
- A350 Stringers – Over 10,000 Produced
- A350 Frames – Over 6,000 produced

MILITARY STRUCTURES
- F-22 Fiber Placed Pivot Shafts
- F-35 Wing Skins – Over 1,000 Produced
- F-35 Fixed Skins – Over 700 Produced
- Nacelles
- C-17 Counter Balance Assembly
- A400M Wing T Stringers

MILITARY SYSTEMS
- Apertures
- Survivable Structures
- Performance Verification
- Sensor Technology
- Low Observable Technology

LAUNCH
- Launch Structures – Over 450 Produced
- Rocket Motor Cases – Over 11,000 Composite Cases Manufactured. With over 5,600 flown successfully
- Composite Overwrapped Pressure Vessels (COPV)
About ATK – Our Place

KEY FEATURES
Located in Clearfield, Utah
- Floor Space: 615,000 sq ft
- 100,000 sq ft Clean Room Capacity
- 15,000 sq ft Office Space
- 2,500 sq ft Freezer Capacity
- Automated Stiffener Forming Capacity
- Automated Linear Stringer Machines
- Automated Radial Frame Machines
- Engine Case Wrap Capacity
- CNC Milling Machine Capacity
- Large Diameter Autoclave Capacity
- NDI Systems - Automated Ultrasonic Inspection (AUIS)
- Water Jet and Machining Capacity
- Hot Drape Forming
KEY FEATURES
Located in Clearfield, Utah
• Floor Space: 403,000 sq ft
• Clean Rooms
• Fiber Placement Machines
• Case Wrappers
• Ovens
• CNC Milling Machines
• Lathes
• Autoclaves
• Automated Ultrasonic Inspection Systems (AUIS)
• ASFM Machines
• Pattern Cutter
• Leica Measurement Systems
• Laser/Radar Measurement Systems
• Virtek Projection Systems

~32,000 sq ft Expansion
2 New FPMs
1 New Autoclave
In the Beginning…
Shop Floor and MFG Systems

Goals
1. Keep current MES’ish Systems
2. Talk to Shop Floor machines

PLM

Oracle 10g

ePIC Data

Solumina / Impresa Data

Engineering Desk PC

EWI & MRP

Oracle 10g

Control PC

Shop Floor Machine

Barcode Scanner

Solumina Shop Floor PC
PLM, EWI, Control PC

- ePIC
  - Part Program
  - Name
  - Revision
  - Location

- Engineering Desk PC

- Solumina
  - Work Order
  - Part Program
  - Name
  - Revision

- Control PC

- Machine Designation
  - Part Programs
  - Output Data Files
  - Output Reports

- 700+ Part Numbers
Prototype Phase

How the customer explained it
How the project leader understood it
How the engineer designed it
How the programmer wrote it
How the sales executive described it

How the project was documented
What operations installed
How the customer was billed
How the helpdesk supported it
What the customer really needed
Take One
- Machine controls say take a hike
- Use EWI for interface
Take Two
- Bug machine controls until they give in
- Use Local DB for interface
Prototype Phase

Take Three?
- IT states we’re nuts
- Starting to panic
We’re going to…
The Aha Moment…
What’s a “Web Service”?

- Web-based software solution for retrieving or sending data
  - Examples:
    - Retrieve weather information for a given city, zip-code, or GIS point
    - Retrieve Job/Work Order data from a MES system
    - Update an ERP system with actual consumption/production information
  - Typically firewall-friendly (single “port”, usually same as standard web traffic (80))
  - Provides a “simple” way to pull or push information between software systems
Advantages

• Can be Java or IIS (needs to be SOAP compliant)

• If the definition (inputs/outputs) is known early, ArchestrA side programing can start immediately

• Any other system involved can call the same web service/code base

• Provides and abstract layer for data that minimizes re-coding in ArchestrA

• Can be swapped out for bus style integration with minimal ArchestrA coding
The **Web Services Description Language** is an **XML-based** **interface description language** that is used for describing the functionality offered by a **web service**. A WSDL description of a web service (also referred to as a **WSDL file**) provides a machine-readable description of how the service can be called, what parameters it expects, and what data structures it returns. It thus serves a purpose that corresponds roughly to that of a **method signature** in a programming language.
It's all about the WSDL
Recipe
- Add one heaping batch of Web Services
- 1 part specific HMI
- Add machine control DB to taste
Finally a good idea…

Well, almost
OK, Now a good idea…

Enter Wonderware
Now for the good stuff…
Apache Web Server

Why Apache
• 40+% of the internet uses it (Google said so)
• It’s free
• Did we mention it’s free?

Apache HTTP Server
Manages Requests

Apache Tomcat Web Server 1
(WSDL to Solumina/ePIC)

Apache Tomcat Web Server 2
(WSDL to Solumina/ePIC)
Server Environment
Status Manager Used for Setup and Balancing of Workers (Tomcat Servers)

JK Status Manager for ut02src1fs:8086

```
Server Version: Apache/2.2.22 (Win32) mod_jk/1.2.37
Server Time: Tue, 08 Oct 2013 13:27:22 Mountain Daylight Time
JK Version: mod_jk/1.2.37
Unix Seconds: 1381260442
```

[Read Only] [Dump] [S=Show only this worker, E=Edit worker, R=Reset worker state, T=Try worker recovery]

Listing Load Balancing Workers (2 Workers) [Hide]

```
[SER] Worker Status for MachineCluster

Type Sticky Sessions Force Sticky Sessions Retries LB Method Locking Recover Wait Time Error Escalation Time Max Reply Timeouts [Hide]
lb True False 2 Request Optimistic 60 30 0

Good Degraded Bad/Stopped Busy Max Busy Next Maintenance Last Reset [Hide]
2 0 0 0 5 58/120 548486

Balancer Members [Hide]

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Hostname</th>
<th>Address:Port</th>
<th>Connection Pool Timeout</th>
<th>Connect Timeout</th>
<th>Prepost Timeout</th>
<th>Post</th>
<th>Time</th>
<th>Reply Timeout</th>
<th>Retries</th>
<th>Recovery Options</th>
<th>Max Packet Size</th>
<th>[Hide]</th>
</tr>
</thead>
<tbody>
<tr>
<td>NodeA1</td>
<td>actok</td>
<td>aip13</td>
<td>ut02svatoma1 10.82.16.111:8009</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>2</td>
<td>0</td>
<td>8192</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>NodeA2</td>
<td>actok</td>
<td>aip13</td>
<td>ut02svatoma2 10.82.16.112:8009</td>
<td>0</td>
<td>0</td>
<td>2</td>
<td>0</td>
<td>0</td>
<td>8192</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Name</th>
<th>Act</th>
<th>State</th>
<th>D</th>
<th>F</th>
<th>M</th>
<th>V</th>
<th>Acc</th>
<th>Sess</th>
<th>Err</th>
<th>CE</th>
<th>RE</th>
<th>Wr</th>
<th>Rd</th>
<th>Busy</th>
<th>Max</th>
<th>Con</th>
<th>Route</th>
<th>RR</th>
<th>Cd</th>
<th>Rs</th>
<th>LR</th>
<th>LE</th>
<th>[Hide]</th>
</tr>
</thead>
<tbody>
<tr>
<td>NodeA1</td>
<td>actok</td>
<td>OK</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>25677 (0/sec)</td>
<td>25677 (0/sec)</td>
<td>0</td>
<td>0</td>
<td>25M</td>
<td>47</td>
<td>73M</td>
<td>138</td>
<td>0</td>
<td>3</td>
<td>3</td>
<td>NodeA1</td>
<td>0</td>
<td>0</td>
<td>548486</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>NodeA2</td>
<td>actok</td>
<td>OK</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>25647 (0/sec)</td>
<td>25647 (0/sec)</td>
<td>3</td>
<td>0</td>
<td>26M</td>
<td>50</td>
<td>68M</td>
<td>130</td>
<td>0</td>
<td>3</td>
<td>3</td>
<td>NodeA2</td>
<td>0</td>
<td>0</td>
<td>548486</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>
```

[Hide]
Tomcat Application Manager

Application Manager used for setup and management of Web Service methods

![Tomcat Web Application Manager](image)

<table>
<thead>
<tr>
<th>Applications</th>
<th>Manager</th>
<th>HTML Manager Help</th>
<th>Manager Help</th>
<th>Server Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>Path</td>
<td>Version</td>
<td>Display Name</td>
<td>Running</td>
<td>Sessions</td>
</tr>
<tr>
<td>/</td>
<td>None specified</td>
<td>Welcome to Tomcat</td>
<td>true</td>
<td>0</td>
</tr>
<tr>
<td>StockExchangeWS</td>
<td>None specified</td>
<td>True</td>
<td>true</td>
<td>0</td>
</tr>
<tr>
<td>Webapp</td>
<td>None specified</td>
<td>Tomcat Documentation</td>
<td>true</td>
<td>0</td>
</tr>
<tr>
<td>Examples</td>
<td>None specified</td>
<td>Servlet and JSP Examples</td>
<td>true</td>
<td>0</td>
</tr>
<tr>
<td>host-manager</td>
<td>None specified</td>
<td>Tomcat Host Manager Application</td>
<td>true</td>
<td>0</td>
</tr>
<tr>
<td>manager</td>
<td>None specified</td>
<td>Tomcat Manager Application</td>
<td>true</td>
<td>1</td>
</tr>
</tbody>
</table>

Deploy directory or WAR file located on server
And now for the Smart Guy…

Phil Zerr

Callisto Integration
Create a custom DLL to wrap web service calls

- Add a Web Reference (Service Reference) to the project
Adding the Web Reference creates a proxy object that exposes all public web service methods.
Create public methods to wrap web service methods.

```java
public static Boolean getWorkOrder(out string outResult, string partNumber, string serialNumber)
{
    Boolean result = false;
    try
    {
        JavaMachineWS.MachineWSService ws = new JavaMachineWS.MachineWSService();
        outResult = ws.getWorkOrder(partNumber, serialNumber);
        result = true;
    }
    catch (Exception ex)
    {
        outResult = ex.Message;
    }
    return result;
}
```
Import the custom DLL into the Galaxy as a Script Function Library
Call public methods from custom DLL in object or graphic scripts

```vbnet
DIM outResult As System.String;
DIM partNumber As System.String;
DIM serialNumber As System.String;

partNumber = Me.getWorkOrder.partNumber;
serialNumber = Me.getWorkOrder.serialNumber;

IF partNumber <> "" AND serialNumber <> "" THEN
    IF MachineInterfaceWW.HMachineInterface.getWorkOrder(outResult, partNumber, serialNumber) THEN
        IF (outResult <> null) THEN
            Me.getWorkOrder.Result = "Success";
            Me.getWorkOrder.WorkOrder = outResult;
        ELSE
            LogMessage("Invalid Return Data");
            Me.getWorkOrder.Result = "Invalid Return Data";
            Me.getWorkOrder.WorkOrder = "";
        ENDIF;
    ELSE
        LogMessage("Call Failed: " + outResult);
        Me.getWorkOrder.Result = "Call Failed: " + outResult;
    ENDIF;
ELSE
    LogMessage("Invalid Input - partNumber: " + partNumber + " serialNumber: " + serialNumber);
    Me.getWorkOrder.Result = "Invalid Input - partNumber: " + partNumber + " serialNumber: " + serialNumber;
ENDIF;
```
Bottom Line

- Web Services CAN and ARE used within Application Server solutions

- Best Practices:
  - ALWAYS “wrap” .NET methods that can throw exceptions in a custom library (DLL)
  - Have these wrapper methods/functions return appropriate data to take necessary action (success or failure)
  - Anytime you are accessing a remote resource, try to do so using Asynchronous Application Server scripts
  - Plan before you code!
    - Using Boolean returns or return codes?
    - What failures could occur and how will it affect your object’s execution?
    - How will you notify appropriate personnel when “failures” occur, and how will they know what action to take?
Lessons Learned

• Provide public properties in custom DLL to set web service connection details
  – This allows for easy configuration to point to development, qual, or production environments

• Return standard types for data
  – Datasets, strings, float, etc.

• When returning datasets, return an empty dataset rather than NULL if no rows
  – Preserves column headers for grid display
Lessons Learned

• Query data for graphics in the graphics.

• Use ArchestrA objects for writing data.

• Reliably updating custom DLL versions (Script Function Library) can be...challenging.
Chris Anderson – ATK
Sr. Manufacturing Analyst
Chris.Anderson@atk.com

Phil Zerr – Callisto Integration
Sr. Systems Engineer
Phil.Zerr@callistointegration.com