

# aras community event

2010 International Conference

## Federation Workshop

Peter Schroer

[www.aras.com](http://www.aras.com)



Hosted by:

**MOTOROLA**

1295 E. Algonquin Road  
Schaumburg, IL 60196

# Welcome



## Workshop Goals

- Understanding of why use Federation
- Enough Knowledge of the Aras Federation Approach to Plan a Project

# Agenda



1. Build a working definition for Federation
  - Understand Federation vs. Integration
2. Discuss Federation Approach Alternatives
3. Aras Innovator Federation
4. Federation Modes
5. Discuss Pseudo-Code Examples
6. Federation Data Access Techniques
7. Additional Resources

# Definition



- ❖ PLM is never standalone: ERP, corporate document management systems, manufacturing and quality software, etc.
- ❖ Replicating data between systems is a common approach, but synchronization of edits between systems is complicated and risky
- ❖ Federation
  - ✓ Creating a pointer within PLM to data objects stored in the other business systems

# Federation vs. Integration



- ❖ We use Federation when we want to see (and manage) the objects in the other system, through the Aras user interface, and use those objects in Aras workflow and configurations
- ❖ Integration (e.g. CAD integration) is a technique for moving data from one system to another, usually triggered by user action of lifecycle transitions
- ❖ Integration is often batch oriented
- ❖ Federation is about real-time accessing and aggregating of data from multiple sources

# Definition (continued)



- ❖ Federation creates a wrapper around data objects from remote systems, and then extends their definition with PLM functionality
- ❖ We aggregate data from many sources in this way in order to:
  - Present the aggregated data in a single window for the user, e.g. a Mash-Up
  - Execute business rules, BI dashboards, or Reports against the consolidated data set
  - Add advanced PLM capabilities such as BPM workflow or Secure web interface to legacy systems

Done well, the end-user is not aware of the sources of data

# Scenarios



## ❖ Corporate doc management system

- Documents checked in to the PLM system are transferred directly to a corporate document archiving solution
- Searching and viewing documents is performed within the PLM interface, directly accessing remotely stored data.

## ❖ Cost and inventory on hand fields in ERP displayed on the PLM Part form

- Material cost may be maintained in the ERP system, but is useful information to engineers using PLM
- The PLM part form contains data from both PLM and a real-time ERP query

# Federation Approaches



## Client Based Federation

- ❖ The PLM client application initiates 2 queries, one to each system: the PLM server and the 3<sup>rd</sup> party server. The data is merged on the client
- ❖ Aras client is HTML / JS and developers can use AJAX to easily make calls from a Form to 3<sup>rd</sup> party servers
- ❖ Technique has limited usefulness beyond merged data onto a form for an aggregated view

# Federation Approaches



## Server Based Federation

- ❖ Data exchange with the other system is executed on the PLM server
- ❖ The PLM client is sent a single data set, and is not aware that core business objects have been modified with remote properties, or that the data objects are not native
- ❖ Advantage because integration of remote data objects is encapsulated on the server, eliminating client customizing
- ❖ Additional advantage that server-side business rules such as Permissions, LifeCycle, and Workflow logic can act on the 3<sup>rd</sup> party data objects

# Aras Innovator Federation



- ❖ Aras Client expects AML to be returned on every transaction - this is key to the simplicity of Federation
- ❖ No customizations normally on the client
  - As long as AML is returned, the client is happy to work with any data it receives
  - Exception is file check-in handling ( \*\* discussed later )
- ❖ Use standard Server Events to add the Federation logic
  - OnAfterGet to merge federated properties
  - OnGet to return 100% federated items

# Simple Exercise



- ❖ Create an ItemType named Test with 1 Property
- ❖ Add a Server Event OnGet for a Method that creates and returns an XML string

```
<Item type='Test' id='1234'><name>Hello</name></Item>
```

- ❖ Select this item on the TOC
- ❖ Grid will populate with the 'fake' data

## ❖ NOTES:

- Had to create an ID for each row

# Federation Modes



1. Federate either a complete ItemType or a subset of Properties on a standard managed ItemType
  
2. Federation Models
  - View only of data from the other system
  - Use remote objects in configurations and workflows
  - Full access: OnGet, OnAdd, OnDelete

# Pseudo Code for View



- ❖ Assume “MyFederation” OnGet event - VB Method
- ❖ Assume entire ItemType has been Federated
- ❖ Properties for the remote object, TOC Access, Form Views are defined in Aras Innovator
- ❖ 2 Modes:
  1. Simple Searching and Viewing
  2. Use Federated Item in Relationships or as Item Properties

For example: Part to Document Relationship with a set of Federated Documents

# Pseudo Code for View #1



- User query parameters from the Simple Search grid row are Properties on the Me object
- Check if this is a Get for one record or a set. **How?**
- Build the query for the remote system, passing the query parameters from the client
- Run the query code ( \*\* discussed next )
- Create an AML string from the query result
- Add a dummy ID for each row
- Return AML

**QUIZ: Is there an ItemType table for this item?**

# Pseudo-Code for View #2



Federated Items will be used as Item Properties,  
or either the Source or Related in a Relationship

- ✓ ItemType is not flagged as Federated this time because we do want an Aras Innovator table

**QUIZ - What properties must this table store?**

# Pseudo-Code for View #2



3 properties are required to maintain configurations

- ID
- Federated-ID
- Keyed-Name

Solution: in the OnGet method we add a function to register the data objects found via user queries, assigning a new Aras Innovator ID to each

*I use a small stored procedure to execute this efficiently*

# Integration Techniques



## ❖ Web Services

- Building an ASPX page as a wrapper
- Using Visual Studio to make a Proxy
- XMLHTTP object

## ❖ ODBC

## ❖ Commercial API from other system

## QUIZ

How to register a DLL (either API or VS Proxy) so that you can use the DLL directly in methods ?

# Other Design Considerations



- ❖ File check-in to a Federated source
  - After check-in move file from Aras vault to remote vault
  - Aras client does direct file upload
  
- Actions for Add, Update and Delete are possible
  - More important to consider permissions and access

# More Resources



## ❖ Federation test package

- Innovator - to – Innovator federation used for testing of the Server Events.

## ❖ Aras Wiki articles