The Industrial Software Revolution Begins Now
WW HMI SCADA-11 System Platform
Best Practices 1: Engineering Efficiencies

Michael Brost
Wonderware Solutions Architect
North America

mike.brost@invensys.com
Topics to be Covered

• Building a Reusable Galaxy Template
• Multi-User Development
• Modeling and Template Organization
• Efficient Scripting
• Database Integration
• System Sizing Guidelines
• Model Based Device Integration
Configuring a Template Galaxy

- Create a Base Application Server Galaxy
- Connect to the Newly Created Galaxy
Load Script Libraries

- Import Script Function Libraries
  - Base Template Library
  - Object Modeling Script Library
  - ww.nasc.btl.modeling.aaSLIB

- Importing Script Libraries First
  - Prevents Validation Errors
  - Fixing Validation Errors is Tedious
Configure Galaxy Style Library

- Quality and Status
  - Globally defines the behavior of animations to changes in data quality
  - Warning, Bad, Uncertain, Initializing
  - Read Errors
  - Communication, Configuration, Software

- Write Status
  - Pending, Operational, Security
  - Overrides for Text, Fill, Line
  - Status indicators require the use of the Status Animation
  - All are automatic and take effect immediately, no deployment required

- Recommend defining a standard for Outline Overrides as a minimum
  - Does not change the appearance of graphics
  - Outline appears around graphic element being animated
  - Preview Legend will show behavior, good documentation of standard
Configure Galaxy Style Library

- Element Styles
  - Styles are used to globally define Graphics Appearance
  - They do not have to be used in Animations
  - Styles are Named
  - Text, Fill, Line and Outline are definable
  - Additional Styles can be added if desired
  - Defined via an XML document
  - Can Be Exported and Imported

- Element Style is both a Configuration and Runtime Settable Property
Configure Alarm Priority Mapping

- Defines the range of Alarm Priorities
  - Critical, High, Medium, Low
- Which Alarms to Historize
- Image to use for Alarm Indication
- Defines which events to Historize
Import Library of Templates

- North American System Consultants: Base Template Library
  - Excellent starting point for your own Library
  - Good example of scripting techniques
  - Contains
    - Application Object Templates
    - Quick Client
    - Model and Graphic Toolset Navigation
    - Will be discussed in Detail Later in this Presentation and Following Presentations
Import Client Controls

- Historian Client Controls
- A little extra in cost but far superior to the TrendClient in the basic galaxy
- C:\Program Files (x86)\Common Files\ArchestrA\Application-Level Controls
- Application-Level Controls
  - aaHistClientTrend
  - aaHistClientQuery
- Building Block Controls
  - aaHistClientTimeRangePicker
  - aaHistClientTagPicker
  - aaHistClientSingleValueEntry
  - aaHistClientActiveDataGrid
  - aaHistClientChart
- Put in their own Graphics Toolset
- Put the TrendClient in its own toolset and hide it
Import Base Template Library

- Import BTL
- Import BTL Master and User Templates
- Import BTL Quick Client aaPKG
- Optional Import the DASABCIP Master and User Templates
- Organize the Template Toolbox after the Imports
- Expand the Read Only Templates Toolset
- Fully Expand the Wonderware Base Templates Tree
- Move the Core Device Integration Objects into the Toolset under this branch
- Delete the now Empty Device Integration toolset
- Move the System and Application Toolsets here as well
- Add any additional Standards
- Languages
- Communications Management
Save the Galaxy as a Template

- Launch the System Management Console (SMC) from the Galaxy Repository.
- Expand the Galaxy Database Manager Tree.
- Right-click on the Standards Galaxy and Backup.
- Place the Standards Galaxy cab file in the directory `C:\Program Files (x86)\ArchestrA\Framework\Bin\BackupGalaxies`.
- Repeat this for all Galaxy Repositories in your organization.
Create Two Galaxies on the Project Repository

- Template Galaxy (Templates Only No Instances)
- Project Galaxy (Templates and Instances Required by Project)

- Base both Galaxies on your Standards Galaxy

- Enable Security on Both Galaxies

- Use OS User or Group if Possible

- Use Galaxy Security if no Active Directory Server is Available

- Do Not Allow Multiple IDE's to Connect with None Security Selected

- Each Developer has their Own User ID

- Unique Checkin/Checkout

- Default Platform, Engine, Area, Security Group

- Use aaPackages to transfer templates.
Different Classes of Objects

- Infrastructure Objects
  - Objects which define the Infrastructure of the Running Galaxy
  - Node Name Specific properties Typically
  - Platforms ($WinPlatform)
    - Node Name of the Operating System to Deploy to on the Network
    - Node Name of Historian to Configure and Send Data To
      - Alarms and Events
      - Platform Attributes to be Historized
  - Application Engines ($AppEngine)
    - Node Name of Historian to Configure and Send Data To
  - Dedicated Device Integration ($OPCClient/$DDESuiteLinkClient)
    - Point to a Specific Node Name for the DAServer
  - Wonderware WorkFlow Gateway ($WorkFlowGateway)

- Infrastructure Objects are Dedicated to a Specific Galaxy’s Network
  - Should not be routinely transferred between Galaxies
  - Accidental Deployment will cause undesired results
Different Classes of Objects

• Project Specific Objects
  • Object specific to the Project/Application being created
    • Areas ($Area)
    • Application
      • Equipment, Calculation, Utility ($UserDefined)
      • AnalogDevice ($Analog Device)
      • DiscreteDevice ($DiscreteDevice) Etc...
    • RedundantDeviceIntegration ($RedundantDIObj)ect
    • InTouch View Engine ($ViewEngine)
    • InTouch Applications ($InTouchViewApp)
    • ArchestrA Graphics
  • These object can be Transferred Safely between Galaxies
    • Properly designed they adapt to the Infrastructure of the Target Galaxy
    • Little or no editing should be required when transferred
Multi-User Development

- Developer Workstation
  - Operating System can be Windows 7
  - Installation
    - System Platform DVD Install
    - System Platform Development Server (Role Install)
      - Remove the InTouch Demo Apps if desired
    - Historian Server Node (Role Install)
      - Testing of Object history and Tiering of Historians
      - Applicable DA Servers if needed
  - This provides for a complete standalone development workstation with all capabilities covered by Dev. Studio License.
Project Infrastructure

Project Servers (Operating System Instances)

Galaxy Repository
Historian Server
Information Server
Workflow Server

Engineering Workstations

EWS 1
EWS 2
EWS 3
EWS 4

AOS
RMC
AOS
Project Infrastructure

Template Galaxy
  Standard Templates
  Project Templates
  Project Galaxy
  Project Infrastructure
  Project Application

Project Servers (Operating System Instances)

Galaxy Repository
Historian Server
Information Server
Workflow Server

Development Galaxy
  Dev. Infrastructure
  Test Templates
  Test Instances

EWS 1
EWS 2
EWS 3
EWS 4

Dedicated DI Clients

Engineering Workstations
Multiple Developers in one Galaxy

- Production Galaxy
- Template Galaxy

Each Development Workstation has a Local Development and Test Galaxy
Scaling out Production w/Galaxy Load

- Good Tool for scaling out the Production Galaxy
- Allows Spreadsheet definition of Instances
- Can also Manipulate Existing Instances
  - Change Areas
  - Attribute Values
- Only Attributes to be Changed are Required
  - Tagname is the Key (Always Required)
  - Tagname and SecurityGroup (Required for new Instances)
- Format is Simple
  - Even the Cancel Button works 😊
- Can Create “Templates” of Non-Templateable Objects
  - DI Networks and Devices
  - Areas and Contained Objects
  - Engine Layouts for Platforms
Galaxy Load to Create Instances
Organize your Templates

• Create a Template Storage Galaxy
  – Maintains current Templates
  – Distribute via Packages
  – Control distribution of updates
  – Should always have Security Enabled

• Levels in hierarchy dictate Ownership within Team/Organization

• Very Few Instances should exist in this Galaxy
  – Necessary for placing template Graphics on InTouch Windows
Base Template Library

Organization/User Owned

Water Filter Project Owned Template

Water Filter Template
Galaxy Design Guidelines (Estimates?)

- **Platforms**
  - Multiple AOS Platforms reduce deployment times
- **Application Engines (Galaxy Work Horse)**
  - 1 Active Engine / Core / 1-2 GB
  - 5-10,000 IO / Engine
  - 2,000 Objects / Engine
  - Standard Engines can Handle a Heavier Load than Redundant Engines
- **View Engines**
  - Can host multiple InTouch App Instances
  - Can serve as an Active Engine in Runtime
    - Template for Configuration Settings
    - Holder of ArchestrA Graphics as Windows
  - Multiple View Engines Can be Used on same Platform (Have a good reason for THIS)
Galaxy Design Guidelines (Estimates?)

- Areas
  - Provides for Object Distribution across Engines
  - Hierarchical Model
    - Alarming and Events
    - Historical Data (If Enabled to 1st Tier Historian)
- Areas are Sisters in Execution Not Hierarchical
- Must have Multiple Areas to support Multiple Engines
- Rollup of Alarm Counts / Enable / Silence / Disable
- Limit of 500 Objects / Area is a good rule of thumb
Galaxy Infrastructure

- Create Templates for Standards
  - Lockable Read Only Attributes
    - Will not be Dumped via Galaxy Dump
    - Propagation will be guaranteed
    - Slightly better performance
  - Writable Attributes needing Initialization in Configuration or Runtime
    - Locking cannot be used
    - Utilize an OnStartup script to set the values
      - Script can be locked to ensure propagation in IDE
      - Attribute values will remain writable at runtime
      - Use Read-only Security for IDE only settings. (enforced even with "None" Model)
      - Protects against Upload Runtime Changes overwriting initial values
Auto Complete Script Editor
Database Integration

• aaDBIntegration Script Library
  • Loaded by default into every Galaxy since v3.0(?)
  • Script Safe .net class library for database tasks
    • Object Classes
      • aaDBClient
        • aaDBCommand
        • aaDBConnection
      • aaDBRow
      • aaDBTransaction
Database Integration

- Enumerations
  - aaDBCommandState
  - aaDBCommandType
  - aaDBConnectionState
  - aaDBConnectionType
  - aaDBParameterDirection
  - aaDBTransactionState
Database Integration

- Connection pooling
- Results Sharing
- Synchronous Transactions
- Asynchronous Transactions
- Engine Safe
- Object Safe
Device Integration Objects

Four types of Device Integration objects are in the Galaxy Toolbox:

- **Di Client Objects**
  - Connect to externally installed and configured DA Servers and Applications

- **Di Network Objects**
  - Contain DA Servers in the Runtime Package which is deployed to the Target Platform

- **Di Device Objects**
  - Configure the Di Network device hierarchy and serve as Di Client objects to the component in the hierarchy of devices on this Network

- **Redundant Di Objects**
  - Choose between two DI Client objects as a provider of Device Items to Application Objects
Redundancy

Treat Redundancy as an Insurance Policy Understand what it covers and what it does not.

- **Device Communication**
  - Selects Between Two Dedicated Communication Paths to a Device
  - Failover Less Than 5 Seconds

- **Application Engine Redundancy**
  - Preserves the aaEngine as a Service
  - Executes a state-full restart of the Engine Service on another Platform
  - Failover Less Than 1 Minute

- **Operating System Redundancy**
  - Managed by Hyper-V, V Motion, or Hardware
  - GR, WIS, RDS, Historian, Workflow
  - Executes a state-full restart of the OS
  - Failover Less Than 10 Minutes (Typically)
Redundancy
Sizing Guidelines

• Operating System Sizing
  • GR
    – Windows 2008 R2, 4 Cores, 4-8GB Memory
    – Recommended to be Virtualized
    – Only way to recover a GR without a Required Deployment
  • AOS
    – Windows 7, 4 Cores, 4GB Memory
    – Virtualize Larger Hardware into Blocks this Size
    – Improves Deployment Speed
    – Failover Performance
    – Upgrade with Minimum Downtime (Following Section)
    – 25,000 IO Per Standard AOS or Redundant AOS Pair (YMMV)
    – Ideally DA Servers are Local
Sizing Guidelines

• Operating System Sizing
  • InTouch Workstation
    – Windows 7, Dual Core, 4GB Memory
    – High clock speed better than more cores
    – Fast Disks or Solid State (Loading windows from disk)
  • InTouch RDS Server
    – Windows 2008 R2
    – Lots of Cores (16), Lots of Memory (48GB)
    – Solid State Disks
    – 25 - 75 Sessions per Server (YMMV)