Multi-discipline Program Office and Functional Integration – Challenges and Solutions

Ravi Rangan, Chief Technical Officer

Outline

- Lifecycle context for Simulation and Modeling
  - Practices and evolution
  - Deployment considerations
- Accounting for patterns
  - CM process dynamics support within domain model
  - Leveraging available infrastructure, and organic evolution
- Use Cases
  - Functional Integration
  - Multi-Discipline Program Office
- Summary
Delivering benefits across many industries

Key Business Drivers

- Global Operations
  - Design anywhere/ build anywhere
- Outsourced Operations and partnerships
  - M&A, outsourcing
- Business Concurrency & Decision Support
  - Cycle time reduction, mass customization
- Bigger investment in upfront design
- Rapid time-to-benefit
Implementation Challenges

© 2007 Centric Software, Inc.

Existing Tools and Methods

• Lack of system engineering mindset
• Optimized around old technologies
• Limited application scope
• Hidden inefficiencies

Next Gen Tools and Methods

• Robust, system engineering mindset
• Flexible and modular to handle new technologies
• Broad application scope

Lesson learned: We need to better understand deployment and adoption patterns

GM: BRYANT’S LAW (modified)

\[ S = (100E + 10P + 1T)\times D \]

where

- **S** = Success + Realized value
- **E** = Empowered, trained people
- **P** = Process
- **T** = Technology
- **D** = Data (legacy, current, future)


Systems Modeling EIM Deployment within an Automotive OEM, Mark Jennings, Ford, ASME EIM Panel 2003

© 2007 Centric Software, Inc.
**Key Patterns**

**Evolution of Business Object Management Practices**

1. Document-based control
3. Part-based control with change management within BOM
4. Part, function and system with change management across configurations

<table>
<thead>
<tr>
<th>Business Value</th>
<th>Breadth of lifecycle addressed</th>
<th>Part Complete System</th>
</tr>
</thead>
<tbody>
<tr>
<td>Low</td>
<td></td>
<td></td>
</tr>
<tr>
<td>High</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

© 2007 Centric Software, Inc.
Functional Silos Prevent Optimal Execution
Crucial Product Data Lives Everywhere

Value
Multi-domain Processes
Insight and Understanding
Information
Data

Business Intelligence
Unstructured and Unmanaged Data
PLM, Doc Mgt, CAD
SCM, CRM, ERP

User Category
Product Professionals
Business Professionals

Intelligent Information Aggregation

Domain Model accounts for Business Method Robustness

Manage as Isolated, un-intelligent content (document centric configuration management)

Account for Impact analysis
Manage relationships for System Effects process continuity

Balance Concurrence, role autonomy, change isolation and Process dynamics through Interfaces
- Delegation
- Lean processes
- Accountability / Traceability
- Uniform and common process

© 2007 Centric Software, Inc.
Solution Drivers

- Streamline design data integration process across product domains
  - Unified view of logical system and functional design
  - Secure management of mappings & allocations between logical system and geometric product structures
- Leverage off-the-shelf connectivity products
- Lower cost of ownership of integrated system
- Rapid deployment with responsive iterative feedback cycles
- Formally accommodate aggregated visibility and allocation use-cases and CM process dynamics

Complex Products - More than BOMs

<table>
<thead>
<tr>
<th>Phase</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>Requirements Analysis</td>
<td></td>
</tr>
<tr>
<td>Abstract Function</td>
<td></td>
</tr>
<tr>
<td>Generalized Function</td>
<td></td>
</tr>
<tr>
<td>Physical Function</td>
<td></td>
</tr>
<tr>
<td>Physical Form</td>
<td></td>
</tr>
<tr>
<td>Production Process</td>
<td></td>
</tr>
<tr>
<td>Operations Process</td>
<td></td>
</tr>
</tbody>
</table>

- Extensive cross-domain mappings & allocations across product domains
- Detailed requirements analysis
- High-level conceptual design
- Generalized function
- Physical function
- Physical form
- Production process
- Operations process

© 2007 Centric Software, Inc.
Complex systems: Elaborate through multiple levels of abstractions

- **Requirements**
- **Functional Design**
- **Logical Design**
- **Physical Design**

Chosen abstraction levels establish process (certain levels skipped)

Capture traceability & rationale, as needed

Allocations for Accountability

Enable CM across existing systems and processes

- Functional/Logical System Design (XXX)
- Component Connectivity (YYY)
- Design Configuration (ZZZ)

SOA-based Connectivity (Synchronized Product Intelligence)

Search, Aggregate & Execute

Centric Connector

© 2007 Centric Software, Inc.
Functional Integration

1. Provides the Full System View
   - Review design perspectives from various workgroup data repositories
   - Combined reconciliation of the Integrated Product Architecture

2. Present relevant information at each level of design abstraction
   - Drill-down to any level of detail
   - Capture and preserve product semantics
   - Aggregate and allocate design perspectives without losing functional integrity

3. Enable information relationships across functional boundaries
   - Native support of Functional Integration concepts
   - Allocation between design perspectives to ensure audit & accountability
   - Configuration Management for Functional Integration
### Functional-Logical Mapping

**Allocation Data**

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
<th>Logical Design</th>
<th>Implementation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Functional-Logical Mapping</td>
<td>Functional Allocation</td>
<td>Allocation of Functional Hierarchy to Logical Design</td>
<td>LD-001, 1</td>
<td>FW-001, 1</td>
</tr>
</tbody>
</table>

**Allocation Data Table**

<table>
<thead>
<tr>
<th>EXE-020221</th>
<th>SFNC-102101</th>
<th>SFNC-102121</th>
<th>SFNC-102130</th>
<th>SFNC-102140</th>
<th>SFNC-102150</th>
<th>SFNC-102160</th>
<th>SFNC-102170</th>
<th>SFNC-102180</th>
</tr>
</thead>
<tbody>
<tr>
<td>EXE-020222</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>EXE-020223</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>EXE-020224</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>EXE-020225</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>EXE-020226</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>EXE-020227</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>EXE-020228</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>EXE-020229</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>EXE-020230</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>EXE-020231</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Allocation Table**

<table>
<thead>
<tr>
<th>ID</th>
<th>Component Type</th>
<th>Description</th>
<th>Signal Name</th>
<th>Signal Type</th>
<th>Signal Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>EVF-00129</td>
<td>Cockpit Sensor</td>
<td>SYS-004-18</td>
<td>Config_0114-1</td>
<td>Input</td>
<td></td>
</tr>
<tr>
<td>EVF-00129</td>
<td>Cockpit Sensor</td>
<td>SYS-004-18</td>
<td>FLW-0004023</td>
<td>Input</td>
<td></td>
</tr>
<tr>
<td>EVF-00129</td>
<td>Cockpit Sensor</td>
<td>SYS-004-18</td>
<td>FLW-0004021</td>
<td>Output</td>
<td></td>
</tr>
<tr>
<td>EVF-00129</td>
<td>Cockpit Sensor</td>
<td>SYS-004-18</td>
<td>FLW-0004020</td>
<td>Output</td>
<td></td>
</tr>
<tr>
<td>EVF-00129</td>
<td>Cockpit Sensor</td>
<td>SYS-004-18</td>
<td>FLW-0004022</td>
<td>Output</td>
<td></td>
</tr>
<tr>
<td>EVF-00129</td>
<td>Cockpit Sensor</td>
<td>SYS-004-18</td>
<td>FLW-0004021</td>
<td>Output</td>
<td></td>
</tr>
<tr>
<td>EVF-00129</td>
<td>Cockpit Sensor</td>
<td>SYS-004-18</td>
<td>FLW-0004023</td>
<td>Output</td>
<td></td>
</tr>
</tbody>
</table>

**Export to Excel**

<table>
<thead>
<tr>
<th>EXE-020221</th>
<th>SFNC-102101</th>
<th>SFNC-102121</th>
<th>SFNC-102130</th>
<th>SFNC-102140</th>
<th>SFNC-102150</th>
<th>SFNC-102160</th>
<th>SFNC-102170</th>
<th>SFNC-102180</th>
</tr>
</thead>
<tbody>
<tr>
<td>EXE-020222</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>EXE-020223</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>EXE-020224</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>EXE-020225</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>EXE-020226</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>EXE-020227</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>EXE-020228</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>EXE-020229</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>EXE-020230</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>EXE-020231</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Project**

- Name: Functional-Logical Mapping
- Type: Functional Allocation
- Description: Allocation of Functional Hierarchy to Logical Design
- Logical Design: LD-001, 1
- Implementation: FW-001, 1
Centric for Functional Integration

1. Provide a Full System View from bottom-up implementations
   - Review design perspectives from various workgroup data repositories
   - Systems level visibility and interaction with Integrated Product Architecture

2. Presents relevant information at each level of design abstraction
   - Drill-down to any level of detail
   - Capture and preserve product semantics
   - Aggregate and allocate design perspectives without losing functional integrity

3. Enable information relationships across functional boundaries
   - Native support of Functional Integration concepts
   - Allocation between design perspectives to ensure audit & accountability
   - Configuration Management for Functional Integration
Barriers Create Operational Problems

Decision quality and team coordination need optimization

- Simple questions difficult to resolve:
  - Are we on schedule?
  - What are the current issues?
  - What are the sources of problems?
  - What needs our attention?
  - What should we do?
  - Has it been done?
Program Office Example

Top Problems Addressed

- Early warning of risks to on-time delivery
- Faster response to customer change requests with improved impacts analysis
- Earlier awareness of cost overruns
- Faster determination of overrun cause and financial responsibility
- Increase program efficiency and productivity utilizing existing resources

Program Office Navigation & Hierarchy

- Decision Center’s navigation & hierarchy is flexible!
- In Decision Center Program Office application, users navigate & information is organized by:
  - Site (List of Programs)
  - Program
  - Team (Functional Teams)
  - Deliverable

- Provides information appropriate for every level in the organization.
- Higher level views (Site, Program, etc.) are “roll-ups” of information created and managed at lower levels.
Centric Program Office
Make strategic program decisions based on multiple data sources

- Program Manager
  - Create, Communicate and Monitor activities and status
  - Visualize all relevant up-to-date program business and product data
  - Easily determine program, project and activity dependencies

Deploy Out-of-the-Box PI Applications
Unique technology combination enables PI

- Collect
  - Only patented, product intelligent search
  - Broadest and deepest off-the-shelf connectivity

- Decide
  - Patented data presentation and integration
  - Multiple, flexible, customer-defined interfaces

- Execute
  - Only execution environment powered by data search
  - Strategic and tactical views

- Assess
  - User-defined business rules for alerts

© 2007 Centric Software, Inc.
Summary

- Systems validation and functional integration requires appropriate levels of abstractions
  - Respecting business object authority semantics
  - with support for allocations, relationship capture across domains
  - Change propagation and change insulation semantics across systems
  - Macro-level program management with actionable performance visibility

- For effective adoption, applications must anticipate:
  - installed IT investments
  - prevalent processes, practice patterns

- Pragmatic solutions available today, based on:
  - Intelligent connectivity (SOA based support)
  - Configurable domain model with process dynamics semantics