Yesterday, Today & Tomorrow
Implementations of the Development Lifecycles

Agenda

- Background
- Yesterday’s Implementation
- Today’s Implementation
- Trends for Future Implementation
'Yesterday' Development Lifecycles

Nine Plus Years - Vintage 2167A

- Variations of Waterfall & Incremental Spiral Lifecycles
- All Lifecycles Implemented Incremental Releases of Functionality
- Informal Risk Management
- User Interface Prototyping
- Requirements Not Clearly Understood By Customer at Start
- Requirements Evolved Throughout Development Life-cycle
- Some Projects Were Based on Legacy Systems
- Cost and Schedule Constraints

'Yesterday'- Continued

- What Worked Well
  - Program Management Review Process
  - Utilization of Subject Matter Experts (SME) Early in Lifecycle
  - Early User Interface Prototyping
  - End User Demonstrations During Development
  - Change Management Process
  - Independent Testing Process (Separate Development & Test Teams)
- What Required Improvement
  - Merging of Build Functionality
  - Major Build / Release Development Schedule Overlap
  - Experience with Methodologies & Tools
  - Metrics Collection
  - Estimation of Effort (i.e. Resources)
‘Today’s Development Lifecycles

- Variations of Incremental & Spiral Lifecycles
- All Lifecycles Implementing Incremental Releases of Functionality
- Formal Risk Management
- User Interface Prototyping
- Requirements Not Clearly Understood by Customer at Start
- Requirements Evolved Throughout Development Life-cycle
- Cost and Schedule Constraints
- Multi-Contracts Working With Multi-Contractors

Summary of 3 of ‘Today’s’ Projects

- Three Independent Contracts
- Multiple Contractors on Each Project
- Two Projects Following Process with Slight Tailoring
- What Works Well on One Project May Not on Another Due To Individual Teams Implementation and Management
- All Teams Working to Build a Single Integrated Product
Summary of Projects

<table>
<thead>
<tr>
<th>Projects</th>
<th>A</th>
<th>B</th>
<th>C</th>
</tr>
</thead>
<tbody>
<tr>
<td>What is Working Well Today</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Program Management Reviews</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Transition between Development Processes</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Task &amp; Change Management</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Tools Which Support the Development Methodology</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Issue &amp; Change Management</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>User Guide Supporting Development</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Metrics Collection and Analysis</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Software Integration Management</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Narrative View of Early Development Products</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Conceptual Modeling of User Stories</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Technical Review Process</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Defect-Driven (Part Reason)</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Change Management Process</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
</tbody>
</table>

What Requires Improvement Today

<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>B</th>
<th>C</th>
</tr>
</thead>
<tbody>
<tr>
<td>Multi-Contract Internal Review Process</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Single-Contract Internal Review Process</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Narrative View of Early Development Products</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Conceptual Modeling of User Stories</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>More Realistic User Development Story Points</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Collection &amp; Analysis of Development Environment Metrics</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Project Management of Schedule Expectations</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Software Integration Management</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Component-Level Integration Planning</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Performance and Stability of Development Tools</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>System Integration Management</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
</tbody>
</table>
‘Tomorrow’

Lifecyle Trends

- Distributed Spiral Development
- Multi-Releases During Development
- Increased User Involvement During Development
- Increased Risk Management Influence on Development
- Seamless System / Software Development
- Accelerated New Technology Analysis & Insertion
- Movement From Cost Plus to Fixed Price Mentality Independent of Contract Type

‘Tomorrow’

Process Trends

- Integrated System and Software Engineering
- Distributed Project Management & Control
- Distributed Technical Review Processes
- Formal Integration Management Processes
- Enhanced Metrics Collection Processes
‘Tomorrow’
Tools Trends

- Seamless Development Environment
- Web-Based Support & Analysis Tools
  - Program Management
  - Risk Management
  - Engineering
  - Quality Control
  - Configuration / Data Management
  - Metrics
- Developed Products Accessible Via Web

Discussions for Working Groups

- Impact of Distributed Development on Spiral Development
- Impact of Integrated Product Teams on Spiral Development
- Methods to Reduce Barriers Between System & Software Engineering Process
- Impact of Fixed Price Contracts with Integrated Product Teams using Spiral Lifecycle