Composable COTS-Based Processes Elements

Ye Yang, USC

USC-CSE ARR’05
Mar. 15th, 2005
Paradigm Changes

- Custom development vs. COTS integration
  - USC e-services data shows an increase of CBA projects from 28% in 1997 to 70% in 2002
  - Standish Group 2000 Chaos: 54%
- Limitations of traditional software lifecycle process models
  - Waterfall-based: poorly-matched
  - Spiral: non-specific
- Need efficient processes and still maintain enough control
A CBA-Oriented Process Framework

CBA Experience Base

- CBA Classification
  - Top Risks
  - Guidelines
  - Process Patterns

CBA Process Decision Framework

Process Elements

- COTS Market/Vendors
- Information, updates, changes

Stakeholders’ WinWin Negotiation

System OC&P Agreements

COTS integration characteristics

Cost exposure

COCOMO II, COCOTS
Problem: Overfocus on Capabilities vs. Integration – Supply Chain Example

• Identify requirements
  – Enterprise Resource Planning (ERP), Advanced Planning and Scheduling (AP&S), Transportation Mgmt. (TM), Custom Relationship Mgmt. (CRM)

• Select best COTS products for each
  – The cost of the COTS products was about $200K

• Find that best COTS products don’t interoperate

• Overrun budget and schedule in recovering
  – $3M budget overrun
  – 8 months delay
  – Many associated system effectiveness shortfalls
How COTS Process Elements Address this Problem

• **Requirements**
  - Balance of prioritized objectives
  - Checklist helps to emphasize interoperability objectives

• **Process framework identifies specific assessment needs for interoperability**
  - Include experimental tailoring and glue code
CBA Process Decision Framework

Start

P1: Identify Objective, Constraints and Priorities (OC&Ps)

P2: Do Relevant COTS Products Exist?

No or Unsure

P3: Assess COTS Candidates

Yes

Single Full-COTS solution satisfies all OC&Ps

P4: Tailoring Required?

Yes

P5: Multiple COTS cover all OC&Ps?

No

Partial COTS solution best

No acceptable COTS-Based Solution

P6: Can adjust OC&Ps?

Yes

P7: Custom Development

No

P8: Coordinate custom code and glue code development

Yes

P9: Develop Custom Code

No

P10: Develop Glue Code

P11: Tailor COTS

P12: Productize, Test and Transition

Deployment
A4. Detailed Assessment

• Prototyping that involves
  – Tailoring to evaluate COTS tailorability (usability)
  – Glue code to test the interoperability among several system components (COTS and/or custom)
How Much Assessment is Enough?

• Use risk analysis to determine
  – Risk Exposure: Probability of loss \( P(L) \) * Size of loss \( S(L) \)

![Diagram showing risk exposure and COTS assessment effort relationship]

- Many Errors: high \( P(L) \)
- Critical Errors: high \( S(L) \)
- Few Errors: low \( P(L) \)
- Minor Errors: low \( S(L) \)
- Few Delays: low \( P(L) \)
- Short Delays: low \( S(L) \)
- Many Delays: high \( P(L) \)
- Long Delays: high \( S(L) \)
Tailoring Process Element

T1: Identify tailoring methods available for the selected COTS components

T2: Clear best tailoring method?
   - No
   - Yes → T3: Perform tailoring effort vs functionality trade-off analysis
   - No → T4: Tailoring-functionality trade-off feasible to satisfy OC&Ps?
      - No → No
      - Yes → A4: Detailed Assessment

A4: Detailed Assessment

G4: Develop glue code and integrate

T6: Perform Tailoring

T5: Design and Plan tailoring using best available tailoring method

P12: Productize, Test and Transition

Tailoring Option | GUI-based | Parameter-based | Script-based |
-----------------|-----------|----------------|-------------|
Design Details Required | Low - None | Low | Detailed |
Complexity | Low | Moderate | Moderate | High |
Adaptability | Low | Low | Moderate | High |
Developer Resources | Low | Low | Moderate | High |

Tailoring Option GUI-based Parameter-based Script-based
Design Details Required Low - None Low Detailed
Complexity Low Moderate Moderate High
Adaptability Low Low - Moderate High
Developer Resources Low Low Moderate - High
Glue-code Process Element

- Determine interconnection topology options
- Minimize the complexity of interactions
- Select connectors (e.g. events, procedure calls, pipes, shared memory, DB, etc.) w.r.t. COTS interfaces
- Identify potential architectural mismatches

G1: Architect and Design Glueware

G2: Architecture Feasible?

G3: Tailoring Required

G4: Develop glue code and integrate

P12: Productize, Test and Transition

A4: Detailed Assessment

P6: Can adjust OC&Ps?

P9: Develop custom code

No acceptable COTS-Based Solution
Problem Avoidance

• Perform detailed assessment to further filter the leading choices
  – identify the major interoperability risk exposures,
  – use the size of the overall risk exposure to scope the more detailed COTS interoperability assessment

• Choose a few key interoperability scenarios and the COTS candidates are selectively tailored to support them

• Prototyping with glue code to identify and resolve interoperability issue earlier

• The overall assessment cost would be much less than the interoperability risk exposure
Experiment and Results

• Apply the Framework and Process Elements in CBA project planning and control

• 13 USC e-services CBA projects
  – Group A: 8 CBA projects that applied.
  – Group B: 5 CBA projects that did not apply.

• Survey Results:

  Comparison of COTS Impacts on the two groups (# responses in Group A: 44; in Group B: 26):

Team Performance:

<table>
<thead>
<tr>
<th></th>
<th># of Teams</th>
<th>Score</th>
<th>Client Eval.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Total</td>
<td>19</td>
<td>650</td>
<td>20</td>
</tr>
<tr>
<td>Group A</td>
<td>8</td>
<td>589.8</td>
<td>18.5</td>
</tr>
<tr>
<td>Group B</td>
<td>5</td>
<td>578.9</td>
<td>17.64</td>
</tr>
</tbody>
</table>

In Group A, percentage of developers reported that:

- Framework can help with:
  - COTS assessment: 81.8%
  - Risk identification and mitigation: 68.2%
  - Life cycle planning: 63.6%
Conclusions

• Accommodating for process flexibility with composable process elements and dominant decision points

• Adapting to frequent go-backs with a recursive, and reentrant Spiral framework
  – Based on new and evolving OC&P’s
  – Based on risk considerations

• Facilitating strategic effort/resource allocation based on COCOTS estimates and risk analysis

• Preliminary experiment results shows notable team performance improvement
Backup Slides
Key Definition and Context

• COTS Based Applications
  • as a system for which
    • at least 10% of the development effort is devoted to COTS considerations
    • at least 30% of the end-user functionality is provided by COTS products

• Different from custom development on COTS platforms
Primary COTS Activities

- **Major effort distributed among COTS assessment, tailoring, and glue code development**
  - COTS assessment: activity whereby COTS products are evaluated and selected as viable components for a user application
  - COTS tailoring: activity whereby COTS software products are configured for use in a specific context
  - COTS glue code development: activity whereby code is designed, developed, and used to ensure that COTS products satisfactorily interoperate in support of the user application.
COTS Characteristics

• COTS characteristics requiring COTS-based process, accommodation
  – Positive: affordability, timelines, tailorability
  – Negative: opacity, incompatibility, uncontrollability
  – Both: proliferation, dynamism

• Accommodation mechanism
  – Assessment: proliferation, opacity, incompatibilities
  – Tailoring: tailorability, incompatibilities
  – Glue code: incompatibility, interoperability, uncontrollability
  – Synchronized refresh: uncontrollability, dynamism
Guidelines

- **COTS Assessment Background (CAB)**
  - Provides the minimum essential set of organization objectives, constraints, priorities (OC&P’s), and situation background needed to perform a COTS/NDI assessment

- **COTS Assessment Plan (CAP)**
  - Covers the minimum essential “why/whereas, what/when, who/where, how, and how much” aspects of the activity being planned

- **COTS Assessment Report (CAR)**
  - Covers the COTS assessment objectives, context, approach, results, conclusions, recommendations, and supporting data, plus other significant topics if any
## Software Development Phase Activities: Custom & COTS

<table>
<thead>
<tr>
<th>Requirements (Inception)</th>
<th>Design (Elaboration)</th>
<th>Implementation (Construction)</th>
<th>Acceptance &amp; Deployment (Transition)</th>
<th>Sustainment (Maintenance)</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Custom Development Activities</strong></td>
<td><strong>Custom Development Activities</strong></td>
<td><strong>Custom Development Activities</strong></td>
<td><strong>Custom Development Activities</strong></td>
<td><strong>Custom Development Activities</strong></td>
</tr>
<tr>
<td>• Ops Concept</td>
<td>• Preliminary Design</td>
<td>• Code &amp; Unit Test</td>
<td>• Acceptance Test</td>
<td>• Operations</td>
</tr>
<tr>
<td>• Planning</td>
<td>• Detailed Design</td>
<td>• Component Test</td>
<td>• Site Installation</td>
<td>• Maintenance</td>
</tr>
<tr>
<td>• Software Reqs</td>
<td></td>
<td>• System Test</td>
<td>• Site Activation</td>
<td>• Enhancements</td>
</tr>
<tr>
<td>• Architecture</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>• Risk, Feasibility</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>COTS Specific Activities</strong></td>
<td><strong>COTS Specific Activities</strong></td>
<td><strong>COTS Specific Activities</strong></td>
<td><strong>COTS Specific Activities</strong></td>
<td><strong>COTS Specific Activities</strong></td>
</tr>
<tr>
<td>• Definition of Objectives Constraints &amp; Priorities</td>
<td>• Tailoring</td>
<td>• Acceptance Test</td>
<td>• Component Refresh</td>
<td>• Component Refresh</td>
</tr>
<tr>
<td>• COTS Component Identification</td>
<td>• Glue code development</td>
<td>• Site Installation</td>
<td></td>
<td></td>
</tr>
<tr>
<td>• COTS Assessment/Selection</td>
<td>• COTS Integration &amp; Test</td>
<td>• Site Activation</td>
<td></td>
<td></td>
</tr>
<tr>
<td>• Prototyping</td>
<td>• Component Refresh</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>• Risk, Feasibility</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>