A RESCUE OF A COTS PUBLIC SECTOR PROJECT USING AGILE, SCRUM, LEAN, AND SAFe PRINCIPLES, PRACTICES AND FRAMEWORKS

AKA

Let me get this straight – A rescue project on a public sector Oracle COTs – I’m in

Ben Blanquera
bblanquera@pillartechnology.com
Agile 2014
A little about me
What we’re going to talk about

• Team structure
• The story
  • The big ugly - first 90 day
  • It still hurts - days 90-180
  • Hitting our groove - days 180-360
  • Countdown to release - days 360-450
• The practices
• Success factors and lesson learned
Challenges

- How quickly could we define and implement a delivery process that leveraged agile principles in the midst of a high-pressure project rescue environment?

- Could we shift the culture to high trust mode?

- Could we break up a large monolithic COTS system release into an incremental delivery model?

- What mechanisms would we need to put into place to manage the complexity of the 40+ integrations?

- What could we do to incorporate traditional QA phases (system integration testing, performance testing, user acceptance testing) into an agile process?
Team structure

- **Program leadership**
  - Configuration teams
    - Configure package — includes dev and testers
  - Business analysis teams
  - Infrastructure
    - Build infrastructure
  - Interfaces and integration
    - Build integrations using WebSphere ESB
  - QA teams
    - Regression and system integration testing
  - Other — Data/Reporting/PeopleSoft
    - Data migration, reporting, document mgmt, PeopleSoft
The Big Ugly
days 0-90
Scrum Framework

Product Backlog → Sprint Backlog → 2-4 Weeks → Potentially shipable Product Increment

24 Hours
It still hurts
days 90-180
Hitting our groove —
days 180 -360
**Illustrative Go To Release Plan**

<table>
<thead>
<tr>
<th>Build</th>
<th>System integration test</th>
<th>User acceptance test</th>
<th>Deploy and cutover</th>
</tr>
</thead>
<tbody>
<tr>
<td>Story testing and acceptance</td>
<td>Performance testing</td>
<td>End user training</td>
<td>Data conversion and load</td>
</tr>
<tr>
<td>Infrastructure upgrades</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Sprints</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>n</td>
<td>n+1</td>
</tr>
</tbody>
</table>
Countdown to release –
days 360 - 450
The practices

- Indicates where our practices mirrored what is described in SAFe.

Program refactoring
Success factors

- The right people with the right attitude and sense of urgency
- Program governance and refactoring
- Release, features, stories
- Transparency and trust
- Continuous Improvement and learning
Lessons learned

- Architect solution path early before bringing majority of dev team on board
- develop the QA/testing strategy early and resource automation activities early,
- focus on and resource devops early to enable rapid and frequent deployments into environments, and
- decide early on data conversion strategy and and do mock data conversions early to use as test data
Thank You

Research paper available here –

Ben Blanquera
bblanquera@pillartechology.com
614.288.7004
Delivery Executive
Pillar Technology
pillartechology.com