Implement Continuous Delivery with OpenEdge

OpenEdge is Shifting Left

Peter Judge
pjudge@progress.com
Agenda

- Setting the Stage – Terminology
- Continuous Integration (CI)
- Continuous Validation (CV)
- Questions?
What do we hope to accomplish?

ERRR...

CAN'T STOP.
TOO BUSY!!
Agenda

- Setting the Stage – Terminology
  - CI + CV = CD
  - Value Stream Mapping
  - Tools
- Continuous Integration (CI)
- Continuous Validation (CV)
- Questions
3 Key Terms

**Continuous Integration**

Continuous Integration is the process of checking in code frequently and having **builds triggered by those check-ins that run lightweight tests and code scans that provide quick feedback to engineers** on the code they recently checked in. Continuous Integration is the responsibility of the development team, supported by configurations that were set up by the systems team.

**Continuous Validation**

Continuous Validation is the process of running **automated functional, acceptance and other relevant tests, which are triggered after successful completion of a Continuous Integration cycle**. The systems team owns the configuration of this environment, but the content of the tests and how they should work comes from development.

**Continuous Delivery**

Continuous Delivery is an extension of Continuous Integration and Validation, which ensures that the steps of the previous processes are configured into a **segmented pipeline that results in a potentially ship-able deliverable** at the end of the pipeline. The configuration of this pipeline is mostly owned by the systems team, with input from/usage by development.
Current Value Stream Mapping

Value-added Time

**Sync Build Events to Check-ins**
- Check-in
  - 2-15 mins
  - 10-90 mins
- Dry Runs
  - 18/day
  - 10-90 mins
- "Full Build"
  - 4-5/week
  - ~10 mins
- "Full Build" timing:
  - 10 mins
  - 1-72 hrs between builds
  - 10-270 mins

**Promotion Testing**
- ~10 mins
- 5-10 hrs
- 23-46 hrs
- 12-32 hrs depending on check-in
- 32-288 hrs depending on check-in
- 10 mins

**Move to Quality Gates**
- Move to Quality Gates
  - 2-5 days
- End Game Testing
  - Months between SP or ESAP
  - 2-5 days

**Elapsed Time**
- 23-46 hrs
- 12-32 hrs depending on check-in
- 32-288 hrs depending on check-in
- 10 mins
Development Pipeline w/ Feedback
Continuous Integration

Continuous Validation
Agenda

- Setting the Stage – Terminology
- Continuous Integration (CI)
  - Starting Point
  - Changes Made
  - Next Steps
  - Metrics
- Continuous Validation (CV)
- Questions
## CI Common Practices

<table>
<thead>
<tr>
<th>Practice</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>Maintain a code repository</td>
<td>🔄</td>
</tr>
<tr>
<td>Automate the build</td>
<td>🔄</td>
</tr>
<tr>
<td>Make the build self-testing</td>
<td>🔄</td>
</tr>
<tr>
<td>Everyone commits to the baseline every day</td>
<td>🔄</td>
</tr>
<tr>
<td>Every commit (to baseline) should be built</td>
<td>🔄</td>
</tr>
<tr>
<td>Keep the build fast</td>
<td>🔄</td>
</tr>
<tr>
<td>Test in a clone of the production environment</td>
<td>🔄</td>
</tr>
<tr>
<td>Make it easy to get the latest deliverables</td>
<td>🔄</td>
</tr>
<tr>
<td>Everyone can see the results of the latest build</td>
<td>🔄</td>
</tr>
<tr>
<td>Automate deployment</td>
<td>🔄</td>
</tr>
</tbody>
</table>
Starting Point

- Scheduled Builds
- Builds Include All Product Areas
- Build Failure in One Product May Impact Others
- System Team Monitoring
Changes Made

- Isolate Different Product Builds Where Possible
- Save Shared Artifacts For Integration
- Build “On Demand”
- Quick Feedback to Developers
Continuous Integration
Next Steps

- Continue to Isolate Builds
- Enable Packaging by Product
- Shift Code Scanning Left
Metrics

- Number of Builds
- Build Cycle Time
- Number of Broken Builds
- Time to Resolve Broken Build
Agenda

- Setting the Stage – Terminology
- Continuous Integration (CI)
- Continuous Validation (CV)
  - Starting Point
  - Changes Made
  - Next Steps
  - Metrics
- Questions
Starting Point

- Testing Scheduled Nightly or Weekend
- Some Testing Groups Run at Release Milestones
Continuous Validation

Developers’ Desktops → CI → Component Testing → Integration Testing → Performance/Load Testing → Released to ESD
# Changes Made – Move to Quality Gates

<table>
<thead>
<tr>
<th>QG #2 – Component Tests</th>
<th>QG #3 – Acceptance Tests</th>
<th>QG #4 – Integration Tests</th>
<th>QG #5 – System Testing</th>
<th>QG #6 – End Game/Manual Testing</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Next level basic Core product coverage</td>
<td>• Core product coverage</td>
<td>• Integration w/ other OE products</td>
<td>• Long running, ‘heavy resource’ tests</td>
<td>• Install</td>
</tr>
<tr>
<td>• New feature testing</td>
<td>• New feature testing</td>
<td>• Purify</td>
<td>• Large DB Testing</td>
<td>• Control Code testing</td>
</tr>
<tr>
<td>Sample Tags</td>
<td>Sample Tags</td>
<td>Sample Tags</td>
<td>Sample Tags</td>
<td>N/A</td>
</tr>
<tr>
<td>• #%cert</td>
<td>• #%cert</td>
<td>• #%nightly</td>
<td>• Table.all</td>
<td></td>
</tr>
<tr>
<td>• #%nightly</td>
<td>• #%nightly</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>• #%sanity</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>• 60-90 minutes</td>
<td>• 2-4 hrs</td>
<td>• 2-4 hrs</td>
<td>• 8-12 hrs</td>
<td>• 1-2 Days (?)</td>
</tr>
<tr>
<td>• ~40% Core code coverage</td>
<td>• ~60% Core code coverage</td>
<td>• ~60% Core code coverage</td>
<td>• ??% code coverage</td>
<td></td>
</tr>
</tbody>
</table>
Side-bar: V-Model

- Concept of Operations
- Requirements and Architecture
- Detailed Design
- Verification and Validation
- Integration, Test, and Verification
- System Verification and Validation
- Operation and Maintenance
- Project Test and Integration
- Project Definition

Time
Next Steps

- Continue to build out Quality Gates
- Complete Automation of Testing Groups, such as Performance
- Shift Left – Testing earlier and more often in the development process
Metrics

- Time required to release Gold Candidate
- Bugs found in each quality gate
- Customer found bugs
- Code coverage
- More ESAP or Canary Launches
Open to Your Feedback. And, on the Edge of Our Seats.

Join the Progress Customer Validation Program today!

- **Roadmap Surveys**—Short surveys on product direction and enhancement prioritizes
- **Virtual Open Houses**—Live, interactive webinars sharing new development and architectural runways
- **Usability Surveys**—Show us how you actually use the product
- **Sprint Reviews**—Ongoing showcases that highlight development progress
- **Pre-release software**—Preview, test and share your feedback

Visit community.progress.com and search: OpenEdge CVP
Enter our **raffle** to win Apple AirPods

https://prgress.co/PUGChallenge