Mono: The Kissing Repository
Project Surfliner’s Monorepo Strategy

Matt Critchlow, UC San Diego
Tamsin Johnson, UC Santa Barbara
What is Project Surfliner?
Surfliner Applications

- Shoreline
  - GeoBlacklight
    - Blacklight
  - Blacklight

- Starlight
  - Spotlight
    - Blacklight

- Lark
  - Valkyrie
Development Methodology

• Follow 12-factor methodology
  • Example: Environment variables for shared deployment

• Agile
  • Combined daily stand-up
  • Two week sprints
  • Retrospectives

• Code Review
• Pair Programming
One (git) Repository
To Rule Them All
Mono Repo

One version control repository shared by all Surfliner products.
Flexible Code Boundaries

- Atomic changes cross-product
- Lock-step releases
Flexible Team Boundaries

- Shared build/test/deploy pipeline
- Global code style and quality standards
- Easy integration testing
Pain Points

Where does it hurt? Where are we growing?
Relearning Dependencies

- Extracting shared services is different than we’re used to.
  - More like working with an inlined dependency than a gem; poly-inlined.
- We can (but mostly don’t) release packages like Gems.
Pipeline Maintenance: Sequential

- 8 jobs for 1844 with refactor-again in 1 minute and 37 seconds (queued for 1 second)
- latest detached
- aa788dcc ...
- 1 related merge request: 1844 lark: isolate HTTP error behavior in Controllers

Pipeline DAG Jobs 8 Tests 0

- Build
  - lark:frontend:n...
  - lark:ruby-build

- Test
  - lark:api:test:unit
  - lark:client:test:...

- Integration
  - lark:deploy:rev...

- Review
  - lark:stop:revie...
  - lark:stop:revie...
<table>
<thead>
<tr>
<th>Status</th>
<th>Pipeline</th>
<th>Commit</th>
<th>Stages</th>
<th>Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>passed</td>
<td>#61752888 by ⭕️</td>
<td>reindex_eve... → a019bc1b</td>
<td>![Checkmark] ![Checkmark] ![Checkmark] ![Checkmark] ![Checkmark]</td>
<td>00:17:22</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Update README for SOLR re-i...</td>
<td>![Checkmark] ![Checkmark] ![Checkmark] ![Checkmark] ![Checkmark]</td>
<td>42 minutes ago</td>
</tr>
<tr>
<td>running</td>
<td>#61749072 by 🤖</td>
<td>master → e4e571d9</td>
<td>![Checkmark] ![Checkmark] ![Checkmark] ![Checkmark] ![Checkmark]</td>
<td>00:12:29</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Updated lark README.</td>
<td>![Checkmark] ![Checkmark] ![Checkmark] ![Checkmark] ![Checkmark]</td>
<td>17 hours ago</td>
</tr>
<tr>
<td>passed</td>
<td>#61746859 by ⭕️</td>
<td>dummy_data → e4e571d9</td>
<td>![Checkmark] ![Checkmark] ![Checkmark] ![Checkmark] ![Checkmark]</td>
<td>00:12:04</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Updated lark README.</td>
<td>![Checkmark] ![Checkmark] ![Checkmark] ![Checkmark] ![Checkmark]</td>
<td>17 hours ago</td>
</tr>
<tr>
<td>running</td>
<td>#61746778 by 🤖</td>
<td>master → bf3c3449</td>
<td>![Checkmark] ![Checkmark] ![Checkmark] ![Checkmark] ![Checkmark]</td>
<td>00:07:34</td>
</tr>
<tr>
<td></td>
<td></td>
<td>lark: Make staging deploy req...</td>
<td>![Checkmark] ![Checkmark] ![Checkmark] ![Checkmark] ![Checkmark]</td>
<td>17 hours ago</td>
</tr>
<tr>
<td>passed</td>
<td>#61743209 by ⭕️</td>
<td>staging-dep... → bf3c3449</td>
<td>![Checkmark] ![Checkmark] ![Checkmark] ![Checkmark] ![Checkmark]</td>
<td>00:10:03</td>
</tr>
<tr>
<td></td>
<td></td>
<td>lark: Make staging deploy req...</td>
<td>![Checkmark] ![Checkmark] ![Checkmark] ![Checkmark] ![Checkmark]</td>
<td>18 hours ago</td>
</tr>
<tr>
<td>passed</td>
<td>#61742382 by ⭕️</td>
<td>master → 8fd4d12c</td>
<td>![Checkmark] ![Checkmark] ![Checkmark] ![Checkmark] ![Checkmark]</td>
<td>00:09:59</td>
</tr>
<tr>
<td></td>
<td></td>
<td>starlight [UCSB]: fix footer link...</td>
<td>![Checkmark] ![Checkmark] ![Checkmark] ![Checkmark] ![Checkmark]</td>
<td>18 hours ago</td>
</tr>
<tr>
<td>passed</td>
<td>#61734817 by ⭕️</td>
<td>reindex_eve... → 498c339a</td>
<td>![Checkmark] ![Checkmark] ![Checkmark] ![Checkmark] ![Checkmark]</td>
<td>00:12:31</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Update README for SOLR re-i...</td>
<td>![Checkmark] ![Checkmark] ![Checkmark] ![Checkmark] ![Checkmark]</td>
<td>19 hours ago</td>
</tr>
<tr>
<td>passed</td>
<td>#61733575 by ⭕️</td>
<td>GL-216 → 8fd4d12c</td>
<td>![Checkmark] ![Checkmark] ![Checkmark] ![Checkmark] ![Checkmark]</td>
<td>00:19:56</td>
</tr>
<tr>
<td></td>
<td></td>
<td>starlight [UCSB]: fix footer link...</td>
<td>![Checkmark] ![Checkmark] ![Checkmark] ![Checkmark] ![Checkmark]</td>
<td>18 hours ago</td>
</tr>
<tr>
<td>passed</td>
<td>#61733501 by ⭕️</td>
<td>dummy_data → f61b9a19</td>
<td>![Checkmark] ![Checkmark] ![Checkmark] ![Checkmark] ![Checkmark]</td>
<td>00:09:51</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Updated lark README.</td>
<td>![Checkmark] ![Checkmark] ![Checkmark] ![Checkmark] ![Checkmark]</td>
<td>20 hours ago</td>
</tr>
</tbody>
</table>
Impact on Outside Collaborators

- The project looks BIG.
  - Multiple layers of READMEs to get to a given product.
  - Other documentation has high surface area.
  - Developing on one application requires checking them all out.

- What if I want to fork the repo?
  - Running the pipelines and contributing MRs is hard; maybe prohibitively so.
  - Permanently forking one application would be a major pain.
    - (maybe this is a feature)
Questions & Discussion
Pipeline #61752888 passed with warnings for a019bc1b on reindex_event_table

Deployed to ucsd.review.lark/reindex... 17 hours ago

lark-ucsd-review-lark-sfhtz

version: "0.1.0"

ucsd.review.lark/reindex_event_table

<table>
<thead>
<tr>
<th>ID</th>
<th>Commit</th>
<th>Job</th>
<th>Created</th>
</tr>
</thead>
<tbody>
<tr>
<td>#183</td>
<td>reindex_event.. → a019bc1b</td>
<td>review_ucsd (#214099491) by 🌟</td>
<td>17 hours ago</td>
</tr>
<tr>
<td></td>
<td>Update README for SOLR re-indexing</td>
<td></td>
<td></td>
</tr>
<tr>
<td>#173</td>
<td>reindex_event.. → 498c339a</td>
<td>review_ucsd (#214033086) by 🌟</td>
<td>20 hours ago</td>
</tr>
<tr>
<td></td>
<td>Update README for SOLR re-indexing</td>
<td></td>
<td></td>
</tr>
<tr>
<td>#158</td>
<td>reindex_event.. → d99e59be</td>
<td>review_ucsd (#214003380) by 🌟</td>
<td>22 hours ago</td>
</tr>
<tr>
<td></td>
<td>Update README for SOLR re-indexing</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Continuous Integration

- Gitlab CI/CD Pipelines
  - Build, Code Quality, Dependency Scanning, Lint, Test, Deploy
- Docker image(s) per application
  - Lark API, Lark Frontend
  - Starlight
  - Shoreline
- Gitlab Image Registry
Continuous Delivery

- Cloud-native infrastructure
  - Docker, Kubernetes (k8s), Helm
- Auto-deploy across environments & campuses
  - Automated review deployments per merge request
  - Automated staging deployments on merge to mainline
  - Push-button production deployment
- Helm + k8s + 12-Factor means:
  - Deployments are repeatable
  - Applications are observable and scalable
  - Rollback is simple