

ACCESSIBLE CONTINUOUS INTEGRATION

A PRIMER



CivicActions

ACCESSIBLE CONTINUOUS INTEGRATION | OUTLINE

- Nerdstein**
- Role Playing**
- Continuous Integration**
- Current Limitations**
- Your Users**
- A Future Vision**
- Accessible Continuous Integration**
- Case Studies**
- A Call to Action**

Nerdstein (Adam)



- Associate Director of Engineering, CivicActions
- Masters of Science, Information Systems Security
- Drupal 8 Maintainer of Taxonomy Menu, Password Policy, Key, Encrypt, Field Encrypt

**Let's do some
role playing!**



KENNETH PARCELL

- 1. Intern in IT Department**
- 2. Hard-working, eager, team player, well loved, still learning IT**
- 3. Screwed up last three code releases**
- 4. Likes knitting and kite flying**



RON SWANSON

1. Runs IT Department

2. Non-technical

3. Low budget

4. Loves scotch and eats meat, actually hates IT



CRUELLA DEVILLE

1. Runs IT Department
2. Non-technical
3. Low budget
4. Loves scotch and eats meat, actually hates IT

Continuous Integration

The Slack conversations are getting ugly...

- Cruella: *Your next Drupal release better be accessible or I'll eat your babies*
- Ron: *Team, go figure this out. Mmm... scotch*
- Kenneth: *Now's my chance! I will make this right!*

IT'S LIFE OR DEATH!

Kenneth stumbles on a concept called Continuous Integration...

- He found an awesome quote to motivate him:
“Don’t solve the same problem more than once”
- Kenneth researches deployments and finds drupal.org articles and blog posts on the topic:
Use Drush to script deployments
- Kenneth develops a bash script of Drush commands to execute the deployments

Kenneth presents his work to Ron...

→ Ron responds, like a boss:

“I was close to firing you, so make sure other engineers can do this. Oh, and pour me more scotch.”

→ Kenneth researches collaborative, team-based tools for continuous integration:

Use Jenkins for distributed builds and standardize continuous integration centrally for a team.

→ Kenneth moves his bash scripts into a Jenkins server and provides his team access to manage and execute builds

ACCESSIBLE CONTINUOUS INTEGRATION | CONTINUOUS INTEGRATION



DevOps automates solutions to longstanding CI problems

- **Continuous learning into applied problem solving**
- **Consistency equates to predictability and stability**
- **Automation over error-prone manual processes**
- **Have no barriers:** *release management, security scanning, log analysis, 508 compliance, automated testing, quality assurance, code reviews, on-demand environments*

**You offer a better
service to your users
with CI practices**

Current Limitations

Kenneth runs the deployment successfully from Jenkins...

→ Kenneth emails Ron and Cruella:

“Your deployment is complete and it actually works!”

→ Cruella responds:

“You fools! I can’t believe you forgot to send me my accessibility report ONCE AGAIN! Why are we even paying you?!?!”

→ Ron chimes in while eating a bacon cheeseburger:

“NOMMMMMM Why can’t your fancy Jenkins tool let Cruella run her own damn accessibility reports, Kenneth?”

ACCESSIBLE CONTINUOUS INTEGRATION | CURRENT LIMITATIONS

Kenneth creates a Jenkins job to execute accessibility scans and provides Cruella access to Jenkins...

→ Cruella has a nuclear meltdown:

“What kind of insane technical system is this? Who do you think I am, Einstein?”

→ Kenneth hangs his head in shame

→ Ron adds yet another item into his list of customer complaints:

- > Your security scanning reports are long and illegible*
- > I can't effectively test my code before it goes to production*
- > I have thousands of unread emails, why do you continue to email me?*

We must evolve continuous integration beyond just a technical practice

- Technical people producing technical solutions is not enough for those we serve**
- The outcomes we desire is better service**
- Services are for our customers**
- Can we at least agree that those we serve should be involved in the conversation?**

**We're missing those
we serve in the process**

**How do you know
your actually improving?**

Your Users

**So... what do your
users need?**

- **They need confidence in your services**
- **They need continued assurance that you are making the correct decisions on their behalf**
- **They need continued trust and that trust is very easily eroded if your services cause users issues**

**So... what do your
users expect?**

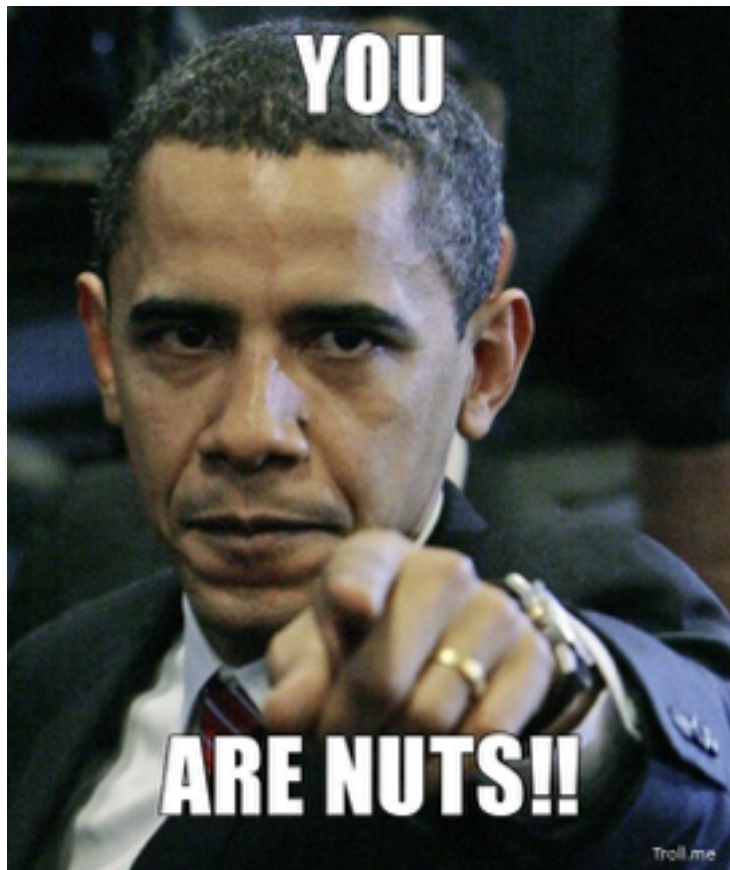
They assume you know and follow best practices

- They want no surprises to arise because you didn't**
- This might not be visible to them**
- Examples:**
 - > Regular site and system updates**
 - > Accessibility and compliance**
 - > Proper permissions and capabilities**

They assume it's comprehensive

- Developing any feature they could ever imagine or see on Facebook (for free, mind you)**
- Examples:**
 - > Visually perfect**
 - > Complete feature parity**
 - > Secure and performant**
 - > Immaculately maintained**

ACCESSIBLE CONTINUOUS INTEGRATION | YOUR USERS



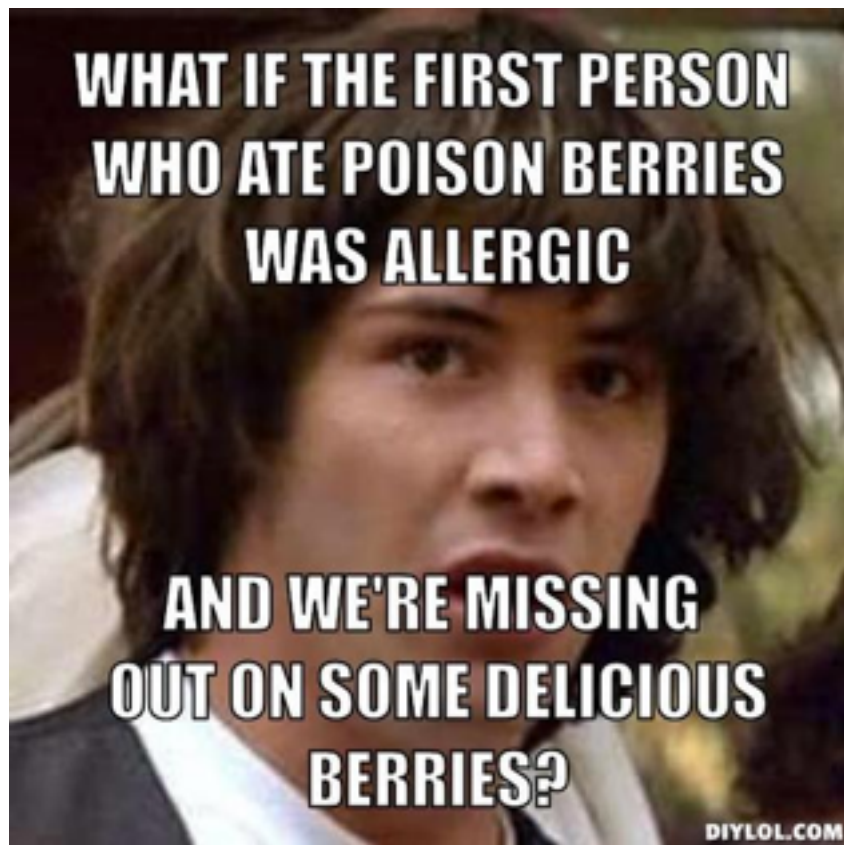
A Future Vision

Innovation is not just technical breakthroughs

- The problem is not the tools, it's that they are not approachable**
- Open a dialogue with those you serve**

Build both technical and social bridges

ACCESSIBLE CONTINUOUS INTEGRATION | CONTINUOUS INTEGRATION



Digital Enablement

Let's promote digital enablement for Continuous Integration...

- Emphasize ease of adoption and removal of barriers to entry**
- Promote effective information sharing and transparency**
- Drive toward usability of your services and tools**
- Deliver comprehensive and streamlined services**

How can we frame high level goals...

- SIMPLE - Processes void of encumbrance**
- USEFUL - Solve meaningful problems**
- FLEXIBLE - Build robust, long-term, unassuming solutions**
- TRANSPARENT - Communicate concisely and frequently**

Take Continuous Integration to the masses



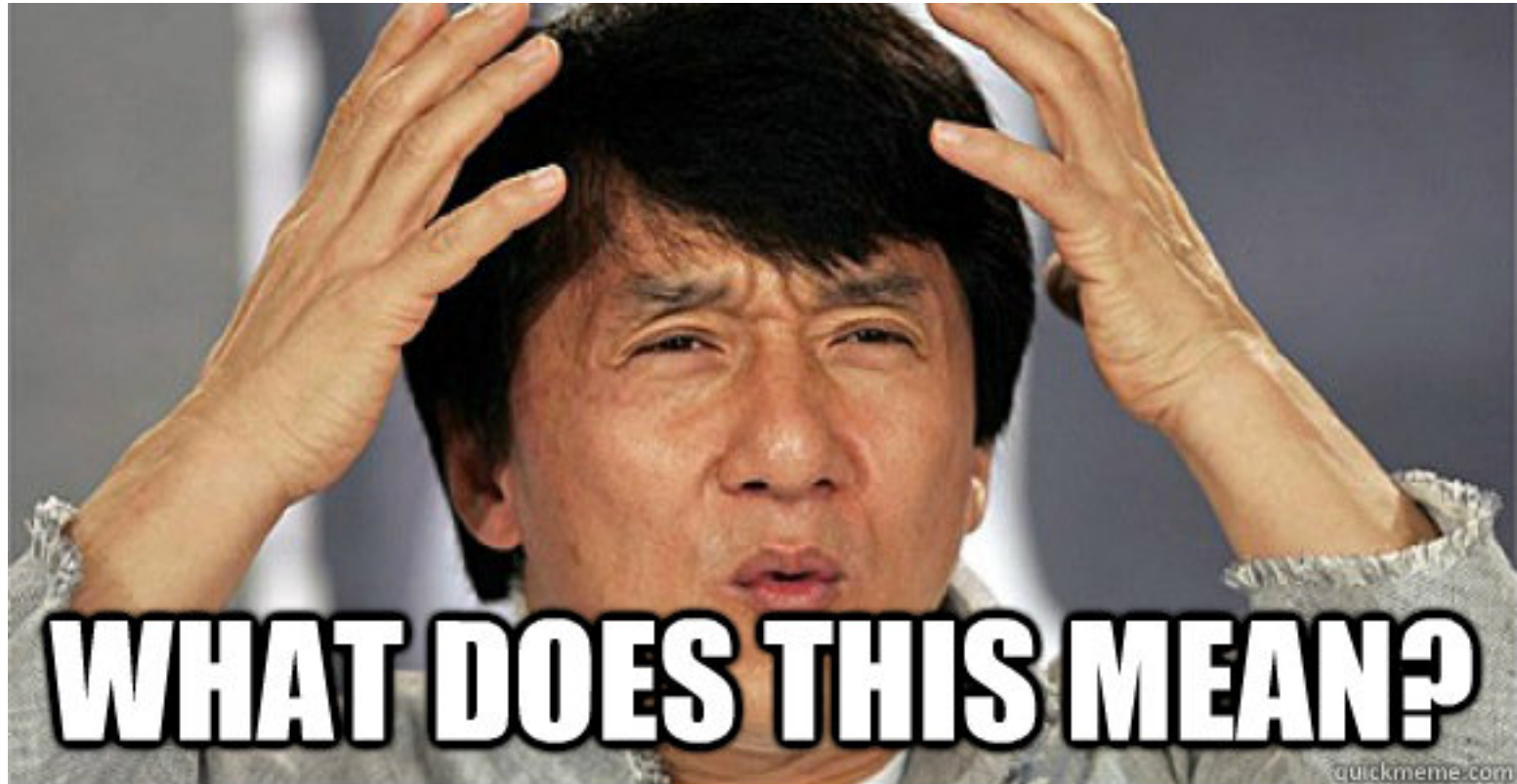
Onward!

Accessible Continuous Integration

Wikipedia defines accessibility as...

The process of creating products that are usable by people with the widest possible range of abilities, operating within the widest possible range of situations.

Do not get confused with **508 Compliance *Accessibility***



There are current practices that build bridges...

- Abstracting technical details**
- Systems integration**
- Streamlined processes**
- Platform and device agnostic**

- 1. Abstracting technical details**
2. Integrate systems
3. End-user involvement
4. Platform and device agnostic

Abstracting technical details...

- KISS concept (Keep it simple, stupid)**
- Systems must promote usability, account for technical literacy**
- Build finely tuned user interfaces, not command lines**
- Limit decision points, add help text, consistent UI design**

1. Abstracting technical details
- 2. Integrate systems**
3. End-user involvement
4. Platform and device agnostic

Integrate systems...

- Select systems that promote interoperability (future proof for continuous learning)**
- Connect systems instead of forcing users to use multiple systems**
- Make use of web services, APIs, plugin systems, and give back to communities so others can benefit**

1. Abstracting technical details
2. Integrate systems
- 3. End-user involvement**
4. Platform and device agnostic

End-user involvement...

- Identify systems in which your users are comfortable using -- enhance them**
- Avoid forcing users to learn too much or use new systems**
- Encourage user testing and feedback loops to participate in Continuous Integration discussions**

1. Abstracting technical details
2. Integrate systems
3. End-user involvement
- 4. Platform and device agnostic**

Platform and device agnostic...

- Users want access everywhere and immediately**
- Avoid systems (like email) where communication can break down**
- Adopt best of breed solutions that don't restrict platforms**
- Systems should not only be on desktops, use of mobile phones, tablets, and refrigerators (Internet of Things)**

Tear down the walls of CI participation



Case Studies

Accessible CI in the wild...

1. Slack and Jenkins

2. On-demand Environments

3. Data Analysis

4. JIRA, BDD and Repositories

Slack and Jenkins is a happy marriage...

- Slack is highly intuitive for non-technical users
- Slack supports custom commands with help text
- Example: Run a security scan from Slack, send condensed summary back to Slack with results
- Commands can include running Jenkins commands
 - ◆ *Abstract parameters and options into separate commands*
 - ◆ *Customize output to Slack so it's relevant to all users*

Accessible CI in the wild...

1. Slack and Jenkins

2. On-demand Environments

3. Data Analysis

4. JIRA, BDD and Repositories

On-demand environments help QA and user engagement...

- Helps project managers and product owners perform testing before releasing code**
- Leverage environment hooks to produce environments for new branches**
- Integrate with a Slack command to produce URL and credentials**
- Simplifies release management for code staging instead of releasing code bundles**

Accessible CI in the wild...

1. Slack and Jenkins
2. On-demand Environments
- 3. Data Analysis**
4. JIRA, BDD and Repositories

Promote transparency through data analysis...

- Put data at your users fingertips, saves email, development time, requests, etc.**
- Provide dynamic dashboards that SUMMARIZE effective information**
- Platform stability, health, and performance metrics (red, yellow, green)**
- Analytics and end-user behaviors (traffic patterns, most viewed, least viewed, etc)**
- Log analysis, trends, and insights**

Accessible CI in the wild...

1. Slack and Jenkins
2. On-demand Environments
3. Data Analysis
- 4. JIRA, BDD and Repositories**

Test driven development practices can be improved by using JIRA and Repo Hooks...

- Development activities can be more secured when validated! Manual testing is not comprehensive and error prone
- Behave for JIRA plugin, integrates with code repository
- Empower users to write automated tests within JIRA tickets
- Leverage hooks in the repository to run automated tests when developers submit pull requests (TravisCI or Jenkins)
- Feedback loops are drastically shortened between user needs and developer's code

A Call to Action

Unlock the potential

Empower your users

Accessible Continuous Integration

ACCESSIBLE CONTINUOUS INTEGRATION | A CALL TO ACTION



Thank you, Drupal North!

Questions?