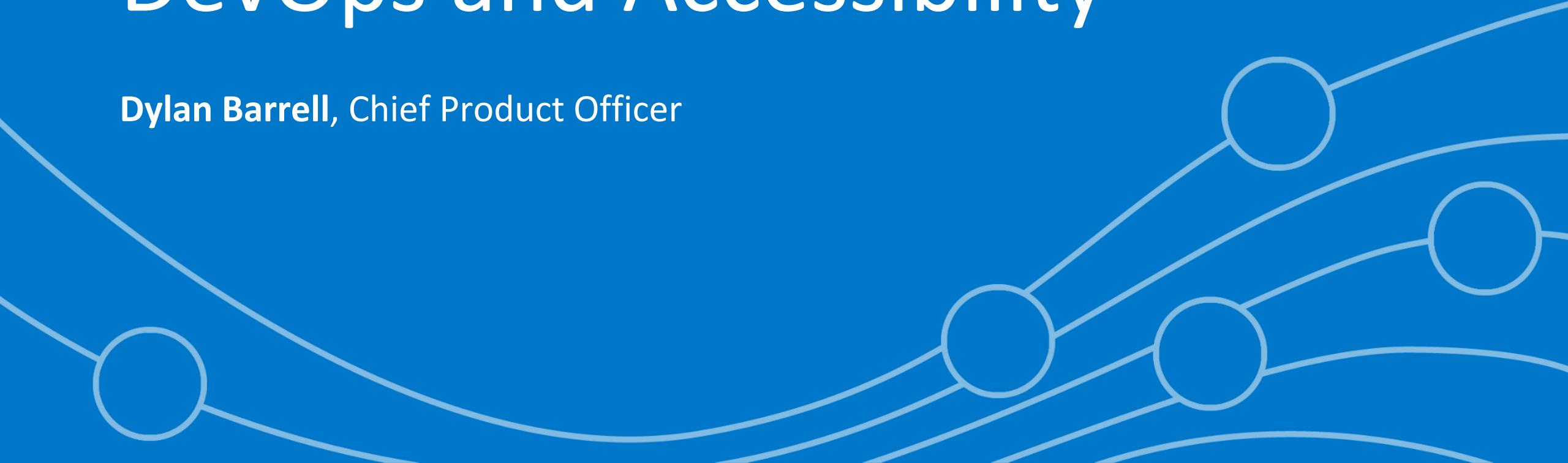




DevOps and Accessibility

Dylan Barrell, Chief Product Officer



Deque

Founded: 1999

Mission: Digital Equality

Focus: To achieve sustainable accessible software development through the application of tools and processes

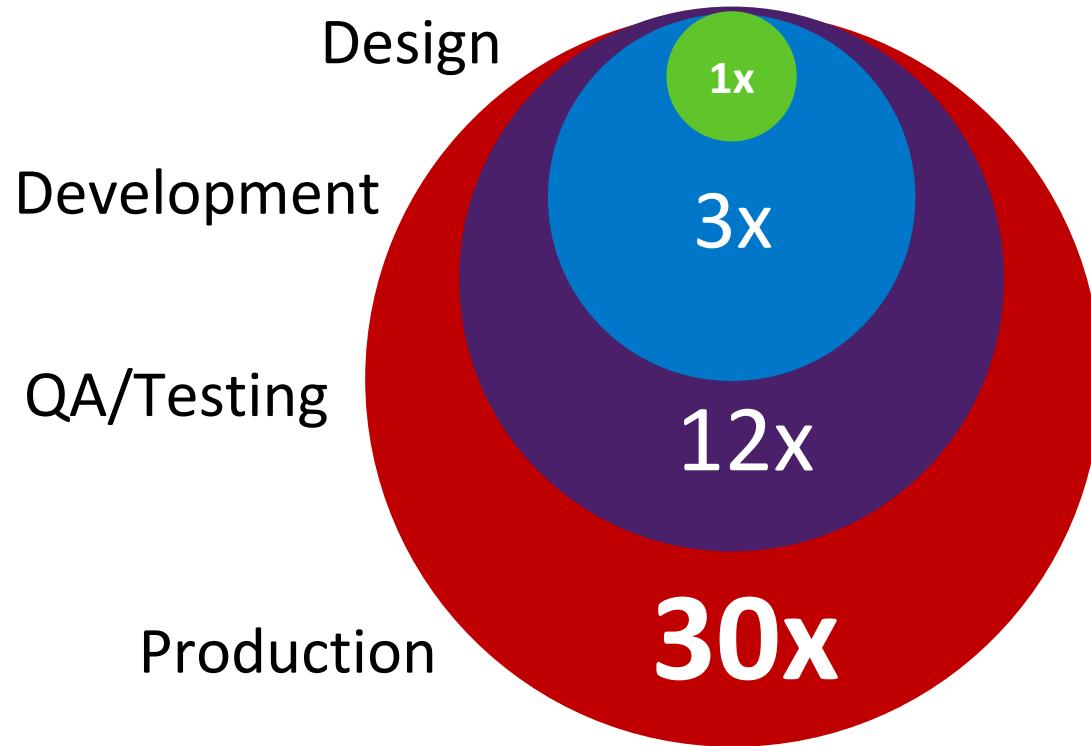
Technologies: axe® suite of tools for automating accessibility testing

Value to the business

- **Increase revenue:** 73% of Americans are influenced by accessibility
- **Lower cost:** Digital sales channels cost far less to run than physical
- **Avoid risk:** Lawsuits and brand damage

What about value to development?

Value in DevOps: Where you find and fix issues matters



What is WCAG?

- Web Content Accessibility Guidelines
- 1.0 published May 1999
- 2.0 published December 2008
 - 2.1 June 2018
 - 2.2 currently in final draft
- WAI-ARIA 1.0 published March 2014
 - added semantics that were/are not available in HTML
- Four principles
 - Perceivable
 - Operable
 - Understandable
 - Robust

Perceivable Example



Operable Example

The screenshot shows the homepage of the deque website. The header features the deque logo and a search bar. Below the header is a navigation menu with links to Tools, Services & Training, Solutions, Resources, Compliance, and Company. A large white callout box is overlaid on the page, containing sections for Tools, Why Axe, and Get Started.

Tools

- axe DevTools**
Prevent web & mobile issues
- axe Auditor**
Full coverage manual audits
- axe Monitor**
Sitewide scanning & reporting

Why Axe

- axe Tools Overview**
Full service platform
- AI**
More automation, less false positives
- Standards**
The world relies on axe-core

Get Started

- axe DevTools Browser Extension**
- axe DevTools Mobile**
- axe DevTools Linter**
- axe Developer Hub (Beta)**
- Axe for Designers (Beta)**
- Deque University**

No-code mobile app testing is here.
[Start testing across every technology and platform now.](#)

Operable Example

The screenshot shows the homepage of the deque website. The header features the deque logo and a search bar. Below the header is a navigation menu with links: Tools, Services & Training, Solutions, Resources, Compliance, and Company. A dropdown menu is open under the Tools link, containing the following items:

- axe DevTools: Prevent web & mobile issues (icon: code editor)
- axe Auditor: Full coverage manual audits (icon: magnifying glass)
- axe Monitor: Sitewide scanning & reporting (icon: waveform)
- Why Axe
 - axe Tools Overview: Full service platform (icon: triangle)
 - AI: More automation, less false positives (icon: robot)
 - Standards: The world relies on axe-core (icon: gear)
- Get Started
 - axe DevTools Browser Extension (icon: plug)
 - axe DevTools Mobile (icon: smartphone)
 - axe DevTools Linter (icon: exclamation mark)
 - axe Developer Hub (Beta) (highlighted in dark blue box, icon: network)
 - Axe for Designers (Beta) (icon: network)
 - Deque University (icon: graduation cap)

At the bottom left, there is a promotional banner with the text "No-code mobile app testing is here." and a link "Start testing across every technology and platform now." with an icon of a smartphone.

Some interesting statistics

- 50 success criteria (SC) in WCAG 2.1 A and AA
- Just 3 account for ~60% of defects
- 6 for ~80% of defects
- **Only 15 success criteria account for ~95% of all defects**

Table 1: WCAG Success Criteria with the Most Issues

#	Success Criteria #	Success Criteria Name	Total issues	Manual issues	Auto issues	Manual %	Auto %	% of ALL Issues by SC	Cumulative % of Issues
1	1.4.3	Contrast (Minimum)	88,714	14,981	73,733	16.89%	83.11%	30.08%	30.08%
2	4.1.2	Name, Role, Value	48,287	22,011	26,276	45.58%	54.42%	16.37%	46.45%
3	1.3.1	Info and Relationships	36,382	19,950	16,432	54.83%	45.17%	12.33%	58.78%
4	4.1.1	Parsing	34,488	3,351	31,137	9.72%	90.28%	11.69%	70.47%
5	1.1.1	Non-text Content	23,701	7,687	16,014	32.43%	67.57%	8.04%	78.51%
6	2.4.3	Focus Order	9,553	9,553	0	100.00%	0.00%	3.24%	81.75%
7	2.1.1	Keyboard	9,412	9,178	234	97.51%	2.49%	3.19%	84.94%
8	2.4.7	Focus Visible	7,312	7,312	0	100.00%	0.00%	2.48%	87.42%
9	1.4.11	Non-text Contrast	4,539	4,539	0	100.00%	0.00%	1.54%	88.96%
10	1.4.1	Use of Color	3,713	3,261	452	87.83%	12.17%	1.26%	90.22%
11	1.3.2	Meaningful Sequence	3,313	3,313	0	100.00%	0.00%	1.12%	91.34%
12	3.3.2	Labels or Instructions	2,537	2,019	518	79.58%	20.42%	0.86%	92.20%
13	2.4.1	Bypass Blocks	2,533	532	2,001	21.00%	79.00%	0.86%	93.06%
14	2.4.2	Page Titled	2,211	1,962	249	88.74%	11.26%	0.75%	93.81%
15	3.1.1	Language of Page	2,173	178	1,995	8.19%	91.81%	0.74%	94.54%
	#.#.#	Rest of WCAG 2.1 A/AA SC	16,090	15,889	201	98.75%	1.25%	5.46%	100.00%
		Totals	294,958	125,716	169,242	42.62%	57.38%		

What tools can help

- axe for Designers - Figma plugin for testing designs and communicating design intent
- axe DevTools Linter - integrate into IDE and CI/CD to find accessibility issues while coding
- axe Developer Hub - no code leveraging of e2e tests for accessibility



DevOps and Accessibility

A Case Study of USAA's Success

Steven Rambach
Senior Software Engineer



THE



WAY

OUR MISSION

is to empower our members to achieve financial security through highly competitive products, exceptional service and trusted advice. We seek to be the #1 choice for the military community and their families.

OUR CORE VALUES

Service. Loyalty. Honesty. Integrity.

OUR BEHAVIORS

We take ownership.

We strive for excellence.

We foster belonging.

OUR PERFORMANCE STANDARDS

Exceptional Service Team for members.

Winning Financial Strength for employees. association.

Community Impact for the for external stakeholders.

OPERATING WITHIN RISK APPETITE

Accessibility in the past

- Manual consultation
- Limited hands-on manual tooling
- Manual testing attestation
- WCAG version mismatches

Transitioning to CI environments

- Streamlining & automating
 - Code review and deploy
 - Vulnerability management
 - Automated testing
 - Unit/Integration/Functional
 - Security
 - Performance
 - *Accessibility*
 - Automated test evidencing
 - ...



Generic diagram for CI things

<https://www.toptal.com/devops/effective-ci-cd-deployment-pipeline>

How can USAA integrate accessibility testing?

- Minimal developer disruption
- Hands-off evidencing



Generic stock image of guy thinking

Accessibility in CI – Strategy

- Minimal developer impact
 - Utilize pre-existing tests
 - DOM readily available
- Evidence testing
 - Test XML's already supported

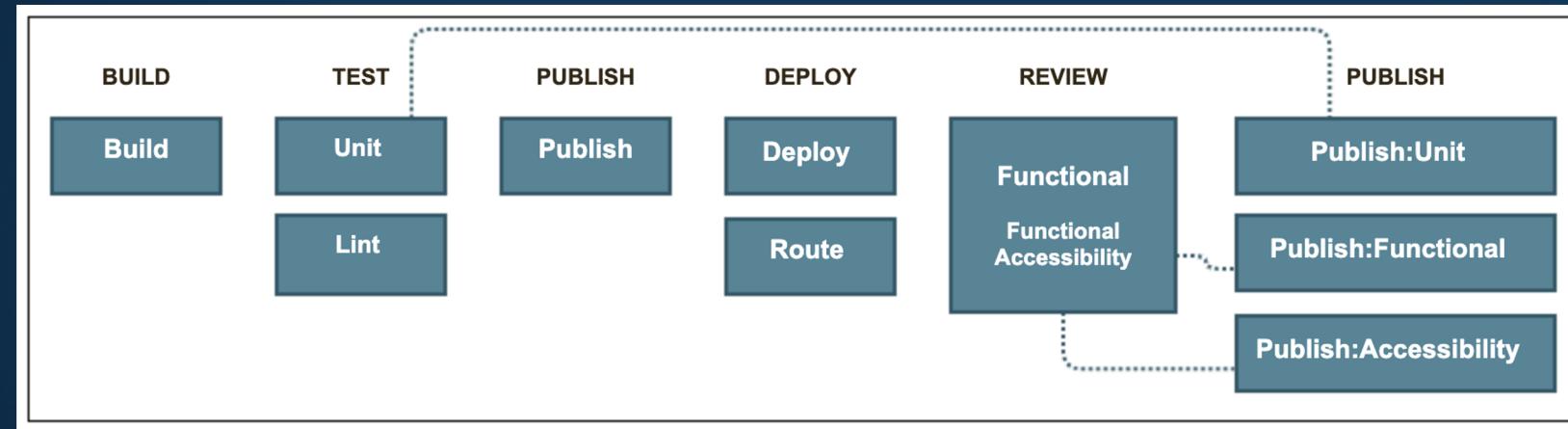
Accessibility in CI – Execution

- One-line page scan
 - No new tests
 - No new frameworks
 - One-time setup
 - No new executions
- Export and evidence accessibility testing findings
 - Leverage existing test publishing

Accessibility in CI – Execution

```
it('should click a link', async () => {  
  
  await $('=Standard Template').click();  
  expect(await browser.getTitle()).toContain('Standard Template');  
  //Scans the current page state for accessibility issues  
  await browser.axeAnalyze('homepage');  
  
});
```

Accessibility in CI – Execution



What have we solved?

- Accessibility scans are now automated in a CI setting
 - Minimal code change
 - No extra tests required
- Evidencing of scans is present, and automated

What's next?

- Expand automated testing past WCAG coverage
- Detect issues earlier than code builds
 - Linters
 - Design stages
- Streamline WCAG scans even further
 - One line of code is great, but what about zero?

