



DIY Accessibility

Kelly Hutchins

Tao Zhang

2018 Esri DEVSummit Conference | Palm Springs, CA

What is accessibility?

- Make content usable by as many people as possible
- About 15% of world population lives with some form of disability: **1 billion** people
- In the US, 1 in 5 adults has a disability
- Disabilities could be temporary or situational

Why is accessibility important?

- People with disabilities deserve equal rights
- The ADA and Section 508
- Accessible interface is about good design and coding practice
- Good accessibility is good user experience

What are we doing?

- Better knowledge sharing
- Review products internally for compliance
- Working accessibility into new features

The screenshot shows the Esri GeoNet website. The top navigation bar includes 'esri', 'ArcGIS', 'Industries', 'About', and 'Support'. The main header features the 'GeoNet The Esri Community' logo and a search bar. Below the header, there are navigation links for 'Home', 'News', 'Communities', 'All Content', 'ArcGIS Ideas', and 'GeoNet Help'. The main content area is titled 'Accessibility' and includes a sub-navigation bar with 'Overview', 'Activity', 'Content', 'People', and 'Calendar'. A 'Log in' button is visible in the top right corner. The main content area contains a 'Welcome to Accessibility' message, a search bar for 'ASK ACCESSIBILITY', and sections for 'GROUP ADMINS', 'LATEST BLOG POSTS', and 'LATEST DISCUSSIONS'.

The screenshot shows the Esri ArcGIS Blog website. The top navigation bar includes 'esri', 'ArcGIS', 'Industries', 'About', and 'Support'. The main header features the 'ARCGIS BLOG' logo and a search bar. The main content area is titled 'New accessibility features in Story Map Journal' and includes a byline 'by Owen Evans on December 7, 2017'. The article text discusses accessibility features in the Story Map Journal app and provides information on what accessibility means and alternative text. The article includes social sharing buttons for Facebook, Twitter, and LinkedIn. The bottom of the page features a section titled 'Alternative Text' with a detailed explanation of the concept.

New accessibility features in Story Map Journal

by Owen Evans on December 7, 2017

Accessibility is an important goal at Esri, and in the last two ArcGIS Online updates (September and December 2017), the Story Map Journal app received significant enhancements related to accessibility. Read on to learn more about these new features and how they work to help you create accessible stories.

What is accessibility?

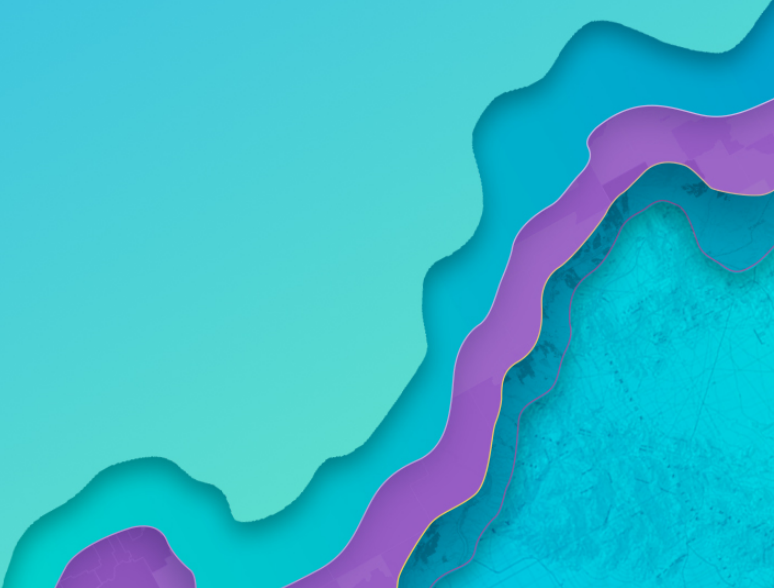
If something is *accessible* it means that people with disabilities can use it. Just like a public building can be made accessible by the installation of an entrance ramp, web content, such as a story map, can also be made accessible by the addition of some key pieces of infrastructure.

Most organizations have accessibility guidelines for web content, and many are required by law to adhere to certain accessibility requirements. For example, web content published by US federal government agencies is subject to [Section 508 standards](#). The World Wide Web Consortium (W3C) also publishes guidelines for web accessibility known as the [Web Content Accessibility Guidelines \(WCAG\)](#).

Alternative Text

Alternative text — or "alt text," as it is often called — provides a textual alternative to web content such as images, maps, videos, charts, and other visual media. You can now use the Map Journal builder to attach alt text to media in the main stage and side panel of your stories. This includes media that appears in the main stage via a [story action](#).

What we will cover today

- Background
 - Automated test
 - Keyboard test
 - Screen reader test
 - Color accessibility
- 

Why accessibility testing?

- Accessibility is about the *experience* of all users.
- Testing is the only way to ensure the experience is accessible.

Functional test

Specification	WCAG 2.0 Success Criteria
Goal	Verify how well web content functions as WCAG 2.0 specified

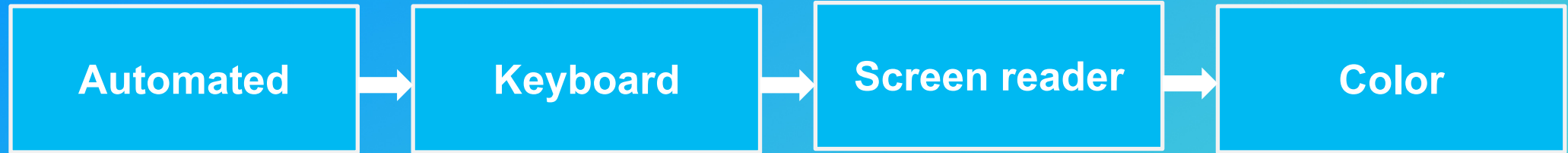
Less subjective compared to usability testing

Anyone can do the test!

Overview of WCAG 2.0

Principles	Success Criteria	Level A	Level AA	Level AAA
1. Perceivable	1.1 Text Alternatives	1.1.1		
	1.2 Time-based Media	1.2.1 – 1.2.3	1.2.4 – 1.2.5	1.2.6 – 1.2.9
	1.3 Adaptable	1.3.1 – 1.3.3		
	1.4 Distinguishable	1.4.1 – 1.4.2	1.4.3 – 1.4.5	1.4.6 – 1.4.9
2. Operable	2.1 Keyboard Accessible	2.1.1 – 2.1.2		2.1.3
	2.2 Enough Time	2.2.1 – 2.2.2		2.2.3 – 2.2.5
	2.3 Seizures	2.3.1		2.3.2
	2.4 Navigable	2.4.1 – 2.4.4	2.4.5 – 2.4.7	2.4.8 – 2.4.10
3. Understandable	3.1 Readable	3.1.1	3.1.2	3.1.3 – 3.1.6
	3.2 Predictable	3.2.1 – 3.2.2	3.2.3 – 3.2.4	3.2.5
	3.3 Input Assistance	3.3.1 – 3.3.2	3.3.3 – 3.3.4	3.3.5 – 3.3.6
4. Robust	4.1 Compatible	4.1.1 – 4.1.2		

Test process



Automated test

The background features a teal-to-blue gradient. In the upper right, there are faint, concentric white wavy lines. In the lower left, there are layered, wavy shapes in shades of purple and teal, resembling a stylized landscape or topographic map.

The screenshot shows the aXe accessibility testing tool interface. The top navigation bar includes tabs for Elements, Console, Sources, Network, Performance, Memory, Application, Security, Audits, aXe-Coconut, and aXe. The aXe logo is in the top left. A sidebar on the left shows a list of violations: 'Violations found' (2), 'Needs review' (3), 'Page must have means to bypass repeated blocks' (1), 'Page must contain one main landmark' (1), and 'Elements must have sufficient color contrast' (3). The main panel displays the details for the selected violation: 'Page must have means to bypass repeated blocks'. It includes an 'Issue description' (Ensures each page has at least one mechanism for a user to bypass navigation and jump straight to the content), an 'Impact' of 'serious', and a 'Learn more' link. The 'Element location' is 'html' and the 'Element source' is '<html lang="en" class="dj_edge dj_gecko dj_ff1 dj_contentbox">'. The 'To solve this violation, you need to:' section lists three issues: 'No valid skip link found', 'Page does not have a header', and 'Page does not have a landmark region'. The bottom of the panel shows 'Issue tags: category: keyboard wcag2a wcag241 section508 section508.22.o'.

- Tests rendered browser DOM
- Aims at no false positives
- Accessible
- Helpful documentation

Practice aXe

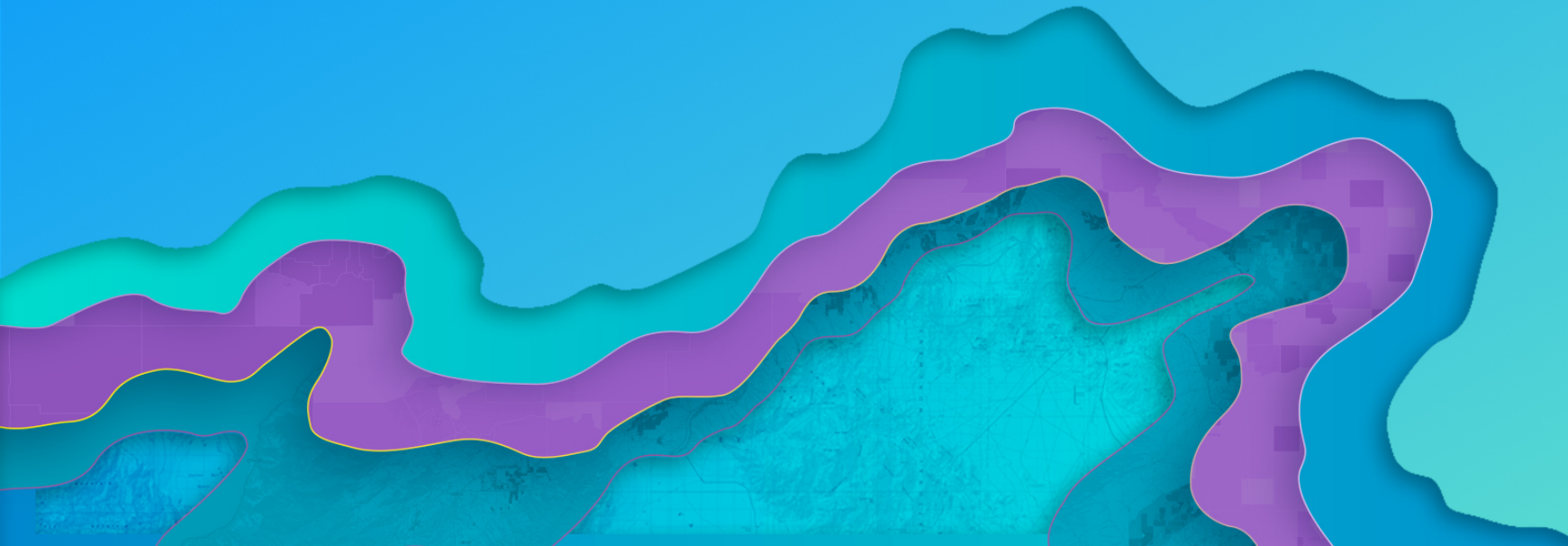
Park Locator

<http://arcg.is/05DzDX>

Automated test

- No automated test tools can definitely prove conformance with any given WCAG Success Criterion.
- Automated test is good starting point but cannot detect all accessibility issues.
- Run automated test of each page state.

Keyboard test



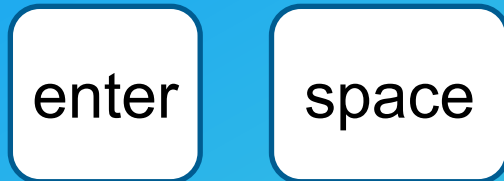
Keyboard navigation



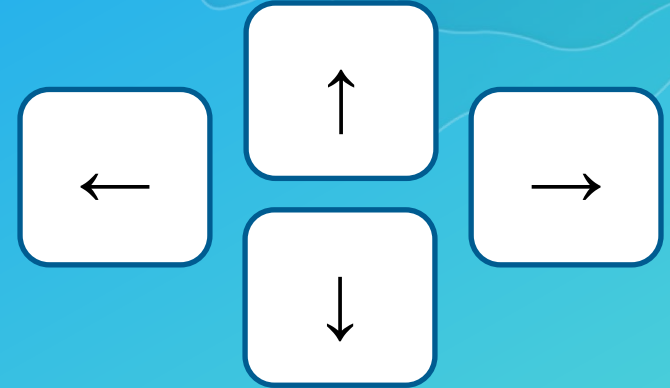
Move keyboard focus



Click links



Click buttons



Menus and some form controls

Expected outcomes

- 2.1.1: Interact with all controls, links, and menus using only keyboard.
- 2.4.7: See what item has focus at all times.
- 2.4.3: Visual focus order matches intended interaction order.
- 2.1.2: No keyboard trap.
- Off-screen content (e.g., responsive navigation) should not receive focus when invisible.

Practice keyboard test

Test cases

- [Park Locator](#)
- [UK Government Digital Service – Test cases](https://arcg.is/1OLC8K) (https://arcg.is/1OLC8K)
- [Accessible University](http://arcg.is/1O45jz) (http://arcg.is/1O45jz)

Tools

- [Enhanced focus](http://arcg.is/19muKy) (http://arcg.is/19muKy)
- `document.activeElement`

Screen reader test



Screen reader

Recommended combinations:

OS	Screen reader	Browser
MacOS	<u>VoiceOver</u>	Safari
Windows	<u>NVDA</u>	Firefox
Windows	<u>JAWS</u>	IE/Edge

Screen reader

	Turn on	Stop	Modifier key
VoiceOver	Command + F5	Command + F5	Control + Option
NVDA	Control + Alt + N	NVDA + Q	Numpad Insert
JAWS	Control + Alt + J	Insert + F4	Numpad Insert

Modifier key: Enter screen reader commands by pressing modifier key and one or more other keys

Screen reader commands

- VoiceOver
- NVDA
- JAWS
- Basic commands for accessibility testing

VoiceOver commands

VO + right/left arrow

Read next/previous item

Control

Stop reading

VO + space

Click link, button, form controls

VO + u

Open rotor

Screen reader testing coverage

Navigation

Headings

Links

Landmarks

Menus

Content

Alt text

Tables

Charts

Interaction

Forms

Dialogs

Messages

Widgets

WAI-ARIA Authoring Practices

TABLE OF CONTENTS

- 1. **Introduction**
- 2. **Read Me First**
 - 2.1 No ARIA is better than Bad ARIA
 - 2.2 Browser and Assistive Technology Support
 - 2.3 Mobile and Touch Support
- 3. **Design Patterns and Widgets**
 - 3.1 Accordion (Sections With Show/Hide Functionality)
 - 3.2 Alert
 - 3.3 Alert and Message Dialogs
 - 3.4 Breadcrumb
 - 3.5 Button
 - 3.6 Checkbox
 - 3.7 Combo Box
 - 3.8 Dialog (Modal)
 - 3.9 Disclosure (Show/Hide)
 - 3.10 Feed
 - 3.11 Grids : Interactive Tabular Data and Layout Containers
 - 3.12 Link
 - 3.13 Listbox
 - 3.14 Menu or Menu bar
 - 3.15 Menu Button
 - 3.16 Radio Group

WAI-ARIA Authoring Practices 1.1

W3C Working Group Note 14 December 2017



This version:

<https://www.w3.org/TR/2017/NOTE-wai-aria-practices-1.1-20171214/>

Latest published version:

<https://www.w3.org/TR/wai-aria-practices-1.1/>

Latest editor's draft:

<https://w3c.github.io/aria-practices/>

Previous version:

<https://www.w3.org/TR/2017/WD-wai-aria-practices-1.1-20170628/>

Editors:

Matt King, Facebook, mck@fb.com
James Nurthen, Oracle Corporation, james.nurthen@oracle.com
Michiel Bijl, Invited Expert
[Michael Cooper](mailto:cooper@w3.org), W3C, cooper@w3.org
Joseph Scheuhammer, Inclusive Design Research Centre, OCAD University (Previous Editor)
Lisa Pappas, SAS (Previous Editor)
Rich Schwerdtfeger, IBM Corporation (Previous Editor)

Copyright © 2015-2017 W3C® (MIT, ERCIM, Keio, Beihang). W3C liability, trademark and permissive document license rules apply.

Abstract

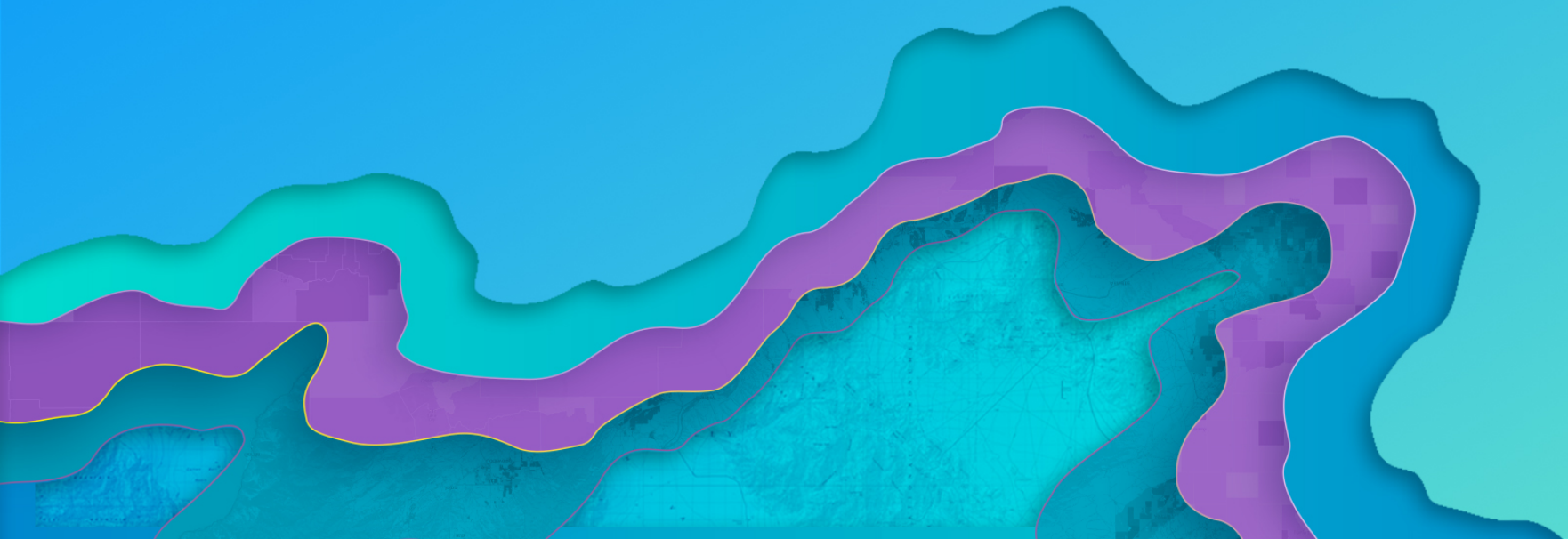
This document provides readers with an understanding of how to use [WAI-ARIA 1.1](#) [[wai-aria-1.1](#)] to create accessible rich internet applications. It describes considerations that might not be evident to most authors from the [WAI-ARIA](#) specification alone and recommends approaches to make widgets, navigation, and behaviors accessible using [WAI-ARIA](#) roles, states, and properties. This document is directed primarily to Web application developers, but the guidance is also useful for user agent and assistive technology developers.

Practice screen reader

Park Locator

<http://arcg.is/05DzDX>

Color test



Expected outcomes

- 1.4.1: Not use presentation that relies solely on color.
- 1.4.3: Color contrast ratio is at least 4.5:1.

Practice color test

Test cases

- Park Locator

Tools

- Contrast ratio calculator
(<http://arcg.is/1m44TW>)

Summary

- Start with automated test, then do keyboard, screen reader, and color test.
- Need to understand [WCAG 2.0 Success Criteria](#).
- Get familiar with ARIA for widgets.

The ultimate decision-maker about whether or not something is accessible, is whether or not people can use it.

Want to learn more about accessibility?

Accessible Web Mapping Apps: ARIA, WCAG and 508 Compliance

Kelly Hutchins, Tao Zhang

8:30 – 9:30 am
Friday, March 9

Smoketree A-E

Schedule

Hands-on Workshops

Tuesday, March 6

1:00 – 2:00 PM

Introducing UX to your GIS Org
Mesquite B

4:00 – 5:00 PM

DIY Usability Testing
Mesquite C

Thursday, March 8

9:00 – 10:00 AM

DIY Accessibility
Mesquite B

1:00 – 2:00 PM

**ArcGIS Runtime: Building Great
User Experience**
Smoketree A-E

2:30 – 3:30 PM

Collaborative Brainstorming
Mesquite G-H

4:00 – 5:00 PM

**Customizing the ArcGIS API for
JavaScript Widgets**
Primrose A

Friday, March 9

8:30 – 9:30 AM

Accessible Web Mapping Apps

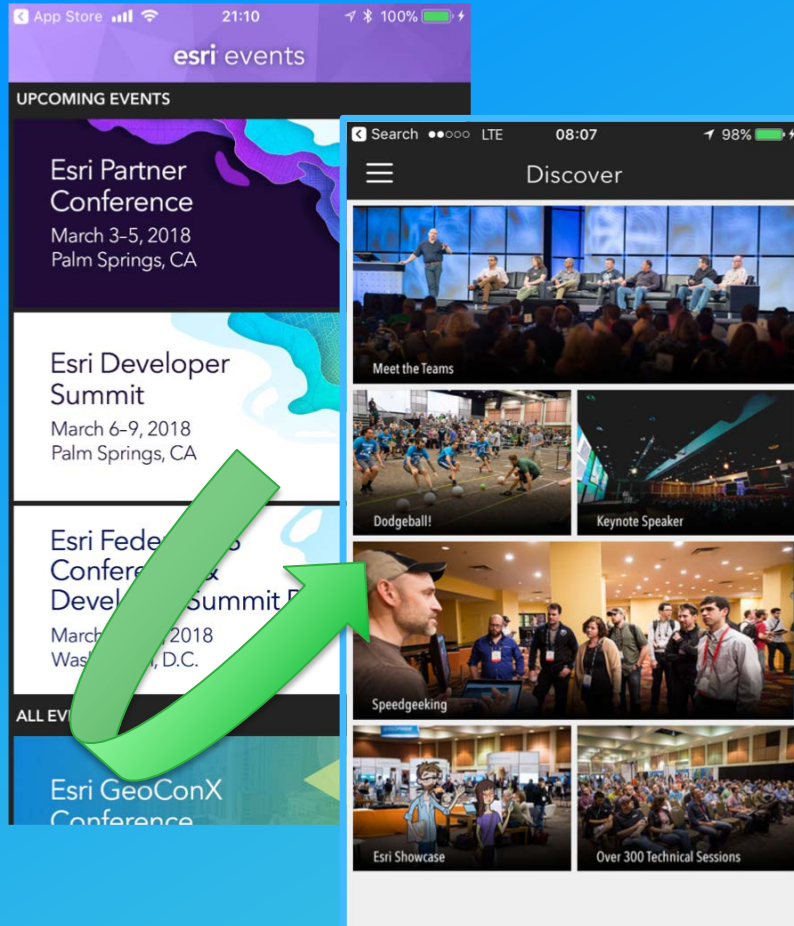
10:00 – 11:00 AM

DIY Usability Testing
Smoketree A-E

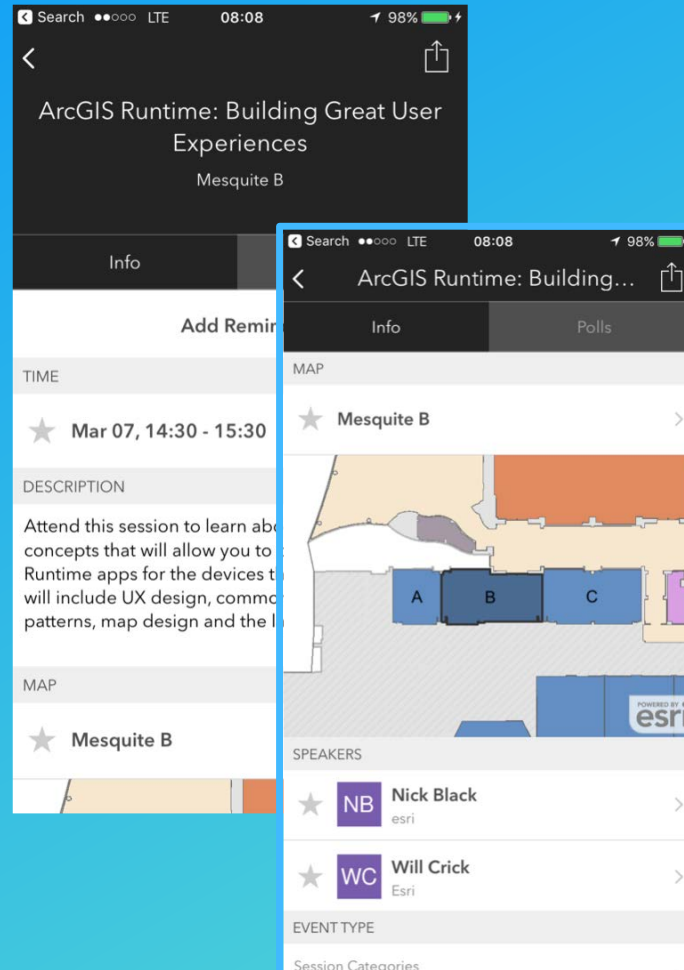


Please Take Our Survey!

Download the Esri Events app
and find your event

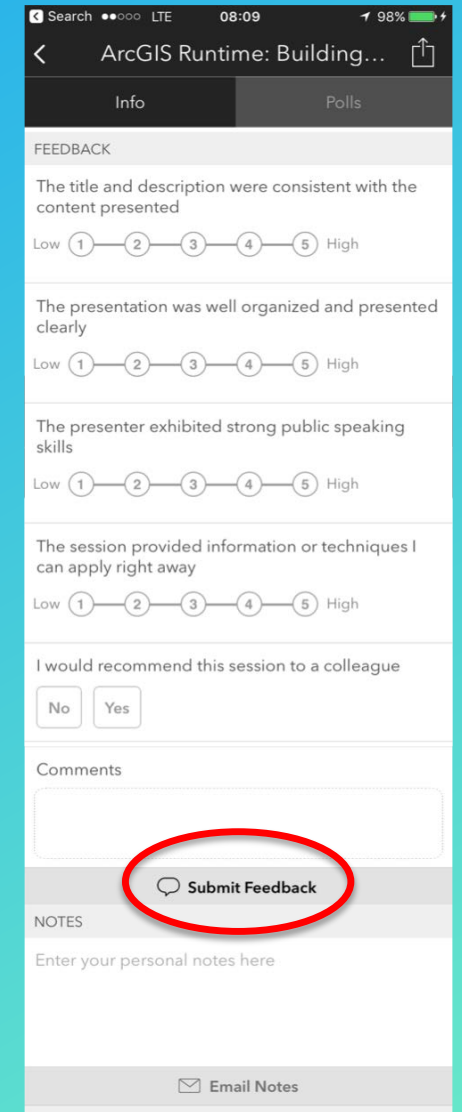


Select the session you
attended



Scroll down to the
“Feedback” section

Complete Answers,
add a Comment,
and Select “Submit”





esri

THE
SCIENCE
OF
WHERE