Accessible tools and data: Lessons from the Web

Andy Boughton

Abecasis Group Meeting

July 20, 2021

When everything is digital, who gets left behind?

"...websites, tools, and technologies are designed and developed so that people with disabilities can use them. More specifically, people can:
- perceive, understand, navigate, and interact with the Web
- contribute to the Web

Web accessibility encompasses all disabilities that affect access to the Web, including:
- auditory [3.1%]
- cognitive [6.3%]
- neurological
- physical [8.2%]
- speech
- visual [3.3%]"

[1] https://www.w3.org/WAI/fundamentals/accessibility-intro/
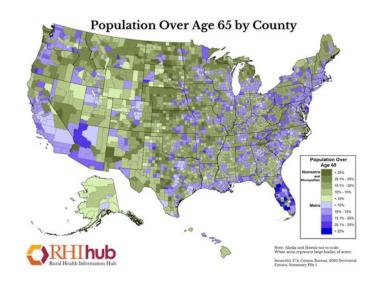[2] https://www.interactiveaccessibility.com/accessibility-statistics (2012)

"Section 504 prohibits discrimination based on disability by federal agencies and recipients of federal assistance... applies to facilities... communications such as websites. So, if your organization receives federal funding or assistance, your website is required to be accessible.

the DOJ ... was considering proposing WCAG 2.0 Level AA as the accessibility standard for websites and web content.... The Revised 508 Standards are based on WCAG 2.0.
However, a final rule specifying technical standards under the ADA has not been adopted. ... if you're subject to the ADA, you have more flexibility in determining how to make your website compliant with the ADA's general requirements of nondiscrimination and effective communication."

[1] https://www.section508.gov/blog/do-section-508-accessibility-standards-apply-to-mywebsite

Accessibility helps everyone

Population Over Age 65 by County

- Colorblind-friendly
  - Figures that print well in black and white
- Low contrast issues
  - Reading phone screen in bright sunlight
- Subtitles (CC)
  - Automatic transcripts of podcasts; captions for TV in restaurants
- Text based communication (TDD)
  - Everyone texts now
- Adapt to new types of technology
  - Mobile friendly websites, voice-activated, etc.
  - This benefits from the shift to REST APIs (separate content from rendering)

[1] https://www.ruralhealthinfo.org/toolkits/aging/1/demographics
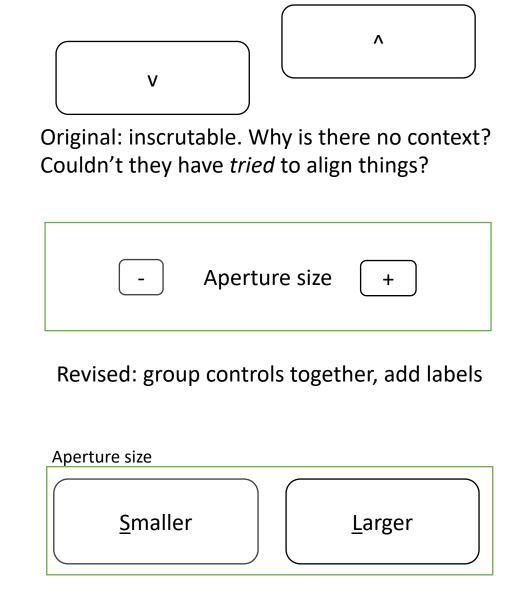
A mouse-only UI?

The challenge: instrument control software for a light-sensitive experiment. Had to turn off the screen during the critical step.

Only one person knew what the buttons did.

A first draft added human friendly labels; this took up space and made buttons very hard to click in the dark

Second draft considered the physical limits for how it would be used

Modern alternatives: physical buttons, touch interface, VUI...

v

^

Original: inscrutable. Why is there no context? Couldn't they have *tried* to align things?

- | Aperture size | +

Revised: group controls together, add labels

Aperture size

Smaller | Larger

Final: add keyboard shortcuts (no mouse required), add alert sounds if a problem occurs

One goal, many pieces

Perceivable - Information and user interface components must be presentable to users in ways they can perceive.

Operable - User interface components and navigation must be operable.

Understandable - Information and the operation of user interface must be understandable.

Robust - Content must be robust enough that it can be interpreted reliably by a wide variety of user agents, including assistive technologies.

[1] https://www.w3.org/TR/UNDERSTANDING-WCAG20/intro.html

Auditory issues: considerations

- Conference talks / lectures
  - Provide live captions/interpreters, or pre-typed notes
- Provide alternatives where audio is critical
  - "Phone number to get login code"
  - Error messages/ alerts
  - Jeopardy video clue

Physical (eg no mouse)

- Ensure that form controls can be reached without a mouse, and in the order expected
    - AJAX forms should still use form tags for grouping
    - Keyboard tab order may not match the order of elements in HTML: ensure that users can reach form elements and buttons in a way that makes sense

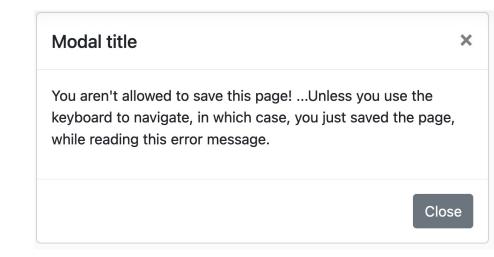- Screen readers might not understand custom widgets or HTML canvas

"...Users might recognize it visually, but there are no machine-readable semantics for an assistive technology." [2]

[1] https://dequeuniversity.com/checklists/web/custom-widgets
[2] https://developer.mozilla.org/en-US/docs/Web/Accessibility/An_overview_of_accessible_web_applications_and_widgets
[3] http://nomouse.org/

Keyboard users break the rules

- Sometimes, UI is designed to only work with "mouse click" events. Keyboard users can bypass the expected rules.
  - Unreachable elements: some buttons cannot be used without a mouse
  - Opposite problem: Keyboard "traps" (can't leave)
- Common issues: modal dialogs; form validation that only disables the button
- Pre-made widget libraries often include accessibility features built in

Modal title ×

You aren't allowed to save this page! ...Unless you use the keyboard to navigate, in which case, you just saved the page, while reading this error message.

Close

https://accessibility.18f.gov/keyboard/

Cognitive

- There are many types of problem; this comes into play when writing tools with a wide audience
  - How to ensure informed consent from someone clicking a web page?
- Common considerations:
  - Distinguishable (eg distraction-free layouts)
  - Readable (short, clear sentences)
  - Other language barriers?
- The most nebulous of the WAI standards: https://www.w3.org/WAI/cognitive/
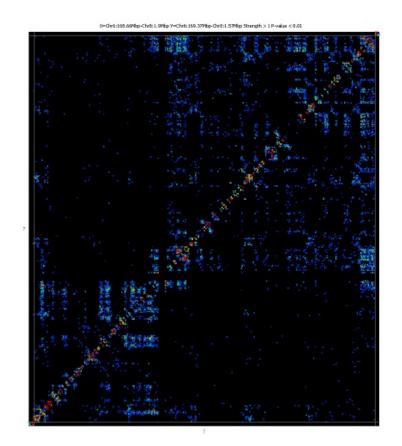- Research studies have domain-specific rules, eg informed consent

[1] https://research-compliance.umich.edu/informed-consent-guidelines

Vision: A complicated topic

- Most interfaces are visual, so this is something that technology tends to focus on

- Many possible aspects:
  - Color blindness / contrast issues
  - Screen readers
  - Features that **shouldn't** be computer readable: *CAPTCHAs*
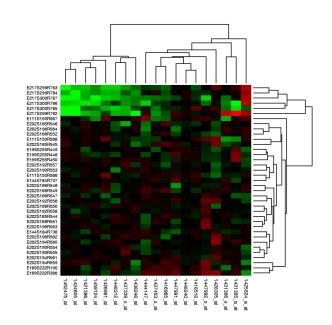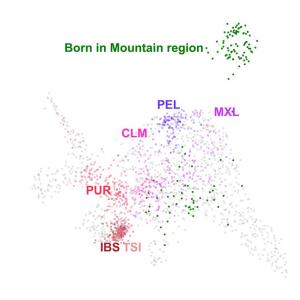
Forms of colorblindness

- Three basic categories, each with several forms [1]:
  - Red-green: ~8% of males and ~0.5% of females of N. Eur descent
  - Blue-yellow
  - Complete
- Heritable and X-linked (more common in males)
  - *"If a submitted manuscript happens to go to three male reviewers of Northern European descent, the chance that at least one will be color blind is 22 percent."* [2]
- There is a continuum: some people have *partial* loss

[1] https://www.nei.nih.gov/learn-about-eye-health/eye-conditions-and-diseases/color-blindness/types-color-blindness
[2] Wong, B. Points of view: Color blindness. *Nat Methods* **8**, 441–441 (2011). (link)

Colors Are Hard

Born in Mountain region

PEL   MXL
CLM
PUR
IBS TSI
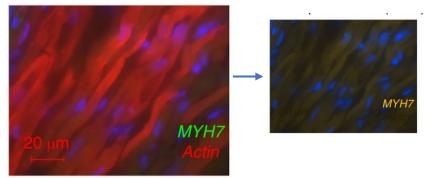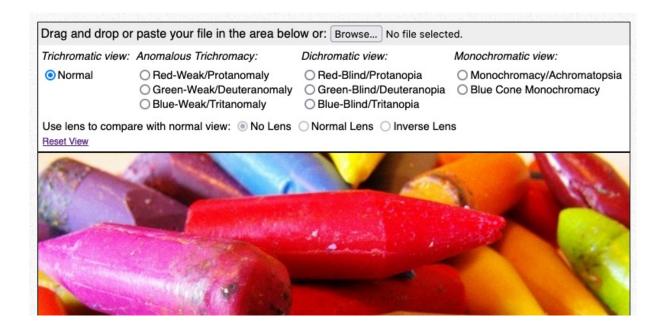
MYH7 expression in RA (HF 6w)

MYH7
Actin
20 μm

MYH7

[1] "HiC Heatmap": https://www.bioinformatics.babraham.ac.uk/projects/seqmonk/Help/3%20Visualisation/3.2%20Figures%20and%20Graphs/3.2.12%20The%20HiC%20Heatmap%20Plot.html

[2] "Gene expression heatmap": https://en.wikipedia.org/wiki/Heat_map#/media/File:Heatmap.png

[3] UMAP plots of populations structure (2020): https://journals.plos.org/plosgenetics/article?id=10.1371/journal.pgen.1008432

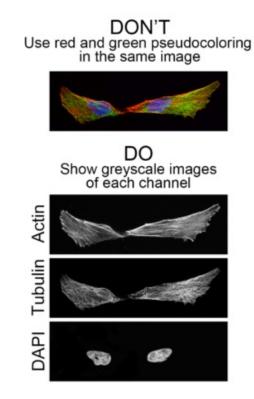[4] It's annual review season! https://www.nature.com/articles/s41588-018-0171-3/figures/4

Experiencing colorblindness

- Online "color-blind simulator" tools let you try any image
- Firefox web browser has a built-in tool for web pages

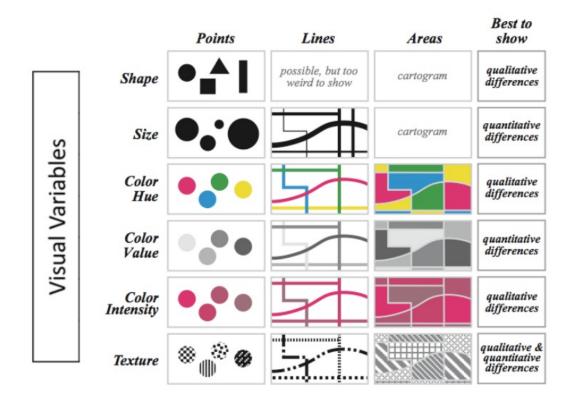# Improving colors: content matters

- Consider what you are using color to convey
  - Combined image or separate pieces of information?
- Choose a color scheme that reflects the type of data
  - Continuous
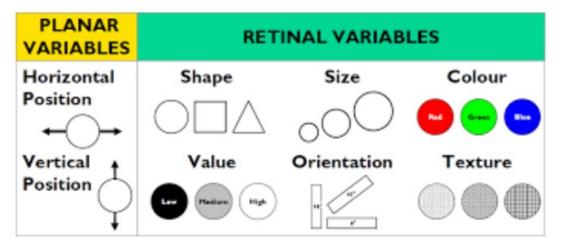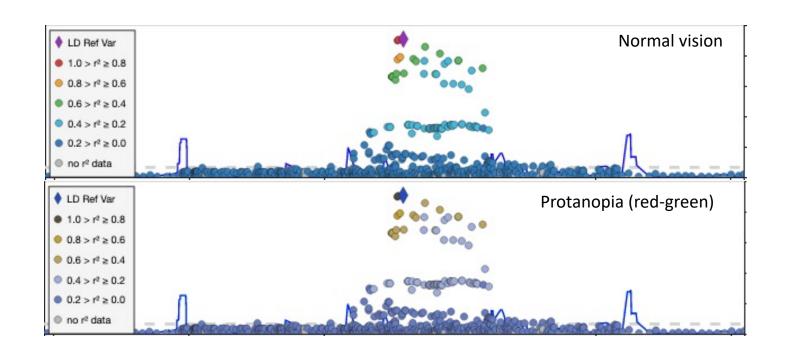 - Sequential
  - Categorical

[1] https://www.ascb.org/science-news/how-to-make-scientific-figures-accessible-to-readers-with-color-blindness/
[2] "Semiology of graphics": https://karlsluis.medium.com/before-tufte-there-was-bertin-63af71ceaa62
[3] "Grammar of Graphics": https://www.amazon.com/Grammar-Graphics-Statistics-Computing/dp/0387245448
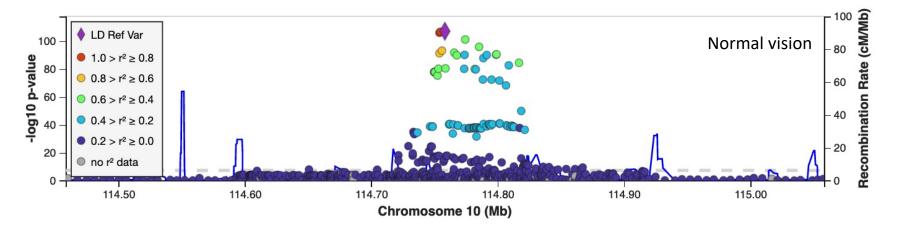
Go Beyond Color

Application to our own tools: LocusZoom.js

- Default LocusZoom color palette is ok, but not great with red-green colorblindness

- Can we find a better color palette?

New LZ.js color map: Turbo
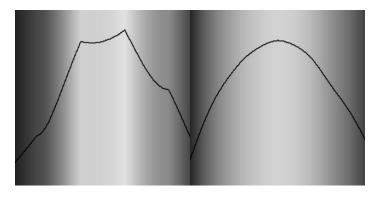

- More saturated colors
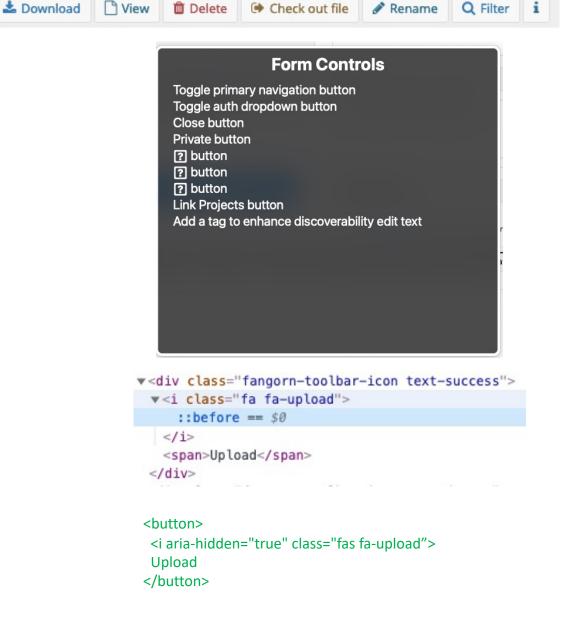- Smoother luminosity gradient than Jet

[1] https://ai.googleblog.com/2019/08/turbo-improved-rainbow-colormap-for.html

Try it out: Screen Readers

- Demonstration videos
  - https://twitter.com/Kristy_Viers/status/1287189581926981634
  - https://www.youtube.com/watch?v=_B1o6DyLCVY

- How to use macOS "VoiceOver" (built in):
  - https://dequeuniversity.com/screenreaders/voiceover-keyboard-shortcuts
  - Caveat: most users are on Windows: https://webaim.org/blog/three-things-voiceover/

- Windows 10 "Narrator":
  - https://support.microsoft.com/en-us/windows/complete-guide-to-narrator-e4397a0d-ef4f-b386-d8ae-c172f109bdb1
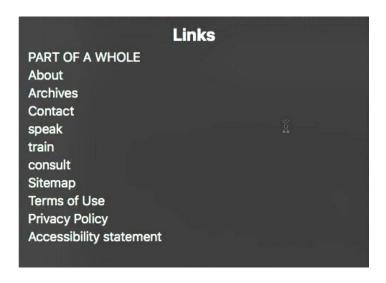
Use semantic markup!

- Using built-in HTML tags lets the browser add accessibility features automatically
  - "You can click this"
  - "Here is a label for this text"

- Code should indicate interactive elements
  - Links for navigation
  - Buttons for click/action
  - Tab order

- Icons or visual-only elements can render your controls invisible to blind people

Form Controls

Toggle primary navigation button
Toggle auth dropdown button
Close button
Private button
? button
? button
? button
Link Projects button
Add a tag to enhance discoverability edit text

```
▼<div class="fangorn-toolbar-icon text-success">
  ▼<i class="fa fa-upload">
      ::before == $0
    </i>
    <span>Upload</span>
  </div>
```

```
<button>
  <i aria-hidden="true" class="fas fa-upload">
  Upload
</button>
```

[1] https://fontawesome.com/v5.15/how-to-use/on-the-web/other-topics/accessibility
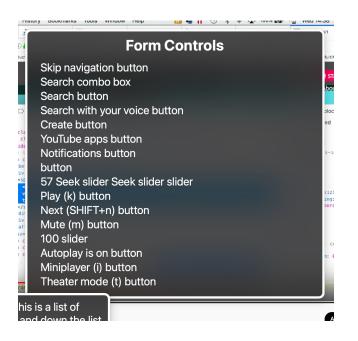
Help the user find content

- Landmarks: Use HTML tags for navigation vs section

- Hierarchical headings give a sense of page outline (H1, H2...)

- ARIA roles that indicate nested UI (tabs, flyouts...)

- Provide context-specific description
    - Link text should describe contents, not just say "click here"

- Form controls: descriptions include keyboard shortcuts or current state where applicable
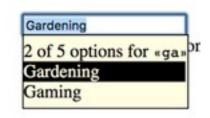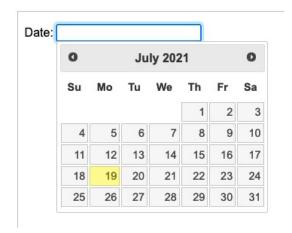

The VoiceOver "rotor": quickly find sections of the page

When semantic markup isn't possible

- Custom HTML widgets require telling the browser how to use the widget

- ARIA roles are special HTML attributes to identify how pieces interact

- The spec is... complicated. Read it as you go.

[1] https://www.accessibility-developer-guide.com/examples/widgets/autosuggest/
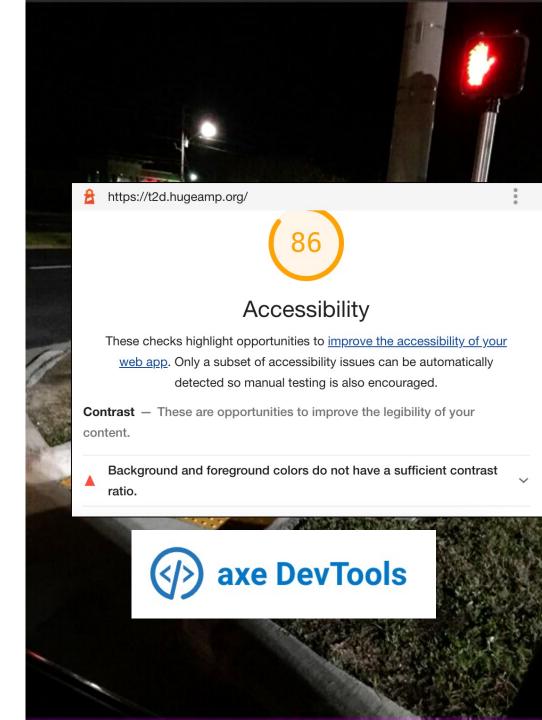
Beyond visualization

- Sonification (turning patterns into music)
  - https://www.youtube.com/watch?v=3EXvR1shVFQ
- WebPlotDigitizer: Extract datapoints by tracing a figure (*)
  - https://apps.automeris.io/wpd/
- Show the data!
  - Tabula: Extract data from tables: https://tabula.technology/
  - Supplemental files / data repositories

(*) Some authors get *really* mad when you access raw data this way, even just to check the results in that paper.

Automated checkers

- Browser based tools to check website
  - Chrome extension: https://wave.webaim.org/
  - Did well in UK test: https://tenon.io/
  - Built into Google Chrome: https://developers.google.com/web/tools/lighthouse
- Automated tools that integrate into CI
  - https://github.com/dequelabs/axe-core
  - https://www.npmjs.com/package/vue-axe
- Not a substitute for human testing. Tend to emphasize what can be easily measurable
  - UK study: best tools found ~38% of barriers [1]
  - Missing one big problem can break everything

[1] https://alphagov.github.io/accessibility-tool-audit/test-cases.html

Some good examples

- Places that have a stake in compliance
    - Federal Agencies: https://18f.gsa.gov/
    - Banks and Airlines (operate in many jurisdictions, lots of $$: lawsuit targets)
        - https://www.delta.com/
        - https://www.chase.com/
- Advocacy organizations: lead by example
    - https://globalaccessibilityawarenessday.org/#site-content

References

- "Practical Hands-on accessibility"- good intro video! https://www.youtube.com/watch?v=_B1o6DyLCVY
- Helpful communities
  - https://globalaccessibilityawarenessday.org/
  - https://www.section508.gov/create/software-websites
  - GSA Guide: https://accessibility.18f.gov/
  - The "no mouse" challenge: http://nomouse.org/
- Standards and guidelines
  - WCAG: https://www.w3.org/TR/UNDERSTANDING-WCAG20/intro.html
  - WAI-ARIA: https://www.w3.org/WAI/standards-guidelines/aria/

Sometimes, you just want data

Don't lock people into a single preferred user interface

Claims and evidence

- Papers have long incorporated claims and evidence together

- As methods become more complex, it is hard to follow up just from figures
  - Access to the raw data is needed

Galileo reported his discovery of Saturn's unusual shape as 2 *visual nouns* that compare clear and murky telescopic views. In Galileo's work *Istoria e dimostrazioni intorno alle macchie solari* (1613), words and images combine to become simply evidence rather than different modes of evidence:

ta imperfezzione dello ftrumento, ò dell'occhio del riguardan-te, perche fendo la figura di Saturno così ⟨⟩ ,come moftra-no alle perfette vifte i perfetti ftrumenti , doue manca tal perfezzione apparifce così ⟨⟩ non fi diftinguendo perfetta-mente la feparazione , e figura delle tre ftelle ; ma io che mil-le volte in diuerfi tempi con eccellente ftrumento l'hò riguar-dato, poffo afficurarla , che in effo non fi è fcorta mutazione

[1] Galileo Galilei, *Istoria e dimostrazioni intorno alle macchie solari* (Rome, 1613), 25. *Discoveries and Opinions of Galileo* (New York, 1957), p. 102, translated by Stillman Drake. Images of Saturn are integrated into the text of 3 letters by Galileo in 1610, *Le Opere di Galileo Galilei*, ed. Antonio Favaro (Florence, 1890–1909), vol. 10, 409–410, 474, 502–504. Illuminated manuscript below: the *Howard Psalter* (14th century), facsimile by Henry Shaw (London, early 1860s).

The shape of Saturn is thus ⟨⟩ as shown by perfect vision and perfect instruments, but appears thus ⟨⟩ where perfection is lacking, the shape and distinction of the three stars being imperfectly seen.[1]

Placed in the familiar typographic context, these extraordinary images become just another sentence element, with no distinction between text and image. Saturn as evidence, image, drawing, graphic, word, noun. Galileo's word/image sentence is one of the best analytical designs ever. It is an excellent precedent for integrating small, detailed images within text, similar to the graphical capital letters in early manuscripts.

Edward Tufte, Beautiful Evidence (2006)

Considerations for sharing

- FAIR Principles
  - **F**indable: Indexed by search, and persistent links
  - **A**ccessible: Respect access rules, but not just "available upon request"
  - **I**nteroperable: Integrate with tools and workflows
  - **R**eusable: Maintain provenance and vocabulary

- Resources and best practices vary by field; ask around!

Where to share: Methods and code

- Genetic data (ask Heather, Albert, Tom, etc how to use these!)
  - dbSNP
  - dbGaP
  - ...others?
- Summary results
  - EBI eQTL Catalogue
  - EBI GWAS Catalogue
- Source code/ methods
  - GitHub ("living" version of source code + provenance)
  - Zenodo (auto-generate author list and citable DOI from a GitHub repo; archives maintained by CERN)

Summary

- Accessibility has many pieces

- There is no substitute for human testing - yet

- Interactive widgets are harder to write, and MUCH harder to make accessible

- Know your requirements up front, and plan for accessibility during initial development