



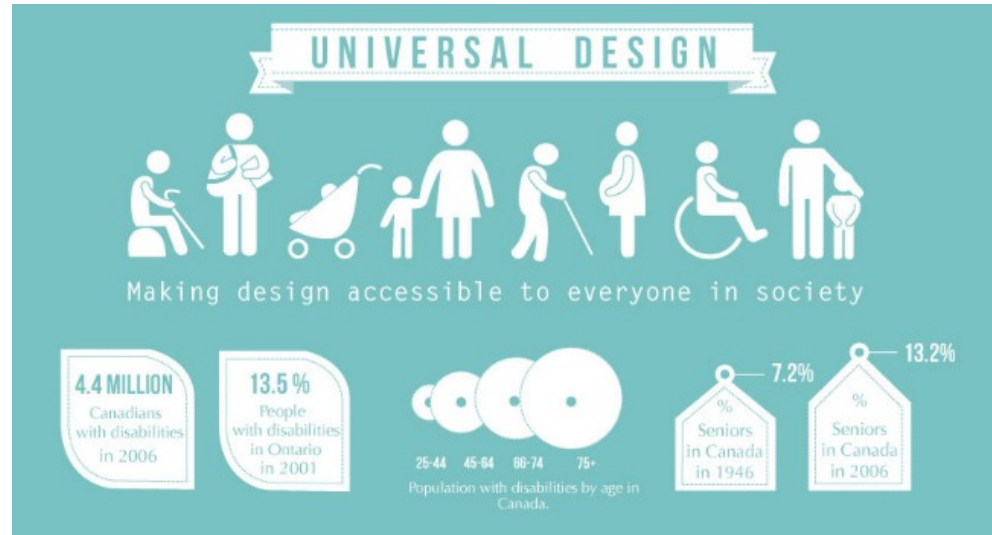
CS1320
***Creating Modern Web and
Mobile Applications***

Lecture 3:

Universal Access

What does Universal Accessibility mean?

- Letting everyone access your web site
- What does everyone include?
 - Non-Native speakers
 - Persons with physical disabilities
 - Persons with mental disabilities
 - Persons with temporary injuries
 - Everyone



Who Are Your Users?

- Do you have a particular set of users in mind?
- Are you designing for yourself?
 - Most people do
- Do you represent all your users?
 - 6-8% of males are color blind
 - 30-70% of CS students have wrist problems
 - 65-75% of people wear glasses or contacts
 - 17% of impairments are uncorrectable
 - 6% of the population
 - ~50% as people get older
- **THERE IS NO TYPICAL USER**



Common Disabilities

- **Vision Problems**
 - Blindness, low-vision, presbyopia, color blindness
- **Hearing problems**
 - Deafness, high-frequency loss
- **Movement problems**
 - Paraplegic, wrist problems, broken arm/hand, MS, ALS
- **Difficulty in reading**
 - Dyslexia, illiterate, non-native speaker



Web Site Accessibility

Making a web site accessible primarily involves:

- A. Ensuring your web site can be used effectively and efficiently with assistive devices such as screen readers
- B. Making sure that disabled users can use your web site directly
- C. Adding special HTML elements to handle various disabilities
- D. Having separate web sites for the blind and other disabled groups
- E. Testing your web site with a broad range of disabled users.

Why You Should Care

- This only affects a small subset of potential users
 - You should have enough users without these
 - It can be a lot of work adapting your app to all potential users
 - Is it worth it?
- YES: It is the right thing to do
 - But lots of companies (esp. startups) don't bother
- YES: It isn't really a small subset
- YES: Its not that hard
- YES: Makes your application better
 - It tends to make you application better in any case
 - Many of the things you do for accessibility help the overall look and feel and usability of the interface
- YES: Required legally



Americans with Disabilities Act

- Requires all businesses make **reasonable** accommodations for **all** handicaps
- It applies to web and mobile applications
 - Universities have been sued for switching to Google Apps
 - <https://www.insidehighered.com/news/2018/12/10/fifty-colleges-sued-barrage-ada-lawsuits-over-web-accessibility>
 - Companies have been sued as well
 - <http://reason.com/blog/2008/08/28/target-settles-ada-lawsuit-ove>
- Anything done for the government (or Brown)
 - Must meet ADA guidelines
 - Might mean everything done in this course



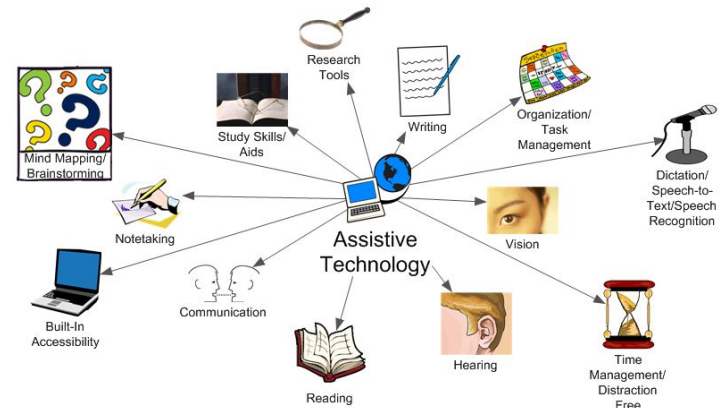
So What Do You Do

- Make your site at least minimally ADA-compliant
 - This will make your web site better as well
- Easiest to do this from the start
 - While designing the user interface
 - While designing the web application
- Much harder to retrofit later on
 - Might not be simple CSS changes
 - Might require a full redesign of your site
- Need to understand what this means
 - Especially when we get to dynamic interfaces



Assistive Technologies

- You can't be expected to do everything for everyone (too many disabilities)
 - Accessibility doesn't have to be built-in
 - **But it has to be available**
 - You should know what technology people actually use
- Assistive Technologies
 - Screen readers
 - Screen magnifiers
 - Assistive display settings
 - Alternative input devices (keyboard only)
 - Video and audio captioning



Experiences

- What accessibility feature did you try?
 - Who used a screen reader?
 - Who tried a high-contrast display?
 - Who tried high-magnification?
 - Anything else
- Could you use the web/applications this way?
 - Why or why not?

How to Make Web Sites & Mobile Apps Accessible

- **General rules**
 - HTML provides features that can enhance accessibility
 - Also features that have the opposite effect
 - You should know what helps and what hinders
 - Use what helps, avoid what hinders
- **The bulk of the work has been done for you**
 - W3C web accessibility content accessibility guidelines
 - Understanding these makes accessibility easier
 - Particular HTML tags and fields to aid accessibility

Web Accessibility

Improve the Web for Everyone

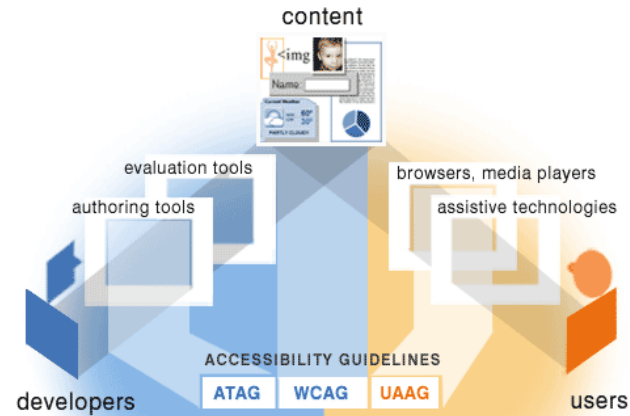
W3C Web Accessibility Initiative

- Provides simple guidelines with priorities
 - Various levels: A, AA, AAA
 - If you meet the guidelines you're doing best effort
- The guidelines contain a lot of common sense
 - Match common user interface guidelines
 - Make your web site better
 - Guidelines apply to web sites in general
 - Go beyond accessibility and address **usability**
 - Guidelines apply to mobile applications as well



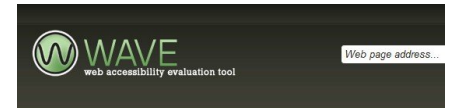
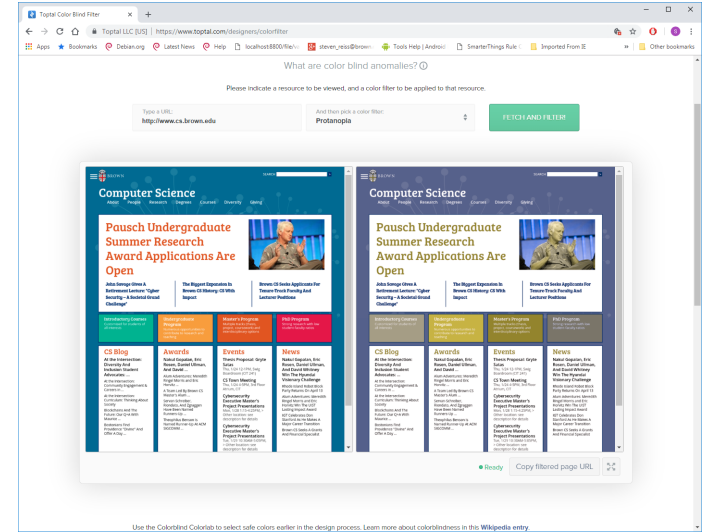
Guideline Examples

- **Understandability guidelines**
 - Make text readable and understandable (e.g. avoid idioms, funny fonts, ...)
 - Make content appear and operate in predictable ways
 - Help users avoid and correct mistakes
- **Robustness guidelines**
 - Maximize compatibility with current assistive tools
 - Maximize compatibility with future assistive tools
- **DESIGNERS:**
 - Read and understand these guidelines
 - Before designing a web site



Guideline Checking

- Several accessibility testers exist
 - <http://wave.webaim.org>
 - Also available as a browser extension
 - <http://cynthiasays.com>
 - <http://achecker.ca/checker/index.php>
 - <http://fae.cita.uiuc.edu>
 - <http://colorfilter.wickline.org> :: color blind views of your page
- Test your web site with real users
 - To ensure it is accessible
 - To ensure you can handle a wide range of users



Welcome to WAVE

WAVE is a free web accessibility evaluation tool provided by [WebAIM](http://www.webaim.org). It is used to aid you than providing a complex technical report, WAVE shows the original web page with emt that page.

Enter a web site address

Enter the URL of the web site you want to evaluate:

Complexities

- What happens with a Front-End heavy application (dynamic web pages)
 - DOM changes dynamically - Html effectively changes on the fly
 - Will the user be able to tell from a screen reader?
 - Will the user see the change if the page is highly magnified
 - This would need to be tested extensively
 - Simple validators don't necessarily help
- What about included documents
 - Word, PDF, PowerPoint, Excel
- We will keep coming back to accessibility



Internationalization

Internationalizing a web site does **not** involve

- A. Having separate web sites for each major country
- B. Localizing all text strings using an appropriate tool
- C. Eliminating icons that have text in them
- D. Avoiding culture-specific symbols
- E. Using library functions for formatting time, currency, etc.

Why Internationalization

- **Where are your users**
 - Will they always be there
 - Is your software portable
- **What are your users' backgrounds**
 - Is English their first language
- **Should you create one or multiple web sites**
 - How easy will it be to maintain
 - How many languages should you accommodate



What is Internationalization

- **Creating source information that is locale independent**
 - **Locale** : set of features defining the user's region
 - en-us, C, de, ...
 - Facilitate customization through localization
 - Much more than simple translation
- **Localization**
 - Adapting a web site to a particular locale
 - Not an attempt to be everything for everybody at once
- **Internationalization is really setting up for *Localization***



What Changes with Localization

- What do you think changes?
 - How many know other languages
 - How many have traveled to or lived in foreign countries



Language Changes

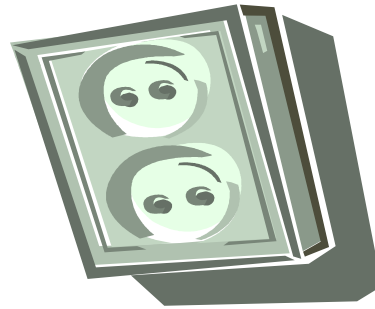
- Translation and automatic translation
- Fonts and character sets
 - Unicode versus ASCII versus UTF-8 versus UTF-16 ...
- Text direction, flow direction
- Lengths of text elements



Symbols and Design Elements



Symbols and Design Elements

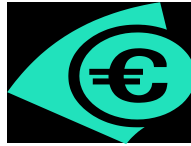


Symbols and Design Elements



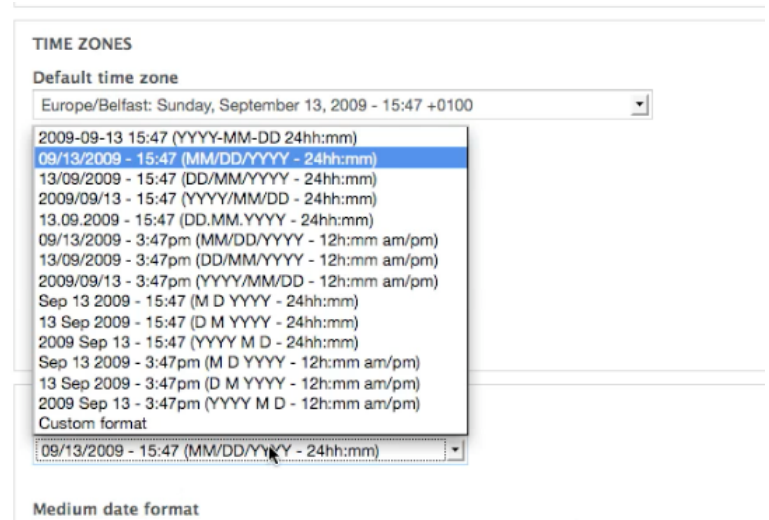
Currency

- Different currency signs and conventions



Numbers, Dates, and Times

- Numbers are represented differently
 - US/UK: 12,345.67; France: 12 345,67
 - Germany: 12.345,67; Asia: 1.2345,67
- Dates
 - Saturday is 02/01/2020 (US)
 - Saturday is 01/02/2020 (Elsewhere)
- Times
 - The meeting is at 3:30pm (US)
 - The meeting is at 15:30 (Elsewhere)
- Percentage
 - Space before percent or not; Percent before or after number



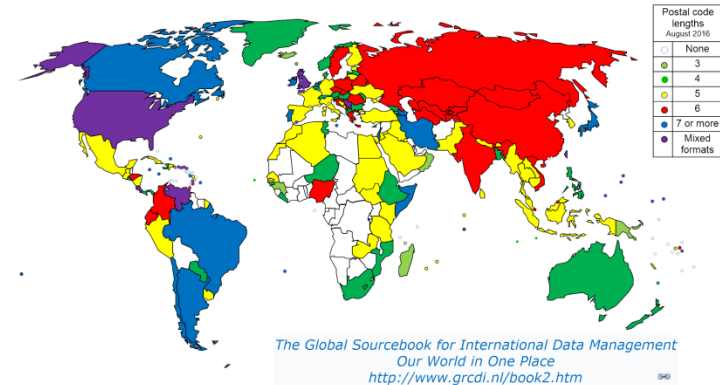
Alphabetization Order

- Unicode order is not correct (even for English)
 - Uppercase / lowercase
 - Accents can affect order
- Where do extra letters go
 - Some accented letters are actually other letters
- Special sort orders for languages
 - Spanish: a,b,c,ch,d,...,l,ll,m,...



Addresses

- **Postal codes**
 - US has 5 (or 9) digit zip codes
 - Other countries have different codes (different length, letters and digits)
- **Telephone numbers**
 - Can have more or fewer digits than US
 - Country codes
- **Names**
 - Salutations
 - Patronymic names
 - Where to display titles, degrees, etc.



Other Changes

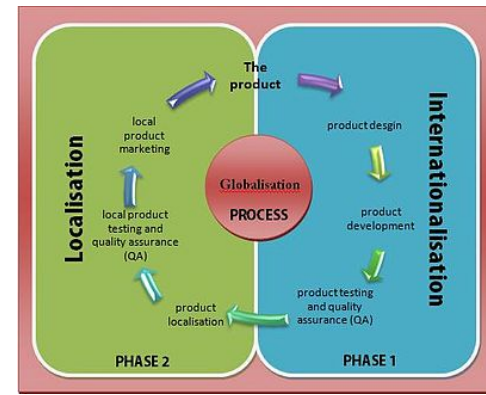
- Units of measure
 - Metric versus English
- Paper sizes
 - Letter/Legal versus A3/A4
 - Printer layout
- Calendars can differ
 - Gregorian
 - Julian
 - Chinese
 - Hebrew

- Privacy Regulations
- Shopping (shipping, customs, ...)

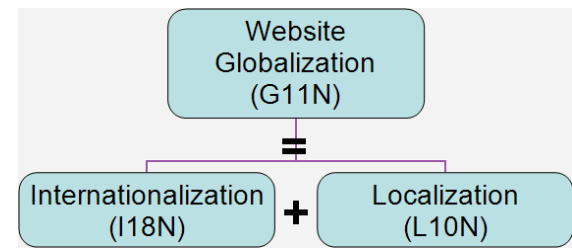
The screenshot shows a 'Localization' settings panel. At the top, there is a 'Localization' tab with a globe icon. Below it is a checkbox labeled 'Check/uncheck all (check boxes if you want to set a custom value for this shop or group shop context)'. The main area contains four rows, each with a checkbox on the left and a unit selection field on the right. The first row is 'Weight unit: lb' with a red asterisk and the subtitle 'The default weight unit for your shop (e.g. kg or lbs)'. The second row is 'Distance unit: mi' with a red asterisk and the subtitle 'The default distance unit for your shop (e.g. km or mi)'. The third row is 'Volume unit: gal' with a red asterisk and the subtitle 'The default volume unit for your shop'. The fourth row is 'Dimension unit: in' with a red asterisk and the subtitle 'The default dimension unit for your shop (e.g. cm or in)'. At the bottom right is a 'Save' button. At the bottom left is a legend: '* Required field'.

Internationalization / Localization

- Language
- Text Direction
- Fonts
- Height and width of labels
- Character set
- Sort order
- Meaning of symbols
- Meaning of colors
- Currency
- Purchase methods
- Number representations
- Units
- Date and time
- Calendars
- Abbreviations, mnemonics, ...
- Slang or jargon, idioms
- Addresses
- Telephone numbers
- Paper sizes (printing)
- Names
- Privacy requirements
- Shopping



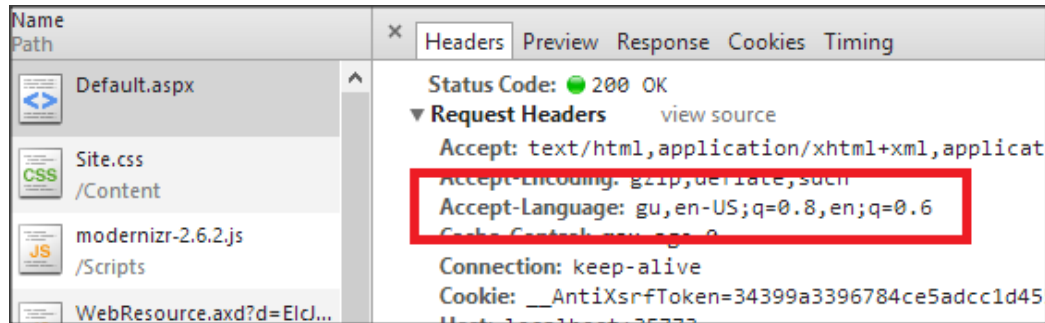
Internationalization



- How do you make your web site handle all these
 - To let your web site be customized to a particular locale (localization)
 - To make it easy to add new locales
- Difficult to do after the fact
 - Error-prone to retrofit
 - You'll miss something (icon/dynamic text/...) [or lots of somethings]
 - Easier to redo the entire web site from scratch
- Fairly simple if done consistently from the start
 - Get in the habit of doing it right
 - **DESIGNERS - work on this from the start if you will need it**

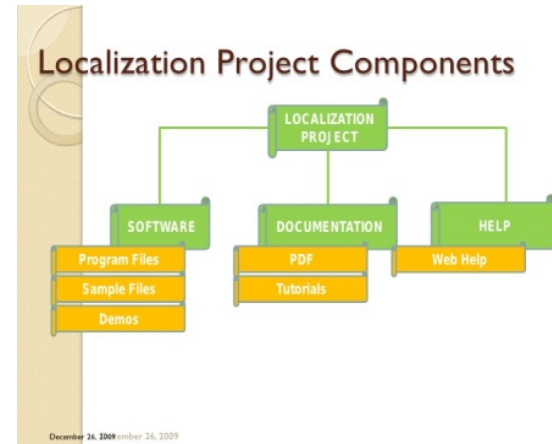
Determining User Locale

- Browser can provide Accept-Language header
- Browser provides IP address
 - Can map IP address to country
 - GeoIP extension to PHP, Node.js
- Buttons to let user set common locales



Basic Techniques for Localization

- Separate structure from presentation (sound familiar?)
 - Replaceable icons and images
 - Replaceable fonts, colors, ...
 - Separate CSS sheets for different locales
- Use library/browser support for text
 - Different character sets
 - Different text directions
- Use localization libraries and functions
 - Numbers, dates, currencies
 - Sorting



Avoiding Internalization Problems

- Avoid text embedded in graphics
 - Use text on top of a structured background
 - Good idea for accessibility as well
- Avoid culture-dependent symbols
 - Use envelopes rather than mailboxes
 - Be wary of icons with cultural meanings (stars, crosses, ...)
 - Choose icons carefully or allow them to be localized
- Internationalize your database
 - Store Unicode, not simple ASCII
 - Use locale-specific sort order (system sort functions)



Handling Static Text

- Externalize all text
 - Button names, navigation terms, error messages ...
 - There should be no actual text in the code or html
 - Front end and back end
- Use a "resource file"
 - Use a different resource file for different locales
 - Can be a directory hierarchy
 - Resource file access in the program
 - Can be a simple assignment of strings to variables
 - Can be a database, Can be an array with known indices
 - Use include facilities (JavaScript, PHP, Node.JS ...)
- Problems: truncation, spacing, ...
 - Use CSS style sheets to manage localization



Packages for Internationalization

- Date and Time formatting libraries
 - Built into latest JavaScript
- Number formatting & Sorting libraries
 - Built into latest JavaScript
- General internationalization libraries
 - gettext (GNU Project)
 - Available in PHP, Django, Ruby, Java, Node.js ...
 - Uses a directory structure of resource files
 - i18n library for node.js

GNU gettext: 1.1 Preparing Program Sources

```
/* Internationalized Jabberwocky program */
#include <libintl.h>
#include <locale.h>
#include <stdio.h>
#include <stdlib.h>

int main(int argc, char *argv[]){
    ...
    setlocale(LC_ALL, "");
    bindtextdomain("jabberwocky", "/usr/share/locale");
    textdomain("jabberwocky");
    ...
    printf(gettext("'Twas brillig, and the slithy toves\n"));
    printf(gettext("Did gyre and gimble in the wabe:\n"));
    ...
    exit(0);
}
```



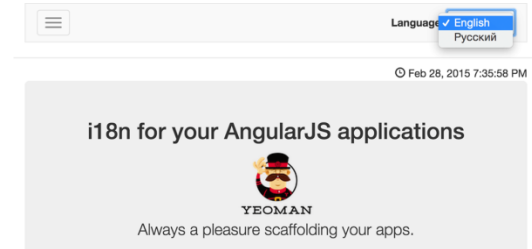
Handling Dynamic Text

- What are the problems?

- `The \${what} is currently unavailable`
 - \$what is one of "server", "connection", ...
- `There are \${n} connections`
 - This can be handled by gettext

- How might you handle this

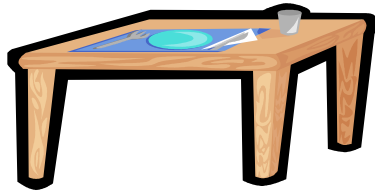
- Use a resource file as with static text
 - Store text in a database or an array of strings
 - Complete messages
- Generate content through a predefined translation function
 - Libraries for this purpose exist



Next Time

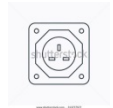
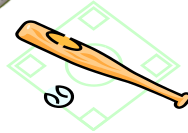
- LAB on HTML/CSS
 - There is (long) prelab to be done in preparation for the lab
 - Bring laptop to class if possible
 - Lab will be collaborative

Symbols and Design Elements



Symbols and Design Elements

- Symbols for the same different things in different cultures



Other Items

- Abbreviations, mnemonics, acronyms
- Slang or jargon, idioms
 - Motherhood and apple pie

Screen Readers

- Change the visual display into audio output
 - Scan a window and read things in the order they appear
 - Some take HTML structure into
 - Useful for blind, dyslexic, illiterate
- Braille displays
 - Provide output as Braille rather than audio
 - Useful if both blind and deaf
 - Might be faster than audio alone
- Effects of these
 - Web page is reduced entirely to text
 - Simple images are meaningless
 - Browsing is a time-based experience
 - Navigation bar at top will be read for every page

Screen Magnifiers

- **Simple solutions**
 - Increasing the font size in the browser
 - Does this work?
 - Web site might not support this (fixed images, fonts)
 - Zoom the browser
 - Decreasing screen resolution of magnifying screen
- **Large scale magnification (400+%)**
 - Might cause loss of context
 - Might make the page difficult to use
 - Especially if there is a lot of blank space

Alternative Input Devices

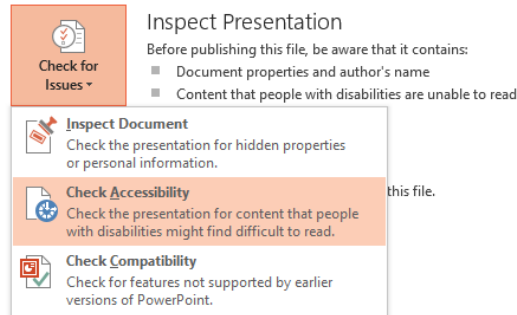
- **Simple alternatives**
 - Sticky keys, slow keys
- **Uses**
 - Repetitive stress injuries are common
 - Blind/low vision cannot use the mouse
 - Some can't use keyboard, but have a mouse equivalent
 - Severe arthritis, MS, ALS, ...
 - Keyboards are difficult to use on phone/tablet
- **Navigate with only the keyboard**
 - Keyboard as the mouse
 - Other devices simulate keyboard input
 - Tab sequences should be logical and valid
 - How do you follow links without a mouse

Guideline Categories

- **Perceivable guidelines**
 - Provide text-alternatives for non-text content
 - Provide captions and other alternatives for multimedia
 - Create content that can be presented in different ways
 - Usable by assistive technologies
- **Operable guidelines**
 - Make all functionality available from keyboard (mouse)
 - Give users enough time to read and use content
 - Do not use content that cause seizures
 - Help user navigate and find content

Checking Accessibility

- Use existing tools for a first approximation
- There is no substitute for using real people



Inspect Presentation
Before publishing this file, be aware that it contains:

- Document properties and author's name
- Content that people with disabilities are unable to read

Check for Issues ▾

- Inspect Document**
Check the presentation for hidden properties or personal information.
- Check Accessibility**
Check the presentation for content that people with disabilities might find difficult to read.
- Check Compatibility**
Check for features not supported by earlier versions of PowerPoint.



gettext (I18N) Usage

- Take you source files
 - Replace all translatable output with `gettext(...)`
 - Can use `_(...)`
- Run `xgettext` on the file
 - This yields a file of all the messages in `messages.po`
- Create translated versions of those messages
 - Automatically or manually in another file
- Set up a hierarchy of `messages.po` files
 - Organized by locale name
- `gettext` in `php/node` will read from the right file