CHARLES F. MUNAT

WEB ACCESSIBILITY

Tenei ko te aha te reira rite ki te he "kehe-abled" tangata i roto i te ao o te "poto taea ngā tinana" tangata. Kihai i nui te ngahau, he? Ite i te iti puputuu? Weriweri? Mahue ki waho? He āwangawanga e pā ana ki te aha ai koe e ngaro?

Kia waiho koe. Na kia koutou whakaaro e pā ana nei o *koutou kiritaki pea e poauau me te hano koutou maukati i*.

This is what it's like to be a "differently-abled" person in a world of "temporarily able-bodied" persons. Not much fun, is it? Feel a little confused? Annoyed? Left out? Concerned about what you might be missing?

You should be. And you should be concerned about *which of your potential customers you are foolishly and unnecessarily locking out*.

ACCESSIBILITY

- Accessibility is the degree to which a product, device, service, or environment *is available to as many people as possible*.

UNIVERSAL DESIGN

- Universal design (often inclusive design) refers to broad-spectrum ideas meant to produce buildings, products and environments that are *inherently accessible to older people, people without disabilities, and people with disabilities*.

- Also called accessible design, barrier-free design, transgenerational or intergenerational design.

SOME ESSENTIAL PRINCIPLES

- Equitable use
- Flexibility in use
- Simple and intuitive
- Perceptible information
- Tolerance for error
- Low physical effort
- Size and space for approach and use

WEB ACCESSIBILITY

- Web accessibility refers to the inclusive practice of making websites *usable by people of all abilities and disabilities*. When sites are *correctly designed, developed and edited,* all users can have equal access to information and functionality.

TYPES OF "DIS"ABILITY

- Visual
- Motor/Mobility
- Auditory
- Seizures
- Cognitive/Intellectual

VISUAL

- Visual impairments including blindness, various common types of low vision and poor eyesight, various types of color blindness.

THE KEY WEB ACCESSIBILITY PRINCIPLES FOR USERS WHO ARE BLIND ARE:

- POUR:
  - Perceivable
  - Operable
  - Understandable
  - Robust

PERCEIVABLE

- Because blind users cannot perceive (see) visual information such as graphics, layout, or color-based cues.

OPERABLE

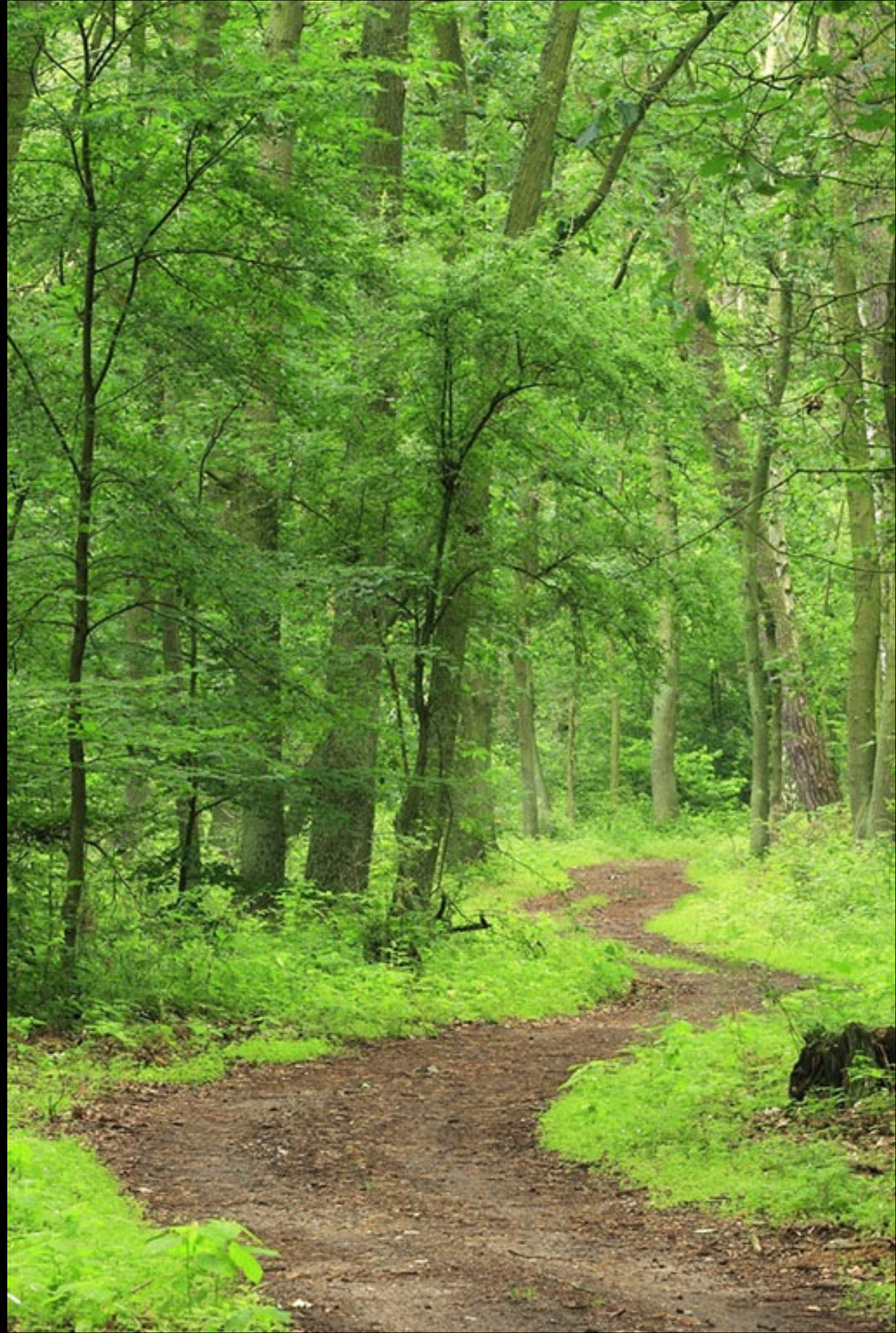
- Because blind users usually depend on a keyboard to operate (navigate) web content functionality, rather than a mouse.

UNDERSTANDABLE

- Because blind users cannot understand content that is presented in an *illogical linear order*, or which contains extraneous text *not meant to be read word for word* or character by character (such as long web addresses).

ROBUST

- Because the assistive technologies used by the blind are not always capable of accessing a broad range of technologies, especially if those technologies are new.



TYPES OF COLOR-BLINDNESS

- Protanopia and protanomaly (red deficiencies)
- Deuteranopia and deuteranomaly (green deficiencies)
- Tritanopia (blue deficiencies)
- Rod monochromacy or achromacy (no color)

DESIGNING FOR COLOR-BLINDNESS

- The key principle of web accessibility for users with color-blindness is *PERCEIVABLE* because they cannot perceive (see) the difference between certain color combinations.

- **Make sure that colors are not your only method of conveying important information.**

MOTOR/MOBILITY

- *Difficulty or inability to use the hands*, including tremors, muscle slowness, loss of fine muscle control, etc., due to conditions such as Parkinson's Disease, muscular dystrophy, cerebral palsy, and stroke.

UNABLE TO USE THE MOUSE

- Make sure that all functions are available from the keyboard (try tabbing from link to link).

UNABLE TO CONTROL THE MOUSE OR THE KEYBOARD WELL

- Make sure that your pages are error-tolerant (e.g. ask for confirmation, as in, "*Are you sure you want to delete this file?*")
- Do not create *small links or moving links*

MAY BE USING VOICE-ACTIVATED SOFTWARE

- Voice-activated software can replicate mouse movement, but not as efficiently as it can replicate keyboard functionality, so *make sure that all functions are available from the keyboard*.

MAY BECOME FATIGUED WHEN USING "PUFF-AND-SIP" OR SIMILAR ADAPTIVE TECHNOLOGIES

- Provide a method for *skipping over long lists of links* or other lengthy content

AUDITORY

- Deafness or hearing impairments, including individuals who are hard of hearing.
- The techniques for providing accessibility for users with disabilities are very straightforward - *provide captions and transcripts* for multimedia content (meaning video content that also has audio) and provide transcripts for audio-only content.

SEIZURES

- Photoepileptic seizures caused by visual strobe or flashing effects
- Because of the potentially serious nature of seizures, developers should be extra careful to avoid any graphics, animations, movies, or other objects *which have strobing, flickering, or flashing effects*.

COGNITIVE/INTELLECTUAL

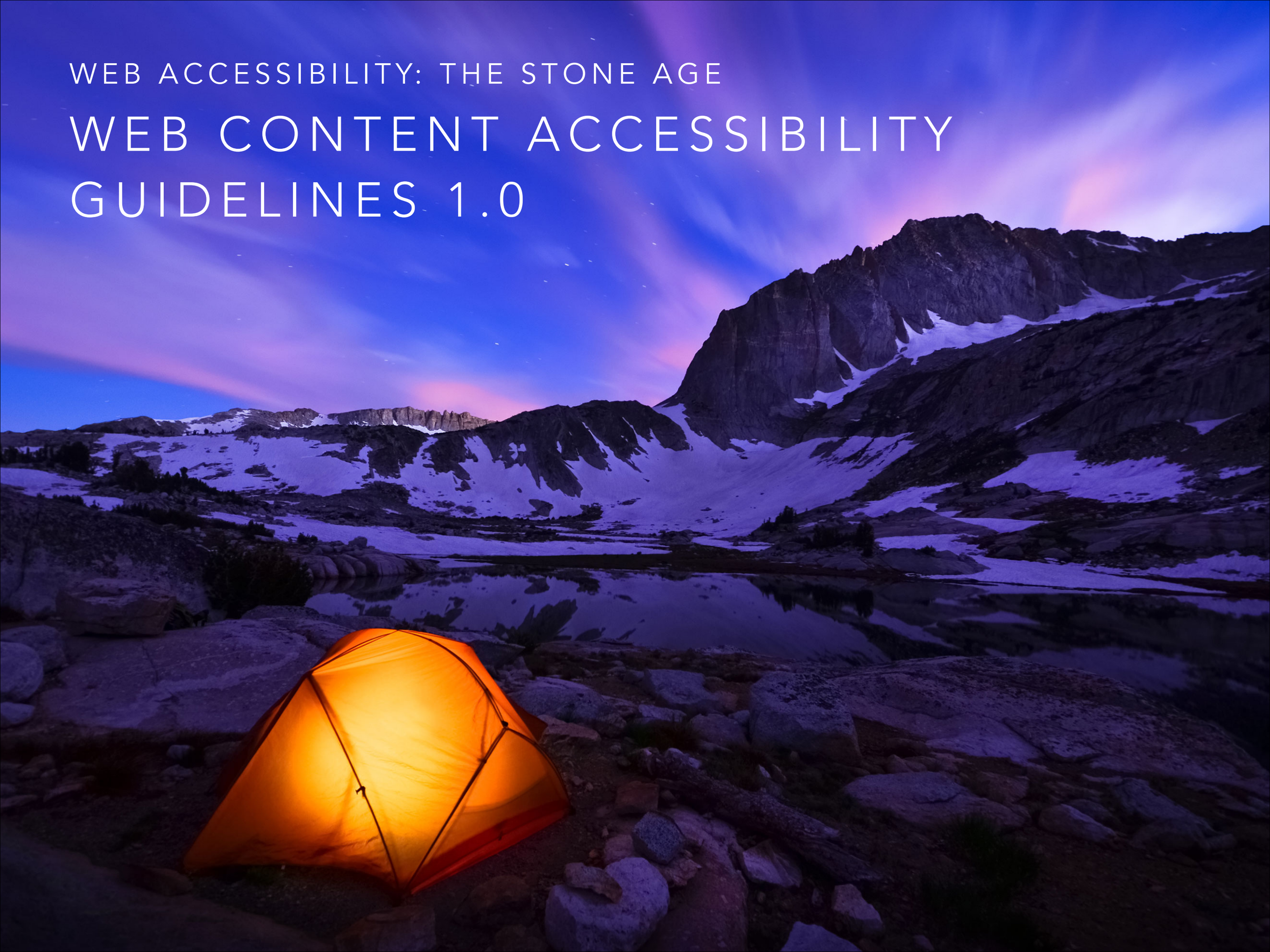
- Memory
- Problem-solving
- Attention
- Reading, linguistic, and verbal comprehension
- Math comprehension
- Visual comprehension

NON-LITERAL TEXT

- Non-literal text, such as sarcasm, satire, parody, allegory, metaphor, slang, and colloquialisms, can be a problem for some readers. In some cases, *readers will not realize that the words are not meant to be understood literally*

WEB ACCESSIBILITY: THE STONE AGE

WEB CONTENT ACCESSIBILITY GUIDELINES 1.0



"The power of the Web is in its universality. Access by everyone *regardless of disability* is an essential aspect."

– TIM BERNERS-LEE

WCAG 1.0 EXAMPLES

- *Ensure graceful transformation:* pages that transform gracefully remain accessible despite physical, sensory, and cognitive disabilities, work constraints, and technological barriers

- *Make content understandable and navigable:* this includes not only making the language clear and simple, but also providing understandable mechanisms for navigating within and between pages

Provide equivalent alternatives to auditory and visual content

GUIDELINE 1

Guideline 3: Use markup and style sheets and do so properly

Guideline 6: Ensure that pages featuring new technologies transform gracefully

Guideline 9: Design for device-independence

Guideline 13: Provide clear navigation mechanisms

A NEW APPROACH

WEB CONTENT ACCESSIBILITY GUIDELINES (WCAG) 2.0

WCAG 2.0 vs WCAG 1.0

- **WCAG 2.0** applies more broadly to different types of Web technologies and to more advanced technologies.
- **WCAG 2.0** requirements are more precisely testable with automated testing and human evaluation.
- **WCAG 2.0** was developed in coordination with international efforts to harmonize on a single standard for Web content.

THE FUTURE

ACCESSIBLE RICH INTERNET APPLICATIONS (ARIA)

BUILDING ACCESSIBLE APPLICATIONS WITH WAI-ARIA

- Each element or widget has *correct and complete semantics that fully describe its behavior* (using element names or roles)

- The *relationships* between elements and groups are defined

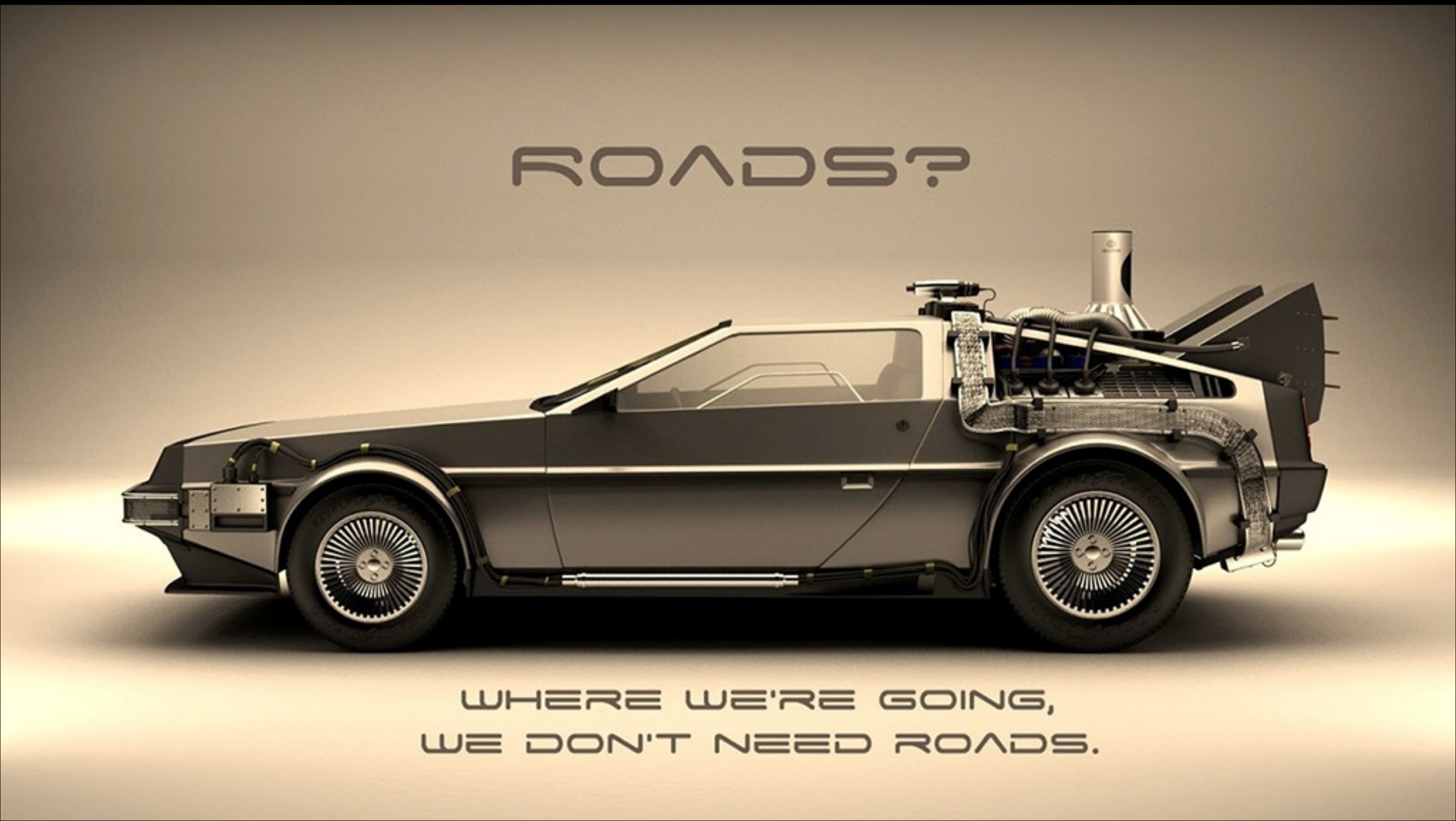
- States, properties, and container relationships are valid for each element's behavior and are accessible via the DOM and the platform accessibility API

BUILDING ACCESSIBLE APPLICATIONS WITH WAI-ARIA (CONTINUED)

- *Keyboard focus should be maintained* for the duration of the user's interaction with the application

- All interactive components should be *keyboard operable*

RECOMMENDED STEPS



USE NATIVE MARKUP WHEN POSSIBLE

- *Use the semantic elements that are defined in the host markup language.* For example, with HTML or XHTML, it is better to use the native checkbox than to use a div element with role="checkbox" as these should already be accessible through your browser.

APPLY THE APPROPRIATE ROLES

- *Set roles* to make sure elements behave predictably and correctly describe the behavior of each element within the application, unless element behaviors are fully described by the native markup language.

PRESERVE SEMANTIC STRUCTURE

- Structural information is critical to providing context to persons with disabilities. *Preserve DOM hierarchy* within structural elements and widgets.
    - Form *logical groups* within user interface widgets
    - Identify large perceivable regions and apply a *landmark* role to those areas
    - Landmark: region of the page intended as a navigational landmark.

BUILD RELATIONSHIPS

- *Look for relationships between elements*, and mark them using the most appropriate property or attribute.

- Examples:

  - aria-labelledby: Identifies the element (or elements) that labels the current element.

  - aria-posinset: Defines an element's number or position in the current set of listitems or treeitems.

SET STATES AND PROPERTIES IN RESPONSE TO EVENTS

- Once the role for an element has been set, *change states and properties as appropriate during the element's life cycle*, usually in response to user input events.

- Example: aria-checked (state) where state is true or false depending on whether the control is checked.

SUPPORT FULL, USABLE KEYBOARD NAVIGATION

- Rich internet applications behave more like desktop applications where the user tabs to significant widgets and uses the arrow keys to navigate within a complex widget, such as a menu or spreadsheet.

- *Tabbing in the document should follow a logical navigation structure.*

- When using these features, it is important as always to ensure keyboard navigation is *logical*.

SYNCHRONIZE THE VISUAL INTERFACE WITH THE ACCESSIBLE INTERFACE

- This will allow the *state* of your UI to be *perceivable* to the user as well as assistive technologies.
- For example, the author should use the appropriate WAI-ARIA attribute on a form element that is visually styled to appear required (aria-required) or a gridcell that is visually styled to appear selected (aria-selected (state)).

DOCUMENT OUTLINES

- The document outlining algorithm is a *mechanism for producing outline summaries of Web pages* based on how they are marked up.

- Every Web page has an outline, and checking it is easy .

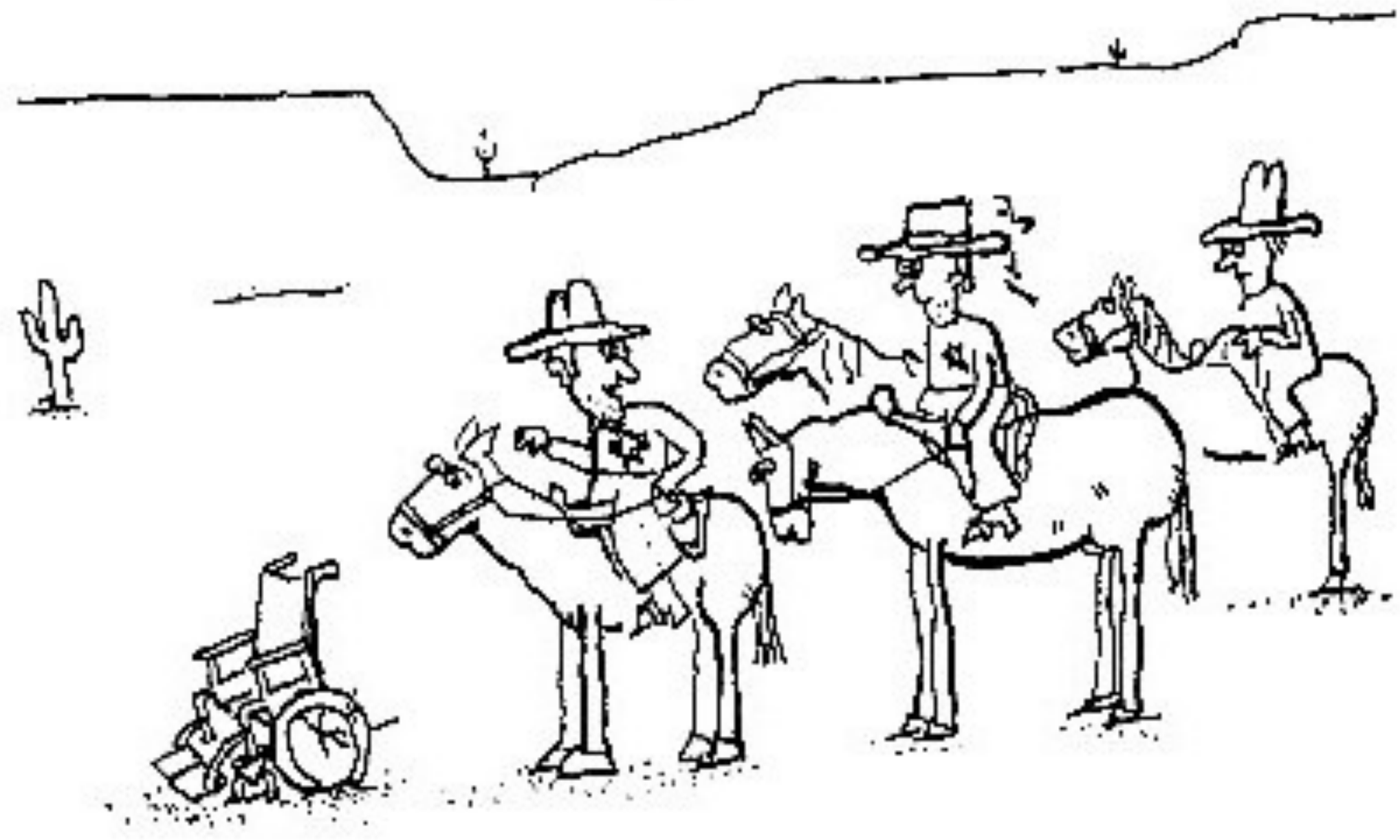
- If you remember the basics—that section, article, aside and nav create sub-sections on Web pages—then you are 90% of the way there.

CHROME EXTENSIONS

- Web Developer
- Accessibility Developer Tools
- ColorPick Eyedropper
- HTML5 Outliner

https://github.com/chasm/webaccess

LINKS TO ACCESSIBILITY RESOURCES



"Don't worry, he won't get far on foot!"

~~GO AWAY NOW~~

DON'T GO AWAY YET