



# **Aligning Requirements for WCAG Success Criteria with Organizational Roles to Deliver Accessible Content**

# Presenter



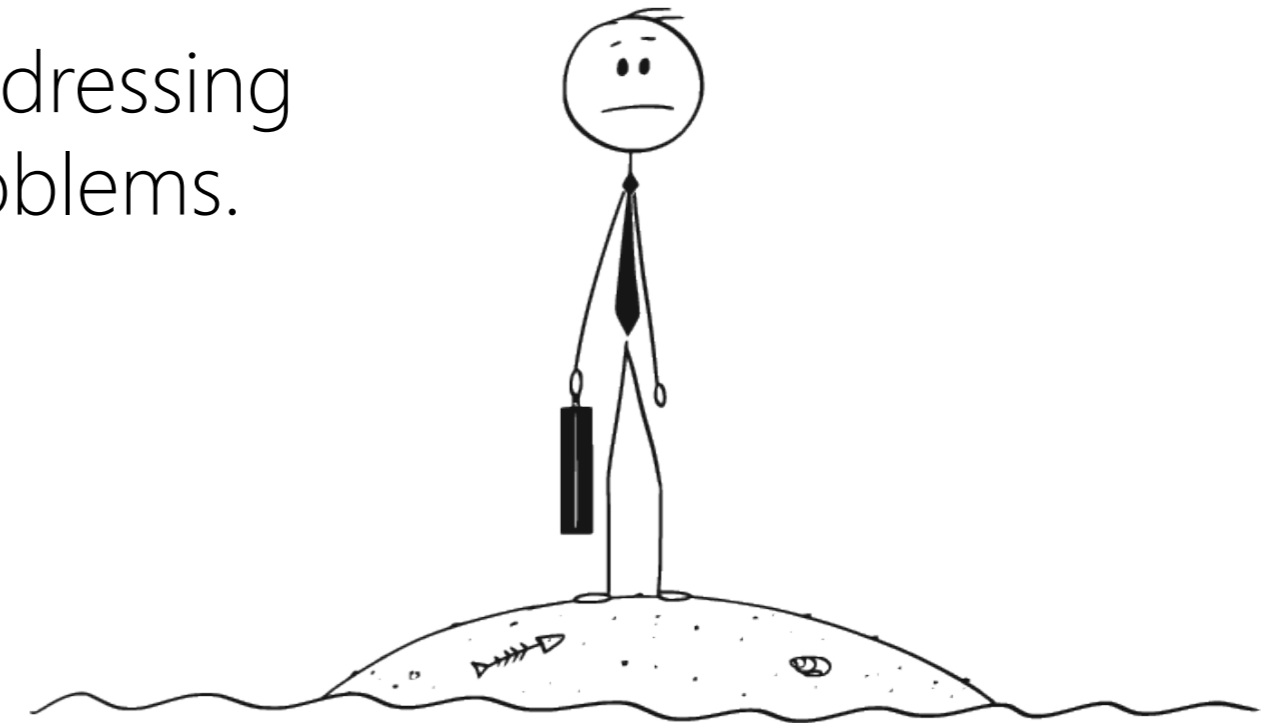
## Jeff Singleton

**Vice President, ADAAlign for Web**  
**Co-Founder of Converge Accessibility**  
**Co-Creator of WebAlign**

- Over two decades of evaluating and supporting information and communication technology
- In depth knowledge of accessibility guidelines, standards and assistive technologies
- Lectures on accessibility topics at National and International Disability and Technology Conferences

# Overview

- W3C's Web Content Accessibility Guidelines (WCAG) can be vague and complex.
- WCAG is primarily developer focused.
- Benefits of addressing those two problems.



## Two Problems

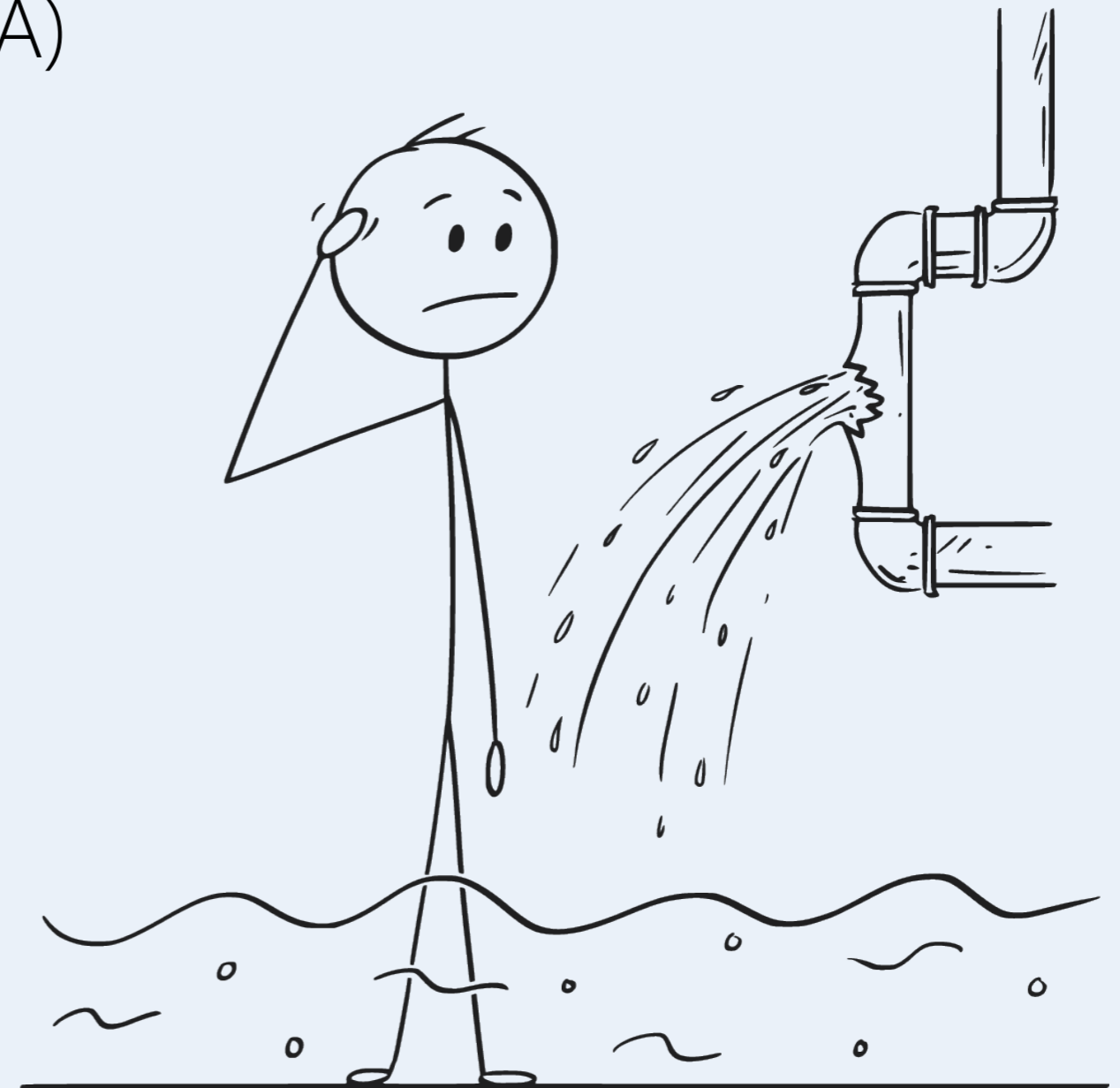
- W3C's Web Content Accessibility Guidelines (WCAG) can be vague and complex.
- WCAG is primarily developer focused.

# WCAG Can be Vague and Complex

Success Criterion 1.3.1 Info and Relationships (Level A)

*"Information, structure, and relationships conveyed through presentation can be programmatically determined or are available in text."*

- Header Structure
- HTML Regions and ARIA Landmarks
- Data Tables vs Layout Tables
- Identifying Header and Data Cells in a Data Table
- Grouping Related Elements
- Programmatically Labeled Forms
- Using Markup Consistent with Intended Purposes



# WCAG is Primarily Developer Focused

## Success Criterion 1.3.1 Info and Relationships (Level A)

### *Sufficient Techniques*

- H51: Using table markup to present tabular information
- H44: Using label elements to associate text labels with form controls
- H85: Using OPTGROUP to group OPTION elements inside a SELECT
- H48: Using ol, ul and dl for lists or groups of links
- H42: Using h1-h6 to identify headings
- ARIA11: Using ARIA landmarks to identify regions of a page
- ARIA12: Using role=heading to identify headings
- ARIA16: Using aria-labelledby to provide a name for user interface controls
- ARIA20: Using the region role to identify a region of the page

Sufficient Techniques - 35

Advisory Techniques - 5

Failure Techniques - 11

# Addressing the Problem

- Clarify Exact Requirements
- Break Requirements into Core Concepts
- Document Each Concept
- Adapt the Concept to Role-Based Guides

# Step 1: Clarify Exact Requirements

## Success Criterion 1.3.1 Info and Relationships (Level A)

*"Information, structure, and relationships conveyed through presentation can be programmatically determined or are available in text."*

- Sufficient Techniques - 35
- Advisory Techniques - 5
- Failure Techniques - 11

## What is required for this criterion?

1. Header Structure
2. HTML Regions and ARIA Landmarks
3. Data Tables vs Layout Tables
4. Identifying Header and Data Cells in a Data Table
5. Grouping Related Elements
6. Programmatically Labeled Forms
7. Using Markup Consistent with Intended Purposes




## Step 2: Break Requirements into Core Concepts (Multiple Concepts)

Success Criterion 1.3.1 Info and Relationships  
(Level A)

*"Information, structure, and relationships conveyed through presentation can be programmatically determined or are available in text."*

Core Concept 1.3.1.1 Header Structure

*"Structure in headers follow a proper hierarchy of header tags."*

1. Header Structure
  2. HTML Regions and ARIA Landmarks
  3. Data Tables vs Layout Tables
  4. Identifying Header and Data Cells in a Data Table
  5. Grouping Related Elements
  6. Programmatically Labeled Forms
  7. Using Markup Consistent with Intended Purposes
- 

## Step 2: Break Requirements into Core Concepts (Single Concept)

### Success Criterion 2.4.1 Bypass Blocks (Level A)

*"A mechanism is available to bypass blocks of content that are repeated on multiple Web pages."*

### Success Criterion 2.4.3 Focus Order (Level A)

*"If a Web page can be navigated sequentially and the navigation sequences affect meaning or operation, focusable components receive focus in an order that preserves meaning and operability."*

### Core Concept 2.4.1.1 Bypass Blocks

*"A mechanism is provided to bypass blocks of content that are repeated on multiple web pages (preferably using a "skip to main content" link)."*

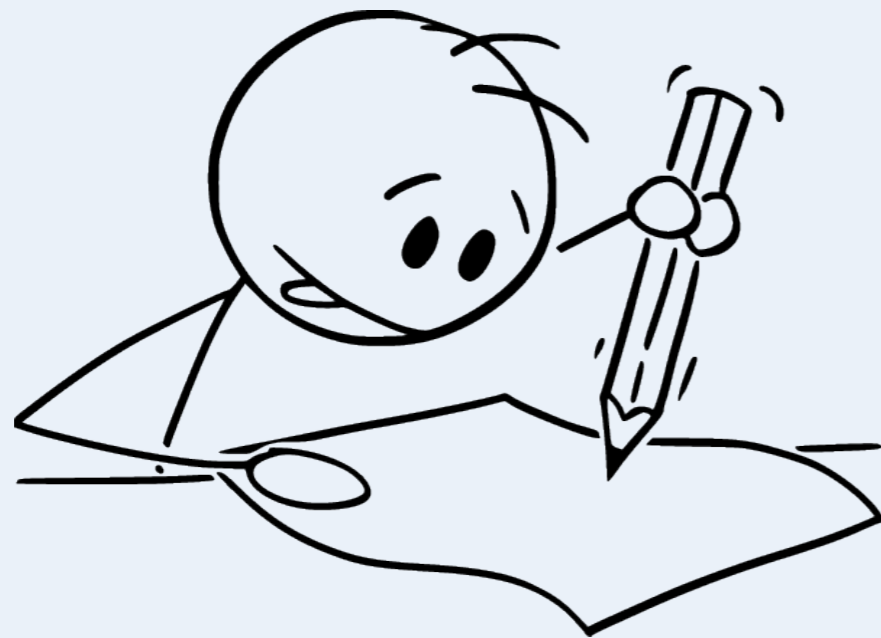
### Core Concept 2.4.3.1 Focus Order

*"Elements capable of receiving focus should receive focus in an order that is logical and preserves meaning and operability."*

## Step 3: Document Each Core Concept

### Core Concept 2.4.3.1 Focus Order

*"Elements capable of receiving focus should receive focus in an order that is logical and preserves meaning and operability."*



### Potential Summary for Core Concept 2.4.3.1 Focus Order

*Ensuring the order in which page content receives focus will help the visual and assistive technology user understand and interact successfully with the web page. If there is more than one order that preserves the meaning and operability of the page content, only one would need to be provided."*

## Step 4: Adapt Core Concepts to Role-Based Guides

### Role Specific Guidance for Core Concept 1.3.1.1 Header Structure

*“Structure in headers follow a proper hierarchy of header tags.”*

#### Pre-Production

“When headings are used in web content, make sure to use header tags. Furthermore, follow a consistent and proper hierarchy starting either with level 1 (only one level 1 heading should exist) or level 2.”

#### Production

“Ensure that the Document Object Model uses a consistent hierarchy of <H1>, <H2>, ... , <H6> tags. If necessary, mix and match aria-level attributes (e.g., role=“heading” aria-level=“7”) with <Hn> tags, provided that these levels respect a proper and consistent hierarchy.”

#### QA/Testing

“Ensure heading levels follow a consistent hierarchy beginning with an <H1> or <H2> tag, via code inspection or using a scanning tool or browser extension. If a mix of <Hn> tags and aria-level attributes are used, ensure that all such headings respect a common, consistent, and proper hierarchy.”

## Benefits

- Unified, focused, on-demand resource
- Avoid lengthy accessibility trainings
- Accounts for all accessibility guidelines/standards
- Supports DEI Initiatives

## Scanners and Overlays

Why don't we just use an automated accessibility scanner or accessibility overlay?

- Not all accessibility scanners are created equal
- Scanners do not catch everything
- Overlays have their own set of issues
- Does not empower everyone to do their part
- Not a solid foundation for an accessibility policy
- Quick Fix Solutions vs Doing the Work



## In Review

- W3C's Web Content Accessibility Guidelines (WCAG) can be vague and complex
- Extend WCAG into a simpler, clearer role-based approach to digital accessibility
- Solid foundation for web accessibility policies and DEI

**Too Much Effort. It Will Never Fly at My Organization!**





## Role-based Resources



**WebAlign**



**Align**

**Thank you!**

Jeff Singleton

[jsingleton@abledocs.com](mailto:jsingleton@abledocs.com)

+1 (425) 306-3123

