

Tableau Dashboard Accessibility Guide

Oregon Health Authority

Version 1.1 published December 2022



Contents

- Introduction 1**
 - Accessibility 1
 - Web content accessibility guidelines (WCAG)..... 1
- Remediation..... 2**
 - 1 – Descriptive links..... 2
 - 2 – Descriptive title 3
 - 3 – Text requirements..... 3
 - 4 – Descriptive headings with key takeaways 6
 - 5 – Narrative uses plain language principles 7
 - 6 – Title, caption, and data table for each visualization 7
 - 7 – Sheet tooltips turned off or non-essential 8
 - 8 – Alt text for images..... 9
 - 9 – Tooltip text for navigation buttons..... 10
 - 10 – Transcript, captions for embedded YouTube videos 11
 - 11 – Descriptions for filter, parameter, set, legend titles..... 11
 - 12 – Instructions use a combination of shape, color, size, direction 12
 - 13 – Definitions of unusual words, abbreviations [optional] 12
 - 14 – Color paired with labeling or shapes..... 12
 - 15 – Text contrast ratio of at least 4.5:1, or 3:1 if Large (18pt or bolded 14pt) ... 15
 - 16 – Non-text contrast ratio of at least 3:1 [Optional]..... 16
 - 17 – Filter or aggregate to reduce the number of marks..... 16
 - 18 – Dashboard objects have friendly names [optional] 16
 - 19 – Focus order follows logical sequence 17
 - 20 – Support mobile layout (resizing 200%) 18
 - 21 – Scroll in one dimension only 18
 - 22 – Pointer targets at least 44 x 44 pixels [Optional]..... 18
 - 23 – Repeated elements are consistent 19
 - 24 – Submitted for usability and accessibility user testing 19
 - 25 – Downloadable sheets are numbered 19
- Testing with screen readers 20**
- Common mistakes and pitfalls 21**
- Additional best practices..... 22**
- Tableau limitations 24**
- Version history 31**

Introduction

Accessibility

For the Oregon Health Authority Public Health Division, accessibility is when a person experiencing an equity barrier is afforded the opportunity to acquire the same information, engage in the same interactions, and experience the same services in an equally integrated, effective and timely manner, and with substantially equivalent ease of use.

The purpose of accessible Tableau design, or accessibility remediation of existing products, is to ensure that information presented is more accessible to all users. This includes those with disabilities or access needs which can be situational, temporary, or permanent¹. Dashboard designers should consider user exclusions by sight, sound, dexterity, and comprehension. Assistive technology is available to help users interact with digital content. Some people use a screen reader or select-to-speak technology on a mobile device to navigate and read the information displayed on the page; others use a keyboard or keyboard alternatives instead of a mouse to navigate a dashboard. Some people use screen magnifiers to increase the size of the text or visualizations. Each of these modalities should be considered when designing or remediating a dashboard.

It is important to consider the concept of [universal design](#), design that is usable by all people, to the greatest extent possible, without the need for adaptation or specialized design. “This is not a special requirement, for the benefit of only a minority of the population. It is a fundamental condition of good design. If [a product] is accessible, usable, convenient and a pleasure to use, everyone benefits.”²

Web content accessibility guidelines (WCAG)

WCAG defines the standards for accessible digital content, including data. WCAG 2.0 Levels A and AA are the legal standard referenced by Section 508 and case law on digital accessibility under the ADA (Americans with Disabilities Act).

This document will describe specific items within the WCAG that were applicable to Tableau accessibility remediations.³ The WCAG guidelines are designed as technical standards for anyone building web content. The guidelines are divided into [4 principles](#):

1. **Perceivable:** Information and components must be perceivable to all users.
2. **Operable:** Users must be able to operate the interface.

¹ Kund E. [Build Accessible Dashboards](#). Tableau Conference 2022.

² [What is Universal Design?](#) National Disability Authority (NDA) Centre for Excellence in Universal Design; c2020.

³ Colby R, Gupton K. [Applying Web Content Accessibility Guidelines WCAG Data Visualizations](#). Tableau; 2021 April 20.

3. **Understandable:** Users must be able to understand both the information and the operation of the interface.
4. **Robust:** Content must be sufficiently robust to allow for interpretation by a variety of users, including those utilizing assistive technologies.

Each principle includes specific guidelines that describe how to meet it, and each guideline has specific criteria that explain it in additional detail. Some WCAG 2.0 success criteria were not applicable to Tableau dashboard reports. Others are marked as optional here because they're not required by law – WCAG 2.1 or higher, or Level AAA – but are still best practices that developers can strive to achieve when possible. Before finalizing remediations, testing should be administered to determine if the content satisfies the criteria. It is recommended that users with disabilities and assistive technology users be involved in the testing.

This document includes explanations of accessibility items, the process for implementing them, and examples if needed.

Remediation

1 – Descriptive links

WCAG Links: [Success Criterion 2.4.4: Link Purpose \(In Context\)](#)
[Success Criterion 2.4.9 Link Purpose \(Link Only\)](#)

Explanation: Whenever a link is present on a dashboard, there needs to be accompanying language provided that describes what the link is and where it will take the user. This ensures that users can make an informed decision before clicking on the link and that they understand what they will see once the link opens. It is also important to note that names of the workbooks and dashboards should be short, intentional, and remain consistent as they generate the dashboard URL. If these names are changed, it will reset the page view counts on Tableau Public and will break links; it is best to keep them the same whenever possible. Likewise, sheet names should also be short, intentional, and remain consistent as they are listed in the download dialog window and may break web scraping process if changed.

Process: Currently this feature is not supported in Tableau Desktop, so it needs to be completed in a web-based version of Tableau (Public or Server). Additionally, note that if a hyperlink is in an edited section (for example, a title, filter, or caption), Tableau will remove all hyperlinks in the section once the text has been edited.

1. Start with the sentence where the link is embedded, leaving the link as a placeholder.
2. Paste the link in the desired section.
3. Press edit and change the default option.

Example: Note the language added prior to the link in the notes below.

please visit the [COVID-19 Regional Support Approach website.](#)

2 – Descriptive title

WCAG Link: [Success Criterion 2.4.2: Page Titled](#)

Explanation: Every dashboard must include a title that describes the content of that page. Titles should also be included for all charts, graphs, or tables. This allows people using assistive technology to read or hear a preview of the content displayed in the data visualizations.

Process: See process for 3.1 below.

3 – Text requirements

3.1 – Title and caption have narrative text

WCAG Link: [Success Criterion 1.4.5: Images of Text](#)

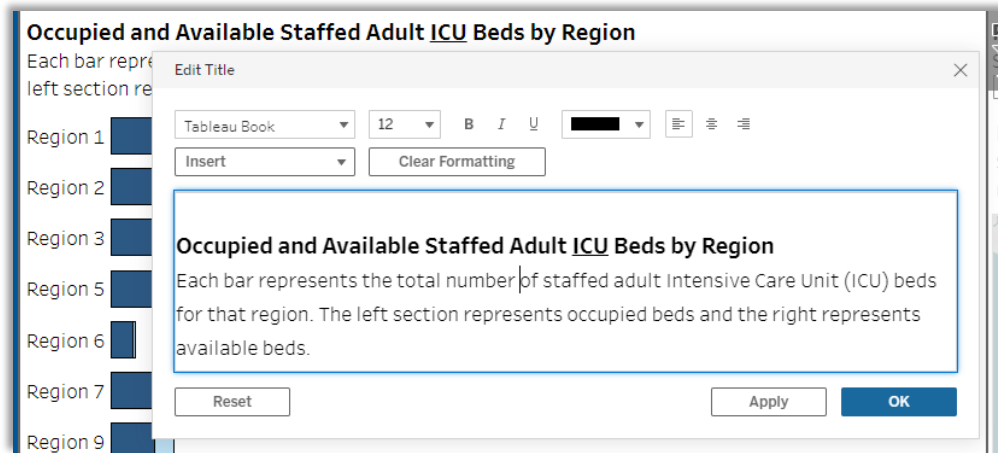
Explanation: Any graph or chart in Tableau must include text that briefly describes the information displayed in the visualization. Screen readers cannot read text that is embedded within an image; labels or text inside a graph will not be heard by non-sighted users. However, screen readers can read text included in titles and captions. When using captions, delete the autogenerated caption text and replace with custom text. Titles and captions must be shown (Select “Show Title” or “Show Caption” from the Worksheet menu) to be read by screen readers. Tableau only puts heading markers (<h1>, <h2>, <h3>) on titles and expands those to captions.

Process:

1. Right click the arrow at the top of the graph or chart and select either “Title” or “Caption”. You will then see the title at the top or the caption at the bottom.
2. Right click on the title or the caption and click “Edit Title” or “Edit Caption.”
3. Add 1-3 sentences of descriptive text.
4. Click “Apply” and then “OK.”

The title of the graph should be bolded and larger than the descriptive text. [Best practice](#) is to use at least 14-point font for the title and 12-point font for the descriptive text, if possible.

Example: In the example below, descriptive text was added above the bar graph to explain what the visualization is displaying. The existing title of the graph – “Occupied and Available Staffed Adult ICU Beds by Region” – remains there, with the added language underneath.



3.2 – Convert text objects to worksheets

WCAG Link: [Success Criterion 1.4.5: Images of Text](#)

Explanation: Text objects in Tableau are not read by screen readers. To ensure that the dashboard narrative is accessible, all text objects must be converted into worksheets.

Process:

1. Check the dashboard for text objects. Complete steps below for each text object.
2. Double click on the text object.
3. Highlight the text and copy it.
4. Delete the text object.
5. At the bottom of the screen, click left of the three + signs to create a new worksheet.
6. Double click the name of the worksheet at the bottom and re-name it.
7. In the new worksheet, double click on the title at the top and paste the text into the title. If there is too much text and the character limit is exceeded, use captions for the remaining text.
8. If there are over 1000 marks on the page, the screen-reader will skip the title. In this case, you may need to add something to the body of the sheet, even if it is only a blank calculated field.
9. Adjust the font, sizing, and background of the title as needed.
10. Return to the dashboard. Select the worksheet from the left side of the screen.
11. Then add the new worksheet to the dashboard. There are two methods for this:

- Click and drag it over to the dashboard where you want the sheet to be.
 - Or right click and select “Add to Dashboard.” The sheet will then be added automatically to the top left corner of the page.
12. Select the arrow in the top right and click “Show Title” and then “Show Caption” for the title and the caption to be read by the screen-reader.
 13. Move and resize the worksheet on the dashboard as needed. It is often helpful to put the new sheet into an existing container or create a new container to ensure a consistent layout.
 14. Repeat for the mobile layout, if customized.

3.3 – Add dynamic text to titles, captions

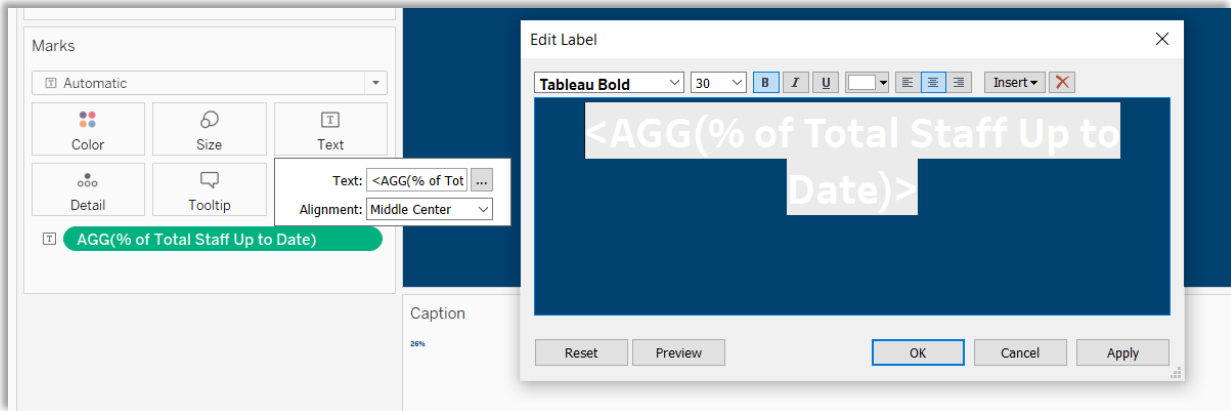
WCAG Link: [Success Criterion 1.4.5: Images of Text](#)

Explanation: Although titles and captions in a worksheet can be read by screen readers, text on the label marks card cannot be read by a screen reader. The best solution to this is to add dynamic text – numbers connected to a measure value or a calculated field that update when new data are uploaded – to the title or captions. In this instance, the title or caption text is repeating information displayed in the data visualization. Adjusting the title or caption text format to match the background ensures that the text will be read aloud by a screen-reader but will not add visual clutter for sighted users.

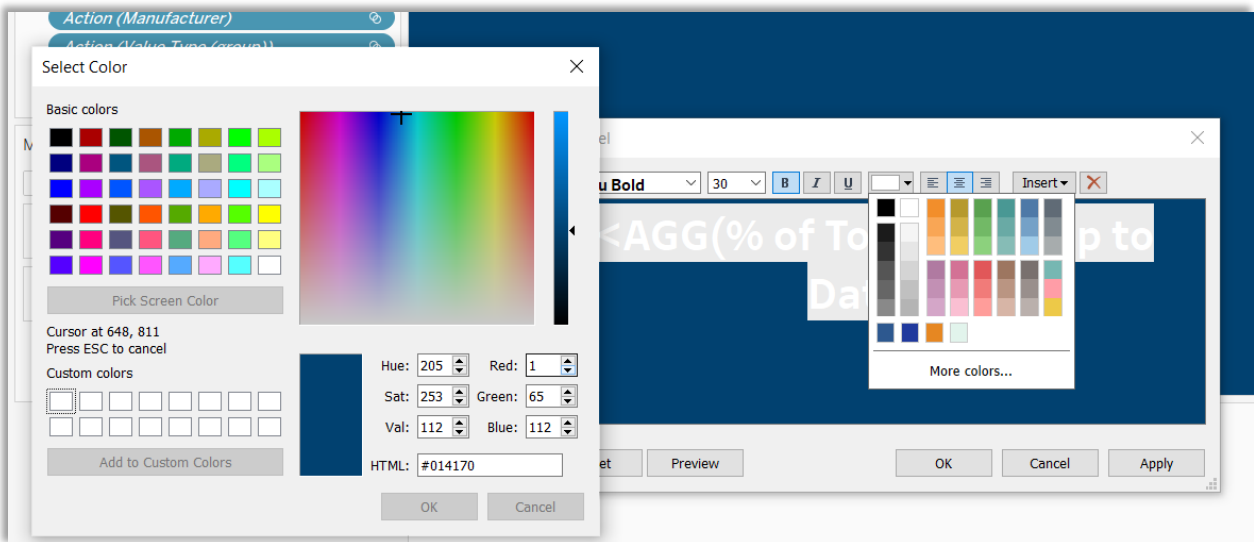
Process:

1. Copy text in the body of the worksheet. In the marks section, click “Text” and then click the 3 dots on the right side. This will open a window with the dynamic text.
2. Highlight the text and copy it. Exit out of the window.
3. Near the bottom of the screen, right click and select “Caption.” You’ll see a box at the bottom.
4. Right click in that box and select “Edit Caption.” A text box will pop up.
5. Paste the text into the caption.
6. Change the size of the text to 8-pt. font or smaller to avoid the captions taking up space.
7. Change the color of the text to match the color of the background of the dashboard to hide the text.
 - a. If the background is white, make the text white.
 - b. If the background is another color, click on “More colors...” then, click on “Pick Screen Color” and use the mouse to select the background color of the worksheet.

Example: In the image below, note the steps required to copy the dynamic text from the label and paste it into the caption.



In the image below, note the steps required to match the color of the caption text to the background color to hide the caption from view completely.



4 – Descriptive headings with key takeaways

WCAG Links: [Success Criterion 2.4.6: Headings and Labels](#)
[Success Criterion 2.4.10: Section Headings](#)

Explanation: Each dashboard must have a heading that describes the most important concepts or takeaways that a user should understand. There is some subjectivity involved in determining how detailed these descriptions need to be, as too much text can be distracting from the rest of the dashboard. Descriptive headings should be added at the top of the dashboard as a part of the title.

5 – Narrative uses plain language principles

WCAG Link: [Success Criterion 3.1.5: Reading Level](#)

For information on plain language, please see the [Oregon Department of Human Services | Oregon Health Authority \(ODHS|OHA\) Plain Language Basics](#):

“Writing in plain language is writing in a way your audience understands. When you use plain language, your audience can find, understand and use what they read.

1. Use common words that clearly and directly state what you mean.
2. Use short, simple sentences.
3. Share information in a logical order.
4. Avoid jargon or abbreviations.
5. Write like you talk. Use active voice and present tense whenever possible.”

Additional plain language resources are available from the [Department of Administrative Services \(DAS\)](#).

6 – Title, caption, and data table for each visualization

WCAG Link: [Success Criterion 1.1.1: Non-text Content](#)

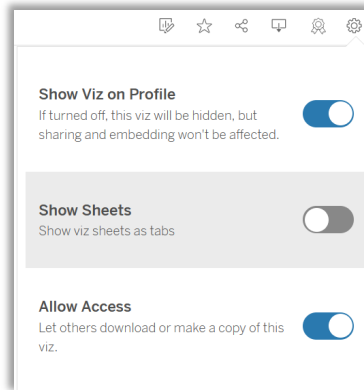
Explanation: Every chart or table must have a title and a caption that describes clearly the information presented in the corresponding visualization, as well as a corresponding data table. For more details on titles/captions, refer to item [3.1 – Title and Caption Have Narrative Text](#).

This refers to there being a textual alternative that serves the equivalent purpose. In practice, this means that when data are visually represented in any graph or chart, there should be a corresponding data table that is either presented to users on the dashboard or available for users to download and view separately. Note that this should be done only when the underlying data of the dashboard does not contain sensitive information. There are some instances where the full access should not be made available to all users due to privacy or security concerns.

Process:

For Tableau Public:

1. On the accompanying summary table of a dashboard, navigate to the workbook published and select settings at the top right.
2. Toggle on the “Allow Access” switch at the bottom as seen below.



For Tableau Server:

1. As a Project Lead, customize the permissions for your Project to grant Guest users the ability to download summary data from the Workbooks tab.

7 – Sheet tooltips turned off or non-essential

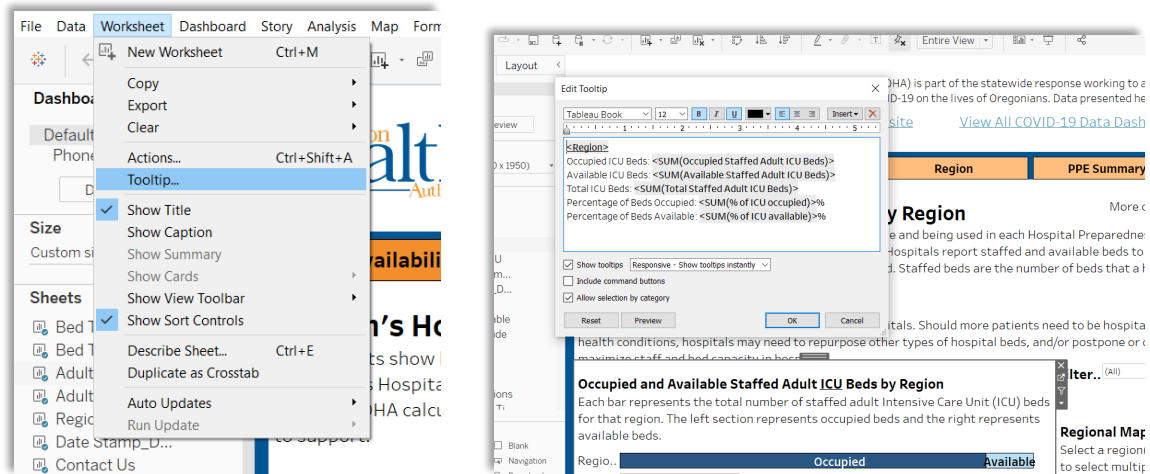
WCAG Link: [Success Criterion 1.1.1: Non-text Content](#)

Explanation: Adding tooltips to display information can be a useful way to include additional information on a dashboard. However, people with visual exclusions and keyboard-only users are not able to view tooltips. Therefore, any data or information included in tooltips must be non-essential, meaning it is not needed to understand the visualization. If the information in the tooltips can be viewed or accessed another way, then the tooltips are considered non-essential and can be included. Tooltips can provide on-demand detail for sighted users, adding valuable context to the chart or graph.

Process: Tooltips can be turned on and off in the worksheets.

1. From the Worksheet menu, select “Edit Tooltip.”
2. Uncheck or check “Show tooltips.”
3. Customize the tooltip text, then click “OK.”

Example: See example below for the steps needed to edit the tooltips.



8 – Alt text for images

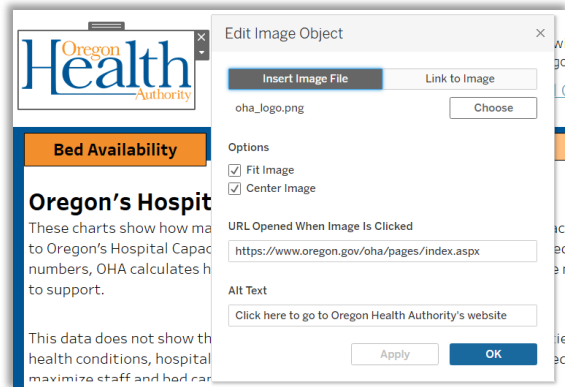
WCAG Link: [Success Criterion 1.1.1: Non-text Content](#)

Explanation: All images on a dashboard must have an accompanying text alternative that conveys the same information. The purpose of this text is to provide context and user guidance for the link embedded within the image. This allows those using screen readers or select-to-speak technology to hear where the image will take them.

Process:

1. Click on the dropdown arrow at the top right of the image or logo and select “Edit Image”.
2. Add a URL. The image’s Alt Text will not be read by screen readers without a URL.
3. Add the Alt Text in bottom section to describe what the link is. For more on how to write image descriptions, see Figure 2.1 in the [ODHS|OHA Accessibility Manual](#).
4. Click “Apply” and then “OK.”

Example:

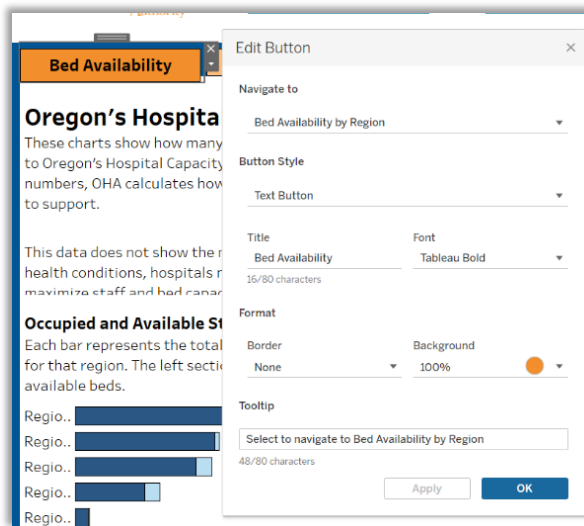


9 – Tooltip text for navigation buttons

WCAG Link: [Success Criterion 1.1.1: Non-text Content](#)

Explanation: Any navigation button that is used to click or move through a dashboard must include a tooltip to explain the function or purpose embedded in the button. Users can click on the navigation buttons to navigate to a different dashboard, so it’s important to have clear instructions.

Example: In the screenshot below, the added tooltip allows the user to see or hear “Select to navigate to Bed Availability by Region” before they click, so that they will understand that they can choose this button to navigate to the Bed Availability dashboard. When there is an image, a screen reader will read out only the word “image”, so the tooltip is crucial to add context.



10 – Transcript, captions for embedded YouTube videos

WCAG Links: [Success Criterion 1.2.1: Audio-only and Video-only \(Prerecorded\)](#),
[Success Criterion 1.2.2: Captions \(Prerecorded\)](#)

For more information on creating accessible videos, please see "[Creating Accessible Videos: A Brief Guide](#)" by Rooted in Rights, supported by the Ford Foundation:

“An accessible video is one that contains content that everyone can consume, including people with disabilities who may be unable to hear the audio or see the images on screen. This means that, among other things, the auditory experience of the video must convey all essential information for those who cannot see, just as the visual experience will need to do the same for people who cannot hear.

This can primarily be achieved by including the following components in—or along with—your video:

- Captions
- Audio description
- A transcript”

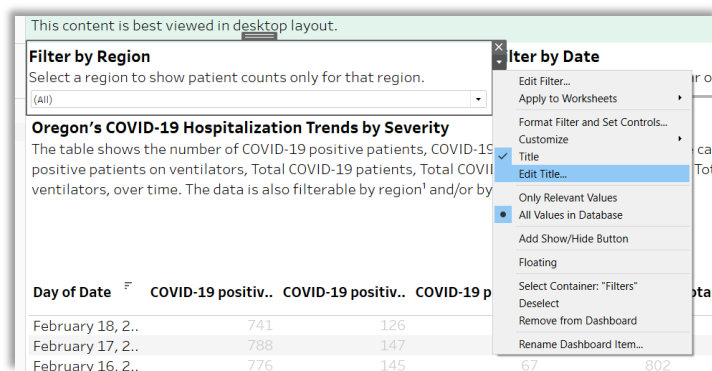
11 – Descriptions for filter, parameter, set, legend titles

WCAG Links: [Success Criterion 1.1.1: Non-text Content](#)
[Success Criterion 2.5.3: Label in Name](#)
[Success Criterion 3.3.2: Labels or Instructions](#)

Explanation: Labels or instructions must be provided for any interactive content that requires user input. Therefore, anything that has a call to action associated with it must have labels or instructions.

Process: This process relates to filters but can be applied to legend titles as well.

1. Right click the filter and click the check box on Title.
2. Right click again and click on “Edit Title.” A text box will pop-up where text can be added.
3. Click “Apply” and then “OK.”



12 – Instructions use a combination of shape, color, size, direction

WCAG Link: [Success Criterion 1.3.3: Sensory Characteristics](#)

Explanation: Instructions for how to view, filter, or otherwise manipulate the content should be based on a combination of shape, color, size, and direction. This helps those who can perceive one of those but not the other; for example, some users cannot see color but can distinguish between shapes. For additional clarity on how to meet this standard, see [item 14 below](#).

Process/Example: In the title of a filter, explain how to filter the data without relying solely on shape, color, size, or direction. For example, instead of including language such as “Filter below” say “Use the drop-down menu below to filter by region.” The second option includes both a direction and a description of what the filter is (a drop down), making it more accessible.

13 – Definitions of unusual words, abbreviations [optional]

WCAG Links: [Success Criterion 3.1.3: Unusual Words](#)
[Success Criterion 3.1.4: Abbreviations](#)

Explanation: Any uncommon or obscure words need to be defined to ensure people can understand the meaning. Moreover, all abbreviations should be written out with appropriate definitions.

Example: For example, PPE includes the full definition (Personal Protective Equipment) as shown in the example below.

Oregon PPE by Hospital

Each week, hospitals report how much PPE (Personal Protective Equipment) they have to Oregon's Hospital Capacity Web System (HOSCAP)¹:

- *Amount of PPE on hand* is the number of PPE items the hospital currently has (for example, number of gloves). ..

14 – Color paired with labeling or shapes

WCAG Link: [Success Criterion 1.4.1: Use of Color](#)

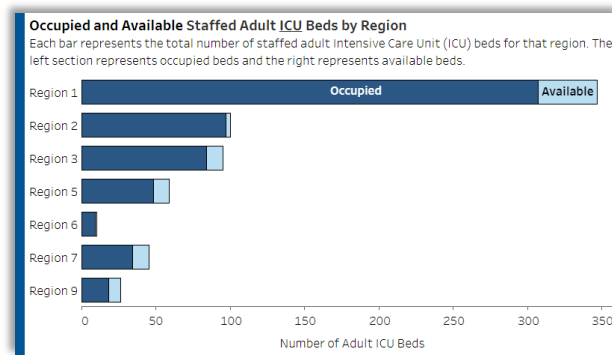
Explanation: Because some dashboard users may have visual exclusions or colorblindness, it is necessary to ensure that color is not the only way to distinguish between two categories or two areas. Additional labeling or shapes should be added as a second way for users to grasp these differences.

Process: There are multiple ways to meet this standard depending on the type of visualization. On grouped bar charts, adding labels for each series can meet the standard. For a stacked bar graph that is split into multiple sections, each with a different color, adding annotations for each section of the bar to clarify what the different areas represent, as shown below in example A, can be an effective solution. For line graphs, adding labels or annotations to the graph is a viable solution, as displayed below in example B. Lastly, adding shapes and colors to represent different categories of data can be effective.

Example A:

1. Right click on the area of the bar where you want the annotation to be.
2. Select “Annotate area.”
3. Type in the text. If the bar is a dark color, the text should likely be white.
4. Right click the annotation box and select “Format.”
5. In the top left, click “Box” and then “None” to remove the box around the annotation.
6. Then, click “Shading” and select “None” to remove any shading.
7. Click “Line” and select “None” to remove any lines.
8. Click and drag the annotation to the correct place on the bar.

Note the “Occupied” and “Available” annotations added to the region 1 bar below. For a single stacked bar chart, labels could be used instead of annotations.



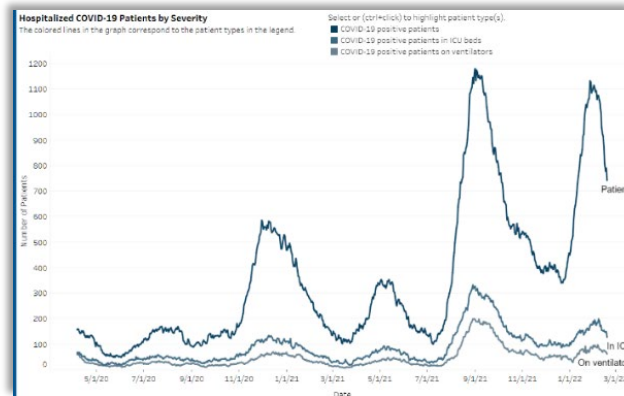
Example B: For line graphs, adding labels to the graph is a viable solution. If you were displaying a static line graph without any filters, annotations could be added instead of labels. However, given the line graph shown below includes multiple filters, annotations are not an appropriate solution as they do not move with the lines once the filters are applied. The labels will adjust and automatically show up adjacent to the last data point, even with filters applied.

1. Create a calculated field to instruct Tableau to create three different labels, one for each measure name displayed on the line graph. The syntax for the calculated field in this example is:


```

      If CONTAINS([Measure Name], 'ventilators') THEN 'On ventilators'
      ELSE IF CONTAINS([Measure Name], 'ICU') THEN 'In ICU'
      ELSE 'Patients' END
      
```
2. Drag the calculated field on to the Label.
3. Click label and select “Most recent” so that the labels are next to the most recent data point.
4. Click label and check “Allow labels to overlap other marks” so all labels are displayed.
5. Click on the labels themselves to adjust the position manually, if necessary.

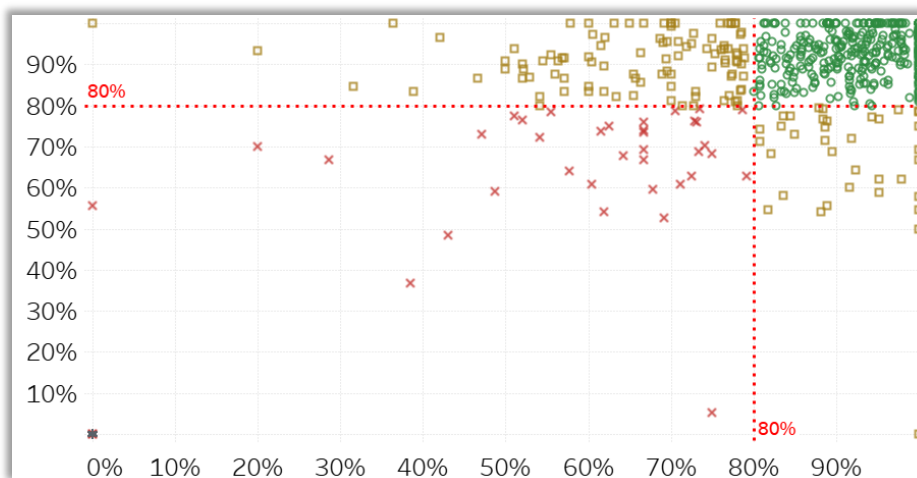
Note the three labels to the right of each of the lines on the graph below.



Example C: Using shapes as an additional marker to differentiate between groups or categories of data is an effective strategy. It may not work well in certain situations, can result in cognitive overload from too many shapes, colors and lines. However, for the scatterplot shown below, it was an appropriate remediation to meet the accessibility standard.

1. Add the measure on the “Color” card to the “Shape” card by dragging the measure name over to “Shape” while holding Ctrl.
2. Click “Shape” and select the “Filled” or “Default” shape palette. Click on the data item on the left and then select which shape on the right to correspond to it. Basic shapes such as squares and circles are best.
3. Describe what each shape represents in the title so that users know what each color and shape represent.

Note the green circles, yellow squares, and red exes shown on the scatterplot below



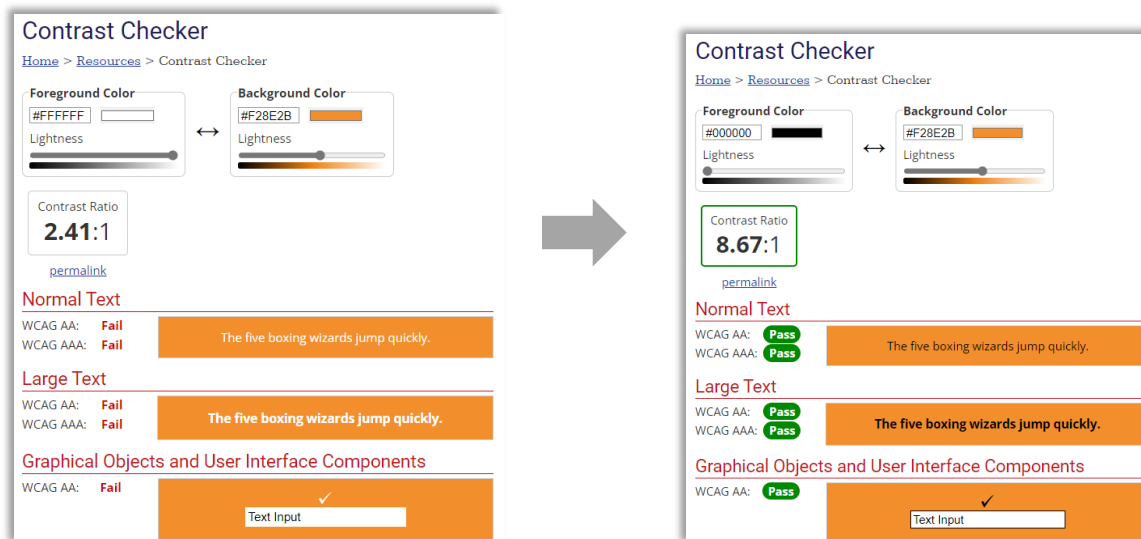
15 – Text contrast ratio of at least 4.5:1, or 3:1 if Large (18pt or bolded 14pt)

WCAG Links: [Success Criterion 1.4.3: Contrast \(Minimum\)](#)
[Success Criterion 1.4.6: Contrast \(Enhanced\)](#)

Explanation: The color contrast ratio between the text and the background must be at least 4.5:1, or 3:1 if the text is large (either at least 18 point or bolded 14 point). This ensures that the text may be differentiated from the background and readable for sighted users.

Process: To ensure that text meets the standard ratio requirement, test the contrast ratios using a contrast ratio calculator such as [WebAIM's Contrast Checker](#). Tap the color code to insert the proper color code for the text (foreground) and background using the eyedrop icon. The page will then update to show the contrast ratio and indicate whether the two colors pass or fail the various standards. This ratio needs to be at least 4.5:1 for small text or 3:1 for large text. A 7:1 ratio is an optional reach goal (Level AAA).

Example: The font inside the orange navigation boxes was changed to black to comply with the required color ratio. White text failed the contrast ratio requirements between the text color and the background color, as shown in the left screenshot below, but the black foreground against the orange background passed the standards.



16 – Non-text contrast ratio of at least 3:1 [Optional]

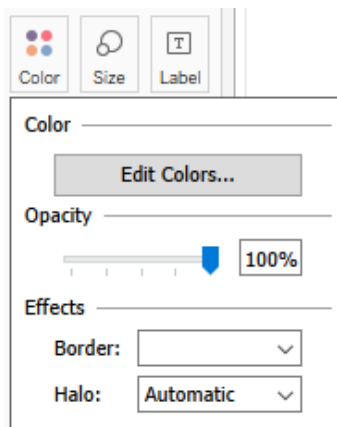
WCAG Link: [Success Criterion 1.4.11: Non-text Contrast](#)

Explanation: The color contrast ratio between any non-text image and the background should be at least 3:1. This ensures that users are able to differentiate the image from the background, especially important for users with colorblindness and low vision. One strategy to meet this standard is to apply a black or white border around adjacent data elements in a chart (for example, stacked bar charts, maps).

Process:

1. To ensure that all non-text meets the standard ratio requirement, test the contrast ratios using a [contrast checker](#).
2. After inserting the color code for the non-text color (foreground) and background, the page will update to show the contrast ratio and indicate whether the two colors pass or fail the various standards.
3. To add a border, click on “Colors” under the marks, and then click Border.

Example:



17 – Filter or aggregate to reduce the number of marks

See the [Tableau Limitations](#) section for a discussion of how to address this.

18 – Dashboard objects have friendly names [optional]

While renaming dashboard items remains a best practice for developers to organize their item hierarchy, this step is no longer needed to facilitate correcting a dashboard’s focus order. [Manually editing the XML focus order](#) is not routinely advisable, as it is labor intensive and has resulted in some workbook instability, including the loss of mobile layouts.

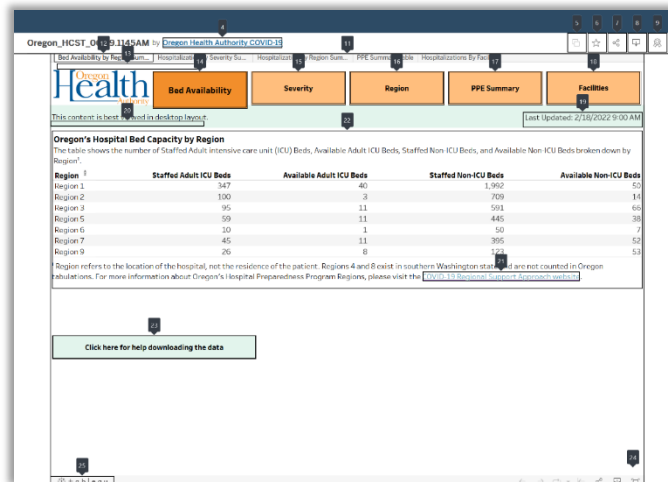
19 – Focus order follows logical sequence

WCAG Links: [Success Criterion 1.3.2: Meaningful Sequence](#)
[Success Criterion 2.4.3: Focus Order](#)

Explanation: Focus order must flow logically and linearly, from top down and/or from left to right. This can be edited in the XML if needed. Tableau [states](#) that it does not currently support a way to set the focus order independently of an object’s location on the dashboard, without unpackaging it and editing the XML. The focus order was based on the order in which an object was placed on the dashboard; meaning if the logo was put in first that would be the first place a screen-reader user would tab to. However, Tableau version 2021.3 changed the default focus order to be based off the object’s position on the page so that it starts with the top left corner, going left to right and top down.

Process: To change the default focus order on workbooks created with older versions of Tableau, Tableau states that “Any dashboard created in a new or existing workbook using Tableau 2021.3 or later is considered a ‘new dashboard’. A new duplicate of an existing dashboard is also a ‘new dashboard’.” Therefore, if a dashboard owner is using Tableau 2021.3 (or newer) and has an older workbook that was made with a previous version, they need to duplicate each dashboard to change the focus order. This cannot be done at the workbook level at this time.

Example:



20 – Support mobile layout (resizing 200%)

WCAG Link: [Success Criterion 1.4.4: Resize text](#)
[W3C User Agent Accessibility Guidelines \(UAAG\) 1.0 Checkpoint 4.1](#)
[Tableau Mobile Layout Guidance](#)

Explanation: To support people with visual exclusions, it is necessary to create web content that allows the user to scale content effectively when zooming. The only way to do that is to build a separate mobile layout. [Tableau documentation](#) shows that the device layout displayed – desktop, phone, or tablet – is determined by the smallest iframe dimension. An [iframe](#), or inline frame, is an HTML object that displays another website within the HTML document. When the corresponding [CSS breakpoint](#) hits (usually at 200%), Tableau will switch from the desktop layout to the mobile layout. If the user is not already zoomed to 200% when landing on the page, they will need to refresh the page to trigger the change from the desktop layout to the mobile layout. Below is a summary of how to make a mobile layout.

Process: It is important that no data or information is lost when transitioning from the default layout to the mobile layout. Screen-reader instructions should be included in the mobile layout as well. Because a table or visualization may need to be altered or tweaked slightly for it to render appropriately in a mobile layout, it is often best practice to duplicate these worksheets and create a separate mobile version so that these mobile version changes do not show up on the desktop layout.

21 – Scroll in one dimension only

WCAG Link: [Success Criterion 1.4.10: Reflow](#)
Optional under WCAG 2.0, required under 2.1

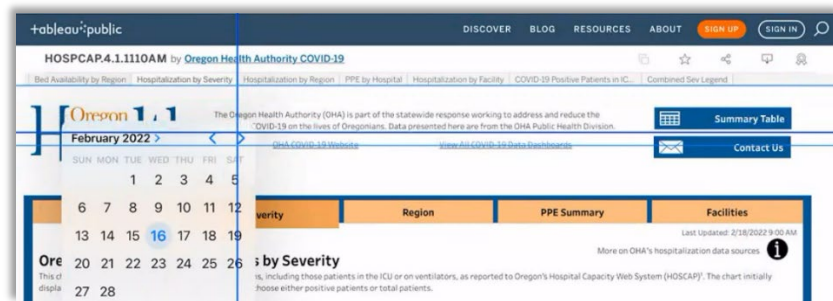
Explanation: To ensure that dashboards look the same when zoomed at 400% as they do at 100%, a page needs to reflow to a single column so that nothing is cutoff on the sides. This allows there to be no loss of information or functionality on the dashboard when changing from 100% to 400%. However, the WCAG scrolling rule does not apply to tables, maps, or graphs within the dashboard. See the [Tableau limitations](#) section for more about this.

22 – Pointer targets at least 44 x 44 pixels [Optional]

WCAG Link: [Success Criterion 2.5.5: Target Size](#)

Explanation: Pointers are any object with a call to action associated with it. Pointer targets need to be resized to be at least 44 width X 44 height. This benefits users with mobility exclusions and those who may be using only one switch to interact with content.

Process: For interactive objects on a worksheet, select layout and update the size accordingly on the top left, so that the width and height of the object is at least 44 X 44.



23 – Repeated elements are consistent

WCAG Links: [Success Criterion 3.2.3: Consistent Navigation](#)
[Success Criterion 3.2.4: Consistent Identification](#)

Explanation: It is important to make sure that dashboards are consistent. This refers to fonts, colors, and sizes within a dashboard and across dashboards, as well as functional components such as filters, navigation buttons, and graphs. Design elements including color palettes, borders, and filter types, should remain the same whenever possible. This helps the user orient themselves to the page without unnecessary distractions.

This item is both a best practice and an accessibility standard, as all users benefit from consistency. However, for those who use screen-readers and screen magnifiers, the need for consistency is even greater as they rely more on page landmarks.

Process: It is a best practice to build one dashboard and then use that same structure for the rest of the dashboards. The preferred strategy is to duplicate the first dashboard and then replace content or visualizations as needed to ensure consistency.

24 – Submitted for usability and accessibility user testing

Explanation: Part of the process of implementing accessibility remediations includes having the dashboards tested by screen-reader users to ensure that they can navigate them. It is a best practice to partner with an accessibility expert or someone who is well-versed in the WCAG guidelines.

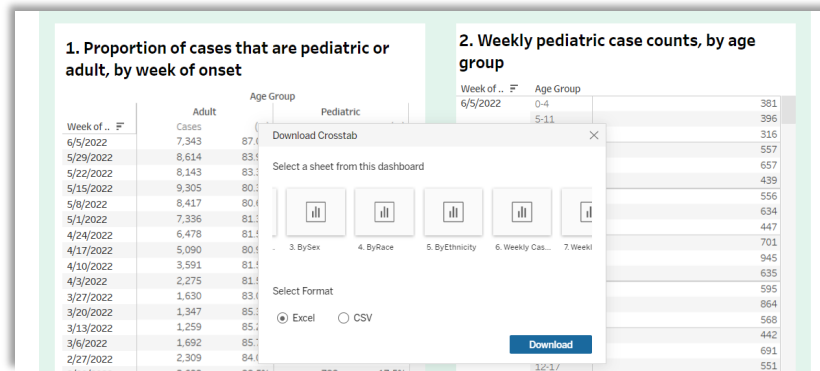
25 – Downloadable sheets are numbered

Explanation: To aid in both consistent summary table dashboard layouts, and to help users navigate visualizations and images within a dashboard, a best practice is to number each data table. This makes it easier for users to navigate and download the data, as it will show the ordered numbers while viewing the crosstabs, as shown in the example below.

Process:

1. In the title of the table enter the number of the sheet.
2. Add a space, and then type in the title of the visualization. Spaces are included to make sure that if there is a number at the front of the title it does not cause confusion.
3. Add the corresponding table number to the title of the sheet at the bottom.

Example:



Testing with screen readers

A crucial part of ensuring that a Tableau dashboard is accessible is testing it with a screen reader and with select-to-speak technology. It is best to do this on Tableau Public or the Tableau Server to replicate the user experience; Tableau Desktop often functions differently. When updating a dashboard, best practice is to retest as described below if possible. There are three main parts of the testing process.

1. **Tabbing:** An important part of ensuring that a Tableau dashboard is accessible is confirming that all essential elements of the dashboard are keyboard navigable – using the tab, shift, and enter keys instead of a mouse. The first step to test whether a dashboard is accessible is to simply start at the top left and use the Tab key to move to the next dashboard element. If certain elements are skipped, it may be necessary to adjust the focus order, the item hierarchy, or to replace inaccessible dashboard elements with accessible ones.
2. **Download a screen reader:** The next step is to use a screen reader to test that all elements on the dashboard can be read aloud. There are several different screen reader applications, all with various advantages and disadvantages. [NVDA](#) is preferred because it is compatible with most systems (iOS and Microsoft) and is relatively easy to use ([basic NVDA commands](#)). Here are some basic instructions [for configuring and using NVDA](#).

3. **Test on mobile:** After building the mobile layouts, test that the dashboard functions well on a mobile device using select-to-speak technology. Both [iPhones \(How-to video for VoiceOver\)](#) and [Android devices \(How-to website for gestures\)](#) come with built in select to speak assistive technology that can be utilized on webpages.

Common mistakes and pitfalls

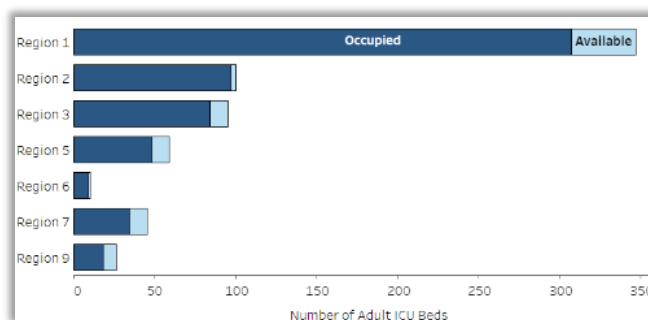
- **Mobile layouts:** When working on the mobile layouts, any updates made to the font, language, size, color, background, or graph axes will also be reflected in both the desktop and mobile layouts. One workaround for this is to duplicate the sheet to create a new mobile version. This allows developers to customize the sheet as needed to fit within the mobile layout without affecting the original default desktop layout. The mobile version of the sheet must first be added to the default layout before it can be added to the mobile layout. The mobile version may be hidden on the default layout within a layout container near the bottom of the dashboard that uses a Show/Hide button. More details can be found in [this Tableau article](#).
- **Colorblind palette:** For data visualizations, Tableau's default palette or colorblind palette may not be the best choice. While these color options may be easier for some users with colorblindness, if the colors are touching each other in any way (for example, to delineate regions on a map), they may not have sufficient contrast to meet WCAG standards. One option is to use a Tableau palette as a starting point and tweak the colors as needed, making them darker or lighter to enhance the contrast ratios. Another option is to add a black or white border between the colors to provide sufficient contrast. It is also important to include a second way to convey meaning in addition to color; this could be direct labels or shapes.
- **Font size:** When adjusting the font size or font type on a dashboard, it is important to change the filters and titles last. For example, you might want the data in a table to be 12-point font but the instructions for filtering to be in 14-point font. If you change the filter to be size 14, and then change the data table to be size 12, the text in the filter will revert to size 12 to match the data table. Therefore, it's always best to finalize the fonts inside the visualization first, and then adjust the filters or titles afterwards.
- **Font choice:** Calibri font is a standard font used by many organizations; however, Calibri is not compatible with Apple products. If the dashboard includes any text in Calibri font, this text will render as Times New Roman when the dashboard is viewed on an iOS device. The recommendation is to use Tableau's fonts (for example, Tableau Book, Tableau Bold, Tableau Regular).

- **Screen reader pausing:** When testing with a screen reader, it might continuously read the title and any captions or text underneath the title without pausing. This can create confusion for those using screen readers, as they may not be able to discern when the title ends and the additional text begins. One solution is to add a hidden period that matches the background color which will cause the screen reader to briefly pause.
- **Editing hyperlinks:** On Tableau Desktop, it is not currently possible to edit the text of a hyperlink. However, on Tableau Public and Tableau Server, developers can edit links and adjust fonts as needed.

Additional best practices

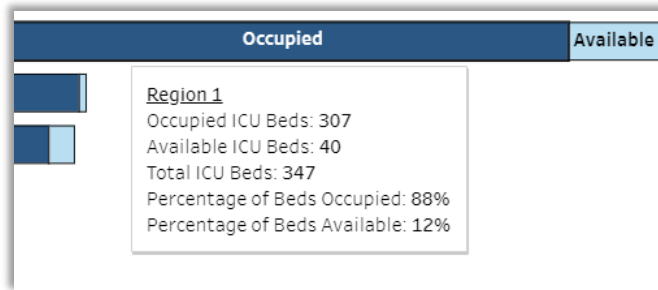
These concepts are not directly referred to within WCAG, but they improve accessibility indirectly via universal best practices.

- **Cognitive overload:** Cognitive overload is the idea that too much information on a dashboard can be overwhelming and can make it difficult to process the information. This is a well-known concept among Tableau developers and in the data visualization world overall. In practice, remembering the potential for cognitive overload should encourage developers to keep dashboards and data visualizations as simple and clean as possible. It is important to remove or delete any numbers, information, lines, or colors that do not relate directly to the visualization. Minimizing the amount of data on the page can help the user understand the main takeaways.
- **Gridlines:** When using line graphs, bar graphs, and scatterplots consider removing the gridlines from the graph. Sometimes they can be helpful for the user to comprehend the meaning of the graph but often they are unnecessary and somewhat distracting. For example, in the bar graph below, it is relatively easy for the user to look at the bars and see that Region 1 has approximately 350 total beds, Region 2 has approximately 100 total beds. Vertical gridlines are not needed for lower levels of precision, if approximate values are sufficient. However, this is not true for every graph; it often is a judgement call and a matter of personal preference.



- Concise language:** Building on the importance of avoiding cognitive overload described above, it is crucial to avoid adding too much text to a dashboard. There should be just enough to meet the accessibility requirements and to ensure that users have the appropriate context, but not too much that it distracts from the visualizations or overwhelms the user.
- Interactivity:** When possible, add interactive filters, highlighting, or actions to dashboard to enhance the user experience. These can take many forms depending on the type of data visualization but including at least one per dashboard helps to keep users engaged and interested.
- Tooltips:** How tooltip data are displayed and formatted impacts how the user experiences a dashboard. Tooltips should enhance the user experience and provide additional detail. For the example below, tooltips show percentages of available and occupied hospital beds out of the total number of hospital beds, which is additional information not directly conveyed through the bars. It is also crucial that Tooltips are formatted in a way that makes them simple and efficient for the user to understand the information. For example, if a visualization has data by county, the tooltips should show the county name at the top in bold and/or underlined font, so that the user can quickly check to see if they are viewing the correct county. As a reminder, tooltips are not accessible for some users so any information included in the tooltips must be non-essential and available elsewhere.

Tooltip example A:



Tooltip example B:

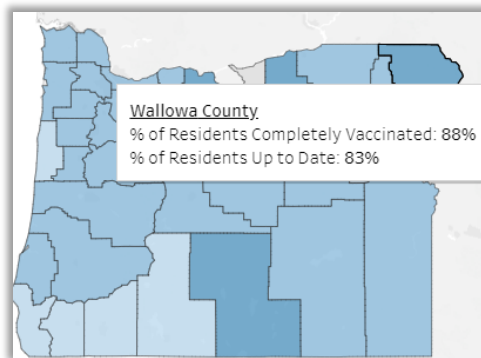


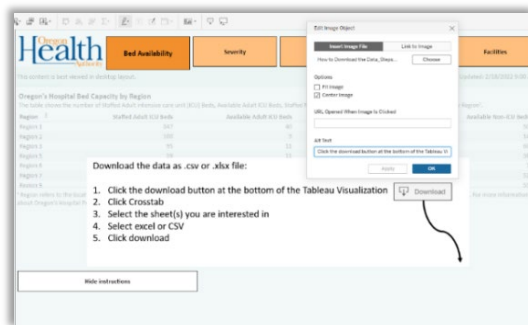
Tableau limitations

This section describes specific accessibility requirements developers are not currently able to meet due to technical limitations in Tableau. The software company has been made aware of these challenges in multiple ways (for example, discussions with customer service team members via phone or email, Tableau forums, other internet forums), but it is unclear which of these limitations, if any, Tableau is working to address. The structure of this section includes a description of each limitation as well as examples and possible solutions where available.

Alt text is provided for images

Limitation: Tableau will read Alt Text for images when using screen readers or select-to-speak technology but only if they have a link associated with them.

Example: In the example below, the intention is not to take screen reader users to another page but rather to ensure that they understand how to download the data. This limitation prevents them from hearing the download instructions provided in the Alt Text of the image. The example below is one where the image is non-essential. However, in other situations there may be images such as logos that are essential will not be read aloud unless there is an associated link.

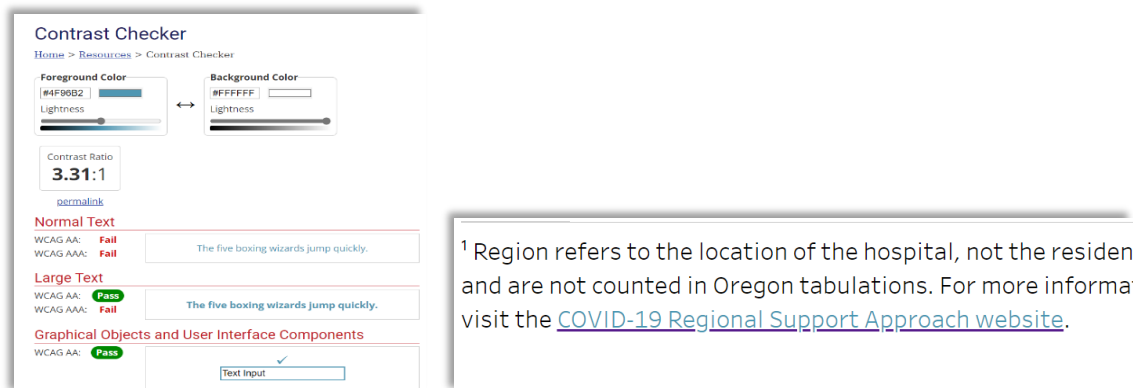


Contrast ratio

WCAG Links: [Success Criterion 1.4.3: Contrast \(Minimum\)](#)
[Success Criterion 1.4.6: Contrast \(Enhanced\)](#)

Limitation: The default color of hyperlinked text does not pass the minimum contrast ratio requirement provided by WCAG. Tableau will automatically change any functional URL link to the underlined blue font, and there is no way to change the font color.

Example: In the example below, note that the contrast checker indicates that the light blue color of the hyperlinked text fails the WCAG standard.



Solution 1: There are potential [solutions for this](#), but they often create more problems than they solve. The first strategy is to convert a field in a sheet to text and then apply a call to action so that when the user clicks the sheet, the hyperlink will open. When the field is in its own sheet, the text can be changed to any color and the hyperlink will still function properly. While this solves the contrast ratio problem, the recommendation is not to implement this for two reasons:

1. The user experience is not ideal. When converting a field to a sheet, the sheet becomes a pointer target. If the user then tries to highlight text in a paragraph they may accidentally click on the link even if when they did not intend to.
2. The sheet can be linked only to one URL, meaning individual sheets are needed for every link included in the dashboard, which could be quite labor intensive. Moreover, having multiple sheets attached to a dashboard slows down the performance of the workbook once it is published and can significantly delay any filtering or actions.

Solution 2: Another potential solution is to switch the worksheet from a white background to a dark background for any sheet that includes text blocks with hyperlinks. A dark background with color #262626 or darker will result in a contrast ratio of at least 4.56:1 with the default hyperlink color, which would pass the WCAG standard. All other text could simply be white.

While this does solve the problem, there are many instances where adding a dark background is not a viable option due to the desired design or appearance of the dashboard. However, this solution is generally preferred over Solution 1 above.

Mobile layout / dashboard resizing

WCAG Link: [Success Criterion 1.4.4: Resize text](#)

Limitation: When viewing a Tableau dashboard on a mobile device using voice over technology (on either iOS or Android devices), the usual double-tap method to click on something has historically not worked with objects that have a call-to-action associated with them. This included navigation buttons, show/hide buttons and filters. There was previously no known solution to that limitation.

Solution: With Tableau’s current version this seems to be working and it is anticipated to get better with future updates. The recommendation is to continue to monitor for potential issues.

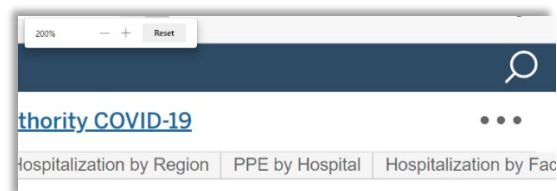
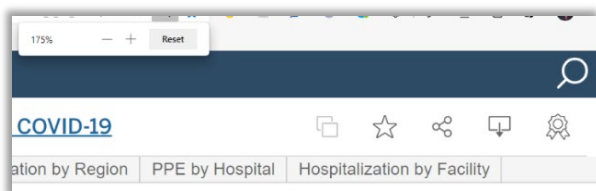
Scrolling in one dimension

WCAG Link: [Success Criterion 1.4.10: Reflow](#)

Limitation: This standard requires that the web page be resized to different CSS (cascading style sheets) breakpoints and that the mobile version be available on the main webpage. In Tableau, there is only one CSS breakpoint, and it can be achieved only after the user zooms to around 200% and then refreshes the page. Tableau does not allow for reflow to occur in a gradient and therefore these pages will not fully meet these criteria. [Here](#) is WCAG’s example of a website reflow.

Example: In the top right corner of any Tableau public webpage are Tableau regular webpage control elements. These are not part of the dashboard but rather are native to the Tableau webpage. When the user zooms in, these control elements are combined into an image with three dots, and then the user needs to click on the three dots to view the control elements in a column. However, that type of consistent zooming does not function similarly on dashboard itself. When zooming to 200%, some of the content is simply cut off and not merged into a single column.

In the examples below, the webpage is reflowing at 175% zoom on the left and to 200% zoom on the right. Note the three horizontal dots in the image on the right.



Below, note that the dashboard cut off both when zooming to 175% (bottom left) and to 200% (bottom right). The webpage will not automatically resize at this point, so users are able to view only a portion of the dashboard on a desktop. If viewing the dashboards on a phone/tablet, users can slide fingers to the side to see the rest of it. If the dashboard has a mobile layout, then when the webpage is refreshed, the dashboard will convert to show the mobile layout if zoomed to around 200% or higher.



Headings

WCAG Link: [Success Criterion 1.1.1: Non-text Content](#)

Limitation: Only the headings should be labeled as <h2>. A widely adopted way of navigating a webpage via a screen-reader is to read out the section headings of a page. This gives users a quick overview of what they can anticipate while being on the page. However, in Tableau, the heading and any text that follows need to be in the <h2> or in the title of the sheet for text to be read by screen-readers. This severely hinders that main approach towards skimming a page while using a screen-reader.

Example: The example below depicts that the heading and text are both labeled as <h2>.

Hospital/Health System	Face Shield		Gloves		Amount on Hand
	Amount on Hand	Burn Rate	Amount on Hand	Burn Rate	
Adventist (HEALTH SYSTEM)	5,413	32	204	1,279,808	22,889
Asante (HEALTH SYSTEM)	8,651	22	389	822,800	98,025
CHS (HEALTH SYSTEM)	4,524	21	273	938,000	12,650
Kaiser (HEALTH SYSTEM)	20,814	141	411	6,838,340	0
Legacy (HEALTH SYSTEM)	13,150	141	93	21,208,900	122,507
OHSU (HEALTH SYSTEM)	207	305		88,620	244
Providence (HEALTH SYSTEM)	41,114	1,370	50	2,761,810	92,000
Previdence (HEALTH SYSTEM)	4,236	605	156	498,220	66,988
Saint Alphonsus (HEALTH SYSTEM)	364	2	404	35,625	12,742

Text boxes

WCAG Link: [Success Criterion 1.1.1: Non-text Content](#)

Limitation: Tableau [states](#) that visualizations are treated by screen readers as images. Tableau will recognize titles and descriptions for corresponding HTML elements <h2> to be detected by screen-readers. Given that text objects do not have titles, descriptions, or captions, they are not able to be picked up by screen-readers.

Solution: Any language in a text object needs to be copied and pasted into a separate worksheet; this language can then either be included in the title or the caption. Screen readers will typically read text that is in is an <h1>, <h2>, or <h3> through <h6> HTML tag. With text that has been put into sheets, screen readers will read additional phrases such as "Marks, access to the view data window is not enabled."

Accessibility and tableau objects

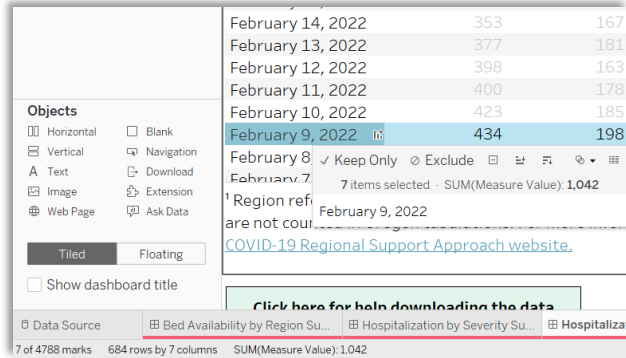
Limitation: Tableau provides a [list of dashboard elements](#) that they designate as inherently accessible and others that are not. Both maps and sliding date filters are considered not WCAG conformant because of their lack of keyboard support. With sliding date filters, they will not be picked up by the focus order and maps do not let the user directly download the data like other visualizations do.

Solution: Dashboard design should be adjusted to provide date filters in another format. For maps, data must be available for download from a summary table.

Number of marks (<1000)

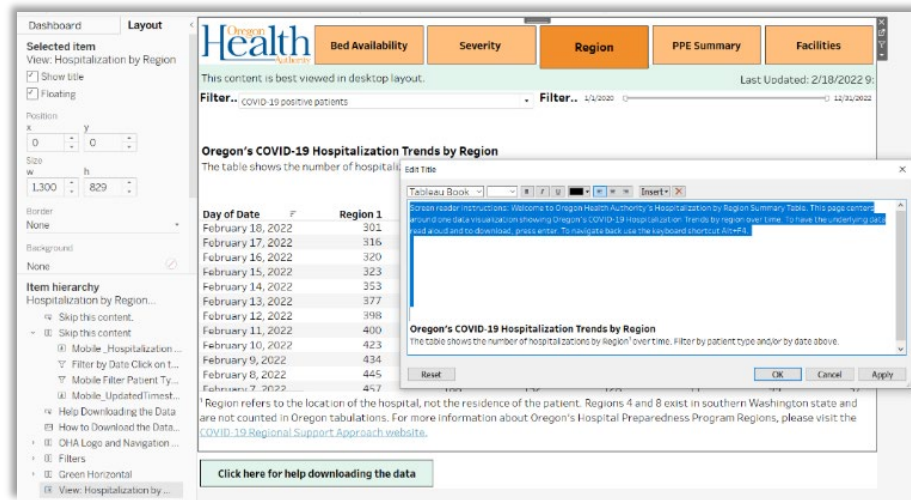
Limitation: Tableau, like other webpages, has a limit on the amount of information that can be displayed on a webpage before it impacts rendering. If the dashboard has over 1,000 marks, the visualizations are generated by the server, and it prevents sheets from using their HTML headings associated with their title and captions. That means that these sheets will not be read aloud unless they are a data visualization of their own. One option is to use a filter or aggregate the data to reduce the number of marks, but this not always possible.

Example 1: When clicking on a mark, Tableau will automatically display the total number of marks in the bottom left corner of the page. In the example below, note the total is 4,788 marks.

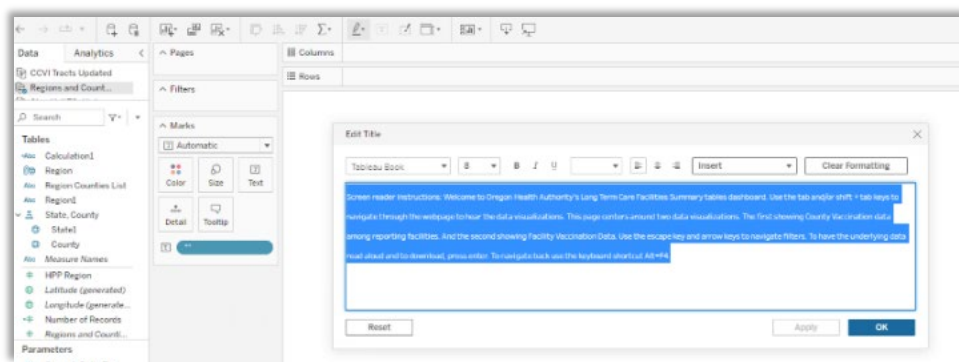


Example 2: Using filters on dashboards that include visualizations with over 1,000 marks does not address the issue because the filters change only the values that are displayed; the filters do not actually decrease the total number of marks. The sliding date filter reduces the mark count to fewer than 1,000, but this filter is not accessible to screen reader users. For data reported daily, it is difficult to meet this threshold.

Solution A: One solution extends the data visualization to the top of the page so that it will be the first object that is selected by screen readers. Add space to position the visualization where it should be on the dashboard, as seen below. Because the data is in its own visualization, it will be now read aloud even if there are more than 1,000 marks.



Solution B: A second solution is to add a blank calculated field (for example, using only quotation marks). The quotation marks will make a blank cell appear on the sheet. This will then allow for the sheet to be detected by screen readers, even when there are over 1,000 marks on a dashboard. Try using a Standard fit, which defaults to showing the calculated field, for the sheet first. Using other fit types – Entire View, Fit Width, and Fit Height – may require additional formatting to display the blank cell and be detected by the screen reader.



Version history

Version	Date	Notes
v1.0	August 2022	Created by: Shannon Allain (OHA), Adam McBride-Smith (OHA) David Gooft (Deloitte), Felipe Henao (Deloitte), Matthew Nicholas (Deloitte) Tableau version 2021.4 was used for the remediations that are described in this document.
v1.1	November 2022	Internal document (OHA 4353) adapted for partner sharing (OHA 4353_A).

To share your experience, ideas on how to make this better, please use the [Tableau Dashboard Accessibility Guide Feedback Form](#)

If you have any questions about accommodations or need any assistance to participate, please call or email: Shannon Allain at shannon.q.allain@dhsosha.state.or.us or 971-277-0990 (voice/text) TTY (711).