ARIA: progressbar role

- Description
  - All descendants are presentational
  - Associated WAI-ARIA roles, states, and properties
- Examples
- Best practices
  - Prefer HTML
- Specifications
- See also

The `progressbar` role defines an element that displays the progress status for tasks that take a long time.

Description

The `progressbar` range widget indicates that a request has been received and the application is making progress toward completing the requested action.

Authors **may** set aria-valuemin and aria-valuemax to indicate the minimum and maximum progress indicator values. Otherwise, their implicit values follow the same rules as HTML's `<input type="range">`:

- If `aria-valuemin` is missing or not a number, it defaults to `0` (zero).
- If `aria-valuemax` is missing or not a number, it defaults to `100`.
- The `aria-valuemin` and `aria-valuemax` properties only need to be set for the `progressbar` role when the progress bar's minimum is not `0` or the maximum value is not `100`.
- The read-only `aria-valuenow` should be provided and updated unless the value is `indeterminate`, in which case don't include the attribute. If set, make sure the `aria-valuenow` value is between the minimum and maximum values.

If the `progressbar` role is applied to an HTML progress element, the accessible name can come from the associated label. Otherwise use `aria-labelledby` if a visible label is present or `aria-label` if a visible label is not present.

All descendants are presentational

There are some types of user interface components that, when represented in a platform accessibility API, can only contain text. Accessibility APIs do not have a way of representing semantic elements contained in a `progressbar`. To deal with this limitation, browsers automatically apply role `presentation` to all descendant elements of any `progressbar` element as it is a role that does not support semantic children.

For example, consider the following `progressbar` element, which contains a heading.

```
<div role="progressbar"><h3>Title of my progressbar</h3></div>
```

Because descendants of `progressbar` are presentational, the following code is equivalent:

```
<div role="progressbar">
  <h3 role="presentation">Title of my progressbar</h3>
</div>
```

From the assistive technology user's perspective, the heading does not exist since the previous code snippets are equivalent to the following in the accessibility tree:

```
<div role="progressbar">Title of my progressbar</div>
```

Associated WAI-ARIA roles, states, and properties

- `aria-valuenow`
  - : Only present and required if the value is not indeterminate. Set to a decimal value between `0`, or `aria-valuemin` if present, and `aria-valuemax` indicating the current value of the progress bar.
- `aria-valuetext`
  - : Assistive technologies often present the value of `aria-valuenow` as a percentage. If this would not be accurate use this property to make the progress bar value understandable.
- `aria-valuemin`
  - : Set to a decimal value representing the minimum value, and less than `aria-valuemax`. If not present, the default value is `0`.
- `aria-valuemax`
  - : Set to a decimal value representing the maximum value, and greater than `aria-valuemin`. If not present, the default value is `100`.

- `aria-label` or `aria-labelledby`
  - : Defines the string value or identifies the element (or elements) that label the progressbar element providing an accessible name. An accessible name is required.

It is recommended to use a native progress or `<input type="range">` elements rather than the `progressbar` role. User agents provide a stylize widget for the progress element based on the current `value` as it relates to the `0`, the minimum value, and the `max` value. When using non-semantic elements, all features of the native semantic element need to be recreated with ARIA attributes, JavaScript and CSS.

Examples

In the example below, the progress bar uses the default values of 0 and 100 for `aria-valuemin` and `aria-valuemax`:

```
<div>
  <span id="loadinglabel">Loading:</span>
  <span role="progressbar" aria-labelledby="loadinglabel" aria-valuenow="23">
    <svg width="100" height="10">
      <rect height="10" width="100" stroke="black" fill="black" />
      <rect height="10" width="23" fill="white" />
    </svg>
  </span>
</div>
```

Using semantic HTML, this could be written as:

```
<label for="loadinglabel">Loading:</label>
<progress id="loadinglabel" max="100" value="23"></progress>
```

Best practices

If the progress bar is describing the loading progress of a particular region of a page, include the `aria-describedby` attribute to reference the progress bar's status, and set the `aria-busy` attribute to `true` on the region until it is finished loading.

Prefer HTML

It is recommended to use a native progress or `<input type="range">` elements rather than the `progressbar` role.

Specifications

See also

- HTML progress element
- Other range widgets include:
  - `meter`
  - `scrollbar`
  - `separator` (if focusable)
  - `slider`
  - `spinbutton`

□□□ #3
□ □□□ □□□ 2026-10-31 03:27:53 UTC
□ cocobook-sync □□□ 2026-11-04 17:33:21 UTC