

# Introducción a WAI-ARIA (2)

Esta lección pertenece al [curso “Aprende Accesibilidad Web Paso a Paso”](#) <sup>1</sup>

Prohibida su distribución sin permiso del autor.

## Presentación

Hola, soy Olga Carreras Montoto, consultora en Experiencia de Usuario y Accesibilidad web en [“Usable y accesible”](#) <sup>2</sup> (España).

Esta lección es la continuación de “Introducción a WAI-ARIA (1)” que te invito a que consultes previamente.

En la lección anterior vimos como la especificación del W3C, WAI-ARIA<sup>3</sup>, nos permite incluir información semántica sobre la estructura de la página y sobre los componentes de la interfaz, su comportamiento y la relación entre los mismos; de manera que esta información pueda ser transmitida a los usuarios que utilizan productos de apoyo, como un lector de pantalla, y facilitarles así la comprensión de la página y su interacción con la misma.

Aprendimos también qué son los roles y los diferentes tipos que existen:

- Roles que definen la estructura de la página (como los *landmark roles*).
  - Por ejemplo, `<div role="main">...</div>`

---

<sup>1</sup> <http://accesibilidadweb.es>

<sup>2</sup> “Usable y accesible”, <http://www.usableyaccesible.com>

<sup>3</sup> W3C, [Accessible Rich Internet Applications \(WAI-ARIA\) 1.0](http://www.w3.org/TR/wai-aria/), <http://www.w3.org/TR/wai-aria/>

- Roles que definen elementos de la interfaz (*widget roles*).
  - Por ejemplo, `<ul role="tree">...</ul>`

A continuación vamos a ver más ejemplos de roles y propiedades.

## Otros roles y propiedades

### Radiogroup

Podemos indicar si una serie de campos están relacionados entre sí, en este caso un grupo de radiobuttons, con el rol “radiogroup”:

```
<ul role="radiogroup">...</ul>
```

Podéis consultar un ejemplo en [“Example 31 - Radiogroup: ARIA CSS Selectors”](#)<sup>4</sup>.

### Aria-required

Podemos indicar si un campo es obligatorio con la propiedad “aria-required”:

```
<input type="text" name="username" id="username"
aria-required="true" />
```

En mi artículo sobre HTML5 y accesibilidad ([“HTML5 y accesibilidad: nuevos tipos de input, atributos asociados y validación nativa”](#)<sup>5</sup>) podéis consultar información sobre este atributo en relación con su homólogo de HTML 5.

---

<sup>4</sup> OAA, “Example 31 - Radiogroup: ARIA CSS Selectors”, <http://oaa-accessibility.org/example/31/>

<sup>5</sup> Carreras Montoto, Olga: “HTML5 y accesibilidad: nuevos tipos de input, atributos asociados y validación nativa”: <http://olgacarreras.blogspot.com.es/2014/03/html5-y-accesibilidad-nuevos-tipos-de.html>, 2014

## Aria-label

Podremos definir la etiqueta, el texto que identifica a un elemento, mediante la propiedad “aria-label”.

```
<div role="navigation" aria-label="Menú principal">...</div>
```

```
<div role="navigation" aria-label="Menú secundario">...</div>
```

Con la propiedad “aria-label” indicamos directamente el texto, por ejemplo para diferenciar dos zonas de menú diferentes. Ambas zonas tienen el mismo rol pero su diferente etiqueta permitirá que sean anunciadas como “Menú principal” o “Menú secundario”.

## Aria-labelledby

También tenemos una propiedad que nos permite etiquetar elementos indicando el id o ids de los elementos que actúan como etiqueta.

```
<p id="report-title">Informe de ventas 2012</p>
```

```
<ul>
```

```
<li><a aria-labelledby="report-title pdf" href="ventas.pdf" id="pdf">PDF</a></li>
```

```
<li><a aria-labelledby="report-title doc" href="ventas.doc" id="doc">Word</a></li>
```

```
</ul>
```

De esta manera, el enlace PDF del ejemplo anterior puede ser etiquetado y anunciado como “Informe de ventas 2012 PDF”.

## Aria-describedby

La propiedad “aria-describedby” nos permite indicar la descripción de un elemento indicando el id o ids del elemento o elementos que lo describen.

```
<input aria-describedby="ayuda" ... />
```

```
<p id="ayuda">La contraseña debe tener mínimo 6  
caracteres.</p>
```

Os invito a que leáis mi artículo [“Ayuda contextual de los formularios más accesible con ‘aria-describedby’ \(WAI-ARIA\)”](http://olgacarreras.blogspot.com.es/2013/12/ayuda-contextual-de-los-formularios-mas.html)<sup>6</sup> donde os explico con detalle cómo podemos asociar semánticamente un campo de formulario con su ayuda contextual para que sea anunciada a los usuarios de productos de apoyo cuando el campo coge el foco.

---

<sup>6</sup> Carreras Montoto, Olga: “Ayuda contextual de los formularios más accesible con “aria-describedby” (WAI-ARIA)”: <http://olgacarreras.blogspot.com.es/2013/12/ayuda-contextual-de-los-formularios-mas.html>, 2013

## Live Regions

Otro problema de accesibilidad en las páginas, que nos permite solucionar WAI-ARIA, son las zonas de la página que se modifican o actualizan automáticamente sin intervención del usuario, y que se llaman “Live Regions”.

Puede ser una zona en la cual se actualicen los resultados deportivos, por ejemplo cada vez que se marca un gol, puede ser un contador, un banner publicitario o diversas imágenes en la cabecera que va cambiando, las últimas noticias o mensajes de un chat, etc.

Los usuarios que utilizan productos de apoyo, como un lector de pantalla, no se percatarán del cambio o de la actualización a menos que el lector de pantalla se lo anuncie.

Para indicar que una zona de la página es viva, que se modifica o actualiza dinámicamente sin intervención del usuario, solo tenemos que incluir a esa zona la propiedad “aria-live”, que puede tener tres valores:

- `aria-live="off"`: los cambios solo se anunciarán al usuario cuando el foco esté en esa región. Será por tanto adecuado usarlo cuando la actualización no es relevante, por ejemplo una rotación de imágenes en la cabecera.
- `aria-live="polite"`: los cambios se anunciarán al usuario pero sin interrumpirle, anunciará el cambio cuando termine la tarea que está llevando a cabo, por ejemplo cuando termine de escribir en un campo o termine de leer un contenido.
- `aria-live="assertive"`: los cambios se anuncian de inmediato, independientemente de lo que el usuario esté haciendo. Teniendo en cuenta que el anuncio puede desorientar a los usuarios o que estos no terminen la tarea que están realizando, solo debe utilizarse para mensajes y

advertencias importantes, por ejemplo los mensajes de error de validación en un formulario.

Puedes ampliar información, ver las buenas prácticas para aplicar el atributo “aria-live” y otros asociados, o consultar ejemplos concretos, en mi artículo [“Live Regions y WAI-ARIA. Cómo mejorar la accesibilidad de contenidos que se actualizan automáticamente”](#)<sup>7</sup>.

En el documento [“WAI-ARIA 1.0 Primer”](#)<sup>8</sup>, podréis consultar los pasos y buenas prácticas para aplicar WAI-ARIA, que tenéis resumidos en mi artículo [“WAI-ARIA. Introducción, referencias, ejemplos, herramientas”](#)<sup>9</sup>, en cual, además, encontraréis enlaces a páginas con muchos ejemplos.

Por otra parte, en el documento [“Técnicas para las WCAG 2.0”](#)<sup>10</sup> de las WCAG 2.0 se recogen las técnicas que se puede emplear para cumplir con los criterios de conformidad de las WCAG 2.0, y allí encontraréis técnicas ARIA. Las resumí en el artículo [“Nuevas técnicas ARIA en las WCAG 2.0”](#)<sup>11</sup> que os invito a que leáis.

---

<sup>7</sup> Carreras Montoto, Olga: “Live Regions y WAI-ARIA. Cómo mejorar la accesibilidad de contenidos que se actualizan automáticamente”:

<http://olgacarreras.blogspot.com.es/2013/11/live-regions-y-wai-aria-como-mejorar-la.html>, 2013

<sup>8</sup> W3C, “WAI-ARIA 1.0 Primer”, <http://www.w3.org/TR/wai-aria-primer/>

<sup>9</sup> Carreras Montoto, Olga: “WAI-ARIA. Introducción, referencias, ejemplos, herramientas”: <http://olgacarreras.blogspot.com.es/2007/02/ajax-accesible-ii-wai-aria.html>, 2014

<sup>10</sup> W3C, “ARIA Techniques for WCAG 2.0”, <http://www.w3.org/TR/WCAG20-TECHS/aria.html>

<sup>11</sup> Carreras Montoto, Olga: “Nuevas técnicas ARIA en las WCAG 2.0. Novedades de la actualización del documento ‘Techniques for WCAG 2.0’ del 11 de marzo de 2014”: <http://olgacarreras.blogspot.com.es/2014/03/nuevas-tecnicas-aria-en-las-wcag-20.html>, 2014

## Autor

Olga Carreras Montoto. Consultora en Experiencia de Usuario y Accesibilidad Web en [“Usable y accesible”](#) (España)

Ya para terminar, estas son las direcciones de mi blog, de mi web y de mi cuenta de Twitter. En ellas puedes encontrar información sobre mí y sobre mi trabajo, así como más artículos sobre accesibilidad.

- Blog: <http://olgacarreras.blogspot.com.es/>
- Web: <http://www.usableyaccesible.com>
- Twitter: <https://twitter.com/olgacarreras>

## Referencias

- Carreras Montoto, Olga: [“WAI-ARIA. Introducción, referencias, ejemplos, herramientas”](#):  
<http://olgacarreras.blogspot.com.es/2007/02/ajax-accesible-ii-wai-aria.html>, 2014
- Carreras Montoto, Olga: [“Nuevas técnicas ARIA en las WCAG 2.0. Novedades de la actualización del documento Techniques for WCAG 2.0 del 11 de marzo de 2014”](#):  
<http://olgacarreras.blogspot.com.es/2014/03/nuevas-tecnicas-aria-en-las-wcag-20.html>, 2014
- Carreras Montoto, Olga: [“Navegación más accesible y semántica en 2 minutos con Landmark Roles \(WAI-ARIA\)”](#):  
<http://olgacarreras.blogspot.com.es/2014/03/navegacion-mas-accesible-y-semantica-en.html>, 2014

- Carreras Montoto, Olga: [“HTML5 y accesibilidad: nuevos tipos de input, atributos asociados y validación nativa”](http://olgacarreras.blogspot.com.es/2014/03/html5-y-accesibilidad-nuevos-tipos-de-input-atributos-asociados-y-validacion-nativa.html):  
http://olgacarreras.blogspot.com.es/2014/03/html5-y-accesibilidad-nuevos-tipos-de.html, 2014
- Carreras Montoto, Olga: [“Live Regions y WAI-ARIA. Cómo mejorar la accesibilidad de contenidos que se actualizan automáticamente”](http://olgacarreras.blogspot.com.es/2013/11/live-regions-y-wai-aria-como-mejorar-la.html):  
http://olgacarreras.blogspot.com.es/2013/11/live-regions-y-wai-aria-como-mejorar-la.html, 2013
- Carreras Montoto, Olga: [“Ayuda contextual de los formularios más accesible con "aria-describedby" \(WAI-ARIA\)”](http://olgacarreras.blogspot.com.es/2013/12/ayuda-contextual-de-los-formularios-mas-accesible-con-aria-describedby.html):  
http://olgacarreras.blogspot.com.es/2013/12/ayuda-contextual-de-los-formularios-mas.html, 2013
- iCITA, [“Tree View Example 1”](http://test.cita.illinois.edu/aria/tree/tree1.php),  
http://test.cita.illinois.edu/aria/tree/tree1.php
- OAA, [“Example 31 - Radiogroup: ARIA CSS Selectors”](http://oaa-accessibility.org/example/31/)  
http://oaa-accessibility.org/example/31/
- W3C, [“ARIA Techniques for WCAG 2.0”](http://www.w3.org/TR/WCAG20-TECHS/aria.html),  
http://www.w3.org/TR/WCAG20-TECHS/aria.html
- W3C, [“Accessible Rich Internet Applications \(WAI-ARIA\) 1.0 W3C Recommendation 20 March 2014”](http://www.w3.org/TR/wai-aria/),  
http://www.w3.org/TR/wai-aria/
- W3C, [“WAI-ARIA 1.0 Primer”](http://www.w3.org/TR/wai-aria-primer/), http://www.w3.org/TR/wai-aria-primer/
- W3C, [“WAI-ARIA Overview”](http://www.w3.org/WAI/intro/aria.php),  
http://www.w3.org/WAI/intro/aria.php
- W3C, [“WAI-ARIA Taxonomy”](http://www.w3.org/TR/wai-aria/rdf_model.html), http://www.w3.org/TR/wai-aria/rdf\_model.html



*Hacer la Web accesible puede mejorar dramáticamente las vidas de las personas y beneficiar a la sociedad en su conjunto. S.L.Henry*