

ACCESIBILIDAD WEB EN OBJETOS FLASH

**Centro de Referencia en Accesibilidad
y Estándares Web**

Copyright © 2007 Instituto Nacional de Tecnologías de la comunicación (INTECO)



El presente documento está bajo la licencia Creative Commons Reconocimiento-No comercial-Compartir Igual versión 2.5 España.

Usted es libre de:

- copiar, distribuir y comunicar públicamente la obra
- hacer obras derivadas

Bajo las condiciones siguientes:

- **Reconocimiento.** Debe reconocer los créditos de la obra de la manera especificada por el autor o el licenciadore (pero no de una manera que sugiera que tiene su apoyo o apoyan el uso que hace de su obra).
- **No comercial.** No puede utilizar esta obra para fines comerciales.
- **Compartir bajo la misma licencia.** Si altera o transforma esta obra, o genera una obra derivada, sólo puede distribuir la obra generada bajo una licencia idéntica a ésta.

Al reutilizar o distribuir la obra, tiene que dejar bien claro los términos de la licencia de esta obra.

Alguna de estas condiciones puede no aplicarse si se obtiene el permiso del titular de los derechos de autor

Nada en esta licencia menoscaba o restringe los derechos morales del autor.

Esto es un resumen legible por humanos del texto legal (la licencia completa) disponible en <http://creativecommons.org/licenses/by-nc-sa/2.5/es/>

El presente documento cumple con las condiciones de accesibilidad del formato PDF (Portable Document Format).

Se trata de un documento estructurado y etiquetado, provisto de alternativas a todo elemento no textual, marcado de idioma y orden de lectura adecuado.

Para ampliar información sobre la construcción de documentos PDF accesibles puede consultar la guía disponible en la sección [Accesibilidad > Formación > Manuales y Guías](#) de la página <http://www.inteco.es>.

ÍNDICE

ÍNDICE	3
1. INTRODUCCIÓN A LA ACCESIBILIDAD WEB EN FLASH	5
1.1. Usos de Flash	5
1.1.1. Funcionalidad y Beneficios de Flash	5
1.1.2. Usos Inapropiados	6
1.1.3. SVG: la alternativa estándar	7
1.2. Sobre la Accesibilidad de Flash	8
1.2.1. Precedentes	8
1.2.2. Requerimientos	8
1.2.3. Limitaciones Actuales	9
2. GENERAR CONTENIDO FLASH ACCESIBLE	12
2.1. Alternativa Obligatoria	12
2.2. Arquitectura de la Accesibilidad en Flash	14
2.2.1. Concepto Jerárquico de la Accesibilidad en Flash	14
2.2.2. Propiedades de Accesibilidad a través de ActionScript	16
2.2.3. Organizar un Objeto Flash Accesible	17
2.3. Contenido textual accesible	18
2.4. Símbolos accesibles: Gráficos, Botones y Clips de Película	20
2.4.1. Gráficos	20
2.4.2. Botones	20
2.4.3. Clips de Película	21
2.5. Proporcionar contexto	22
2.6. Orden de Lectura	24
2.7. Acceso y navegación con teclado	25
2.8. Control sobre animaciones	26
2.9. Componentes accesibles	28
2.10. Color y Contraste	29
2.10.1. Dependencias del color	29
2.10.2. Contraste	30
2.11. Control, Subtitulado y Transcripción para Audio y Vídeo	31
2.11.1. Control	31



Instituto Nacional
de Tecnologías
de la Comunicación

2.11.2.	Subtitulado	32
2.11.3.	Transcripción	34
3.	VALIDACIÓN DE LA ACCESIBILIDAD EN FLASH	35
3.1.	Alternativa Obligatoria	35
3.2.	Orden de Lectura y Orden de Tabulación	35
3.3.	Color y Contraste	36
3.3.1.	Color	36
3.3.2.	Contraste	37
	MATERIAL DE SOPORTE	39
	Guías de Accesibilidad en Flash	39
	Software	39
	Otras referencias	40

1. INTRODUCCIÓN A LA ACCESIBILIDAD WEB EN FLASH

Adobe Flash es una tecnología basada en gráficos vectorizados cuyo principal enfoque es el tratamiento de animaciones e implementación de entornos multimedia. Actualmente tiene soporte sobre diferentes aplicaciones y plataformas pero su mayor campo de acción es el de la **Web**. A través de este documento, se proporciona información acerca de cómo proporcionar un nivel de Accesibilidad adecuado para cualquier objeto Web basado en esta tecnología.

Antes de profundizar en los pasos necesarios, resulta imprescindible analizar el propósito para el que Flash fue diseñado, su funcionalidad y sus orígenes.

1.1. USOS DE FLASH

El objetivo de este apartado es el de ubicar el concepto de “*utilizar cada tecnología para lo que fue diseñada*” o lo que es lo mismo: despejar las dudas acerca de cuando utilizar Flash y cuando optar por otras tecnologías W3C¹ al diseñar y desarrollar un sitio Web.

1.1.1. Funcionalidad y Beneficios de Flash

Los sitios Web y objetos programados en Flash, proporcionan entornos multimedia capaces de ofrecer interacción y movimiento. El propósito para el que está diseñada esta herramienta de desarrollo, es la presentación y la interacción visual en el ámbito multimedia y resulta realmente potente en dichos objetivos, pero no es la herramienta ideal para generar documentos Web completos o maquetar un portal. Es en este aspecto donde el desarrollador debe tomar decisiones: no es lo mismo construir el sitio Web para el catálogo de una biblioteca que hacerlo para la promoción de una película. En el primer caso, deben primar los contenidos estructurados y una funcionalidad útil e intuitiva mientras que, en el segundo, el grafismo y el apartado multimedia pueden llegar a tener más protagonismo. En cualquier caso, tendrá difícil justificación una Web desarrollada íntegramente en Flash (más información en la sección [Usos inapropiados](#)).

La capacidad que ofrece Flash es siempre a un nivel de interacción muy específico: aquel donde prime el apartado visual y el movimiento o en el que la información no pueda ser presentada por otras tecnologías Web.

Si bien es cierto que en cada nueva versión del software de desarrollo de Flash se ofrecen potentes prestaciones para construir sitios Web y ofrecer una funcionalidad similar a la de los documentos estructurados (*HTML*², *XHTML*³, etc), su aun limitado soporte para la

¹ W3C: World Wide Consortium (<http://w3.org>)

² HTML: Hyper-text Transfer Protocol

³ XHTML: Extended Hypert-text Transfer Protocol

accesibilidad y su funcionalidad exclusiva sobre determinados sistemas operativos y agentes de usuario hacen que, de momento, el desarrollador deba plantearse en cada momento una serie de cuestiones: *¿Necesito utilizar un objeto Flash para esta sección en concreto? ¿Hay alguna tecnología estándar que me permita presentar la información de manera similar? ¿Puedo garantizar la accesibilidad de los objetos programados?...* En función de esas respuestas y con lo expuesto anteriormente, es relativamente fácil decidir si un elemento en Flash es beneficioso o puede llegar a suponer un obstáculo.

En ningún caso se trata de evitar el uso de esta tecnología, si no de todo lo contrario: usarla con todas sus prestaciones y capacidades para los casos en los que suponga una mejora evidente de los contenidos del sitio. Es decir, cuestionarse si es la mejor opción de entre todas las tecnologías Web disponibles: sin llegar a discutir sus beneficios y su potencial, Flash fue diseñado como herramienta de diseño y animación de gráficos vectorizados y su uso en una Web en la que se desee alcanzar un nivel de accesibilidad adecuado, debe estar siempre justificado.

1.1.2. Usos Inapropiados

En función de lo expuesto anteriormente, un objeto programado en Flash puede estar siendo utilizado de manera poco apropiada en un Sitio Web. No se trata de generalizar ni de aplicar fórmulas matemáticas a los malos usos de la tecnología pero sí hay una serie de aspectos a tener en cuenta:

- **Sitio Web 100% en Flash:** Es habitual encontrarse sitios Web en los que prima el apartado visual que implementan toda su funcionalidad y contenidos sobre esta plataforma. Aunque en el mejor de los casos se aporte una alternativa equivalente en formato HTML o XHTML (ver sección de [Alternativa Obligatoria](#)), nunca es aconsejable este tipo de abusos de la tecnología ya que se limita un acceso globalizado a los contenidos. Es necesario recordar siempre la importancia de los estándares frente a otras tecnologías por lo que, un sitio enteramente en Flash, no es la mejor de las opciones.
- **Contenido Web Estático:** como norma general, no es buena práctica utilizar Flash para contenido Web estático ya que para ese tipo de funcionalidad y presentación la solución óptima es el uso de documentos estructurados (*HTML*, *XHTML*) y hojas de estilo en cascada (*CSS*⁴, *XSL*⁵). El uso de estas tecnologías, potencia la accesibilidad de los contenidos y proporciona soporte para cualquier entorno y agente de usuario.

⁴ **CSS:** Cascading Style Sheets

⁵ **XSL:** The Extensible Stylesheet Language Family

- **Navegación e Interacción:** Fundamentar la navegación e interacción de un sitio Web en tecnología Flash, puede llegar a suponer un problema. Pese a que el objeto Flash destinado a tal efecto contenga las alternativas textuales a cada enlace y elemento del menú, esta tecnología no aporta el sentido lógico que puede llegar a ofrecer actualmente un documento estructurado (a través de una lista de enumeración, por ejemplo).

1.1.3. SVG: la alternativa estándar

Siempre que sea posible, lo óptimo para fomentar la accesibilidad de los contenidos es ofrecer tecnologías estándar.

En el campo de los gráficos vectorizados, el **W3C** recomienda la tecnología **SVG**⁶, que muestra de algunas de las siguientes ventajas:

- Fácilmente *editable* (el código fuente es **XML** y **CSS**).
- Pueden hacerse búsquedas en el código del gráfico.
- Los textos del gráfico pueden presentarse en el idioma preferido del usuario, de manera sencilla.
- Puede reutilizarse una hoja de estilos **CSS** para varios gráficos.
- Es un estándar abierto con implementaciones distintas y extensibles.
- Se le pueden aplicar efectos típicos de las imágenes *bitmap* a imágenes vectoriales (rellenos degradados, efectos, etc).
- Pueden generarse gráficos automáticamente, transformando el código **XML**.

Las **limitaciones** actuales de la tecnología **SVG** son su aún reducido soporte en algunos agentes de usuario (sigue siendo necesario el uso de *plug-ins*) y su incapacidad para incluir directamente elementos multimedia como vídeos o sonidos (característica con la que sí cuentan los entornos *Flash*).

⁶ **SVG**: Scalable Vector Graphics

1.2. SOBRE LA ACCESIBILIDAD DE FLASH

1.2.1. Precedentes

La primera versión de Flash se remonta a 1996, cuando *Macromedia* comercializa la conversión y adaptación del programa **Future Splash Animator**⁷ en colaboración con sus creadores: la compañía *FutureWaves*.

Durante los últimos años de la década pasada, *Macromedia* sigue comercializando sucesivas versiones de Flash que van introduciendo mejoras y funcionalidad a la herramienta pero no será hasta la **versión 6.0** cuando se implemente el soporte para **Accesibilidad**. A partir de esa versión, y ya de manos de la compañía *Adobe*⁸, se siguen implementando mejoras del motor de Flash en cuanto a Accesibilidad Web.

1.2.2. Requerimientos

Para que un desarrollador pueda generar contenido accesible a través de Flash, o un usuario pueda beneficiarse de la Accesibilidad de dichos contenidos, existen una serie de **requerimientos**⁹:

- **Flash Player 6 o superior**
- **Windows 98, 2000, XP o Vista**
- **Microsoft Internet Explorer 5 o superior**
- **Un lector de pantalla:**
 - GW Micro Window Eyes 4.2 o superior
 - Freedom Scientific JAWS 4.5, 6.1 o superior
 - IBM Home Page Reader 3.04 o superior
 - Dolphin HAL 6.50
 - KDS PC Talker (Japonés)

⁷ Programa de diseño y animación 2-D orientado a la Web y basado en el primitivo programa de dibujo vectorial **SmartSketch**.

⁸ Adobe absorbió la compañía *Macromedia* el 3 de Diciembre de 2005.

⁹ Fuente:

http://www.adobe.com/resources/accessibility/best_practices/best_practices_acc_flash.pdf

1.2.3. Limitaciones Actuales

A partir de los [Requerimientos](#) y lo visto hasta el momento, se puede establecer, de manera específica, las **limitaciones** actuales **para generar contenido accesible en objetos Flash**:

1.2.3.1. Limitaciones del entorno de usuario

- Sistema Operativo. El usuario que desee tener acceso a la información de un objeto Flash a través de software de soporte de Accesibilidad (un lector de pantalla, por ejemplo) deberá contar con un sistema operativo **Microsoft Windows**. Limitar el acceso a la información en función de una única plataforma o entorno de usuario supone una barrera significativa para muchos individuos.
- Navegador. La accesibilidad en los objetos Flash sólo está disponible a través del navegador **Microsoft Internet Explorer**. En otros navegadores, como es el caso de **Mozilla Firefox**, no hay soporte para la Accesibilidad de estos objetos. Esta dependencia del agente de usuario creado por Microsoft, viene dada por la tecnología **Microsoft Active Accesibility** (MSAA¹⁰) y su incompatibilidad con el motor actual de la mayoría de los navegadores. **MSAA** es la única vía que tiene Flash, por el momento, para ofrecer compatibilidad con lectores de pantalla y otras ayudas técnicas.

¹⁰ Tecnología basada en **Component Object Model (COM)** de Microsoft que permite a las aplicaciones de Windows poder utilizar herramientas de soporte de accesibilidad (lectores de pantalla, por ejemplo).

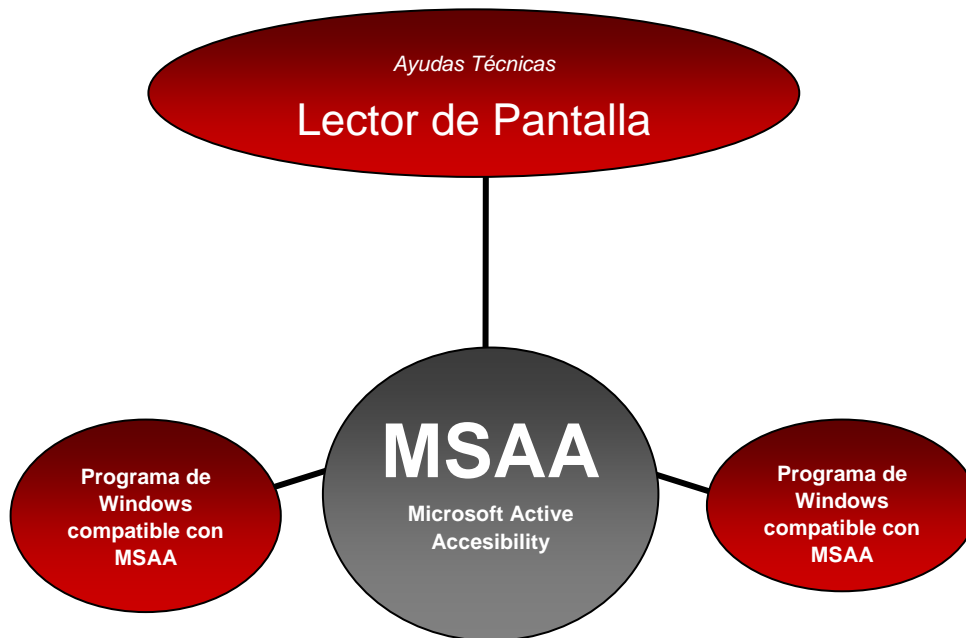


Figura 1. Esquema de dependencias de la tecnología MSAA (Microsoft Active Accessibility).

Además, en las últimas versiones de **Microsoft Internet Explorer**, para poder tabular a través de los contenidos de un objeto Flash (o cualquier otro objeto incrustado) este se debe activar antes a través de la barra espaciadora o haciendo *click* sobre el mismo, lo que limita aún más el acceso a la información (lectores de pantalla como JAWS no advierten al usuario de este tipo de peculiaridades sobre el control de estos contenidos):

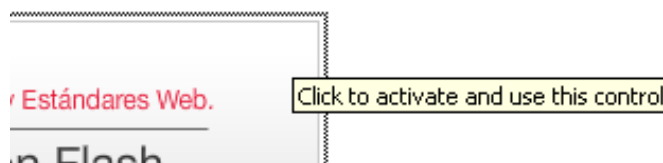


Figura 2. Detalle del tip mostrado por Internet Explorer 7 para activar el control de los objetos Flash.

- Flash Player o plug-in de navegador: Al no formar parte de los estándares, para acceder a un elemento o película Flash se necesita un entorno propio. Esta rutina o *plug-in* debe instalarse en la máquina para que el agente de usuario tenga soporte para los objetos incrustados en la Web.

1.2.3.2. Limitaciones propias de la tecnología Flash

- Contenido basado en animación y gráficos vectorizados. Pese a que los objetos Flash son capaces de incluir textos accesibles por las ayudas técnicas, la propia herramienta de desarrollo de Flash está enfocada al diseño gráfico vectorizado y los entornos multimedia y, por tanto, resulta complejo dotar a todas las capas de contenido de las alternativas equivalentes.
- Flash como programa propietario. Siendo un programa propietario, Flash se aleja de la máxima de usar siempre tecnologías *al alcance de cualquier usuario*. Y es que pese a que *Flash Player* es una aplicación o *plug-in* gratuito, fomentar el uso de Flash en la Web de forma indiscriminada frente a los estándares, reduce el campo de acción de cualquier desarrollador e impide globalizar el contenido Web. Por eso, lo oportuno es buscar siempre alternativas gratuitas y estándar que garanticen el acceso a los contenidos desde cualquier plataforma y entorno.
- Profundidad de los elementos de película. Cada objeto o película en Flash, está [estructurado de forma jerárquica](#), en función del tiempo y con una base de programación orientada a objetos llamada *ActionScript*. Esta abstracción de los bloques de información dificulta que principios tan valorados de accesibilidad como el orden de lectura de la información, o la navegación a través de ayudas técnicas, puedan ser aplicados con solvencia.

1.2.3.3. Limitaciones del entorno de desarrollo

- No hay soporte para accesibilidad. Desde el propio entorno de desarrollo de Flash no hay posibilidad de comprobar la accesibilidad de los contenidos que se implementan (a través **Control – Probar Película**). Esto obliga al desarrollador a compilar el fichero SWF y a generar un fichero HTML para realizar las [Pruebas de Accesibilidad](#) desde Internet Explorer.

2. GENERAR CONTENIDO FLASH ACCESIBLE

2.1. ALTERNATIVA OBLIGATORIA

Independientemente del nivel de accesibilidad del propio objeto Flash, este debe llevar asociada siempre una alternativa en formato textual estructurado (HTML o XHTML) equivalente en funcionalidad y contenido.

El objetivo de dicha alternativa es dar soporte para todos aquellos usuarios que no tengan acceso a los contenidos del objeto bien sea por [limitaciones del entorno de usuario](#), o bien por las [limitaciones propias de la tecnología](#).

Teniendo en cuenta que en la Web un objeto Flash estará incrustado en los propios documentos estructurados (HTML o XHTML) o enlazado directamente (como archivo **SWF**¹¹ independiente), la alternativa podrá ser presentada de dos maneras distintas: **incluida en el propio documento** o **presentada en un documento externo**.

Imaginemos que tenemos un *banner* realizado en Flash, animado con tres secuencias que representan, a su vez, tres objetivos:



Figura 3. Primera secuencia de Animación: "Tecnología próxima a la Sociedad"



Figura 4. Segunda secuencia de Animación: "Tecnología al alcance de los Ciudadanos"



Figura 5. Tercera secuencia de Animación: "Desarrollo de proyectos Tecnológicos"

Para aportar la alternativa necesaria, deberemos incluir la información estructurada pertinente dentro de la etiqueta **<object>** (utilizada por el *plug-in* de Flash Player para

¹¹ **SWF**: Ficheros compilados de Flash

mostrar la animación compilada en SWF). Como en este caso se trata de identificar tres frases que identifican tres objetivos de actuación, se pueden estructurar como lista desordenada en HTML (elemento `` y ``):

```
<object type="application/x-shockwave-flash" data="flash.swf" id="flash">

  <param name="allowScriptAccess" value="sameDomain" />
  <param name="movie" value="flash.swf" />
  <param name="quality" value="high" />
  <param name="bgcolor" value="#FFF" />
  Objetivos de actuación:
  <ul>
    <li>Tecnología próxima a la sociedad.</li>
    <li>Tecnología al alcance de los ciudadanos.</li>
    <li>Desarrollo de proyectos Tecnológicos.</li>
  </ul>
</object>
```

Nota importante: Para que el Flash degrade en su alternativa correctamente, debe eliminarse la línea de código correspondiente a la etiqueta `embed` generada al publicar el objeto en HTML.

En caso de que la información de la alternativa sea demasiado extensa o se desee dividir la información en varios documentos, se puede hacer una llamada a un documento externo que proporcione contenido equivalente:

```
<object type="application/x-shockwave-flash" data="flash.swf" id="flash">

  <param name="allowScriptAccess" value="sameDomain" />
  <param name="movie" value="flash.swf" />
  <param name="quality" value="high" />
  <param name="bgcolor" value="#FFF" />
  <a href="info_ampliada.html">Información sobre los Objetivos</a>
</object>
```

En este caso, se debe facilitar en la página de destino un enlace que permita volver al flujo natural de navegación.

Pese a que no es considerado una buena práctica, en el caso de que sea necesario enlazar con el fichero en formato Flash (SWF) de manera independiente a la estructura lógica de navegación se debe aportar **siempre** una fuente de información alternativa:

```
<ul>
  <li><a href="flash.swf">Objetivos (Objeto Flash SWF)</a></li>
  <li><a href="objetivos.html">Objetivos (HTML Accesible)</a></li>
</ul>
```

2.2. ARQUITECTURA DE LA ACCESIBILIDAD EN FLASH

El objetivo de este apartado es establecer las bases para construir un objeto Flash accesible:

2.2.1. Concepto Jerárquico de la Accesibilidad en Flash

La estructura de un objeto Flash se basa en jerarquías de capas y niveles de abstracción (objetos gráficos, variables, clips de película, etc). Las características de **Accesibilidad** siguen este modelo de Herencia de modo que pueden establecerse de manera específica a cada uno de los niveles. En el ejemplo siguiente, se muestra un objeto Flash sencillo con un elemento gráfico y un clip de película, que a su vez contienen un componente de tipo botón:

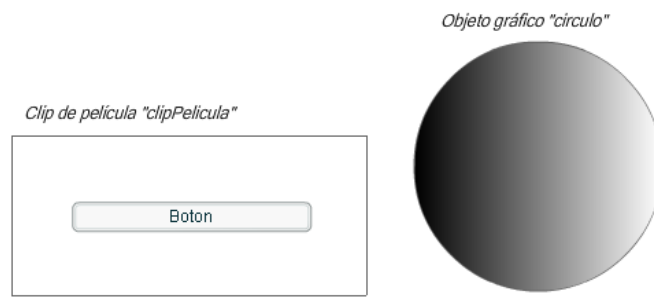
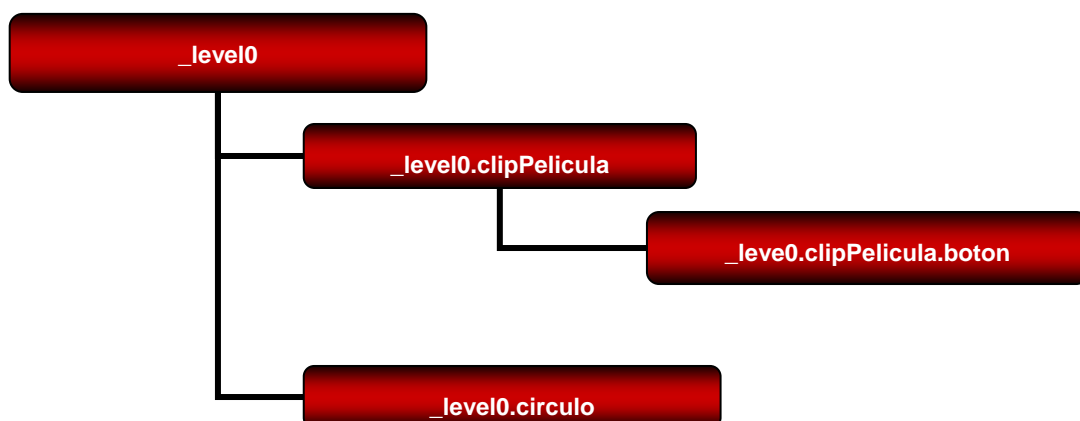


Figura 6. Ejemplo de Objeto Flash Sencillo

Teniendo en cuenta que los objetos están en el primer **frame** (o fotograma) de la línea de tiempo y que a cada una de las instancias se les ha asignado los nombres clipPelícula, boton y circulo, respectivamente, se podría establecer la siguiente estructura lógica a partir de la instancia raíz (_level0 ó _root):



A la hora de proporcionar a cada objeto del árbol de contenidos propiedades de Accesibilidad propias, el entorno de desarrollo¹² facilita un cuadro diálogo específico (menú **Ventana – Otros paneles – Accesibilidad**):

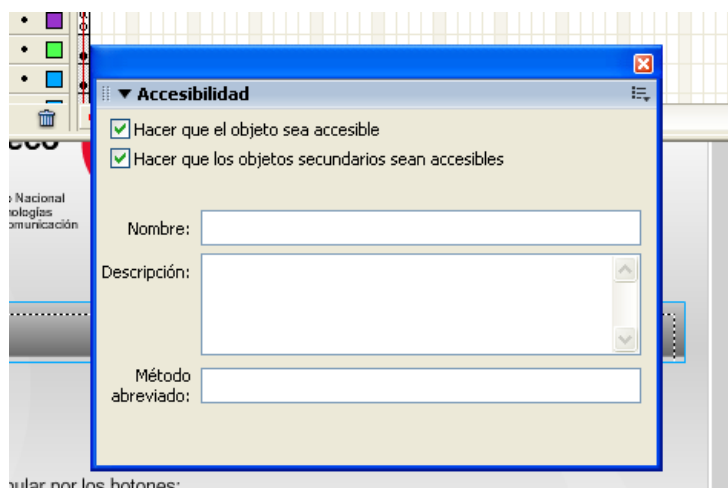


Figura 7. Propiedades de Accesibilidad para un clip de Película

A través de esta herramienta, se puede definir en mayor o menor profundidad el nivel de accesibilidad de cada instancia y objeto del escenario (salvo para Formas, Símbolos Gráficos y Textos estáticos):

- Con la propiedad **Hacer que el objeto sea accesible** (disponible para Clips de Película, Botones y Textos Dinámicos y de Introducción de datos), se puede activar o desactivar la accesibilidad para un objeto determinado. Esto puede resultar útil para determinadas instancias de película que tengan un papel meramente decorativo y no aporten información útil al usuario.
- A través de la opción **Hacer que los objetos secundarios sean accesibles** (disponible sólo para Clips de Película), se puede delimitar el acceso a los contenidos para las instancias dependientes del objeto *MovieClip*. En el ejemplo anterior, si se desmarcara esta opción para el objeto “clipPelícula” (*_level0.clipPelícula*), el objeto “boton” (*_level0.clipPelícula.boton*) dejaría de ser accesible.
- Los campos **Nombre**, **Descripción** y **Método Abreviado** permiten asociar información textual alternativa y funcionalidad accesible:

¹² Característica disponible a partir de la Versión 6 de Flash: ver [Precedentes](#).

- **Nombre** (disponible para Clips de Película, Botones y Textos de Introducción de datos). Aporta información concreta sobre el objeto. El texto de este campo debe ser breve y conciso.
- **Descripción** (disponible para Clips de Película, Botones y Textos Dinámicos y de Introducción de datos). Proporciona información extendida del objeto. Por ejemplo, en el caso de un botón, puede aportar un texto explicativo acerca de su funcionalidad.
- **Método Abreviado** (disponible para Clips de Película, Botones y Textos de Introducción de datos). Permite asociar scripts de escucha o *Listeners* a determinados atajos del teclado.

2.2.2. Propiedades de Accesibilidad a través de ActionScript

Para definir el nivel de accesibilidad de cada instancia de contenido, de manera independiente al cuadro de diálogo habilitado para tal efecto, puede utilizarse el entorno de programación *ActionScript*. Existen una serie de propiedades que equivalen a los parámetros de dicho panel de Accesibilidad:

Propiedad	Tipo	Equivalencia en Panel de Accesibilidad	Aplica a...
.silent	<i>booleano</i>	Hacer que el objeto/película sea accesible (lógica inversa)	<i>Película principal</i> <i>Botones</i> <i>Clips de Película</i> <i>Texto Dinámico</i> <i>Texto de Entrada</i>
.forcesimple	<i>booleano</i>	Hacer que los objetos secundarios sean accesibles (lógica inversa)	<i>Película principal</i> <i>Clips de Película</i>
.name	<i>cadena</i>	Nombre	<i>Película principal</i> <i>Botones</i> <i>Clips de Película</i> <i>Texto de Entrada</i>
.description	<i>cadena</i>	Descripción	<i>Película principal</i> <i>Botones</i> <i>Clips de Película</i> <i>Texto Dinámico</i> <i>Texto de Entrada</i>

Ejemplo: un clip de película en el que se muestre la animación de una luna orbitando un planeta (definido con el nombre de instancia *orbitaClip*), podrían definirse las opciones de accesibilidad:

```
_root.orbitaClip._accProps = new Object();  
_root.orbitaClip._accProps.name = "Luna orbitando un planeta";  
_root.orbitaClip._accProps.forcesimple = true;  
Accessibility.updateProperties();
```

En este caso, a parte de proporcionar un **nombre** para el objeto, se está aplicando la lógica inversa sobre la propiedad “**Hacer que los objetos secundarios sean accesibles**” (*forcesimple=true*) por lo que cualquier objeto o símbolo incluido dentro de este clip de película será no accesible. Cada vez que se modifiquen las propiedades de accesibilidad de un objeto se debe llamar a la función `Accessibility.updateProperties()` para que se actualicen los comportamientos del clip de película en tiempo real.

2.2.3. Organizar un Objeto Flash Accesible

Estructurar un objeto Flash con propiedad, ayudará a conseguir un nivel de Accesibilidad adecuado.

En primer lugar, se debe configurar la accesibilidad para el **Clip de Película Principal** (*_root, _level0*). Al igual que sucede para cualquier clip de película, para la instancia raíz se deben especificar una serie de parámetros (para mostrar el cuadro de diálogo en el entorno de desarrollo, bastará con hacer *click* fuera de la de la región de **Escenario**¹³):

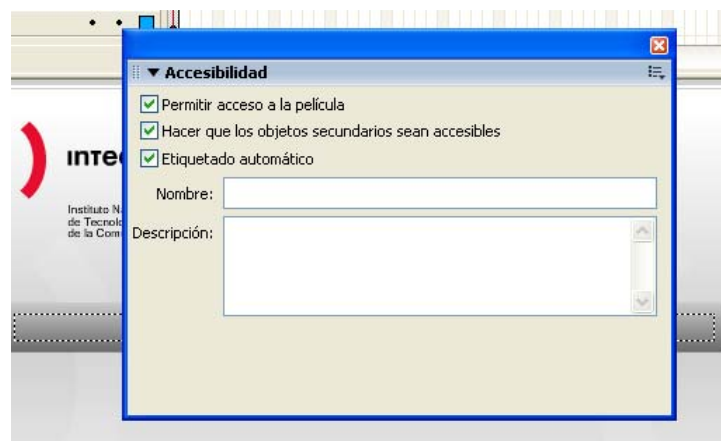


Figura 8. Propiedades de Accesibilidad para Clip de Película Principal (*_root* o *_level0*)

La única característica que difiere de cualquier otra instancia de película, es la propiedad **Etiquetado automático**. Con esta opción marcada, se provoca que en botones y clips de

¹³ Región definida para la ubicación de las instancias de objeto.

película que actúen como botón, se tomen los objetos de texto como **etiqueta**. Si se desmarca, se debe asignar la etiqueta a cada uno de estos elementos manualmente.

Una vez definidas las propiedades para la película principal o instancia raíz, el desarrollador debe plantearse la necesidad de organizar el contenido de forma que se adapte a cualquier usuario independientemente de su condición. La clave para conseguir una estructura accesible, es separar los objetos con Contenido Textual de los objetos de Presentación (animaciones, gráficos) para posteriormente asociar sus características de manera independiente y según sea necesario.

2.3. CONTENIDO TEXTUAL ACCESIBLE

Todos los objetos textuales incluidos en un *clip* (ya sean estáticos, dinámicos o de introducción de texto) son Accesibles a través de cualquier ayuda técnica, según la propia arquitectura de la accesibilidad en Flash¹⁴. A pesar de ello, se debe comprobar que cada instancia de película o de gráfico que contenga información textual, sea siempre accesible. En la fase de desarrollo se deberá comprobar para cada bloque estructural a través del cuadro de diálogo de Accesibilidad: “*Hacer que el objeto sea accesible*” y “*Hacer que los objetos secundarios sean accesibles*” (detalladas en el [Concepto Jerárquico de la Accesibilidad Flash](#)) deben estar activas en función de la estructura de la interfaz y de los contenidos que se esté utilizando.

Es importante en este punto dejar constancia de cómo las ayudas técnicas (como lectores de pantalla) acceden a la información textual. Dado que no se trata de un documento estructurado, cada bloque de texto que se incluya se tomará como la parte de un todo; de tal forma que si no se separa la información a través de signos de puntuación o separaciones lógicas entre cada unidad lingüística, se interpretará sin pausa alguna.

Valdría como ejemplo un *banner*: tres objetos textuales dentro de una misma instancia representando una sola frase:

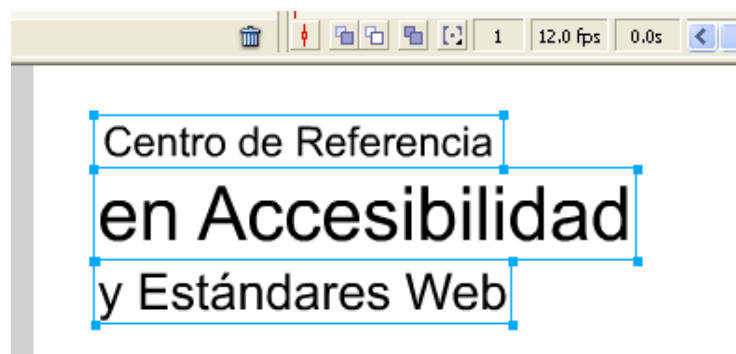


Figura 9. Tres objetos textuales que completan una sola frase

¹⁴ Siempre teniendo en cuenta que la Accesibilidad está disponible desde la **versión 6** del software de desarrollo de Flash. Ver sección: Precedentes.

Pese a que visualmente puede parecer evidente que se trata de una frase con 8 palabras, para un software de lectura de pantalla, la información literal es la siguiente: “Centro de Referencia en Accesibilidad Estándares Web” (se ha perdido parte de su valor semántico por no incluir los elementos de espaciado). Para estos casos, si no es posible incluir la frase completa en un solo objeto textual (por que se vaya a animar cada línea, por ejemplo) al menos debe asegurarse que no se pierde su significado introduciendo un espacio al final de cada línea.

Si por la complejidad de la animación de cada bloque textual se perdiera el orden lógico de lectura, deberá plantearse lo que se proponía en la sección [Organizar un Objeto Flash Accesible](#): dividir el contenido textual de la parte meramente presentacional o visual. Para el ejemplo puesto en marcha, podría ocurrir:

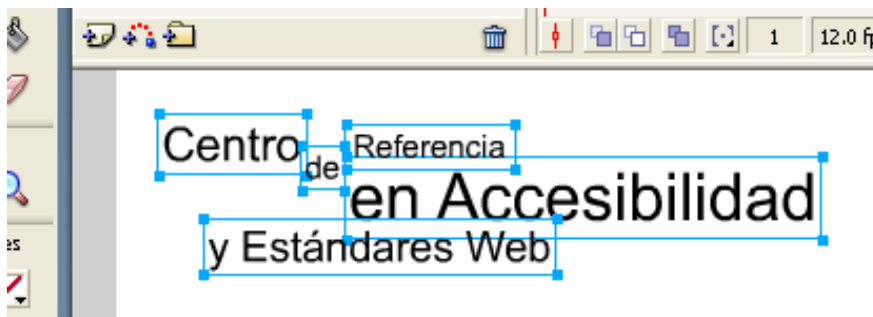


Figura 10. Bloques de texto animados que pierden el orden lógico de lectura

Como primer paso de la solución podría crearse un clip de película con la animación de los bloques de texto y **desmarcar las opciones de accesibilidad**:

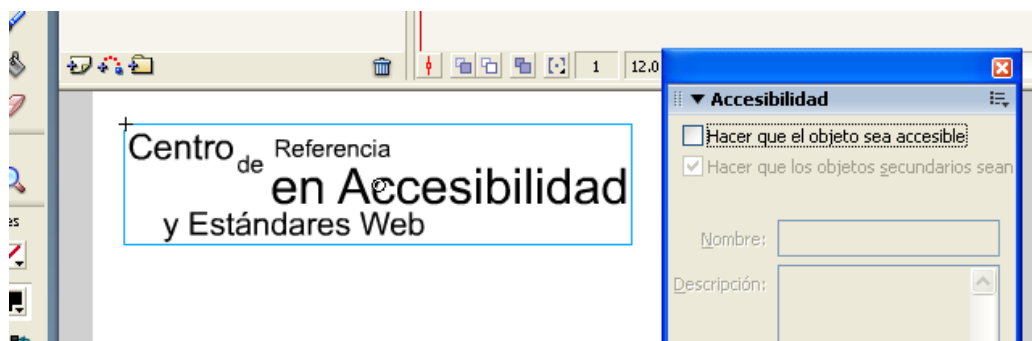


Figura 11. Clip de Película con la animación de los bloques de texto

De esta forma se oculta el bloque de información ilegible de cara a las ayudas técnicas. Sólo faltaría ofrecer la información redundante acompañando al nuevo objeto creado: bien de forma sutil (aportando un texto redundante bien formado pero de poca relevancia visual) o bien utilizando alguna técnica de de ocultación (utilizando un color frontal del texto idéntico al color de fondo o ubicando objetos fuera del escenario, por ejemplo):

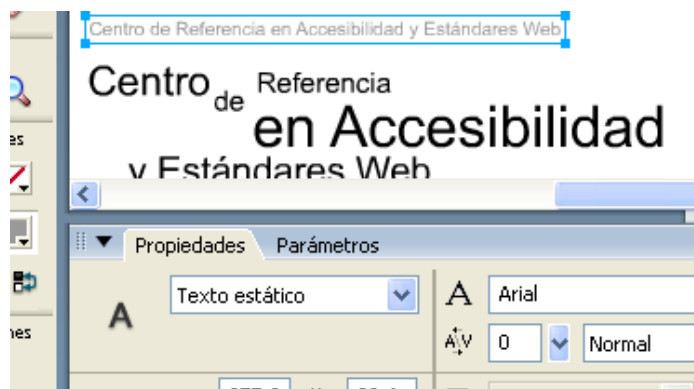


Figura 12. Bloque de texto con información redundante acompañando al clip no accesible.

Utilizando este tipo de soluciones, se asegura que no haya pérdidas de información o problemas de legibilidad. Se deberá comprobar siempre (a través de las ayudas técnicas) que los bloques de texto de la película son interpretados de manera idéntica por cualquier individuo.

Aportando todos los signos de puntuación necesarios en los objetos textuales y especificando las propiedades de Accesibilidad para los clips de película **que los incluyan, se evitarán pérdidas de significado en los contenidos.**

2.4. SÍMBOLOS ACCESIBLES: GRÁFICOS, BOTONES Y CLIPS DE PELÍCULA

Los objetos o instancias en Flash están divididos en tres tipologías de contenido llamadas **símbolos**: estas pueden ser **Gráficos**, **Botones** o **Clips de Película**.

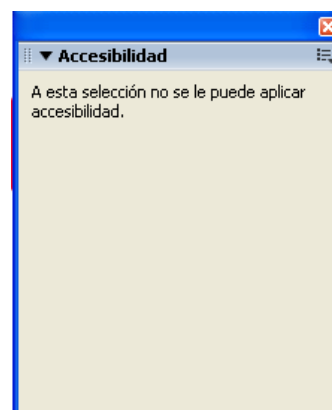
2.4.1. Gráficos

Los símbolos gráficos, no poseen opciones de Accesibilidad. Cualquier instancia definida como tal, será identificada por la plataforma como elemento presentacional y sin información relevante.

Por tanto, los símbolos gráficos sólo deben ser creados cuando no se trate con contenido textual relevante o no se proporcionen mecanismos de interacción.

2.4.2. Botones

Las instancias definidas como botón tienen características peculiares: poseen internamente cuatro estados identificados como fotogramas (*Reposo*, *Presionado*, *Sobre* y *Zona Activa*) y **actúan como elemento de interacción**. Por ello, disponen de opciones de accesibilidad:



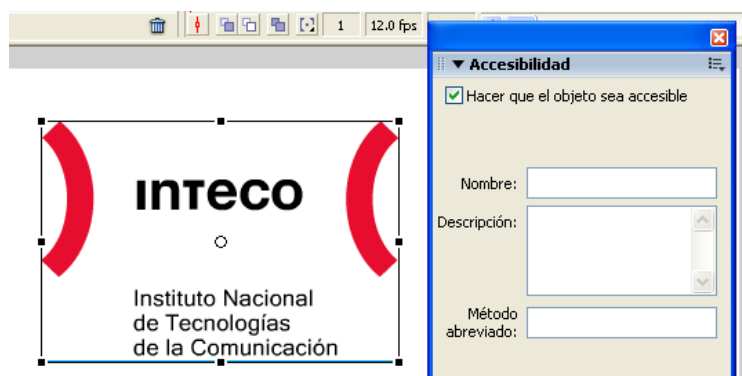


Figura 13. Opciones de Accesibilidad para un símbolo de botón

Si el símbolo no posee un bloque de texto, se deberá especificar un valor para el campo **Nombre**. En el ejemplo, si el logotipo fuera una *mapa de bit* (o una imagen plana; sin contenido textual), un texto apropiado para el Nombre sería “INTECO: Instituto Nacional de Tecnologías de la Comunicación”.

Si por el contrario, el símbolo contiene algún elemento textual, no es necesario definir la etiqueta del elemento ya que el propio motor asume el texto como etiqueta del control.

Para el ejemplo anterior, se pueden definir las propiedades de Accesibilidad a través del lenguaje de *ActionScript* al igual que lo hacemos a través del entorno gráfico:

```
_root.logoInteco._accProps = new Object();  
_root.logoInteco._accProps.name = "INTECO: Instituto Nacional de Tecnologías de la  
Comunicación";  
_root.logoInteco._accProps.forcesimple = true;  
Accessibility.updateProperties();
```

2.4.3. Clips de Película

Los símbolos de Película son los objetos más complejos ya que actúan de contenedores de otros símbolos siguiendo una [estructura jerárquica](#). En función de si los contenidos que incluye el símbolo son unidades de información importantes o forman parte de un elemento presentacional que no aporta datos relevantes, se deberá marcar o desmarcar la opción “**Hacer que el objeto sea accesible**”.

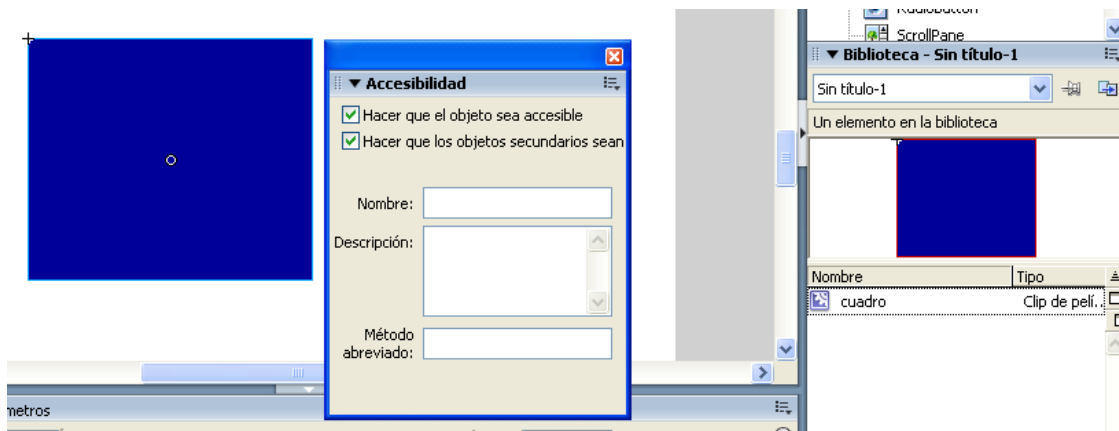


Figura 14. Detalle de creación de un clip de Película

Si el clip de película incluye símbolos sobre los que se deben aplicar a su vez opciones de accesibilidad, la opción “**Hacer que los objetos secundarios sean accesibles**” debe estar activada.

2.5. PROPORCIONAR CONTEXTO

Los elementos y símbolos de un objeto Flash, no sólo deben ser accesibles en sí mismos sino que debe aportarse toda la información necesaria sobre el contexto, estado y entorno en el que se encuentran.

La mejor forma de comprender este principio es mediante la observación de un caso práctico:

Generalmente, el control del volumen en cualquier entorno multimedia suele estar representado por una barra (horizontal o vertical) que permite, por medio de un desplazamiento en las dos direcciones posibles, aumentar o disminuir el valor. Este tipo de convenciones no deberían darse por supuestas si se quiere proporcionar funcionalidad a cualquier individuo independientemente de la forma en la que acceda a los contenidos de un objeto Flash.

Supongamos que el control de volumen está incluido en un clip de película y se desea proporcionar toda la información relevante sobre **contexto y funcionalidad**. Podría ser identificativa la descripción: “*Barra de desplazamiento vertical para controlar el volumen del sonido*”.

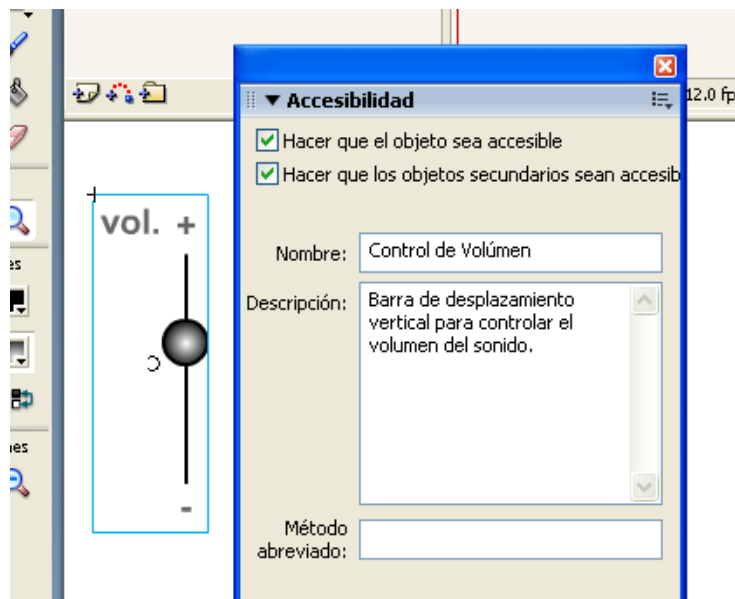


Figura 15. Propiedades de accesibilidad para un el Clip de Película que contiene el control de volúmen

A su vez, el propio control (símbolo de tipo botón sobre el que actúa el efecto de desplazamiento) debería no sólo proporcionar información sobre su **función** sino también sobre su **funcionalidad (información de contexto)**:

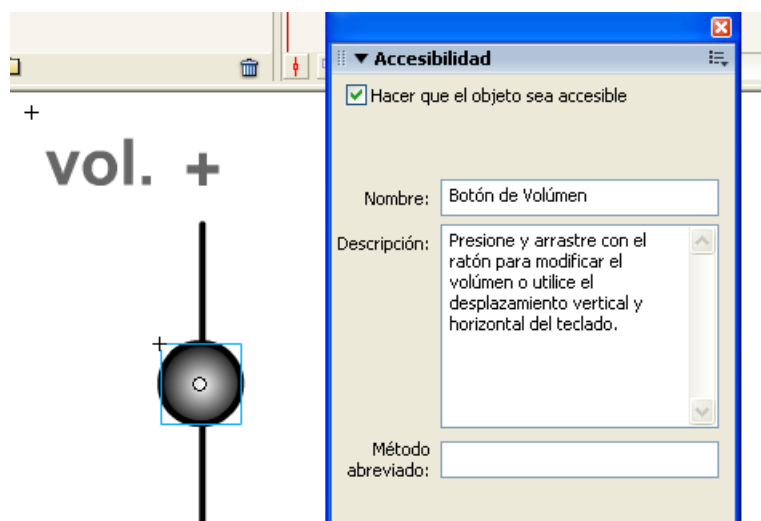


Figura 16. Propiedades de accesibilidad para el elemento botón.

Para aportar toda la información de contexto posible, se podría incluso proporcionar en todo momento la información relativa a los cambios de estado del elemento. Dado que la variación del volúmen se producirá en tiempo real en función de las acciones de los usuarios, tendremos que asignar las opciones de accesibilidad a través de *ActionScript*:

MATERIAL DE SOPORTE

GUÍAS DE ACCESIBILIDAD EN FLASH

- Flash 8 Accessibility: <http://www.adobe.com/resources/accessibility/flash8/>
- Flash Accessibility Presentation:
https://admin.adobe.acrobat.com/_a295153/p29082639/
- Centro de Recursos de Accesibilidad Adobe: <http://www.adobe.com/es/accessibility/>
- Best Practices for Accessible Flash Design:
http://www.adobe.com/resources/accessibility/best_practices/best_practices_acc_flash.pdf

SOFTWARE

- **JAWS** (Lector de Pantalla) Demo para Windows. Descarga de la versión de 30 minutos de uso: http://www.freedomscientific.com/fs_downloads/jaws.asp
- **Web Accessibility Toolbar** para Internet Explorer y Opera (para desactivación de objetos Flash y comprobación de alternativa):
<http://www.visionaustralia.org.au/ais/toolbar/>
- **Flashblock** - Extensión para *Mozilla Firefox* (permite detectar y activar los objetos Flash de un determinado documento).
- **Colour Contrast Analyser 1.1**: <http://www.wat-c.org/tools/CCA/1.1/>
- **ColorDoctor 2.01** (Comprobaciones de color):
<http://www.fujitsu.com/global/accessibility/assistance/cd/>
- **AccRepair** para Flash (herramienta de reparación – *programa propietario*):
http://www.hisoftware.com/accrepair_flash/index.html
- **Adobe Captivate 2** (entre otras funcionalidades, permite realizar subtulado y transcripción en objetos Flash - *programa propietario*):
<http://www.adobe.com/products/captivate/>

OTRAS REFERENCIAS

- *Blog de Accesibilidad de Adobe:* <http://blogs.adobe.com/accessibility/>
- *Flash MX Accessibility Issues:*
<http://www.oreillynet.com/pub/a/javascript/2002/11/08/flashmx.html>
- *Creating Accessible Macromedia Flash Content:*
<http://www.webaim.org/techniques/flash/>
- *Making Flash Usable for Users With Disabilities:*
<http://www.useit.com/alertbox/20021014.html>
- *Flash Access: Unclear on the Concept:* <http://alistapart.com/articles/unclear/>
- *Web Usability: Flash and Accessibility:* <http://www.usability.com.au/resources/flash.cfm>