VÍDEOS INTERACTIVOS CON DESCARTES

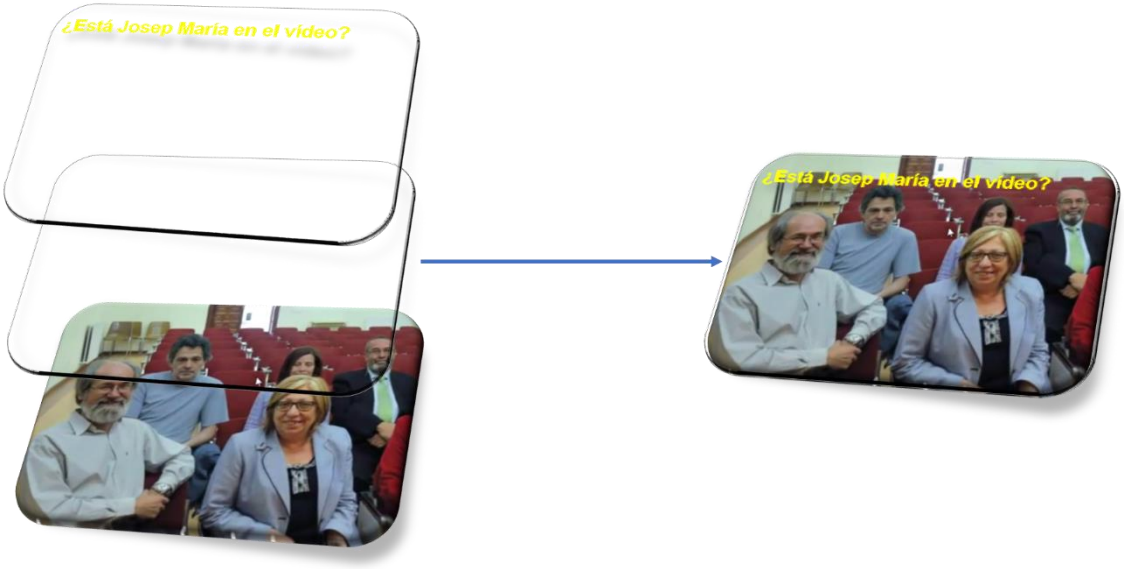
Juan Guillermo Rivera Berrío

Institución Universitaria Pascual Bravo

VÍDEOS INTERACTIVOS

Los vídeos interactivos, normalmente, son vídeos a los que se le agregan capas transparentes, con el fin de sobreponer elementos externos como imágenes, textos complementarios, cuestionarios y actividades interactivas. Actualmente se vienen desarrollando varias aplicaciones que permiten el diseño de este tipo de vídeos, por ejemplo http://h5p.org/interactive-video

El editor de Descartes brinda la posibilidad de sobreponer capas transparentes, lo cual abre un gran abanico de alternativas o, si se prefiere, de diversos modelos de vídeos interactivos como los que se explicarán en este documento. En la siguiente imagen se ilustra, en una forma simple, cómo se sobrepone al vídeo una capa transparente y un elemento de texto.



COMUNICACIÓN DE UN ESCENA CON UN HTML EMBEBIDO EN UN HTMLIFRAME

En la documentación técnica y de usuario de Descartes en su versión compatible con HTML5, en el apartado http://descartesjs.org/documentacion/?p=2709, se describe cómo se realiza una comunicación bidireccional entre una escena y una página HTML embebida en dicha escena. En general, la escena envía a través de comandos del tipo E.set('variable', valor) un valor hacia un elemento del código HTML, el cual se actualiza; por otra parte, en el HTML embebido hay otras variables cuyos valores se envían a la escena.

Aprovechando esta comunicación, se han diseñado otros modelos de vídeos interactivos, que complementan el modelo desarrollado y descrito ampliamente en el apartado http://descartesjs.org/documentacion/?p=2381. A diferencia del modelo anteriormente documentado, los nuevos modelos utilizan vídeos que se reproducen en una página HTML, que luego se embebe en la escena, con la ventaja adicional de incluir vídeos, también embebidos, de YouTube.

Es importante recordar que el objetivo del vídeo interactivo es permitir al usuario interactuar con la escena, en la cual el objeto principal es el vídeo. La interacción se realiza a través de preguntas o de la selección de opciones presentadas como botones o puntos gráficos dibujados sobre el vídeo. Esta interacción se logra si sólo se habilitan algunos controles de vídeo, que no permitan al usuario interferir en su reproducción como normalmente lo puede hacer. La comunicación bidireccional escena – HTML posibilita una gran variedad de alternativas o modelos de vídeos interactivos.

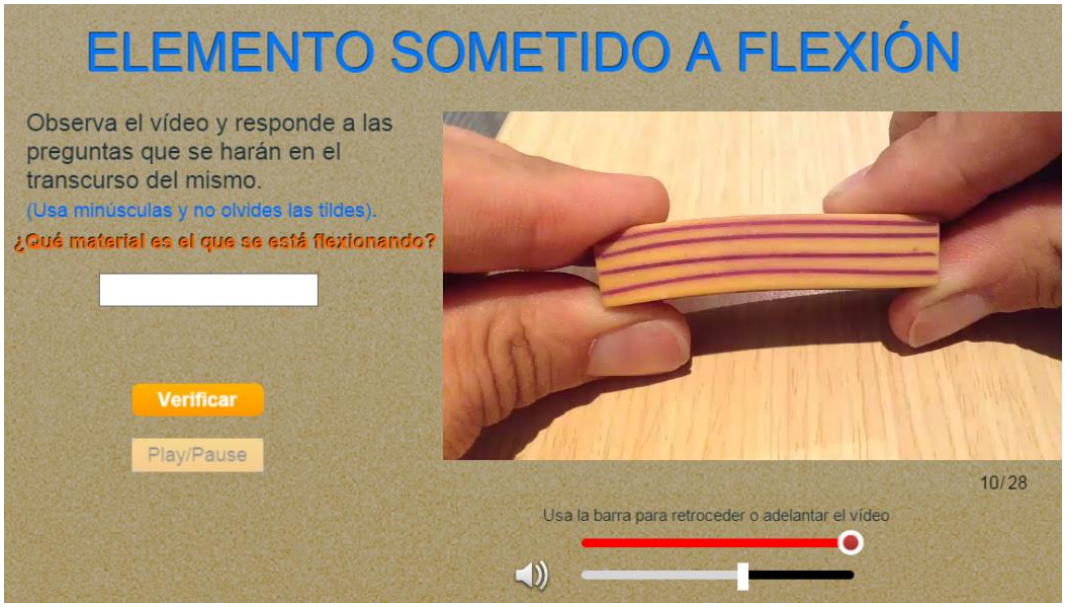
En este documento presentamos algunas de estas alternativas. Se hará una descripción técnica de los comandos JavaScript relacionados con la reproducción del vídeo y, por otra parte, se explicará cómo modificar la escena para incorporar otros vídeos y las interacciones correspondientes.

La hemos dividido en dos secciones; en la primera, se presentan modelos que incorporan vídeos almacenados en una carpeta de cada modelo; es decir, modelos para ejecución en local, si bien es posible reproducirlos en línea en ordenadores y algunos dispositivos móviles, su propósito principal es poderlos descargar e instalar en ordenadores que presentan baja conectividad; en la segunda sección, se presentan modelos que incluyen enlaces a vídeos de YouTube que, obviamente, requieren conexión a Internet, si bien es posible usar cualquier vídeo entre la gran abundancia que presenta este popular repositorio, recomendamos diseñar nuestros propios vídeos y luego subirlos a YouTube, un ejemplo de ello es el último modelo. Cada sección guarda estrecha similitud en sus aspectos técnicos y de programación, que se diferencia por ligeros cambios en las palabras reservadas o comandos utilizados por el API de YouTube.

1. Modelos con vídeos en local

1.1 Vídeo interactivo con preguntas de respuesta escrita

En la siguiente imagen se aprecia la escena con los elementos que contiene este modelo.



MODIFICACIÓN DE LA ESCENA

La intervención de la escena se realiza con el editor de Descartes, cargando el archivo **indexb.html**. Los únicos elementos a intervenir se encuentran en la opción **Programa** y en el algoritmo de **INICIO**.

Título de la escena. Basta cambiar el título que se encuentra ente comillas simples:

titulo='ELEMENTO SOMETIDO A FLEXIÓN'

Ubicación y nombre del vídeo. En la carpeta **videos** se encuentra el vídeo que cargará la escena, para este modelo lo hemos llamado video1 con formato mp4, por ello observarás la siguiente línea:

url='videos/video1.mp4'

Ahora, si nuestro vídeo tiene otro nombre y otro formato compatible con HTML5 (ogv o webm), sólo tienes que cambiarlo por el que se encuentra entre las comillas simples. Por ejemplo, en la carpeta **videos** hemos incluido el tráiler de *big buck bunny* en formato webm, que podemos cambiar en la escena así:

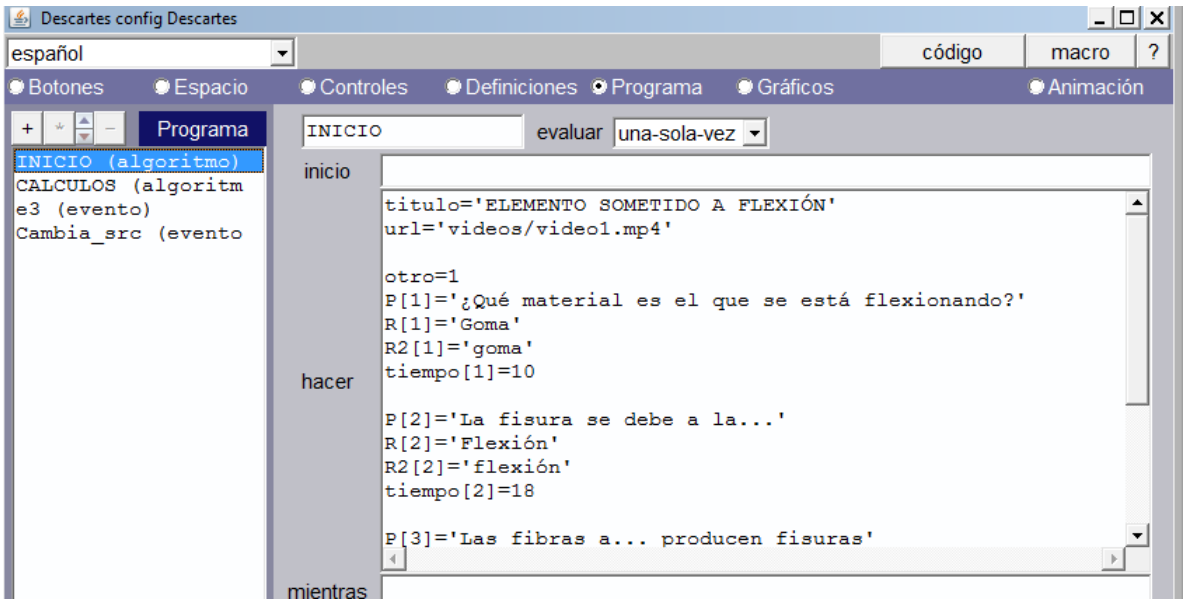
url='videos/ big-buck-bunny_trailer.webm'

Preguntas y respuestas. La escena está diseñada para un máximo de 10 preguntas. La primera pregunta tiene los siguientes elementos:

P[1]='¿Qué material es el que se está flexionando?'
R[1]='Goma'
R2[1]='goma'
tiempo[1]=10

Hemos incluido dos respuestas para validar respuestas con la primera letra en mayúscula o no. Se debe incluir el tiempo desde el inicio del vídeo en el cual se hará la pregunta. En forma similar se hacen las demás preguntas; por ejemplo, la pregunta dos:

P[2]='La fisura se debe a la...'
R[2]='Flexión'
R2[2]='flexión'
tiempo[2]=18



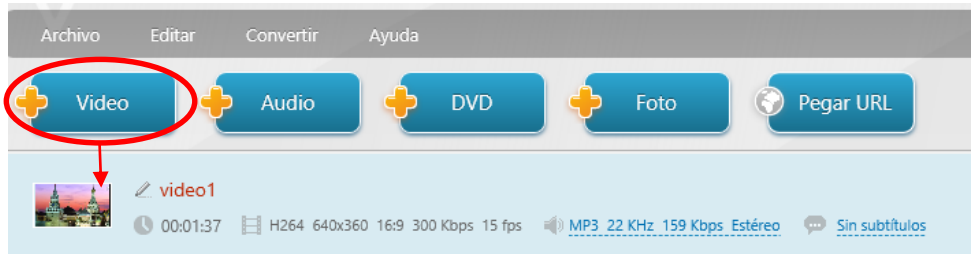
Otros elementos a tener en cuenta para modificar la escena, que se intervienen fuera de ella, son:

1.1.1 Carpeta de imágenes. Contiene imágenes como botones, fondo, sonido, las cuales pueden ser modificadas al gusto de quien vaya a modificar la escena.

1.1.2 Carpeta de vídeos. Contiene el vídeo a reproducir, este vídeo debe ser compatible con HTML5: mp4, ogg, ogv o webm. Es importante comprender que no todo formato mp4 garantiza un buen funcionamiento del vídeo, por ello, se recomienda utilizar un convertidor de vídeo. Uno de ellos, de descarga libre, es *Freemake Video Converter*, que se puede obtener desde este enlace: http://www.freemake.com/

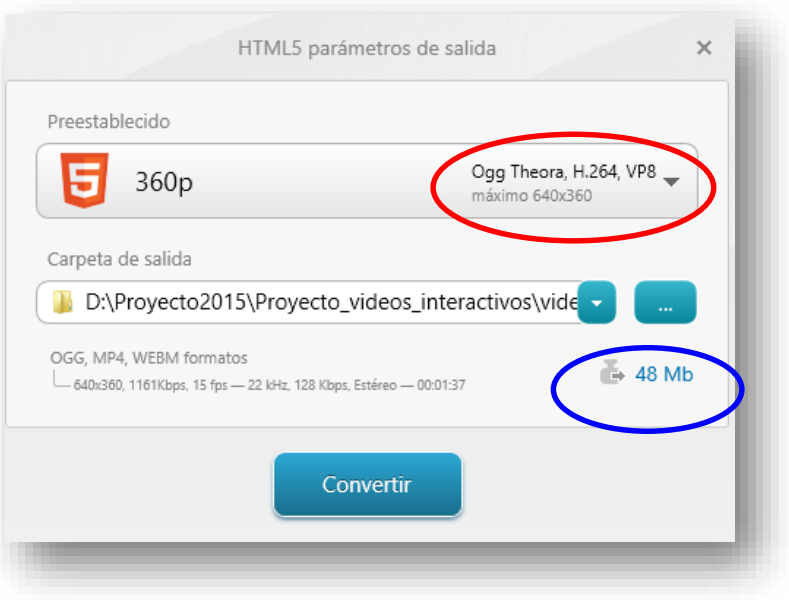
Para este caso especial, los pasos a seguir son:

- Elección del vídeo



- Selección del formato. Elegimos HTML5

- Definición de parámetros. Si se desea reducir el tamaño del vídeo, se pueden cambiar sus dimensiones (resaltado en rojo en la imagen) y su peso (resaltado en azul en la imagen).

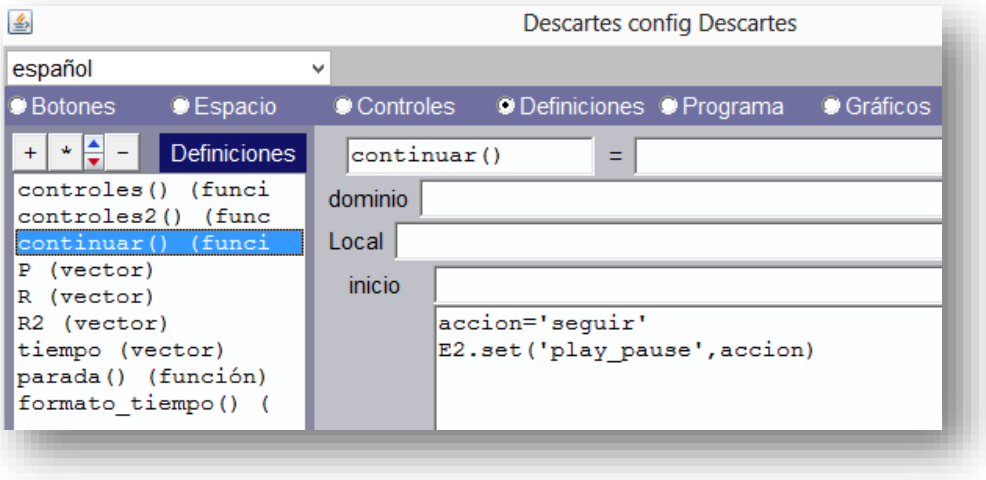


- Conversión. Luego del proceso de conversión, obtendremos tres vídeos en formatos mp4, ogv y webm, cada uno con un tamaño aproximado de 16 Mb, que suman los 48 que escogimos en el paso anterior (para tamaños menores, se debe elegir un límite menor, con alguna pérdida de calidad). Escogemos unos de estos tres vídeos para nuestra escena.

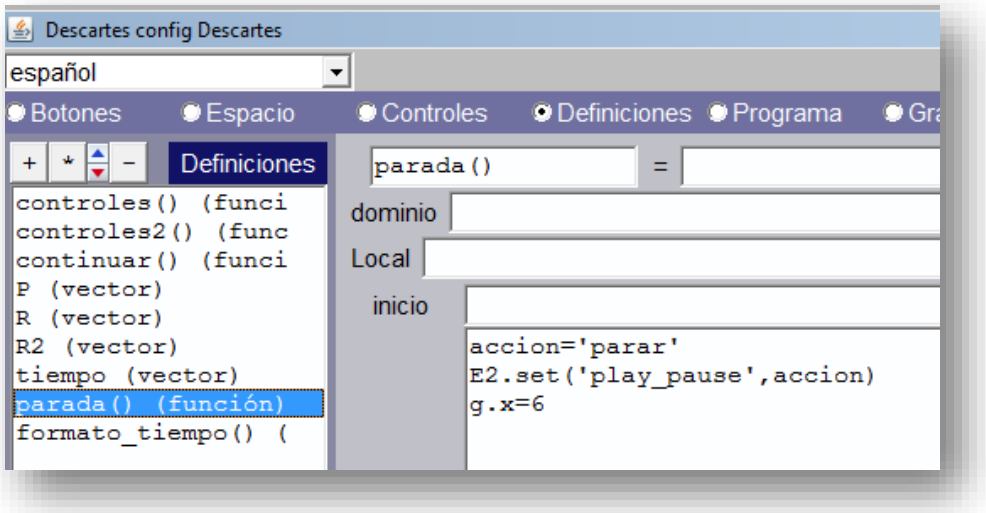
ASPECTOS TÉCNICOS EN LA ESCENA

En el código de esta escena se incluyen instrucciones que permiten comunicarla con el archivo ivideo.html, que se incluye en un espacio HTMLIframe, y que es el que gestiona el vídeo en sí. Procedemos a describir los elementos de la escena que permiten gestionar el vídeo.

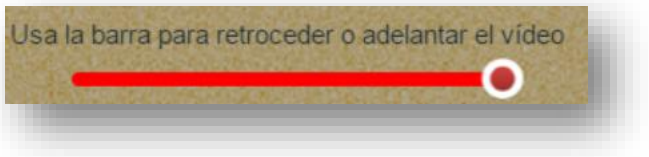
Play_pause. Variable que permite enviar órdenes al archivo HTML para que detenga o reanude la reproducción del vídeo. Si deseamos enviar el valor 'seguir' a la variable **play_pause**, podemos hacerlo con instrucciones como:



E2 es el espacio HTMLIframe. Obsérvese la función **parada(),** en la escena, que se ejecuta a través de un evento que controla el tiempo de reproducción para cada pregunta. La acción, entonces, sería 'parar' o detener el vídeo para responder a la pregunta.



actualizar_tiempo. Variable que reasigna el tiempo en segundos a partir del cual se debe reanudar la reproducción del vídeo. En este modelo se controla con un control gráfico (**g**):



La orden para actualizar el tiempo de reproducción es:

E2.set('actualizar_tiempo',g.x*avanzado/6)

Donde la variable **avanzado** es el tiempo transcurrido en el momento que se detiene la reproducción. El tiempo del vídeo se actualiza con valores entre 0 y el tiempo transcurrido (avanzado).

cambia_volumen. Variable que modifica el valor del volumen entre un valor mínimo (0) y un valor máximo (100). Al igual que la variable anterior, se controla con un control gráfico (**sonido**):

La orden para modificar el volumen es:

E2.set('cambia_volumen',(sonido.x-1)*20)

cambia_video. Como lo indicamos antes, esta variable nos permite cambiar el nombre yn la dirección de la url del vídeo que, obviamente, se extiende para vídeos en línea (puede porbar, por ejemplo, con url='http://tinyvid.tv/file/3r438pnhwxdsf.ogg', que carga un vídeo de los muppets en formato ogg). La orden para modificar la fuente (src) del vídeo es:

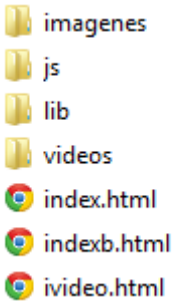
E2.set('cambia_video',url)

Finalmente, desde el archivo HTML se capturan los valores del tiempo de reproducción (t) y de la duración total del vídeo (t2), a través de las siguientes instrucciones **tiempo1=_Eval_(t)** y **tiempo_reproduccion=_Eval_(t2)**, respectivamente. Esta captura permite controlar los tiempos para realizar las preguntas, además de presentar el tiempo de reproducción en la escena (ver resaltado en rojo en la imagen).

ASPECTOS TÉCNICOS DE LOS ARCHIVOS JS

A continuación se describen los aspectos técnicos de este objeto. Esta sección sólo es de interés para aquellos usuarios que deseen conocer en detalle cómo se gestiona el vídeo y la comunicación entre la escena y el archivo ivideo.html. Como puede observarse, en el diseño de este objeto se ha tratado de aislar al desarrollador de la escena de esta parte técnica.

En la carpeta denominada "video_interactivo_modelo1_Local", se encuentran los siguientes elementos:



La explicación de los comandos JavaScript utilizados en el HTML embebido en la escena, será común para los demás ejemplos del apartado uno.

1.1.3 Carpeta lib. Contiene el intérprete de Descartes

1.1.4 Carpeta js. Contiene dos archivos. El primero, **resp.js**, es la librería que permite que la escena sea de diseño responsive o adaptable (véase el apartado http://descartesjs.org/documentacion/?p=2662). La segunda, **videolocal.js**, es la librería que permite la comunicación con la escena, así:

1.1.5 Mensajes hacia la escena. Para la manipulación del vídeo desde la escena, se requiere conocer el tiempo reproducido del vídeo (variable t) y la duración del vídeo (variable t2). La captura de estas dos variables se logra a través de los comandos:

```
seg = Math.round(myVid.currentTime);
```

```
duracion = Math.round(myVid.duration);
```

Donde myVid es la variable que contiene el nombre del video, la cual se declara con el comando:

```
myVid = document.getElementById("video1");
```

Para el caso de la primera variable a enviar (t) a la escena, se usan los siguientes comandos:

```
Contador.innerHTML = seg;
```

```
document.getElementById("texto_a_enviar").value = seg;

document.getElementById("texto_a_enviar").click();
```

En los cuales, la variable **texto_a_enviar** se maneja a través de un **input** creado en el archivo **ivideo.html**, que explicamos más adelante. La variable contenido envía a la escena el texto a enviar (seg) con el nombre (t), a través del siguiente bloque de comandos:

```
var contenido = document.getElementById("texto_a_enviar");
        contenido.addEventListener('click', function (evt) {
           window.parent.postMessage({
              type: "set",
              name: "t",
              value: contenido.value
           }, '*');
           window.parent.postMessage({
              type: "update"
           }, '*');
        });
```

En forma similar se envía la variable t2.

1.1.6 Mensajes desde la escena. Desde la escena se maneja un mensaje del tipo set que puede ser asociado a alguna de las siguientes variables: play_pause, retrocede, actualizar_tiempo, o cmabia_volumen (en la descripción de la escena se explicita cómo se envían estos mensajes).

play_pause. Se esperan dos posibles valores: 'parar' o 'seguir', lo que permite parar o reanudar el vídeo, así:

```
var data = evt.data;
            if ((data.type === "set")&&(data.name === "play_pause")) {
              // data.name es el nombre de la variable
              // data.value es el valor de la variable

              if (data.value == "parar") {
                 myVid.pause();
              } else if (data.value == "seguir") {
                 myVid.play();
              }

           }
```

Variables de avance y retroceso del vídeo (retrocede, actualiza_tiempo). Permiten modificar el tiempo de reproducción del vídeo:

```
function skip(value) {
      myVid.currentTime += value;
      }

function actualiza(value) {
      myVid.currentTime = value;
      }

function actualiza2(value) {
      myVid.currentTime = value;
      }
```

Variable de control de volumen (cambia_volumen). Asigna valores entre 0 (mínimo) y 1 (máximo) al volumen del vídeo.

```
function actualiza3(value) {
      myVid.volume = value/100;
      }
```

Variable que cambia la url del vídeo. Asigna el valor de la url definida por la escena.

```
function actualiza4(value) {
      myVid.src = value;
      }
```

1.1.7 Archivo ivideo.html. Es el HTML que se embebe en la escena, para la modificación de la escena no hay necesidad de intervenirla. Algunos bloques de diseño importantes son:

<script src="js/videolocal.js"></script>. Invoca la librería descrita en el apartado anterior

<input id="texto_a_enviar" type="text" style="VISIBILITY:hidden;display:none" />. Comando de entrada al que se le asigna el valor de la variable **texto_a_enviar**. Para nuestro caso es la variable **t**, la propiedad VISIBILITY permite ocultar el cuadro de texto que genera este comando.

En la librería videolocal.js se simula la tecla intro (enter) para enviar automáticamente este mensaje a la escena, con el comando:

document.getElementById("texto_a_enviar").click();

Vídeo. La reproducción del vídeo se realiza con la etiqueta HTML5 <video>, la cual permite que el vídeo se adapte al tamaño del espacio HTMLIframe de la escena. El bloque es el siguiente:

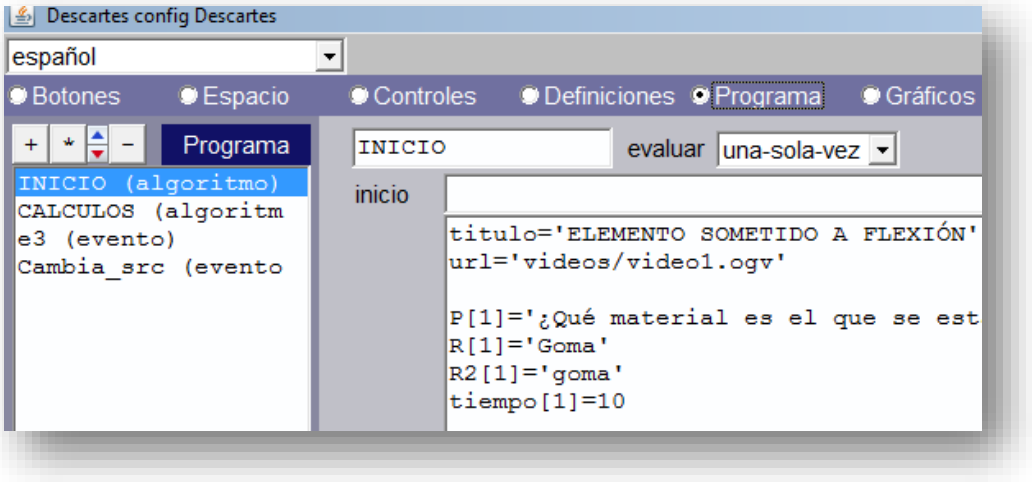
```
<video autoplay autobuffer id="video1" style="width:100%; height: auto !importante;">
```

```
        <source src="videos/video1.mp4" type="video/mp4">
        <source src="videos/video1.ogv" type="video/ogg">
        <source src="videos/video1.webm" type="video/webm">
</video>
```

Como se observa, se incluye la propiedad de autoreproducción, la ubicación y el nombre del vídeo y el tamaño adaptable.

1.2 Vídeo interactivo con preguntas de respuesta escrita (modelo 2). Es el mismo modelo anterior, pero las preguntas se superponen sobre el vídeo.

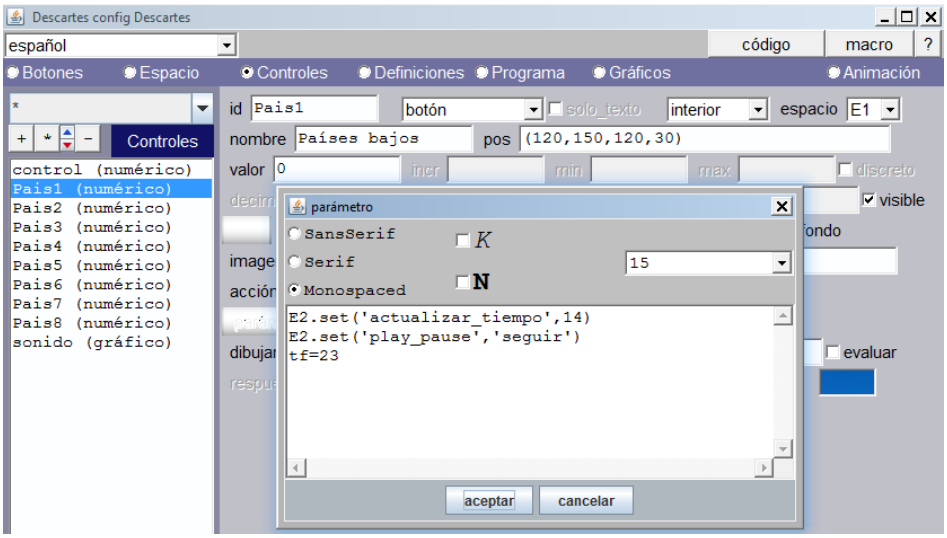
El vídeo utilizado para este ejemplo está en formato ogv.



1.3 Vídeo interactivo con selección de fragmentos de vídeo. Este modelo permite seleccionar fragmentos de vídeo a través de un tiempo inicial y uno final. Para ello, el que desee cambiar el vídeo, debe reproducirlo y tomar nota de los puntos de corte inicial y final. En nuestro ejemplo, hemos seleccionado ocho fragmentos de vídeo que se activan por medio de controles tipo botón.

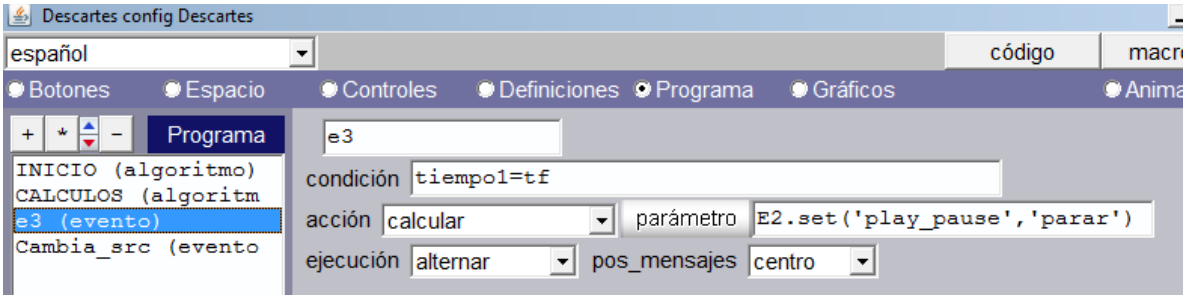
MODIFICACIÓN DE LA ESCENA

El título y la url del vídeo se modifican tal como se explicó en el primer modelo. Los tiempos de cada fragmento se modifican en los parámetros de cada botón, tal como se aprecia en la siguiente imagen.



Nótese que el tiempo inicial se define en la orden **E2.set('actualizar_tiempo',14)**, que para este caso sería 14 segundos. Esta orden y su efecto en el archivo HTML se explicó en el primer modelo.

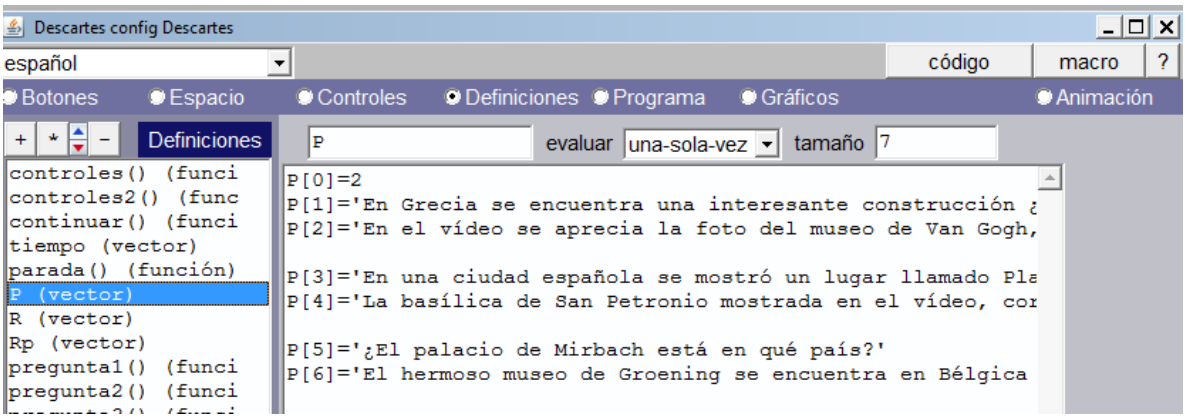
El tiempo final se designa por la instrucción: tf=23. Este límite se controla por un evento en la escena:



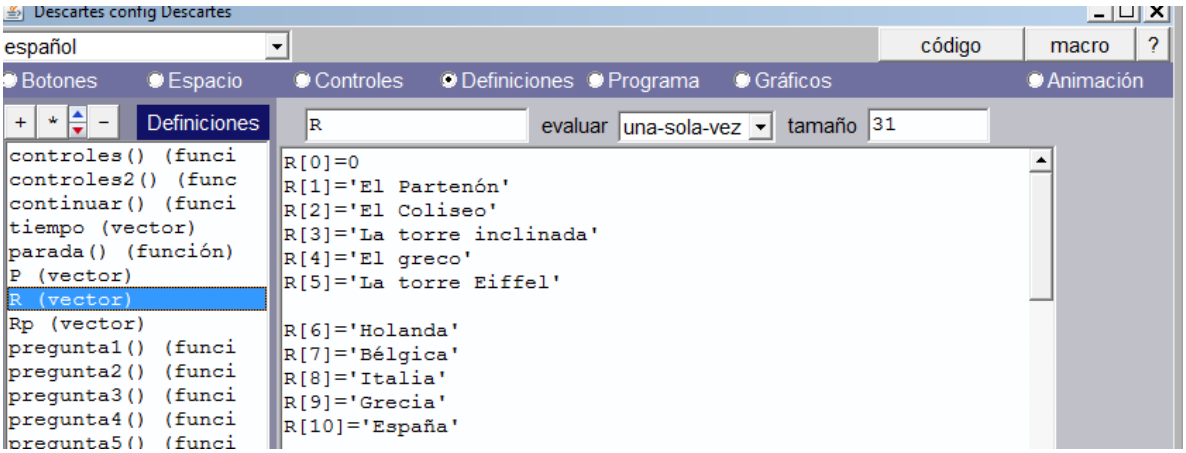
1.4 Vídeo interactivo con preguntas de selección múltiple. Este modelo es similar al segundo modelo, cambia en el tipo y número de preguntas.

MODIFICACIÓN DE LA ESCENA

Preguntas. Las preguntas se modifican en el vector P. La escena está diseñada para un máximo de seis preguntas, que se presentan en tres momentos de corte del vídeo. Se han utilizado funciones de cadena, que permitan preguntas de texto largo, obviamente sin abusar de los mismos.

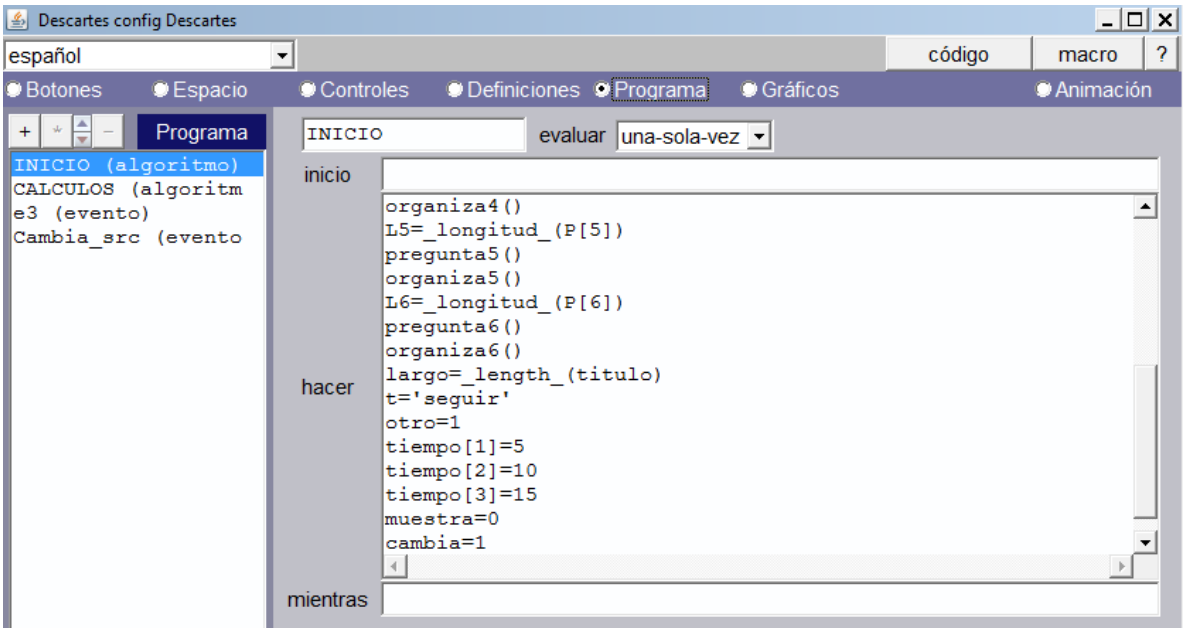


Respuestas. Cada pregunta tiene cinco respuestas asociadas, donde la primera es la correcta, la cual no se presentará, necesariamente, en esa misma posición, pues se distribuyen aleatoriamente.



Tiempo para presentar las preguntas. Cada pegunta aparecerá en un momento determinado de la reproducción del vídeo, que para nuestro ejemplo se ha determinado a los 5, 10 y 15 segundos. El vídeo de este ejemplo nada tiene que ver con las preguntas, es sólo ilustrativo.

La modificación de esto tiempos se hace en el algoritmo de INICIO.



Finalmente, se invita a desarrollar otros modelos a partir de lo explicado en esta primera parte. Es posible incorporar otro tipo de preguntas u otras acciones que combinen los modelos anteriores.

2. Modelos con vídeos de YouTube

2.1 Vídeo interactivo con preguntas de respuesta escrita

En la siguiente imagen se aprecia la escena con los elementos que contiene este modelo.

MODIFICACIÓN DE LA ESCENA

La intervención de la escena se realiza con el editor de Descartes, cargando el archivo **indexb.html**. Los únicos elementos a intervenir se encuentran en la opción **Programa** en el algoritmo de **INICIO** y el evento **e4**.

Título de la escena. Basta cambiar el título que se encuentra ente comillas simples:

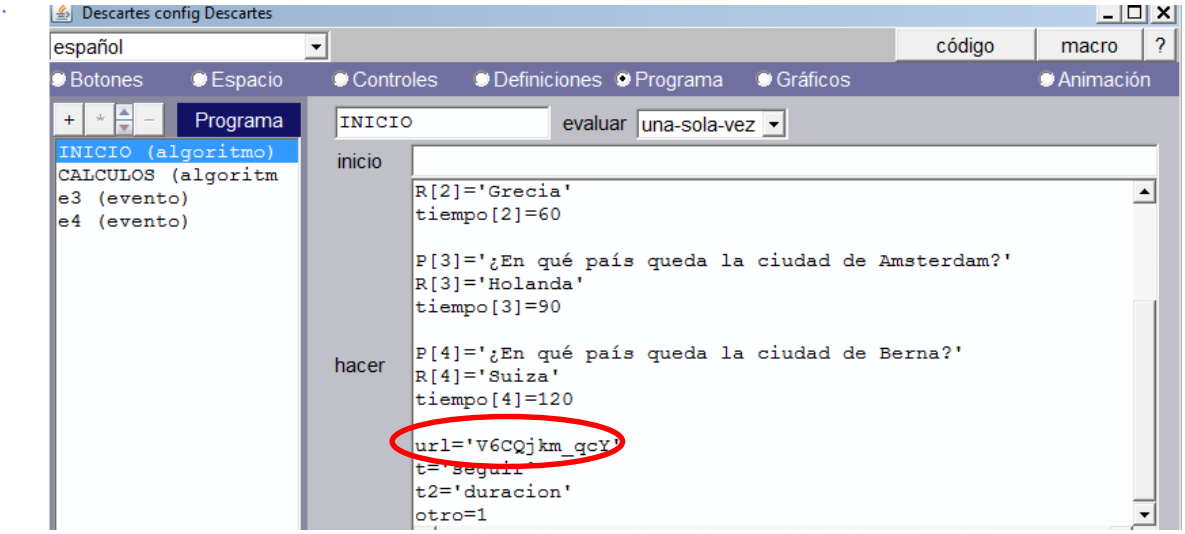
titulo= CIUDADES DE EUROPA'

Identificación del vídeo (id). El **id** de los vídeos en YouTube se presenta luego de la expresión **watch?v=**; por ejemplo, el primer vídeo del canal de YouTube de la Red Educativa Digital Descartes se encuentra en el enlace:

https://www.youtube.com/watch?v=podf-jQjxaM

El **id** del vídeo, entonces, sería **podf-jQjxaM**

Una vez identificado el vídeo que vamos a usar, vamos al algoritmo y modificamos la expresión **url='V6CQjkm_qcY'**



Preguntas y respuestas. La escena está diseñada para un máximo de 10 preguntas. La primera pregunta tiene los siguientes elementos:

P[1]='¿En qué país queda la ciudad de Amberes?'
R[1]='Bélgica'
tiempo[1]=50

Se debe incluir el tiempo (en segundos) desde el inicio del vídeo en el cual se hará la pregunta. En forma similar se hacen las demás preguntas; por ejemplo, la pregunta dos:

P[2]='¿En qué país queda el Partenon?'
R[2]='Grecia'
tiempo[2]=60

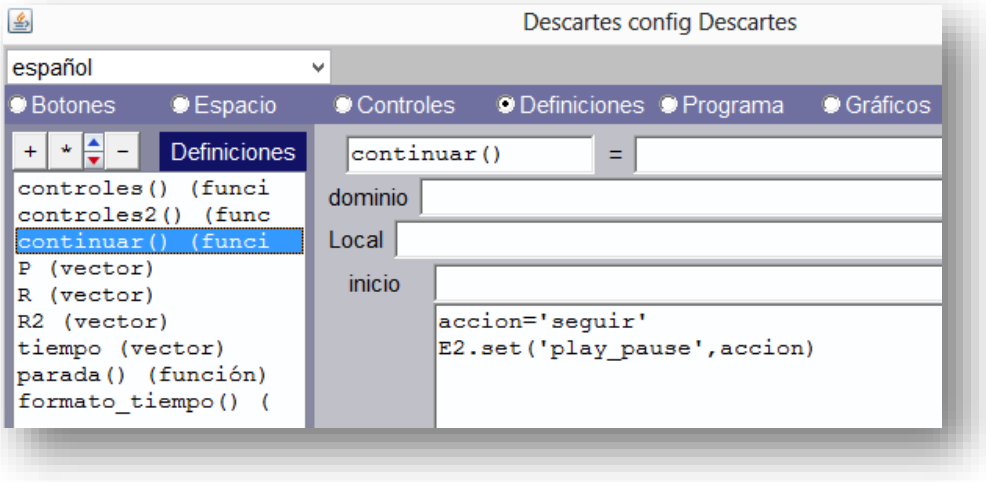
Vídeo con el logo. Al inicio de cada vídeo aparece el logo de la Institución Universitaria Pascual Bravo (mi Institución). Tienes tres opciones: i) Conservar este logo, ii) crear tu logo y subirlo a YouTube, luego incorporas el **id** obtenido (ver apartado sobre el archivo ivideo.html) o, iii) eliminar el logo. Si la decisión es la última opción, basta cambiar en el evento **e4**, la condición por **tiempo1>0**.

Otros elementos a tener en cuenta para modificar la escena, que se intervienen fuera de ella, son las imágenes, que explicamos en la sección anterior.

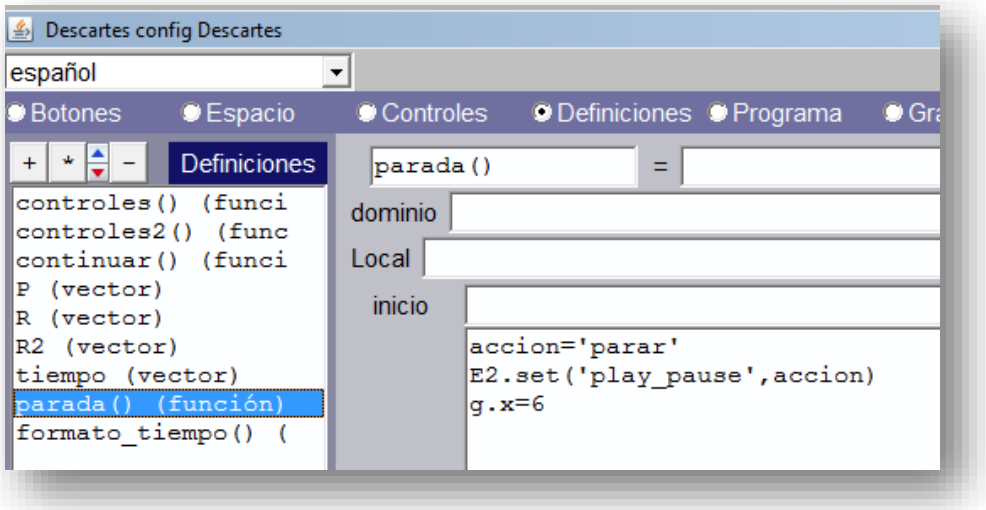
ASPECTOS TÉCNICOS EN LA ESCENA

En el código de esta escena se incluyen instrucciones que permiten comunicarla con el archivo ivideo.html, que se incluye en un espacio HTMLIframe, y que es el que gestiona el vídeo en sí. Procedemos a describir los elementos de la escena que permiten gestionar el vídeo. Repetimos varias explicaciones dadas en la sección anterior, de tal forma que el usuario que está sólo interesado en estos modelos, no tenga que consultar dicha sección.

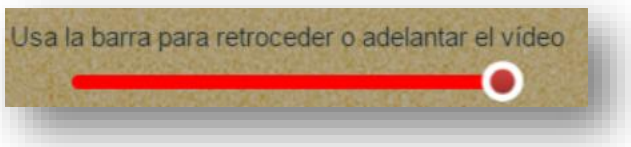
Play_pause. Variable que permite enviar órdenes al archivo HTML para que detenga o reanude la reproducción del vídeo. Si deseamos enviar el valor 'seguir' a la variable **play_pause**, podemos hacerlo con instrucciones como:



E2 es el espacio HTMLIframe. Obsérvese la función **parada(),** en la escena, que se ejecuta a través de un evento que controla el tiempo de reproducción para cada pregunta. La acción, entonces, sería 'parar' o detener el vídeo para responder a la pregunta.



actualizar_tiempo. Variable que reasigna el tiempo en segundos a partir del cual se debe reanudar la reproducción del vídeo. En este modelo se controla con un control gráfico (**g**):



La orden para actualizar el tiempo de reproducción es:

E2.set('actualizar_tiempo', g.x*avanzado/6)

Donde la variable **avanzado** es el tiempo transcurrido en el momento que se detiene la reproducción. El tiempo del vídeo se actualiza con valores entre 0 y el tiempo transcurrido (avanzado).

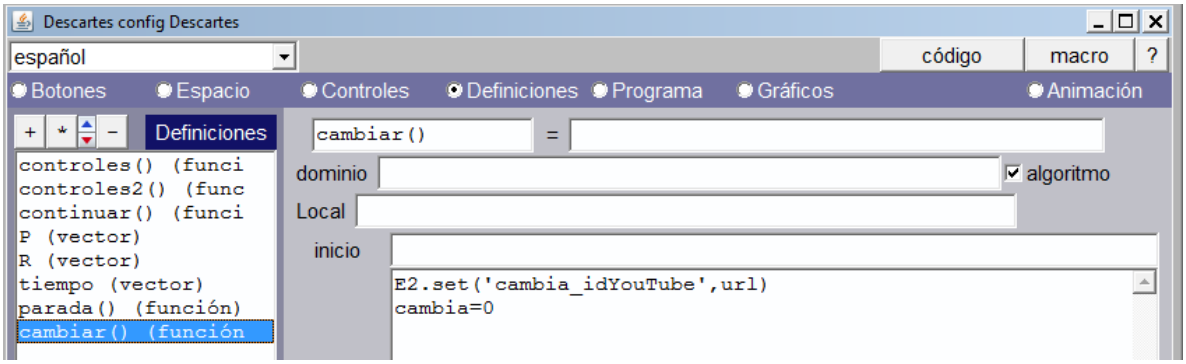
cambia_volumen. Variable que modifica el valor del volumen entre un valor mínimo (0) y un valor máximo (100). Al igual que la variable anterior, se controla con un control gráfico (**sonido**):

La orden para modificar el volumen es:

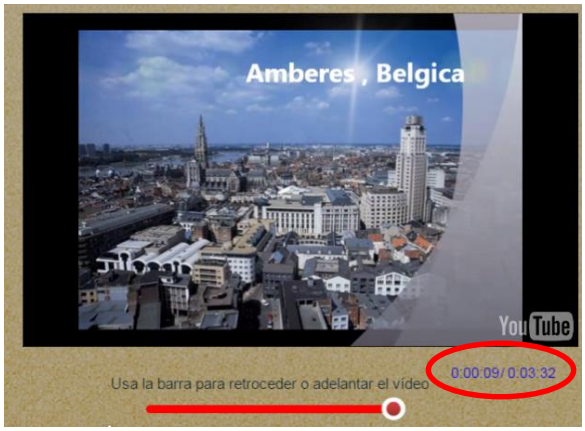
E2.set('cambia_volumen', (sonido.x-1)*20)

cambia_video. Como lo indicamos antes, la identificación de nuestro vídeo la asignamos a la variable url del vídeo. La orden para modificar el enlace de YouTube del vídeo es:

E2.set('cambia_idYouTube', url)



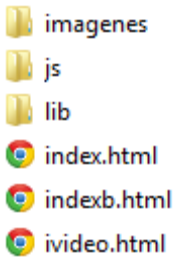
Finalmente, desde el archivo HTML se capturan los valores del tiempo de reproducción (t) y de la duración total del vídeo (t2), a través de las siguientes instrucciones **tiempo1=_Eval_(t)** y **tiempo_reproduccion=_Eval_(t2)**, respectivamente. Esta captura permite controlar los tiempos para realizar las preguntas, además de presentar el tiempo de reproducción en la escena y su duración (ver resaltado en rojo en la imagen).



ASPECTOS TÉCNICOS DE LOS ARCHIVOS JS

A continuación se describen los aspectos técnicos de este objeto. Esta sección sólo es de interés para aquellos usuarios que deseen conocer en detalle cómo se gestiona el vídeo y la comunicación entre la escena y el archivo ivideo.html. Como puede observarse, en el diseño de este objeto se ha tratado de aislar al desarrollador de la escena de esta parte técnica.

En la carpeta denominada "video_interactivo_modelo1_YouTube", se encuentran los siguientes elementos:



La explicación de los comandos JavaScript utilizados en el HTML embebido en la escena, será común para los demás ejemplos de esta sección.

2.1.1 Carpeta lib. Contiene el intérprete de Descartes

2.1.2 Carpeta js. Contiene dos archivos. El primero, **resp.js**, es la librería que permite que la escena sea de diseño responsive o adaptable (véase el apartado http://descartesjs.org/documentacion/?p=2662). La segunda, **youtube.js**, es la librería que permite la comunicación con la escena, así:

2.1.3 Mensajes hacia la escena. Para la manipulación del vídeo desde la escena, se requiere conocer el tiempo reproducido del vídeo (variable t) y la duración del vídeo (variable t2). La captura de estas dos variables se logra a través de los comandos:

```
seg = Math.round(player.getCurrentTime());
duracion = Math.round(player.getDuration());
```

Donde player es la variable que contiene el nombre del video, la cual se declara con el comando:

```
player = document.getElementById("video1");
```

Para el caso de la primera variable a enviar (t) a la escena, se usan los siguientes comandos:

```
Contador.innerHTML = seg;
document.getElementById("texto_a_enviar").value = seg;
document.getElementById("texto_a_enviar").click();
```

En los cuales, la variable **texto_a_enviar** se maneja a través de un **input** creado en el archivo **ivideo.html**, que explicamos más adelante. La variable contenido envía a la escena el texto a enviar (seg) con el nombre (t), a través del siguiente bloque de comandos:

```
var contenido = document.getElementById("texto_a_enviar");
        contenido.addEventListener('click', function (evt) {
           window.parent.postMessage({
              type: "set",
              name: "t",
              value: contenido.value
           }, '*');
           window.parent.postMessage({
              type: "update"
           }, '*');
        });
```

En forma similar se envía la variable t2.

2.1.4 Mensajes desde la escena. Desde la escena se maneja un mensaje del tipo set que puede ser asociado a alguna de las siguientes variables: play_pause, retrocede, actualizar_tiempo, o cambia_volumen (en la descripción de la escena se explicita cómo se envían estos mensajes).

play_pause. Se esperan dos posibles valores: ‘parar’ o ‘seguir’, lo que permite parar o reanudar el vídeo, así:

```
var data = evt.data;
        if ((data.type === "set")&&(data.name === "play_pause")) {
              if (data.value == "parar") {
                 player.pauseVideo();
                                          } else if (data.value == "seguir") {
                 player.playVideo();
```

```
        }
    }
```

Variables de avance y retroceso del vídeo (retrocede, actualiza_tiempo). Permiten modificar el tiempo de reproducción del vídeo:

```
function skip(value) {
    player.currentTime += value;
    }


function actualiza(value) {
    player.seekTo(value);
    }
```

Variable de control de volumen (cambia_volumen). Asigna valores entre 0 (mínimo) y 100 (máximo) al volumen del vídeo.

```
function actualiza3(value) {
    player.setVolume(value);
    }
```

Variable que cambia la url del vídeo. Asigna el valor de la url definida por la escena.

```
function actualiza4(value) {
    player.loadVideoById(value);
            player.playVideo();
    }
```

2.1.5 Archivo ivideo.html. Es el HTML que se embebe en la escena, para la modificación de la escena no hay necesidad de intervenirla. Algunos bloques de diseño importantes son:

<script src="js/youtube.js"></script>. Invoca la librería descrita en el apartado anterior

<input id="texto_a_enviar" type="text" style="VISIBILITY:hidden;display:none" />. Comando de entrada al que se le asigna el valor de la variable **texto_a_enviar**. Para nuestro caso es la variable **t**, la propiedad VISIBILITY permite ocultar el cuadro de texto que genera este comando.

En la librería youtube.js se simula la tecla intro (enter) para enviar automáticamente este mensaje a la escena, con el comando:

document.getElementById("texto_a_enviar").click();

<iframe>. La reproducción del vídeo se realiza con la etiqueta HTML5 <video>, la cual permite que el vídeo se adapte al tamaño del espacio HTMLIframe de la escena. El bloque es el siguiente:

<iframe id="video" src="http://www.youtube.com/embed/j8SdsVIBbZQ?enablejsapi=1&html5=1&rel=0&autoplay=1&controls=0&showinfo=0" width="550px" height="330px" frameborder="0" allowfullscreen></iframe>

Aquí es importante describir varios aspectos:

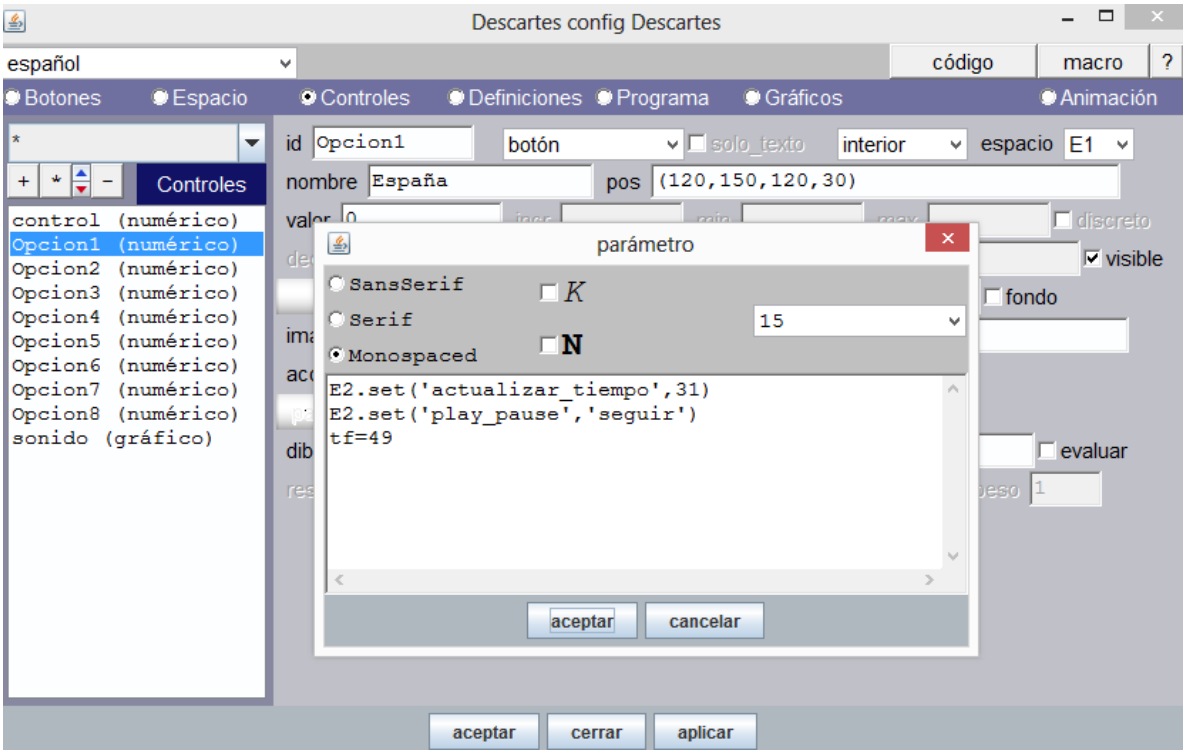
- En la fuente del vídeo (src) aparece el **id** j8SdsVIBbZQ, que corresponde al vídeo logo explicado en el apartado de modificación de la escena.
- **enablejsapi=1** es un comando para que se habilite el API de YouTube, que es lo que permite que podamos intervenir el vídeo.}
- Como se observa, se incluye la propiedad de autoreproducción (autoplay).
- Deshabilitación de los controles de YouTube (controls=0)

2.2 Vídeo interactivo con selección de fragmentos de vídeo. Este modelo permite seleccionar fragmentos de vídeo a través de un tiempo inicial y uno final. Para ello, el que desee cambiar el vídeo, debe reproducirlo y tomar nota de los puntos de corte inicial y final.

En nuestro ejemplo, hemos seleccionado ocho fragmentos de vídeo que se activan por medio de controles tipo botón.

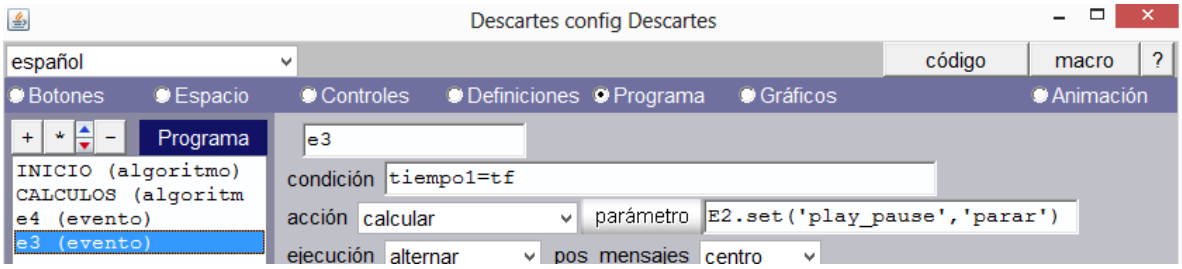
MODIFICACIÓN DE LA ESCENA

El título y la url del vídeo se modifican tal como se explicó en el primer modelo. Los tiempos de cada fragmento se modifican en los parámetros de cada botón, tal como se aprecia en la siguiente imagen.

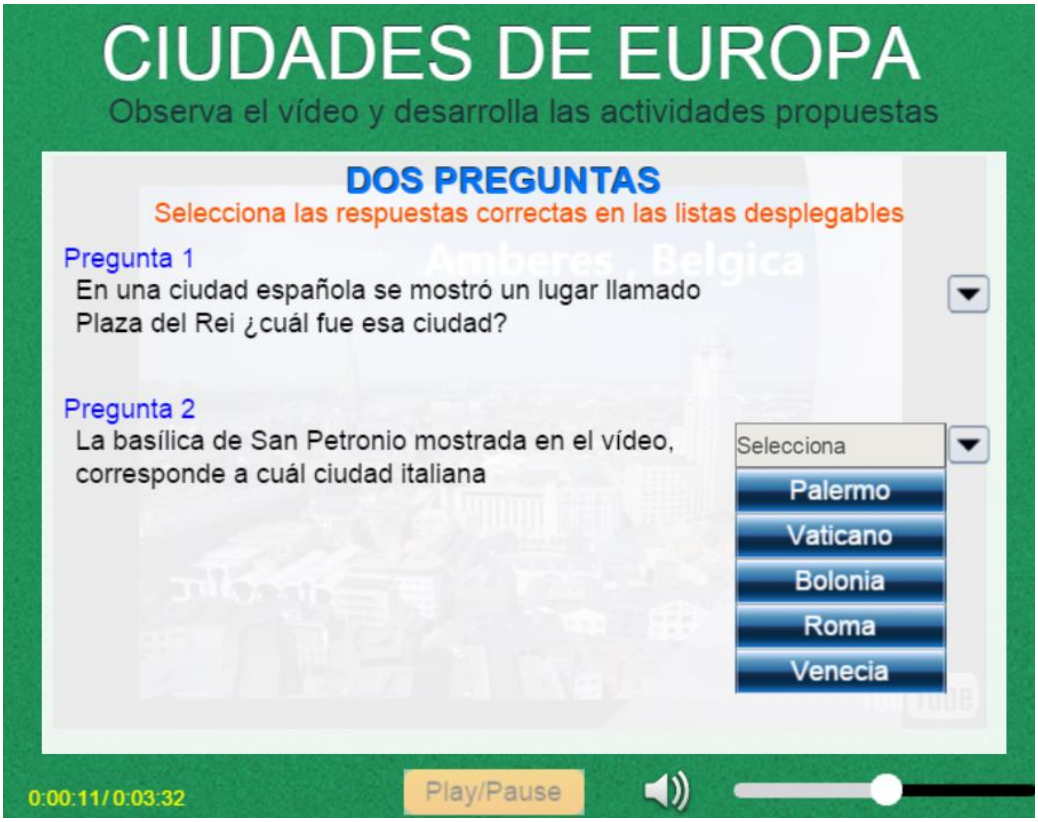


Nótese que el tiempo inicial se define en la orden **E2.set('actualizar_tiempo', 31)**, que para este caso sería 31 segundos. Esta orden y su efecto en el archivo HTML se explicó en el primer modelo.

El tiempo final se designa por la instrucción: tf=49. Este límite se controla por un evento en la escena:

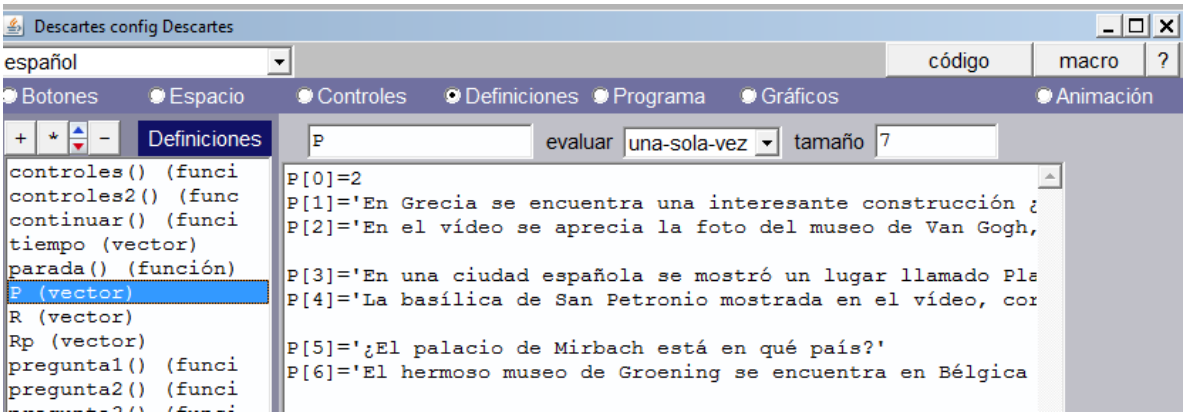


2.3 Vídeo interactivo con preguntas de selección múltiple. Este modelo es similar al segundo modelo, cambia en el tipo y número de preguntas.

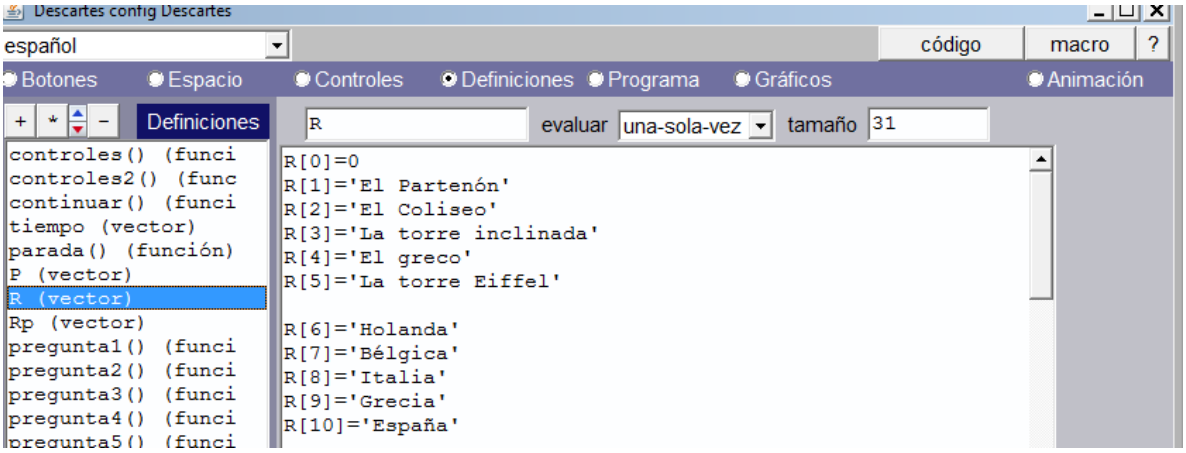


MODIFICACIÓN DE LA ESCENA

Preguntas. Las preguntas se modifican en el vector P. La escena está diseñada para un máximo de seis preguntas, que se presentan en tres momentos de corte del vídeo. Se han utilizado funciones de cadena, que permitan preguntas de texto largo, obviamente sin abusar de los mismos.

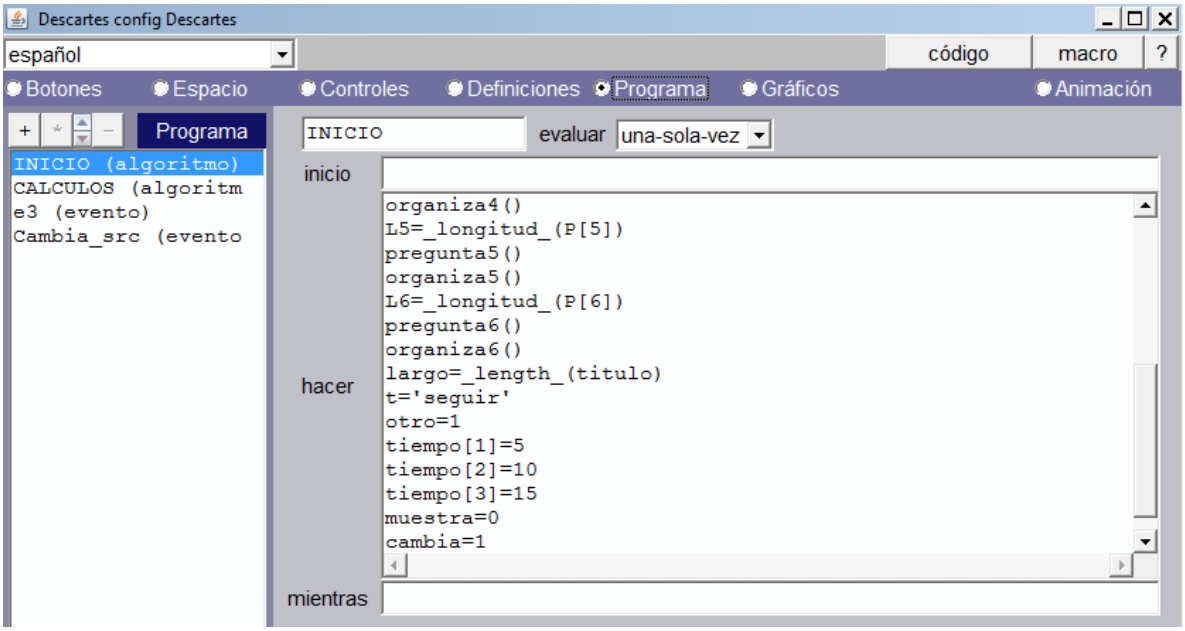


Respuestas. Cada pregunta tiene cinco respuestas asociadas, donde la primera es la correcta, la cual no se presentará, necesariamente, en esa misma posición, pues se distribuyen aleatoriamente.



Tiempo para presentar las preguntas. Cada pegunta aparecerá en un momento determinado de la reproducción del vídeo, que para nuestro ejemplo se ha determinado a los 5, 10 y 15 segundos. El vídeo de este ejemplo nada tiene que ver con las preguntas, es sólo ilustrativo.

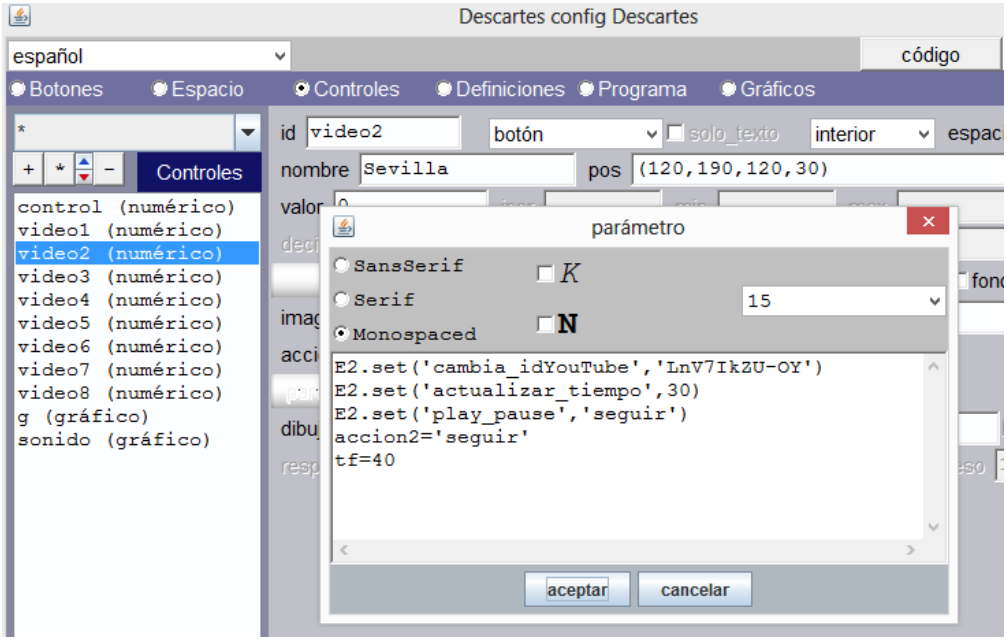
La modificación de esto tiempos se hace en el algoritmo de INICIO.



2.4 Vídeo interactivo con enlaces a otros vídeos. Este modelo es similar al segundo, su diferencia la dan los enlaces a nuevos vídeos, mientras que en el segundo se direccionaba al mismo vídeo en un punto temporal diferente. Este modelo no se presentó en la sección en local, en tanto que demandaría la inclusión de varios vídeos con la consecuencia en peso que ello implica.

MODIFICACIÓN DE LA ESCENA

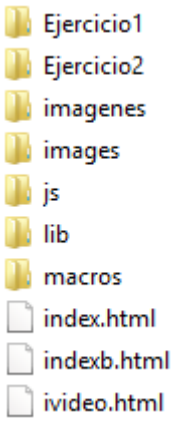
Los cambios son similares a los del segundo modelo, excepto por los parámetros de cada botón. Obsérvese, en la imagen, que se debe incluir el **id** del vídeo al cual el botón redirecciona la escena.



2.5 Vídeo interactivo con enlaces a otras escenas interactivas de Descartes. Una de las aplicaciones interesantes de los vídeos interactivos, es la posibilidad de incluir varias escenas interactivas, que se cargan de acuerdo a los tiempos que definamos para que el vídeo haga una pausa. En este ejemplo, hemos incluido dos escenas.

MODIFICACIÓN DE LA ESCENA

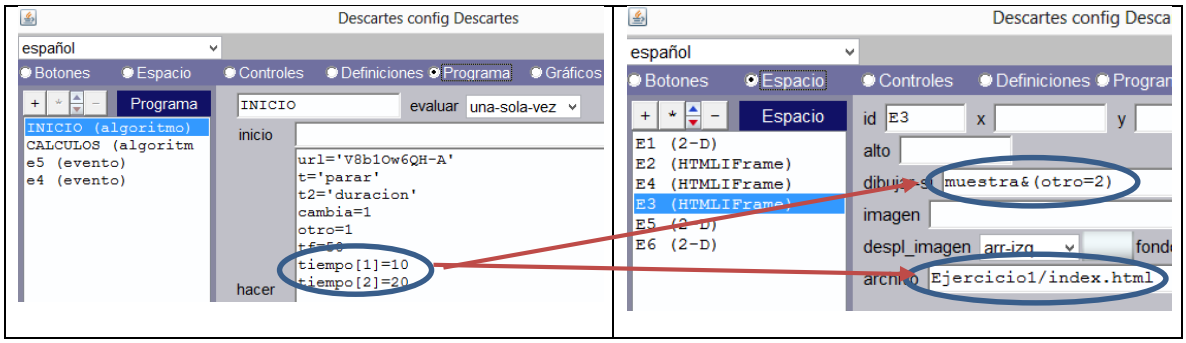
La url del vídeo se cambia en el algoritmo INICIO.



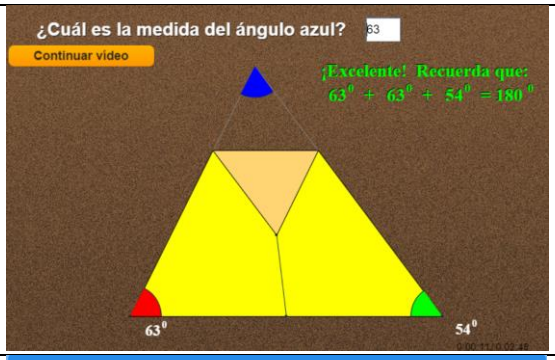
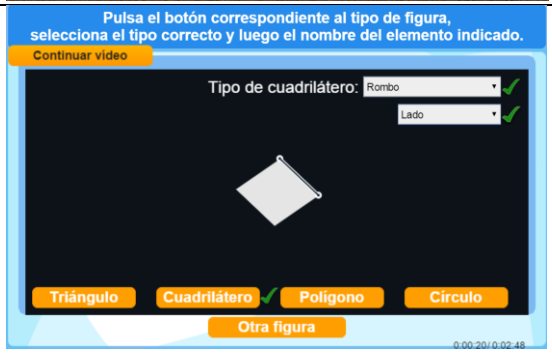
Las dos escenas del ejemplo se encuentran en las carpetas Ejercicio1 y Ejercicio2 (nombres que el usuario puede cambiar). La primera corresponde a una actividad del Proyecto Canals, la segunda es del Proyecto Pizarra Interactiva.

Podemos incluir las escenas Descartes que deseemos, las cuales están asociadas a los tiempos de parada (pause) del vídeo. Para el ejemplo, cuando el tiempo es 10 segundos, se carga la escena "Ejercicicio1", a los 20 segundos, se carga la escena "Ejercicio 2".

Las modificaciones, entonces, consiste en modificar los tiempos de parada en el algoritmo INICIO y el nombre de las escenas en los espacios HMTLIframe. Si se desean más escenas, habrá que crear nuevos espacios.



En la siguiente tabla se observan algunas imágenes del vídeo y las escenas interactivas.

VÍDEO	ESCENA INTERACTIVA
	
	

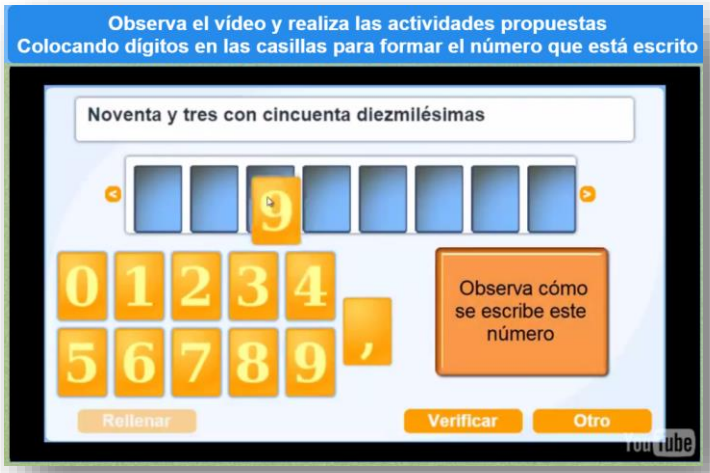
2.6 Vídeo interactivo y escena interactivas simultáneas. Una alternativa al modelo anterior, es incluir los comandos de vídeo en una escena interactiva. Dado que cada escena tiene sus particularidades, no damos un instructivo. Sin embargo, algunos elementos comunes son: inclusión de un HTMLIframe con el vídeo, controles para reproducir o detener el vídeo (en el modelo 6, véanse la funciones parada() y controles()), vector de control de tiempo y las instrucciones generales que se dan en los algoritmos de INICIO y CALCULOS. En la escena, se incluiría un control de retorno al vídeo.

A continuación, mostramos dos ejemplos de este tipo de modelo:

- Escena de escritura de decimales del Proyecto Pizarra Interactiva. Corresponde al modelo cuya carpeta es **video_interactivo_modelo6_YouTube**, el vídeo lo hemos diseñado con Camtasia y subido a YouTube.

- Escena de colorea comunidades del Proyecto GEOgráfica. Corresponde al modelo cuya carpeta es **colorea(video_interactivo)-JS**. En este caso, se muestra un vídeo de la región andaluza, luego aparece la escena interactiva para colorear una provincia, si esto se hace bien, se muestra un vídeo de esa provincia, además de colorearla.

A continuación se muestran imágenes de estos dos ejemplos:


Observa el vídeo y realiza las actividades propuestas
Colocando dígitos en las casillas para formar el número que está escrito
Noventa y tres con cincuenta diezmilésimas
9
0 1 2 3 4
5 6 7 8 9
,
Observa cómo se escribe este número
Rellenar
Verificar
Otro

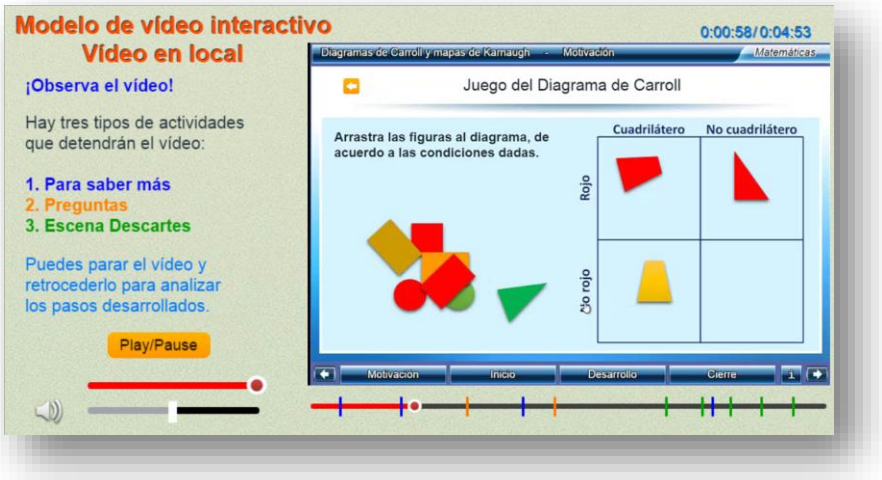
Geografía: Continente Europeo. España
Colorea Comunidades
CIUDAD REA
Bolaloázar
RIO ZUJAR
SIERRA MORENA
CanalSurTelevisión
Andalucía

Geografía: Continente Europeo. España
Colorea Comunidades
Vamos a colorear:
Andalucía
¡Excelente!, ahora
Haz clic en la provincia de:
Jaén
Almería
Intentos = 1

3. Modelo con marcas en la línea de tiempo

Finalmente, incluimos un último modelo (tanto en local como para YouTube) con una característica especial: marcas en la línea de tiempo del vídeo. El modelo está diseñado de tal forma que permite una edición sencilla desde el editor de Descartes, con la posibilidad de incluir tres tipos de actividades: imágenes con información complementaria, preguntas de selección múltiple y una escena interactiva[1].

En las siguientes imágenes se aprecia la línea de tiempo incluida y las marcas con diferentes colores que indican el tipo de actividad:



[1] Las escenas, para el modelo diseñados, son de Descartes; sin embargo, se pueden incluir escenas de GeoGebra o de cualquier otra herramienta.

MODIFICACIÓN DE LA ESCENA

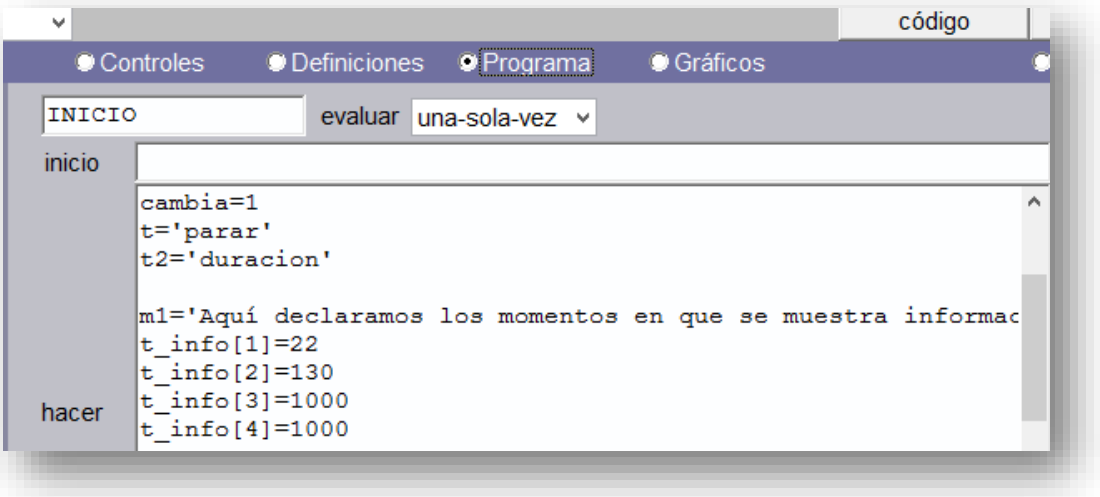
La explicación que se da a continuación, sirve tanto para el modelo con vídeo en local como el que reproduce un vídeo en YouTube.

Para modificar el vídeo interactivo, nos centramos únicamente en las tres actividades que interactúan con el vídeo.

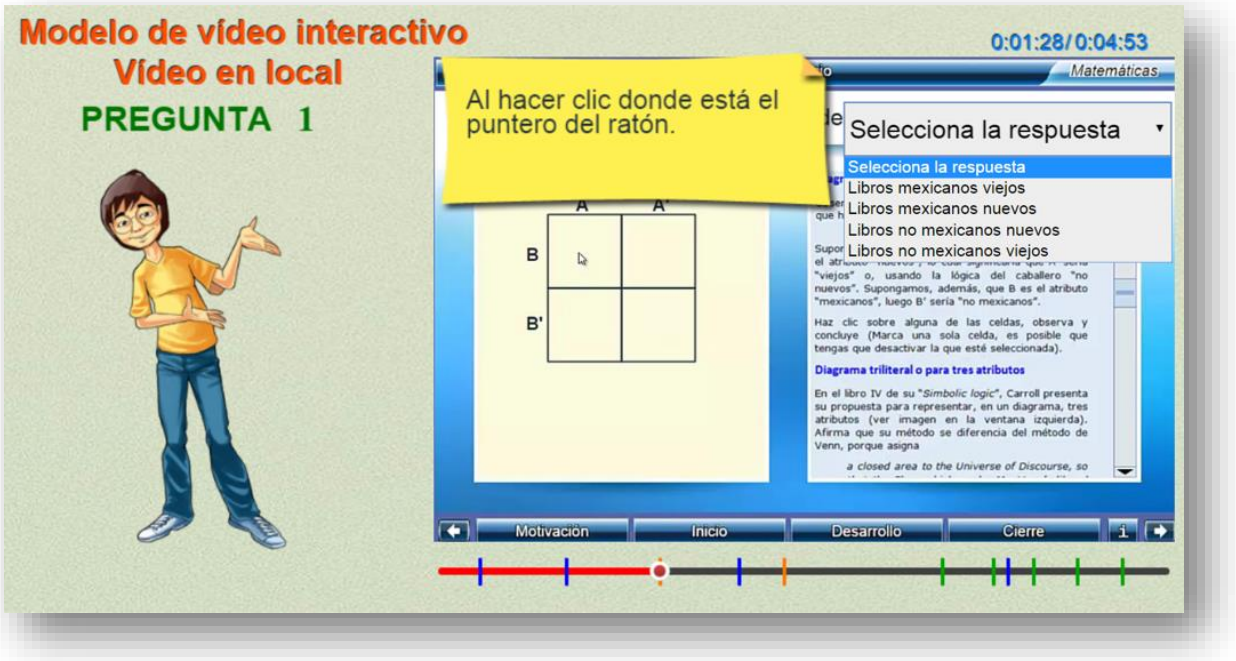
Actividad "Para saber más". Se presentan en la línea de tiempo marcadas con color azul. Su objetivo es presentar una imagen con información complementaria. Esta actividad detiene la reproducción del vídeo, muestra la imagen y un botón de cierre que permite la reanudación del vídeo. Para su modificación se requiere:

- **Modificar imágenes**. El modelo contempla la posibilidad de incluir cuatro imágenes, suficientes para un vídeo corto (no más de 7 minutos); sin embargo, es posible incluir más si se desea. Estas imágenes deben ir en la carpeta imágenes con los nombres i1.jpg, i2.jpg,… con un tamaño sugerido de 600x400 píxeles. Si no se desea este tipo de actividad en el vídeo, se recomienda dejar, al menos, la imagen i1.jpg para evitar lentitud en el editor de Descartes, pues por su diseño trata de leer al menos esta primera imagen.

- **Modificar los tiempos en la línea de tiempo**. En el editor de Descartes, en la opción programa (algoritmo INICIO), podemos modificar los valores del vector **t_info**. Estos valores son los segundos transcurridos en el vídeo para presentar una imagen. Si se desean más de cuatro imágenes, basta copiar las líneas adicionales; por ejemplo, para una quinta imagen a los dos minutos: t_info[5]=120. Si no se desea incluir imágenes, asignamos un tiempo de 1000 segundos o más. En la siguiente imagen, se observa que aparecerán sólo dos imágenes en el vídeo interactivo.



Actividad "Preguntas". Se presentan en la línea de tiempo marcadas con color naranja. Son preguntas de selección múltiple, que reanudan el vídeo una vez se seleccione una respuesta.



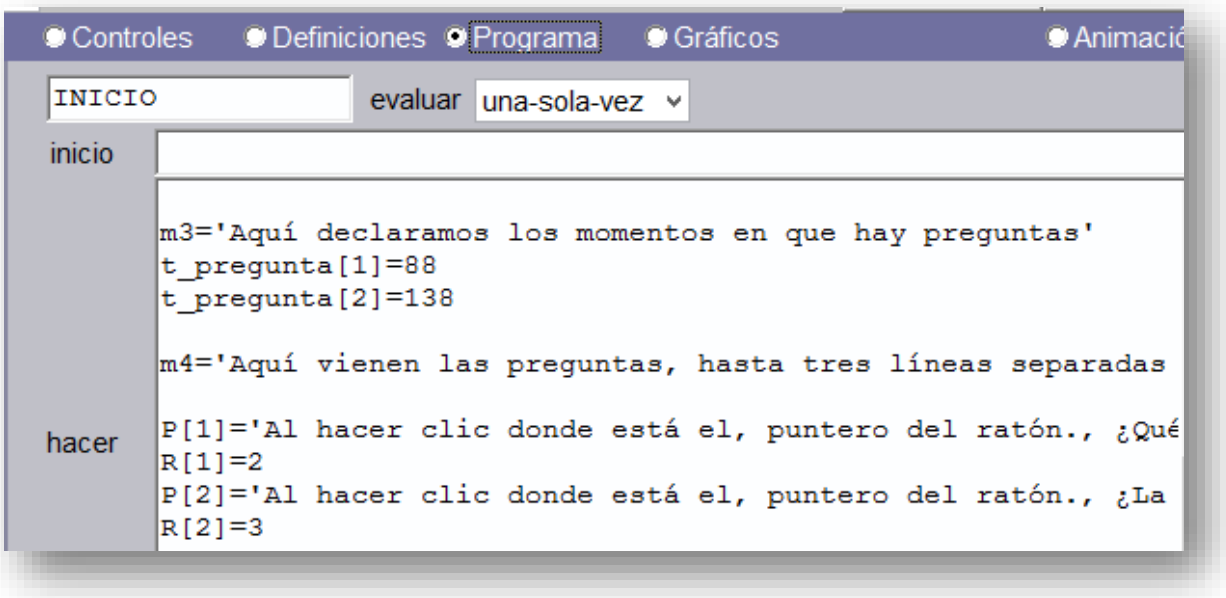
Para su modificación se requiere:

- **Modificar los tiempos en la línea de tiempo**. En el editor de Descartes, en la opción programa (algoritmo INICIO), podemos modificar los valores del vector **t_pregunta**. Estos valores son los segundos transcurridos en el vídeo para presentar una imagen. Si se desean más preguntas, basta copiar las líneas adicionales; por ejemplo, para una tercera pregunta a los dos minutos: t_pregunta[3]=120.

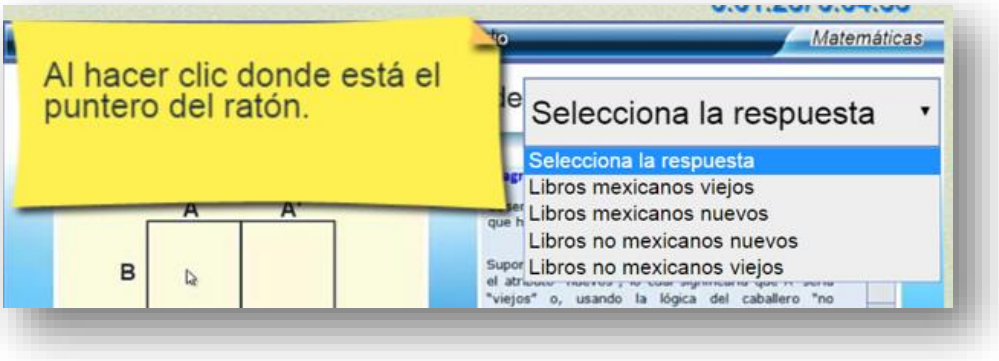
  Si no se desea incluir preguntas, asignamos un tiempo de 1000 segundos o más.

- **Modificar preguntas y respuestas**. En el mismo algoritmo anterior del editor Descartes, podemos modificar los valores de las preguntas asignadas al vector **P**. Como se observa en la imagen anterior, el enunciado de la pregunta puede tener más de una línea (para el modelo máximo tres), que se separan por comas. En nuestro ejemplo, se tienen formuladas dos preguntas P[1] y P[2]. Por otra parte, en el vector **R**, se asignan las posiciones en la que se encuentran las respuestas correctas (ver siguiente apartado).

  En la siguiente imagen, se observa que aparecerán sólo dos preguntas en el vídeo interactivo.

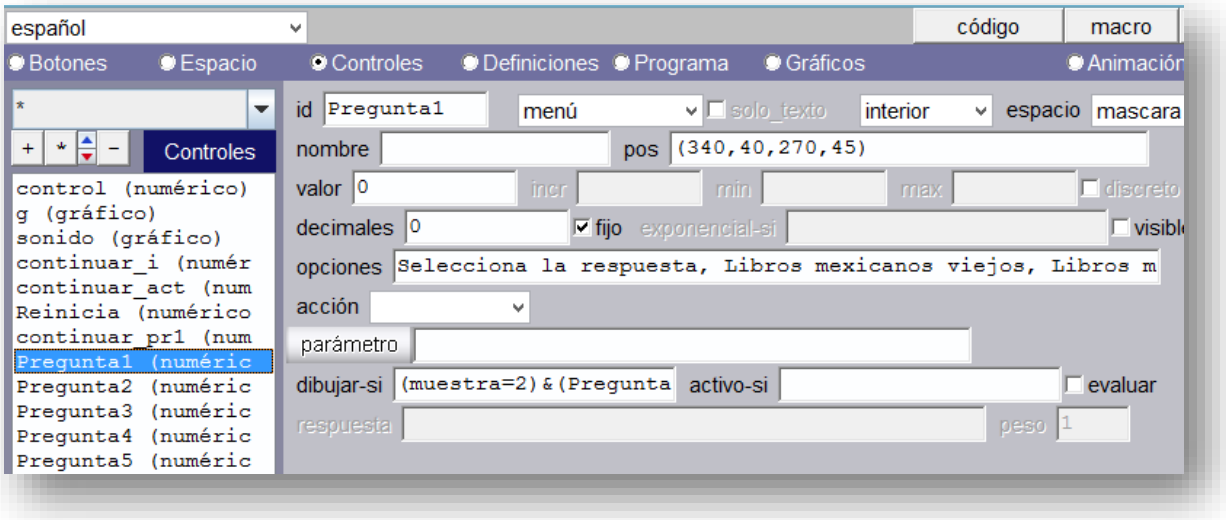


- **Modificar el menú de respuestas**. Como se trata de preguntas tipo selección múltiple, por cada pregunta aparecen cuatro opciones. Para nuestro ejemplo la respuesta correcta sería "Libros mexicanos nuevos", que corresponde a la posición dos; es decir, R[1] = 2



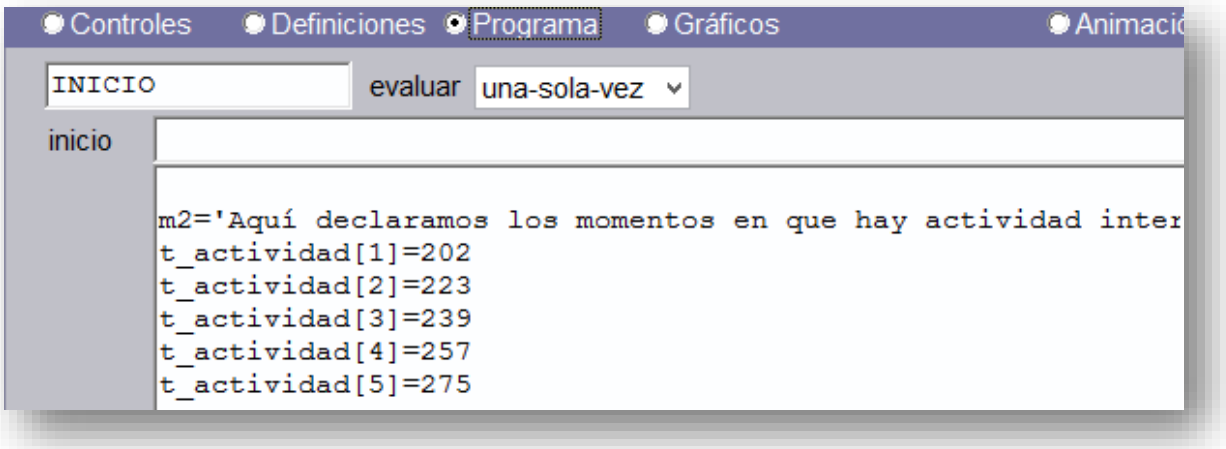
Este menú se modifica en los controles tipo **menú** que aparecen en el apartado **Controles** del editor de Descartes, basta cambiar el contenido de la casilla **opciones**, cada repuesta se separa con coma. La primera opción "Selecciona respuesta" no se modifica (ver siguiente imagen).

El modelo está diseñado para un máximo de cinco preguntas. Si no se desean hacer preguntas, es suficiente con modificar los tiempos del vector **t_pregunta**; es decir, no es necesario eliminar los cinco controles de preguntas.

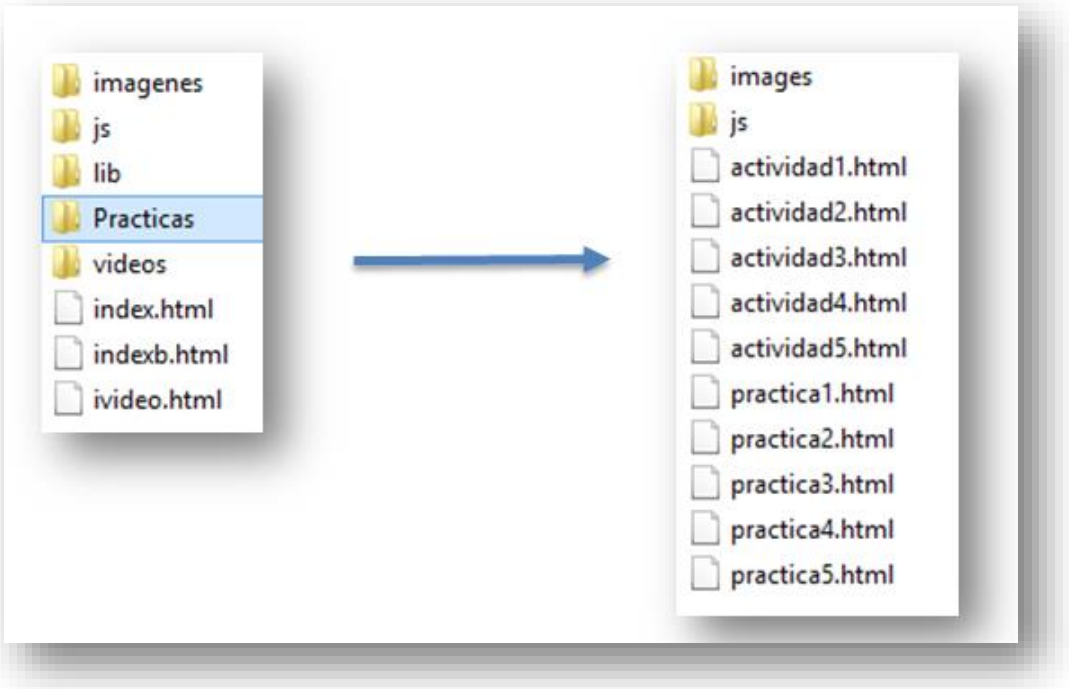


Actividad "Escena Descartes". Se presentan en la línea de tiempo marcadas con color verde. Para el modelo son escenas, pero igual podemos incluir otro tipo de escenas que estén adaptadas a HTML5, GeoGebra por ejemplo (ver ejemplo al final).

- **Modificar los tiempos en la línea de tiempo**. En el editor de Descartes, en la opción programa (algoritmo INICIO), podemos modificar los valores del vector **t_actividad**. Estos valores son los segundos transcurridos en el vídeo para presentar una de las escenas interactivas. Si se desean más actividades, basta copiar las líneas adicionales; por ejemplo, para una sexta actividad a los cinco minutos: t_actividad[6]=300. Al igual que en los caso anteriores, basta con asignar un tiempo grande (1000, por ejemplo) para no incluir este tipo de actividades.



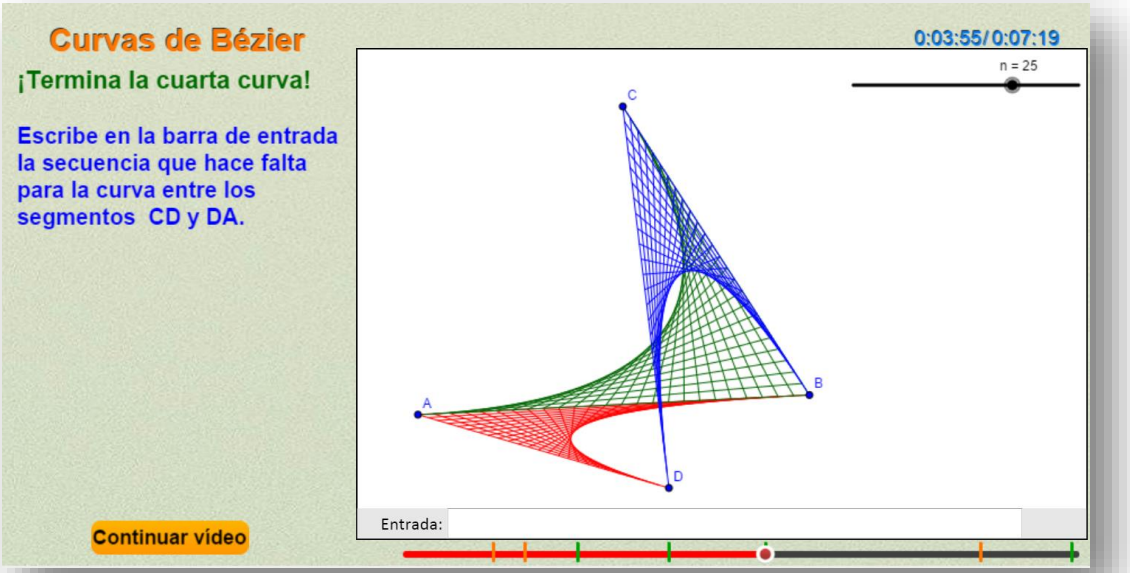
- **Modificar las escenas Descartes**. En la carpeta **Practicas** se guardan las escenas que vamos a utilizar. Cada escena debe nombrarse como practica1.html, practica2.html,…, con los correspondientes archivos y carpetas de soporte.



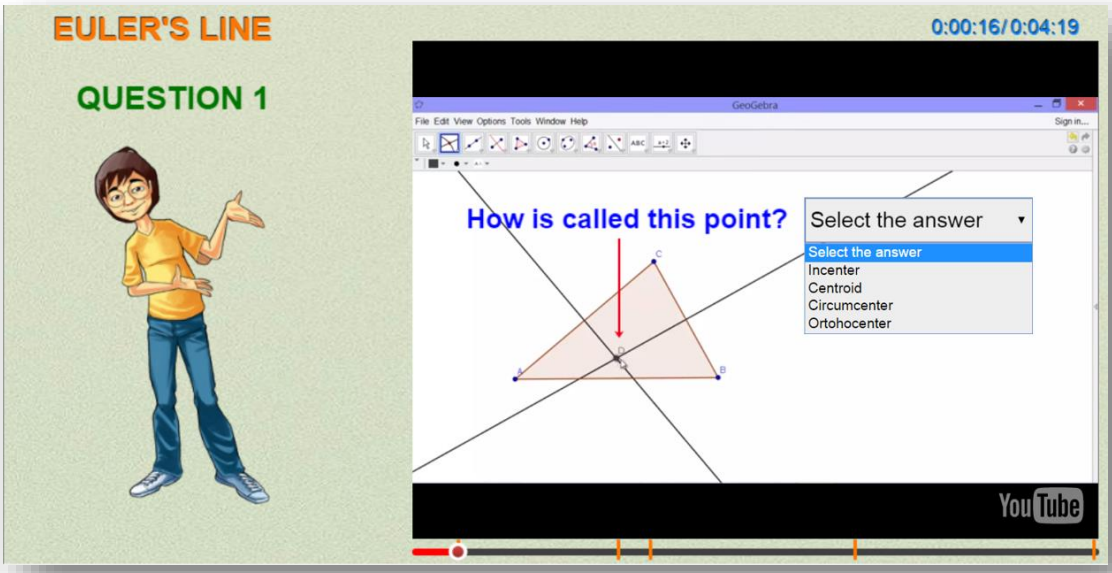
- **Escenas en formato adaptable**. El modelo permite la inclusión de escenas con las siguientes dimensiones: 970x550 para vídeos en local y 790x520 para vídeos de YouTube, cada una de ellas tienen un tratamiento de diseño adaptable para ajustarlas a la escena principal. Si se desean tamaños diferentes, se recomienda consultar cómo realizar la adaptación en http://descartesjs.org/documentacion/?tag=adaptable

En las siguientes imágenes, se pueden apreciar algunos ejemplos que utilizan escenas de Descartes y otros con escenas de GeoGebra.

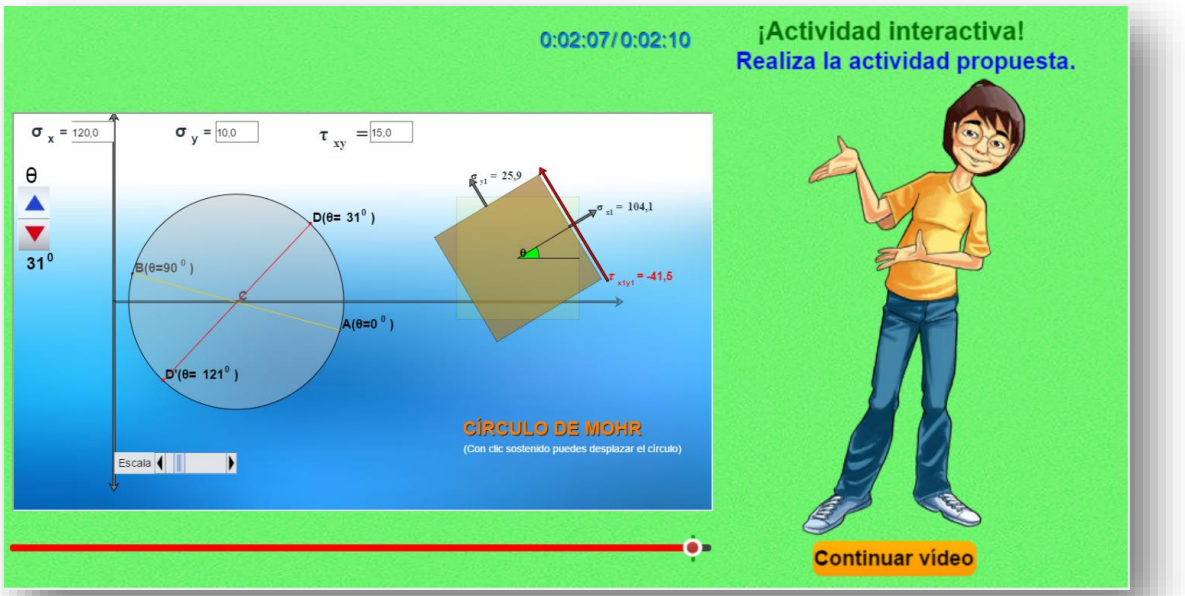
Vídeo interactivo con escenas de GeoGebra y Descartes, sólo se usan preguntas y actividades de GeoGebra.



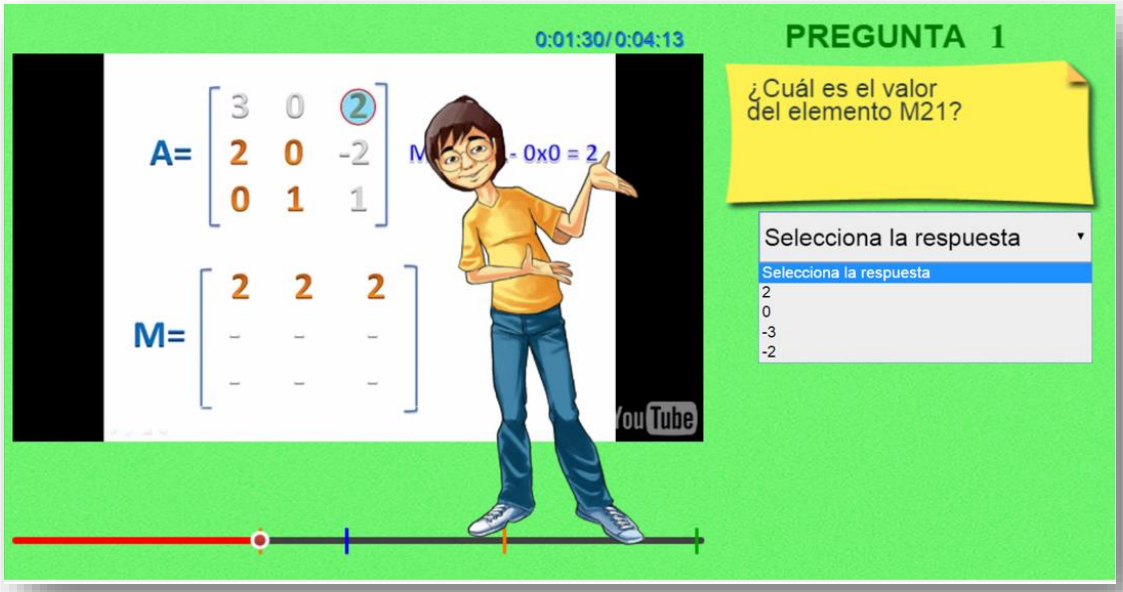
Vídeo interactivo con escenas de GeoGebra. Este ejemplo se preparó para publicarlo en el foro de GeoGebra de usuarios de habla inglesa.



Vídeo interactivo con escenas de Descartes y vídeo en local (incluye animación de https://www.moovly.com/).



Vídeo interactivo con escenas de Descartes y vídeo en YouTube.



Nota final: Cada uno de los ejemplos descritos en este documento, pueden ser consultados y descargados en http://descartesjs.org/documentacion/

CRÉDITOS

Diseño del proyecto y de los modelos: Juan Guillermo Rivera Berrío

Desarrollo de la comunicación escena – HTML: Joel Espinosa Longi

Desarrollo de la comunicación etiqueta <video> con la escena: Ramiro Lopera Sánchez y Juan Felipe Gómez Toro

Desarrollo de la comunicación <iframe> YouTube con la escena: Juan Guillermo Rivera Berrío.

Revisión de contenidos: José R. Galo Sánchez

Ilustraciones capítulo 3: Salvador Saucedo Ipiña