

# Harmony22

**Toon Boom  
Harmony22**

**Juegos**



---

**TOON BOOM ANIMATION INC.**

4200 St.Laurent Blvd, Suite 1020  
Montreal, Quebec, Canada  
H2W 2R2

---

**+1 514 278 8666**

**contact@toonboom.com**  
**toonboom.com**

## Avisos legales

Toon Boom Animation Inc.  
4200 Saint-Laurent, Suite 1020  
Montreal, Quebec, Canadá  
H2W 2R2

Tel.: +1 514 278 8666

Fax: +1 514 278 2666

[toonboom.com](http://toonboom.com)

## Exención de responsabilidad

El contenido de este documento es propiedad de Toon Boom Animation Inc. y cuenta con derechos de propiedad intelectual. Cualquier reproducción total o parcial está completamente prohibida.

El contenido de este documento está cubierto por una garantía limitada específica y por exclusiones y limitación de responsabilidad según el Contrato de licencia aplicable completado por los términos y condiciones especiales para el formato de archivo de Adobe®Flash® (SWF). Para obtener más información, consulte el Contrato de licencia, así como dichos términos y condiciones especiales.

Algunos iconos en este documento se han proporcionado con la versión Font Awesome Free 5.6.1 generada por Font Awesome. Estos iconos se proporcionan bajo la licencia CC BY 4.0. Para obtener más información sobre Font Awesome, visite <https://fontawesome.com>. Para obtener información sobre la licencia de Font Awesome Free, visite <https://fontawesome.com/license/free>.

Algunos iconos en este documento se han proporcionado con la fuente Glyphicons Halflings generada por Glyphicons. Para obtener más información sobre Glyphicons, visite <https://www.glyphicons.com/>.

## Marcas comerciales

Toon Boom® es una marca comercial registrada. Harmony™ y el logotipo de Toon Boom son marcas comerciales de Toon Boom Animation Inc. Todas las otras marcas comerciales son propiedad de sus respectivos dueños.

## Fecha de publicación

22/02/2023

Copyright © 2023 Toon Boom Animation Inc., una empresa de Corus Entertainment Inc. Reservados todos los derechos.

# Tabla de contenido

<b>Tabla de contenido</b>	<b>3</b>
<b>Acerca de los juegos</b>	<b>4</b>
<b>Notas de versión de Harmony Gaming SDK 2022</b>	<b>5</b>
<b>Capítulo 2: exportación Harmony de datos de para juegos</b>	<b>10</b>
<b>Capítulo 3: Acerca de la creación de recursos de juego</b>	<b>14</b>
Directrices de rigging de juego	15
Directrices de deformación de juego	17
Acerca de las deformaciones de huesos de juego	18
Directrices de la cuchilla de juego	19
Creación de notas de metadatos	20
Consejos de animación de juegos	22
Animación de varias secuencias	23
Acerca de la cámara ortográfica	25
<b>Capítulo 4: Acerca de la exportación a Unity</b>	<b>26</b>
Configuración de anclas	27
Exportación de cartas de sprites	28
Resoluciones de sprites	30
Uso de Bake_Groups	32
Exportación a Easel JS	34
Variaciones de paletas	36
<b>Capítulo 5: Acerca del Harmony Unity SDK</b>	<b>38</b>
Generador de vista previa de Harmony	42
Manipulación de recursos	44
Acerca del proyecto de ejemplo de Unity	50
Acerca de la interfaz de Unity	51
Importación de archivos de Harmony en Unity	53
Flujo de trabajo de archivo de archivo TBG en Unity	55
Flujo de trabajo de carpeta XML en Unity	62
Adición de un Harmony Renderer a objetos de juego vacíos	66
Configuración de colisiones en Unity	68
Glosario de juego	69
Índice	88

## Acerca de los juegos

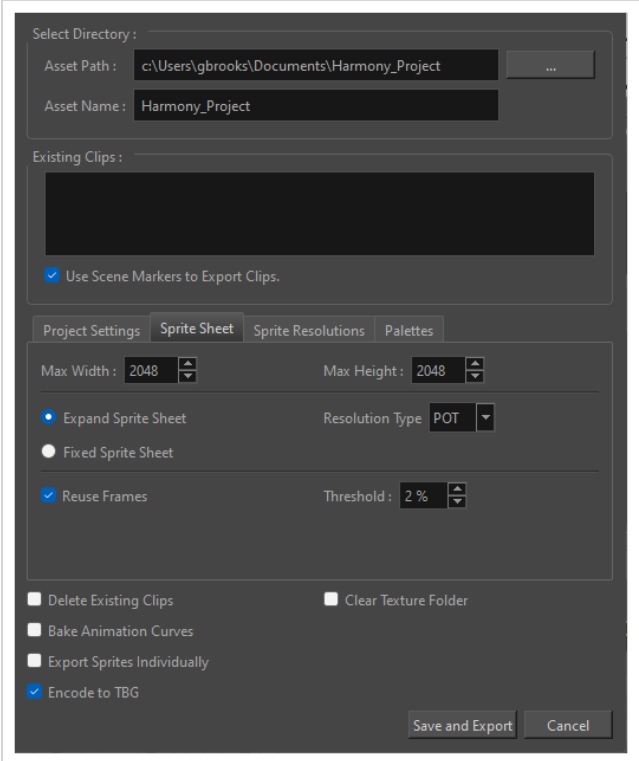
Harmony permite exportar animaciones creadas en Harmony a motores de juego externos. Esto le permite diseñar personajes con movimientos sencillos que pueden utilizarse en un motor de juego.

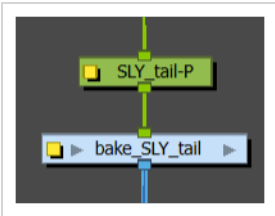
Unity es un ejemplo de motor de juego que puede incorporar los datos de Harmony. Toon Boom tiene un SDK de Harmony que puede descargar de la tienda de recursos de Unity para tener control sobre el personaje una vez se haya importado a Unity.

Esta guía le mostrará cómo exportar animaciones desde Harmony. Además, también se detallará Harmony Game SDK para la integración de Harmony con Unity.

# Notas de versión de Harmony Gaming SDK 2022

## Exportación desde Harmony

Característica	Descripción
<p>Nuevas opciones de exportador</p>	<p>Para facilitar un mejor soporte dentro de los editores de juegos, la interfaz Export to Sprite Sheet (Exportar a carta de sprites) ha recibido algunas características adicionales para empaquetar más información en el archivo exportado.</p>  <p><b>Nuevas opciones</b></p> <ul style="list-style-type: none"> <li>• Pestaña de selección Palettes (Paletas): excluye las paletas que no quiere incluir en el archivo exportado; esto cubrirá todas las posibles variaciones de paletas que tendrán como resultado la visualización de diferentes colores en el personaje</li> <li>• Encode to TBG (Codificar en TBG): comprime los archivos XML al formato de archivo TBG personalizado para que se puedan importar a Unity con los Unity Custom Importers. Cuando los archivos se importen, se tratarán como cualquier recurso en Unity e incluirán sub-assets para texturas, sprites y animaciones. Así se mejora la ergonomía de la creación y el desplazamiento de los datos de los personajes, y la comprobación de los cambios en el control de origen.</li> <li>• Rendimiento mejorado con «Save and Export»</li> </ul>

Característica	Descripción
	(Guardar y exportar): el proceso de exportación tarda, por lo general, 1/3 del tiempo comparado con Harmony 21, tanto para las carpetas XML como para archivos .tbg
Grupos «bake_»	<p>Actualmente, hay muchos nodos que no se pueden renderizar en el Game SDK. Como medio para incorporar el valor de dichos nodos a su exportación, existe una nueva función para agrupar dibujos y los nodos que les afectan en un nuevo grupo con el prefijo «bake_» (p. ej., bake_hips).</p>  <p>Cualquier grupo que lleve en su nombre el prefijo «bake_» se analizará en busca de fotogramas únicos de animación para efectos, movimientos y sustituciones de dibujo. Cuando se exporte a una carta de sprites, cada fotograma único de este grupo se exportará como un sprite. Si muchos fotogramas de la línea de tiempo no tienen nada cambiado dentro de ese grupo, todos se considerarán el mismo sprite exportado en la carta de sprites.</p>

## Importación de archivos TBG en Unity

Característica	Descripción
Automatic Importing Process	<p>Unity detectará automáticamente los cambios en los archivos TBG y los volverá a importar utilizando un importador personalizado. A continuación, el archivo se puede arrastrar a la escena como un prefab, tal y como se esperaría de un archivo FBX.</p> <p>Se generarán texturas, sprites, animaciones y recursos TbgStore, y proporcionarán como sub-assets debajo del recurso prefab TBG principal. Estos recursos no se tendrán que gestionar por separado en el control de origen y solo existirán en la caché de la librería de Unity. Junto al personaje, se genera un recurso de AnimatorController y carta de sprites, ya que estos no pueden funcionar como sub-assets.</p>
Importer Settings	<p>El proceso de importación se puede modificar desde la configuración de la Inspector View.</p> <p>Configuración de animaciones</p> <ul style="list-style-type: none"> <li>Discretization Step (Paso de discretización): El número de columnas/filas en las que se dividirán los sprites si los huesos los están deformando.</li> <li>Framerate (Fotogramas por segundo): El número de fotogramas por segundo a los</li> </ul>

Característica	Descripción
	<p>que se ejecuta la animación.</p> <ul style="list-style-type: none"> <li>• Stepped (En pasos): Deshabilitar la interpolación entre fotogramas creados.</li> <li>• Create Animator Controller (Crear controlador de animador): Deshabilitar la interpolación entre fotogramas creados.</li> <li>• AnimatorController: Proporciona un espacio de trabajo para la transición de animaciones entre sí dadas ciertas condiciones. Se puede hacer referencia a él en comandos de juego para activar animaciones durante la partida.</li> <li>• Maintain Curves on Cloned Clips (Mantener curvas en clips clonados): Las curvas duplicadas de los sub-assets del clip dentro del archivo .tbg (mencionado en AnimatorController) se sustituirán por los nuevos datos del archivo .tbg actualizado.</li> </ul> <p>Configuración del material</p> <ul style="list-style-type: none"> <li>• Shader: Se utiliza para renderizar todos los sprites del prefab</li> <li>• sRGB Texture (Textura SRGB): Mapear el espacio de color gamma de las texturas en el renderizado del espacio de color lineal.</li> <li>• Filter Mode: Cómo se interpolan los píxeles contiguos a la textura.</li> <li>• Mipmap Enabled (MipMap habilitado): Generar texturas de menor resolución para renderizar a distancias mayores.</li> <li>• Create Sprite Atlas (Crear atlas de sprites): Si no se hace referencia a ningún Sprite Atlas a continuación, se creará un nuevo recurso SpriteAtlas en el proyecto junto al archivo .tbg</li> <li>• SpriteAtlas (Atlas de sprites): Todos los sprites del archivo .tbg se pueden agregar a un SpriteAtlas; cuando se haya hecho, será capaz de combinar varios SpriteRenderers nuevos en un solo drawcall, mejorando el rendimiento del renderizado.</li> </ul>

## Renderización de archivos TBG en Unity

Característica	Descripción
Jerarquía de prefab	<p>La estructura del prefab generado en Unity debe coincidir con la estructura jerárquica configurada en la línea de tiempo de Harmony. La estructura de la vista de nodos de Harmony de las composiciones que entran en el nodo de salida se excluyen durante la exportación.</p> <p>Se pueden adjuntar nuevos GameObjects a cualquiera de estas transformaciones pegs hijas. Esto permite adjuntar equipo de procedimiento.</p> <p>Además, cualquier GameObject se puede habilitar o deshabilitar en tiempo de ejecución, por lo que las adiciones visuales opcionales en los personajes se pueden ocultar hasta que se necesiten. La jerarquía también se podría descomponer, reestructurar o duplicar por completo en tiempo de ejecución. Esto permite hacer explotar personajes o disparar ciertos dibujos como proyectiles.</p>

Característica	Descripción
Configuración del TBG Renderer	La resolución, paletas, materiales, colores y máscaras se pueden editar desde el Inspector, así como los comandos personalizados del GameObject. El material y el color de los SpriteRenderers también pueden configurarse de forma individual por dibujo, por lo que puede cambiar los colores de los accesorios de un personaje o aplicar efectos especiales a algunos dibujos mediante shaders personalizados.
Integración del 2D Animation Package	Los dibujos deformados por huesos de juego en Harmony se traducirán en huesos SpriteSkin en Unity utilizando el «2D Animation Package» de Unity. Este paquete ofrece soporte para la deformación de malla de varios hilos y de alto rendimiento para SpriteRenderers, así como controladores de interfaz de usuario para mover huesos en la Scene View. La cadena de huesos se puede animar mediante procedimientos utilizando el «IK Manager 2D» de Unity para fusionar efectos de apuntar, inclinarse, sacudirse, agitar, etc. después del paso de actualización de la animación.
Animaciones	<p>Todos los datos de animación se exportan desde Harmony por peg, por lo que todas las rotaciones y escalas intermedias se representan en la ventana Animation (Animación) en Unity. Las animaciones se pueden modificar una vez que su recurso AnimationClip asociado se duplica desde un sub-asset de TBG a un nuevo recurso en el proyecto, lo que permite agregar eventos y curvas adicionales dentro de Unity.</p> <p>Los AnimationClips se seleccionan a través de un AnimatorController, lo que proporciona un medio de transición entre animaciones, ya sea inmediatamente o con una fusión suave de transformaciones de peg.</p>
Integración del gráfico del shader	Como parte del SDK, se proporcionan gráficos de shaders compatibles que se pueden duplicar y modificar, para crear interfaz sencilla que permite introducir efectos visuales personalizados específicos del juego en sus personajes.

## Mejoras en HarmonyRenderer (carpetas XML)

Característica	Descripción
Vista previa de HarmonyProject	Al seleccionar un personaje en la carpeta de recursos de Unity, aparecerá una vista previa del personaje en la Preview Window. La Preview Window reproduce una vista previa de las animaciones del personaje y tendrá la posibilidad de cambiar las máscaras sin necesidad de arrastrar el personaje a la Scene View.



## Correcciones

En esta versión de Harmony Game SDK se han resuelto los siguientes problemas:

### HarmonyRenderer

- Anteriormente, las anclas perdían memoria al actualizar las transformaciones de ancla. Una actualización del plugin de C++ HarmonyRenderer resuelve este problema.
- Los HarmonyRenderers han mejorado el rendimiento de renderizado. Ahora recibirá nuevas mallas del plugin de C++ HarmonyRenderer que se pueden pasar directamente al backend de C++ de Unity utilizando las nuevas funciones de generación de mallas de Unity. Así se reduce de forma considerable el tiempo de actualización de las animaciones en los personajes de HarmonyRenderer, lo que permite tener más personajes en pantalla a la vez.

## Capítulo 2: exportación Harmony de datos de para juegos

Hay dos pipelines principales para exportar datos desde Harmony al motor de juego:

- [exportación Harmony de datos de para juegos](#) en la página 10
- [exportación Harmony de datos de para juegos](#) en la página 10



### Creando una animación en Harmony

Hay que tener en cuenta algunas cosas antes de empezar a crear rigs de personajes y animación para juegos. Artistas y programadores deben colaborar para asegurarse de que se cumplan sus necesidades.

- ¿Para qué plataformas se creará su juego? Windows, Mac, Móvil, iOS, PS, XBOX, etc.
- ¿Qué motor de juego utilizará?
- ¿Qué estilo de animación se utilizará para crear el aspecto del juego? Dibujo a mano, cut-out, con texturas, etc.

Estas son solo algunas de las preguntas que debemos plantearnos antes de empezar. Todas influyen en la forma de diseñar, crear y animar los personajes.

Por ejemplo, si tiene previsto crear un juego móvil para smartphone, lo más probable es que quiera que ocupe menos de 50 MB para que se pueda descargar sin tener que estar conectado a una red Wi-Fi. En ese caso, lo más importante es crear personajes eficientes con cartas de sprites muy ajustadas y reutilizar gran parte de la animación para que el tamaño de los archivos sea reducido. Durante el proceso, necesitará:

- Crear rigs y animar personajes en Harmony.
- Extraer los datos de Harmony.
- Importar los datos de Harmony en el motor de juego.

Tenga en cuenta que, si trabaja con un motor personalizado, también puede procesar los datos de Harmony exportados y utilizarlos en un motor de juego personalizado. Si necesita ayuda para adaptar los datos a su motor, póngase en contacto con [store.toonboom.com/contact/support](https://store.toonboom.com/contact/support).

Si está creando un juego para consolas como PS o Xbox, tiene libertad para crear texturas más grandes. Puede animar fotograma a fotograma, con un personaje cut-out, o de ambas formas.

Si simplemente va a exportar en una secuencia fotograma a fotograma, puede utilizar todas las herramientas de Harmony sin limitaciones. A continuación, puede procesar una secuencia de imágenes exportada a una carta de sprites.

## Exportación de datos de juego en bruto (Export to Sprite Sheet (Exportar a carta de sprites))



La exportación de datos de juego en bruto es adecuada cuando se desea que el tamaño de los archivos sea lo más pequeño posible. Toon Boom le permite convertir datos directamente desde la escena de Harmony para incorporarlos a un motor de juego. Puede extraer la información de esqueleto, información de dibujo, datos de animación de fotogramas clave, así como deformaciones (solo huesos y articulaciones), cuchilla, nodos de transparencia y columnas de tiempo.

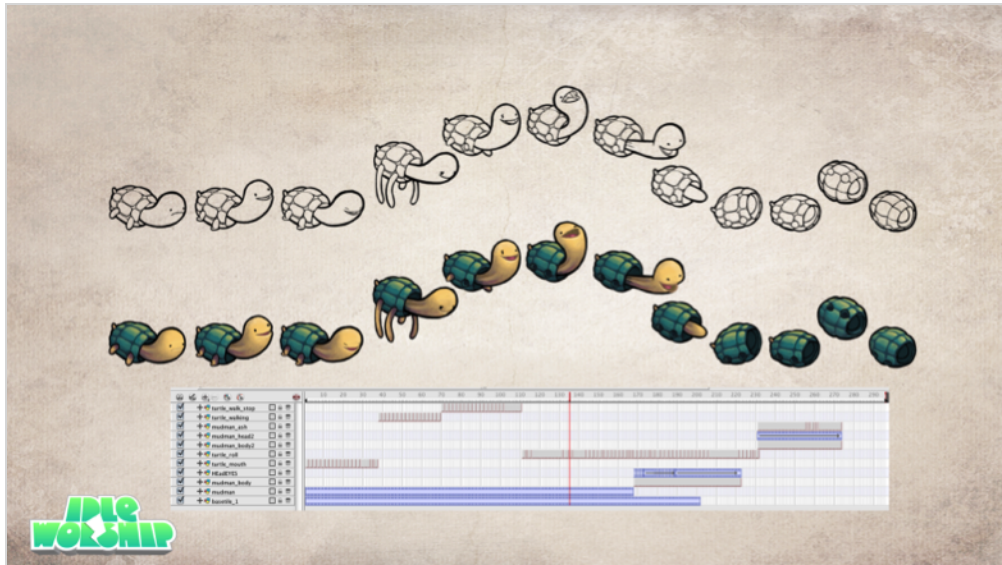
- **Ventaja:** esta es la exportación más ligera, y mantendrá un tamaño pequeño para los archivos, lo que es ideal para aplicaciones móviles.
- **Desventaja:** las herramientas que puede utilizar en Harmony son algo limitadas. Puede utilizar herramientas como el morphing, así como los deformadores de curvas o de distorsión de envoltorio, pero tendrá que desensamblarlos en dibujos para que se interpreten correctamente en el motor de juego. Puede utilizar efectos de cuchilla (enmascaramiento), pero no puede ponerlos en cascada, es decir, no puede tener más de uno en una cadena de jerarquía. Los deformadores de hueso de juego se pueden usar en su rig sin tener que ensamblarlos en dibujos.

Sin embargo, incluso con estas limitaciones, puede crear animaciones muy buenas de personajes cut-out en Harmony y extraer todos los datos compatibles. Al mover, rotar, escalar y sesgar las distintas capas de dibujo, puede crear animaciones con aspecto avanzado.

Cuando extraiga los datos, tendrá cartas de sprites que contendrán solo los dibujos de las partes del cuerpo utilizadas en su archivo de escena de Harmony. También puede admitir varias animaciones, como reposo, carrera y salto, reutilizando el mismo esqueleto y los mismos dibujos.

Toon Boom ha integrado plenamente esta solución con el motor de renderizado para desarrollo de juegos de Unity. Si está creando el juego en Unity, tiene un pipeline perfecto sin necesidad de volver a tratar los datos en el motor de juego.

## Exportación fotograma a fotograma (Export to Easel JS (Exportar a Easel JS))



Puede exportar desde Harmony utilizando una secuencia de imágenes, que puede volverse a compilar en una carta de sprites.

**Ventaja:** al animar, puede utilizar cualquier herramienta de Harmony.

**Desventaja:** estas cartas de sprites pueden llegar a ser muy pesadas y terminar como texturas grandes que hay que procesar en el juego. Esto puede estar bien para muchos juegos, como los de consola, pero cuando se desarrollan juegos para dispositivos móviles, como iOS o Android, el espacio de texturas debe ser muy ligero. Además, el tamaño de los archivos debe ser inferior a 50 MB para facilitar la descarga sin necesidad de estar conectado a una red Wi-Fi.

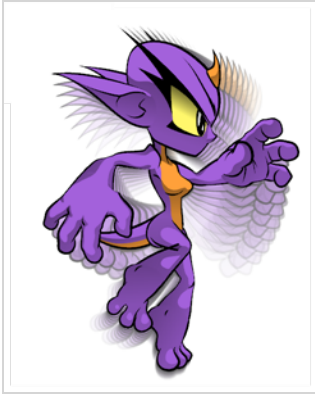
Hay un comando disponible en Harmony que permite exportar la animación fotograma a fotograma al motor de juego EaselJS. EaselJS es un motor de juego javascript compatible con web. Utiliza una estructura de datos json para asignar cada fotograma a una entrada de la carta de sprites. La estructura de datos es fácil de entender y, si es necesario, se puede adaptar o convertir a un motor de juego diferente.

Para exportar a EaselJS, debe agregar el comando en la barra de herramientas Scripting (Comandos) (TB\_ExportToEaselJS). Este comando está incluido en Harmony. Consulte la guía de comandos para obtener más información sobre cómo agregar el comando a la barra de herramientas Scripting (Comandos) y sobre los parámetros de la interfaz.



## Capítulo 3: Acerca de la creación de recursos de juego

En esta sección, encontrará directrices para el rigging, la deformación, la cuchilla y muchos consejos de animación útiles.



## Directrices de rigging de juego

A continuación, se ofrece una lista de directrices generales que debe tener en cuenta a la hora de crear un rig en el personaje. Cuando planifique el rig del personaje para el juego, tenga en cuenta el estilo del personaje y cree su paleta de colores personalizada. Sin embargo, debe tener en cuenta algunas limitaciones si piensa extraer los datos del juego:

- Configure la escena de Harmony para que tenga una resolución cuadrada (p. ej., 1024 x 1024). Puede hacerlo en el cuadro de diálogo Scene Settings (Configuración de escena); consulte la guía de referencia .
- Dibuje solo en las capas de línea y de color. La información contenida en las capas de superposición y de subcapa no se exportará a Unity. Sin embargo, se puede utilizar como capa de referencia, pero no con algo que se vaya a exportar.
- Empuje las capas al espacio Z si necesita reordenarlas. Sin embargo, no se admiten desplazamientos de Z significativos dentro de un rig de personaje.
- Haga que cada capa de su motor de juego sea una escena independiente en Harmony. Si tienes dos personajes a distinta profundidad, póngalos en archivos de escena separados.
- Los rigs de personajes en grupos en el nivel raíz de la escena de Harmony se renderizarán en un solo plano en Unity, pero utilizarán cartas de sprites y conjuntos de datos de animación separados. Téngalo en cuenta para las escenas en las que puede haber más de un personaje interactuando entre sí.
- No utilice el espacio 3D. Si desea configurar las cosas en el espacio 3D, puede hacerlo cuando llegue al motor de juego.
- Establezca los puntos de pivote en las capas de peg utilizando la herramienta Rotate (Rotar) para establecer el pivote en toda la capa. Se recomienda usar pivotes de peg sobre pivotes de dibujo. También debe establecer los puntos de pivote en las capas de dibujo, incluso si no anima en ellas, ya que le permitirá recuperar la información más adelante en el motor de juego si necesita poner un ancla en una capa de dibujo.
- No utilice morphing. Aún no se admite en los motores de juego.
- Tanto el morphing como las deformaciones se pueden usar y luego desensamblarlos en dibujos individuales. Estos dibujos aparecerán como dibujos nuevos en la carta de sprites. No lo haga con demasiada frecuencia, ya que aumentará el espacio de la textura.
- Preste atención a dónde coloca el personaje antes de exportarlo. El pivote maestro de su objeto de juego exportado será el centro de su escena en Harmony (0,0).
- Asegúrese de tener una pantalla al final de la jerarquía.
- Al hacer rigging, tenga en cuenta que el exportador no admite algunos nodos. Como resultado, los motores de juego no interpretarían bien las estructuras de nodos. Por lo tanto, se recomienda evitar el uso de estructuras de nodos complejas. Para hacerlo:
  - Construya su personaje con dibujos adjuntos a pegs. Los pegs pueden conectarse a otros pegs para formar jerarquías.
  - Debe tener todos los pegs conectados a un único peg maestro en la parte superior de la vista de nodos

- Por lo general, los dibujos deben conectarse a las composiciones para determinar el orden de los dibujos que aparecen en pantalla
- Si los dibujos deben cambiar de orden durante la animación, puede modificar la profundidad de z para cambiar qué dibujo se muestra primero
- Si no consigue que sus animaciones funcionen bien solo con dibujos y pegs, hay otros nodos que puede utilizar que se pueden exportar:
  - **Cuchillas:** por lo general, solo cortan un dibujo contra otro dibujo (p. ej., cortar una cara hasta los límites de una cabeza). No es posible conectar en cadena la salida de una cuchilla con la entrada de otra.
  - **Deformaciones de hueso de juego:** por lo general, solo deforman un dibujo. El esqueleto óseo puede ramificarse. Las salidas cinemáticas se pueden usar al adjuntar dibujos a los extremos de los huesos (p. ej., pie en el extremo de una pierna)



#### NOTA

Aún puede utilizar estructuras de nodos más complicadas dentro de «bake\_groups». Consulte [Uso de grupos de ensamblaje](#).

Si tiene en cuenta estos consejos, podrá crear un personaje de juego 2D ajustado y eficiente en Harmony, y sacar partido de todas las herramientas.

Estas son algunas de las cosas que debe hacer:

- Crear una jerarquía simple de relaciones padre-hijo en la vista de línea de tiempo.
- Utilizar capas de pegs para contener datos de animación de fotogramas clave, configuradas en posición separada.
- Utilizar capas de dibujo para dibujar sobre ellas y crear dibujos nuevos cuando sea necesario.
- Utilizar la herramienta Rotate (Rotar) para configurar los puntos de pivote en las capas de pegs.
- Nombrar las capas de forma adecuada para que, si necesita recuperar el punto de pivote de una capa específica más adelante en el motor de juego, puede reconocer fácilmente la capa que necesite. Si tiene un grupo A de nivel superior que tiene un grupo hijo dentro de él (grupo B) y la capa de dibujo es hija del grupo B, entonces la capa de dibujo se exporta como A\_B\_DrawingLayer.
- Establezca las anclas donde quiera que se muestren los pegs en Unity como transformaciones.

Puede utilizar cualquier herramienta de dibujo que desee: herramientas Pencil (Lápiz) y Brush (Pincel), líneas con textura, áreas sólidas y gradientes. Cada dibujo individual se renderizará y ensamblará más tarde en una carta de sprites.



#### NOTA

Dado que el motor de juego de Unity no admite caracteres Unicode, se recomienda evitar su uso en escenas concebidas para juegos.



## Directrices de deformación de juego

Harmony puede exportar las deformaciones al formato XML del motor de juego. Solo las jerarquías formadas por huesos de juego pueden exportarse de forma correcta. Además, se han implementado salidas cinemáticas para complementar los huesos de juego.

Las deformaciones en Harmony incluyen un amplio conjunto de funciones. Sin embargo, algunas de estas funciones no son compatibles con el SDK del motor de juego y no deben utilizarse para jugar. Por lo tanto, los deformadores de hueso de juego no admiten las siguientes características estándar del deformador de hueso:

- Zonas de influencia
- Deformaciones de curva o envolvente
- Tener más de una pose única en la jerarquía de deformación
- Animación de los pegs de dibujo bajo los grupos de deformación

La deformación en el SDK del motor de juego no se comporta exactamente como la deformación de hueso en Harmony.

Para cumplir con la mayoría de los motores de juego y mantener cálculos rápidos, el SDK implementa un algoritmo de máscara de base lineal para fusionar los huesos en las articulaciones. Es posible que note algunas diferencias en función de la curvatura de las articulaciones utilizadas. Dicho esto, en comparación con las deformaciones de hueso normales, los GameBones se asemejan más en Harmony a lo que se verá una vez que se importe a Unity.



### NOTA

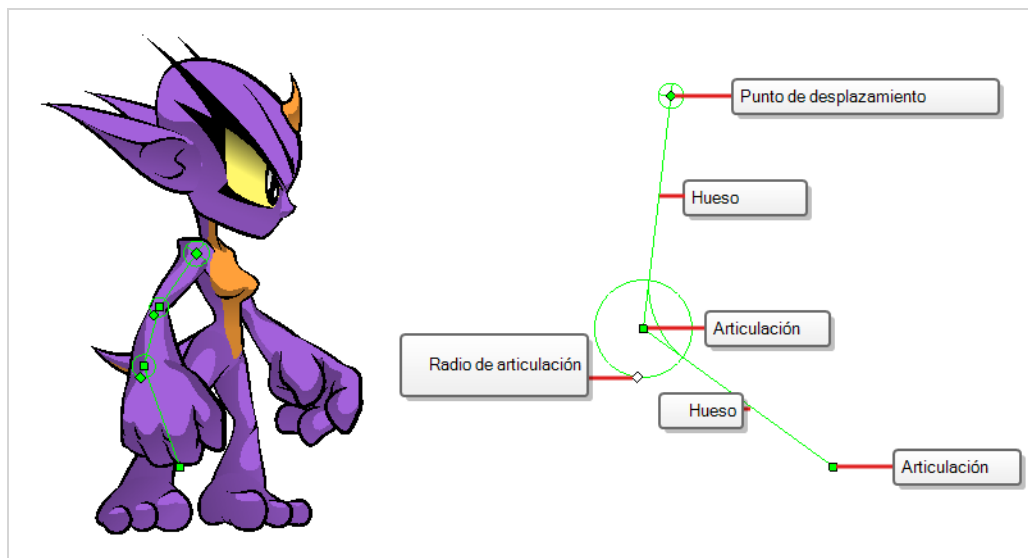
\* No disponible actualmente en la implementación Cocos2d-x del SDK del motor de juego.

## Acerca de las deformaciones de huesos de juego

T-RIG-007-003

La deformación de hueso para juego es parecida a la deformación de hueso. Permite crear una estructura similar al hueso en la que cada parte es sólida, pero las articulaciones son flexibles. Esto es sobre todo útil para la animación de los miembros de un personaje, como los brazos o las piernas, u otras partes que se puedan articular, como torsos o dedos. Por ejemplo, se puede utilizar una deformación de hueso de juego para articular un brazo que está hecho de un solo dibujo, de modo que el brazo y el antebrazo se puedan mover de forma independiente, sin necesidad de dibujar la parte superior del brazo y el antebrazo en capas diferentes. Harmony deformará el dibujo para que parezca que está articulado. Las distintas partes de una deformación de hueso de juego se pueden rotar alrededor de su articulación, extenderse y acortarse, proporcionándole las mismas capacidades que las articulaciones de animación en distintas capas, sin tener que preocuparse por la separación de piezas, puntos de pivote o recorte de contornos.

La deformación de hueso de juego es diferente de la deformación de hueso en la que se optimiza para motores de juego como Unity. Por lo tanto, generalmente se usa para desarrollo de juegos y no en producciones animadas. Las diferencias entre las deformaciones de hueso y las de hueso de juego son que las deformaciones de hueso de juego no tienen una propiedad de sesgo ni de región de influencia. Los pliegues de la articulación también parecen ligeramente más redondos.



---

## Directrices de la cuchilla de juego

La operación de cuchilla, o máscara, en Harmony se utiliza para recortar dibujos con formas personalizadas. El SDK del motor de juego implementa cuchillas y cuchillas invertidas con las siguientes limitaciones:

- Para un sprite, solo se puede aplicar un único dibujo de una cuchilla cuando se renderiza. Esto también se aplica a una composición de varios dibujos de mate. El SDK del motor de juego solo utilizará el primer dibujo de mate durante el renderizado y descartará los demás.
- Un dibujo deformado no se puede cortar, pero un dibujo cortado sí se puede deformar.



### NOTA

No disponible actualmente en la implementación Cocos2d-x del SDK del motor de juego.

## Creación de notas de metadatos

A medida que cree recursos para el juego en Harmony, es posible que quiera tomar notas sobre la escena o sobre partes específicas del personaje o props para el programador. Estas notas incrustadas que se exportarán con sus recursos a Unity se conocen como metadatos.

### Cómo acceder a la vista de editor de metadatos

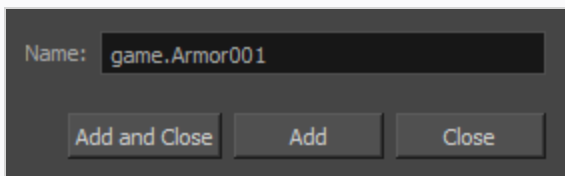
- En la esquina superior derecha de una vista, haga clic en el botón Add View (Agregar vista) **+** y seleccione **Metadata Editor (Editor de metadatos)**.
- En el menú superior, seleccione **Windows > Metadata Editor (Ventanas > Editor de metadatos)**.

### Cómo crear metadatos de escena

1. En la vista de editor de metadatos, en la sección Scene Metadata (Metadatos de escena), haga clic en el botón más **+** para crear una nueva entrada de metadatos.

Aparecerá el cuadro de diálogo Add Metadata (Agregar metadatos).

2. En el cuadro de diálogo Add Metadata (Agregar metadatos), introduzca el nombre de la nueva entrada de escena.



#### NOTA

Para cualquier información de ancla o prop creada en el editor de metadatos, la convención de nomenclatura debe utilizar siempre el prefijo «game.». Unity reconoce automáticamente el término «game.» como información de metadatos. Por ejemplo: *game.Armor001*.

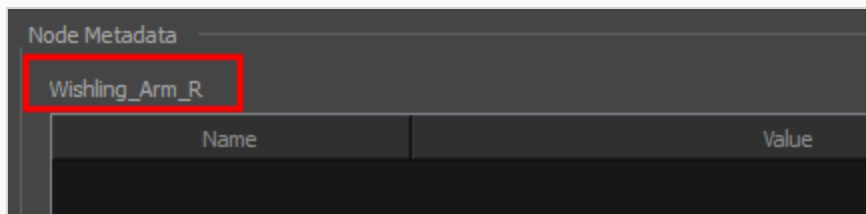
3. Haga clic en **Add and Close (Agregar y cerrar)** si solo desea agregar una entrada. Haga clic en **Add (Agregar)** si desea agregar varias entradas. Continúe agregando varias entradas, haciendo clic en **Add (Agregar)** después de cada una. Haga clic en **Close (Cerrar)** cuando haya terminado.
4. En la vista de editor de metadatos, haga doble clic en el campo de valor de la primera capa para que se pueda editar.
5. Introduzca la información de valor para esta entrada.
6. Continúe agregando información de valor para todas las entradas.

Esta información se exportará con los recursos de Harmony. Una vez en Unity, los metadatos de escena aparecerán en Inspector View > Metadata (Vista del Inspector > Metadatos) cuando el recurso se seleccione en la vista de jerarquía.

## Cómo crear metadatos del nodo

1. En la vista de línea de tiempo, haga clic en la capa a la que desea adjuntar metadatos.

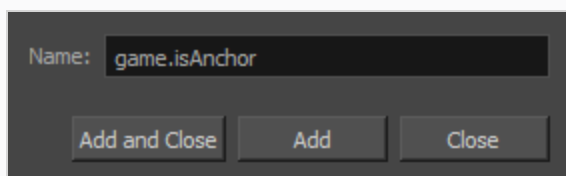
El nombre de la capa aparece cerca de la parte superior de la sección de metadatos del nodo.



2. En la sección Node Metadata (Metadatos del nodo), haga clic en el botón más **+** para crear una nueva entrada de metadatos.

Aparecerá el cuadro de diálogo Add Metadata (Agregar metadatos).

3. En el cuadro de diálogo Add Metadata (Agregar metadatos), introduzca el nombre de la nueva entrada de nodo.



### NOTA

Para cualquier información de ancla o prop creada en el editor de metadatos, la convención de nomenclatura debe utilizar siempre el prefijo «game.». Unity reconoce automáticamente el término «game.» como información de metadatos. Por ejemplo: *game.Armor001*.

4. Haga clic en **Add and Close (Agregar y cerrar)** si solo desea agregar una entrada. Haga clic en **Add (Agregar)** si desea agregar varias entradas. Continúe agregando varias entradas, haciendo clic en **Add (Agregar)** después de cada una. Haga clic en **Close (Cerrar)** cuando haya terminado.
5. En la vista de editor de metadatos, haga doble clic en el campo de valor de la primera capa para que se pueda editar.
6. Introduzca la información de valor para esta entrada.
7. Continúe agregando información de valor para todas las entradas.

Esta información se exportará con los recursos de Harmony; en concreto, esta información estará vinculada a la capa seleccionada. Una vez en Unity, los metadatos de escena aparecerán en Inspector View > Metadata (Vista del Inspector > Metadatos) cuando el recurso se seleccione en la vista de jerarquía.

## Consejos de animación de juegos

Al animar para juegos, dependiendo del tipo de juego que esté creando, puede que necesite limitar su animación. Por ejemplo, si crea juegos para móviles y quiere que el tamaño de los archivos sea pequeño y la reproducción rápida en todos los dispositivos, límitese a una animación de fotogramas clave sencilla con el menor número posible de cambios de dibujo. Si crea juegos para consola, tiene libertad para crear más dibujos con mayor complejidad. Puede estudiar con los desarrolladores las limitaciones de las plataformas a las que exporta y lo que admite su motor de juego.

Aquí puede encontrar algunos consejos para una animación eficiente y ligera:

- Utilice principalmente transformaciones, como mover, rotar, escalar y sesgar.
- Cree cambios de dibujos adicionales cuando sea necesario.
- Si utiliza deformadores de curva y de distorsión de envoltorio o morphing, tendrá que desensamblar los dibujos para exportarlos. Tenga cuidado al hacerlo, ya que es posible que desee mantener un número de dibujos reducido. No desensamble una secuencia entera, sino solo dibujos seleccionados. No necesita ensamblar los deformadores de hueso de juego.
- Cuanto más grandes sean los dibujos en la vista de dibujo, más píxeles ocuparán en el tamaño de textura en la carta de sprites. Cuando configure su rig, asegúrese de no escalar capas individuales utilizando un fotograma clave con la herramienta Transform (Transformar). Si desea ampliar o reducir, utilice la herramienta Select (Seleccionar). De este modo, las cosas mantendrán el mismo tamaño relativo en la carta de sprites. Cuando exporte las cartas de sprites, también puede establecer la resolución de la carta de sprites en el comando para que los dibujos puedan reducirse para dispositivos más pequeños.
- Solo se exportarán a la carta de sprites los dibujos que estén expuestos en la escena. Por ejemplo, si tiene 10 dibujos en la vista de librería, pero solo aparecen en escena dos de ellos, tan solo se exportarán esos dos. Así se mantiene la carta de sprites lo más ajustada posible.

---

## Animación de varias secuencias

Lo más probable es que tenga varias animaciones para sus personajes. Por ejemplo, una secuencia de reposo, una secuencia de carrera, una secuencia de acción, etc. Necesita trabajar en una estructura específica para poder exportar todas estas animaciones a una única carta de sprites.

Hay dos flujos de trabajo diferentes que puede usar:

- Flujo de trabajo 1: escenas separadas
- Flujo de trabajo 2: separación mediante marcadores de escena

### Flujo de trabajo 1: escenas separadas

En primer lugar, cree un archivo de escena con el nombre del personaje, como **Space Duck**. Este es el archivo en el que puede crear o importar su rig de juego. En el menú superior, seleccione **File > Save as New Version (Archivo > Guardar como nueva versión)**, y póngale a esta nueva versión el nombre de la animación. Por ejemplo, **Idle**.

Cada vez que necesite hacer una nueva animación utilizando el mismo personaje, elija **Save As New Version (Guardar como nueva versión)**. Al finalizar, tendrá algo como esto:

**Escena:** Space Duck

**Versiones:**

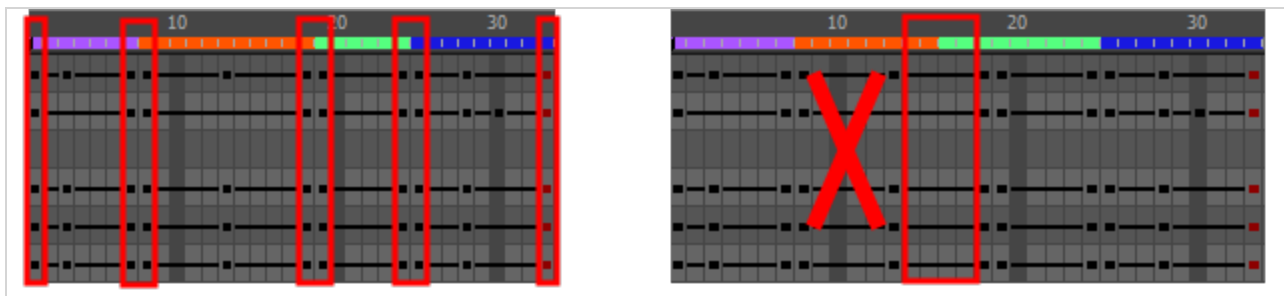
- Reposo
- Correr
- Saltar
- Disparar

Cuando ejecute el comando de exportación, exportará los dibujos de la escena actual a la carpeta de exportación. También le indicará si hay otras versiones de escenas que ya se hayan exportado a esa carpeta. Si es así, la carta de sprites se volverá a compilar para incluir todos los dibujos de todas las animaciones. Esto permite que se reutilicen los dibujos el máximo posible.

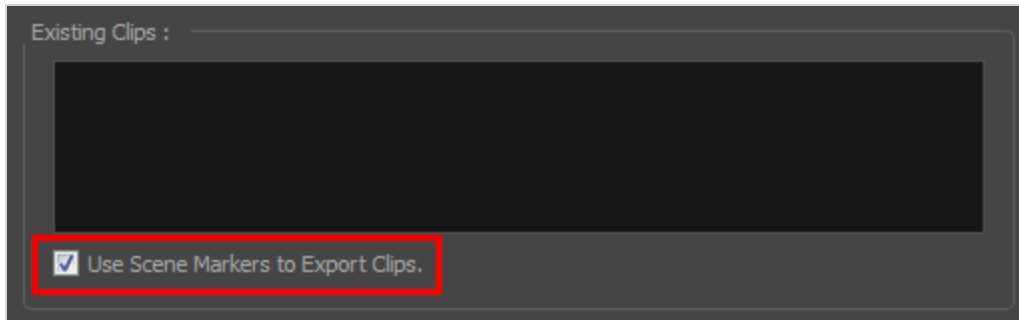
### Flujo de trabajo 2: separación mediante marcadores de escena

También puede crear todas las animaciones de sus personajes en una única escena, una tras otra, como reposo, correr, saltar y disparar. Luego, utilice los marcadores de escena para marcar y separar las animaciones individuales.

Cuando marque los rangos de fotogramas de animación individuales, asegúrese de que empiecen y terminen con un fotograma clave. No cree marcadores de escena para un rango de fotogramas que comience o termine en mitad de un movimiento interpolado.



Cuando exporte su carta de sprites, en el cuadro de diálogo Export To Sprite Sheet (Exportar a carta de sprites), asegúrese de marcar la opción **Use Scene Markers to Export Clips (Utilizar marcadores de escena para exportar clips)**. Esta opción está activada de forma predeterminada.



Los clips animados se dividen y se enumeran en stage.xml de la misma manera que aparecerían si hubiera exportado cada secuencia animada de escenas separadas a la misma ubicación de archivo.



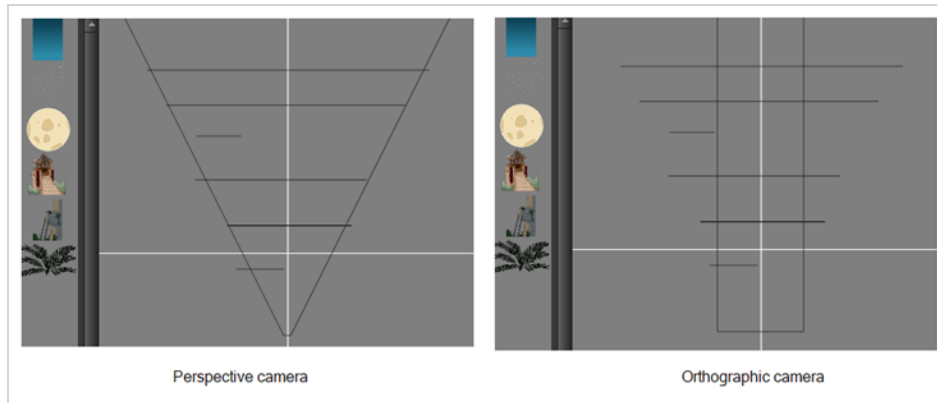
---

## Acerca de la cámara ortográfica

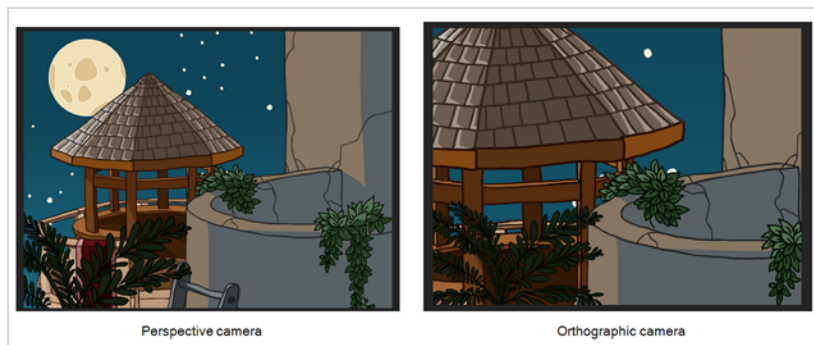
In Harmony, there are two types of cameras available:

- Perspective
- Orthographic

The orthographic camera is specific to the gaming pipeline. It changes the camera type from perspective to orthographic. It becomes a camera without vanishing points. This means there is no more perspective in the Camera view. Objects, when moved on the Z-axis, will not change in size or scale.



The orthographic camera can be set in the Scene Settings dialog box. In order to create scenes with the orthographic camera, by default, you need to create a new custom scene resolution.



## Capítulo 4: Acerca de la exportación a Unity

Si tiene un motor de juego personalizado, puede tomar los datos de Harmony exportados y seguir con su proceso habitual. O puede modificar los comandos de exportación para adaptarlos a sus necesidades.



Antes de exportar una escena:

- Asegúrese de configurar la pantalla como **Display (Visualizar)** y no como Display All (Visualizar todo).
- Asegúrese de guardar su escena. Harmony realiza la exportación a partir de los archivos tvg; las actualizaciones no guardadas que haya agregado no se exportarán.
- Configure la escena de Harmony para que tenga una resolución cuadrada (p. ej., 1024 x 1024). Seleccione Scene > Scene Settings (Escena > Configuración de escena).

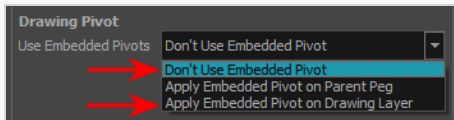
## Configuración de anclas


Antes de exportar a Unity, se debe plantear qué establecer como anclas.

Un ancla es un punto del personaje al que desea hacer referencia en el motor de juego durante la animación del personaje.

### Cómo establecer un ancla

1. Cree una nueva capa de dibujo y cámbiele el nombre según el ancla que esté creando.
2. Dentro de las propiedades de capa de su capa de dibujo, asegúrese de que la opción de pivote esté establecida en **Don't Use Embedded Pivot (No usar pivote incrustado)** o **Apply Embedded Pivot on Drawing Layer (Aplicar pivote incrustado en la capa de dibujo)**. La opción Apply Embedded Pivot on Parent Peg (Aplicar pivote incrustado en peg padre) no se transfiere bien a Unity y perderá la información del pivote.



3. Una vez hecho esto, puede utilizar cualquiera de las herramientas de animación para cambiar la posición del punto de pivote de la capa, como la herramienta Rotate (Rotar) de la barra de herramientas Advanced Animation (Animación avanzada). Debe poder encontrar las coordenadas de pivote a partir de las propiedades de capa de la capa de dibujo. Si no se cambia la posición del pivote, su lugar predeterminado será el centro de la escena de Harmony (0,0).
4. Una vez configurado, seleccione la capa de dibujo y haga clic en el botón Toggle Anchor (Alternar ancla)  en la barra de herramientas Game (Juego). También puede crear una selección múltiple para establecer un ancla en varias capas de dibujo de forma simultánea.

La capa de dibujo se destaca en rojo en la vista de línea de tiempo y de nodo. Ahora puede utilizar el ancla para adjuntar un prop.

## Exportación de cartas de sprites

La ventana Export to Sprite Sheets (Exportar a cartas de sprites) exporta a varias resoluciones, generando varios archivos .xml y una o varias cartas de sprites dependiendo de cuántas resoluciones de sprites haya definido.

Así se guardan diferentes animaciones del mismo personaje en el mismo nombre. Por ejemplo, si hay una animación de reposo, una de correr y una de saltar, todas deben compartir el mismo nombre de recurso. Puede verlo como la recogida global de animaciones. Dentro están las diferentes versiones de escenas guardadas cuyos dibujos puede reutilizar para todas las animaciones de ese conjunto de personajes. Cada versión de escena se mostrará como un elemento en la lista.

Cuando se exporta una animación, solo se exportan los dibujos utilizados en esa escena. Todos los dibujos se exportan primero de forma individual y luego se unen en un atlas en una carta de sprites.

Si ha guardado varias animaciones con el mismo nombre de recurso (es decir, SpaceDuck: correr, reposo), volverá a unir en atlas la carta de sprites para incluir todos los dibujos de todas las animaciones de esa carpeta, creando un nuevo archivo de animación, pero reutilizando el mismo esqueleto.



### NOTA

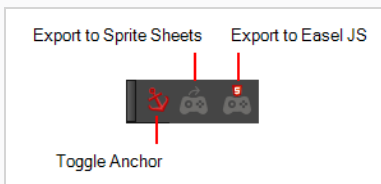
Programmers may be interested to note that the XML data exported by Harmony can be converted afterwards to a more optimized binary data structure. To convert XML to Binary format, use the Xml2Bin utility. This utility converts the XML data structure generated through the Toon Boom Harmony software to a compressed binary data structure. This utility is available in the gaming SDK under /HarmonyGameSDK/Plugins/.


- Plugins/Mac:-Precompiled binary for Mac OSX.
- Plugins/Windows\x86: Precompiled binary for Windows.
- Samples/HarmonyGameSDKSource/Utils/Xml2Bin: Xml2Bin sources.
- Samples/HarmonyGameSDKSource/Utils/Xml2Bin/proj.mac.Xml2Bin.xcodeproj: XCode project for Mac OSX.
- Samples/HarmonyGameSDKSource/Utils/Xml2Bin/proj.win32/Xml2Bin.sln: Visual Studio 2010 solution for Windows.

The C++ code that handles the data structure can be reused and parsed in your own code if you want to integrate with other game engines.

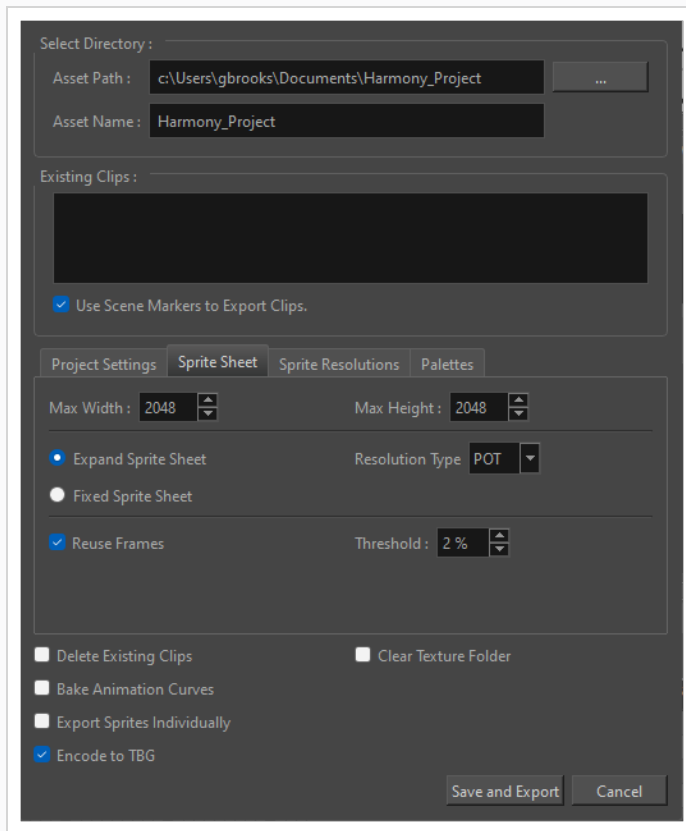
## Cómo exportar cartas de sprites

1. En la barra de herramientas Game (Juego), al seleccionar **Windows > Toolbars > Game (Ventanas > Barras de herramientas > Juego)**.



2. Ejecute el comando haciendo clic en el botón Export to Sprite Sheets (Exportar a cartas de sprites)  en la barra de herramientas Game (Juego).

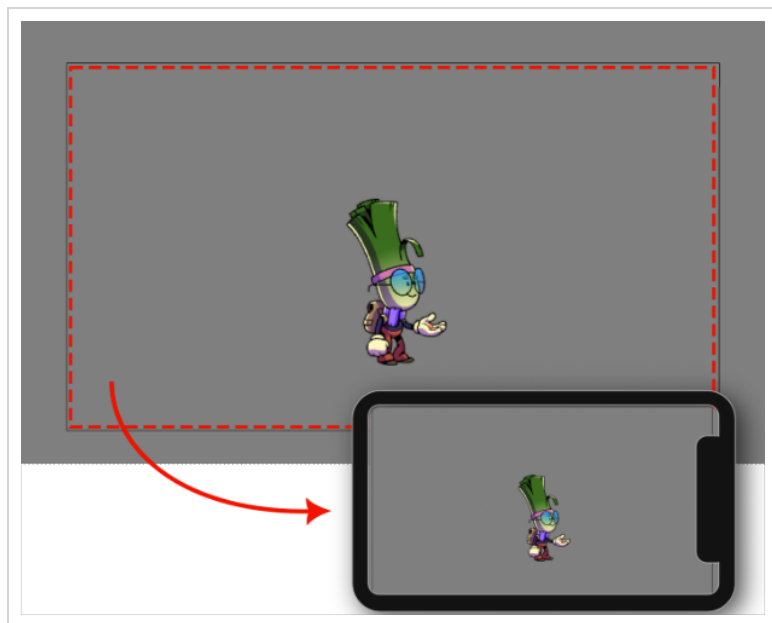
Se abrirá la ventana Export to Sprite Sheets (Exportar a cartas de sprites).



3. Establezca la ruta de almacenamiento en la carpeta correcta del proyecto de Unity si desea que se actualice de forma automática. Si no, puede guardar en cualquier lugar y luego transferir la animación a su proyecto de Unity o a cualquier otro motor de juego que desee utilizar.
4. Configure las preferencias de su carta de sprites.
5. Haga clic en **Export (Exportar)**.

## Resoluciones de sprites

Puede definir el número de píxeles de un sprite en función de la resolución objetivo del juego. Su resolución objetivo puede basarse en el hardware al que se dirige, como la pantalla de un televisor o un teléfono.




Se creará una nueva carta de sprites para cada resolución de destino especificada en la pestaña Resolutions (Resoluciones) de la ventana Export to Sprite Sheets (XML) (Exportar a cartas de sprites (XML)). Para acceder a esta ventana, debe activar la barra de herramientas Game (Juego) seleccionando **Windows > Toolbars > Game (Ventanas > Barras de herramientas > Juego)**.



Tenga en cuenta que el tamaño del sprite será el tamaño del personaje cuando se creó en Harmony.

### Cómo configurar varias resoluciones de sprites

1. Habilite la barra de herramientas Game (Juego) seleccionando **Windows > Toolbars > Game (Ventanas > Barras de herramientas > Juego)**.
2. Seleccione Export to Sprite Sheets (Exportar a cartas de sprites). 🛠️ Se abrirá la ventana Export to Sprite Sheet (Exportar a carta de sprites).

3. En el cuadro diálogo Export to Sprite Sheet (XML) (Exportar a carta de sprites (XML)), seleccione la pestaña Sprite Resolutions (Resoluciones de sprite). En esta pestaña se mostrarán todas las resoluciones objetivo actuales.
4. Haga clic en el icono Add (Agregar) . Se creará una nueva fila para una nueva resolución de destino.
5. Introduzca el nombre, el ancho y la altura de la nueva resolución de destino.

Las resoluciones de destino predeterminadas son HD, FullHD y QuadHD. Las resoluciones de pantalla indicadas representan el dispositivo de destino en el que se podría reproducir.

Puede agregar o eliminar de forma manual resoluciones de esta lista en función de la resolución de destino y el hardware del que disponga. Una vez importada en el motor de juego, será posible seleccionar qué resolución se está mostrando al cambiar la carta de sprites renderizada.



#### NOTAS

- Las resoluciones de sprites se ven afectadas por la escala de las pinceladas de dibujo.
- El escalado de dibujos o sus pegs no afectará a la resolución de los sprites de salida.
- Si desea aumentar la resolución de sprites, puede escalar las capas hijas de un dibujo.

## Uso de Bake\_Groups

Un Bake\_Group es una agrupación de nodos de dibujo y efectos que se pueden exportar juntos como un único dibujo unificado en una carta de sprites.

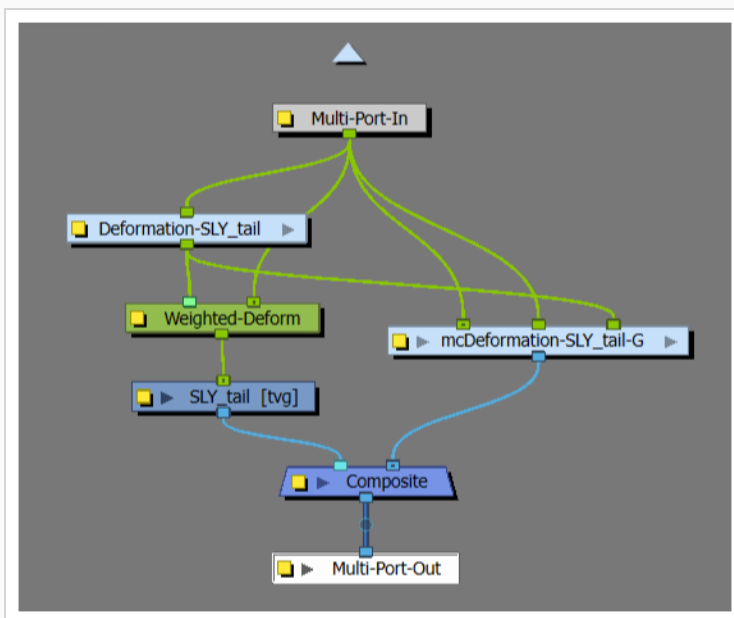
### Cómo crear un grupo ensamblado

1. Agrupe las capas que vaya a ensamblar. Se recomienda agrupar las capas desde la vista de nodos.



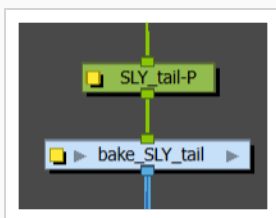
#### NOTA

Un bake\_group tan solo debe tener una entrada y una salida. Esto generalmente significa que un nodo Composite (Composición) se debe incluir dentro del bake\_group para que la exportación funcione correctamente. Considere también qué pegs se necesitan para guiar los dibujos, ya que todos los pegs deben proceder al final de un único peg fuera del grupo.



2. En la línea de tiempo, agregue el prefijo «bake\_» delante del nombre del grupo.

Vea en el ejemplo siguiente qué aspecto debe tener un nodo de grupo con el prefijo «bake\_». El «grupo» etiquetado como «hips» (caderas) debe convertirse en «bake\_hips» para que se aplique el ensamblaje.





Si se incluye el prefijo «bake\_» en el nombre del grupo, este se analizará en busca de fotogramas que contengan efectos únicos, movimientos y sustituciones de dibujos, y cada fotograma único se exportará como un sprite en la carta de sprites.

Se crearán nuevos dibujos animados para estos sprites generados y se proporcionarán al motor de juego. El motor de juego no tendrá acceso a ninguna de las estructuras dentro de este grupo; en su lugar, tratará a todo el grupo como si fuera un único dibujo con sustituciones de dibujo.


## Exportación a Easel JS

La ventana Export to Easel JS (Exportar a Easel JS) le permite aplanar una secuencia de imágenes de su animación. Incluso si tiene una marioneta totalmente en rig o una sola capa de dibujo con su secuencia de animación, el resultado seguirá siendo una salida aplanada de cada fotograma agrupada en su carta de sprites. Esto permite una mayor flexibilidad y libertad de trabajo, ya que tiene acceso a cualquier herramienta o módulo de efectos que desee utilizar. Sin embargo, puede dar lugar a archivos más pesados en función de la longitud, la complejidad y el tamaño de exportación de la animación.

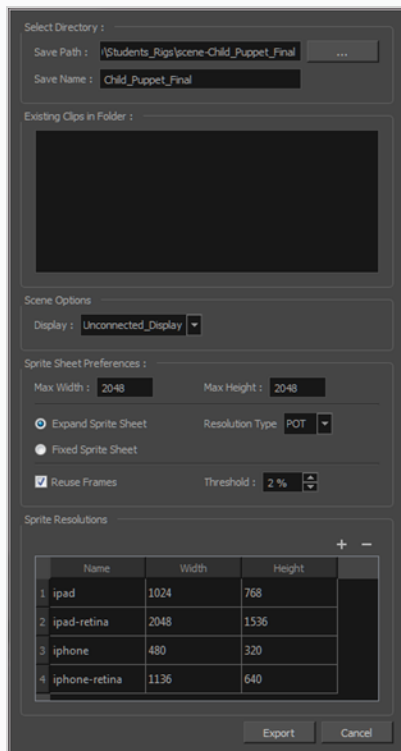
### Cómo exportar a Easel JS

1. En la barra de herramientas Game (Juego), al seleccionar **Windows > Toolbars > Game (Ventanas > Barras de herramientas > Juego)**.



2. Ejecute el comando haciendo clic en el botón Export to Easel JS (Exportar a Easel JS)  en la barra de herramientas Game (Juego).

Se abre la ventana Export to Easel JS (Exportar a Easel JS).



3. Establezca la ruta de almacenamiento en la carpeta correcta del proyecto de Unity si desea que se actualice de forma automática. Si no, puede guardar en cualquier lugar y luego transferir la

animación a su proyecto de Unity o a cualquier otro motor de juego que desee utilizar.

4. Configure las preferencias de la carta de sprites.
5. Haga clic en **Export (Exportar)**.

## Variaciones de paletas

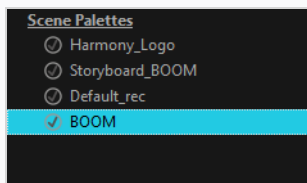
La creación de versiones de personajes con diferencias en la paleta es importante cuando tenga previsto agregar la personalización de personajes como una característica de su juego. En Harmony, puede crear varias paletas para un personaje, de modo que pueda tener distintos aspectos en el juego.


En Harmony, una vez creado el personaje y sus paletas, puede exportar el personaje a cartas de sprites para cada paleta.



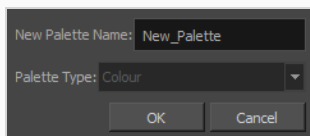
### Cómo crear variaciones de paletas para el personaje de un juego

1. In the Colour view, select a palette to clone.



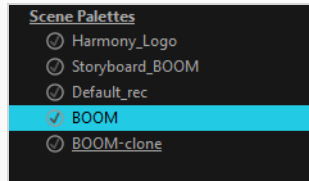
2. Do one of the following: From the Colour menu, select **Palettes > Clone** or right-click and select **Clone**.
  - Right-click on the palette and select **Clone**.
  - Open the Colour menu , then select **Palettes > Clone**.

The Clone palette dialog box opens




3. In the **New Palette Name** field, enter the name for the new palette. By default, the palette's name is based on the original palette name followed by "-clone". The palette list above this field lists the palettes that already exist in the selected location. Make sure you enter a name that is not already used by another palette.
4. Click **OK**.

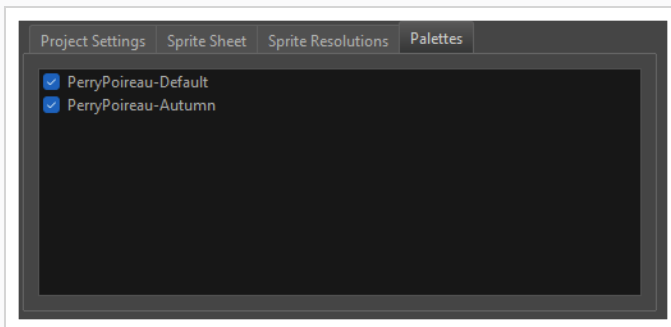
The cloned palette appears in the palette list.



All Scene Palettes will appear in the Palettes tab of the Export to Sprite Sheet window.

## Cómo exportar una carta de sprites para cada variación de paleta

1. Habilite la barra de herramientas Game (Juego) realizando una de las siguientes acciones:
  - En el menú superior, seleccione Windows > Toolbars > Game (Ventanas > Barras de herramientas > Juego).
  - Haga clic con el botón derecho en cualquier barra de herramientas existente y seleccione **Game (Juego)**.
2. En la barra de herramientas Game (Juego), seleccione Export to Sprite Sheets (Exportar a cartas de sprites) .
3. Abra la pestaña Palettes (Paletas). Verá todas las paletas de su proyecto y sus clones.



### NOTA

Solo aparecerán las paletas y sus clones. Los duplicados de paletas no aparecerán.

4. Seleccione los nombres de las paletas que desea exportar. Cada paleta marcada se exportará con el personaje como su propia carta de sprites.
5. Exporte su carta de sprites.

Una vez importado el personaje a Unity, se podrá seleccionar qué paleta se muestra cambiando la carta de sprites renderizada.

## Capítulo 5: Acerca del Harmony Unity SDK

Toon Boom proporciona un paquete de Unity en la Unity Asset Store que contiene todos los comandos necesarios para importar los datos exportados desde Harmony. Una vez desempaquetado en un nuevo proyecto de Unity, la siguiente estructura está disponible dentro de la carpeta Toon Boom Harmony Gaming SDK:

- Documentación (información para desarrolladores sobre la integración de Unity)
- Plugins (librerías de SDK)
- Muestras (escenas de demostración, comandos de exportación, fuente del SDK y un generador de vista previa)
- Tiempo de ejecución (comandos de Harmony que se ejecutan durante el juego)
- Editor (comandos de Harmony que solo se ejecutan en el editor y no aparecen en las compilaciones)
- Shaders (proporciona shaders que renderizan correctamente las características específicas de Harmony)
- Materiales (materiales previamente creados que utilizan shaders de Harmony)

Estos paquetes se pueden gestionar de forma manual a través de la interfaz del administrador de paquetes de Unity. Los desarrolladores pueden cambiar los paquetes a una versión más reciente o más antigua en caso de que las nuevas versiones de Unity tengan ciertas incompatibilidades de paquetes con los comandos de Harmony.

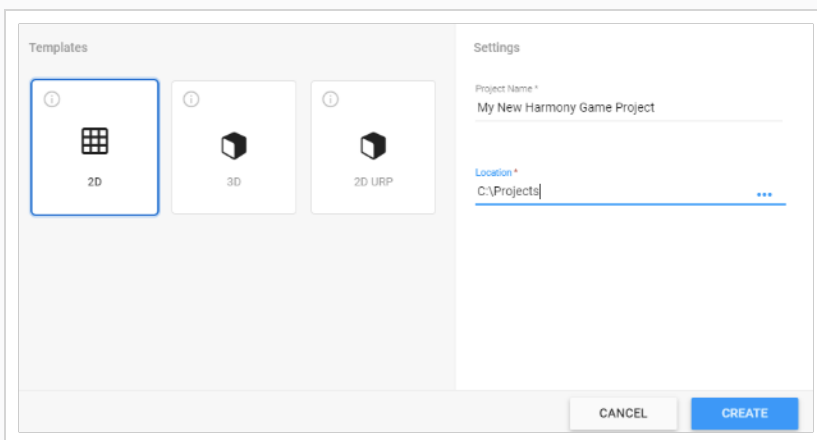
### Cómo crear un proyecto de Unity SDK

1. Cree un proyecto de juego de Unity vacío. Se recomienda crear un proyecto de juego 2D de Unity vacío, ya que es lo más conveniente cuando se trabaja en renderizado y física 2D.



#### NOTA

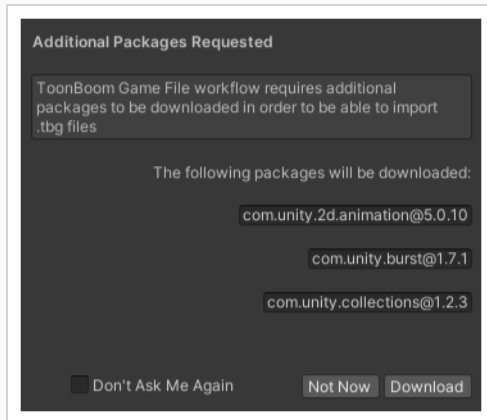
Asegúrese de estar utilizando la versión 2020.3.36 o posterior de Unity, ya que las versiones anteriores de Unity no tienen todas las características necesarias para dar soporte a la importación y el renderizado de archivos TBG.



2. Instale el Harmony Game SDK disponible en la Unity Asset Store. Si su SDK se descargó de forma manual, puede hacer doble clic en el archivo .unitypackage mientras su proyecto de Unity está abierto.

La ventana emergente Additional Packages Requested (Paquetes adicionales solicitados) aparecerá para asegurarse de que tiene paquetes de Unity compatibles que faciliten la animación 2D optimizada en tiempo de ejecución.

3. Haga clic en **Download (Descargar)**.



Una vez que Unity termine de descargar los paquetes necesarios y de compilar todos los comandos, su proyecto estará listo para comenzar a importar personajes de Harmony.

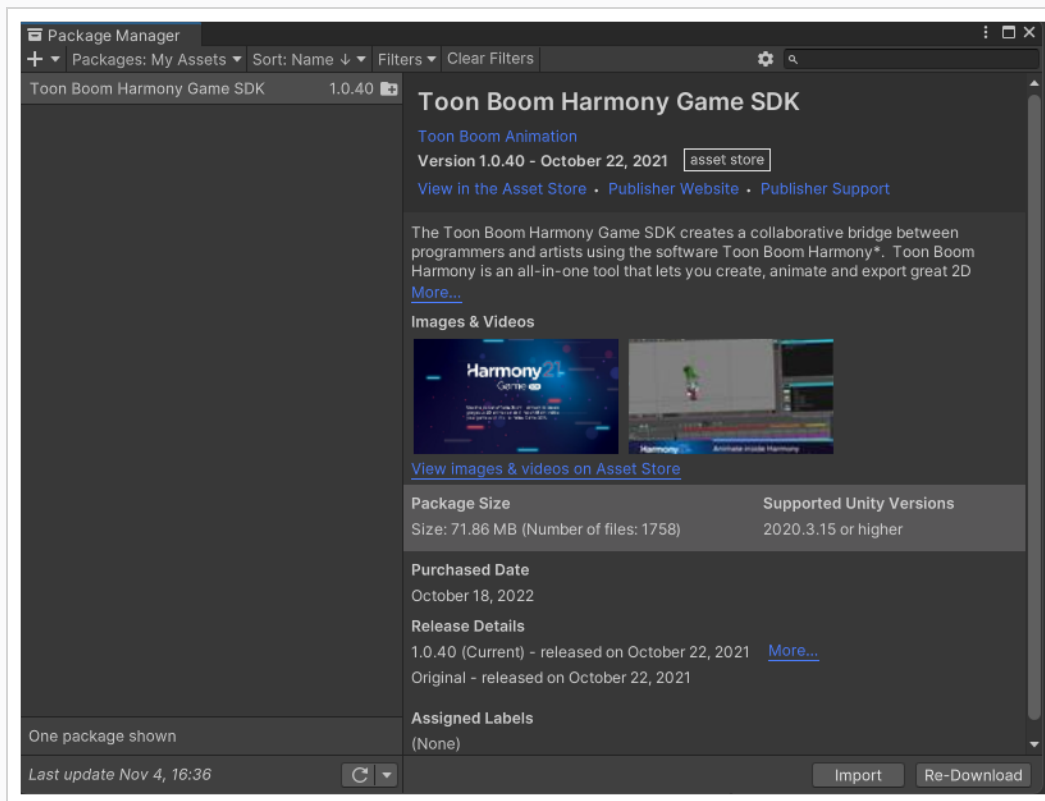


#### NOTAS

Cuando haya instalado el Toon Boom Harmony Game SDK, se deberá importar el paquete Harmony SDK en cada nuevo proyecto creado en Harmony.

### Cómo importar un paquete a un proyecto

1. Abra la tienda de recursos seleccionando Window > Asset Store (Ventana > Tienda de recursos). Se abrirá la pestaña Asset Store (Tienda de recursos).
2. En el menú Asset Store (Tienda de recursos), seleccione **Open Package Manager (Abrir administrador de paquetes)**. Se abrirá la ventana Package Manager (Administrador de paquetes).



3. En el menú desplegable Packages (Paquetes), situado en la parte superior izquierda del administrador de paquetes, habrá cuatro opciones:

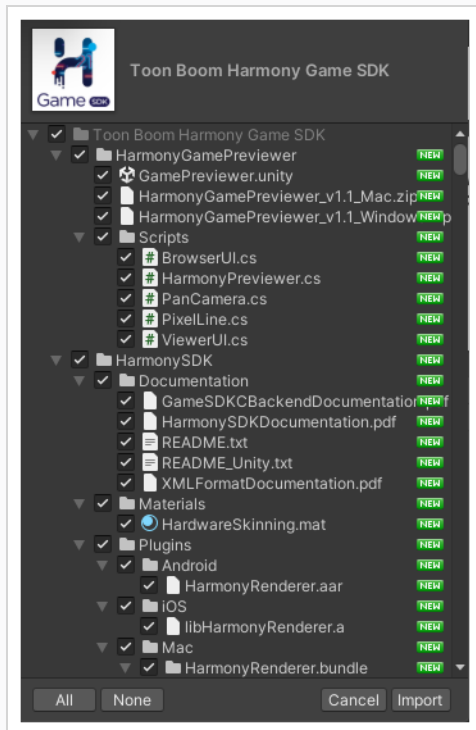
- Packages: Unity Registry (Paquetes: Registro de Unity)
- Packages: In Project (Paquetes: En el proyecto)
- Packages: My Assets (Paquetes: Mis recursos)
- Packages: Built-in (Paquetes: Incorporados)

4. Seleccione **Packages: My Assets (Paquetes: Mis recursos)**.

Se mostrará una lista con sus recursos descargados a continuación. A la derecha de la pestaña Package Manager (Administrador de paquetes), aparecerá información sobre el recurso seleccionado.

5. Cuando haya seleccionado el paquete de Harmony, seleccione Import (Importar) en la parte inferior del administrador de paquetes. Aparecerá la ventana Import Unity Package (Importar paquete de Unity) para mostrar los archivos del recurso.





## 6. Selección Import (Importar).

## Generador de vista previa de Harmony

En esta sección se explica cómo utilizar el generador de vista previa de Harmony para ver animaciones sin necesidad de instalar Unity.

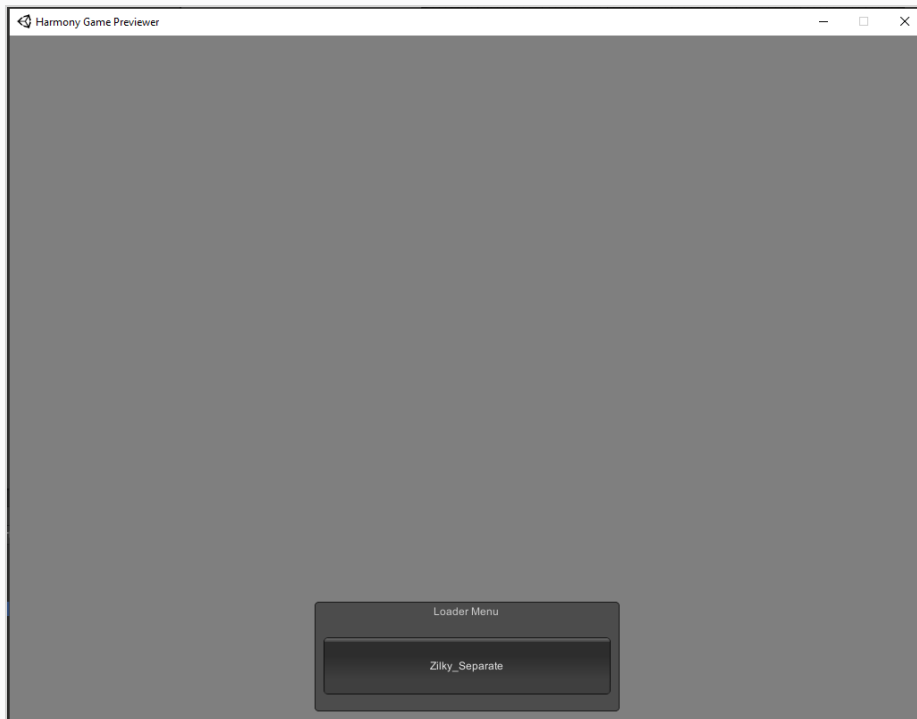
Puede encontrar una compilación para Windows o macOS dentro de la carpeta Harmony Game Previewer que solo tiene que descomprimir.

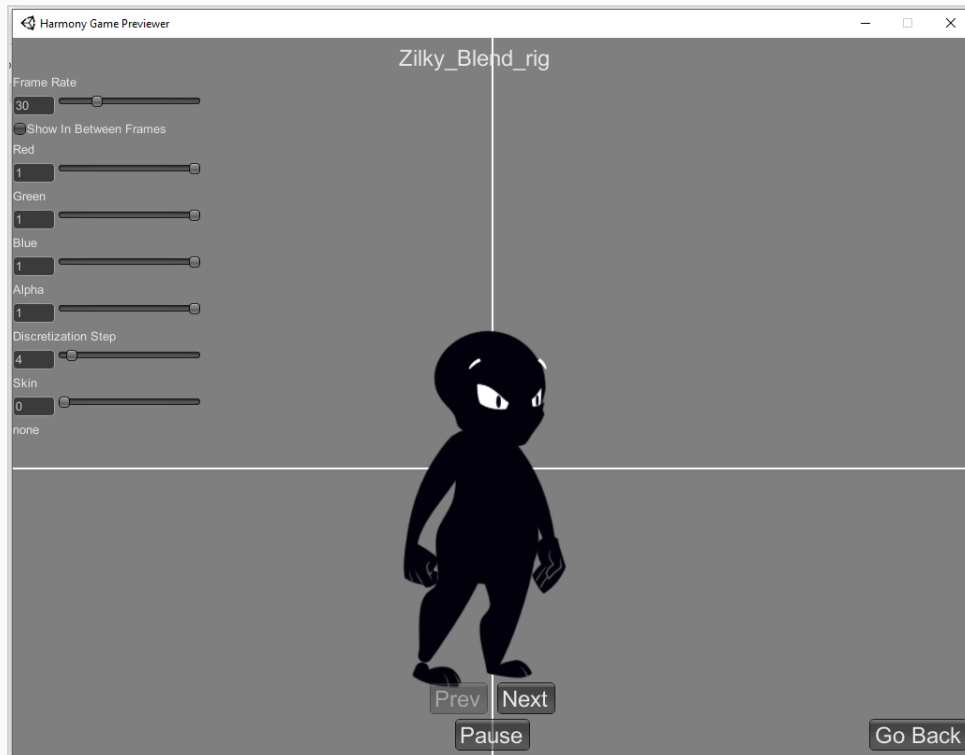


### NOTA

No descomprima la compilación dentro de su proyecto Unity; se aconseja descomprimirla en su propia carpeta.

### Cómo utilizar el generador de vista previa





1. Ejecute *Harmony Game Previewer.exe* o *Harmony Game Previewer.app*.
2. Seleccione qué proyecto desea cargar del menú del cargador. Estos proyectos se cargan desde la carpeta Harmony Resources en el mismo directorio que *Harmony Game Previewer.exe* o *Harmony Game Previewer.app*.
3. Desplácese por las animaciones con el botón Prev / Next (Anterior / Siguiente).
4. Ajuste la frecuencia de fotogramas, el tinte, el discretization step y la máscara con los controles de la parte superior izquierda.
5. Haga clic en el botón Go back (Volver) para volver al menú anterior y cargar una animación diferente.

## Manipulación de recursos

En esta sección, se describirán los componentes de Harmony Mesh y Harmony Renderer en Unity.

### Flujo de trabajo de TBG

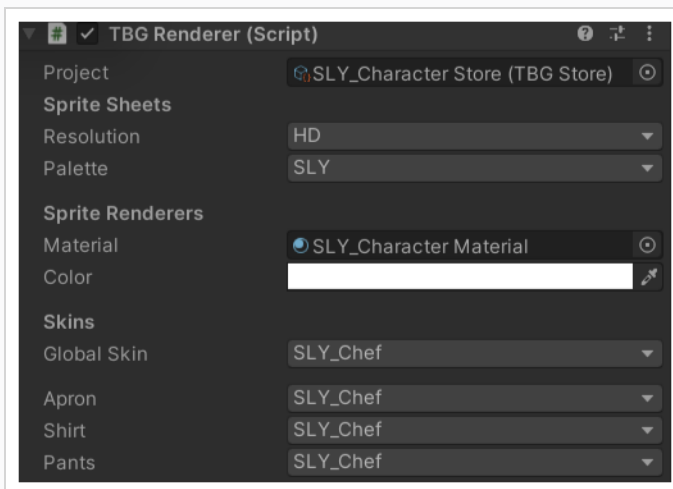
Si está importando sprites a Unity a través del flujo de trabajo de TBG, las opciones de manipulación de recursos para archivos TBG se mostrarán en la Inspector View. Las opciones de manipulación de recursos mostradas son diferentes de las opciones de manipulación de recursos mostradas en el flujo de trabajo XML.

#### Renderizador de recursos TBG

Harmony Renderer para el flujo de trabajo de archivos TBG se utiliza para controlar la animación y el renderizado de los recursos de Harmony para archivos TBG, así como para exponer variables importantes que puede querer cambiar durante el tiempo de ejecución.

### Cómo manipular los recursos de renderizado del flujo de trabajo de TBD

1. Seleccione un personaje en la escena. La sección TBG Renderer (Script) aparecerá en la pestaña Inspector a la derecha.



2. Controle los recursos del personaje configurando lo siguiente:
  - **Project (Proyecto):** Referencia al recurso TBGStore. Este recurso lo genera el importador y contiene los datos del proyecto TBG que no están cubiertos por los AnimationClips generados.
  - **Resolution (Resolución):** La resolución se utiliza para mostrar este personaje. Coincide con los nombres de las cartas de sprites asociadas.
  - **Palette (Paleta):** La paleta se utiliza para mostrar este personaje. Coincide con los nombres de las cartas de sprite asociadas.
  - **Material:** El material que se utiliza para sombrear este personaje. La máscara de hardware es el material sin iluminación predeterminado que se ofrece con el SDK.
  - **Color:** Opción que multiplica el color seleccionado en el recurso.

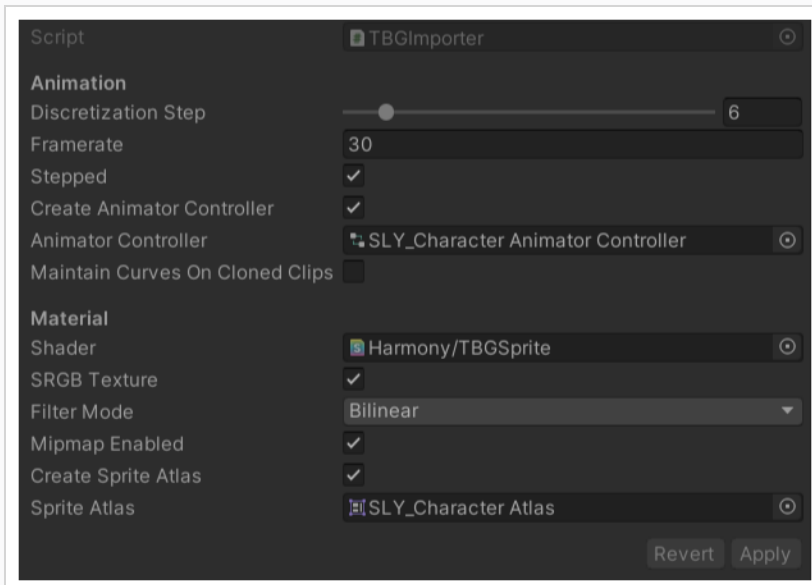
- **Skins (Máscaras):** Permite cambiar en tiempo real la máscara que ha exportado desde Harmony. Cada grupo tiene su propio menú desplegable para cambiar la máscara, así como un desplegable global para cambiar la máscara de todos los grupos a la vez.

## TBGImporter (Importador de TBG)

Cuando los archivos TBG se importan a Unity, los recursos de Harmony se convierten en prefabs, AnimationClips y otros recursos de Unity. El importador de TBG permite especificar cómo se crean los recursos y las animaciones.

### Cómo controlar la configuración para importar recursos TBG de Harmony a Unity

1. Seleccione un archivo de personaje .tbg en la vista de proyecto. Se abrirá la pestaña Inspector.
2. En la parte superior de la Inspector View (Vista del inspector), haga clic en **Select (Seleccionar)** para el modelo. Se abrirá la vista de TBGImporter y se mostrarán sus opciones.



3. Controle cómo se importarán las animaciones configurando lo siguiente:

- **Configuración de animaciones**

- **Discretization Steps (Pasos de discretización):** Control deslizante que representa el número de columnas/filas en las que se dividirán los sprites si los huesos los están deformando.
- **Frame rate (Fotogramas por segundo):** Fotogramas por segundo de la animación que utiliza el sistema de animación de Unity para reproducir los fotogramas de Harmony. Debe coincidir con los fotogramas por segundo de Harmony.
- **Stepped (En pasos):** Si está marcado, las animaciones pasarán entre fotogramas enteros y, por tanto, no se interpolarán de manera fluida. Si no está marcado, la animación se reproducirá entre fotogramas enteros para obtener una animación fluida.

- **Create Animator Controller (Crear controlador de animador):** Si está marcado, este importador creará automáticamente un nuevo Animator Controller que contendrá todos los AnimationClips generados y colocará el recurso junto al prefab en la vista de proyecto.
- **Animator Controller (Controlador de animador):** Cuando se realicen cambios en el archivo .tbg, dicho Animator Controller se actualizará con los nuevos AnimationClips.
- **Maintain Curves on Cloned Clips (Mantener curvas en clips clonados):** Si está marcada, las curvas duplicadas de los sub-assets del clip dentro del archivo .tbg (mencionado en AnimatorController) se sustituirán por los nuevos datos del archivo .tbg actualizado.
- **Configuración del material**
  - **Shader (Sombreador):** Cuando se crea el prefab, todos los SpriteRenderers harán referencia a un material generado usando este shader.
  - **SRGB Texture (Textura sRGB):** Si está marcada, mapeará el espacio de color de gamma de las texturas generadas en el renderizado del espacio de color lineal.
  - **Filter Mode (Modo de filtro):** Cómo se interpolan los píxeles contiguos a la textura.
  - **Mipmap Enabled (MipMap habilitado):** Generar texturas de menor resolución para renderizar a distancias mayores.
  - **Create Sprite Atlas (Crear atlas de sprites):** Si está marcada, este importador creará automáticamente un nuevo Sprite Atlas que contendrá todos los sprites generados y colocará el recurso junto al prefab en la vista de proyecto.
  - **Sprite Atlas (Atlas de sprites):** Cuando se realicen cambios en el archivo .tbg, dicho Sprite Atlas se actualizará con los nuevos sprites.

## Flujo de trabajo de XML

Si está importando sprites a Unity a través del flujo de trabajo de XML, las opciones de manipulación de recursos para archivos XML se mostrarán en la Inspector View. Las opciones de manipulación de recursos mostradas son diferentes de las opciones de manipulación de recursos mostradas en el flujo de trabajo TBG.

## Harmony Mesh

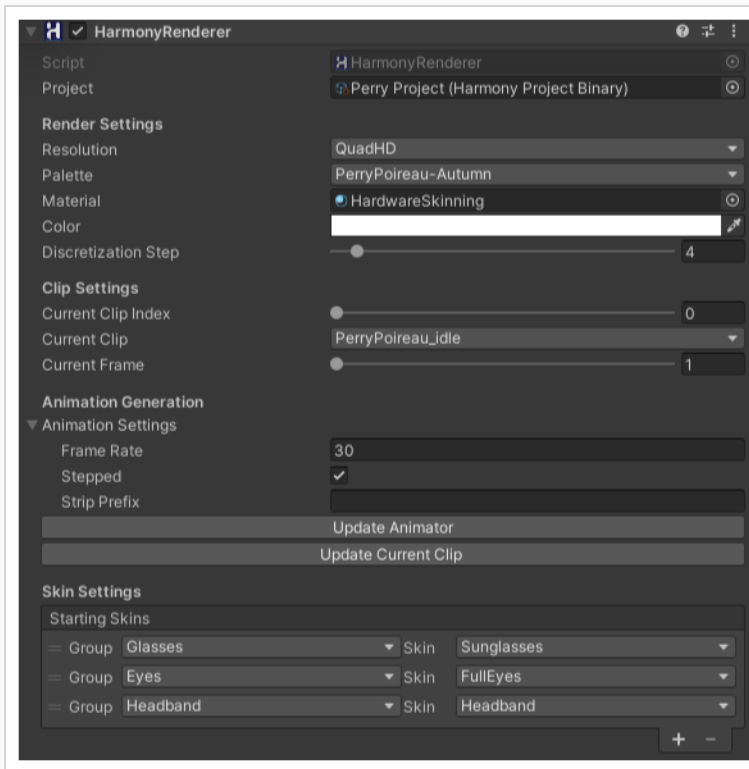
Harmony Renderer utiliza el componente Harmony Mesh para crear mallas de recursos basadas en los dibujos expuestos en la animación.

## Renderizador de recursos XML

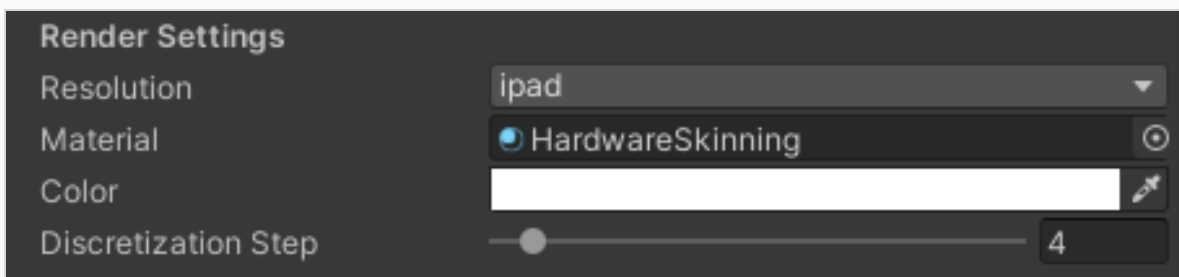
Harmony Renderer para el flujo de trabajo de XML se utiliza para controlar la animación y el renderizado de los recursos de Harmony para archivos XML, así como para exponer variables importantes que puede querer cambiar durante el tiempo de ejecución.

## Cómo manipular los recursos de renderizado del flujo de trabajo de XML

1. Seleccione un personaje en la escena
2. En la Inspector View (Vista del inspector), seleccione Add Component (Agregar componente) en la parte inferior de la vista. Aparecerá la sección HarmonyRenderer



3. Controle los recursos del personaje configurando lo siguiente:
  - **Project (Proyecto):** Referencia al recurso binario del proyecto de Harmony. Este recurso lo genera el importador y contiene los datos del proyecto de Harmony en un formato binario eficiente.
  - **Render Settings (Configuración de render):** La configuración de render se utiliza para modificar el recurso, ya sea cambiando la resolución o el shader utilizado para renderizar o la calidad de la deformación.

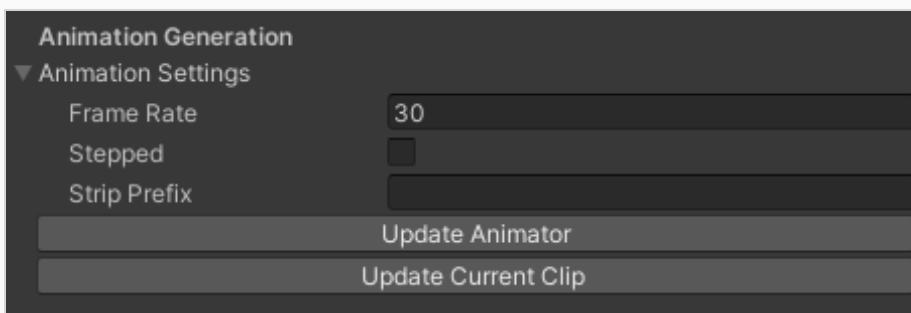


- **Resolution (Resolución):** La resolución se utiliza para mostrar este personaje. Coincide con los nombres de las cartas de sprites asociadas.

- **Palette (Paleta):** La paleta se utiliza para mostrar este personaje. Coincide con los nombres de las cartas de sprites asociadas.
- **Material:** El material que se utiliza para sombrear este personaje. La máscara de hardware es el material sin iluminación predeterminado que se ofrece con el SDK.
- **Color:** Opción que multiplica el color seleccionado en el recurso.
- **Discretization Steps (Pasos de discretización):** Control deslizante que define el orden de todos los dibujos deformados.
- **Clip Settings (Configuración de clips):** La configuración de clips se utiliza para ver las animaciones sin abrir la ventana Animation (Animación). Esto solo afecta al proyecto cuando está en modo editor; cuando se juega, el animador anula los ajustes dentro de la configuración de clips.



- **Current Clip Index (Índice de clips actual):** Índice de clips numérico para mostrar. Se puede animar en la ventana Animation (Animación).
- **Current Clip (Clip actual):** Menú desplegable para seleccionar qué animación se debe mostrar. Coincide con el índice de clips actual.
- **Current Frame (Fotograma actual):** El número de fotogramas que se muestran actualmente.
- **Animation Settings (Configuración de animación):** Esta configuración es la misma que la de la importación de recursos. Es útil al importar o volver a importar modificaciones de recursos dentro de Harmony. Una vez se realizan las modificaciones, haga clic en Update Animator (Actualizar animador) o Update Current Clip (Actualizar clip actual) para guardar las modificaciones.



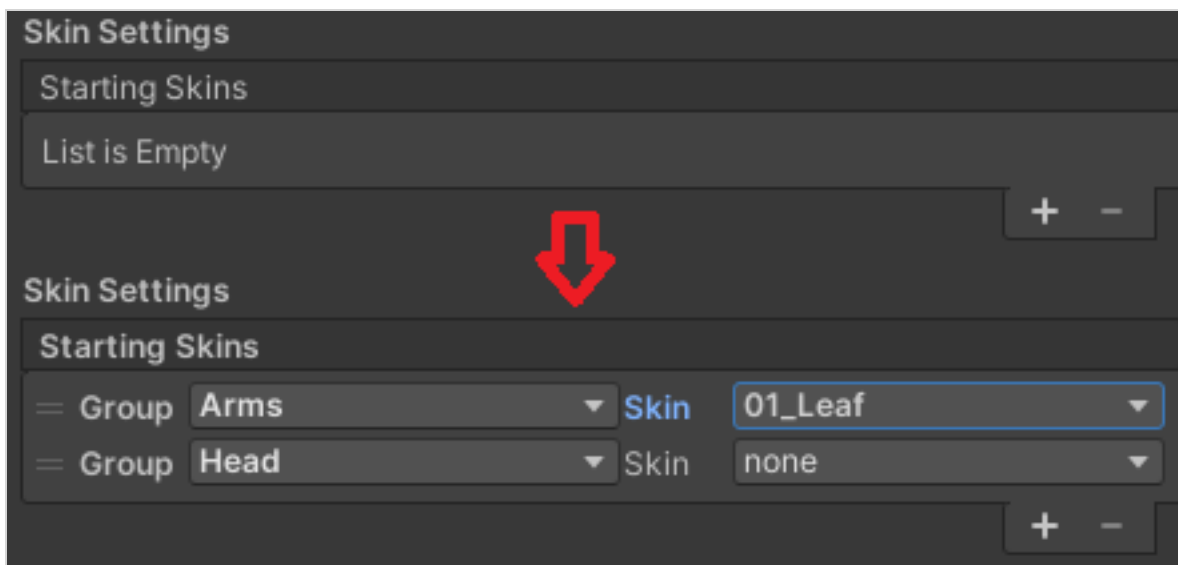
- **Frame rate (Fotogramas por segundo):** Fotogramas por segundo de la animación que utiliza el sistema de animación de Unity para reproducir los fotogramas de Harmony. Debe coincidir con los fotogramas por segundo de Harmony.
- **Stepped (En pasos):** Si está marcado, las animaciones pasarán entre fotogramas enteros y, por tanto, no se interpolarán de manera fluida. Si no está marcado, la animación se reproducirá entre fotogramas enteros para obtener una animación fluida.



- **Strip Prefix (Eliminar prefijo):** Se utiliza para eliminar un prefijo en el nombre de la animación. Por ejemplo, si queremos que «Wishling\_Idle» se quede en «Idle», definimos el prefijo en «Wishling\_». Esto es útil cuando el nombre de la animación de los juegos es el mismo para todos los personajes.
- **Update Animator (Actualizar animador):** Haga clic en este botón para actualizar todo el conjunto de animaciones de este personaje para que coincida con lo que hay en el proyecto Harmony. Se actualizarán las animaciones existentes con la nueva configuración introducida y se agregarán las que falten.
- **Update Current Clip (Actualizar clip actual):** Se actualizará solo el clip seleccionado.

### Skin Settings (Configuración de la máscara)

La configuración de la máscara permite modificar en tiempo real la máscara que ha exportado desde Harmony. Para cambiar una máscara, primero debe pulsar el botón + y agregar un grupo. Debe agregar la misma cantidad de grupos que tenía en la exportación de Harmony. Si no tenía grupos, agregue uno y déjelo en todos. Puede definir qué grupo desea modificar y, a continuación, qué máscara asignar a cada grupo. Más tarde, podrá eliminar un grupo seleccionando cualquier grupo y pulsando el botón -.



### Cómo cambiar una máscara en el flujo de trabajo de XML

1. Haga clic en el botón + para agregar un grupo/grupos.
2. Modifique el grupo y asígnele la máscara.
3. Elimine un grupo seleccionando cualquier grupo y haciendo clic en el botón -.

## Acerca del proyecto de ejemplo de Unity

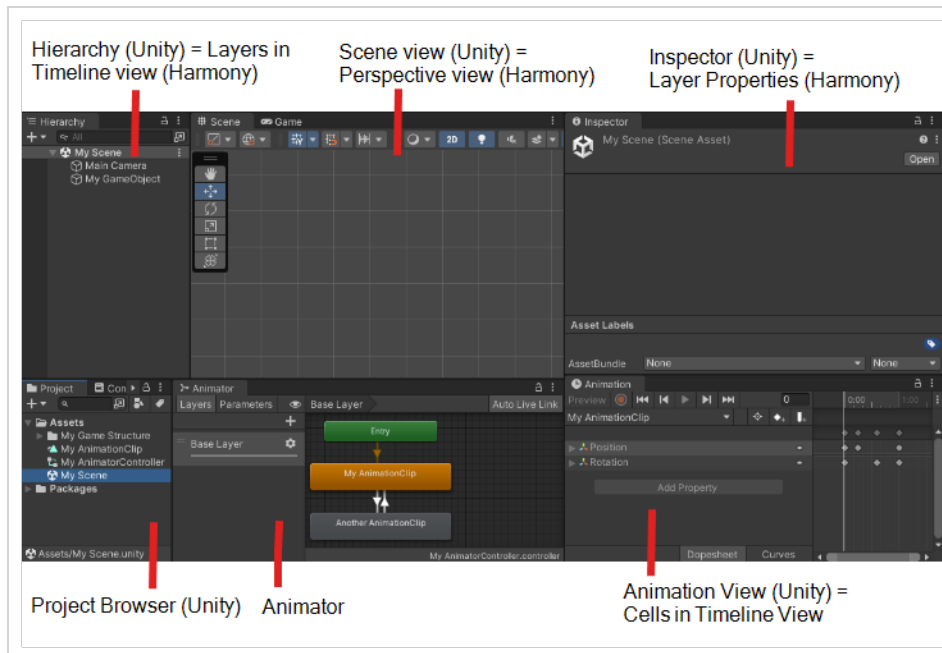
Toon Boom incluye un proyecto de ejemplo de Unity que contiene todos los comandos necesarios para importar los datos exportados desde Harmony. Dentro de este proyecto, hay una carpeta de recursos que contiene las siguientes carpetas:

- Plugins
- Escenas (escenas de demostración y un generador de vista previa)
- Comandos (todos los comandos de Harmony )
- Recursos de transmisión (aquí es donde deben exportarse todos los archivos de escena de Harmony)

Al configurar el comando Harmony, puede exportar automáticamente las animaciones a la carpeta StreamingAssets (Recursos de transmisión). De este modo, Unity cargará dinámicamente los recursos más actualizados en cuanto aparezcan en la carpeta. También puede colocar manualmente los datos de Harmony exportados en esta ubicación.

## Acerca de la interfaz de Unity

Aquí puede encontrar los principales componentes de la interfaz de Unity y sus equivalentes en Harmony:



Unity	Harmony	Descripción
Scene view (Vista de escena)	Vista de perspectiva	Aquí podrá configurar la escena, seleccionando y colocando entornos, el reproductor, la cámara, los enemigos y todos los demás GameObjects.
Game view (Vista de juego)	Vista de cámara	La vista renderizada a partir de las cámaras de su juego. Es representativa del juego final publicado.
Inspector	Layer Properties (Propiedades de capa)	Muestra información detallada acerca del GameObject seleccionado, incluidos todos los componentes adjuntos y sus propiedades.
Hierarchy (Jerarquía)	Vista de capas en la línea de tiempo	Muestra la jerarquía de elementos en la escena y le permite establecer relaciones padre-hijo para los diferentes objetos del juego.
Animation (Animación)	Vista de celdas en la línea de tiempo	Muestra los fotogramas clave de una animación por transformación que se esté animando.
Animator (Animador)		Muestra un gráfico de estados y transiciones que le permite configurar cómo responderán sus personajes a los eventos del juego y a las acciones de los jugadores.

Navegador de proyectos	---	Le permite acceder y administrar los recursos de un proyecto.
------------------------	-----	---

Aquí puede encontrar algunos tipos de datos con los que interactúa Unity y sus equivalentes en Harmony:

Unity	Harmony	Descripción
Sprite	Dibujo	Una sola imagen que se puede mostrar en un SpriteRenderer.
SpriteRenderer (Renderizador de sprites)	Capa	Muestra un sprite cada vez. Los sprites se pueden intercambiar durante una animación para crear el movimiento de una parte del cuerpo del personaje.
Transform (Transformación)	Peg	Se utiliza para formar la jerarquía del cuerpo de un personaje. Se puede escalar, rotar o trasladar para crear el movimiento de una parte del cuerpo del personaje. El sesgo debe derivarse de una jerarquía de escalas y rotaciones.
GameObject (Objeto de juego)		Siempre se combina con una transformación. Los GameObjects incluyen componentes que pueden mostrar y simular la partida. El SDK ofrece comportamientos personalizados que se ejecutan en GameObjects para mostrar gráficos específicos de Harmony.
Asset (Recurso)		Datos básicos en un proyecto Unity que se utilizan para gráficos, física, partida y efectos de audio. Por lo general, estos son archivos, pero pueden presentarse como sub-assets dentro de archivos más grandes. Todas las texturas, AnimationClips, sprites, AnimatorControllers, AudioClips y prefabs son recursos.

## Importación de archivos de Harmony en Unity

Una vez que haya terminado de crear sus ilustraciones y movimientos de personajes animados repetitivos en Harmony, es hora de importarlos a Unity para integrarlos en el juego.

### Cómo importar archivos de Harmony directamente en Unity

1. Cree un proyecto 2D de Unity.
2. El proyecto de SDK Harmony se debe importar a su proyecto para que los recursos se incorporen. Realice una de las siguientes acciones:
  - Vaya a la Unity Asset Store ([assetstore.unity3d.com/es/#!/content/31211](https://assetstore.unity3d.com/es/#!/content/31211)) y guarde el archivo Harmony Game SDK en su ordenador. A continuación, impórtelo en su proyecto desde: **Top Menu > Asset > Import Package > Custom Package (Menú superior > Recurso > Importar paquete > Paquete personalizado)**.
  - Busque Harmony Game SDK en la Unity Asset Store y luego descargue e importe el paquete. Puede acceder a la Asset Store (Tienda de recursos) desde: **Top Menu > Window > Asset Store (Menú superior > Ventana > Tienda de recursos)**.



#### NOTA

Para cada nuevo proyecto de Unity, debe repetir el paso 2, ya que contiene los archivos y carpetas necesarios para que funcione el proceso de importación de Harmony.

Existen dos flujos de trabajo principales para exportar e importar datos de Harmony a Unity

- **Archivos TBG/TBGRenderer**

Este es el flujo de trabajo preferible si:

- Utiliza exclusivamente Unity y quiere la experiencia más natural a la forma de hacer las cosas en Unity.
- Quiere importar más rápido con menos pasos.
- Quiere producir una jerarquía significativa de transformaciones y SpriteRenderers que se puedan ajustar y con los que pueda jugar. Esto permitirá a aquellos con poca o ninguna experiencia en la creación de comandos agregar efectos de procedimiento únicos a los personajes.
- Quiere sacar partido de la malla de sprites de Unity combinando los atlas de sprites y los controladores de animadores con la superposición y la fusión de transiciones. Así se abre la posibilidad de utilizar paquetes existentes o comandos sencillos para la IK, la física de sacudidas, los muñecos de trapo (ragdolls), etc.

- **Carpetas XML / HarmonyRenderer**

Este es el flujo de trabajo preferible si:

- Utiliza Unity para crear prototipos y quiere movilidad a otros motores de juegos. El flujo de trabajo XML es transferible, ya que recurre a una librería de C++ que es portátil a otros motores.

- Quiere más espacio para jugar con soluciones personalizadas de transmisión de recursos y quiere un punto de partida para crear efectos de procedimiento personalizados de alto rendimiento.

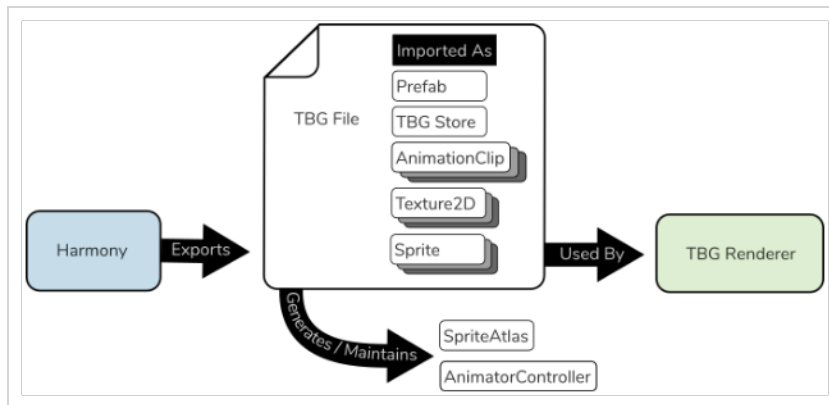


#### NOTAS

- Si lo desea, puede cambiar de flujo de trabajo más adelante.
- Ambos sistemas predeterminado utilizan Animator Controllers para elegir y reproducir animaciones en el personaje, por lo que los comandos que interactúen únicamente con el animador del personaje funcionarán de forma parecida. Sin embargo, los comandos que cambian máscaras o resoluciones, o que proporcionan efectos de procedimiento, se tendrán que reescribir o crear a medida para cada flujo de trabajo.
- En general, la transición de HarmonyRenderers a TBGRenderers funcionará mejor, ya que HarmonyRenderers tiene un subconjunto de las características disponibles en TBGRenderers de forma predeterminada. Siempre es buena idea mantener el acceso a sus proyectos originales de Harmony .xstage si necesita volver a exportar con distinta configuración o un flujo de trabajo diferente.

## Flujo de trabajo de archivo de archivo TBG en Unity

En el flujo de trabajo de archivo TBG (Toon Boom Gaming), Unity interpretará de forma inmediata un archivo TBG exportado desde Harmony como un nuevo recurso al que puede hacer referencia para derivar nuevas texturas, sprites, datos de proyecto y un prefab del archivo. El prefab resultante se puede arrastrar a la escena de Unity para visualizarlo y animarlo inmediatamente. Cuando el archivo TBG se sobrescriba a partir de una nueva exportación en Harmony, el importador de TBG puede realizar el proceso de reimportación de forma automática para actualizar el prefab, los sprites y las texturas existentes. Esto permite a los artistas iterar rápidamente sobre sus personajes y ver los resultados de forma inmediata en el juego.



El prefab TBG Renderer resultante contiene una jerarquía completa de transformaciones y SpriteRenderers para mostrar el personaje en un rig compatible con Unity. Además, se incluye un comando TBG Renderer que garantiza que los sprites se intercambien durante el progreso de una animación y que las cuchillas se apliquen correctamente a los SpriteRenderers relevantes.



### NOTA

El formato XML utilizado en los archivos TBG es una versión más reciente y puede no ser compatible con el flujo de trabajo de la carpeta XML para personajes más complicados. Conviene conservar los archivos de Harmony originales y volver a exportarlos más adelante con otra configuración.

## Cómo importar un archivo TBG en Unity

1. En Harmony, busque el botón Export to Sprite Sheets (Exportar a cartas de sprites) 🎮 en la barra de herramientas Game (Juego).
2. En la ventana Export to Sprite Sheets (Exportar a cartas de sprites), consulte **Encode as TBG (Codificar como TBG)**.
3. Para la ruta de recurso, introduzca la ruta del archivo donde desea guardar el personaje.
4. Seleccione **Save and Export (Guardar y exportar)**. Aparecerá una barra de carga que completará el proceso de exportación y generará un nuevo archivo TBG.
5. Abra su proyecto de Unity. Unity debe cargar el TBG de forma automática como un prefab.

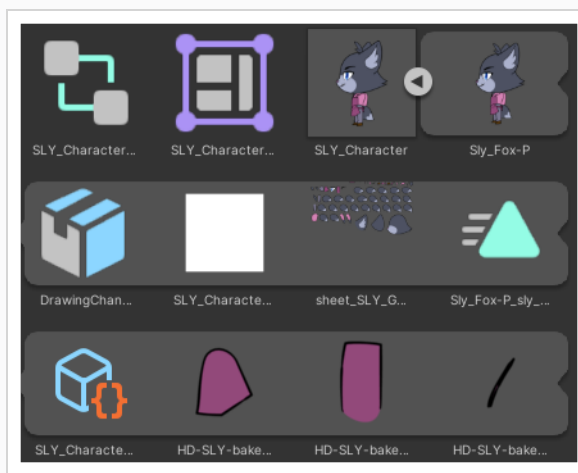
**NOTA**

Si el personaje no aparece como un prefab, asegúrese de que:

- Tiene HarmonySDK instalado para su proyecto de Unity. Se debe instalar para cada nuevo proyecto que se cree.
- No tiene un marcador de escena que se extienda más allá de la exposición de un personaje.

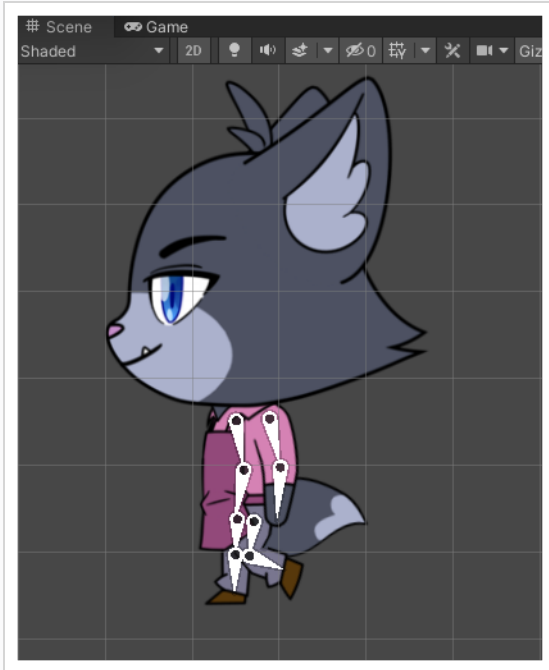
En su proyecto de Unity, debe haber:

- sub-assets para texturas, sprites, datos de TBG Store y AnimationClips.
- Dos nuevos recursos para el SpriteAtlas y el AnimatorController del archivo además del prefab TBG.

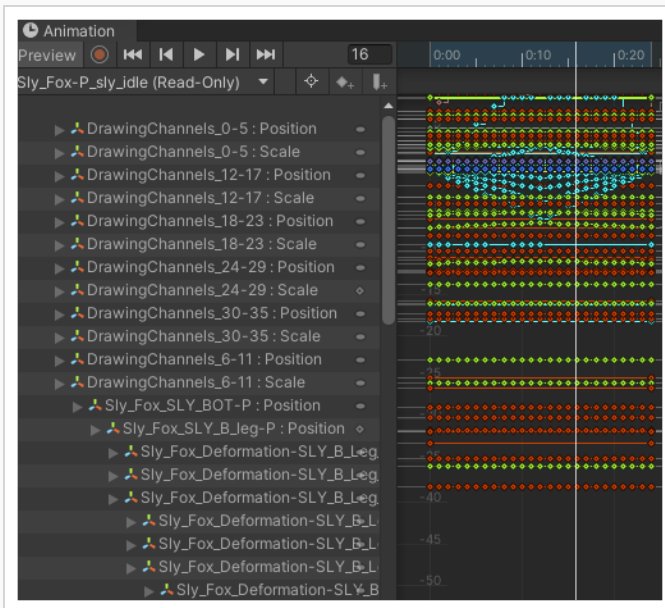


El prefab TBG resultante se puede arrastrar a la jerarquía, donde se mostrará y será interactivo.

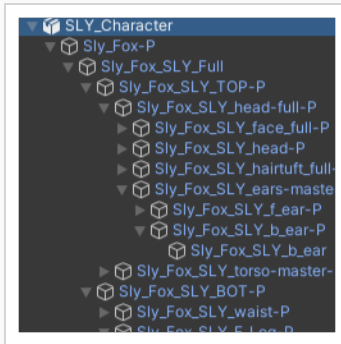




Al seleccionar el prefab en la jerarquía, podrá ver las animaciones del personaje desde la vista de animación.

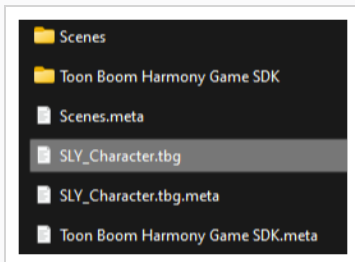


Al abrir el prefab en la vista de jerarquía haciendo clic en la flecha pequeña, podrá explorar la jerarquía del personaje de forma muy similar a lo que se muestra en la línea de tiempo de Harmony.

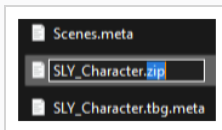


## Cómo revisar el contenido de un archivo TBG de forma manual

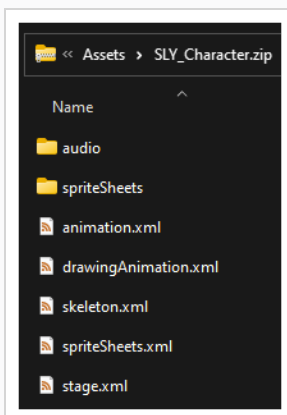
1. Busque el archivo TBG en el explorador de archivos del ordenador



2. Si utiliza Windows o Mac, cambie el nombre de la extensión del archivo, ya que los ordenadores Windows y Mac no reconocen la extensión de archivo .tbg.



Una vez cambiado de nombre, Windows y Mac le permitirán explorar el contenido del archivo .zip para que pueda asegurarse de que las cartas de sprites tienen un tamaño razonable o de que los archivos de datos tienen el contenido que espera.



**NOTA**

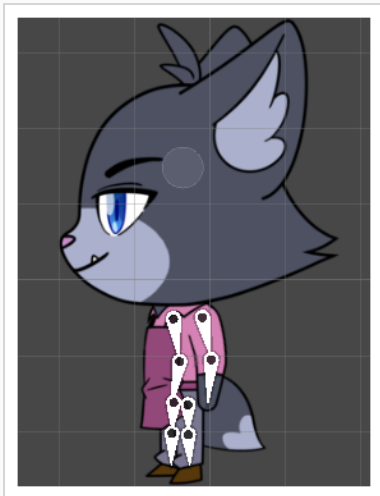
Para asegurarse de que Unity no tenga ningún problema, puede volver a cambiar la extensión del archivo a .tbg antes de reanudar el trabajo con Unity. Por lo general, este proceso de inspección solo es necesario con ordenadores que no tengan instalado Unity.

## Adición de cinemática inversa (IK) a un prefab de TBG Renderer

Puede utilizar la IK 2D de Unity para agregar objetivos de procedimiento a las extremidades de sus personajes. Puede utilizar SLY\_Character de la escena de ejemplo del flujo de trabajo de TBG para este proceso.

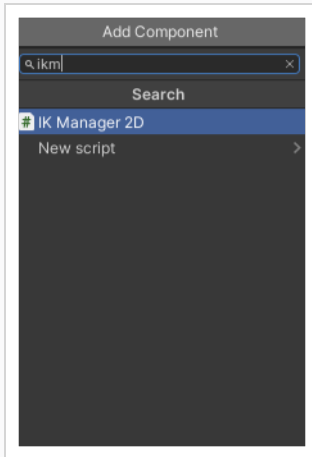
### Cómo agregar IK a un prefab de TBG Renderer

1. Busque y arrastre el SLY\_Character desde la vista de proyecto a su Scene View, y pulse F para encuadrar el personaje en su vista.

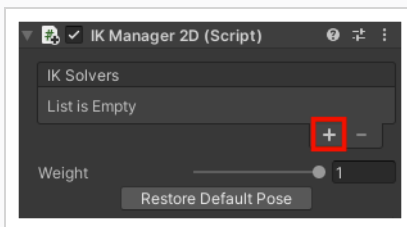
**NOTA**

Asegúrese de que se seleccione el SLY\_Character en la jerarquía.

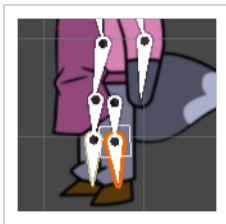
2. En la Inspector View (Vista del inspector), agregue un componente IK Manager 2D.



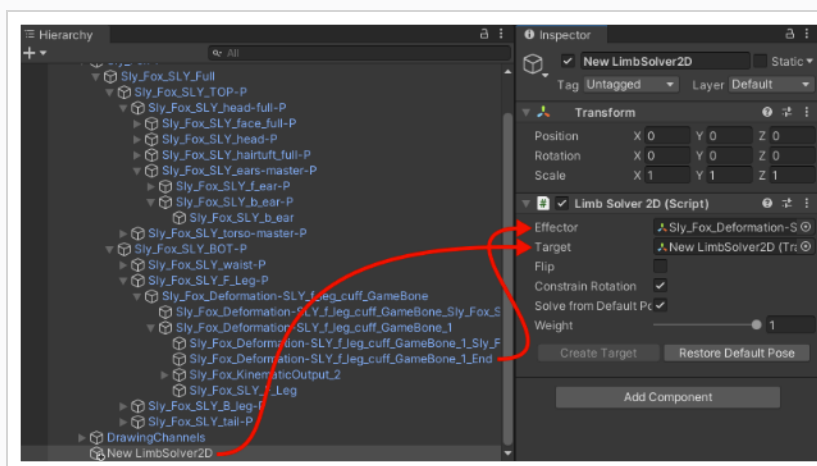
3. En el componente IK Manager 2D, haga clic en el botón + para agregar un nuevo solucionador de IK y seleccione **Limb (Extremidad)**.



4. Haga clic en el hueso inferior derecho de las piernas de Sly para exponer su lugar en la jerarquía. Utilizaremos su GameObject hermano que tiene «...\_GameBone\_1\_end» en el nombre.



5. Con el objeto de juego «New LimbSolver2D» seleccionado, arrastre «New LimbSolver2D» a la ranura objetivo del comando LimbSolver2D y arrastre el GameObject «\_leg\_cuff\_GameBone\_1\_end» a la ranura de Effector (Generador de efectos).



Una vez hecho esto, se puede moverse por la transformación Nuevo New LimbSolver2D y ver cómo se dobla la extremidad para adaptarse al objetivo.



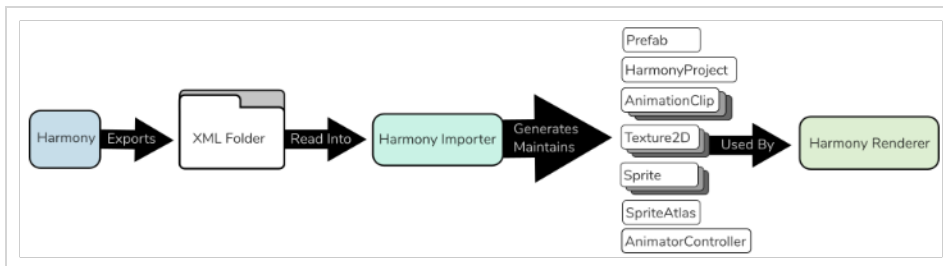
Para cambiar la dirección en la que se dobla la rodilla, podemos hacer clic en la casilla «Flip» (Voltear) del componente Limb Solver 2D.

## Flujo de trabajo de carpeta XML en Unity

En el flujo de trabajo de la carpeta XML, un desarrollador puede crear un recurso de importador de Harmony único para hacer referencia a una carpeta XML exportada desde Harmony. Este importador se puede utilizar luego para generar un recurso de proyecto de Harmony y un nuevo prefab de Harmony Renderer.

Para cada archivo XML que se vuelva a exportar, los desarrolladores deben establecer de forma manual las rutas de importación y exportación, y realizar el proceso de importación desde el recurso de importador de Harmony.

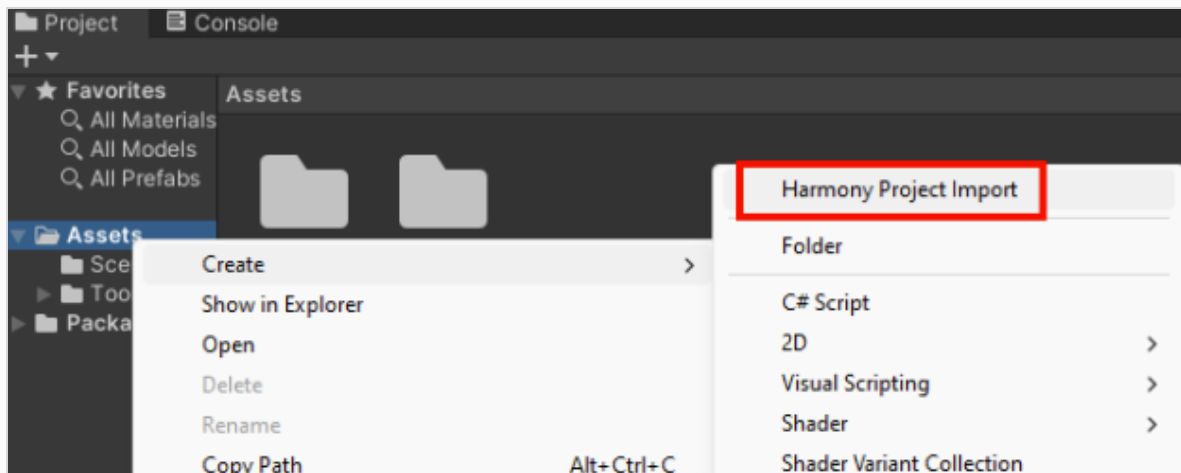
Este flujo de trabajo ofrece flexibilidad en cuanto al lugar donde pueden residir los datos XML. Por ejemplo, puede residir dentro o fuera del proyecto de Unity. Sin embargo, una desventaja de este flujo de trabajo es que se pueden cometer más errores, ya que hay más pasos implicados. Por ejemplo, la ruta de archivo para el importador de TBG tendrá que actualizarse de forma manual al mover archivos de un lugar a otro.



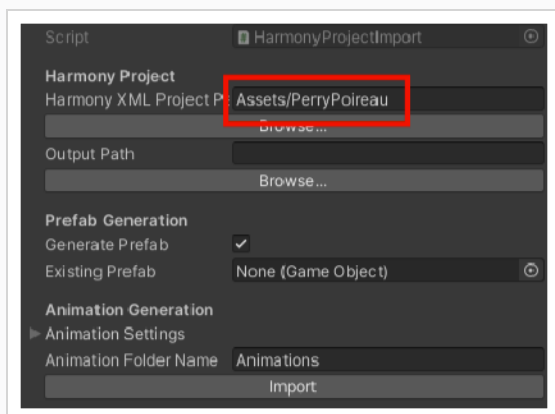
El prefab de Harmony Renderer resultante permite un renderizado de personajes de alto rendimiento mediante la generación de una malla dentro de un plugin de C++. Esto es preferible para multitudes de personajes que no requieren complejas funciones de renderizado personalizadas.

### Cómo importar una carpeta XML en Unity

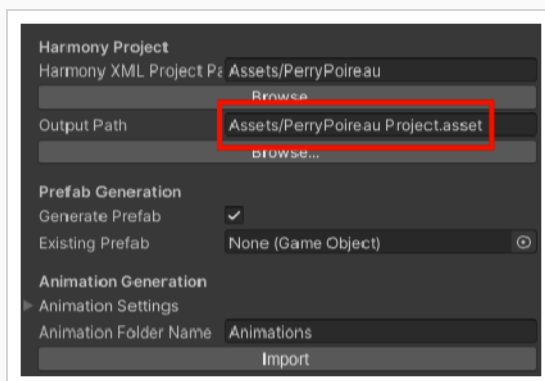
1. En Harmony, busque el botón Export to Sprite Sheets (Exportar a cartas de sprites) 🎮 en la barra de herramientas Game (Juego).
2. Seleccione **Save and Export (Guardar y exportar)**. Asegúrese de que la opción Export as TBG (Exportar como TBG) no esté marcada en el cuadro de diálogo Export to Sprite Sheets (Exportar a cartas de sprites).
3. Guarde el XML en una carpeta dentro de la carpeta de recursos de su proyecto de Unity.
4. En Unity, cree un nuevo recurso de importador de Harmony mediante una de las siguientes acciones:
  - En menú desplegable Assets (Recursos) del menú superior, seleccione **Create > Harmony Project Import (Crear > Importación de proyecto de Harmony)**.
  - Haga clic con el botón derecho en la carpeta Assets (Recursos) en la vista de proyecto y seleccione **Create > Harmony Project Import (Crear > Importación de proyecto de Harmony)**.



5. Nombre el importador de Harmony con el sufijo «Importer» para evitar futuras confusiones.
6. Copie la ruta relativa de la carpeta exportada y péguela en el nuevo importador de Harmony.

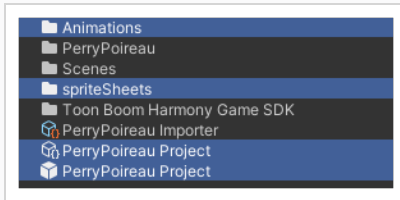


7. Copie la ruta relativa del recurso de importador de Harmony, péguela en el importador de Harmony, sustituya «Importer» por «Project» en la ruta.

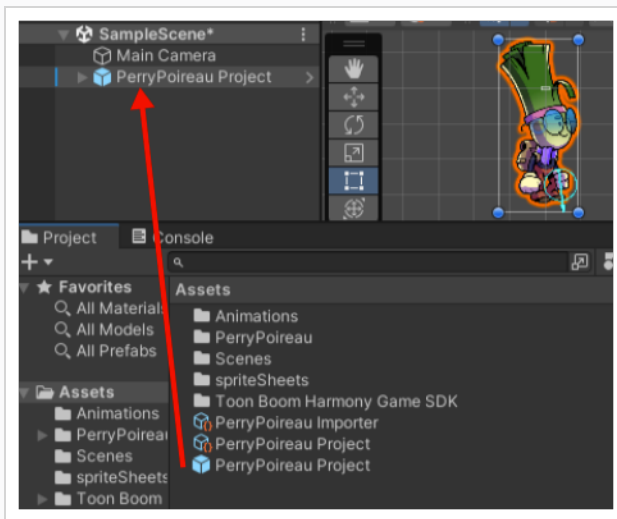


8. Haga clic en **Import (Importar)**.

Se creará un nuevo recurso de HarmonyProject, así como un nuevo prefab con Harmony Renderer que haga referencia al HarmonyProject.



El recurso prefab se puede arrastrar desde la vista de proyecto a la vista de jerarquía y a la Scene View para mostrar el personaje en el juego.



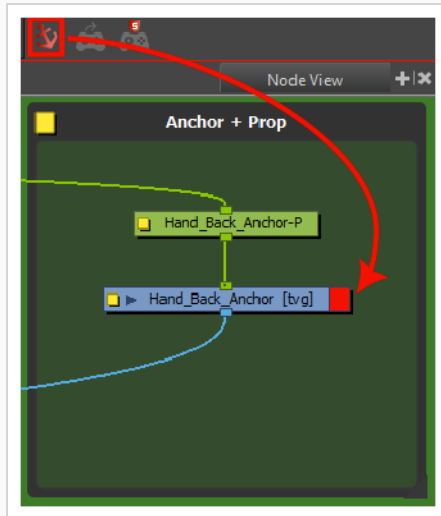
## Importación de anclas

En Harmony, las anclas ofrecen una manera de indicar qué pegs y dibujos son posiciones de interés para los comandos de juego en Unity. Se puede asignar un ancla a cualquier dibujo o peg.

### Cómo importar un ancla a Unity

1. Seleccione una capa en la vista de línea de tiempo o un nodo en la vista de nodos y seleccione el botón Toggle Anchor (Alternar ancla) en la barra de herramientas Game (Juego).





## 2. Importe el personaje a Unity.

La próxima vez que se lleve a cabo el proceso de importación de XML, la nueva ancla aparecerá en Unity bajo el prefab importado del personaje como una nueva HarmonyAnchor.



Este GameObject seguirá la posición, rotación y escala de la animación actual que se esté reproduciendo en el personaje. Esto permite adjuntar de forma dinámica objetos de juego como hijos a las anclas para transportarlos, equiparlos y utilizarlos.



### NOTA

El desplazamiento de las posiciones de las anclas no afectará a los gráficos de Harmony Renderer. Para ver y afectar a toda la estructura del personaje en Unity y no solo a las anclas, es posible que desee considerar el uso del flujo de trabajo de TBG

## Adición de fuentes de audio

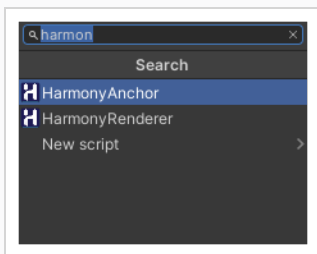
Cuando se crean archivos de escena con audio en Harmony, el audio se exporta a la carpeta del proyecto. El importador de Harmony creará automáticamente un AudioSource y la agregará al prefab generado. Harmony Renderer puede entonces activar la reproducción de los archivos AudioClip en los puntos durante la reproducción de la animación tal y como se creó en Harmony.

## Adición de un Harmony Renderer a objetos de juego vacíos

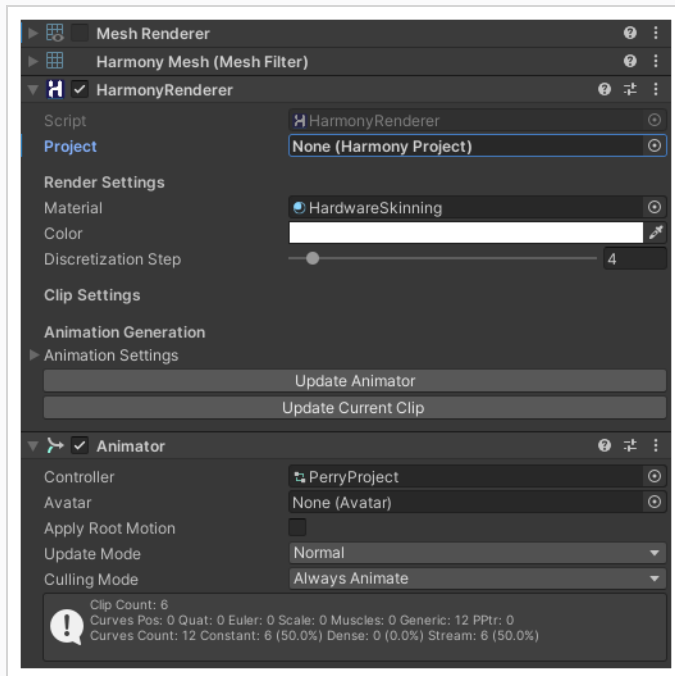
Si no tiene el prefab de su importador de Harmony, aún es posible crear un nuevo componente de Harmony Renderer en un objeto de juego vacío para obtener el mismo resultado final.

### Cómo utilizar un GameObject vacío

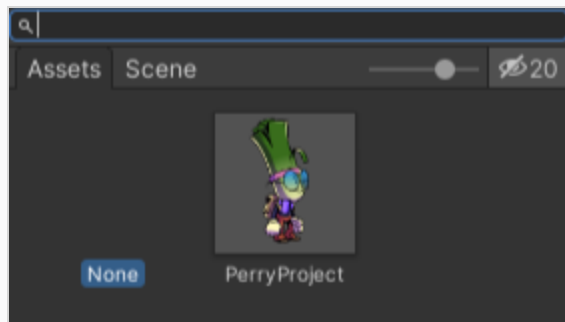
1. Seleccione **GameObject > Create Empty (GameObject > Crear vacío)**.
2. Cambie el nombre del GameObject vacío para que quede claro en la jerarquía. Dado que estamos utilizando el archivo de demostración de PerryPoireau, cambie el nombre del GameObject a **SpaceCat** mediante una de las siguientes acciones:
  - Haga doble clic en el nombre y cambie su nombre en la jerarquía.
  - Seleccione el GameObject y cambie su nombre en el Inspector.
3. En este punto, es un objeto vacío. El acceso a los datos de Harmony se realiza mediante comandos.
  - Seleccione el GameObject PerryPoireau.
  - En el Inspector, vaya a **Add Component > Scripts > Harmony Renderer (Agregar componente > Comandos > Renderizador de Harmony)**.



En el Inspector aparecen cuatro nuevas secciones denominadas Mesh Renderer (Renderizador de malla), Harmony Mesh (Filtro de malla) y Harmony Renderer (Renderizador de Harmony).



A partir de aquí, es necesario adjuntar un proyecto de Harmony a este Harmony Renderer recién creado. Al hacer clic en el círculo situado a la derecha de la ranura vacía (proyecto de Harmony), puede seleccionar el proyecto que ya ha importado siguiendo los pasos para crear un proyecto de Harmony utilizando el flujo de trabajo de la carpeta XML.



## Configuración de colisiones en Unity

Es útil extraer la información de la caja delimitadora si desea hacer que algo colisione con el personaje. Cuando coloque un personaje 2D en una escena 3D, puede que desee usar física para hacer que el personaje 2D colisione con el plano 3D.

Los colisionadores se mejoraron para proporcionar varios tipos de colisionadores diferentes para su uso en Unity. Puede seleccionar colisionadores de polígono y colisionadores de caja para obtener una caja delimitadora más precisa.

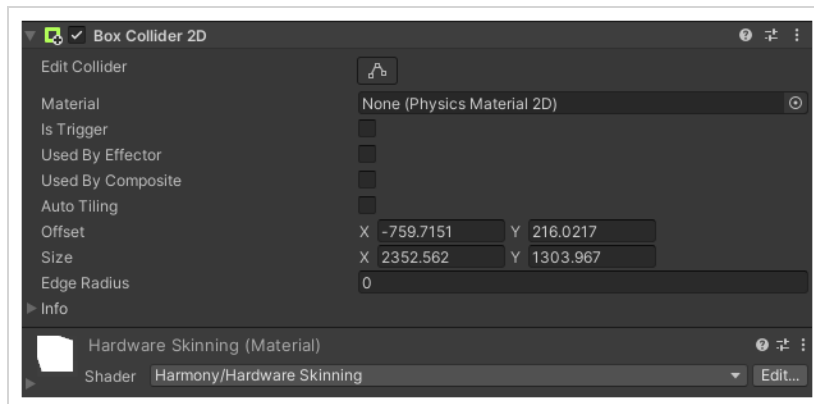
### Cómo configurar colisiones

1. Seleccione su personaje importado.
2. En el Inspector, seleccione **Add Component > Physics 2D > Box Collider 2D (Agregar componente > Física 2D > Colisionador de caja 2D)**.
3. Para habilitar la física en el personaje, vaya a **Add Component > Physics 2D > Rigid Body 2D (Agregar componente > Física 2D > Cuerpo rígido 2D)**.



#### NOTA

También puede utilizar la física 3D en los archivos importados.



Puede ajustar las variables y ver los resultados

## Glosario de juego

<b>2D Animation Package</b>	Plugin original para Unity (creado por Unity) que facilita el procesamiento y la animación avanzados de sprites 2D (deformaciones, rigging, teselación).
<b>3D estereoscópico</b>	Imágenes o películas que generan dos imágenes diferentes, cada uno desde el punto de vista de uno de los ojos humanos. Cuando se visualizan o proyectan a través de dispositivos especiales, como una pantalla 3D o gafas de realidad virtual, se crea la ilusión que los elementos de la imagen se encuentran enfrente del espectador.
<b>aberración cromática</b>	En óptica, una aberración cromática (CA), acromatismo o distorsión cromática, es un tipo de distorsión en el que una lente no puede enfocar todos los colores en el mismo punto de convergencia.
<b>aceleración</b>	En animación, la velocidad es la aceleración y la desaceleración de un movimiento. Puede ser un movimiento creado mediante una curva de función o una serie de dibujos animados. Otros términos comunes para aceleración gradual y desaceleración gradual son aceleración y desaceleración.
<b>aceleración de entrada</b>	Aceleración drástica al principio de la acción.
<b>aceleración de salida</b>	Aceleración drástica al final de la acción.
<b>aceleración gradual</b>	Aceleración gradual en la acción. Conocida también como aceleración.
<b>alimentador de papel automático</b>	Método automatizado de introducir dibujos en un escáner en el que hay varios dibujos apilados en un alimentador de hojas. Cuando se activa el escáner, los dibujos se digitalizan de forma consecutiva, sin ninguna otra intervención del usuario.
<b>Ancla</b>	Los nodos en Harmony pueden marcarse como anclas para exponer su posición, rotación y escala en el motor de juego durante el tiempo de ejecución. Los archivos TBG no requieren anclas.
<b>animación</b>	Simulación de movimiento que se crea dibujando o editando fotogramas individuales.
<b>animación cut-out</b>	El proceso conocido como animación cut-out es la acción de animar personajes compuestos por varias piezas moviéndolos de fotograma en fotograma. La animación cut-out se puede generar por ordenador o realizarse del modo tradicional utilizando papel.

<b>animación directa</b>	Técnica en la cual se dibuja por orden una secuencia completa desde la primera posición hasta la última. Esta metodología requiere muy poca planificación. El lugar en el que el personaje termina y cómo llega hasta allí pueden ser una sorpresa tanto para el público como para el animador. Este enfoque es espontáneo y creativo, pero puede generar resultados imprecisos.
<b>animación pose a pose</b>	El proceso de animación pose a pose es la acción que consiste en crear todas las poses de acción principales, denominadas poses clave, y después colocar las poses secundarias entre las claves. Las poses secundarias se denominan descomposición. Por último, el animador rellena los huecos con los dibujos intercalados para lograr una animación fluida.
<b>animación secundaria</b>	Movimiento secundario provocado por la acción principal. Por ejemplo, un personaje que corre y lleva puesta una capa. La acción principal es el cuerpo que corre. Esto provocará que la capa siga el movimiento, aunque no se moverá al mismo tiempo, sino que reaccionará varios fotogramas después y seguirá la curva de movimiento principal.
<b>animación sin papel</b>	El proceso de animación sin papel es la acción de animar digitalmente. El principal proceso de animación sin papel es dibujar la animación, fotograma a fotograma, directamente en el software.
<b>animación tradicional</b>	Tipo de proceso de animación mediante el cual todas las secuencias de animación se dibujan a mano en papel antes de digitalizarlos o tintarlos en celdas.
<b>animática</b>	Una película hecha con secuenciación de paneles en un storyboard, en la que se establece el tiempo aproximado de cada panel para la acción que representa y se sincroniza cada panel con la pista de sonido. Las animáticas se utilizan para convertir el storyboard en un borrador de la película final con el fin de determinar el tiempo asignado a cada escena y a cada acción y sincronizar la acción con la pista de sonido.
<b>AnimationClip</b>	Unity Asset que contiene una serie de curvas para diferentes campos de datos en un prefab (cambios de posición, rotación, escala de
<b>AnimatorController</b>	Los Animator Controllers deciden qué animación debe reproducirse durante la partida en función del estado actual del personaje.
<b>anime</b>	Animación japonesa.
<b>Archivo .tbg</b>	El formato de archivo de juego de Toon Boom (Toon Boom Game o .tbg), se creó a partir de Harmony y Unity lo incorporó, de la misma forma que

	en el futuro lo harán otros motores de juego.
<b>arco</b>	La acción no suele producirse de forma directa, sino que más bien se suele desplegar en lo que los narradores denominan un arco. El propósito de un arco de historia es mover un personaje o situación de un estado o escenario al siguiente.
<b>arrastrar</b>	Operación que consiste en hacer clic en un elemento, mantenerlo pulsado y, a continuación, moverlo mediante el movimiento del cursor del ratón. Esto se puede realizar con el botón izquierdo del ratón o con la punta del lápiz.
<b>artista de layout</b>	Artista que dibuja el fondo y crea la cámara y la guía de campo, haciendo coincidir la escena y el movimiento de la cámara. Dibujará en el modelo las poses de acción principales.
<b>atajos de teclado</b>	Tecla de teclado o combinación de teclas de teclado que se asigna para realizar un comando específico en la aplicación al pulsarlos.
<b>Automatic Importing Process</b>	Importará el archivo automáticamente sin intervención del usuario cuando el archivo se cree o se modifique/sobrescriba/actualice.
<b>baja resolución</b>	Formato idóneo para vídeos destinados a la web, donde el tamaño y la descarga rápida del archivo de vídeo tienen prioridad respecto a la calidad. Una imagen de baja resolución es aquella que no contiene detalles precisos.
<b>bake_group</b>	Los bake_groups son grupos en la vista de nodos cuyo nombre se cambia a «bake_...»; además, un bake_group solo admite 1 puerto de entrada y 1 puerto de salida. Durante la exportación desde Harmony, estos grupos se analizan en busca de cambios y se exportan tantos sprites como variaciones únicas de datos haya en el grupo, así como una animación para mostrar cada sprite cuando proceda. Dentro de un bake_group se puede utilizar cualquier efecto: brillos, deformaciones con forma libre, capa de color/capa de línea.
<b>bocetos</b>	Es un nombre común en una película de animación para un dibujo que se utiliza como referencia, pero no forma parte de la imagen final. Un layout es un boceto. Es el esbozo del esqueleto de una animación o diseño. Los bocetos consisten principalmente en líneas y formas esbozadas, pero también pueden contener detalles de diseño.
<b>campo</b>	Unidad de medida que se utiliza en la animación tradicional para registrar y realizar un seguimiento de la cámara, así como de su proximidad a la escena.

<b>canal alfa</b>	Canal de imagen que contiene información de transparencia. De partida, una imagen tiene tres canales: rojo, verde y azul (RGB). El canal alfa es el cuarto canal (A). El mate, o información de transparencia, se almacena en este cuarto canal. Una imagen sin canal alfa siempre es opaca.
<b>capa</b>	En animación, una capa es una columna, nivel o personaje individual. Las capas de una escena están superpuestas para formar la imagen final.
<b>capa de fondo</b>	En animación, una capa de fondo es una parte específica de la decoración colocada detrás de la animación principal.
<b>capa superior</b>	Parte del entorno de una escena, como una silla o un arbusto, que se coloca delante de la animación principal.
<b>carta de color</b>	Una carta de color es una tarjeta que contiene un color sólido con el mismo tamaño que la cámara. La carta de color rellena el fondo con un color sólido cuando no se ha incluido ninguna imagen de fondo.
<b>carta de exposición</b>	Hoja con varias columnas verticales y fotogramas horizontales que se utiliza para indicar el tiempo de una escena. Cada columna representa una capa de escena. Los números de dibujos de cada columna se indican y se repiten a lo largo de la cantidad de fotogramas determinada en la que deben aparecer. La carta de exposición la utilizan los animadores, directores y otros miembros del equipo para trazar la secuencia y el tiempo de las imágenes, el diálogo, los efectos de sonido, las pistas de sonido y los movimientos de cámara. Conocida también como carta de rodaje.
<b>carta de fonemas</b>	Carta basada en los ocho fonemas de la animación (A, B, C, D, E, F, G y X, que se utiliza para representar el silencio) utilizados para la sincronización de labios.
<b>carta de rodaje</b>	La utilizan los animadores, directores y otros miembros del equipo para trazar la secuencia y el tiempo de las imágenes, el diálogo, los efectos de sonido, las pistas de sonido y los movimientos de cámara. Conocida también como carta de exposición.
<b>Carta de sprites</b>	Varios sprites aplanados en una sola imagen. Algunas cartas están en una cuadrícula de tamaño fijo y otras tienen sprites de tamaño variable. El motor de juego debe entender dónde se encuentra cada sprite en la carta para poder extraer correctamente los datos de sprite de la carta.
<b>celda</b>	En la animación tradicional, una celda (también denominada celuloide) es una hoja transparente en la que la animación se tinta y se pinta antes de enviarla a la cámara. El trazado de la imagen se dibuja en la parte



	delantera de la celda y, a continuación, se colorea por la parte trasera.
<b>ciclo</b>	Serie de fotogramas de animación que se pueden repetir para simular un movimiento continuo. Por ejemplo, un personaje a menudo camina mediante una animación en la que da dos pasos, uno con cada pie y, a continuación, repitiendo la animación.
<b>ciclo de andar</b>	Serie de dibujos "en el lugar" que describen cómo camina un personaje. La ilusión de movimiento se crea utilizando paneos de fondo. Para evitar realizar innumerables dibujos, los animadores suelen realizar un ciclo de andar para un personaje.
<b>cinemática directa</b>	La cinemática directa es una función utilizada para animar principalmente personajes 3D y marionetas cut-out con jerarquía. Se utiliza para animar una marioneta a partir de una de sus partes padres, como un hombro, y hacer que el resto del brazo se mueva con ella como un todo.
<b>cinemática inversa (IK)</b>	Función utilizada principalmente para animar personajes 3D y marionetas cut-out con jerarquía. La cinemática inversa animará una marioneta desde una de las extremidades, como una mano, y permitirá que el resto del cuerpo la siga con naturalidad.
<b>círculo cromático</b>	Visualización del espectro de colores en forma de círculo.
<b>clean up</b>	Una vez que los bocetos se han evaluado y aprobado, se quita todo el ruido de la imagen (líneas sobrantes, notas, etc.) para crear dibujos finales que se pueden tinter, pintar y filmar. El proceso de clean up hace referencia al trazado de una línea limpia sobre un boceto para conseguir la versión final o a la eliminación de la suciedad y las líneas adicionales que han quedado tras el proceso de digitalización.
<b>CMYK</b>	Acrónimo en inglés de cian, magenta, amarillo y negro. Estos colores forman el modelo estándar utilizado en un proceso denominado impresión offset.
<b>código de tiempo</b>	Información de tiempo impresa en un clip de película para indicar la escena, hora, minuto y segundo que se muestran actualmente en pantalla.
<b>composición</b>	La composición es la acción de incorporar todos los elementos de una escena para crear el resultado final antes del renderizado. Por ejemplo, el artista de la composición importará todas las secuencias, el fondo, las superposiciones y las subcapas de la animación en la escena y los colocará correctamente. A continuación, el artista definirá el encuadre de cámara y lo animará en caso necesario. Por último, el animador creará

	<p>todos los efectos generados por ordenador para el proyecto.</p>
<b>corte</b>	<p>Cuando una escena pasa a la escena siguiente de un fotograma a otro sin ningún efecto visual.</p>
<b>corte de salto</b>	<p>Corte brusco entre dos escenas. Normalmente, un corte de salto no es visualmente agradable. Normalmente lo provoca el final de una escena y el comienzo de otra, con una imagen parecida. La falta de diferencia provoca que el ojo vea un pequeño salto entre las dos escenas.</p>
<b>Create Animator Controller</b>	<p>Opción para crear un Animator Controller cuando el archivo TBG se importa por primera vez</p>
<b>curva Bézier</b>	<p>Método para definir líneas curvas inventado por el matemático francés Pierre Bézier. Una curva Bézier es una curva matemática o paramétrica. Los puntos de control definen las curvas Bézier. En Harmony, las curvas Bézier tienen un punto de inicio, un punto de final y dos puntos de control interiores que influyen la curva entre los puntos de inicio y final.</p>
<b>depuración de sonido</b>	<p>Proceso que permite escuchar el sonido en tiempo real, mientras mueve el cabezal de reproducción hacia delante o hacia atrás. Es muy útil para realizar el ajuste preciso de una sincronización de labios.</p>
<b>desaceleración</b>	<p>Desaceleración gradual en la acción. Conocida también como desaceleración gradual.</p>
<b>desaceleración gradual</b>	<p>Desaceleración gradual en la acción. Conocida también como desaceleración.</p>
<b>descomposición</b>	<p>En animación cut-out, la descomposición es la acción de descomponer un personaje en trozos para crear una marioneta con articulaciones. Para descomponer un personaje, el artista corta partes, como las manos y los brazos, del modelo del personaje y las pega en capas independientes. A continuación, las articulaciones se fijan y se definen los pivotes. En la animación tradicional, una descomposición es una pose de animación que normalmente se encuentra entre dos poses clave. Las poses clave son las poses principales de una animación y las descomposiciones son poses secundarias que ayudan a describir el movimiento y la curva de rotación (normalmente denominada arco).</p>
<b>desglose de pista de audio</b>	<p>Desglose de una pista de película animada en sonidos individuales para generar la posición fotograma a fotograma precisa de cada sonido.</p>
<b>desplazamiento</b>	<p>Pequeño empujón (izquierda, derecha, arriba, abajo, hacia delante o</p>

<b>pequeño</b>	hacia atrás) realizado con las teclas de flecha del teclado en un elemento seleccionado. El desplazamiento pequeño se utiliza para mover una selección de manera muy ligera y precisa.
<b>detección automática de sincronización de labios</b>	Asignación automática de dibujos en un elemento a la carta de fonemas generada para un sonido. Puede ahorrar tiempo al sincronizar los labios una pista de voz.
<b>detección manual de sincronización de labios</b>	Intercambio manual de dibujos de posición de la boca para que coincidan con una pista de voz. Para este proceso, se utilizan tanto la depuración de sonido (escuchar una onda de sonido descompuesta fotograma a fotograma) como las sustituciones de dibujos.
<b>diálogo</b>	Texto hablado por un personaje en una película o animación.
<b>Discretization Step</b>	El nivel de precisión con el que se segmentará un sprite si se deforma. Cuanto mayor sea el número, más suaves serán las formas de los sprites que están muy deformados.
<b>diseño del personaje</b>	Cada personaje de una película animada se dibuja desde varios ángulos en formato de estilo póster denominado hoja de modelo, que sirve como referencia para los animadores.
<b>dpi</b>	Siglas en inglés para puntos por pulgada, que representa la medida estándar de resolución para impresoras computerizadas. En ocasiones se aplica a pantallas, en cuyo caso es más exacto denominarlo píxeles por pulgada. En cualquier caso, el punto es el elemento diferenciado más pequeño que forma la imagen.
<b>eje</b>	Línea imaginaria alrededor de la cual rota un objeto. Para gráficos 2D, hay dos ejes: X (horizontal) e Y (vertical). Para gráficos 3D, hay tres ejes: X (horizontal), Y (vertical) y Z (profundidad). En animaciones que rotan constantemente, el elemento de eje especifica alrededor de qué eje rota el objeto. Un número negativo provoca que una animación rote en sentido contrario a las agujas del reloj, mientras que un número positivo hace que una animación rote a la derecha.
<b>escena</b>	Plano en una película o programa. Una secuencia está compuesta por varias escenas. Una escena cambia a otra mediante un corte simple o por transición.
<b>estilo de paleta</b>	Un estilo de paleta es una segunda versión de una paleta existente con un ligero cambio en el tinte y el valor. Un estilo de paleta se puede utilizar para crear la versión nocturna de una paleta. También se le puede denominar paleta de clonado.

<b>exposición</b>	En animación, una exposición es el número de celdas en las que un dibujo aparece en la escena. Para que un dibujo parezca más grande, la exposición se debe ampliar durante un número de celdas mayor.
<b>FBX</b>	Formato de archivo FBX: formato de archivo estándar en la industria del modelado 3D y el desarrollo de juegos 3D, popularizado por 3ds Max
<b>film-1.33</b>	Resolución idónea para el formato de película panorámico que se ajusta a la proporción estándar de 4:3 píxeles.
<b>film-1.66</b>	Resolución idónea para el formato de película panorámico que se ajusta a la proporción de 16:9 píxeles.
<b>Filter Mode</b>	Cómo mezclar píxeles contiguos: la opción Nearest (Más cercano) muestra bordes gruesos alrededor de los píxeles (capa de píxeles), mientras que Bilinear (Bilineal) y Trilinear (Trilineal) intentan mezclar los datos de color de los píxeles vecinos para suavizar la imagen.
<b>flipping</b>	En la animación tradicional, flipping es la acción de pasar por los dibujos de una secuencia de animación muy rápidamente con el fin de ver la animación en movimiento. Flipping también puede ser la acción de crear una transformación en espejo de un objeto.
<b>Flujo de trabajo de TBG</b>	Proceso para exportar un personaje desde Harmony e importarlo a un motor de juego utilizando archivos TBG como tipo de archivo intermedio.
<b>Flujo de trabajo de XML</b>	Proceso para exportar un personaje desde Harmony e importarlo a un motor de juego utilizando carpetas XML como tipo de archivo intermedio.
<b>fondo</b>	Ilustraciones que llenan el campo de la cámara y que se encuentran detrás de los personajes y props. Normalmente, estas ilustraciones representan la decoración o escena en la que ocurre la acción.
<b>fonema</b>	Unidad de sonido de un idioma.
<b>fotograma</b>	Un fotograma de animación es una imagen fotográfica única en una película. En la animación tradicional, el estándar norteamericano contiene generalmente 24 fotogramas por segundo, mientras que el estándar europeo es de 25.
<b>fotograma clave</b>	Posiciones importantes en la acción que definen los puntos inicial y final de cualquier acción. Un fotograma clave es una posición generada por ordenador en un momento específico (fotograma) y en una trayectoria

	determinada.
<b>fotograma clave de movimiento</b>	Fotograma clave con interpolación generada por ordenador.
<b>fotograma clave de Stop-motion</b>	Fotograma clave sin interpolación generada por ordenador.
<b>fotogramas por segundo</b>	Representan la velocidad a la que se reproducen los fotogramas por segundo. Se calculan por lo general en fotogramas por segundo. Por ejemplo, una escena se podría reproducir a 12, 24, 25, 30 o 60 fotogramas por segundo o a cualquier otro número. Es la medida de la frecuencia (tasa) a la que un dispositivo de imagen produce imágenes consecutivas únicas, denominadas fotogramas. El término se aplica por igual a los gráficos por ordenador, videocámaras, cámaras de película y sistemas de captura de movimiento. La tasa de fotogramas se suele expresar principalmente en fotogramas por segundo (fps) y en monitores de barrido progresivo en hercios (Hz).
<b>Framerate</b>	Número de fotogramas por segundo que se mostrarán al reproducir una animación.
<b>función</b>	Movimiento, trayectoria o ruta generados por ordenador al que se pueden conectar parámetros de elementos, otras trayectorias y efectos. La función se puede controlar agregando fotogramas clave y puntos de control en la curva de función.
<b>fundido de entrada/fundido de salida</b>	El fundido de entrada o fundido de salida es un efecto de transición utilizado para abrir o cerrar una secuencia. Un fundido de entrada se produce cuando la primera escena aparece progresivamente, desde una transparencia total a su opacidad total. Un fundido de salida se produce cuando la escena desaparece progresivamente, desde la opacidad total a la transparencia total.
<b>fundido encadenado</b>	Cuando una escena pasa a la escena siguiente mediante un fundido en el que desaparece una y aparece la siguiente.
<b>gama</b>	Rango de colores que puede representar un dispositivo determinado.
<b>guion</b>	Texto original que contiene la información de toda la película o el programa. En animación, el guion contiene todas las descripciones de la ubicación, el diálogo, el tiempo, etc. Un proyecto comienza con un guion.
<b>HDTV</b>	Acrónimo en inglés de televisión de alta definición, que es el estándar de

	resolución actual para producciones televisivas. La resolución de producciones de HDTV es 1280×720 (HD ready) o 1920×1080 (full HD), y sus fotogramas por segundo normalmente oscilan entre 23.976 y 30.
<b>hoja de campos</b>	Guía que contiene todas las unidades de campo que los artistas de animación y layout utilizan para definir el tamaño del campo de la cámara, así como sus movimientos.
<b>hold</b>	Fotograma en la animación en el cual el personaje mantiene su posición sin moverse. Se puede crear un hold entre dos fotogramas clave cualesquiera.
<b>HSV</b>	Siglas en inglés para matiz, saturación y valor. Método para definir los colores en cuanto a matiz (tinte), saturación (sombra) y valor (tono o luminancia).
<b>Huesos</b>	Segmentos rígidos del cuerpo de un personaje, utilizados como guía para deformar un personaje, generalmente las piernas y los brazos.
<b>Huesos de juego</b>	Deformación en Harmony que simula el funcionamiento de los huesos en los motores de juego, parecido a las deformaciones normales de los huesos en Harmony, pero con un algoritmo de fusión diferente.
<b>IK Manager 2D</b>	Plugin original para Unity (creado por Unity) que facilita la cinemática inversa (IK) para personajes 2D.
<b>imagen en miniatura</b>	Imagen muy pequeña utilizada como una referencia o indicador.
<b>Importer Settings</b>	Configuración disponible en el recurso desde la ventana Inspector que cambia la forma en que el recurso se convierte de datos en el archivo a datos que Unity comprende.
<b>ink and paint</b>	El proceso ink and paint es la acción de pintar las zonas vacías y colorear las líneas en los dibujos de animación finales siguiendo un modelo de color.
<b>Inspector View</b>	Vista en Unity que muestra el contenido de un GameObject: la transformación, componentes para renderizado y física, y comportamientos personalizados.
<b>intercalado</b>	Dibujos que existen entre las poses clave. Se dibujan para crear transiciones fluidas entre poses.

<b>interpolación</b>	Movimiento generado por ordenador creado entre dos fotogramas clave. Se puede crear interpolación, o no, entre los fotogramas clave.
<b>layout</b>	Paso de comunicación entre el storyboard y la animación. Es la acción de poner el storyboard en el modelo. Es decir, el personaje se dibuja siguiendo el diseño del paquete de modelo, de modo que el animador puede comenzar a trabajar. El artista de layout dibuja el fondo y crea la cámara y la guía de campo, haciendo coincidir la escena y el movimiento de la cámara. Por último, el animador dibuja en el modelo las poses de acción principales.
<b>layout y pose</b>	Acción que consiste en poner el storyboard en el modelo (a la escala adecuada) para que el animador comience a trabajar.
<b>librería</b>	Zona de almacenamiento que contiene plantillas y recursos que se pueden reutilizar en un proyecto o escena.
<b>línea de acción</b>	Línea central en una pose de acción. Cuando se pintan poses de acción sin tocar la línea de acción se garantiza que la pose sea dinámica y que su silueta se ajuste a la dirección de la acción.
<b>línea de tiempo</b>	Representación horizontal de los elementos, el tiempo y los fotogramas clave de una escena.
<b>líneas vectoriales transparentes</b>	Cuando no hacen referencia a trazos reales, las líneas vectoriales transparentes son contornos invisibles que se utilizan para definir zonas de color o para cerrar zonas de color abiertas.
<b>mapa de bits</b>	Imagen definida por una malla de píxeles, así como por el color individual de cada uno de sus píxeles. Los gráficos de mapas de bits se caracterizan por ser preferibles a los gráficos vectoriales para ilustraciones muy detalladas o fotorrealistas. Sin embargo, los gráficos de mapas de bits no se pueden escalar, rotar o sesgar sin que la imagen pierda calidad.
<b>Máscaras</b>	Colecciones alternativas de dibujos en la línea de tiempo para una capa. Se pueden crear y alternar utilizando la vista de máscara en Harmony. Por ejemplo, el traje alternativo de un personaje, que puede cambiarse, o nuevas partes del cuerpo de un personaje, que se muestran u ocultan de forma opcional durante el juego.
<b>mesa de disco</b>	Equivalente al disco/mesa de animación, es un dispositivo que permite rotar el espacio de trabajo para mayor comodidad mientras se dibuja.
<b>mesa de luz</b>	Dispositivo que permite ver otras capas en transparencia mientras trabaja

	en una capa determinada.
<b>MipMap</b>	Cuando MipMap está marcado en la configuración de importación en Unity, indica que el usuario desea tener versiones de menor resolución de las texturas generadas para renderizar a escalas más pequeñas/distancias más lejanas.
<b>modelo de color</b>	Diseño de color oficial que se debe utilizar para pintar la animación. Un modelo es el diseño de personaje, prop o ubicación definitivo que cada artista debe seguir para la producción.
<b>modelo/modelo de color</b>	Diseño de personaje, prop o ubicación definitivo que cada artista debe seguir para una producción. Un modelo de color es el diseño de color oficial que se debe utilizar para pintar la animación.
<b>morphing</b>	Función que permite crear dibujos generados por ordenador entre un dibujo de origen y otro de destino. La animación creada con la función de morphing se puede reutilizar en distintos proyectos.
<b>multiplano</b>	Efecto de pasar por diversos niveles de dibujos para crear una sensación de profundidad en un plano. Un multiplano es una escena en la que las capas están colocadas a distintas distancias de la cámara, de modo que, cuando la cámara se mueve, se produce una ilusión de profundidad. Con un multiplano, toda la perspectiva y la escala se calculan automáticamente.
<b>NTSC</b>	Antiguo sistema de difusión televisiva analógico estándar utilizado en Norteamérica.
<b>PAL</b>	Resolución que funciona mejor con el formato europeo para pantallas de televisión y de ordenador, ya que los píxeles rectangulares se muestran con una orientación distinta.
<b>paleta maestra</b>	Grupo de colores atribuidos a un personaje o prop. La paleta se utiliza durante toda la producción para mantener la coherencia en el aspecto y para garantizar que se utilicen los mismos colores durante la producción. Conocida también como paleta.
<b>paleta/paleta maestra</b>	Una paleta o paleta maestra es un grupo de colores atribuidos a un personaje o prop. La paleta se utiliza durante todo el proyecto para mantener la coherencia en el aspecto y para evitar cambios de color durante la animación. También se le denomina paleta maestra.



<b>panear</b>	Mover la cámara atravesando la escena en cualquier dirección.
<b>panel</b>	En un storyboard, un panel es un fotograma en un plano. Un plano puede estar compuesto por uno o varios paneles.
<b>papel cebolla</b>	Función que permite ver los dibujos anterior y siguiente de una secuencia.
<b>paso con doble bote</b>	En los fotogramas clave y las posiciones de paso de un paso con doble bote, el cuerpo es menor que una línea de referencia dibujada en posición vertical. Durante los intercalados, el cuerpo está situado sobre esta línea, ofreciendo así un aspecto de rebote.
<b>peg</b>	En la animación tradicional, es una herramienta utilizada para garantizar el registro preciso de la acción a medida que se mueven las capas de celdas. En la animación digital, en la que se realiza un rigging de marionetas más avanzado, se pueden usar capas de pegs. Las capas de pegs son capas de trayectoria que no contienen dibujos. Son rutas de movimiento que puede usar para agregar articulaciones de ruta. Para las últimas, también puede utilizar una herramienta de cinemática inversa.
<b>pivote</b>	Punto alrededor del cual un peg o dibujo rota.
<b>píxel</b>	Elemento más pequeño de una imagen mostrado en un monitor o pantalla de televisión. El píxel, abreviatura en inglés de elemento de imagen, es un punto único en una imagen gráfica. Es una pequeña muestra de una imagen, un punto, un cuadrado o una sección muy pequeña compuesta de filtrado suavizado. Si acerca el zoom lo suficiente en una imagen digital, verá los píxeles, que parecen pequeños cuadrados de distintos colores e intensidad.
<b>planificación de layout</b>	Dibujo de las funciones principales de la escena que se utiliza como referencia al planificar la animación y ejecutar sus etapas iniciales.
<b>plano</b>	Escena en una película o programa. Una secuencia está compuesta por varios planos. Un plano cambia a otro mediante un corte simple o por transición.
<b>plano de situación</b>	Escena en la cual el espectador puede ver toda el área en la que se está produciendo una secuencia. Por ejemplo, si un niño está jugando en el patio delantero de su casa, el plano de situación sería una escena en la que el espectador puede ver la casa, el patio, parte de la calle y los edificios situados alrededor del punto de acción central. Esto ayuda al espectador a entender la ubicación de la historia y la orientación de la escena.

<b>plantilla</b>	Recurso almacenado en la librería que se puede reutilizar en cualquier proyecto. Una plantilla puede ser un dibujo, una serie de fotogramas clave, un archivo de sonido, un panel, un personaje cut-out, un efecto, una trayectoria, una animación o cualquier otra cosa que se utilice en la animación.
<b>pose clave</b>	Posiciones importantes en la acción que definen los puntos inicial y final de cualquier transición suave. Las claves, o poses clave, son los dibujos principales de una secuencia de animación que describen el movimiento. Por ejemplo, si se agita un brazo, las claves corresponderán al brazo de una extremidad del movimiento de onda y a la otra extremidad. Al realizar un flipping de dichos dibujos, el animador puede ver el esqueleto del movimiento sin tener todos los dibujos.
<b>posición de paso</b>	Al dibujar una secuencia en la que un personaje camina, la posición de paso es el punto en el que una pierna pasa a la otra.
<b>Prefab</b>	Jerarquía de GameObjects, transformaciones, componentes y comportamientos que pueden construirse en la vista de jerarquía y luego guardarse en la vista de proyectos. Se pueden poner instancias de prefabs en la jerarquía para tener varios duplicados del mismo objeto (muchos personajes, muchas balas, muchos árboles).
<b>Preview Window</b>	Sección de la pestaña Inspector que muestra el aspecto del recurso, generalmente con modelos, materiales y texturas.
<b>proceso de layout</b>	Paso de comunicación entre el storyboard y la animación.
<b>proporción</b>	Proporción entre las dimensiones de ancho y altura de una escena, fotograma o formato de película. Actualmente, la proporción más popular es 16:9. Las pantallas antiguas utilizaban una proporción de 4:3.
<b>renderizado</b>	Paso final de la animación por ordenador. Durante el renderizado, el ordenador toma cada píxel que aparece en la pantalla y procesa todos los componentes. Además, agrega desenfoque de movimiento antes de que produzca una imagen final. Es el proceso de calcular las imágenes finales después del proceso de composición.
<b>resolución</b>	Tamaño de una escena, calculado por lo general en píxeles. Por ejemplo, la resolución NTSC es de 720 x 480. La resolución debe coincidir con la salida final: HDTV, film-1.33, film-1.66, NTSC, PAL, baja.
<b>RGB</b>	Siglas en inglés para rojo, verde y azul. Método que permite definir el color especificando las cantidades de estos tres componentes del color.

<b>rigging</b>	Proceso de conectar las distintas partes de una marioneta cut-out.
<b>rotoscopia</b>	Técnica de animación en la que los animadores trazan un movimiento de película de acción en vivo, fotograma a fotograma, para usarlo en películas animadas. Es el acto de esbozar sobre material de acción en vivo para crear una secuencia animada.
<b>Scene View</b>	Visualización del juego en Unity. Complementa la vista de jerarquía, donde la Scene View muestra cómo se verá para un jugador, y la jerarquía muestra cómo se ve el juego para el motor.
<b>secuencia</b>	Serie de escenas o planos que forman una parte distinta de la historia o película, normalmente conectados por la unión de la ubicación o el tiempo.
<b>Shadergraph</b>	Formato de archivo de Unity que Unity proporciona con una interfaz de usuario y un importador de comandos, y que permite la creación visual y la edición de shaders sin necesidad de escribir código.
<b>símbolo</b>	Un icono combina animación, ilustración o capas en un único objeto que puede controlar en una capa. También puede crear iconos de cada parte del cuerpo en marionetas cut-out. En un icono puede colocar lo que desee. Puede utilizar símbolos para animar una marioneta o crear animaciones reutilizables, como un parpadeo.
<b>sincronización de labios</b>	Proceso que consiste en sincronizar la boca de un personaje con sonidos en la pista de sonido del diálogo. La boca se ajusta fotograma a fotograma para hacer coincidir el sonido del diálogo y aparentar que el personaje está hablando. La sincronización de labios se puede utilizar para cualquier secuencia de sonido y no solo para el habla. Por ejemplo, puede hacer que un pájaro pío o que un lobo aúlle a la luna.
<b>slugging</b>	Indicación de los tiempos de inicio y detención de un diálogo y las acciones relevantes.
<b>Sombreador</b>	Instrucciones que informan al juego de cómo mostrar los datos de malla y textura proporcionados (colores, iluminación, suavidad, curvatura, mezcla, etc.)
<b>Sprite Atlas</b>	La versión de Unity de una carta de sprites toma una colección de sprites y los aplana en una sola imagen. Cada sprite del atlas señalará los datos en el atlas durante el renderizado, lo que reduce la sobrecarga de rendimiento de muchos sprites diferentes.

<b>sRGB Texture</b>	Cuando sRGB está marcado en la configuración de importación en Unity, le indica a Unity que los datos en la textura se codificaron en espacio de color lineal, que coincide con el modo de renderizado lineal de Unity.
<b>Stepped</b>	Indica si mantener la animación a la frecuencia de fotogramas prevista por el artista o adivinar los fotogramas intercalados cuando se muestran a una frecuencia de fotogramas mayor.
<b>storyboard</b>	Plan visual de todas las escenas y planos de una animación. El storyboard indica lo que sucederá, cuándo lo hará y cómo se distribuyen los objetos en una escena.
<b>Sub-Asset</b>	Recurso que se empaqueta dentro de un recurso existente, y, por lo general, se asocia con el recurso principal. En el caso de los archivos TBG, un personaje consta de un recurso prefab y varios sub-assets para texturas, sprites, AnimationClips, materiales y TBGStore.
<b>tableta/lápiz</b>	Dispositivo utilizado junto con un ratón, o en lugar del mismo, para mover el puntero (en ocasiones denominado cursor) por la pantalla del ordenador.
<b>TBG</b>	El formato de archivo de juego de Toon Boom (Toon Boom Game o .tbg), se creó a partir de Harmony y Unity lo incorporó, de la misma forma que en el futuro lo harán otros motores de juego.
<b>TBG Renderer</b>	Comportamiento en tiempo de ejecución adjunto a un prefab durante el proceso de importación de TBG que gestiona el cambio de resoluciones, paletas, máscaras y la actualización de los sprites de cuchilla.
<b>TBG Store</b>	Recurso generado durante la importación de TBG que almacena datos útiles acerca del personaje a los que el TBG Renderer hace referencia durante el tiempo de ejecución.
<b>Totales acumulados de columna</b>	Campo de texto que contiene información de diálogo, efectos, sonido o slugging en un storyboard.
<b>transición</b>	Efecto colocado entre dos escenas cuando se pasa de una a otra. Los efectos de transición comunes son el fundido encadenado y el barrido.
<b>trayectoria</b>	Ruta o trayectoria generada por ordenador que pueden seguir los elementos. La trayectoria se puede controlar mediante puntos de control, fotogramas clave y velocidad.

<b>trazar y pintar</b>	Una vez que los bocetos de las animaciones se han limpiado y se ha realizado una prueba de línea o de lápiz final, cada dibujo se traza y se pinta para la animación final. En el mundo digital actual, esto se puede realizar de maneras distintas a los métodos de celuloide o acetato tradicionales.
<b>Unity Asset</b>	Los fragmentos de datos en Unity, que genera Unity y son nativos de Unity, y se empaquetan en un juego durante el proceso de creación del mismo. Los generan las herramientas de Unity y durante la importación de archivos externos.
<b>Unity Custom Importer</b>	Importador en Unity que se define en la interfaz de comandos de C# por proyecto, que asigna una extensión de archivo (p. ej., «.tbg») a un proceso de conversión de datos de archivo en Unity Assets.
<b>usar carta de rodaje</b>	Asignar un dibujo determinado a un rango de fotogramas.
<b>vector</b>	Una imagen basada en vectores está compuesta por puntos y curvas Bézier. El ordenador lee los puntos y traza los segmentos, vinculándolos para reproducir la forma de imagen. Una imagen vectorial no contiene tamaño fijo ni resolución. El gráfico se puede ampliar y distorsionar según se desee y el sistema volverá a calcular los segmentos y reconstruirá las formas. Las imágenes vectoriales se trasladan y se muestran en píxeles una vez que el cálculo se ha realizado.
<b>velocidad</b>	En animación, la velocidad es la aceleración o la desaceleración de un movimiento. Se puede alcanzar mediante una curva de función o una serie de dibujos animados. Otros términos comunes para aceleración gradual y desaceleración gradual son aceleración y desaceleración.
<b>vibración de cámara</b>	La vibración de cámara se produce en una escena cuando la cámara se mueve ligeramente y rápidamente en varias direcciones. Este movimiento da la impresión de impacto, vibración o, por ejemplo, baches en la carretera.
<b>XML</b>	Formato de archivo XML: formato de archivo estándar para almacenar datos de forma estructural, parecido en su funcionamiento al formato JSON
<b>zona</b>	Área que se puede pintar con color.
<b>zona segura</b>	Zona central del fotograma de una escena que no quedará recortada por el fotograma de TV. Como un fotograma de TV corta un margen del tamaño del fotograma original, al mantener una zona segura se garantiza que la acción principal de la escena siga siendo claramente visible una

vez que la película se proyecte en televisión.



---

# Índice

## A

animación  
    varias secuencias, juegos 23

## E

exportación  
    fotograma a fotograma, juegos 12

## J

juegos 10  
    animación de varias secuencias 23  
    directrices 14  
    exportación de datos de juego en bruto 11  
    exportación fotograma a fotograma 12  
    Harmony 10  
    recursos 14



