

# Curso LDraw, 12ª entrega

## Gestión de MILS con BlueBrick

por Jetro



En HispaBrick Magazine® 005 presenté el programa BlueBrick, un gestor visual de trazados y dioramas que puede facilitar mucho el trabajo en equipo para la presentación de eventos. Como en este número se ha presentado el concepto colaborativo de los MILS aprovecho esta entrega del tutorial sobre LDraw y herramientas asociadas para mostraros cómo emplear BlueBrick para organizar dioramas colaborativos con MILS.

### ¿Por qué Bluebrick?

El concepto MILS está pensado para dos cosas:

- 1) es modular, lo que implica que se puede construir en tamaños y configuraciones diferentes
- 2) facilita la colaboración; de lo contrario no haría falta seguir un estándar común.

Esto significa que en un diorama creado con la ayuda de MILS se puede adaptar fácilmente al espacio disponible y permite la colaboración de varias personas en un mismo diorama. Empleando BlueBrick esas colaboraciones se pueden planificar de forma sencilla sin que los participantes del diorama tengan que reunirse físicamente.

### Crear MILS para BlueBrick

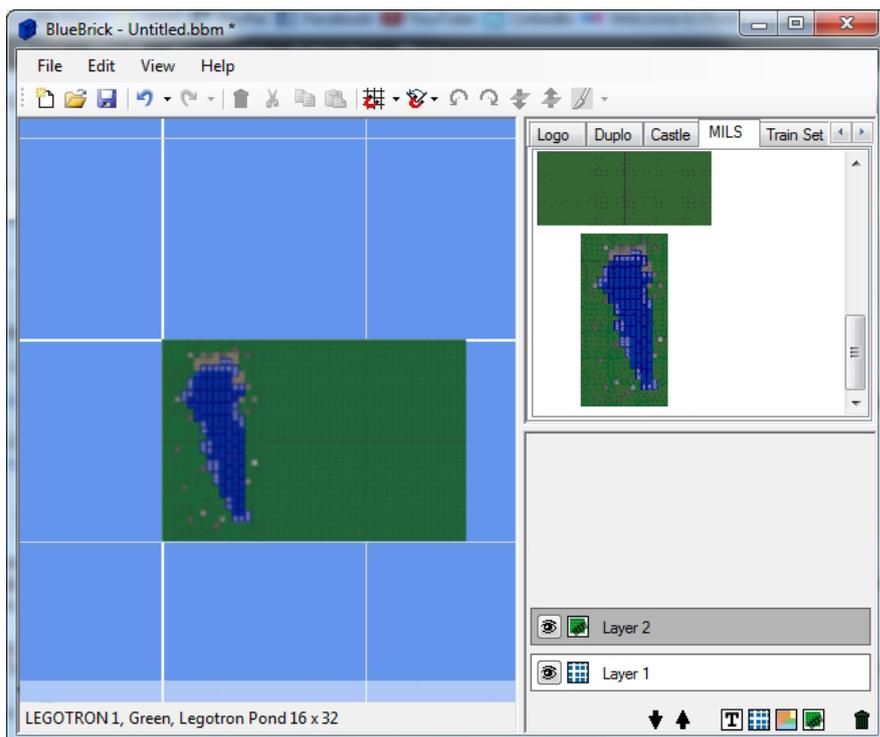
Empecemos con lo más sencillo: una BTM lisa[1]. Como la vista superior de una BTM es idéntica a la de un baseplate 32x32 podemos aprovechar esa imagen (ya disponible en BlueBrick) para crear la primera BTM. Si aún no tienes BlueBrick puedes descargarlo desde su página web[2]. El programa no requiere de ninguna instalación ya que se ejecuta desde la misma carpeta en la que se extrae[3]. Para simplificar las cosas en este tutorial todas las ubicaciones de archivos se indicarán a partir de la carpeta "BlueBrick".

Los elementos de la biblioteca de piezas de BlueBrick se encuentran ubicadas en \BlueBrick\parts. Notarás que dentro de esa carpeta hay subcarpetas para cada tipo de elemento (por ejemplo \BlueBrick\parts\Baseplate). Estas carpetas son las que determinan el nombre y la cantidad de pestañas del navegador de elementos que se encuentra en la parte superior derecha de la ventana del programa BlueBrick. Como en este tutorial vamos a crear un nuevo tipo de elemento empezaremos por crear una nueva carpeta llamada MILS (\BlueBrick\parts\MILS – la

carpeta no se verá como pestaña hasta la siguiente vez que iniciamos BlueBrick). De la carpeta \BlueBrick\parts\baseplate copiaremos los archivos 3867.2.gif y 3867.2.xml y las pegaremos en la carpeta \BlueBrick\MILS que habremos hecho anteriormente. Como se trata de un Módulo de Terreno Básica renombraremos los archivos a BTM.2.gif y BTM.2.xml respectivamente[4].

A continuación vamos a modificar el archivo XML asociado a esta imagen. Para poder modificar el archivo hay que abrirlo en un editor de texto como por ejemplo el Bloc de notas (en Windows esto se consigue haciendo un clic derecho sobre el archivo y seleccionando "Abrir con" del menú emergente donde se nos presentará una selección de programas capaces de abrir el archivo). Al abrir el archivo notaremos enseguida que refleja el nombre del elemento/módulo.

Entre las etiquetas <Description></Description> aparecen dos parejas de etiquetas identificativas de idioma. De momento están disponibles las etiquetas de inglés <en> y francés <fr>. Si quieres preparar el archivo para futuras traducciones del programa puedes añadir la etiqueta <es></es> para español. Lo que aparece entre estas etiquetas es el nombre que aparece en la barra de estado de BlueBrick al colocar el cursor sobre la imagen en miniatura del módulo en la pestaña MILS de BlueBrick. Modifica el texto que se encuentra entre esas parejas de etiquetas para que ponga BTM en todos los casos.



La información que aparece más abajo en el archivo XML es relativa al código de este elemento en Track Designer(TD). Como las BTU no existen en TD, pero en este caso es idéntico que un baseplate 32x32, podemos dejar la información tal cual para asegurar una mayor compatibilidad del elemento con otros programas.

## Crear un módulo personalizado

El siguiente paso sería crear una representación de un módulo personalizado. Entre las imágenes del artículo sobre MILS hay una con un estanque. ¿Cómo podríamos conseguir tener ese módulo disponible en BlueBrick? Hay diferentes maneras de hacerlo y para este tutorial describiré solo una que me ha dado buenos resultados.

El primer paso sería recrear la vista superior del módulo en un editor LDraw (MLCad, LeoCAD, LDCad, Bricksmith...). Para este tutorial he recreado el módulo en MLCad. Como lo que necesitamos es una vista superior con la menor distorsión posible y con una iluminación neutra, un método sencillo es la de hacer una captura de pantalla de la vista superior del módulo directamente desde MLCad. Para ello, primero maximiza la ventana de MLCad y amplía al máximo el área de vista superior del módulo. Luego abre el menú contextual (clic derecho) y selecciona el nivel de zoom "fit" (ajustar) para ver el módulo al tamaño máximo posible en esa ventana.

Pulsando el botón "Impr pant" se hace una captura de pantalla que habrá que recortar hasta dejar solo el módulo. Para ello he usado el editor de imágenes GIMP [5] un programa libre con licencia GNU. Abre un lienzo nuevo con un tamaño igual o superior a la resolución de pantalla que usas. Ahora pega el pantallazo que hiciste anteriormente (Ctrl+V o Editar > Pegar). A continuación escoge la herramienta de selección de un rectángulo (arriba en el panel flotante derecho) y recorta grosso modo el módulo, dejando solo blanco alrededor del mismo. En el menú "imagen" encontrarás 3 herramientas muy útiles que usarás a continuación. La primera es "Recortar a la selección" que dejará solamente el área que habías seleccionado anteriormente. Ahora que solo hay un borde blanco alrededor de la imagen, puedes usar la opción "Autorecortar imagen" que eliminará el borde blanco de forma automática. Finalmente hay que ajustar el tamaño de la imagen al tamaño que necesita BlueBrick: la imagen debe medir 8 pixeles por cada stud de ancho o alto que tiene. En el caso de una baseplate de 32x32 el tamaño de la imagen debería ser de (8x32) 256x256. Esto se puede hacer con la opción "Escalar la imagen...". Ya solo falta convertir la imagen a formato GIF. Esto se hace desde Archivo > Guardar como y seleccionando el tipo de archivo que se desea.

Ahora ya solo falta el archivo XML correspondiente. Al igual que hemos hecho en el ejemplo del BTM, es conveniente buscar un archivo XML existente de un elemento que se parece al BTM personalizado que acabamos de crear. Los pasos a seguir a continuación son los mismos que se detallan más arriba. Para facilitar el trabajo en grupo sería conveniente usar un nombre de módulo que use una identificación de su autor como primer elemento. Como BlueBrick muestra los elementos en orden alfabético, así nos aseguraremos de que los módulos de un mismo autor estarán agrupados. Hasta aquí ha llegado el artículo sobre MILS y por lo tanto hasta aquí llega esta entrega del tutorial LDraw. En el siguiente número seguiré los pasos explicados en la segunda entrega del artículo sobre MILS para adaptarlo a la gestión de módulos en BlueBrick.

[1] El ejemplo que sigue es una manera sencilla de crear una BTM a partir de un elemento existente. Puedes descargar una BTM creada específicamente para el grupo MILS (imagen y archivo XML) en la página de HispaBrick Magazine®.

[2] <http://bluebrick.lswproject.com/>

[3] La versión más reciente de BlueBrick en el momento de escribir este artículo es la 1.7.1. En esta versión se han hecho algunas modificaciones con el objeto de hacerla plenamente compatible con Mono y poder usar el programa BlueBrick tanto bajo Linux como Windows. Si tienes una versión anterior a la 1.7.1. es altamente recomendable actualizar a una versión más reciente para asegurar la plena compatibilidad de todos los elementos.

[4] BlueBrick extrae información sobre los elementos de diferentes sitios. El color de un determinado elemento se indica en el propio nombre del archivo. Un baseplate 32x32 (código BL 3811) lleva como nombre de elemento (y por tanto del archivo GIF y XML) un sufijo que indica su color. De este modo un baseplate 32x32 azul tiene como nombre 3811.1, uno verde 3811.2 y uno tan 3811.19. El código de color proviene de la numeración de los colores en LDraw. Para distinguir una BTU verde (para un diorama con campo) de uno blanco (para un diorama con nieve) se puede usar este sufijo en el nombre del elemento.

#

