

LbaNet map file information

In this document will be explained how the map data is stored and read by the LbaNet game. It will provide useful information for people who are interested in creating their own world and add it to the game.

Data directory structure

The data directory is the central place where all maps information is stored to be read by the game. It has the following structure:

- Bricks directory: contains all brick images used by the game. Those images are currently in .png format but could also be in .tga format. If you want to create new bricks and add them to the game this would be the place to put them. I will come back at later stage on how to add bricks to the game.
- Maps directory: contains one text file per map describing how bricks are put together on the map. This is the place to put newly created map information. I will come back later to how map information is stored into the text file.
- Music directory: contains all music of the game. Music can be off format .mid .mp3 .wav .ogg. This is the place where to put any new music.
- Some remaining HQR files: I could not get rid of those one as they are used to load character and animation. Will certainly be removed some time later when I will rework the animation part.
- Xml files: there will be one xml file per world. It means that when you want to create a new world that will be the place to put you xml world description file. I will come back later on how the information is stored inside the xml file.
- lba_brick_info.txt: this file is pretty important. It contains a list of bricks of the game. Details will be explained later.

Tips on developing new world

Before going into details, I would like to suggest you not to overwrite or update existing inside directory of the Data directory. Indeed, the directory already there contains information about the original LBA1 and LBA2 games.

As an example, if I want to create a new brick. I will not put my brick into /Data/Bricks/Lba1/. I will create a new directory called /Data/Bricks/Myworld/ and put the brick file inside. This way I am sure that I will still be able to play to the original LBA1 world.

Same for creating new map, I will create a new directory /Data/Maps/Myworld/ and add my file Mymap.txt into it.

Important notice, before editing the lba_brick_info.txt file and add new brick into it, please send me a pm or an email in order for me to give you unique brick indexes that you can use for your new bricks. Just give me the number of bricks that you created and I will give you a range of indexes. Indeed, if two people are not doing that and create new bricks at the same time with the same indexes, they will get into trouble when merging their work.

How to add new brick image to the game

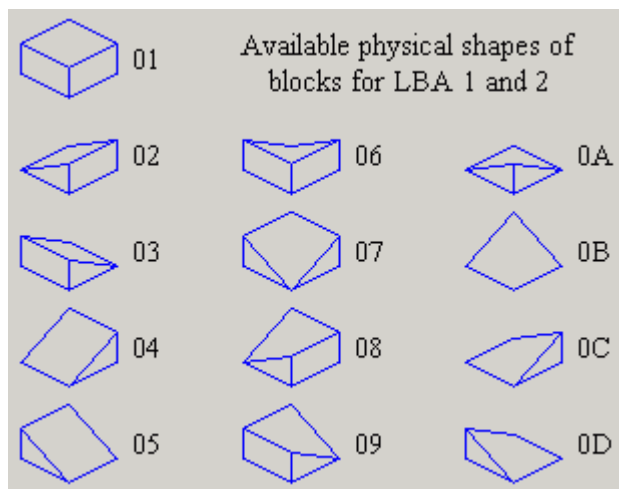
First, the brick image needs to have a width of 48 pixels and a Height of 38 pixels to be correctly read by the game. The image format can be .png or .tga. The image support alpha channel for transparency, it basically means that you can now create kind of glass bricks where people could see trough.

- Copy the new bricks into the brick directory (e.g. /Data/Bricks/Myname/newbrick.png)
- Edit the file /Data/lba_brick_info.txt with a text editor. The format of the file is the following:

Each line contain one brick, the first column represent the brick index, the second represent the path of the brick image file, the third represent the brick shape.

Brick shape can be of the following:

- 0: no shape, twinsen can pass trough it
- 1: solid structure
- 2: stairs: top left edge at the ground
- 3: stairs: top right edge at the ground
- 4: stairs: bottom left edge at the ground
- 5: stairs: bottom right edge at the ground
- 6: double-sided stairs 1 (corner at the ground, two egdes at the ceiling): top corner at the ground
- 7: double-sided stairs 1: bottom corner at the ground
- 8: double-sided stairs 1: left corner at the ground
- 9: double-sided stairs 1: right corner at the ground
- 10: double-sided stairs 2 (two edges at the ground, cornet at the ceiling): top corner at the ground
- 11: double-sided stairs 2: bottom corner at the ground
- 12: double-sided stairs 2: left cornet at the ground
- 13: double-sided stairs 2: right corner at the ground
- 15: water, gas, fire (twinsen die when stepping on it)



So for each new brick that you created and added to the brick directory, you will need to add a line in this file.

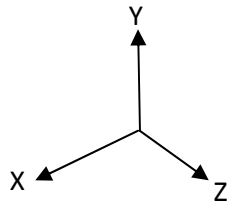
Important notice, before editing the lba_brick_info.txt file and add new brick into it, please send me a pm or an email in order for me to give you unique brick indexes that you can use for your new bricks. Just give me the number of bricks that you created and I will give you a range of indexes. Indeed, if two people are not doing that and create new bricks at the same time with the same indexes, they will get into trouble when merging their work.

How to create new map

If we simplify it, a map is nothing more than a cube of brick image put together. So basically the text file used to describe a map is just storing the representation of this cube.

Now lets get into the details and let me show explain it the way information are stored there. I would suggest that you open a map file with a text editor to have a look at it during you read this section (e.g. Data/Maps/Lba1/ map0.txt).

The first line of the text describe the size of the cube in the following order sizeX(length) sizeY(height) sizeZ(width)



In the original game the map size was always 64 25 64. But now there is now size limit. However keep in mind that the bigger the map is, the more place it will take in memory, so a very big map can be a problem for people having old computers.

The second line of the file represents the number of different bricks that will be used for the map. So if for example your map only uses 10 different brick image, you would put the number 10 there.

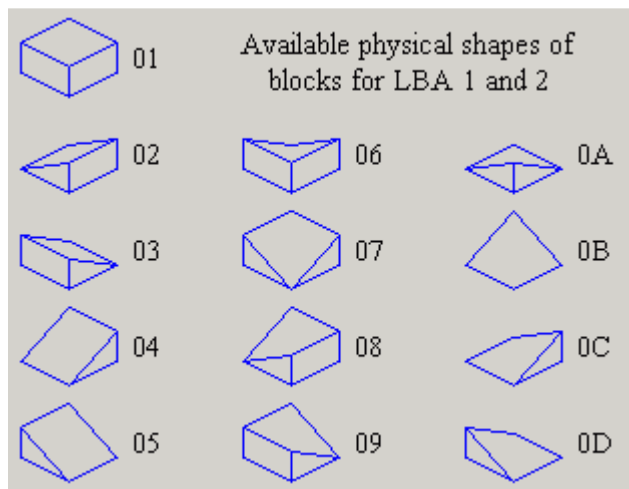
The next section of the file gives the list of all brick image that will be used by the map. So again, let say you have your 10 bricks, here you will put the 10 indexes of the bricks you are using. In order to know the brick index of a brick image, please refer to the file "lba_brick_info.txt" and to the previous section of this document.

Finally the last section of the file represents the map cube information. For each "floor" of your map (depends of your map height) you will have a matrix of indexes, each indexes will represent the brick to display at position X Y Z of the map.

But be careful, the indexes used there are not the index of the "lba_brick_info.txt" file but the index of your brick list you defined first in the file. However, the indexes from 0 to 13 are reserved indexes for empty bricks. So you will need to add a 14 offset to your index before adding it to the matrix.

So what are those reserved indexes for? They are used to represent invisible bricks. Meaning brick having now image, they will be completely transparent. Therefore they are not listed in the lba_brick_info.txt file. Here are the indexes used:

- 0: invisible brick with no shape
- 1: invisible brick with solid structure
- 2: invisible brick with stairs: top left edge at the ground
- 3: invisible brick with stairs: top right edge at the ground
- 4: invisible brick with stairs: bottom left edge at the ground
- 5: invisible brick with stairs: bottom right edge at the ground
- 6: invisible brick with double-sided stairs 1
- 7: invisible brick with double-sided stairs 1: bottom corner at the ground
- 8: invisible brick with double-sided stairs 1: left corner at the ground
- 9: invisible brick with double-sided stairs 1: right corner at the ground
- 10: invisible brick with double-sided stairs 2
- 11: invisible brick with double-sided stairs 2: bottom corner at the ground
- 12: invisible brick with double-sided stairs 2: left corner at the ground
- 13: invisible brick with double-sided stairs 2: right corner at the ground



As I am sure that you got lost at this point, let me make a small example of a fake map.
Let's create a map called "smallmap.txt".

The map would contain the following information

4 2 4

3

55

56

57

0 0 0 0

0 14 15 0

0 16 14 0

0 0 0 0

1 1 1 1

2 2 2 2

3 3 3 3

4 4 4 4

Lets take it line by line

4 2 4 the map will have a size of 4 for X 2 for Y 4 for Z

3 I will use 3 different brick in my map

55 the first brick I will use

56 the second brick I will use

57 the third brick I will use

0 0 0 0 Here start my first floor (1 of 2 Y)
for X=1 and Z from 1 to 5 define the brick index (in this case the whole column contains
empty bricks.

0 14 15 0 two empty brick and two other brick 14 represent the first brick that I defined before
(brick 55 of "lba_brick_info.txt") (index starting from 0 with a 14 offset) 15 = brick 56

0 16 14 0 16 = brick 57 14= brick 55

0 0 0 0 empty bricks

1 1 1 1 empty bricks with a solid shape (use for invisible wall)

2 2 2 2 empty bricks with a stair shape (use for invisible stair)

3 3 3 3 empty bricks with a stair shape (use for invisible stair)

4 4 4 4 empty bricks with a stair shape (use for invisible stair)

Hope it is clearer by now ;-)

Anyway, the best ways to understand is by creating your own map and try them in game.

Just start with a very small map to see how things work then you can extend later.

Of course I know that a visual editor would be very welcome there. But for now there is no such thing so you can only edit map manually. Another way is to edit maps with LBArchitect, then convert them into xml with a tool. I will explain how to work with this tool at later stage.

How to create a new world

So now you know everything of how to create you own maps and you are ready to put the maps you created together into your own world. Or you are too lazy to do the map stuff and you just want to create a world with original LBA1 and LBA2 maps putting them together in different order.

So how do we do that, simple the whole world description will be into a unique xml file. So there will be one xml file per world. If you put your newly created xml file into the data directory and restart the game, you will be able to directly play it without having to change the game executable at all :-P

So now lets put our hands in the dirt, what exactly this xml file contain? I would suggest you to open Data/ GiantCitadelWorld.xml with a text editor while reading this section so that you can have an example. (xml is a bit similar to html, so if you now the one it should not be hard to understand the other one)

Here is the basic structure of the lba world xml file:

`<?xml version="1.0" ?>` do not mind this line, it just need to be present on top of every xml files

`<World name="Lba1" firstmap="Map0_1" firstsparea="FirstSpawning">` tag used to describe your world
I come back to that in a moment

`<description>Reconstruction of the LBA1 original world.</description>` here you can enter a short description of your world, what was your idea behind it. It will be used at later stage by the user to select the world he wants to play

`<maps>` tag used to begin the listing of maps

... here you will find a list of maps, I will come to that part later

`</maps>` tag used to end the listing of maps

`</World>` tag used to end your world

Important notice, when you open a tag, please make sure that I get closed before the end of the file otherwise you xml file will not be valid.

Not lets come back to this line:

```
<World name="Lba1" firstmap="Map0_1" firstsparea="FirstSpawning">
```

In this line, you will first give a name to your world e.g. **name="whatever_you_want"**

The second part is used to define the first map the player will arrive when he start the game e.g. **firstmap="myfirstmap"**. The name of the map need to be related to one map of the list of maps you will define afterwards, but I will come back into that.

The third part define the position in the map the player will arrive when he start the game. This position is defined by an identifier. We will see at later stage what this identifier refer to.

e.g. **firstsparea="my_first_spawning_place"**

So here is the result so far:

```
<?xml version="1.0" ?>
<World name="myfirstworld" firstmap=" myfirstmap" firstsparea="my_first_spawning_place">
  <description>My first world, that is awesome!</description>
  <maps>
  </maps>
</World>
```

It is actually a valid xml file, which describe a world named myfirstworld. The world is currently totally empty, there is not map bound to it. So now let see how to add a map to our world.

Here is the structure of a map:

```
<Map name="Map0_1" type="interior" music="Music/Midi_mi_win18.midi"> map begin tag
  <description>Citadel Island, Prison</description> short description of the map
  <files> files begin tag
    <file name="Maps" path="Maps/Lba1/map0.txt" /> give the path of the map info file to load
  </files> files end tag
  <spareas> spawning area begin tag
    <sparea name="FirstSpawning" posX="340" posY="40" posZ="340" RotationAtArrival="0" />
  </spareas> spawning area end tag
  <exits> exit begin tag
  </exits> exit end tag
</Map> map end tag
```

Lets first focus on the first line :

```
<Map name="Map0_1" type="interior" music="Music/Midi_mi_win18.midi">
```

The first part give a name to the map e.g **name="myfirstmap"**. This name should be a unique identifier used as channel name for the chat.

The second part defines whether the map is of "interior" or "exterior" type.

e.g. **type="exterior"**

The third part gives the path of the music to be played when the player is on this map

e.g. **music="Music/Midi_mi_win18.midi"**

Then come next:

```
<files>  
  <file name="Maps" path="Maps/Lba1/map0.txt" />  
</files>
```

This part provides the path of the map info file to be read by the game in order to obtain the map cube information. Please refer to the section "How to create new map" for more details on those files.

In this example the path is "Maps/Lba1/map0.txt" which is the description file of the LBA1 citadel prison.

Let's move to next part:

```
<spareas>  
  <sparea name="FirstSpawning" posX="340" posY="40" posZ="340" RotationAtArrival="0" />  
</spareas>
```

This part defines all possible spawning area of twinsen in the map. Each map should have at least one spawning area otherwise the player will never be able to arrive there. However you could have more than one spawning area depending on where you come from. For example the LBA1 citadel prison map has 2 spawning area, one from the big doors upstairs and one from the electricity panel on the basement.

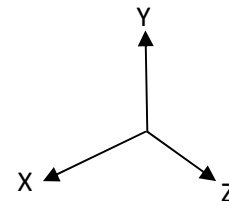
So first part define the name of the spawning area e.g. **name="FirstSpawning"**

Followed by that, you define the position of the spawning point in the map

e.g. **posX="340" posY="40" posZ="340"**

the value should be multiplied by a factor 10

in this example, twinsen will arrive at position X:34 Y:4 Z:34 of the map



RotationAtArrival="0" is currently not used in the game you can just leave it like this.

So now back to our previous example, let's add a map to our world:

```
<?xml version="1.0" ?>
<World name="myfirstworld" firstmap="myfirstmap" firstsparea="my_first_spawnning_place">
  <description>My first world, that is awesome!</description>
  <maps>
    <Map name="myfirstmap" type="interior" music="Music/Midi_mi_win18.midi">
      <description>Citadel Island, Prison</description>
      <files>
        <file name="Maps" path="Maps/Lba1/map0.txt" />
      </files>
      <spareas>
        <sparea name="my_first_spawnning_place" posX="340" posY="40" posZ="340"
          RotationAtArrival="0" />
      </spareas>
      <exits>
      </exits>
    </Map>
  </maps>
</World>
```

Here we are, we now have our first world containing one map. You can try to save this to a text file, rename it with the .xml extension and put it into the data directory. Then start the game to see the result.

I highlighted 2 things in this file which are important to understand. Remember when we were defining our world in the beginning we had to tell in which map the user will arrive when connecting to our world. e.g. **firstmap="myfirstmap"**. So now we have to make sure that the name of our added map is the same as the name we writted down as the value of the firstmap field. (See the red part of the text)

Again, when defining our world we had to enter the name of our first spawning area giving the position on the map when connecting to the game (check the orange part of the text).

So now we have our first world with one map, we can already play on it. But it will soon get boring as we have only one map in there. So let's add some more maps and see how to make twinsen being able to go from one map to another one.

We already know how to define a map, so let's add one to the list.

```
<?xml version="1.0" ?>
<World name="myfirstworld" firstmap="myfirstmap" firstsparea="my_first_spawnning_place">
  <description>My first world, that is awesome!</description>
  <maps>

    <Map name="myfirstmap" type="interior" music="Music/Midi_mi_win18.midi">
      <description>Citadel Island, Prison</description>
      <files>
        <file name="Maps" path="Maps/Lba1/map0.txt" />
      </files>
      <spareas>
        <sparea name="my_first_spawnning_place" posX="340" posY="40" posZ="340"
          RotationAtArrival="0" />
      </spareas>
      <exits>
      </exits>
    </Map>

    <Map name="mysecondmap" type="exterior" music="Music/Midi_mi_win17.midi">
      <description>Citadel Island, out of Prison</description>
      <files>
        <file name="Maps" path="Maps/Lba1/map1.txt" />
      </files>
      <spareas>
        <sparea name="spawnning1" posX="370" posY="20" posZ="130" RotationAtArrival="0" />
      </spareas>
      <exits>
      </exits>
    </Map>

  </maps>
</World>
```

Ok now we have 2 maps in our world, but twinsen is still stuck in the first one, how is it possible to tell the game that we want twinsen to exit the first map to go to the second one. Here is where the exit part of the map description comes into play.

Here is an example:

```
<exits>
  <exit name="myfirst_exit" TopRightX="38" TopRightY="20" TopRightZ="15"
    BottomLeftX="37" BottomLeftY="14" BottomLeftZ="10" newMap=" mysecondmap "
    spawning="spawning1" />
</exits>
```

So lets see in details what it means:

name="myfirst_exit" give a name to your exit

TopRightX="38" TopRightY="20" TopRightZ="15" BottomLeftX="37" BottomLeftY="14"

BottomLeftZ="10" this define a the position and the size of a exit zone inside the map. If twinsen enter the exit cube by any side, he will exit the map.

newMap=" mysecondmap " give the name of the destination map when we exit

spawning="spawning1" give the name of the spawning area where twinsen will appear in the new map

So now let's add an exit to our first map and make the exit go to the second map.

```
<?xml version="1.0" ?>
<World name="myfirstworld" firstmap="myfirstmap" firstsparea="my_first_spawning_place">
  <description>My first world, that is awesome!</description>
  <maps>

    <Map name="myfirstmap" type="interior" music="Music/Midi_mi_win18.midi">
      <description>Citadel Island, Prison</description>
      <files>
        <file name="Maps" path="Maps/Lba1/map0.txt" />
      </files>
      <spareas>
        <sparea name="my_first_spawning_place" posX="340" posY="40" posZ="340"
          RotationAtArrival="0" />
      </spareas>
      <exits>
        <exit name="myfirst_exit" TopRightX="38" TopRightY="20" TopRightZ="15"
          BottomLeftX="37" BottomLeftY="14" BottomLeftZ="10" newMap="mysecondmap"
          spawning="spawning1" />
      </exits>
    </Map>

    <Map name="mysecondmap" type="exterior" music="Music/Midi_mi_win17.midi">
      <description>Citadel Island, out of Prison</description>
      <files>
        <file name="Maps" path="Maps/Lba1/map1.txt" />
      </files>
      <spareas>
        <sparea name="spawning1" posX="370" posY="20" posZ="130" RotationAtArrival="0" />
      </spareas>
      <exits>
      </exits>
    </Map>

  </maps>
</World>
```

Here we are, so now we have two maps connected to each other. The rest is easy, you just need to rinse and repeat for each map you want to add to the world. Have fun creating your own map. And do not forget to share your creation on the MBN forum!