

Notation of GraviTrax® marble runs

July 29, 2021

GraviTrax® is a marble run building system from Ravensburger. To start with you need at least one starter set. There is an official app to register own runs. That app focuses on graphics effects and visualizing runs while the notation proposed here aims at a compact text format and can be used even without support from a program. However a program can be useful to check the recipes for errors, output human readable instructions and more. The descriptions of such marble runs can be noted in files. A program **gravi** has been written to store many such descriptions in a database. This allows to quickly determine the number of different elements for a run, register owned construction sets, visualize marble runs, output build instructions for runs, export stored runs and more.

Elements

Each element is given a **short name** (see table). It is used in the description lines for placing the elements on the planes. The characters **3..9** are reserved for self made height tiles (see below).

_	Base Plate
^	Transparent plane
=	Small transparent plane
o	Marble
1	Height tile large
+	Height tile small
3..9	Height tiles (self made)
A	Launch pad
B	Balcony
C	Curve
D	Freefall (Drop)
E	Double Balcony
F	Flip
G	Catcher
H	Hammer
I	Tunnel Straight
J	Jumper
K	Scoop (Cascade)
L	Pillar
M	Magnetic Cannon
N	Volcano
O	Open Basket
P	Splash
Q	Looping
R	Trampoline
S	Switch
T	Tunnel Curve
U	Tunnel Switch
V	Vortex
W	3 in 1
X	Junction
Y	2 in 1 (same tile as Switch)

Z	Landing
xA	Zipline begin
xB	Bridge Tile
xD	Dipper
xF	Lift
xG	Base Tile for Inserts
xH	Spiral (Helix)
xK	Catapult
XL	Tunnel Pillar
xM	Mixer
xR	Transfer
XS	Spinner
xT	Tiptube
xV	Dispenser
xZ	Zipline end
a	Rail Bernoulli Short
b	Rail Bernoulli
c	Rail bent left turn (anti clockwise)
d	Rail bent right turn (clockwise)
e	Finish line
f	Lift Tube
g	Rail Overlong
h	Spiral Curve
l	Spiral in
j	Spiral out
l	Rail Long
m	Rail Medium
q	Rail Overlong Slow
r	Angled Base (Trampoline)
s	Rail Short
t	Tunnel Vertical
u	Drop Rail Convex

v	Drop Rail Concave
xa	Zipline Rail
xb	Unfolding Rail Unit
xi	Lift in
xj	Lift out
xl	Wall long
xm	Wall medium
xs	Wall short
xt	Flextube

Placement of elements on the board

Each tile with its up to 3 connecting rails and up to 3 marbles is described by a single line of the form

Position [*Height*][*Tile*][*Detail*][*Orientation*] [*Rail*][*Direction*] [*o*][*Color*][*Position*]

where *Tile* and *Rail* are symbols from the table above. For the meaning of the other parts in the line please see the explanations below. A sample line looks as follows

55 3Cf md

and means that at position 5,5 three large height tiles are placed with a curve tile on top. The rail connections on that curve tile with orientation f are north, north east, south and north west. There is a medium length connecting rail in direction south. Each rail needs to be described only once, i.e. in this example the rail from position 8,5 to position 5,5 is already known. The height is a string of height tile symbols. For consecutive 1 elements its sum can be written instead as long as it stays below 10.

Element positioning

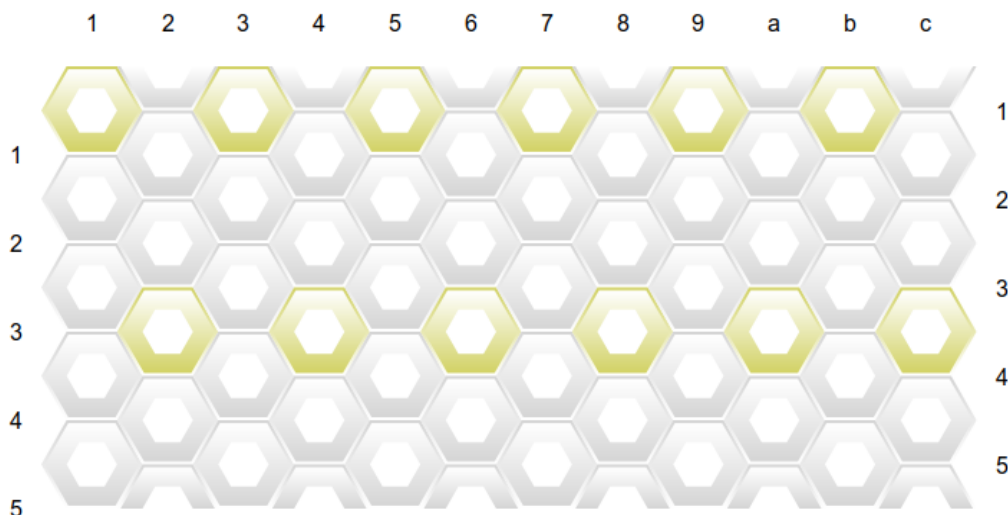
A **position** is described by **xy** with a row number *x* and a column number *y*.

The position is either a relative position on a plate <row><column> or an absolute position numbering all rows and columns from **1..9a-z**. Absolute position numbering can hence be used only for runs with less than 36 rows or columns, that is 5 plates horizontal and 6 plates vertical.

If the description starts with a base plate line (symbol **_**), relative coordinates are used, if such lines are missing then absolute coordinates in the range **1..9a..z** are used. For relative positions the **Row number** *x* runs from 1..6. The **Column number** is then in the range **1..5**. For larger marble runs a different notation with ground plane numbering has to be used,

Rails need a position and a direction for its placement. On a description line the originating position is already given, it is the position of a tile (the first information in the line). Therefore only the direction away from the tile is required. This is noted down with the letters **a..f**, where **a** means the upward direction, the subsequent ones go clockwise up to **f** (upper left).

To illustrate the absolute position numbering, a 1 x 2 board is shown in the following picture:



Ground plane

The ground plane is built of rectangular cardboard pieces (base plates with the symbol $_$) that can be put together.

We choose the **orientation** of the ground plane such, that always a **green hexagon** is in the **upper left corner** of the card pieces. This is also the orientation used in the instruction booklets. The notation gets wrong if this condition is not met!

When using relative positions we need to say on which base plate the tiles are placed. That is done with lines of the form

$_ <row> <column>$

For absolute positions such lines must not be used.

Transparent planes

Transparent planes (\wedge) (and \equiv , see below) have to be described by a line giving the center position xy on the ground plane:

$xy \wedge$

followed by lines describing what is on that plane. To better visualize when the description for a new transparent plane begins, in addition a level line can be used:

Level $<number>$

The level number is a sequence number. On stacked planes lower level planes should get smaller numbers. For all elements placed on a transparent plane the height of the elements has to be given relative to that plane, not counting from the ground (level 0). A Level 0 line can be omitted if there is only one level or if level 0 lines are coming first. While the level lines are optional it is highly recommended to keep them, as they define the level numbers. Otherwise they get assigned sequential numbers for every line with a \wedge character. If relative positions are used, the positions on the transparent planes are relative to its center, which is at position 33. The possible relative positions are shown in the picture in the light blue area.

The speed starter set introduced smaller transparent planes. They will be denoted using the symbol \equiv . For these planes the center is at position 22 and the possible relative positions are around that position. Other than that the description of small transparent planes is the same as for the larger transparent planes.



Orientation of elements

As all tiles have a hexagonal form, at most 6 different orientations are possible. Connecting elements such as rails are characterized by two positions and do not need an orientation value. The 6 possible orientations are denoted by the letters **a..f** where **a** is the orientation pointing north (up, 12 o'clock) and the other ones are clockwise labeled **b..f**. The following picture illustrates the chosen orientation convention

In the starter set booklet most of the elements are displayed. Orientation **a** is always the one where the front side of the element (south west in the drawings) is pointing north. (Exception launch pad, there the shown picture is orientation **b**)

Orientation **a** is always the one, where

1) the marble leaves the element (if a direction for marbles is given, e.g. hammer, magnetic cannon etc.). For the volcano direction **a** is the one where the marble enters the tile.

2) multiple paths end in one (e.g. switch, 2 in one, 3 in one) or

3) a rail connection is possible. This leaves an ambiguity for curves and junctions. For a junction position **a** is where rail connections are north, north east, south east and south west, for a curve where rail connections exist at the north north east, south east and south west sides. If on the launch pad not all marbles are placed, an orientation should be used where a marble is sitting.

Orientation	a	b	c	d	e	f
Curve						
2 in 1, Switch						
Junction						
Freefall (Drop), Catcher						
Balcony						
Straight Tile						
Basic Tile						

Height tiles

There exist large height (**1**) (1 unit) and small height (**+**) (0.5 units) tiles (element names in bold letters in parentheses) that can be stacked. On top of the stacked tiles marble run tiles such as curves (**C**), junctions (**X**) etc. can be placed. The heights of these tiles are described by **a sequence of height symbols** or a

number of height units under it. Only up to 9 height units may be noted. Larger numbers can be reached by concatenating more digits, which are then added. Thus +92 and 29+ both mean 11.5 height units.

It was proposed to build additional (wooden) height tiles to overcome the often limiting number of height tiles available when building more complex runs. Such tiles with a height being a multiple of one height unit can be noted as well and do have element characters **3..9**. This is only used when checking the required number of elements for a given marble run against the number of owned elements.

The Starter set vertical introduced more height elements, pillars, tunnel pillars and double balconies. It qualifies them as height elements as they can be stacked on top of each other in any order. In contrast to other height elements the notation of tunnel pillars and double balconies need an orientation letter appended.

Rails and Walls

The notation of height elements follows usually by a tile description. Afterwards the walls, rails and marbles get described. For rails the symbol name and the direction have to be entered. The starting point is the tile described on the same line and the direction is counting clockwise from **a..f** away from the starting point. Up to three rail descriptions are possible. The vertical starter set adds walls, which are also noted as rails, as they are connections between tiles. Up to 3 walls can be attached to a tile in addition. For walls a digit between the symbol name and the orientation character has to be put, if the wall is not attached to the pillar closest to the ground plane. This digit is the pillar count.

Marbles

For better documentation of the placement of marbles (element **o**) it **can** be noted where marbles are placed at the beginning of a run and which color they have. The color names silver, red, green and blue are coded in the detail field **S,R,G** and **B**. Its positions on a tile can also be specified by the orientations **a..f**. A repeat count in front of the **o** is possible for e.g. the many marbles needed for the lifter (see below).

Switches

The initial position of the moving part on a switch tile may be important for the correct working of a run. Therefore the initial switch state can be noted for switches, tunnel switches and the dipper by adding a detail field, a **+** or **-** sign (clockwise or anti clockwise turned) to the switch definition (**S**, **U** and **xD** respectively).

Example: *S+a*

Flextube

A flextube (**xt**) has a variable direction of the outgoing marble. That has to be noted as a detail before the flextube direction (the direction of the incoming marble) and after the symbol (**xt**).

Bridges

A variable even number of bridge elements (at least two) can be used on bridge tiles. The number of elements used can be inserted as a detail after the element symbol (**xB**) and before the direction (**a..f**). If no number is given, then 4 is assumed. Only with 4 elements unfolding bridges are possible.

Drop rails with open basket

A collecting basket (**O**) is usually placed under the drop rails. It is noted like a normal stone, but without any height and direction (both are defined by the rail on which it is located).

Trampoline

One or two angled bases can be placed under the trampoline tiles. These bases also have an orientation (for orientation **a** the thick side is pointing up). In order to describe the placement and positioning of the angled bases, after the **R** character (for trampoline) up to two specifications *<orientation>* can follow in the detail field.

Example: *33 1Raa*

Spiral

The same notation as for unfolding bridges applies to spirals. The number of elements (at least two, since start and run-out are always required) must be specified between the element name (**xH**) and the character describing the orientation. The orientation is defined by the outgoing path, not the start of the spiral. The number of elements used can be easily determined, it is twice the number of height units between the start and the run-out element

Lifter

In addition to the ball inlet and outlet, several tubes can be used to increase the height of the lift. The number of transparent elements (at least 2) is given after the element letter (**xF**). If no number is given, the default is 2, i.e. only inlet and outlet were used. The direction of the ball outlet has to be put before the orientation of the stone, which is defined by the ball inlet. A complete description of a lift then looks like this:

xy zxF[Number of transparent Elements]<Inlet direction><Orientation> [rails]

Mixer

The mixer is a tile that has the same symmetry as the starting tile. It is noted with **xM**. There are only two orientations that differ. The specialty is the green moving part which, depending on its position, determines where the ball rolls. If you want, you can write the direction of the outgoing ball as an additional information between **xM** and the orientation of the mixer. For the orientation a, c and e of the mixer, only directions a, c or e are possible, for b, d and f the other three directions.

Splinter

The splinter is a tile that, depending on the position of the rotating green flap, directs the ball in one or the other direction. The position of the flap therefore decides in which direction the ball runs. As with the junction tile, up to four rails can be connected at the top of the stone, so the orientation is defined the same as for the junction tile. For orientation a, the green flap points upwards slightly to the right. So you don't need any additional information for the initial position of the flap. The splinter is given the symbol **xV**, whereby the V is intended to indicate that the ball falls down, similar to the vortex.

Balconies

The vertical starter set introduces two types of balconies. Both act as height elements and get noted together with other height elements. For simple balconies the hole they are sitting in is coming first with positions **1..9a..d** (hole 1 .. 13). then comes the symbol **B** for the balcony followed by the orientation character. If the balcony is attached to a wall where several walls with the same direction and position are stacked on top of each other then the (lower) pillar number (counting from the bottom) where the wall is attached to has to be inserted as a detail before the orientation character.

Double balconies have two positions and need to get registered with these positions twice. The first position is that in a stack of height elements. The double balcony has to be noted there with its orientation together with the other height elements. Then on a separate line the second position has to be written down, followed by the symbol **E** and the orientation. If more than one double balconies exist at a position in the same

direction, its sequence number (counting from bottom) has to be entered as a detail in the line with the second position.

The elements on top of the balconies (height tiles, another tile and rails) are noted afterwards.

Notation of marble tracks

A marble track description has to start with a line

Name *<unique name of the track>*

followed by optional lines

Date *<free form date string>*

Author *<name of the creator of this description>*

Source *<where was the track originally described/found>*

Then follow element placement lines as already described above with a position, the tile placement, zero to three rail and wall descriptions and zero to three marble data. Instead on putting descriptions on separate lines the information blocks can be separated by a **semicolon**.

Lines starting with # (**comment lines**) and empty lines are ignored.

As an example a complete description file is shown below:

```
Name Track A
Date Feb 27, 2020
Author John Doe
Source Construction plan booklet
43 5Ac sc ocS; 55 3Cf md; 85 3Cf lb; 69 Ce; 6a Cf; 7a Zb
```

Notation of owned construction sets and elements

The number and kind of construction sets can also be noted in a file. It has to start with an owner line

Owner *<name>* [*****]

followed by lines describing the number and the name of owned sets and elements. If the owner line is marked with a star then that owner becomes the default owner used for the determination of required material.

With the exception of the starter set names all names can be abbreviated to at minimum 2 characters as long as they are unique, its names are as follows:

Starter Set
XXL Starter Set
Starter Set Vertical
Starter Set Speed
Building Extension
Tunnel
Trax
Bridges Extension
Extension Vertical
Lifter
Hammer
Looping

Volcano
Transfer
Flipper
Scoop
Catapult
Cable Car
Trampoline
Magnetic cannon
Jumper
Spiral
TipTube
Splinter

Dispenser
Spinner

Dipper
Flextube

Additional or missing elements can also be entered, where the number of elements can be positive or negative and the elements are named by its (one or two letter) symbol (see table above).

The material lines are of the form

```
<number> [x] <set name|element character>
```

A sample file looks then as follows:

```
Owner John Doe
 2 Starter
 1 Volcano
 2 x 4
-1 x m
```

Also here a **semicolon** can be used to separate information blocks

Software

The concept above has been implemented in a perl program. Apart from a standard perl it requires the database module DBI and DBD::SQLite, SVG and Locale::Maketext::Lexicon for the locale support. If svg files should be converted using the attached `svg_to_png` program, then `Image::LibRSVG` is required in addition. Support for other DB modules is currently not planned. During the first run a `.gravi.db sqlite3` file is created in the users home directory. When instructions for a marble run are displayed, afterwards a svg file (best viewed in a web browser) can be created.

Installation on Linux

Assuming that perl is installed, there are no special features during installation. The newest `Game-MarbleRun-xy.tar.gz` is downloaded (xy is the version number), unpacked and in the new version drawing is changed. Then the usual procedure for installing perl modules follows:

```
tar xf Game-MarbleRun-xy.tar.gz
cd Game-MarbleRun-xy
perl Makefile.PL
make
make test
make install
```

If the program should not be installed centrally, a good recipe is to use the `local::lib` perl module described e.g. at <https://metacpan.org/pod/local::lib>.

Installation on Windows

The installation under Windows requires a working Strawberry-Perl (<https://strawberryperl.com/>). The security warning must be ignored when installing Strawberry perl. When strawberry perl is installed, you open the command prompt. Then you have to install some required perl modules and enter the other commands similar to Linux. As `DBD :: SQLite` should already be included with Strawberry Perl that the `cpan` command does nothing .

```
cpan install DBD::SQLite
cpan install Locale::Maketext::Lexicon
tar xf Game-MarbleRun-xy.tar.gz
```

The last command creates a directory with the software. You need to change into that directory:


```
cd Game-MarbleRun-xy
```

and execute there the following commands:

```
perl Makefile.PL  
gmake  
gmake test  
gmake install
```

If the procedure above is too complicated or strawberry perl should not be installed, a ready to run gravi.exe can be downloaded from the web site <https://www.zeuthen.desy.de/~friebel/gravitemp/>. Please use it at your own risk.

Since the cmd program or a similar terminal program has to be used to start the program, the characters there are normally printed with the Windows encoding. As the program gravi switches to Unicode and outputs also some less common Unicode characters, some characters get displayed incorrectly. To correct this, the font used by cmd has to be changed. To do this, right-click in the upper bar of the cmd program. In the opening menu select the Font tab and choose an appropriate font, e.g. "DejaVu Select Sans Mono" or "Source Code Pro". Only these two fonts have all special characters used. This setting needs to be done only once, Windows will remember the font choice.

Program usage

On the first invocation of gravi the database gets created. You are asked if a sample demo track should be included in the database. If you accept then the build instructions for this track are printed and you are asked to enter a file name if a picture should be created. In that case a file with the extension .svg is written that can be viewed with an appropriate program (web browser or graphics program).

The first step should be the registering of tracks and maybe material. The provided tracks get registered by

```
gravi -a runs
```

provided you are in the directory containing the runs directory. To see what is possible with the gravi program you can call the program with the -h option:

```
gravi -h
```

All other activities like selecting, printing and deleting marble runs are described in the help displayed. Currently the program is available in two languages, English and German. The language can be selected with the -l option followed by the two letter language code (en or de).

© Wolfgang Friebel 2020, 2021