The SMBX1...64 *.LVL file specification

Reverse-engined by Wohlstand 02/12/2014
Updated 02/03/2020 to clarify with the source code

Level file is a TEXT file. All parameters are written sequentially with separating by CRLF new-line character (This file format requires CRLF line feed only and LF-only file will cause crash of SMBX).

Introduction

Standard parameters:

Standard size of one block	32x32 pixels
Possible on screen display height	19 blocks
Possible on screen display width	25 blocks
Height of screen	600 pixels (19 blocks without 8 pixels)
Width of screen	800 pixels (25 blocks)
Max level space size:	419998×419998 pixels (but it is allowed to come out of limits)

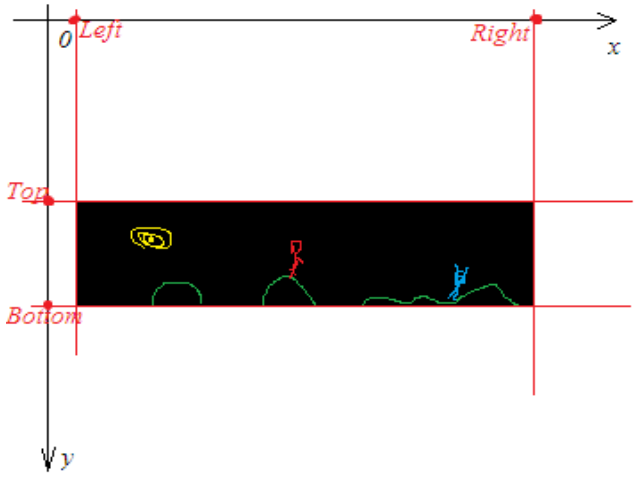
Limits of objects on one level map:

Blocks:	20000
NPCs:	5000
Background objects:	8000
Doors:	200

Level coordinate space

All elements of a level: blocks, Background Objects (BGOs), NPC's, warp points, liquid zones, playable characters start points, are in united space which dividing to 21 section (for file formats older 8 are 6 sections). Coordinate system has a pixels units.
X axis is directed from left to right.
Y axis is directed from up to down.
Minimal and maximal values of X and Y are equal to “Double” C/C++ type.

Each section is declared by the position of each side of the section.
height and width can be calculated with a formula:
$W = |L-R|$ $H = |T-B|$

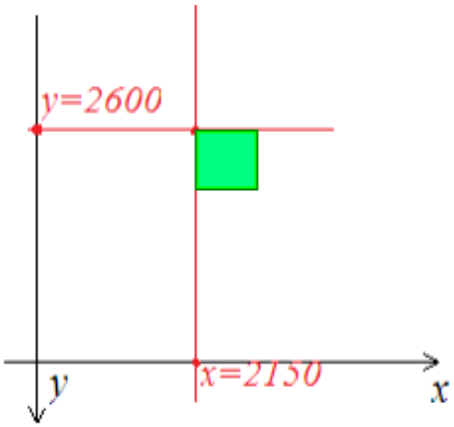


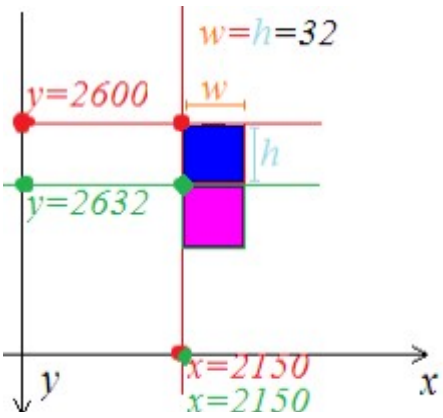
The coordinates of an object's placement is set concerning its upper left corner:
In this example, the mushroom's coordinates on the current section is: X=2150; Y=2600

and the block coordinates are: X=2150; Y=2632

Remember!

As the Y axis is turned to move an object down, it is necessary to add to the Y offset and to move up, it is necessary to subtract.





File Format Specification

Structure

- [header]
- [sections settings]
- [player's start points]
- [blocks on level]
- “next”
- [background tiles]
- “next”
- [NPS's options]
- “next”
- [Warps/Doors options]
- “next”
- [Water/Quicksand]
- “next”
- [Layers]
- “next”
- [Events]

The reference designations:

- standard parameter
- **Comment title**
- Comment description
- **loop**
- variable
- *Special option, used only under special conditions, differently is absent*
- The option isn't known yet
- File format version limit
- Data type

File format version:

The version number of the file format defines data present or absent in the file.

--File begin--

First 3th lines (Header):

Parameter	Description
64	(unsigned int) File format number (last in SMBX is 64)
0	(unsigned int) number of stars on this level (>= 17)
"Level name"	(string) Level title (>=60)

Section properties

(loop 21 times) (if <=7, loop 6 times)	
-200000	(long) **Left** size (-left/+right)
-200768	(long) **Top** size (-down/+up)
-200000	(long) **Bottom** size (full screen is -200600 -down/+up)
-199200	(long) **Right** size (full screen is -199200 +left/-right)
24	(unsigned int) Music number (24 is enable custom music file)
16291944	(long) Background color *(In old versions editors)* [0] = black, [16291944] = blue, [10520656] = dark blue-green
#FALSE#	(bool) Is Level wrap
#TRUE#	(bool) enable Off Screen exit
13	(unsigned int) Background number
#FALSE#	(bool) No turn back (>=1)
#TRUE#	(bool) Under water (>=30)
"somefile.mp3"	(string) Custom music file (>=2)
(End loop)	

Note: Format 0 is an intro.dat file of SMBX 1.0

Player start points

-199996	(long) First player Position x (+left/-right) (0 – player point isn't set)
-200600	(long) First player Position y (+up/-down) (0 – player point isn't set)
24	(unsigned int) Width of character (0 – is is first player point isn't set)
54	(unsigned int) Height of character (0 – is is first player point isn't set)
-199242	(long) Second player Position x (0 – player point isn't set)
-200068	(long) Second player Position y (0 – player point isn't set)
24	(unsigned int) Width of character (0 – is is second player point isn't set)
60	(unsigned int) Height of character (0 – is is second player point isn't set)

Blocks

Description of the blocks placed on a map:

(Loop = how many blocks are in this level)	
-241408	(long) Block position x
-180512	(long) Block position y
32	(unsigned int) Height
32	(unsigned int) Width
63	(unsigned int) Block ID
0	(unsigned int) Containing NPC number 0 – empty, 1-99 coins, or 1000+npc_id – NPC. **Note:** SMBX1 and SMBX2 formats (<18) have a special values of NPC-ID: 100 => 9, 101 => 1, 102 => 14, 103 => 34, 104 => 35 Coin number same which in newer formats.
#FALSE#	(bool) Invisible
#TRUE#	(bool) Slippery (>=61)
"Default"	(string) Layer name (>=10)
"BlockDestroy"	(string) Block destroy event name (>=14)
"hitme"	(string) Block hit event name (>=14)
"NoMore"	(string) "No more object in layer" event (>=14)
(Loop end)	

Note: Array must be sorted by x and by y;

Marker between Backgrounds and blocks:

"next"

Background objects description:

(Loop = how many backgrounds are in this level)	
-199808	(long) Position x
-200480	(long) Position y
15	(unsigned int) Background-1 number
"Default"	(string) Layer name (>=10)
(Loop end)	

Note: Array must be sorted by x and by y and grouped by special priory by ID;

Marker between NPC and Backgrounds:

"next"

NPC descriptions:

(Loop = how many NPCs are in this level)	
-199966	(long) Position x
-200480	(long) Position y
-1	(int) [-1] left, [0] random/none, [1] right
91	(unsigned int) NPC number
71	*(int) Special option:* *(>=15 for NPC76 and >=30 for NPC28)*
	Included NPC: used only if the NPC is
	Buuble**(283)**/Burred**(91)**/Lakitu**(284)**/Egg**(96)**
	CoopaTroopa algorithm: (NPC-76, 121, 122, 123, 124, 161, 176, 177) and Paragoombas: NPC-243, 244
	0 chase, 1 jump, 2 hover L/R, 3 hover U/D, >4 idle in air
	Cheep-cheep algorithm: (NPC-28, 229, 230, 232, 233, 234, 236)
	0 Swim, 1 jump, 2 projective, 3 swim L/R, 4 swim U/D
	Firebar position (NPC 260) 0-32
	Warp to section: -1-20 (Section number -1) **(NPC-288, 289)**
12	(int) *Second special option:* **(Only for NPC-91 with special=288)**
	Section number for included magic potion: -1-20
#FALSE#	(bool) Generator enable (>=3)
3	*(int) Special:* Generator direction: [1] up, [2] left, [3] down, [4] right (>=3)
2	*(int) Special:* Generator type [1] Warp, [2] Projective (>=3)
155	*(unsigned int) Special:* Generator period (sec*10) [1-600] (>=3)
""	(string) Message by this NPC talkative (>=5)
#FALSE#	(bool) Friendly NPC (>=6)
#FALSE#	(bool) Don't move NPC (>=6)
#FALSE#	(bool) Legacy Boss (>=9)
"Default"	(string) Layer name (>=10)
"Activate"	(string) Activate event (>=10)
"GOldCoin D"	(string) Death event (>=10)
"Talk"	(string) Talk event (>=10)
"NoMoreObj"	(string) No more object in layer event (>=14)
"AttachToLayer"	(string) Layer name to attach (>=63)
(Loop end)	

Marker between NPC and Doors:

"next"

Doors description:

(Loop = how many doors are in this level)	
-199824	(long) Entrance position x
-200224	(long) Entrance position y
-199824	(long) Exit position x
-200224	(long) Exit position y
3	(unsigned int) Entrance direction: [3] down, [1] up, [2] left, [4] right
3	(unsigned int) Exit direction: [1] down [3] up [4] left [2] right
1	(unsigned int) Door type: [1] pipe, [2] door, [0] instant
"file.lvl"	(string) Warp to level (>=3)
0	(unsigned int) Normal Entrance / To Warp [0-100] (>=3)
#FALSE#	(bool) Level Entrance (can not enter) (>=3)
#FALSE#	(bool) Level Exit (End of level) (>=4)
-1	(long) Wrap to X on world map (-1 is empty) (>=4)
-1	(long) Wrap to Y on world map (-1 is empty) (>=4)
5	(unsigned int) Need a stars for enter (>=7)
"Default"	(string) Layer name (>=12)
#FALSE#	(bool) <unused>, always FALSE (Called as “Hidden by layer”)(>=12)
#FALSE#	(bool) No Yoshi (>=23)
#FALSE#	(bool) Allow NPC (>=25)
#FALSE#	(bool) Locked (>=26)
(Loop end)	

Marker between Doors and water/quicksand:

"next" (>=10)

Water/Quicksand descriptions:

(>=29), else skip section

(Loop = how many water ranges are in this level)	
-159968	(long) Position X
-160096	(long) Position Y
160	(unsigned long) Width
64	(unsigned long) Height
0	(float) <unused>, always 0 (called as “Buoy”)
#TRUE#	(bool) Is Quicksand, else Water (>=62)
"Default"	(string) Layer name
(Loop end)	

Marker between water/quicksand and layers:

"next" (>=10)

Layers descriptions:

(>=10), else skip section

(Loop = how many layers are in this level)	
"Default"	(string) Layer name
#FALSE#	(bool) Is Hidden layer
(Loop end)	

Marker between layers and events:

"next" (>=10)

Events descriptions:
(>=10), else skip section

(Loop = how many events are in this level)	
"New Event"	(string) Event name
"All super-puper!!!)))))))"	(string) Show message after start event (>=11)
0	(unsigned int) Play sound number (0 is don't play sound) (>=14)
0	(unsigned int) End game type (0 – none, or 1 – Bowser Defeat)(>=18)
↓ **(loop 20 times = Show/hide/toggle layers lists)**	
"hideme1"	(string) Hide layer
"showme1"	(string) Show layer
"Toggleme1"	(string) Toggle layer (>=14)
↑ **(loop 20 times end)**	
Warning: In SMBX exist bug: if you add 21'st layer in any list, you will lose opportunity to delete layers from list. Max 21 layers	
""	(string) Empty string (must by for SMBX)
""	(string) Empty string (must by for SMBX)
""	(string) Empty string (must by for SMBX) (>=14)
↓ **(loop 21 times start (for each 21 sections)** (>=13)	
-1	(int) Set Music ([-1] don't change; [-2] default; or number of music)
-1	(int) Set Background ([-1] don't change; [-2] default; or # of back)
-1	(long) Set Position ([-1] don't change; [-2] default;
	or **LEFT** x coordinates for section=current_loop)
0	(long) **TOP** y coordinates for section=current_loop
0	(long) **BOTTOM** y coordinates for section=current_loop
0	(long) **RIGHT** x coordinates for section=current_loop
↑ **(Loop 21 times end)**	
"Trigger event"	(string) Trigger event (>=26)
1532	(unsigned int) trigger delay in deciseconds. I. e. 153,2 sec(>=26)
#FALSE#	(bool) No Smoke (>=27)
#FALSE#	(bool) Hold **ALT-JUMP** player control (>=28)
#FALSE#	(bool) Hold **ALT-RUN** player control (>=28)
#FALSE#	(bool) Hold **DOWN** player control (>=28)
#FALSE#	(bool) Hold **DROP** player control (>=28)
#FALSE#	(bool) Hold **JUMP** player control (>=28)
#FALSE#	(bool) Hold **LEFT** player control (>=28)
#FALSE#	(bool) Hold **RIGHT** player control (>=28)
#FALSE#	(bool) Hold **RUN** player control (>=28)
#FALSE#	(bool) Hold **START** player control (>=28)
#FALSE#	(bool) Hold **UP** player control (>=28)
#FALSE#	(bool) Auto start (>=32)
"MoveMe"	(string) Layer name for movement (>=32)
0	(float) Layer moving speed – horizontal (-Left/+Right) (>=32)
0	(float) Layer moving speed – vertical (+Up/-Down)(>=32)
0	(float) Move screen horizontal speed (-Left/+Right) (>=33)
0	(float) Move screen vertical speed (+Up/-Down) (>=33)
0	(int) Scroll section x, (in file value is x-1) (>=33)
(Loop end)	

--End of file--

Attachments:

Initial section left-top positions

(Initial size of each section is 800x600 pixels)

(Section Center)	Section (X and Y axis ranges)
-200000	**01** (-190000 : -219999)
-180000	**02** (-170000 : -189999)
-160000	**03** (-150000 : -189999)
-140000	**04** (-130000 : -149999)
-120000	**05** (-110000 : -129999)
-100000	**06** (-90000 : -109999)
-80000	**07** (-70000 : -89999)
-60000	**08** (-50000 : -69999)
-40000	**09** (-30000 : -49999)
-20000	**10** (-10000 : -29999)
0000	**11** (9999 : -9999)
20000	**12** (10000 : 29999)
40000	**13** (30000 : 49999)
-60000	**14** (50000 : 69999)
80000	**15** (70000 : 89999)
100000	**16** (90000 : 109999)
120000	**17** (100000 : 129999)
140000	**18** (130000 : 149999)
160000	**19** (150000 : 169999)
180000	**20** (170000 : 189999)
200000	**21** (190000 : 209999)

Background object Order Priority table

Value	BGO ID list	Comment
10	14, 75, 76, 77, 78	[background-2] Backgrounds
20	11, 12, 60, 61	SMB3 Goal zone
25	66, 158, 159, 172	[background-1] Waterfall
26	26, 65, 82, 83, 164, 165, 166, 167, 168, 169	Water
30	52, 79	Blk. dungeon block, Tree Trunk
75	2, 3, 4, 5, 6, 7, 8, 9, 10, 13, 15, 16, 17, 18, 19, 20, 21, 22, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 47, 53, 54, 55, 56, 57, 58, 59, 62, 63, 64, 67, 80, 81, 84, 85, 86, 89, 90, 91, 93, 94, 95, 96, 97, 98, 100, 101, 102, 103, 108, 109, 110, 111, 112, 113, 114, 115, 116, 117, 118, 119, 120, 121, 122, 123, 124, 125, 126, 127, 128, 132, 133, 134, 135, 136, 142, 144, 146, 147, 148, 149, 150, 151, 152, 153, 160, 161, 162, 163, 170, 171, 173, 174, 175, 176, 177, 178, 179, 180, 181, 182, 183, 184, 185, 186, 189, 190	Most of BGOs
76	129, 130, 131	Wooden fence
77	1	Small bush
80	48, 139, 140	Ghost house
90	70, 71, 72, 73, 74, 141	Rails, Rail lift buffer, Ghosthouse door
98	87, 88, 92, 104, 105, 107	Doors
99	99	Glass window fragment
125	23, 24, 25, 45, 46, 49, 50, 51, 68, 69, 106, 137, 138, 143, 145, 154, 155, 156, 157, 187, 188	Foreground-1 BGOs

How SMBX orders BGOs in a list: There's a mix of using hardcoded conditionals, but also some lookup from a table of boolean values, where this table of boolean values is only written once and not changed. Therefore in result BGOs are randomly sorting insode it's order group. Sort algorithm implementation inside SMBX is not stable, therefore every save results different order of BGO's.

PGE File Library orders BGOs by this order priorities table and by order of placement which gives a same result on every file save.

BGO Priority and sorting source code

```
'finds where the backgrounds should be put to set drawing priority
Public Function BackGroundPri(A As Integer) As Double
    'Lower Numbers get drawn first
    With Background(A)
        If .Type = 11 Or .Type = 12 Or .Type = 60 Or .Type = 61 Then
            BackGroundPri = 20
        ElseIf .Type = 65 Or .Type = 26 Or .Type = 82 Or .Type = 83 Or _
               .Type = 164 Or .Type = 165 Or .Type = 166 Or .Type = 167 Or _
               .Type = 168 Or .Type = 169 Then 'WATER
            BackGroundPri = 26
        ElseIf .Type = 168 Or .Type = 159 Or .Type = 172 Or .Type = 66 Or .Type = 158 Then
            BackGroundPri = 25 'WATER FALLS
        ElseIf .Type = 75 Or .Type = 76 Or .Type = 77 Or .Type = 78 Or .Type = 14 Then
            BackGroundPri = 10
        ElseIf .Type = 79 Or .Type = 52 Then
            BackGroundPri = 30
        ElseIf .Type = 70 Or .Type = 71 Or .Type = 72 Or _
               .Type = 73 Or .Type = 74 Or .Type = 141 Then
            BackGroundPri = 90
        ElseIf .Type = 139 Or .Type = 140 Or .Type = 48 Then
            BackGroundPri = 80
        ElseIf .Type = 65 Or .Type = 165 Then
            BackGroundPri = 150
        ElseIf Foreground(.Type) = True Then
            BackGroundPri = 125
        ElseIf .Type = 66 Then
            BackGroundPri = 50
        ElseIf .Type = 99 Then
            BackGroundPri = 99 'Always doors + 1
        ElseIf .Type = 87 Or .Type = 88 Or .Type = 92 Or .Type = 107 Or _
               .Type = 105 Or .Type = 104 Then 'Doors
            BackGroundPri = 98
        ElseIf .Type >= 129 And .Type <= 131 Then
            BackGroundPri = 76
        ElseIf .Type = 1 Then
            BackGroundPri = 77
        Else
            BackGroundPri = 75
        End If
        BackGroundPri = BackGroundPri + Background(A).Location.X / 10000000
    End With
End Function

Public Sub BackgroundSort()
    Dim A As Integer
    Dim B As Integer
    Dim tempBackground As Background
    Dim sortAgain As Boolean
    Do
        sortAgain = False
        For A = 1 To numBackground
            For B = 1 To numBackground
                If B <> A Then
                    If BackGroundPri(A) < BackGroundPri(B) And A > B Then
                        tempBackground = Background(A)
                        Background(A) = Background(B)
                        Background(B) = tempBackground
                        sortAgain = True
                    End If
                End If
            Next B
        Next A
    Loop While sortAgain = True
End Sub

Public Sub qSortBackgrounds(min As Integer, max As Integer) 'quicksort the backgrounds
    Dim medBackground As Background
    Dim medBackgroundPri
    Dim hi As Integer
    Dim lo As Integer
    Dim i As Integer
    If min >= max Then Exit Sub
    i = Int((max + min) / 2)
    medBackground = Background(i)
    medBackgroundPri = BackGroundPri(i)
    Background(i) = Background(min)
    lo = min
    hi = max
    Do
        Do While BackGroundPri(hi) >= medBackgroundPri

            hi = hi - 1
            If hi <= lo Then Exit Do
        Loop
        If hi <= lo Then
            Background(lo) = medBackground
            Exit Do
```

```
        End If
        Background(lo) = Background(hi)
        lo = lo + 1
        Do While BackGroundPri(lo) < medBackgroundPri
            lo = lo + 1
            If lo >= hi Then Exit Do
        Loop
        If lo >= hi Then
            lo = hi
            Background(hi) = medBackground
            Exit Do
        End If
        Background(hi) = Background(lo)
    Loop
    qSortBackgrounds min, lo - 1
    qSortBackgrounds lo + 1, max
End Sub
```