The PGE *.LVLX file file specification

Created by Wohlstand (17th of July, 2014; Updated on 17th of December, 2024)

This is a text file which saves a level map. All parameters are written with having own markers per every field, and there are divided by data sections which are contains only items of defined type (blocks, background objects, NPC's, etc.).

Introduction

Standart parameters:

Standart size of one block	32x32 pixels
On screen can showing height	19 blocks
On screen can showing width	25 blocks
Big height of screen	600 pixels (19 blocks without 8 pixels)
Big width of screen	800 pixels (28 blocks)
Small width of screen	512 pixels (16 blocks)
Small height of screen	480 pixels (15 blocks)

Default section positions

(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)

- The standard size of one section zone is 29999×29999 pixels
- Y is always equal to X as Section centre coordinates
- where x=0 and y=0 – is a centre of 11th section.

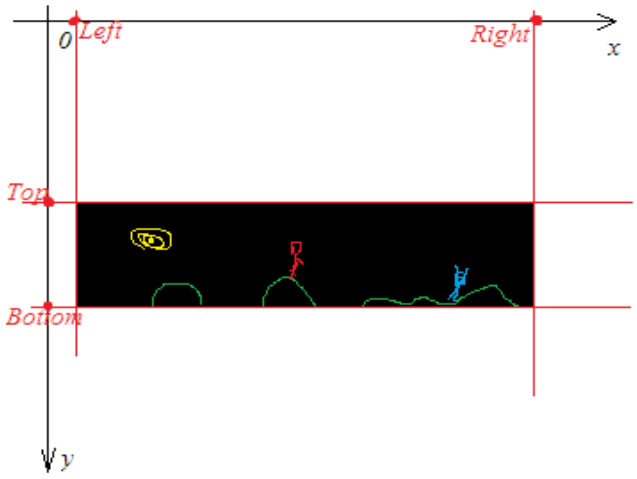
For converting from absolute coordinates to the relative of centre by one section:

$$X_{n-section} = X_{absolute} - X_{Current\ section\ center}$$

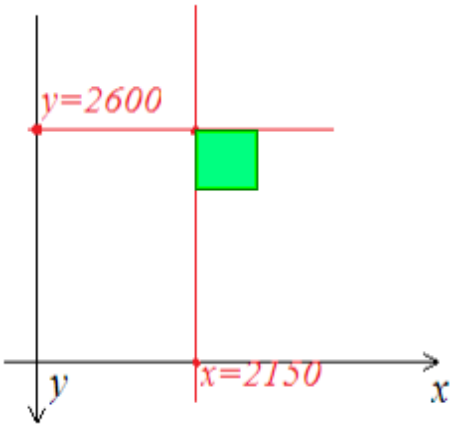
$$Y_{n-section} = Y_{absolute} - Y_{Current\ section\ center}$$

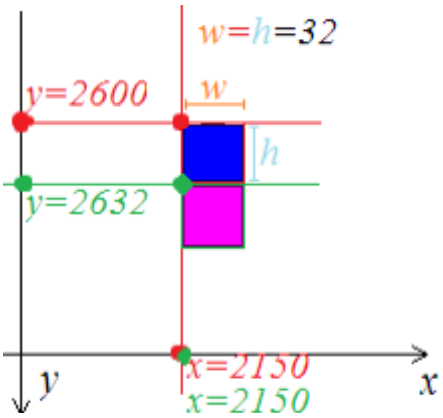
The section size and position are defined by the position of each side of the section.
height and width are calculated with a formula:

$W = |L\text{-}R|$ $H = |T\text{-}B|$



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

The reference designations:

- standard parameter
- currently not implemented or not used parameter
- **Comment title**
- Comment description
- ~~Legacy parameter~~
- **loop**
- variable
- *Special option, using only under special conditions, differently is absent*
- File format version limit
- Data type

File format version:

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

Sections

Each data type separated by markers:
started from line **DATA1** and closed with line **DATA1_END**

for example:

SECTION
SC:1;L:-32445;R:-32436;T:-43623;B:-32677;MZ:43;MF:"test.ogg";BG:34; ← some data
SECTION_END

List of available sections:

Marker	Description
HEAD	File header
SECTION	Level sections
META_BOOKMARKS	Position bookmarks
META_SYS_CRASH	**Editor-only** state meta-data to restore after emergency level rescue
STARTPOINT	Player's start points
BLOCK	Blocks present on the map
BGO	BGO present on the map
NPC	NPC present on the map
PHYSICS	Physical environment zones on the map
DOORS	Warps and doors available on the map
LAYERS	Layers
EVENTS	Action-styled events
VARIABLES	Local level variables are can be used with scripts and conditions
ARRAYS	List of array names
SCRIPTS	Local level scripts are allows you exchange level with interactions
EVENT	Sub-section of action-styled events, contains list of actions
EVENTS_CLASSIC	SMBX64-compatible classic events
CUSTOM_ITEMS_38A	Fine tune settings for in-game object types in format of SMBX-38A

Data

Each data entry have each line. Parameters separated with a semicolon “;”. All parameters must have a markers. Marker and value separated by colon “:”. Non-exist markers will be skipped. Not allowing to use parameters without markers. Data-type for each parameter defining by its internal. Data-types closed by its markers. Always possible add new marker with possible to save compatible with old versions.

Example of data entry:

```
ID:24;X:-4146;Y:23566;V:-1;ZO:0;SP:-1;L:"Default"
```

Data types syntax:

12345	unsigned int/long
-31414	signed int/long
13.45	unsigned float/double
-34.772	signed float/double
"Hello world"	string
H48656c6c6f20776f726c64	hex encoded string
[123,54,243,33]	int/long array
["test", "cat", "dog"]	string array
0/1	Boolean
011011010	Boolean array
B12FD24	Byte-encoded Boolean array

Also inside strings can be used special safe constants:

\n	New line
\”	Safe Quotes
\\	Safe slash
\;	Semicolon
\:	colon
\[	[
\]	]
\,	Comma
\%	Percent

User's variables:

@p@	Current character's name

Data markers

Legend:

Normal entry
Reserved for the future but wasn't implemented at the PGE-FL. It's possible that this value will be implemented, or will be removed as unnecessary.
Deprecated fields remained for backward compatibility with old files.
Sub-sections encoded as a serialized PGE-X data stored in a single string.

File header: (Section HEAD)

Contains only one single data row with level-wide properties.

Marker	Description
TL	(string) Level title
SZ	(unsigned int) Number of stars
DL	(string) Target level to open when player will fail (all playable characters are dead)
DE	(unsigned int) Target level warp-ID (0 – regular entrance, >=1 – warp-ID of enter)
NO	(string list) Player names override per index/id.
CPID	(string) Unique configuration package identifier string.
XTRA	(string) General-purpose level-wide extra settings (usually stored in JSON format).
MUS	(string list) Level-wide list of external music files.
GS	(bool) Use episode global switches (switch states will be saved)
PS	(int) Physics type (Internal physics configuration ID)
CC	(bool array) Controls flags (allow/deny control features for whole level)
CT	(int) Start level with character ID ([-1] – last used character)
TL	(long) Time limit in seconds (0 - disabled)
TA	(int) Time limit type (Kill player, Trigger event)

Level Section (Section SECTION):

Contains a list of sections available on the level and their settings.

Marker	Description
SC	(unsigned int) Number of section. Maximum allowed ID is 1000.
L	(long) Left side position X.
T	(long) Top side position Y.
B	(long) Bottom side position Y.
R	(long) Right side position X.
MZ	(unsigned int) Internal music ID.
MF	(string) Custom music relative file path.
ME	(string) Index of the external music entry the level header's list.
BG	(unsigned int) Internal background ID.
BF	(string) Custom background relative file path.
CS	(bool) Connect horizontal section sides (Horizontal wrap).
CSV	(bool) Connect vertical section sides (Vertical wrap).
OE	(bool) Off-screen exit flag.
SR	(bool) Right-way scroll only, camera blocked to love to left (No Turn-back).
SL	(bool) Left-way scroll only, camera blocked to move right.
SD	(bool) Up-way scroll only, camera blocked to move down.
SU	(bool) Down-way scroll only, camera blocked to move up.
UW	(bool) Set the underwater environment on entire section.
XTRA	(string) General-purpose per-section extra data string (Primarily in JSON format)
GR	(int) Gravity Value
WE	(bool) Section wind flag
WD	(int) Section wind direction
WF	(int) Wind strength value
WT	(int) Weather effects
CC	(bool array) Controls flags (allow/deny control features for this section only)

Position bookmarks (Section META_BOOKMARKS):

Contains a list of position bookmarks that allows quickly scroll to the remembered position of the editor camera.

Marker	Description
BM	(string) Bookmark title
X	(long) Position X
Y	(long) Position Y

Editor-only state recovery meta-data (Section META_SYS_CRASH):

Preservation of currently-editing file state used by editor to restore the currently-editing level file after fatal errors.

Marker	Description
UT	(bool) Untitled flag (if it's a new file that was never saved on the disk).
MD	(bool) Was file modified after last save or not?
FF	(int) Recent file format ID.
FV	(unsigned int) Version of file format (SMBX64 and SMBX-38A only).
N	(string) File name.
P	(string) Directory path where original level is placed (if not untitled).
FP	(string) Absolute path to the original level file (if not untitled).

Players start points (Section STARTPOINT):

Declares a list of points where players will start playing the level by default (if not entered by warp or resumed the game on checkpoint after they failed previous attempt).

Marker	Description
ID	(unsigned int) Player ID
X	(long) Position X
Y	(long) Position Y
D	(int) Start direction: -1 is left, 1 is right, 0 is auto depending on half of section place

Blocks (Section BLOCK):

Declares a list of solid block objects available on the level.

Marker	Description
ID	(unsigned long) Block ID
X	(long) Position X
Y	(long) Position Y
W	(unsigned int) Width
H	(unsigned int) Height
AS	(bool) Enable the auto-scale of physical size to the size of graphical frame.
CXN	(string) Custom base filename of the graphic (If empty, use default graphics).
GXX	(int) Texture offset by X (used for texture that contains multiple variants of same)
GXY	(int) Texture offset by Y
CN	(int) NPC Content (0 – empty, <0 – coins number, >0 – NPC-ID)
IV	(bool) Invisible flag
SL	(bool) Slippery flag
MA	(unsigned int) Assign a special auto-motion AI (in SMBX-38A it's wings type)
S1	(long) Special value 1 (General-purpose integer, differently used by AI)
S2	(long) Special value 2 (General-purpose integer, differently used by AI)
LR	(string) Layer
ED	(string) Event slot "Destroyed"
EH	(string) Event slot "Hit"
EE	(string) Event slot "Layer is empty"
PTR	(string) Unique item pointer (Using only for item specific events)
XTRA	(string) General-purpose per-block extra data string (Primarily in JSON format)

Background object (Section BGO):

Declares a list of visual decorations (called as "background objects") available on the level.

Marker	Description
ID	(unsigned long) Background object ID
X	(long) Position X
Y	(long) Position Y
GXX	(int) Texture offset by X (used for texture that contains multiple variants of same)
GXY	(int) Texture offset by Y
ZO	(double) Z-Offset: Sets the relative offset to the default Z-order value of BGO-ID.
ZP	(int) Z-Position: Sets one of built-in Z-values: • 2 – Foreground 2 (Over everything) • 1 – Foreground 1 (Over everything but under lava blocks) • 0 – Default (Keep default Z-value value of current BGO-ID) • -1 – Background 1 (Under everything but over resizeable blocks) • -2 – Background 2 (Under everything)
SP	(int) SMBX64 sort priority override (Used during resaving into SMBX LVL file)
LR	(string) Layer
PTR	(string) Unique item pointer (Using only for item specific events)
XTRA	(string) General-purpose per-BGO extra data string (Primarily in JSON format)

Non-Playable Characters (Section NPC):

Declares a list of interactive objects (non-playable characters, items, obstacles, etc.) available on the level.

Marker	Description
ID	(unsigned long) NPC ID
X	(long) Position X
Y	(long) Position Y
CXN	(string) Custom base filename of the graphic (If empty, use default graphics).
GXX	(int) Texture offset by X (used for texture that contains multiple variants of same)
GXY	(int) Texture offset by Y
OW	(long) Override width (0 – keep default)
OH	(long) Override height (0 – keep default)
D	(int) Direction (-1 left, +1 right, 0 random)
CN	(long) Content of NPC (used by NPC containers only). 0 – empty. Depending on NPC-AI, this might be a container that releases its content by interaction or other conditions, or an enemy or generator who replicates content continuously.
S1	(long) Special value 1 (General-purpose integer, differently used by AI)
ZP	(int) Z-Position: Sets one of built-in Z-values: • 2 – Foreground 2 (Over everything) • 1 – Foreground 1 (Over everything but under lava blocks) • 0 – Default (Keep default Z-value value of current NPC-ID) • -1 – Background 1 (Under everything but over resizeable blocks) • -2 – Background 2 (Under everything)
S2	(long) Special value 2 (General-purpose integer, differently used by AI)
GE	(bool) Generator flag
GT	(int) Generator type
GD	(int) Generator direction: • 0 – Centre/Custom (Enables feature such as custom angle, branches, range, etc). • 1 – Up. • 2 – Left. • 3 – Down. • 4 – Right.

	• 9 – Up-Left. • 10 – Left-Down. • 11 – Down-Right. • 12 – Up-Right.
GM	(unsigned int) Generator period in deciseconds value.
GA	(double) Generator custom angle in degrees (applies if direction value is 0)
GB	(unsigned int) Generator number of branches (applies if direction value is 0)
GR	(double) Generator angle range (applies if direction value is 0)
GS	(double) Generator, custom initial NPCs speed (applies if direction value is 0)
MG	(string) NPC message
FD	(bool) Friendly flag
NM	(bool) Idle flag
BS	(bool) Boss algorithms
LR	(string) Layer
LA	(string) Attach layer to this NPC: solid objects of layer will follow the NPC's move
EA	(string) Event slot "Activate"
ED	(string) Event slot "Death/Take/Destroy"
ET	(string) Event slot "Talk"
EE	(string) Event slot "Layer is empty"
EG	(string) Event slot "Grab"
EO	(string) Event slot "Touch"
EF	(string) Event slot "Next frame" (trigger event every game logic loop)
PTR	(string) Unique item pointer (Using only for item specific events)
XTRA	(string) General-purpose per-NPC extra data string (Primarily in JSON format)

Physical environment zones (Section PHYSICS):

Declares a list of physical environment zones on the level that alternates physics behaviour in their scopes.

Marker	Description
ET	(unsigned int) Environment type: • 0 – Water. • 1 – Quicksand. • 2 – Custom liquid. • 3 – Gravitational field. • 4 – Trigger an event once when detected collision with a player. • 5 – Continuously trigger an event while player touches the zone. • 6 – Trigger an event once when detected collision with a player or NPC. • 7 – Continuously trigger an event while player or NPC touches the zone. • 8 – Mouse click hot zone: trigger an event on mouse click in the zone. • 9 – Execute a script function on any collision. • 10 – Mouse click hot zone: execute a script function on mouse click in the zone. • 11 – Trigger an event on any collision. • 12 – Air chamber: cancels other physical effects, useful at underwater sections. • 13 – Trigger an event once when detected collision with an NPC. • 14 – Continuously trigger an event while NPC touches the zone. • 15 – The field that hurts any NPC that touches it. • 16 – Sub-area: can be used for making secret rooms.
X	(long) Position X
Y	(long) Position Y
W	(unsigned int) Width or Radius if Height is -1 (The shape is circle)
H	(unsigned int) Height or -1 to turn shape into the circle
FR	(double) Custom friction (Applying if "Custom Liquid" environment type is active)
AD	(double) Custom Acceleration direction (Degrees 0...360, -1 means stream is off)
AC	(double) Custom Acceleration (Applying for "Custom Liquid")
MV	(double) Maximum velocity (Applying for "Custom Liquid")
EO	(string) Touch event
HN	(bool) Enable hurtful for NPC
HP	(bool) Enable hurtful for Player
HV	(int) Hurtful strength (<0 – health up, 0 – safe, 1 – damage, 2, fast damage, 10 – fatally [as lava])
ST	(bool) Stream (for example, wind or water stream)
SD	(int) Stream direction
SS	(int) Stream strength
LR	(string) Layer
PTR	(string) Unique item pointer (Using only for item specific events)
XTRA	(string) General-purpose per-PEZ extra data string (Primarily in JSON format)

Warps and Doors (Section DOORS):

Declares a list of connected teleports that allows player to go between two points with various effects and behaviour.

Marker	Description
IX	(long) Entrance position X
IY	(long) Entrance position Y
OX	(long) Exit position X
OY	(long) Exit position Y
IL	(unsigned int) Entrance length
OL	(unsigned int) Exit length
DT	(unsigned int) Warp type: • 0 – Instant with zeroing of momentum (horizontal and vertical speeds). • 1 – Pipe. • 2 – Door. • 3 – Portal: Instant warp with remaining momentum same.
ID	(unsigned int) Entrance direction: • 1 – Up • 2 – Left • 3 – Down • 4 – Right
OD	(unsigned int) Exit direction • 1 – Down • 2 – Right • 3 – Up • 4 – Left
WX	(long) Exit to world map at X (If both X and Y are -1 – exit to world map disabled)
WY	(long) Exit to world map at Y (If both X and Y are -1 – exit to world map disabled)
LF	(string) Exit to another level file (full filename or relative path)
LI	(unsigned int) Target Warp ID. 0 – enter at the default start point.
ET	(bool) Is a level entrance (The entry will contain only exit point).
EX	(bool) Is a level exit (The only entrance point, entering which ends the level).
SL	(unsigned int) Stars/Leeks needed for entrance.
SM	(string) Custom message for needed starts/leeks if not enough collected.
SH	(bool) Don't show the number of stars in the destination level.
NV	(bool) Deny vehicles flag: vehicle will be removed until finishing the level.
AI	(bool) Allow items flag: allows player to carry item through the warp.
LC	(bool) Locked door flag: the door won't entered until using a key.
LB	(bool) Locked door, bomb required, flag until bomb will explode near the door.
HS	(bool) When enabled, the inter-level scene (with player and lives) will be skipped.
AL	(bool) Allow inter-level items: player will be able to carry item between levels.
SR	(bool) Required a special state to enter (For example, if a small/tiny state is).
STR	(bool) To enter this warp is required to stand on the ground or placed object.
TW	(bool) Two-way door flag: the warp can be entered from both sides.
TE	(unsigned int) Transition effect: • 0 – No effect or (depending on engine) a very quick fade if going between sections • 1 – Scroll (works when both points are in the same section) • 2 – Fade through black • 3 – Fade with the circular hole on the black • 4 – Horizontal flip of the screen • 5 – Vertical flip of the screen
PT	(bool) Cannon shoot exit (pipe only): player will exit with a speed value
PS	(float) Cannon shoot exit projectile speed
LR	(string) Layer

EE	(string) On-Enter event slot (Gets activated instantly when player enters the door)
EEX	(string) On-Exit event slot (Gets activated after warp/door animation finishes)
PTR	(string) Unique item pointer (Using only for item specific events)
XTRA	(string) General-purpose per-warp extra data string (Primarily in JSON format)

Layers (Section LAYERS):

Declares a list of groups of objects which are used to apply group actions by events or scripts.

Marker	Description
LR	(string) Layer title
HD	(string) Hidden flag
LC	(string) Locked flag

Variables (Section VARIABLES):

A list of level-wide integer variables that can be used in scripts and events.

Marker	Description
N	(string) Name of variable
V	(string) Value of variable (any type, encoded as ASCII/UTF-8 string)
G	(bool) Is this variable global (The variable will be visible by all levels of episode)

Arrays (Section ARRRAYS):

A list of level-wide arrays that can be used in scripts and events.

Marker	Description
N	(string) Name of array

Scripts (Section SCRIPTS):

Contains a list of level built-in script codes, may be in different languages, supported by different engines.

Marker	Description
N	(string) Name of script (Equivalent of lua function name for TeaScript)
L	(int) Code of language (0 – Lua, 1 – LunaScript/LunaDLL Autocode, 2 - TeaScript)
S	(string) Source code of script

Custom items settings (Section CUSTOM_ITEMS_38A):

Contains a list of level built-in custom item settings inherited from the 38A.

Marker	Description
T	(unsigned int) The type of object: • -1 – Unknown • 0 – Block • 1 – BGO • 2 – Effect
ID	(long) The ID of object
D	(string array) The list of "key=value" strings that should be parsed as pair of 10-base integers (if pair is invalid, file will be invalidated)

Classic Events (Section EVENTS_CLASSIC):

Declares a list of classic-style events from legacy and compatible engines. Unlike modern events, they contains everything at the single bloated structure.

Marker	Description
ET	(string) Title
MG	(string) Message text
SD	(unsigned int) Play sound ID
EG	(unsigned int) End Game algorithm
LH	(string array) hide layers
LS	(string array) show layers
LT	(string array) toggle layers
SM	(string array) Music section sets (old style, legacy parameter)
SB	(string array) Background section sets (old style, legacy parameter)
SS	(string array) Size section sets (old style, legacy parameter)
SSS	(string array) Encoded sub-entries of setting of properties for specific sections
MLA	(string array) Encoded sub-entries of moving layers list
SNPC	(string array) Encoded sub-entries of NPCs to spawn
SEF	(string array) Encoded sub-entries of effects to spawn
UV	(string array) Encoded sub-entries of variables to update
TE	(string) Trigger event
TD	(unsigned int) Trigger delay
DS	(bool) Disable smoke
TSCR	(string) Run script on triggering of this event
TAPI	(int) Trigger of internal API ID function (table of available internal functions coming soon!)
AU	(bool) Auto-start event
AUC	(string) Condition to auto-start of this event
PC	(bool array) Player's control hold keys: every bit at the array has its own meaning: • 0 – Hold UP button of player controls. • 1 – Hold DOWN button of player controls. • 2 – Hold LEFT button of player controls. • 3 – Hold RIGHT button of player controls. • 4 – Hold RUN button of player controls. • 5 – Hold JUMP button of player controls. • 6 – Hold DROP/SELECT button of player controls. • 7 – Hold START button of player controls. • 8 – Hold ALT-RUN button of player controls. • 9 – Hold ALT-JUMP button of player controls. • 10 – Enable controls hold. • 11 – Enable the full keyboard lock mode. If array is shorter than all possible values, then missing bits considered being false.
ML	(string) Movement layer
MX	(int) Movement layer speed X
MY	(int) Movement layer speed Y
AS	(int) Auto-scroll section ID
AX	(int) Auto-scroll speed X
AY	(int) Auto-scroll speed Y

Section settings sub-entries (The SSS field of Classic Events)

The PGE-X encoded sub-section that contains a per-sections list of setup change commands.

Marker	Description
ID	(int) Section ID for which need to change settings
SL	(long) Left boundary position
ST	(long) Top boundary position
SB	(long) Bottom boundary position
SR	(long) Right boundary position
SXX	(string) Expression for X position of section calculation
SYX	(string) Expression for Y position of section calculation
SWX	(string) Expression for width of section calculation
SHX	(string) Expression for height of section calculation
MI	(int) Music ID to change
MF	(string) Custom music file path to change
ME	(int) The external music index from the level header's list to change
BG	(int) Background ID to change
AS	(bool) Enable auto-scrolling for this section
AST	(int) Type of auto-scroll: • 0 – Simple (the static move of camera) • 1 – Advanced (custom camera path with speed and shape [lines or curves])
ASP	(int array) The list of 4-digits groups: each of them is an advanced auto-scrolling path stops with parameters (in a scope of same group): • 0 – The X position of stop • 1 – The Y position of stop • 2 – Type of the next path (0 - line, 1 – Bezier curve Right, 2 – Bezier curve Left) • 3 – Speed of auto-scroll in frames per 1/65 of second. Important: Array must contain multiple to 4 entries or be empty, otherwise, consider as an invalid data and fail to load the file.
AX	(float) Auto-scroll speed X
AY	(float) Auto-scroll speed Y
AXX	(float) Expression for Auto-scroll speed X calculation
AYX	(float) Expression for Auto-scroll speed Y calculation

Moving layers sub-entries (The MLA field of Classic Events)

The PGE-X encoded sub-section that contains a list of moving layers commands.

Marker	Description
LN	(string) The name of the layer to set its moving
SX	(float) The horizontal speed of the layer in ticks per 1/65 of second.
SY	(float) The vertical speed of the layer in ticks per 1/65 of second.
SXX	(string) Expression for horizontal speed of layer move calculation
SYX	(string) Expression for vertical speed of layer move calculation
MW	(unsigned int) The way of the moving layer: • 0 – Set the move speed • 1 – Immediately move the layer to X/Y coordinates

Spawn NPC sub-entries (The SNPC field of Classic Events)

The PGE-X encoded sub-section that contains a list of NPCs to spawn commands.

Marker	Description
ID	(unsigned int) The ID of NPC to spawn on event trigger.
SX	(long) The target X position of NPC spawn
SY	(long) The target Y position of NPC spawn
SXX	(string) Expression for the target spawn X position of NPC to spawn
SYX	(string) Expression for the target spawn Y position of NPC to spawn
SSX	(float) The initial horizontal speed of the NPC in ticks per 1/65 of second.
SSY	(float) The initial vertical speed of the NPC in ticks per 1/65 of second.
SSXX	(string) Expression for initial horizontal speed of NPC calculation
SSYX	(string) Expression for initial vertical speed of NPC calculation
SSS	(long) The initial “special value 1” or “content” of the NPC.

Spawn effects sub-entries (The SEF field of Classic Events)

The PGE-X encoded sub-section that contains a list of effects to spawn commands.

Marker	Description
ID	(unsigned int) The ID of the effect to spawn on event trigger.
SX	(long) The target X position of effect spawn.
SY	(long) The target Y position of effect spawn.
SXX	(string) Expression for the target spawn X position of effect to spawn.
SYX	(string) Expression for the target spawn Y position of effect to spawn.
SSX	(float) The initial horizontal speed of the effect in ticks per 1/65 of second.
SSY	(float) The initial vertical speed of the effect in ticks per 1/65 of second.
SSXX	(string) Expression for initial horizontal speed of effect calculation.
SSYX	(string) Expression for initial vertical speed of effect calculation.
GT	(bool) Spawn effect with the gravity enabled.
FP	(int) Spawn effect with the custom frame-rate.
TTL	(int) The maximum time of effect life in ticks per 1/65 of second.

Update variables sub-entries (The UV field of Classic Events)

The PGE-X encoded sub-section that contains a list of variables to update commands.

Marker	Description
N	(string) The name of variable to update
V	(string) The expression that should assign a new value to the target variable

Action-styled events entries (Section EVENT)
IMPORTANT: THIS SECTION IS A DRAFT AND HAS NO WORKING PROTOTYPES YET. THIS MAY CHANGE AT ANY TIME UNTIL THE IMPLEMENTATION WILL BE STABILIZED.

Action-styled events contains only single entry with one marker ET, what used as event title.

Actions entries example

ACT_XXX:"TEXT" ← Single-paranetric
ACT_XXX;ID:1;VL:"4564" ← Multi-parametric
DO_XXX ← Action without parameters

Actions list:

Show message: ACT_MSG

Marker	Description
-	(string) MessageText

Play sound: ACT_SND

Marker	Description
-	(unsigned int) Play sound

Hold player's keys: ACT_PCNT

Marker	Description
-	(bool array) Hold control keys

Reset player's hold keys: DO_PCNT_R

Marker	Description
-	-

Hide layers: ACT_LHIDE

Marker	Description
NS	(bool) without smoke effect
LS	(string array) layers

Show layers: ACT_LSHOW

Marker	Description
NS	(bool) without smoke effect
LS	(string array) layers

Toggle layers: ACT_LTOGGLE

Marker	Description
NS	(bool) without smoke effect
LS	(string array) layers

Toggle layers: ACT_LTOGGLE

Marker	Description
NS	(bool) without smoke effect
LS	(string array) layers

Change section music ID: ACT_SETMUS

Marker	Description
SI	(unsigned int) Section ID
MZ	(unsigned long) Music ID

Change section custom music file field: ACT_SETMUSF

Marker	Description
SI	(unsigned int) Section ID
MF	(string) Music file path

Change section background ID: ACT_SETBG

Marker	Description
SI	(unsigned int) Section ID
BG	(unsigned long) Background ID

Change section custom background file field: ACT_SETBGF

Marker	Description
SI	(unsigned int) Section ID
BF	(string) Background file path

Change section custom background file field: ACT_SRESIZE

Marker	Description
SI	(unsigned int) Section ID
L	(long) left side position X
T	(long) top side position Y
B	(long) bottom side position Y
R	(long) right side position X

Reset music ID to default: ACT_SMUS_R

Marker	Description
-	(unsigned int) Section ID

Reset background ID to default: ACT_SMUS_R

Marker	Description
-	(unsigned int) Section ID

Reset section size to default: ACT_SSIZE_R

Marker	Description
-	(unsigned int) Section ID

Reset custom music file field to default: ACT_SMUSF_R

Marker	Description
-	(unsigned int) Section ID

Reset custom background file field to default: ACT_SMUSF_R

Marker	Description
-	(unsigned int) Section ID

Set event trigger: ACT_TRIGGER

Marker	Description
TE	(string) Event title
TT	(unsigned int) Delay, d-seconds (1/10 sec)

Set layer speed motion: ACT_MOTEL

Marker	Description
LT	(string) Layer title
SX	(double) Speed X
SY	(double) Speed Y
AX	(unsigned double) Acceleration X
AY	(unsigned double) Acceleration Y

Set event trigger: ACT_AUTOSCRL

Marker	Description
SI	(string) Layer title
SX	(double) Speed X
SY	(double) Speed Y
AX	(unsigned double) Acceleration X
AY	(unsigned double) Acceleration Y

Delay next action: ACT_DELAY_NEXT

Marker	Description
-	(long) Delay time, ms

Loop event X times: ACT_LOOP

Marker	Description
-	(long) Loop times

Loop event forever: ACT_LOOP_FOREVER

Marker	Description
-	-

Abort event loop: ACT_LOOP_ABORT

Marker	Description
-	(string) Name of event which in loop

Change Item setting: ACT_CHITEM

Marker	Description
PTR	(string) Target Item pointer
IT	(unsigned int) Item type
S	(string) Setting name
V	(string) Setting value

Cut-scene mode Events

Loop event forever: ACT_CUTSCENE

Marker	Description
-	(bool) Enable cut-scene mode (Stop any AI's algorithms)

Change Item setting: ACT_CTRL_NPC

Marker	Description
PTR	(string) Target Item pointer
S	(string) Command name (Move, talk msg, jump, teleport to, etc)
V	(string) Command value

Change Item setting: ACT_CTRL_PLAYER

Marker	Description
ID	(unsigned int) Playable character ID
S	(string) Command name (Move, talk msg, jump, teleport to, etc)
V	(string) Command value

Global events and switches

Change Item setting: ACT_GLOBAL_EVENT_DO

Marker	Description
E	(string) Global event name
S	(string) Command name (Move, talk msg, jump, teleport to, etc)
V	(string) Command value

Change Item setting: ACT_GLOBAL_SWITCH

Marker	Description
SN	(string) Global switch name
SW	(bool) Switch state