

# The PGE \*.WLDX file description

Created by Wohlstand (September, 26, 2014)

This is a world map and episode settings coding file. The file is saved in a text format. All parameters have each markers and separated by sections which contains only items of defined type.

## Introduction

### Standard parameters:

Size of one tile: 32×32 pixel  
Standard view port size 668×404 pixels

## File Format Specification

### The reference designations:

- standard parameter
- **Comment title**
- Comment description
- **loop**
- **variable**
- Special option, using only under special conditions, differently is absent
- **Data type**

### 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
META_BOOKMARKS	Position bookmarks
VARIABLES	Global variables and flags list
TILES	Tiles present on the map
SCENERY	Scenery present on the map
PATHS	Paths present on the map
LEVELS	Level entrances present on the map
MUSICBOXES	Music boxes present on the map

## 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 sprcial safe constants:

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

## Data markers

### File header:

Marker	Description
TL	(string) Episode title
DC	(bool array) Disable selectable of some characters from start game page
IT	(string) Intro level which will start automatically
HB	(bool) Hub-styled episode (world map will not be used)
RL	(bool) Restart last level on player's fail
SZ	(unsigned int) Total number of stars
CD	(string) Credits of this episode

### Position bookmarks:

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

### Variables:

Marker	Description
TP	(string) Variable type (Boolean, int, float, string)
VN	(string) Variable name
VV	(*) Default value of variable [type is defined by variable type]

### Tiles:

Marker	Description
ID	(unsigned long) Tile ID
X	(long) Position x (+right/-left)
Y	(long) Position y (-up/+down)

### Scenery:

Marker	Description
ID	(unsigned long) Scenery object ID
X	(long) Position x (+right/-left)
Y	(long) Position y (-up/+down)

### Paths:

Marker	Description
ID	(unsigned long) path ID
X	(long) Position x (+right/-left)
Y	(long) Position y (-up/+down)

### Music boxes:

Marker	Description
ID	(unsigned long) World Music ID
X	(long) Position x (+right/-left)
Y	(long) Position y (-up/+down)
MF	(string) Music file name

**Levels:**

<b>Marker</b>	<b>Description</b>
ID	(unsigned long) Level icon ID
X	(long) Position x (+right/-left)
Y	(long) Position y (-up/+down)
LF	(string) Level file name
LT	(string) Level title
EI	(unsigned int) Array ID of target warp (0 – begin level from start points)
ET	(int) Level top exit type (-1 is 'any')
EL	(int) Level left exit type (-1 is 'any')
ER	(int) Level right exit type (-1 is 'any')
EB	(int) Level bottom exit type (-1 is 'any')
WX	(long) Go to world map position x (-1 is normal level point)
WY	(long) Go to world map position y
AV	(bool) Always visible
SP	(bool) Is Game start point
BP	(bool) Path background
BG	(bool) Big path background