NAME

mapper – battle grid save file format

DESCRIPTION

The version 23 file format for the `gma-mapper`(6) battle grid program is described in this document.

The file is UTF-8-encoded Unicode, of which 7-bit ASCII is a subset.

The first line MUST have the form

> `__MAPPER__:`*version*

where *version* is the map file format version this file conforms to. Following this are a number of data records of the form

> *type object*

The *object* is a JSON-formatted data structure appropriate for the *type* of value being described by this record. The line MUST begin with the *type* designation. The JSON-formatted *object* data must either be completely contained on one line, or be continued over multiple following lines, all of which SHOULD be indented by at least one space or tab. A line MUST NOT begin with the character « unless it is the start of a new record or the EOF marker (i.e., this character in the first position on a line marks the start of a new record).

The delimiter characters « and » are Unicode codepoints U+AB and U+BB respectively.

JSON field names are case-sensitive. Their values are strings unless otherwise noted.

The final line of the file MUST be the text «`__EOF__`» to indicate the end of the data.

The following data records MAY be present:

«`__META__`»
> File metadata. The JSON object may contain the following fields:
>
> `Timestamp` *(int)*
>> Modificaion time as a number of seconds from the epoch.
>
> `DateTime`
>> The modification time as a human-readable string value in any form. (Software always uses the *Timestamp* field; this is only for the convenience of human readers.)
>
> `Comment`
>> Random comments about the file.
>
> `Location`
>> The name or description of the location described by the file.
>
> `FileVersion` *(int)*
>> When a file is read, this field is populated with the input file's format version number. It is ignored if present when writing a map file.

«*element_type*»
> Defines a map element using a JSON structure identical to the JSON data defined in the mapper protocol in `gma-mapper`(6) for the LS-*element_type* command. This includes *element_type*s `ARC`, `CIRC`, `LINE`, `POLY`, `RECT`, `SAOE`, `TEXT`, and `TILE`.

«`IMG`» Declares the server ID by which a named image may be retrieved by the client (or a local path-name if the file is local) using a JSON data structure similar to that of the `AI` server command. See the example below.

«`MAP`» Declares the existence of a map file which may be retrieved from the server using a JSON data structure identical to that of the `L` server command, except that only the *File* field is used.

«`CREATURE`»
> Places a creature token on the map using a JSON structure identical to that used by the `PS` server command.

COMPATIBILITY

Programs are expected to silently ignore data fields which they don't expect, and to default any missing fields with the zero value for their data type. Thus, it is not always necessary for the file version number to increase if a change in data fields would produce a benign default behavior given these assumptions.

EXAMPLE

```
__MAPPER__:23
«__META__» {
  "Timestamp": 1650611775,
  "DateTime": "Fri Apr 22 00:16:15 PDT 2022",
  "Location": "Room 12 of the Dungeon"
}
«LINE» {
  "ID": 1,
  "Fill": "#00e400",
  "X": 425,
  "Y": 107,
  "Z": 1,
  "Points": [
    {"X": 433, "Y": 337},
    {"X": 553, "Y": 392},
    {"X": 600, "Y": 243},
    {"X": 424, "Y": 108}
  ],
  "Width": 5
}
«RECT» {
  "ID": 2,
  "Line": "black",
  "Fill": "#00e400",
  "Width": 5,
  "X": 593,
  "Y": 122,
  "Z": 2,
  "Points": [
    {"X": 807, "Y": 443}
  ]
}
«IMG» {
  "Name": "re5_e2",
  "Sizes": [
    {"File": "SJy9rpbTDElsCfklYHMvsgre5_e2@50", "Zoom": 1.0},
    {"File": "SJy9rpbTDElsCfklYHMvsgre5_e2@100", "Zoom": 2.0}
  ]
}
«__EOF__»
```

This file contains two objects:

#1 is a line in the shade of green with (R=0, G=0xe4, B=0), with line segments of width 5 going between points (425,107) − (433,337) − (553,392) − (600,243) − (424,108).

#2 is a black rectangle filled in with the same shade of green as line #1, with a line width of 5, between opposing corners (593,122) and (807,443).

Additionally, it indicates that an image called "re5_e2" may be found with server ID "SJy9rpbTDElsCfklYHMvsgre5_e2@50" at zoom factor 1.0, or with server ID "SJy9rpbTDElsCfklYHMvsgre5_e2@100" at

zoom factor 2.0.

SEE ALSO

gma-mapper(6), gma-rendersizes(6).

FILES

The exact location of cache files may vary depending on the operating system platform. For example, on macOS the cache directory is found in ˜/Library/Caches/gma-mapper. On Linux and FreeBSD it is located in ˜/.caches/gma-mapper. On Windows it is located in %LOCALAPPDATA%\gma-mapper. If no specific cache directory can be determined, ˜/.gma/mapper/cache is used by default.

˜/.gma/mapper/cache/*name*@*zoom*.gif
    The GIF-format file for image with the given *name* and scaled to the given *zoom* factor.

˜/.gma/mapper/cache/*id*.map
    The map file with server ID *id*.

HISTORY

This describes the 4.x versions of mapper.

Changes as of File Format 2

Version 2 files differ from version 1 files in that they add the JOIN, SPLINE, START, EXTENT and ARC-MODE attributes, the arc object type, and the "no-fill" (empty string) mode for the FILL attribute.

Changes as of File Format 3

This file retains compatibility with the version 2 format, with the following differences.

Each object now has a z coordinate in addition to the (x,y) coordinates of its reference point as in previous versions. The z coordinate indicates the display "stacking order" of objects on the screen. Each object is "flat" and are rendered in order from the smallest to the largest value of z so that the lowest z value is at the "bottom" with the others overlapping on top of them. These serve only to indicate the relative positioning of objects. There is no absolute meaning to any particular value for z.

Added the HIDDEN, LEVEL, and GROUP object attributes.

Changes as of File Format 4

This format is compatible with File Format 3, except that it adds the following attributes to support tile objects:

LAYER   The layer attribute may now include the value tiles which is rendered below everything else, including the grid lines.

IMAGE   This attribute appeared for the first time in this version.

SIZE   This attribute appeared for the first time in this version.

Programs which read newer version files should deal correctly with older files which do not contain the newer elements.

Changes as of File Format 5

This introduced images, creatures, and spell areas of effect.

Changes as of File Format 6

The image definition syntax was extended to allow externally downloaded image files.

Changes as of File Format 7

Additional creature attributes were added, such as NOTE and SKIN.

Changes as of File Format 8

Added text objects. This includes the addition of the ANCHOR, FONT, and TEXT attributes.

Changes as of File Format 9

Added ARROW and DASH attributes. For backward compatibility with format 8, if either of these are missing from the file, defaults are assumed.

Changes as of File Format 10

Added `ELEV` and `MOVEMODE` attributes.

Changes as of File Format 11

Added `HEALTH` attribute.

Changes as of File Format 12

Added `F` record type.

Changes as of File Format 13

Added the optional *blur* element of the `HEALTH` attribute. Removed the `SELECTED` attribute. This is a transient state used during the mapper program's operation and has no point being saved with the object. This attribute was moved to the private name `_SELECTED` instead.

Changes as of File Format 14

Added the `STATUSLIST` attribute.

Changes as of File Format 15

Added the `SKINSIZE` attribute.

Changes as of File Format 16

Added the `LOCKED` attribute.

Changes as of File Format 17

`SIZE` only applies to creature tokens. The version 16 document incorrectly specified that map elements (notably tiles) also had a `SIZE` attribute but this was never implemented. Instead, they had unofficial attributes `_BBHEIGHT` and `_BBWIDTH` which have now been promoted to official status as `BBHEIGHT` and `BBWIDTH`.

File Formats 18 and 19

These are reserved for use in case map format changes are needed before moving to map version 4.x.

Changes as of File Format 20

This version introduced the JSON-based file format, replacing the older key/value line sequence.

Changes as of File Format 21

The fields for creatures have been changed as documented in `gma-mapper-protocol`(7).

Changes as of File Format 22

Animation support introduced in protocol 407 alters the format of the `IMG` records in this file correspondingly.

AUTHORS

Steve Willoughby / steve@madscience.zone; Elevation and movement modes added by John Mechalas.

BUGS

The `LEVEL`, `GROUP`, and `LAYER` attributes are not actually implemented yet in the mapper program. They're documented here and will be accepted by the mapper but none of their behavior documented above will actually happen.

The `token` object type is deprecated and clients are free to be unaware of its existence.

The current implementation of file reading code in GMA will accept input more permissively than this spec requires, strictly speaking (e.g., an otherwise valid JSON object where the final brace is not by itself on a line), but this behavior should not be construed as a requirement for other code. File readers and writers should follow this specification as the authoritative standard.

Changes as of File Format 23

Map elements may now have a `Stipple` attribute.

COPYRIGHT