
GameSave.Info Documentation

Release 2.0.2

Matthew Barbour

February 07, 2014

1	API Reference	3
2	XML Format Reference	5
2.1	Introduction	5
2.2	Creating the XML file	6
2.3	<programs>	6
2.4	<expansion>	7
2.5	<game>	8
2.6	<mod>	9
2.7	<system>	9
2.8	<title>	9
2.9	<version>	9
3	Indices and tables	13

Contents:

API Reference

XML Format Reference

Now hold up! You don't have to learn the XML format to get games added to GameSave.Info! Just use the Analyzer program included with [MASGAU](#) to send me a report on a game (it even has a button to automatically e-mail it for you!), and I'll add it for you! But, if you really want to understand how GameSave.Info and MASGAU sources their game data, feel free to read on!

2.1 Introduction

GameSave.Info uses XML files checked into Github to describe where each game keeps its settings and saves. As you may or may not know, an XML file is little more than a specially typed text file, and can be created in programs as simple as notepad. All you do is fire up your favorite text editor and type the correct lines. Here's an excerpt from the d.xml file:

```
<game name="DeusEx" added="1955-11-05T00:00:00" updated="1955-11-05T00:00:00">
  <title>Deus Ex</title>
  <version os="PS2" region="USA">
    <title>Deus Ex: The Conspiracy</title>
    <ps_code prefix="SLUS" suffix="20111"/>
    <contributor>GameSave.Info</contributor>
  </version>
  <version os="Windows">
    <locations>
      <path ev="installlocation" path="DeusEx"/>
      <path ev="installlocation" path="GOG.com\Deus Ex"/>
      <path ev="steamcommon" path="deus ex"/>
      <registry root="local_machine" key="SOFTWARE\GOG.com\GOGDEUSX" value="PATH"/>
      <registry root="local_machine" key="SOFTWARE\Unreal Technology\Installed Apps\Deus Ex" value="P">
      <shortcut ev="startmenu" path="Programs\Deus Ex\Play Deus Ex.lnk" detract="System"/>
      <shortcut ev="startmenu" path="Programs\GOG.com\Deus Ex GOTY\Deus Ex GOTY.lnk" detract="System">
    </locations>
    <files>
      <include path="Save"/>
    </files>
    <files type="Settings">
      <include path="System" filename="*.ini"/>
    </files>
    <linkable path="Save"/>
    <identifier path="Save"/>
    <contributor>GameSave.Info</contributor>
  </version>
```

```
<comment>The best game EVER!</comment>
</game>
```

This looks more intimidating than it is. Each tag is explained on its own page, but before you read those let's define some terms:

```
<game name="DeusEx">
```

Element or Tag That little bit of code is an element. Elements are surrounded by < and >. Some elements have an opening and closing, like <game></game> where </game> is the closing element. Anything between the opening and closing element are considered “children” of that element.

Attribute The word “name” in the above is an attribute. It's like a property of an element.

NOTE: IN PATHS AND FILENAMES LEAVE OFF ALL LEADING AND TRAILING SLASHES (\ AND /)

ANOTHER NOTE: There is a [schema file on GitHub](#). If you know what that means, use it.

2.2 Creating the XML file

For the most part, you don't need to create an XML file from scratch. Just check out the data files from the repo and edit the ones already there. All GameSave.Info files must start with a standard XML declaration:

```
<?xml version="1.0" encoding="UTF-8"?>
```

The encoding attribute is important. Due to needing to support many non-ASCII characters, GameSave.Info XML files must all be encoded in UTF-8. With that added, you next have to add the root *programs element*.

2.3 <programs>

2.3.1 Example

```
<programs majorVersion="2" minorVersion="0" revision="2" updated="2013-06-15T13:45:04" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  ...
</programs>
```

2.3.2 Description

The root element for the GameSave.Info XML format is the “programs” element. There can only be one “programs” element per XML file. The name “programs” was chosen because an individual file can contain data on not just games, but also mods and system tools.

2.3.3 Attributes

majorVersion (integer) (required) This defines the major version for the XML file format. GameSave.Info and MASGAU both make use of this to determine whether they are compatible with a particular format of an XML file, and whether it is possible for it to automatically update the file to a newer format. Major version changes are defined as changes that drastically alter the entire format of the file.

minorVersion (integer) (required) This is used in the same manner as majorVersion. Minor version changes are defined as changes that do not affect the file as a whole, but still can impact compatibility with older versions of GameSave.Info or MASGAU.

revision (integer) (required) This is used in the same manner as minorVersion. Revisions are defined as changes that fo not affect the file or compatability, but still require a change to the schema file that defines the format.

updated (timestamp) (required) This is used by MASGAU's auto-update system to keep track of when a file was last updated.

xmlns:xsi (string) (required) This lets the XML parser know that the file uses an XML Schema.

xsi:noNamespaceSchemaLocation (string) (required) This tells the XML parser what the filename of the XML Schema is.

2.3.4 Child Elements

<game> (optional) Defines a game.

<mod> (optional) Defines a mod.

<expansion> (optional) Defines an expansion.

<system> (optional) Defines a system application.

NOTE: THE PROGRAMS ELEMENT REQUIRES AT LEAST ONE CHILD. CHILDREN CAN BE OF ANY TYPE LISTED HERE, AND IN ANY ORDER.

2.4 <expansion>

2.4.1 Example

```
<expansion name="MechWarrior4BlackKnight" for="MechWarrior4Vengeance">
  ...
</expansion>
```

2.4.2 Description

Use this if the entry is for an expansion pack, add-on or DLC for another game. In this example, Mechwarrior 4: Black Knight is an expansion for MechWarrior 4: Vengeance. The "for" attribute is required for an expansion, and MUST reference another game in the XML file. "Stand-alone expansions" do NOT get to be marked as an expansion. The term is an oxymoron, and makes no sense.

2.4.3 Attributes

for Designates the name of the game that the expansion is for. This allows GameSave.Info to establish links between expansions and the base game.

2.4.4 Child Elements

2.4.5 Parent Element

programs Root element that contains all program-type elements.

2.5 <game>

2.5.1 Example

```
<game name="DeusEx">  
  ...  
</game>
```

2.5.2 Description

Each game is defined within a program element. There are four types of program elements:

1. <game>
2. <expansion>
3. <mod>
4. <system>

The main purpose of this tag is to provide a unique internal name for the game. This unique name must follow these rules:

1. No spaces, and no symbols.
2. Use [CamelCase](#) for legibility.
3. Always use numbers instead of [roman numerals](#) (or other representations of numbers), for sorting purposes.
4. All versions of a game go under the same game tag. For example, the Deus Ex game tag contains a version for both Windows and PlayStation 2.

There are actually several variations on this tag, and you should try to use the one appropriate for your entry:

`<mod name="NamelessMod" for="DeusEx">` Use this if the entry is for an MOD for another game. In this example, The Nameless Mod is a MOD for Deus Ex. The “for” attribute is required for a MOD, and **MUST** reference another game in the XML file.

`<system name="GamesForWindows">` Use this when describing system data.

There is a completely optional “follows” attribute that can be added to any of these variations:

`<game name="DeusExInvisibleWar" follows="DeusEx">` It basically just indicates that the entry is somehow a follow-up (or sequel) to the indicated other entry. It’s not parsed or used anywhere yet, but one day maybe.

If a game is marked as deprecated, it means that the i.

Obviously your closing tag should match your opening tag. Other than this, the contained tags are all the same.

2.5.3 Attributes

2.5.4 Child Elements

2.5.5 Parent Element

programs Root element that contains all program-type elements.

2.6 <mod>

2.7 <system>

2.8 <title>

<title>Deus Ex</title>

Between the two title tags you just type up the name of the game. This should be the name that was first attached to a game when it was release, other names would be delegated to version titles, which we will talk about later. Try to include the entire name, no reason to skimp on length. It might be tempting to shorten Penny Arcade Adventures: On The Rain Slick Precipice Of Darkness Chapter One to Penny Arcade Adventures 1, but resist it.

2.9 <version>

<version os="Windows">

The version tag is used to specify the versions of the game the contained saves are compatible with. In this example, it's compatible with Windows. This does NOT mean this save will only work on Windows, only that it's only for the Windows version of the game. This save would also be compatible with Linux if you have Deus Ex installed under WINE. The reason it is organized like this is because GameSave.Info also doubles as the data source for the game save backup program MASGAU.

There are 5 attributes that allow us to describe a unique game save version:

os The operating system the save is compatible with. Possible values: #. Android #. DOS #. Linux #. OSX #. PS1 #. PS2 #. PS3 #. PSP #. Windows #. WindowsVista #. WindowsXP

platform The technology platform the save is compatible with. Possible values: #. Flash #. RenPy #. ScummVM #. SteamCloud #. UbisoftSaveStorage

region The region of the world the save is for. Possible values include any 3-letter ISO 3166-1 country code, or any of these two-letter continent codes: #. AF - Africa #. AS - Asia #. EU - Europe #. NA - North America #. SA - South America #. OC - Oceania (Antarctica has its own country code, so I won't support its continent code)

media The delivery medium of the game version that the save is compatible with. Possible values: #. CD #. Download #. Floppy #. GoG #. Steam

release The release of the game the save is compatible with. This is freeform, can be anything at all. Some examples: #. CollectorsEdition #. TitaniumEdition #. Gold #. HD #. Remastered #. GOTY

If a save is compatible with more than one thing in any of these categories, just don't specify the attribute. Try to keep the version specification as general as possible, while simultaneously making sure that a save would not accidentally get labelled as belonging to the wrong version of the game.

<version os="Windows" media="CD" release="TitaniumEdition" region="USA"> This example states that the contained saves are only compatible with the Windows version of the Titanium Edition of the game that was released on CD in the USA. This example is fake, I have yet to encounter saves that had such specific requirements.

<version media="Steam"> An important distinction should be made between a version for Steam and a version specifically for Steam Cloud data. The above is for the former, and the below for the latter.

<version platform="SteamCloud"> You could also specify media="Steam" on this, and it would be accurate but since SteamCloud automatically implies Steam, it's not necessary.

My policy right now on DOS games is to label it as DOS if the save produced is only compatible with the DOS version of the game. If there exists a Windows version of a DOS game, and the saves are compatible with both, then both of the games' information would be combined into one Windows profile, such as with Master Of Orion 2 or Descent II.

If you omit these attributes, then it is saying that the saves described are compatible with all versions of the game. This is pretty rare, but these games do exist. One example is fs2_open:

```
<game name="fs2_open"> <title>fs2_open</title> <version>

  <locations> <path ev="installlocation" path="fs2_open"/> <parent name="FreeSpace2"
    os="Windows"/>
</locations> <files>

  <include path="dataplayers"/>

</files> <identifier filename="fs2_open*" /> <comment>Doesn't have a default in-
  stall folder, so might require an Alt. Install Path.</comment> <contribu-
  tor>GameSave.Info</contributor>

</version>
</game>
```

If a game's saves were to work across just Linux and Windows, I would also not add a platform attribute, even if there was a Mac version with incompatible saves. By adding an additional Mac-specific version we would be declaring such an incompatibility.

You can specify more than one version of a game within the same game tag:

```
<game name="MechWarrior2">

  <title>MechWarrior 2: 31st Century Combat</title> <version os="Windows">

    <locations> <path ev="installlocation" path="ActivisionBattlePackMW2"/>
      <shortcut ev="startmenu" path="ProgramsBattlePackMechWarrior 2Mech-
        Warrior 2 Uninstall.lnk"/>
    </locations> <files type="Mechs">

      <include path="mek"/>
    </files> <files>

      <include filename="userstar.bwd"/>
    </files> <files type="Settings">

      <include filename="MW2PRM.CFG"/> <include file-
        name="MW2REG.CFG"/>
    </files> <contributor>GameSave.Info</contributor>

  </version> <version os="Windows" release="TitaniumEdition">

    <title>MechWarrior 2: 31st Century Combat: Titanium Edition</title> <loca-
      tions>

      <path ev="installlocation" path="ActivisionTitaniumMechwarrior2"/>
      <path ev="altsavepaths" path="MechVMgamesmw2-
        31stcc-tt"/> <registry root="local_machine"
        key="SOFTWAREActivisionActivenetApplications1020.2.1"
        value="Cwd"/> <shortcut ev="startmenu"
        path="ProgramsTitaniumMechwarrior2Play MechWarrior2.lnk" de-
        tract="splash"/>
```

```
</locations> <files type="Mechs">
  <include path="mek"/>
</files> <files>
  <include filename="userstar.bwd"/>
</files> <files type="Settings">
  <include filename="MW2PRM.CFG"/>      <include file-
  name="MW2REG.CFG"/>
</files> <contributor>GameSave.Info</contributor>
</version>
</game>
```

As you can see we only specify a version title when that version has a title different than the main one specified under the game tag.

`<version os="Windows" virtualstore="ignore" detect="required">` There are two additional attributes demonstrated here:

`virtualstore` - Specified if the game ignores VirtualStore in Windows Vista and later. Can be set to "ignore" or "use". Default is use. `detect` - Specifies whether the game's save location cannot be predicted without an existing save location. Can be either "required" or "optional". Default is "optional". `<version deprecated="true">` If a version is marked as deprecated, it means that the information provided is no longer considered correct. It's kept only for posterity and backwards-compatibility.

Indices and tables

- *genindex*
- *modindex*
- *search*