

Version Compatibility

Table of Contents

- [1 Overview](#)
- [2 Introduction](#)
- [3 Tag your plugin](#)
 - [3.1 Attribute Description and Usage](#)
 - [3.2 Subsystem Hierarchy](#)
 - [3.3 Notes](#)
- [4 Plugin Installer](#)
- [5 MediaPortal and TV Server SubsystemVersions](#)
- [6 Changelog](#)

Overview

How to implement version compatibility in your plugin

Introduction

As of v 1.2.0 MediaPortal introduced a [Version Compatibility](#) system to ensure plugins are compatible with a user's installed version of MediaPortal.

There are two main cases for version compatibility:

1. Plugin designed for a version of MediaPortal **older** than the one the user has installed.
2. Plugin designed for a version of MediaPortal **newer** than the one the user has installed.

Thus two new attributes have been added to allow you to tag your plugin for version compatibility in each of these cases.

Tag your plugin

Plugins **must** tag their plugin using the two new attributes:

- **CompatibleVersion** - the version of MediaPortal the plugin was built against
- **UsesSubsystem** - which MediaPortal subsystems the plugin uses

Both attributes can be used on the Assembly level or on the Class level.

Warning: Any plugin that is not tagged using the above attributes will be deemed incompatible with MediaPortal.

Attribute Description and Usage

In order to tag plugins with version and compatibility information, the following attributes have been defined:

CompatibleVersion(*designedForVersion*[, *minRequiredVersion*]) This attribute is used to specify against which version of MediaPortal the plugin was built. It may also optionally specify the minimum MediaPortal version required so that it can work properly. Versions are specified as strings in the form "x.y.z.w" (e.g. "1.1.7.0" is 1.2.0 Beta). If *minRequiredVersion* is omitted it is assumed equal to *designedForVersion*. Any version of MediaPortal earlier than *minRequiredVersion* will consider the plugin incompatible and will not load it. If any subsystems used by the plugin have breaking changes in any version later than *designedForVersion*, the plugin will be considered incompatible and will not be loaded.

The attribute can be applied to either or both the assemblies or classes. When applied to an assembly it will affect all contained classes that do not have the same attribute applied upon them.

The attribute can be applied only once on each assembly and/or class. **UsesSubsystem**(*subsystem*[, *used*]) This attribute is used to specify which subsystems the plugin is using. This information combined with the **CompatibleVersion** attribute information allows the Compatibility Manager to determine if a plugin is still compatible with a version of MediaPortal released after the release of the plugin. Specify only one subsystem (as a string) in the *subsystem* parameter. The optional parameter *used* (default is true) can be used to specify if the subsystem is used or not. This is especially useful if you have marked an assembly as using a specific subsystem but you want to exclude some classes.

The attribute can be applied to either or both the assemblies or classes. When applied to an assembly it will effectively be applied to all contained classes. If some of the contained classes are also tagged with the same subsystem(s) the attribute applied to the classes takes precedence.

The attribute can be applied as many times as needed, once for each used subsystem, to tag each assembly and/or class.

Note: Valid subsystems are defined in [MediaPortal-1](#) / [Common-MP-TVE3](#) / [Common.Utils](#) / [CompatibilityInfo.cs](#)

Basically any plugin will be compatible as long as:

1. **minRequiredVersion** is earlier than the current version of MediaPortal
2. no subsystems used by the plugin have breaking changes in any version of MediaPortal later than the **designedForVersion**.

The most common case for incompatibility is when there are required skin changes in MediaPortal, since most plugins use the MP.SkinEngine subsystem.

Subsystem Hierarchy

To unambiguously describe the functionality used, we have come up with a hierarchy of subsystems. The full set of MediaPortal functionality that plugin developers can use, has initially been categorized into a number of top-level subsystems. Each subsystem can consist of a number of lower level subsystems. The functionality described by a higher level subsystem is always a superset of the functionalities of the lower level subsystems that comprise it.

Notes

To optimize load speed and maintain plugin integrity, each assembly is first checked for compatibility based on the assembly level attributes. If during that check it is deemed incompatible, none of the contained plugins will be loaded, even if individual plugins inside the assembly may be compatible.

Plugin Installer

The new MPMaker will **only** pick up the attributes you set on your assembly. So if you want the compatibility info displayed by the Plugin Installer to correctly reflect your plugin's settings, you have to set attributes as shown in the example below:

1. Reference the new *Common.Utils.dll* from MediaPortal install dir.
2. The easiest way is to add these lines to your *AssemblyInfo.cs*:

```
using MediaPortal.Common.Utils;
[assembly: CompatibleVersion("1.1.6.27644")]
[assembly: UsesSubsystem("MP.SkinEngine")]
[assembly: UsesSubsystem("MP.Config")]
```

MediaPortal and TV Server SubsystemVersions

The following list contains all the SubsystemVersions currently available for MediaPortal and TV Server:

```
// MediaPortal subsystems' breaking versions
```

```
[assembly: SubsystemVersion("MP", "1.1.6.27644")]
```

```
[assembly: SubsystemVersion("MP.SkinEngine", "1.3.100.0")] // MP 1.4.0 Pre Release
```

```
[assembly: SubsystemVersion("MP.SkinEngine.Core", "1.1.6.27644")]
```

```
[assembly: SubsystemVersion("MP.SkinEngine.Controls", "1.3.100.0")]
```

```
[assembly: SubsystemVersion("MP.SkinEngine.Dialogs", "1.1.6.27644")]
```

```
[assembly: SubsystemVersion("MP.Input", "1.1.6.27644")]
```

```
[assembly: SubsystemVersion("MP.Input.Mapping", "1.1.6.27644")]
```

```
[assembly: SubsystemVersion("MP.Input.Keyboard", "1.1.6.27644")]
```

```
[assembly: SubsystemVersion("MP.Input.Mouse", "1.1.6.27644")]
```

```
[assembly: SubsystemVersion("MP.Players", "1.1.6.27644")]
```

```
[assembly: SubsystemVersion("MP.Players.DVD", "1.1.6.27644")]
```

```
[assembly: SubsystemVersion("MP.Players.Video", "1.1.6.27644")]
```

```
[assembly: SubsystemVersion("MP.Players.TV", "1.1.6.27644")]
```

```
[assembly: SubsystemVersion("MP.Players.Music", "1.3.100.0")]
```

```
[assembly: SubsystemVersion("MP.DB", "1.1.6.27644")]
```

```
[assembly: SubsystemVersion("MP.DB.Music", "1.5.100.0")]
```

```
[assembly: SubsystemVersion("MP.DB.Videos", "1.2.100.0")] // MP 1.3.0 Alpha 1
```

```
[assembly: SubsystemVersion("MP.DB.Pictures", "1.1.6.27644")]
```

```
[assembly: SubsystemVersion("MP.Filters", "1.1.6.27644")]
```

```
[assembly: SubsystemVersion("MP.Filters.TsWriter", "1.1.6.27644")]
```

```
[assembly: SubsystemVersion("MP.Filters.MPFileWriter", "1.1.6.27644")]
```

```
[assembly: SubsystemVersion("MP.Filters.StreamingServer", "1.1.6.27644")]
```

```
[assembly: SubsystemVersion("MP.Filters.TsReader", "1.1.6.27644")]
```

```
[assembly: SubsystemVersion("MP.Filters.DVBSubs", "1.1.6.27644")]
```

```
[assembly: SubsystemVersion("MP.Filters.AudioSwitcher", "1.1.6.27644")]
```

```
[assembly: SubsystemVersion("MP.Filters.IPTV", "1.1.6.27644")]
```

```
[assembly: SubsystemVersion("MP.Externals", "1.1.6.27644")]
```

```
[assembly: SubsystemVersion("MP.Externals.MediaInfo", "1.1.6.27644")]
```

```
[assembly: SubsystemVersion("MP.Externals.SQLite", "1.1.6.27644")]
```

```
[assembly: SubsystemVersion("MP.Externals.Gentle", "1.3.100.0")] // MP 1.4.0 Pre Release
```

```
[assembly: SubsystemVersion("MP.Externals.Log4Net", "1.3.100.0")] // MP 1.4.0 Pre Release
```

```
[assembly: SubsystemVersion("MP.Externals.BASS", "1.7.0.0")]
```

```
[assembly: SubsystemVersion("MP.Externals.HTMLAgilityPack", "1.5.100.0")] // Added after 1.6 pre-release
```

```
[assembly: SubsystemVersion("MP.Config", "1.1.6.27644")]
```

```
[assembly: SubsystemVersion("MP.Config.DefaultSections", "1.1.6.27644")]
```

```
[assembly: SubsystemVersion("MP.Plugins", "1.1.6.27644")] // reserved for plugin defined subsystems
```

```
[assembly: SubsystemVersion("MP.Plugins.Music", "1.6.100.0")]
```

```
[assembly: SubsystemVersion("MP.Plugins.Videos", "1.6.100.0")]
```

```
[assembly: SubsystemVersion("MP.Plugins.Pictures", "1.6.100.0")]
```

```
[assembly: SubsystemVersion("MP.Plugins.PowerScheduler", "1.1.6.27644")]
```

```
[assembly: SubsystemVersion("MP.Plugins.TV", "1.1.6.27644")] // This refers to TVPlugin
```

```
//[assembly: SubsystemVersion("MP.Plugins.3rdParty", "1.1.6.27644")] // reserved for 3rd party plugin
```

```
// TV Server subsystems' breaking versions
```

```
[assembly: SubsystemVersion("TVE", "1.1.6.27644")]
```

```
[assembly: SubsystemVersion("TVE.DB", "1.1.6.27644")]
```

```
[assembly: SubsystemVersion("TVE.Controller", "1.1.6.27644")]
```

```
[assembly: SubsystemVersion("TVE.Scheduler", "1.1.6.27644")]
```

```
[assembly: SubsystemVersion("TVE.Config", "1.1.6.27644")]
```

```
[assembly: SubsystemVersion("TVE.Config.Controls", "1.1.6.27644")]
```

```
[assembly: SubsystemVersion("TVE.Plugins", "1.1.6.27644")]
```

```
[assembly: SubsystemVersion("TVE.Plugins.PowerScheduler", "1.1.6.27644")]
```

```
[assembly: SubsystemVersion("TVE.Plugins.XmlTV", "1.1.6.27644")]
```

```
[assembly: SubsystemVersion("TVE.Plugins.WebEPG", "1.1.6.27644")]
```

```
[assembly: SubsystemVersion("TVE.Plugins.TvMoview", "1.1.6.27644")]
```

```
[assembly: SubsystemVersion("TVE.Plugins.ServerBlaster", "1.1.6.27644")]
```

Changelog

Change	Date	Version
Version Compatibility	2010/12/24	1.1.0 to 1.2.0
WindowPluginBase Class	2012/06/24	1.2.0 to 1.3.0
Compatibility Subsystems	2013/05/01	1.3.0 to 1.4.0

HTML Agility Pack	2013/10/22	1.5.0 to 1.6.0
LastFM/AudioScrobbler	2013/11/07	1.5.0 to 1.6.0
IMDB class changes	2012/07/05	1.2.0 to 1.3.0
VideoDatabase interface changes	2012/07/05	1.2.0 to 1.3.0