

PublicWare Music Archive Plugin Documentation

Making Music Player Plugins for Music Archive 1.0 to 1.10

Version 1.0. Released: April 14, 2003

Author: Anders M. Hansen, PublicWare

1. Intro

This document contains a description of functions, procedures and types that is used by a Music Archive plugin to handle and send information from and to Music Archive. The information should be sufficient to make it possible to create your own plugins, and there by connect Music Archive with other music players than those connected trough the standard PublicWare Music Archive plugins.

A good help to better understand this document and the functions it descried. Then see the Music Archive General Plugin Design package, it contains working code that can led to a better understanding of what this document descries.

The package can be downloaded from here:

<http://www.PublicWare.dk/EN/Development/Plugin/Sample>

2. Legal notes:

This document is delivered as is, and there is no guarantee of any sort. All use of this document or parts of the document is one the readers own responsibility.

All rights involving this document are owned by Anders M. Hansen and PublicWare. The document may freely be used, as long as it is in agreement with these legal notes. Copying and distribution of the document is allowed, as long as it is in agreement with these legal notes. The document **MUST** be copied and/or distributed in it's original form as first delivered from PublicWare and without changes.

The document may not directly or indirectly be used commercially.

The document may not be sold, rent out or used to other business purposes.

If desired to use the document commercially or in another way that is not in agreement with these notes, then special agreement can be made with Anders M. Hansen and PublicWare.

The document may not be changed in any way.

Comments or suggestions to the document can be emailed to: Feedback@PublicWare.dk

Copyright © 2003 Anders M. Hansen. All rights reserved.

3. Table of Contents

PublicWare Music Archive Plugin Documentation.....	1
1. Intro	1
2. Legal notes:	1
3. Table of Contents	1
4. General design.....	2
4.1. Plugin folder	2
4.2. Plugin settings.....	2
5. Types	2
5.1. TInfo record.....	2
6. Functions	3
6.1. Info procedure.....	3

6.2. OpenSong function	4
6.3. AddSong function.....	5
6.4. InfoDlg procedure.....	5
6.5. ConfigDlg procedure	6

4. General design

Music Archive Music Player Plugins, are basic Dynamic Link Library (DLL) files that contain the curtain number of functions and procedures. Music Archive then calls these functions and procedures when the plugin is in use.

4.1. Plugin folder

In a standard configuration all Music Archive plugins are located in the main application folder. Music Archive does how ever give the user the option to change the plugin folder. When installation of a new plugin the plugin installer must therefore have in mind that plugin path be altered or that the application it self is installed in a special folder.

The solution is that the plugin installer locks in the Windows Registry to locate the right folder to install the plugin into. The value is stored in

HKEY_CURRENT_USER\Software\PublicWare\Music Archive\Plugin\PluginPath. This value is a normal string, with one important deference. The value can contain dynamic strings that are replaced dynamically by Music Archive. There are two dynamic parameters in Music Archive version 1.0 to 1.10. The two are %APPPATH% and %DBPATH%. The %APPPATH% parameter is replaced by the application path. The %DBPATH% is replaced by the location of the current archive database. A plugin installer must be aware of these parameters.

4.2. Plugin settings

Plugin specific settings can be saved directly by the plugin. If desired by the programmer to save the settings under the PublicWare/Music Archive registry folder then there is reserved room to do this, without coming in contact with the rest of Music Archive. The plugin settings should be saved in the HKEY_CURRENT_USER\Software\PublicWare\Music Archive\Plugin\%PLUGINNAME%, folder where %PLUGINNAME% is replaced by the a the plugin name or filename. It is important to notes that the settings could be saved in a sub folder to the ...Music Archive/Plugin folder, because the folder it self contains information used by Music Archive, and a plugin could interfere with the Music Archive application it self.

5. Types

This section contains a description of all type records that is needed to make a Music Archive plugin.

5.1. TInfo record

The record is used by the Info procedure to return information about the plugin to Music Archive.

CompBuild: Integer;
 PluginType: Integer;
 Name: PChar;
 Version: PChar;

Company: PChar;
Comment: PChar;

Values

CompBuild

CompBuild inform Music Archive if the plugin is compatible with the current version of Music Archive.

The CompBuild to Music Archive 1.0, 1.01 and 1.10 is 3.

PluginType

PluginType inform Music Archive what type of plugin the plugin is.

The PluginType of a music player plugin is 1.

Name

Name returns a name string of the plugin to Music Archive.

Version

Version returns a version string of the plugin to Music Archive. General the version string should be constructed in the following format: Version . Reversion . Build (example: 1.01.231).

Company

Company returns a string with the name of the company who made the plugin.

Comment

The comment returns a comment string to Music Archive. The comment string can contain anything and is displayed to the user.

Remarks

No remarks

See also

Info

6. Functions

This section contains a description of all functions and procedures that is used in the plugin by Music Archive.

6.1. Info procedure

The Info procedure sends information to Music Archive about the plugin, and the functions in the plugin.

```
Info(  
    var Info: TInfo  
);
```

Parameters

Info

The Info parameter is a TInfo record that is fill by the plugin with information about the plugin. The information is used by Music Archive to display information about the plugin to the user and how the plugin works.

Return values

The Info parameter returns information about the plugin. (Read about the parameter in the parameter section).

Remarks

No remarks

See also

TInfo

6.2. OpenSong function

The OpenSong function is called from Music Archive when the music player should open a new song.

```
Integer OpenSong(  
    FileName: Pchar    //address of the songs filename  
);
```

Parameters

FileName

Points to a null-terminated character string that contains the complete filename and path of the song that should be opened by the music player.

Return values

If the function succeeds, the return value should be 0.

If the function fails, the return value should be 1.

If the function should be disabled in Music Archive the return value should be 2.

Remarks

When Music Archive loads the plugin, the OpenSong function is called with no filename specified. The string is empty. The plugin should then return 2 if the function should be disabled in the Music Archive menus. If the function shall continue to be enabled, then the function should return 0.

When more songs are opened at the same time, only the first is open with the OpenSong function. The rest is open with the AddSong function

See Also

AddSong

6.3. AddSong function

The AddSong function is called from Music Archive when a song should be added to the music players playlist.

```
Integer AddSong(  
    FileName: PChar    // address of the songs filename  
);
```

Parameters

FileName

Points to a null-terminated character string that contains the complete filename and path of the song that should be added to the music players playlist.

Return values

If the function succeeds, the return value should be 0.

If the function fails, the return value should be 1.

If the function should be disabled in Music Archive the return value should be 2.

Remarks

When Music Archive loads the plugin, the AddSong function is called with no filename specified. The string is empty. The plugin should then return 2 if the function should be disabled in the Music Archive menus. If the function shall continue to be enabled then the function should return 0.

See Also

OpenSong

6.4. InfoDlg procedure

The InfoDlg procedure is called from Music Archive when the user clicks the Plugin information button. The plugin should when display a dialog with information about the plugin.

InfoDlg();

Parameters

No parameters

Return values

No return values

Remarks

If the plugin does not have a information dialog. Then the button can be disabled in the Info procedure

See also

Info, ConfigDlg

6.5. ConfigDlg procedure

The ConfigDlg procedure is called from Music Archive when the user clicks the Plugin configuration button. The plugin should when display a dialog with configuration options to the plugin.

ConfigDlg();

Parameters

No parameters

Return values

No return values

Remarks

If the plugin does not have a configuration dialog. Then the button can be disabled in the Info procedure

See also

Info, InfoDlg