Helium Music Manager Simple API

Version 1.8, 2007-09-25

Introduction

The Helium Music Manager Simple API or SA for short allows you to quickly communicate with Helium Music Manager from an external application.
The API currently contains only basic features, but might be extended in the future. Suggestions are welcome.
The API is built on windows messages as in Winamp's API, in order that it should be as easy as possible to interact with the SA. The code examples in this document are written for Borland Delphi, but it should be simple enough to use the same code in VB and C/C++.

All of the messages except IPC_NEW_COVER_NOTIFICATION are sent in the form:

```
  Result := SendMessage(hwnd_helium, WM_HMM_IPC, (parameter), IPC_*);
```

(See the specification for IPC_NEW_COVER_NOTIFICATION below)

The parameter `hwnd_helium` can be retrieved with the following code:

```
  hwnd_helium := FindWindow(PChar(CLASS_NAME), nil);
```

`CLASS_NAME` is equal to:

```
  Const CLASS_NAME = 'THeliumMainForm';
```

If Helium Music Manager is not started or the window cannot be found for some other reason then `hwnd_helium` will be equal to zero.

`WM_HMM_IPC` is defined as:

```
  const WM_HMM_IPC = WM_USER;
```

Not all messages return a result and not all messages require a parameter. This is described in detail below.

Callbacks

The messages `IPC_GET_CURRENT_TRACK`, `PC_GET_LIST_ITEM` and `IPC_NEW_COVER_NOTIFICATION` are a bit special and requires more pogramming to handle than for the other messages.
Both these messages will return data back to the calling application via a callback, therefore before calling these messages you must ensure that you have defined this callback in your application. The callback itself is defined by an IPC message, `IPC_SET_CALLBACK_HWND`.

NEW_

To define this callback, send the following IPC to Helium Music Manager:

```
  SendMessage(wnd, WM_USER, Self.Handle, IPC_SET_CALLBACK_HWND);
```

`Self.handle` in this case is the window handle of your application, which Helium Music Manager will return data to.

The data returned from `IPC_GET_CURRENT_TRACK` and `IPC_GET_LIST_ITEM` comes as a `WM_COPYDATA` message which you will need to handle.

IPC_NEW_COVER_NOTIFICATION is also sent as a WM_COPYDATA message with the difference that you will not invoke this resulting message, Helium Music Manager will notify you automatically.

In short, you should handle them in your application as shown in the code below:

```
Type
  TForm1 = class(TForm)
  Private
    …
  public
    procedure WMCOPYDATA(var Msg: TWMCopyData); message WM_COPYDATA;
  end;
…
procedure TForm1.WMCOPYDATA(var Msg: TWMCopyData);
var
  sText: array[0..4096] of Char;
begin
  case Msg.CopyDataStruct.dwData of
    IPC_GET_CURRENT_TRACK:
    begin
      StrLCopy(sText, Msg.CopyDataStruct.lpData, Msg.CopyDataStruct.cbData);
      // Do something with the resulting string here
    end;
    IPC_GET_LIST_ITEM:
    begin
      StrLCopy(sText, Msg.CopyDataStruct.lpData, Msg.CopyDataStruct.cbData);
      // Do something with the resulting string here
    end;
    IPC_NEW_COVER_NOTIFICATION:
    begin
      StrLCopy(sText, Msg.CopyDataStruct.lpData, Msg.CopyDataStruct.cbData);
      // Do something with the resulting string here
    end;
  end;
end;
```

Adding files to the Active Playlist

To add a file to the Active Playlist, a special message needs to be sent from the client application to Helium Music Manager. This message should be sent as a WM_COPYDATA message.
An example of how this can be implemented is shown in the code below:

```
var
  cds: TCopyDataStruct;
  hwnd_helium: hwnd;
begin
  hwnd_helium := FindWindow(PChar(CLASS_NAME), nil);
  cds.dwData := IPC_ADDFILE_PLAY_PLAYLIST;
  cds.lpData := PChar(“c:\testfile.mp3”);
  cds.cbData := strlen(cds.lpData) + 1;
  SendMessage(hwnd_helium, WM_COPYDATA, 0, LongInt(@cds));
end;
```

Explanation of the parameters in the TCopyDataStruct structure:

- dwData: The message number
- lpData: A pointer to the filename that should be added to the Active Playlist
- cbData: The length of the filename plus one (for null padding)

Detailed list of all available IPC messages

IPC_PLAY = 100;

This message will start to play the active track in Helium Music Manager's Active Playlist
Parameters: None
Result: None

IPC_PLAYPAUSE = 101;

This message will pause the playback (if playing), otherwise it will resume the playback (if paused)
Parameters: None
Result: None

IPC_FORCEPAUSE = 102;

This message will force the playback to pause
Parameters: None
Result: None

IPC_STOP = 103;

This message will stop the playback
Parameters: None
Result: None

IPC_NEXT = 104;

This message will skip to the next track in the Active Playlist and start to play this track
Parameters: None
Result: None

IPC_PREVIOUS = 105;

This message will skip to the previous track in the Active Playlist and start to play this track
Parameters: None
Result: None

IPC_SET_VOLUME = 108;

This message will set the volume of the playback in Helium Music Manager
Parameters: Volume (0-100) in wParam of the message
Result: None

IPC_GET_VOLUME = 109;

This message will retrieve the current output volume in Helium Music Manager
Parameters: None
Result: The current volume (0-100)

IPC_GET_CURRENT_TRACK = 110;

This message will retrieve information about the track that is currently being played.
Parameters: None
Result: None
Remarks: See the callback topic above
Details: The information is in the following form:

- Title (including subtitle and remix if available)
- Artist (including secondary artist if available)
- Album
- Genre
-
- Year
- Comment

- Current Track
- The length of this file in seconds
-
- The filename including the full path for this file
-
- The rating of this file (0-10)
- The full path and filename to the assigned album cover
- Composer
- Lyricist
- Publisher
- Conductor
- Producer
- Copyright

separated by a

When a stream is played in Helium Music Manager, the following data will be returned:

- Title (retrieved from the stream-data)
- Additional stream information
- Reserved for future use
- Reserved for future use
- Reserved for future use
- Reserved for future use
- Reserved for future use
- Reserved for future use
- The full URL to the stream
- Reserved for future use
- Reserved for future use
- Reserved for future use
- Reserved for future use
- Reserved for future use
- Reserved for future use
- Reserved for future use
- Reserved for future use

All fields are separated by a tab character.

-
-
-
-
-

IPC_GET_DURATION = 113;

This message will retrieve the duration for the track currently being played
Parameters: None
Result: The duration (in seconds)

IPC_SET_POSITION = 114;

This message will set the position for the track currently being played

Parameters: The new position in wParam of the message. (In seconds)

IPC_IS_PLAYING = 115;
This message will let you know if Helium Music Manager is currently playing a song.
Parameters: None
Result: 1 if playing, 0 otherwise

IPC_IS_PAUSED = 116;
This message will let you know if Helium Music Manager currently is paused.
Parameters: None
Result: 1 if paused, 0 otherwise.

IPC_GET_LIST_LENGTH = 117;
This message will tell you have many entries there are in the Active Playlist in Helium Music Manager.
Parameters: None
Result: The number of entries in the Active Playlist.

IPC_SET_LIST_POS = 118;
This message will select a specific position in the Active Playlist and start to play this track.
Parameters: The desired position in wParam of the message.
Result: None

IPC_GET_LIST_ITEM = 119;
This message will retrieve information about a specific entry in the active playlist.
Parameters: Entry number in wParam of the message.
Result: None
Remarks: See the callback topic above
Details: The information is in the following form:

- Artist (including secondary artist if available)
- Title (including subtitle and remix if available)
- Length in seconds
- The filename including full path
- Rating (0-255)

All fields are separated by a tab character.

IPC_SET_CALLBACK_HWND = 120;
This message will set the callback window handle used for callback messages.
Parameters: Your window handle in wParam of the message
Result: 0 if the callback could not be set, 1 if the callback was successfully set.
Remarks: See the callback topic above

IPC_GET_LIST_POS = 121
This message will return the active position in the Active Playlist
Parameters: None
Result: Active position of the Active Playlist

IPC_GET_POSITION = 122
This message will return the current playing position in seconds
Parameters: None
Result: The player position

IPC_TRACK_CHANGED_NOTIFICATION = 123

If the client has registered itself with a callback to Helium Music Manager, Helium Music Manager will notify the client using this message each time a new track is played in Helium Music Manager.
Parameters: None
Result: None

IPC_SHOW_HELIUM_WINDOW = 124

This message will show Helium Music Managers main window.
Parameters: None
Result: None

IPC_GET_PLAYER_STATE = 125

This message will return the current state of the player.
Parameters: None
Result:

0. Player is stopped
1. Player is playing
2. Player is paused

IPC_PLAYER_STATE_CHANGED_NOTIFICATION = 126

When the playing state in Helium Music Manager changes, this message will be sent to a client.
Parameters: The new player state (as described for IPC_GET_PLAYER_STATE) in wParam
Result: None

IPC_AUTOENQUEUE_OPTIONS = 127

Opens the Auto-enqueue options dialog.
Parameters: None
Result: None

IPC_SET_REPEAT = 128

Call IPC_SET_REPEAT with 0 in wParam to set Auto-enqueue to stop.
Call IPC_SET_REPEAT with 1 in wParam to set Auto-enqueue to repeat.
Parameters: Mode in wParam:

0. Set Auto-Enqueue to Stop
1. Set Auto-Enqueue to Repeat

Result: None

IPC_SHUTDOWN_NOTIFICATION = 129

This message will be sent to a client when Helium Music Manager shuts down.
Parameters: None
Result: None

IPC_GET_REPEAT = 130

This message will return the current repeat state
Parameters: None
Result: 1 if Repeat is set, 0 otherwise

IPC_CLOSE_HELIUM = 131

Send this message to closedown Helium Music Manager. Please note that a confirmation message may be shown to the user, before Helium Music Manager can close.
Parameters: None
Result: None

IPC_SET_RATING = 132

Send this message to rate the song which is currently being played in Helium Music Manager. This function is equivalent to clicking on the rating control in the Internal Player. Please note that a confirmation message may be shown to the user, before Helium Music Manager can close.

Rating is a byte set to one of the following values:

0. No rating
1. 0.5 stars
2. 1 star
3. 1.5 stars
4. 2 stars
5. 2.5 stars
6. 3 stars
7. 3.5 stars
8. 4 stars
9. 4.5 stars
10. 5 stars

Parameters: The new rating in wParam
Result: None

IPC_RATING_CHANGED_NOTIFICATION = 639

When the rating is changed in Helium Music Manager for the file currently being played, Helium Music Manager will send a notification message (similar to IPC_TRACK_CHANGED_NOTIFICATION) to a registered client, containing the new rating (0-10).
Parameters: The new rating value in wParam
Result: None

IPC_NEW_COVER_NOTIFICATION = 800

When a new album cover is either added/updated or removed from Helium Music Managers Internal Player, a connected client will be notified with this message.
The full path and filename to the new cover is contained in the <COPYDATA_STRUCT>.

The message is sent from Helium Music Manager in the following form:

```
  SendMessage(hwnd_client, WM_COPYDATA, 0, <COPYDATA_STRUCT>);
```

A client should handle this message the same as the messages IPC_GET_CURRENT_TRACK and IPC_GET_LIST_ITEM.

IPC_GET_CURRENT_LYRICS= 801;

This message will retrieve the lyrics for the track currently being played. The lyrics are sent back to the client with a WM_COPYDATA message.
Parameters: None
Result: None
Remarks: See the callback topic above

IPC_ADDFILE_PLAY_PLAYLIST= 802;

Use this message to add a specific file to Helium Music Managers Active Playlist. If the file is added properly, it will automatically be started.

Parameters: None
Result: None
Remarks: This message is send as a WM_COPYDATA message from the client. Please see the comments in the topic “Adding files to the Active Playlist”

IPC_ADDFILE_QUEUE_PLAYLIST= 803;

Use this message to add a specific file to Helium Music Managers Active Playlist. The file will only be enqueued, eg. it won’t start to play automatically.
Parameters: None
Result: None
Remarks: This message is send as a WM_COPYDATA message from the client. Please see the comments in the topic “Adding files to the Active Playlist”