|  |
| --- |
| Podcast Utilities |
| Podcast Utilities |
| User Guide |
|  |
| **Derek Wilson** |
| **4/1/2011** |

|  |
| --- |
| How to use and configure Podcast Utilities |

# Contents

[1. Contents 2](#_Toc292524442)

[1. What is Podcast Utilities? 4](#_Toc292524443)

[1.1. Key benefits 4](#_Toc292524444)

[2. Installing Podcast Utilities 4](#_Toc292524445)

[2.1. Prerequisites 4](#_Toc292524446)

[2.2. Copying the distribution 4](#_Toc292524447)

[3. SyncPodcasts 5](#_Toc292524448)

[3.1. Typical Usage 5](#_Toc292524449)

[3.2. Command Line 5](#_Toc292524450)

[3.3. Example Control File 5](#_Toc292524451)

[3.4. Control File Format 6](#_Toc292524452)

[3.4.1. Global Section 6](#_Toc292524453)

[3.4.1.1. sourceRoot 6](#_Toc292524454)

[3.4.1.2. destinationRoot 7](#_Toc292524455)

[3.4.1.3. playlistFilename 7](#_Toc292524456)

[3.4.1.4. playlistFormat 7](#_Toc292524457)

[3.4.1.5. freeSpaceToLeaveOnDestinationMB 7](#_Toc292524458)

[3.4.1.6. sortfield 7](#_Toc292524459)

[3.4.1.7. sortdirection 7](#_Toc292524460)

[3.4.2. Podcast Section 7](#_Toc292524461)

[3.4.2.1. folder 7](#_Toc292524462)

[3.4.2.2. pattern 8](#_Toc292524463)

[3.4.2.3. number 8](#_Toc292524464)

[3.4.2.4. sortfield 8](#_Toc292524465)

[3.4.2.5. sortdirection 8](#_Toc292524466)

[4. GeneratePlaylist 9](#_Toc292524467)

[4.1. Typical Usage 9](#_Toc292524468)

[4.2. Command Line 9](#_Toc292524469)

[4.3. Control File Format 9](#_Toc292524470)

[4.3.1. Global Section 9](#_Toc292524471)

[4.3.1.1. destinationRoot 9](#_Toc292524472)

[4.3.1.2. playlistFilename 9](#_Toc292524473)

[4.3.1.3. playlistFormat 9](#_Toc292524474)

[4.3.2. Podcast Section 9](#_Toc292524475)

[4.3.2.1. folder 10](#_Toc292524476)

[4.3.2.2. pattern 10](#_Toc292524477)

# What is Podcast Utilities?

Podcast utilities are a set of utilities designed to help manage digital media on removable drives.

## Key benefits

There are many sophisticated media management programs available for organising digital media however the key differences that Podcast Utilities has are

* Portable  
  Podcast Utilities is a “portable” application, in that it does not need to be installed and can be run from any drive type, local network or removable flash drive
* Configurable  
  Almost any synchronisation solution can be configured, for example copy 3 of this podcast, all of these podcasts, in this order, and leave a certain amount of space on the device
* Lightweight  
  The application does not use large amounts of memory in a system tray icon all the time its not being used, it only uses system resources when its being run.
* Scriptable  
  Podcast Utilities lends itself to being run as a scheduled task or from a script. The API is “headless” and can easily be called and used from other applications
* Open Source  
  The source is open and available
* Free  
  There is no cost to using Podcast Utilities for any use.

# Installing Podcast Utilities

Podcast Utilities does not require any special installation, just copy the files from the dirstibution ZIP into any folder and run any of the utilities. The utilities can be run from local hard disks, network shares or removable media such as flash drives.

## Prerequisites

The following are required to install PodcastUtilities

1. A machine running Windows XP / Vista / 7
2. .NET Framework 3.5

## Copying the distribution

Unzip all the files from the distribution ZIP into any folder.

You will need to create your own control XML file for your specific media files, an example file is provided in the distribution ZIP along with this document.

# SyncPodcasts

SyncPodcasts is the command that will copy digital media from a source location to a destination.

## Typical Usage

SyncPodcasts is typically used when you have a large number of digital media files and you want to transfer a subset of these files to a removable drive. Only files that are not present in the destination are copied, any files that have been removed in the source are also removed in the destination.

## Command Line

The command line to run SyncPodcasts is

SyncPodcasts <controlfile>

Where <controlfile> is the filename of the XML file to control how SyncPodcasts will work. For example if your control file is called MyPhone.XML then the command line would be

SyncPodcasts MyPhone.xml

## Example Control File

The control file is an XML file and must be a legal XML file, in that elements must have a start and an end. The elements for example <sourceRoot> must be specified in the correct upper and lower case as shown here. This is an example of a control file

<podcasts version="1.0">

<global>

<sourceRoot>.\downloads</sourceRoot>

<destinationRoot>W:\Podcasts</destinationRoot>

<playlistFilename>podcasts.wpl</playlistFilename>

<playlistFormat>wpl</playlistFormat>

<freeSpaceToLeaveOnDestinationMB>2000</freeSpaceToLeaveOnDestinationMB>

<sortfield>name</sortfield>

<sortdirection>asc</sortdirection>

</global>

<podcast>

<folder>Test Match Special</folder>

<pattern>\*.mp3</pattern>

<number>-1</number>

</podcast>

<podcast>

<folder>Hanselminutes</folder>

<pattern>\*.mp3</pattern>

<number>-1</number>

</podcast>

<podcast>

<folder>From Our Own Correspondent</folder>

<pattern>\*.mp3</pattern>

<number>5</number>

</podcast>

</podcasts>

This control file will copy all the mp3 podcasts in “Test Match Special” and “Hansleminutes” folders followed by the first five mp3 files in the folder “From Our Own Correspondant” The source files are in a folder called downloads in the current folder and the destination folder is in “w:\Podcasts”.

A WPL playlist will be generated and at least 2GB of space will be left on the drive.

## Control File Format

The format of the control file is XML, so all <elements> must have a closing element </elements>

The root element of the file is <podcasts>

### Global Section

There is only one global section and it has settings that can only be set once or default values that can be overridden for each podcast

#### sourceRoot

This is the path to the source media files. It can be a relative or absolute path.

If it is a relative path (for example “.\media”) it is relative to the current working directory, usually this is the current directory if you are running the utility from a cmd prompt.

An absolute path (for example “m:\media”) will remain unaffected by where the utility is run from. Do remember that removable media can be referred to by different drive letters on different machines, however you can use the “Computer Management” console to permanently set the drive letter for a removable media device on a given computer. You can launch the Computer Management console by typing “compmgmt.msc” into a command prompt



In Windows 7 select Action -> All Tasks -> Change Drive Letters and Paths

#### destinationRoot

This is the path that we shall copy to. This must be an absolute pathname for example p:\media.

#### playlistFilename

This is the name of the file to write the playlist that contains all the media in the destination. The playlist contains all files in the destination not just the files copied in this run.

#### playlistFormat

The format the playlist should be in. Possible values are wpl or asx, (upper or lower case).

WPL format is a windows playlist for Windows Media Player and Windows Mobile based devices, ASX is a format used by many media players. Generate the playlist format that works with your player.

#### freeSpaceToLeaveOnDestinationMB

The space in MB to leave in the device, for example 2000 is 2GB. Copying will stop when there is less that this space left, however if there is less that this space left when SyncPodcasts is run then files will not be deleted to free up space.

#### sortfield

The field to be used to sort the media files. If we want to copy the “first” 5 files of a given podcast then this setting is used to determine which files are first.

If the value of this setting is creationtime (upper or lower case) then the files are sorted by the date and time the file was created otherwise any other value means they are sorted by name. Many podcasts incorporate a numbering system into the filenames so that using the filename will work fine however sometimes they either have no number or it is a random id in those cases you may want to sort the files on the bases of the date the file was downloaded.

This setting can be overridden on a podcast by podcast basis.

#### sortdirection

This field is also used to control the selection of media files, possible values are desc or asc , (upper or lower case) for descending or ascending sort order.

This setting can be overridden on a podcast by podcast basis.

### Podcast Section

After the global section there can be any number of podcast sections. Each section describes one folder of media files to be synced.

Each section must have a value for folder, pattern and number and can optionally override sortfield and sortdirection.

#### folder

This is the folder relative to the source that will be used to copy from and relative to the destination to copy to.

For example if the sourceRoot is “C:\Podcasts” and the destinationRoot is “X:\Media” and folder is “fred” then we will copy from “C:\Podcasts\fred” to “X:\Media\fred”

#### pattern

The is the filename pattern to look for when copying files, for example \*.mp3 will match all the mp3 files.

#### number

This is the number of files to copy. -1 means that all file in the source folder will be copued.

#### sortfield

This is the same as the setting in the global section, if it is missing then the one from the global section is used

#### sortdirection

This is the same as the setting in the global section, if it is missing then the one from the global section is used

# GeneratePlaylist

GeneratePlaylist is the command that will create a playlist file from a folder structure

## Typical Usage

GeneratePlaylist is typically used when you have a large number of digital media files yhat you have copied to a location and you would like to generate a playlist for all the files.

## Command Line

The command line to run GeneratePlaylist is

GeneratePlaylist <controlfile>

Where <controlfile> is the filename of the XML file to control how GeneratePlaylist will work. For example if your control file is called MyPhone.XML then the command line would be

GeneratePlaylist MyPhone.xml

GeneratePlaylist is the action performed by SYncPodcasts after all the copying has been done.

## Control File Format

The format of the control file is that same as that for SyncPodcasts however only some of the elements are used they are

The format of the control file is XML, so all <elements> must have a closing element </elements>

The root element of the file is <podcasts>

### Global Section

There is only one global section and it has settings that can only be set once or default values that can be overridden for each podcast

#### destinationRoot

This is the path to look for media files for the playlist. It is also the place where the playlist is intended to be used, all media file references in the playlist file are relative to this folder.

#### playlistFilename

This is the name of the file to write the playlist that contains all the media in the destination. The playlist contains all files in the destination not just the files copied in this run.

#### playlistFormat

The format the playlist should be in. Possible values are wpl or asx, (upper or lower case).

WPL format is a windows playlist for Windows Media Player and Windows Mobile based devices, ASX is a format used by many media players. Generate the playlist format that works with your player.

### Podcast Section

After the global section there can be any number of podcast sections. Each section describes one folder of media files to be synced.

Each section must have a value for folder and pattern. All files found in the folders specified are included in the playlist that is generated

#### folder

This is the folder relative to the destination root to look for media files in.

For example if the destinationRoot is “X:\Media” and folder is “fred” then we will look in the folder “X:\Media\fred” for media files to place in the playlist.

#### pattern

The is the filename pattern to look for when copying files, for example \*.mp3 will match all the mp3 files.