|  |
| --- |
| Podcast Utilities |
| Podcast Utilities |
| User Guide |
|  |
| **Derek Wilson** |
| **7/10/2011** |

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

# Contents

[1. Contents 2](#_Toc298079796)

[1. What is Podcast Utilities? 4](#_Toc298079797)

[1.1. Benefits 4](#_Toc298079798)

[2. Installing Podcast Utilities 4](#_Toc298079799)

[2.1. Prerequisites 4](#_Toc298079800)

[2.2. Copying the distribution 4](#_Toc298079801)

[2.3. Distribution contents 5](#_Toc298079802)

[3. DownloadPodcasts 6](#_Toc298079803)

[3.1. Typical Usage 6](#_Toc298079804)

[3.2. Command Line 6](#_Toc298079805)

[4. SyncPodcasts 6](#_Toc298079806)

[4.1. Typical Usage 6](#_Toc298079807)

[4.2. Command Line 6](#_Toc298079808)

[5. GeneratePlaylist 6](#_Toc298079809)

[5.1. Typical Usage 6](#_Toc298079810)

[5.2. Command Line 6](#_Toc298079811)

[6. Control File Format 8](#_Toc298079812)

[6.1. Example Control File 8](#_Toc298079813)

[6.1.1. Global Section 9](#_Toc298079814)

[6.1.1.1. sourceRoot 9](#_Toc298079815)

[6.1.1.2. destinationRoot 10](#_Toc298079816)

[6.1.1.3. playlistFilename 10](#_Toc298079817)

[6.1.1.4. playlistFormat 10](#_Toc298079818)

[6.1.1.5. freeSpaceToLeaveOnDestinationMB 10](#_Toc298079819)

[6.1.1.6. sortfield 10](#_Toc298079820)

[6.1.1.7. sortdirection 10](#_Toc298079821)

[6.1.1.8. maximumNumberOfConcurrentDownloads 10](#_Toc298079822)

[6.1.1.9. feed section 11](#_Toc298079823)

[6.1.1.10. format 11](#_Toc298079824)

[6.1.1.11. namingStyle 11](#_Toc298079825)

[6.1.1.12. maximumDaysOld 11](#_Toc298079826)

[6.1.2. Podcast Section 11](#_Toc298079827)

[6.1.2.1. folder 11](#_Toc298079828)

[6.1.2.2. pattern 12](#_Toc298079829)

[6.1.2.3. number 12](#_Toc298079830)

[6.1.2.4. sortfield 12](#_Toc298079831)

[6.1.2.5. sortdirection 12](#_Toc298079832)

[6.1.2.6. feed section 12](#_Toc298079833)

[6.1.2.7. url 12](#_Toc298079834)

[6.1.2.8. format 12](#_Toc298079835)

[6.1.2.9. namingStyle 12](#_Toc298079836)

[6.1.2.10. maximumDaysOld 12](#_Toc298079837)

# What is Podcast Utilities?

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

## 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. The number of concurrent downloads can easily be configured.
* 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.
* No need for an iTunes account  
  Podcasts promoted on iTunes can be downloaded however there is no need to have an iTunes account.

# 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.

## Distribution contents

The distribution contains the utilities in a binary form for running on Windows. The utilities (exe files) and their supporting dlls can be copied to any folder. Before you can use the utilities you will need to setup your control file. An example control file is included in the distribution.

If you are a developer and want to write code using PodcastUtilities then you need to add a reference to PodcastUtilities.Common.dll, this is the only file you need from the distribution though you may find it useful to refer to the Documentation.chm file for documentation on the API. PodcastUtilities.Common.dll is designed to be used “headless”, it makes no calls to win forms, web forms or console output.

# DownloadPodcasts

DownloadPodcasts is the command to be used to download media from all the feed urls specified int he control file. You need to have internet connectivity for this command to work.

## Typical Usage

Use this command to download podcasts either directly to a media playing device attached to your computer or to download media to your computer and then use SyncPodcasts to copy a subset to your media player.

## Command Line

The command line to run DownloadPodcasts is

DownloadPodcasts <controlfile>

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

DownloadPodcasts MyPhone.xml

# 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

# 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

All the utilities make use of a control file to control how they operate. 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>

## 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>

<maximumNumberOfConcurrentDownloads>20</maximumNumberOfConcurrentDownloads> <feed>

<maximumDaysOld>31</maximumDaysOld>

<namingStyle>pubdate\_url</namingStyle>

<format>rss</format>

</feed>

</global>

<podcast>

<feed>

<url>http://downloads.bbc.co.uk/podcasts/fivelive/tms/rss.xml</url>

</feed>

<folder>Test Match Special</folder>

<pattern>\*.mp3</pattern>

<number>-1</number>

</podcast>

<podcast>

<feed>

<url>http://www.hanselminutes.com/hanselminutes\_MP3Direct.xml</url>

</feed>

<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>

If this control file was used with the SyncPodcasts utility then it would 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.

If it was used with DownloadPodcasts then it would download all the episodes of Test Match Special and Hansleminutes published in the last 31 days. It would attempt to download 20 episodes at a time.

### 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. The sourceRoot is ued by DownloadPodcasts as the root folder to download episodes to, SyncPodcasts uses it as the place to read media from.

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

SyncPodcasts uses this as the path that we shall copy to, GeneratePlaylists uses it as the root folder to find media in. This must be an absolute pathname for example p:\media. GeneratePlaylist

#### playlistFilename

SyncPodcasts and GeneratePlaylist uses this as 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.

If you use the DownloadPodcasts utility to download files then you can use the namingStyle setting to control the filename to ensure that the filename can be sorted by publishing date.

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.

#### maximumNumberOfConcurrentDownloads

This element is only used by DownloadPodcasts and controls how many downloads are running at the same time. Your computer or the server that holds the podcasts may limit the number of connections it will accept.

#### feed section

This section is only used by DownloadPodcasts

#### format

The only supported podcast feed format is RSS

#### namingStyle

This is the style to use to name the file downloaded from a feed. The possible values are

|  |  |
| --- | --- |
| **Value** | **Meaning** |
| pubdate\_url | Published date in the YYYY MM DD HH MM (to enable the files to be sorted by published date) followed by the filename element from the end of the URL |
| pubdate\_title\_url | Published date in the YYYY MM DD HH MM (to enable the files to be sorted by published date) followed by the feed title (the folder used to store the episodes) followed by the filename element from the end of the URL |
| <anything else> | The filename element at the end of the URL for this episode |

#### maximumDaysOld

This can be used to prevent old episodes from being downloaded. The age of a podcast is calculated by comparing the current date time with the published date time in the feed.

### 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 or podcasts to be downloaded.

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

If you want to use DownloadPodcasts to download the episodes then you will need a feed/url element, you can optionally override any other element in the feed section of the global section.

#### folder

SyncPodcasts uses this as 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”

GeneratePlaylists uses this as the folder to find media to add to the playlist.

It is also used by DownloadPodcasts as the folder relative to the sourceRoot to download files to and also as the title of the podcast feed if the title is to be included in the episode file name using the namingStyle element.

#### pattern

SyncPodcasts uses this as the filename pattern to look for when copying files, for example \*.mp3 will match all the mp3 files.

#### number

SyncPodcasts uses this as 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

#### feed section

This section is only used by DownloadPodcasts

#### url

This is the url to the podcast feed, often this will be displayed as “RSS Feed” on a web site. If this is not present then DownloadPodcasts will ignore the podcast element.

#### format

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

#### namingStyle

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

#### maximumDaysOld

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