Format of DownloadMsConferences Config File

This XML file is divided into two parts: <Topics> and <Conferences>

<Conferences>

Each <Conference> within the <Conferences> group has three elements.

<Conference>

<Name>Build 2015</Name>

<TargetDir>D:\Downloads\VideoTest\BUILD 2015</TargetDir>

<RssUrl>http://s.ch9.ms/events/build/2015/rss/mp4</RssUrl>

</Conference>

|  |  |
| --- | --- |
| **Name** | The name of the conference that you want to download session videos and/or slides from. This shows up in the combo box that allows you to select the conference. |
| **TargetDir** | The directory where you want the session data to be download to. |
| **RssUrl** | The URL of the RSS file (in .xml format) that defines the information about the sessions for the specified conference. |
| **SpecialTopics** | The same format as the <Topics> data (but with the <SpecialTopics> tag). This allows you to override (i.e. replace) the default Topics on a conference-by-conference basis. |

Topics

With some conferences having literally hundreds of sessions, we didn’t want to dump everything into a single directory. One advantage of this is to be able to shunt certain session data to, say, an Azure sub-directory of the <TargetDir>, and others to, say, a Windows 10 sub-directory. Now the user can, for example, ignore everything in the Azure sub-directory (if that’s what s/he wants) and concentrate on sessions about Windows 10.

The Topics are processed in order against the Title of the session. Given the list of Topics below (current as of January 12, 2016), a given session title would be searched for the string “Xamarin” (in a case-insensitive fashion). If “Xamarin” were found anywhere in the Title, the session file(s) (video and/or slides) would be placed in the “Xamarin” sub-directory. If “Xamarin” were *not* found in the Title, it would be checked against each subsequent string in turn. If none of the Topics were found, the file(s) would be placed in the <TargetDir> directory for the conference.

Some of the entries have a backslash in the text. This supports a type of aliasing. For example, <string>Cloud\Azure</string>. This says to search the Title for “Cloud”, but if that text is found, don’t create a “Cloud” sub-directory and place the file(s) there. Rather put them in the Azure sub-directory.

The order of the Topics can be important. During development, the “.Net” string was placed before the “ASP.NET” entry. Because the program takes the first entry it finds, *all* ASP.NET articles were placed in the .Net sub-directory and none in the ASP.NET sub-directory. So the entry for ASP.NET was moved to be *before* the entry for .Net.

Thus there is the question of where to put a Title of, say, “Using C# to develop Windows Store, Android and IOS apps using Xamarin”. So does this get filed into the “C#” sub-directory, or “Windows Store”, or “Android”, or “IOS” or “Xamarin”? So even though “Xamarin” is the last word in the Title, since it is first in the list below, the session data will be placed in the “Xamarin” sub-directory. There was the possibility that we could have put the files(s) in the Xamarin directory and placed links to it in the other directories, but this wasn’t done.

Sub-directories are created “on-demand”. So if, say, there are no Titles with, say, IOS in the Title, then no IOS sub-directory will be created.

<Topics>

<string>Xamarin</string>

<string>Universal\Windows 10</string>

<string>Windows 10</string>

<string>Windows 8</string>

<string>Windows Store</string>

<string>Windows Phone</string>

<string>WP8\Windows Phone</string>

<string>Azure</string>

<string>Cloud\Azure</string>

<string>Android</string>

<string>iOS</string>

<string>Linux</string>

<string>Entity Framework</string>

<string>Windows Server</string>

<string>DevOps</string>

<string>Graphics</string>

<string>C#</string>

<string>HTML</string>

<string>ASP.NET</string>

<string>.Net</string>

<string>XAML</string>

<string>Visual Studio</string>

<string>C++</string>

<string>TypeScript</string>

<string>JavaScript</string>

<string>Office</string>

<string>XBox</string>

</Topics>