MARQUIS PFR INDEX WEB SERVICE API

SPECTRA LOGIC UPDATE - NOVEMBER 2016

INTRODUCTION

PFR Indexing can be performed one of two ways:

- Automatically, by configuring the PFR Index Service to scan for new media files and to automatically generate the index files as they are found;
- Manually, but making calls to this Index Web Service API in order to create each index file as required.

There is no “right” or “better” way to perform indexing, the option chosen will depend upon the nature of the workflow in your organization and whether you have the development resources available to program against a Web Service API such as this.

It is recommended that within a single installation of PFR, all indexing is performed by one of the two methods above and not with a mixture.

Regardless of which method of indexing is utilized, the functionality is provided as part of the PFR Index Service (a standard Windows Service application).

API DETAIL

The PFR Index Web Service API is a REST Interface that listens on TCP/IP port 60786.

INDEXING A FILE

Call Name	indexfile
Call Type	GET
Parameter	filepath
	The full path (via a mapped drive on the StorNext MDC) to the media file to be indexed.
Returns	See section below – Returned XML

Notes

This method will block while the index is created and will only return when either the index file has been created or for some reason it has not been possible to create the index file.

On HFS systems that leave stub files on disk, such as StorNext, if a request is made to create an index for a media file which has been truncated by StorNext, this call will cause the entire media file to be restored by StorNext.

The Web Service will support multiple concurrent calls to this command.

Example

The following example shows the HTTP request to send PFRIndexer a request to index the media file S:\Test\File1.mov:

```
http://PFRIndexer:60786/indexfile?filepath=S:\Test\File1.mov
```

REQUESTING FILE INDEX STATUS

Call Name	filestatus
Call Type	GET
Parameter	filepath
	The full path (via a mapped drive on the StorNext MDC) to the media file whose status is to be retrieved.
Returns	See section below – Returned XML

Notes

This method will block while retrieving the index status for a previously indexed file.

This method internally uses an XML file, generated by the indexer to retrieve the detailed status. This XML file is only generated by Quantum PFR version 1.1 and later. For files that were indexed in an earlier version of Quantum PFR, minimal information will be retrieved (just whether the file had been indexed or not).

The Web Service will support multiple concurrent calls to this API call.

Example

The following example shows the HTTP request to request from server PFRIndexer the status of media file S:\Test\File1.mov:

```
http://PFRIndexer:60786/filestatus?filepath=S:\Test\File1.mov
```

RETURNED XML

The return from either of the above calls is a packet of XML that describes the result of the call. In effect, internally, when requested to perform an index on a file, we then request its status and make that the return packet of the call.

Failed To Index

```
<?xml version="1.0" encoding="utf-8" ?>
<IndexerReport IndexResult="Failed" IndexTime="2011/10/21 15:30:15"/>
```

Good fileindex or filestatus call

```
<?xml version="1.0" encoding="utf-8" ?>
<IndexerReport IndexResult="Succeeded" IndexTime="2011/10/21 11:40:53"
FileStartTC="01:00:00;00" FileDuration="1800" FileFrameRate="29.97"/>
```

filestatus when file not present

```
<?xml version="1.0" encoding="utf-8" ?>
<IndexerReport IndexResult="Error File Not Found"/>
```

filestatus when file not indexed

```
<?xml version="1.0" encoding="utf-8" ?>
<IndexerReport IndexResult="Not Indexed"/>
```

REQUESTING FILE BYTE OFFSETS

Call Name	fileoffsets
Call Type	GET
Parameter	filepath The full path (via a mapped drive) to the media file whose partial offsets are being requested. tcin Timecode of the first frame requested tcout Timecode of the last frame requested **`fileframerate`** Frame rate, as returned in the file status report
Returns	See section below – Returned XML

Notes

This method will block whilst retrieving the start and end byte offsets for the requested timecodes. The offsets are extended in order to handle GOP and interleave ordering.

Timecode format should be in form hh:mm:ss:ff for non-drop framerates and hh:mm:ss;ff for drop framerates

Example

The following example shows the HTTP request to retrieve the byte offsets from the media file S:\Test\File1.mov:

`http://PFRIndexer:60786/fileoffsets?filepath=S:\Test\File1.mov&tcin=00:00:10:00&tcout=00:05:00:00&fileframerate=25`

RETURNED XML

The return from the fileoffsets call is a packet of XML that describes the result of the call.

Good fileoffsets call

```
<?xml version="1.0" encoding="utf-8" ?>
< fileoffsetvalues fileoffsetsResult="Succeeded” in_bytes="0x0060000"
out_bytes=”0x0080000"/>
```

fileoffsets when file not present

```
<?xml version="1.0" encoding="utf-8" ?>
< fileoffsetvalues fileoffsetsResult="Error File Not Found"/>
```

PARTIAL FILE REQUEST

Call Name	partialfile
Call Type	PUT
Parameter	filepath
	The full path to the media file whose partial offsets are being requested.
	tcin
	Timecode of the first frame requested
	tcout
	Timecode of the last frame requested
	`fileframerate`
	Frame rate, as returned in the file status report
	in_byte
	Byte offset of start of partial file relative to original file
	out_byte
	Byte offset of end of partial file relative to original file
	part_file
	Full UNC path to partial restored file fragment
	out_filename
	output file name for partial media file (care should be taken that this does not clash with other part restores, e.g. from other sections of the same source file)
Returns	Standard HTTP Status code

Notes

This method will use the parameters supplied to generate a Marquis XML file that will be used to create the partial output file.

PARTIAL FILE REQUEST STATUS

Call Name	partialfilestatus
Call Type	GET
Parameter	targetpartialname
	The out_filename value passed in the Partial File Request
Returns	See section below – Returned XML

Notes

This method will return status (% complete) for the creation of a partial media file initiated using the Partial File Request API call.

RETURNED XML

For Existing Job:

```
<?xml version="1.0" encoding="utf-8" ?>
< partialfilestatus phase="[Pending|Parsing|Transferring|Complete|Failed]"
percentcomplete="26"/>
```

For Non-existent Job:

```
<?xml version="1.0" encoding="utf-8" ?>
< partialfilestatus error="Job not found"/>
```