

Public Support - Support Request #19454

Get time stamp / time range with ADTF\_DATTOOL

2023-02-20 12:45 - hidden

<b>Status:</b>	Closed	<b>Product Issue Numbers:</b>  <b>Affected Products:</b> ADTF File Library 0.9.2 (BETA) <b>Platform:</b> Windows 10 64bit <b>Topic:</b> FileLibrary::DatProcessing <b>FAQ Links:</b>
<b>Priority:</b>	Normal	
<b>Category:</b>		
<b>Customer:</b>	VW	
<b>Department:</b>	EHHV	
<b>Requester's Priority:</b>	Normal	
<b>Support Level:</b>	2nd Level	
<b>Resolution:</b>	Solved Issue	

Description

Support Anfrage:

ich habe mehrere ADTFDAT-Files, welche Lidarscanner-Daten als einen Stream enthalten, bei dem die MediaDescription "anonymous" ist.

Ich möchte jetzt gern den Zeitstempel des 1. MediaSamples dieses Streams auslesen.

Dazu habe ich das bei ADTF3.15.2 mitgelieferte adtf\_dattool mit dem CSV-Prozessor verwendet, welches leider wegen der nicht vorhandenen MediaDescription eine Exception erzeugt.

Daher habe ich angefangen, auf Basis der mitgelieferten Quellcodes dieses CSV-Prozessors einen eigenen Prozessor zu programmieren, welcher sich nur um die Zeitstempel der MediaSamples kümmert.

Leider weiß ich jetzt nicht, wie ich meinen eigenen Prozessor jetzt zum Laufen bekomme. Ich habe das Binary in das vorhandene bin-Verzeichnis des adtf\_file unter pkg kopiert, aber der Aufruf des Tools liefert:

exception: no factory for processor timestamp

Jetzt weiß ich auch nicht mehr weiter. Wie bringe ich das zum Laufen? Außerdem musste ich mein Projekt als "Debug" bauen, da im Release-Mode folgende Fehler kommen (allein schon beim Bau-Versuch der unveränderten mitgelieferten Quellen):

Fehlerliste						
Gesamte Projektmappe 21 Fehler 0 von 10 Warnungen 0 Mitteilungen Erstellen + IntelliSense Fehlerliste durchsuchen						
	Code	Beschreibung	Projekt	Datei	Zeile	Unterdrückungszust...
	LNK2038	Konflikt ermittelt für "_ITERATOR_DEBUG_LEVEL": Der Wert "2" stimmt nicht mit dem Wert "0" in canwriter.obj überein.	canwriter	adtf_file.lib(stream_type.obj)	1	
	LNK2038	Konflikt ermittelt für "RuntimeLibrary": Der Wert "MDd_DynamicDebug" stimmt nicht mit dem Wert "MD_DynamicRelease" in canwriter.obj überein.	canwriter	adtf_file.lib(stream_type.obj)	1	
	LNK1000	Internal error during LIB::Search	canwriter	LINK	1	
	LNK2038	Konflikt ermittelt für "_ITERATOR_DEBUG_LEVEL": Der Wert "2" stimmt nicht mit dem Wert "0" in fileaccess.obj überein.	fileaccess	adtf_file.lib(adtf3_stream_type_seri...	1	
	LNK2038	Konflikt ermittelt für "RuntimeLibrary": Der Wert "MDd_DynamicDebug" stimmt nicht mit dem Wert "MD_DynamicRelease" in fileaccess.obj überein.	fileaccess	adtf_file.lib(adtf3_stream_type_seri...	1	
	LNK1000	Internal error during LIB::Search	fileaccess	LINK	1	
	LNK2038	Konflikt ermittelt für "_ITERATOR_DEBUG_LEVEL": Der Wert "2" stimmt nicht mit dem Wert "0" in referencedfiles.obj überein.	referencedfiles	adtf_file.lib(default_sample.obj)	1	
	LNK2038	Konflikt ermittelt für "RuntimeLibrary": Der Wert "MDd_DynamicDebug" stimmt nicht mit dem Wert "MD_DynamicRelease" in referencedfiles.obj überein.	referencedfiles	adtf_file.lib(default_sample.obj)	1	
	LNK1000	Internal error during LIB::Search	referencedfiles	LINK	1	
	LNK2038	Konflikt ermittelt für "_ITERATOR_DEBUG_LEVEL": Der Wert "2" stimmt nicht mit dem Wert "0" in csv_processor.obj überein.	csv_processor	adtf_file.lib(object.obj)	1	
	LNK2038	Konflikt ermittelt für "RuntimeLibrary": Der Wert "MDd_DynamicDebug" stimmt nicht mit dem Wert "MD_DynamicRelease" in csv_processor.obj überein.	csv_processor	adtf_file.lib(object.obj)	1	

Gäbe es grundsätzlich noch einen einfacheren Weg, den Zeitstempel des 1. MediaSamples eines Streams auszulesen? Klar, ich könnte einen Debugger ranhängen, aber das finde ich nicht so gut, wenn ich das wiederholt machen muss für eine Vielzahl an DATFILES.

Lösung:

Das geht mit dem adtf\_dattool ganz leicht:

```
adtf_dattool --liststreams <datfile>
```

liefert dann folgendes:

```
adtf_dattool --liststreams ~/devel/sdk/adtf3/src/examples/datfiles/example_file.adtfdat

~

adtfdat:
  VIDEO:
    type: adtf/image
    processors:
      time range (ns): [405356000, 14805306000]
      items: 874
  NESTED_STRUCT:
    type: adtf2/legacy
    processors:
      time range (ns): [0, 14805306000]
      items: 595
```

## History

### #1 - 2023-02-20 13:41 - hidden

- Project changed from Public Support to 20
- Description updated
- Status changed from New to In Progress
- Customer set to VW
- Department set to EHHV
- Topic set to FileLibrary::DatProcessing
- Affected Products ADTF File Library 0.9.2 (BETA) added

### #2 - 2023-02-20 14:03 - hidden

Hi Stefan,

das geht mit dem adtf\_dattool ganz leicht:

```
adtf_dattool --liststreams <datfile>
```

liefert dann folgendes:

```
adtf_dattool --liststreams ~/devel/sdk/adtf3/src/examples/datfiles/example_file.adtfdat

~

adtfdat:
  VIDEO:
    type: adtf/image
    processors:
      time range (ns): [405356000, 14805306000]
      items: 874
  NESTED_STRUCT:
    type: adtf2/legacy
    processors:
      time range (ns): [0, 14805306000]
      items: 595
```

Grüße,

Martin

### #3 - 2023-02-21 06:09 - hidden

- Status changed from In Progress to Customer Feedback Required

### #4 - 2023-02-21 06:45 - hidden

Hi Martin,

danke für die Lösung - da hätte ich ja echt auch selber drauf kommen können =)).

Beste Grüße,  
Stefan

#5 - 2023-02-21 07:04 - hidden

- Subject changed from *HowTo: Entwickeln eigener Prozessoren für ADTF\_DATTOOL* to *Get time stamp / time range with ADTF\_DATTOOL*
- Description updated
- Status changed from *Customer Feedback Required* to *To Be Closed*
- Resolution set to *Solved Issue*

#7 - 2023-02-21 07:30 - hidden

- Platform *Windows 10 64bit* added

#9 - 2023-03-13 07:52 - hidden

- Project changed from *20* to *Public Support*
- Status changed from *To Be Closed* to *Closed*
- Private changed from *Yes* to *No*

Files

image001.png	72.4 KB	2023-02-20	hidden
--------------	---------	------------	--------