

## Public Support - Support Request #952

### Trying to increase stack size for a selected filters

2017-08-01 18:27 - hidden

<b>Status:</b>	Closed		
<b>Priority:</b>	Normal		
<b>Category:</b>			
<b>Customer:</b>	BOSCH	<b>Product Issue Numbers:</b>	<a href="https://www.cip.audi.de/jira/browse/ACORE-7239">https://www.cip.audi.de/jira/browse/ACORE-7239</a>
<b>Department:</b>	CC-DA/ESI	<b>Affected Products:</b>	ADTF 2.10.0, ADTF 2.11.0, ADTF 2.12.0, ADTF 2.12.1, ADTF 2.13.0, ADTF 2.13.1, ADTF 2.13.2, ADTF 2.13.3, ADTF 2.14.0, ADTF 2.9.1
<b>Requester's Priority:</b>	Normal	<b>Platform:</b>	Windows 7 64bit
<b>Support Level:</b>	3rd Level	<b>Topic:</b>	ADTF::Common
<b>Resolution:</b>	Solved Issue	<b>FAQ Links:</b>	

#### Description

##### Supportanfrage

Wenn ich mich richtig erinnere, wurde beim Übergang von 2.9.0 auf 2.9.1 die Stack-Größe auf einen kleineren Wert eingestellt. Das hat bei einigen unserer Filter (z. B. dem Framegrabber-Filter) zu Problemen geführt, weil dort ganze Bilder auf dem Stack abgelegt wurden (ob das sinnvoll ist oder nicht sei dahingestellt). Mit der kleineren Stackgröße reichte der Speicher nicht mehr aus. Das gleiche Problem ist nun an anderen Stellen bei Filtern aus unserer SIL-Umgebung aufgetaucht, die gerade von ADTF 2.9.0 auf 2.13 portiert wird.

Ich war der Meinung man kann in Visual Studio die Stack-Größe für jeden Filter individuell einstellen. Die Kollegen haben mir nun zurückgemeldet, dass man das zwar kann, es aber keine Auswirkung hat (die Stack-Größe wird vom ADTF-Framework übernommen?).

Eine Lösung für das Problem ist die Filter - wie unseren Framegrabber-Filter - umzuschreiben und die Daten statt auf dem Stack auf dem Heap abzulegen.

Es kam nun die Frage auf, ob es auch eine Lösung ohne Code-Änderung gibt. Die ursprüngliche Anfrage nach einem ADTF 2.13 wieder mit größerer Stack-Size habe ich bereits verneint.

Habt Ihr noch eine andere Idee oder einen Weg einzelnen Filtern einen größeren Stack zu spendieren?

##### Lösung

Das Compiler Flag `/STACK:[0-9]` ist der Wert, der per Default verwendet werden soll, wenn man einen Thread erzeugt.

D.h. man weist nicht einem Filter Speicher zu, sondern den `CreateThread` aufrufen unter Windows. Daher kann das auch nicht hochgesetzt werden für alle.

Wenn man synchron in einem `OnPinEvent` arbeitet, dann arbeitet in Caller Kontext, der den Stack für diesen Thread erzeugt hat. D.h. die Stackgröße für einen Thread kann man nur verändern durch folgende Möglichkeiten:

1. Das `OnPinEvent` darf nicht synchron, sondern muss asynchron in einem eigenen thread verarbeitet werden:
  2. a) Dabei muss man einen `::CreateThread` Aufruf selber implementieren (siehe [https://msdn.microsoft.com/en-us/library/windows/desktop/ms682453\(v=vs.85\).aspx](https://msdn.microsoft.com/en-us/library/windows/desktop/ms682453(v=vs.85).aspx)) dort kann man die Stackgröße angeben
  2. b) Ein `a_utils::cThread` verwenden und die Compileroption `/Stack:<mysize>` verwenden.

#### History

##### #1 - 2017-08-02 09:06 - hidden

- Project changed from Public Support to 5
- Topic set to ADTF::Common

##### #2 - 2017-08-02 17:26 - hidden

- Status changed from New to In Progress

Hallo Roland,

mir ist keine andere Lösung bekannt als die Verwendung des heap als "Datenablage".

@Florian, ist dir da was bekannt?

### #3 - 2017-08-09 09:28 - hidden

- File *cmake\_variable\_stack.png* added

Hallo Roland,

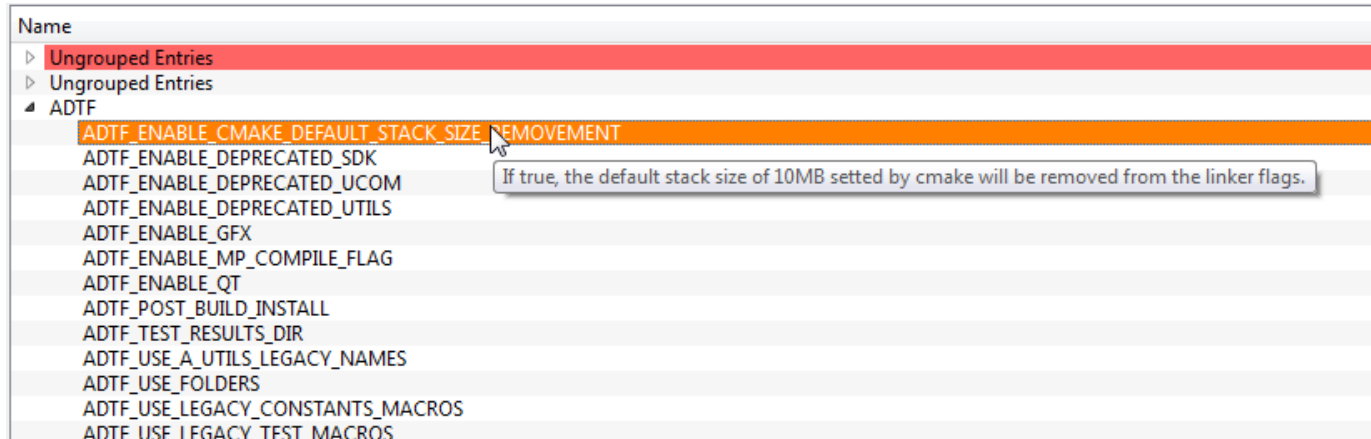
ab ADTF 2.12 ist das über CMake konfigurierbar:

- ACORE-7239: Make Changes from #22212 (Reserved stack memory for threads is to large) available in the ADTFConfig.cmake
  - (alte Redmine ID 22212)

und somit für den Anwender und seine Komponenten auch zugänglich.

Ansonsten hätte das Feature nur auf bereits kompilierte Lieferungen eine Auswirkung.

In der ADTFConfig.cmake wird die Cache Variable bereitgestellt.



Beantwortet das deine Frage ?

### #4 - 2017-08-09 09:29 - hidden

- Status changed from *In Progress* to *Customer Feedback Required*

### #5 - 2017-08-09 09:37 - hidden

- Resolution set to *Solved Issue*

- Product Issue Numbers set to <https://www.cip.audi.de/jira/browse/ACORE-7239>

### #6 - 2017-08-09 13:00 - hidden

Hallo Florian,

nein, so ganz ist es mir noch nicht klar, wo und wie die CMake Variable Auswirkungen hat.

Beim Bauen von ADTF ist die Variable offensichtlich auf TRUE gesetzt und die Stackgröße somit auf 1MB eingestellt.

Wenn ich nun einen eigenen Filter baue und die Variable auf dem Defaultwert FALSE lasse, dann bleibt es bei 10 MB Stackgröße. Aber was gilt denn nun, wenn man den Filter unter ADTF laufen lässt, die 1 MB vom Framework oder die 10 MB vom Filter?

Unserer Erfahrung sind es die 1MB vom Framework und dann bring uns die Variable nichts, da wir ja das ADTF-Framework nicht mit der Defaulteinstellung FALSE bauen können.

Mit freundlichen Grüßen / Best regards

Roland Herrmann  
CC-DA/ESI3

### #7 - 2017-08-09 14:41 - hidden

Hallo Roland,

wenn ihr nicht unsere gelieferte .cmake verwendet um zu bauen müsst ihr euch selbst um folgende Konfiguration kümmern: (siehe ADTFConfig.cmake)

```
# Remove /STACK:10000000 set by CMake. This value for stack size
# is very high, limiting the number of threads we can spawn.
# Default value used by Windows is 1MB which is good enough.
STRING(REGEX REPLACE "/STACK:[0-9]+" "" CMAKE_EXE_LINKER_FLAGS_REM "${CMAKE_EXE_LINKER_FLAGS}")
SET(CMAKE_EXE_LINKER_FLAGS "${CMAKE_EXE_LINKER_FLAGS_REM}" CACHE STRING "" FORCE)
```

```
MESSAGE (STATUS "Removed default /STACK from CMAKE_EXE_LINKER_FLAGS : ${CMAKE_EXE_LINKER_FLAGS}")
STRING (REGEX REPLACE "/STACK:[0-9]+" "" CMAKE_MODULE_LINKER_FLAGS_REM "${CMAKE_MODULE_LINKER_FLAGS}")
SET (CMAKE_MODULE_LINKER_FLAGS "${CMAKE_MODULE_LINKER_FLAGS_REM}" CACHE STRING "" FORCE)
MESSAGE (STATUS "Removed default /STACK from CMAKE_MODULE_LINKER_FLAGS : ${CMAKE_MODULE_LINKER_FLAGS}")
STRING (REGEX REPLACE "/STACK:[0-9]+" "" CMAKE_SHARED_LINKER_FLAGS_REM "${CMAKE_SHARED_LINKER_FLAGS}")
SET (CMAKE_SHARED_LINKER_FLAGS "${CMAKE_SHARED_LINKER_FLAGS_REM}" CACHE STRING "" FORCE)
MESSAGE (STATUS "Removed default /STACK from CMAKE_SHARED_LINKER_FLAGS : ${CMAKE_SHARED_LINKER_FLAGS}")
```

Beantwortet das deine Frage?

**#8 - 2017-08-09 15:00 - hidden**

Hallo Stephan,

nein, beantwortet meine Frage noch nicht.

Wenn ich das ADTF CMake Framework zum Bauen eines Filters verwende und dort die Größe für den Stack auf den CMake-Standardwert 10 MB eingestellt ist, ich den Filter dann in einem ADTF laufen lasse, welches mit den Einstellung für eine Stackgröße von 1MB gebaut wurde, wie groß ist dann der Stack, den mein Filter erhält? 10 MB wie beim Bauen des Filters angegeben oder 1 MB wie beim Bauen des ADTF Frameworks selber angeben?

Mit freundlichen Grüßen / Best regards

Roland Herrmann

**#9 - 2017-08-10 13:47 - hidden**

Hallo Roland,

...wie groß ist dann der Stack, den mein Filter erhält?

Ich gehe davon aus, das die 1MB des ADTF-Frameworks verwendet wird.

Falls du möchtest frage ich Pierre, sobald er aus dem Urlaub in KW35 wieder da ist. Er kann diese Frage beantworten.

Das CMake Flag wurde ursprünglich eingeführt, damit man als User die Möglichkeit hat von der damals default stack Größe von 10MB auf 1MB zu verkleinern, weil frühere ADTF Versionen auch unter 32Bit Windows Systemen verwendet wurden.

So wie ich das sehe, kommt ihr bei der Portierung eurer Filter nicht darum deren source code zu ändern, damit nur Speicher am heap allokiert wird. Ich hoffe ich konnte dir mit dieser Antwort weiterhelfen.

**#10 - 2017-08-14 13:38 - hidden**

Hallo Roland,

bitte um Feedback zum Ticketstatus bzw. der Erwartungshaltung.

**#11 - 2017-08-14 16:30 - hidden**

Hallo Stephan,

da es mir wichtig ist die Frage nach dem Stack korrekt zu klären, möchte ich Dich bitten bei Pierre nachzufragen. Ich gehe zwar auch davon aus aber Glauben heißt nicht Wissen.

Mit freundlichen Grüßen / Best regards

Roland Herrmann

**#12 - 2017-08-14 16:38 - hidden**

- Status changed from Customer Feedback Required to In Progress

Auf Wunsch des Anwenders Verlängerung des Due Dates auf 30.08.2017.

**#13 - 2017-09-04 14:03 - hidden**

- Status changed from In Progress to Customer Feedback Required

Hallo Roland,

hier die Antwort von Pierre:

Das Compiler Flag /STACK:[0-9] ist der Wert, der per Default verwendet werden soll, wenn man einen Thread erzeugt.

D.h. man weist nicht einem Filter Speicher zu, sondern den CreateThread aufrufen unter Windows. Daher kann das auch nicht hochgesetzt werden für alle.

Wenn man synchron in einem OnPinEvent arbeitet, dann arbeitet in Caller Kontext, der den Stack für diesen Thread erzeugt hat. D.h. die Stackgröße für einen Thread kann man nur verändern durch folgende Möglichkeiten:

1. Das OnPinEvent darf nicht synchron, sondern muss asynchron in einem eigenen thread verarbeitet werden:
2. a) Dabei muss man einen ::CreateThread Aufruf selber implementieren (siehe [https://msdn.microsoft.com/en-us/library/windows/desktop/ms682453\(v=vs.85\).aspx](https://msdn.microsoft.com/en-us/library/windows/desktop/ms682453(v=vs.85).aspx)) dort kann man die Stackgröße angeben
2. b) Ein a\_utils::cThread verwenden und die Compileroption /Stack:<mysize> verwenden.

Ich hoffe deine Frage ist damit beantwortet. Bitte um Feedback bis spätestens Mittwoch den 06.09.2017

**#14 - 2017-09-04 14:30 - hidden**

Hallo,

Frage ist beantwortet. Ticket kann geschlossen werden.

Danke  
Roland

Mit freundlichen Grüßen / Best regards

Roland Herrmann

**#15 - 2017-09-06 09:14 - hidden**

- Subject changed from *Erhöhung Stack Size für einzelne Filter möglich?* to *Trying to increase stack size for a selected filters*
- Description updated
- Status changed from *Customer Feedback Required* to *To Be Closed*
- Support Level changed from *2nd Level* to *3rd Level*

**#16 - 2017-09-06 09:14 - hidden**

- Status changed from *To Be Closed* to *Closed*

**#17 - 2018-01-25 11:48 - hidden**

- Project changed from *5* to *Public Support*
- Private changed from *Yes* to *No*

**Files**

---

cmake_variable_stack.png	14.9 KB	2017-08-09	hidden
--------------------------	---------	------------	--------