Public Support - Support Request #10133

Change active configuration after ADTF launch

2020-02-03 09:01 - hidden

Status:	Closed			
Priority:	Normal			
Category:				
Customer:	BOSCH	**Product Issue Numbers:**		
Department:		**Affected Products:**	ADTF 2.13.2	
Requester's Priority:	Normal	**Platform:**	Windows 10 64bit	
Support Level:	2nd Level	**Topic:**	ADTF::Common	
Resolution:	Solved Issue	**FAQ Links:**		

Description

Supportanfrage

We have an following requirement

- [Existing] ADTF is launched with parameter(mentioning the active configuration) by an exe
- [Existing] An registered ADTF service which gets initialized will then control the RL State (Application, Running and XSystem) based on external trigger/event
- [Requirement] Is there a way after ADTF launch and RLState:XSystem, switch the active configuration to any one which exists, after which an RLState to Application and Running could be applied to the one configuration which is made active.

Lösung

I would recommend to use the Python Support Service and a script, which maybe does the same as your self-implemented service:

- See also The ADTF Python Package

But if you want to do yourself within a service, you can set it using the ISessionManager interface... something like:

```
cObjectPtr<ISessionManager> pSessionManager;
RETURN_IF_FAILED(_runtime->GetObject(NULL, IID_ADTF_SESSION_MANAGER, (tVoid**) &pSessionManager,
__exception_ptr));
RETURN_IF_FAILED(pSessionManager->SetActiveConfiguration("my_active_configuration",
__exception_ptr));
```

You can also write an own application, see Creating a working combination of the Session and Filtergraph manager services

History

#1 - 2020-02-03 09:49 - hidden

- Project changed from Public Support to 5

- Topic set to ADTF::Common

- Customer set to BOSCH

#2 - 2020-02-04 09:52 - hidden

- Status changed from New to In Progress

#3 - 2020-02-04 10:26 - hidden

- Status changed from In Progress to Customer Feedback Required

Hi Robin,

I would recommend to use the Python Support Service and a script, which maybe does the same as your self-implemented service:

- See also The ADTF Python Package

But if you want to do yourself within a service, you can set it using the [ISessionManager](#) interface... something like:

```
cObjectPtr<ISessionManager> pSessionManager;
RETURN_IF_FAILED(_runtime->GetObject(NULL, IID_ADTF_SESSION_MANAGER, (tVoid**) &pSessionManager,
__exception_ptr));
RETURN_IF_FAILED(pSessionManager->SetActiveConfiguration("my_active_configuration", __exception_ptr));
```

You can also write an own application, see [Creating a working combination of the Session and Filtergraph manager services](#)

#4 - 2020-02-05 04:56 - hidden

Thanks for the suggestion !!
We shall try using the "ISessionManager" and do support us if we are to face any issue in coming days.
As of now we can close this ticket.

#5 - 2020-02-05 10:43 - hidden

- *Project changed from 5 to Public Support*

- *Subject changed from [Support] Change active configuration after ADTF launch to Change active configuration after ADTF launch*

- *Description updated*

- *Private changed from Yes to No*

- *Resolution set to Solved Issue*

#6 - 2020-02-05 10:43 - hidden

- *Status changed from Customer Feedback Required to To Be Closed*

#8 - 2020-07-07 12:49 - hidden

- *Status changed from To Be Closed to Closed*

```
cObjectPtr<ISessionManager> pSessionManager;
RETURN_IF_FAILED(_runtime->GetObject(NULL, IID_ADTF_SESSION_MANAGER, (tVoid**) &pSessionManager,
__exception_ptr));
RETURN_IF_FAILED(pSessionManager->SetActiveConfiguration("my_active_configuration", __exception_ptr));
```