

## Public Support - Support Request #10193

### Properties Changable at Runtime

2020-02-04 19:22 - hidden

<b>Status:</b>	Closed	
<b>Priority:</b>	Normal	
<b>Category:</b>		
<b>Customer:</b>	AUDI	<b>Product Issue Numbers:</b>
<b>Department:</b>	EF	<b>Affected Products:</b> ADTF 3.6.3
<b>Requester's Priority:</b>	Low	<b>Platform:</b> Windows 10 64bit
<b>Support Level:</b>	2nd Level	<b>Topic:</b> ADTF::Common
<b>Resolution:</b>	No Customer Feedback	<b>FAQ Links:</b>

#### Description

##### Supportanfrage

Is it possible to change filter properties in CE and propagate them live to a running config? Is there anything else a filter can do with a property except reading it at initialization with RegisterPropertyVariable() ?

##### Lösung

Is it possible to change filter properties in CE and propagate them live to a running config?

No, because the CE is decoupled from runtime, its for setting up a session **before** launching, as well as the ADTF Config Tool. To change properties during runtime, please refer to the ADTF Control, in interaction mode as well as in scripting automatization. Teaser: In ADTF 3.7 there will be UI Service and RPC Tooling as well for this use case, but currently the ADTF Control is your best choice (and after as well in humble headless option...)

Is there anything else a filter can do with a property except reading it at initialization with RegisterPropertyVariable() ?

Note that do not Register a property, you register a property\_variable which maps to a property accessed by user or code. At any time in your component you can receive the current value.

Teaser: ADTF 3.7 provides a property\_changed callback for each registered property so do not have to poll for changes anymore. Of course you can use a cProperty directly but it has no benefits compared to a property\_variable.

Which feature are you missing ?

What's your aim ?

Note: RegisterPropertyVariable does not read, it creates the above mentioned mapping. you do not have to read, you just register in the constructor and then use the property\_variable (with its current value) whenever and wherever you want.

The linked guide for properties is a good start, as well as our Demos/examples.

#### History

##### #1 - 2020-02-05 09:57 - hidden

- Project changed from Public Support to 11

- Status changed from New to In Progress

- Topic set to ADTF::Common

##### #2 - 2020-02-05 10:06 - hidden

- Status changed from In Progress to Customer Feedback Required

Hello,

~~Yes, this is possible.~~

Please have a look at our [ADTF 3 Guides, section Change Properties of ADTF Components while a Session is running](#)

Please report, if this helped

**#3 - 2020-02-05 10:08 - hidden**

@Florian

are there further information on this?

**#4 - 2020-02-05 10:17 - hidden**

Hi Falk,

Is it possible to change filter properties in CE and propagate them live to a running config?

No, because the CE is decoupled from runtime, its for setting up a session **before** launching, as well as the ADTF Config Tool. To change properties during runtime, please refer to the ADTF Control, in interaction mode as well as in scripting automatization. Teaser: In ADTF 3.7 there will be UI Service and RPC Tooling as well for this use case, but currently the ADTF Control is your best choice (and after as well in humble headless option...)

Is there anything else a filter can do with a property except reading it at initialization with RegisterPropertyVariable() ?

Note that do not Register a property, you register a property\_variable which maps to a property accessed by user or code. At any time in your component you can receive the current value. Teaser: ADTF 3.7 provides a property\_changed callback for each registered property so do not have to poll for changes anymore. Of course you can use a cProperty directly but it has no benefits compared to a property\_variable.

Which feature are you missing ?  
What's your aim ?

Note: RegisterPropertyVariable does not read, it creates the above mentioned mapping. you do not have to read, you just register in the constructor and then use the property\_variable (with its current value) whenever and wherever you want. The linked guide for properties is a good start, as well as our Demos/examples.

**#5 - 2020-02-13 14:45 - hidden**

- Project changed from 11 to Public Support
- Description updated
- Private changed from Yes to No
- Resolution set to No Customer Feedback

**#6 - 2020-02-14 08:11 - hidden**

- Status changed from Customer Feedback Required to To Be Closed

**#9 - 2020-07-07 12:49 - hidden**

- Status changed from To Be Closed to Closed