

USING GLOBAL PARAMETERS IN OPENDEVELOPER

What are Globals and Why are they Useful?

Global parameters are a set of properties (such as Channel or Options) in TTX that are inherently used when a method designed to use them is called. Global parameters reduce the amount of values that need to be specified each time the method is called.

How are they Modified?

Global parameters are modified through a set of methods that can be called to either change a global parameter's value or return the currently set value.

The method used will depend on the type of global parameter. String type global parameters such as FillItem or Options will use the global parameter method associated with Strings. Likewise, number based global parameters such as Channel will utilize the global parameter method associated with number values.

Global Parameter Dependency Reference List

There are several methods that inherently use the global parameters whenever they are called. This reference list shows these methods and the global parameters they are dependant on.

GetEpocsV - RespectOffsetEpoc

GetEpocsExV - RespectOffsetEpoc, T1, T2, MaxReturn

GetValidTimeRangesV - RespectOffsetEpoc, MaxReturn

SetEpocTimeFilter - RespectOffsetEpoc, AutoRefEpoc

SetEpocTimeFilterV - RespectOffsetEpoc, AutoRefEpoc

SetFilterArray - RespectOffsetEpoc

SetFilterWithDescEx - RespectOffsetEpoc

ReadEventsSimple - T1, T2, MaxReturn, Channel, SortCode, Options

ReadWavesOnTimeRangeV - T1, T2, SortCode, FillItem, FillValue, WaveSF, WavesMemLimit

ReadWavesV - T1, T2, Channel, SortCode, Options, FillItem, FillValue, WaveSFEvent, WaveSF, WavesMemLimit

Methods for Modifying Globals

SetGlobalV - set a numeric global parameter to the numeric value specified.

SetGlobalStringV - set a string global parameter to the string value specified.

SetGlobals - set multiple global parameters of different types using a single call.

ResetGlobals - Reset all global parameters to their default value

GetGlobalV - Return the current value of the specified numeric global parameter.

GetGlobalStringV - Return the current value of the specified string global parameter.

Tip

The SetGlobals method is extremely useful for modifying the values of multiple global parameters in one call regardless of the type of value expected.

This method accepts a string argument for all globals being set. Both string and numeric global parameters are passed as strings.

Matlab Example Code	Description
TTX.SetGlobalV('Channel', 1) events = TTX.ReadEventsSimple('Snip')	Sets the global parameter Channel to 1, then reads only Channel 1 of the event Snip.
TTX.SetGlobalStringV('Options', 'FILTERED')	Sets the Options global parameter to 'FILTERED'. Events read will be FILTERED based on EPOC values used to set the filter (see SetFilter or SetEpocTimeFilterV).
TTX.SetGlobals('Options = FILTERED; Channel = 1; T1 = 1; T2 = 10')	Sets several global parameters with a single method call. A single string expression is used to set the values.
TTX.ResetGlobals	Resets all global parameters to their default settings.
TTX.GetGlobalV('Channel')	Returns the current value of the global parameter Channel.



GLOBAL PARAMETERS

Here is a listing of all global parameters that are associated with OpenDeveloper.

Name	Default Value	Valid Values	Description
<i>AutoRefEpoch</i>	1 (Enabled)	0 (Disabled), 1	Automatically references time stamps to an event's epoch.
<i>Channel</i>	0 (All Channels)	Any Integer	Specifies any or all channel numbers.
<i>FillItem</i>	'DataPoints'	DataPoints, Name, Sort, Time, Freq, xIndex, yIndex, zIndex, Channel, FixedNum	Specifies the value that the returned matrix will be filled with.
<i>FillValue</i>	1	Any Number	Specifies the value that the returned matrix will be filled with if FillItem = 'FixedNum'
<i>MaxReturn</i>	100000	Any Integer	Specifies the maximum number of events to be returned.
<i>Options</i>	'ALL'	ALL, NEW, FILTERED	Specifies whether all events are extracted, filtered or new.
<i>RespectOffsetEpoc</i>	1	0, 1	Filters out the events that occur after the offset of the buddy epoch, otherwise it will include all events until the next onset.
<i>SortCode</i>	0 (All)	Any Integer (0-31)	Specifies the sort codes to be included.
<i>T1</i>	0.0	Any Number	Specifies the starting point for dependant methods.
<i>T2</i>	0.0	Any Number	Specifies the stopping point for dependant methods.
<i>WavesMemLimit</i>	33554432 (~32 MB)	Any Number in Bytes	The maximum memory that can be used by a single method.
<i>WaveSF</i>	0 (Hz)	Any Number	Specifies the sampling frequency, in Hz, used to sample a certain event.
<i>WaveSFEvent</i>	''	Any event string (except epoch events) present in the block	Specifies an event whose sampling frequency is used to sample a certain event.

