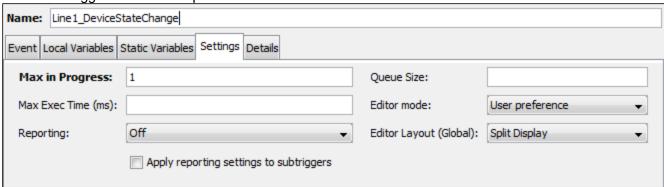You are here: IoT Gateway User Guide > Projects and triggers > Defining a trigger > Trigger settings

Topic updated on September 21, 2023

Trigger settings

The **Settings** tab allows you to set parameters for the trigger that provide specific execution processing such as the number of triggers that can be processed at the same time.

Parameter descriptions

Parameter	Description
Max in Progress	Sets the number of trigger instances that are allowed to execute concurrently. The default value is one. If more than this number are scheduled to execute concurrently, the trigger instances will be queued for later execution based on the **Queue Size** parameter. For example a schedule event type, with a periodic frequency set to 100 milliseconds, but the trigger executions are taking longer than 100 milliseconds. This will lead to multiple trigger instances that are scheduled to execute concurrently.
Max Exec Time (ms)	The maximum execution time for the trigger, in milliseconds. If the trigger execution exceeds this time, a warning message is logged in the Exceptions Log (even though the **Reporting** parameter might be set to off). This warning message in the Exceptions Log is for informational purposes, it does not end or change the trigger's execution.
Reporting	The trigger reporting option controls when a trigger report is generated and written to the Reports Log. Care should be taken when specifying the reporting option for triggers, based on the need for the additional information, the frequency of the execution of the triggers, and the availability of system resources on the node (CPU, memory, disk). For all options other than Off, the trigger is generated as the trigger steps through the trigger's actions. Once the trigger execution completes as successful or an error, the trigger report may be discarded or written to the Reports Log.
Apply reporting settings to subtriggers	An option to have the trigger reporting setting selected for this trigger applied to all "called" subtriggers. If selected, the trigger reporting selection (On, Off, etc.) is passed to a subtrigger that is "called" using the **Execute SubTrigger** action. This passing of the reporting selection applies to all levels of called subtriggers.

Parameter	Description
	For example: trigger 1 calls subtrigger 2, which calls subtrigger 3 and subtrigger 4. Trigger 1's reporting selection is passed to, and overrides, the reporting setting in subtrigger 2, subtrigger 3, and subtrigger 4.
Queue Size	The number of trigger instances that can be queued for later execution when the number of concurrent trigger instance execution reaches the **Max in Progress** parameter. Trigger instances queued are executed once another executor becomes available (its trigger instance ends execution). The default value is blank, which is the same as a zero, which indicates the queue is disabled. If the **Queue Size** value is reached, then additional trigger instances are marked as **Overflow** and are not executed.
Editor mode	The trigger editor to use when editing this trigger: • **List** - use the List Editor. Trigger actions are specified in a list form, with the action's routes set by selecting from a list. • **Canvas** - use the Canvas Editor. Trigger actions are positioned in a drag and drop flow chart form, with action routes indicated by connecting action output ports to other action's input ports. • **User preference** - Remembers the last editor mode used for the trigger.
Editor Layout (Global)	A Global setting, for all triggers, for the trigger editor's display layout style.

Use of trigger settings during development, debug and production

As an application's triggers are developed and debugged, the settings can be used to aid the development process and to understand a trigger's execution path.

Once the application is put into production, the settings can be monitored to understand how the triggers are executing in different production environment scenarios.

For example:

- The **Reporting** parameter can be set to On when developing a trigger and then set to Off before going into production.

- The **Max in Progress** parameter can be used to ensure that only one instance of a trigger is executing concurrently (value set to 1) or it can be used to allow multiple concurrent instances of the trigger to execute concurrently. Allowing multiple instances of a trigger to execute concurrently may require serialization or concurrency concepts in the trigger's application logic.

- The **Queue Size** parameter can be used to allow temporary spikes in trigger instances being scheduled for execution to be queued for later execution, instead of discarding the instances as an **Overflow**.

- An increase in a trigger's **Overflow** counter (in the trigger list in the project's tab on the Projects window) may mean that the **Max in Progress** and **Queue Size** parameters need to be adjusted based on the trigger's event frequency and the trigger's execution time.

About Telit | Contact Us | Legal Notices | Terms of Service | Privacy Policy