### evtctxadd

Creates the event context with the specified key, if the input data matches the conditional expression.

#### Syntax

evtctxadd dynamic=t key=KEY\_FIELD CONDITIONAL\_EXPR

or

evtctxadd [expire=INT{mon|d|h|m|s}] [maxrows=INT] [timeout=INT{mon|d|h|m|s}] topic=STR key=KEY\_FIELD CONDITIONAL\_EXPR

Required Parameter

**dynamic=BOOL**

Option to enable dynamic options (default: f).

* t: Gets topic, expire, timeout, maxrows options from input records. You cannot use the topic, expire, timeout, and maxrows options when dynamic=t.
* f: Disables dynamic option.

**topic=STR**

Name of event context. the topic acts like the table name in an in-memory database. You cannot use this option when dynamic=t.

**key=KEY\_FIELD**

Name of the field to record the unique key that distinguishes the event context.

**CONDITIONAL\_EXPR**

Conditional expression for creating an event context.

Optional Parameter

**expire=INT{mon|d|h|m|s}**

Expiration period in units of mon (month), d (day), h (hour), m (minute), and s (second). The event context is deleted once a specified expiration period has passed from when the event context is created. Once the expiration period is set, it is not extended, even if there is input data that matches the conditional expression CONDITIONAL\_EXPR. You cannot use this option when dynamic=t.

**maxrows=INT**

Maximum number of rows to store in the event context (default: 10). You cannot use this option when dynamic=t.

**timeout=INT{mon|d|h|m|s}**

Length of time after the last event received until timeout. You can specify in units of mon (month), d (day), h (hour), m (minute), and s (second). You cannot use this option when dynamic=t.

#### Usage

**Generate a timeout if it takes more than 10 seconds to receive a response after sending a message.**

evtctxadd topic=txmatch key=txkey timeout=10s type == "send"| evtctxdel topic=txmatch key=txkey type == "recv"

The example query command consists of the following event context creation/deletion commands:

* If the **type** value is send, use the evtctxadd command to create the event context.
* If the **type** value is recv, use the [evtctxdel](https://docs.logpresso.com/en/query/evtctxdel-command) command to delete the event context.

When the event context condition occurs, both commands distinguish the event context by associating the topic txmatch and the event context key field **txkey**.

Now, if the following event data is passed as an input,

json "{'txkey':'001122', 'type':'send'}"json "{'txkey':'001122', 'type':'recv'}"

The event context is created when the first data is provided. Different events occur depending on the time the second data is provided.

* If the second data is entered within 10 seconds, an event context deletion (EventCause.REMOVAL) event occurs.
* If the second data is entered after 10 seconds, or if it is not provided, a timeout (EventCause.TIMEOUT) event occurs.

Depending on the cause of the deletion of the event context, you can perform different processing with subsequent commands.