

SQL Server: Introduction to Extended Events

Module 6: Defining Event Sessions

Jonathan M. Kehayias
Jonathan@SQLskills.com



Introduction

- Event sessions have minimum requirements for creation and modification
- Predicates can be defined in different ways that can affect event session performance
- Actions can be used to add data to events, but should only be used when necessary
- In this module we'll cover:
 - Event session requirements
 - Basic predicate usage
 - Understanding short-circuiting
 - Adding data with actions

Event Session Requirements

- **Requires at least one event for CREATE**
 - Targetless event sessions are allowed to facilitate side-effecting actions like a `create_dump_all_threads` to capture a stack dump of the server when an event fires
- **No requirements for ALTER**
 - All of the targets and events can be dropped from an existing session
 - Dropping all events allows in-memory targets to stop collecting data for data analysis

Basic Predicate Usage

- **Predicates define the logical rules for which an event can fire in a session**
- **Predicates can be defined two ways:**
 - Using basic operators common in Transact-SQL (=,<>,,>,>=,<,<=,LIKE)
 - Using textual comparators (equal_uint64, like_i_sql_unicode_string, etc,)
- **Boolean expressions can be joined and grouped together logically using AND/OR with parenthesis**
- **Inverse predicates can be created using NOT before the Boolean evaluation**
- **The predicate for a single event is limited to 3000 characters**
 - Make use of standard arithmetic operators to shorten predicate length when possible

Understanding Short Circuiting

- Predicates are evaluated in the logical order written for the event that is firing
- Extended Events predicates use short-circuit evaluation as soon as the first predicate clause evaluates to false
- Predicates should be defined such that the predicates that are most likely to evaluate to false occur earliest in the predicate order
- Predicates on event column data should occur before predicates on global data to reduce the need to collect the global data for evaluation if the earlier predicates evaluate to false

Adding Data with Actions

- Actions execute synchronously on the firing thread and should be used only where they actually add benefit to the event data
- Large actions like `sql_text` may require additional consideration for event size during session configuration to prevent event loss
- Don't add actions to perform post-collection event correlation; use the causality tracking session option instead
- Actions may not return data for every event, even though they can be added to every event
 - Execution context can prevent collection of "global state data" like the `sql_text`

Summary

- Event sessions require a single event to be created
- Altering an event session can remove all events and targets from the session
- Predicates can be defined using the standard logical operators common in Transact-SQL or using textual comparators
- Predicates can short-circuit evaluation to prevent firing and prevent the need to collect global data for evaluation
- Actions can add data to events but may not be available for all thread contexts
- The next module will look at:
 - Event session management