DBA-498-P

Extended Events Deep Dive

Jonathan Kehayias
Principal Consultant
SQLskills.com

About Jonathan Kehayias

- Consultant/Trainer/Author/Speaker
- Principal Consultant, SQLskills.com

 Email: jonathan@sqlskills.com
  Blog: http://sqlskills.com/blogs/jonathan
  Twitter: @SQLPoolBoy

- Author: SQL Server 2008 Internals and Troubleshooting
- Author: Troubleshooting SQL Server; A Guide for the Accidental DBA
- Microsoft Certified Master: SQL Server 2008, SQL Server MVP
- Author of Using Extended Events whitepaper on MSDN
- Regular presenter at PASS Summit, SQL Saturdays, SQLBits and SQL Connections conferences
- Developer Extended Events Manager SSMS Addin for SQL Server 2008 Open Source project on Codeplex
- Course instructor for *Microsoft Certified Master – Database* and *SharePoint* qualifications

SQL Server MVP Deep Dives, Volume 2

ONE SURGERY.
A lifetime of smiles.

Operation Smile

after   before
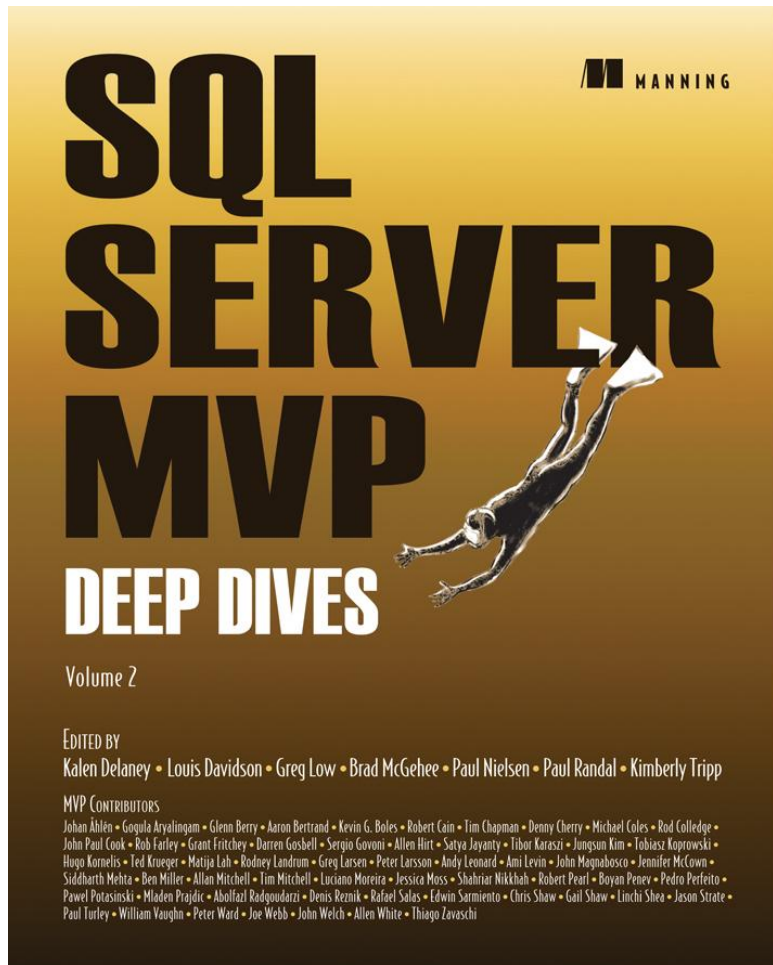
www.operationsmile.org

www.manning.com/delaney

What is Extended Events

- Advanced event collection infrastructure introduced in SQL Server 2008

- Highly flexible implementation allows complex configurations for event collection that simplify problem identification

DEMO

Simplified Problem Identification

Extended Events vs SQL Trace

SQL Trace

- Events fire based on server wide bitmap

- All data buffered to trace controller and then sent to the traces

    - Entire Events may be discarded by a trace at filtering

    - Excess columns will be discarded from the dataset if not part of the trace definition

- Rowset provider and trace file collect all events and do not provide aggregation
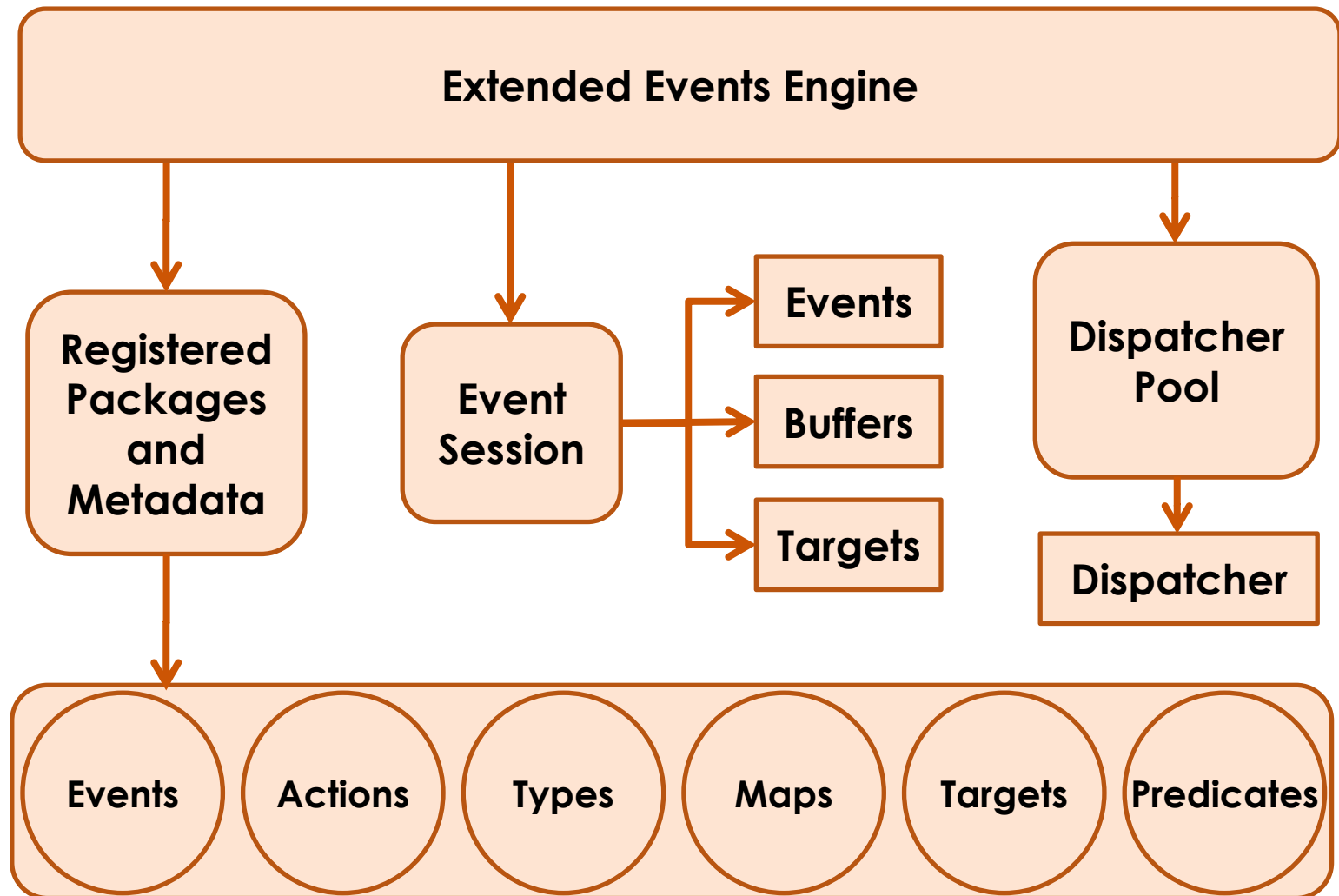
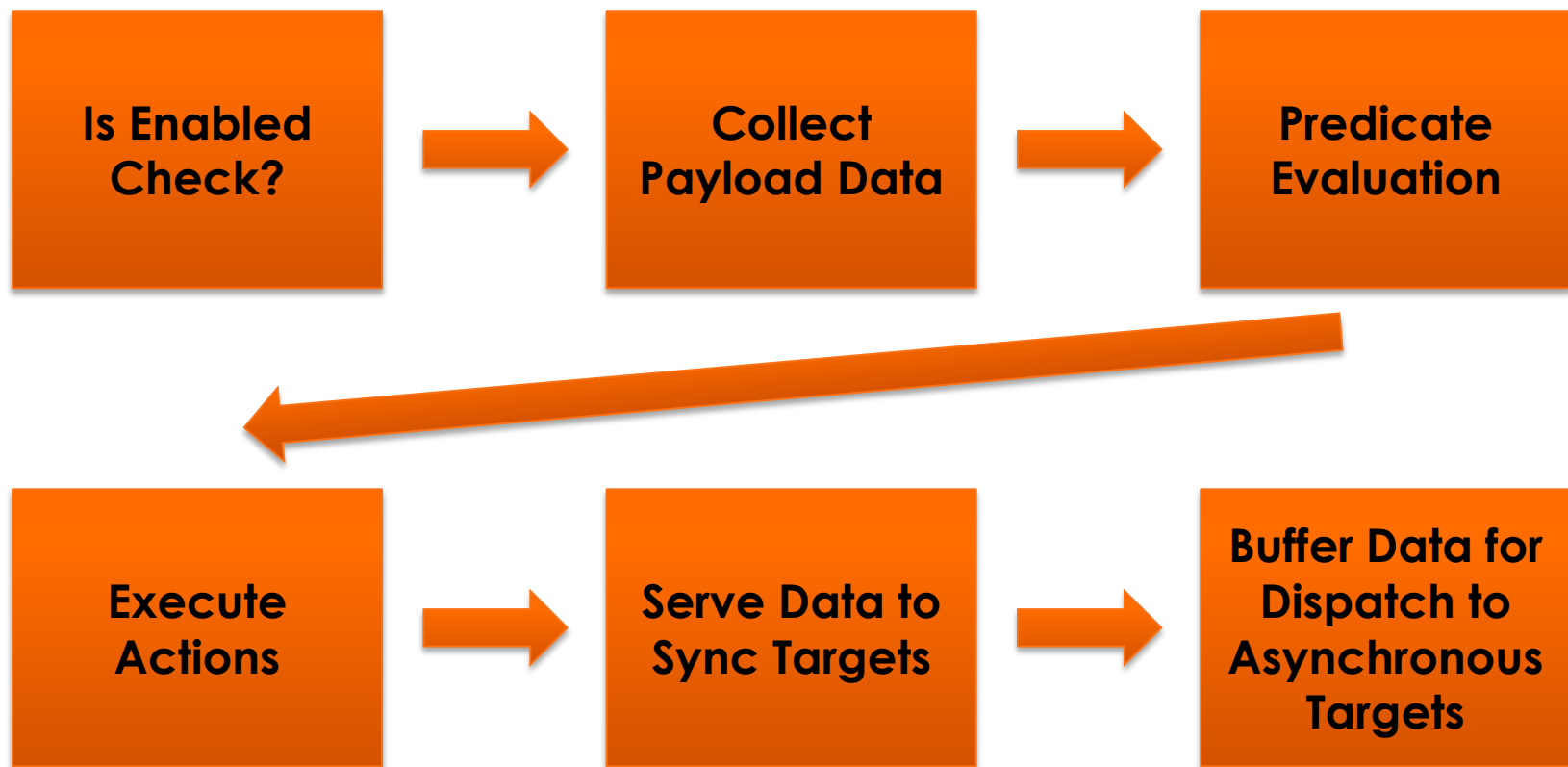# SQL Trace Architecture
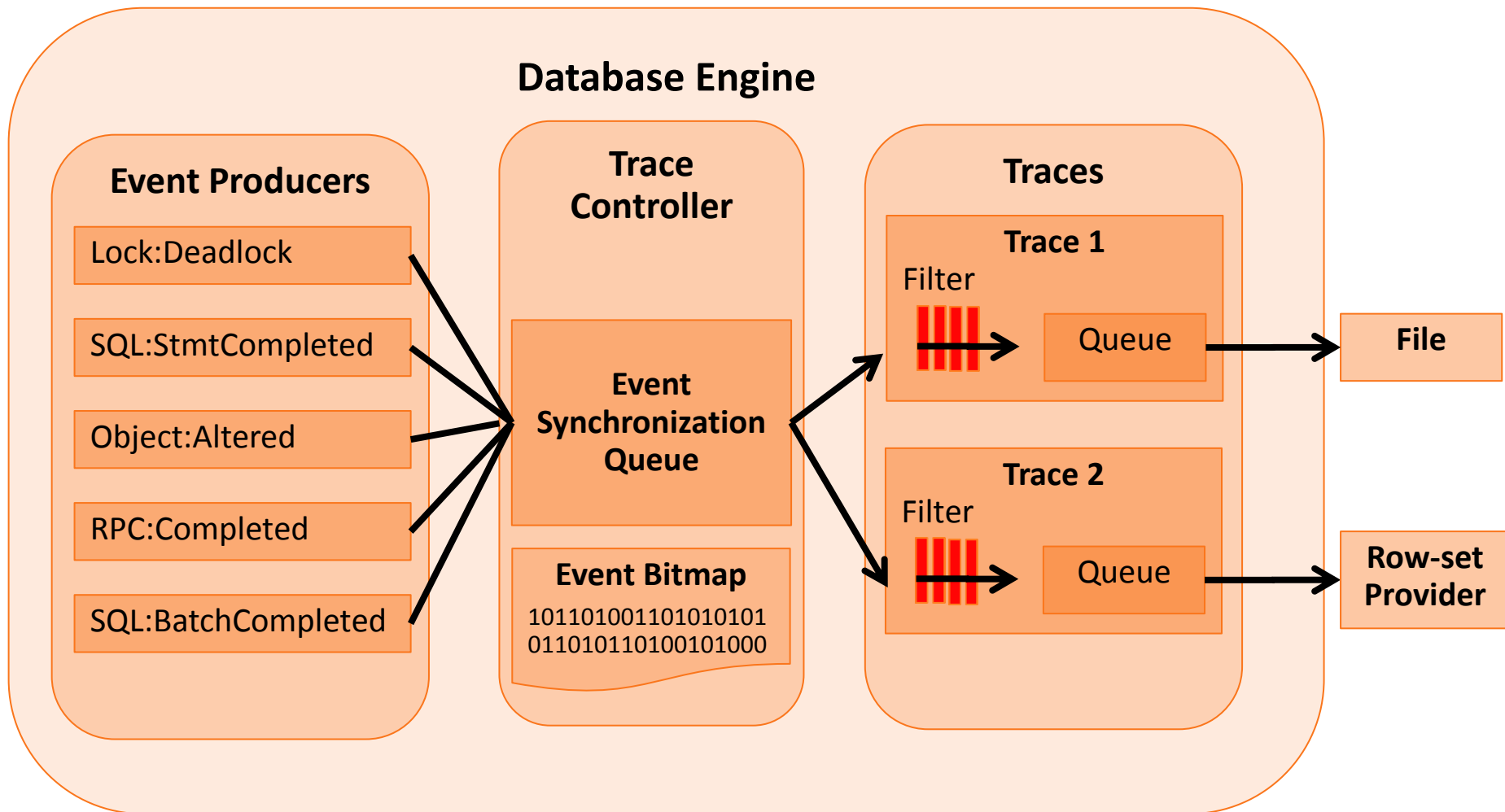
Extended Events vs SQL Trace

Extended Events

- Events fire based on session level configuration

- Events only fire if predicate evaluation succeeds reducing excess

- Additional data only collected when event fires

- New targets provide specialized aggregation of data for complex analysis

Extended Events Architecture

Extended Event Lifecycle

Is Enabled Check?	→	Collect Payload Data	→	Predicate Evaluation

Execute Actions	→	Serve Data to Sync Targets	→	Buffer Data for Dispatch to Asynchronous Targets

What No UI Support?

- Managed through DDL commands, system catalog views and DMV's

- No native SSMS support or tools to manage Extended Events in SQL Server 2008
  - Extended Events Addin on Codeplex provides integrated management in SSMS

- New SSMS integration in SQL Server Denali CTP3
  - Is not backwards compatible with SQL Server 2008
  - In memory targets still require knowledge of XML to extract event data

Metadata Views

sys.dm_xe_packages
- Contains a entry for each of the packages registered in the Extended Events Engine
- Each package will have a unique guid, which is used to map its objects to it

sys.dm_xe_objects
- Contains information about the objects (events, actions, predicates, targets, types and maps) available in the packages registered in the Extended Events Engine
- Objects have the package_guid of the package that loaded them
- The object_type column determines the type of object

Metadata Views

sys.dm_xe_object_columns

- Contains information about the columns, or data elements, that exist for a specific object
  - readonly - additional system metadata about an event, that allows the integration with Event Tracing for Windows
  - data - the data elements that are returned by default when the event fires
  - customizable – control collection of additional data elements in the event payload that have a higher cost to collect

sys.dm_xe_map_values

- Contains key/value pairs for each of the Maps defined in the system
- Maps are linked by their object_package_guid to the specific package that created them

Demo

Querying Metadata Views

Session Definition Catalog Views

sys.server_event_sessions

- Contains the name and session level options for each event session that exist inside of the Extended Events Engine

sys.server_event_session_events

- Contains information about the events and predicates that are a part of an event session

sys.server_event_session_actions

- Contains one row for each action for each event in an event session

Session Definition Catalog Views

sys.server_event_session_targets

- Contains one row for each of the configured targets that are defined for an event session

sys.server_event_session_fields

- Contains one row for each of the configured options for each target defined for an event session

Active Session DMVs

sys.dm_xe_sessions

- Contains the event session name, memory buffer configuration, event loss information, and date/time the event session was started for each active event session (STATE=START) in the SQL Server Instance

sys.dm_xe_session_events

- Contains information about each of the events and the event predicate for events defined in an active event session

sys.dm_xe_session_event_actions

- Contains one row for each action that is defined on an event in an active event session

Active Session DMVs

sys.dm_xe_session_targets

- Contains the name, type of target, execution statistics and an XML column for each target that exists for an active event session

- The data for memory resident targets is in the xml column and for persisted targets this will contain execution statistics

sys.dm_xe_session_object_columns

- Contains information about the configuration options of each of the targets, as well as information about the customizable columns for events in the event session

Demo

Querying Active Session DMVs

Management DDL

CREATE EVENT SESSION
- Creates a new event session based on the events, actions, predicates, targets, and session options provided
- All event sessions are created in a stopped state

ALTER EVENT SESSION
- Add or remove events and targets from an event session
- Change session configuration options for a stopped event session
- Alter the state of an event session to start or stop

DROP EVENT SESSION
- Removes an event session from the system entirely
- Memory resident targets are not available after a event session is dropped

Event Session Management

- Event sessions must be explicitly started after creation
- Active event sessions can be modified with immediate effect (except for session options)
- Started event sessions can be dropped without being stopped
- Changing the configuration of an event/target requires that the event/target be dropped and added back to the event session with the new configuration
- Dropping all events from a active session can stop event collection allow for data analysis when using in-memory targets

Event Session DDL

CREATE EVENT SESSION

ALTER EVENT SESSION

DROP EVENT SESSION
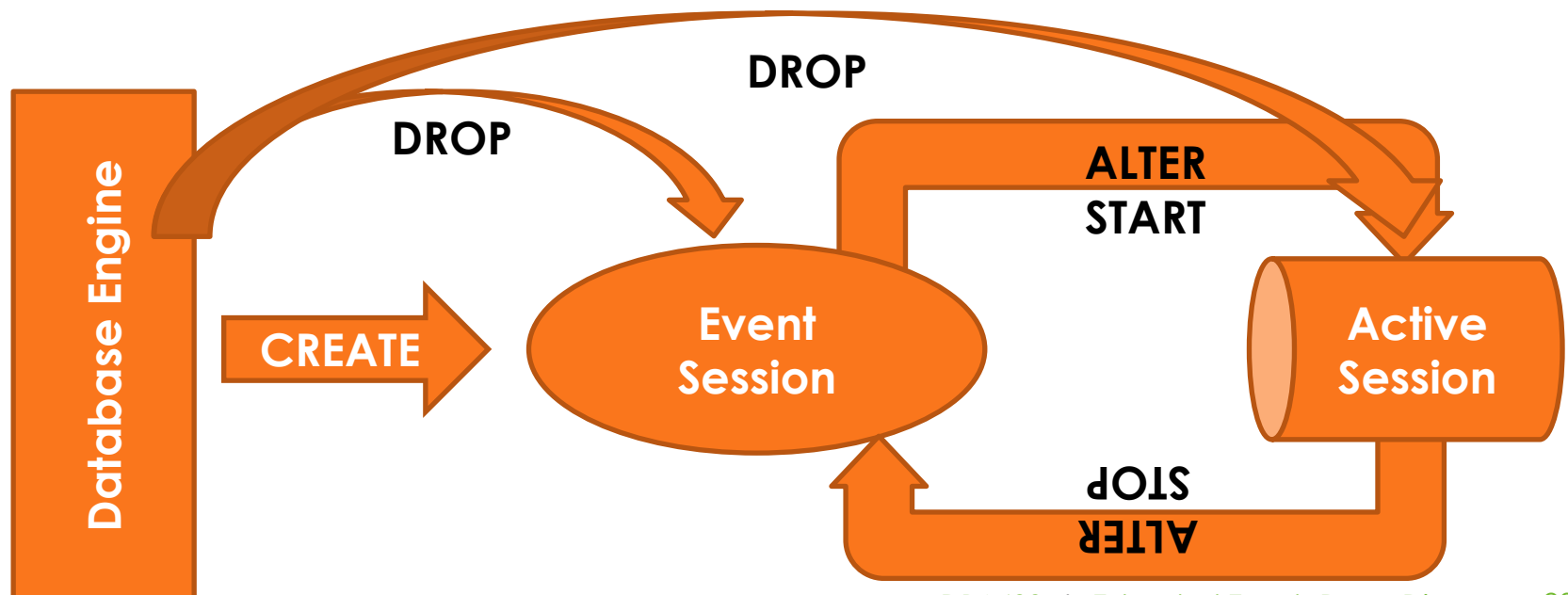
Creating an Event Session

```
CREATE EVENT SESSION [long_duration_statements]
    ON SERVER
ADD EVENT sqlserver.sql_statement_completed
(       ACTION
                (       sqlserver.tsql_stack,
                        sqlserver.sql_text )
        WHERE
                (       sqlserver.is_system = 1
                        AND duration > 5000000 ))
ADD TARGET package0.ring_buffer
    (  SET MAX_MEMORY = 1024    )
WITH (MAX_DISPATCH_LATENCY = 15 SECONDS)
```

Working with Event Sessions

```
ALTER EVENT SESSION [long_duration_statements]
ON SERVER
STATE=START | STOP


ALTER EVENT SESSION [long_duration_statements]
ON SERVER
ADD EVENT sqlserver.sql_statement_starting
ADD TARGET package0.asynchronous_file_target
( SET filename =
'C:\SQLPASS\long_duration_statements.xel',
metadatafile =
'C:\SQLPASS\long_duration_statements.mta')
```

Querying Event Session Data

```
SELECT
      CAST(target_data as xml) AS target_data
FROM sys.dm_xe_sessions AS s
JOIN sys.dm_xe_session_targets AS t
      ON s.address = t.event_session_address
WHERE s.name = 'long_duration_statements'
  AND t.target_name = 'ring_buffer'
```

Demo
Managing Event Sessions

Building Event Sessions

Selecting Events
  Configuring Customizable Columns
Actions
  Data Adding
  Side impacting
Predicates
  Boolean operations
  Textual Comparators
Targets
Event Session Options

Customizable Columns

Add/remove additional data, of potentially limited usage, into the event payload that may have additional cost to collect (e.g. statement or batch_text)

DDL for customizable column usage

SET customizable_column = <value> [,...n]

The column size may exceed the normal buffer size for an event session resulting in large buffer usage

- If an appropriate value for MAX_EVENT_SIZE has not been set in the session options, large events may be lost
- Consider allocating additional memory buffer space if collecting large customizable columns and testing the impact of CPU partition modes

Demo

Using Customizable Columns

Data adding Actions on Events

- Actions may not return data for every event, even though they can be added to every event
  - Execution context can prevent the collection of "global state data"
- Actions execute synchronously on the firing thread and should be used only where they actually add benefit to the event data
- Don't add actions to do event correlation use causality tracking instead

sqlserver.sql_text Action

- Is actually the InputBuffer for the executing session and not the actual sql_text.

- InputBuffer masking of sensitive data using keyword match list for words like "password" blocks data collection

- Use TRACK_CAUSALITY and sql_statement_starting/completed event collect_statement customizable column in Denali instead

- Large actions like sql_text may require additional consideration for event sizing during session configuration to prevent event loss

Side effecting Actions on Events

Allow side effects to occur when an event fires

- Performing a memory dump of a single thread
- Performing a memory dump of all threads
- Inserting a debug break - This really does what it says it does!

Keep in mind that these still execute synchronously and take extreme caution when using one of these in a production environment

Predicates

- Define short circuits on event data before global state data where possible to prevent the need to gather the state data

- Textual comparators provide the most flexibility in filtering but are harder to use

  - E.g. Bitmask checking, greater than last max value, less than last min value, and divides evenly by int64 can perform event sampling not possible with basic Boolean expressions

Demo

Defining Predicates

Event Session Options

TRACK_CAUSALITY

- Attaches a guid and sequence number to events for correlation of what events led to other events

STARTUP_STATE

- Start the event session automatically with SQL Server

MAX_DISPATCH_LATENCY

- Dispatch events to targets faster
- May impact the dispatcher pool if multiple event sessions exist

EVENT_RETENTION_MODE

- Determines whether single events, entire buffers, or no events can be lost by the event session
- No event loss can significantly impact performance

Event Session Options

MAX_MEMORY

- MAX_MEMORY is not the actual max memory for the event session since buffers align on 64K boundaries

MAX_EVENT_SIZE

- Determine what the maximum event size may be based on the event definitions
- Can not be smaller than a single memory buffer for the event session and the large buffers will be in addition to the regular buffers for the session

MEMORY_PARTITION_MODE

- Determines the number of memory buffers created for the event session; none=3 buffers, per_node=3 buffers per NUMA node, per_cpu=2.5 buffers per scheduler

Demo

Configuring Event Session Options

ring_buffer Target

- General purpose in-memory target that performs FIFO collection of events
- DMV limitations in SQL Server 2008 may result in unreadable XML from sys.dm_xe_session_targets
- Ideal target for small data sets where event loss is acceptable

Options

- **Max_memory** – the maximum amount of memory in KB to be used; older events are dropped when this is reached
- **Occurrence** – sets the preferred number of events of each type to keep

bucketizer (histogram) Target

- Specialized in-memory target that collects events into buckets simplifying analysis of event frequency

- Does not store actual event data, only the count of occurrence

pair_matching Target

- Specialized target that matches events based on a specified criteria and discards the matched pairs, leaving unmatched events only in the data

- Careful selection of the pairing criteria has to be made or invalid event pairings will occur resulting in incorrect analysis

  - E.g. lock_acquired and lock_released are not fired the number of times when lock escalation occurs

- Troubleshooting incorrect matching should be done using TRACK_CAUSALITY and the ring_buffer or file target

- Same output XML as the ring_buffer for event data

event_counter Target

- Counts how many events occurred for the event session

- Use this when defining an event session for an unknown workload to determine how many events you can expect based on your event sessions definition as a part of planning the appropriate targets to use for the session

asynchronous_file_target Target

- Similar to the trace file in SQL Trace, this target collects event data for long term detailed analysis

- Event data has the same XML scheme as the ring_buffer target

- Target data must be queried through the SQL Server database engine in SQL Server 2008

- Denali CTP3 includes a .NET API for reading the file outside the database engine

etw_classic_sync_target Target

- Provides integration between SQL Server Eventing and Event Tracing for Windows

- Built on the classic provider from Windows Server 2000 and not the manifest provider from Windows Server 2008

- Creates an XE_DEFAULT_ETW_SESSION in windows, and only one exists for the entire server, making segregation of data from multiple instances impossible

- ETW session requires command line flush and close even if the Event Session in SQL is dropped

Demo

Working with Targets

Extended Events SSMS Addin

- SSMS support for Extended Events in SQL Server 2008
  - Future release will provide backwards compatibility between Denali and 2008 in a single UI
- Allow creation and scripting of Event Sessions
- View session data for started sessions
- View server metadata
- SMO type functionality through library implementation provides integration with PowerShell for SQL Server 2008

Demo

Extended Events SSMS Addin for 2008

Changes in SQL Denali

Implements all of the events from SQL Trace

- Some events have been renamed to match SQL Trace event names

SSMS UI support for Extended Events in Denali versions only

- New Event Session Wizard simplifies event session creation

- Grouping, filtering, and aggregating event data possible using tabular grid view of data from file target

Streaming provider implemented with .NET API support for third party applications

Demo

Changes in SQL Server Denali

Questions?

Resources

XEvent a Day Blog Series

http://sqlskills.com/blogs/jonathan/category/XEvent-a-Day-Series.aspx

My Blog Posts

http://sqlskills.com/blogs/jonathan/category/Extended-Events.aspx

Using SQL Server 2008 Extended Events Whitepaper

http://msdn.microsoft.com/en-us/library/dd822788(v=sql.100).aspx

Extended Events SSMS Addin

http://extendedeventmanager.codeplex.com/

Complete the Evaluation Form to Win!

Win a $100 gift card – every day – just for handing in your completed form. Each session evaluation form represents a chance to win.

Pick up your evaluation form:

- In each presentation room
- Online on the PASS Summit website

Drop off your completed form:

- Near the exit of each presentation room
- At the Registration desk
- Online on the PASS Summit website

Thank you

for attending this session and the
2011 PASS Summit in Seattle