

Analyzing SQL Server Query Plans

CAPTURE QUERY PLANS USING EXTENDED EVENTS
AND TRACES



Pinal Dave

SQL SERVER PERFORMANCE TUNING EXPERT

<https://blog.sqlauthority.com>



Agenda



Introduction to query plans

SQL Profiler

SQL Traces

SQL Extended Events

Demos



“We fail more often because we solve the **wrong** problem than because we get the **wrong** solution to the **right** problem.”

Russell L. Ackoff

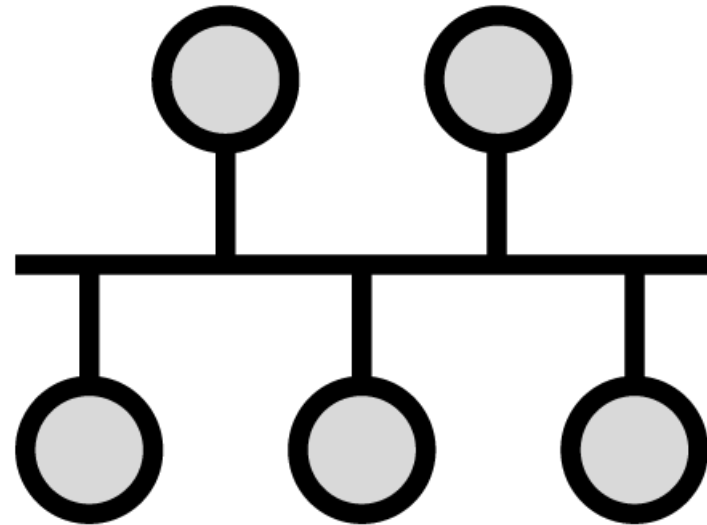
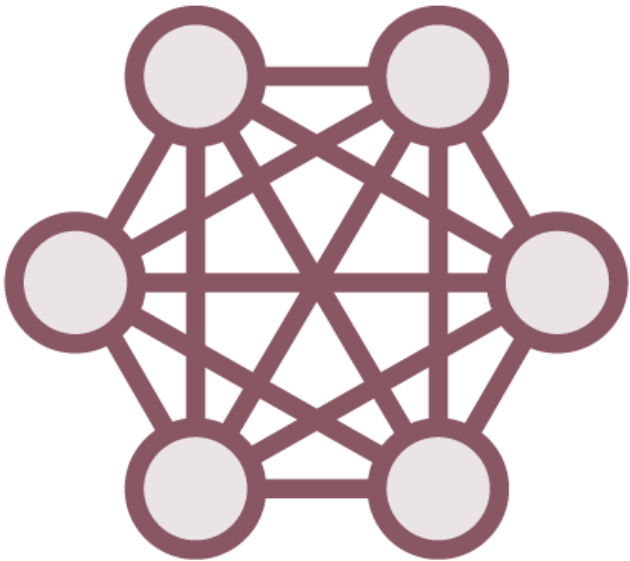
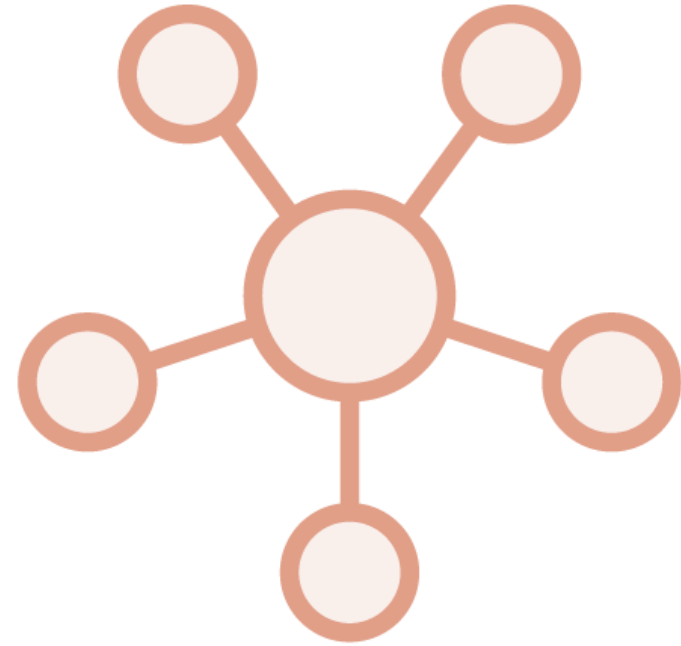
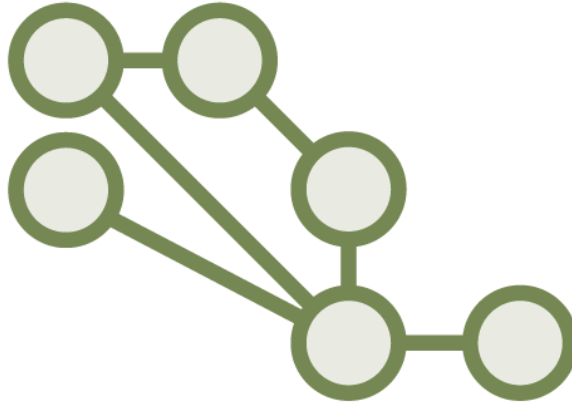
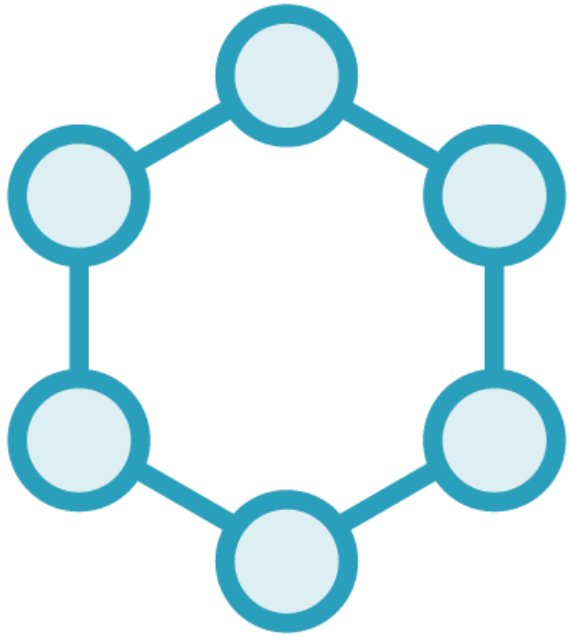


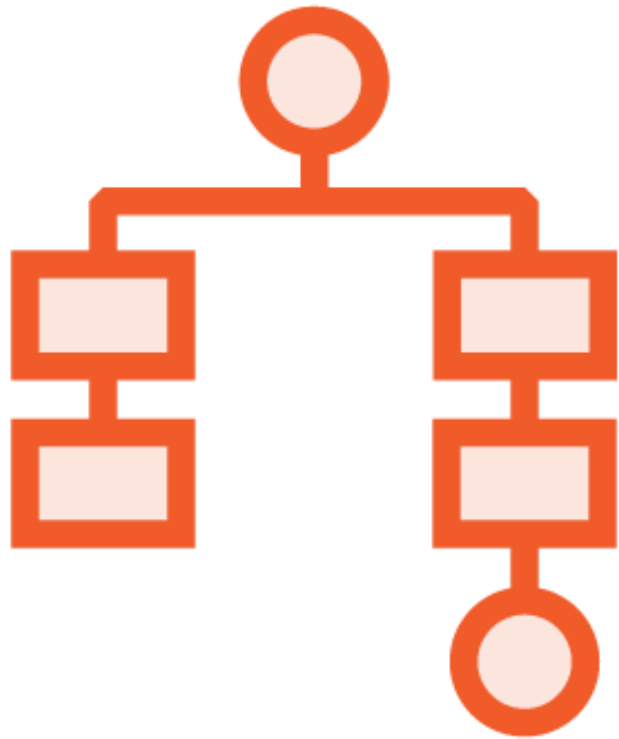
What is a query plan?

A query plan (or query execution plan) is an ordered set of steps used to access data in a SQL relational database management system.

- Wikipedia







Query execution plan – important stages

- Query parsing
- Query optimization
- Query execution
- Query plan cached

Why analyze query plans?

- Identify expensive operators
- Understand the data flow
- Diagnose blocking operators
- Determine state of statistics
- Classify resource consumption



Demo



Execution plans

- Graphical
- Textual
- XML



How to analyze query plans?

- Profiler (old and famous)
- Trace (deprecated)
- Extended Events (this module)
- Query Store (module 4)



What is SQL Profiler?

SQL Profiler is graphical user interface to SQL Trace to monitor various events running on a SQL Server instance.





Why not use SQL Profiler?

- Resource intensive

What are SQL Traces?

SQL traces gather events at the instance level and report them efficiently to users. SQL Profiler can create traces.





Why not use SQL Traces?

- Deprecated



Demo



SQL Profiler



Demo



SQL Traces



What are Extended Events?

Extended events are a lightweight monitoring tool which collects data about inner operations of SQL Server and helps troubleshoot performance problems.





Why use Extended Events?

- Lightweight on resources
- Extensive server events
- Portable

Demo



Extended Events



Summary



Extended events are the most efficient way to analyze query plans for SQL Server

