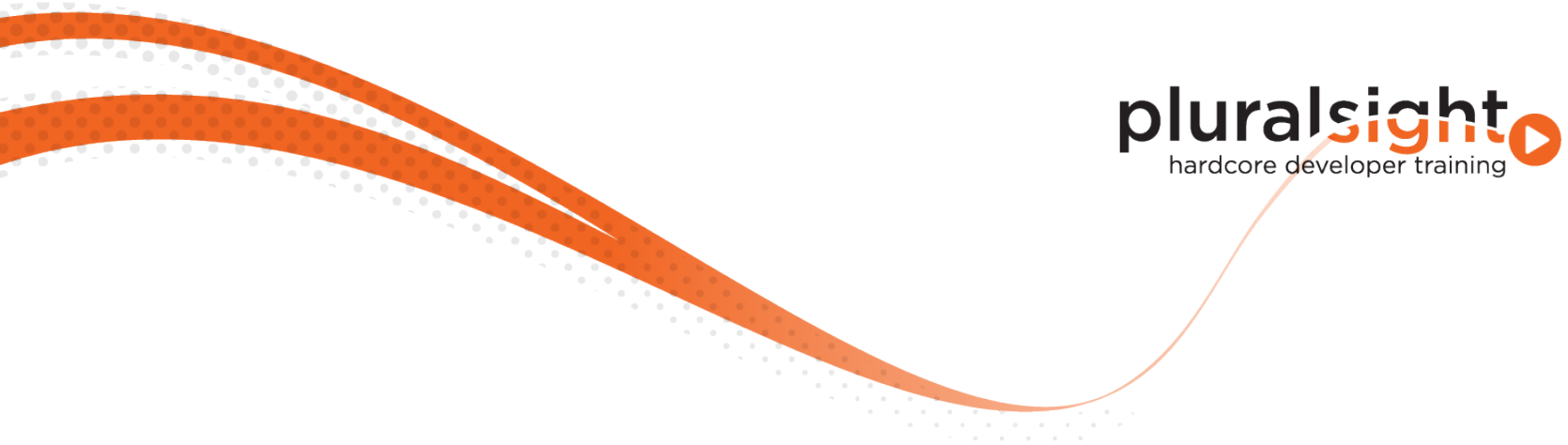
SQL Server: Performance Troubleshooting Using Wait Statistics

Module 1: Introduction

Paul S. Randal
Paul@SQLskills.com



pluralsight
hardcore developer training

Introduction

- **This course is applicable to all versions from SQL Server 2005 onwards**
- **Common requests to troubleshoot performance**
  - 'It feels like the server is running slower than it was'
  - 'Things are slower and we think it's the database'
  - The classic: 'We didn't change anything…'
- **Where do you start when troubleshooting poor performance?**
  - Hardware?
  - I/O subsystem?
  - Indexing?
  - Application?
- **Often people waste time flailing around with no clear direction**

Where To Start Troubleshooting

- **SQL Server keeps track of what is delaying query execution**
  - Contention for resources
- **Ask SQL Server where the contention is**
  - Provides an excellent starting point for performance troubleshooting
  - Combine with other sources of information
    - DMVs
    - Performance counters
    - Query plan analysis
- **Known as the Waits and Queues performance tuning methodology**
  - Contention = a thread waits for a queue (i.e. a resource to become available)

- **How to interpret the data?**

Doctor, Doctor… My Knee Hurts!

- **Three people go to the doctor complaining that their knee hurts**
  - Sounds like the start of a bad joke…

- **Patient 1 symptom: left knee hurts**
  - Root cause: right foot injury
- **Patient 2 symptom: left knee hurts**
  - Root cause: thyroid problem
- **Patient 3 symptom: left knee hurts**
  - Root cause: left knee injury

- **Three scenarios with identical symptoms, but different root causes**
  - Performance troubleshooting is the same!

Don’t Assume the Symptom is the Root Cause

- **Performance troubleshooting is not an exact science**
  - The same symptoms can result from many root causes

- **For example, how many different things could cause I/O latencies?**
  - Overloaded/incorrectly-configured I/O subsystem
  - Synchronous I/O subsystem mirroring
  - Buffer pool memory pressure
    - From plan cache bloat
    - From external Windows pressure
    - From an ad-hoc query
    - Inefficient query plan
  - Network latency
  - And more…

Interpreting the Data

- **Don't do 'knee-jerk' performance tuning**
  - Work through the data to see what may be the root cause
  - You'll end up spending less time overall

- **Proficiency in using wait statistics data comes from:**
  - Retrieving the data correctly
  - Understanding what common wait types mean
  - Recognizing patterns
  - Avoiding inappropriate Internet advice
  - Practice!

- **Even better is to have a series of snapshots of wait statistics over time**
  - Allows identification of changes and the time of the change
  - Allows trending

Course Structure

- Module 2: SQL Server Threading Model
- Module 3: Waits
- Module 4: Latches and Spinlocks
- Module 5: Troubleshooting Patterns
- Module 6: Summary