Deadlocking for Mere Mortals

Jonathan Kehayias
Senior Consultant, SQLskills.com
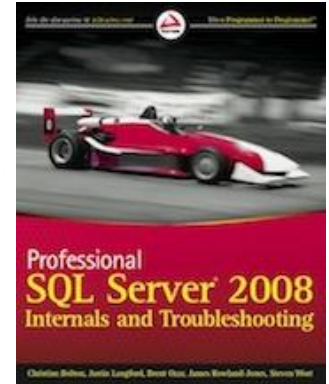
5/18/2011

Author/Instructor: Jonathan Kehayias

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

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

- Contributing Author: SQL Server 2008 Internals and Troubleshooting
- Author: SQL Server; Common Problems and Practical Troubleshooting
- Microsoft Certified Master: SQL Server 2008
- SQL Server MVP since October 2008
- Author of Using Extended Events whitepaper on MSDN
- Developer Extended Events Manager SSMS Addin for SQL Server 2008 Open Source project on Codeplex
- Member of the Tampa SQL Server User Group and PASS

Everything you need to know to prevent Deadlocks

Use WITH(NOLOCK)

for all SELECT statements

Questions?

Please Complete the Evaluation Form

Pick up your evaluation form:

- In each presentation room

Drop off your completed form

- Near the exit of each presentation room
- At the registration area

Presented by Dell

Everything you **DON'T** need to know to prevent Deadlocks

Use WITH(NOLOCK)
for all SELECT statements

This is not the Answer!

Overview

- What is a Deadlock
- Collecting Deadlock Graphs
- Anatomy of a Deadlock
- Reading Deadlock Graphs
- Resolving Deadlocks
- Questions

What is a Deadlock

- A condition when two or more processes are holding locks on resources and are each waiting for the other to release it's locks to progress, or where two or more processes are waiting for locks on resources in a circular chain

Process A → Resource 1 ← Process B

Process A → Resource 2 ← Process B

SQL Server and Deadlocks

- Lock Monitor Thread

- Error 1205

- Victim Selection
  - Deadlock Priority
  - Cost of Rollback

Collecting Deadlock Graphs

- Trace Flags (1222, 1205, 1204)
- SQL Trace / Profiler
- Event Notifications / WMI
- Extended Events

Anatomy of a Deadlock
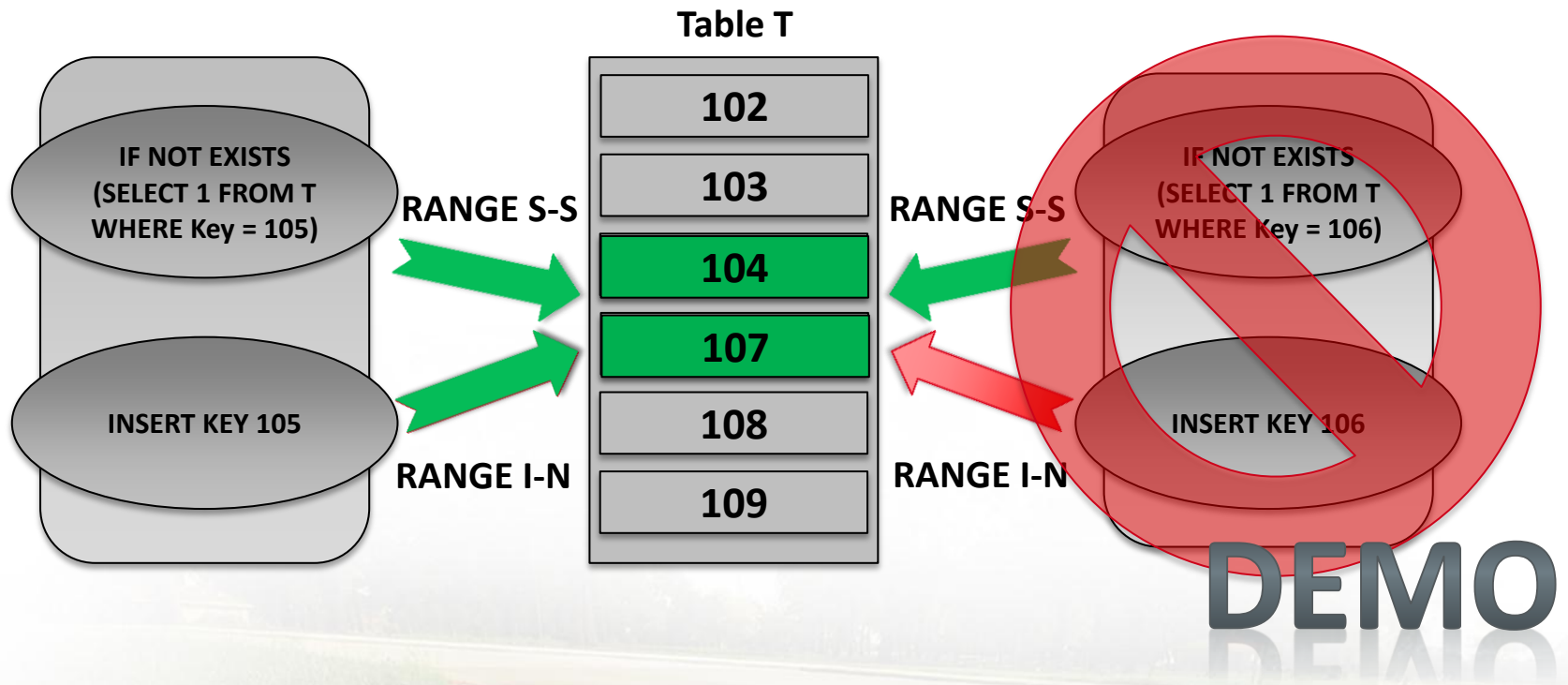
- Bookmark Lookup Deadlock

Reading Deadlock Graphs

- deadlock victim
- process-list
  - process id, waitresource, spid, priority, isolationlevel, inputbuf
- resource-list
  - owner-list, waiter-list, objectname, indexname

Anatomy of a Deadlock

- Serializable Lock Conversion Deadlock

Table T

IF NOT EXISTS (SELECT 1 FROM T WHERE Key = 105)
INSERT KEY 105

RANGE S-S

RANGE I-N

102
103
104
107
108
109

RANGE S-S

RANGE I-N

IF NOT EXISTS (SELECT 1 FROM T WHERE Key = 106)
INSERT KEY 106

DEMO

Resolving Deadlocks

Just use WITH(NOLOCK)
for every SELECT

This is not the Answer!

Resolving Deadlocks

- Change indexing to cover queries
- Enable Read Committed Snapshot Isolation
  - Writers won't block readers
  - Reads won't block writers
- Change Isolation Level
- Use locking hints to force specific lock types to prevent lock conversion
- Change application code to handle 1205 errors, log them and then resubmit the transaction to SQL Server again

Review

- What is a Deadlock
- Collecting Deadlock Graphs
- Anatomy of a Deadlock
- Reading Deadlock Graphs
- Resolving Deadlocks
- Questions

QUESTIONS?

Resources

- Bart Duncan's Blog
  http://blogs.msdn.com/b/bartd/archive/tags/sql+deadlocks/

- My Blog Posts
  http://sqlblog.com/blogs/jonathan_kehayias/archive/tags/Deadlock/default.aspx

- My Articles on SQL Server Central
  http://www.sqlservercentral.com/Authors/Articles/Jonathan_Kehayias/244648/

- My Videos on SQL Share
  http://www.sqlshare.com/Search.aspx?terms=deadlock

Please Complete the Evaluation Form

Pick up your evaluation form:

- In each presentation room

Drop off your completed form

- Near the exit of each presentation room
- At the registration area

Presented by Dell

THANK YOU!

Presented by Dell

For attending this session and
PASS SQLRally Orlando, Florida