




Paul Nielsen

- SQL Server 2000 Bible
- SQL Server 2005 Bible
- NORDic O/R DBMS in Action

paul@SQLServerBible.com
www.SQLServerBible.com

Colorado PASSCamp 2006

Colorado PASSCamp 2006

Performance Decisions

Paul Nielsen
www.SQLServerBible.com



A myriad of Performance Ideas, tips and tricks out there...

avoid Joins

Indexing

Raid5 vs SAN

Update Cursors

DeNormalize

Run the Wizard

Colorado PASSCamp 2006

Optimization Theory

- Inherent dependencies
- a framework for performance based on these dependencies.

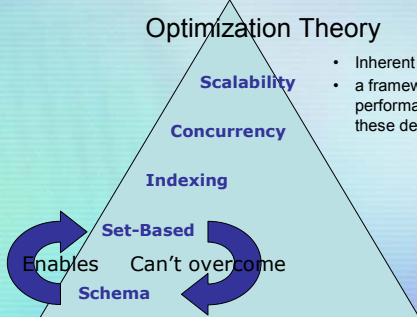
Scalability

Concurrency

Indexing

Set-Based Schema


Enables Can't overcome



Colorado PASSCamp 2006

Key Performance Decisions

- Schema
- Set-based Code
- Indexing
- Concurrency
- Scalability

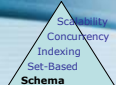


Colorado PASSCamp 2006

Performance Decision: Schema


Avoid Over-Complexity

- Normalization as Grammar
- Generalization
- Evaluate Multiple Normalized Solutions



Colorado PASSCamp 2006

Performance Decision: Schema

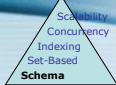


Select Keys Carefully

- Natural Primary Key
 - Pros: FK Saves Joins, Human Understandable
 - Cons: Human Understandable, Page splits ?, complex FKs
- Identity Primary Keys
 - Pros: Fast, permanent
 - Cons: FKs require more joins
- GUID Primary Keys NewSequentialID()
 - Pros: Replication, more unique
 - Cons: Page splits, fewer rows on data page
- Composite Primary Keys
 - Pros: Works well with Natural-only Schemes, skip levels of joins
 - Cons: Complex FKs

Colorado PASSCamp 2006

Performance Decision: Schema

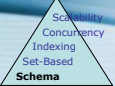


Handle Optional Data

- The Problem of Null
 - Vague meaning
- Possible Solutions
 - Null Row
 - Null Value
 - Surrogate Nulls

Colorado PASSCamp 2006

Performance Decision: Schema




Enforce an Abstraction Layer

- Improve Performance
- Long-Term Extensibility
 - App Language Lifecycle
 - Database Lifecycle
- Physical Location ?
 - Reports?

Colorado PASSCamp 2006

Performance Decision: Set-Based




Enable Set-Based Code

- Clean Schema easy to understand and query
- Set-Based can't overcome Poor Schema
 - Bucket Schemes
 - Overly-Complex Schema
 - Null-Row Schema

Colorado PASSCamp 2006

Performance Decision: Set-Based




Use Set-Based Code

<u>to solve:</u>	<u>use:</u>
• Complex B-Logic	Query, Subqueries, CTEs
• Dynamic Code Iteration	Cursor
• Denormalizing a List	Multiple Assignment Variable, Cursor
• Crosstab Query	Pivot, Case Expression
• Navigating Hierarchy	UDF, CTE
• Cumulative Totals	Cursor

Colorado PASSCamp 2006

Performance Decision: Set-Based




Refactor Complex Logic Cursors

- Power of Time Optimization
- Query Technique
 - Case Expression
 - Embed Logic in UDF
 - Multiple Queries

Colorado PASSCamp 2006

Performance Decision: Indexing




Enable Indexing

- Enables Indexing
 - Clean Schema
 - Abstraction Layer
 - Set-Based Code
- Indexes can't overcome
 - Iterative Code
 - Difficult to Index without known set of data access
 - Overly Complex Schema nearly impossible to plan indexes

Colorado PASSCamp 2006

Performance Decision: Indexing




Use Cluster Indexes

- Avoid BookMark Lookups
- Group Multiple Rows on same page(s)

Colorado PASSCamp 2006

Performance Decision: Indexing

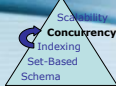


Use a Comprehensive Strategy

- Analyze all access by columns
- Design Index for reuse
- NC indexes leaf level includes Clustered Index
- Covering Indexes are *FAST!*

Colorado PASSCamp 2006

Performance Decision: Concurrency

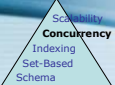


Enable Concurrency

- Enables Concurrency
 - Clean Schema
 - Abstraction Layer
 - Set-Based Code
 - High Performance Indexing
- Locking and Blocking fixes can't overcome
 - Long Transactions due to poor indexes, iterative code, or complex Schema
 - NoLock is not the answer

Colorado PASSCamp 2006

Performance Decision: Concurrency



Tighten Transactions

- Tight Sproc Logical Transactions
- Logic in Sproc not Trigger

Colorado PASSCamp 2006

Performance Decision: Concurrency

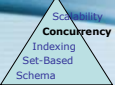


Appropriate Isolation Level

- Read Committed probably best (default)
- Lower isolation carefully

Colorado PASSCamp 2006

Performance Decision: Concurrency

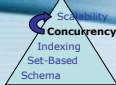


Consider Snapshot Isolation

- Writers block Writers
- Writers don't block Readers
- Enabled at Database Level
- Heavy use of TempDB

Colorado PASSCamp 2006

Performance Decision: Scalability

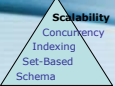


Enable Scalability

- Schema, Queries, Indexes, and Concurrency are Scalable
- Adv Scalable options can't overcome lack of good design

Colorado PASSCamp 2006

Performance Decision: Scalability

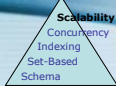


Partition Tables

- Divide a Table to filegroups based on data boundaries
- Improve Parallel Operations
- Maintain Smaller Partitions
- Switch & Rolling Partitions

Colorado PASSCamp 2006

Performance Decision: Scalability

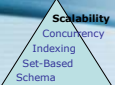


Use Index Views

- Denormalized Materialized View
- Fast for complex joins of large tables
- Minimal overhead when updating

Colorado PASSCamp 2006

Performance Decision: Scalability

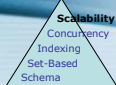


Use Service Broker

- High Payload Work Queue
- Load Balance over time

Colorado PASSCamp 2006

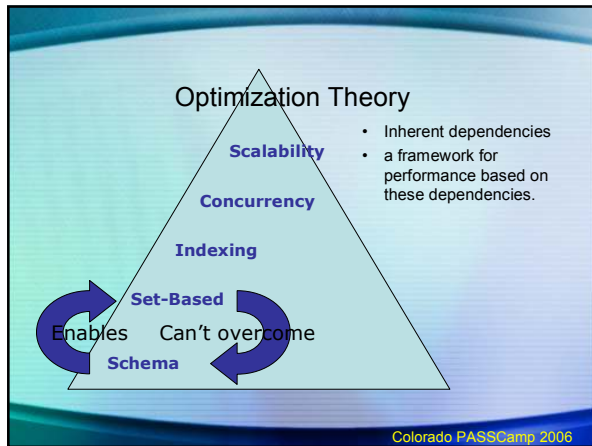
Performance Decision: Scalability



Dedicate TempDB Drive

- C: System, SQL Server
- D: Swap File
- E: TempDB, Master, Model, MSDB
- F: TempDB TLog
- G: Transaction Log
- H:... Data FileGroups

Colorado PASSCamp 2006



Performance Decision: Enjoy PASS Camp

Four Challenges

- Make a friend who will meet you at a PASS Chapter
- Go to a session outside your comfort zone
- Ask a question
- Win something

Colorado PASSCamp 2006