The New Locking, Logging, And Recovery Architecture of Microsoft SQL Server 7.0

David Campbell
Microsoft Corporation, Redmond, WA, USA
davidc@microsoft.com

The Microsoft SQL Server storage engine was rearchitected to support row level locking in Version 7.0. This required significant changes throughout the store; from page update primitives, logging and recovery, to access-methods. One interesting engineering aspect of this project was the fact that we kept the system running during the entire transformation of the underlying architecture – a task we liken to transforming a zebra into a cheetah by transplanting one organ at a time. This talk will focus on how real-world pragmatics collided with academic purity and what solutions resulted.

A brief overview of the pre-existing locking, access methods, and recovery systems will be followed by a discussion of the new architecture and how we changed a strictly page based concurrency control and recovery system into a highly concurrent storage system that has significantly better performance and concurrency behavior than its predecessor.

We will discuss:

- How we originally implemented a standard key-range locking architecture as described in the literature, and then had to extend it to process deletes as updates to improve concurrency for a number of real-world workloads.
- How we implemented a multi-granular lock protocol for B-trees that includes both page and row granularity locking, and the benefits and complications that resulted.
- How we added a run-time cost-based optimization scheme to determine the appropriate locking granularity for each access method scan.
- How we changed allocation and index structural modifications from a transaction consistent policy to an action consistent policy where the changes are performed under short-term "system" transactions and how these system transactions are used in a multi-level recovery scheme.
- How we addressed the issue of log reservation for compensation logging of undo actions.
- How we maintained the behavior of prior releases such as minimal logging for bulk update operations including index creation.
- How we instrumented and tested the system to ensure it was ready for production use.

The talk will conclude with a presentation of the lessons learned from this effort.

Proceedings of the 25th VLDB Conference, Edinburgh, Scotland, 1999.