

“Data Access Hacks and Shortcuts” Stephen Forte

May 22, 2009

[MS Techdays 2009] Summary “Data Access Hacks and Shortcuts” Stephen Forte

Not very interesting so I haven't a lot of notes. For details you should have a look at the conference slides.

Pass .NET collections to Stored Procedures

SQL Sever 2008 allows table valued variables. This in combination with ADO .Net 3.51 allows you to pass DataSets as variables to Stored Procedure. Moreover C# 3.0 can convert any collection, implementing *IEnumerable*, into a DataSet. So custom collections can be passed to stored procedures as variables.

Spying on your DB Server

LINQ will convert IQueryable to TSQL queries. You can spy on these queries by using following tools:

- SQL Server Profiler (<http://msdn.microsoft.com/en-us/library/ms187929.aspx>) allows you to record all incoming request on your DB Server.
- LINQ ObjectDumper (<http://msdn.microsoft.com/en-us/library/bb397968.aspx>) outputs the results of LINQ queries for testing purposes.

Over-normalization tackled with Views

If your DBA has gone overboard and created an over-normalized database it is better to create views then to have queries with lots of unions between tables. These views will deconstructs complex data and optimize lookups.

What about updates? You still can access your tables in the normal way.

Binding REST data to Silverlight

Use LINQ to ADO.NET Data Services on the Silverlight client leverages your investments in LINQ but requires you to be 100% asynchronous.

Reporting DB design Online transactions have different needs when accessing a DB compared to reporting. The first are optimized for inserts. The second focus on fast selects. Why are we trying to do this on the same DB?

It is better to create a reporting DB next to the traditional OLTP DB. Move aggregated data to your data warehouse that is denormalized for fast access to optimally support reporting.

How to keep both in sync? You can create ETL code to move form your OLTP DB to your data warehouse but it should be clear there always will be latency between what's online and what's available in the data warehouse. Allowing no latency will remove all advantages because every update is synchronized immediately and affecting the reporting part.

www.vanderbist.com