SQL 2012 Developer, Part 10 of 13: Partitions

Meet the expert: Frank Tillinghast is a senior consultant with MTOW Software Solutions. He is a Microsoft Certified Solution Developer and has been developing applications for over 15 years. Most of his time is spent consulting for companies nationwide with troubled projects or mentoring projects to successful completion. When he is not developing software or mentoring clients, Frank is teaching other developers. He has taught thousands of application developers how to create business solutions with Visual Studio .NET. VB.NET, ASP.NET, Visual C#, and SQL Server.

Prerequisites: This course assumes that you have a basic familiarity with the concept of relational databases and a basic understanding of what SQL Server is and the high-level tools in it, as well as how to create and manage objects using Management Studio. You should also have a basic understanding of how SQL Server implements security, including its authentication and authorization schemes, and how to assign permissions on securable objects to principals. You should know the fundamentals of Transact-SQL to write queries to retrieve and update data, and how to execute scripts using the query editor in Management Studio.
It will be very helpful, but not absolutely necessary, to have experience with .NET development using Visual Studio 2010 or 2012 for the portions of the course that deal with SQLCLR development. The .NET code included in this course is written in C#, but if you prefer Visual Basic it should be fairly easy to convert the code.

Runtime: 01:22:12

Course description: In this course we'll explore partitioning, and specifically how it's implemented using SQL Server 2012. Partitioning is a way to manage large amounts of data, and while certainly not a new concept, SQL Server has made improvements over the years that make your job as a database architect and developer easier. We'll start off by discussing the evolution of partitioning in SQL Server. You'll learn about partition keys, schemas and functions and how they affect tables, indexes, and the organization of your data. Once you have an understanding of the basics, we'll move on to discuss methods for querying partitions. As time goes on, it may be necessary to make adjustments to your partitions. Next you'll learn how to merge or split partitions, switch out partitions, and make modifications to partition schemes. The course then concludes with some tips on how performance tuning.

Course outline:

Partitioning
- Introduction
- Table-Based Partitioning
- Reasons For Partitioning
- SQL Server Partitioning
- SQL Server 6.5 and Earlier
- SQL Server 7.0
- SQL Server 2000
- SQL Server 2005/2008
- SQL Server 2012
- Range Partitions
- Partition Key
- Using Multiple Filegroups
- Index Partitioning
- Summary

Creating Partition Tables
- Introduction
- Creating Partitioned Tables
- LEFT and RIGHT Boundaries
- Datetime Values
- Demo: Partition Function
- Creating a Partition Scheme
- Schemes and Filegroups
- Demo: Adding Filegroups
- Demo: Partition Scheme
- Partitioning Tables
- Partitioned Index
- Demo: Partitioned Table
- Demo: Partitioned Index
- Summary

Querying Partitions
- Introduction
- Querying Partitions
- Using the PARTITION Function
- Demo: PARTITION Function
- Catalog Views
- sys.partition_functions
- sys.partition_range_values
- sys.partition_schemes
- sys.data_spaces
- Demo: Catalog Views
- Summary

Managing Partitions
- Introduction
- Managing Partitions
- Modifying Partitions
- Modifying a Partition Function
- Splitting Partitions
- Merging Partitions
- Switching Partitions
- Modifying a Partition Scheme
- Backing Up Partitions
- Performance Considerations
- Summary