

Foreword by Tobias Ternström
Lead Program Manager, Microsoft SQL Server Engine team

Microsoft® SQL Server® 2012 High-Performance T-SQL Using Window Functions



Itzik Ben-Gan



Microsoft® SQL Server® 2012 High-Performance T-SQL Using Window Functions

Apply powerful window functions in T-SQL—and increase the performance and speed of your queries

Optimize your queries—and obtain simple and elegant solutions to a variety of problems—using window functions in Transact-SQL. Led by T-SQL expert Itzik Ben-Gan, you'll learn how to apply calculations against sets of rows in a flexible, clear, and efficient manner. Ideal whether you're a database administrator or developer, this practical guide demonstrates how to use a range of T-SQL querying solutions to address common business tasks.

Discover how to:

- Go beyond traditional query approaches to express set calculations more efficiently
- Delve into ordered set functions such as rank, distribution, and offset
- Implement hypothetical set and inverse distribution functions in standard SQL
- Use strategies to improve sequencing, paging, filtering, and pivoting
- Increase query speed using partitioning, ordering, and coverage indexing
- Apply new optimization iterators such as Window Spool
- Handle common issues such as running totals, intervals, medians, and gaps



Get code samples on the web

Ready to download at
<http://go.microsoft.com/fwlink/?Linkid=246708>

For **system requirements**, see the *Introduction*.

microsoft.com/mspress

ISBN: 978-0-7356-5836-3



U.S.A. \$36.99

Canada \$38.99

[Recommended]

Databases/Microsoft SQL Server



About the Author

Itzik Ben-Gan, a Microsoft MVP for SQL Server since 1999, is cofounder of SolidQ, a company that provides consulting and training services for the entire Microsoft data platform. He writes numerous articles for *SQL Server Pro* magazine and speaks at industry events such as the Professional Association for SQL Server (PASS) and Microsoft TechEd.

DEVELOPER ROADMAP

Start Here!

- Beginner-level instruction
- Easy to follow explanations and examples
- Exercises to build your first projects



Step by Step

- For experienced developers learning a new topic
- Focus on fundamental techniques and tools
- Hands-on tutorial with practice files plus eBook



Developer Reference

- Professional developers; intermediate to advanced
- Expertly covers essential topics and techniques
- Features extensive, adaptable code examples



Focused Topics

- For programmers who develop complex or advanced solutions
- Specialized topics; narrow focus; deep coverage
- Features extensive, adaptable code examples



Microsoft®

Microsoft SQL Server 2012 High-Performance T-SQL Using Window Functions

Table of Contents

Cover

Title Page

Copyright Page

Contents at a Glance

Foreword

Introduction

SQL Windowing

- Background of Window Functions

 - Window Functions Described

 - Set-Based vs. Iterative/Cursor Programming

 - Drawbacks of Alternatives to Window Functions

- A Glimpse of Solutions Using Window Functions

- Elements of Window Functions

 - Partitioning

 - Ordering

 - Framing

- Query Elements Supporting Window Functions

 - Logical Query Processing

 - Clauses Supporting Window Functions

 - Circumventing the Limitations

Table of Contents

Potential for Additional Filters

Reuse of Window Definitions

Summary

A Detailed Look at Window Functions

Window Aggregate Functions

Window Aggregate Functions Described

Supported Windowing Elements

Further Filtering Ideas

Distinct Aggregates

Nested Aggregates

Ranking Functions

Supported Windowing Elements

ROW_NUMBER

NTILE

RANK and DENSE_RANK

Distribution Functions

Supported Windowing Elements

Rank Distribution Functions

Inverse Distribution Functions

Offset Functions

Supported Windowing Elements

LAG and LEAD

FIRST_VALUE, LAST_VALUE, and NTH_VALUE

Summary

Ordered Set Functions

Hypothetical Set Functions

RANK

DENSE_RANK

Table of Contents

PERCENT_RANK

CUME_DIST

General Solution

Inverse Distribution Functions

Offset Functions

String Concatenation

Summary

Optimization of Window Functions

Sample Data

Indexing Guidelines

POC Index

Backward Scans

Columnstore Indexes

Ranking Functions

ROW_NUMBER

NTILE

RANK and DENSE_RANK

Improved Parallelism with APPLY

Aggregate and Offset Functions

Without Ordering and Framing

With Ordering and Framing

Distribution Functions

Rank Distribution Functions

Inverse Distribution Functions

Summary

T-SQL Solutions Using Window Functions

Virtual Auxiliary Table of Numbers

Table of Contents

Sequences of Date and Time Values

Sequences of Keys

Update a Column with Unique Values

Applying a Range of Sequence Values

Paging

Removing Duplicates

Pivoting

TOP N per Group

Mode

Running Totals

Set-Based Solution Using Window Functions

Set-Based Solutions Using Subqueries or Joins

Cursor-Based Solution

CLR-Based Solution

Nested Iterations

Multirow UPDATE with Variables

Performance Benchmark

Max Concurrent Intervals

Traditional Set-Based Solution

Cursor-Based Solution

Solutions Based on Window Functions

Performance Benchmark

Packing Intervals

Traditional Set-Based Solution

Solutions Based on Window Functions

Gaps and Islands

Gaps

Islands

Table of Contents

Median

Conditional Aggregate

Sorting Hierarchies

Summary

Index

About the Author

Survey