GroupBy

A practical look at Temporal Tables

Randolph West

Data Platform Time Lord

GROUPBY

CODE OF CONDUCT

The Quick Version

We are dedicated to a harassment-free experience for everyone, regardless of who you are and what makes you *you.* We recognize the right of any individual to attend and participate. Anyone. This is included but not limited to gender identity and expression, sexual orientation, disability, physical appearance, body size, race, religion, or any other classification, affiliation, or label.

We do not tolerate harassment in any form. For the duration of your engagement with GroupBy and its programs, you are expected to act appropriately and to adhere to this Code of Conduct. This includes conduct in-person and online, at the conference itself, as well as any non-conference programs that may include participants: including talks, workshops, parties, on social media, and other online forums. GroupBy participants violating these rules may be sanctioned or expelled without a refund (if that applies) at the discretion of the conference organizers.

You can review the full policy at: **GroupBy.org/Code-of-Conduct**

GROUPBY 2021 | **OCT 26-27**

Core Sponsors

redgate

solarwinds

Media Sponsors

redgate

solarwinds

SSG
SQL Solutions Group

Virtual Group Sponsors

minionware

GroupBy

Who is Randolph West?

Author

Actor

Consultant

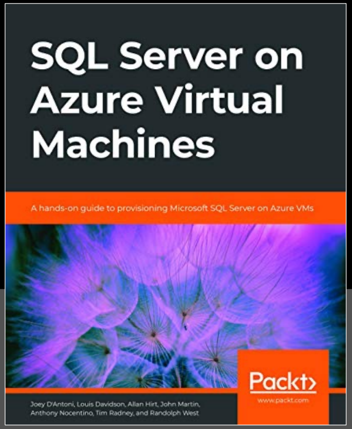
C#

SQL Server

Chocolate

Pronouns: *they/them*

Email: *r@ndolph.ca*

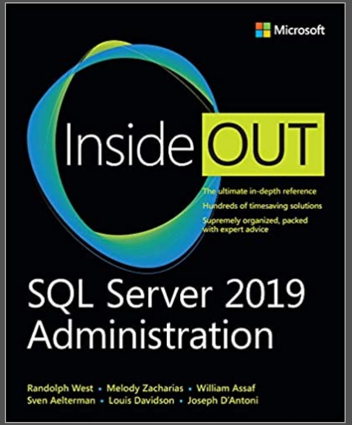


SQL Server on Azure Virtual Machines

A hands-on guide to provisioning Microsoft SQL Server on Azure VMs

Packt>
www.packt.com

Joey D'Antoni, Louis Davidson, Allan Hirt, John Martin, Anthony Nocentino, Tim Radney, and Randolph West



Microsoft

Inside OUT

The ultimate in-depth reference
Hundreds of timesaving solutions
Supremely organized, packed with expert advice

SQL Server 2019 Administration

Randolph West • Melody Zacharias • William Assaf
Sven Aelterman • Louis Davidson • Joseph D'Antoni

Microsoft

Inside OUT

The ultimate, in-depth reference
Hundreds of timesaving solutions
Supremely organized, packed with expert advice

SQL Server 2017 Administration

William Assaf • Randolph West • Sven Aelterman • Mindy Curnutt
Foreword by Patrick LeBlanc, Microsoft

MVP Microsoft® Most Valuable Professional

Azure Data Community

DATA SATURDAYS

BORNSQL.ca

Types of system-versioned tables

- Temporal tables (our main focus)
- Ledger tables (Azure SQL only)

System-versioned temporal tables

Keep a full history of data changes

System-versioned temporal tables

Allow easy point-in-time analysis

System-versioned temporal tables

Period of validity for each row is managed by the database engine

System-versioned temporal tables

- two `PERIOD` columns
- `datetime2` data type
- record validity period per row whenever a row is modified

System-versioned temporal tables

- reference a *history* table
- with a mirrored schema
- store previous version of the row whenever a row is modified

When should I use *temporal* tables?

Audit all data changes and perform data forensics when necessary

This is a marketing slide

When should I use *ledger* tables?

Provide evidence of tampering during a forensic investigation because hashes of database state are stored offsite

You can use temporal and ledger tables together, with some limitations

Why temporal?

Reconstruct the state of the data at any time in the past

Why temporal?

Calculate trends over time

Why temporal?

Maintain slowly-changing dimensions for decision-support applications

Why temporal?

Recover from accidental data changes and application errors

This is the killer feature

Why temporal?

HIDDEN period columns provide backward compatibility

Why temporal?

History tables can be created manually, or by the database engine

Limitations and considerations

Primary Key in the current table,
no primary key in history table
(no other constraints either)

Limitations and considerations

History table must be stored in the same database as current table

Limitations and considerations

History table is PAGE compressed by default

If you create the history table manually, you must enable compression

Limitations and considerations

Blob data incurs significant storage and performance penalties: `(n)varchar(max)`, `varbinary(max)`, `(n)text`, and `image`

Limitations and considerations

With system versioning, you cannot:

- TRUNCATE TABLE
- Modify the history table directly

Limitations and considerations

Read them all:

https://docs.microsoft.com/sql/relational-databases/tables/temporal-table-considerations-and-limitations

Managing historical data retention

- Stretch database
- Table partitioning
- Custom cleanup
- Retention policy (2017+ and Azure)

https://docs.microsoft.com/sql/relational-databases/tables/ manage-retention-of-historical-data-in-system-versioned-temporal-tables

Memory-optimized temporal tables

- Current table in-memory
- History table on disk
- Internal staging table in-memory
- Works on Standard Edition

Please don’t use in-memory tables on Standard Edition

Demo