Tabular Models: Easier & Faster Than Cubes; Really?

the good, the bad, the ugly & the beautiful
paul turley

Mentor, SolidQ
SQL Server MVP
Author
BI Solution Architect

paul.turley@SolidQ.com
www.SqlServerBiBlog.com
Tabular Models: Easier & Faster Than Cubes; Really?

The future of data analysis is the new, in-memory xVelocity Tabular analytic engine but is this new product ready for enterprise solutions? Is it easier to build Tabular models than multidimensional cubes in SQL Server 2012? Maybe... Depends on what you need to do. Is it faster? Heck, yea. Taking off the Microsoft BI marketing hat, let’s take a tabular journey together and learn some lessons from the first generation of real, enterprise-scale solutions: The Good, the Bad, the Ugly and the Beautiful.
Marketing

real-world practice
<table>
<thead>
<tr>
<th>Semantic Modeling Options</th>
<th>Scale</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>PowerPivot - Desktop</strong></td>
<td>x 1 million rows</td>
</tr>
<tr>
<td>• Simple, fast development</td>
<td></td>
</tr>
<tr>
<td>• May be promoted to SharePoint or BSM Tabular</td>
<td></td>
</tr>
<tr>
<td><strong>PowerPivot - SharePoint</strong></td>
<td>x 10 million rows</td>
</tr>
<tr>
<td>• Collaborative</td>
<td></td>
</tr>
<tr>
<td>• Server scale</td>
<td></td>
</tr>
<tr>
<td><strong>BISM Tabular Model</strong></td>
<td>x 100 million rows x 10 billion rows</td>
</tr>
<tr>
<td>• Partitions for scale</td>
<td></td>
</tr>
<tr>
<td>• Admin: processing options, security control</td>
<td></td>
</tr>
<tr>
<td><strong>BISM Multidimensional (SSAS OLAP)</strong></td>
<td>millions, billions</td>
</tr>
<tr>
<td>• Optimized for dimensional structures &amp; sum-able aggregates</td>
<td></td>
</tr>
<tr>
<td>• Mature features &amp; scripting flexibility</td>
<td></td>
</tr>
</tbody>
</table>
terminology

- VertiPaq
- VertiScan
- xVelocity
- Column Store
- Tabular Model
- PowerPivot
- DAX
the Good, Bad, the Ugly

... and the Beautiful
the good

VertiPaq engine is stable & reliable
performs very well in most scenarios
in many ways is much less complex than multidimensional SSAS
can be easier & faster to implement
basic DAX is easy to learn
the bad

PowerPivot is a v.2 product
tabular SSAS is a v.1 product
design patterns & best practices are still emerging
one model.bim file = one developer working at a time
support community is thriving but still small
the DAX editor in PowerPivot & SSDT is quirky
the ugly

the v.1 SSDT designer can be cumbersome
all design work is performed while connected to the workspace database
all changes are written & updated, one property at a time
not all SSAS features are currently supported
the beautiful

• in-memory, tabular model technology is the promise of the future
• many useful & valuable features are supported today
• PowerPivot models upgrade seamlessly to tabular model
• tabular is SSAS!
• features not implemented now will be available in future builds
• clients that support SSAS multidimensional support tabular
• tabular understands MDX & DAX queries
tabular data sources

• all standard sources: all relational databases, text files, Excel, SSAS
• data feeds: OData, ATOM Feed, SSRS reports
• clipboard
• anything that looks like a table
• Excel linked tables not supported in SSAS tabular model
• Azure Marketplace & Suggest Related
tabular models are not a replacement for data quality governance & ETL
challenges & gotchas

• workspace database updates can be time-consuming
  • renaming objects
  • setting properties
• workspace database can be too big for development machine
• workspace database stability issues
• diagram view usability issues with large models
• DAX editor auto-completion creates syntax errors
• SSAS may expose two copies of a table
defining model metadata

• important to be thorough
• easy to miss steps
• iteration is common

  • rename objects
  • hide from client tools
  • change data types
  • set formatting
  • set the sort by column
  • date table
  • column categories

model & deployment management

• use partitions to manage development working set
• workspace database should be a local instance
• process one or a few partitions to populate workspace database
• deploy to de
target and workspace databases
naming conventions & friendly names

A semantic model is the buffer zone between enterprise data and a self-service user interface. Convert database object names to user-familiar names & structures. Column & field names:

In large models, use a view to map data source columns to friendly names.
what is DAX?

Google:

Bing:

Wikipedia:

“Wikipedia does not have an article with this exact name.”
(as of 28 Nov 2012)

...so I posted an article:

“Data Analysis Expressions, or DAX, is the native formula and query language for Microsoft PowerPivot and SQL Server Analysis Services Tabular models. DAX includes some of the functions that are used in Excel formulas, and additional functions that are designed to work with relational data and perform dynamic aggregation.”
“DAX is simple but DAX is not always easy”
-Alberto Ferrari; author, BI expert

Complete DAX documentation:
select * from $system.mdschema_functions where origin = 3 or origin = 4
review: best practices & learnings

• install a local instance of SSAS tabular on development machine & use for workspace database
• map column names in views
• name measure base columns differently than measures
• create measures & hide measure base columns
• hide everything you don’t want to expose to users – keys, utility columns & measure base columns
• copy DAX expression before making changes
• use partitions to minimize model data size
resources

• my blog: SqlServerBiBlog.com
• TechNet Wiki (SQL Server / Analysis Services)
• Training: SolidQ.com/training
• Books: