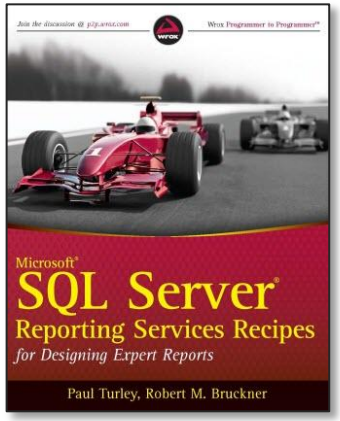
MDX Essentails

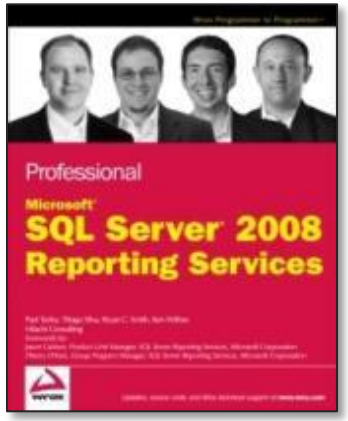
Paul Turley
Mentor, SQL Server MVP

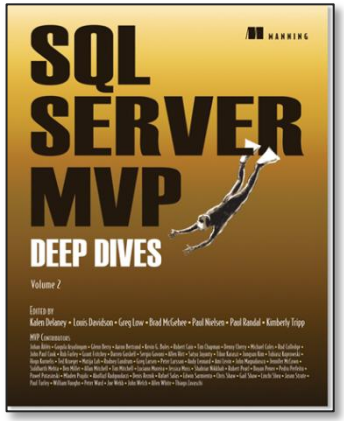
Introduction

Paul Turley
Mentor, SQL Server MVP

SqlServerBiBlog.com

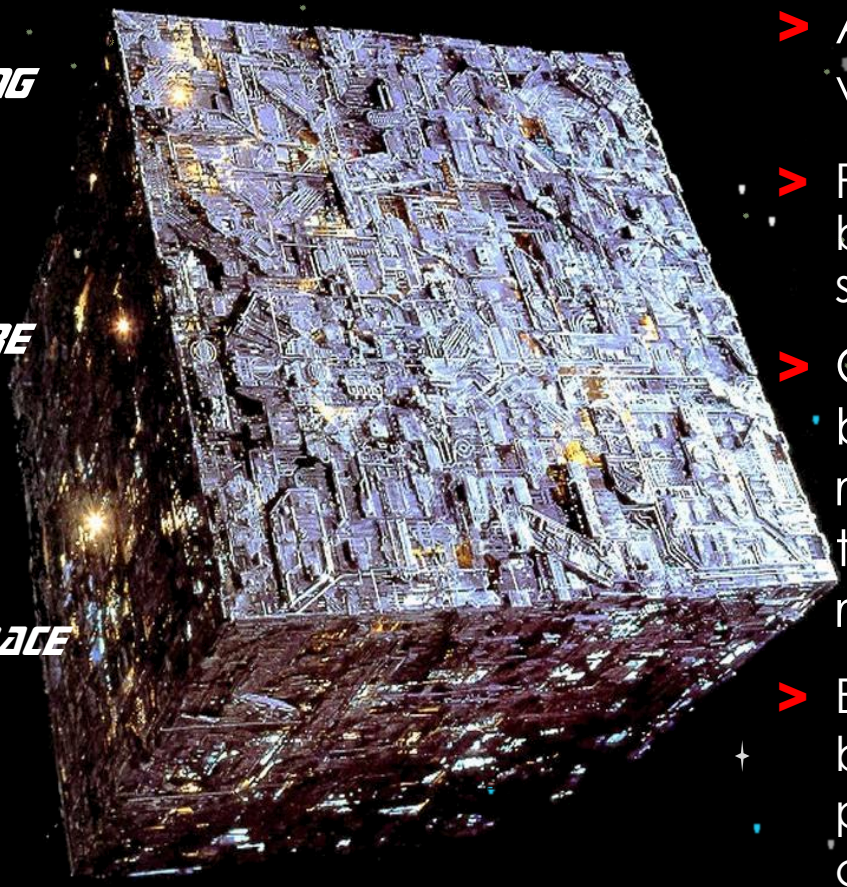






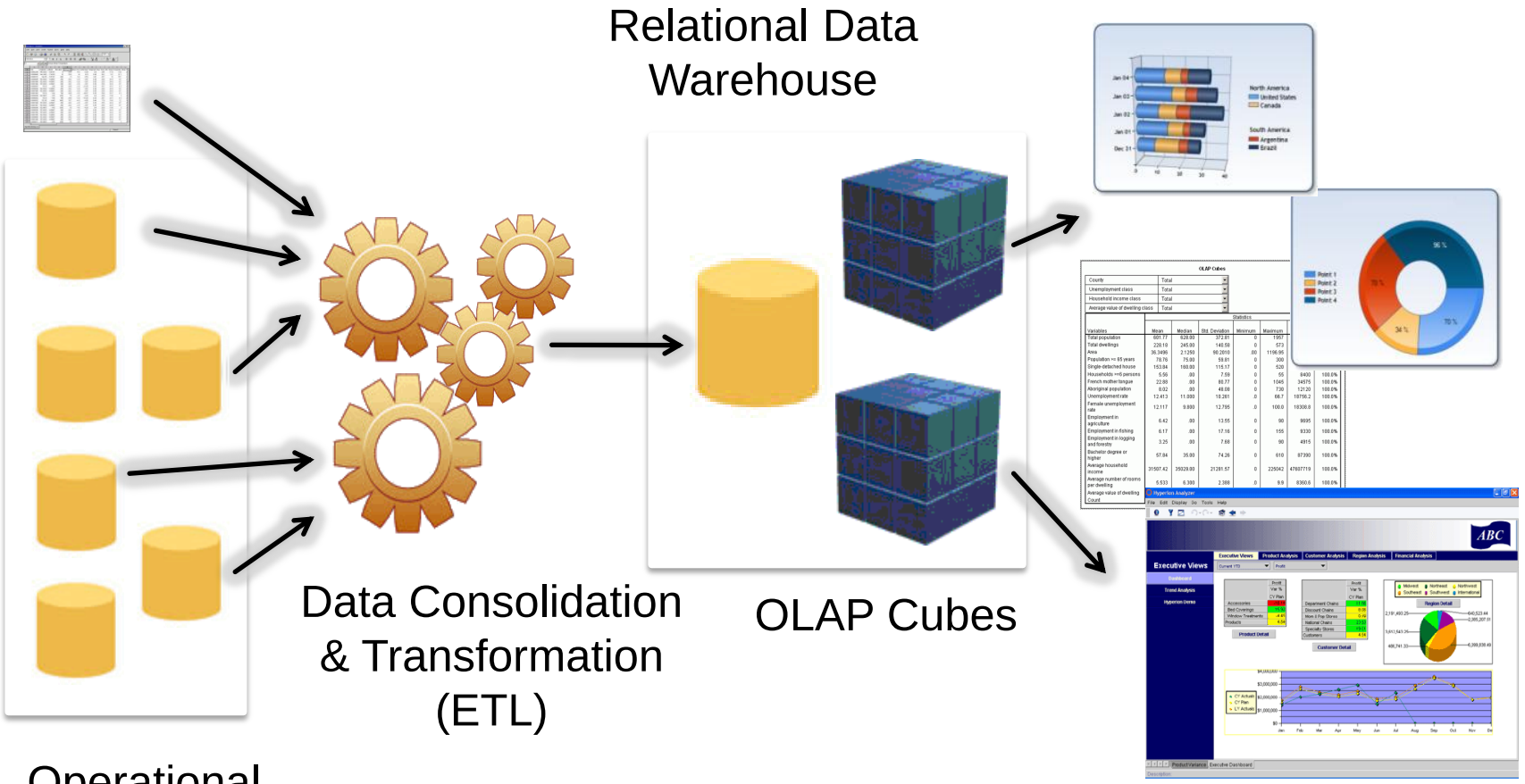
What Can You Do with a Cube?

- Destroy anything in its path
- Assimilate entire civilizations
- Create a mega race of neo-humanoid androids with a single collective consciousness



- Aggregate very large volumes of data
- Present browse-able business information for self-service reporting
- Create high-value business reports that render in a fraction of the time of a relational data source
- Encapsulate complex business rules into predefined hierarchies, calculations, business measures and KPIs

The Business Data Continuum



MDX

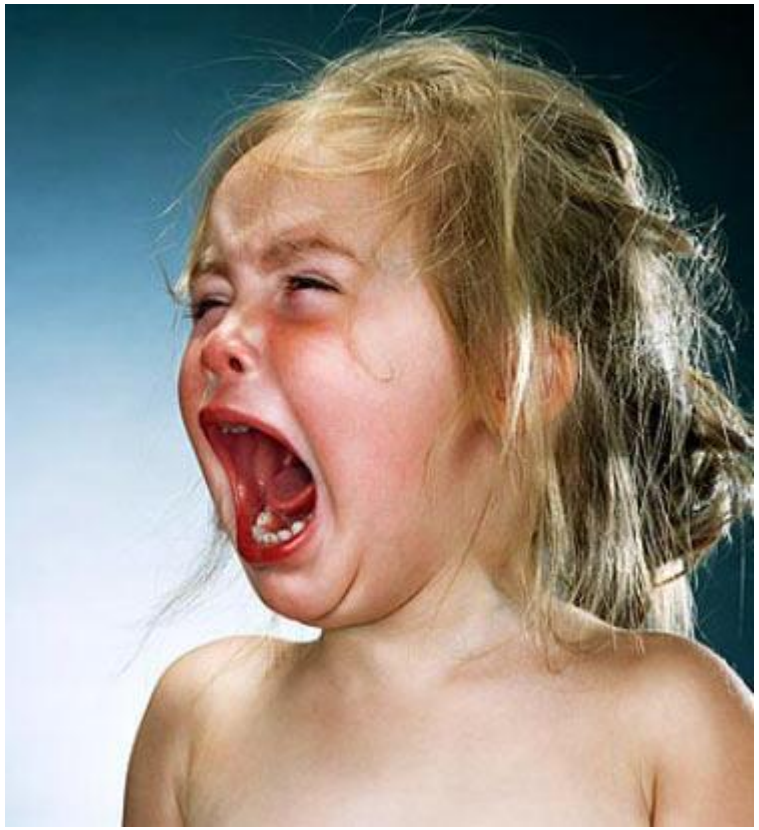
M ulti
D imensional
e**X** pressions

- A query language developed & used by Microsoft & other vendors for OLAP data products
- Part of the OLE DB for OLAP 1997 specification held by Microsoft (not ANSI or ISO)
- Specification was written mainly by Mosha Pasumansky from Microsoft, originally from Panorama

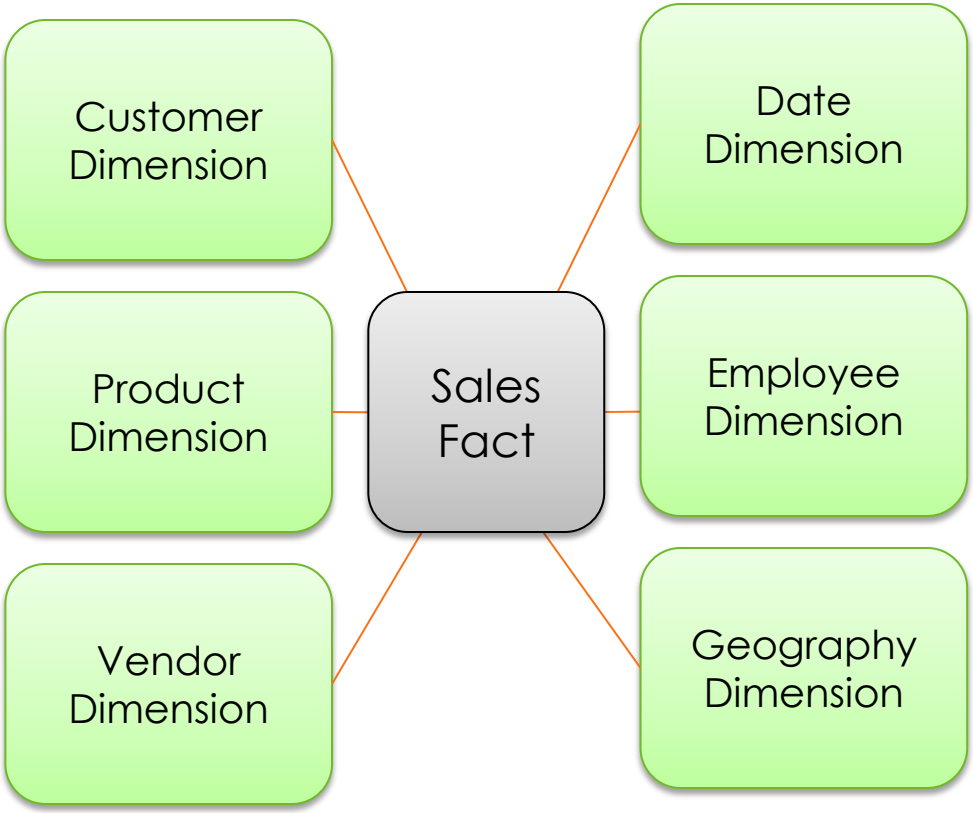
Why Can't I Use SQL?

Multidimensional storage is a fundamental paradigm shift from relational storage

MDX is optimized for dimensional hierarchies & not table/index scans

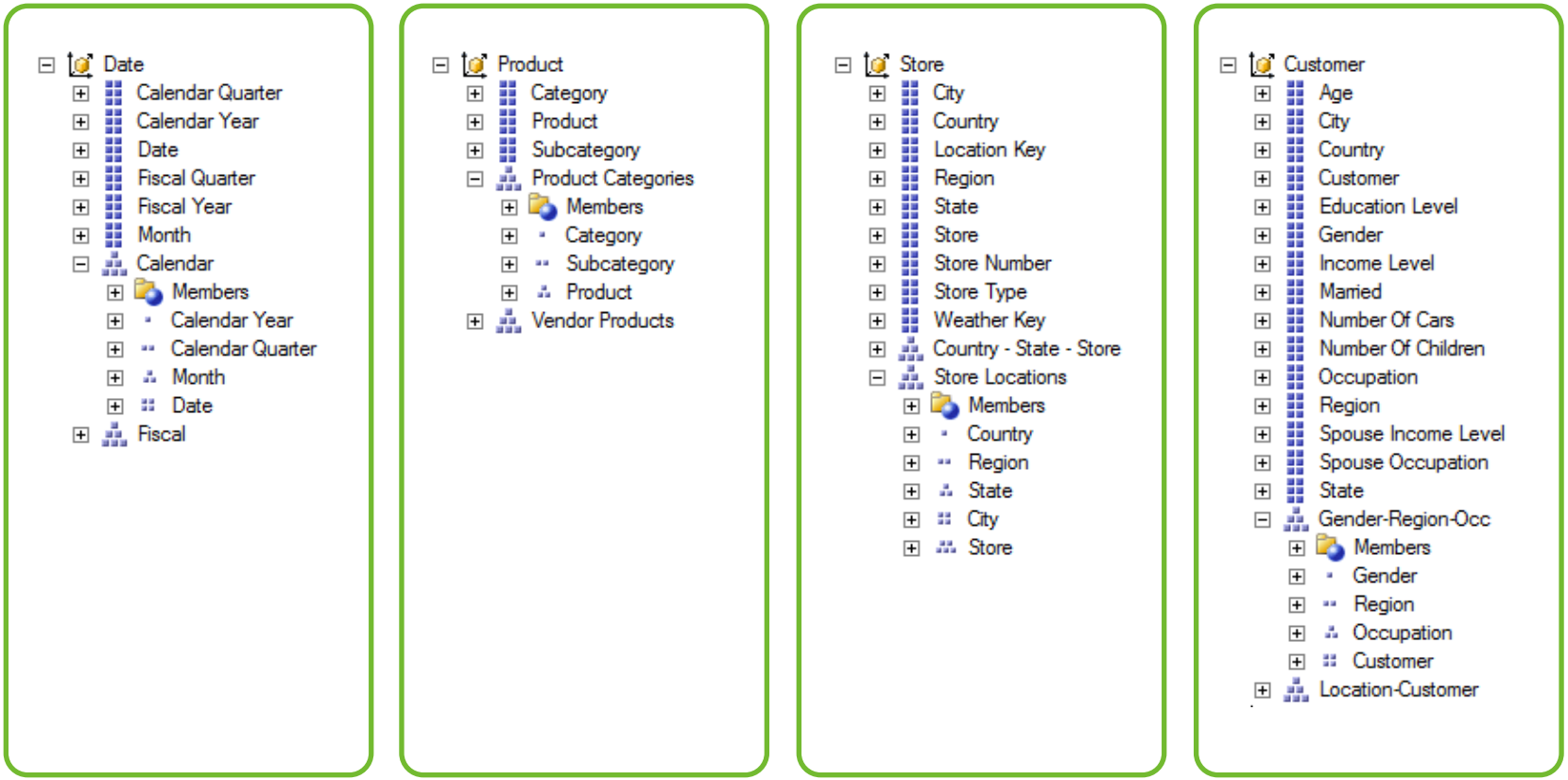


Dimensional Data Warehouse Design



Dimensions

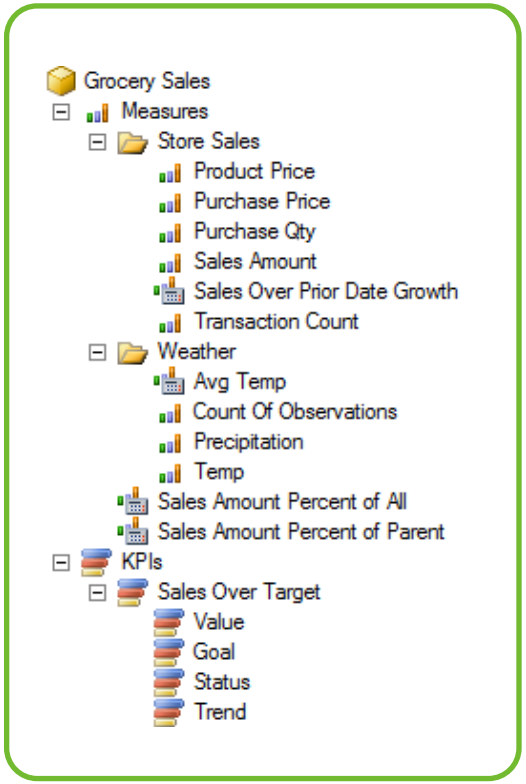
Dimension > Hierarchy > Level > Member



Measures

Organized in measure groups

Derived from numeric fields or SQL calculations



Basic Query Syntax

```
SELECT
    < member or set > on < Columns | Axis(0) | 0 >,
    < member or set > on < Rows | Axis(1) | 1 >
FROM
    < cube or subcube expression >
WHERE
    < member or set > ;
```

```
SELECT
    { [Sales Amount], [Order Quantity] } on Columns,
      [Category].Members on Rows
FROM
      [Adventure Works]
WHERE
      [CY 2001] ;
```

Filtering

Slicer

```
SELECT … on Columns, … on Rows
FROM < cube name >
WHERE { [Category].[Bikes], [Category].[Clothing] }
;
```

Subcube

```
SELECT … on Columns, … on Rows
FROM
    (
      SELECT { [Category].[Bikes], [Category].[Clothing] } on 0
      FROM < cube name >
    )
 ;
```

Sets

Combine members from same hierarchy using braces

{ [Year].[2005], [Year].[2006] }

Tuples

Combine members from different hierarchies using parentheses

([Category].[Bikes], [Year].[2006])

Add measure to return a value from the tuple:

```
(Measures.[Sales Amount]
    , ( [Category].[Bikes], [Year].[2006] ) )
```

Calculated Members

Calculation added to Measures:

```
WITH Member [Measures].[Product Percent of Parent] AS
     [Measures].[Sales Amount] / ( [Measures].[Sales Amount]
   , [Product].[Product Categories].Parent )
   , Format_String = "Percent"
```

Used in Query:

```
SELECT { [Measures].[Sales Amount]
            , [Measures].[Product Percent of Parent] } on Columns
  , NON EMPTY [Product Categories].Category.Members on Rows
FROM < cube name > ;
```

Calculated Members

Calculation added to Hierarchy:

```
WITH Member [Product].[Product Categories].[All].[Category Average] AS
     Avg([Product].[Product Categories].[Category])
```

Used in Query:

```
SELECT { [Measures].[Sales Amount]
          , [Measures].[Product Percent of Parent] } on Columns
  , NON EMPTY [Product Categories].Category.AllMembers on Rows
FROM < cube name > ;
```

Member Functions

.CurrentMember	Returns the current member of a hierarchy Typically used in a calculated member expression
.Parent	Returns the parent member of a member
Ancestor ( )	Returns a member at a level above a member in a user hierarchy – by level name or distance
.PrevMember	Returns a sibling or cousin one position prior
ParallelPeriod ( )	Returns a member at the same level and ordinal position of a member, relative to a specified ancestor
.Item ( )	Returns a specified ordinal member of a set
IsEmpty ( )	Returns True for a tuple containing a value
Is Null	Returns True if a member exists

Set Functions (1)

.Children	Returns set a level below a member
Descendants ( )	Returns set any level below a member (named level or distance from level)
Head ( )	Returns the first X members of a set
Tail ( )	Returns the last X members of a set
Union ( ) or **Set1 + Set2**	Combines 2 sets into one set (Distinct or All)
Intersection ( )	Returns the common members of 2 sets
Except ( ) or **Set1 - Set2**	Returns members of 1st set that don't exist in 2nd set

Set Functions (2)

Order ( )	Returns a set in the order of a specified member/measure
TopCount ( )	Returns top X ranking members based on a measure
Hierarchize ( )	Organizes & sorts members of a set by hierarchal structure. Typically used on Unioned sets .
Filter ( )	Filters a set based on a Boolean expression
Exists ( )	Filters one set by another set from the **same dimension** Returns only members of the 1st set, not the 2nd set
NonEmpty ( )	Filters one set by a set from a **different dimension** Returns only members of the 1st set, not the 2nd set

Set Functions (3)

Sum ( )	Returns a numeric sum for a measure over a set
Aggregate ( )	Like Sum() but uses the measure's AggregateFunction
LastPeriods ( )	Returns a set of previous periods including current
PeriodsToDate ( )	Returns a set of current level members sharing a common time level ancestor
YTD ( )	Simplified PeriodsToDate( ) with fixed time level QTD, MTD, WTD - YTD
Generate ( )	Returns a string from a set, concatenating a member property separated by a specified delimeter
SetToStr ( )	Returns a string from a set. Delimits full UniqueName reference for each member separated by commas

Crossjoins

Same Dimension

Returns coexisting members

```
([Product].[Category].[Category], [Product].[Color].[Color])
([Product].[Category].[Category], [Product].[Color].[Red])
```

Different Dimensions

Returns Cartesian product

```
([Product].[Color].[Color], [Date].[Calendar Year].[Calendar Year])
([Product].[Color].[Color], [Date].[Calendar Year].[2001])
```

Questions

PASS 2011 SUMMIT

Thank You

Resources

Paul's Blog……SqlServerBiBlog.com

paul@SqlServerBiBlog.com

SQL Server 2008 MDX
Bryan C Smith, Ryan Clay
Microsoft Press

SQL Server 2008 Analysis Services
Scott Cameron
Microsoft Press