



ROCK YOUR CODE
TOUR • 2017

CHICAGO

September 18, 2017
9:00am – 6:00pm

Level: Intermediate

SQL Server 2016 for Developers

Andrew Brust

Senior Director, Market Strategy and
Intelligence, Datameer

Leonard Lobel

Chief Technology Officer
Sleek Technologies

-  Datameer Senior Director,
Market Strategy and Intelligence
- Big Data blogger for ZDNet
- Microsoft Regional Director, MVP
- Co-chair Visual Studio Live!
- Twitter: @andrewbrust



Andrew's Blog (bit.ly/bigondata)

The screenshot shows the ZDNet blog interface. At the top, there's a navigation bar with the ZDNet logo, a search icon, and categories like CLOUD, CXO, and HA. Below the header, the title 'Big on Data' is displayed in large, bold, black font. A subtitle below it reads 'Veteran data geek Andrew Brust covers Big Data technologies including Hadoop, NoSQL, Dat...'. On the left, there's a sidebar with a bio for 'ANDREW BRUST' and a photo of an elephant. The main content area features a post thumbnail with a blue background and white text. To the right, there's a section titled 'LATEST POSTS' with a link to 'Hortonworks results, Databricks release and more top the week's Big Data news'. At the bottom right, there's a Visual Studio LIVE! logo.

Meet Lenni



Leonard Lobel

- CTO & Co-Founder
 - Sleek Technologies, Inc.
- Principal Consultant
 - Tallan, Inc.
- Microsoft MVP
 - Data Platform
- Co-organizer
 - NYC .NET Developers Group
- Trainer/Speaker/Author
- Programming since 1979

Contact

- Email: lenni.lobel@sleektech.com
- Blog: lennilobel.wordpress.com
- Twitter: [@lennilobel](https://twitter.com/lennilobel)



Agenda

- **Introduction and Overview**
- **PART I**
 - New SQL Server 2016 Features
- **PART II**
 - Data Warehousing
- **PART III**
 - Big Data and Analytics
- **PART IV**
 - Beyond Relational
- **Demos, demos, demos!**
 - http://bit.ly/vsldc2017_sqlworkshop



Download Slides and Code

[http://bit.ly/
vslchicago2017_sqlworkshop](http://bit.ly/vslchicago2017_sqlworkshop)

(all lower case!)



Schedule

- Workshop Begins
 - 9:00 AM
- Morning Break (15 minutes)
 - 11:00 AM
- Lunch (1 hour)
 - 1:00 PM
- Afternoon Break (15 minutes)
 - 4:00 PM
- Workshop Ends
 - 6:00 PM



Introduction and Overview



SQL Server Versions 2008 – 2016

(Developer Features)

2008

- Table-valued parameters (TVPs)
- New date and time types
- MERGE
- INSERT OVER DML
- GROUPING SETS
- hierarchyid
- FILESTREAM
- Geospatial
- Transparent Data Encryption (TDE)
- Change Data Capture(CDC)
- SQL Audit

2012

- SQL Server Data Tools (SSDT)
- Windowing (OVER) Enhancements
- 14 new general-purpose functions
- 8 new analytic windowing functions
- Server-side paging
- Sequences
- Metadata Discovery
- FileTable
- Columnstore indexes

2016

- DROP IF EXISTS (DIE)
- SESSION_CONTEXT
- Dynamic data masking (DDM)
- Row-level security (RLS)
- Always encrypted
- Stretch database
- JSON
- Temporal
- PolyBase
- QueryStore
- R integration
- Hekaton improvements

2008 R2

- BI Refresh

2014

- Hekaton

Visual Studio  LIVE!
EXPERT SOLUTIONS FOR .NET DEVELOPERS



SQL Server 2016 New Features for Developers

2008

- Table-valued parameters (TVPs)
- New date and time types
- MERGE
- INSERT OVER DML
- GROUPING SETS
- hierarchyid
- FILESTREAM
- Geospatial
- Transparent Data Encryption (TDE)
- Change Data Capture(CDC)
- SQL Audit

2012

- SQL Server Data Tools (SSDT)
- Windowing (OVER) Enhancements
- 14 new general-purpose functions
- 8 new analytic windowing functions
- Server-side paging
- Sequences
- Metadata Discovery
- FileTable
- Columnstore indexes

2016

- **DROP IF EXISTS (DIE)**
- **SESSION_CONTEXT**
- **Dynamic data masking (DDM)**
- **Row-level security (RLS)**
- **Always encrypted**
- **Stretch database**
- **JSON**
- **Temporal**
- **PolyBase**
- **QueryStore**
- **R integration**
- **Hekaton improvements**

2008 R2

- BI Refresh

2014

- Hekaton

Visual Studio  LIVE!
EXPERT SOLUTIONS FOR .NET DEVELOPERS

Visual Studio Live! Chicago 2017

The diagram illustrates the evolution of SQL Server features over time, categorized into three main sections: 2008, 2012, and 2016. It also includes a preview for 2014 and 2008 R2.

SQL Server 2016 New Features for Developers

Exploring SQL Server Tools and Language Enhancements

2008

- Table-valued parameters (TVPs)
- New date and time types
- MERGE
- INSERT OVER DML
- GROUPING SETS
- hierarchyid
- FILESTREAM
- Geospatial
- Transparent Data Encryption (TDE)
- Change Data Capture(CDC)
- SQL Audit

2012

- SQL Server Data Tools (SSDT) Enhancements
- Windowing (OVER) Enhancements
- 14 new general-purpose functions
- 8 new analytic windowing functions
- Server-side paging
- Sequences
- Metadata Discovery
- FileTable
- Columnstore indexes

2016

- DROP IF EXISTS (DIE)
- SESSION_CONTEXT
- Dynamic data masking (DDM)
- Row-level security (RLS)
- Always encrypted
- Stretch database
- JSON
- Temporal
- PolyBase
- QueryStore
- R integration
- Hekaton improvements

2014

- Hekaton

2008 R2

- BI Refresh

Visual Studio LIVE!
EXPERT SOLUTIONS FOR .NET DEVELOPERS

The diagram illustrates the evolution of SQL Server features over time, categorized into four main sections: 2008, 2012, 2016, and a specific section for Native File Streaming in 2012-2014. It also includes a preview for 2014 and 2008 R2.

SQL Server 2016 New Features for Developers

Exploring SQL Server Tools and Language Enhancements

SQL Server 2012-2014 Native File Streaming

2008

- Table-valued parameters (TVPs)
- New date and time types
- MERGE
- INSERT OVER DML
- GROUPING SETS
- hierarchyid
- FILESTREAM
- Geospatial
- Transparent Data Encryption (TDE)
- Change Data Capture(CDC)
- SQL Audit

2012

- SQL Server Data Tools (SSDT) Enhancements
- Windowing (OVER) Enhancements
- 14 new general-purpose functions
- 8 new analytic windowing functions
- Server-side paging
- Sequences
- Metadata Discovery
- FileTable
- Columnstore indexes

2016

- DROP IF EXISTS (DIE)
- SESSION_CONTEXT
- Dynamic data masking (DDM)
- Row-level security (RLS)
- Always encrypted
- Stretch database
- JSON
- Temporal
- PolyBase
- QueryStore
- R integration
- Hekaton improvements

2014

- Hekaton

2008 R2

- BI Refresh

Visual Studio LIVE!
EXPERT SOLUTIONS FOR .NET DEVELOPERS

What's New in SQL Server 2016

- DIE
 - Drop If Exists
- Session context
 - Stateful dictionary
- Dynamic data masking
 - Mask sensitive columns
- Row-level security
 - Filter/block row-level access
- Always encrypted
 - Client-side encryption
- Stretch database
 - Hybrid cloud feature
- Temporal data
 - Point-in-time data access
- Built-in JSON support
 - Store/retrieve/transform JSON
- In-Memory OLTP
 - Improvements over 2014
- Query Store
 - Record & recall query plans
- PolyBase
 - Azure Hadoop/Blob integration
- R Integration
 - Advanced analytics



Data Warehousing

- BI Concepts Dimensional Model, Star Schema
- MPP, APS, Azure SQL DW
- Columnar Databases
- Columnstore indexes
- Vector Processing
- Batch Mode



Big Data and Analytics

- SQL Server Analysis Services
- Reporting Services, “Mobile” Reports
- Power BI
- Big Data, HDInsight, Hadoop and Spark
- PolyBase
- SQL Server R Services
- Other components



Beyond Relational

- XML
 - Native data type for XML storage in the database
- JSON
 - Built-in support for JSON storage in the database
- FILESTREAM
 - Enhance storage of unstructured (BLOB) data in the database
- FileTable
 - Combines FILESTREAM with hierarchyid to expose a logical Win32 file system
- Geospatial
 - Support for planar (flat-earth) and geodetic (round-earth) models



2016

PART I

New SQL Server 2016 Features



2016

Drop If Exists (DIE)



2016

Just DIE (Drop If Exists) Please!

- If you hate this...

```
IF OBJECT_ID('dbo.Product', 'U') IS NOT NULL  
    DROP TABLE dbo.Product;  
  
IF EXISTS (SELECT * FROM sys.triggers  
    WHERE name = 'trProductInsert')  
    DROP TRIGGER trProductInsert
```

- ...you're gonna love this!

```
DROP TABLE IF EXISTS dbo.Product  
DROP TRIGGER IF EXISTS trProductInsert
```



2016

Objects That Can DIE

- AGGREGATE
- ASSEMBLY
- DATABASE
- DEFAULT
- INDEX
- PROCEDURE
- ROLE
- RULE
- SCHEMA
- SECURITY POLICY
- SEQUENCE
- SYNONYM
- TABLE
- TRIGGER
- TYPE
- VIEW



2016

Dynamic Data Masking (DDM)

Visual Studio  EXPERT SOLUTIONS FOR .NET DEVELOPERS

2016

Introducing Dynamic Data Masking (DDM)

- Limit exposure to sensitive data by masking
 - Full – entire column is masked
 - Partial – show starting and/or ending characters of the column data, mask the rest with a custom string
 - Email – show the first character of the column data, mask the rest with XXX@XXXX.com
 - Random – entire column is replaced by random values
- Reveals masked data to queries
 - Data in the database is not changed
- Enforced at the database level
 - No impact at the application level

Visual Studio  EXPERT SOLUTIONS FOR .NET DEVELOPERS

2016

Masking Table Columns

```
CREATE TABLE Customer(  
    FirstName varchar(20)  
        MASKED WITH (FUNCTION='partial(1, "...", 0)'),  
    LastName varchar(20),  
    Phone varchar(12)  
        MASKED WITH (FUNCTION='default()'),  
    Email varchar(200)  
        MASKED WITH (FUNCTION='email()'),  
    Balance money  
        MASKED WITH (FUNCTION='random(1000, 5000)'))  
  
ALTER TABLE Customer  
ALTER COLUMN LastName  
ADD MASKED WITH (FUNCTION='default()')
```

2016

Masking Different Data Types

Masking Function	Behavior	Strings	Numbers	Dates	Other Types
------------------	----------	---------	---------	-------	-------------

2016

Masking Different Data Types

Masking Function	Behavior	Strings	Numbers	Dates	Other Types
<code>default()</code>	Show xxxx mask (strings), or minimum value (other types)	Yes	Yes	Yes	Yes



2016

Masking Different Data Types

Masking Function	Behavior	Strings	Numbers	Dates	Other Types
<code>default()</code>	Show xxxx mask (strings), or minimum value (other types)	Yes	Yes	Yes	Yes
<code>partial(a, 'x', b)</code>	Show first a characters, custom mask, and last b characters	Yes	No	No	No



2016

Masking Different Data Types

Masking Function	Behavior	Strings	Numbers	Dates	Other Types
<code>default()</code>	Show xxxx mask (strings), or minimum value (other types)	Yes	Yes	Yes	Yes
<code>partial(a, 'x', b)</code>	Show first a characters, custom mask, and last b characters	Yes	No	No	No
<code>email()</code>	Show first character and <code>XXX@XXXX.com</code>	Yes	No	No	No



2016

Masking Different Data Types

Masking Function	Behavior	Strings	Numbers	Dates	Other Types
<code>default()</code>	Show xxxx mask (strings), or minimum value (other types)	Yes	Yes	Yes	Yes
<code>partial(a, 'x', b)</code>	Show first a characters, custom mask, and last b characters	Yes	No	No	No
<code>email()</code>	Show first character and <code>XXX@XXXX.com</code>	Yes	No	No	No
<code>random(a, b)</code>	Show random value between a and b	No	Yes	No	No



2016

Discovering Masked Columns

- sys.columns
 - is_masked
 - masking_function
- sys.masked_columns
 - Inherits from sys.columns
 - Filters to show only masked columns
 - WHERE is_masked = 1



2016

Mask Permissions

- DDM is based on user permissions
- Create a table with masked columns
 - No special permission required
- Add, replace, or remove a column mask
 - Requires ALTER ANY MASK permission
- View unmasked data in masked columns
 - Requires UNMASK permission
- Updating data in a masked column
 - No special permission



2016

Dynamic Data Masking (DDM)

demo

Visual Studio  EXPERT SOLUTIONS FOR .NET DEVELOPERS

2016

DDM Limitations and Considerations

- DDM cannot be used with
 - FILESTREAM columns
 - COLUMN_SET, or a sparse column that's part of a COLUMN_SET
 - Computed columns
 - But will return masked data if it depends on a masked column
 - Key for FULLTEXT index
 - Encrypted columns (Always Encrypted)
- Masking is a one-way street
 - Once masked, the actual data can never be obtained
 - An ETL process from a source with masked columns results in an irreversible data loss when loaded into the target environment

Visual Studio  EXPERT SOLUTIONS FOR .NET DEVELOPERS

2016

Row-level Security (RLS)

Visual Studio  EXPERT SOLUTIONS FOR .NET DEVELOPERS

2016

Introducing Row-level Security (RLS)

- Restrict access to individual rows in a table
 - Create predicate functions
 - Write custom logic to control user access to every row
- Security policy
 - Bind the functions to tables as a filter or block predicate
 - SQL Server filters and blocks user access to individual rows
 - Can enable/disable the policy as desired

Visual Studio  EXPERT SOLUTIONS FOR .NET DEVELOPERS

2016

Filter and Block Predicates

- Filter predicate
 - SELECT, UPDATE, DELETE
 - Can't select, update, or delete rows that violate the predicate
- Block predicate
 - AFTER INSERT, AFTER UPDATE
 - Can't insert or update rows to values that would violate the predicate
 - BEFORE UPDATE, BEFORE DELETE
 - Can't update or delete rows that violate the predicate
 - Implied when combined with filter predicate



2016

Combining RLS Predicates

Predicate	SELECT/UPDATE/DELETE rows that violate the predicate	INSERT rows with violating values	UPDATE rows to violating values
-----------	--	-----------------------------------	---------------------------------



2016

Combining RLS Predicates

Predicate	SELECT/UPDATE/DELETE rows that violate the predicate	INSERT rows with violating values	UPDATE rows to violating values
Filter	No	Yes	Yes



2016

Combining RLS Predicates

Predicate	SELECT/UPDATE/DELETE rows that violate the predicate	INSERT rows with violating values	UPDATE rows to violating values
Filter	No	Yes	Yes
AFTER INSERT block	Yes	No	Yes



2016

Combining RLS Predicates

Predicate	SELECT/UPDATE/DELETE rows that violate the predicate	INSERT rows with violating values	UPDATE rows to violating values
Filter	No	Yes	Yes
AFTER INSERT block	Yes	No	Yes
AFTER UPDATE block	Yes	Yes	No



2016

Combining RLS Predicates

Predicate	SELECT/UPDATE/DELETE rows that violate the predicate	INSERT rows with violating values	UPDATE rows to violating values
Filter	No	Yes	Yes
AFTER INSERT block	Yes	No	Yes
AFTER UPDATE block	Yes	Yes	No
BEFORE UPDATE block	No	N/A	N/A



2016

Combining RLS Predicates

Predicate	SELECT/UPDATE/DELETE rows that violate the predicate	INSERT rows with violating values	UPDATE rows to violating values
Filter	No	Yes	Yes
AFTER INSERT block	Yes	No	Yes
AFTER UPDATE block	Yes	Yes	No
BEFORE UPDATE block	No	N/A	N/A
BEFORE DELETE block	No	N/A	N/A



2016

Creating Security Predicate Functions

- Write a security predicate function
 - Ordinary inline table-valued function (TVF)
 - Must be schema-bound
 - Accept any parameters of any type
 - Map these parameters to column values
- Implement your own custom logic in T-SQL
 - Examine the row via the columns passed in as parameters
 - Determine if access should be allowed or denied
 - Return a scalar 1 (allow) or nothing at all (deny)
 - Encapsulate logic inside WHERE clause of a single SELECT statement inside the TVF



2016

Creating Security Predicate Functions

```
CREATE FUNCTION sec.fn_MySecurityPredicate(@Parm1 AS int, ...)
    RETURNS TABLE
    WITH SCHEMABINDING
AS
    -- SQL Server passes in column values of each row via parameters

    RETURN
        SELECT 1 AS Result
        WHERE ...
            -- Custom logic here examines the parameters (column values)
            -- passed in, and determines the row's accessibility
```



2016

RLS Security Policy

- Create a security policy
 - Add filter and block predicates to the policy
- Bind each predicate function to a table
 - Map table columns to the TVF parameters
 - SQL Server will call the TVF to determine the accessibility of each row



2016

RLS Security Policy Examples

- With filter predicate

```
CREATE SECURITY POLICY sec.MySecurityPolicy  
ADD FILTER PREDICATE sec.fn_MySecurityPredicate( Col1, ... )  
ON dbo.MyTable  
WITH ( STATE = ON )
```

- With AFTER INSERT and AFTER UPDATE block predicates

```
CREATE SECURITY POLICY sec.MySecurityPolicy  
ADD BLOCK PREDICATE sec.fn_MySecurityPredicate( Col1, ... )  
ON dbo.MyTable AFTER INSERT,  
ADD BLOCK PREDICATE sec.fn_MySecurityPredicate( Col1, ... )  
ON dbo.MyTable AFTER UPDATE,  
WITH ( STATE = ON )
```



2016

Getting Started with Row-Level Security (RLS)

demo



2016

Identifying Users for RLS

- Credentials supplied for the database connection
 - SQL Server login (username and password)
 - Windows authentication
 - Obtain the username from DATABASE_PRINCIPAL_ID
- Different strategy required for n-tier applications
 - Typically, all users connect to the database using the same service account from the application tier
 - DATABASE_PRINCIPAL_ID is the same for every user
- Solution: Use new SESSION_CONTEXT feature
 - Store the application level user ID as a readonly value in session context



2016

Introducing SESSION_CONTEXT

- What is “session context”?
 - Stateful dictionary (key/value pair) object
 - Key is a Unicode string
 - Value is a sql_variant (any data type)
 - Retains state for the lifetime of the connection
- SESSION_CONTEXT()
 - Built-in function that returns a value from session context by key
- sp_set_session_context
 - System stored procedure that stores a value to session context
 - Specify @key, @value, and optionally, @read_only



2016

Using Row-Level Security (RLS) with SESSION_CONTEXT() for N-Tier Applications

demo



2016

Always Encrypted (AE)



2016

Traditional SQL Server Encryption Features

- Column (cell-level) encryption
 - Uses certificates or symmetric keys
- Database (page-level) and backup encryption
 - Transparent Data Encryption (TDE)
 - Uses TDE certificate with database encryption keys (DEKs)
- Keys and certificates are stored in the database
 - Risk of security breach at the database level
- Data is only encrypted “at rest”
 - Risk of security breach while “in flight”



2016

Introducing Always Encrypted

- Always Encrypted in SQL Server 2016
 - Based on keys managed outside the database
 - Keys are never revealed to SQL Server
- Separating those who own the data from those who manage it
 - Uses client side drivers to encrypt/decrypt on the fly
- SQL server is incapable of decrypting on its own
 - Data is always encrypted in flight
- Enable Always Encrypted
 - Use T-SQL or the Always Encrypted Wizard in SSMS



2016

Encryption Types

- Randomized
 - Unpredictable, more secure
 - No support for equality searches, joins, grouping, indexing
 - Use for data that is returned but not queried
- Deterministic
 - Predictable, less secure
 - Use for data that must be queried
 - Easier to guess by examining encryption patterns
 - Increased risk for small value sets (e.g., True/False)



2016

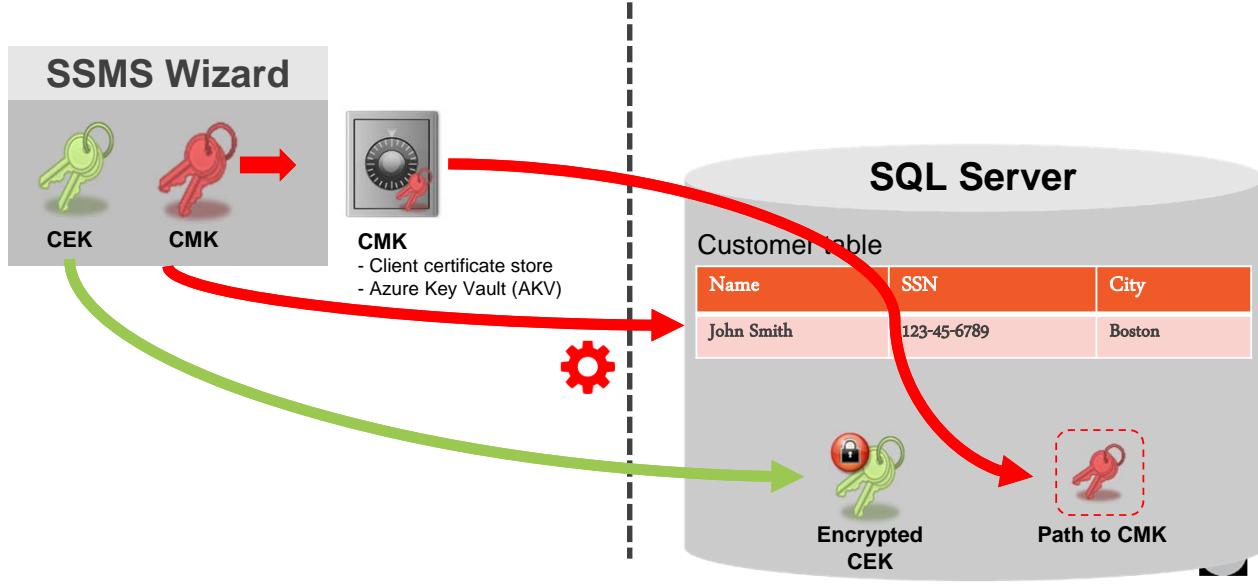
Encryption Keys

- Column Encryption Keys (CEK)
 - Used to encrypt values in specific columns
 - Encrypted versions of each CEK is stored in the database
- Column Master Keys (CMK)
 - Used to encrypt all the CEKs
 - Must be stored externally in a secure key store
 - Key store providers: Azure Key Vault, Certificate store, HSM
- CMK rotation
 - Each CEK can have two encrypted values from two CMKs



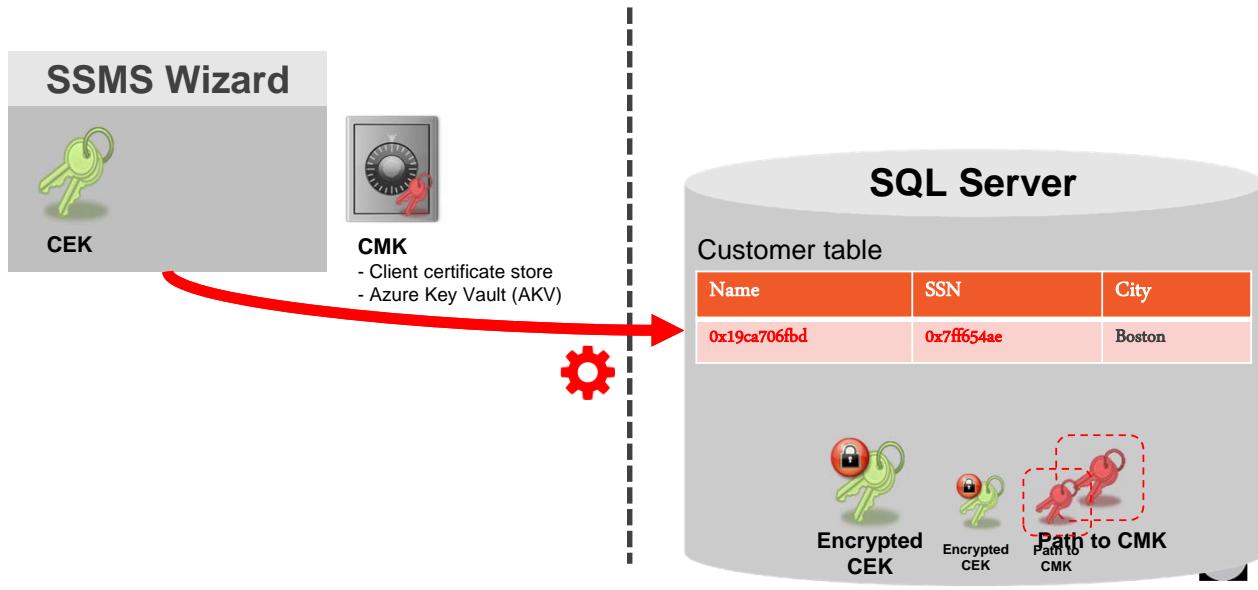
2016

Always Encrypted Workflow

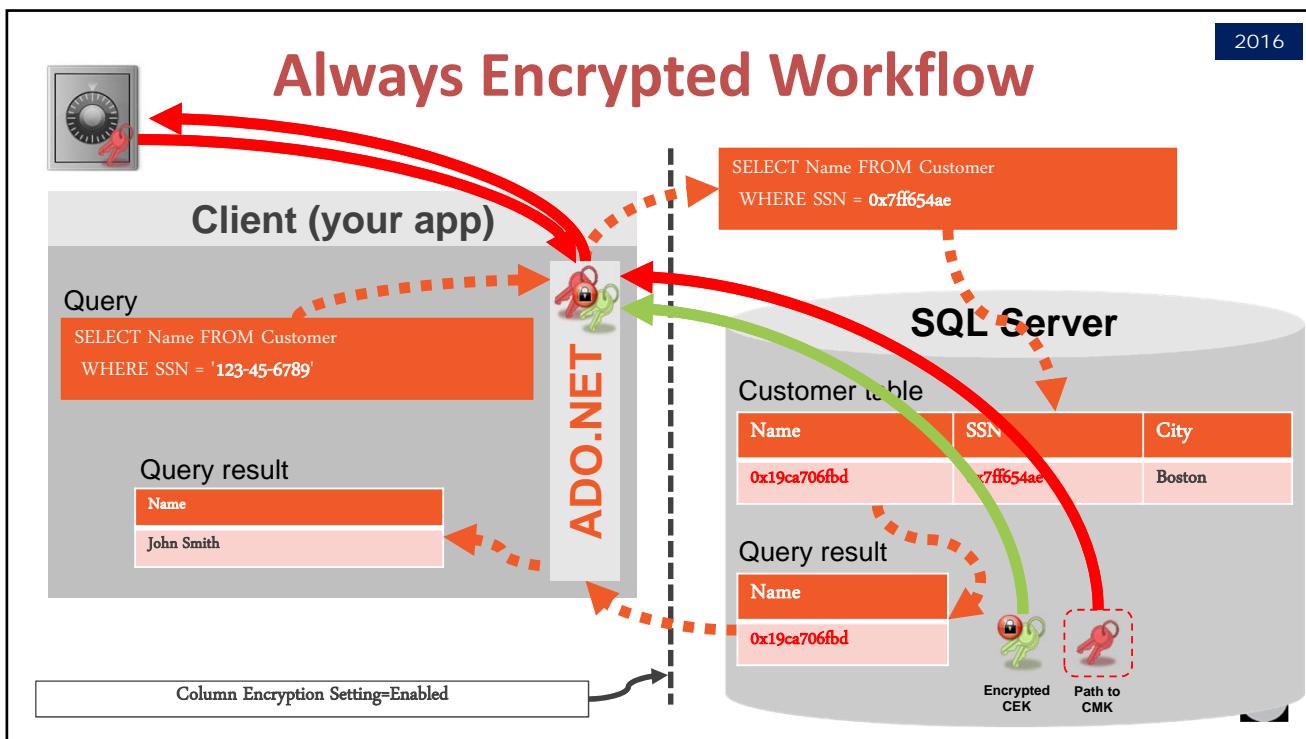


2016

Always Encrypted Workflow



2016



2016

Always Encrypted Wizard (SSMS)

- Creates CMK in either:
 - Local Windows Certificate Store
 - Azure Key Vault
- Creates CEKs
 - Then encrypts them from the CMK
- Deploys to database:
 - Encrypted CEKs
 - Path to CMK
- Runs encryption migration
 - Queries the unencrypted table
 - Encrypts client-side (within SSMS)
 - Creates new encrypted temp table
 - Swaps in the new temp table to replace the old unencrypted table

2016

Always Encrypted Catalog Views

- **sys.column_master_keys**
 - Identifies each CMK
 - Contains external path to CMK location
- **sys.column_encryption_keys**
 - Identifies each CEK
- **sys.column_encryption_key_values**
 - Contains CMK-encrypted values of each CEK
- **sys.columns**
 - New metadata columns to identify encrypted columns



2016

Always Encrypted (AE)
demo



2016

CMK Rotation

- CEKs encrypt all your sensitive data
 - Which is why they are encrypted by a CMK
- The CMK encrypts all your CEKs
 - When the CMK is compromised, all your sensitive data is compromised
- Solution: Rotate the CMK
 - Create a new CMK
 - Re-encrypt the CEKs with the new CMK
 - PowerShell script available at
 - <https://blogs.msdn.microsoft.com/sqlsecurity/2015/08/13/always-encrypted-key-rotation-column-master-key-rotation/>
 - SQL Server Management Studio has integrated GUI support



2016

AE Limitations and Considerations

- Unsupported data types
 - xml, rowversion, image, ntext, text, sql_variant, hierarchyid, geography, geometry
- Also not supported for
 - FILESTREAM, ROWGUIDCOL, sparse, or partitioning columns
 - Fulltext indexes
 - Columns with default constraints
 - Temporal tables
 - Stretch database



2016

AE Limitations and Considerations (cont.)

- Entity Framework 6 considerations
 - <http://blogs.msdn.com/b/sqlsecurity/archive/2015/08/27/using-always-encrypted-with-entity-framework-6.aspx>
- Additional management to install certificates on all clients
- And more...
 - <http://blogs.sqlsentry.com/aaronbertrand/t-sql-tuesday-69-always-encrypted-limitations/>



2016

Stretch Database



2016

Introducing Stretch Database

- Store portions of a database in the cloud
- Remote Data Archive (RDA)
 - Keep “hot” data in local SQL Server database (on-premises)
 - Seamlessly migrate “cold” data to Azure SQL Database
- Stretch Database Advisor
 - Downloadable as part of the SQL Server 2016 Upgrade Advisor
 - Helps identify database and table candidates for stretch
- Enable stretch database
 - Use T-SQL or the Enable Database for Stretch Wizard in SSMS



2016

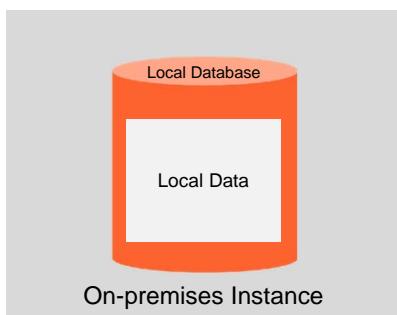
Stretch Database Terminology

- Local database
 - The on-premises SQL Server database being “stretched”
- Eligible data
 - Data in the local database that has not yet been moved to the cloud
- Remote endpoint
 - The Azure SQL Database in the cloud
- Remote data
 - Data that has already been moved from the local database to the cloud

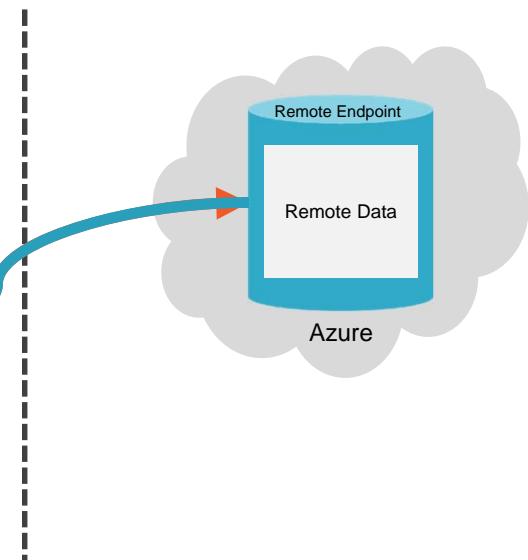
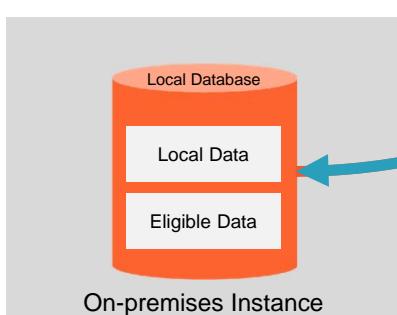


2016

Stretch Database Workflow

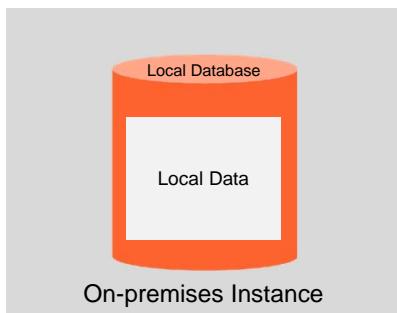


Stretch Database Workflow



2016

Stretch Database Workflow



- `sys.sp_rda_deauthorize_db`
- `sys.sp_rda_reauthorize_db`



2016

Stretching a table
demo

Visual Studio LIVE!
EXPERT SOLUTIONS FOR .NET DEVELOPERS

2016

Stretching Selected Rows

- The entire table doesn't need to be stretched
 - Write a predicate function to filter the rows to be migrated
 - Easily filter by date, status, or other constant
- Filter predicate restrictions
 - Expression must be deterministic
 - Can't inherently implement a sliding window function
- Update filter predicates in place
 - Signature must not change
 - New function must be less restrictive than current one



2016

Stretch Filter Predicate Functions

```
CREATE FUNCTION sec.fn_MyStretchPredicate(@Parm1 AS int, ...)
RETURNS TABLE
WITH SCHEMABINDING
AS
    -- SQL Server passes in column values of each row via parameters

    RETURN
        SELECT 1 AS IsEligible
        WHERE ...
            -- Custom logic here examines the parameters (column values)
            -- passed in, and determines if the row should be migrated
```



2016

Stretch Filter Predicate Functions

```
CREATE FUNCTION sec.fn_MyStretchPredicate(@Parm1 AS int, ...)  
RETURNS TABLE  
WITH SCHEMABINDING  
AS  
-- SQL Server passes in column values of each row via parameters  
  
RETURN  
    SELECT 1 AS IsEligible  
    WHERE ...  
-- Custom logic here examines the parameters (column values  
-- passed in, and determines if the row should be migrated
```



2016

Stretching selected rows
demo



Stretch Pricing

2016

The image shows two separate browser windows side-by-side, both displaying the same "Compute" pricing table from the Microsoft Azure website. The left window shows a smaller set of performance levels (100 to 6000) with corresponding prices. The right window shows a larger set of performance levels (100 to 6000) with higher prices. Both tables have columns for "PERFORMANCE LEVEL (DSU)" and "PRICE".

PERFORMANCE LEVEL (DSU)	PRICE
100	\$2.50/hr
200	\$5/hr
300	\$7.50/hr
400	\$10/hr
500	\$12.50/hr
600	\$15/hr
1000	\$25/hr
1200	\$30/hr
1500	\$37.50/hr
2000	\$50/hr
3000	\$75/hr
6000	\$150/hr

PERFORMANCE LEVEL (DSU)	PRICE
100	\$1,860/mo
200	\$3,720/mo
300	\$5,580/mo
400	\$7,440/mo
500	\$9,300/mo
600	\$11,160/mo
1000	\$18,600/mo
1200	\$22,320/mo
1500	\$27,900/mo
2000	\$37,200/mo
3000	\$55,800/mo
6000	\$111,600/mo

Visual Studio **LIVE!**
EXPERT SOLUTIONS FOR .NET DEVELOPERS

Stretch Database Limitations and Considerations

2016

- Unsupported table types
 - Memory-optimized tables (Hekaton)
 - Replicated tables
 - FILESTREAM/FileTable
 - Tables enabled for Change Tracking or Change Data Capture (CDC)
 - Tables with more than 1023 column or 998 indexes
- Unsupported data types and column properties
 - timestamp, sql_variant, xml, geography, geometry, hierarchyid, CLR UDTs, COLUMN_SET, computed columns
- Unsupported indexes
 - XML, full-text, spatial, indexed views into the table

Visual Studio **LIVE!**
EXPERT SOLUTIONS FOR .NET DEVELOPERS

2016

Stretch Database Limitations and Considerations (cont.)

- Unsupported constraints
 - Check constraints
 - Default constraints
 - Foreign key constraints out of the table (parent tables)
- Uniqueness not enforced UNIQUE constraints
- UPDATE and DELETE not supported
- ALTER TABLE not supported
- Views over stretched tables
 - Can't create index on the view
 - Can't update or delete from the view (but you can insert into the view)



2016

Temporal Data



2016

Introducing Temporal Tables

- System version tables
- Point-in-time data access
 - Query updated and delete data, not just current data
 - Seamless and transparent
- Four primary use cases
 - Time travel
 - Slowly changing dimensions
 - Auditing
 - Accidental data loss recovery



2016

Using Temporal

- Create an ordinary table
 - Must have primary key column
 - Must have two period (start and end date) **datetime2** columns
- Enable the table for temporal
 - Creates history table with same schema, but without constraints
 - Automatically records updates and deletes to the history table
- Query to point in time
 - Include **FOR SYSTEM_TIME AS OF** in your SELECT statement
- Manage schema changes
 - ALTER TABLE automatically updates the history table
 - Some schema changes (e.g., new IDENTITY or computed columns) require turning temporal off, applying the changes to both tables, and then turning it back on



2016

Creating a Temporal Table

```
CREATE TABLE Department
(
    DepartmentID      int NOT NULL IDENTITY(1,1) PRIMARY KEY,
    DepartmentName   varchar(50) NOT NULL,
    ManagerID        int NULL,
    ValidFrom        datetime2 GENERATED ALWAYS AS ROW START NOT NULL,
    ValidTo          datetime2 GENERATED ALWAYS AS ROW END    NOT NULL,
    PERIOD FOR SYSTEM_TIME (ValidFrom, ValidTo)
)
WITH (SYSTEM_VERSIONING = ON (HISTORY_TABLE = dbo.DepartmentHist))
```



2016

Querying a Temporal Table

```
DECLARE @ThirtyDaysAgo datetime2 =
DATEADD(d, -30, SYSDATETIME())
```

```
SELECT *
FROM Employee
FOR SYSTEM_TIME AS OF @ThirtyDaysAgo
ORDER BY EmployeeId
```



2016

Temporal Data

demo

Visual Studio **LIVE!**
EXPERT SOLUTIONS FOR .NET DEVELOPERS

2016

Temporal Limitations and Considerations

- Triggers
 - INSTEAD OF triggers are unsupported
 - AFTER triggers are supported on the current table only
- Cascading updates and deletes are not supported
- In-memory OLTP (Hekaton) is not supported
- FILESTREAM/FileTable is not supported
- INSERT and UPDATE statements cannot reference the period columns
- Works with other new SQL Server 2016 features
 - DDM, RLS, Always Encrypted, Stretch

Visual Studio **LIVE!**
EXPERT SOLUTIONS FOR .NET DEVELOPERS

PART II

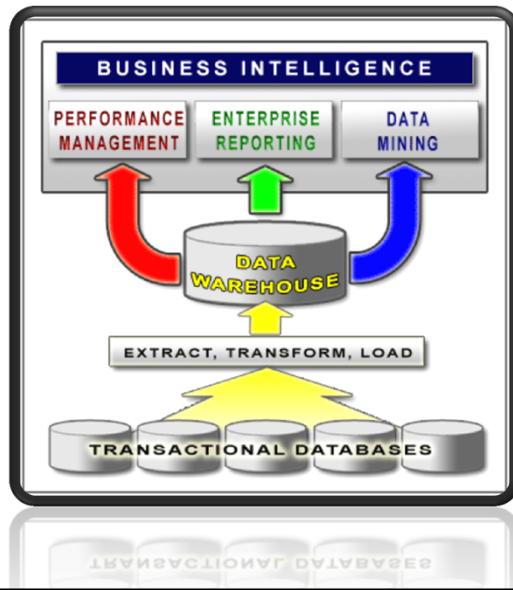
Data Warehousing



BI Concepts

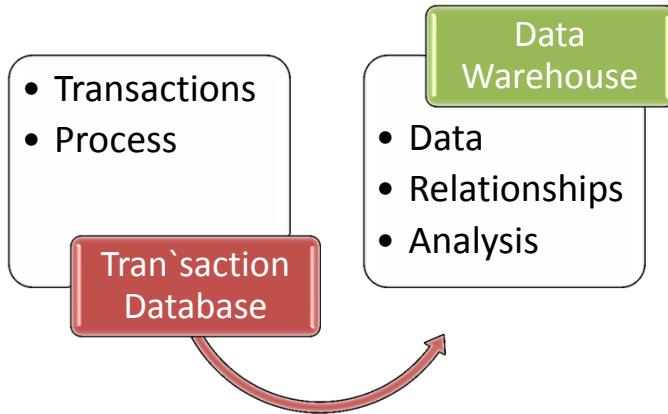


Business Intelligence



Visual Studio **LIVE!**
EXPERT SOLUTIONS FOR .NET DEVELOPERS

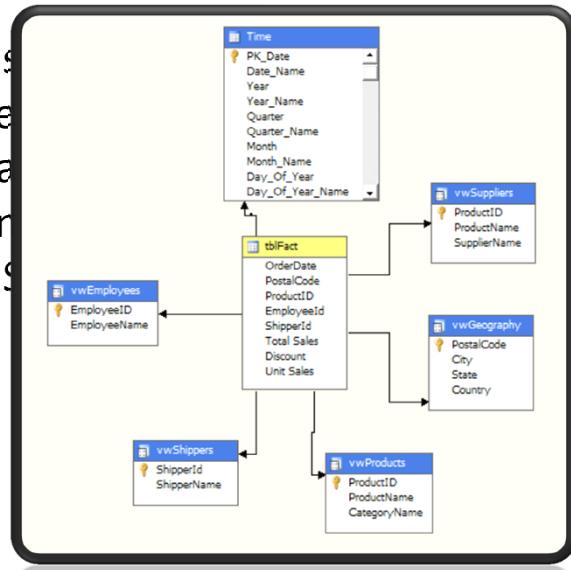
Preparing For Business Intelligence



Visual Studio **LIVE!**
EXPERT SOLUTIONS FOR .NET DEVELOPERS

Dimensional Model

- Measures
- Dimensions
- Hierarchies
- Grains
- Star Schema



Visual Studio **LIVE!**
EXPERT SOLUTIONS FOR .NET DEVELOPERS

Star Schemas

- Physical data model
- Central fact table
- Multiple dimension tables
 - Used to constrain fact table queries



Visual Studio **LIVE!**
EXPERT SOLUTIONS FOR .NET DEVELOPERS

Example Data Request

- Get Total Sales By State, By Month for a Calendar Year For Country = USA and Calendar Year = 1996

Visual Studio **LIVE!**
EXPERT SOLUTIONS FOR .NET DEVELOPERS

Data Warehouse Query

	STATE	Month_Name	(No column name)
1	NM	August 1996	3343.60
2	WY	August 1996	48.00
3	ID	December 1996	6038.60
4	OR	December 1996	780.00
5	WY	December 1996	3391.20
6	NM	July 1996	624.80
7	WA	July 1996	676.00
8	NM	November 1996	1731.20
9	WA	November 1996	2856.00
10	WY	November 1996	141.60
11	AK	October 1996	934.50

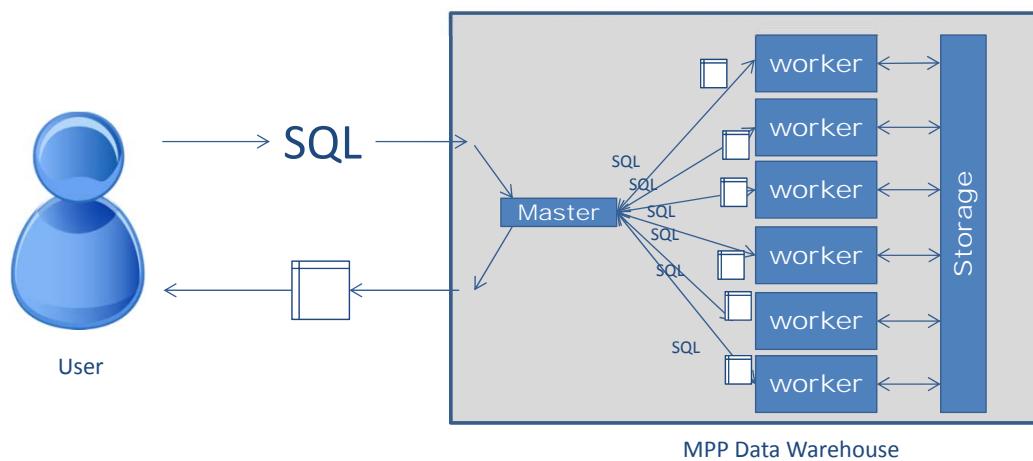
Visual Studio **LIVE!**
EXPERT SOLUTIONS FOR .NET DEVELOPERS

What is MPP?

- Massively Parallel Processing
 - A cluster of individual RDBMS instances (worker nodes)
 - One master node, in front
 - Takes query, delegates parts of it to different worker nodes
 - Combines worker nodes' results, returns as single result set
 - Thus, appears as a single RDBMS
 - Send it one query, get back one result set
 - But query is highly parallelized, so it's fast
 - Perfect for data warehouses
 - Bears some resemblance to MapReduce
 - Examples include Teradata, HP Vertica, IBM Netezza, Pivotal Greenplum

Visual Studio **LIVE!**
EXPERT SOLUTIONS FOR .NET DEVELOPERS

What is MPP?



Visual Studio **LIVE!**
EXPERT SOLUTIONS FOR .NET DEVELOPERS

MPP + Columnar

- Together, these greatly accelerate DW performance.
- Far superior to a scaled-up SQL Server Enterprise box
- Most DW platforms combine these two technologies
- Add vector processing and it's a big deal



MPP at Microsoft?

- Yes, resulting from 2008 acquisition of DATAAllegro
 - Open source MPP based on Ingres, written in Java, running on Linux
- Project Madison
 - Apply DATAAllegro architecture using SQL Server, .NET and Windows
 - Released as SQL Server Parallel Data Warehouse (PDW)
 - Now called Analytics Platform System (APS)



Microsoft APS

- Sold as appliance, from Dell and HP
- Not 100% T-SQL compatible, but close
- Extends SQL Server's cost-based optimizer to include cluster-distribution of queries



Azure SQL Data Warehouse

- In preview
- Think of it as PDW/APS in the cloud
- Scale compute and storage separately
- Pause and resume
- Competitors: Amazon Redshift, Snowflake Elastic DW, Cazena, IBM DashDB



Columnstore Indexes



In Analytics...

- Geared to reporting and visualization
 - Read frequently, write seldom
- Table scans are expected
- Aggregation (think GROUP BY) is *de riguer*
- Extensive normalization is bad
- You only care about values in a small set of columns...maybe even just one
 - The rest are used with WHERE and HAVING, to filter
- Tables that track location and time are common



Column-Oriented Stores

- Imagine, instead of:

Employee ID	Age	Income
1	43	90000
2	38	100000
3	35	100000

- You have:

Employee ID	1	2	3
Age	43	38	35
Income	90000	100000	100000

- Perf: values you wish to aggregate are adjacent
- Efficiency: great compression from identical or nearly-identical values in proximity
- Fast aggregation and high compression means huge volumes of data can be stored and processed, in RAM



A History of Columnstore Indexes

- SQL Server 2012: Nonclustered Columnstore Indexes (NCCIs) added to product
 - Read only
- SQL Server 2014: Clustered Columnstore Indexes (CCIs) added
 - Read/Write
- SQL Server 2016: Numerous enhancements to CCIs



Vector Processing

- Intel x86 CPUs have, since supported “single instruction multiple data” (SIMD) operations since the 1990s
- These process data in parallel, handling multiple data points simultaneously
- This is called vector processing
- SQL Server NCCIs and CCIs can take advantage of it



Useful Applications

- Data Warehouse/Data Mart scenarios
- In combination with DirectQuery feature in SSAS Tabular and Power BI (covered soon)
- In combination with R Services

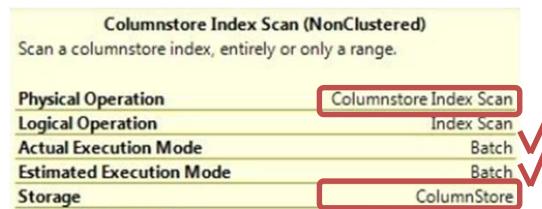


Vector Processing and “Batch” Mode

- SQL Server can burst into a vector processing mode
- Instead of iterating through rowsets, one row at a time, it can handle rows in batches
- So it's called “batch mode” and it's *fast*
 - (Not to be confused with batch processing, which can be slow)

Visual Studio **LIVE!**
EXPERT SOLUTIONS FOR .NET DEVELOPERS

Sanity Check



Visual Studio **LIVE!**
EXPERT SOLUTIONS FOR .NET DEVELOPERS

Making Sure it Works

- CLs are fastest when Batch mode kicks in
 - Difference can be negligible otherwise
 - Check Query Plan to make sure
- And meet the prerequisites...



Prerequisites

1. More than one CPU core

- (Careful on those VMs!)

2. Maximum Degree of Parallelism (MDOP) set to 0

- Or a value between 2 and 64, if you want to limit it
- Use SSMS server properties sheet or `sp_configure` and `RECONFIGURE`

Lots of data

- Millions of rows, or don't bother



Columnstore Indexes

demo

Credit: Microsoft SQL Server 2016 CTP3 Samples



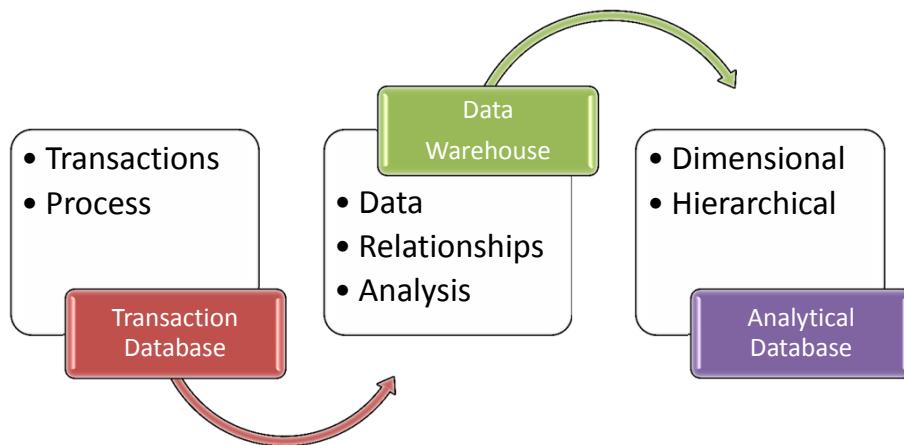
PART III Big Data and Analytics



Analysis Services

Visual Studio **LIVE!**
EXPERT SOLUTIONS FOR .NET DEVELOPERS

Data Migration



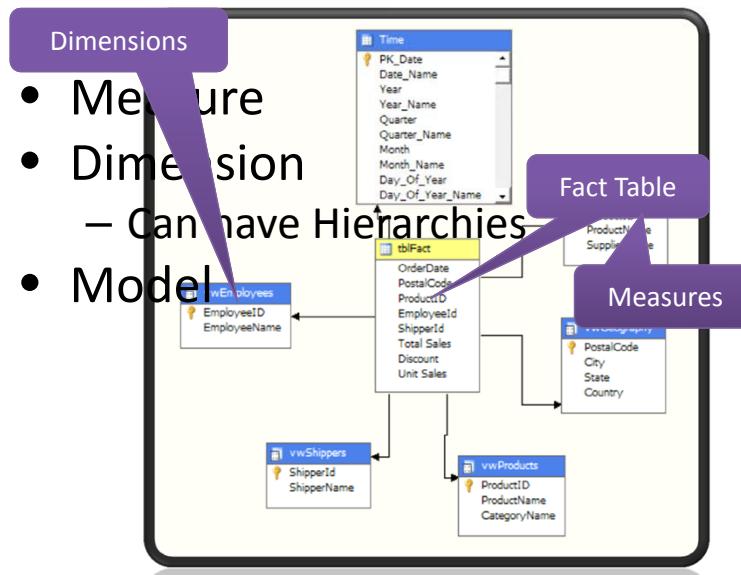
Visual Studio **LIVE!**
EXPERT SOLUTIONS FOR .NET DEVELOPERS

SQL Server Analysis Services

- Built for analysis
 - Included with SQL Server Standard, Enterprise
 - And you can use the Microsoft stack that you know and love

Visual Studio® LIVE!
EXPERT SOLUTIONS FOR .NET DEVELOPERS

From Data Warehouse to OLAP



Visual Studio® LIVE!
EXPERT SOLUTIONS FOR .NET DEVELOPERS

Analysis Services Modes

- **Multidimensional or Tabular**
- Tabular is newer, same tech as Excel/PowerPivot data models and Power BI
- Lots of investment in Tabular in SSAS 2016
- We'll look at Tabular today



Analysis Services Tabular Mode

- SSAS Tabular Mode uses a columnar storage engine in place of a multidimensional one
- Must choose mode for SSAS instance at install time
- Can have default instance with one, named instance with the other
- Can create an SSAS Tabular database project by importing an Excel workbook with PowerPivot model
- SSAS tabular models support partitions, roles, translations, display folders

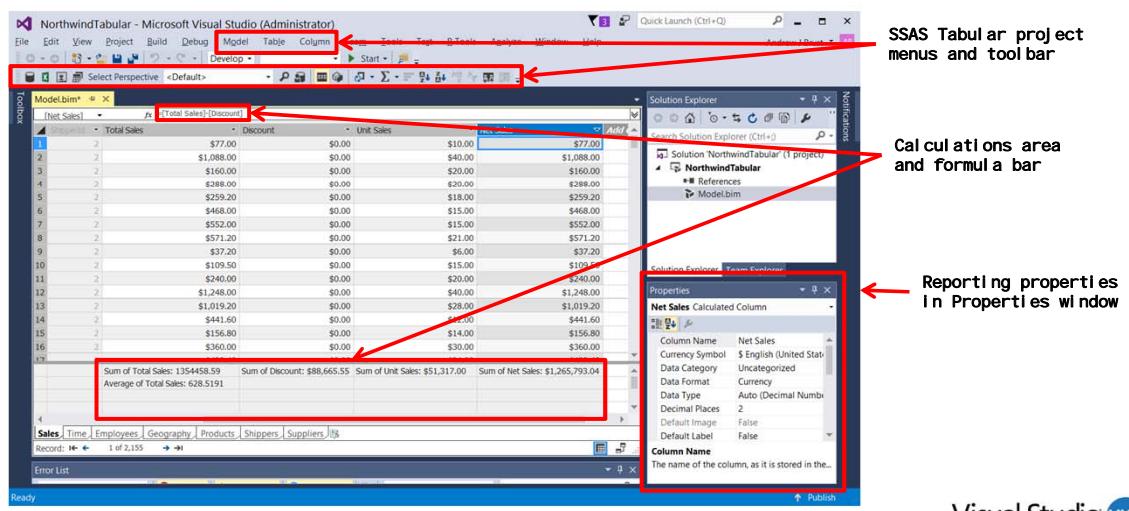


Calculated Columns and DAX

- Formula-based columns may be created
 - Formula syntax is called DAX (Data Analysis eXpressions).
 - Not to be confused with MDX or DMX. Or DACs.
 - DAX expressions are similar to Excel formulas
 - Work with tables and columns; similar to, but distinct from, worksheets and their columns (and rows)
 - =FUNC('table name'[column name])
 - =FUNCX('table name', <filter expression>)
 - FILTER(Resellers,[ProductLine] = "Mountain")
 - RELATED(Products[EnglishProductName])
 - DAX expressions can be heavily nested

Visual Studio® LIVE!
EXPERT SOLUTIONS FOR .NET DEVELOPERS

SSAS Tabular Project in SSDT (Visual Studio)



Building OLAP Cube With BIDS

- Use SQL Server Developer Tools (in VS 2015)
- Select Business Intelligence (in tree view @ left under Templates)
- Then Analysis Services Tabular Project, enter project name, click OK
- Then:



Visual Studio **LIVE!**
EXPERT SOLUTIONS FOR .NET DEVELOPERS

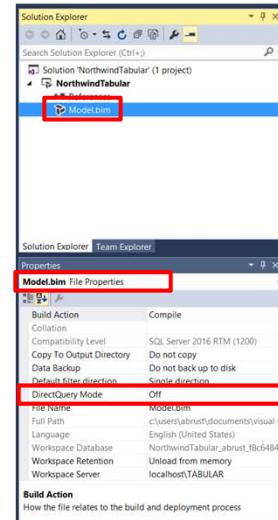
Creating an SSAS Tabular Project

demo

Visual Studio **LIVE!**
EXPERT SOLUTIONS FOR .NET DEVELOPERS

DirectQuery Mode

- In DQ mode, model defines schema, but is not used for data
- Queries issued directly against source
- Similar to ROLAP storage for conventional cubes
- Combine with Columnstore indexes for fast, real-time querying
- DirectQuery now supports: SQL Server, APS, Oracle and Teradata



Visual Studio **LIVE!**
EXPERT SOLUTIONS FOR .NET DEVELOPERS

Reporting Services

Visual Studio **LIVE!**
EXPERT SOLUTIONS FOR .NET DEVELOPERS

What's New in Reporting Services?

- The Web UI has been completely revamped...*finally*
- KPIs and “mobile” reports added
 - Technology from Datazen Acquisition
- Conventional SSRS reports renamed “paginated” reports
- Report Builder updated. Sunburst, Treemaps added.
- Can pin individual objects in report to Power BI dashboards



SSRS + SSAS = YES(S)

- Reporting Services can query Analysis Services databases, including multidimensional cubes and tabular models, directly. PowerPivot too.
- Uses MSOLAP OLE DB provider and issues MDX queries
 - Has its own query designer for MDX
- Beware: SSRS essentially “flattens” SSAS data
 - Conforming multidimensional data to relational structures



SSRS + SSAS Tabular demo

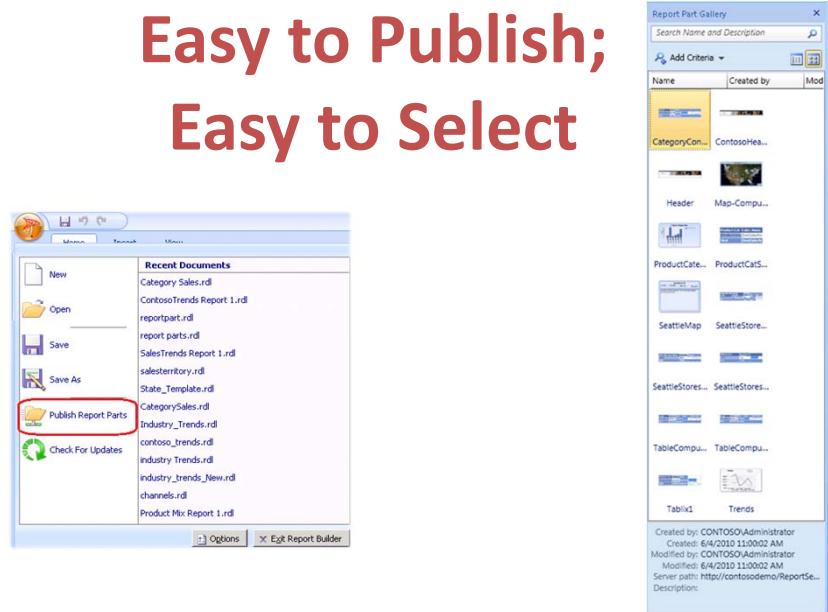


Report Parts

- Skilled SSRS designers can publish report parts
 - From Report Builder 3.0 or VS report projects
- End users can pick them from a gallery
 - A task pane, in Report Builder 3.0, with search capability
 - Cannot select from VS report designer
- What can be published?:
 - Tablixes (i.e. tables, matrices)
 - Rectangles
 - Images, Charts, Gauges, Maps
 - Parameters and Lists
- All aided by ability to share Datasets and Data Sources



Easy to Publish; Easy to Select



Visual Studio **LIVE!**
EXPERT SOLUTIONS FOR .NET DEVELOPERS

What's Where?

Component	On-Prem	Cloud	Mobile
Paginated reports	●	Via PBI dashboard pinning or SQL Server on Azure VM	✗
KPIs and Mobile Reports	●	Via SQL Server on Azure VM only	Via Power BI mobile apps
Power BI	Via SSRS server + Power BI Desktop or Pyramid Analytics now; SSRS report Manager/browser access to come	●	●

Visual Studio **LIVE!**
EXPERT SOLUTIONS FOR .NET DEVELOPERS

Mobile Reports

- Derived from Datazen application
- Create them in Mobile Report Publisher
- As with Report Builder, save them locally or on SSRS server.
 - Files have a .rsmobile extension
- Access on desktop via Report Manager (in Web browser)
- Access on mobile devices via Power BI apps
 - Which get configured to point to SSRS instance
 - Replaces Datazen mobile app



Mobile Reports: Authoring

- Based on controls and a grid
- Build for desktop form factor first
 - Despite the name
- Create secondary versions for tablet and phone
- Connect to Excel spreadsheets or SSRS Shared Datasets.
- Set visual properties in Layout tab; data properties in Data tab
 - Option buttons very important
 - Settings and Preview are the only other tabs



SQL Server Mobile Reports

demo

Credit: TechNet Virtual Lab: “Exploring What's New in
SQL Server 2016 Reporting Services for Mobile
Reporting”



Power BI





Power BI

- Desktop, Mobile and Cloud BI
- Based on same columnar, in-memory BI engine as SQL Server Analysis Services Tabular mode
- Two subscription levels: \$0 and \$10/month/seat
- Easy to use, extensible, embeddable, connects to a huge array of conventional and cloud data sources
 - Growing DirectQuery support
- Highly integrated across Microsoft stack



Power BI Ingredients



Power BI Desktop

- Windows desktop app
- Acquire, shape data query editor
- Visualize data with report view



Browser Environment

- www.powerbi.com
- Edit, consume
- On-prem gateways



iOS, Android, Widows Store Apps

- iPad, iPhone, Android phone
- Windows tablets, PCs
- Consumption only



Power BI Desktop

- Windows Desktop Application
- Has a “main window,” akin to the Excel Power View Add-In, for report authoring and some data modeling
 - Report view
 - Data view
 - Relationships view
- Has a Query Editor window, akin to the Excel Power Query Add-In, for data import and transformation
- Can save files (.pbix) locally and publish them to powerbi.com

Visual Studio EXPERT SOLUTIONS FOR .NET DEVELOPERS



Power BI Query Editor: Overview

- Launched with Get Data option (from ribbon or splash page)
- Re-entered using Edit Queries ribbon button
- Use it to import and shape data
- Use Close & Load ribbon button when done
- Try not to confuse this window with the data view in the main window

Visual Studio EXPERT SOLUTIONS FOR .NET DEVELOPERS

Power BI Desktop Data Sources



File

- Excel
- CSV
- XML
- Text
- JSON
- Folder
- SharePoint Folder

Database

- SQL Server Database
- Access Database
- Oracle Database
- IBM DB2 Database
- MySQL Database
- PostgreSQL Database
- Sybase Database
- Teradata Database
- SAP HANA Database

Azure

- Microsoft Azure SQL Database
- Microsoft Azure SQL Data Warehouse
- Microsoft Azure Marketplace
- Microsoft Azure HDInsight
- Microsoft Azure Blob Storage
- Microsoft Azure Table Storage
- Azure HDInsight Spark (Beta)
- Microsoft Azure DocumentDB (Beta)
- Microsoft Azure Data Lake Store (Beta)

Online Services

- SharePoint Online List
- Microsoft Exchange Online
- Dynamics CRM Online
- Facebook
- Google Analytics
- Salesforce Objects
- Salesforce Reports
- appFigures (Beta)
- GitHub (Beta)
- MailChimp (Beta)
- Marketo (Beta)

- QuickBooks Online (Beta)
- SparkPost (Beta)
- Smartsheet
- SQL Sentry (Beta)
- Stripe (Beta)
- SweetIQ (Beta)
- Twilio (Beta)
- tyGraph (Beta)
- Webtrends (Beta)
- Zendesk (Beta)



Data Shaping, Manipulation



- Configure data types
- Rename and remove columns
- Insert custom columns
- Split columns (i.e. convert delimited text to columns)
- Sort, Group
- Pivot, unpivot columns
- Filter data
- Replace values
- Keep top n rows, or specific range of rows
- Remove duplicates



Power BI Query Editor



demo

Visual Studio **LIVE!**
EXPERT SOLUTIONS FOR .NET DEVELOPERS



Power BI Reports Overview

- Data exploration and visualization client
- Visualizations work as filters, too
- Design and view experiences are unified
- Power View is Excel, SharePoint counterpart
 - Silverlight-based
 - SharePoint version can export to PowerPoint

Visual Studio **LIVE!**
EXPERT SOLUTIONS FOR .NET DEVELOPERS



The Views

-  Report: the report designer/viewer
-  Data: where you can model the data
 - Rename/delete/hide columns and tables
 - Sort by a column (ascending or descending)
 - Add DAX measures and calculated columns
 - Set data types and categories
-  Relationships – Where you can view and edit relationships
 - But you must create them with the Manage Relationships dialog



Power BI Reports



demo



PowerBI.com

- Authoring and consumption tool
- Connectivity to social and SaaS data sources
- Special connectors for on-prem data sources
- Can create three things
 - Dataset
 - Report
 - Dashboard
- Publish report from PBI Desktop, get link to cloud version
 - Also, “Quick Insights”

Visual Studio **LIVE!**
EXPERT SOLUTIONS FOR .NET DEVELOPERS

PowerBI.com Data Sources



Require Pro account:



Visual Studio **LIVE!**
EXPERT SOLUTIONS FOR .NET DEVELOPERS

Gateways

- Permit refresh on cloud copy of report connected to on-premises data source
- Personal gateway must be installed on local machine
- Enterprise gateway installed on server and shared
- Require Pro account
- DirectQuery/“Connect live”
 - Azure SQL DB, Azure SQL DW, Spark (on HDInsight or on-prem)
 - With gateway: SQL Server SSAS (Tabular or MD), Oracle, Teradata, SAP HANA



Power BI and Reporting Services

- Reporting Services (2016) visualizations can be pinned to Power BI dashboards
- This is not enabled by default
- And configuration is not easy
 - Involves the Report Manager UI and Azure Active Directory



Power Q&A

- Natural language query interface to data in underlying model
- Available at top of dashboard
- Generates Power View visualization as you type
- Visualization is pin-able
- Featured Q&A questions/suggestions



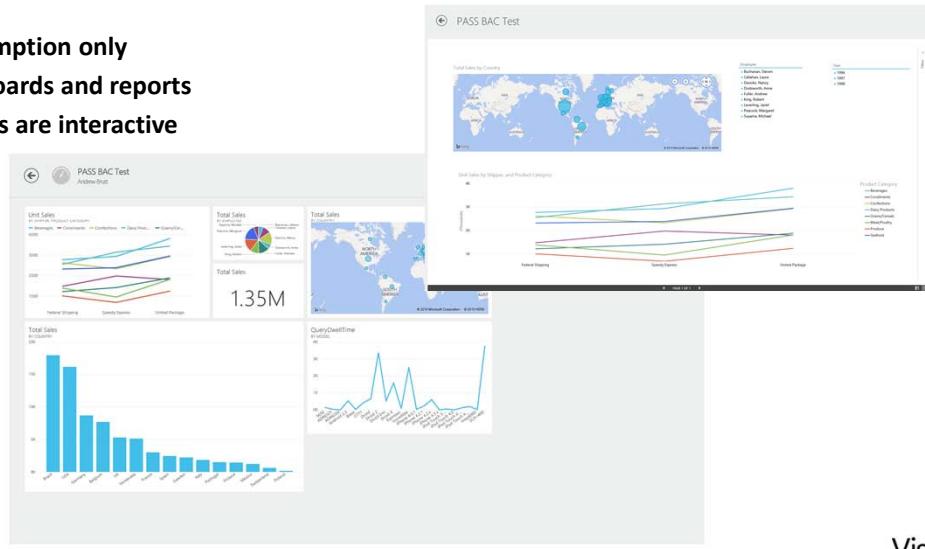
PowerBI.com

demo



Windows Universal, iOS, Android Apps

- Consumption only
- Dashboards and reports
- Reports are interactive



Visual Studio **LIVE!**
EXPERT SOLUTIONS FOR .NET DEVELOPERS

Power BI Windows Universal App

demo

Visual Studio **LIVE!**
EXPERT SOLUTIONS FOR .NET DEVELOPERS

Big Data

Visual Studio  EXPERT SOLUTIONS FOR .NET DEVELOPERS

What is Big Data?

- 100s of TB into PB and higher
- Involving data from: financial data, sensors, web logs, social media, etc.
- Parallel processing often involved
 - Hadoop is emblematic, but other technologies are Big Data too
- Processing of data sets too large for transactional databases
 - Analyzing *interactions*, rather than *transactions*
 - The three V's: Volume, Velocity, Variety
- Big Data tech sometimes imposed on small data problems

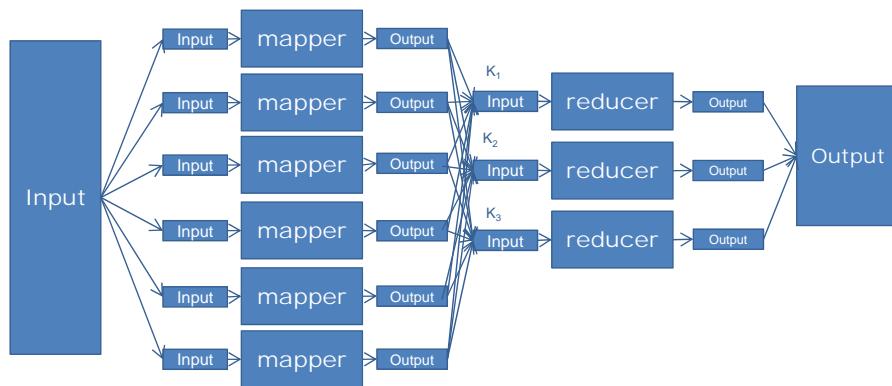
Visual Studio  EXPERT SOLUTIONS FOR .NET DEVELOPERS

What's MapReduce?

- “Big” data input accepted in file form
- Data is partitioned and sent to *mappers* (nodes in cluster)
- Mappers pre-process data into KV pairs, then all output for (a) given key(s) goes to a *reducer*
- Reducers aggregate; one line of output per unique key, with one value
- Map and Reduce code natively written as Java functions

Visual Studio **LIVE!**
EXPERT SOLUTIONS FOR .NET DEVELOPERS

MapReduce, in a Diagram



Visual Studio **LIVE!**
EXPERT SOLUTIONS FOR .NET DEVELOPERS



HDFS

- File system whose data gets distributed over commodity drives on commodity servers
- Data is replicated
- If one box goes down, no data lost
 - “Shared Nothing”
 - Except the name node
- BUT: Immutable
 - Files can only be written to once
 - So updates require drop + re-write (slow)
 - You can append though
 - Like a DVD/CD-ROM



The Hadoop Stack

Log file integration



Machine Learning/Data Mining



RDBMS Import/Export



Query: HiveQL and Pig Latin



Database



MapReduce, HDFS



Hadoop Distributions

- Cloudera (CDH)
- MapR
 - Network File System replaces HDFS
- Hortonworks (HDP)
- Open Data Platform initiative (ODPi)
 - IBM InfoSphere BigInsights
 - HDFS<->DB2 integration
- And Microsoft...

 cloudera

 MAPR

 Hortonworks

 OPEN DATA PLATFORM

 IBM
InfoSphere
BigInsights

Visual Studio  LIVE!
EXPERT SOLUTIONS FOR .NET DEVELOPERS

Microsoft HDInsight



- Developed with Hortonworks and incorporates Hortonworks Data Platform (HDP) *for Windows*
- Windows Azure HDInsight and Microsoft HDInsight Server
 - Single node preview runs on Windows client
 - Also Hortonworks HDP for Windows
 - Also HDInsight with Analytics Platform System
- Includes ODBC Drivers for Hive
- All contributed back to open source Apache project

Visual Studio  LIVE!
EXPERT SOLUTIONS FOR .NET DEVELOPERS



Azure HDInsight Provisioning

The screenshot shows the Azure portal interface for creating a new HDInsight cluster. The left sidebar shows 'New' selected. The main area displays 'Data + Analytics' and 'FEATURED APPS' sections. In the 'FEATURED APPS' section, 'HDInsight' is highlighted with a red box. The 'Cluster Type configuration' step is also highlighted with a red box. The 'Cluster Type' dropdown is set to 'Windows'. The 'Operating System' dropdown is also set to 'Windows'. The 'Version' dropdown is set to 'Hadoop 2.6.0 (HDI 3.2)'. The 'PREMIUM (PREVIEW)' button is visible at the bottom.

Visual Studio LIVE!
EXPERT SOLUTIONS FOR .NET DEVELOPERS



Azure HDInsight Provisioning

The screenshot shows the 'Cluster Type configuration' page. The 'Cluster Type' dropdown is set to 'Hadoop'. The 'Operating System' dropdown is set to 'Linux'. The 'Version' dropdown is set to 'Hadoop 2.6.0 (HDI 3.2)'. A red box highlights the 'Linux' option in the operating system dropdown. A red arrow points from the text 'Linux only' to the 'Linux' option. A red box highlights the 'PREMIUM (PREVIEW)' button at the bottom right.

Visual Studio LIVE!
EXPERT SOLUTIONS FOR .NET DEVELOPERS

Working with HDInsight (Linux)



- Apache Ambari
 - For Hive queries and cluster monitoring
- Access via PowerShell and HDInsight cmdlets
 - Need to install PowerShell for Microsoft Azure
 - **Run your PowerShell client as administrator**
- SSH into head node
 - Use PuTTY or new Bash shell on Windows 10
 - To ssh-*clustername*.azurehdinsight.net

Visual Studio EXPERT SOLUTIONS FOR .NET DEVELOPERS

Submitting, Running and Monitoring Jobs (Linux)

- Upload a JAR
- Run at command line (PowerShell or Linux Command line) passing JAR name and params

Visual Studio EXPERT SOLUTIONS FOR .NET DEVELOPERS

Running MapReduce Jobs (Linux)

demo



Hive



- Used by most BI products which connect to Hadoop
- Provides a SQL-like abstraction over Hadoop
 - Officially HiveQL, or HQL
- Works on own tables, but also on HBase
- Query generates MapReduce job (or Tez DAG), output of which becomes result set
- Microsoft has Hive ODBC driver
 - Connects Excel, Reporting Services, PowerPivot, Analysis Services *Tabular Mode* (only)





Hive demo

Visual Studio **LIVE!**
EXPERT SOLUTIONS FOR .NET DEVELOPERS

Spark



- Wildly popular open source project, focuses on distributed in-memory processing versus on-disk
- Can use it independently of Hadoop, but most people use it with Hadoop/HDFS
- Very popular component: Spark SQL
 - Allows HiveQL queries against Spark (Power BI can use this)
- Also: Spark Streaming, MLlib, GraphX
- Spark now supported on HDInsight Linux clusters as preview...

Visual Studio **LIVE!**
EXPERT SOLUTIONS FOR .NET DEVELOPERS

Spark on HDInsight



- HDInsight Spark clusters include the Jupyter “notebook” user interface
- Allows interspersal of text, code, and code output, including visualizations
- Supports Python (PySpark) and Scala
- Includes seven *very* helpful tutorial notebooks



Spark

demo



Power BI Against Hadoop and Spark

- Power BI can connect to HDFS directly
- Power BI can connect to HDInsight storage directly
 - Even if no cluster running
- Power BI can use Hive ODBC driver
- Power BI in the cloud can connect to Spark on HDInsight



PolyBase

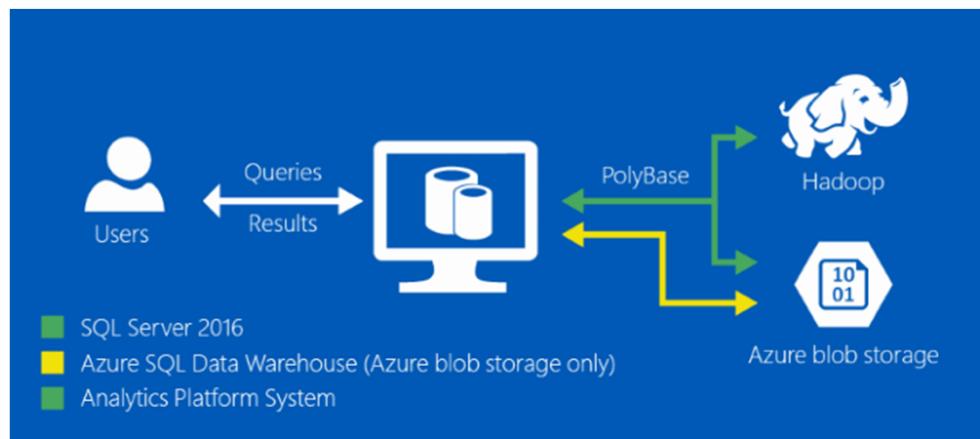


What's PolyBase?

- A “bridging” technology to connect SQL Server to data in Hadoop or Azure Blob Storage
- Makes the Hadoop data look like SQL Server data via “EXTERNAL” tables
- Query as normal; even join with physical tables
- First appeared in Parallel Data Warehouse/APS and Azure SQL DW
- Now included in SQL Server 2016 Enterprise
- Can create physical table with CREATE TABLE...AS SELECT... (CTAS)

Visual Studio **LIVE!**
EXPERT SOLUTIONS FOR .NET DEVELOPERS

PolyBase



Visual Studio **LIVE!**
EXPERT SOLUTIONS FOR .NET DEVELOPERS

Notes

- Data may be moved and processed by SQL Server's engine and optimizer, or may be “pushed down” to Hadoop, or both
- For DW versions of SQL, query is distributed
- Config can be tricky
- Java install is a prerequisite



Relevant T-SQL

- Prepatory:
 - EXEC sp_configure 'hadoop connectivity', x
 - RECONFIGURE;
 - CREATE MASTER KEY ENCRYPTION
- Next:
 - CREATE DATABASE SCOPED CREDENTIAL
 - CREATE EXTERNAL DATA SOURCE
 - CREATE EXTERNAL FILE FORMAT
 - CREATE EXTERNAL TABLE



PolyBase demo

Credit: SQL Server 2016 CTP 3 Samples



R Services



What's R?

- Designed as a sort of open source SPSS/SAS
- Hugely popular, huge ecosystem, tons of free extension packages
- A favorite of data scientists; extremely useful for predictive analytics
- Microsoft acquired Revolution Analytics, key commercial entity behind R (closed April, 2015)
- Microsoft is now integrating R across the entire stack. SQL Server R Services is probably the most noteworthy such integration



Some R Concepts



- Assignment and vectors

```
x <- c(10.4, 5.6, 3.1, 6.4, 21.7)
x[6]
```
- Vector operations

```
y <- mean(x)
z <- x+2
```
- Data frames
- Models
- Packages



Tooling

- Included with all distros: R and Rscript command line, Rgui
- Most Popular: Rstudio
 - Separate, open source download
- For the MS crowd: R Tools for Visual Studio



R Studio
demo



Classic R

Runs on a single workstation

Puts all data in memory

Good for speed

Bad for scale

Not easily operationalized or integrated by developers

Visual Studio **LIVE!**
EXPERT SOLUTIONS FOR .NET DEVELOPERS

Challenges with Embracing Open Source R

Data Movement



Moving data from the database to the R Runtime becomes painful as data volumes grow and carries security risks

Operationalization



How do I call the R script from my production application?

Scale / Performance



R runs single threaded and only accommodates datasets that fit into available memory

Slide source: Microsoft



Revolution Analytics

- Built a version of R that could run over a cluster of servers, instead of a physical workstation
- Integrated it with various MPP data warehouse products and Hadoop distributions
- Suddenly, R is massively parallel and doesn't need to move the data
- Commercial product, but:
 - Revolution had an open source distribution too
 - Team at Revolution very involved in open source R as committers



Different Versions

Revolution Analytics Name	Microsoft Name	Deployment/License
Revolution R Open (RRO)	Microsoft R Open (MRO)	Open Source/Free
Revolution R Enterprise (RRE)	Microsoft R Server	Commercial, licensed
Revolution R Enterprise DistributedR	Microsoft R Services	Embedded in SQL Server, and other products

- And don't forget, there is a “standard” open source R distribution



SQL Server R Services

- Integration of R in SQL Server
- Adaptation of Revolution R Enterprise (RRE)
- Manifests in two ways
 - Embed R in T-SQL using `sp_execute_external_script`
 - Use in Dynamic SQL or in stored procedures
 - Run R code on client but have tasks execute on SQL Server
 - Can use any client mentioned on previous slide
 - Using rxBlah equivalents to standard R functions



Integration Byproducts

- Any R Script can be wrapped in T-SQL
- So any R script can be called by any developer
- And all data stays right where it is, in SQL Server
- R can and will benefit from Columnstore indexes!



Packages, Plots

- External packages are available and loadable in to R Services.
- These include packages that plot and map data
- Resulting images can be returned as varbinary(max) and can be easily rendered in Reporting Services reports



R in SQL Server

demo

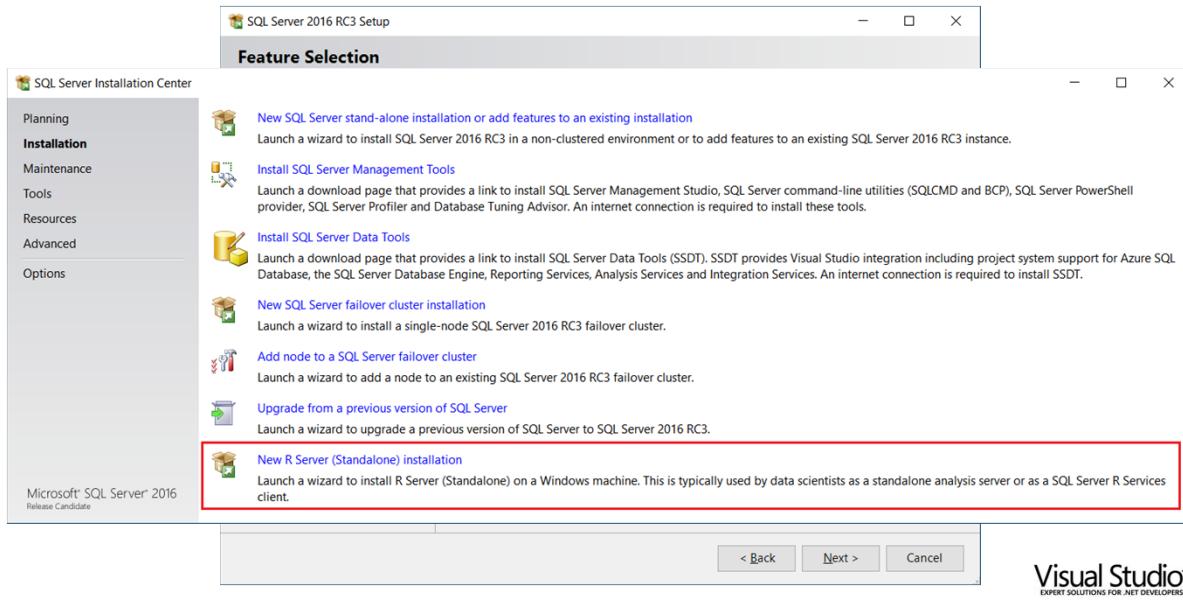


Scoring R Models in SQL Server

- R-based predictive models can be serialized into a binary string, and stored in a database
- Stored procedure can de-serialize and load the model, then score SQL Server data against it, all on the server
- So you can do operationalized predictive analytics in SQL Server, with no need to use another product

Visual Studio  EXPERT SOLUTIONS FOR .NET DEVELOPERS

Which R?



Visual Studio  EXPERT SOLUTIONS FOR .NET DEVELOPERS

Installing Packages

- Run `install.packages()` from one of the front-ends
- Make sure you're in the correct directory or the package won't be accessible!
 - For R Services, go to
 - `C:\Program Files\Microsoft SQL Server\MSSQL13.MSSQLSERVER\R_SERVICES`
 - NOT
 - `C:\Program Files\Microsoft SQL Server\130\R_SERVER`
- You might also have R Open:
 - `C:\Program Files\Microsoft\MRO\R-3.2.4`



R Data Sources and Visualizations in Power BI

- These features leverage R scripts and translation between selected columns and an R DataFrame
- R Script produces a Dataset when used as a source
- R script consumes, manipulates and plots data
- Only works on Windows desktop



Other R Integrations in the MS Stack

- Azure Machine Learning
- HDInsight
- Visual Studio (as mentioned)
- Also: DeployR integration with Excel
 - Offered by Revolution Analytics pre-aquisition



OTHER COMPONENTS



Integration Services (SSIS)

- Microsoft's ETL (Extract, Transform and Load) tool
- Stable, mature, reliable
- New: Support for Hadoop/HDFS
- New: Support for SQL Server R Services
 - Including data extract/load; model build/run/train
- New: Support for SSAS Tabular mode
- New: Connector for SAP BW
- Also: Look for SSIS Azure Feature Pack in SQL Server 2016 Feature Pack



Master Data Services (MDS): Microsoft's Master Data Management (MDM) tool

- Examples:
 - Sales states, countries, currencies, customer types
 - Customers, products
 - Think of “lookup tables” or just think of dimensions!
 - Slowly changing non-transactional entities in your data
- What gets stored:
 - Schemas
 - Any hierarchies
 - The data!
- Other features:
 - Collections, business rules, security, workflows
 - Versioning
- Lots of new investment in 2016!



Data Quality Services (DQS)

- Data Cleansing Tool
- New to SQL Server 2012
- Result of Zoomix acquisition
- Uses Artificial Intelligence algorithms to detect invalid data and perform matching (for de-duplication)
- Allows manual intervention, too
- Can integrate with MDS and SSIS
- Cleaner data = better adoption of your BI project



StreamInsight

- Microsoft's Complex Event Handling (CEP) Product
- Processes data streams that are fast and high-volume
- Highly parallel C# code assures low latency, high throughput
- Not based on SQL Server, though that is its “ship vehicle”



Putting it All Together

- Columnstore indexes with SSAS Tabular or Power BI
- Columnstore indexes with R
- PolyBase with Hadoop
- Analysis Services and Reporting Services
- Reporting Services and Power BI (*)
- Power BI and Hadoop, or Spark
- R and SQL Server, Reporting Services, Power BI and Spark



PART IV Beyond Relational



2016

JSON

Visual Studio  EXPERT SOLUTIONS FOR .NET DEVELOPERS

2016

Built-In JSON Support

- Capabilities
 - Format and export JSON from relational queries
 - Store and query JSON inside the database
- Conceptually similar to XML support
 - Simpler model
 - No native “json” data type; uses nvarchar(max)
- Why no native type?
 - Easier migration to leave json columns as ordinary string types
 - Cross-feature compatibility (e.g., Hekaton, temporal)
- No custom JSON indexes
 - Optimize JSON queries using standard indexes
 - Create computed columns over desired properties, and then index the computed columns

Visual Studio  EXPERT SOLUTIONS FOR .NET DEVELOPERS

2016

Bidirectional JSON Transformation

Number	Date	Customer	Price	Quantity
SO43659	2011-05-31T00:00:00	MSFT	59.99	1
SO43661	2011-06-01T00:00:00	Nokia	24.99	3

Visual Studio **LIVE!**
EXPERT SOLUTIONS FOR .NET DEVELOPERS

2016

Bidirectional JSON Transformation

```
[  
  {  
    "Number": "SO43659",  
    "Date": "2011-05-31T00:00:00"  
    "AccountNumber": "AW29825",  
    "Price": 59.99,  
    "Quantity": 1  
  },  
  {  
    "Number": "SO43661",  
    "Date": "2011-06-01T00:00:00"  
    "AccountNumber": "AW73565",  
    "Price": 24.99,  
    "Quantity": 3  
  }  
]
```

Number	Date	Customer	Price	Quantity
SO43659	2011-05-31T00:00:00	MSFT	59.99	1
SO43661	2011-06-01T00:00:00	Nokia	24.99	3

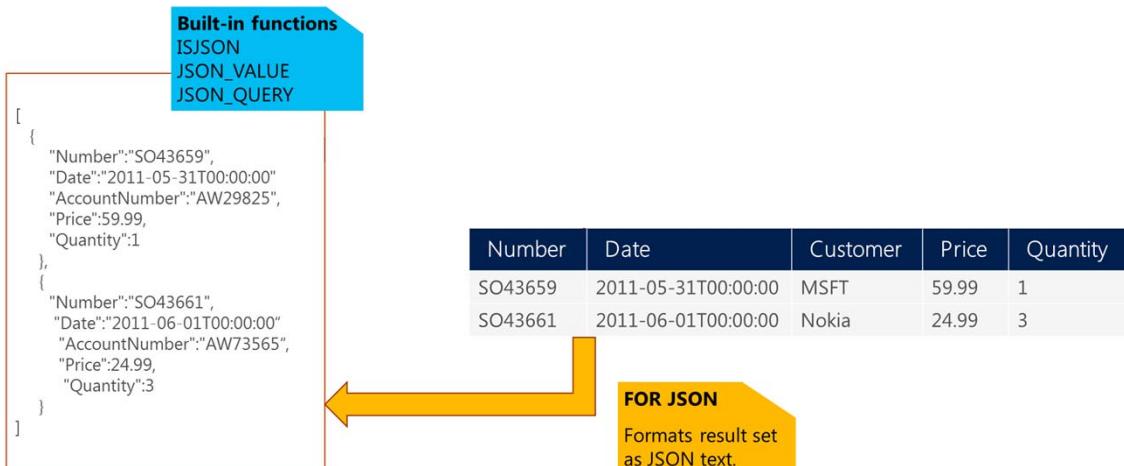
FOR JSON

Formats result set
as JSON text.

Visual Studio **LIVE!**
EXPERT SOLUTIONS FOR .NET DEVELOPERS

2016

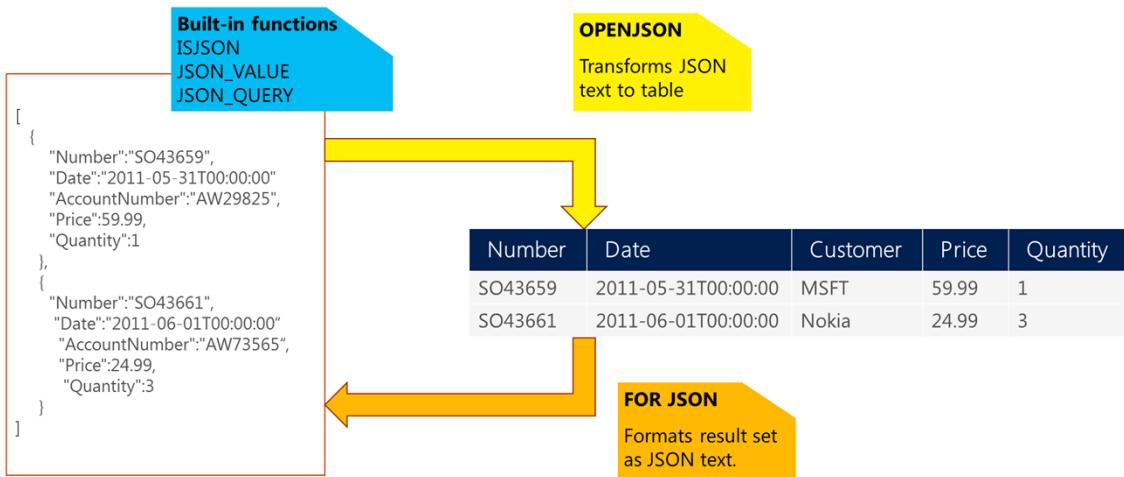
Bidirectional JSON Transformation



Visual Studio **LIVE!**
EXPERT SOLUTIONS FOR .NET DEVELOPERS

2016

Bidirectional JSON Transformation



Visual Studio **LIVE!**
EXPERT SOLUTIONS FOR .NET DEVELOPERS

2016

FOR JSON Clause

- Append to SELECT statements to generate results in JSON format
 - Example:
SELECT * FROM Customer FOR JSON AUTO
- FOR JSON AUTO
 - Creates nested structure based on table hierarchy
- FOR JSON PATH
 - Creates nested structure based on column aliases



2016

FOR JSON Formatting Options

- WITHOUT_ARRAY_WRAPPER
 - Don't generate [] syntax (single JSON object)
- ROOT
 - Generate single root wrapper object around the results
- INCLUDE_NULL_VALUES
 - Generate properties for NULL columns



2016

Generating JSON *demo*

Visual Studio **LIVE!**
EXPERT SOLUTIONS FOR .NET DEVELOPERS

2016

Built-in JSON Functions

- ISJSON
 - Validates for well-formed JSON
 - Use in check constraints for NVARCHAR columns containing JSON
- JSON_QUERY
 - Queries by path expression and returns a nested object/array
 - Similar to *xml.query*
- JSON_VALUE
 - Queries by path expression and returns a scalar value
 - Similar to *xml.value*
- No JSON “DML”
 - Cannot directly modify JSON content
 - No equivalent to *xml.modify*

Visual Studio **LIVE!**
EXPERT SOLUTIONS FOR .NET DEVELOPERS

2016

JSON Path Expressions

- Reference JSON properties using a JavaScript-like syntax

Syntax	Description
\$	References the entire JSON object
\$.property1	References a top-level property in the JSON object
[\$5]	References the sixth element in the JSON array
\$.property1.property2 .array1[5].property3 .array2[15].property4	References a complex nested property in the JSON object



2016

JSON Query Example

```
SELECT
    Id,
    OrderNumber,
    OrderDate,
    JSON_VALUE(OrderDetails, '$.Order.ShipDate')
FROM
    SalesOrderRecord
WHERE
    ISJSON(OrderDetails) AND
    JSON_VALUE(OrderDetails, '$.Order.Type') = 'C'
```



2016

Storing and Querying JSON

demo

Visual Studio  EXPERT SOLUTIONS FOR .NET DEVELOPERS

2016

Shredding JSON

- OPENJSON table-valued function (TVF)
 - Provides a rowset view over a JSON document
 - Shreds single JSON document into multiple rows
- What does it do?
 - Iterates through objects (if JSON array) or properties (if JSON object)
 - Generates a row for each object/property with key, value, and type
- Discoverable schema
 - Key, value, and type columns
- Explicit schema
 - Include columns, data types, and property-to-column mapping rules

Visual Studio  EXPERT SOLUTIONS FOR .NET DEVELOPERS

2016

Shredding JSON *demo*

Visual Studio  EXPERT SOLUTIONS FOR .NET DEVELOPERS

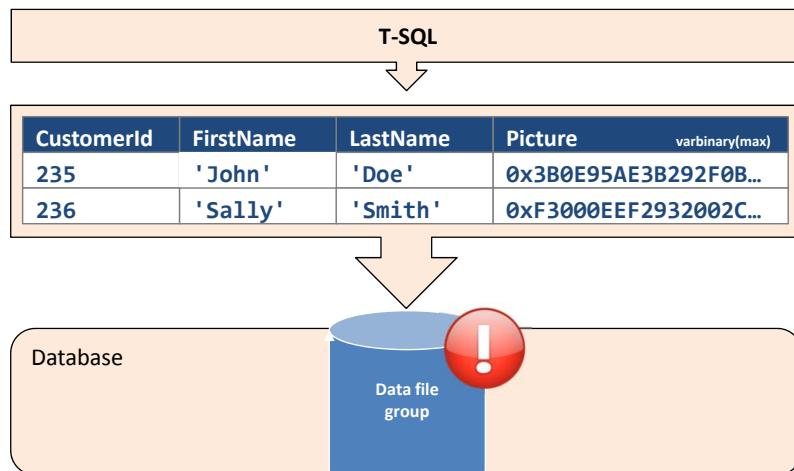
2008+

FILESTREAM

Visual Studio  EXPERT SOLUTIONS FOR .NET DEVELOPERS

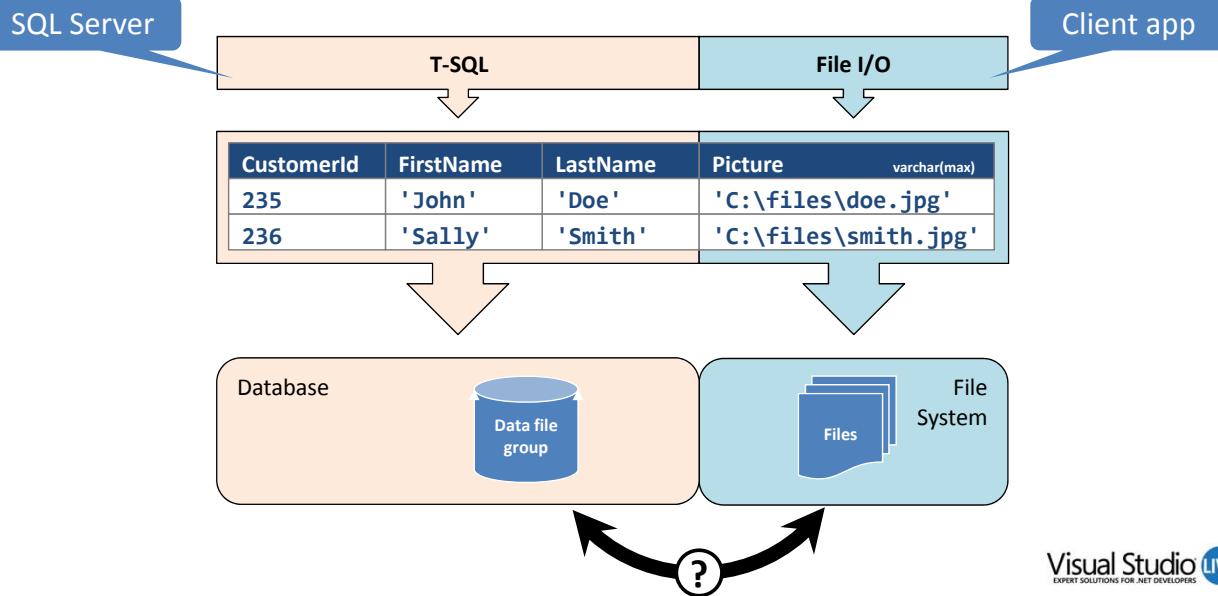
2008+

BLOBs in the Database

Visual Studio **LIVE!**
EXPERT SOLUTIONS FOR .NET DEVELOPERS

2008+

BLOBs Outside the Database

Visual Studio **LIVE!**
EXPERT SOLUTIONS FOR .NET DEVELOPERS

2008+

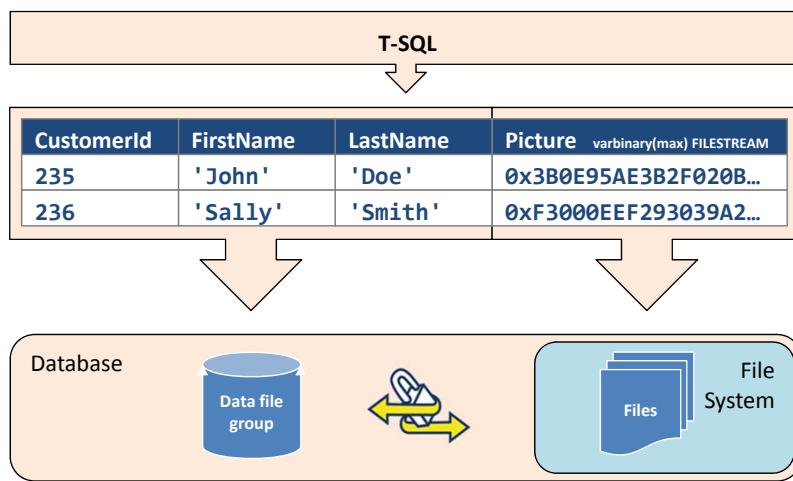
BLOBs Using FILESTREAM

- Transparently store varbinary(max) data in the file system
 - Declare column as “varbinary(max) FILESTREAM”
 - File system is optimized for storing and streaming BLOBS
- Integrated management
 - BLOBS are logically part of the database (backup, restore, etc.)
 - But they are stored physically separate as a file group mapped to the file system
- Simplified programming
 - Just use T-SQL, or the streaming API
 - SQL Server transparently links rows in relational tables to BLOBS in the file system
- Transactional
 - SQL Server integrates with the NTFS file system
 - Database transactions wrap NTFS transactions

Visual Studio  EXPERT SOLUTIONS FOR .NET DEVELOPERS

2008+

BLOBs Using FILESTREAM (T-SQL)



Visual Studio  EXPERT SOLUTIONS FOR .NET DEVELOPERS

2008+

Enabling FILESTREAM

- FILESTREAM is disabled by default
 - Can be enabled for T-SQL or T-SQL + streaming API access
- To use FILESTREAM, it must be enabled twice
 - By the Windows administrator
 - Installs a file system filter driver
 - By the SQL Server administrator
- Set the same access level each time
 - Windows and SQL admins must agree!

Visual Studio  EXPERT SOLUTIONS FOR .NET DEVELOPERS

2008+

Determining the Access Level

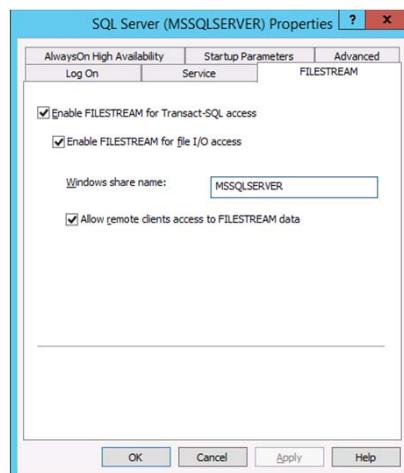
- Disabled
 - FILESTREAM is not enabled
- T-SQL
 - FILESTREAM is enabled for T-SQL access
 - Complete abstraction
- T-SQL + file system I/O (local)
 - FILESTREAM is enabled for both T-SQL access and streamed file system access
 - Streaming API access available only to client applications running on the local server
- T-SQL + file system I/O (remote)
 - Streaming API access available to client applications running anywhere on the network

Visual Studio  EXPERT SOLUTIONS FOR .NET DEVELOPERS

2008+

Enabling FILESTREAM (Windows)

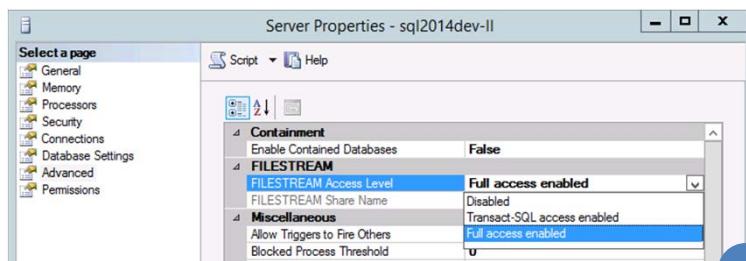
- Enable it either:
 - During setup
 - With SQL Server Configuration Manager
- Set to desired access level
 - Disabled
 - T-SQL only
 - T-SQL + file I/O
 - local only
 - remote
- Can't script with T-SQL

Visual Studio LIVE!
EXPERT SOLUTIONS FOR .NET DEVELOPERS

2008+

Enabling FILESTREAM (SQL Server)

- Enable it in SSMS:
 - Server Properties dialog



- Or, enable it with T-SQL
 - `EXEC sp_configure filestream_access_level, n
RECONFIGURE`

n =
0: disabled
1: T-SQL only
2: T-SQL + file I/O

Visual Studio LIVE!
EXPERT SOLUTIONS FOR .NET DEVELOPERS

2008+

Preparing the Database for FILESTREAM

- Create a FILESTREAM filegroup
 - For a new database, using CREATE DATABASE
 - For an existing database, using ALTER DATABASE
- Add a FILESTREAM container to the filegroup
 - Point to a path on the NTFS file system
 - Contents of each varbinary(max) FILESTREAM column will be stored as a distinct file in this path
- Use multiple filegroups and containers
 - Load balance the BLOB load across multiple disks
 - Can assign different tables to different filegroups

Visual Studio  EXPERT SOLUTIONS FOR .NET DEVELOPERS

2008+

Creating a FILESTREAM-enabled Database

```
CREATE DATABASE PhotoLibrary
ON P Must exist Must not exist
    (NAME = PhotoLibrary_data,
     FILENAME = 'C:\DB\PhotoLibrary_data.mdf'),
FILE Must exist Must not exist GROUP1 CONTAINS FILESTREAM
    (NAME = PhotoLibrary_photos,
     FILENAME = 'C:\DB\Photos')
LOG Must exist Must not exist
    (NAME = PhotoLibrary_log,
     FILENAME = 'C:\DB\PhotoLibrary_log.ldf')
```

Visual Studio  EXPERT SOLUTIONS FOR .NET DEVELOPERS

2008+

Creating FILESTREAM Columns

- Define BLOB columns as “varbinary(max) FILESTREAM”
 - Multiple BLOB columns are permitted per table
- Table requires ROWGUIDCOL column
 - Attribute applied to a uniqueidentifier (GUID) column
 - Must be primary key or have unique constraint
 - Cannot be NULL
- Access using standard T-SQL
 - Back-end NTFS file system storage is completely transparent
 - Inline short BLOBs
 - Import from files using OPENROWSET with SINGLE_BLOB option



2008+

Creating FILESTREAM Columns

```
CREATE TABLE Product(  
    ProductId int IDENTITY PRIMARY KEY,  
    BlobId uniqueidentifier ROWGUIDCOL NOT NULL UNIQUE  
    Summary varchar(max),  
    Photo varbinary(max) FILESTREAM)
```

Required ID

BLOB



2008+

Getting Started with FILESTREAM

demo

Visual Studio **LIVE!**
EXPERT SOLUTIONS FOR .NET DEVELOPERS

2008+

Storage vs. Access

- FILESTREAM solves the *storage* scalability problem
 - But what about *access*?
- T-SQL works
 - But is it optimal?

`SELECT * FROM PhotoAlbum`

Results		Messages	
PhotoId	RowId	PhotoDescription	Photo
1	FB9501B0-E580-E411-80C0-080027729686	Text file	0x31303A333020504D2031322F31302F32303134
2	6E1262E-E680-E411-80C0-080027729686	Document icon	0x4749463839610C000E00B30000FFFFFC6DEC6C0C0C000...
3	92629D68-E680-E411-80C0-080027729686	Mountains	0xFFD8FFE000104A46494600010201004800480000FFED0A9...

- SQL Server needs to perform file I/O
 - Must process entire file contents in memory

Visual Studio **LIVE!**
EXPERT SOLUTIONS FOR .NET DEVELOPERS

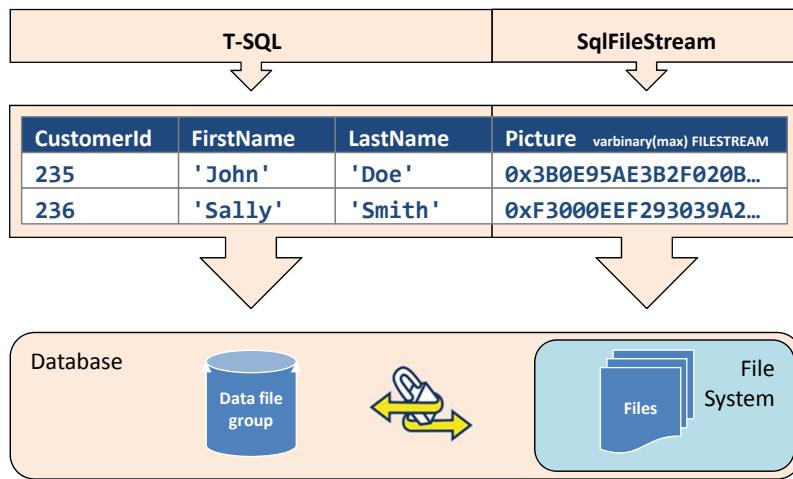
2008+

Introducing SqlFileStream

- Build a .NET streaming client using SqlFileStream
 - Wraps OpenSqlFilestream SQL Server native client API
- Inherits from System.IO.Stream
 - Use standard .NET stream coding patterns
- Included in System.Data.dll
 - No additional references needed
- Use T-SQL to read/write all *non*-BLOB data
 - Use SqlFileStream to read/write all BLOB data
 - Safe and direct streaming access to the SQL Server-controlled file system
- Requires a database transaction
 - Automatically wraps an NTFS file system transaction

2008+

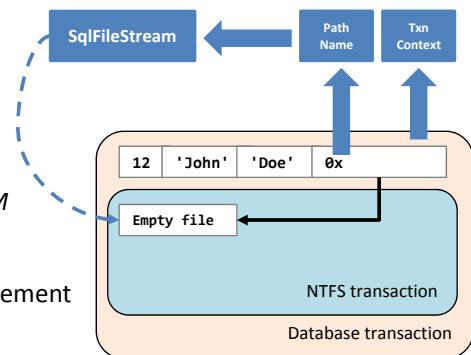
BLOBs Using SqlFileStream



2008+

Storing BLOBs with SqlFileStream

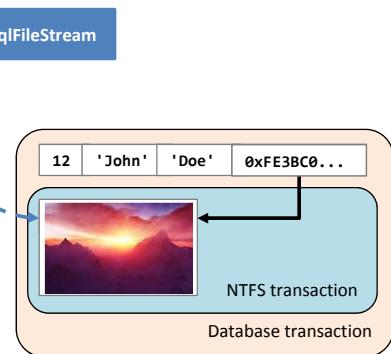
- Start a database transaction
- INSERT new row
 - Store a zero-length binary value for the BLOB column(s)
- Retrieve BLOB path name and transaction context
 - Call `PathName()` function on `varbinary(max) FILESTREAM` column
 - Call `GET_FILESTREAM_TRANSACTION_CONTEXT`
 - Obtain values using an OUTPUT clause in the INSERT statement
- Instantiate `SqlFileStream` object
 - Pass BLOB path name and transaction context to constructor

Visual Studio LIVE!
EXPERT SOLUTIONS FOR .NET DEVELOPERS

2008+

Storing BLOBs with SqlFileStream

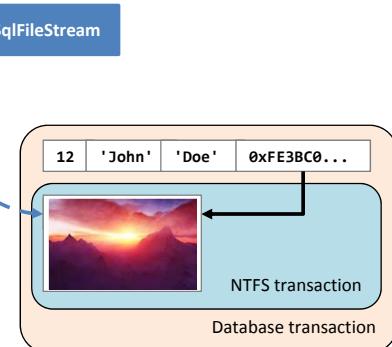
- Start a database transaction
- INSERT new row
 - Store a zero-length binary value for the BLOB column(s)
- Retrieve BLOB path name and transaction context
 - Call `PathName()` function on `varbinary(max) FILESTREAM` column
 - Call `GET_FILESTREAM_TRANSACTION_CONTEXT`
 - Obtain values using an OUTPUT clause in the INSERT statement
- Instantiate `SqlFileStream` object
 - Pass BLOB path name and transaction context to constructor
- Write to the stream
 - Then close it

Visual Studio LIVE!
EXPERT SOLUTIONS FOR .NET DEVELOPERS

2008+

Storing BLOBs with SqlFileStream

- Start a database transaction
- INSERT new row
 - Store a zero-length binary value for the BLOB column(s)
- Retrieve BLOB path name and transaction context
 - Call `PathName()` function on `varbinary(max) FILESTREAM` column
 - Call `GET_FILESTREAM_TRANSACTION_CONTEXT`
 - Obtain values using an OUTPUT clause in the INSERT statement
- Instantiate `SqlFileStream` object
 - Pass BLOB path name and transaction context to constructor
- Write to the stream
 - Then close it
- Commit the database transaction
 - Automatically commits the NTFS file system transaction

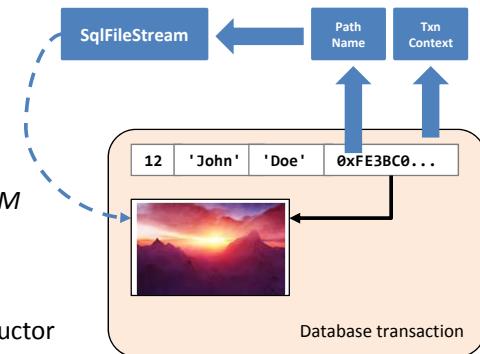


Visual Studio **LIVE!**
EXPERT SOLUTIONS FOR .NET DEVELOPERS

2008+

Retrieving BLOBs with SqlFileStream

- Start a database transaction
- SELECT existing row
 - Don't include the BLOB column(s)
- Retrieve BLOB path name and transaction context
 - Call `PathName()` function on `varbinary(max) FILESTREAM` column
 - Call `GET_FILESTREAM_TRANSACTION_CONTEXT`
- Instantiate `SqlFileStream` object
 - Pass BLOB path name and transaction context to constructor
- Read from the stream
 - Then close it
- Commit the database transaction



Visual Studio **LIVE!**
EXPERT SOLUTIONS FOR .NET DEVELOPERS

2008+

Using SqlFileStream

demo

Visual Studio  EXPERT SOLUTIONS FOR .NET DEVELOPERS

2008+

Improving I/O Scalability

- Distribute FILESTREAM storage by table
 - Each table (or table partition) can be assigned to a different filegroup
 - Use FILESTREAM_ON clause in CREATE TABLE statement
 - Create each filegroup's container on a separate disk volume
- Distribute FILESTREAM storage within a table
 - Partition the table
 - Use FILESTREAM_ON clause to assign a different filegroup to each partition
 - Use multiple containers 2012+
 - Single filegroup for the table will access its containers in round-robin fashion

Visual Studio  EXPERT SOLUTIONS FOR .NET DEVELOPERS

2008+

Multiple FILESTREAM Filegroups

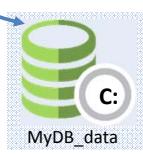
- CREATE DATABASE MyDB
ON PRIMARY
(NAME = MyDB_data, FILENAME = 'C:\DB\MyDB_data.mdf'),
FILEGROUP MyDB_docs CONTAINS FILESTREAM DEFAULT
(NAME = MyDB_docs1, FILENAME = 'D:\DB\MyDB_docs'),
FILEGROUP MyDB_photos CONTAINS FILESTREAM
(NAME = MyDB_photos1, FILENAME = 'E:\DB\MyDB_photos'), [2012+]
(NAME = MyDB_photos2, FILENAME = 'F:\DB\MyDB_photos'),
LOG ON
(NAME = MyDB_log, FILENAME = 'X:\DB\MyDB_log.ldf')

Visual Studio  EXPERT SOLUTIONS FOR .NET DEVELOPERS

2008+

Multiple FILESTREAM Filegroups

- CREATE DATABASE MyDB
ON PRIMARY
(NAME = MyDB_data, FILENAME = 'C:\DB\MyDB_data.mdf'),
FILEGROUP MyDB_docs CONTAINS FILESTREAM DEFAULT
(NAME = MyDB_docs1, FILENAME = 'D:\DB\MyDB_docs'),
FILEGROUP MyDB_photos CONTAINS FILESTREAM
(NAME = MyDB_photos1, FILENAME = 'E:\DB\MyDB_photos'), [2012+]
(NAME = MyDB_photos2, FILENAME = 'F:\DB\MyDB_photos'),
LOG ON
(NAME = MyDB_log, FILENAME = 'X:\DB\MyDB_log.ldf')

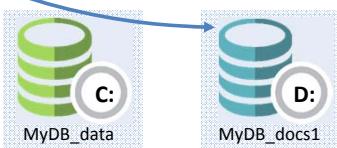


Visual Studio  EXPERT SOLUTIONS FOR .NET DEVELOPERS

2008+

Multiple FILESTREAM Filegroups

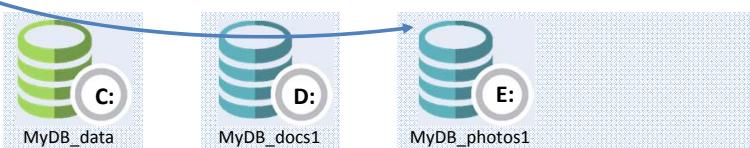
- CREATE DATABASE MyDB
ON PRIMARY
(NAME = MyDB_data, FILENAME = 'C:\DB\MyDB_data.mdf'),
FILEGROUP MyDB_docs CONTAINS FILESTREAM DEFAULT
(NAME = MyDB_docs1, FILENAME = 'D:\DB\MyDB_docs'),
FILEGROUP MyDB_photos CONTAINS FILESTREAM
(NAME = MyDB_photos1, FILENAME = 'E:\DB\MyDB_photos'),
(NAME = MyDB_photos2, FILENAME = 'F:\DB\MyDB_photos'),
LOG ON
(NAME = MyDB_log, FILENAME = 'X:\DB\MyDB_log.ldf')

Visual Studio LIVE!
EXPERT SOLUTIONS FOR .NET DEVELOPERS

2008+

Multiple FILESTREAM Filegroups

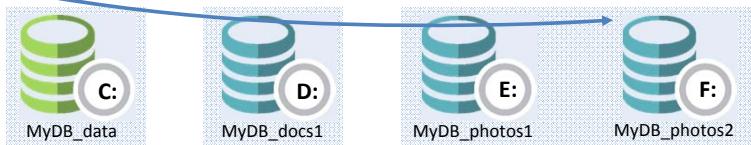
- CREATE DATABASE MyDB
ON PRIMARY
(NAME = MyDB_data, FILENAME = 'C:\DB\MyDB_data.mdf'),
FILEGROUP MyDB_docs CONTAINS FILESTREAM DEFAULT
(NAME = MyDB_docs1, FILENAME = 'D:\DB\MyDB_docs'),
FILEGROUP MyDB_photos CONTAINS FILESTREAM
(NAME = MyDB_photos1, FILENAME = 'E:\DB\MyDB_photos'),
(NAME = MyDB_photos2, FILENAME = 'F:\DB\MyDB_photos'),
LOG ON
(NAME = MyDB_log, FILENAME = 'X:\DB\MyDB_log.ldf')

Visual Studio LIVE!
EXPERT SOLUTIONS FOR .NET DEVELOPERS

2008+

Multiple FILESTREAM Filegroups

- CREATE DATABASE MyDB
ON PRIMARY
(NAME = MyDB_data, FILENAME = 'C:\DB\MyDB_data.mdf'),
FILEGROUP MyDB_docs CONTAINS FILESTREAM DEFAULT
(NAME = MyDB_docs1, FILENAME = 'D:\DB\MyDB_docs'),
FILEGROUP MyDB_photos CONTAINS FILESTREAM
(NAME = MyDB_photos1, FILENAME = 'E:\DB\MyDB_photos'),
(NAME = MyDB_photos2, FILENAME = 'F:\DB\MyDB_photos'),
LOG ON
(NAME = MyDB_log, FILENAME = 'X:\DB\MyDB_log.ldf')



Visual Studio EXPERT SOLUTIONS FOR .NET DEVELOPERS

2008+

Multiple FILESTREAM Filegroups

- CREATE DATABASE MyDB
ON PRIMARY
(NAME = MyDB_data, FILENAME = 'C:\DB\MyDB_data.mdf'),
FILEGROUP MyDB_docs CONTAINS FILESTREAM DEFAULT
(NAME = MyDB_docs1, FILENAME = 'D:\DB\MyDB_docs'),
FILEGROUP MyDB_photos CONTAINS FILESTREAM
(NAME = MyDB_photos1, FILENAME = 'E:\DB\MyDB_photos'),
(NAME = MyDB_photos2, FILENAME = 'F:\DB\MyDB_photos'),
LOG ON
(NAME = MyDB_log, FILENAME = 'X:\DB\MyDB_log.ldf')



Visual Studio EXPERT SOLUTIONS FOR .NET DEVELOPERS

2008+

Multiple FILESTREAM Filegroups

```
CREATE TABLE Candidate(
    CandidateId int IDENTITY PRIMARY KEY,
    BlobId uniqueidentifier ROWGUIDCOL NOT NULL UNIQUE,
    Position varchar(max),
    Resume varbinary(max) FILESTREAM)
FILESTREAM_ON MyDB_docs
```



```
CREATE TABLE Product(
    ProductId int IDENTITY PRIMARY KEY,
    BlobId uniqueidentifier ROWGUIDCOL NOT NULL UNIQUE,
    ProductDescription varchar(max),
    Photo varbinary(max) FILESTREAM)
FILESTREAM_ON MyDB_photos
```



2012+

Visual Studio LIVE!

2008+

Limitations and Considerations

- Database Snapshots
 - Not supported for FILESTREAM filegroups
- Integrated security required for streaming API
 - SQL Server authentication not supported
- Transparent Data Encryption (TDE)
 - Supported, but won't encrypt files
- Mirroring
 - Not supported; use AlwaysOn
- NTFS file system
 - SMB network attached storage not supported
- Replication and log shipping
 - All participating servers must be running SQL Server 2008 or higher
 - For replication, the ROWGUIDCOL column must have a NEWSEQUENTIALID or NEWID default
- SQL Server Express edition
 - Fully supported
 - Database size limit (10GB) does not include FILESTREAM data

Visual Studio LIVE!

2008+

Best Practices

- Don't use FILESTREAM with very small BLOBS (< 1 MB)
 - That's just overkill
- For larger BLOBS, use SqlFileStream
 - T-SQL is OK for smaller BLOBS, but SQL Server memory resources take a hit for larger ones
- If reads require only the first few bytes, use T-SQL with SUBSTRING
- Disable short (8.3) filenames
 - fsutil behavior set disable8dot3 1
- Disable last access time
 - fsutil behavior set disablelastaccess 1
- Tweak cluster size if you have predictable size patterns
 - Tradeoff between disk space (larger clusters) and fragmentation + I/O (smaller clusters)
 - format F: /FS:NTFS /V:MyFILESTREAMVolume /A:64K
- Defragment periodically



2008+

More Information

- Pluralsight
 - SQL Server 2012-2014 Native File Streaming
- The Art of SQL Server FILESTREAM
 - <http://www.amazon.com/The-Art-SQL-Server-FILESTREAM/dp/1906434891>
- Full-text search (FTS)
 - <http://msdn.microsoft.com/en-us/library/ms142571.aspx>
- Semantic search
 - <http://msdn.microsoft.com/en-us/library/gg492075.aspx>
- WCF streaming with MTOM
 - [http://msdn.microsoft.com/en-us/library/ms733742\(v=vs.110\).aspx](http://msdn.microsoft.com/en-us/library/ms733742(v=vs.110).aspx)
- OpenSqlFilestream
 - <http://msdn.microsoft.com/en-us/library/bb933972.aspx>



2008+

hierarchyid

Visual Studio  EXPERT SOLUTIONS FOR .NET DEVELOPERS

2008+

What is hierarchyid?

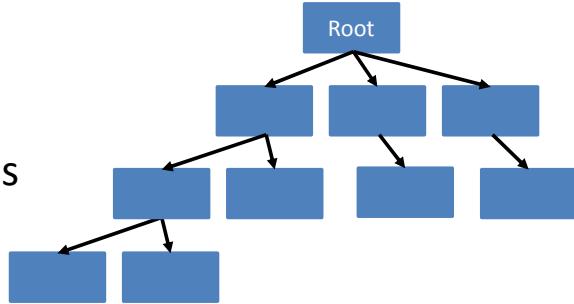
- System CLR data type
 - Extremely compact variable-length binary format
- Enables a robust hierarchical structure over a self-joining table
 - Each row is a node with a unique hierarchyid value
 - Contains the path in the hierarchy to the node... down to the sibling ordinal position
- Invoke methods in T-SQL
 - Efficiently query the hierarchy
 - Arbitrarily insert, modify, and delete nodes
 - Reparent entire sub-trees with a single update

Visual Studio  EXPERT SOLUTIONS FOR .NET DEVELOPERS

2008+

Hierarchical Storage Scenarios

- File system
- Product categories
- Business organization charts
- Forum and mailing list threads
- Content management
- Many more...
 - Unlimited breadth and depth
 - Recursive iteration



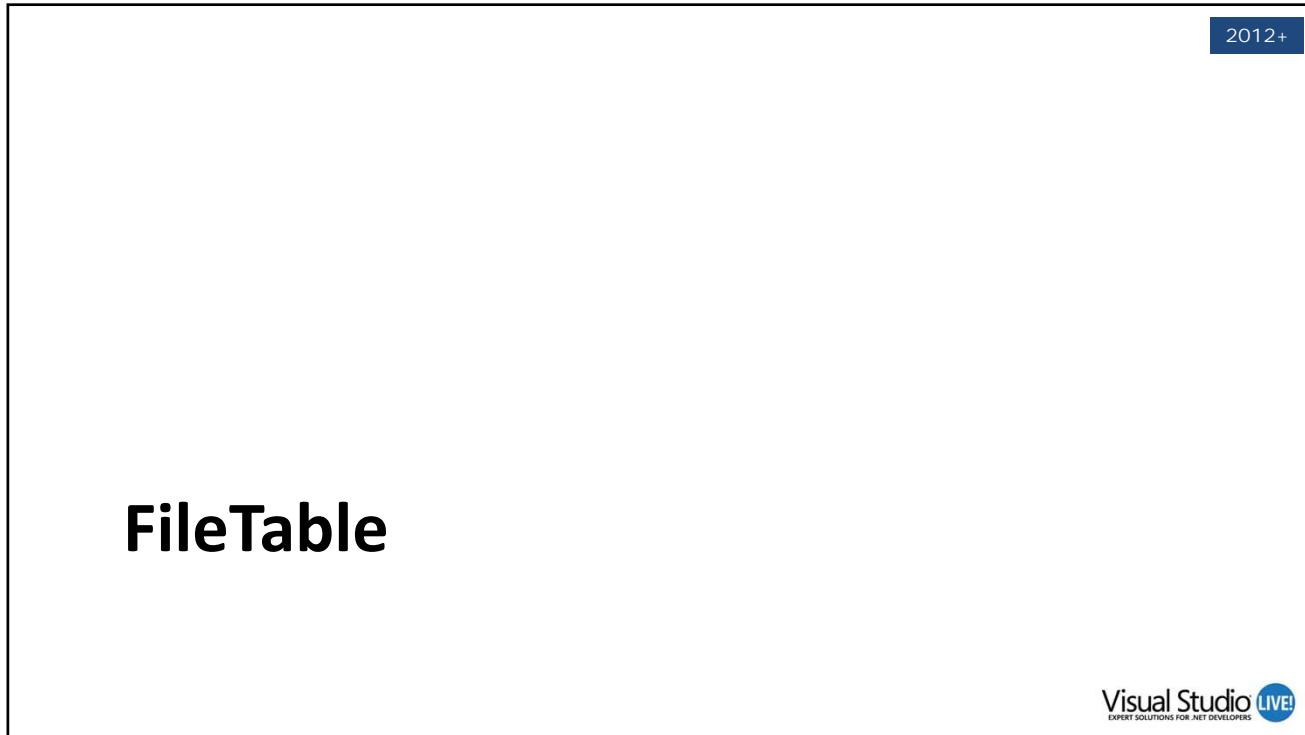
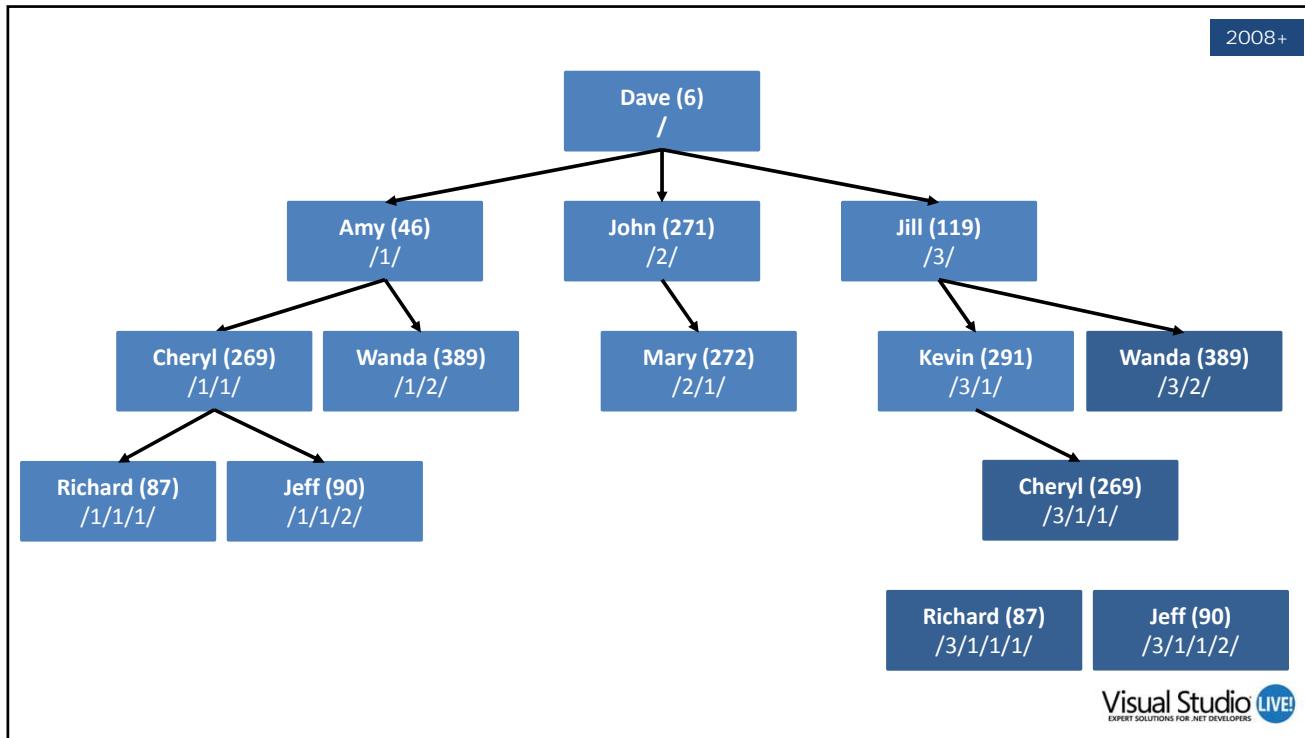
Visual Studio EXPERT SOLUTIONS FOR .NET DEVELOPERS

2008+

Common hierarchyid Methods

- GetAncestor
 - Find a node's parent
- GetDescendant
 - Generate a new child node
- GetLevel
 - Return node depth
- GetReparentedValue
 - Move a node
- GetRoot
 - Reference the root node
- IsDescendantOf
 - Return a subtree
- Parse
 - Convert string format to binary value
- ToString
 - Convert binary value to string format

Visual Studio EXPERT SOLUTIONS FOR .NET DEVELOPERS



2012+

Introducing FileTable

- FILESTREAM is great, but
 - It's *purely* a storage abstraction layer
 - Accessible only to developers and admins
 - The back-end file system is obfuscated and not accessible to users
- Introducing FileTable
 - It's an ordinary table
 - Each row represents either a file or folder
 - It's a functional file system
 - The entire table surfaces as a Windows file share
- FileTable =
 - FILESTREAM + hierarchyid + Windows file system API



2012+

FileTable Schema

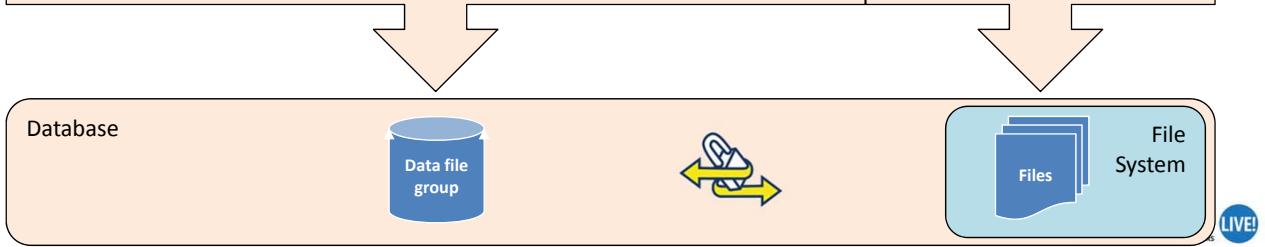
Column Name	Data Type	Description
stream_id	uniqueidentifier ROWGUIDCOL	Unique row identifier
file_stream	varbinary(max) FILESTREAM	BLOB content (NULL if directory)
name	nvarchar(255)	Name of file or directory
path_locator	hierarchyid	Location of file or directory within the file system hierarchy
creation_time	datetimeoffset(7)	Created
last_write_time	datetimeoffset(7)	Last modified
last_access_time	datetimeoffset(7)	Last accessed
is_directory	bit	0 = file, 1 = directory
is_offline	bit	Storage attributes
is_hidden	bit	
is_READONLY	bit	
is_ARCHIVE	bit	
is_SYSTEM	bit	
is_TEMPORARY	bit	



2012+

FileTable Access

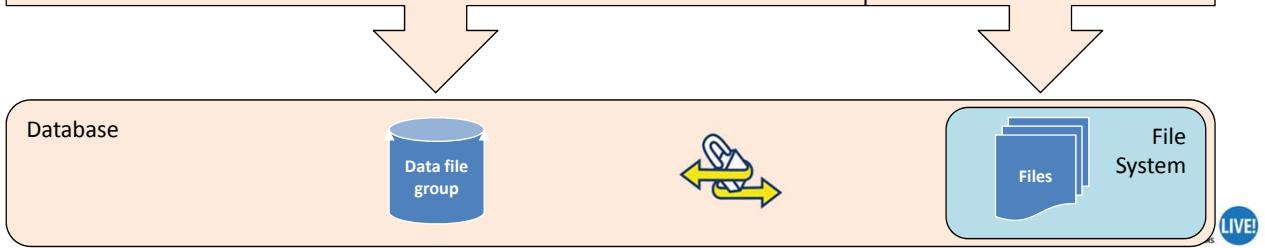
stream_id	name	path_locator	is_directory	...	file_stream
27D8D4AD-D100-39...	'Financials'	0xFF271A3562...	1	...	NULL
78F603CC-0460-73...	'ReadMe.docx'	0xFF59345688...	0	...	0x3B0E956636AE3B2F020B...
207D4A96-E854-01...	'Budget.xlsx'	0xFD0011039A...	0	...	0xF3F359000EEF293039A2...



2012+

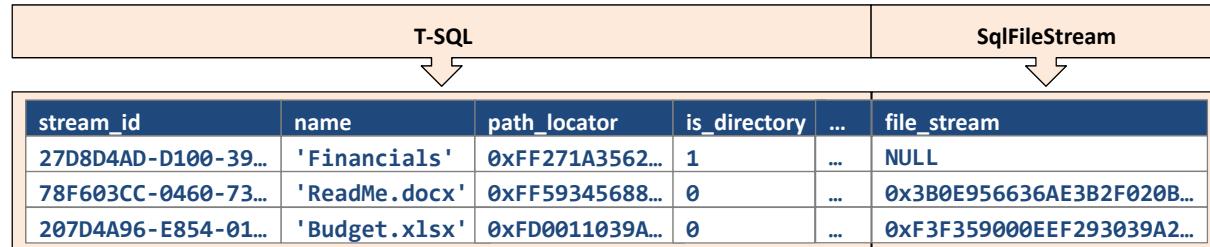
FileTable Access via T-SQL

stream_id	name	path_locator	is_directory	...	file_stream
27D8D4AD-D100-39...	'Financials'	0xFF271A3562...	1	...	NULL
78F603CC-0460-73...	'ReadMe.docx'	0xFF59345688...	0	...	0x3B0E956636AE3B2F020B...
207D4A96-E854-01...	'Budget.xlsx'	0xFD0011039A...	0	...	0xF3F359000EEF293039A2...



2012+

FileTable Access via SqlFileStream



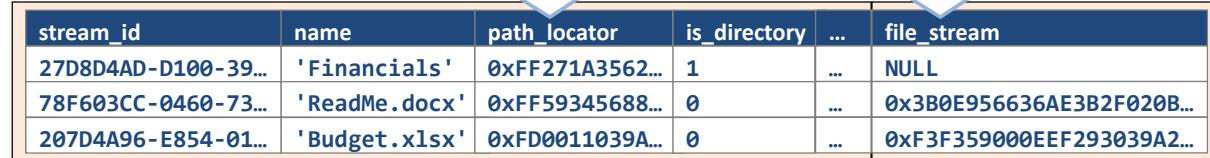
Database



LIVE!

FileTable Access via Windows File System

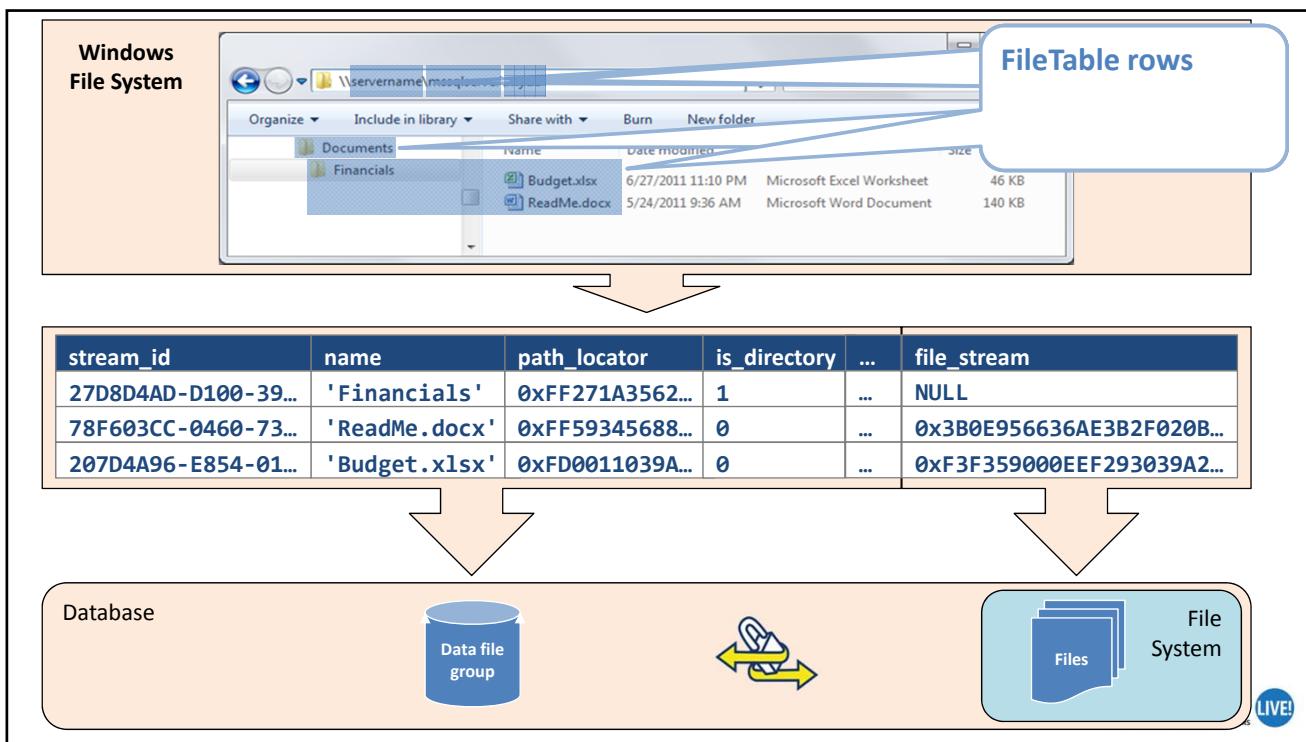
hierarchyid

varbinary(max)
FILESTREAM

Database



LIVE!



2012+

FileTable Prerequisites

- Prerequisites at the instance level
 - FILESTREAM must be enabled for streaming API access
- Prerequisites at the database level
 - Must have a FILESTREAM filegroup and container
 - Set a root directory name for all FileTables in the database
 - Can be different than the database name
 - Surfaces as a folder in the Windows file share for the server instance
 - Enable non-transactional FILESTREAM access for the database
 - Traditional T-SQL and streaming API access is still transactional
 - Can be enabled for readonly or full non-transactional access

2012+

Creating a FileTable-Enabled Database

```
CREATE DATABASE PhotoLibrary
ON PRIMARY
    (NAME = PhotoLibrary_data,
     FILENAME = 'C:\DB\PhotoLibrary_data.mdf'),
FILEGROUP FileStreamGroup1 CONTAINS FILESTREAM
    (NAME = PhotoLibrary_photos,
     FILENAME = 'C:\DB\Photos')
LOG ON
    (NAME = PhotoLibrary_log,
     FILENAME = 'C:\DB\PhotoLibrary_log.ldf')
WITH FILESTREAM
    (DIRECTORY_NAME='PhotoLibrary',
     NON_TRANSACTED_ACCESS=FULL)
```



2012+

FileTable-Enabling an Existing Database

```
ALTER DATABASE PhotoLibrary
SET FILESTREAM
    (DIRECTORY_NAME='PhotoLibrary',
     NON_TRANSACTED_ACCESS=FULL)
```



2012+

Creating a FileTable

- FileTable has a fixed schema
 - You don't (can't) supply a column list
 - You can specify a collation for the name column
 - But it must be a case-insensitive collation
- Exposes a subfolder beneath the database folder in the emulated file system
 - Root directory for the FileTable
 - Named after the table (can be different)
 - Rows in the FileTable surface as files and folders beneath this subfolder



2012+

Creating a FileTable

```
CREATE TABLE FinDoc AS FileTable  
FILESTREAM_ON MyDB_Docs  
WITH (FILETABLE_DIRECTORY = 'Financial Documents')
```



2012+

FileTable Functions

- **FileTableRootPath**
 - Returns the path for a specified FileTable
- **GetFileNamespacePath**
 - Returns the path for a specific file_stream instance (folder or file)
- **GetPathLocator**
 - Returns the hierarchyid for a specific path



2012+

FileTable Namespace

- Refers to a FileTable's hierarchical structure
 - Machine name, instance share name, database name, FileTable name
 - Root path is `\servername\instance\database\filetable`
- FileTable semantics
 - Uses many constraints, computed columns, and defaults
 - Ensures consistency, which adds overhead
- Improve performance for bulk operations
 - Disabling the namespace disables constraints
 - Temporarily disable the namespace to improve performance for bulk operations



2012+

FileTable Catalog Management Views

- Changed catalog management view
 - **sys.tables**
 - New is_filetable column
- New catalog management views
 - **sys.database_filestream_options**
 - Returns FileTable settings for a database
 - **sys.filetables**
 - Returns information about the FileTables in the database
 - **sys.filetable_system_defined_objects**
 - Returns all the system-generated FileTable defaults and constraints



2012+

Non-Transacted Access DMV/Stored Proc

- New dynamic management view
 - **sys.dm_filestream_non_transacted_handles**
 - Returns currently open non-transactional file and folder handles
- New stored procedure
 - **sp_kill_filestream_non_transacted_handles**
 - Terminates handles to files open for non-transactional access



2012+

FileTable demo

Visual Studio  EXPERT SOLUTIONS FOR .NET DEVELOPERS

2012+

FileTable Limitations

- Limited security
 - SQL Server, not the file system, controls security
 - No row-level security = no file permissions
 - Prevent create and delete
 - Deny DELETE and/or INSERT permission on the FileTable
 - Cannot prevent overwriting
- Fixed schema
 - Extend with custom metadata using 1:1-related table
- No support for memory-mapped files

Visual Studio  EXPERT SOLUTIONS FOR .NET DEVELOPERS

Thank You!

- Contact me
 - lenni.lobel@sleektech.com
- Visit my blog
 - lennilobel.wordpress.com
- Follow me on Twitter
 - [@lennilobel](https://twitter.com/lennilobel)
- Thanks for coming! ☺

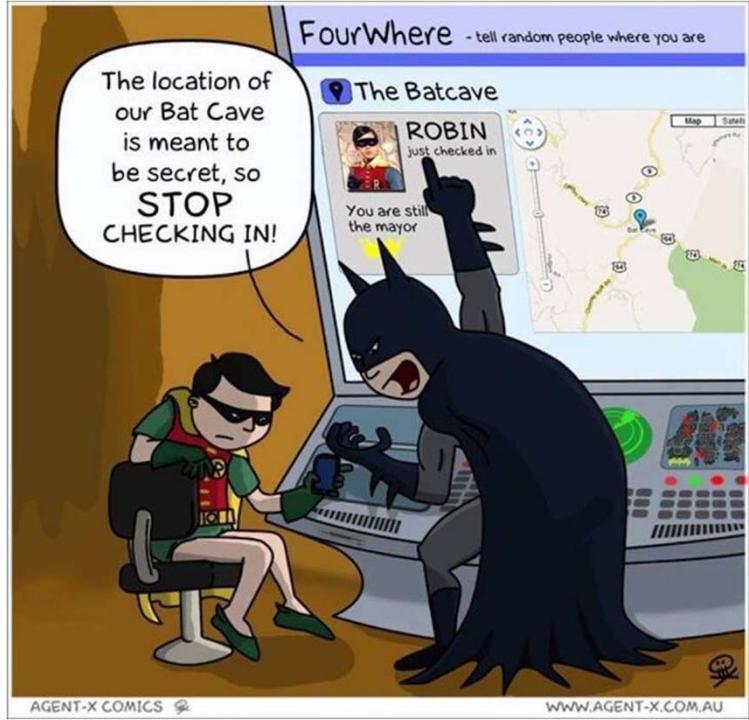


***“Programmers – we do precision
guesswork based on unreliable
data provided by those of
questionable knowledge”***

Did you know?



Hey Robin...

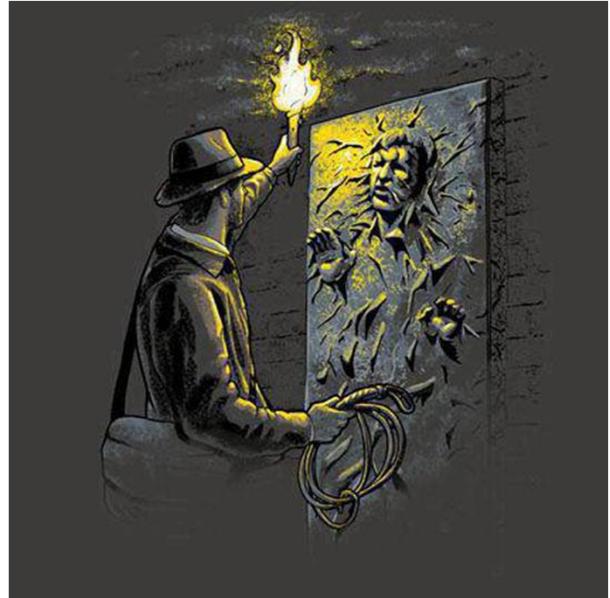


***"The problem with internet quotes
is that you can't always depend on
their accuracy"***

– Abraham Lincoln

Did you know?

Finding yourself...



Visual Studio **LIVE!**
EXPERT SOLUTIONS FOR .NET DEVELOPERS

“Always code as if the guy who ends up maintaining your code will be a violent psychopath who knows where you live.”

– Martin Golding

Did you know?

Visual Studio **LIVE!**
EXPERT SOLUTIONS FOR .NET DEVELOPERS

**Does this
describe you
too?**



"According to your LinkedIn profile you're a focused, disciplined achiever. According to your Facebook photos you love Jack Daniels and are pretty comfortable with your body."

***"Programmers – we fix problems
that you don't know you have, in a
way that you don't understand"***

Did you know?