Angela Henry

DATA TYPES DO MATTER

ANGELA HENRY

Angela is a DBA/BI Developer, living in High Point, NC and loves what she does. She's worked with all versions of SQL Server & worn all the hats that come with dealing with SQL Server throughout the years: developer, administrator, data architect and BI developer.

MSCE: Business Intelligence

Microsoft Data Platform MVP

She is also the local group leader for the PASS Triad SQL User Group in Greensboro, NC.

Tweets at @SQLSwimmer
Blogs at SQLSwimmer.wordpress.com

In her spare time you can probably find her in or at the pool, she's an avid US Masters & USA Swimming Swimmer, Coach and Instructor.

MUSIC CITY TECH

Thanks to our sponsors!

MUSIC CITY CODE
MUSIC CITY AGILE
MUSIC CITY DATA

Platinum

Microsoft

Silver

MINDSEYE

Leading EDJE

JET BRAINS

INGRAM CONTENT GROUP

cardinal
PEOPLE.IDEAS.SOLUTIONS.

LIFEWAY | IT

NASHVILLE SOFTWARE SCHOOL

Friday attendee party brought to you by VACO Free Yourself.

OVERVIEW

Data Quality

Storage

Performance

SQL SERVER DATA TYPES

user-defined data types
sql_variant
xml
datetimeoffset
datetime2
datetime
smalldatetime
date
time
float
real
decimal
money
smallmoney
bigint
int
smallint
tinyint
bit
ntext
text
image
timestamp
uniqueidentifier
nvarchar (including nvarchar(max))
nchar
varchar (including varchar(max))
char
varbinary (including varbinary(max))
binary

DATA QUALITY

Character Data Types - Everyone’s favorite

- For Temporal Data (Dates/Times)
- For Numbers

Demo

DATA QUALITY

Comparison Operators

Sorting

Garbage In = Garbage Out

STORAGE

They’re just numbers, right?!

It’s just string data, who cares?

JUST NUMBERS

Data type	Range	Storage
bigint	-2^63 to 2^63-1 -9,223,372,036,854,775,808 to 9,223,372,036,854,775,807 (6 commas!!)	8 Bytes
int	-2^31 to 2^31-1 (-2,147,483,648) to (2,147,483,647)	4 Bytes
smallint	-2^15 to 2^15-1 (-32,768) to (32,767)	2 Bytes
tinyint	0 to 255	1 Byte

Demo

NOT JUST NUMBERS

Data type	Range	Storage	Impact
bigint	-2^63 to 2^63-1 -9,223,372,036,854,775,808 to 9,223,372,036,854,775,807 (6 commas!!)	8 Bytes	200% > int 400% > smallint 800% > tinyint
int	-2^31 to 2^31-1 (-2,147,483,648) to (2,147,483,647)	4 Bytes	200% > smallint 400% > tinyint
smallint	-2^15 to 2^15-1 (-32,768) to (32,767)	2 Bytes	200% > tinyint
tinyint	0 to 255	1 Byte	

JUST STRING DATA

Data Type	Range	Storage
Char	Fixed Length 1 to 8000	Length of data
Varchar	Variable Length 1 to 8000 or MAX → 2^31 - 1 Bytes (2GB)	Length of data + 2 Bytes
nChar	Fixed Length 1 to 4000	2 * Length of data
nVarchar	Variable Length 1 to 4000 or MAX → 2^31 – 1 Bytes (2GB)	2 * Length of data + 2 Bytes

Demo

NOT JUST STRING DATA

Data Type	Range	Storage	Impact
Char	Fixed Length 1 to 8000	Length of data	200% > varchar with same lengths
Varchar	Variable Length 1 to 8000 or MAX --> 2^31 - 1 Bytes (2GB)	Length of data + 2 Bytes	
nChar	Fixed Length 1 to 4000	2 * Length of data	200% > nvarchar with same lengths
nVarchar	Variable Length 1 to 4000 Or MAX → 2^31 – Bytes (2GB)	2 * Length of data + 2 Bytes	

AND THE WINNER IS.. ~ 10MM ROWS

Data Types	Data Only	Data With Index
TinyInt → BigInt	70MB	140MB
Varchar(255) → Char(255)	2.5GB	5GB

PERFORMANCE

Execution Plans

I/O

EXECUTION PLANS

SARGability - 'SARG' stands for **S**earch **ARG**ument, and it *means* that the predicate can be executed using an index seek

Predicate is <expression> <operator> <expression>

- DateAdd(day, 1, myDateCol) = @myDate is NOT SARGable
- myDateCol = @myDateVariable IS SARGable when they are the same data type

CONVERSION PRECEDENCE

1 user-defined data types (highest)
2 sql_variant
3 xml
4 datetimeoffset
5 datetime2
6 datetime
7 smalldatetime
8 date
9 time
10 float
11 real
12 decimal
13 money
14 smallmoney
15 bigint
16 int
17 smallint
18 tinyint
19 bit
20 ntext
21 text
22 image
23 timestamp
24 uniqueidentifier
25 nvarchar (including nvarchar(max))
26 nchar
27 varchar (including varchar(max))
28 char
29 varbinary (including varbinary(max))
30 binary (lowest)

Demo - Execution Plans

FAT TABLES

```
CREATE TABLE Demo.SuppliersOriginal(
        SupplierID int NOT NULL,
        SupplierName nvarchar(100) NOT NULL,
        SupplierCategoryID int NOT NULL,
        PrimaryContactPersonID int NOT NULL,
        AlternateContactPersonID int NOT NULL,
        DeliveryMethodID int NULL,
        DeliveryCityID int NOT NULL,
        PostalCityID int NOT NULL,
        SupplierReference nvarchar(20) NULL,
        BankAccountName nvarchar(50) NULL,
        BankAccountBranch nvarchar(50) NULL,
        BankAccountCode nvarchar(20) NULL,
        BankAccountNumber nvarchar(20) NULL,
        BankInternationalCode nvarchar(20) NULL,
        PaymentDays int NOT NULL,
        InternalComments nvarchar(max) NULL,
        PhoneNumber nvarchar(20) NOT NULL,
        FaxNumber nvarchar(20) NOT NULL,
        WebsiteURL nvarchar(256) NOT NULL,
        DeliveryAddressLine1 nvarchar(60) NOT NULL,
        DeliveryAddressLine2 nvarchar(60) NULL,
        DeliveryPostalCode nvarchar(10) NOT NULL,
        PostalAddressLine1 nvarchar(60) NOT NULL,
        PostalAddressLine2 nvarchar(60) NULL,
        PostalPostalCode nvarchar(10) NOT NULL,
        LastEditedBy int NOT NULL)
```

```
CREATE TABLE Demo.SuppliersFat(
        SupplierID int NOT NULL,
        SupplierName char(255) NOT NULL,
        SupplierCategoryID int NOT NULL,
        PrimaryContactPersonID int NOT NULL,
        AlternateContactPersonID int NOT NULL,
        DeliveryMethodID int NULL,
        DeliveryCityID int NOT NULL,
        PostalCityID int NOT NULL,
        SupplierReference char(255) NULL,
        BankAccountName char(255) NULL,
        BankAccountBranch char(255) NULL,
        BankAccountCode char(255) NULL,
        BankAccountNumber char(255) NULL,
        BankInternationalCode char(255) NULL,
        PaymentDays int NOT NULL,
        InternalComments char(2000) NULL,
        PhoneNumber char(25) NOT NULL,
        FaxNumber char(255) NOT NULL,
        WebsiteURL char(256) NOT NULL,
        DeliveryAddressLine1 char(255) NOT NULL,
        DeliveryAddressLine2 char(255) NULL,
        DeliveryPostalCode char(255) NOT NULL,
        PostalAddressLine1 char(255) NOT NULL,
        PostalAddressLine2 char(255) NULL,
        PostalPostalCode char(255) NOT NULL,
        LastEditedBy int NOT NULL)
```

Demo – Performance IO

IMPACT ON PERFORMANCE

Implicit Data Type Conversions

Table Scans/Index Scans

More Pages Mean More IO