

# Predstavenie podpory Geo-Spatial dát na SQL Serveri 2008

*Igor Stanek, Chastia s.r.o., Chief Technology Officer*



Microsoft MVP

# Agenda

- *Priestorové dáta*
- *Aplikácie pracujúce s priestorovými dátami*
- *Podpora v SQL Serveri 2008*
- *Typy súradníc*
- *Svetový geodetický systém*
- *Nové dátové typy*
- *Reprezentácia dát*
- *Indexovanie*

# Priestorové dáta

- *Vektorové*
  - *Body*
  - *Čiary*
  - *Polygóny*
  
- *Rastrové*
  - *Satelitné snímky*
  - *Letecké fotografie*

# Mapové aplikácie

- *Mnoho aplikácií využíva mapovanie*
- *Mapy môžu byť hlavným výstupom pre tieto aplikácie*
  
- *Príklady:*
  - *Obchodné vizualizácie (Virtual Earth)*
  - *Katastrálne aplikácie*
  - *GISy*
  - *Facility Management*
  - *a mnohé ďalšie...*

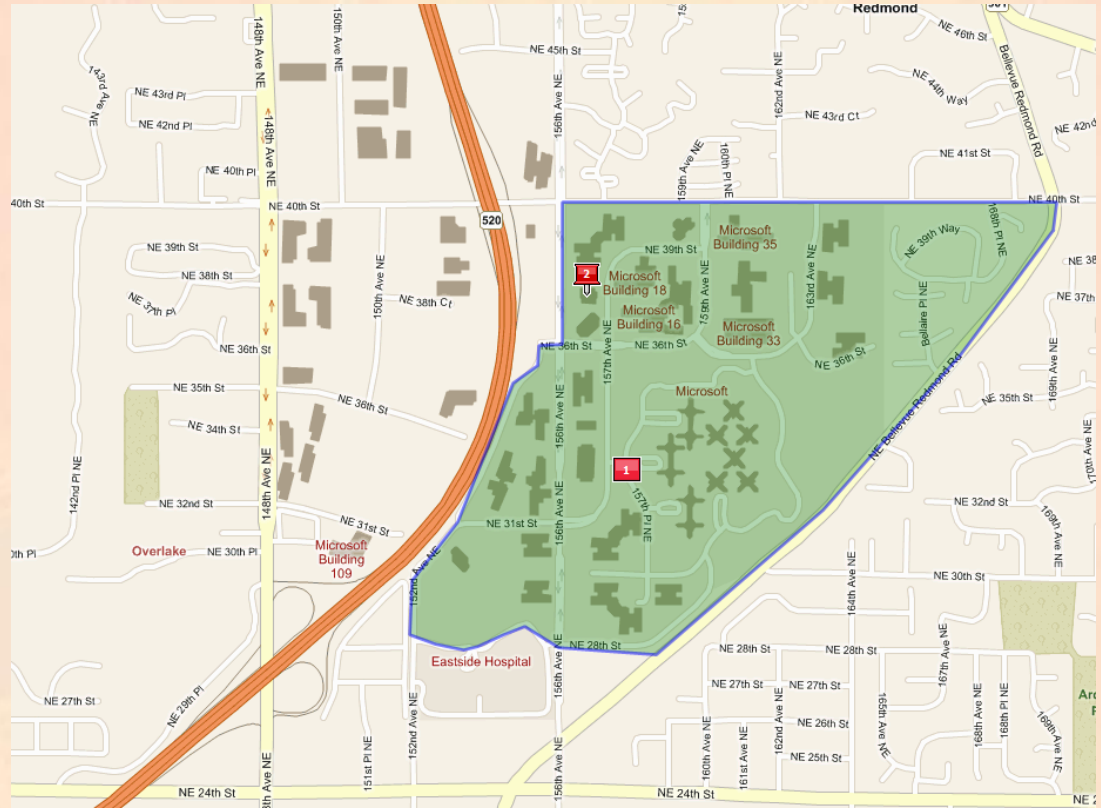
# Aplikácie bez máp

- *Niektoré aplikácie používajú spatial dáta bez použitia máp*
- *Príklady:*
  - *Vyhľadávanie spojov MHD*
  - *...*

# SQL Server 2008

- *Nová podpora pre priestorové dáta*
- *Podpora vektorových dát*
- *Podpora geo-priestorových dát*
  - *Priestorové dáta s umiestnením na povrchu Zeme*
- *2 druhy súradníc*
  - *Planárne*
  - *Geodetické*
- *2D*

# Ukázkový dotaz



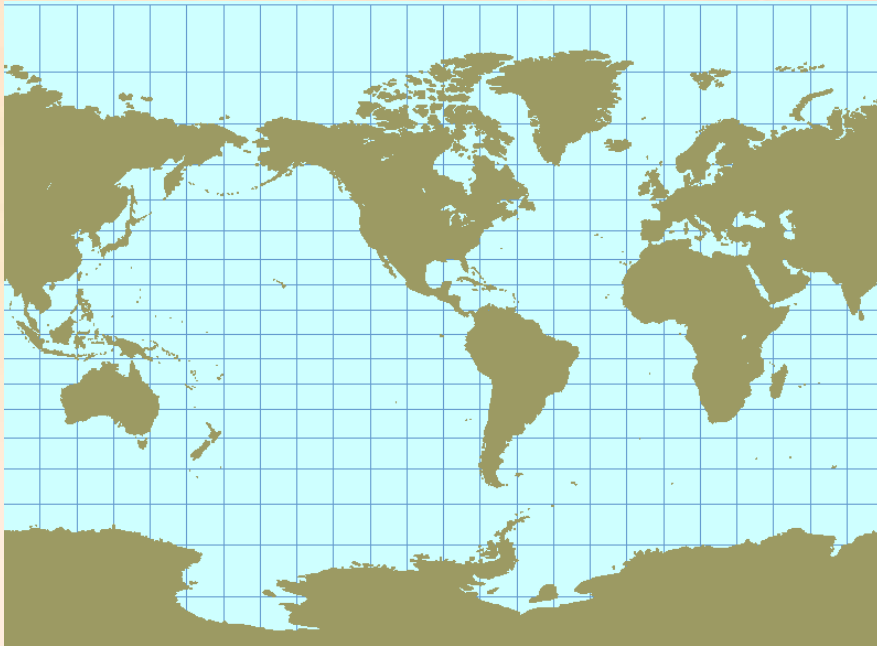
```
SELECT *  
FROM roads  
WHERE roads.geom.Intersects(@ms)=1
```

# Zložitejšie výpočty...



# Planárne vs. Geodetické súradnice

*Model plochej Zeme*



*Model zaoblenej Zeme*



# Kedy použiť jednotlivé typy

## *Planárne – plochá predstava Zeme*

- *Lokálne grafické systémy a interiéry budov*
- *Čiary, body a oblasti pomocou súradníc X, Y*
- *Jednoduchšie výpočty*

## *Geodetické – zaoblená Zem*

- *Geodetický model – Glóbus*
- *Čiary, body a oblasti pomocou zemepisnej dĺžky a šírky*
- *Umiestnenie reštaurácií, kín, divadiel...*
- *Zložitejšia aritmetika*

# Geografické dátové typy

- *SQLCLR od verzie SQL Server 2005*
- *Tvorba kódu a dátových typov pomocou CLR*
  - *Pomocou ľubovoľného .NET jazyka*
- *Spatial typy sú tiež implementované v CLR*
- *SQL User Defined Types*
- *Assembly*
  - *Microsoft.SqlServer.Types*

# Geografické dátové typy

*Geometry*

*Geography*

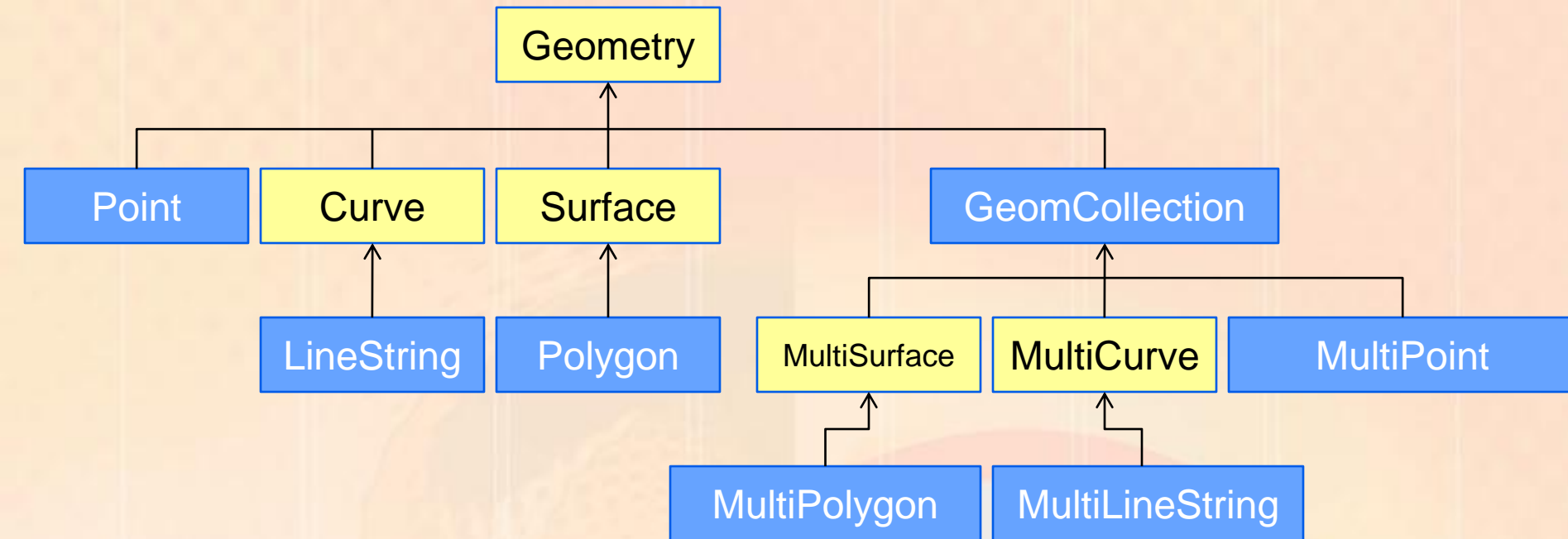
- *Možná práca pomocou ADO.NET*
- *Jednoduché prepojenie na mapové systémy*
  - *Virtual Earth, Google Maps*

# Svetový geodetický systém WGS 84

- *Určovanie polohy pomocou GPS*
  - *Najčastejšie systém World Geodetic System - WGS*
  - *Navrhnutý v roku 1984*
- **WGS 84**
  - *Spatial Reference System (SRID = 4326)*
  - *Definícia súradnicového systému*
  - *Pravotočivá kartézska sústava súradnic so stredom v ťažisku Zeme*
    - *Kladná os X smeruje k priesečníku nultého poludníku a rovníku*
    - *Kladná os Z k severnému pólu*
    - *Kladná os Y je na X a Z kolmá v smere doľava (90° východnej dĺžky a 0° šírky)*

# Geometry a Geography

- *Abstraktné dátové typy*



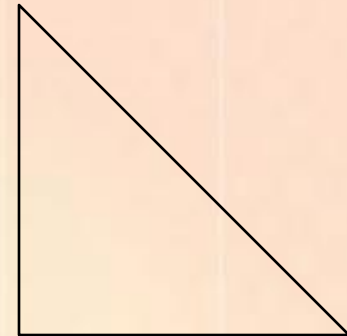
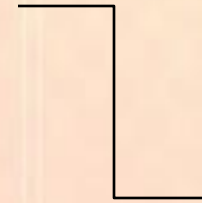
- *V jednom stĺpci môžeme mať uložené rôznorodé geo-priestorové informácie*

# Dátové typy - Point

- *Presná pozícia*
- *Súradnice: X, Y*
- *Nepovinné: Z, M*
  - *Z – výška*
  - *M – mierka*
  - *Je možné definovať, ale nepoužívajú sa pri výpočtoch*

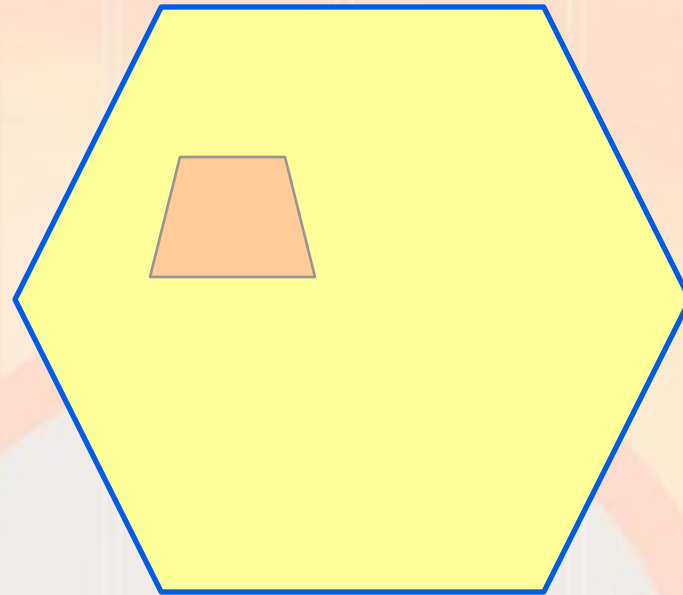
# Dátové typy – LineString

- *Line*
  - *úsečka medzi 2 bodmi*
- *LineString*
  - *úsečky medzi sekvenciou bodov*
- *Simple LineString*
  - *ak sa úsečky nepretínajú*
- *Ring LineString*
  - *ak je počiatočný a koncový bod rovnaký*
- *Vlastnosti*
  - *Dĺžka*



# Dátové typy – Polygon

- *Uzavretý 2D útvar*
  - *Definovaný pomocou Ring LineString*
- *Môže obsahovať diery*
  
- *Vlastnosti*
  - *Dĺžka*
  - *Plocha*



# Dátové typy – Collection

- *Kolekcie typov*
  - *Point*
  - *LineString*
  - *Polygon*

# Reprezentácia dát

- Podľa konzorcia OGC
- Podpora 3 formátov
  - Well-Known Text (WKT)
  - Well-Known Binary (WKB)
  - Geography Markup Language (GML)
- Najčitateľnejší zápis pre človeka - WKT

```
POINT(10 10)
POINT(10 10 10 1)           -- súradnice X Y Z M
LINESTRING(0 0, 10 10)
POLYGON(0 0, 0 10, 10 10, 10 0, 0 0)
```

# Práca s geospatial typmi

- *Práca s Geometry a Geography je úplne rovnaká*
- *Vytvorenie inštancie*

```
DECLARE @g Geometry

SET @g = Geometry::STGeomFromText('LINESTRING(0 0, 10 10, 21 2)', 0)

SET @g = Geometry::Parse('LINESTRING(0 0, 10 10, 21 2)')

-- tu sa implicitne zavolá Parse()
SET @g = 'LINESTRING(0 0, 10 10, 21 2)'
```

# Možnosti vytvorenia inštancií

- *Použitím Well-Known Text (WKT)*
  - *STGeomFromText, Parse, STPointFromText, STMPPointFromText, STLineFromText, STMLineFromText, STPolyFromText, STMPolyFromText, STGeomCollFromText*
- *Použitím Well-Known Binary (WKB)*
  - *STGeomFromWKB, STPointFromWKB, STMPPointFromWKB, STLineFromWKB, STMLineFromWKB, STPolyFromWKB, STMPolyFromWKB, STGeomCollFromWKB*
- *Použitím GML*
  - *GeomFromGml*

*Práce s instanciami*

# **ŽIVÁ UKÁŽKA 1**

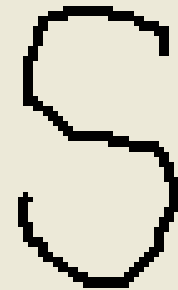
*Práce s datovým typem GEOMETRY v C# aplikacii*

# **ŽIVÁ UKÁŽKA 2**

# Transformácie

- *Existuje množstvo metód na transformáciu*
  - *Geometry -> Geometry*
  - *Geography -> Geography*

```
DECLARE @g GEOMETRY
SET @g = 'LINESTRING ( 69 26, 69 23, 69 21, 67 20, 65 20, 63
18, 58 17, 52 17, 51 17, 49 17, 45 18, 44 20, 44 21, 42 26, 42
29, 42 32, 42 35, 43 35, 47 38, 50 41, 55 42, 58 42, 65 44, 66
44, 67 44, 68 45, 69 47, 70 48, 70 50, 71 51, 70 56, 68 60, 68
63, 66 65, 65 66, 63 68, 60 71, 59 71, 57 71, 55 71, 51 69, 45
65, 44 63, 42 62, 41 59, 41 57, 41 56, 41 54, 42 53 )'
```



# Príklady transformácií I.

```
SELECT @g.STEnvelope().ToString()
```

```
Result: POLYGON ((41 17, 71 17, 71 71, 41 71, 41 17))
```

```
SELECT @g.STConvexHull().ToString()
```

```
Result: POLYGON ((71 51, 70 56, 68 63, 66 65, 65 66,  
63 68, 60 71, 59 71, 57 71, 55 71, 51 69, 45 65, 42 62,  
41 59, 41 57, 41 56, 41 54, 42 26, 44 20, 45 18, 49 17,  
51 17, 52 17, 58 17, 63 18, 67 20, 69 21, 71 51))
```

```
SELECT @g.STBuffer(5).ToString()
```

```
Result: POLYGON ((49 12, 51 12, 52 12, 58 12,  
58.246719360351562 12.00609016418457,  
58.492688179016113 12.024333953857422,  
58.737458229064941 12.054683685302734,  
58.98058032989502 12.097097396850586,  
(... snipped for clarity ...)  
48.693824768066406 12.009382247924805, 49 12))
```



# Príklady transformácií II.

```
SELECT @g.STIntersection(@h).ToString();
```

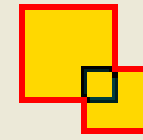
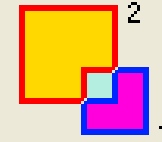
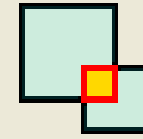
```
Result: POLYGON ((30 30, 40 30, 40 40, 30 40, 30 30))
```

```
SELECT @g.STSymDifference(@h).ToString();
```

```
Result: MULTIPOLYGON (((40 30, 50 30, 50 50, 30 50, 30 40, 40 40, 40 30)), ((10 10, 40 10, 40 30, 30 30, 30 40, 10 40, 10 10)))
```

```
SELECT @g.STUnion(@h).ToString();
```

```
Results: POLYGON ((10 10, 40 10, 40 30, 50 30, 50 50, 30 50, 30 40, 10 40, 10 10))
```



# Inštančné metódy - Geography

- *STArea*
- *STAsBinary*
- *STAsText*
- *STBoundary*
- *STBuffer*
- *STCentroid*
- *STContains*
- *STConvexHull*
- *STCrosses*
- *STDifference*
- *STDimension*
- *STDisjoint*
- *STDistance*
- *STEndpoint*
- *STEnvelope*
- *STEquals*
- *STExteriorRing*
- *STGeometryN*
- *STGeometryType*
- *STInteriorRingN*
- *STIntersection*
- *STIntersects*
- *STIsClosed*
- *STIsEmpty*
- *STIsRing*
- *STIsSimple*
- *STIsValid*
- *STLength*
- *STNumGeometries*
- *STNumInteriorRing*
- *STNumPoints*
- *STOverlaps*
- *STPointN*
- *STPointOnSurface*
- *STRelate*
- *STSrid*
- *STStartPoint*
- *STSymDifference*
- *STTouches*
- *STUnion*
- *STWithin*
- *STX*
- *STY*
- *AsGml*
- *AsTextZM*
- *BufferWithTolerance*
- *InstanceOf*
- *IsNull*
- *M*
- *MakeValid*
- *Reduce*
- *ToString*
- *Z*
- *Lat*
- *Long*
- *NumWings*
- *RingN*

- *STArea*
- *STAsBinary*
- *STAsText*
- *STBoundary*
- *STBuffer*
- *STCentroid*
- *STContains*
- *STConvexHull*
- *STCrosses*
- *STDifference*
- *STDimension*
- *STDisjoint*
- *STDistance*
- *STEndpoint*
- *STEnvelope*
- *STEquals*
- *STExteriorRing*
- *STGeometryN*
- *STGeometryType*
- *STInteriorRingN*
- *STIntersection*
- *STIntersects*
- *STIsClosed*
- *STIsEmpty*
- *STIsRing*
- *STIsSimple*
- *STIsValid*
- *STLength*
- *STNumGeometries*
- *STNumInteriorRing*
- *STNumPoints*
- *STOverlaps*
- *STPointN*
- *STPointOnSurface*
- *STRelate*
- *STSrid*
- *STStartPoint*
- *STSymDifference*
- *STTouches*
- *STUnion*
- *STWithin*
- *STX*
- *STY*
- *AsGml*
- *AsTextZM*
- *BufferWithTolerance*
- *InstanceOf*
- *IsNull*
- *M*
- *MakeValid*
- *Reduce*
- *ToString*
- *Z*
- *Lat*
- *Long*
- *NumWings*
- *RingN*

*Práce s datovým typem GEOGRAPHY v ASP.NET aplikácii s vizualizáciou pomocou Virtual Earth*

# **ŽIVÁ UKÁŽKA 3**

# Indexovanie

- *Nový typ indexu*
  - *SPATIAL INDEX*
- *Urýchlenie práce s dátami typu Geometry alebo Geography*

```
CREATE SPATIAL INDEX sixd  
ON T(geography)
```

*Hotová aplikácia postavená nad SQL Server 2008 Spatial Data*

*[www.ontheroad.to](http://www.ontheroad.to); [pavlis.ontheroad.to](http://pavlis.ontheroad.to)*

# **ŽIVÁ UKÁŽKA 4**

# Odkazy

- 8 dielny seriál
  - <http://jasonfollas.com/blog/archive/2008/03/14/sql-server-2008-spatial-data-part-1.aspx>
- Geospatial Data Generator
  - <http://mikeo.co.uk/demo/sqlspatial/default.aspx>
- <http://www.microsoft.com/sqlserver/2008/en/us/spatial-data.aspx>
- MSDN Library
  - [http://msdn.microsoft.com/en-us/library/bb933790\(SQL.100\).aspx](http://msdn.microsoft.com/en-us/library/bb933790(SQL.100).aspx)



*Aký systém súradníc sa používa najčastejšie v GPS?*

**RÝCHLOSTNÁ SÚŤAŽ 😊**

SQL Server DevCon  
Praha

2008

# Predstavenie podpory Geo-Spatial dát na SQL Serveri 2008

*Igor Stanek, Chastia s.r.o., Chief Technology Officer*

