Our Sponsors

Microsoft

PASS

idera

MELISSA DATA

CUBIDO

AN ACP GROUP COMPANY

ApexSQL
Scaling Analysis Services in the Cloud

by Gerhard Brückl

gerhard@gbrueckl.at
blog.gbrueckl.at
About me

Gerhard Brückl

Working with Microsoft BI since 2006
Windows Azure / Cloud since 2013

focused on Analytics and Reporting

Blog: blog.gbrueckl.at
eMail: gerhard@gbrueckl.at
Agenda

- Why do we need to scale?
- How can we scale
  - Scale Up
  - Scale Out
- Windows Azure
Why do we need to scale?

- Growing amount of data
- Growing amount of users
- Actuality of data
- Complexity of data loads
- Latency (remote locations)
Things we want to achieve

- Better Query Performance
  - Single User
  - Concurrent User
- Faster availability of data
- High Availability
- Easy maintenance
- Flexible resource usage / peak times
- Better Processing Performance
How can we scale?

Scale Up
- Increase resources of current machine

Scale Out
- Add further machines
- Create “Farm”
Scale Up

Increase

- CPU (clock rate vs. #cores)
- Memory
- IO
- (Network)
Scale Up

Pros

- Solves most issues
- Straight forward / simple
- No change in architecture

Cons

- Can only Scale Up to a certain point
- Expensive in terms of high-end hardware
Scale Up

CPU
- Parallel Processing / Querying
- Concurrent users
- (faster calculations)

Memory
- Caching – relieve IO / CPU

I/O
- Faster initial Querying / Processing
## Scale Up

<table>
<thead>
<tr>
<th></th>
<th>Increase CPU</th>
<th>Increase Memory</th>
<th>Increase IO</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cube size (in general)</td>
<td>+</td>
<td>+</td>
<td>+</td>
</tr>
<tr>
<td>Query Performance (single user)</td>
<td>+</td>
<td>+</td>
<td>+</td>
</tr>
<tr>
<td>Query Performance (multiple users)</td>
<td>+</td>
<td>+</td>
<td>~</td>
</tr>
<tr>
<td>Processing Performance</td>
<td>+</td>
<td>~</td>
<td>+</td>
</tr>
<tr>
<td>Actuality of data</td>
<td>+</td>
<td>~</td>
<td>+</td>
</tr>
<tr>
<td>High Availability</td>
<td>~</td>
<td>~</td>
<td>~</td>
</tr>
</tbody>
</table>
Scale Out

- Increase number of machines
- Distribute work load
- Handle peaks
Scale Out

Pros

- “unlimited” scalability
- Flexibility
  - easy to extend / shrink farm
  - Can be leveraged during data loads

Cons

- Change in Architecture
- Harder to maintain
- “Only” solves concurrency issues
- No improvements for single queries
## Scale Out

<table>
<thead>
<tr>
<th></th>
<th>Increase Number of Machines</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cube size (in general)</td>
<td>~</td>
</tr>
<tr>
<td>Query Performance (single user)</td>
<td>~</td>
</tr>
<tr>
<td>Query Performance (multiple users)</td>
<td>+</td>
</tr>
<tr>
<td>Processing Performance</td>
<td>+</td>
</tr>
<tr>
<td>Actuality of data</td>
<td>+</td>
</tr>
<tr>
<td>High Availability</td>
<td>+</td>
</tr>
</tbody>
</table>
Windows Azure and Analysis Services

- **Software as a Service (SaaS)**
  - e.g. Office 365, SharePoint Online, ...

- **Platform as a Service (PaaS)**
  - e.g. SQL Azure Databases, ...

- **Infrastructure as a Service (IaaS)**
  - e.g. hosted virtual machine
Windows Azure IaaS

- Hosted Infrastructure
- Virtual Machines
- Virtual Storage
- Virtual Network
Choose a Region

Create an Affinity Group
- Per Region
- Per Subscription

Create Network
- Per Affinity Group
Windows Azure Virtual Network

Per Region / Affinity Group

Place VMs in same network

Integrate on-site services
  - Point-to-Site Connectivity
  - Site-to-Site Connectivity
Windows Azure Storage Account

Per Region / Affinity Group

Blobs / Tables / Queues
  ▪ VHDs

Locally Redundant
Geo Redundant
Geo Redundant (Read-Only)
Windows Azure Virtual Machine

Per Region / Affinity Group / Virtual Network

Created from Gallery
- Public Templates
- Private Templates

Different Sizes \(\rightarrow\) Scale Up

Availability Sets \(\rightarrow\) Scale Out
## Windows Azure Scale Up

<table>
<thead>
<tr>
<th>Size</th>
<th># of Cores</th>
<th>Memory</th>
<th># of Disks</th>
<th>Price / Month *</th>
</tr>
</thead>
<tbody>
<tr>
<td>Extra small</td>
<td>1 (shared)</td>
<td>768</td>
<td>1</td>
<td>315 €</td>
</tr>
<tr>
<td>Small</td>
<td>1</td>
<td>1,750</td>
<td>2</td>
<td>354 €</td>
</tr>
<tr>
<td>Medium</td>
<td>2</td>
<td>3,500</td>
<td>4</td>
<td>404 €</td>
</tr>
<tr>
<td>Large</td>
<td>4</td>
<td>7,000</td>
<td>8</td>
<td>504 €</td>
</tr>
<tr>
<td>Extra Large</td>
<td>8</td>
<td>14,000</td>
<td>16</td>
<td>1,008 €</td>
</tr>
<tr>
<td>A5</td>
<td>2</td>
<td>14,000</td>
<td>4</td>
<td>526 €</td>
</tr>
<tr>
<td>A6</td>
<td>4</td>
<td>28,000</td>
<td>8</td>
<td>747 €</td>
</tr>
<tr>
<td>A7</td>
<td>8</td>
<td>56,000</td>
<td>16</td>
<td>1,495 €</td>
</tr>
<tr>
<td>(A8)</td>
<td>8</td>
<td>56,000</td>
<td>? / SSD</td>
<td>?</td>
</tr>
<tr>
<td>(A9)</td>
<td>16</td>
<td>112,000</td>
<td>? / SSD</td>
<td>?</td>
</tr>
</tbody>
</table>

*) SQL Server SE + Windows
Windows Azure Scale Up

Simply change size
  - Restart required

Limited Capacities
  - CPU
  - Memory
  - IO

Use dedicated machines!
Windows Azure
Scale Up – Analysis Services

Things to consider:

- Adjust I/O sub-system
  - Add VHDs / rebuild Storage Pool
- Adjust memory limits
  - Absolute settings
- Adjust CPUs
  - Thread settings
  - Group Affinity
Windows Azure
Scale Up – Analysis Services

For frequent Scale Ups / Scale Downs
- Use default settings
- Use only relative settings
  - Memory settings in %
  - Thread settings 0 or <0
# Windows Azure
## Scale Up – Analysis Services

### On-Premise vs. Cloud

<table>
<thead>
<tr>
<th></th>
<th>On-Premise</th>
<th>Cloud “A7”</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>CPU</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>NUMA Nodes</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>Logical Cores</td>
<td>36</td>
<td>8</td>
</tr>
<tr>
<td><strong>Memory</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Physical Memory</td>
<td>512 GB</td>
<td>56 GB</td>
</tr>
<tr>
<td>SSAS / TotalMemoryLimit</td>
<td>100 GB</td>
<td>47 GB (85%)</td>
</tr>
<tr>
<td><strong>IO</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Disks (SSAS only)</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>RAID-Set</td>
<td>RAID-0</td>
<td>RAID-0</td>
</tr>
</tbody>
</table>

**CPU x4**

**Memory x2**
Windows Azure
Scale Up – Analysis Services

Processing Performance:

Test SSAS-DB: 26 GB
Test SQL-DB: 106 GB

<table>
<thead>
<tr>
<th></th>
<th>On-Premise</th>
<th>Cloud “A7”</th>
</tr>
</thead>
<tbody>
<tr>
<td>Parallel Processing</td>
<td>00:13:15</td>
<td>01:25:41</td>
</tr>
<tr>
<td>Serial Processing</td>
<td>01:36:05</td>
<td>05:17:28</td>
</tr>
</tbody>
</table>
Windows Azure
Scale Up – Analysis Services

Query Performance:
7 Queries, +1 User/min, 60 mins

<table>
<thead>
<tr>
<th>Response time in Seconds</th>
<th>On-Premise</th>
<th>Cloud “A7”</th>
</tr>
</thead>
<tbody>
<tr>
<td>Single Query1</td>
<td>3.25</td>
<td>15.30</td>
</tr>
<tr>
<td>Single Query2</td>
<td>51.60</td>
<td>86.30</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Finished Tests</th>
<th>On-Premise</th>
<th>Cloud “A7”</th>
</tr>
</thead>
<tbody>
<tr>
<td>After 20 minutes/users</td>
<td>347</td>
<td>166</td>
</tr>
<tr>
<td>After 40 minutes/users</td>
<td>871</td>
<td>346</td>
</tr>
<tr>
<td>After 60 minutes/users</td>
<td>1,356</td>
<td>480 (*)</td>
</tr>
</tbody>
</table>
Windows Azure Scale Up – Analysis Services

Scales with Number of CPUs

Memory for concurrency

IO for bigger databases
Windows Azure
Scale Out

Easily create/add new VMs
- Images
- Script

Built-In Load Balancing
- Availability Sets
- Traffic Manager

Unused VMs create no costs
Windows Azure Availability Sets

Per Cloud Service
- Several VMs share same Public Port

Used for
- High Availability
  - FaultDomains
  - Update Domains
- Load Balancing
Windows Azure Traffic Manager

Per Subscription

- Several Cloud Services share URL

Used For

- Availability
- Failover
- Performance
Windows Azure Availability Sets and Traffic Manager

<table>
<thead>
<tr>
<th></th>
<th>Traffic Manager</th>
<th>Availability Set</th>
</tr>
</thead>
<tbody>
<tr>
<td>Regions</td>
<td>Any</td>
<td>one</td>
</tr>
<tr>
<td>Round Robin</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Performance</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>Failover</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>Routing-Level</td>
<td>DNS</td>
<td>TCP / UDP</td>
</tr>
</tbody>
</table>
Windows Azure Availability Sets

Auto-Scale Feature

- Scale by Metric
- Scale by Schedule

Only for Availability Sets

Also Third party tools
# Windows Azure Analysis Services Settings

<table>
<thead>
<tr>
<th>Memory</th>
<th>Default Value</th>
<th>Suggested Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>LowMemoryLimit</td>
<td>65</td>
<td>85</td>
</tr>
<tr>
<td>TotalMemoryLimit</td>
<td>80</td>
<td>90</td>
</tr>
<tr>
<td>PreAllocate</td>
<td>0</td>
<td>85</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>OLAP/Process</th>
<th>Default Value</th>
<th>Suggested Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>AggregationMemoryLimitMin</td>
<td>10</td>
<td>5</td>
</tr>
<tr>
<td>AggregationMemoryLimitMax</td>
<td>80</td>
<td>10</td>
</tr>
</tbody>
</table>
Windows Azure
Scaling I/O for Analysis Services

Affinity Groups

Storage Account
- Locally Redundant
- Geo Redundant
- Geo Redundant (Read-Only)
Windows Azure
Scaling I/O for Analysis Services

Limits Storage Account (Locally Redundant)
- 10 Gb/s in / 15 Gb/s out
- 20,000 Transaction/s

Blobs and VHDs
- 500 IO/s
- 60 MB/s
Windows Azure
Scaling I/O for Analysis Services

Windows Server 2012 Storage Pools
- Abstraction Layer
- (Software) RAID

GUI very buggy
→ User PowerShell instead!

RAID / Storage Pools vs. Single Drives
Windows Azure
Scaling I/O for Analysis Services

Theory – 16 Disks:
60 MB/s * 16 \rightarrow 960 MB/s
500 IO/s * 16 \rightarrow 8,000 IO/S

Reality – “A7” with 16 Disks:
\sim 110 MB/s
\sim 6,500 IO/s
Windows Azure
Scaling I/O for Analysis Services

8k Random Reads

16k Random Reads

32k Random Reads

64k Random Reads
Windows Azure
Scaling I/O for Analysis Services

IO scales linear for small blocks
No improvements for bigger blocks
Windows Azure
Scaling I/O for Analysis Services

The special Temporary Storage (D:\)
- superfast but volatile storage

8k Random Reads

120,000 IO/s

64k Random Reads

2,100 MB/s
Windows Azure
Scaling I/O for Analysis Services

**MSDN:** “...performance is not guaranteed to be predictable”

Deleted on resets, reboots, fail overs etc.

Scale-Outs with separate Query and Processing Servers
Windows Azure
Scaling I/O for Analysis Services

Distributing IO – Remote Partitions

Further Investigation needed
Windows Azure Processing Analysis Services

Cloud Service

SSAS1

SSAS2

SSAS3

SSAS4
Conclusion & Findings

Scale Up

Easy to do

Only limited scalability

Watch out for:

- Disk / IO Setup
- absolute SSAS Settings
Conclusion & Findings
Scale Out

Almost everything can be automatized
- Create VMs
- Setup SSAS

Very flexible
- Shutdown unused VMs
- Auto-Scale Service
Conclusion & Findings

Scale Out

Use Availability Sets if possible
- Fault Domains
- Update Domains

Use Traffic Manager for
- Global Scale-Out
- Fail-Over
Conclusion & Findings
Scale Out

“Sticky Session” not supported!
Don’t use “DirectReturn”

Possible issues with Session Objects
Calculated members/sets
- Excel Pivot Tables
- XL Cubed
- ...
Writeback
Conclusion & Findings

IO

Use Storage Pools

Max # of Disks

Upcoming Changes
- A8 / A9 VMs
- Internal Changes
THANK YOU!