

Real World Azure

# Elasticity from on-premise to Cloud (and back)

---

Christian Weyer, thinktecture

[christian.weyer@thinktecture.com](mailto:christian.weyer@thinktecture.com)



Windows Azure by Example

# Elasticity from on-premise to Cloud (and back?)

---

Christian Weyer, thinktecture

[christian.weyer@thinktecture.com](mailto:christian.weyer@thinktecture.com)



# Christian Weyer and thinktecture

- Solution architect and principal consultant at thinktecture
- Focus on
  - distributed applications
  - service orientation, workflows
  - cloud computing
  - interoperability
  - end-to-end solutions
  - Windows Server, WCF, WF, MSMQ, Windows Azure platform
- Microsoft MVP for Windows Azure (Architecture)
- Independent Microsoft Regional Director for Germany
- <http://blogs.thinktecture.com/cweyer>
- [christian.weyer@thinktecture.com](mailto:christian.weyer@thinktecture.com)



# A short trip through the Windows Azure Cloud

---

- Introduction
- Windows Azure Compute – Development & Deployment
- Windows Azure Storage – Runtime Assistant
- SQL Azure – Database as a Service
- Scalability – More and more
- Windows Azure Web-Worker – Cloud Value-add
- „... and back ...“ ? – The Cloud is not everything



# Windows Azure Platform

---

- In many minds “Azure” is mostly glorified hosting
  - but it is Platform-as-a-Service (PaaS), not IaaS, not hosting
- The Windows Azure Platform is actually more than the sum of its pieces
- The Cloud is the perfect (the natural) environment for distributed applications & the idea of service orientation
- COA - Cost-Oriented Architecture
  - the architect needs to have controller’s blood in her veins

# Windows Azure Platform



- Scalable compute and storage
- Automated application/service management
- Familiar tools, technologies, languages



- Relational storage for the cloud
- Consistent development model
- Automated database management



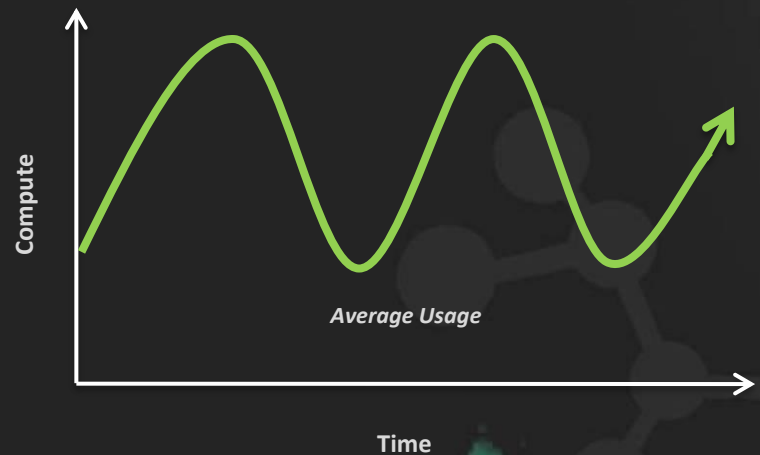
- Connect existing apps to the cloud
- Connect through network boundaries
- Easily control authorization to apps
- Cache data for low latency and costs



## Predictable Bursting

- Services with e.g. seasonal bursts
- Peaks predictable, can be planned
  - planned elasticity
- High IT complexity and low efficiency
- Typical examples
  - online shops, order systems, sports information systems

### *Predictable Bursting*





# A Sample Application

---

- *MvcMusicStore* – simple MVC3 sample application
  - running inside IIS 7.5
  - using SQL Server 2008 R2
  - designed to be a ,stateless‘ application
    - think about album images et. al., though
  - not aware of the Cloud
- **Azure-ize this!**

*[we are not interested in the quality of the sample’s code and architecture here 😊]*



## Windows Azure Compute - Development

---

- Create Windows Azure project in Visual Studio 2010
- Add *MvcMusicStore* project to the solution
  - make sure MVC assemblies are deployable
- Inspect & adjust Service Definition
- Add Windows Azure diagnostics code
- Consider startup tasks to run logic during role/VM initialization
- Local compute emulator is not feature-equivalent

# Windows Azure Compute – Deployment

---

- Create hosted service via portal
  - consider geo location in different regions
  - use Traffic Manager to route user requests
- Establish DNS mapping to Windows Azure hosted service
- Register management certificate
- Configure RDP
- Manual deployment
  - use WebDeploy during development lifecycle
- Develop & test automated deployment (e.g. via command line or PowerShell)

## Windows Azure Storage – Runtime Assistant

---

- Containers & tables for diagnostics & tracing
  - important for runtime monitoring
  - produces costs
- Put commonly used data (like images) into blob storage
  - consider caching data
- Use blob storage as deployment mechanism
  - e.g. together with web-worker



# SQL Azure – Database as a Service

---

- Create database via portal
- Create application user via T-SQL
- Change connection string
- Add connection/error handling to .NET data access code
  - 'code far' can be problematic
- Opt.: establish data sync between on-premise & Cloud
- Implement & test backup
  - via database COPY
  - via PowerShell to Blob Storage
  - via Data Sync

# Scalability – More and more

---

- Consider VM sizes (scale up)
  - more cores, more RAM, more bandwidth – more expensive
- Consider more instances (scale out)
  - based on technical & business PKIs
  - use custom code solution
  - use 3rd party Cloud services
- SQL Azure sharding (e.g. with SQL Azure Federations)
- Consider moving to a NoSql approach
  - may use Windows Azure Table Storage
- Use AppFabric Caching Service to minimize round trips
  - for latency & costs

## Windows Azure Web-Worker – Cloud Value-add

---

- Worker functionality in web role
  - web runs in IIS, worker in separate process
    - both monitored processes
  - usually used for background work
    - e.g. simple warm-up functionality, polling tasks
  - can be used for hosting arbitrary server processes
  - cost-efficient
- *Example:* easy deployment mechanism for web sites in Windows Azure
  - poor man's multi tenancy

„... and back ...“ ? – The Cloud is not everything

---

- Plan for falling back from Cloud to on-premise?
  - for cost reasons
  - for data security/sovereignty reasons
  - proper DNS setup is important
- ***But fact is:*** if you are in the Cloud and have optimized for it – there is no easy way back





# Summary – Azure-ized

---

- Windows Azure enables you to run your applications and services in the Cloud
  - PaaS: no focus on infrastructure
- Elasticity possible from on-premise to Cloud
  - also in the Cloud through role instance management
- Not always as easy as deploying existing app to Cloud
  - design & optimize for the Cloud to get the best out of it
  - ... there may be no easy wack back
- Windows Azure & the whole platform are still young
  - the services will mature, the tools have to mature!

# Resources

---

- [christian.weyer@thinktecture.com](mailto:christian.weyer@thinktecture.com)
- <http://weblogs.thinktecture.com/cweyer>
- Windows Azure blog category
  - <http://weblogs.thinktecture.com/cweyer/azure/>



# Resources

---

- MusicStore
  - <http://mvcmusicstore.codeplex.com/>
- Windows Azure, SQL Azure, Windows Azure AppFabric Portal
  - <http://windows.azure.com>
- Windows Azure Service Management CmdLets
  - <http://archive.msdn.microsoft.com/azurecmdlets>
- Best Practices for Handling Transient Conditions in SQL Azure Client Applications
  - <http://blogs.msdn.com/b/appfabriccat/archive/2010/10/28/best-practices-for-handling-transient-conditions-in-sql-azure-client-applications.aspx>
- SQL Azure Data Sync
  - <http://www.microsoft.com/en-us/sqlazure/datasync.aspx>
- Cerebrata Windows Azure tools
  - <http://www.cerebrata.com>
- AzureWatch
  - <http://www.paraleap.com/azurewatch>

# Christian Weyer and thinktecture

---

- In-depth consulting and training for software architects & developers
- Focus on
  - distributed applications
  - service orientation, workflows
  - cloud computing
  - interoperability
  - end-to-end solutions
  - Windows Server, WCF, WF, MSMQ, Windows Azure platform
- Contact: [christian.weyer@thinktecture.com](mailto:christian.weyer@thinktecture.com)

# Get Started with Windows Azure For Free Today!

---

- **MSDN Subscriber**

- Activate Your Free Included MSDN Benefits via <http://tinyurl.com/activatemsdnazurebenefits>

- **Individual:**

- Get a Free Azure Introductory via <http://tinyurl.com/freeintroazureoffer>
  - Free Computation hours and Storage
- Get 30 Days Free Windows Azure via <http://www.windowsazurepass.com>
  - Select Belgium and enter Promo code: AZP001

- **Partner**

- Get free monthly access to Azure with Partner Cloud Essentials via <http://www.microsoftcloudpartner.com/>

# Start Developing on the Windows Azure Platform



1. Activate your Benefits (see previous slide)
2. Get the Tools via  
**<http://tinyurl.com/toolsforazure>**
3. First learn how to create an application via  
**<http://tinyurl.com/deployazureapplication>**



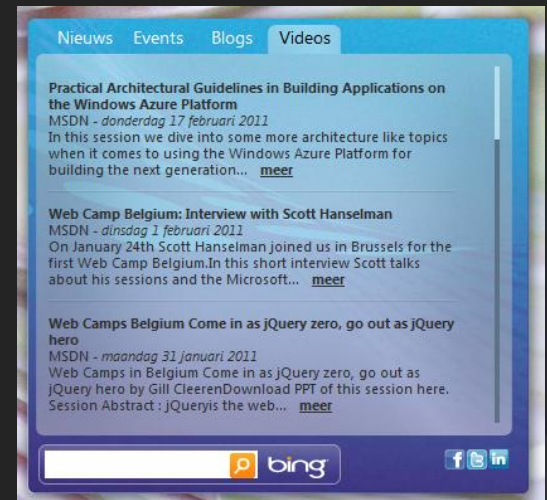
# Stay up to date with MSDN Belux

- Register for our newsletters and stay up to date:  
<http://www.msdn-newsletters.be>
  - Technical updates
  - Event announcements and registration
  - Top downloads
- Follow our blog  
<http://blogs.msdn.com/belux>
- Join us on Facebook  
<http://www.facebook.com/msdnbe>  
<http://www.facebook.com/msdnbelux>
- LinkedIn: <http://linkd.in/msdnbelux/>
- Twitter: [@msdnbelux](https://twitter.com/msdnbelux)

## Download

MSDN/TechNet Desktop Gadget

<http://bit.ly/msdntngadget>





# TechDays 2011 On-Demand

---

- **Watch** this session on-demand via Channel9  
<http://channel9.msdn.com/belux>
- Download to your favorite MP3 or video player
- Get access to slides and recommended resources by the speakers



# THANK YOU

---

