



# Windows Azure Compute Myth Busters

Rainer Stropek  
software architects gmbh  
[rainer@software-architects.at](mailto:rainer@software-architects.at)

# Introduction

- software architects gmbh
- Rainer Stropek

Developer, Speaker, Trainer

MVP for Windows Azure

[rainer@timecockpit.com](mailto:rainer@timecockpit.com)



@rstropek



<http://www.timecockpit.com>  
<http://www.software-architects.com>

IT &  
Dev CONNECTIONS  
powered by Microsoft®

# Vorkenntnisse und Ziele

- Vorkenntnisse:
  - Schon einmal von Azure gehört
- Ziele:
  - Was bedeutet Azure für Entwickler?
  - Was bleibt gleich, was ist neu?
  - Tipps, Tricks und typische Fallen
- Kein Ziel:
  - Marketingsession



Um mit Cloud Computing zu starten,  
muss ich **viel lernen – keine Zeit** dafür!

Just another  
**Windows Server**

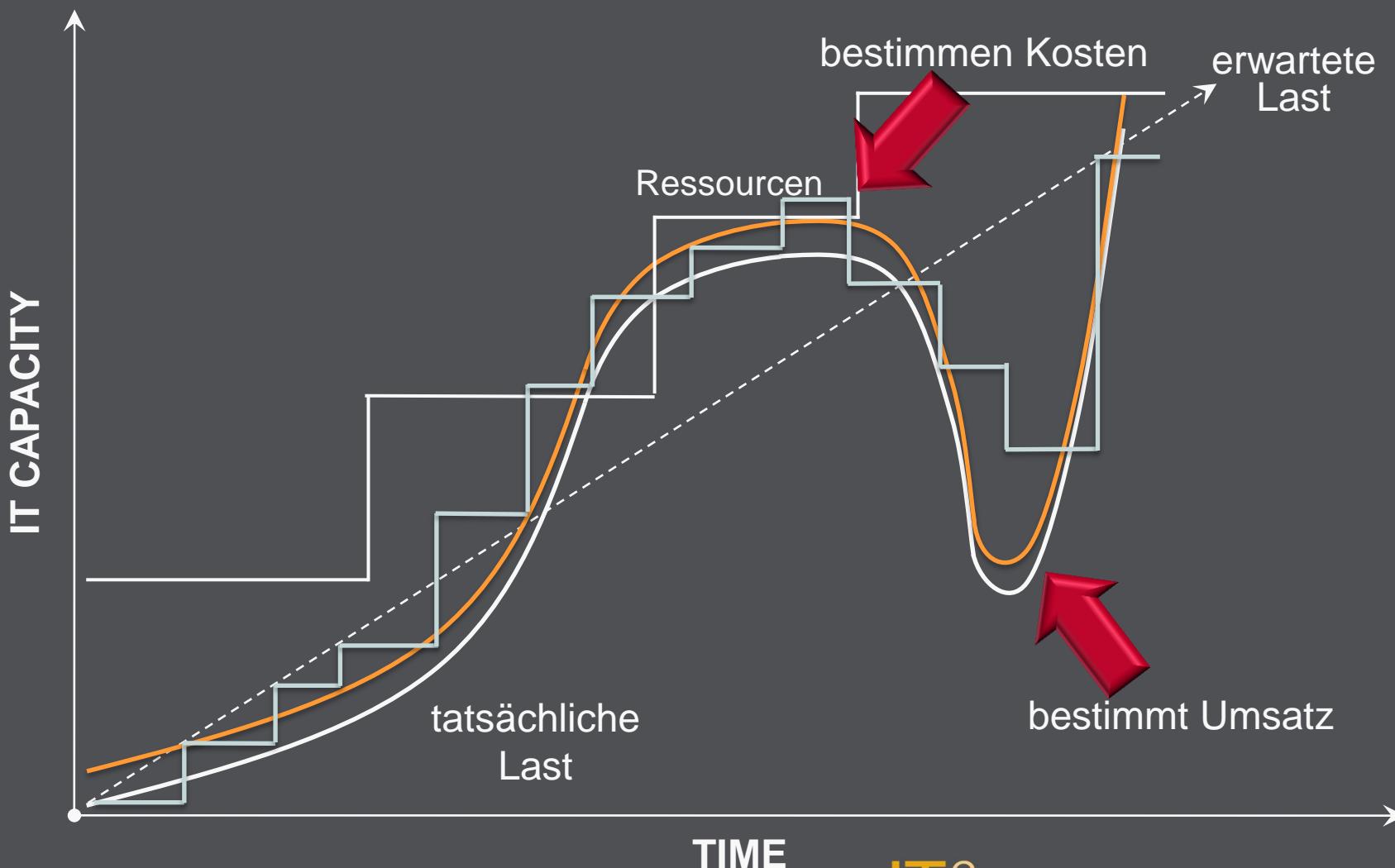
Just another  
**Server Cluster**

Just another  
**IIS**

Just another  
**Web Farm**

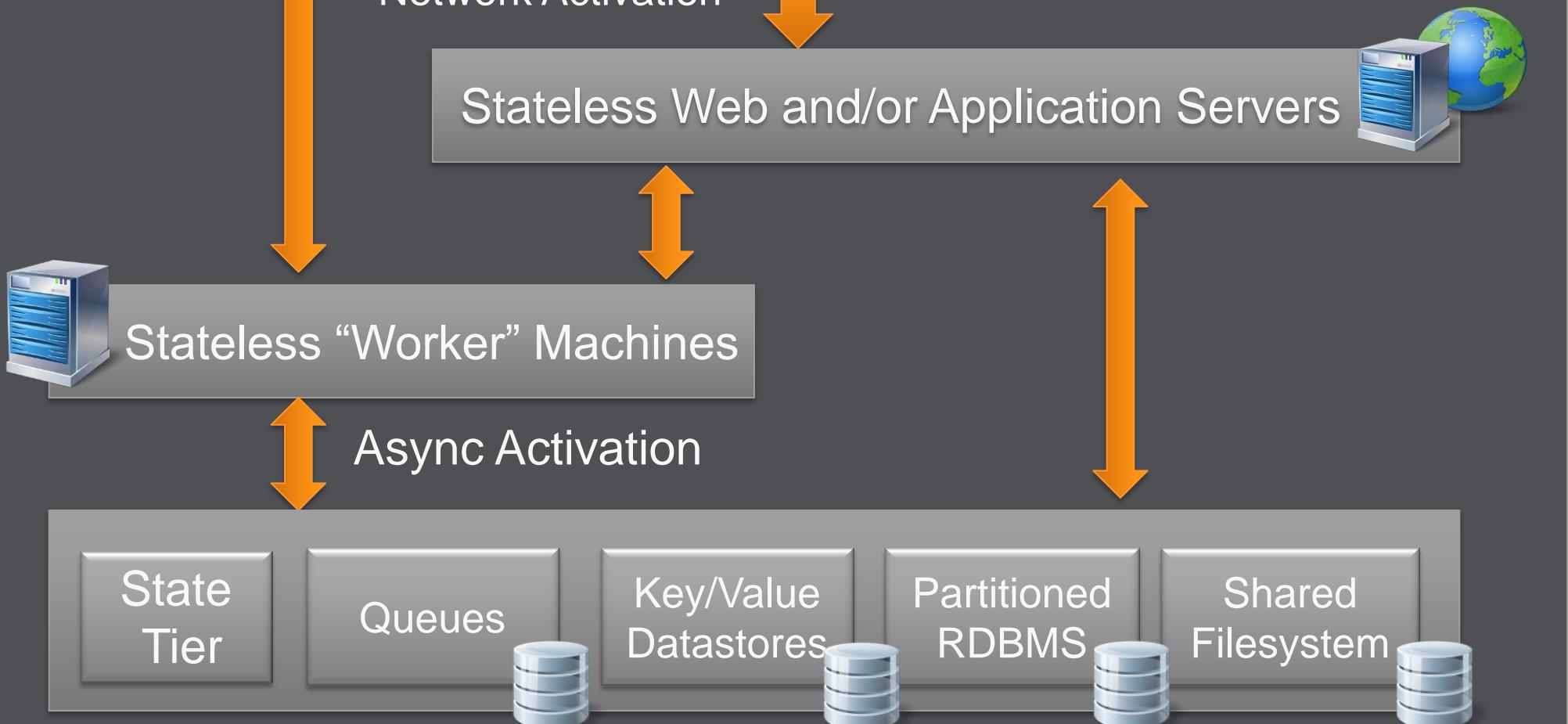
Just another  
**SQL Server**

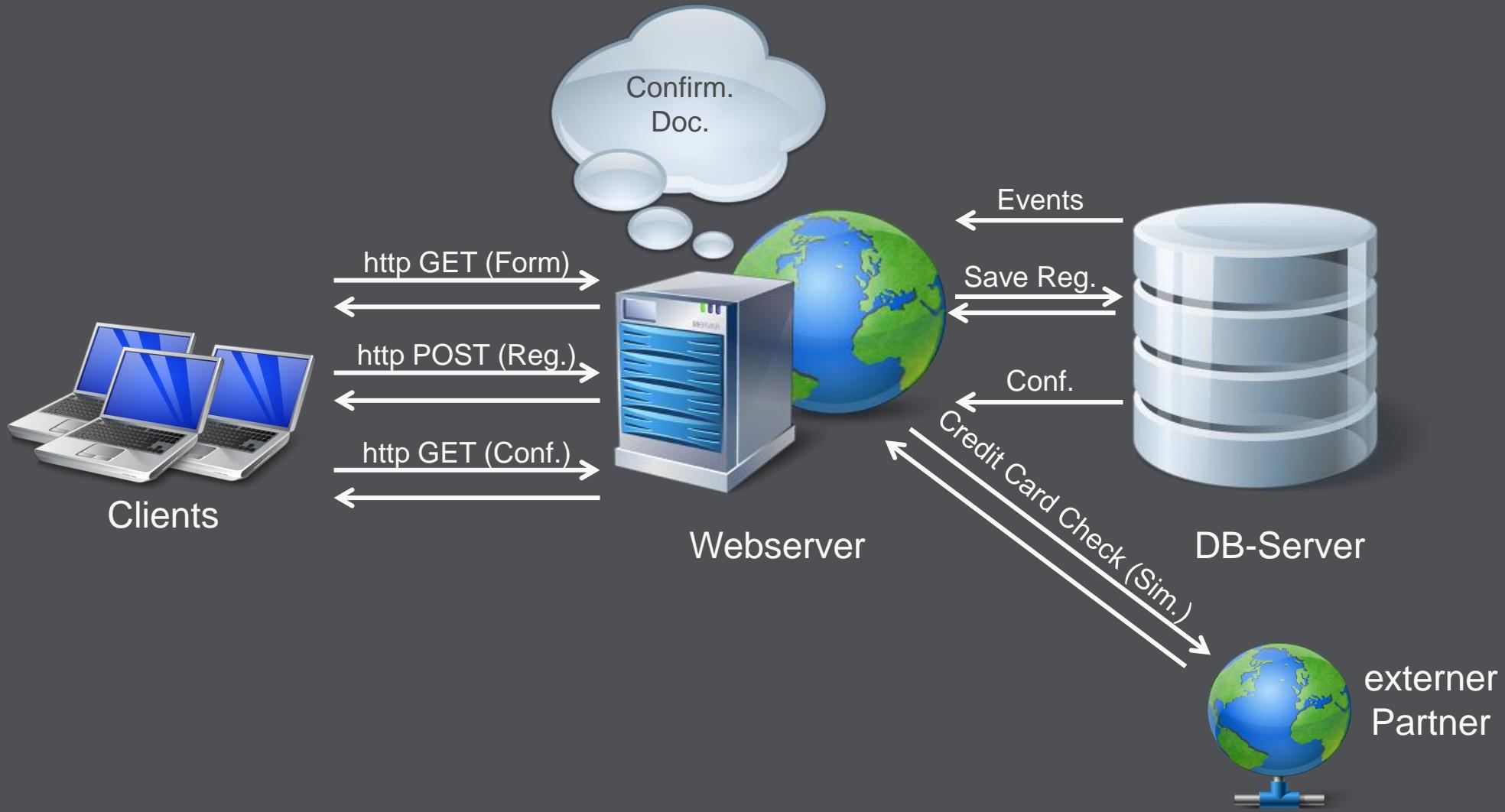
Just another  
**SQL Server Cluster**





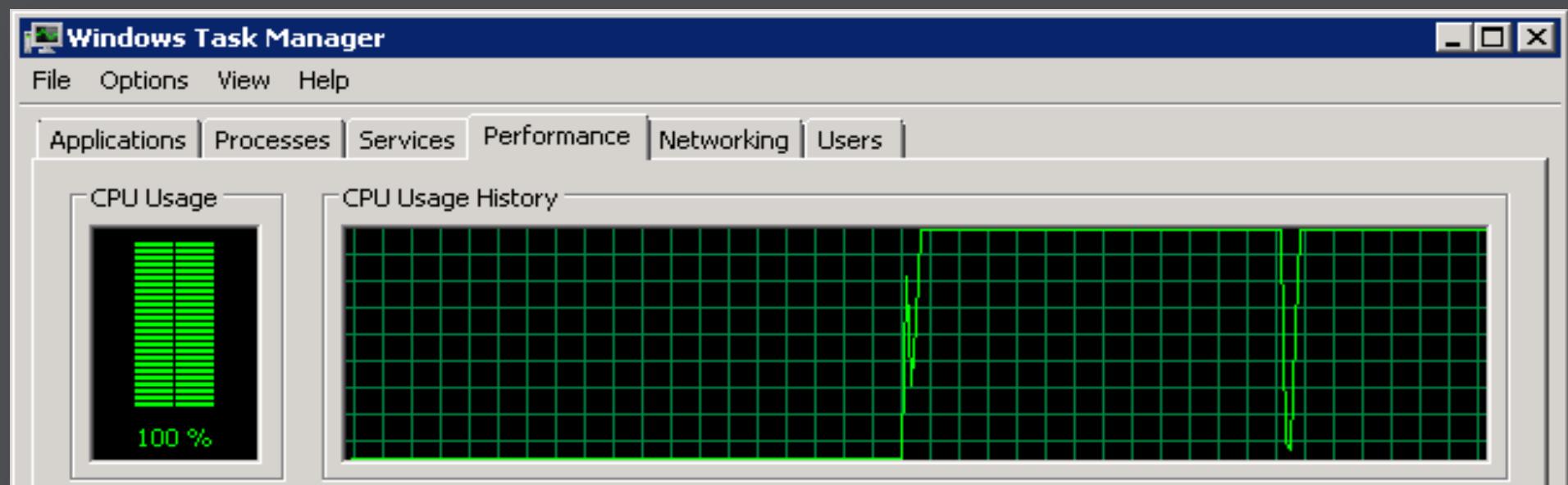
## Intelligent Network Load Balancer

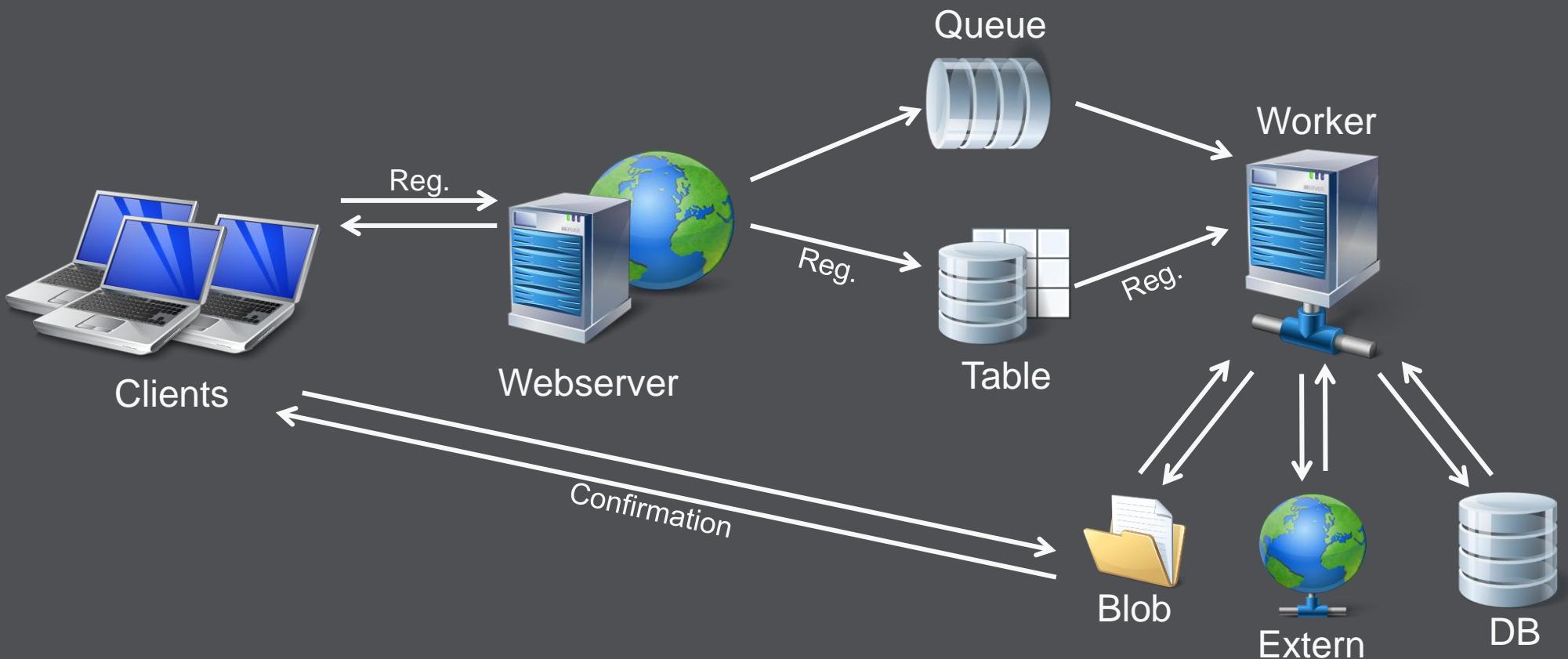




Error code	Resource	Requests	Average response time	Max response time
500	/PerformRegistration.aspx?Eve...dToMonth=01&ValidToYear=2010	7	11.338 s	34.288 s
408	/PerformRegistration.aspx?Eve...dToMonth=01&ValidToYear=2010	693	35.0 s	35.0 s

No response within 35 seconds



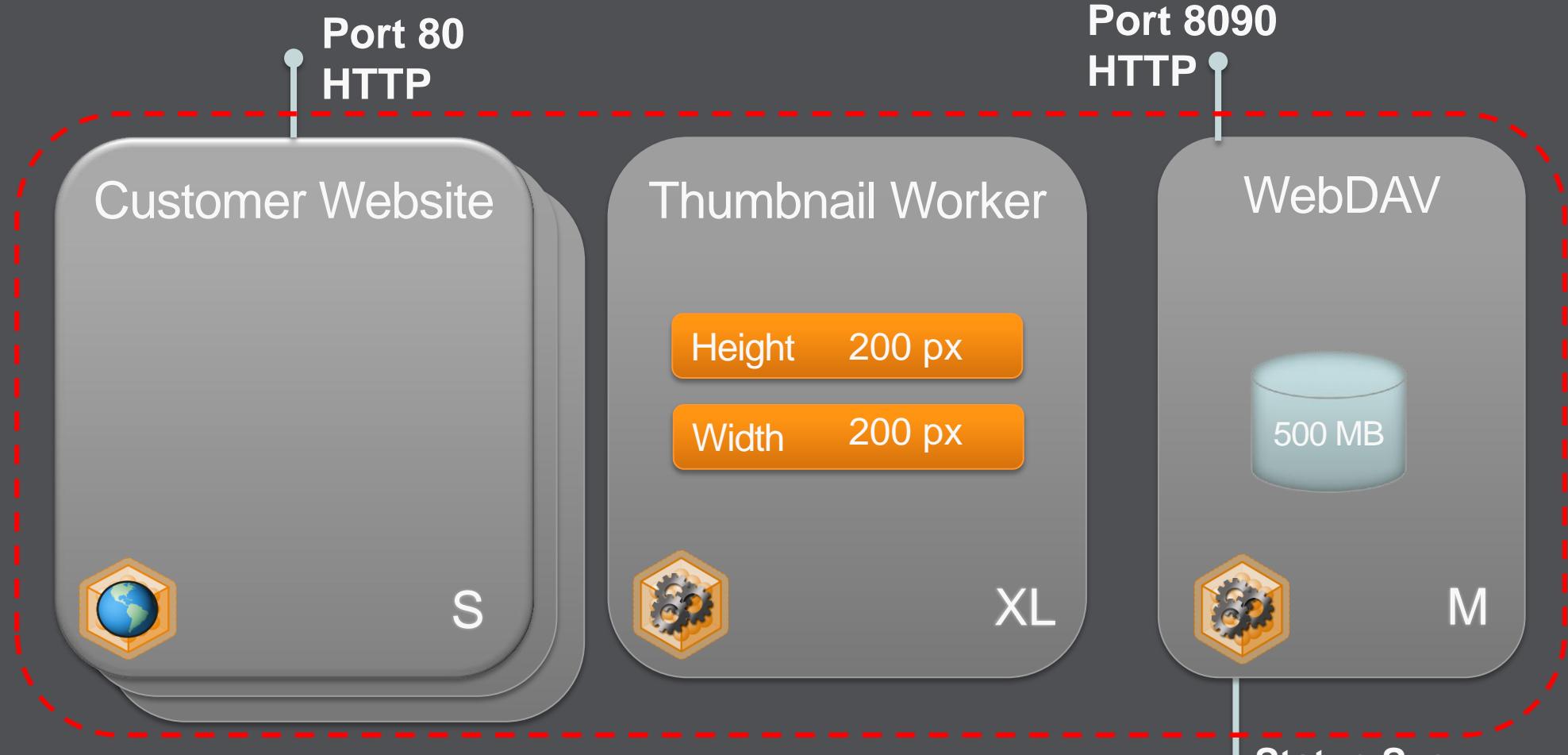


*demo*

## Load-Test auf verteilter Architektur

<http://www.loadstorm.com>

# Service, Rollen, Instanzen



Service Definition

IT & Dev CONNECTIONS  
powered by Microsoft®

*demo*

# Walkthrough: Konfigurationseinstellungen

.cscfg und .csdef statt web.config



In Azure **zahlt** man dauernd,  
selbst während der Entwicklung!

# Windows Compute Emulator

- *Windows Azure Compute Emulator* aka **DevFabric**
  - Teil des Windows Azure SDK → kostenlos
- Simuliert Windows Azure während des Entwicklungsprozesses
  - Zum Debugging
  - Zur Kostenreduktion
  - Für Offline-Szenarien
- Emulator ≠ Windows Azure
  - Emulator kann auf lokale Ressourcen zugreifen
  - Lokale Ressourcen stehen in der Cloud eventuell nicht zur Verfügung
  - Test im Emulator ersetzt Test in der echten Cloud nicht!

# Windows Compute Emulator

- Voraussetzungen
  - [Windows Azure SDK](#) und Azure-Tools für VS
  - Visual Studio 2010
  - IIS und SQL Server 2008 R2 (siehe auch [MSDN](#))
- Installation
  - Installation von SDK und Tools
  - Emulator konfigurieren (siehe auch [MSDN](#))
- Emulatoren unterstützen nur lokalen Zugriff
  - Tipp: Verschiedene Blogartikel beschreiben, wie Emulatoren über das Netzwerk zum Testen verfügbar gemacht werden können (z. B. [Emmanuel's Blog](#))

*demo*

# Windows Compute Emulator

Lokale Entwicklungsumgebung für die Cloud

The background of the slide is a dense, overlapping pile of ripe peaches, showing their characteristic orange-yellow color and slight texture. A white rectangular box containing the main text is centered over this fruit.

**Cloud Computing ist auch nichts anderes als Hosting – alter Hut!**

# Service Deployment

Service

Model

D  
N  
S

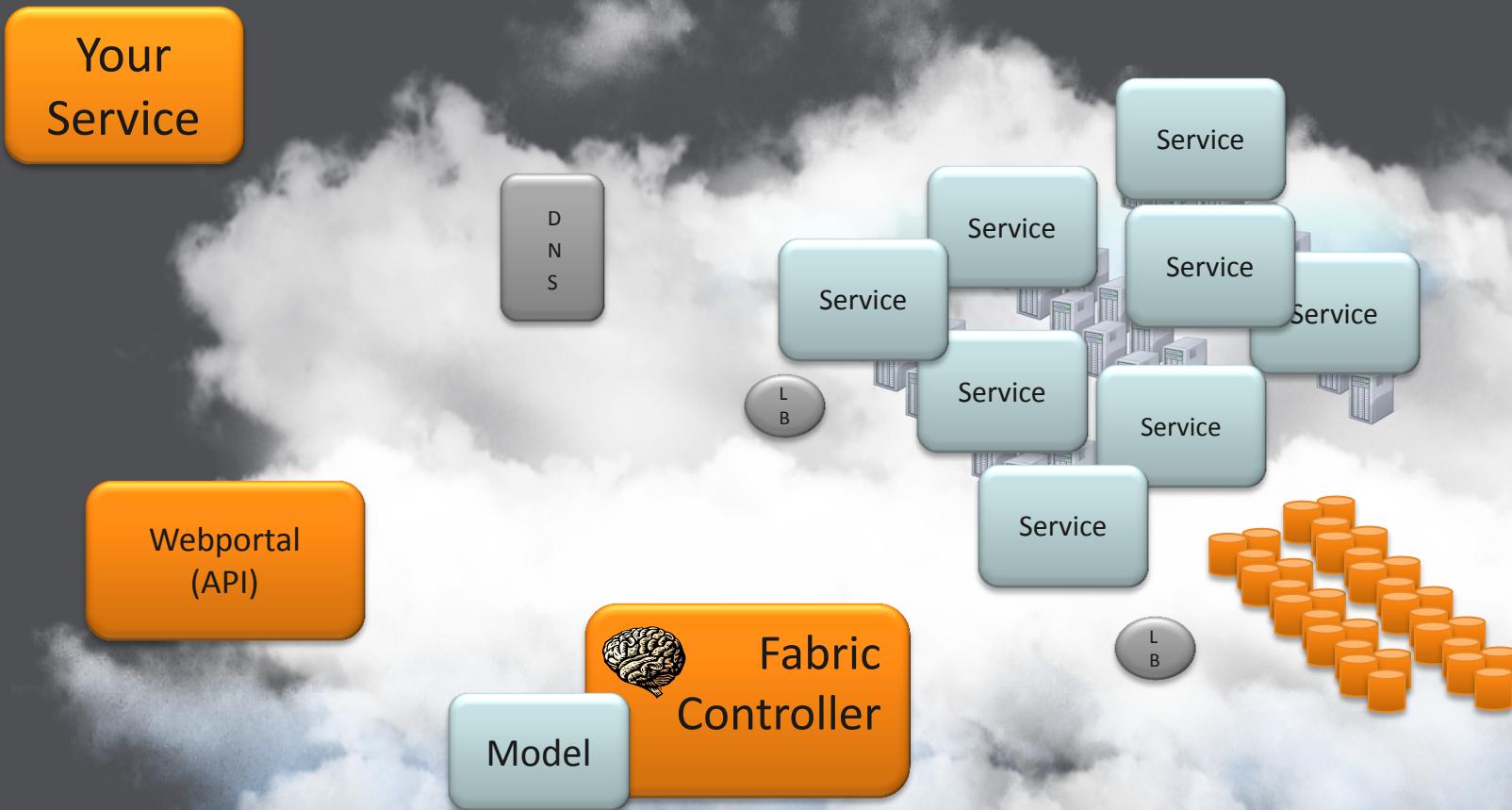
Webportal  
(API)



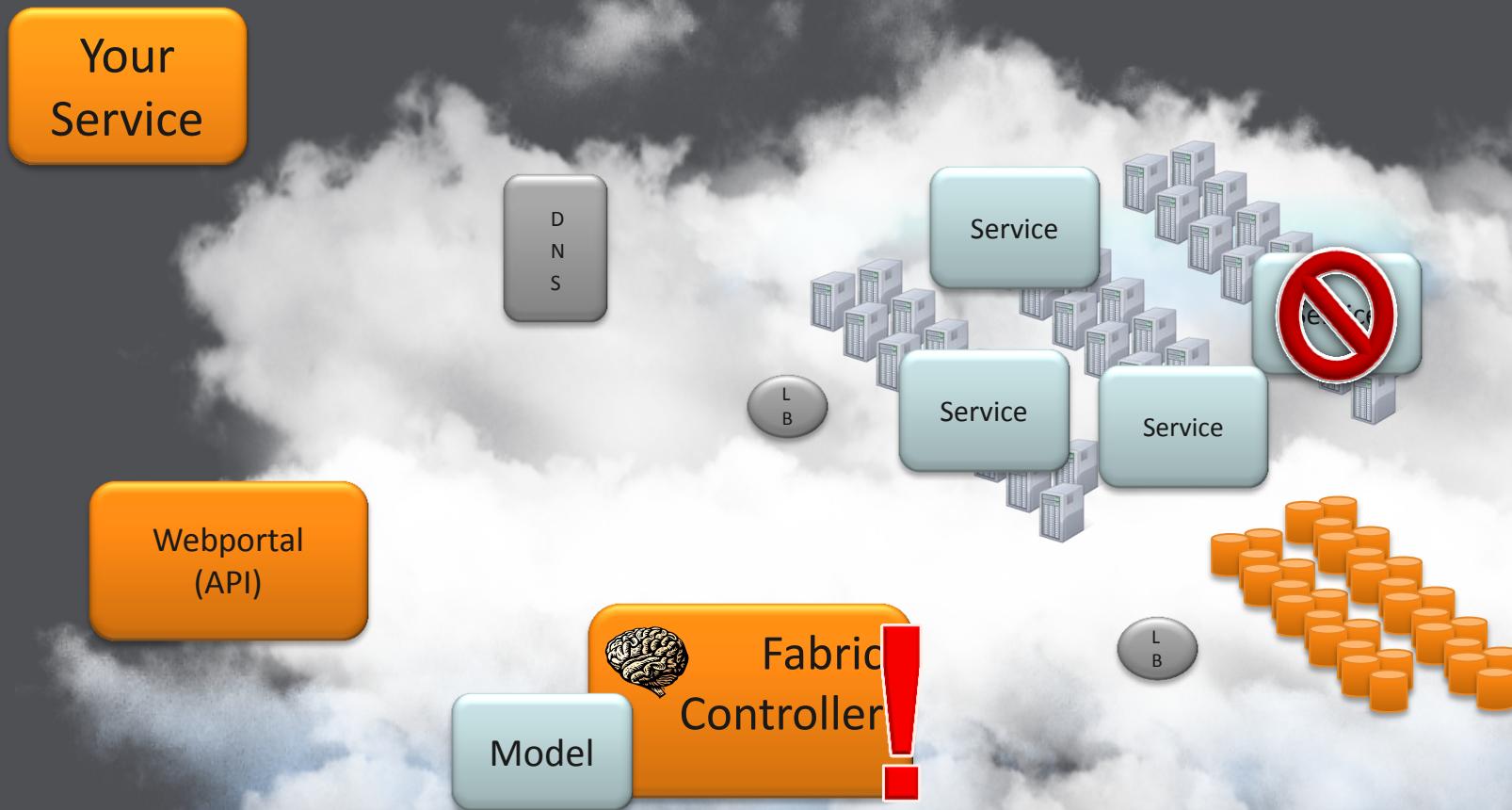
L  
B

L  
B

# Service Scaling



# Service Monitoring & Recovery



*demo*

Webfarm

Windows Azure Compute Emulator

# **BigMat**

MATERIALS BANYOLES

MAGAZINS PROFESSIONALS  
PER A LA CONSTRUCCIÓ

C/ Sant Andreu, 103-111 - 17834 BANYOLES  
Tel. 972 571 817 - Fax 972 580 940  
E-mail: sumiampu@bigmat.es



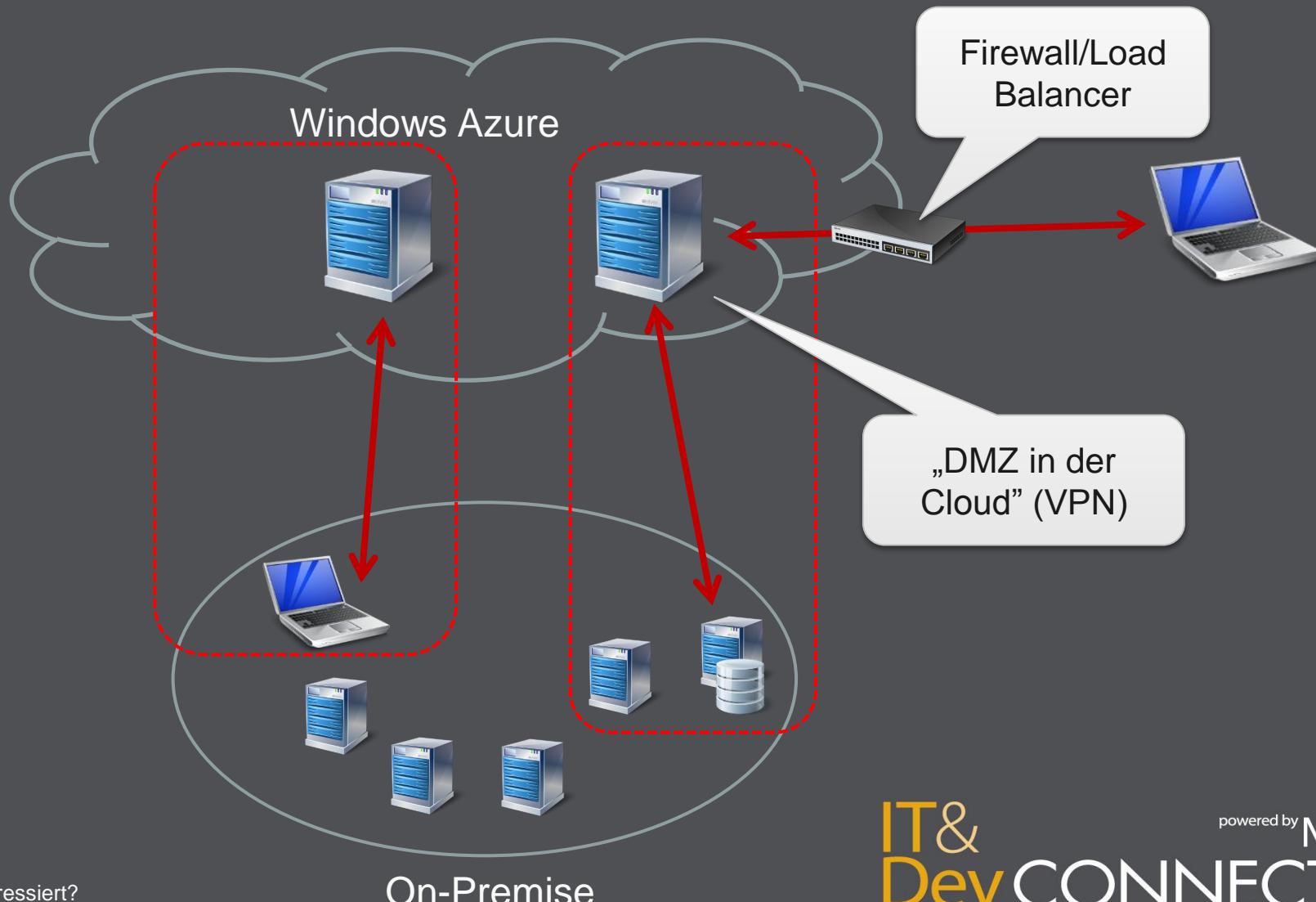
**PROHIBIT EL PAS A TOTA  
PERSONA ALIENA A L'OBRA**

**Cloud Computing ist ein Sicherheitsrisiko!**

# Informationssicherheit

- Datenschutz
  - Gesetzliche Rahmenbedingungen beachten
  - Mögliche Lösung: Windows Azure Connect
- Datensicherheit
  - In der Cloud möglicherweise deutlich besser
  - Vertraulichkeit – Zertifikate, Verschlüsselung etc.
  - Verfügbarkeit – Cluster, Ausfallszenarien etc.
  - Physischer Zugriffsschutz

# Windows Azure Connect (in CTP)



An Details interessiert?

<http://msdn.microsoft.com/en-us/gg502176>

*demo*

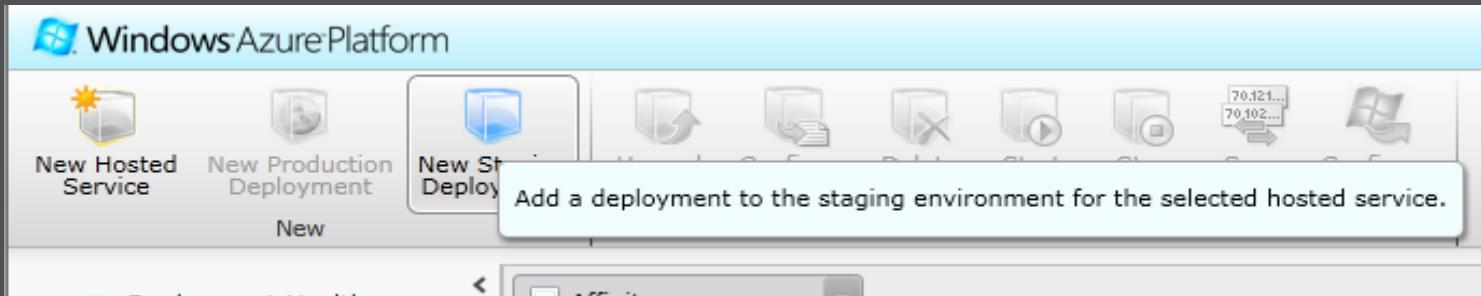
## Windows Azure Connect

Aus der Cloud auf on-premise Ressourcen zugreifen



**Das Deployment in Azure ist so kompliziert –  
nicht verwendbar!**

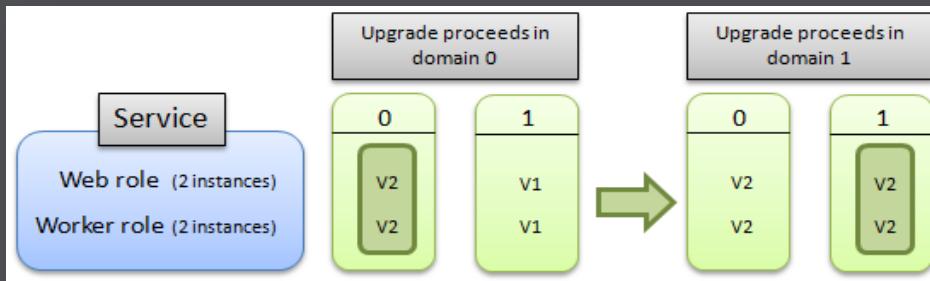
# Trennung in „Production“ und „Staging“



- **Production**
  - `http://<myapp>.cloudapp.net`
- **Staging**
  - `http://<guid>.cloudapp.net`
  - Zum Testen und zum Vorbereiten einer neuen Version

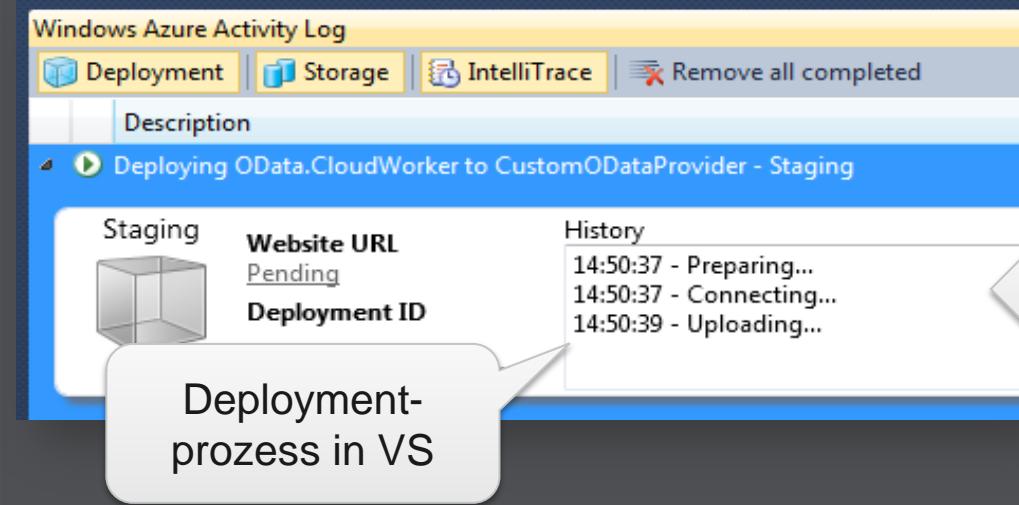
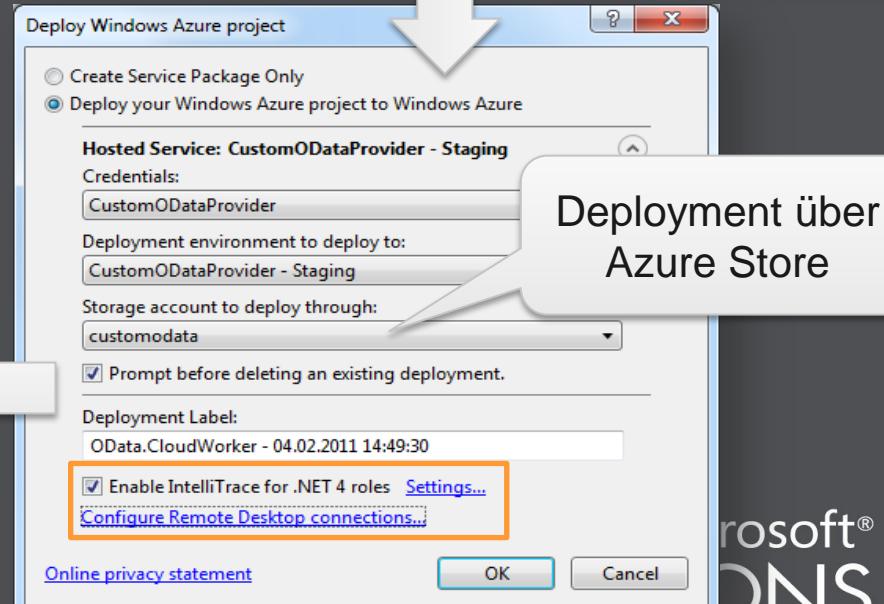
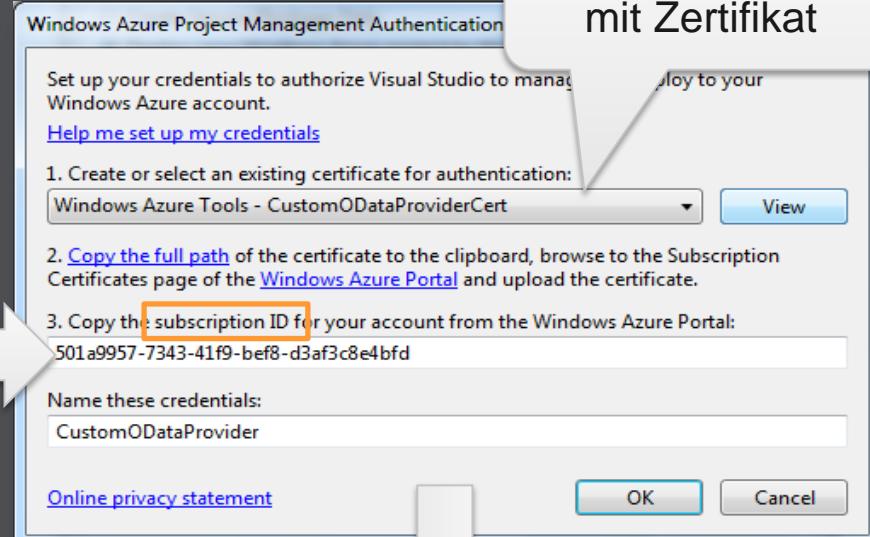
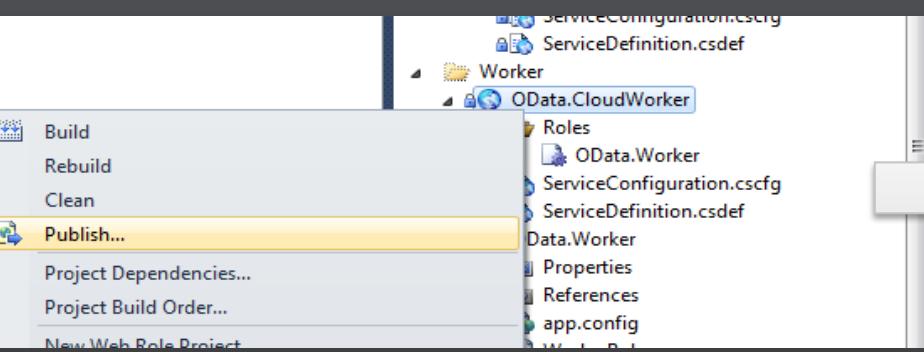
# Typen von Deployments

- **In-Place Update**
  - Kann auf Production und Staging angewandt werden
  - Servicemodell darf sich nicht unterscheiden (z. B. gleiche Anzahl Rollen)



- **VIP Swap (Virtual IP Swap)**
  - Schaltet um zwischen Prod ↔ Staging
  - Servicemodell darf sich ändern; Endpoints müssen ident bleiben

# Deployment mit VS



Deployment-  
prozess in VS

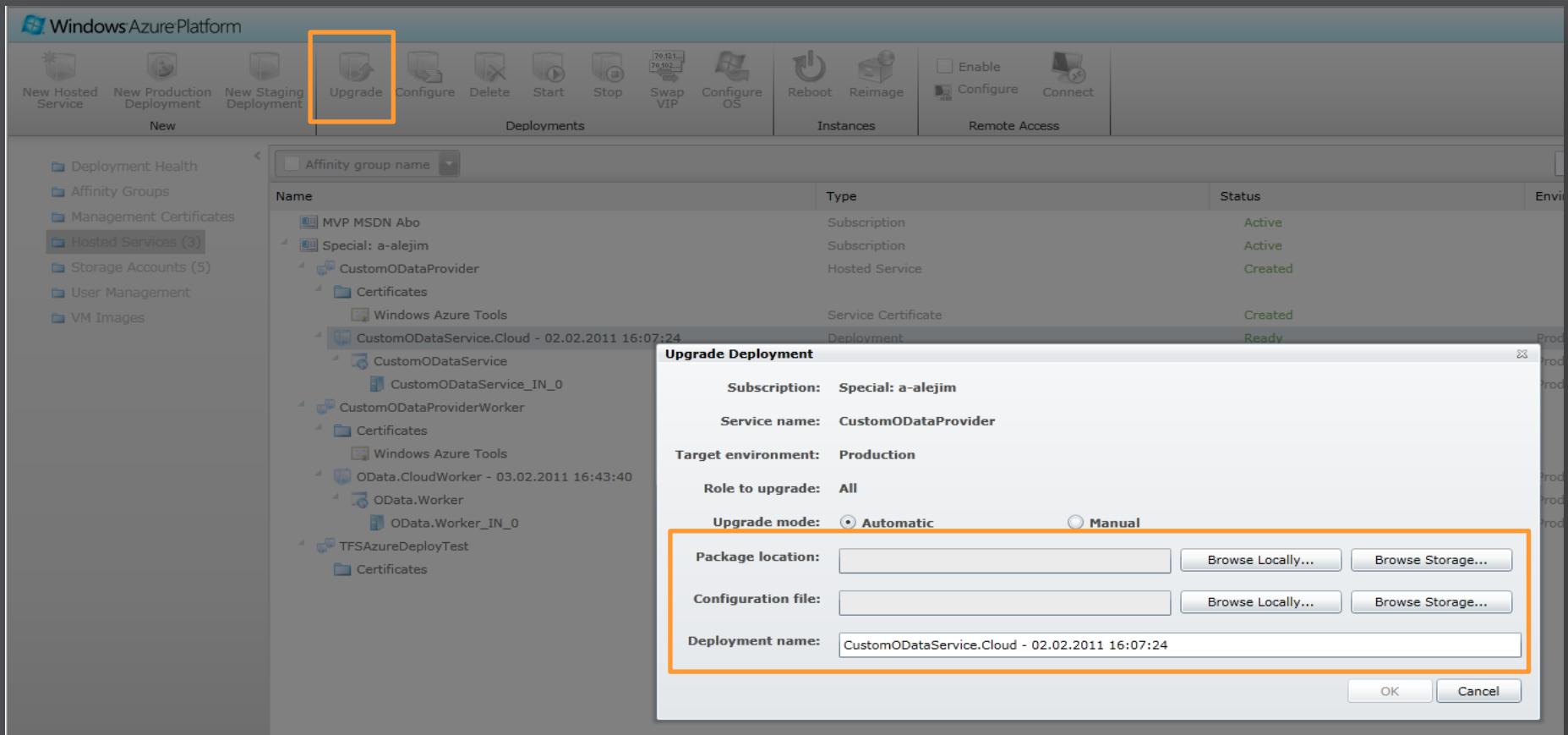
Authentifizierung  
mit Zertifikat

Deployment über  
Azure Store

Microsoft®  
Windows Azure

# Deployment über das Portal

## <http://windows.azure.com>



# VIP Swap



- Prod enthält V<sub>x</sub>
- Deploy V<sub>next</sub> nach Staging
  - Staging-Umgebung mit Staging-DB verbinden
  - Finale QS
  - Staging-Umgebung mit Prod-DB verbinden
  - Finale QS, Warm-up
- VIP Swap durchführen, jetzt ist V<sub>next</sub> online
- Staging stoppen **und löschen**

*demo*

VIP Swap

Prod ↔ Staging

# Windows Azure SDK-Deployment-Tools

- CSPack.exe
  - Deploymentpaket für Emulator oder Azure-Deployment
  - Wird typischerweise mit Visual Studio gemacht
- CSRun.exe
  - Pakete in Emulator deployen und starten
  - Wird typischerweise mit Visual Studio gemacht
  - Tipp: Tester können CSRun verwenden, um Anwendungen ohne VS und Source-Code zu starten
- CSUpload.exe – VHDS in Azure laden
- CSManage.exe
  - Tool (Sample) zum Automatisieren von Azure-Servicetätigkeiten
  - Beispiele: Deployment, Auto-Scaling etc.

*demo*

## Anwendung mit CSRun starten

Automatisches Deployment in den Emulator

# Für die Profis ...

- Generische Rollen, die Assemblies zur Laufzeit z. B. aus dem Blob-Store nachladen
- Möglich mit Web- und Worker-Rolle
  - Relativ einfach: Worker-Rolle
  - Relativ schwierig: Web-Rolle
- Tipp: eigene AppDomains zum leichteren Entladen der Assemblies
- Weitere Literatur
  - [Blogartikel von Rainer Stropek](#)
  - [AppDomain-Klasse](#)
  - [Assembly-Klasse](#)



**Entweder alles in Azure oder nichts –  
viel zu hohes Risiko!**

# „Design to Cost“

- Suchen und nutzen Sie Quick Wins
- Jede Komponente kann einzeln genutzt werden
  - Blob-Store für Backup
  - Table-Store für Logging
  - Queues und AppFabric zur Anwendungsintegration
  - Webserver mit On-Premise-DB
  - SQL Azure zur Datenintegration über Standorte hinweg
  - ...
- Kreatives Kombinieren der einzelnen Komponenten ist gefragt



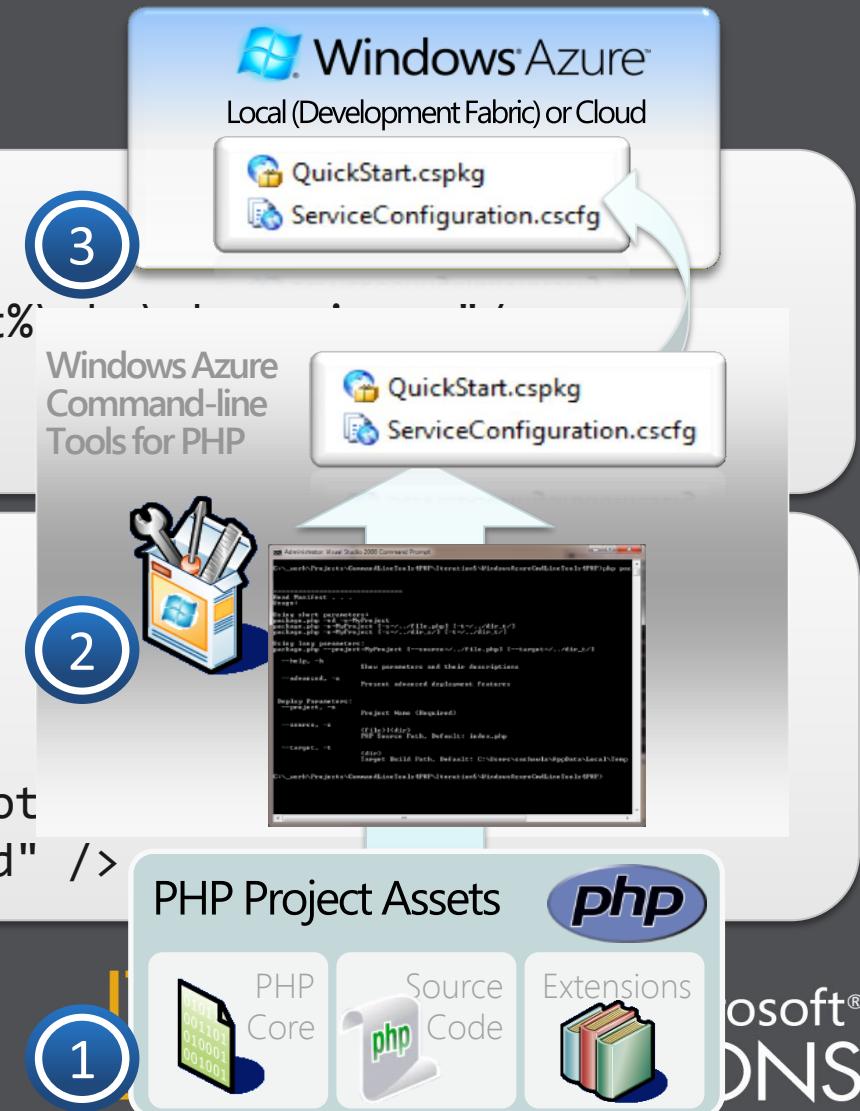
Azure ist **proprietär**,  
nur Windows und .NET!

# „Just another IIS“

- Z. B. PHP über FastCGI

```
<system.webServer>
  <fastCgi>
    <application fullPath="%RoleRoot%">
  </fastCgi>
</system.webServer>
```

```
<add name="PHP via FastCGI"
      path="*.php"
      verb="*"
      modules="FastCgiModule"
      scriptProcessor="%RoleRoot%"
      resourceType="Unspecified" />
```



# PHP-SDK für Windows Azure

- Wrapper Libraries für REST-Services
- Hilfsklassen für Azure-Storage

```
require_once 'Microsoft/WindowsAzure/Storage/Blob.php';

$storageClient = new Microsoft_WindowsAzure_Storage_Blob();

$result = $storageClient->putBlob('testcontainer', 'example.txt',
'/home/maarten/example.txt');
```

- SQL Server-Treiber, Diagnostics
- AppFabric-SDK



NO ALCOHOLIC  
BEVERAGES



NO  
SMOKING



NO GLASS  
CONTAINERS



NO DOGS  
OR PETS



NO MOTOR  
VEHICLES



NO  
LITTERING



NO OPEN  
FIRES



NO  
CAMPING



NO  
DIVING



NO  
BASEBALL



NO BOAT  
LAUNCHING

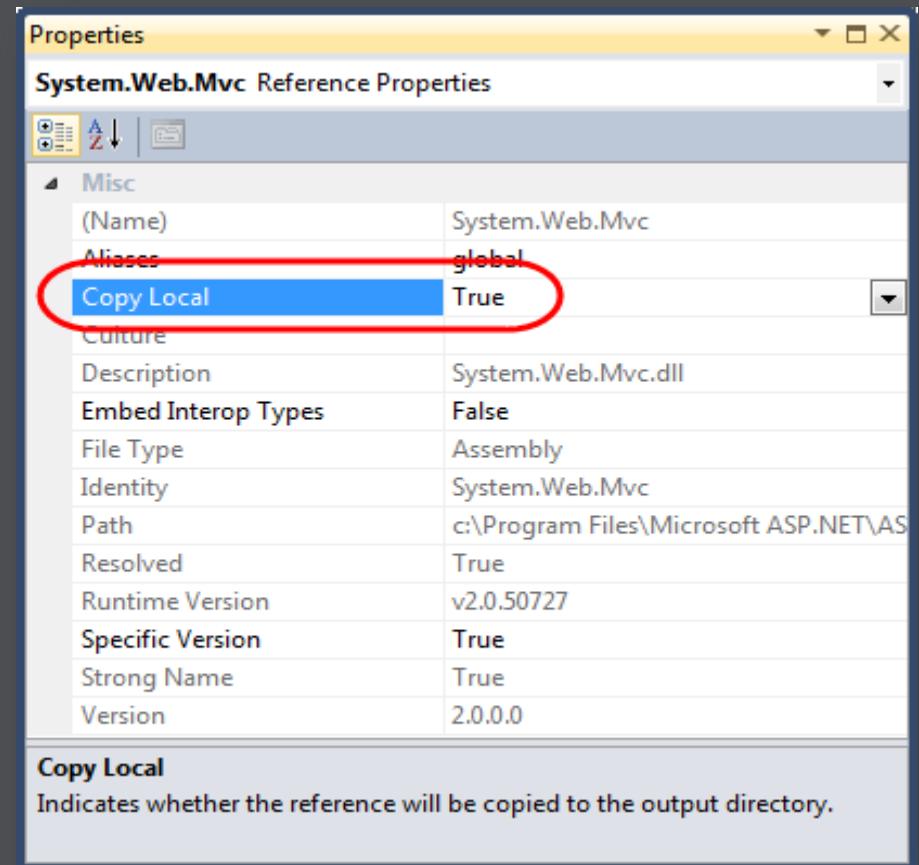


NO  
VENDORS

In Azure kann man  
**keine Drittkomponenten verwenden!**

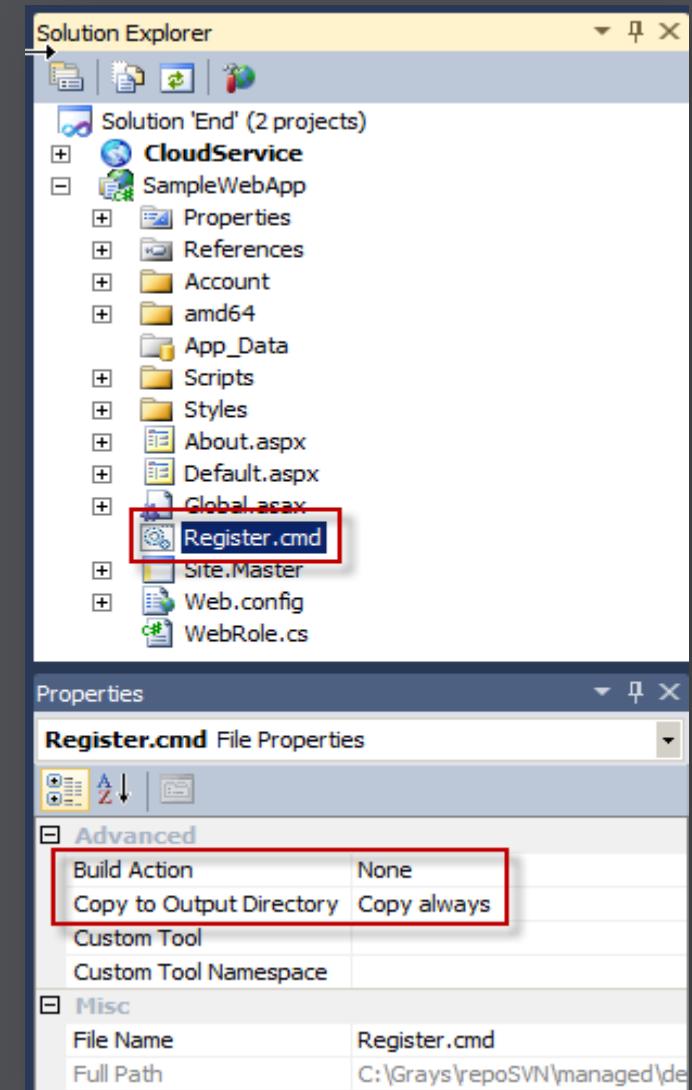
# Assemblies

- Copy Local = true für zusätzliche, referenzierte Assemblies
- Sonstige Dateien deployen:  
siehe [Adding Files to your Windows Azure Service Package](#)



# Startup Scripts

```
<ServiceDefinition [...]>
  <WebRole name="SampleWebApp">
    <Startup>
      <Task commandLine="Register.cmd"
            executionContext="elevated"
            taskType="simple" />
    </Startup>
  </WebRole>
</ServiceDefinition>
```



*demo*

# Unmanaged Code in Azure

Elevated Startup Tasks

# Wann VM-Rolle?

- Lange laufende Installationsroutinen
- Installationsroutinen, die nicht automatisiert werden können
  - Manueller Eingriff notwendig
- Instabile Installationsroutinen

# Die VM-Rolle ist nicht ...

- ... Infrastructure as a Service
- ... eine Plattform für Servervirtualisierung
- ... eine Plattform für Desktopvirtualisierung

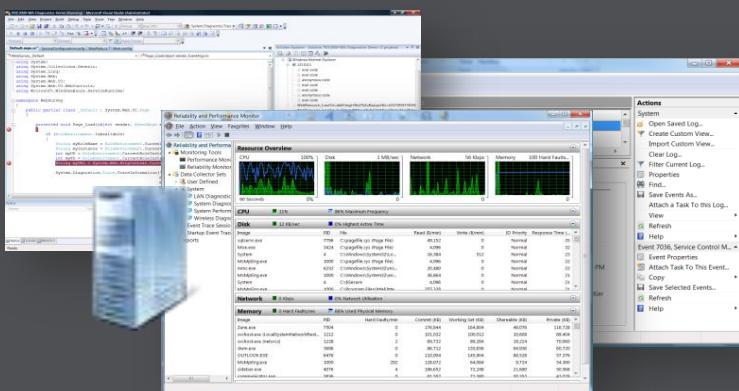


Azure ist ein **schwarzes Loch** –  
man sieht nicht hinein!

# Monitoring On-Premise vs. Cloud

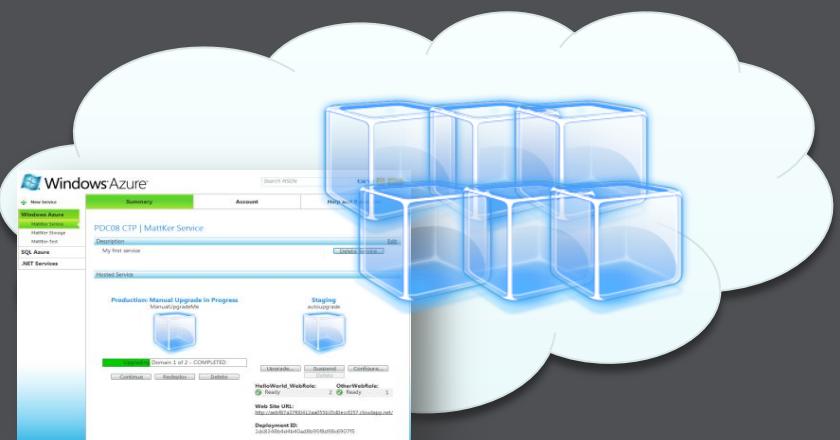
## On-Premise

- Statische Umgebung
- Gut bekannte Umgebung
- Meist wenige Server



## Cloud

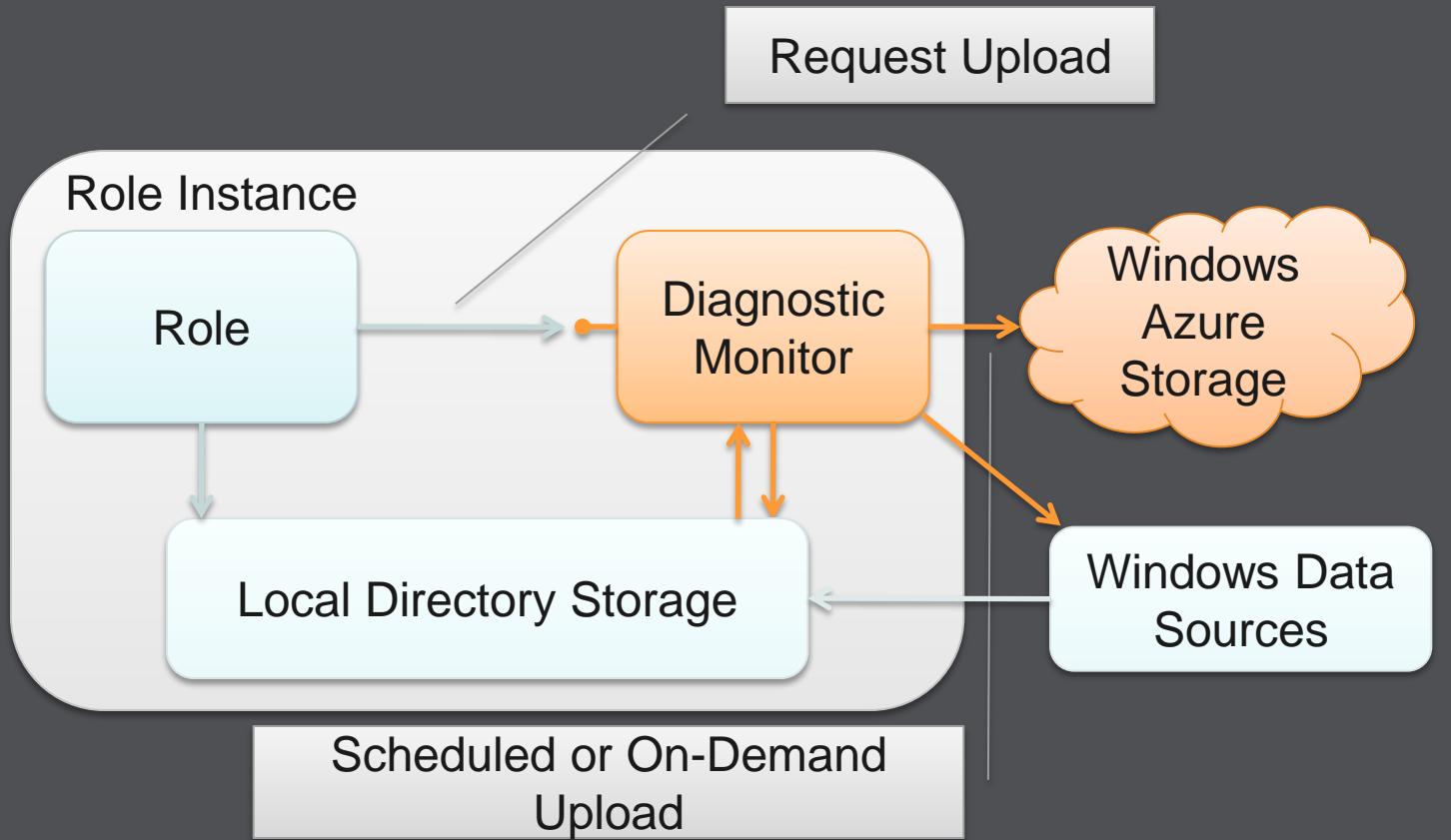
- Dynamische Umgebung
- Viele Instanzen, elastisch
- Viele Knoten



# RDP vs. Diagnostics

- Remote Access mit RDP
  - Während der Entwicklung
  - Troubleshooting in speziellen Situationen (z. B. Speicher- oder CPU-Bedarf)
- Diagnostics
  - Laufendes Monitoring
  - Langzeitstatistiken („Telemetrie“)
  - Überwachung des Gesamtsystems steht im Vordergrund

# Azure Diagnostics



# Mission Impossible?

```
<?xml version="1.0" encoding="UTF-8" ?>
- <osm version="0.6" generator="pbf2osm">
  <node id="172539" lat="52.5651847" lon="13.3354546" version="9" changeset="5702878" user="Woschl" uid="121042" timestamp="2010-09-06T21:0
  <node id="172540" lat="52.5647252" lon="13.3364064" version="7" changeset="5702878" user="Woschl" uid="121042" timestamp="2010-09-06T21:0
  <node id="172541" lat="52.5655270" lon="13.3362226" version="2" changeset="728814" user="bahnpirat" uid="13203" timestamp="2009-03-03T14:1
  <node id="172542" lat="52.5660003" lon="13.3375554" version="3" changeset="728814" user="bahnpirat" uid="13203" timestamp="2009-03-03T14:1
  <node id="172543" lat="52.5663124" lon="13.3394369" version="4" changeset="3410834" user="toaster" uid="10549" timestamp="2009-12-20T01:32
  <node id="172544" lat="52.5666165" lon="13.3432402" version="5" changeset="3410834" user="toaster" uid="10549" timestamp="2009-12-20T01:32
- <node id="172545" lat="52.5670070" lon="13.3466339" version="5" changeset="5701736" user="Woschl" uid="121042" timestamp="2010-09-06T19:3
  <tag k="highway" v="traffic_signals" />
</node>
- <way id="30770007" version="2" changeset="2121805" uid="6669" user="Elwood" timestamp="2010-09-06T21:0
  <nd ref="172539" />
  <nd ref="172540" />
  <nd ref="172541" />
  <nd ref="172542" />
  <tag k="access" v="permissive" />
  <tag k="highway" v="residential" />
  <tag k="maxspeed" v="5" />
  <tag k="name" v="Wolkenburgweg" />
  <tag k="postal_code" v="14169" />
</way>
- <way id="30770008" version="3" changeset="2121805" uid="6669" user="Elwood" timestamp="2010-09-06T21:0
  <nd ref="172542" />
  <nd ref="172543" />
  <tag k="access" v="permissive" />
  <tag k="highway" v="residential" />
  <tag k="maxspeed" v="5" />
  <tag k="name" v="Lohrbergweg" />
  <tag k="postal_code" v="14169" />
</way>
- <way id="30770010" version="1" changeset="99086" uid="72235" user="Basstoelpel" timestamp="2010-09-06T21:0
  <nd ref="172544" />
  <nd ref="172545" />
  <tag k="highway" v="footway" />
</way>
</osm>
```

The diagram illustrates the process of migrating OpenStreetMap (OSM) data into a relational database. On the left, a large block of XML code represents the OSM data. An orange arrow points from this code towards a database schema on the right. The database schema is shown in a tree view:

- LoadTesting
- ParallelProgrammingDemo
- Tables
  - System Tables
  - dbo.Highway
    - Columns
      - HighwayID (PK, int, not null)
      - HighwayGeo (geography, null)
      - HighwayType (nvarchar(100), null)
      - StartingNodeID (int, null)
      - EndNodeID (int, null)
    - Keys
    - Constraints
    - Triggers
    - Indexes
    - Statistics
  - Views
  - Synonyms
  - Programmability
  - Security

ft®

*demo*

# RDP in der Worker-Rolle

Import OpenStreetMap in SQL Azure

# Top-10-Tipps für den Weg in Azure

1. Beste Voraussetzung: 100 % Managed Code
2. Design To Cost (DTC)
3. PaaS statt IaaS
4. Scale out, not up
5. Asynchrone Verarbeitung
6. Optimieren ist sexy!
7. Load Testing
8. Multi Tenancy
9. Skaleneffekte durch Automatisieren und Standardisieren nutzen
10. Lernen, lernen, lernen

# Your Feedback is Important

Please fill out a session evaluation form.

Thank you!