DCGI

DEPARTMENT OF COMPUTER GRAPHICS AND INTERACTION

WA2 | PRACTICE B

Windows Azure Platform Introduction

PRACTICE OUTLINE

- Test
- Windows Azure data management

WA | SESSION MANAGEMENT

- Table Storage
  - + Low cost, - might be slow
- SQL Azure
  - + Low cost, + Faster than table, - maintenance
- Hosting In-Role Cache on Dedicated Roles (Windows Azure Cache)
  - dedicated or co-located
  - + Fast, - Extra cost for dedicated
- Windows Azure Cache Service (Preview)
  - - cost

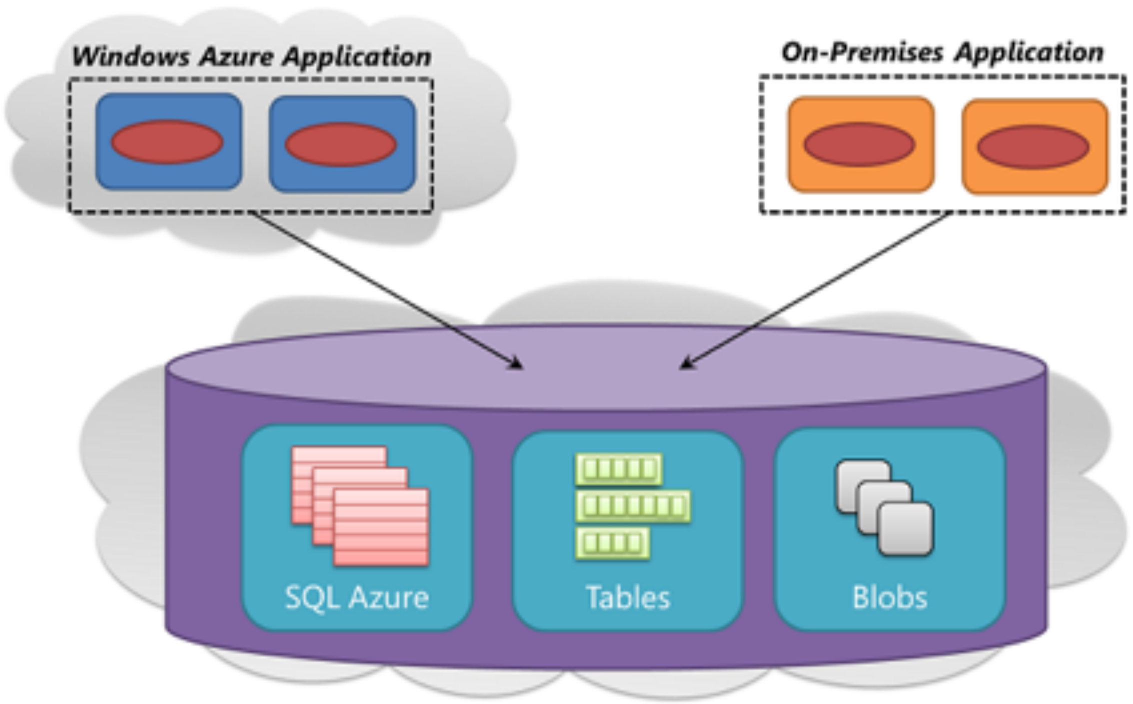
WA | VIRTUAL MACHINE SIZES

instance	CPU Cores	Memory
extra small	shared	768 MB
small	1	1.75 GB
medium	2	3.5 GB
large	4	7 GB
extra large	8	14 GB

WA | DATA MANAGEMENT

- instances are monitored and can be restarted
  - if crashed or unresponsive
  - for maintenance (e.g. OS upgrade)
  - restart deletes locally stored data
- persistent data must be stored out of instance
- solution: SQL Azure, tables, blobs

WA | DATA MANAGEMENT

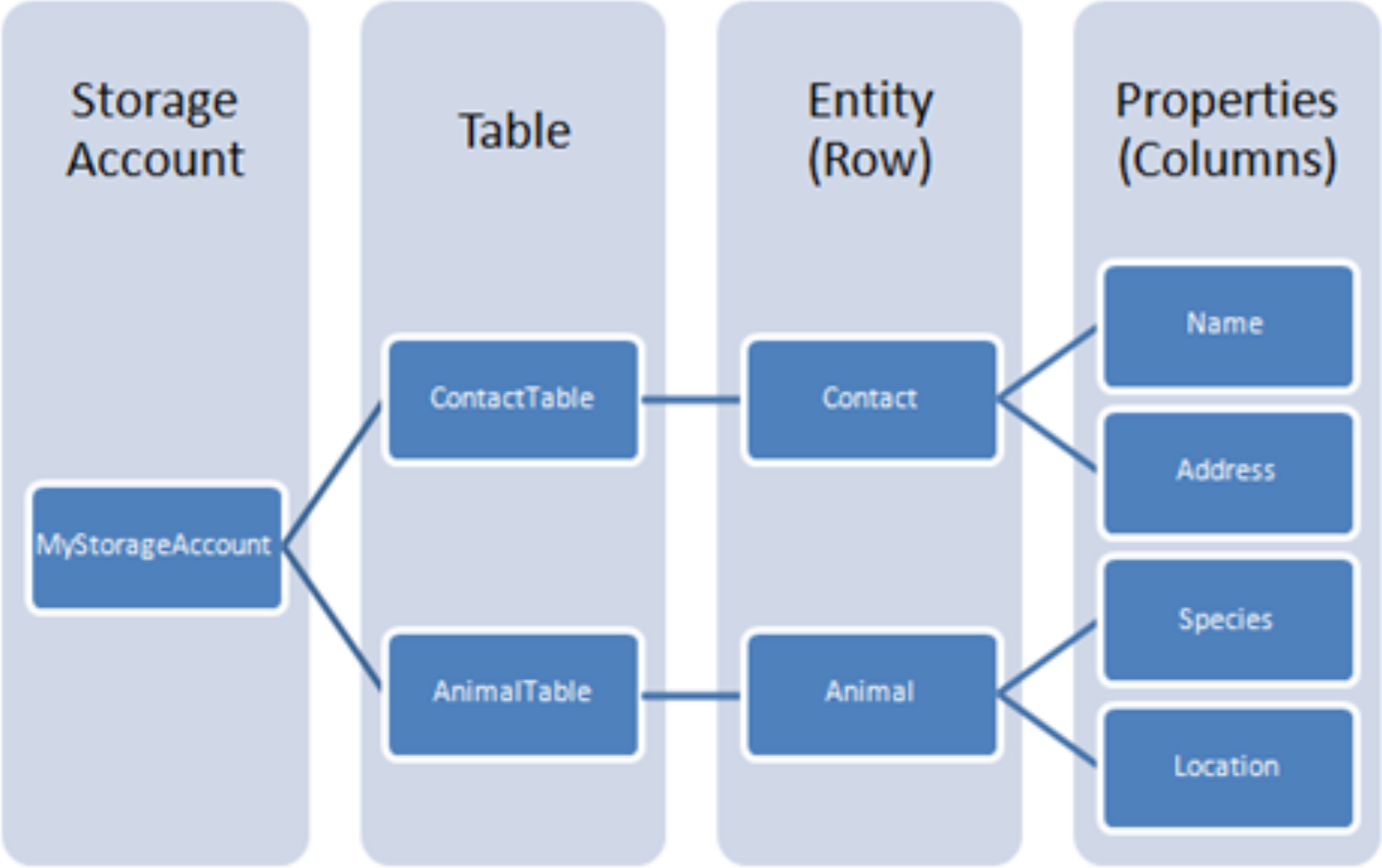


- data automatically replicated (3 comp.)
- access:
  - WA apps - .NET API
  - On-Premises apps - REST API

WA | DATA MANAGEMENT: TABLE STORAGE

- doesn't provide relational storage
- associative storage for objects
- simple but scalable
- less expensive than SQL Azure
- up to 1 TB

WA - DATA MANAGEMENT
TABLES



WA | DATA MANAGEMENT: TABLE STORAGE TRANSACTIONS

- Only entities in same table and same partition group
- Both Rest and .NET CL API
- Requirements
  - Same PartitionKey Value
  - An entity can appear only once
  - up to 100 entities, up to 4MB

WA | DATA MANAGEMENT: BLOB STORAGE

- for unstructured binary data
- e.g. for storing video, images, data backup
- up to 1 TB
- Windows Azure Drives - use blobs for emulating persistent file system

WA - DATA MANAGEMENT | SQL AZURE

- cloud based analogy to SQL Server
- access
  - entity framework
  - ADO.NET
  - other familiar technologies
  - even from On-Premiss apps
- Only subset of data types
- XML-based format for data transfer

TASK 1 | BLOB STORAGE EXAMPLE

- Get solution from previous practice available on Edux
- Implement ISimpleStorage in order to save data into a file stored in Blob storage
  - Create CloudStorageAccount instance.
  - Implement getter Messages {get;}
    - Create instance of Cloud Blob Client
    - Get instance of particular CloudBlobContainer (if exists)
    - Get instance of particular CloudBlob (if exists);
    - Get content and convert it to List<string>, return.
  - Implement method AddMessage
    - Create instance of Cloud Blob Client
    - Get old content (using Messages {get;}
    - Create new content by adding actual message.
    - Get instance of particular CloudBlobContainer (if exists, if not create one)
    - Get instance of particular CloudBlob (if exists);
    - Upload new content into blob

TASK 2 | TABLE STORAGE EXAMPLE

- Implement ISimpleStorage in order to save data into a file stored in the Table Storage
- Implement getter Messages {get;}
    - Get collection of message entries using Select() method.
    - Prepare List<string> collection for output
- Implement method AddMessage
    - Use AddMessage method of data source

THANK YOU

MACIKMIR@FEL.CVUT.CZ