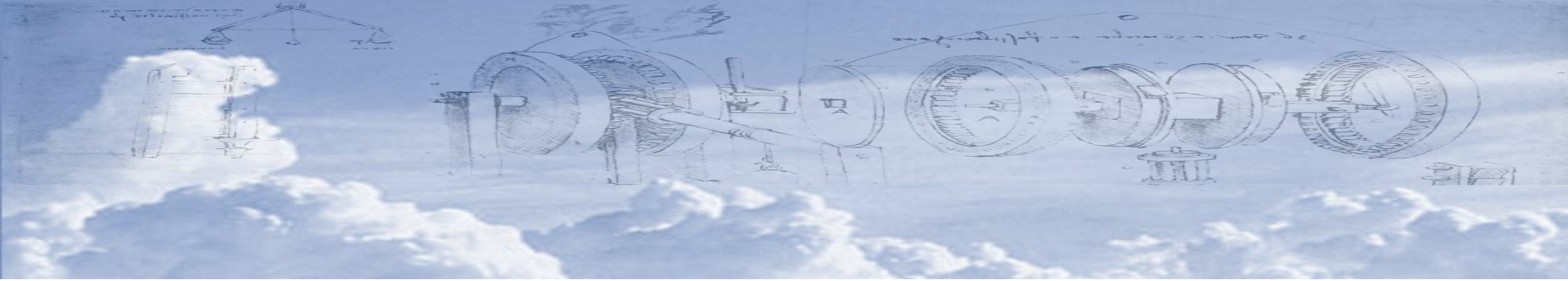


Cloud Computing for Science

Part 1. Managing Data in the Cloud

Dennis Gannon

Motivating Examples

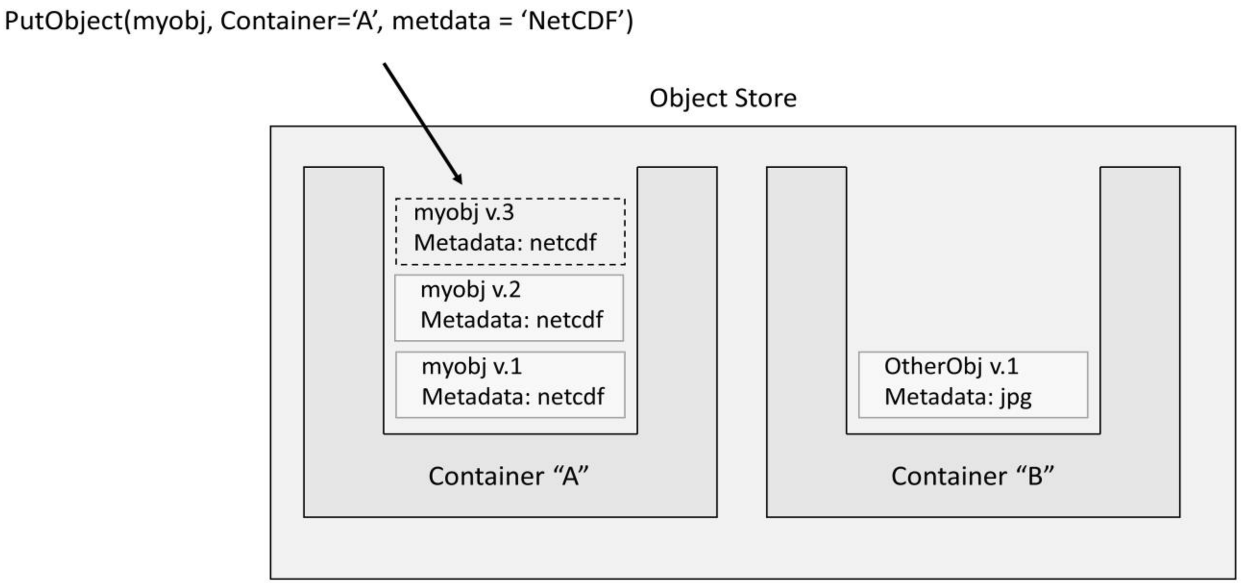
- BIG objects
    - Climate scientists with big simulation output files in NetCDF.
        - Assume 10 TB in big objects
- Many small CVS files
    - Environmental Engineers with 1 Million records of observations each in CSV format
        - May be 100 TB total
- Streams
    - Scientists with a distributed collection of several thousand instruments.
        - Each generates a stream of records that must be collected and analyized every few hours or continuously

Types of Cloud Data Storage Systems

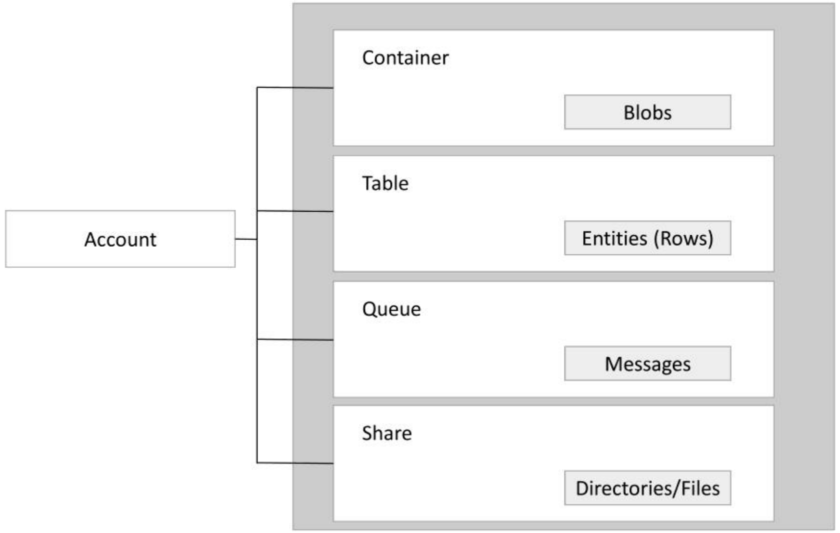
- Basic Blob Object store
  - Buckets of immutable objects
  - Highly scalable & reliable
- Databases
  - SQL Style relational databases
  - NoSQL storage
  - Data warehouses
- Attached File stores
- Graph databases
- Streaming systems

Object stores

- Amazon AWS
  - S3- buckets of immutable objects
    - Organized in a 2-level folder system
    - Each object has associated metadata
- Microsoft Azure
  - Storage accounts contain blob storage along with tables, queues and file shares.
  - Blob containers are similar to S3
- Google Cloud Storage
  - Different models based on availably and cost
- OpenStack does not have a standard but semi-standards exist and various ones are used in various deployment



Azure storage account structure



Relational Databases

- Amazon AWS
  - Relational Data Services (RDS)
  - Aurora
- Microsoft Azure
  - Azure SQL
    - 3 service tiers
    - Premium tier up to 1TB
    - Up to 30000 concurrent sessions
  - Azure Data Lake
- Google Cloud Storage
  - Cloud Spanner (beta)
    - Full relational
    - Strongly consistent
    - Scalable to thousands of servers

- Aurora is distributed
  - Scalable from 2 vCPUs to 32 vCPUs
  - Data up to 64TB
  - MySQL compatible
  - Fully geo-replicated

- Data Lake is a platform
  - Structured & unstructured data
  - Scalable to petabytes
  - Distributed and designed to support analytics

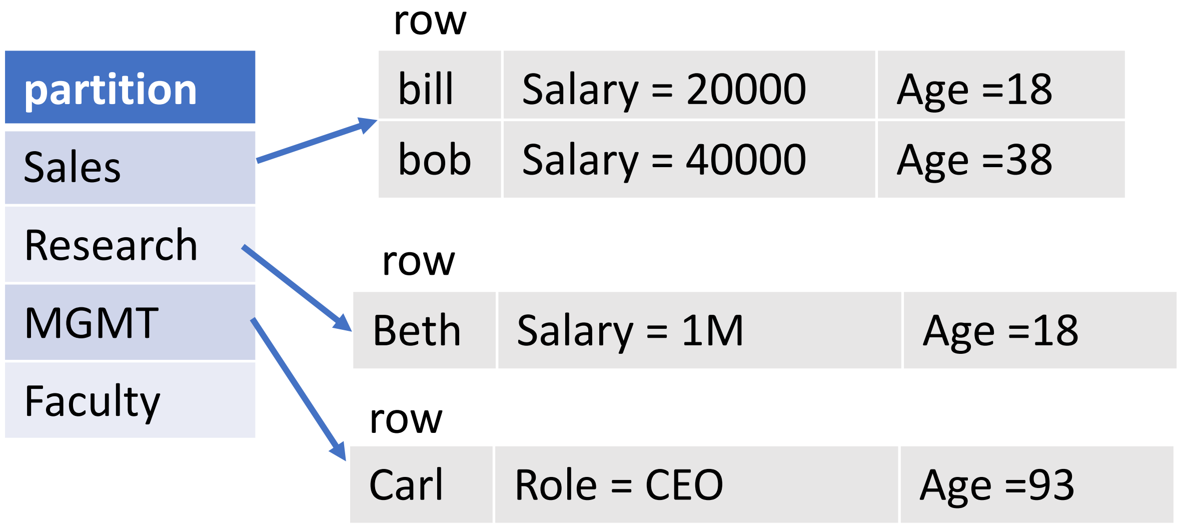
NoSQL Storage Systems

- Main Concepts
  - Designed for massive scale
    - Distributed over many storage nodes
  - Support some SQL operations (not joins)
  - May be only eventually consistent
  - Different types
    - Key-value
    - Column oriented
    - Document style
    - Graph

Relational table

Name	Job	Salary	age
Bill	Sales	$20000	18
Beth	Research	$1m	35
Carl	CEO	0	93
Jill	Prof	$100000	24

NoSQL Key-value system



partition
Sales
Research
MGMT
Faculty

row

bill	Salary = 20000	Age =18
bob	Salary = 40000	Age =38

row

Beth	Salary = 1M	Age =18

row

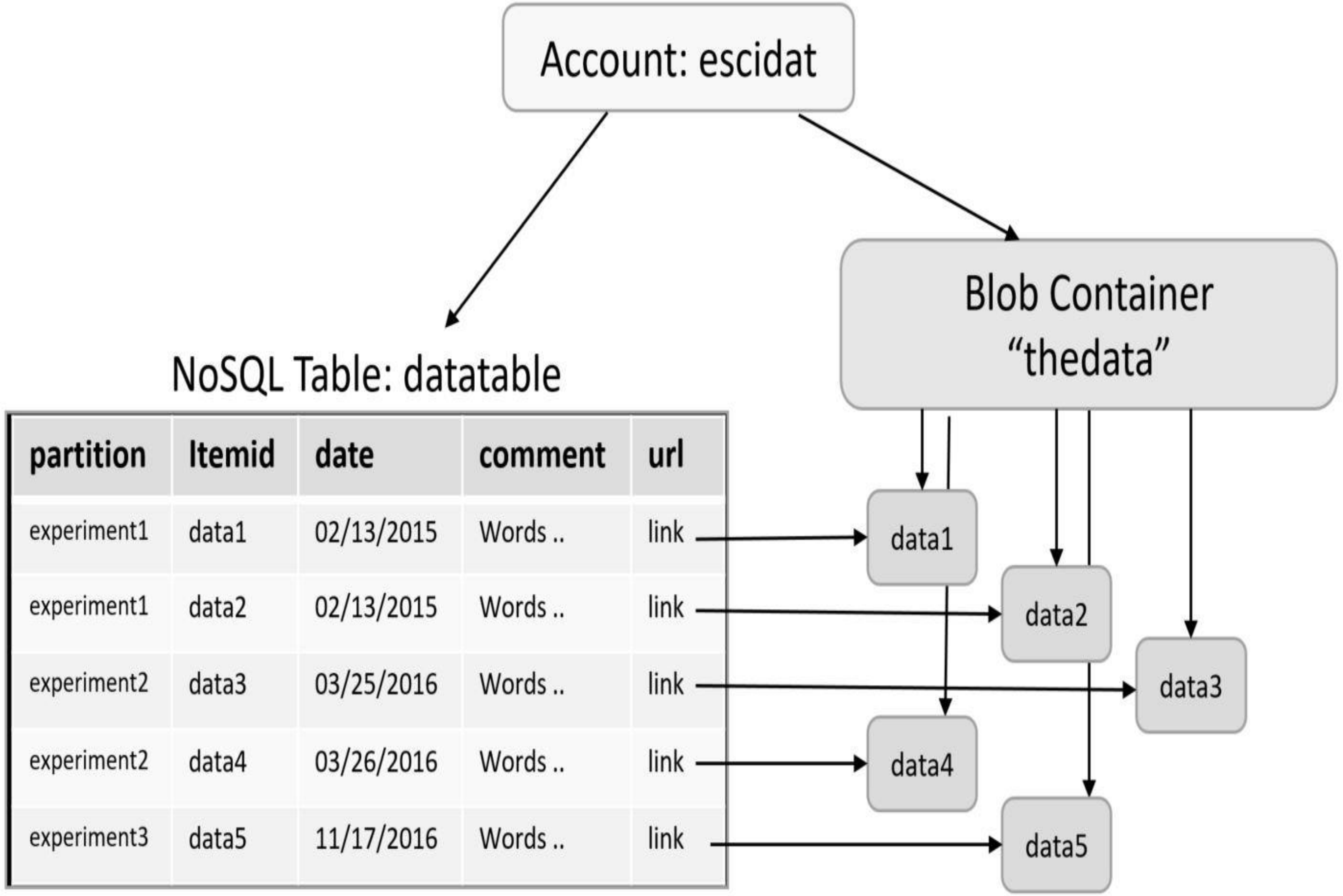
Carl	Role = CEO	Age =93

Key-value NoSQL examples *similar* to

Amazon AWS DynamoDB
Azure Tables
Google BigTable and Cloud DataStore

A simple example

- A table and data blobs
- Suppose you have a set of experiment data containing
  - An experiment number
  - A data item number
  - A date
  - Some comment data
  - A very large binary object
- Build a table of the experiments with a url link to the data in blob store.



Account: escidat

Blob Container
“thedata”

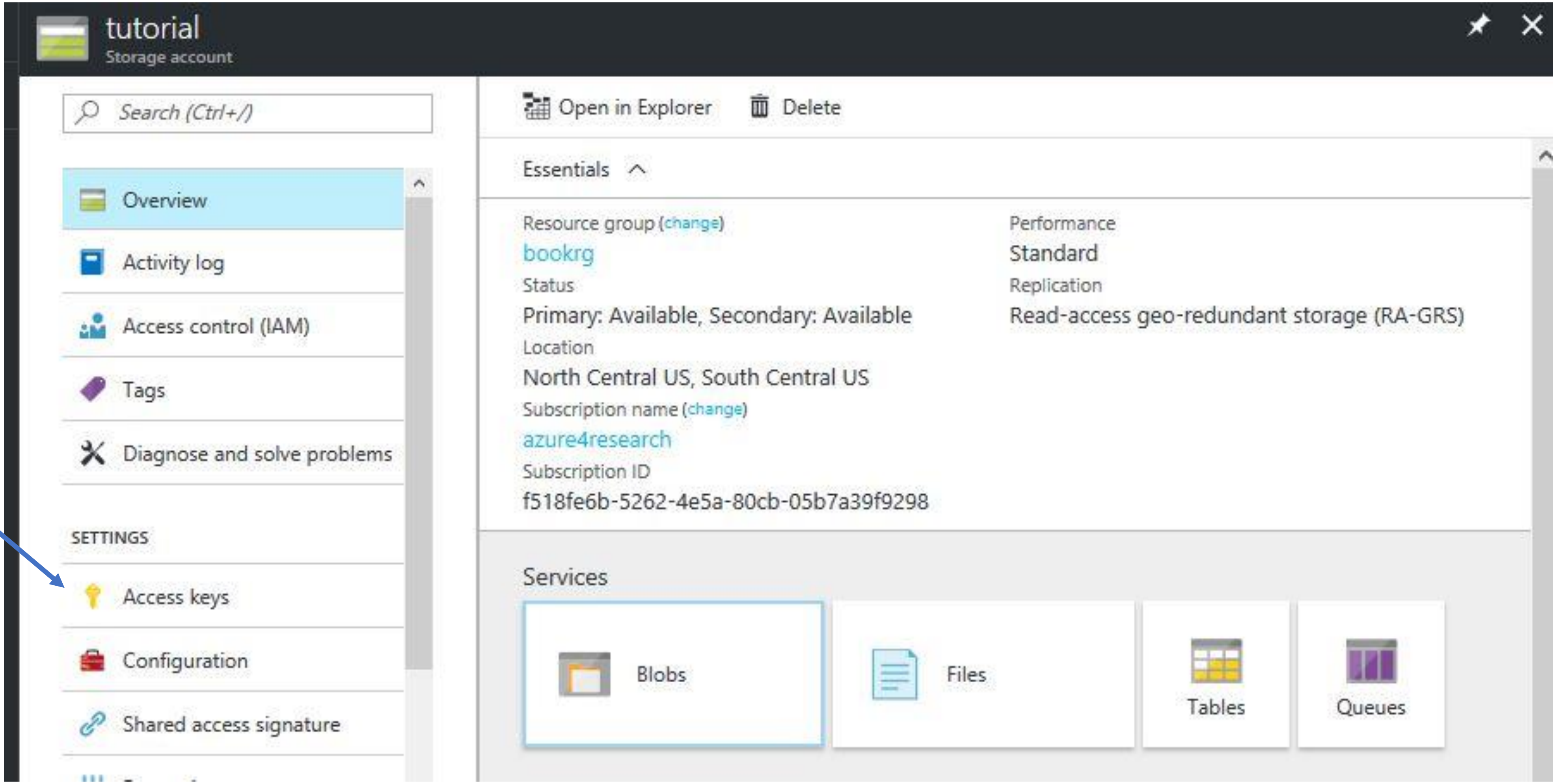
NoSQL Table: datatable

partition	Itemid	date	comment	url
experiment1	data1	02/13/2015	Words ..	link
experiment1	data2	02/13/2015	Words ..	link
experiment2	data3	03/25/2016	Words ..	link
experiment2	data4	03/26/2016	Words ..	link
experiment3	data5	11/17/2016	Words ..	link

Azure Solution

Let’s create a new storage account “tutorial” to hold the blobs and table

To get an access key
Click here



Resource group (change)
bookrg

Status
Primary: Available, Secondary: Available

Location
North Central US, South Central US

Subscription name (change)
azure4research

Subscription ID
f518fe6b-5262-4e5a-80cb-05b7a39f9298

Performance
Standard

Replication
Read-access geo-redundant storage (RA-GRS)

Get the container for the demo

- Make sure you have docker installed on your machine
- Download Docker for your pc or mac
  - https://docs.docker.com/engine/installation/
- Then do

```
docker run -i -t -p 8888:8888 dbgannon/tutorial
```

- This will take a while
- When it is up go to https://localhost:8888
  - You will need to add security exceptions in the browser.   It is safe.
  - Password is “tutorial”
  - Open azure.ipynb in Jupyter
- Or, if using a different jupyter, download https://SciengCloud.github.io/azure.ipynb
- Using the azure portal create a storage account and have the key ready

Azure Solution – now create table and blob container

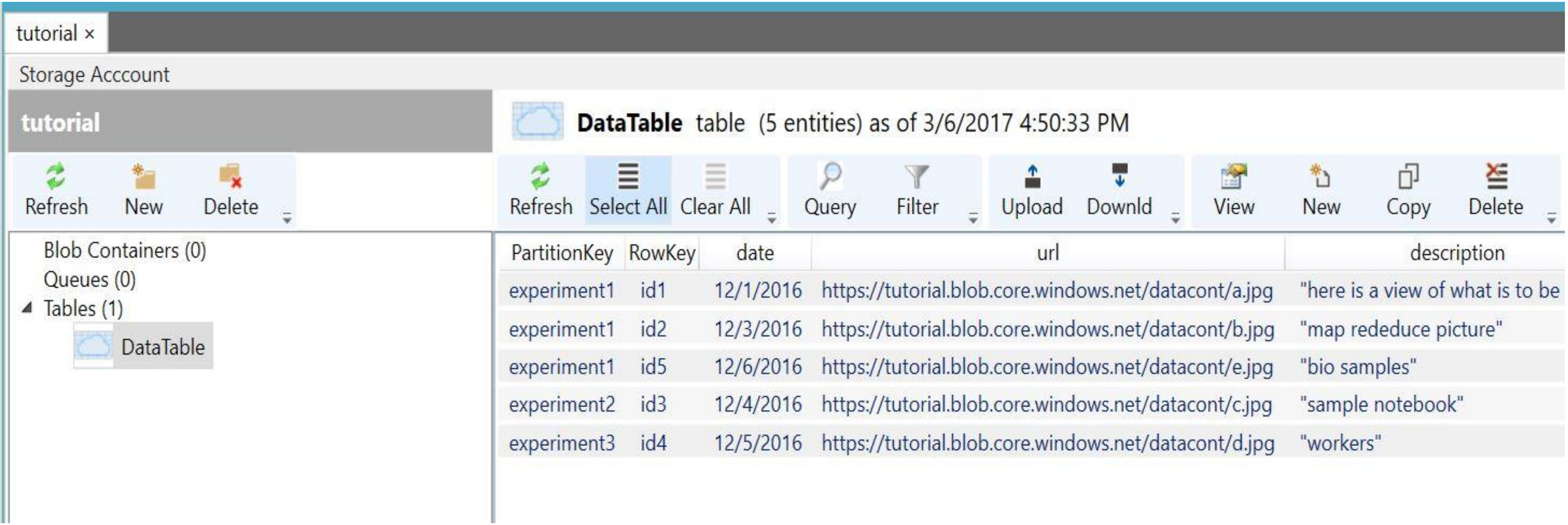
```
import csv
import sys
import azure.storage
from azure.storage.table import TableService, Entity
from azure.storage.blob import BlockBlobService
from azure.storage.blob import PublicAccess

block_blob_service = BlockBlobService(account_name='tutorial',
       account_key=‘biglongaccesskey')

block_blob_service.create_container('datacont',
             public_access=PublicAccess.Container)

table_service = TableService(account_name='tutorial',
       account_key=‘samebiglongkey’)


if table_service.create_table('DataTable'):

    print "table created"

else:

    print "table already there"
```

Azure Solution

- Assume data objects are stored in files in /home/me/ and there is a CSV file “thedata” with rows
  - Experiment name, item id, date, filename, comment string

```
with open('/datadir/experiments.csv', 'rb') as csvfile:
    csvf = csv.reader(csvfile, delimiter=',', quotechar='|')
    for item in csvf:
             print item
             block_blob_service.create_blob_from_path(
                     'datacont', item[3],
                    "/datadir/"+item[3]
              )
        url = "https://"+account+".blob.core.windows.net/datacont/"+item[3]
        metadata_item = {'PartitionKey': item[0], 'RowKey': item[1],
                           ‘description' : item[4], 'date' : item[2], 'url':url}
        table_service.insert_entity('DataTable', metadata_item)
```

Use Azure Storage Explorer to inspect the table



DataTable table (5 entities) as of 3/6/2017 4:50:33 PM

PartitionKey	RowKey	date	url	description
experiment1	id1	12/1/2016	https://tutorial.blob.core.windows.net/datacont/a.jpg	"here is a view of what is to be
experiment1	id2	12/3/2016	https://tutorial.blob.core.windows.net/datacont/b.jpg	"map rededuce picture"
experiment1	id5	12/6/2016	https://tutorial.blob.core.windows.net/datacont/e.jpg	"bio samples"
experiment2	id3	12/4/2016	https://tutorial.blob.core.windows.net/datacont/c.jpg	"sample notebook"
experiment3	id4	12/5/2016	https://tutorial.blob.core.windows.net/datacont/d.jpg	"workers"

Next VMs and Containers