

Building Services using Azure Batch

Marc van Houten

Cloud Solution Architect



Agenda

1. Brief overview of Azure Batch
2. Demo + code
3. Things to keep in mind

Overview Azure Batch

Big Compute Evolution

On-premises Clusters

- Easy scaling to reduce runtimes
- Job scheduling and management
- Compute node provisioning
- *Hardware is paid for*



Hybrid Clusters

- Extend cluster to cloud to handle peak demand
- **HPC Pack Burst to Azure**
- Manage like machines on your network with VPN or Express Route



IaaS Clusters

- Deploy cluster all in cloud
- Move existing applications
- Support projects & testing
- Gallery images and **templates** to deploy
- Flexible VM configuration



Cloud Native

- Azure Batch: Native cloud scheduler
- DevOps, not infrastructure management
- Small to very large deployments
- Elasticity with auto-scale
- Use within a service or to offer SaaS

Two Types of Applications

Embarrassingly parallel:

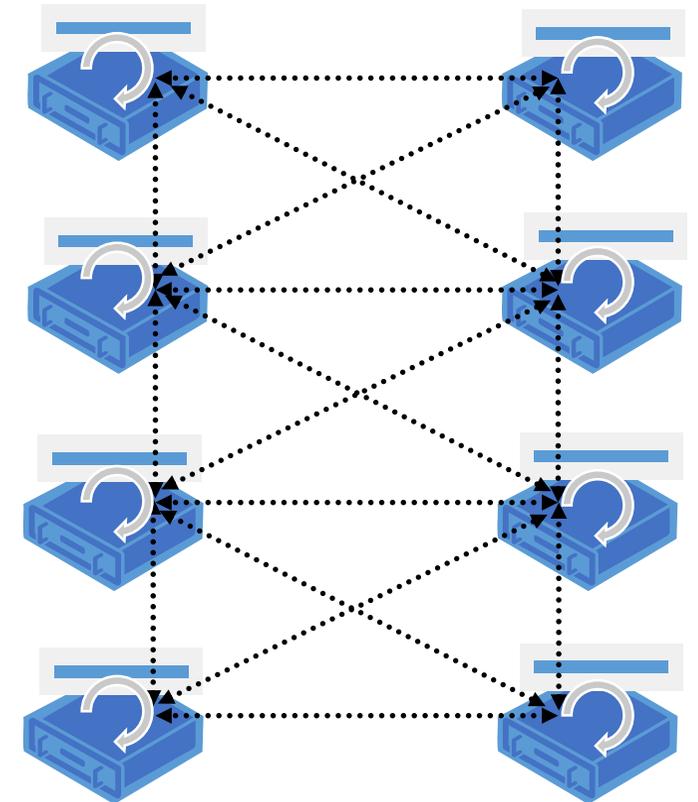
- VMs don't need to talk to each other, or very little cross-node communication
- Usually a parameter sweep, a job splitting, or a search/comparison through data
- Examples: Monte Carlo simulations, image/video rendering, genetic algorithms, sequence matching, file processing

→ Great workload for the cloud, look for them!

Tightly coupled:

- Nodes need to talk to each other constantly
- Requires a fast interconnection network (low latency and high throughput)
- Examples: automotive crash simulation, fluid dynamics, climate modeling, reservoir simulation, manufacturing design

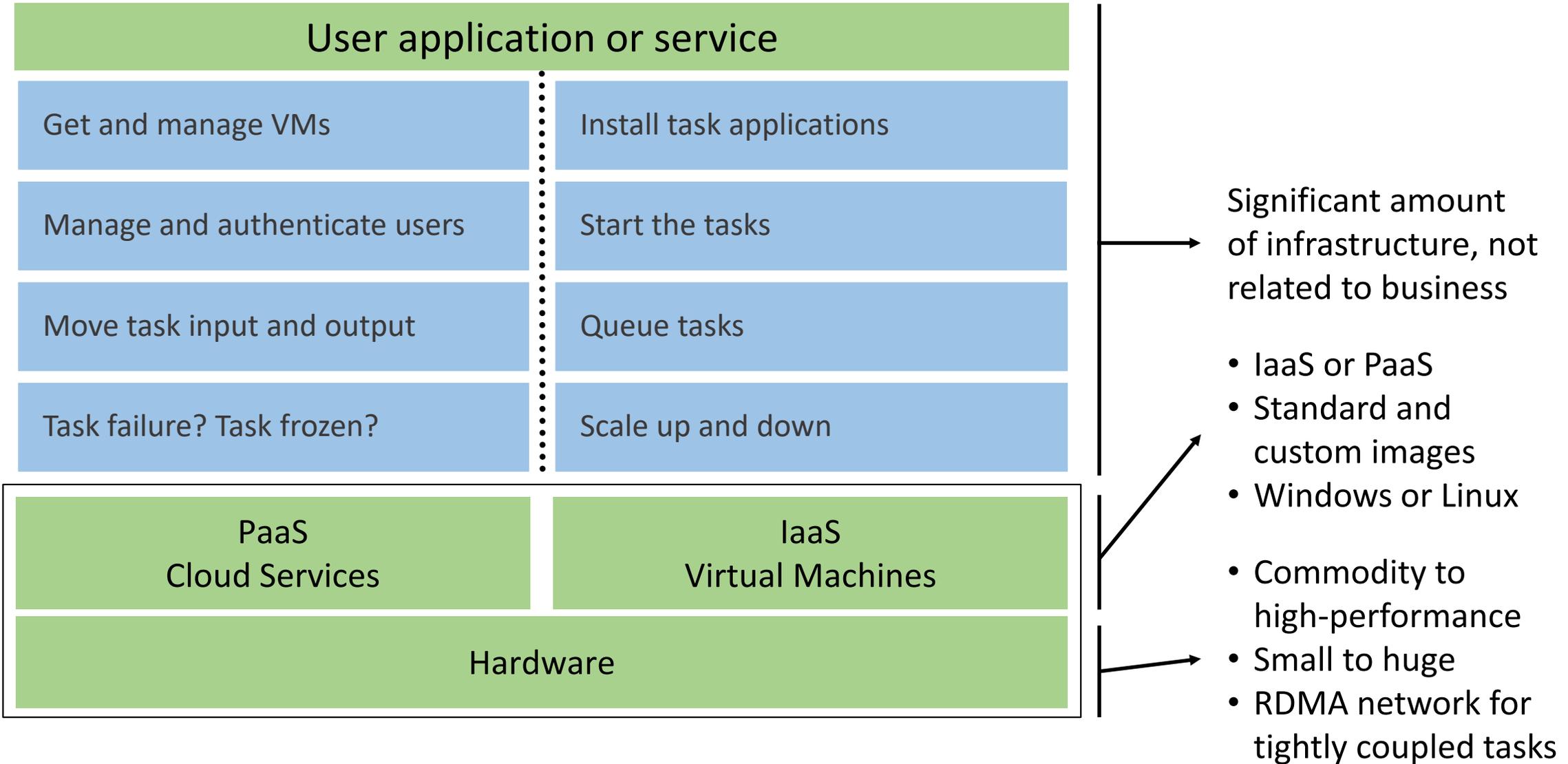
→ Now possible in cloud with HPC VMs with RDMA on Azure



What is Batch being used for?

- **Media transcoding**
 - Media Services - audio and video transcoding, etc.
 - TV Everywhere – video transcoding using ffmpeg
 - Xbox video – video pre-processing
- **Rendering**
 - HoloLens team renders test environments using Blender
 - Engineering companies rendering pipeline using 3DSMax and V-Ray on Linux VMs
- **Test execution**
 - Azure Engineering CloudValidate service
 - Intune runs 20K tests in 20 mins following check-in's
 - Financial risk analysis testing
- **Risk analysis**
 - Towers Watson RAFM
 - Other insurance ISV
 - Banking & financial services company
- **Data ingestion and processing**
 - Azure StorSimple team ingest and pre-process to prepare for HDInsight jobs
 - Nightly data ingestion and processing of automotive data
 - ADF data copy activities
- **Pricing models**
 - Energy pricing using python Monte Carlo quant models
 - R and Spark models for hotels
- **Engineering simulations**
 - Engineering ISV customers can run apps at scale
- **Image processing**
 - Analyze MRI scans for sign of dementia
- **Misc**
 - Tracking drug company spending against doctor prescriptions
 - Aircraft route optimization and simulation

Task Execution Pattern on Azure



Task Execution Pattern on Azure

User application or service

Azure Batch

App lifecycle, job dependencies, data movement, client plugins

VM management, job queue, and task dispatch

PaaS
Cloud Services

IaaS
Virtual Machines

Hardware

Cloud-enable applications,
made available as SaaS:

- Higher-level set of capabilities
- Minimizes required Azure knowledge
- Foundational batch processing service
- Don't worry about the "plumbing"

Scenarios

- I'm a developer producing an Azure-based application that needs to perform a lot of compute-intensive work
 - Engineering stress analysis, fluid dynamics
 - Monte Carlo simulations performing risk calculations
 - Transcoding lots of video or audio
 - Image processing; e.g. create multiple resolutions, extract meta-data
 - Image analysis

- I have an application or script that I need to scale massively but I haven't got access to the code or I don't want to change the code
 - Custom script
 - 3rd party executable

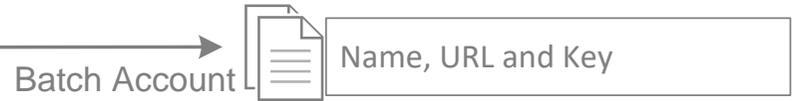
I want to focus on my business and not invest in building & managing my own batch processing infrastructure

Azure Batch Concept



Client or application

1 Create Batch Account



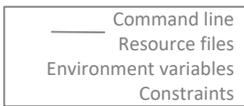
2 Create Storage Account



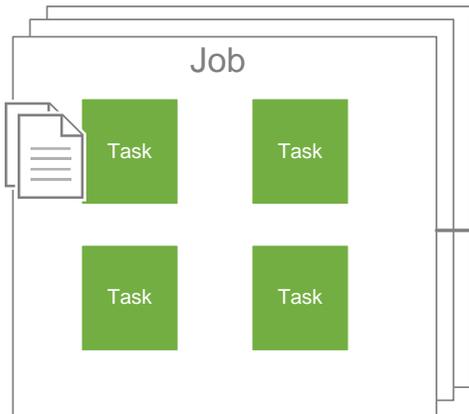
3 Upload data

4 Upload Program

5 Specify Pool



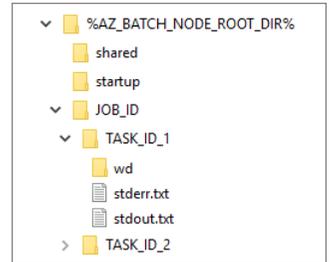
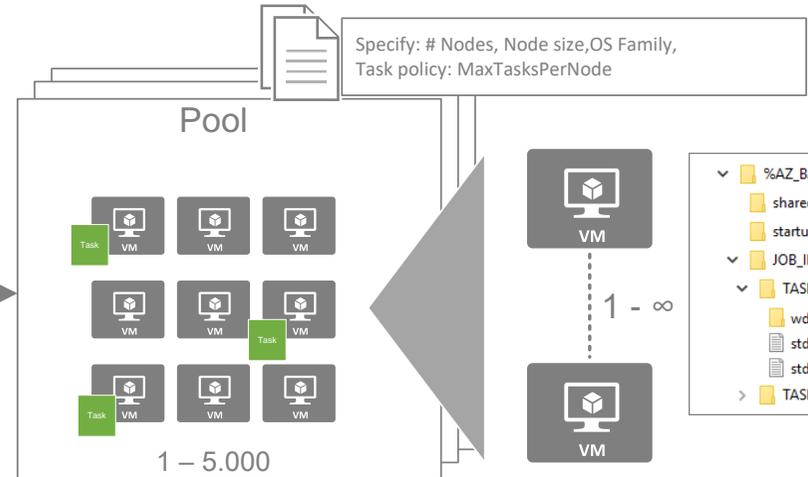
6 Create job



7 Add tasks to job

8 Submit job

9 Download output

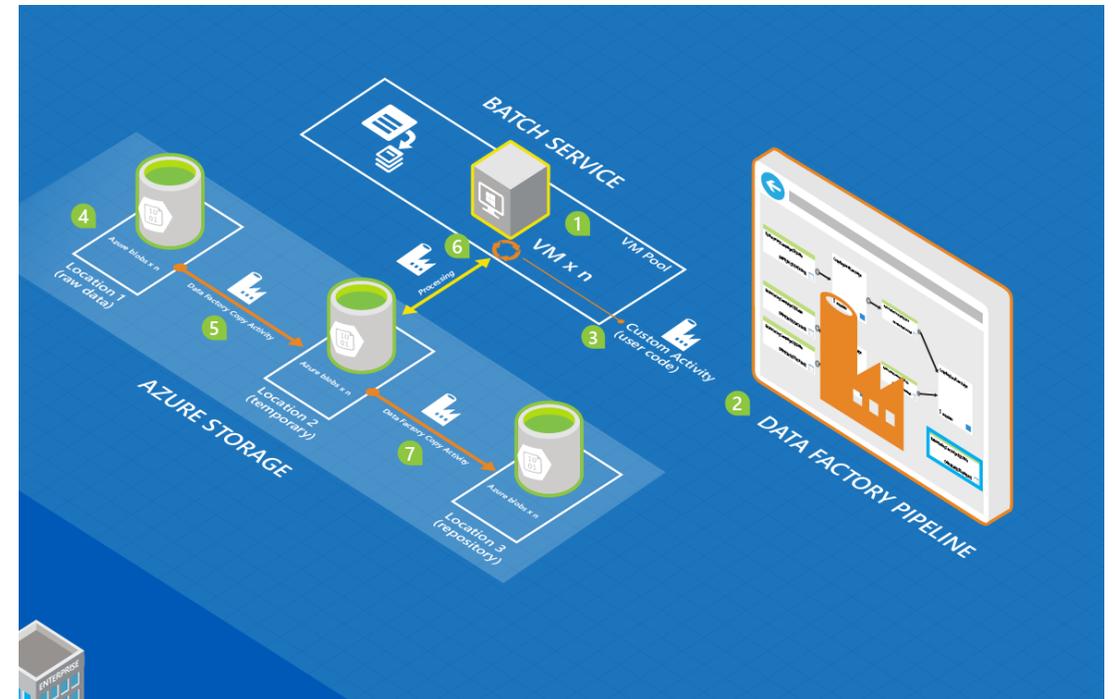


Job Manager Task	used in controlling and/or monitoring job execution and is scheduled to execute before the other tasks in a job
Prep and Release Task	The job preparation task runs on all compute nodes scheduled to run tasks, before any of the other job tasks are executed. the job release task runs on each node in the pool that executed at least one task

Start Task	Is executed each time a compute node joins the pool, and when a node is restarted
Multi-Instance Task	is configured to run on more than one compute node simultaneously

Additional scenario's

- Integrate Azure Batch in your Data Factory Pipeline
- Use WebJobs or Azure functions to kick-off Azure Batch jobs
- Create plug-ins into existing applications



Azure Batch support

API	API reference	Download	Code samples
Batch REST	MSDN	N/A	MSDN
Batch .NET	MSDN	NuGet	GitHub
Batch Python	readthedocs.io	PyPI	GitHub
Batch Node.js	github.io	npm	-
Batch Java (preview)	github.io	Maven	GitHub

Windows or Linux VM

Only Pay for Compute hours (minutes)

DEMO

Demo – massive image processing

- Image processing is done with ImageMagick.exe
- For the demo we process 450 (medium size) JPG images
- Job will resize (10%), grayscale and change the format (GIF) of the images

1. Create Batch account
2. Optional: upload images that needs to be processed.
3. Create and upload Application Package
4. Define Job Preparation Task to pre-configure the nodes
5. Create a Pool
6. Create a Job
7. Define the tasks
8. Run

Things to keep in mind

Things to keep in mind

- Custom VM not supported yet; So use a Job Preparation task to pre-configure the nodes
- Quota limits for each resource (see documentation)
- Understand Azure storage limits (60 MB/sec per blob)
 - Dash Server (<https://github.com/MicrosoftDX/Dash>) to scale out
- Large-scale data movement between storage and 100's or 1000's VM's
- Handling large files (block blobs)
- Network gateway limit of 2GB
- What regions are the services or sizes in (moving towards mega hubs)
- API and .NET libraries changes frequently

New Azure Batch capabilities

Monday, September 5, 2016

- In the last few weeks, we have added some important new capabilities to the Azure Batch service:
- Batch pool support for **Linux VMs and VM scale sets in general availability:**
The support for Linux and Windows virtual machines that was made available as preview in April is now generally available. Pool allocation performance and robustness for large numbers of VMs were improved during the preview period.
- **Linux MPI:**
In December 2015, support for MPI applications was added for Windows VMs. We have now made MPI support available for Linux VMs. For more information, see the [blog post](#).
- **Task application packages:**
In addition to the ability to specify a list of application packages for a pool, it's now possible to specify an application package list for tasks. For more information, see the [MSDN documentation](#) and the [Azure article](#).
- **Virtual network support for Windows VMs:**
All VMs in a pool can now be part of an Azure virtual network. More information is available in [MSDN documentation](#).
- Job auto-completion:
A property can now be set on a job that enables the job to be automatically moved to the completed state when all tasks associated with the job have finished. Prior to this feature, all tasks had to be monitored and the job explicitly terminated to mark it as completed. For more information, see the [documentation for adding a job](#).
- Persisted job and task output to Azure Storage:
Store job and task output files and log files to the linked Azure storage account by using a new .NET class library. The persisted files can be listed and viewed in the Azure portal. For more information, see the [Azure article](#).
- Azure portal:
We continue to update the Batch capabilities in the Azure portal on a weekly basis, with improved functionality and monitoring capabilities

