Orleans + YAMS

Deployment of distributed applications

Jakub Konecki

jakub.konecki@applicita.com
jkonecki@gmail.com
@jakubkonecki

applicita

Your Business. Our Technology. Real Potential.

Technology stack

- Orleans
- Orleankka
  https://github.com/OrleansContrib/Orleankka
- Event sourcing - Streamstone
  https://github.com/yevhen/Streamstone
- CQRS - DocDB
- WebAPI
- Angular SPA

Development + Deployment

- VSTS
  - code
  - work
  - build services
- Octopus Deploy
- YAMS
- Everything is in Azure
  - 4 environments
  - Resource Manager template to provision environment

Monitoring *

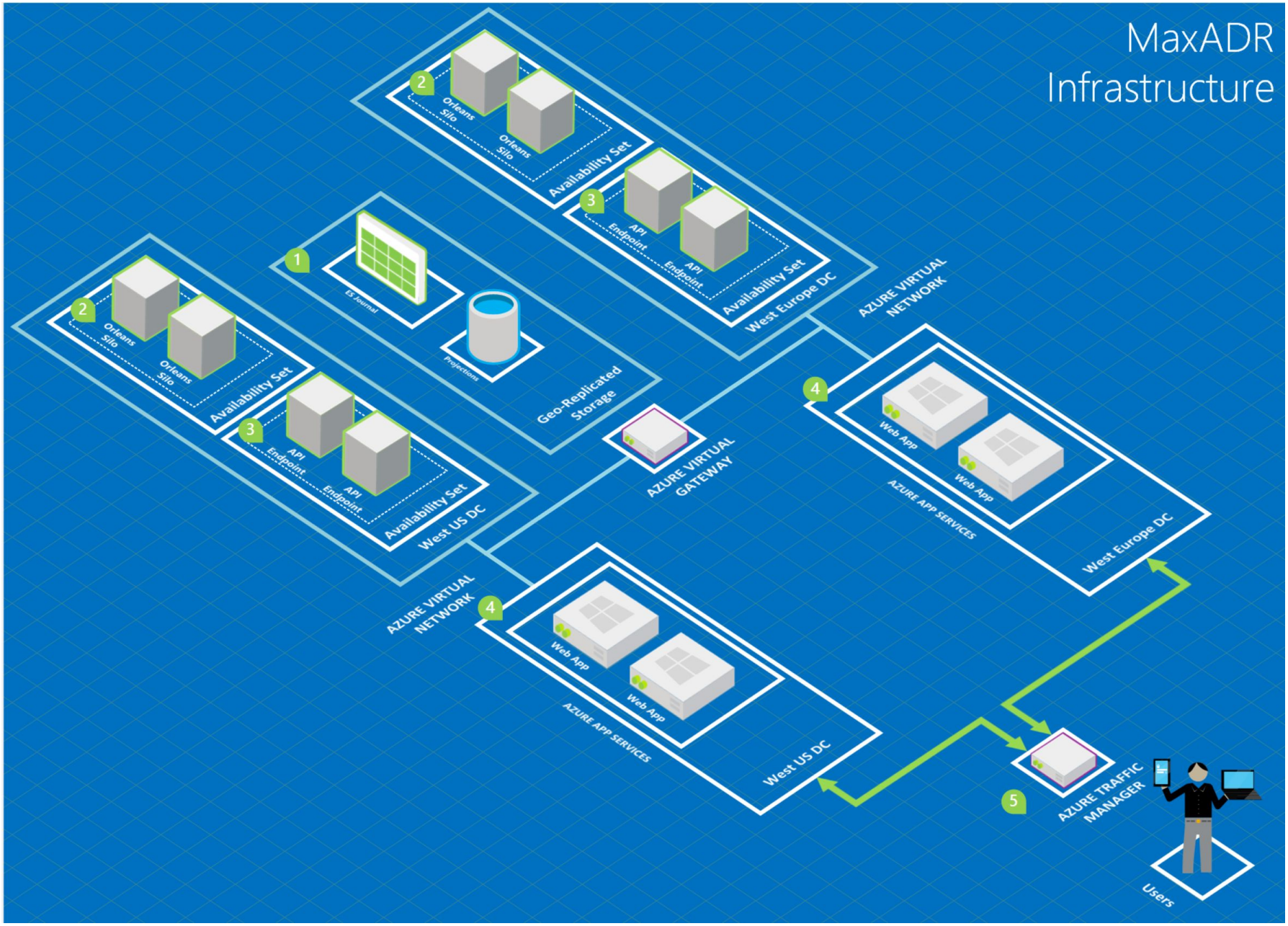
- Serilog → Seq
  - Streamstone read/write
  - DocDB read/write
  - Orleans message processing
- AppInsights
- Azure Alerts

For more awesome monitoring / telemetry watch:

Meetup 11: A monitoring and visualisation show with Richard Astbury, Dan Vanderboom and Roger Creyke

https://youtu.be/WiAX_eGEuyo

* Ashkan asked about monitoring on gitter


MaxADR
Infrastructure
2
Orleans Silo
Orleans Silo
Availability Set
3
API Endpoint
API Endpoint
Availability Set
West Europe DC
1
ES Journal
Projections
Geo-Replicated Storage
2
Orleans Silo
Orleans Silo
Availability Set
3
API Endpoint
API Endpoint
Availability Set
West US DC
AZURE VIRTUAL NETWORK
AZURE VIRTUAL GATEWAY
4
Web App
Web App
AZURE APP SERVICES
West Europe DC
AZURE VIRTUAL NETWORK
4
Web App
Web App
AZURE APP SERVICES
West US DC
5
AZURE TRAFFIC MANAGER
Users

So, what is this YAMS thing, anyway?

I'm not going to tell you, watch:

Meetup 9: Nehme Bilal and Reuben Bond talk about deploying Orleans with YAMS and Service Fabric

https://youtu.be/w__D7gnqeZ0

YAMS starts at 25:30
https://youtu.be/w__D7gnqeZ0?t=1530

YAMS Host

Just a cloud service (one time deployment) - is it worth a NuGet package?

```
var yamsConfig = new YamsConfigBuilder(
    // mandatory configs
    DeploymentIdUtils.CloudServiceDeploymentId,
    RoleEnvironment.CurrentRoleInstance.UpdateDomain.ToString(),
    RoleEnvironment.CurrentRoleInstance.Id,
    config.CurrentRoleInstanceLocalStoreDirectory)
    // optional configs
    .SetCheckForUpdatesPeriodInSeconds(config.UpdateFrequencyInSeconds)
    .SetApplicationRestartCount(config.ApplicationRestartCount)
    .Build();

_yamsService = YamsServiceFactory.Create(yamsConfig,
    deploymentRepositoryStorageConnectionString: config.StorageDataConnectionString,
    updateSessionStorageConnectionString: config.StorageDataConnectionString);

await _yamsService.Start();
```

https://github.com/Microsoft/Yams/blob/master/Docs/Deploy_YAMS.md

** Add your own logging / error handling*

Packaging YAMS applications

Orleans + 4 Web APIs hosted on YAMS

Web App deployed to App Service

Merge to *develop* or *master* branch:

1. Boring compilation bit
2. VSTS build process generates NuGet package for each application
3. VSTS pushes packages to Octopus Deploy NuGet feed
4. VSTS generates a *release* in Octopus Deploy
   - we're using semver (*GitVersion*)
   - *Stable* and *Unstable* channels in Octopus

Octopus Deployment Process

YAMS Uploader

https://github.com/Applicita/YamsUploader

Uploads binaries to Storage, updated YAMS *DeploymentConfig.json*

Thanks, Alex Prooks!

YAMS Uploader Octopus Step Template

https://library.octopusdeploy.com/step-templates/a1d95c5f-42fb-43b3-8bee-74a255f2ae71/actiontemplate-yams-uploader

Easy configuration for YAMS Uploader in Octopus - we've tamed PowerShell gods for you!

Fun fact!

Number of YAMS uploader projects per hemisphere: 1

Check out Reubens YAMS Deployer:
https://github.com/ReubenBond/YamsDeploy

Blue / Green environments

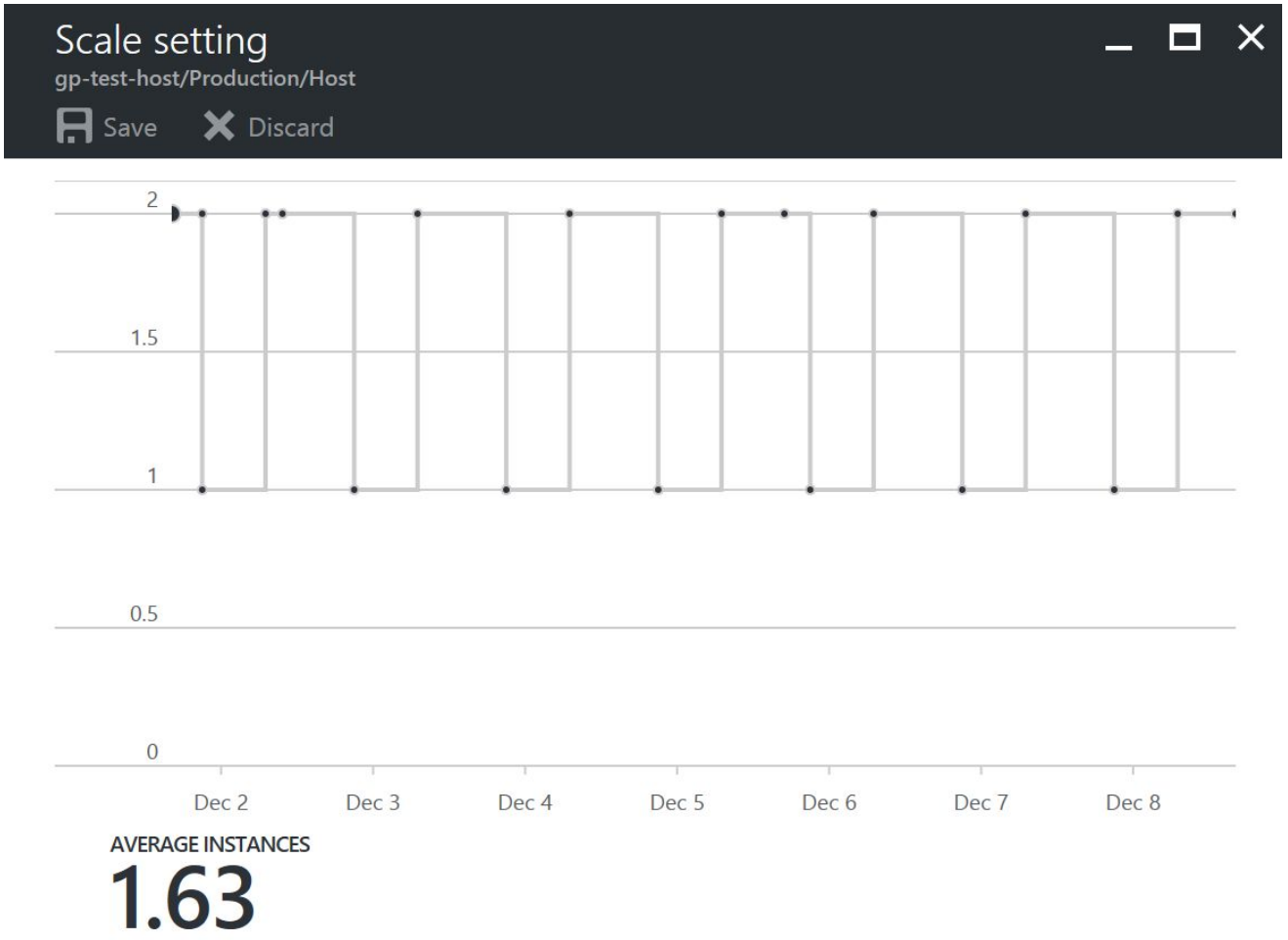
- 2 separate YAMS clusters
- 2 separate Web Apps
- Traffic Manager profile for Web Apps and each API
- 1 Event Store / DocumentDB
- 2 Orleans clusters - but only one running at a time!

Manual switching between environments - when one DC goes offline.

The perfect world:

Using Orleans Multi-Cluster support and distributed grains to have both clusters running at the same time. Traffic Manager profiles can be changed to use performance rule to direct requests to the nearest API / Cluster. We're waiting for Orleans to support Event Sourced grains.

Azure Cloud Service scaling - now in new Portal



Thank you! Any questions?