Journey of containerization on AWS

Nico Schilling  |  09.09.2019
EUROPEAN LEADER IN PUBLIC CLOUD

with local presence in 8 countries.

Nordcloud was born in the cloud 2011 and has grown to be the European leader in public cloud infrastructure solutions and cloud native application services.

We are ranked globally #2 by Gartner capability assessment in Mode 2 use cases such as supporting agile applications and cloud native transformation.

Our highly skilled organisation comprises of

300+ cloud experts
180+ certifications
200+ business and technical accreditations.
OUR EXPERTISE IN MULTI-CLOUD HAS MADE US

Strategic partner for AWS, Google and Microsoft.

Partners with the leading technology companies from their respective fields.
OUR SERVICES

BUSINESS OPTIMISATION
CLOUD FOUNDATION

- Build a robust and future-proof foundation for your applications and servers.
- Introducing maximum agility and flexibility for your business, creating the foundation for business innovation.

BUSINESS ACCELERATION
CLOUD-NATIVE APPLICATIONS

- Boost your business and stay ahead of the competition with cloud native applications.
- Benefit from scalability, as well as easier and more flexible management.

BUSINESS OPTIMISATION
IT TRANSFORMATION

- Improve your processes and operations by moving IT to the public cloud.
- Cut time to market to just minutes (compared to legacy months).

BUSINESS ACCELERATION
DATA-DRIVEN BUSINESS

- Benefit from new business insights through machine learning and artificial intelligence.
- Improve decision making, adapt to changes quickly and open up new business opportunities.
Where we started from?

- A legacy java application in customer branch offices
- Complex network
- A team with nearly no knowledge about Cloud and Container
- No specification / no guidelines about the architecture
Where we started from?

- Application can't be changed
- Thousands of branches
- Master data on local server
What do we want to achieve?

• One API endpoint for all branches
• Find a „good“ solutions in terms of architecture, price, operation
• Ramp up the cloud / container knowledge inside the team
• Iterate in an agile way
First shot
## First shot

### Achievements

- Infrastructure as Code (CloudFormation)
- Network separation
- On-premise connection
- Handling of the application
- Provide a working environment

### Trade off

- No scaling at all
- Manual installation of application
- Management overhead for EC2
- Only for one branch office
- Suboptimal runtime environment
- Expensive
What could help us to eliminate a lot of trade-offs in one step?

Dealing with the trade-offs
Let’s put it in a container!

Putting the application in a container, helped us with the following:

- A container can be scaled easier
- The Dockerfile describes the installation of the container
- No management overhead for the operating system
- Very small management for Docker orchestration by using AWS ECS Fargate
- Runtime environment way more suitable
Second shot with containers!
Second shot!

What the team achieved?

- Infrastructure as Code (CloudFormation)
- Network separation
- On-premise connection
- The handling of the application
- Provide a working environment
- Installation is automated via Dockerfile
- Scaling is easier
- Eliminated OS management

Trade-offs?

- Docker deployment by hand
- Docker container is very hardcoded
All good things come in threes!

What we did in this stage? CI/CD for Docker:

- AWS Cloud
- Public Subnet
- AWS Fargate
- AWS CodeCommit
- AWS CodePipeline
- AWS CodeBuild
- Infrastructure Developer
All good things come in threes!

What the team achieved?

• Infrastructure as Code (CloudFormation)
• Network separation
• On-premise connection
• The handling of the application
• Provide a working environment

• Installation is automated via Dockerfile
• Scaling is easier
• Eliminated OS management

• Docker deployment by hand

Trade-offs?

• Docker container is very hardcoded
What could the future bring?

In the future two huge points needs to be addressed:

• Hardcoded Docker Image
  • Split Image into a base and branch image
  • Eliminate multiple processes in one Container

• Support the operational processes more
  • Stream logs to Cloudwatch logs
  • Build CloudWatch Dashboards to support operation with more insides
A journey of containerization on AWS

Three stages until now

1. **MVP**
   - Provide something that runs and can be used by the development team

2. **Containerization**
   - Eliminate the most critical trade-offs from stage 1

3. **Automation**
   - Let’s make it more comfortable for ourselves
Things for the next project:

• Trade-offs are okay – as long as you have a clear pipeline to fix them

• Be clear about your trade offs

• One trade off should never exist: Security

• Not starting with the whole Cloud World at ones, makes it a lot easier for newbies

• Agility helps a lot finding a suitable solution

• External consulting helps with guidance, upskilling and direction
THANK YOU!

Let's stay in touch
LinkedIn: https://www.linkedin.com/in/nicoschilling/
Email: nico.schilling@nordcloud.com

Image credits:
- Photo by Danielle MacInnes on Unsplash
- Photo by bruce mars on Unsplash
- Photo by frank mckenna on Unsplash
- Photo by Octavian Rosca on Unsplash
- Photo by Joanna Kosinska on Unsplash