

David Barbarin



From development to production



# TUGA IT

## SUMMER EDITION

LISBON, JULY 19-21, 2018

# THANK YOU TO OUR SPONSORS



Microsoft

GOLD SPONSOR

**bi4all**

CREATING BUSINESS INTELLIGENCE

SILVER SPONSOR

|create|**it**|

INNOVATING **LIFE**

TUGA BEER SPONSOR

**FARFETCH**

SWAG SPONSOR



# > whoami

## David Barbarin

### Senior Consultant

Architecture, Performance and high-availability

 [@mikedavem](#)

 [David Barbarin](#)

 [Blog](#)



**Microsoft**  
**CERTIFIED**  
*Master*

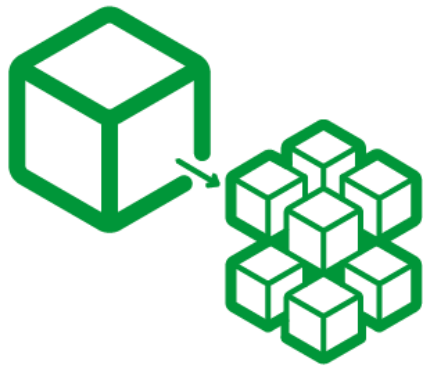


# Agenda

- Docker / DevOps / Microservices and DBAs
- Docker, Software lifecycle management and CI/CD
- Development and production scenario feedbacks
- Q/A



# Docker / DevOps / Microservices and DBAs



# DevOps / Microservices

## Microservices

- > Application paradigm from leaders like Amazon, Netflix, Facebook, Google, eBay
  - > Monolithic vs collaborating services oriented architecture (SOA)
  - > Development engineering in parallel
  - > Embrace emerging technologies and patterns
  - > More flexible / scalable / robust applications

## DevOps

- > To provide a better quality of services to business (Dev + Ops)
- > To accelerate Time to Market (new way to deal with changes)
- > Automation CI/CD and monitoring but not only!

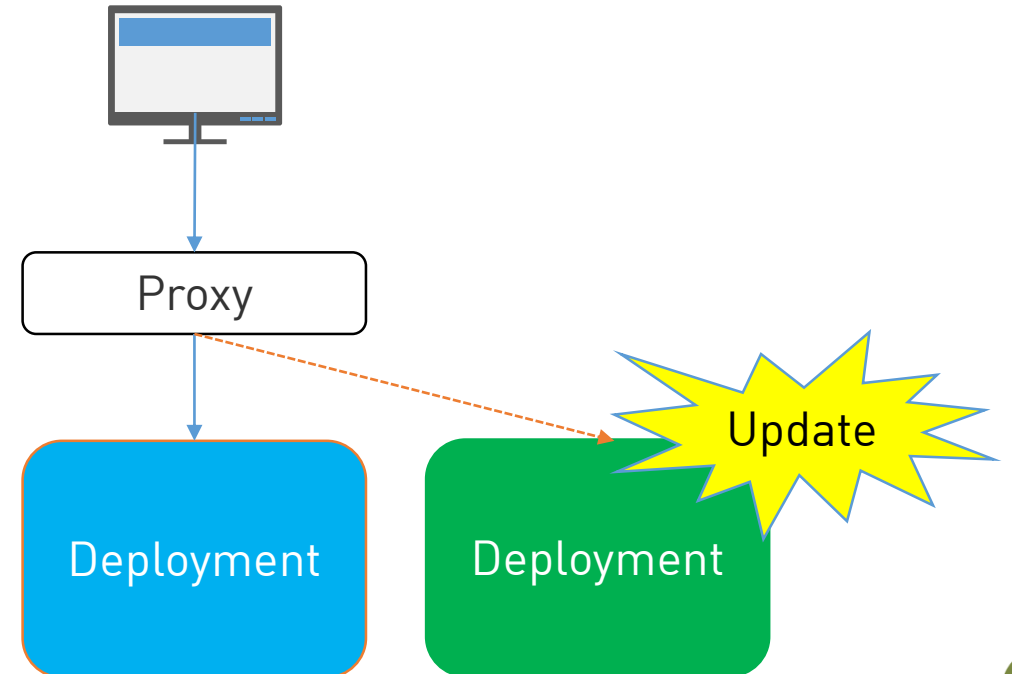
# Customer case



# DevOps / Microservices

## Microservices and challenges (for DBAs)

- > Deployment = Blue/Green, Canary deployments
- > Evolving your SGBDR = (Sharding, schema changes, referential integrity)



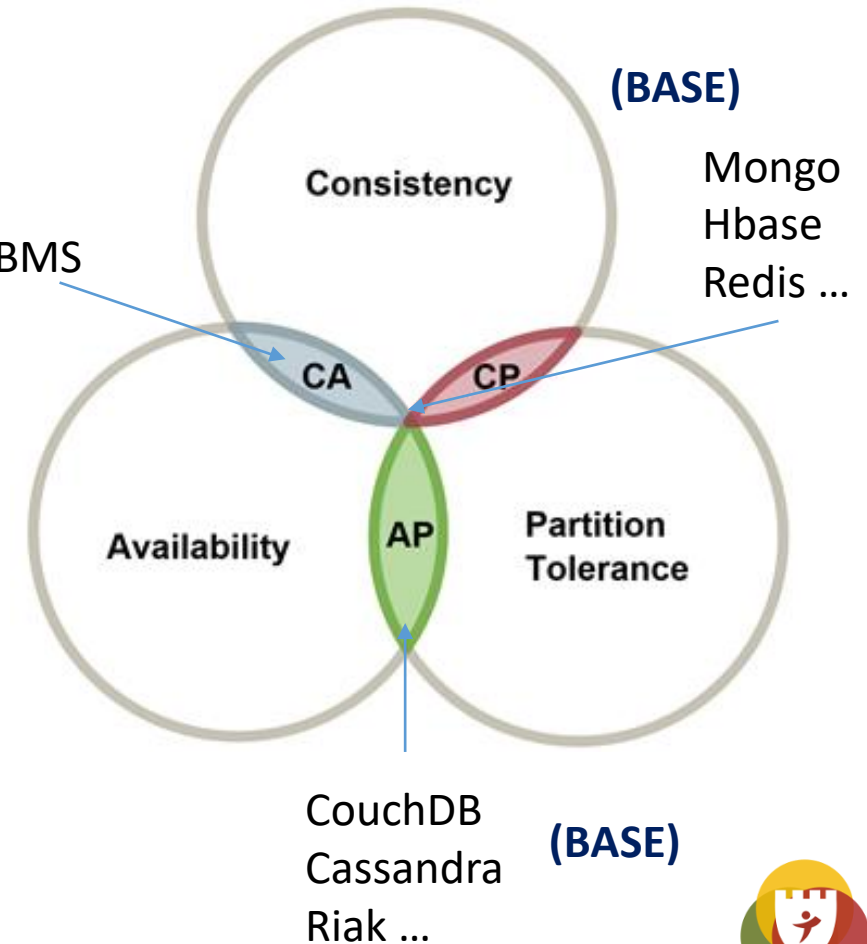


# DevOps / Microservices

## Microservices and challenges (for DBAs)

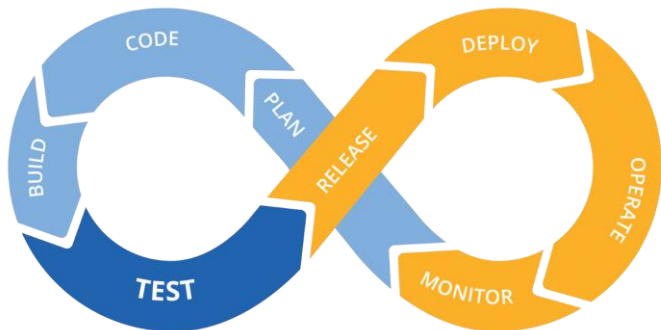
- > Deployment = Blue/Green, Canary deployments
  - > Evolving your SGBDR = (Sharding, schema changes, referential integrity)
- > Decentralized data management
  - > Definition of domain boundaries (Domain Driven Design)
  - > Consistency model = Eventual vs Strong consistency (CAP theorem)

(ACID) RDBMS



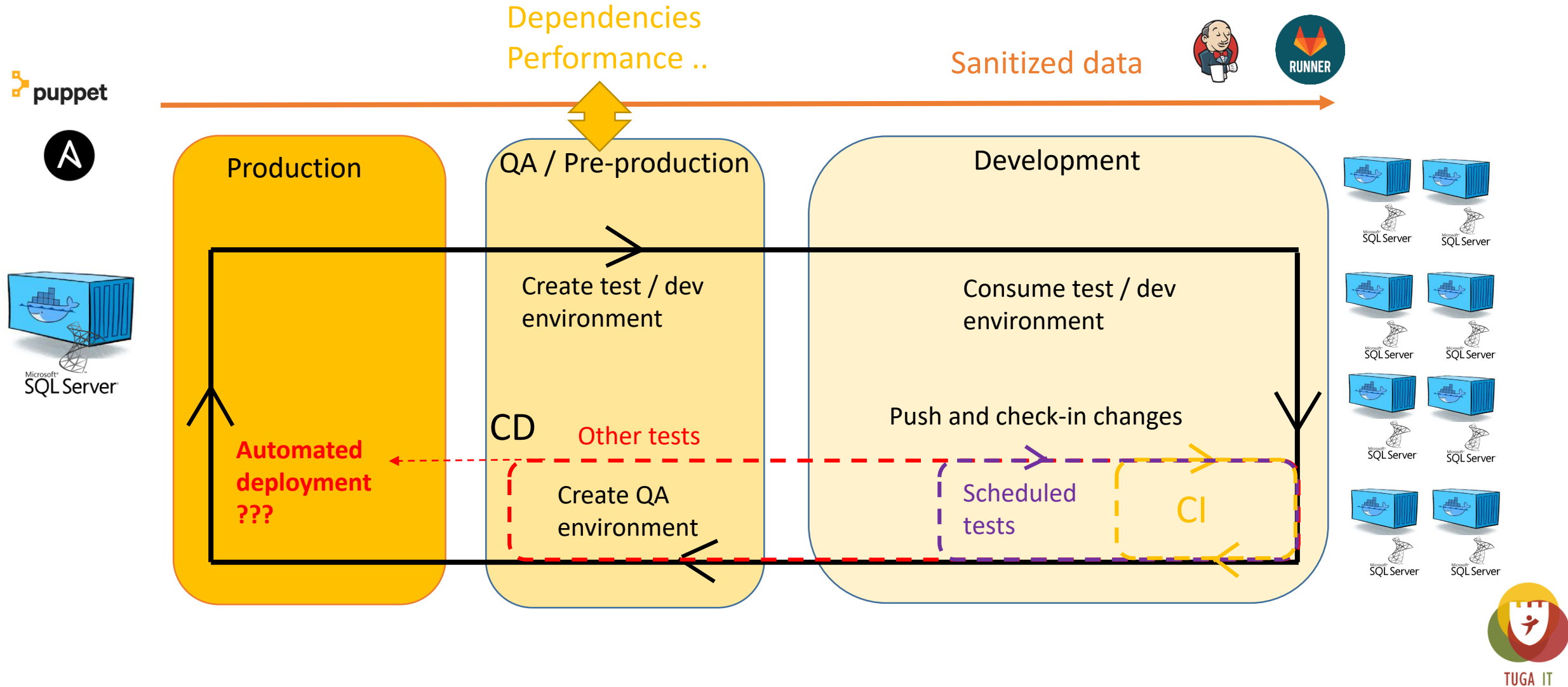


# Docker, Software lifecycle management and CI/CD



# Docker and Software lifecycle management

## CI/CD pipeline

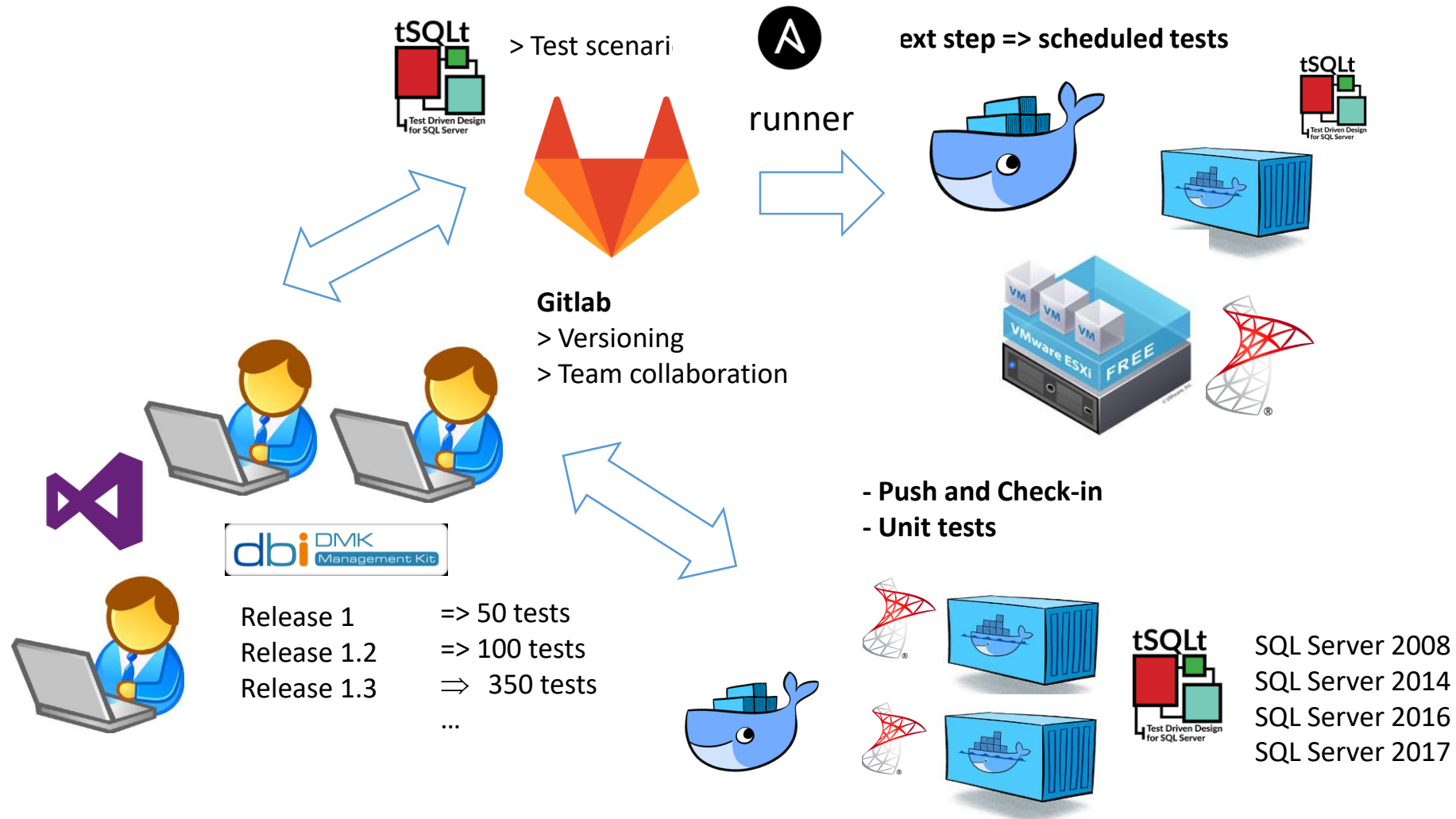




# Development and production feedbacks

# Development and production feedbacks

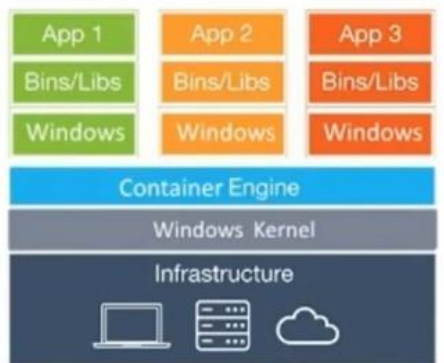
## DMK tool development



# Development and production feedbacks

## Development – Customer CI/CD & self-service environment

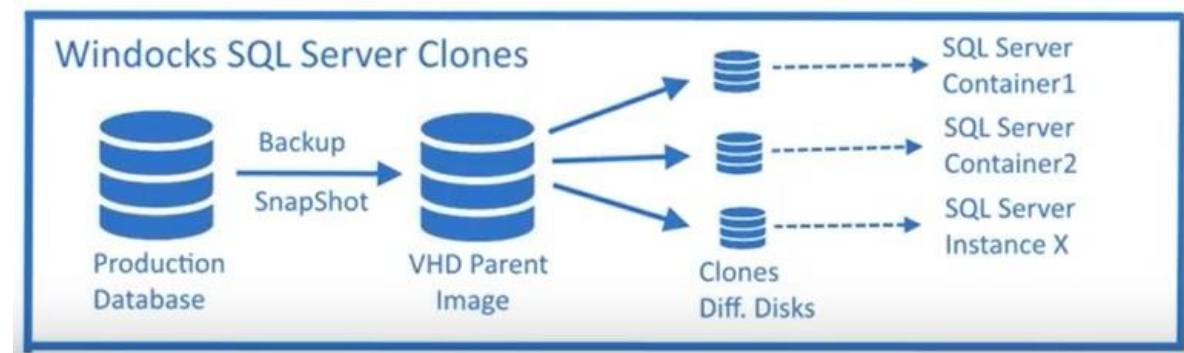
- > What about SQL Server versions prior to SQL2017?
- > I don't want to maintain myself Docker images – time consuming
- > No individual docker engine by developer => internal rules
  - > I want to provide a consolidated self-service environment for developers
- > What about disk space consumption? (Win SQL docker image + DB)
- > What about authentication method? (Windows preferred method)



Microsoft Container Architecture

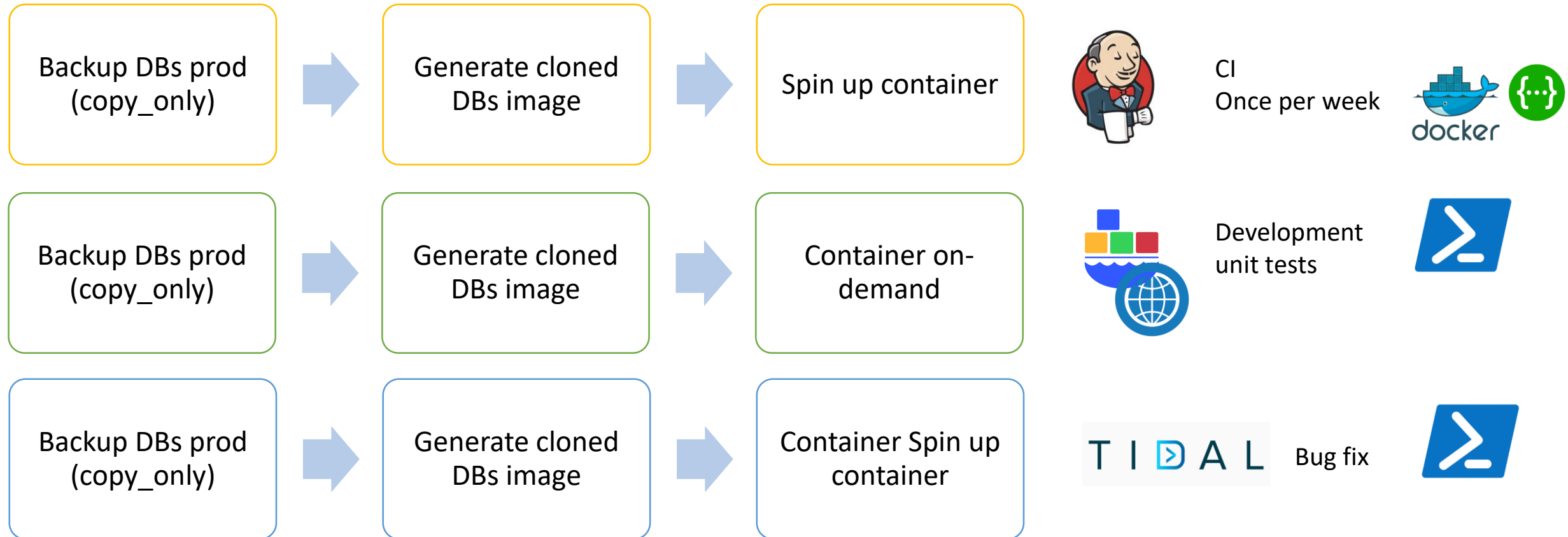


WinDocks Container Architecture



# Development and production feedbacks

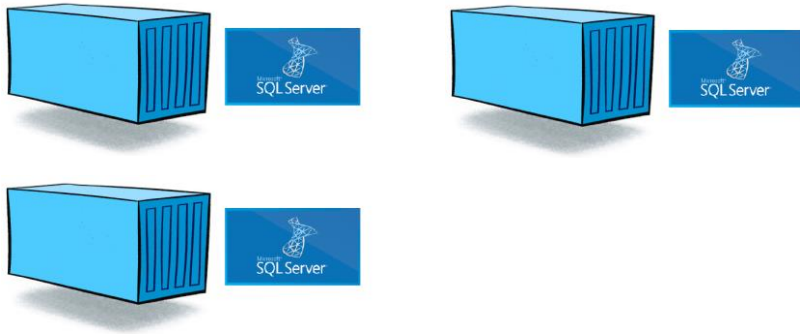
## Development – Customer CI/CD & self-service environment



# Development and production feedbacks

## Production customer case

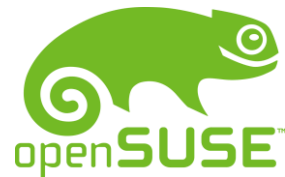
> docker run -d ... microsoft/mssql-server-linux



### Why using a such infrastructure in production?

- > Strong Open Source and Linux culture (Windows OS is not an option)
- > Migration from Oracle to SQL Server
- > Docker is a big part of their main strategy
  - > Recent applications are moving on Docker

Docker Engine standalone



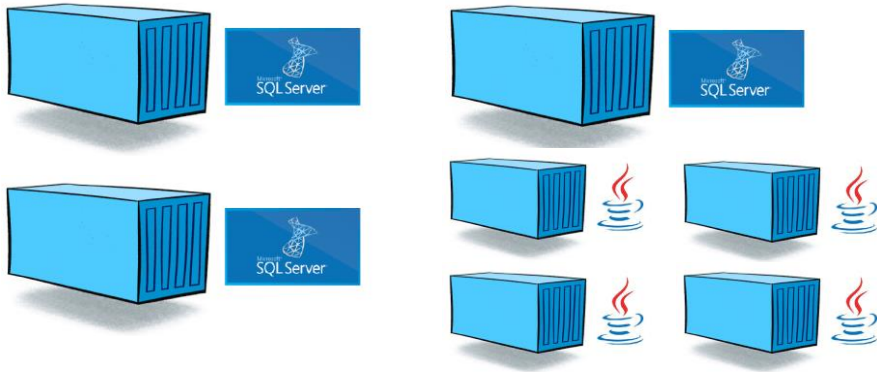
NFS



# Development and production feedbacks

## Production customer case

> docker run -d ... microsoft/mssql-server-linux



Docker Engine standalone



NFS

### Why using a such infrastructure in production?

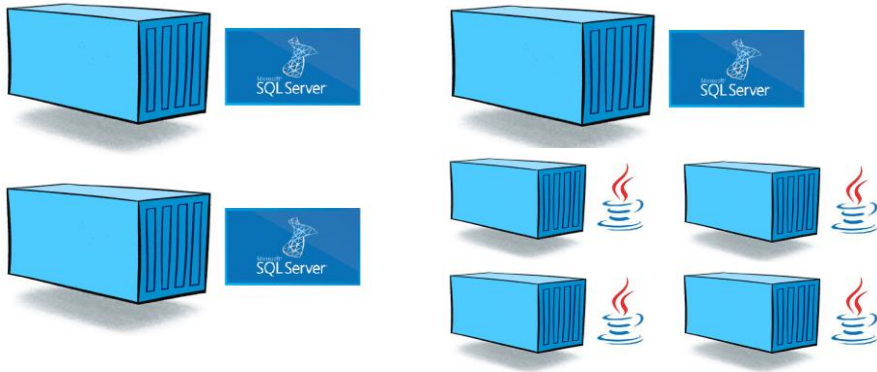
- > Strong Open Source and Linux culture (Windows OS is not an option)
- > Migration from Oracle to SQL Server
- > Docker is a big part of their main strategy
  - > Recent applications are moving on Docker

# Development and production feedbacks

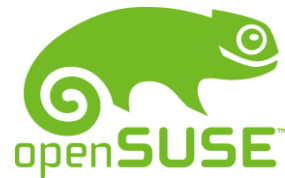
## Production customer case

> docker run -d ... microsoft/mssql-server-linux

**Production leads to different concerns:**



Docker Engine standalone



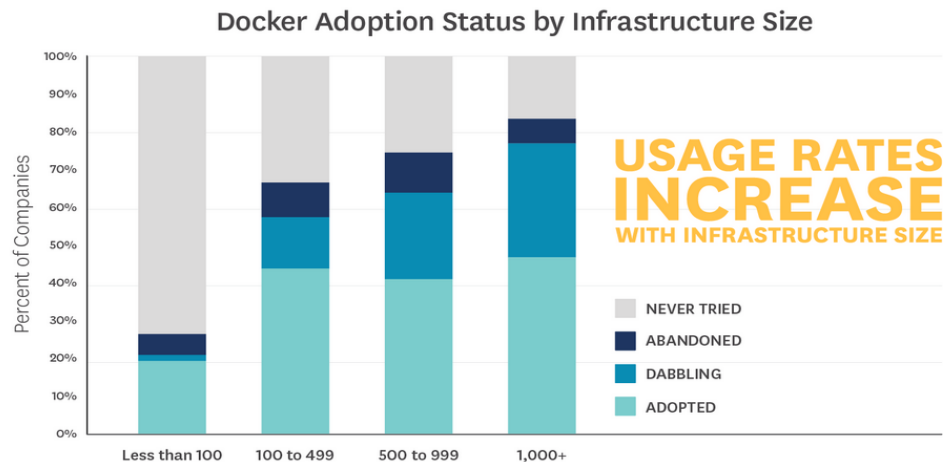
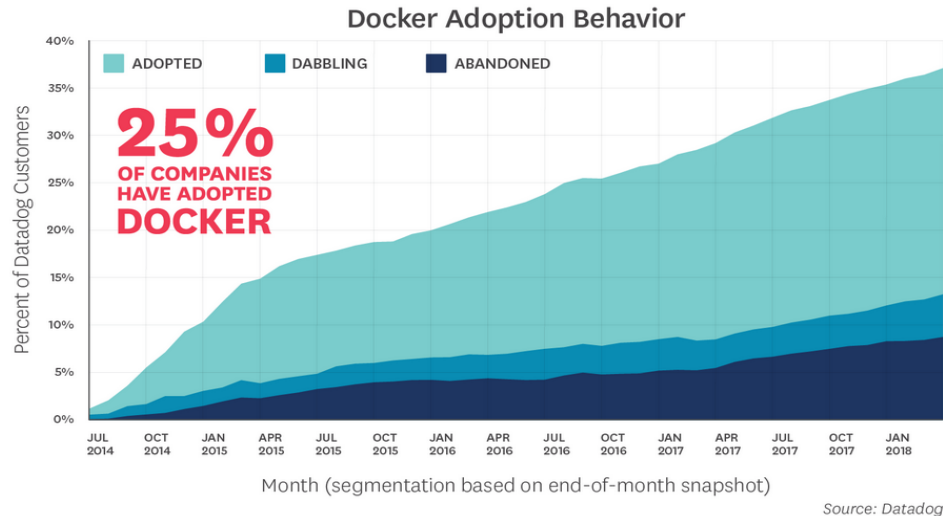
NFS

# Development and production feedbacks

## Production customer case

### Production leads to different concerns:

> Product's maturity ??? (Persistent storage was a blocker for adoption)

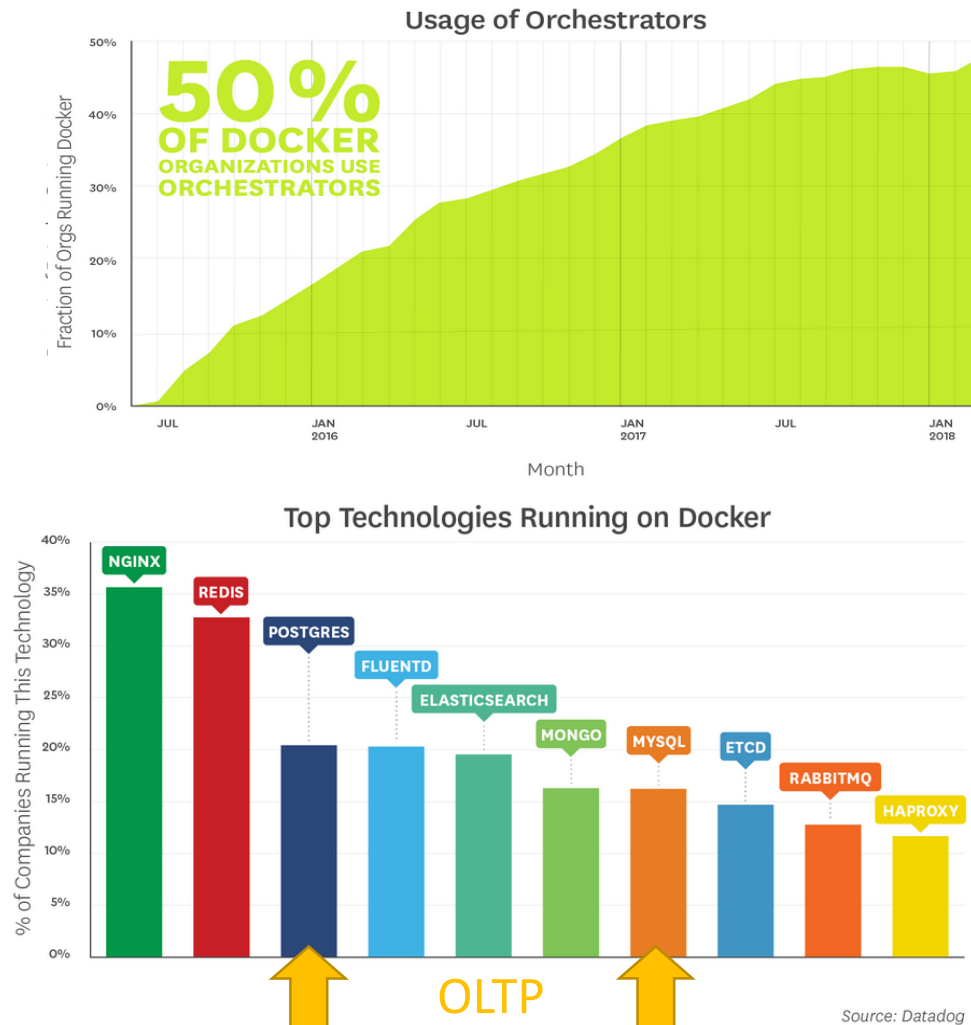


# Development and production feedbacks

## Production customer case

### Production leads to different concerns:

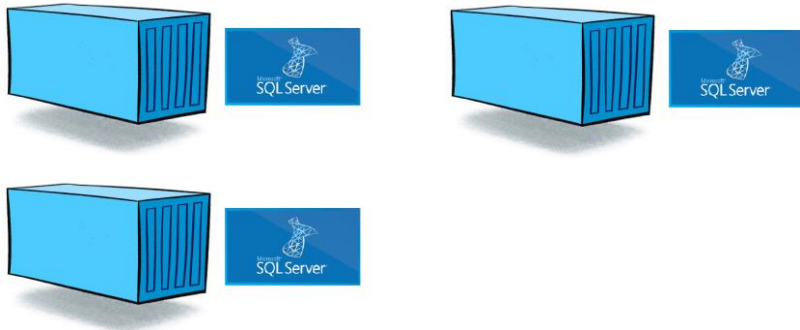
> Product's maturity ??? (Persistent storage was a blocker for adoption)



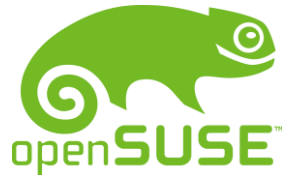
# Development and production feedbacks

## Production customer case

> docker run -d ... microsoft/mssql-server-linux



Docker Engine standalone



NFS

### Production leads to different concerns:

- > Product's maturity ??? (Persistent storage was a blocker for adoption)
- > Docker CE vs ~~Docker EE (UCP, K8s, security features, monitoring..)~~
- > MSSQL Docker images are not "production ready"
  - > Configuration / Performance best practices
  - > DB maintenance & backup (By instance or ~~Third-party tools?~~)
  - > Resource governance (Memory, CPU capping ...)

### Security

- > SQL secrets, OS (Minimal OS, Network, Accesses, Storage ...)

### HA / Load balancing

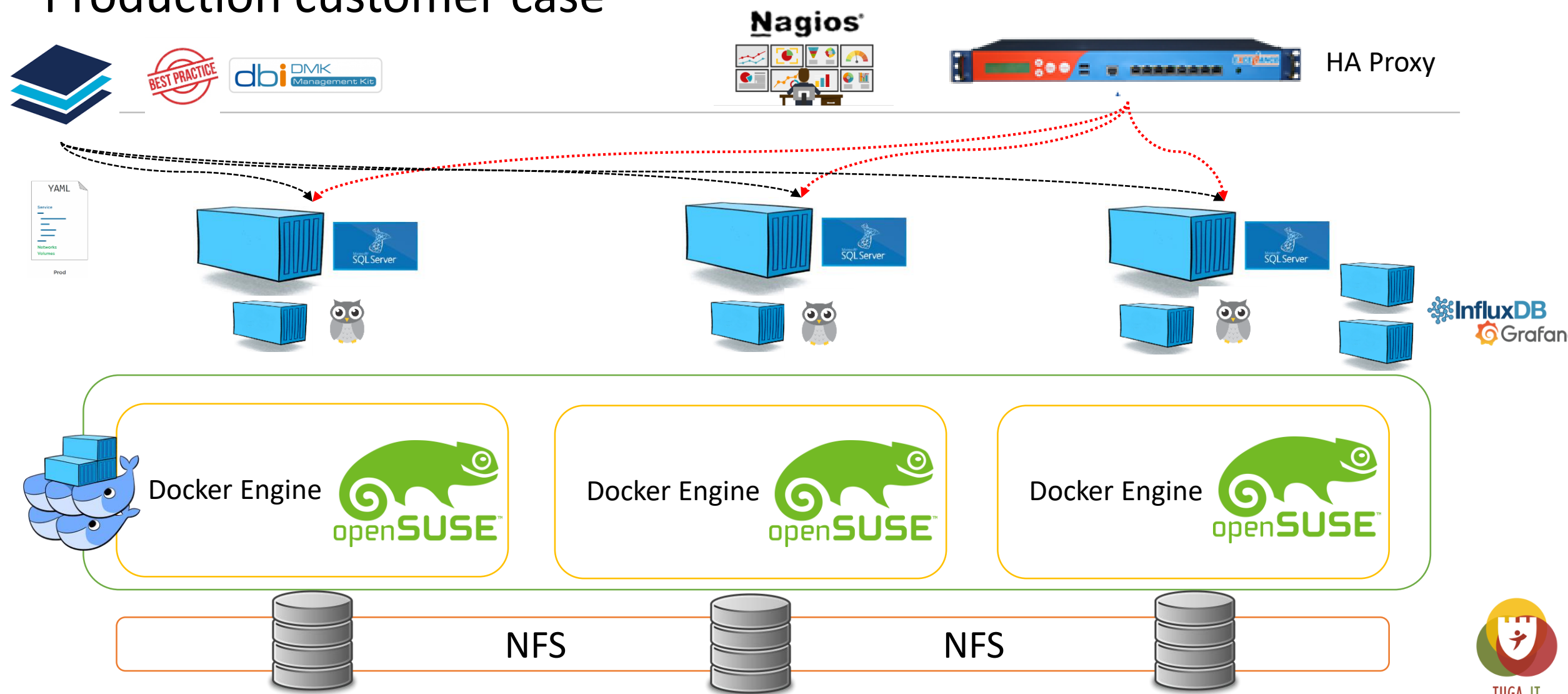
- > Orchestrators (Docker Swarm / ~~Kubernetes / OpenShift, Cloud ...~~)

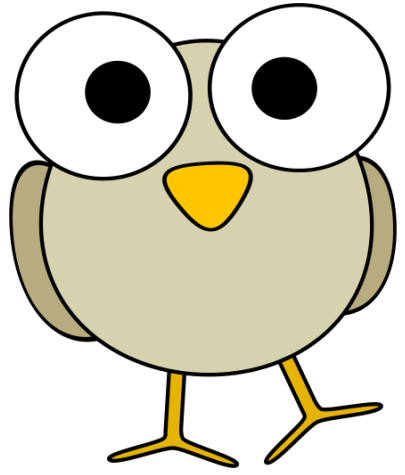
### New application deployment model

- > Stacks / services

# Development and production feedbacks

## Production customer case





??

Questions?

Thanks for listening!



PLEASE FILL IN THE  
EVALUATION FORM.

YOUR OPINION IS  
**IMPORTANT!**



AT THE ENTRANCE AFTER THE LAST SESSION OF THE DAY



#TUGABEER sponsored by

|create|**it**|  
INNOVATING LIFE



# THANK YOU TO OUR SPONSORS



Microsoft

GOLD SPONSOR

**bi4all**

CREATING BUSINESS INTELLIGENCE

SILVER SPONSOR

|create|**it**|

INNOVATING LIFE

TUGA BEER SPONSOR

**FARFETCH**

SWAG SPONSOR

