

# DEVOPS MAGIC

Creating S4 & ECC Replicas in Minutes

# Introductions



**Nathan Pinger**  
COO - Nuve Platform

# Agenda

Motivation

DevOps *Magic*

New Possibilities

Conclusions

# Agenda

Motivation

DevOps *Magic*

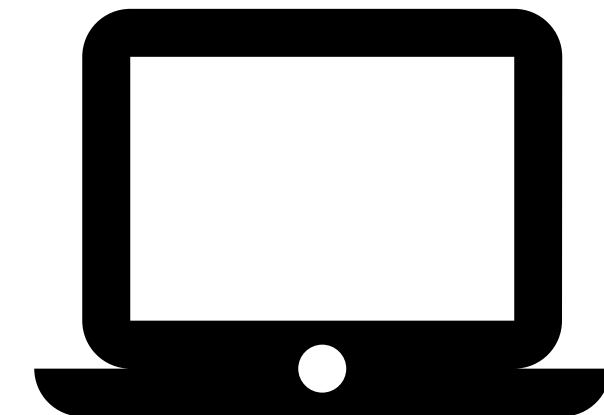
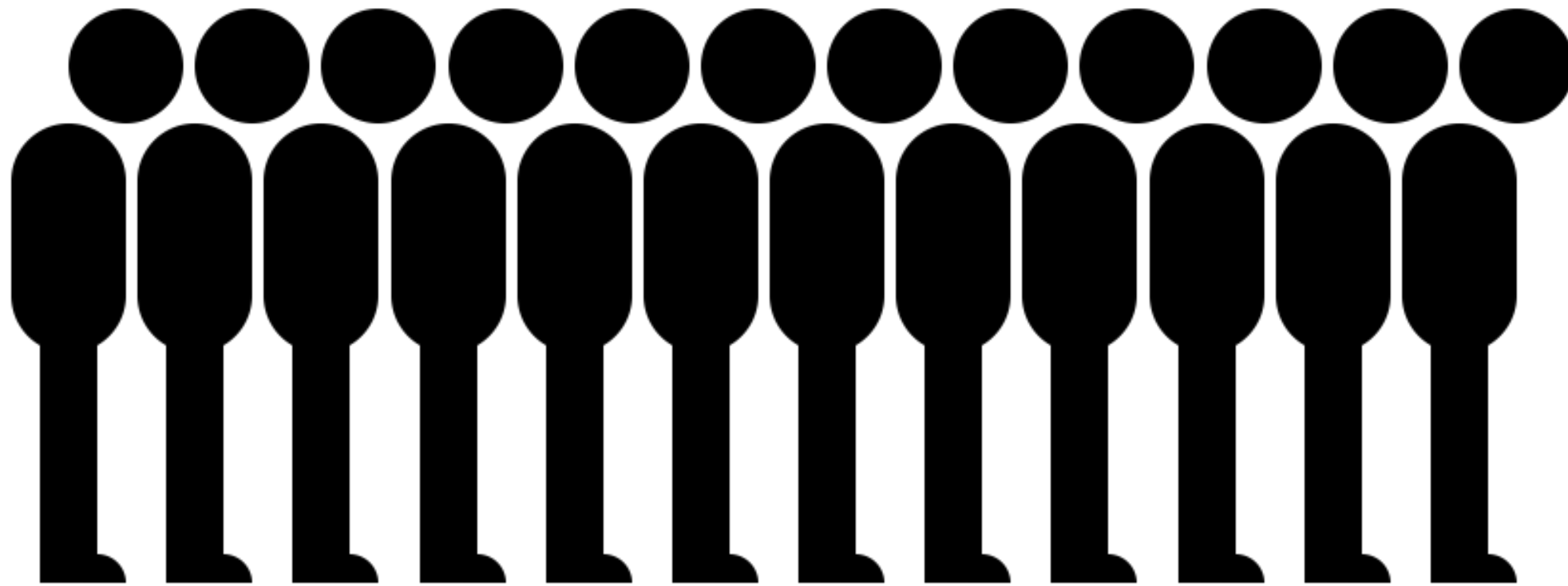
New Possibilities

Conclusions

# Motivation: Shared Environments Are A Hassle!

Waiting in line for access to a system is a drag.

Making a mistake on that system is worse, because it blocks others.



## Motivation

There is a Playbook

Lessons from modern software development . . .

- Isolate Environments
- Develop and Experiment in Parallel
- Integrate When Ready

# Motivation

Goals of the Playbook

The goals these changes achieve:

Reduce risk:

- Mistakes don't block others
- Mistakes don't block you

Better results:

- Better data --> Better tests
- Happier developers
- More projects running in parallel

# Motivation: How is this Possible Now?

Maturing infrastructure, provisioning, version control, and workflows leads to a completely new way of developing on SAP systems.



Mature  
Infrastructure



Mature  
Provisioning



Mature Version  
Control



**GitHub**



**abapGit**



# Motivation

A Note on Costs

Cost cutting is a reality, but remember . . .

Developer time is more expensive than hardware.

A team of 3 devs **waiting 7 business days** for access to a system is **more costly** than running another system\*.

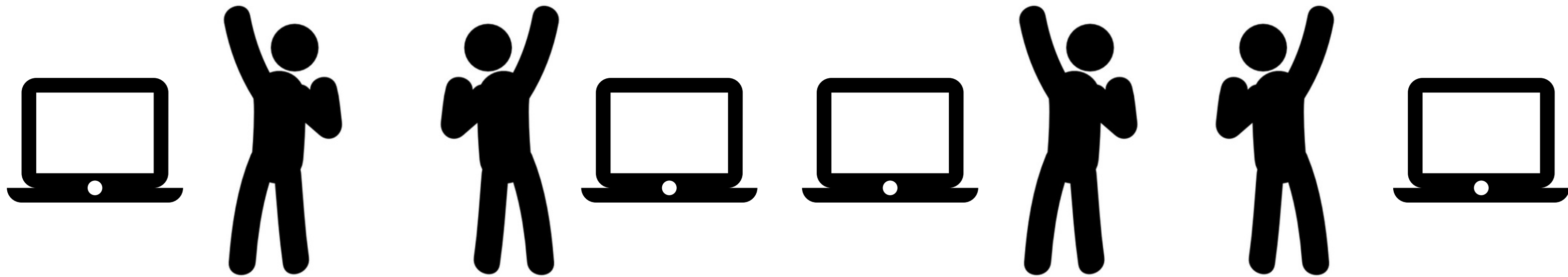
\*Rough figures:

3 Person Dev Team = \$27,500 / month

1.5TB S/4HANA system = \$8,750/month

# Isolated Systems Simplify Things

Let's move to a world where each team (and developer) can do their work.



# Agenda

Motivation

DevOps Magic

New Possibilities

Conclusions

# DevOps Magic

Options to Implement

How it can work with SAP . . .

- Lots of Basis Resources
- Custom DevOps Processes
- Containerization and Replication

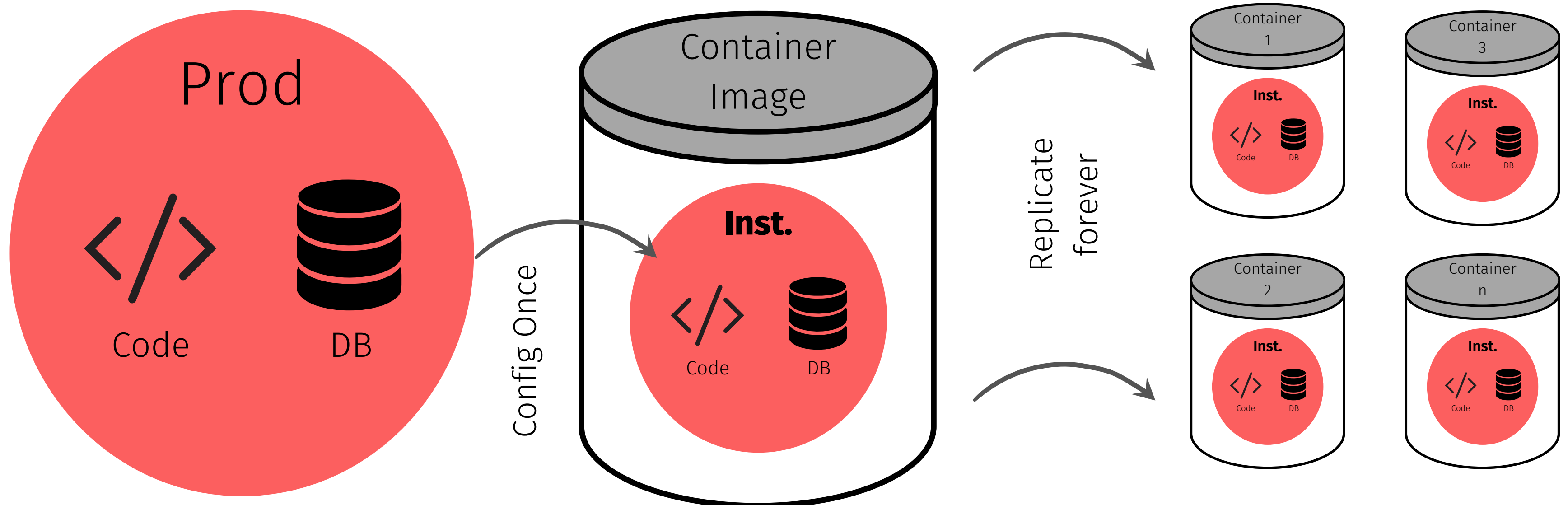
# DevOps Magic

Containers change the game

Stand by for a brief  
demonstration . . .

# DevOps Magic: Mechanics

Replicate prod into a container. Spin up new instances at will.



## DevOps Magic

Containers change the game

What containers allow . . .

- On-demand environments
- Minutes to provision  
(ephemeral and cost effective)
- No waiting for other devs
  - Faster development
  - Lower risk development

# DevOps Magic

## Key Considerations

### To Ensure High Quality Dev/Test Cycles

- Use systems that better represent production
- QA as you develop (and then QA again with isolated QA environments)
- Merge (transport or other) when you are confident the solution works



# Agenda

Motivation

DevOps *Magic*

New Possibilities

Conclusions



## **New Possibilities**

Object un-lock

The Benefits . . .

Low Risk

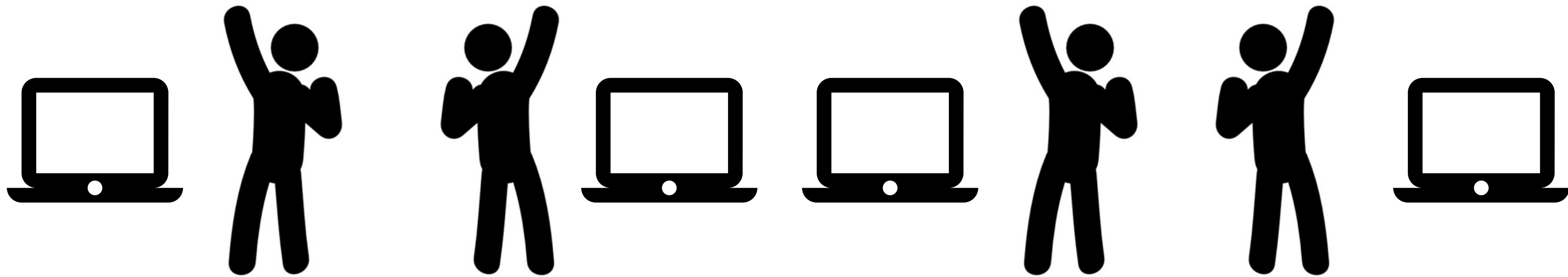
- Your work does not impact others
- Try as much as you want, and reset if it fails

No Waiting

- Eliminate object locks

# New Possibilities: Question

What is possible when each team (or dev) has their own system?



# New Possibilities: Concrete Uses



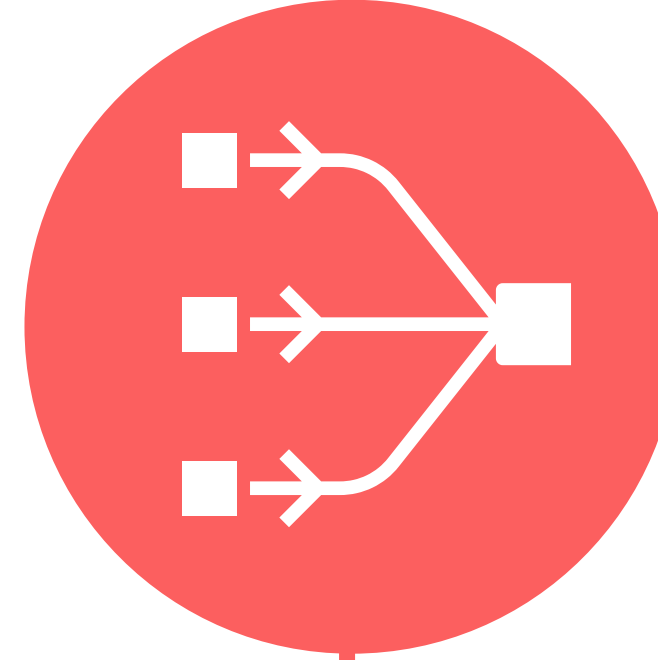
Pre-Sales



M&A



Migrations  
(ECC->S4)



Parallel  
Projects



Testing  
(Automated)

## New Possibilities

A Quick Case Study

Working with a client on an ECC to S4 migration.

Only had 1 S4 instance (future prod).

One partner wanted to keep moving with the migration on a tight timeline.

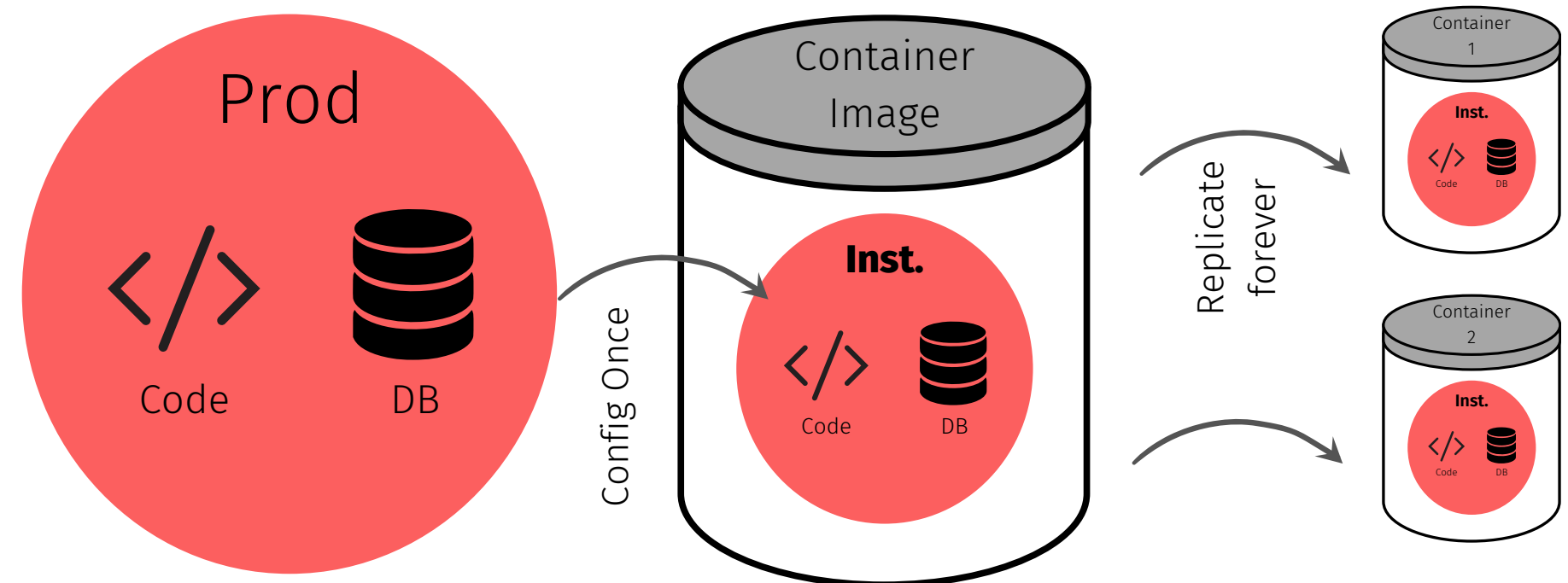
Second partner wanted access to the same system, but would block the migration.

# Parallel Workstreams

A Quick Case Study

Solution:

- Containerize prod from a back up (3 day turnaround)
- Provision 1st copy for Partner A (3 hours)
- Provision 2nd copy for Partner B (3 hours)
- Provisioning happens in parallel



# Agenda

Motivation

DevOps *Magic*

New Possibilities

Conclusions

# Conclusions

The Future is Coming

This trend is inevitable . . .

- Hardware costs falling
- Developer costs rising
- Infrastructure maturing



## Conclusions

The Future is Coming

Teams are already innovating

- In house solutions
- 3rd party platforms

## Conclusions

The Future is Coming

The playbook exists for change:

- Talk with developers from other spaces (Web, DevOps, SRE)
- Practical considerations
  - Stability vs. Speed (vs both)?
  - Start small, then expand
  - Step by step vs. big bang

# Isolated Enviroments

Containers change the game

Back to the demo . . .

# Do you have any questions?

Feel free to reach out!



**Nathan Pinger**

@nuveplatform

npinger@nuveplatform.com

(312) 379-9795