

Managing Clouds and Containers

You've made the initial journey into the Cloud. Now what? The promise was that it would be cheap and easy (compared to legacy on-prem), but... you've probably stepped on some traps you didn't spot until it was too late.

You can go from "Did we make a mistake to go cloud?" to "Superhero of the Cloud" if you take a look at the dangers of cloud laziness AND the opportunities of embracing the most modern cloud practices.

Key Terms

Micro-purchasing:

- ▶ Cloud computing bought purely on usage. Every tiny bit.

Infrastructure-as-a-Service (IaaS):

- ▶ Cloud computing offerings that provide VMs, containers, storage, and supporting networking infrastructure.

Containers:

- ▶ Next Gen VMs that leverages the host OS kernel.

DevOps:

- ▶ Practices and tools for collaboration combining Dev and Ops teams.

Infrastructure as Code (IaC):

- ▶ Ability to automatically manage tech stacks through software, rather than manually.



The Cloud Giveth & Spendeth... Without Endeth

(LEAP TO CHAPTER 1)

- ▶ Infinitely cloud - infinitely flexible AND expensive, without visibility into ALL of your cloud spend in PROD *and* other environments.
- ▶ Micro-Purchasing adds up, creating a Black Hole of Dark Spend - "unofficial" cloud use that happens 24/7.
- ▶ Grab a clipboard and cost flashlight to hunt your spend



One Does Not Just "Go to the Cloud"

(LEAP TO CHAPTER 2)

- ▶ Environment sizing = voodoo guesswork. Until you could get ALL of the data around performance and capacity, automatically.
- ▶ There are at least five types of cloud compute instances. And infinite chances to underprovision or overspend... or both!



Autoscaling isn't magic - plan ahead for the flexibility you need for scale

(LEAP TO CHAPTER 3)

- ▶ It's tempting to keep the legacy of horizontal scaling (throwing more boxes at the datacenter)... but often the wrong approach for today's apps
- ▶ Webs, microservices and containers - just the tip of the iceberg of great use cases for vertical scaling



From containers to K8s - orchestrate before they overrun your admins

(LEAP TO CHAPTER 4)

- ▶ Containers are the next evolution of virtualization, so take the lessons learned from the past and get smarter with containers before they multiply
- ▶ Automation to create containers on-demand needs to be balanced out with orchestrating the ongoing management of containers. Enter... extensive Open Source options



DevOps isn't just for DevOps and developers; it's expanding from "typical" code to IaC

(LEAP TO CHAPTER 6)

- ▶ Managing virtual infrastructure through Infrastructure as Code (IaC) provides your infrastructure monkeys the powers that devs have taken for granted since the rise of DevOps. Version control, rollback and more!
- ▶ The Continuous Journey - Continuous Integration (CI), Development (CD), and now Optimization (CO) - making it all MUCH smarter.
- ▶ CO needs feeding! Feed process workload and app data for always updated machine learning models to optimize both your evolving code AND the infrastructure that morphs with it.

Download the Full Gorilla Guide!

The Densify Gorilla Guide to Cloud & Container Resource Management is here to help. You'll learn how to control your environment in a practical and cost-effective way, to get the maximum benefit of the cloud with minimum hassle.

Inside this Guide:

- ▶ The risk of uncontrolled micro-purchasing
- ▶ The importance of cloud resource management
- ▶ Container orchestration and optimizing Kubernetes resource efficiency
- ▶ The benefits of infrastructure as code
- ▶ Why automation is the key to proper DevOps

GET YOUR COPY!

