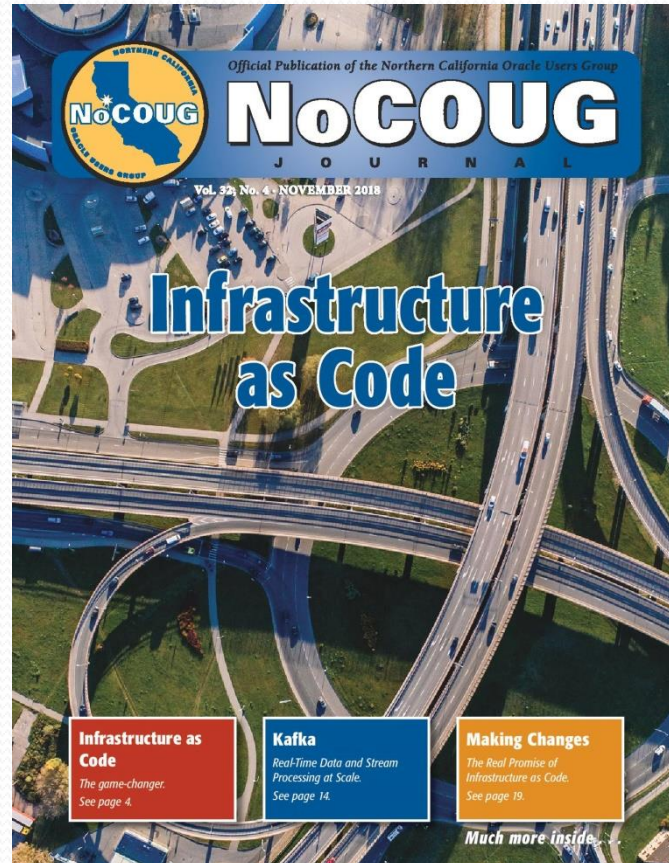# Trends and Game-Changers in the Cloud

Iggy Fernandez

"Infrastructure as Code" demo by Artem Danielov

# NoCOUG Journal

# Latest Issue

# NoCOUG Journal Archive

wget www.nocoug.org/Journal/NoCOUG_Journal_{2001..2018}{02..12..3}.pdf

# Agenda

- Competition from Oracle
  - Bonus: Oracle Database licensing in the cloud
- Infrastructure as Code
  - Bonus: Demo of creating a RAC cluster
- Cloud-Native SQL Databases

# Competition from Oracle

# State of the Cloud

| 2017 Global IT Spend | $3.5 Trillion |
|---|---|
| 2017 Public Cloud Revenue | $270 Billion |
| 2016 Operational DBMS Market | $34.4 Billion |
| 2016 Oracle Corporation Revenue | $37 Billion |
| 2017 AWS Revenue | $17.5 Billion |
| 2016 Oracle Database Revenue | $13.9 Billion |
| 2016 Open Source Database Market | $1 Billion |
| 2016 NoSQL Database Market | $1.5 Billion |
| 2016 AWS Database Revenue | $1.7 Billion |

# Oracle Cloud Infrastructure: Complete Services

**COMPUTE**
**Bare metal, GPUs, VMs**

Up to 52 CPU cores, 8 GPUs, 768 GB RAM, 51 TB local NVMe SSD, 5M IOPS

**STORAGE**
**NVMe, Block, File, Object, Archive**

Predictable IOPS Block Storage for up to 98% less than AWS, High scale file & object storage

**DATABASE**
**Bare metal, VMs, RAC, Exadata**

Up to millions of transactions per second; Full RAC and Active Data Guard support

**CONTAINERS**
**Containers and Kubernetes**

Fully managed, certified Kubernetes service with Docker containers

**NETWORKING**
**VCN, Load Balancing**

Isolated private networks with reserved IPs, subnets, security lists, firewalls, load balancing

**EDGE**
**DNS, VPN, FastConnect, Email**

Global DNS, global private connectivity at up to 97% less, email delivery

**RAVELLO**
**Migrate VMware or KVM**

Move VM environments, retaining existing networking, to the cloud

**CLOUD AT CUSTOMER**
**IaaS, PaaS, Exadata On-premises**

Subscription-priced cloud infrastructure, PaaS, and database managed by Oracle

**DATA MOVEMENT**
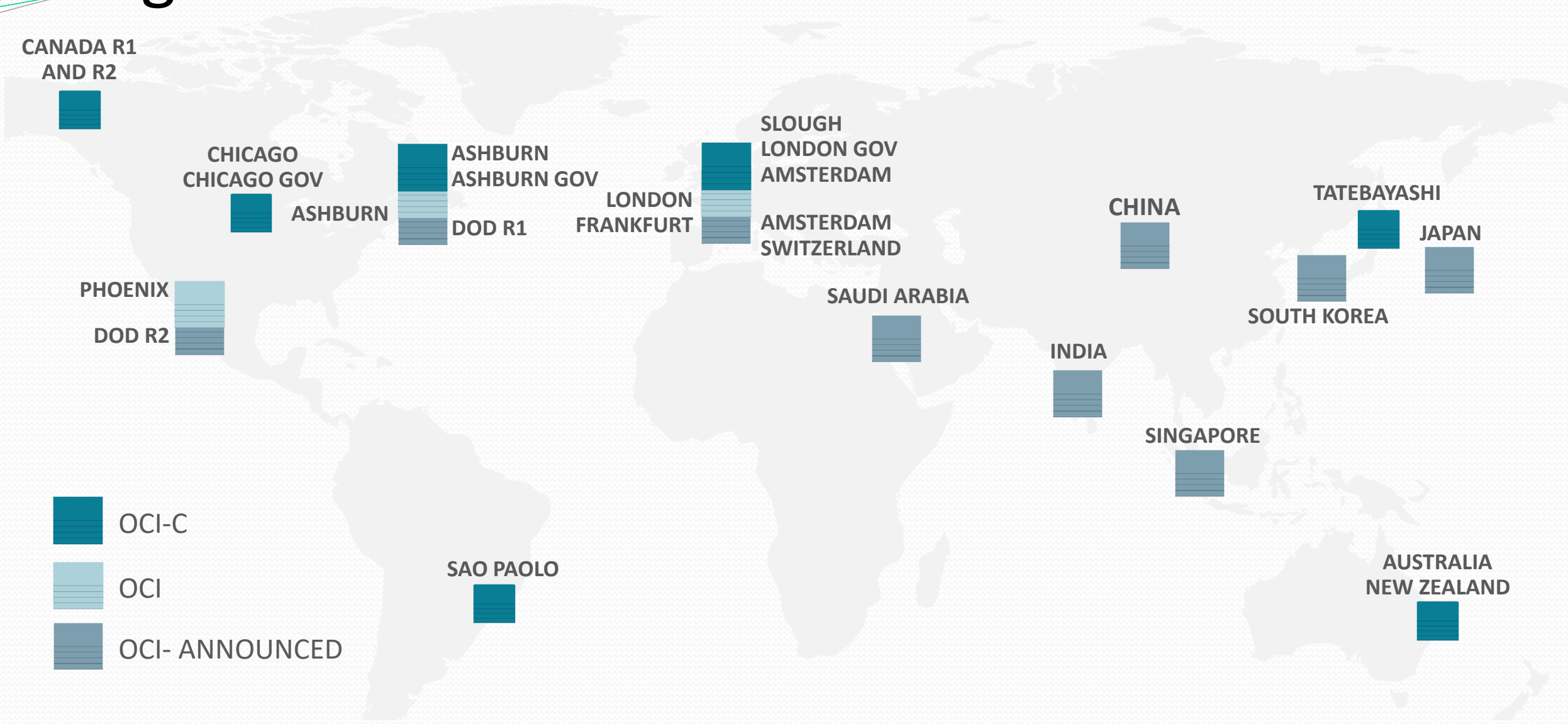**Storage appliance, Data Transfer**

Software NAS gateway, data ingest service with full chain of custody (HDD or appliance)

# Some Highlights

- Flat, non-blocking network
  - Predictable low latency without noisy neighbors
- Off-box IO virtualization
  - Secure bare metal servers without Oracle management software overhead
- Direct-attached NVMe storage
  - Millions of IOPS if you need it

# Regions

CANADA R1
AND R2

CHICAGO
CHICAGO GOV

ASHBURN
ASHBURN GOV

ASHBURN

DOD R1

PHOENIX

DOD R2

LONDON
FRANKFURT

SLOUGH
LONDON GOV
AMSTERDAM

AMSTERDAM
SWITZERLAND

SAUDI ARABIA

CHINA

TATEBAYASHI

JAPAN

SOUTH KOREA

INDIA

SINGAPORE

OCI-C

OCI

OCI- ANNOUNCED

SAO PAOLO

AUSTRALIA
NEW ZEALAND

# Network Design

- Flat network, based on Clos network design, speeds traffic by reducing the switches between any two hosts

- High speed interconnects:  2 x 25Gbps bandwidth

- Predictable, low latency  < 100μs expected one-way latency between hosts in an availability domain

# No charge for storage performance

- Oracle
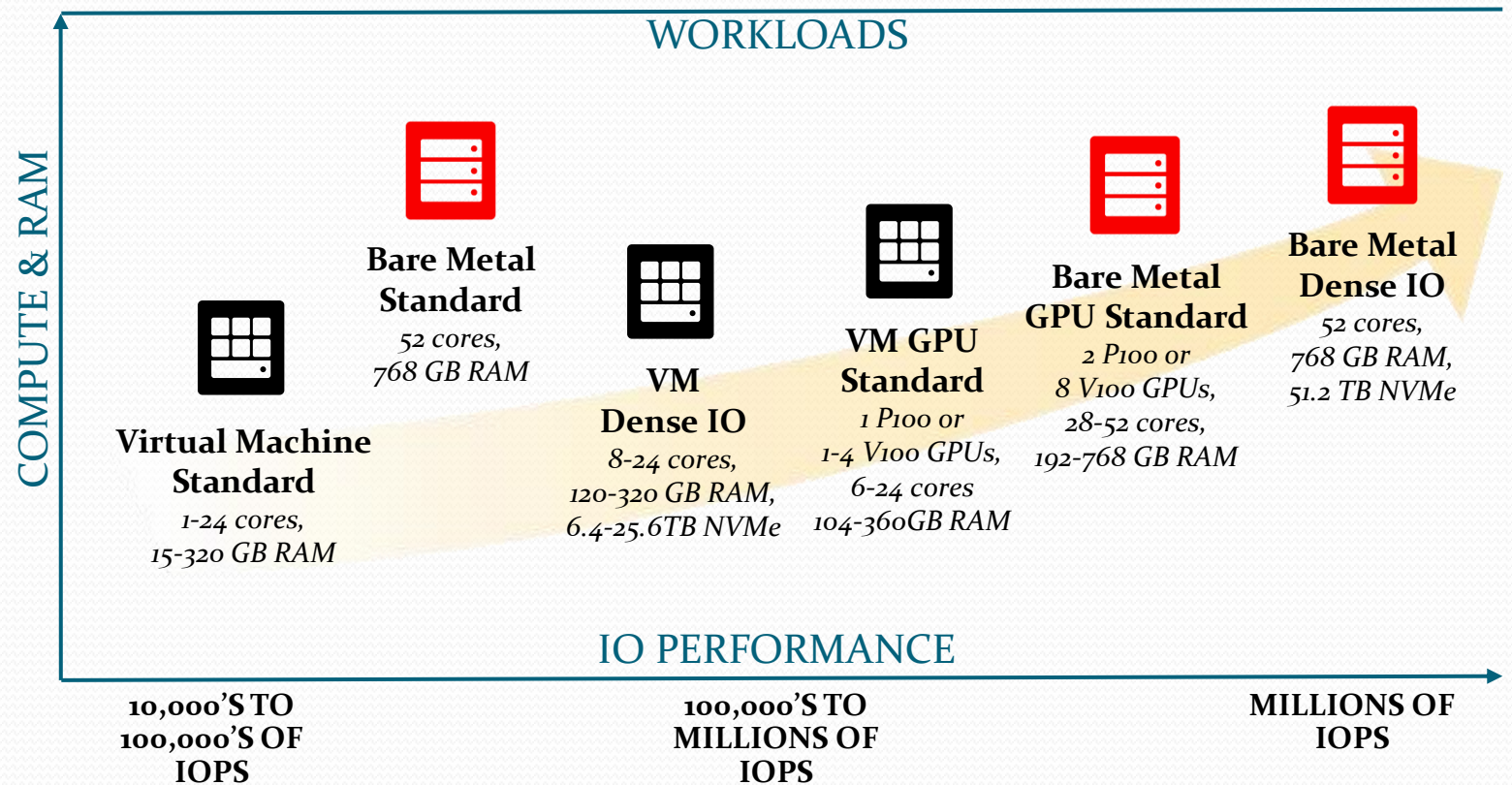  - NVMe SSD
- AWS
  - Provisioned IOPS SSD

# Storage Types

- **Local NVMe SSD storage**
  - VMs and Bare Metal compute with up to 51 TB local storage; backed by performance SLA
- **Block Storage**
  - Network NVMe SSD block storage with 60 IOPS/GB; Max 25K IOPS, 320 MB/s per volume; backed by performance SLA
- **Object and Archive Storage**
  - Object storage with S3 and HDFS compatibility
- **File Storage**
  - Low cost network NVMe SSD file storage with 150 MB/s per TB

# Shapes

- **Shapes**
  - Virtual Machines or Bare Metal
  - Up to 52 Intel Xeon cores
  - Up to 8 V100 GPUs per instance
  - Up to 51 TB local NVMe SSD storage
  - Dual 25 Gbps network interfaces
  - Up to 1 PB network NVMe SSD block storage per instance

WORKLOADS

COMPUTE & RAM

**Bare Metal Standard**
*52 cores, 768 GB RAM*

**Virtual Machine Standard**
*1-24 cores, 15-320 GB RAM*

**VM Dense IO**
*8-24 cores, 120-320 GB RAM, 6.4-25.6TB NVMe*

**VM GPU Standard**
*1 P100 or 1-4 V100 GPUs, 6-24 cores 104-360GB RAM*

**Bare Metal GPU Standard**
*2 P100 or 8 V100 GPUs, 28-52 cores, 192-768 GB RAM*

**Bare Metal Dense IO**
*52 cores, 768 GB RAM, 51.2 TB NVMe*

IO PERFORMANCE

10,000'S TO 100,000'S OF IOPS
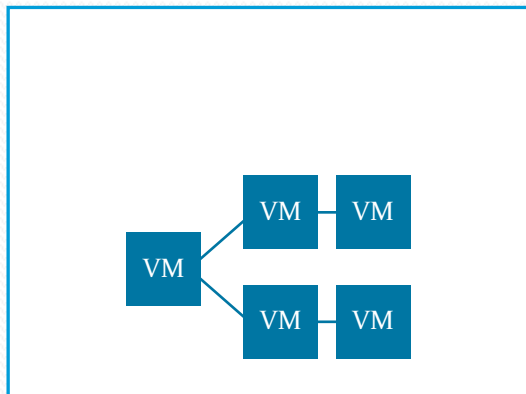
100,000'S TO MILLIONS OF IOPS

MILLIONS OF IOPS

# Cloud at Customer

- Oracle Cloud at Customer
- Exadata Cloud at Customer
- Big Data Cloud at Customer
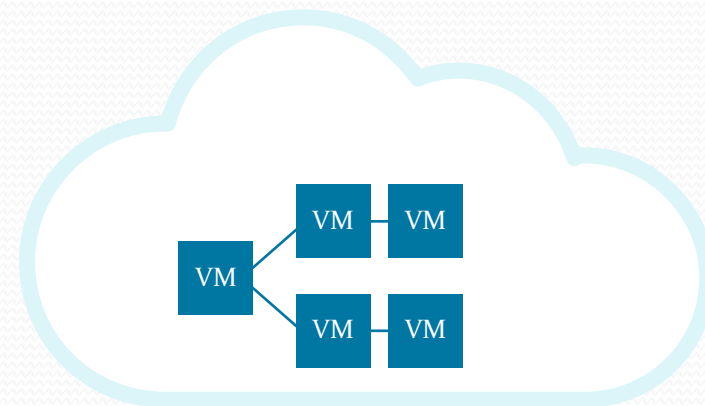- SaaS Cloud at Customer

# Lift-and-shift Vmware and KVM apps

- Applications move to the cloud "as-is," without requiring changes of any kind to the VMs, networking or storage

PRODUCTION APPLICATION ENVIRONMENT

EXACT REPLICA IN THE PUBLIC CLOUD

# Volume Groups

- Group together multiple block storage volumes and perform crash-consistent, point-in-time, coordinated backups and clones across all the volumes in the group

- Point-in-time consistent and coordinated backups and clones of running enterprise applications that span multiple storage volumes across one or more compute instances

# Data Transfer Appliance

- Oracle-branded, purpose-built storage appliance to cost-effectively and easily migrate data to the cloud
- Up to 150TB per appliance
- Free

# Autonomous Everything

- Autonomous Data Warehouse
- Autonomous Transaction Processing

# Oracle Database licensing in the cloud

- Per-core and per-thread licensing are not compatible with the Oracle contract

- The core-factor is not applicable to cloud providers other than Oracle

- *"If a cloud provider can offer Oracle contract-compliant and audit-compatible mechanisms for tracking all physical processors where Oracle software is installed and/or running, then the customer may be able to consider applying their hardware-based licenses in these clouds."*—Reference NoCOUG Journal

  - Dedicated hosts (EC2 only)

# Infrastructure as Code

# What?

- Tools that allows you to programmatically manage, version, and persist your IT infrastructure as code

- Use declarative syntax to describe your infrastructure

- Persist descriptions in configuration files that can be shared, reviewed, edited, versioned, preserved, and reused

Reference: Infrastructure as Code: Managing Servers in the Cloud by Kief Morris

# Problems

- Server Sprawl
- Configuration Drift
- Snowflake Servers
- Fragile Infrastructure
- Erosion

Reference: Infrastructure as Code: Managing Servers in the Cloud by Kief Morris

# Principles

- Systems Can Be Easily Reproduced
- Systems Are Disposable
- Cattle, Not Pets
- Systems Are Consistent
- Processes Are Repeatable
- Design Is Always Changing

Reference: Infrastructure as Code: Managing Servers in the Cloud by Kief Morris

# Practices

- Use Definition Files
- Self-Documented Systems and Processes
- Automatically Generating Documentation
- Version All the Things
- Continuously Test Systems and Processes
- Small Changes Rather Than Batches
- Keep Services Available Continuously

Reference: Infrastructure as Code: Managing Servers in the Cloud by Kief Morris

# Outcomes

- Every element of the infrastructure can be rebuilt quickly, with little effort
- All systems are kept patched, consistent, and up to date
- Standard service requests, including provisioning standard servers and environments, can be fulfilled within minutes, with no involvement from infrastructure team members
- SLAs are unnecessary
- Maintenance windows are rarely, if ever, needed
- Changes take place during working hours, including software deployments and other high-risk activities

Reference: Infrastructure as Code: Managing Servers in the Cloud by Kief Morris

# Examples

- Proprietary
  - Cloud Formation (Amazon Web Services)
  - Azure Resource Manager (Microsoft Azure)
- Open-Source
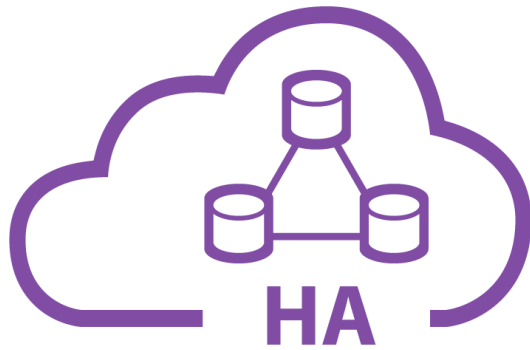  - Terraform (Oracle Cloud Infrastructure)

# Languages

- JSON
- YAML
- HCL

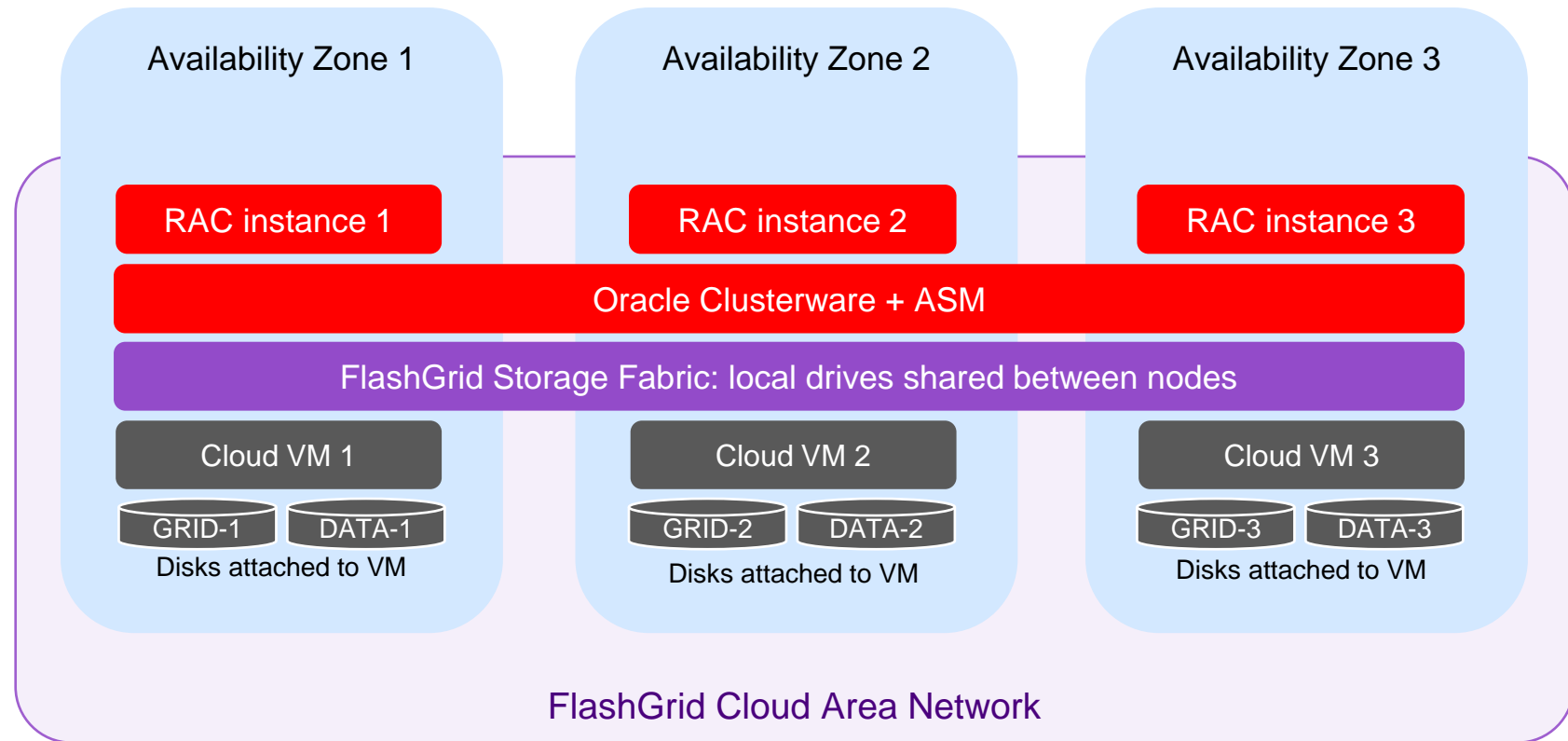# Infrastructure-as-Code Demonstration

Creating an entire Oracle RAC cluster using a JSON configuration file

- FlashGrid SkyCluster Intro
- FlashGrid SkyCluster Architecture:
- Demo: configuring and deploying SkyCluster with Oracle RAC in AWS EC2

# FlashGrid SkyCluster Intro

- ❖ Active-Active high availability with 2+ database nodes

- ❖ Infrastructure-as-Code deployment with a few mouse clicks

- ❖ Proven Oracle RAC database engine

- ❖ 24/7 support

**Run your most critical Oracle databases in the cloud**

# FlashGrid SkyCluster | Oracle RAC in Public Clouds

flash**grid**

| Availability Zone 1 | Availability Zone 2 | Availability Zone 3 |
|---|---|---|
| RAC instance 1 | RAC instance 2 | RAC instance 3 |

**Oracle Clusterware + ASM**

**FlashGrid Storage Fabric: local drives shared between nodes**

| Cloud VM 1 | Cloud VM 2 | Cloud VM 3 |
|---|---|---|
| GRID-1  DATA-1 | GRID-2  DATA-2 | GRID-3  DATA-3 |
| Disks attached to VM | Disks attached to VM | Disks attached to VM |

**FlashGrid Cloud Area Network**

- ❖ FlashGrid Cloud Area Network creates a high-speed network overlay with multicast, VIPs, and QoS
- ❖ FlashGrid Storage Fabric creates shared storage from locally attached disks
- ❖ Proven Oracle Clusterware and ASM for high availability and data mirroring

# IaC Demo: Deploying SkyCluster with Oracle RAC in AWS

flash**grid**

Start the video to see a demo of creating a RAC cluster in AWS using IaC approach.

Note that there is one more layer on top of IaC that we call Engineered-System-as-Code. It makes deployment of the cluster even easier.

# Cloud-native databases

# Flashback to 2007
# Dynamo Requirements: Extreme Availability

"Customers should be able to view and add items to their shopping cart even if disks are failing, network routes are flapping, or data centers are being destroyed by tornados. Therefore, the service responsible for managing shopping carts requires that it can always write to and read from its data store, and that its data needs to be available across multiple data centers."

Reference: Dynamo: Amazon's Highly Available Key-value Store

# Flashback to 2007
# Dynamo Requirements: Extreme Performance

"There are many services on Amazon's platform that only need primary-key access to a data store. For many services, such as those that provide best seller lists, shopping carts, customer preferences, session management, sales rank, and product catalog, the common pattern of using a relational database would lead to inefficiencies and limit scale and availability. Dynamo provides a simple primary-key only interface to meet the requirements of these applications."

Reference: Dynamo: Amazon's Highly Available Key-value Store

# Flashback to 2007
# Dynamo Requirements: Extreme Scale

"Since each service uses its distinct instance of Dynamo, its initial design targets a scale of up to hundreds of storage hosts [only]."

Reference: Dynamo: Amazon's Highly Available Key-value Store

# Dynamo Requirements Summary

1. Extreme availability
2. Extreme performance
3. Extreme scalability

# Dynamo Solution: Functional Segmentation

- Best seller lists, shopping carts, customer preferences, session management, sales rank, and product catalog
- Increases overall site availability by avoiding a single point of failure
- No distributed transactions (eventual consistency)

# Dynamo Solution: Sharding

- **employee** (employee#, name, birthdate)
- **jobhistory** (employee#, jobdate, title)
- **salaryhistory** (employee#, jobdate, salarydate, salary)
- **children** (employee#, childname, birthyear)

# Dynamo Solution: Replication

- Shards are replicated
- No distributed transactions (Eventual consistency)

# Dynamo Solution: BLOBs

- Shopping carts are stored as binary objects (i.e., blobs) identified by unique keys. No operations span multiple data items and there is no need for relational schema.

- *"You can keep a car in a file cabinet because you can file the engine components in files in one drawer, and the axles and things in another, and keep a list of how everything fits together. You can, but you wouldn't want to."*—Esther Dyson

# Dynamo Solution: Summary

- Functional segmentation
- Sharding
- Asynchronous replication
- Key-value (BLOBs)
- Autocommit
- No distributed transactions
- Eventual consistency
- No SQL

# But Wait, There's More!

- Cloud-native
  - No VMs to configure
  - No backups to configure
  - No standby databases to configure
  - No software to choose
  - No patches to install

# Google Cloud Spanner

- Functional segmentation
- Sharding
- ~~Asynchronous replication~~ Synchronous replication
- ~~Key-value (BLOBs)~~ Relational model, interleaved tables (Oracle-style table clusters)
- ~~Autocommit~~ Transaction support
- ~~No distributed transactions~~ Distributed transactions
- ~~Eventual consistency~~ ACID consistency
- ~~No SQL~~ SQL
- Google Cloud Platform only (Uses atomic clocks and GPS)

# But Wait, There's More!

- Cloud-native
  - No VMs to configure
  - No backups to configure
  - No standby databases to configure
  - No software to choose
  - No patches to install

# Calvin/FaunaDB

- Functional segmentation
- Sharding
- Synchronous replication
- ~~Relational model~~ "Document-relational" model
- ~~Table clusters~~
- Transaction support
- Distributed transactions
- ACID consistency
- ~~SQL~~ "Relationally complete" FQL (SQL support is on the roadmap)
- ~~Google Cloud Platform only (uses atomic clocks and GPS)~~
- 500,000 TPC-C transactions per second on a cluster of commodity machines

# But Wait, There's More!

- Cloud-native
  - No VMs to configure
  - No backups to configure
  - No standby databases to configure
  - No software to choose
  - No patches to install

# Thank you!

https://www.linkedin.com/in/iggyfernandez/

https://www.linkedin.com/in/artem-danielov-66668a1/