For Oracle

MICROSERVICES ARCHITECTURE ADVANCED

By Dr. Vishwanath Rao

Breaking Up Monoliths – Pros and Cons

- Traditional Monolithic Applications and Their Place
- Disadvantages of Monoliths
- Developer's Woes
- Architecture Modernization
- Architecture Modernization Challenges
- Microservices Architecture is Not a Silver Bullet!
- What May Help?
- In-Class Discussion
- Summary

Microservice Development

- What are Microservices?
- Microservices vs Classic SOA
- Principles of Microservices Architecture Design
- Domain-Driven Design
- Domain-Driven Design – Benefits
- Microservices and Domain-Driven Design
- Designing for failure
- Microservices Architecture – Pros
- Microservices Architecture – Cons

- Docker and Microservices
- Microservice Deployment with Docker – Workflow
- Writing Dockerfile
- Kubernetes
- Microservices and Various Applications
- Web Applications

Web Applications – Reference Architecture
Web Applications – When to use?

Single Page Applications
Single Page Applications – Benefits
Traditional Enterprise Application Architecture
Sample Microservices Architecture
Serverless & Event-driven Microservice

Twelve-factor Applications

Twelve-factor Applications
Twelve Factors, Microservices, and App Modernization
The Twelve Factors
Categorizing the 12 Factors

12-Factor Microservice Codebase
12-Factor Microservice Dependencies
12-Factor Microservice Config
12-Factor Microservice Backing Services
12-Factor Microservice Build, Release, Run
12-Factor Microservice Processes
12-Factor Microservice Port Binding
12-Factor Microservice Concurrency
12-Factor Microservice Disposability
12-Factor Microservice Dev/Prod Parity
12-Factor Microservice Logs
12-Factor Microservice Admin Processes

Kubernetes and the Twelve Factors – 1 Codebase
Kubernetes and the Twelve Factors –

2 Dependencies

Kubernetes and the Twelve Factors – 3 Config
Kubernetes and the Twelve Factors – 4 Backing Services
Kubernetes and the Twelve Factors – 5 Build, Release, Run
Kubernetes and the Twelve Factors – 6 Processes
Kubernetes and the Twelve Factors – 7 Port Binding
Kubernetes and the Twelve Factors – 8 Concurrency
Kubernetes and the Twelve Factors – 9 Disposability
Kubernetes and the Twelve Factors – 10 Dev/Prod Parity
Kubernetes and the Twelve Factors – 11 Logs
Kubernetes and the Twelve Factors – 12 Admin Processes

REST Services

Many Flavors of Services
Understanding REST
Principles of RESTful Services
REST Example – Create
REST Example – Retrieve
REST Example – Update
REST Example – Delete
REST Example – Client Generated ID

SOAP Equivalent Examples
REST Example – JSON
Famous RESTful Services
Additional Resources

What is gRPC?
Protocol Buffers
REST vs. gRPC
Protobuf vs. JSONHTTP/2 vs. HTTP 1.1
HTTP/2 vs. HTTP 1.1 (Contd.)

Messages vs. Resources and Verbs
Streaming vs. Request-Response
Strong Typing vs. Serialization
Web Browser Support
REST vs. gRPC – In a Nutshell

Microservices with Node.js

- What is Node.js?
- Node’s Value Proposition
- Example of a Node.js App: a Simple Web Server
- Node.js Project Types
- Managing Large Applications
- Core Modules
- Why Node.js uses JavaScript?
- The Traditional Concurrency Support Model
- Disadvantages of the Traditional Approach
- Event-Driven, Non-Blocking I/O
- The Success Callback Function
- Using Node Package Manager (NPM)
- NPM Registry (Repository)
- NPM Enterprise
- Package Life-Cycle Management
- Local and Global Package Installation Options
- Listing and Using Module Versions

- The Express Package
- Installing and Using Express
- Defining Routing Rules in Express
- Route Path
- The Response Object
- A Simple Web Service with Express ExampleThe MEAN Stack

- Spring Booting Your RESTful Web Service
- Spring Boot Skeletal Application Example
- Converting a Spring Boot Application to a WAR File

Docker Introduction

- What is Docker
- Where Can I Run Docker?
- Installing Docker Container Engine
- Docker Machine

- Docker and Containerization on Linux
- Linux Kernel Features: cgroups and namespaces
- The Docker-Linux Kernel Interfaces
- Docker Containers vs Traditional Virtualization
- Docker Integration
- Docker Services
- Docker Application Container Public Repository
- Competing Systems
- Docker Command Line
- Starting, Inspecting, and Stopping Docker Containers
- Docker Volume
- Dockerfile
- Docker Compose
- Using Docker Compose
- Dissecting docker-compose.yml
- Specifying services
- Dependencies between containers
- Injecting Environment Variables
- runC Overview
- runC Features
- Using runC
- Running a Container using runC

Introduction to Kubernetes

- What is Kubernetes
- What is a Container
- Container – Uses
- Container – ProsContainer – Cons
- Composition of a Container
- Control Groups
- Namespaces
- Union Filesystems
- Popular Containerization Software
- Microservices
- Microservices and Containers / Clusters
- Microservices and Orchestration
- Microservices and Infrastructure-as-Code
- Kubernetes Container Networking
- Kubernetes Networking Options
- Kubernetes Networking – Balanced Design

Leading Practices for Microservice Logging

- Logging Challenges
- Leading Practices
- Correlate Requests with a Unique ID
- Include a Unique ID in the Response
- Send Logs to a Central Location
- Structure Your Log Data
- Add Context to Every Record
- Examples of Content
- Write Logs to Local Storage
- Collecting Logs with Fluentd
- Leading Practices for Microservice Logging Summary
- Metrics Using Prometheus
- Overview
- Prometheus
- Prometheus Architecture
- Service Discovery
- File-based Service Discovery
- Istio and Prometheus
- Exposing Metrics in Services
- Querying in Prometheus
- Grafana
- Business Metrics
- Metrics Using Prometheus Summary
- Tracing Using Jaeger
- OpenTracing
- Jaeger
- Jaeger Architecture Diagram
- Jaeger Client Libraries
- Jaeger Sampling

Traffic Routing Patterns

- Edge Proxy Server
- Request Handling
- Filters
- Filter Architecture
- API Gateway for Routing Requests
- API Gateway for Routing Requests (Contd.)
- API Gateway – Example

Rate Limiting
Rate Limiting – Business Cases
Configuring Rate Limiting in NGINX
Circuit Breaker
Design Principles
Design Principles (continued)
Cascading Failures
Bulkhead Pattern
Circuit Breaker Pattern
Thread Pooling
Request Caching
Request Collapsing
Fail-Fast
Fallback
Circuit Breaker Solutions
Load Balancing in Microservices
Server-side load balance
Client-side Load Balance
Architecture
Service Mesh
Service Mesh (Contd.)
Service Mesh Solutions
Content Delivery Network (CDN)
How does a CDN Work?
Benefits of using a CDN
CDN Solutions