



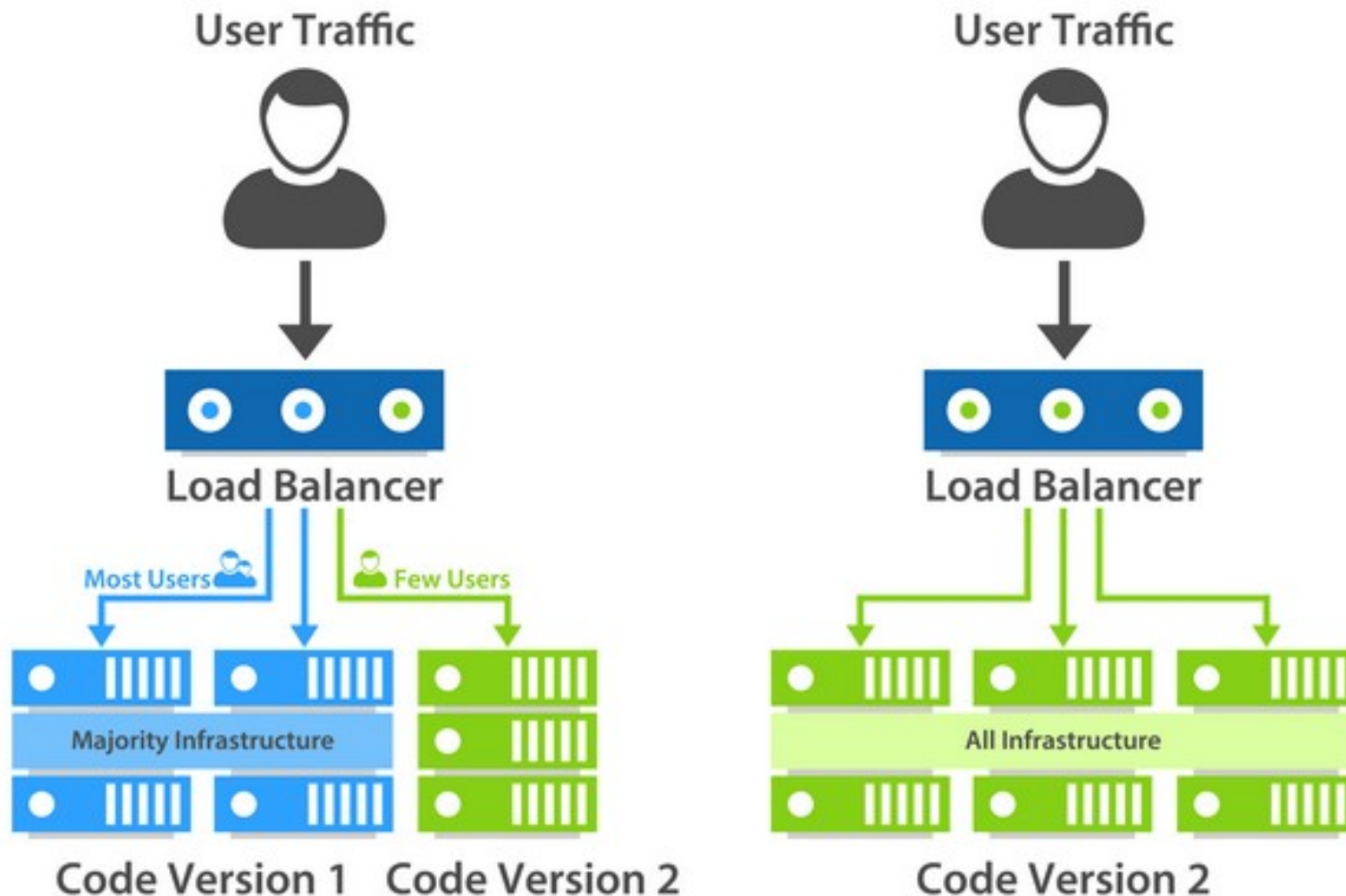
# **Advanced docker / orchestration**



# Orchestration basics

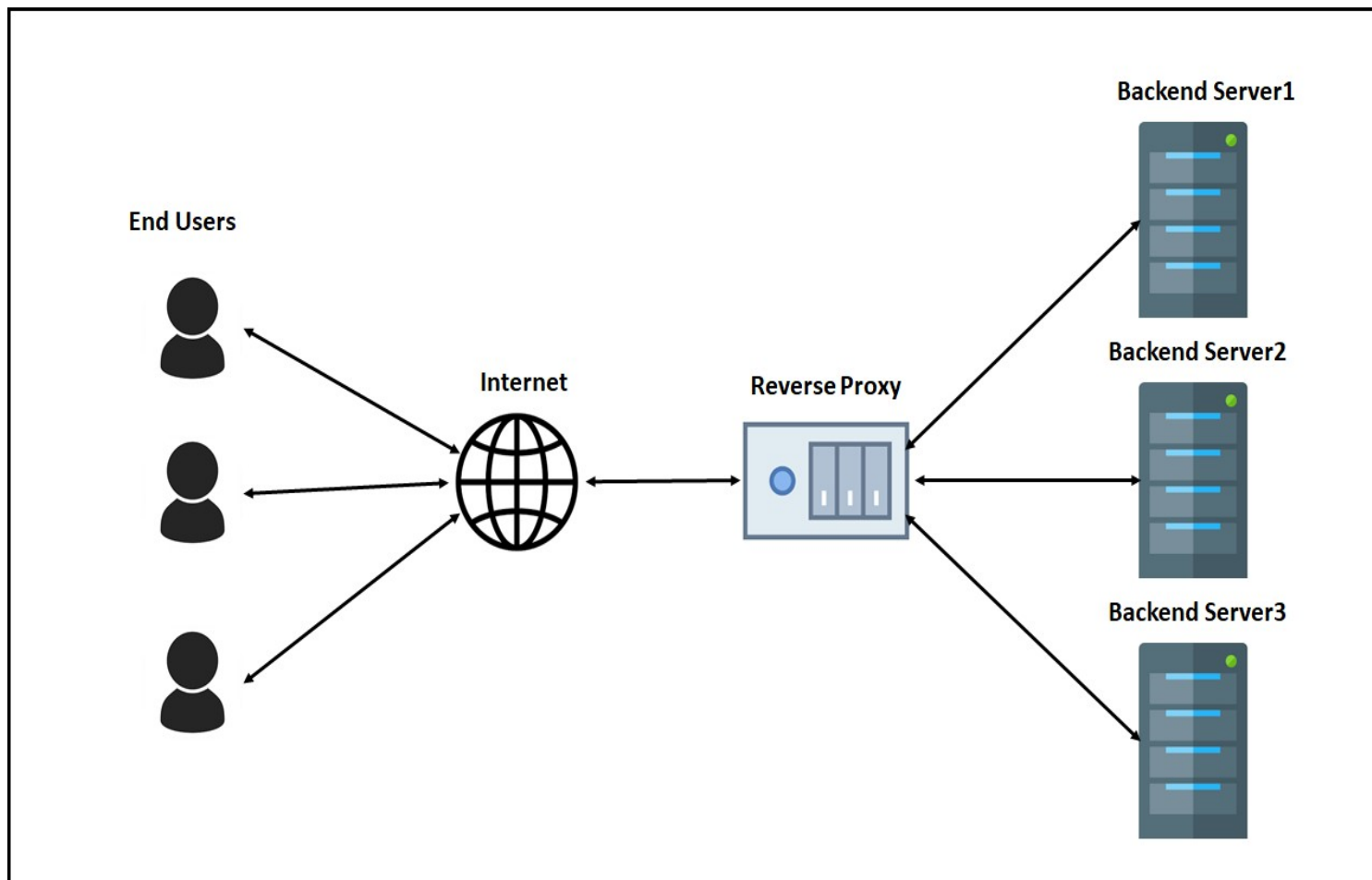
- Run an ecosystem of applications
- Prevent downtime during deployments
- Networking and routing capabilities
- Scalability capabilities

# Deployments downtime



# Networking / routing

- K8s offer a wide range of routing possibilities  
es. Nginx, traefik, calico, istio

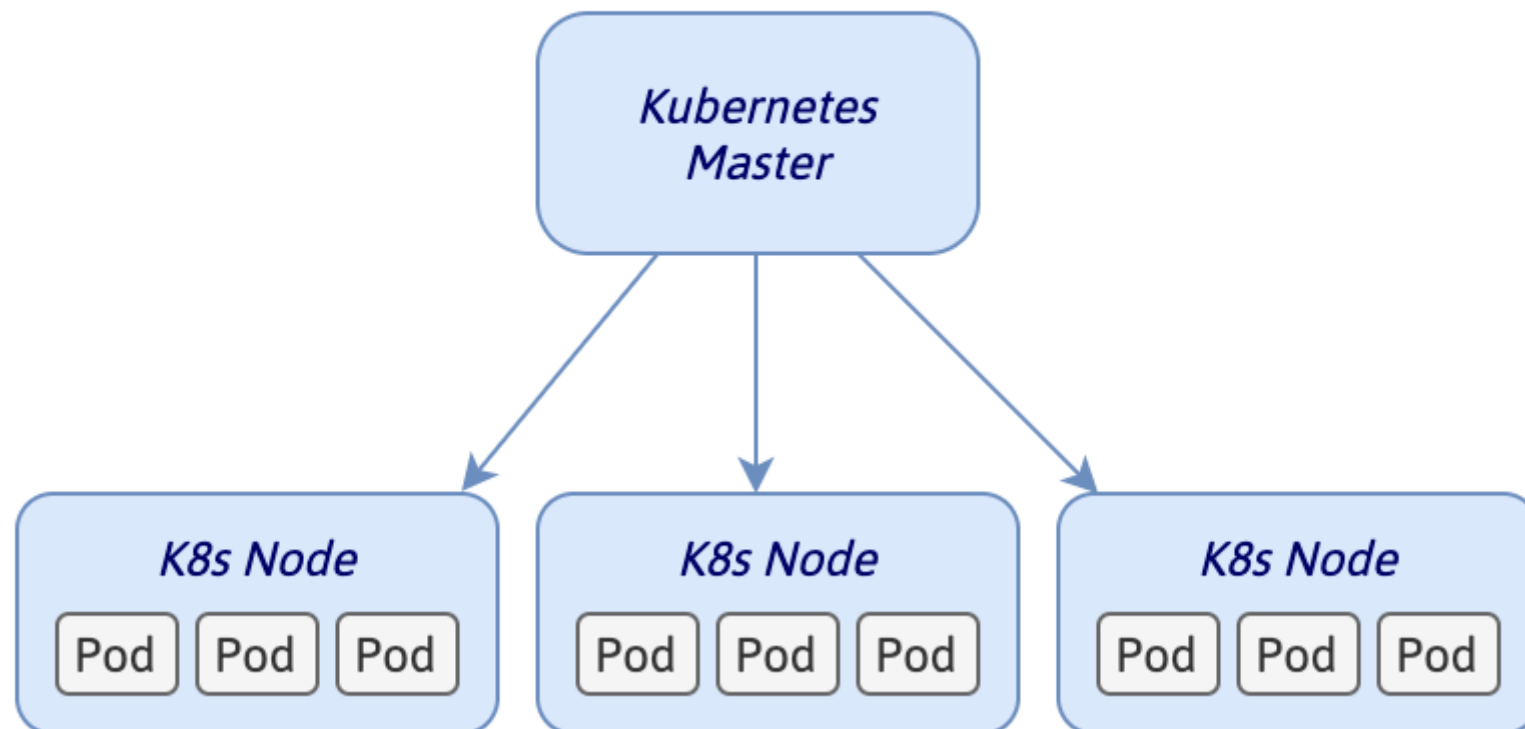


# Scalability

- K8s clusters are composed by nodes. A pool of resources (CPU / RAM) is then shared across pods. A Pod is a set of containers



Kubernetes Cluster





# Orchestration basics

- **Docker-compose**, docker swarm
- Kubernetes, minikube, microk8s
- Openshift (Paas)
- Rancher (hybrid cloud)
- Others: Mesos, Nomad



## Orchestration basics

Docker compose has limited orchestration capabilities:

- use `depends_on` to define start priorities
- Restart policy
- volumes and network
- Basically is a transposition of the docker syntax to a yaml file

# Volumes declaration syntax

- Docker host-mounted volumes

Syntax: `/host/path:/container/path`

- Docker named volumes

Syntax: `volume_name:/container/path`

- Sharing volumes

Syntax: `--volumes-from container_name`



# Volumes declaration syntax

Can create subnets. This enforce isolation between containers

```
version: '3.6'services:
  api-gateway:
    container_name: api-gateway
    image: api-gateway
    networks:
      - gateway
    ports:
      - 9090:8080
    restart:
      on-failure
  api-gateway-replica:
    container_name: api-gateway-replica
    image: api-gateway
    networks:
      - gateway-replica
    ports:
      - 9092:8080
    restart:
      on-failure
networks:
  gateway: {}
  gateway-replica: {}
```

Can customize the ip range of docker.

```
Docker network ls
Docker network inspect my_network
Cat /etc/docker/daemon.json
```



# Assignment 2

- Fetch data from crypto exchange api Kraken
- Adapt the docker-compose project in the ynov-docker repository to let interact the application with the database.
- Show the data via a front end or use grafana.
- Deploy on a local machine via docker-compose.

# Monitoring

- A classic monitoring stack is composed by prometheus, grafana and alertmanager
- Grafana is a multi-platform open source analytics and interactive visualization web application. It provides charts, graphs, and alerts
- Prometheus is a time series database built using a HTTP pull model, with flexible queries and real-time alerting
- Examples: <http://claveblanca.ddns.net:3000>
- <https://github.com/ynov-campus-sophia/devops/tree/main/architecture/monitoring>

# Monitoring examples

