HIGHER SCHOOL OF ECONOMICS
NATIONAL RESEARCH UNIVERSITY

ISP RAS

Adaptation of the TOSCA standard model for the Kubernetes container environment

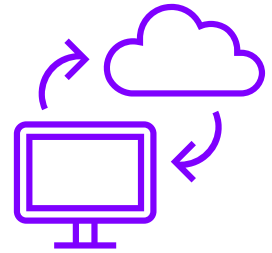
Authors:

Borisova Alexandra
Shvetcova Valeriya
Borisenko Oleg

Presenter:

Borisova Alexandra

Object domain

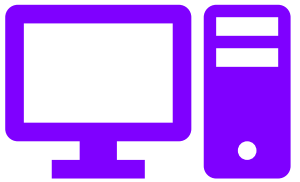


Software Developer

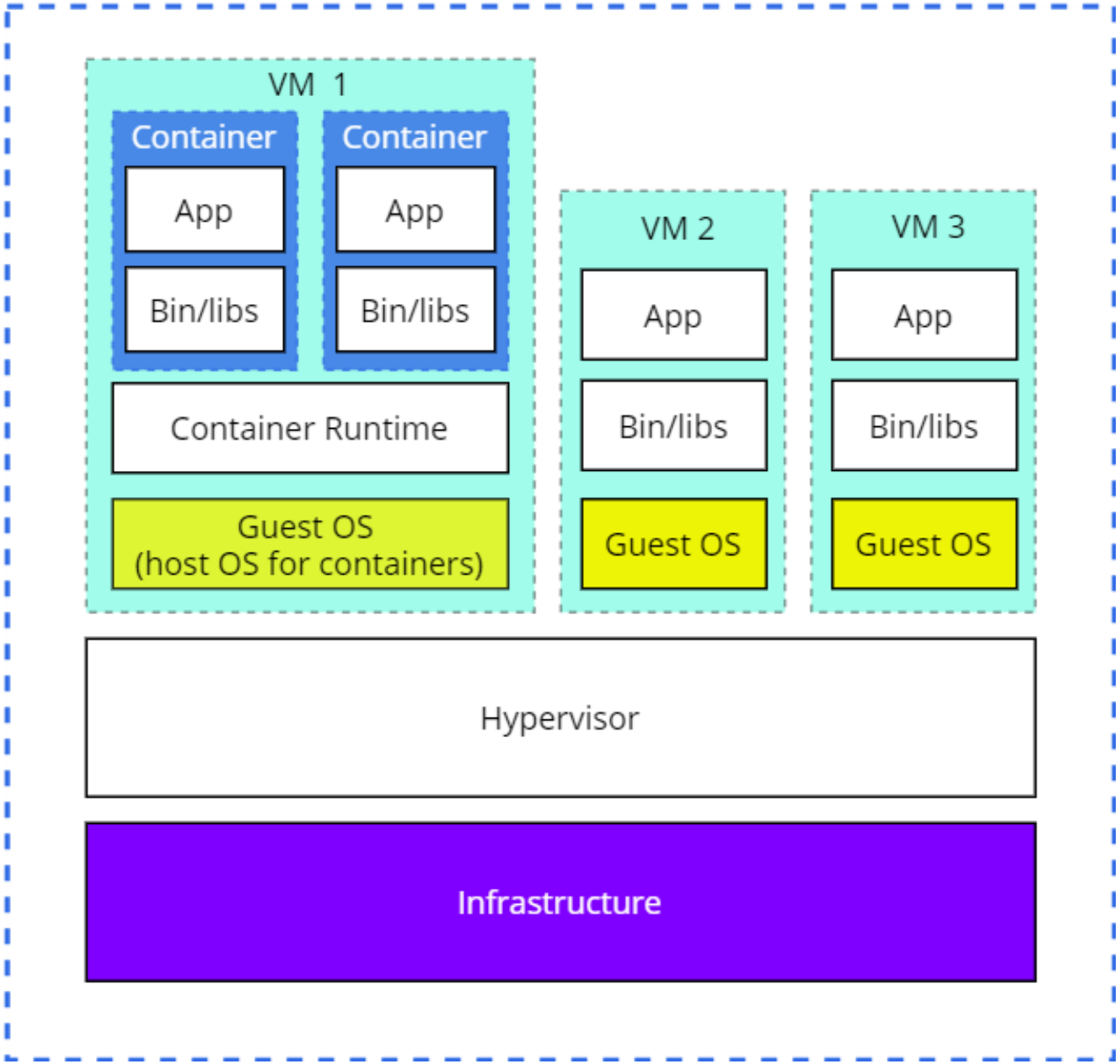
Software Application

deployment

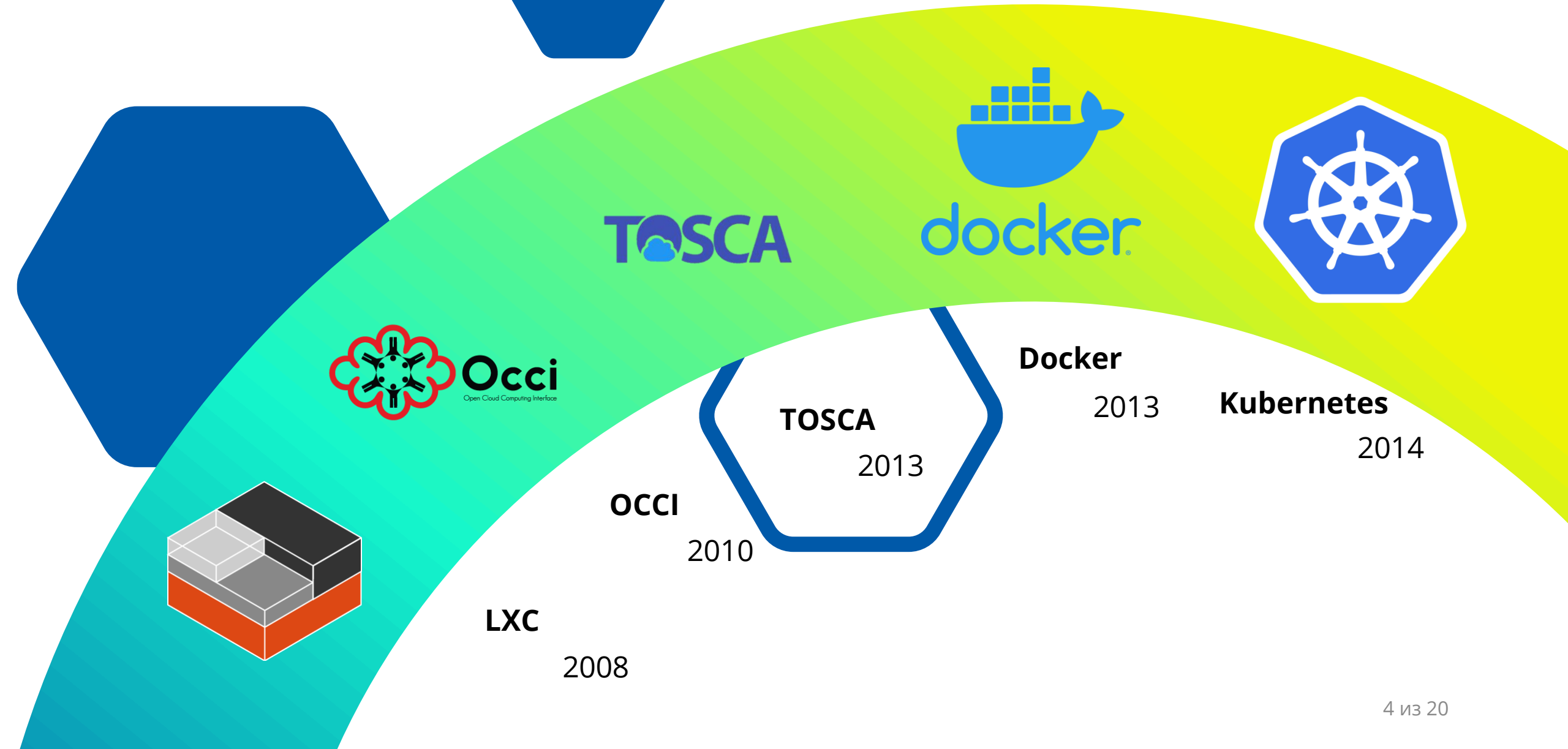
Physical or Virtual machines?

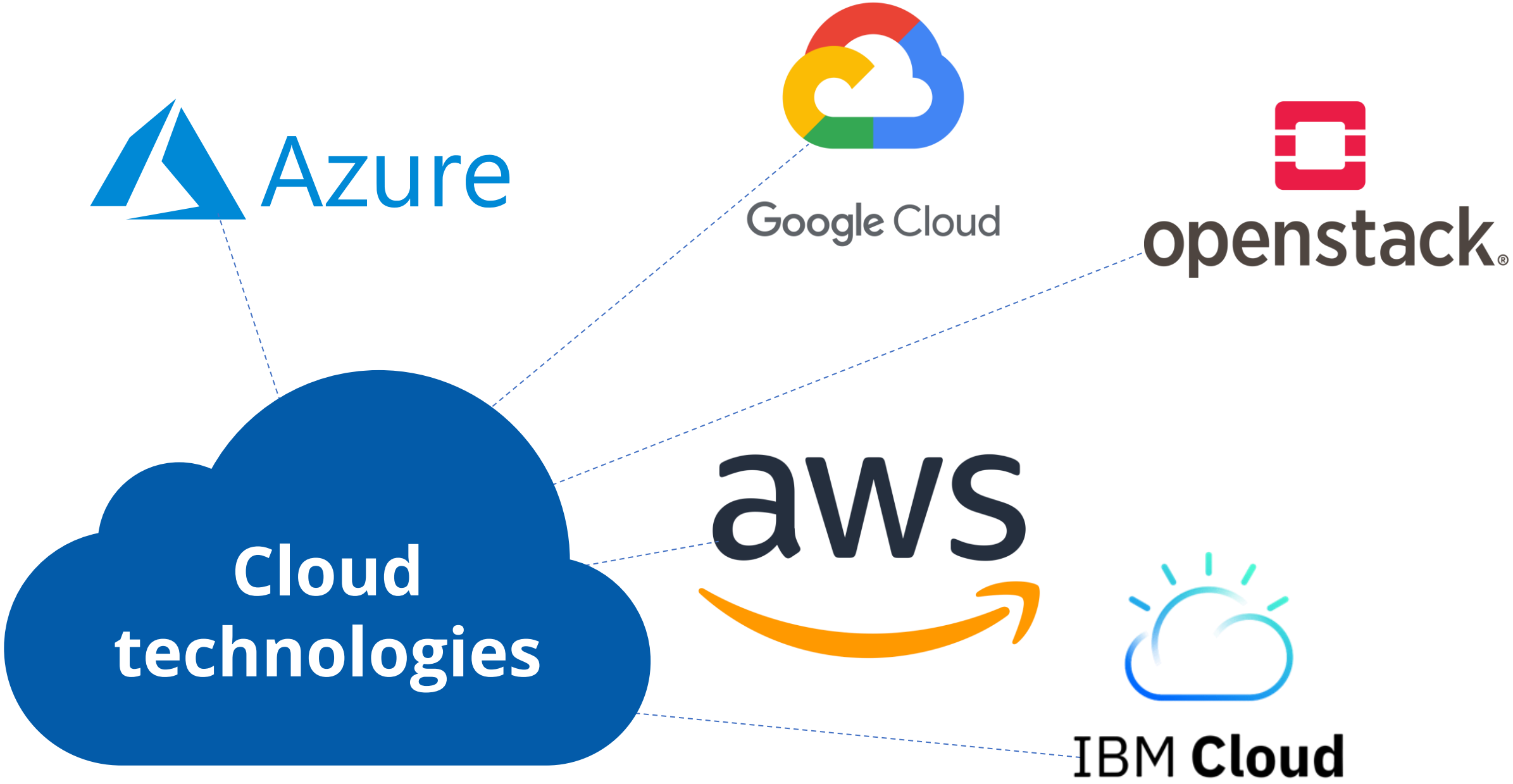


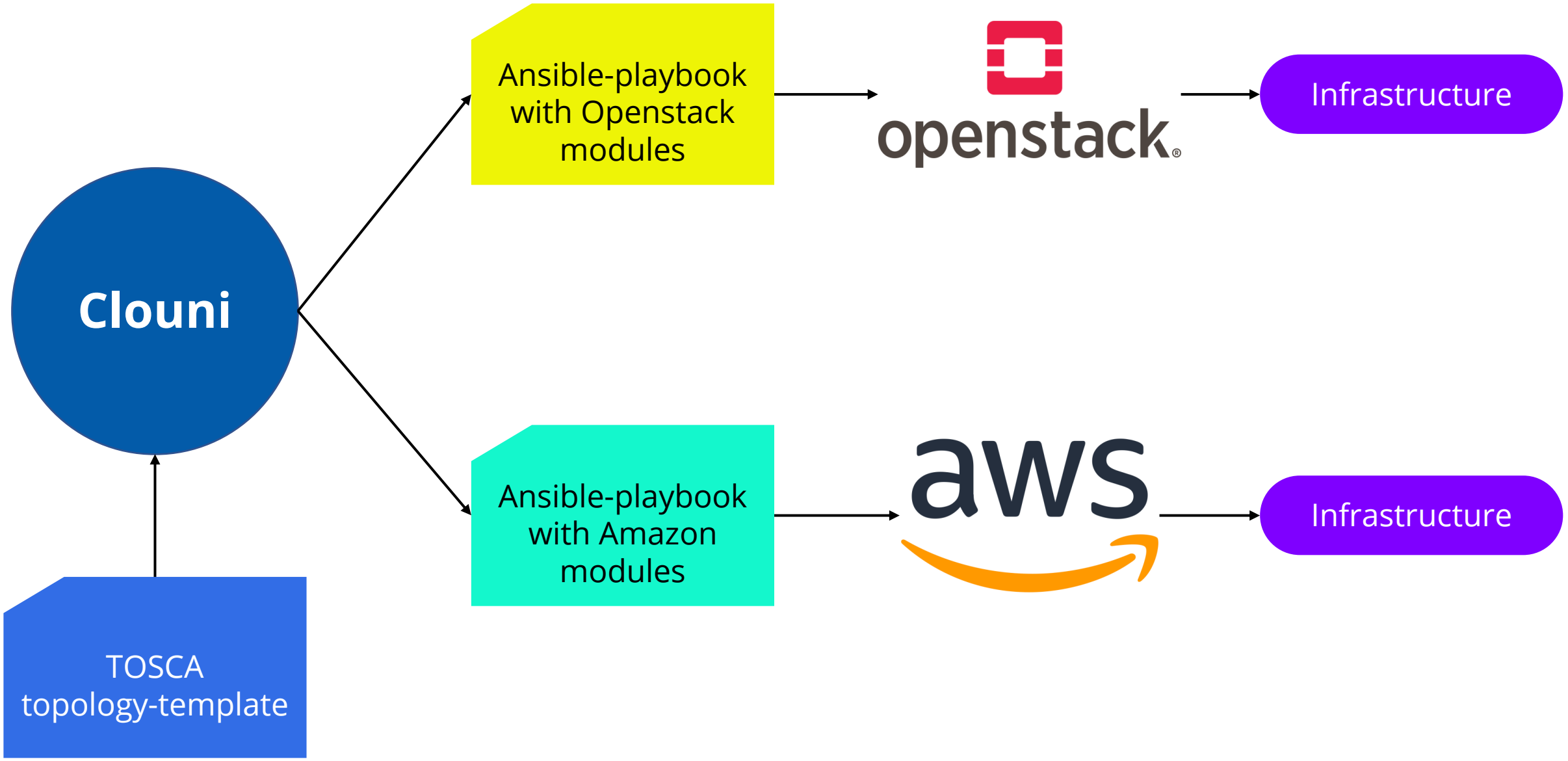
Containerization and Virtualization



History

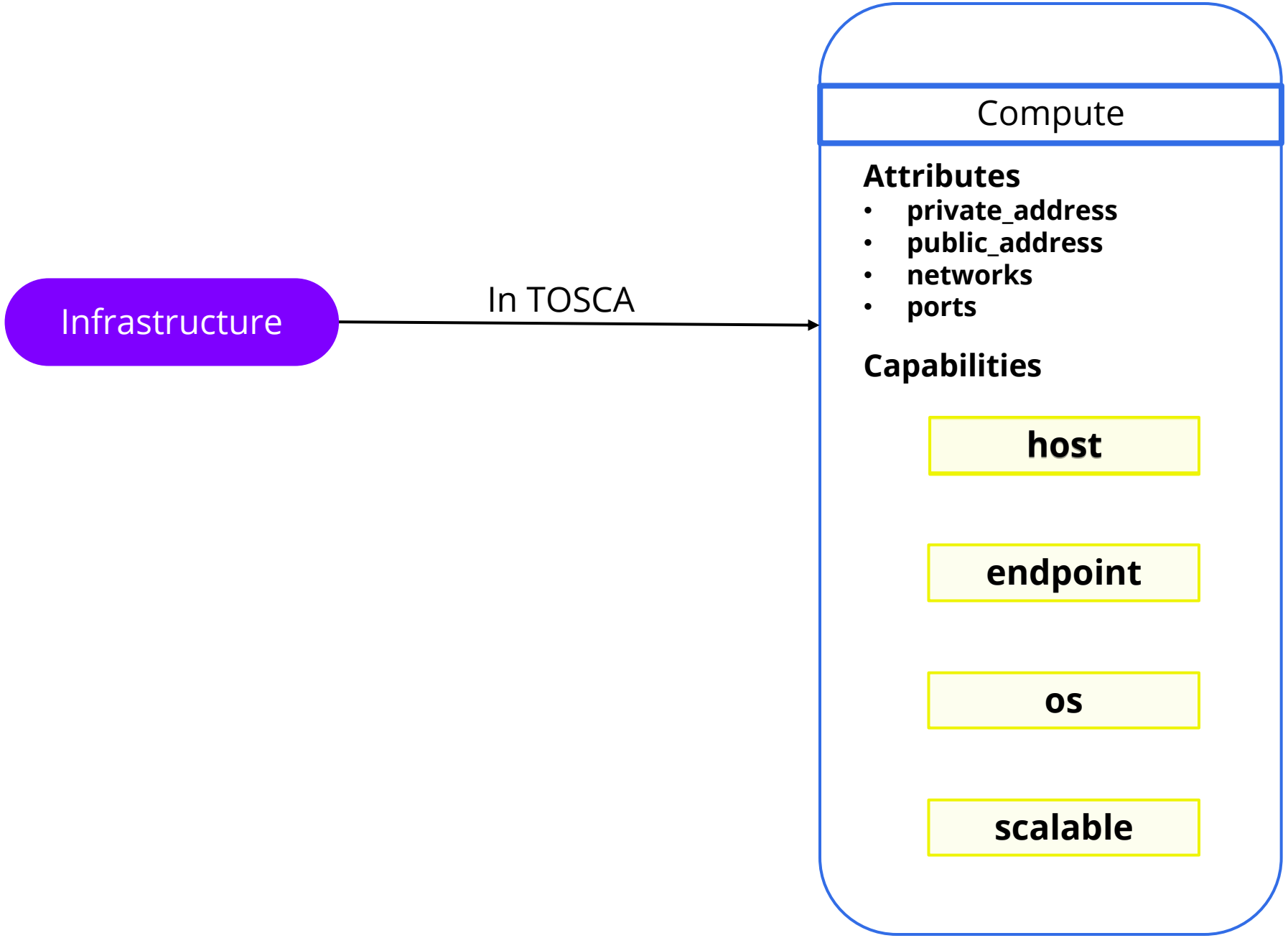

TOSCA
docker
Occi
Open Cloud Computing Interface
Docker
2013
Kubernetes
2014
TOSCA
2013
OCCI
2010
LXC
2008

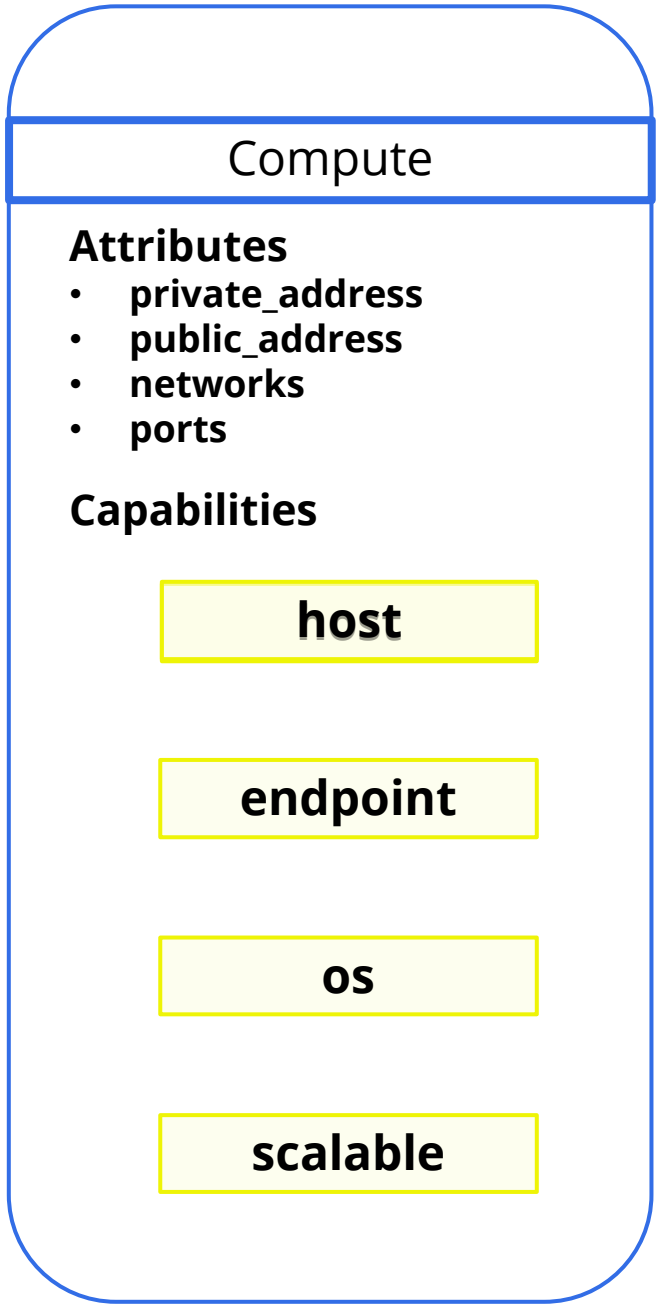

Azure
Google Cloud
openstack
Cloud technologies
aws
IBM Cloud


Clouni
TOSCA
topology-template
Ansible-playbook
with Openstack
modules
openstack
Infrastructure
Ansible-playbook
with Amazon
modules
aws
Infrastructure

Tasks for adaptation

1. Explore TOSCA Standard
2. Explore Kubernetes API
3. Add Kubernetes provider to Clouni
4. Deploy test infrastructure


Infrastructure
In TOSCA
Compute
Attributes
private_address
public_address
networks
ports
Capabilities
host
endpoint
os
scalable



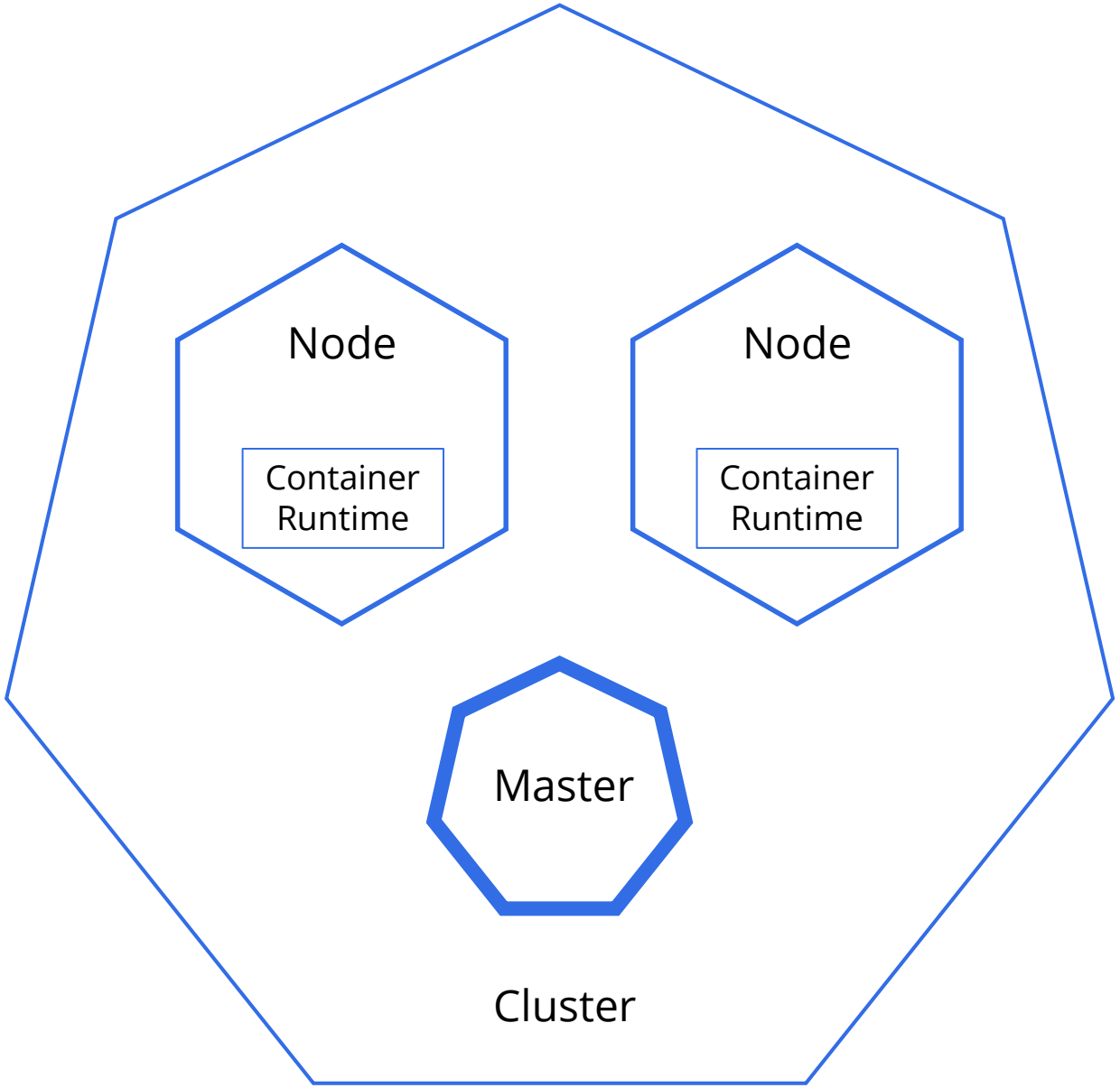
host - Indicates that the node can act as a container for (or a host for) one or more other declared Node Types

endpoint - This is the default TOSCA type that should be used or extended to define a specialized administrator endpoint capability.

os - This is the default TOSCA type that should be used to express an Operating System capability for a node.

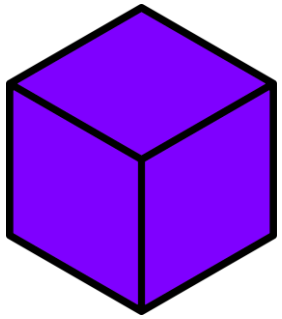
scalable - This is the default TOSCA type that should be used to express a scalability capability for a node

Kubernetes components

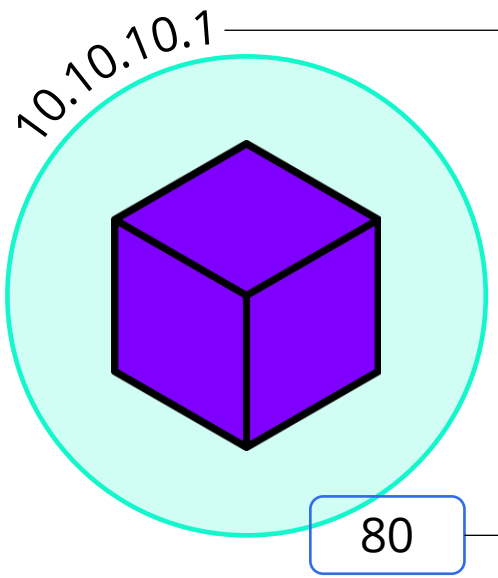


Kubernetes cluster consists of **Master node** and worker **Nodes.**
Master coordinates the cluster.
Nodes are the workers that run applications.
A Node is a VM or a physical computer.

Kubernetes components



Containerized Application with image



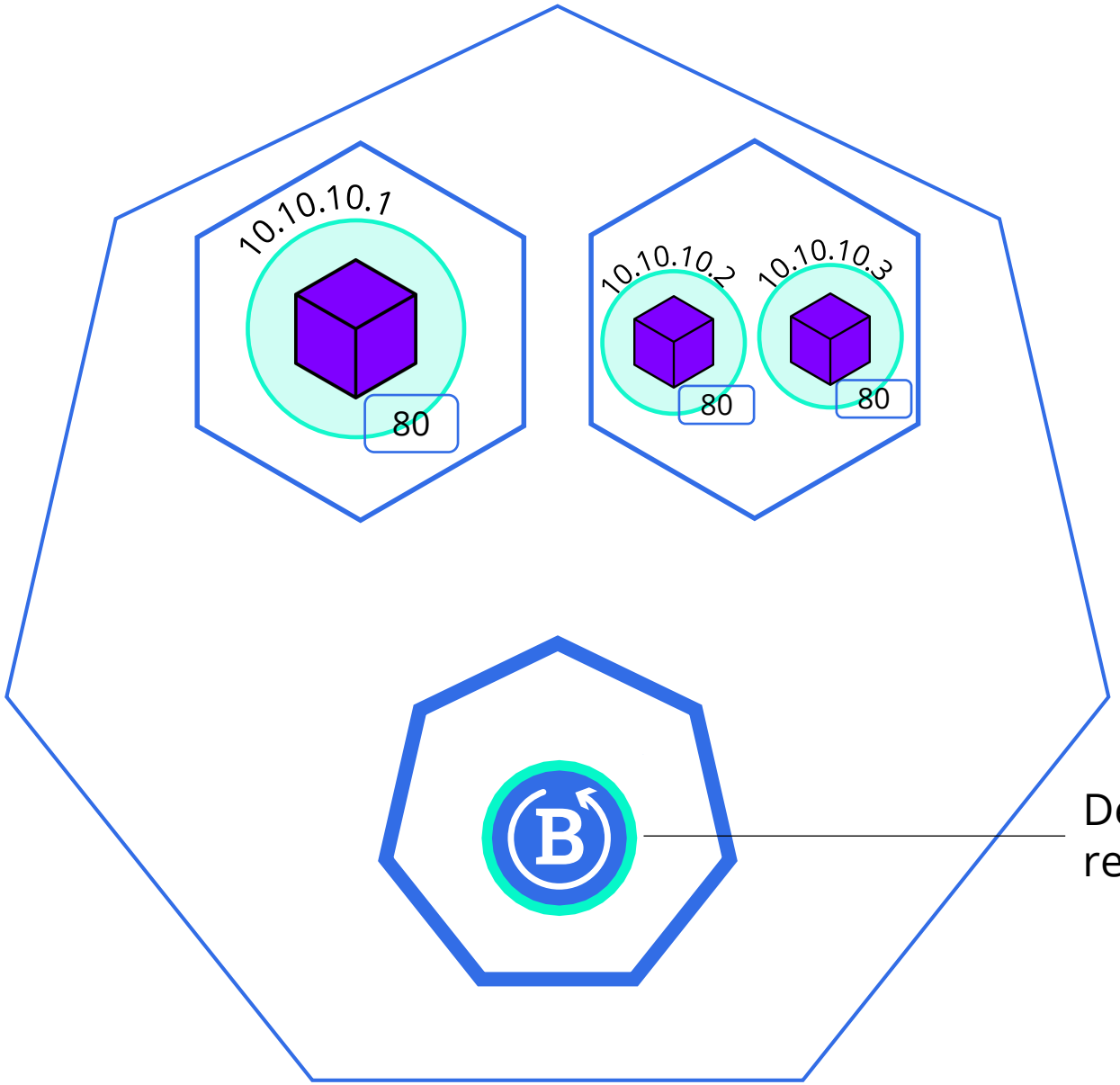
IP address

Pod

container port, protocol

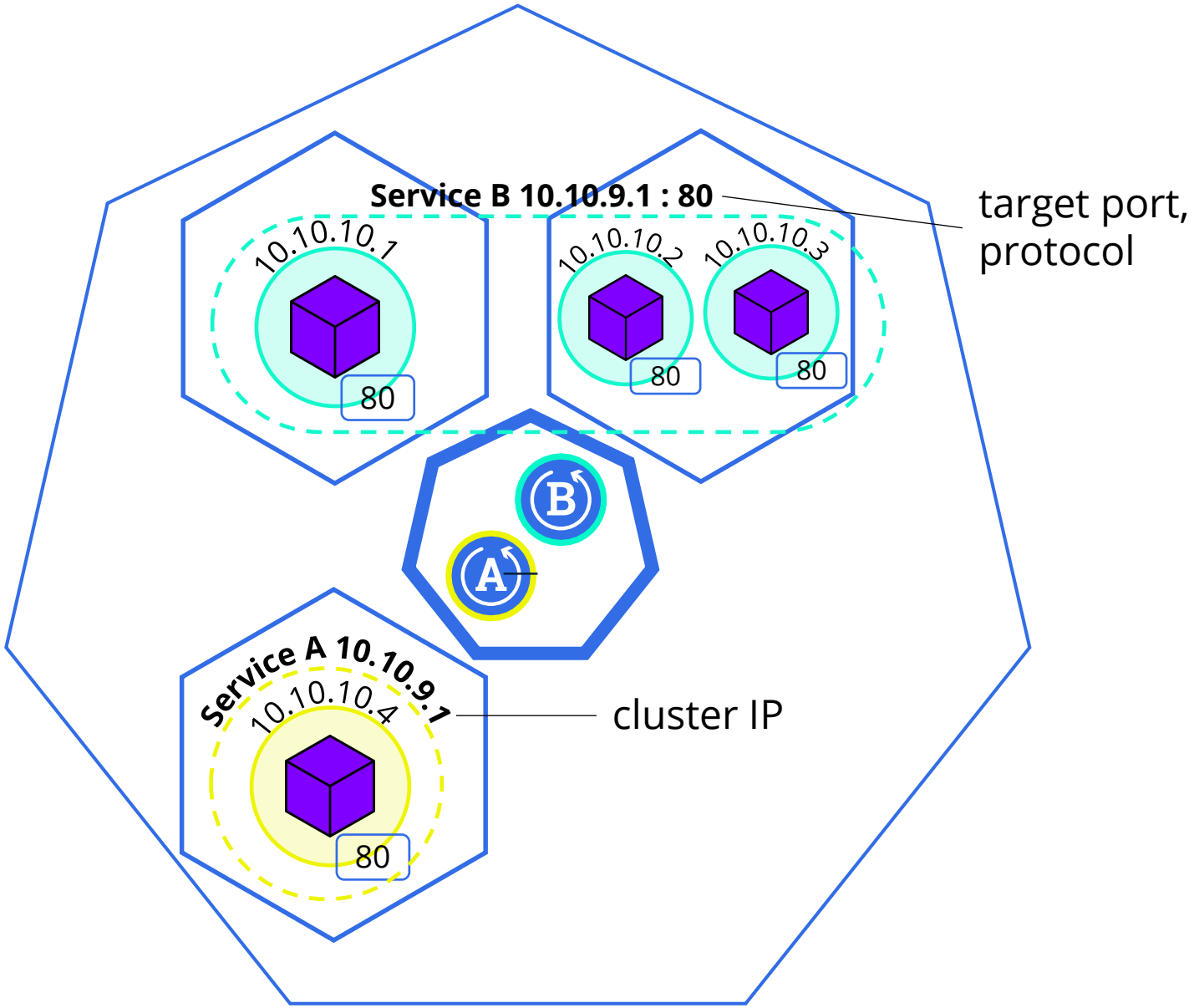
A **Pod** is a group of one or more application containers (such as Docker) and includes shared storage (volumes), IP address and information about how to run them

Kubernetes components



A **Deployment** is responsible for creating and updating instances of application.

Kubernetes components



A **Service** routes traffic across a set of Pods. Services are the abstraction that allow pods to die and replicate in Kubernetes without impacting your application.

Compute

Attributes

- **private_address**
- **public_address**
- networks
- ports

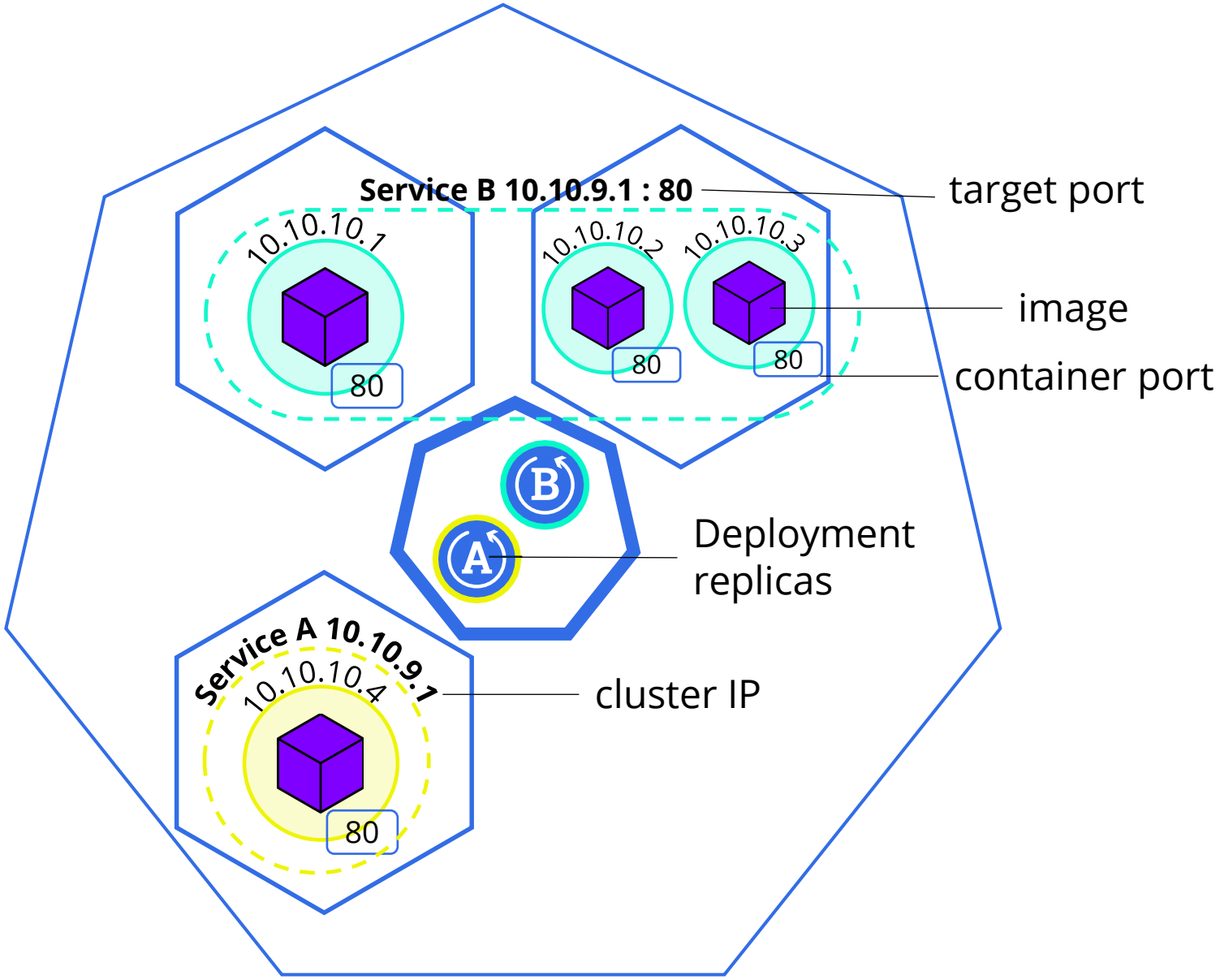
Capabilities

host

endpoint

os

scalable



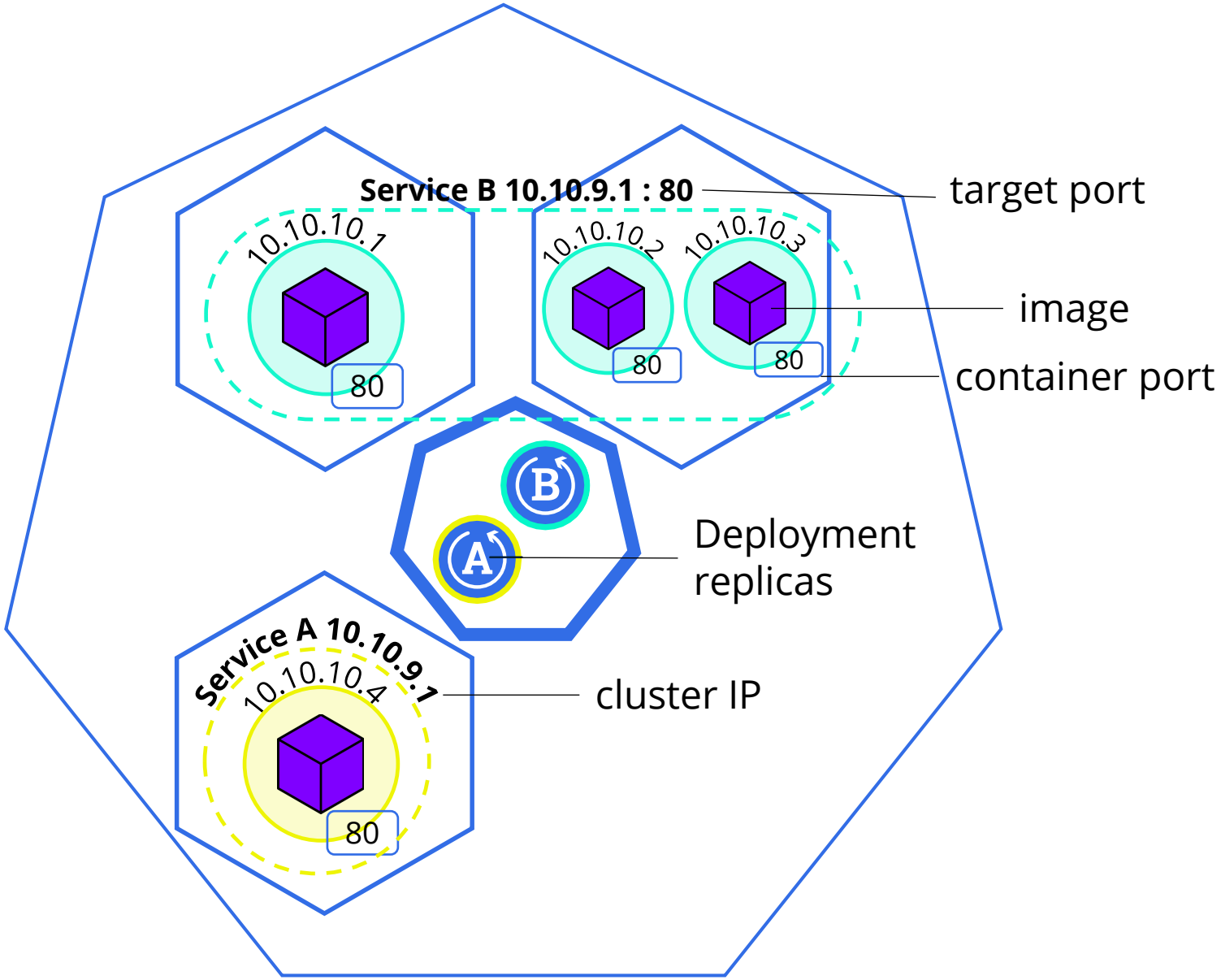
Compute

Attributes

- **private_address**
- **public_address**

Capabilities

host	num_cpus cpu_frequency disk_size mem_size
endpoint	protocol port secure url_path port_name network_name Initiator ports
os	architecture type distribution version
scalable	min_instances max_instances default



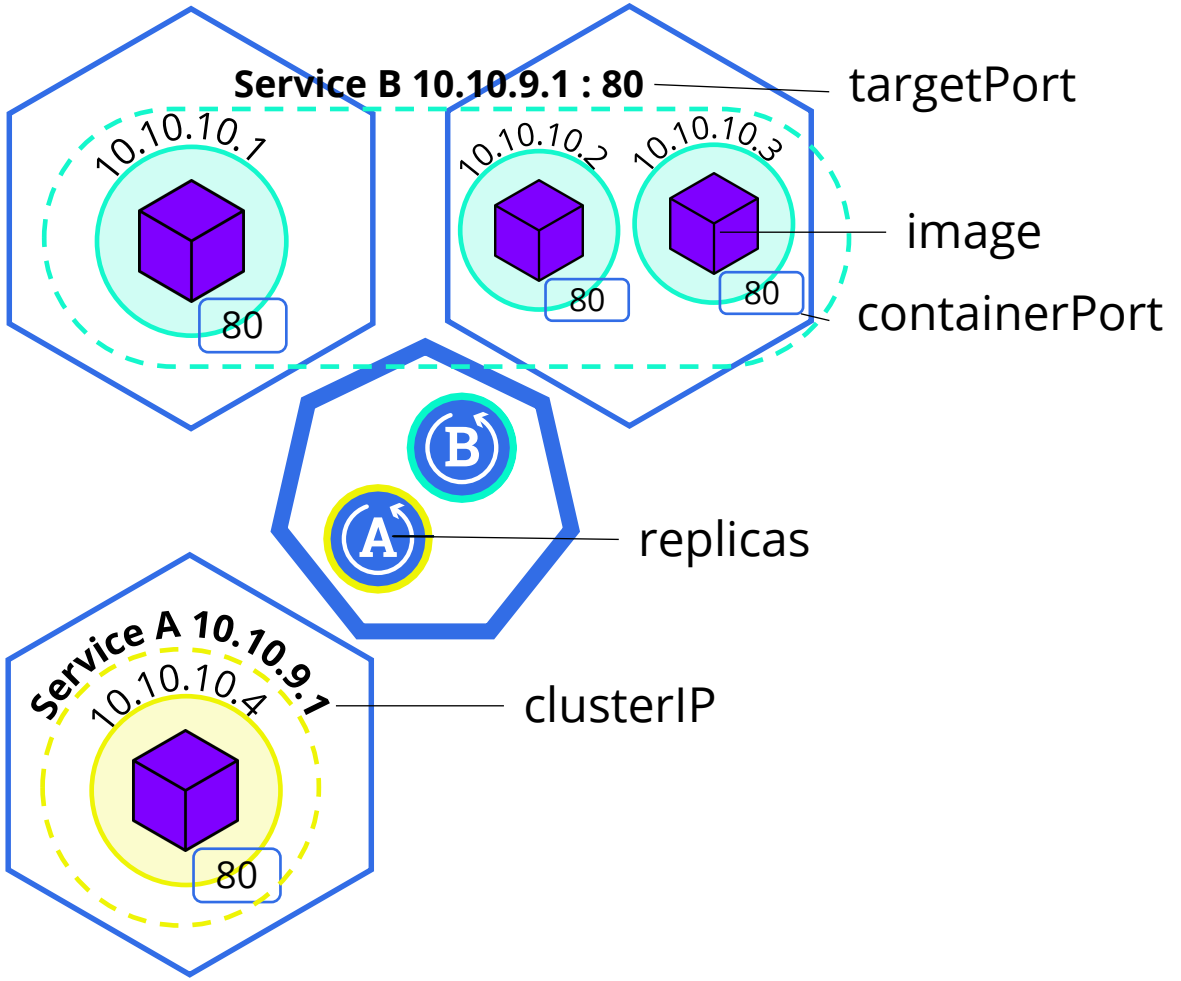
Compute

Attributes

- **private_address** — Service clusterIP &
- **public_address** — externalIPs

Capabilities

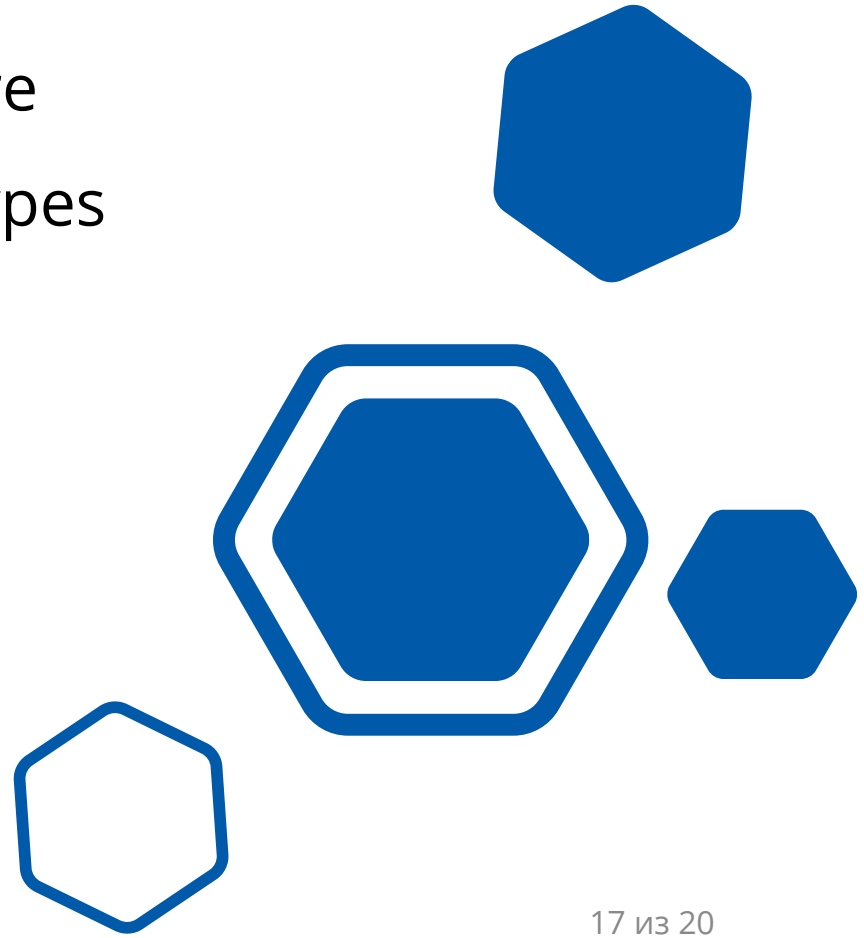
host	num_cpus mem_size	Pod args & limits:memory
endpoint	protocol port port_name	Pod protocol & containerPort & Service targetPort
os	type distribution version	Pod image
scalable	min_instances	Deployment replicas



After mapping

A few steps to add Kubernetes as a provider to Clouni :

1. Make Kubernetes-dependent TOSCA types
2. Make a YAML file in a special Clouni format, where described mapping between TOSCA normative types and Kubernetes-depended types
3. Add script, which translates TOSCA topology-template with Kubernetes-depended types to Kubernetes manifest in YAML.



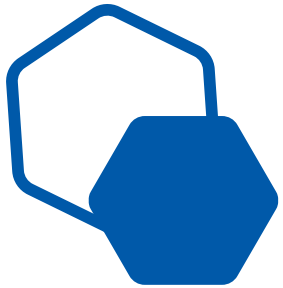
* YAML - is a human-readable data-serialization language

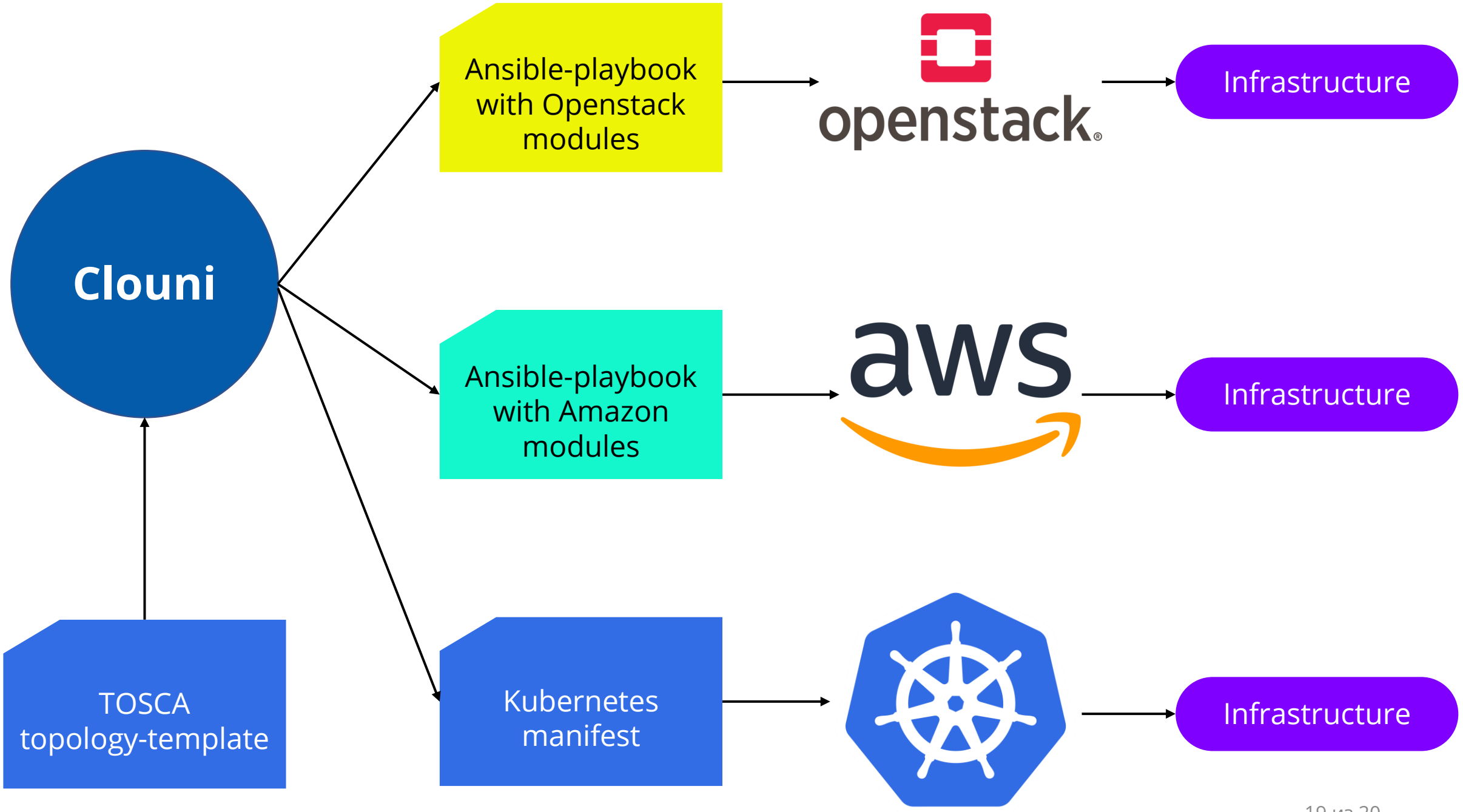
```yaml
tosca_definitions_version: tosca_simple_yaml_1_0

imports:
  - toscatranslator/common/TOSCA_definition_1_0.yaml

topology_template:
  node_templates:
    app:
      type: tosca.nodes.Compute
      attributes:
        private_address: 10.233.0.2
        public_address: 80.11.12.10
      capabilities:
        scalable:
          properties:
            min_instances: 3
        host:
          properties:
            num_cpus: 2
            mem_size: 25 MB
        os:
          properties:
            type: ubuntu
            distribution: xenial
        endpoint:
          properties:
            protocol: tcp
            port: 80
```



```yaml
apiersion: apps/v1
kind: Deployment
metadata:
  name: testapp-deployment
  lables:
    app: testapp
spec:
  replicas: 3
  selector:
    matchLabels:
      app: testapp
  template:
    metadata:
      lables:
        app: testapp
    spec:
      containers:
        - name: testapp-container
          image: ubuntu:xenial
          ports:
            - name: testapp-ports
              containerPort: 80
          resources:
            limits:
              memory: 25 Mi
          args:
            - -cpus
            - "2"
---
```

```yaml
apiVersion: v1
kind: Service
metadata:
  name: testapp-service
spec:
  clusterIP: 10.233.0.2
  externalIPs: [80.11.12.10]
  ports:
    - protocol: TCP
      port: 80
      targetPort: 80
  selector:
    app: testapp
```



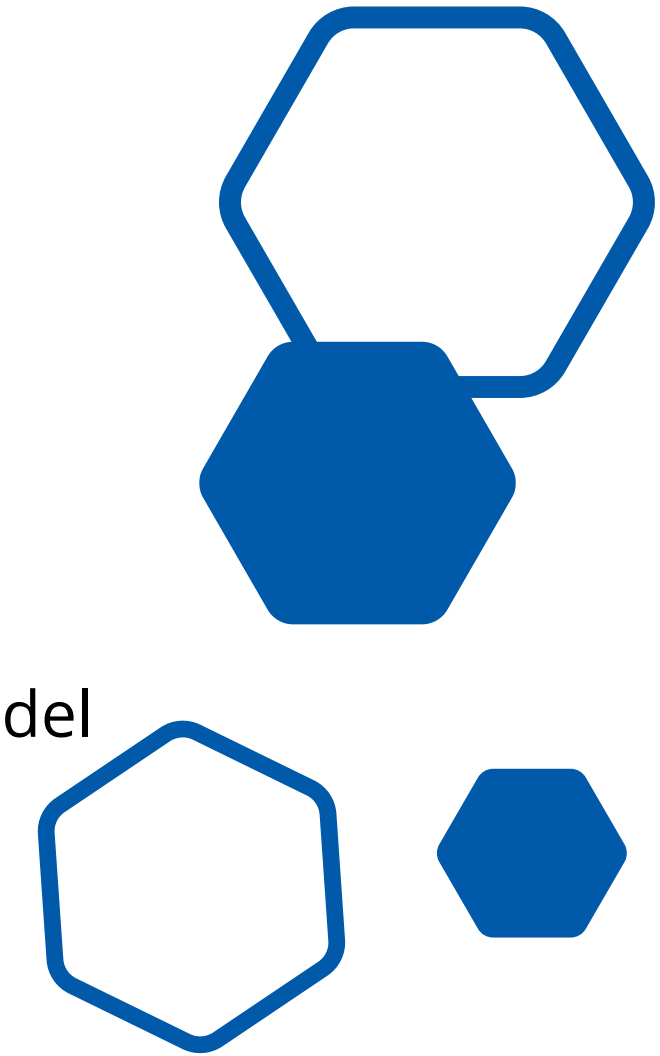

Clouni
Ansible-playbook with Openstack modules
openstack
Infrastructure
Ansible-playbook with Amazon modules
aws
Infrastructure
TOSCA topology-template
Kubernetes manifest
Infrastructure

Future work

- Explore TOSCA Standard
- Explore Kubernetes API
- Add Kubernetes provider to Clouni
- Deploy test infrastructure

<---------- we're here

- Add description of other nodes of the standard model
- Describe the scenarios for mapping nodes



Thank you for your attention!