Consortium GARR THE ITALIAN EDUCATION & RESEARCH NETWORK

A practical use case: deploying k8s on GARR Cloud with Juju

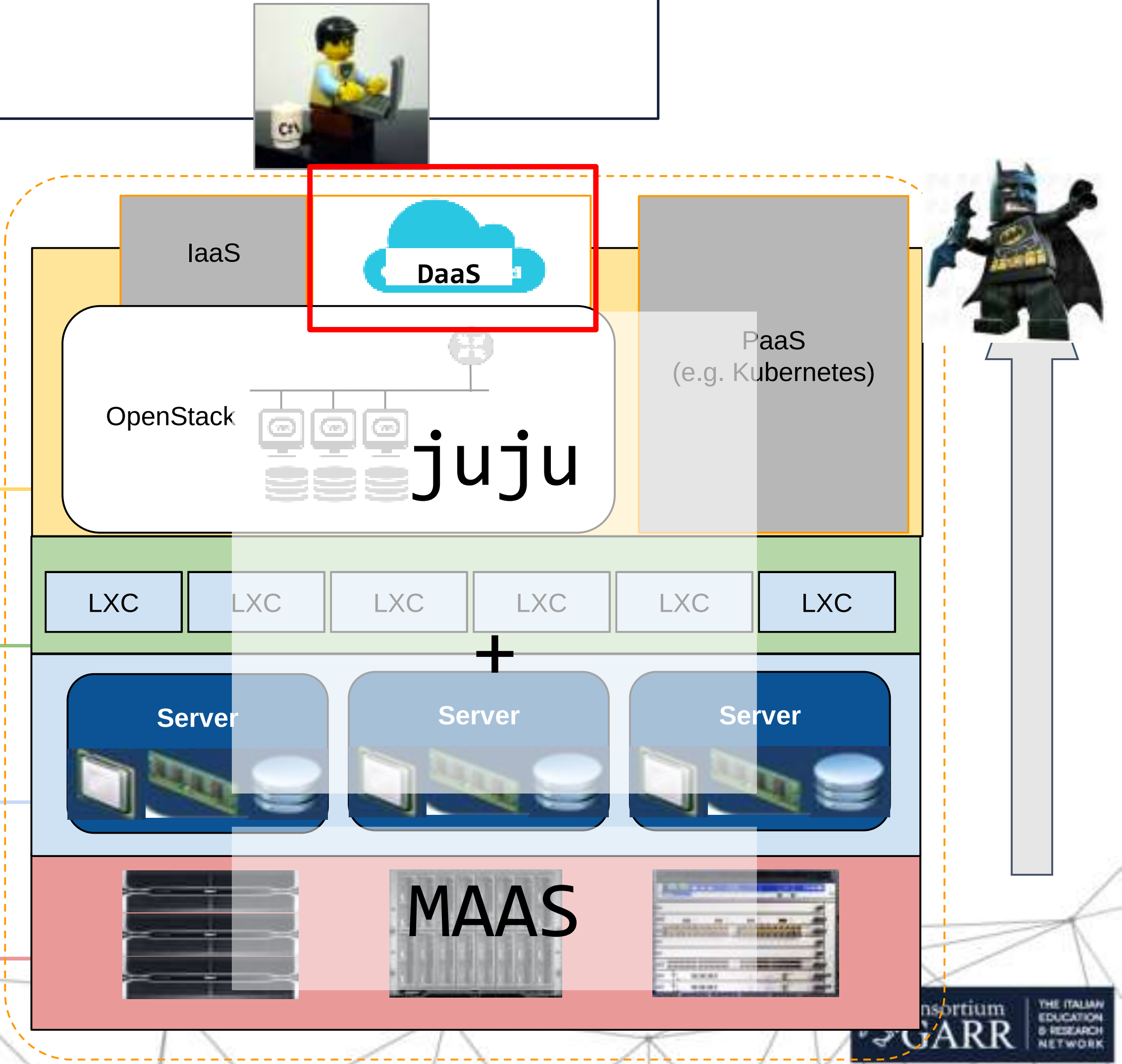
Matteo Di Fazio
Marco Lorini
Delia Passalacqua

Roma, 21/11/2019

Workshop EAP Connect

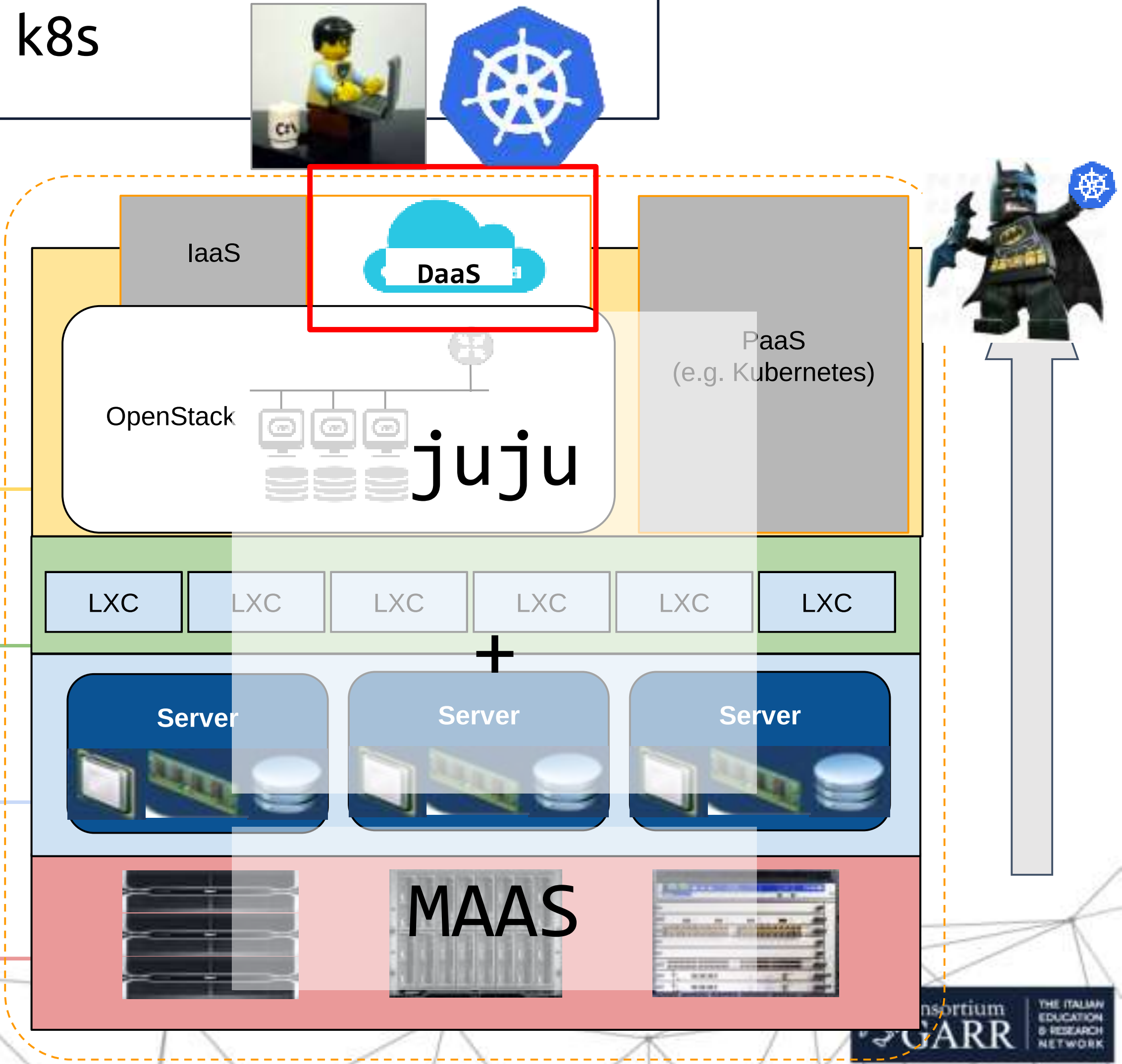
Use case: our architecture

1. Application Services
2. Infrastructure *Virtualization*
3. Operating System
4. Physical resources



Use case: our architecture + k8s

1. Application Services
2. Infrastructure *Virtualization*
3. Operating System
4. Physical resources



Use case

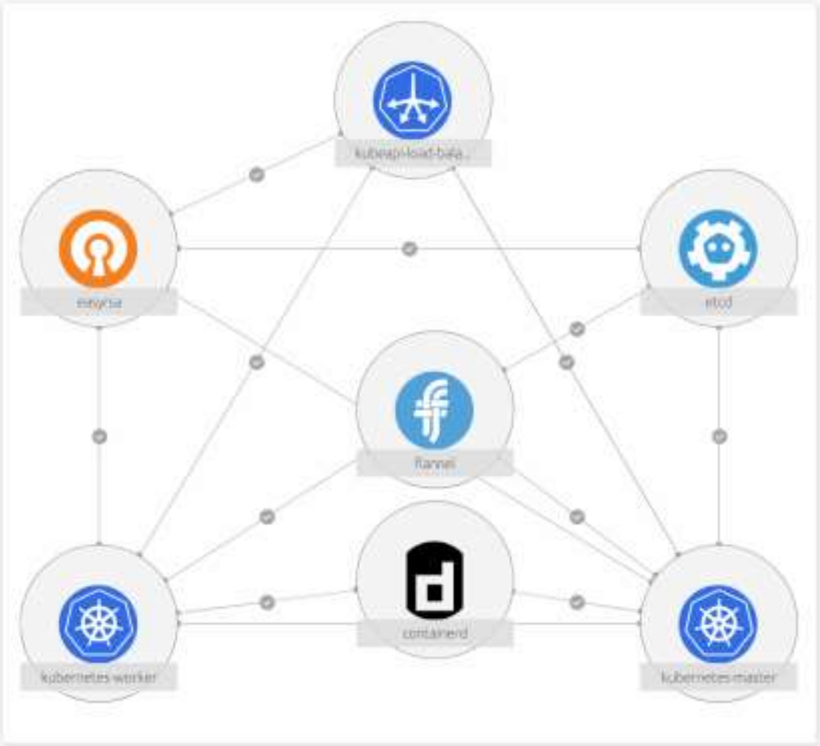
- Necessity to provide one k8s cluster per user
  - k8s multitenancy model does not capture some use cases
- k8s cluster fully integrated with Openstack native features (LBaaS, cinder volumes)

Our approach

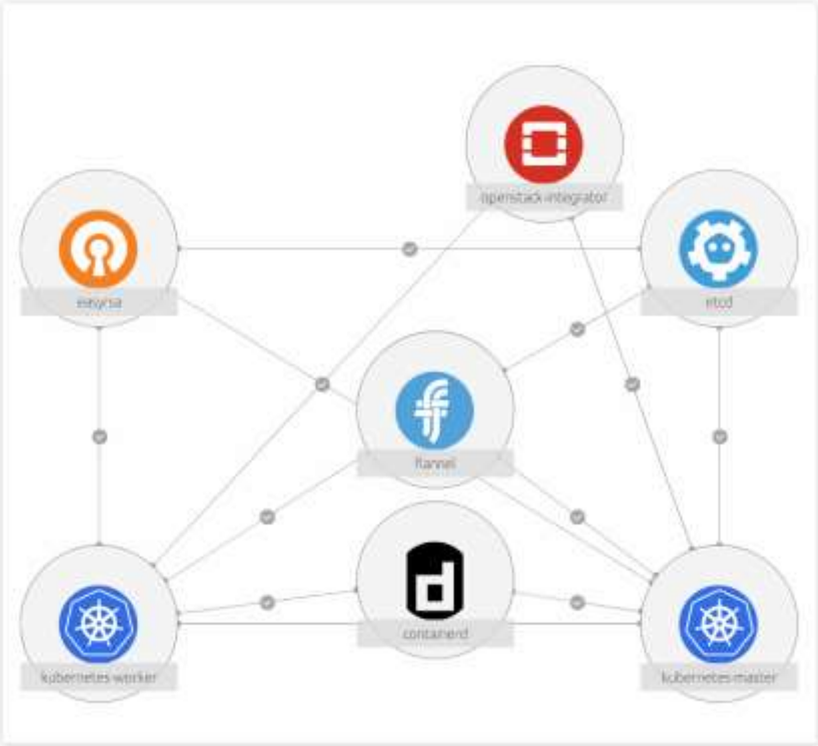
- Custom charm bundle
- Started from a preexisting k8s bundle (*The Charmed Distribution of Kubernetes* by Canonical)
- Modify k8s bundle integrating Openstack features

Comparing bundles

Original Kubernetes bundle



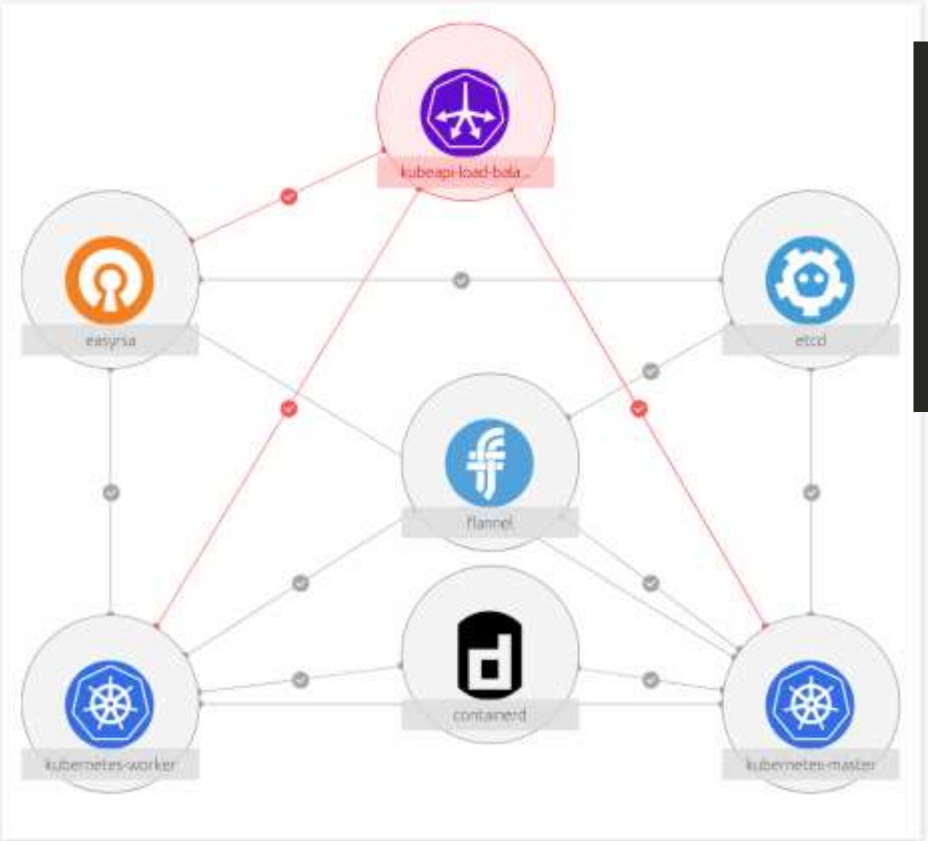
Our modified version

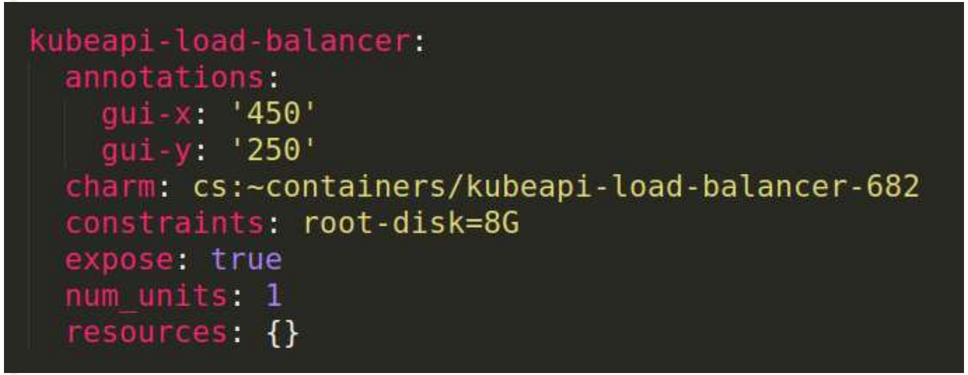


Why OpenStack Integrator?

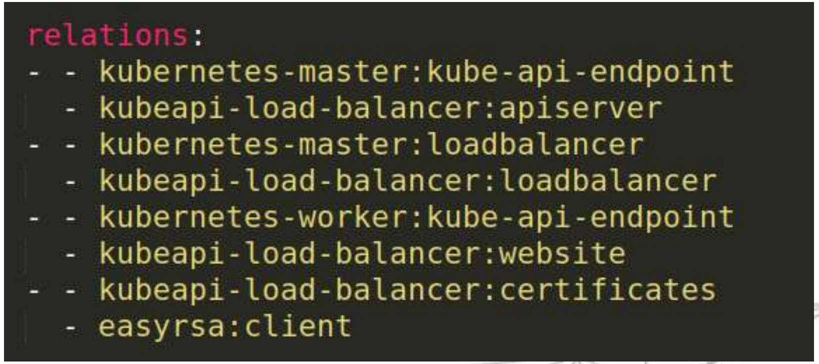
- Simplifies working with Charmed Kubernetes on OpenStack
- Allows use of Openstack load balancers for exposed k8s services
- Allows use of cinder storage volumes

Step 1: remove kubeapi loadbalancer



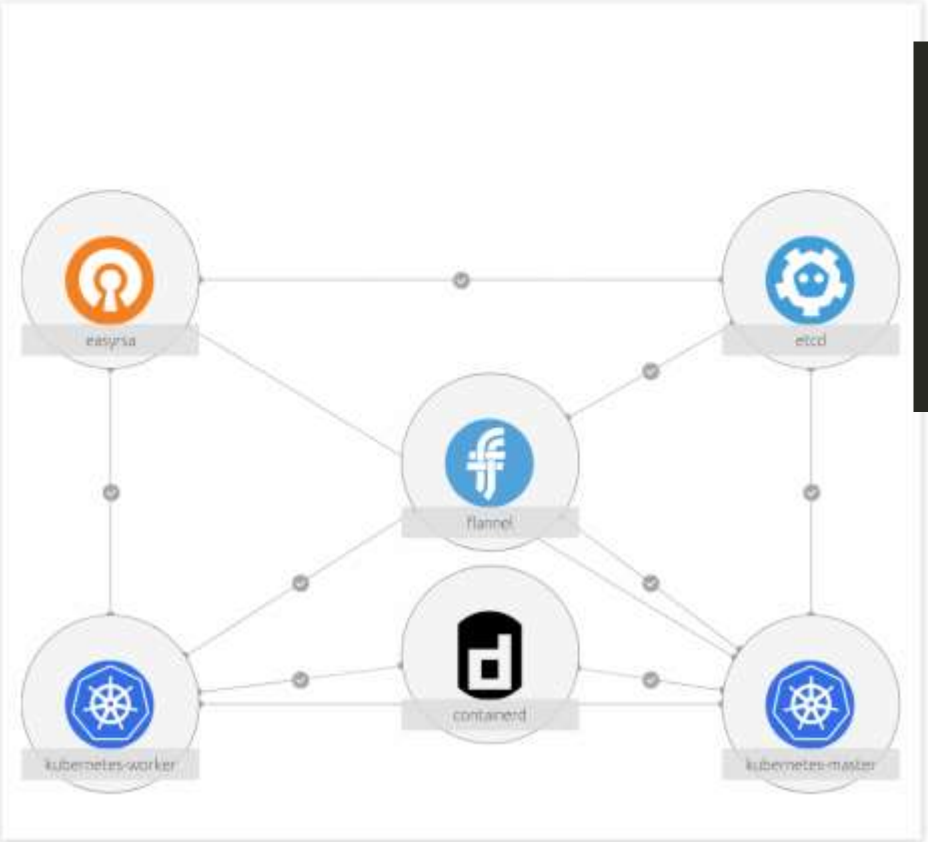


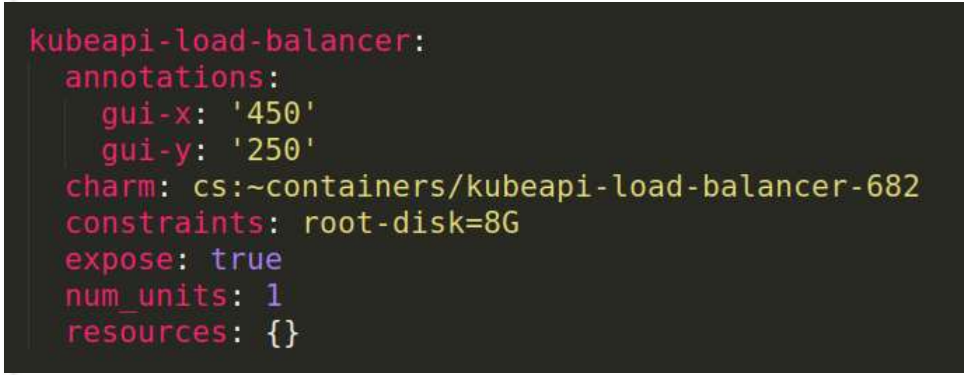
```yaml
kubeapi-load-balancer:
  annotations:
    gui-x: '450'
    gui-y: '250'
  charm: cs:~containers/kubeapi-load-balancer-682
  constraints: root-disk=8G
  expose: true
  num_units: 1
  resources: {}
```



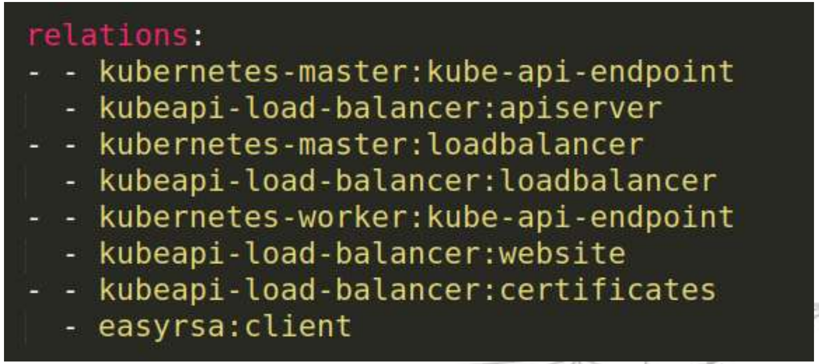
```yaml
relations:
- - kubernetes-master:kube-api-endpoint
  - kubeapi-load-balancer:apiserver
- - kubernetes-master:loadbalancer
  - kubeapi-load-balancer:loadbalancer
- - kubernetes-worker:kube-api-endpoint
  - kubeapi-load-balancer:website
- - kubeapi-load-balancer:certificates
  - easyrsa:client
```

Step 1: remove kubeapi loadbalancer



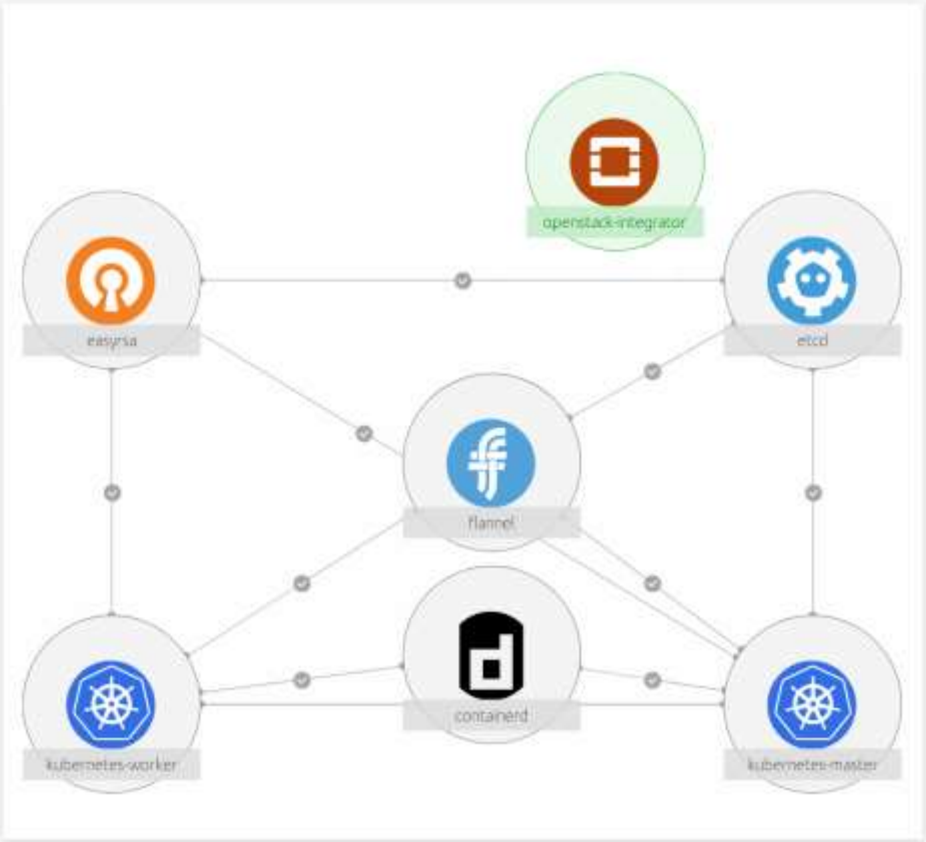


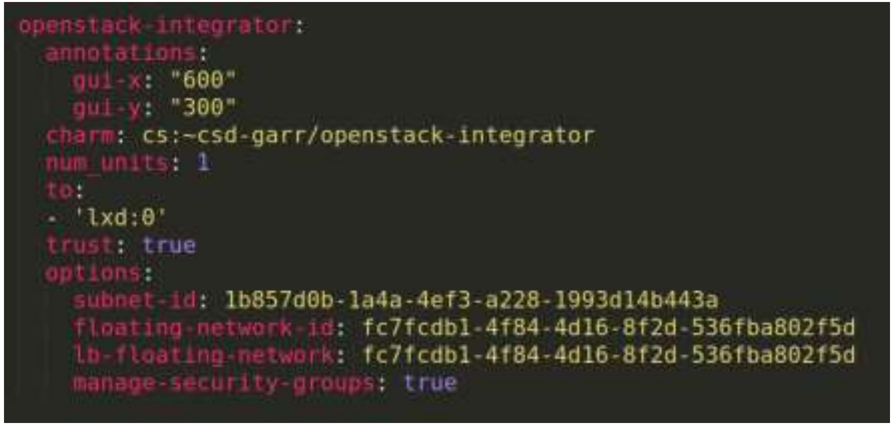
```
kubeapi-load-balancer:
  annotations:
    gui-x: '450'
    gui-y: '250'
  charm: cs:~containers/kubeapi-load-balancer-682
  constraints: root-disk=8G
  expose: true
  num_units: 1
  resources: {}
```



```
relations:
- - kubernetes-master:kube-api-endpoint
  - kubeapi-load-balancer:apiserver
- - kubernetes-master:loadbalancer
  - kubeapi-load-balancer:loadbalancer
- - kubernetes-worker:kube-api-endpoint
  - kubeapi-load-balancer:website
- - kubeapi-load-balancer:certificates
  - easyrsa:client
```

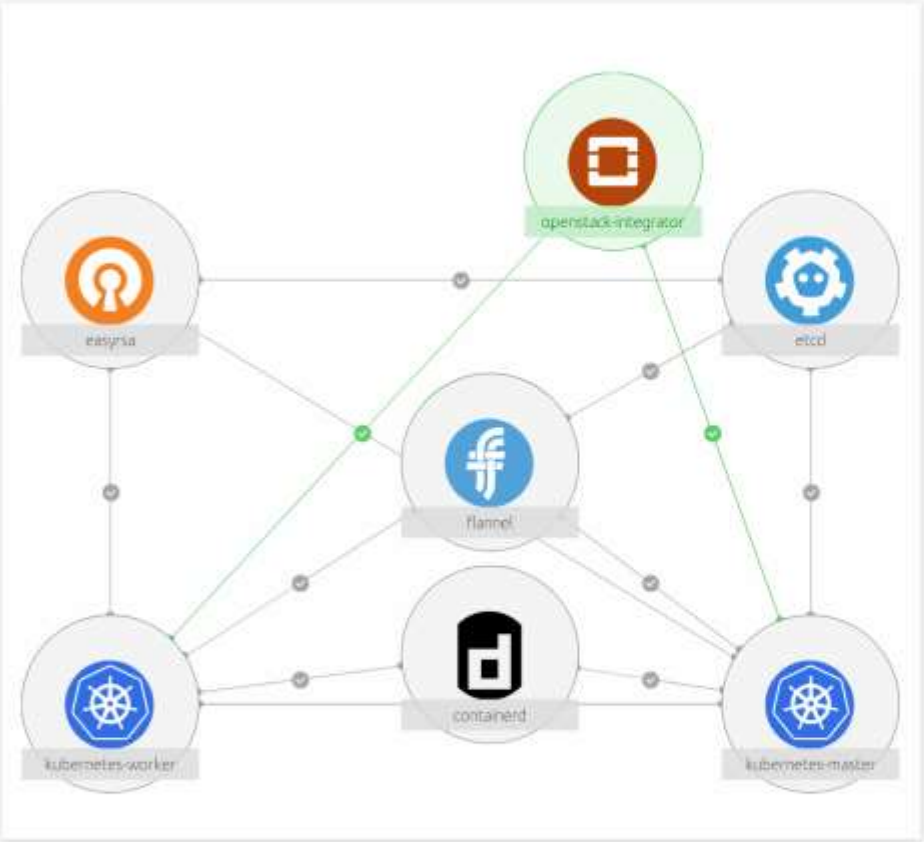
Step 2: add openstack integrator

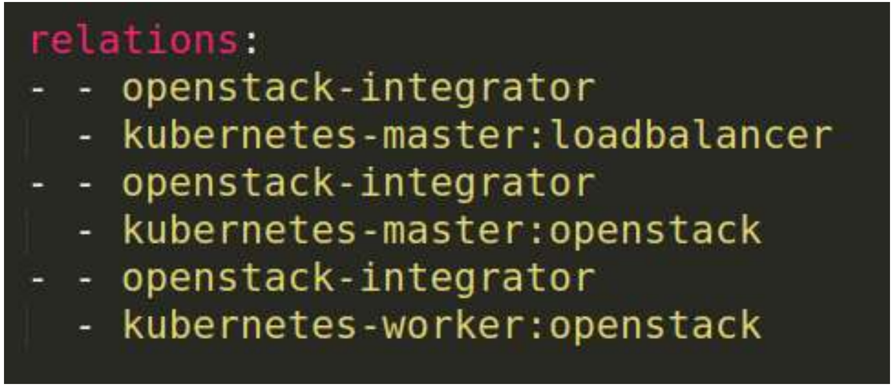




```yaml
openstack-integrator:
  annotations:
    gui-x: "600"
    gui-y: "300"
  charm: cs:~csd-garr/openstack-integrator
  num_units: 1
  to:
  - 'lxd:0'
  trust: true
  options:
    subnet-id: 1b857d0b-1a4a-4ef3-a228-1993d14b443a
    floating-network-id: fc7fcdb1-4f84-4d16-8f2d-536fba802f5d
    lb-floating-network: fc7fcdb1-4f84-4d16-8f2d-536fba802f5d
    manage-security-groups: true
```

Step 3: add openstack integrator's relations

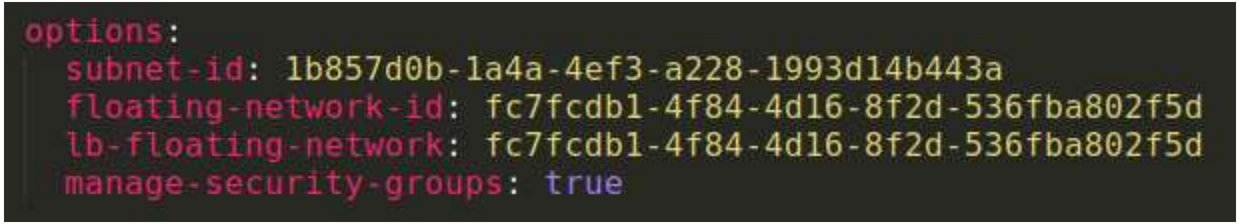




```
relations:
- - openstack-integrator
  - kubernetes-master:loadbalancer
- - openstack-integrator
  - kubernetes-master:openstack
- - openstack-integrator
  - kubernetes-worker:openstack
```

Openstack integrator's options

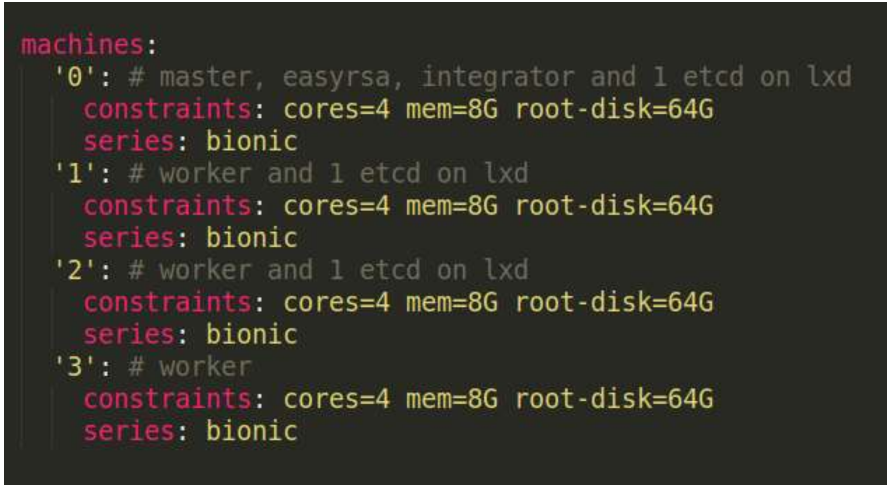
- subnet-id: the subnet in which load balancers should be created
- floating-network-id: the network in which floating ips related to k8s services should be created
- lb-floating-network: the network in which a floating ip for the k8s master load balancer should be created
- manage-security-groups: automatically create and manage security groups when a service is exposed



```
options:
  subnet-id: 1b857d0b-1a4a-4ef3-a228-1993d14b443a
  floating-network-id: fc7fcdb1-4f84-4d16-8f2d-536fba802f5d
  lb-floating-network: fc7fcdb1-4f84-4d16-8f2d-536fba802f5d
  manage-security-groups: true
```

Bundle optimization

- physical resources optimization:
  - 4 VMs (1 x Master, 3 x Worker)
  - lxd containers for openstack-integrator, easyrsa, etcd components
- architectural optimization: etcd HA (3 etcd units distributed on different nodes)



```
machines:
  '0': # master, easyrsa, integrator and 1 etcd on lxd
    constraints: cores=4 mem=8G root-disk=64G
    series: bionic
  '1': # worker and 1 etcd on lxd
    constraints: cores=4 mem=8G root-disk=64G
    series: bionic
  '2': # worker and 1 etcd on lxd
    constraints: cores=4 mem=8G root-disk=64G
    series: bionic
  '3': # worker
    constraints: cores=4 mem=8G root-disk=64G
    series: bionic
```

Troubleshooting

Problems we encountered:

1. bug in openstack integrator charm
   - security groups port setting command fails because "--format=yaml" option is unrecognized.
2. bug in cdk-addons snap package
   - storageclass object gets created repeatedly every 5 minutes

Solutions:

1. Submitted patch to the official repository
2. Known and solved bug, fix ready for the next release

Thanks for your attention!