ubuntu

Deploying full prometheus stacks via juju charms, *lightning'ly*

~ JuanJo Ciarlante - aka *jjo* ~
@xjjo
jjo@canonical.com

~ promcon.io - 2016-08-25 ~

CANONICAL

ubuntu

About me

- Working at Canonical Ltd as Cloud Reliability Engineer
  - since 2012
- Worked at Google Switzerland as Site Reliability Engineer
  - between 2007-2012
- FOSS contributor
  - since 1995

  So ... you don't like the ':' used for linux interfaces aliases ?

  => blame the guy at stage

ubuntu

Juju in a few words

- service orchestration tool
- multi-provider:
  - clouds: AWS, GCE, Azure, Rackspace, Openstack
  - metals: MaaS
  - containers: LXC or LXD (on v2)
- services get deployed and setup by charms:
  - set of scripts invoked at different stages

Juju orchestration - webapp deploy

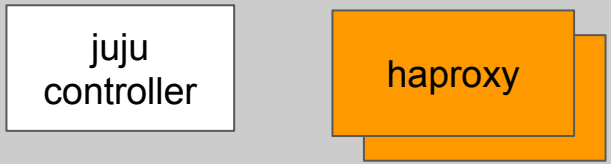
ubuntu

juju
controller

```
$ juju bootstrap
```

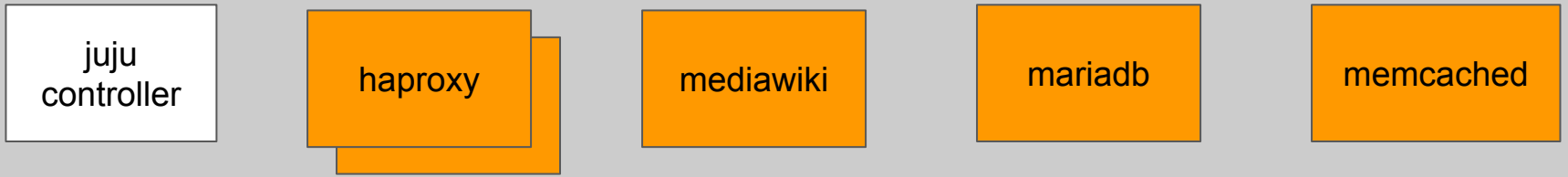
CANONICAL

Juju orchestration - webapp deploy



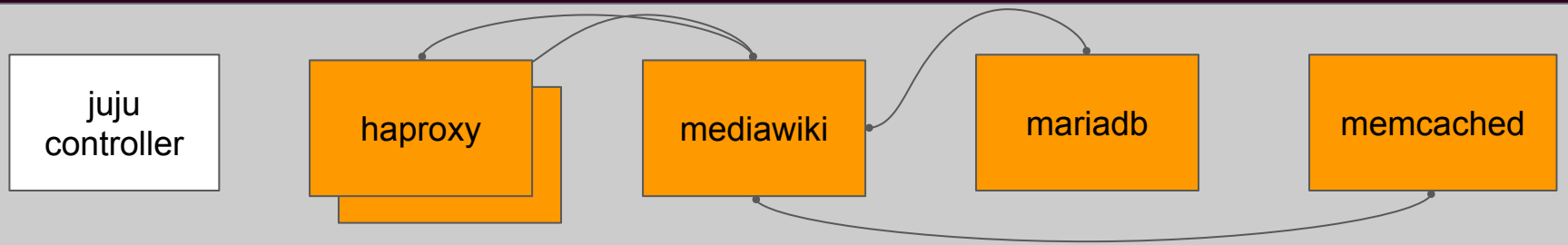

```
$ juju bootstrap
$ juju deploy -n2 haproxy
```

Juju orchestration - webapp deploy



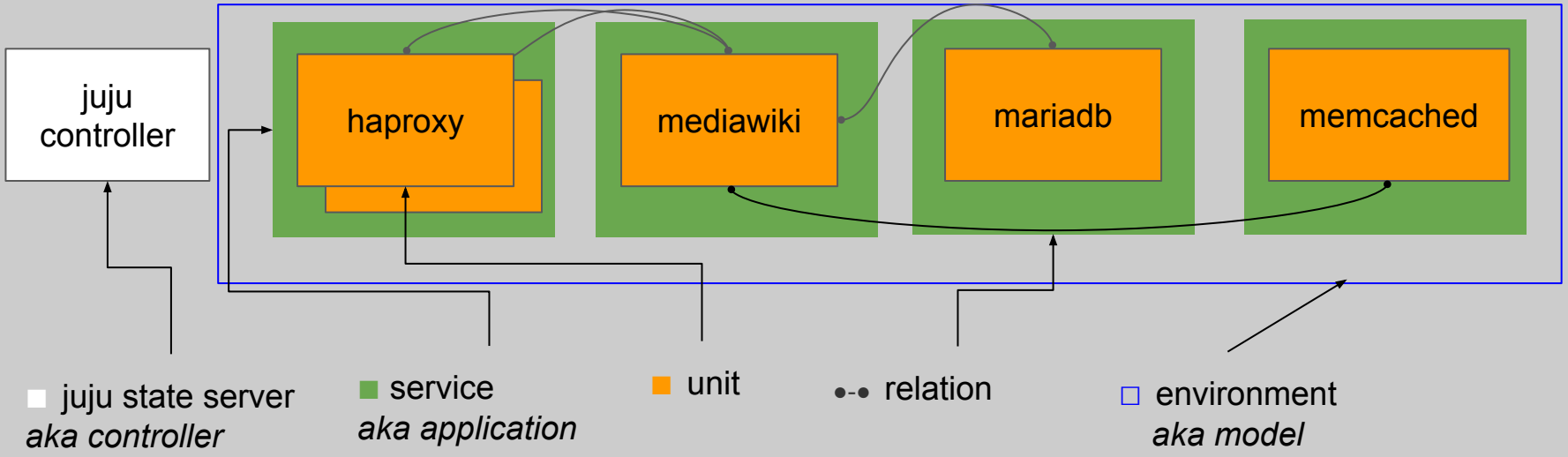

```
$ juju bootstrap
$ juju deploy -n2 haproxy
$ juju deploy mediawiki
$ juju deploy mariadb
$ juju deploy memcached
```

Juju orchestration - webapp deploy




```
$ juju deploy -n2 haproxy
$ juju deploy mediawiki
$ juju deploy mariadb
$ juju deploy memcached
$ juju add-relation haproxy mediawiki
$ juju add-relation mediawiki:db mariadb:db
$ juju add-relation mediawiki memcached
```

Juju orchestration - glossary~ish



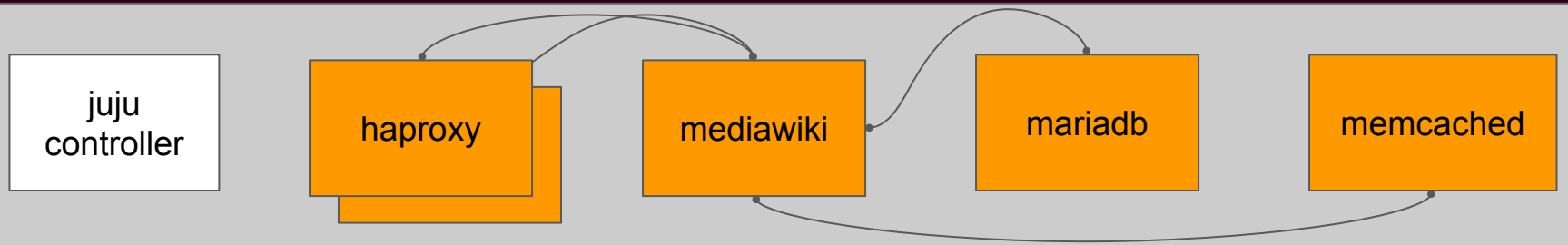
- ***charm*** code is specified by URL, then locally **run** at **units**
- service stacks can be specified as a ***juju bundle*** file (e.g. wiki-bundle.yaml)

ubuntu

What we use prometheus for

- **Infrastructure (phy nodes and overcloud services)**
  - /metrics: snmp exp., pushgateway, collectd exp., telegraf
  - maint. by: ops
- **Openstack (and storage-layer) metrics**
  - /metrics: collectd exp., pushgateway, telegraf, (wip) mtail
  - maint. by: ops
- **Online services**
  - /metrics: telegraf
  - maint. by: devs, ops

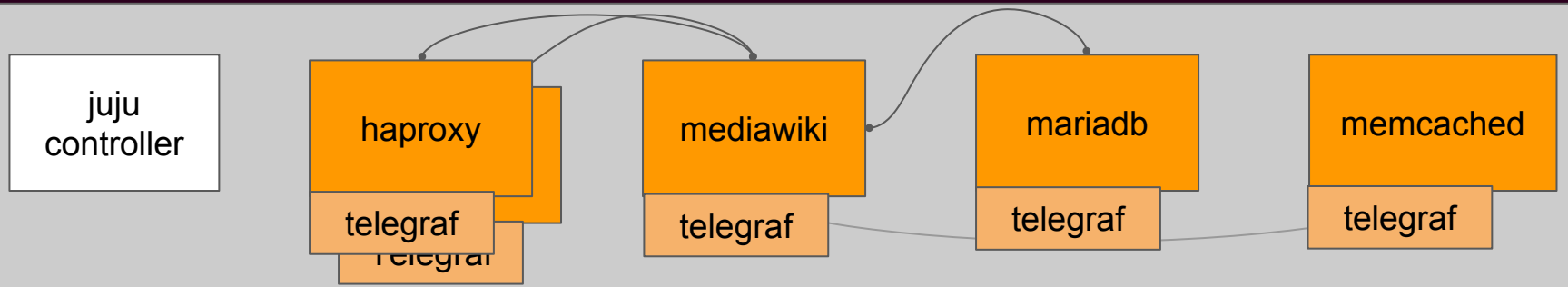
How we deploy prometheus




```
$ juju add-relation haproxy mediawiki
$ juju add-relation mediawiki:db mariadb:db
$ juju add-relation mediawiki memcached
$ juju deploy prometheus
$ juju deploy telegraf  # <- subordinate charm => no unit created
```

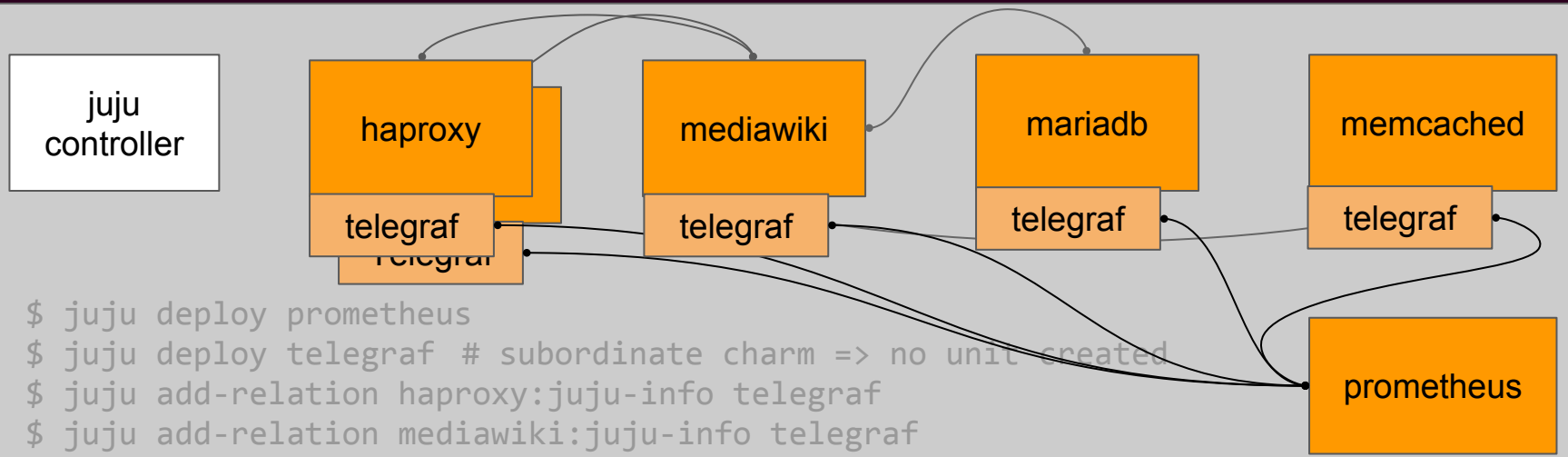
prometheus

How we deploy prometheus




```
$ juju deploy prometheus
$ juju deploy telegraf  # subordinate charm => no unit created
$ juju add-relation haproxy:juju-info telegraf
$ juju add-relation mediawiki:juju-info telegraf
$ juju add-relation mariadb:juju-info telegraf
$ juju add-relation memcached:juju-info telegraf
```

How we deploy prometheus

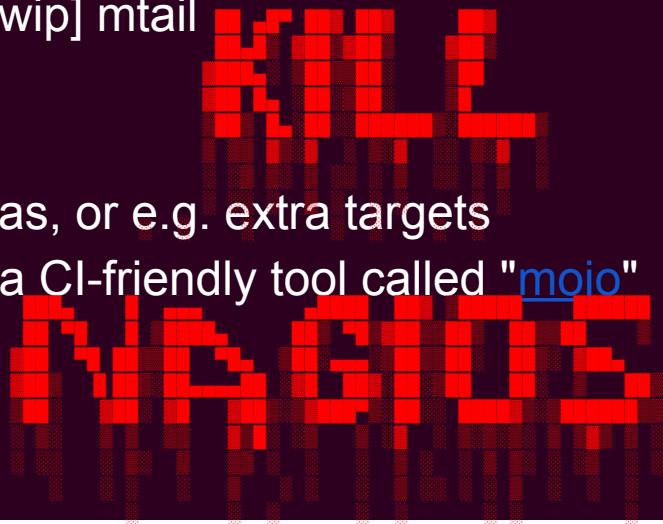



```
$ juju deploy prometheus
$ juju deploy telegraf  # subordinate charm => no unit created
$ juju add-relation haproxy:juju-info telegraf
$ juju add-relation mediawiki:juju-info telegraf
$ juju add-relation mariadb:juju-info telegraf
$ juju add-relation memcached:juju-info telegraf
$ juju add-relation prometheus telegraf
```

How we deploy prometheus (wrap-up)

ubuntu

- all these charms and required deb packages are publicly available as FOSS
- charms we developed, and use:
  - prometheus
  - telegraf, snmp-exporter, collectd-exporter, [wip] mtail
  - grafana
  - [wip] alertmanager
- most charms allow specifying extra config stanzas, or e.g. extra targets
- our deploys are specified by *bundles*, driven by a CI-friendly tool called "mojo"
- more details w/ examples, next week at:
  - http://github.com/jjo/promcon



BootStack-PS4.5/perf-irate

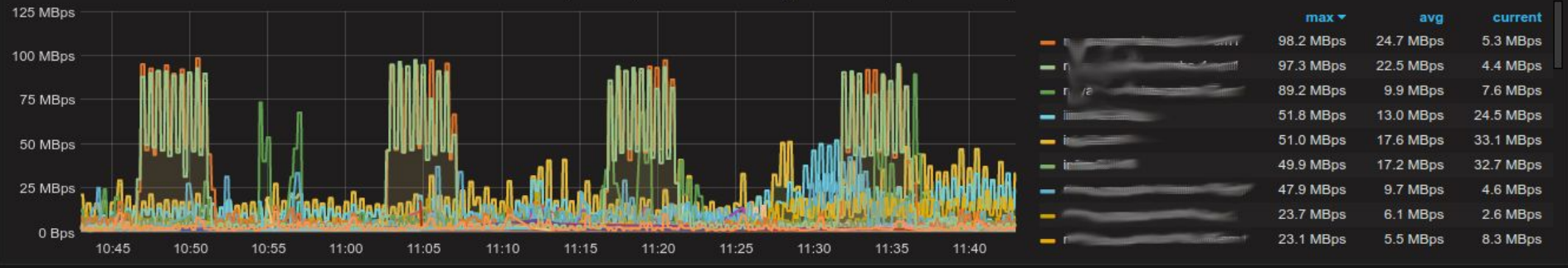
Zoom Out | Last 1 hour UTC

top: 20 | service: .*nova.* | host: .* | disk: sd[b-z][0-9]* | iface: [eb].* | rate_type: irate | rate_min_bps: 10e3 | rate_min_pps: 100 | lat_min_ms: 1

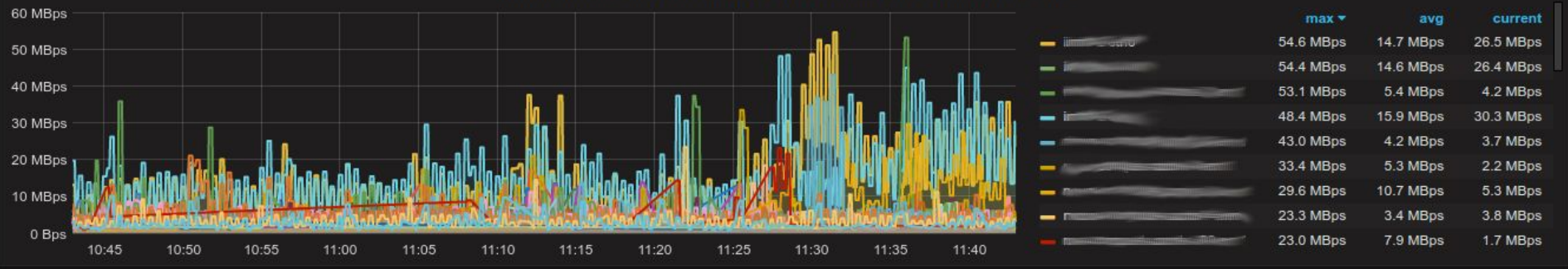
LOAD PERFORMANCE METRICS

NETWORK PERFORMANCE METRICS [IRATE]

.*nova.* (hosts: .*): network rx irate by host-iface (top 20 thru time) > 10e3



.*nova.* (hosts: .*): network tx irate by host-iface (top 20 thru time) > 10e3



DISKS PERFORMANCE METRICS - THROUGHPUT IOPS [IRATE]

.*nova.* (hosts: .*) read IOPs irate (top 20 over time) > 100

ubuntu

&& thank you :)

@xjjo

CANONICAL