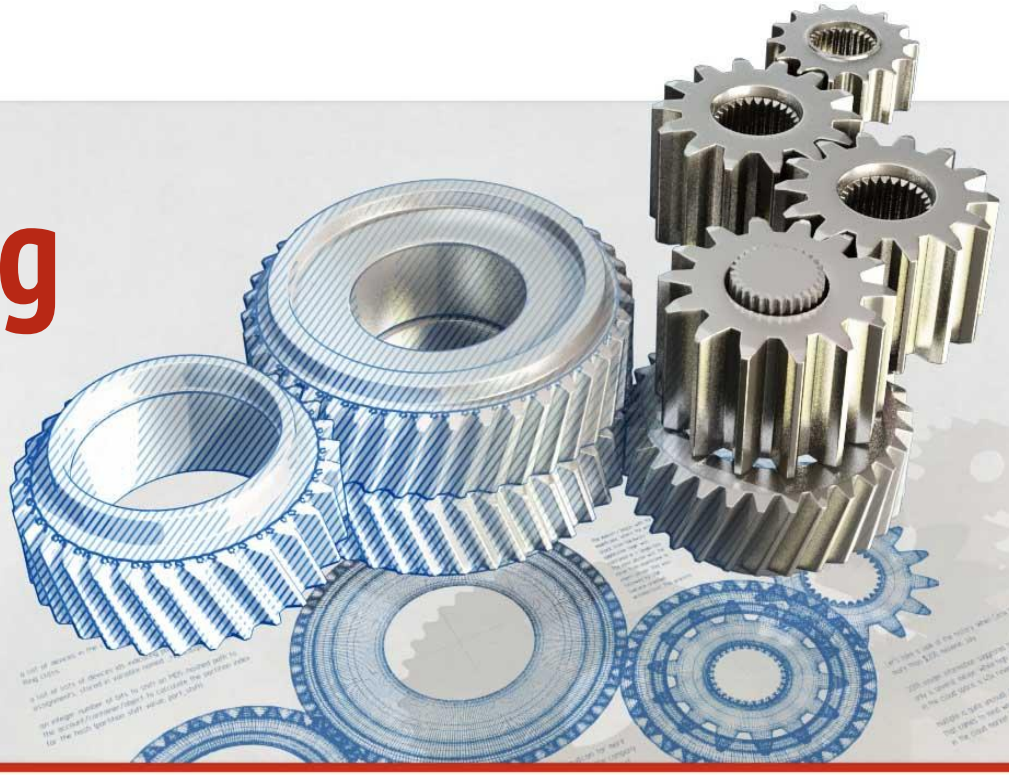
MIRANTIS

Load Balancing as-a-service



Ilya Shakhat
Eugene Nikanorov

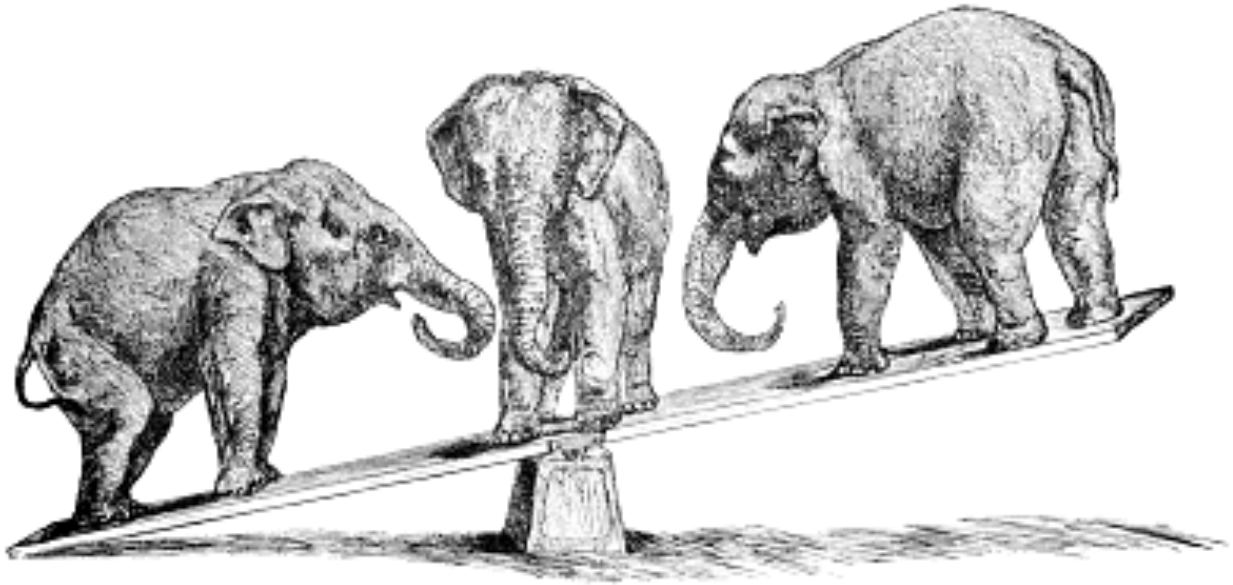
Thanks!

Avishay, Eugene, KC, Leon, Mark, Oleg, Youcef

... and all who helped with discussions, specifications and testing

MIRANTIS

Agenda



- History
- Grizzly
  - features
  - architecture
  - workflow
- Future

MIRANTIS

History

- Stand-alone solutions:
  - Atlas (Java)
  - Equilibrium (Python)
- In Grizzly became Quantum extension

MIRANTIS

Features Overview

- REST API
- CLI
- UI in Horizon
- On-host HAProxy
- DevStack out-of-box support

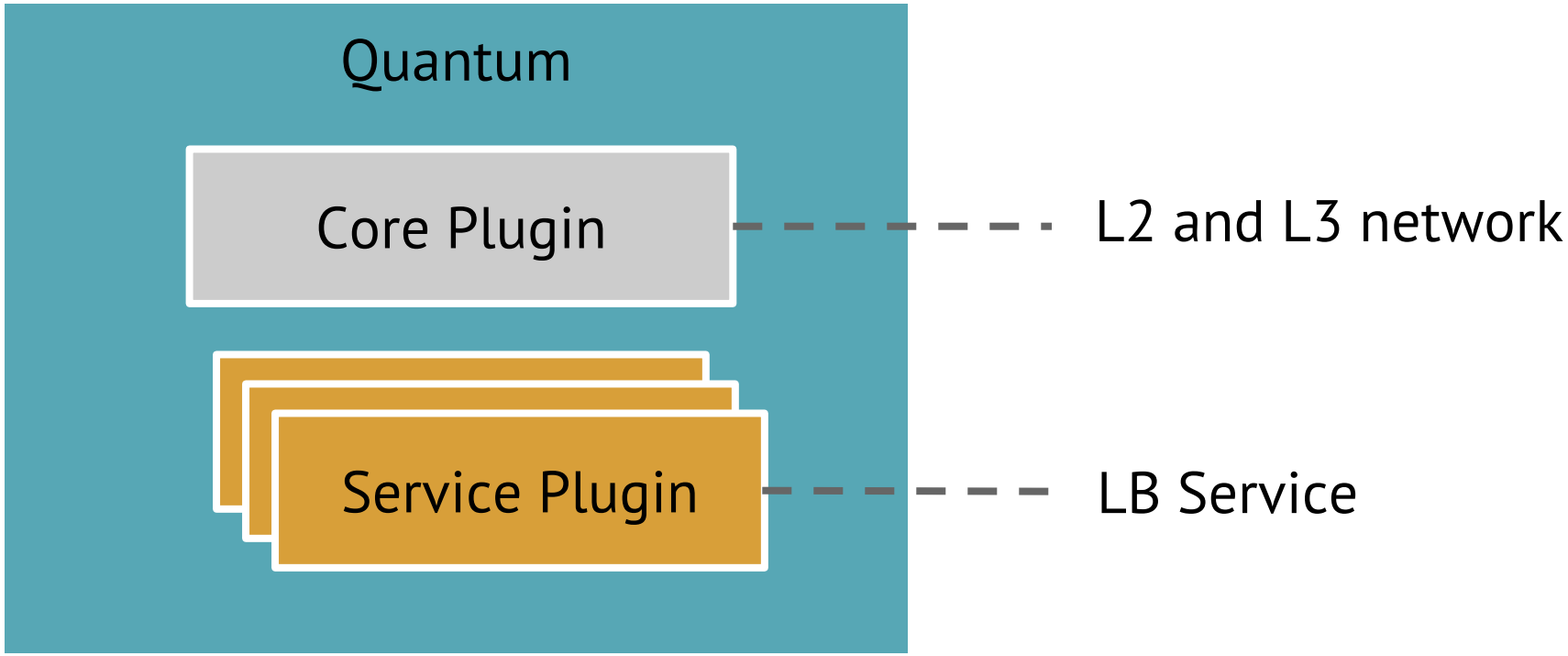
MIRANTIS

Load Balancing Features

- Load balancing between services on VMs
- Load balancing methods (e.g. round-robin)
- Session persistence
- Health monitoring (TCP, HTTP)
- Connection limit

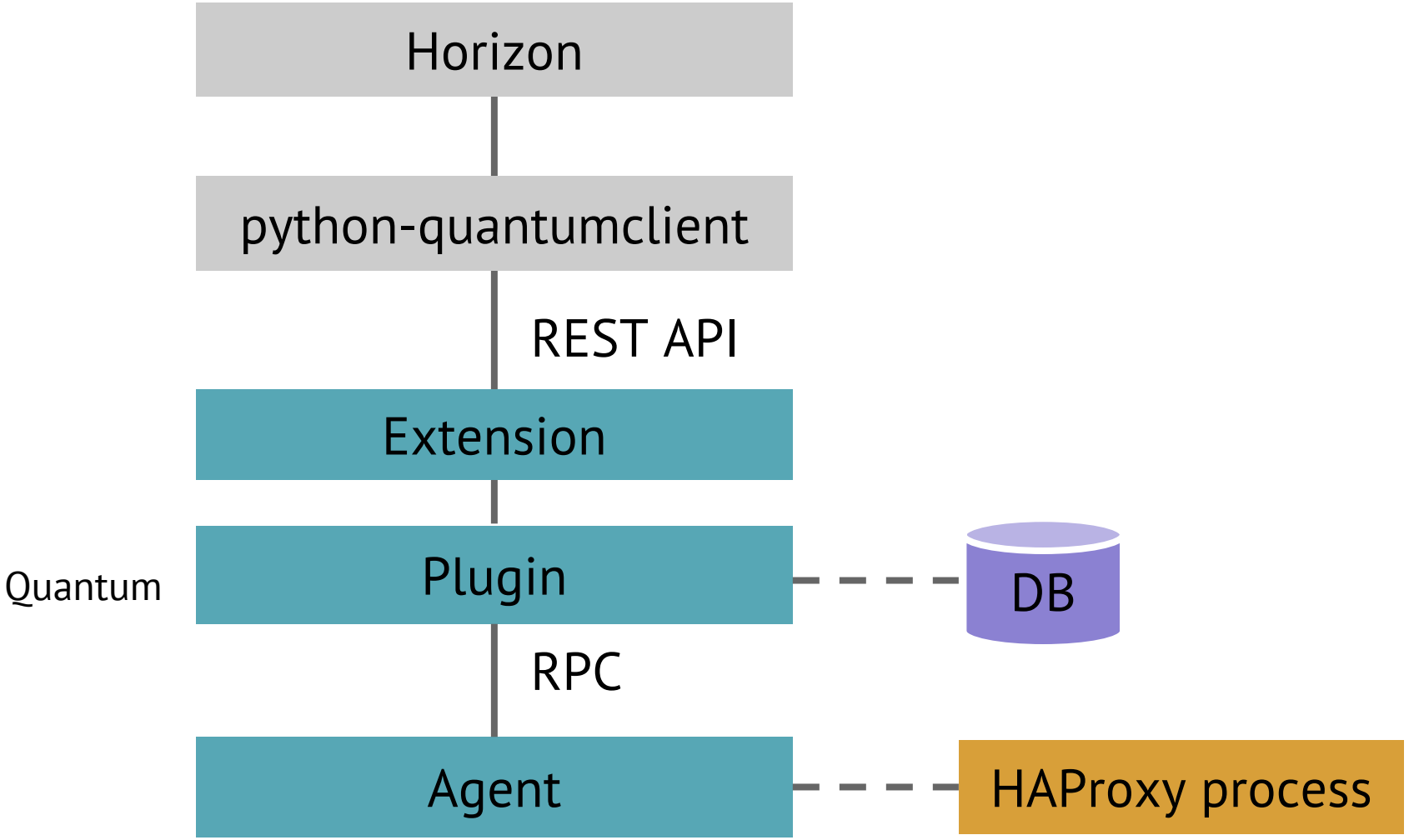
MIRANTIS

Quantum Service Plugins



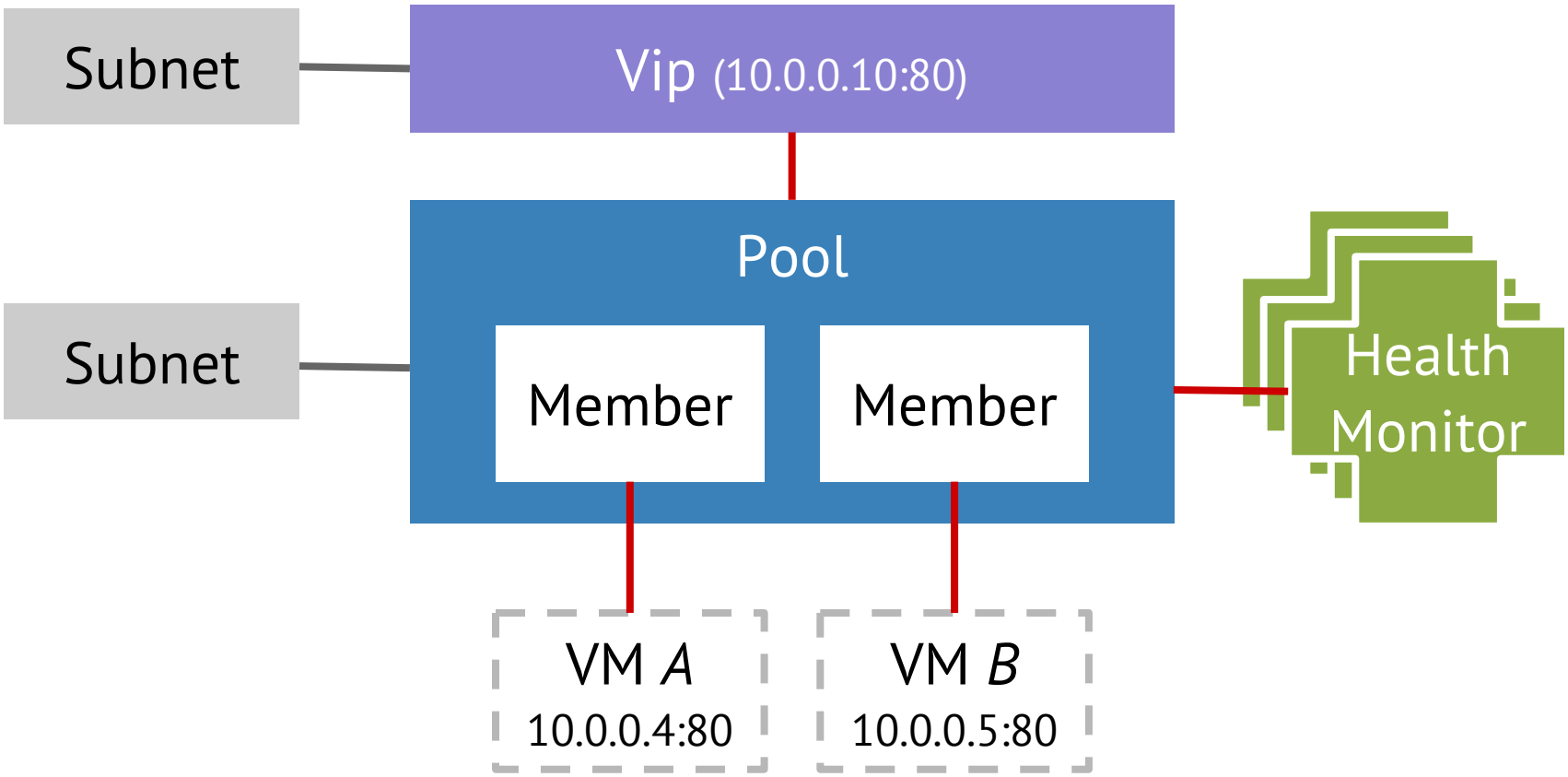
MIRANTIS

Architecture



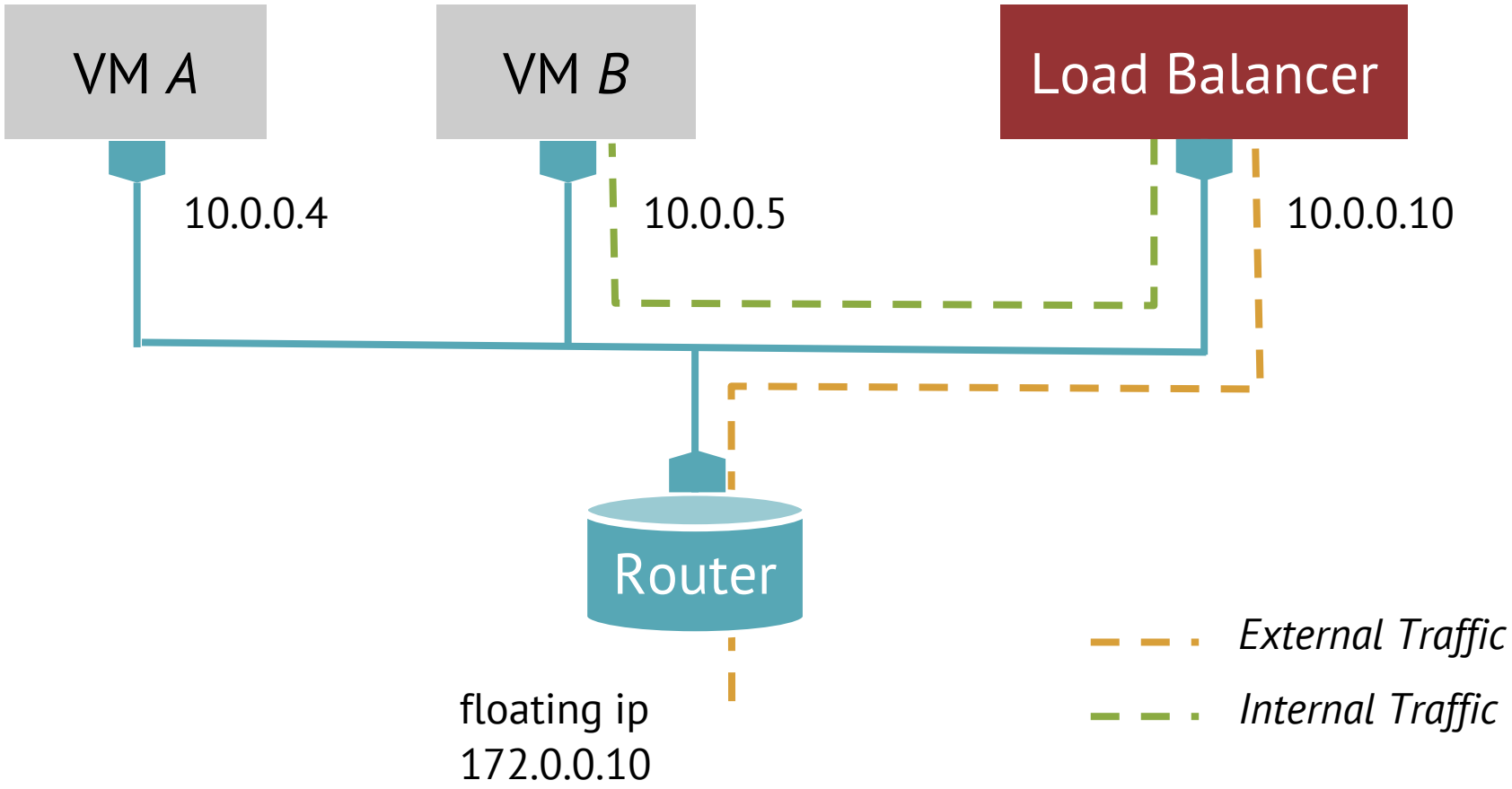
MIRANTIS

Model



MIRANTIS

Wiring



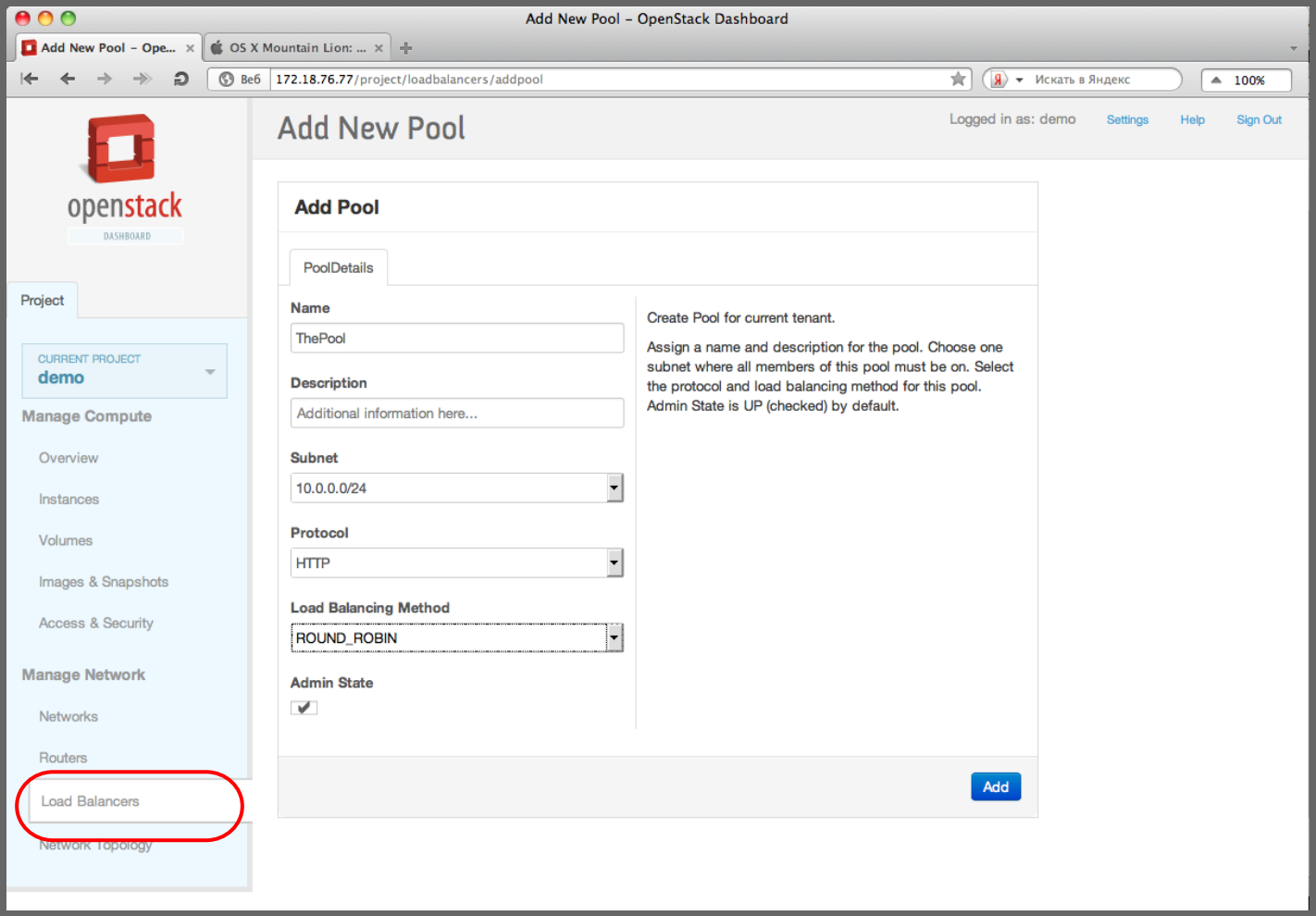
MIRANTIS

Workflow

1. Create pool
2. Create members (1 per VM / service)
3. Create vip for the pool
4. (opt.) Create health monitor and associate with Pool

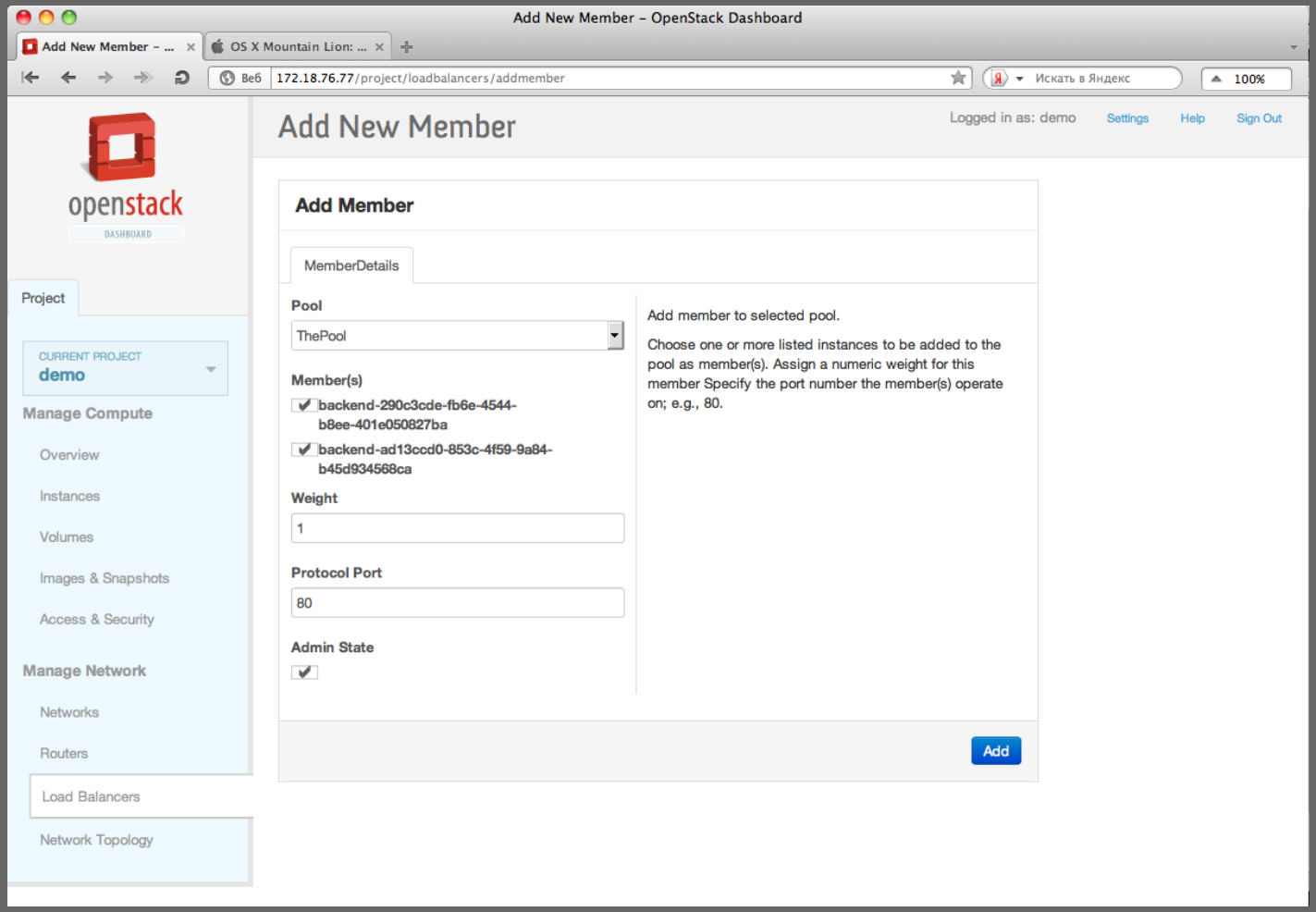
MIRANTIS

UI | Step 1



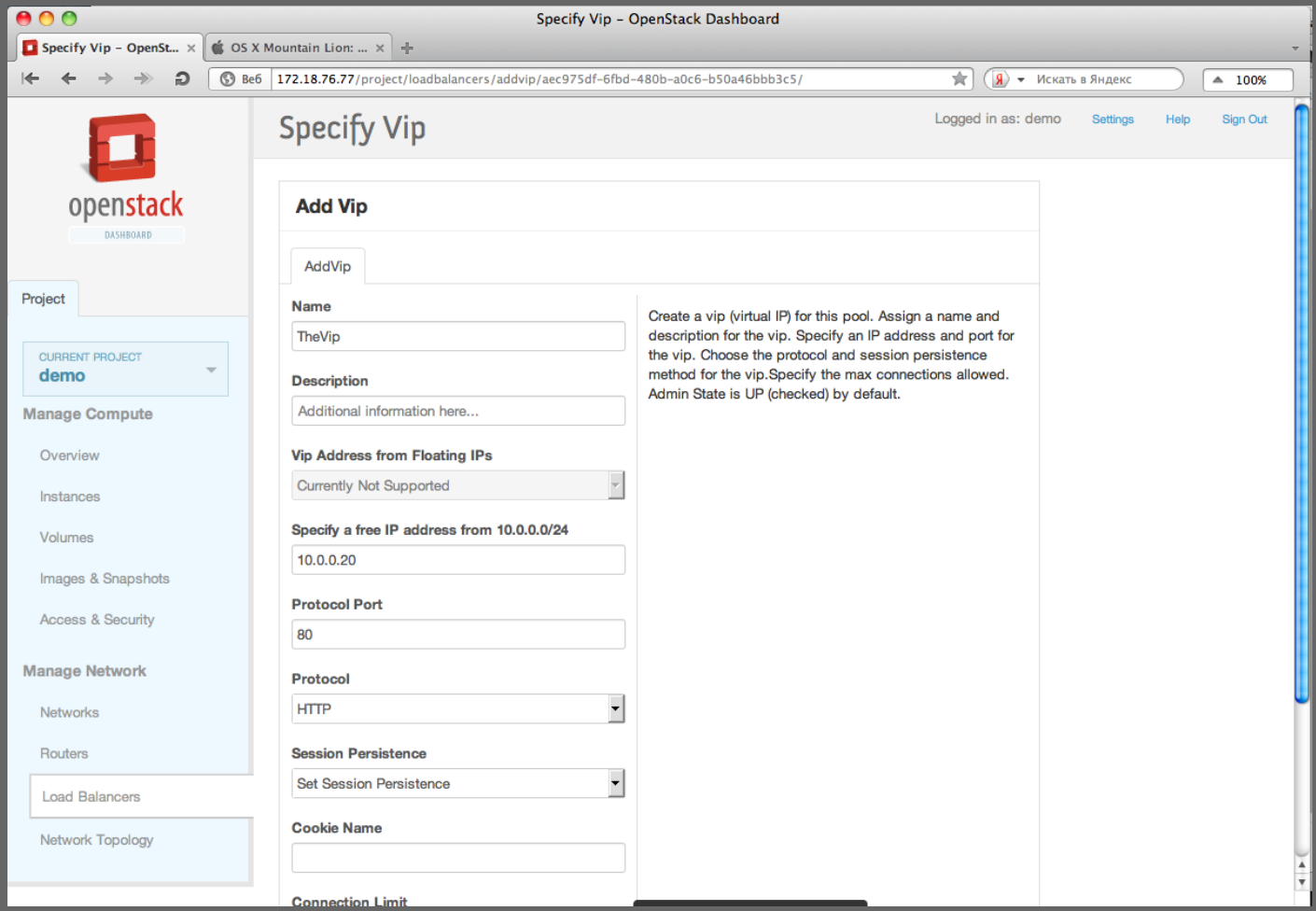
MIRANTIS

UI | Step 2



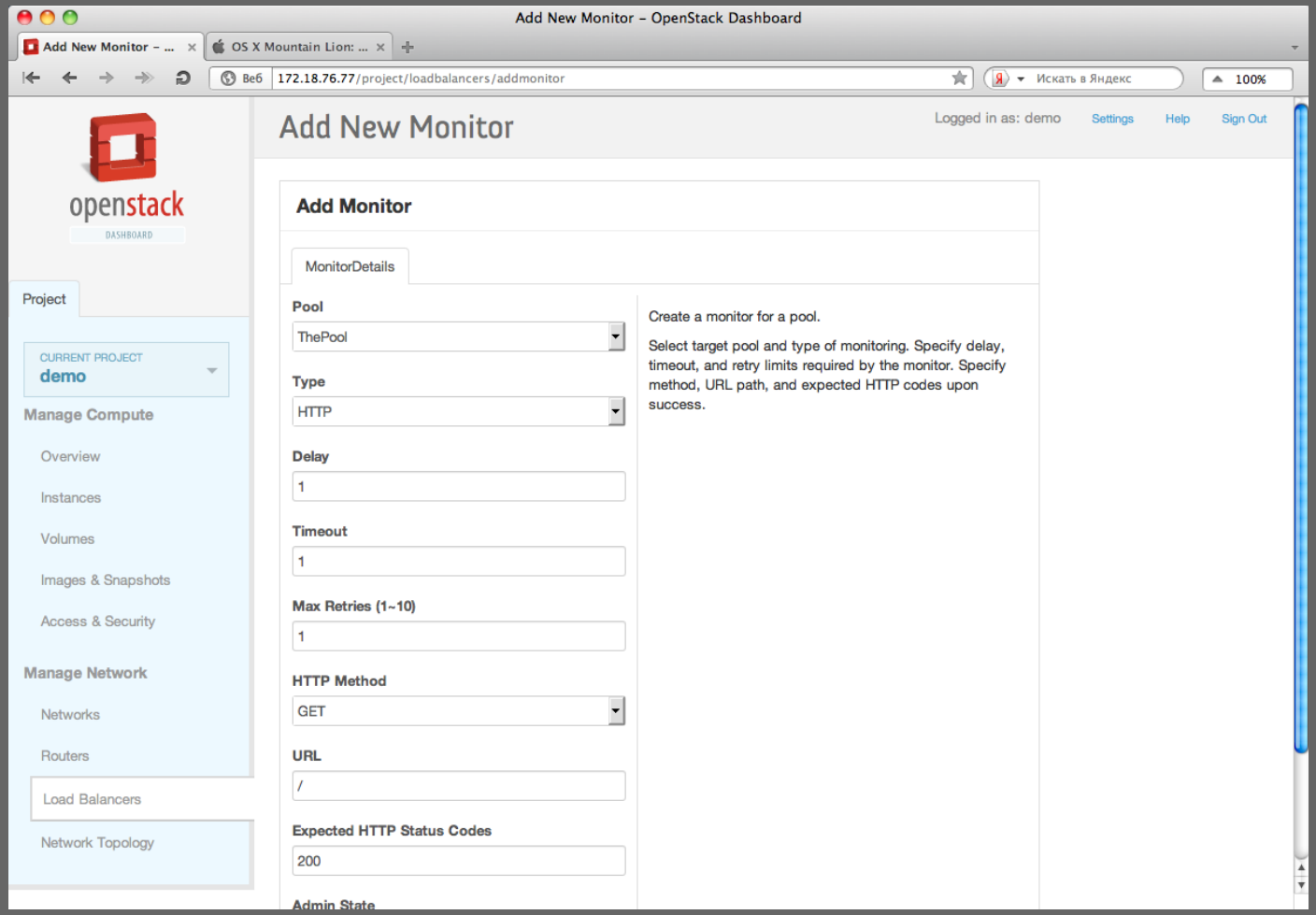
MIRANTIS

UI | Step 3



MIRANTIS

UI | Step 4



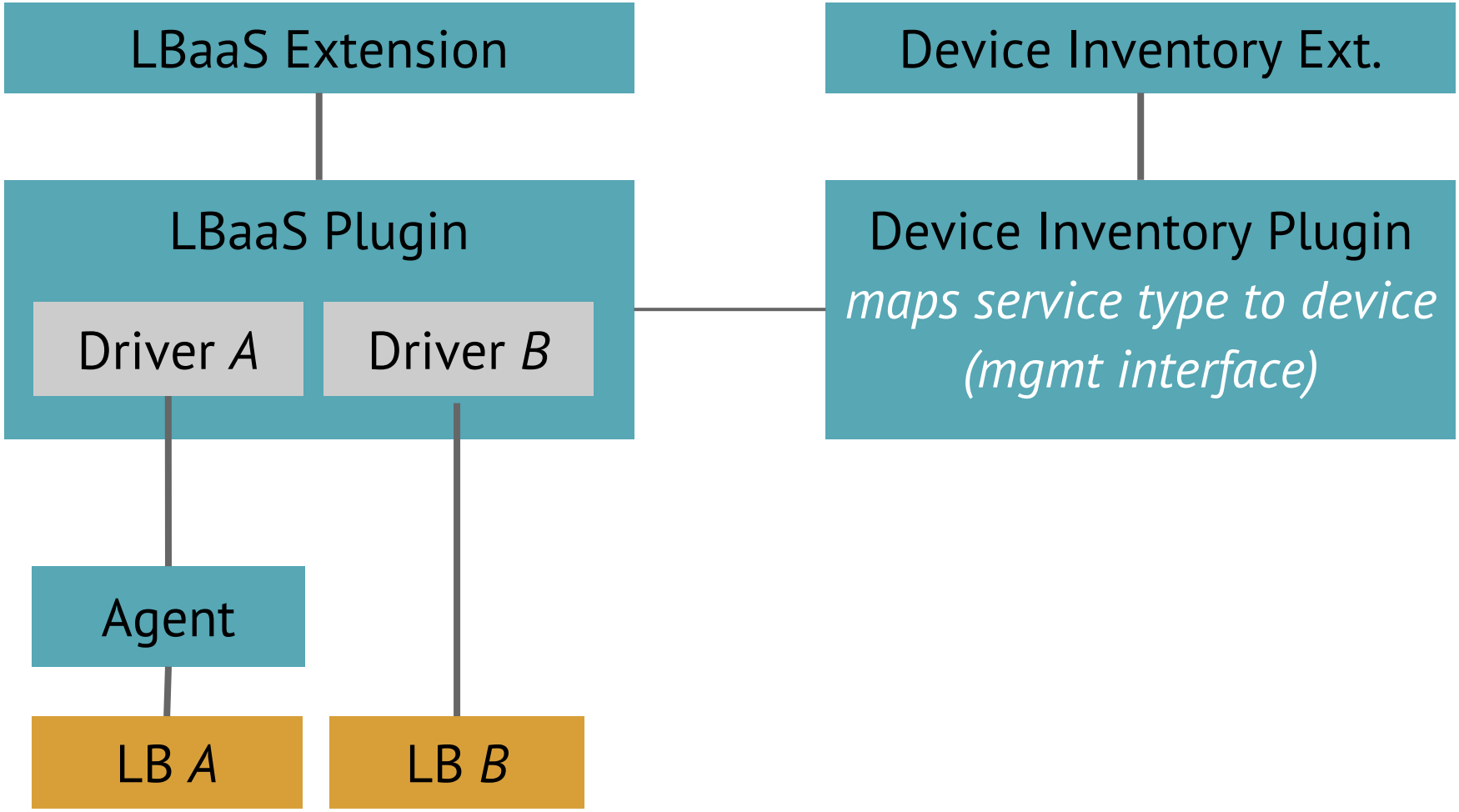
MIRANTIS

LBaaS Next

- Support of HW / VM LBs
  - driver API
- Device Inventory
- Service insertion modes
  - routed mode
  - 1-arm
  - other modes

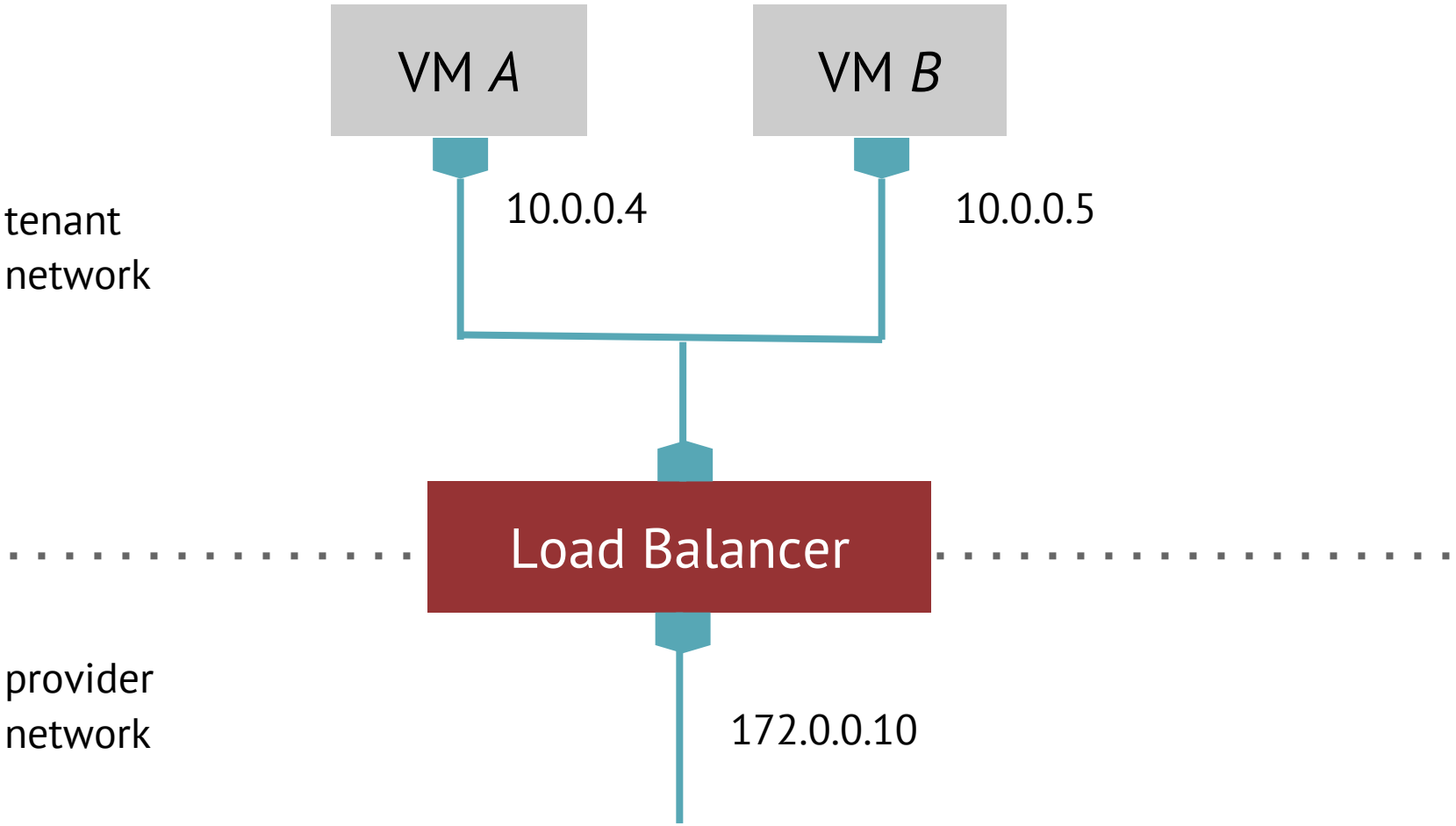
MIRANTIS

Architecture



MIRANTIS

Routed Mode Insertion



MIRANTIS

Q&A

MIRANTIS