우린 그냥 시작했을 뿐인데 

클라우드가 돌아가고 있습니다

: Openstack을 활용한 학생용 클라우드 인프라 구축기

아주대학교 소프트웨어융합대학

소학회 Aolda

이나현, 이찬주, 김화균, 박병언

#고민

#탐색

#도전

목차

01

We are

아올다 소개 및 설립 동기 | 아올다 히스토리

Aolda

아주대학교 학생 클라우드 운영소학회

#소개

클라우드 인프라 개발

Openstack 기반
학생용 클라우드 구축

학생 대상 프로젝트 운영환경 제공

장기적 배포/운영 경험 지원을 위한
‘아올다 클라우드’ 서비스 및
도메인 제공

기술경험 공유

소학회 내 활동 블로그 운영 및
인적 네트워킹 활동 진행

Aolda

아주대학교 학생 클라우드 운영소학회

#설립동기

부담스러운
클라우드 배포비용

상용 클라우드의
지원기간 제한

활동 증빙제출의
부담

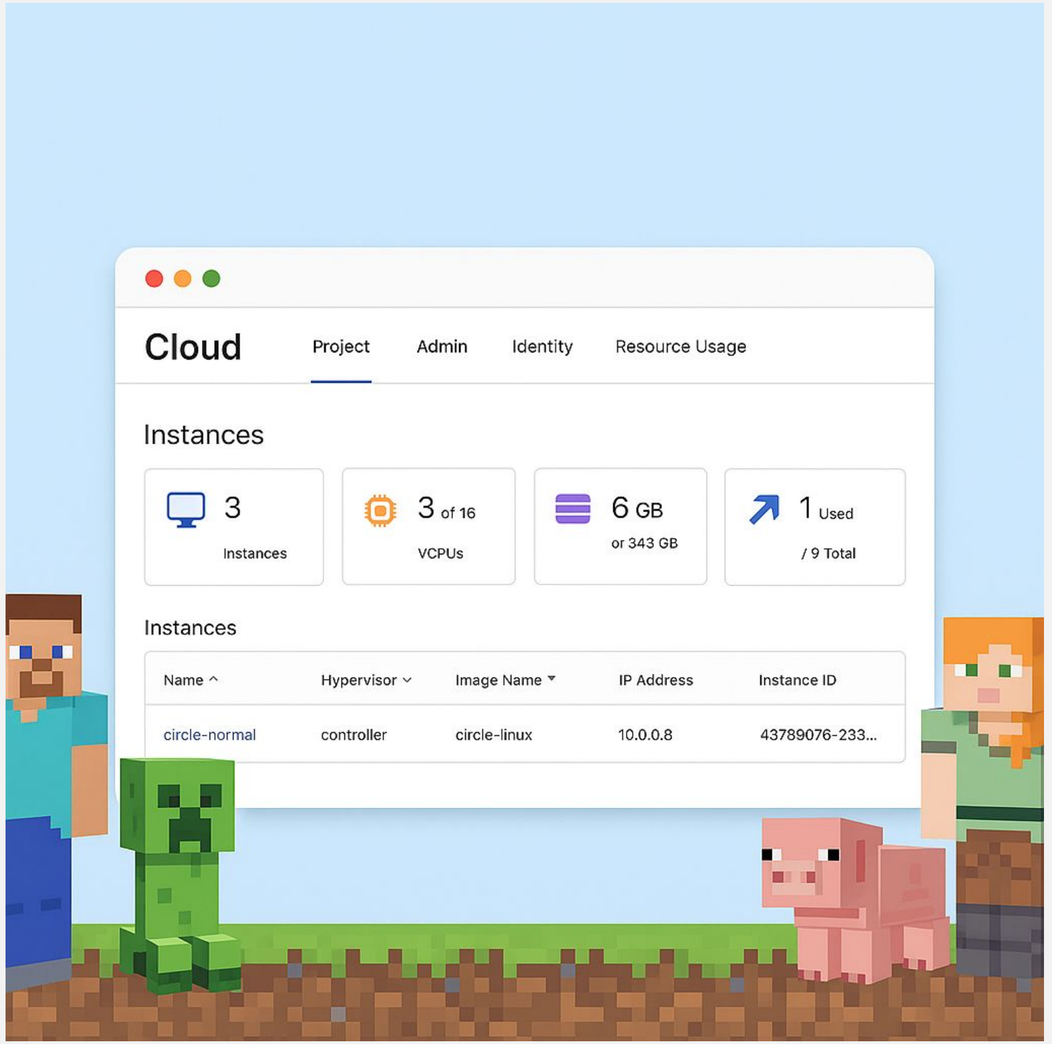
▶ 클라우드 배포환경 제공을 통한 깊은 학습의 기회 제공

사실은 ...

거창한 계획은 없었어요.

어 ? 근데 사양이 생각보다 좋았어요.

학교 전체가 함께 써보면 어떨까?

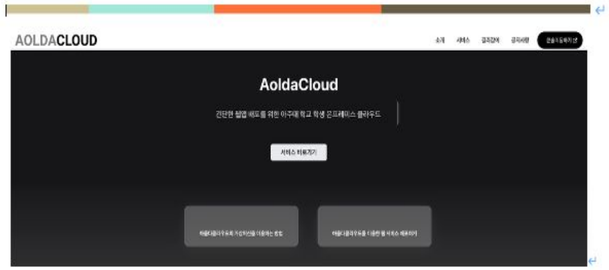


Aolda의 첫 시작

Aolda의 첫 시작

검색 ⌘F

학장님 허락이 떨어졌습니다. "아주대학교 SW중심대학사업의 지원을 받아" 로 해서 제출하시면 되겠습니다.



Aolda Cloud

웹 앱 배포를 위한 아주대학교 학생 클라우드

Last Update : 2024년 4월 3일

작성자

Aolda의 확장

아주대학교 AJOU UNIVERSITY | SW융합교육원 Center for Excellence in SW

교과목 및 산학 과제에서의 활용 계획

- 오픈소스소프트웨어개론 - 학생들이 과제로 개발한 웹, Docker, Kubernetics 기반 프로젝트 실습
- 웹시스템설계 - Docker 실습 및 시험 서비스 개발
- 실전코딩1 - Docker 기반 실습 및 서비스 개발
- 실전코딩2 - IOT 기반 서비스의 서버 프로그래밍 실습
- 자기주도연구, 자기주도프로젝트, 파란학기 - 개발된 서비스를 지속적으로 운영, 모니터링
- 캡스톤디자인 - 개발된 학생 프로젝트를 지속적으로 운영, 모니터링
- SW창업관련 교육 - 시범 서비스를 개발하여, 교내에서 운영해 봄으로써, 고객의 반응을 모니터링

목표

"학생으로서 해볼 수 있는 최고의 클라우드 경험을 한다"라는 가치를 실현하기 위해 아래와 같은

장비 구매 요청

위 목표들을 달성하기 위해 추가로 다음 장비를 구매해 주실 것을 요청 드립니다.

분류	이름	개수	가격 (최저가, 배송비 제외)	비고
서버 (구매 보류)	TYAN TAKO-KST208-(B72S7-20C23)	2ea	17,520,000원 = 8,760,000 * 2	- [다나와 링크] - 전력 총 1400W
Rack	세이프네트워크 HPS-35SP 방음방진 서버랙	1ea	1,100,000원	- [다나와 링크]
네트워크 스위치 (변경됨)	CISCO SG95-16 스위치허브	1ea	209,000원	- https://item.gmarket.co.kr/Item?goodscode=2526342406
관리(모니터)	ATEN 8포트	1ea	1,890,000원	- [다나와 링크]

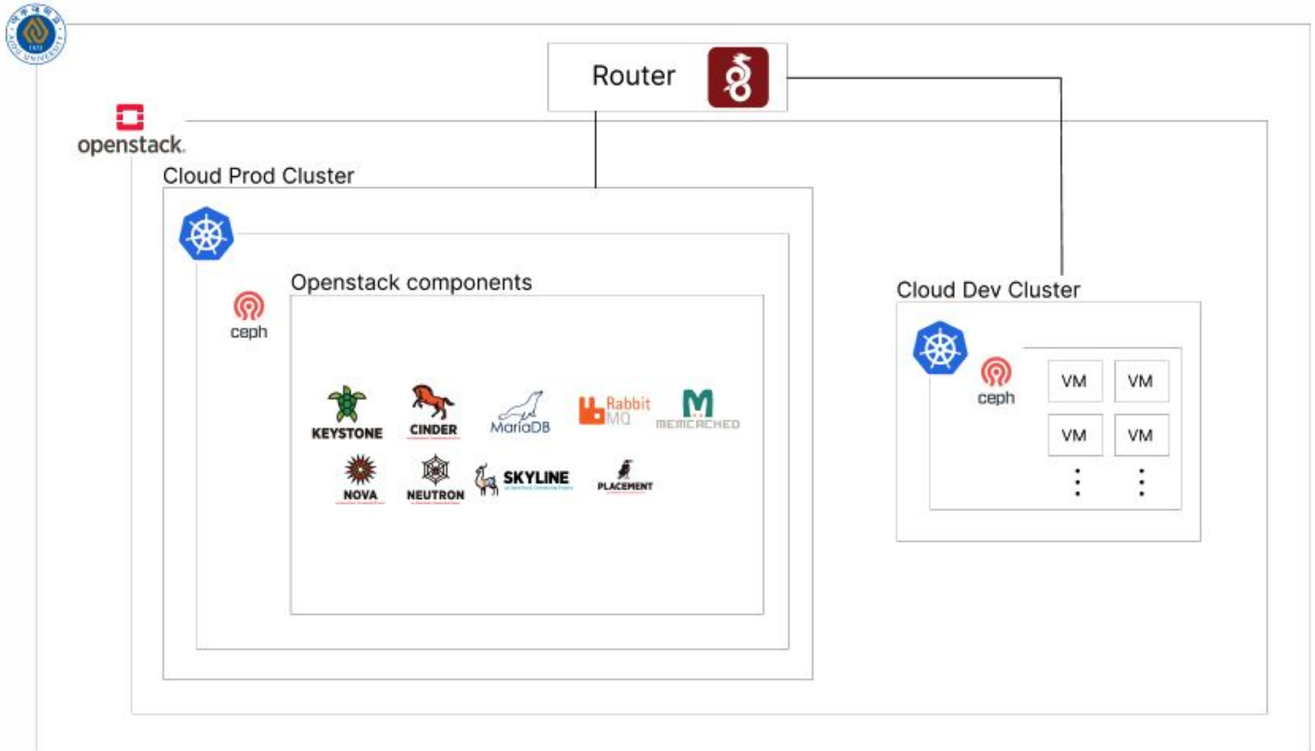
Aolda의 확장

Aolda의 현재

02

We think

아올다 기술스택 선정과정

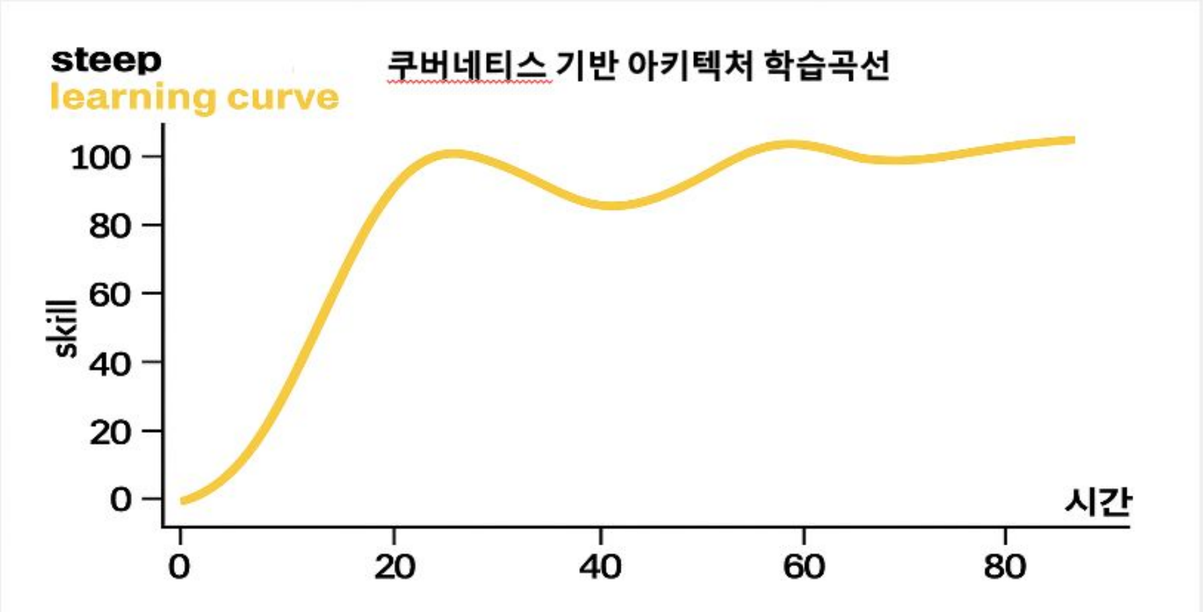


초기 목표 및 접근법

- 오픈스택과 스토리지 모두 쿠버네티스 환경에서 통합 관리
- 오퍼레이터 패턴을 활용한 오픈스택 서비스 제공
- rook-ceph를 통한 스토리지 클러스터 관리
- 쿠버네티스에서 오픈스택과 스토리지 통합 관리

이 접근법은 초반에는 매력적이었지만,
운영 중복성 및 복잡성 문제 발생

문제점 1: 가파른

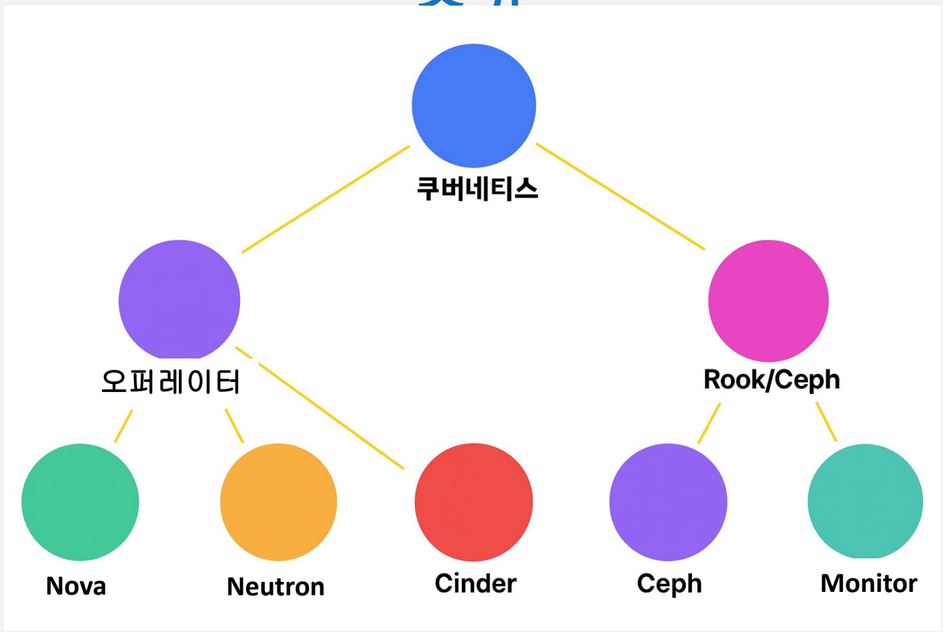


복잡한 기술 스택

높은 학습 장벽

통합 관리의 어려움

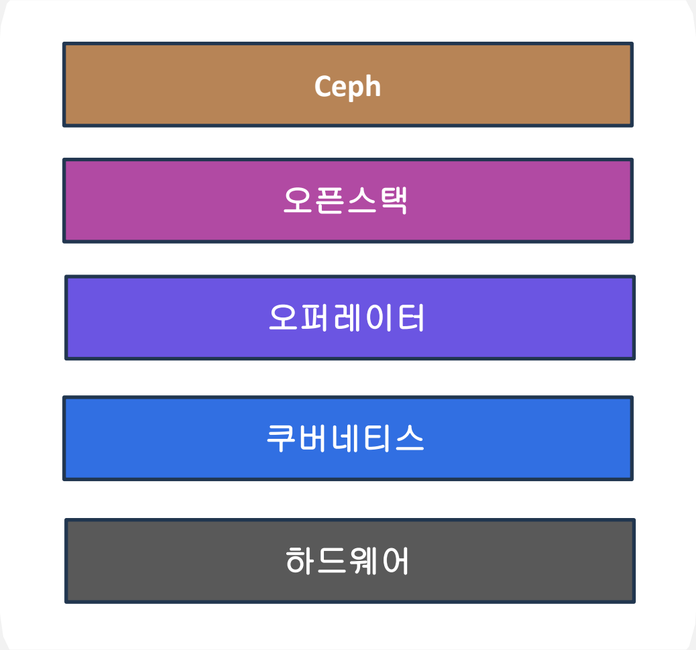
문제점 2: 관리 포인트



복잡한 관리 계층 구조

운영 복잡성 증가

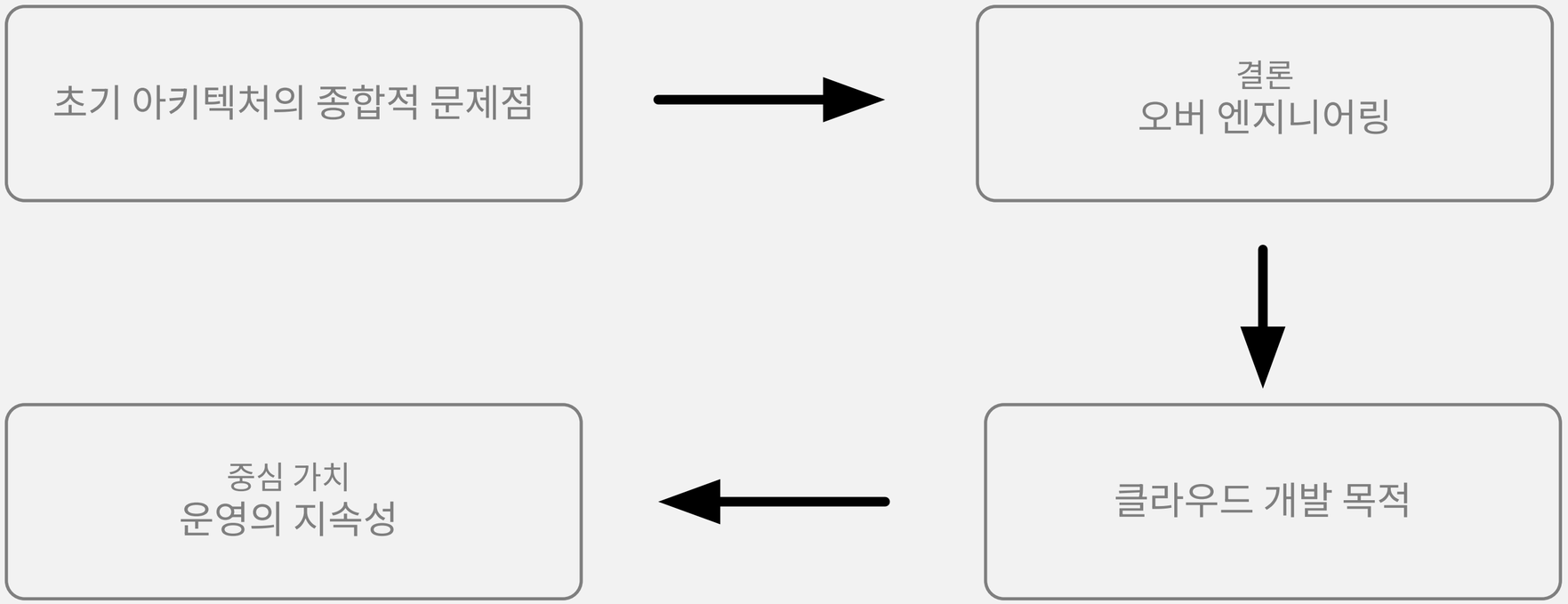
문제점 3: 계층으로 인한 문제 발생 지점 파악



복잡한 관리 계층 구조

문제 해결 시간 증가

Docker와 Kolla-Ansible로의 전환



오버 엔지니어링에서 지속 가능성으로

Docker로의 전환

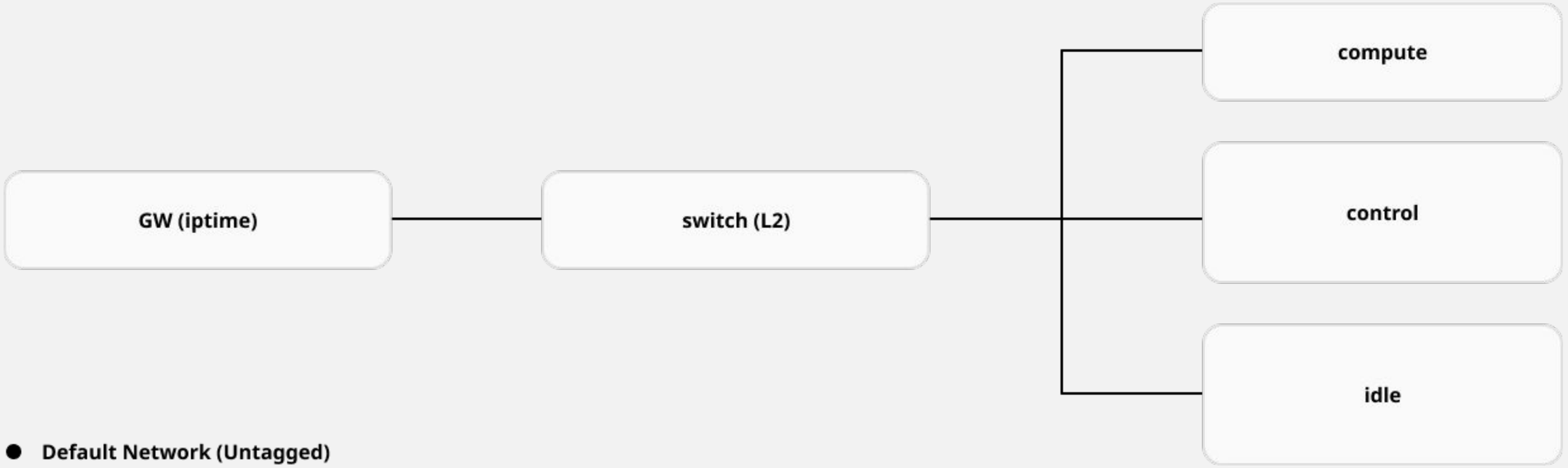
Kolla-Ansible 활용 목적

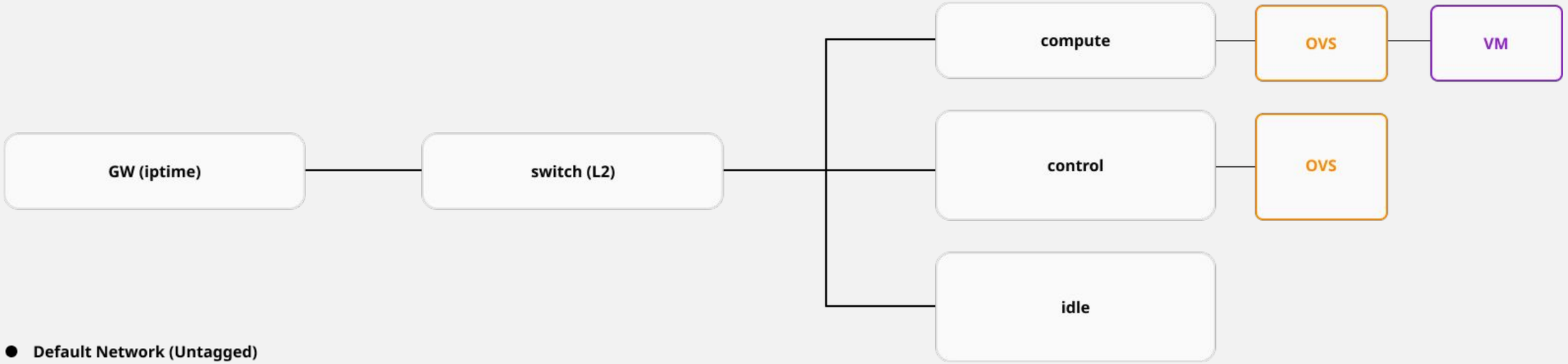
기대효과

03

We did

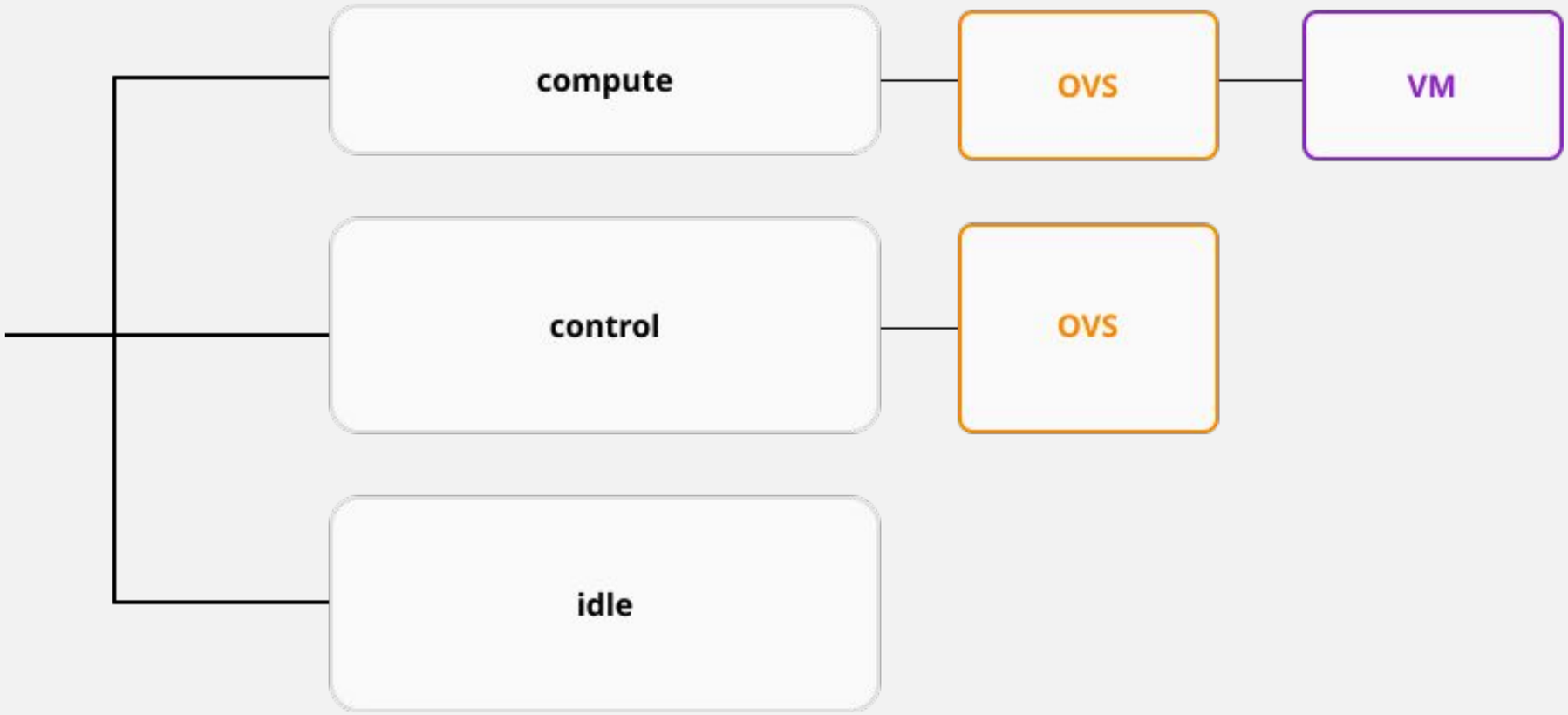
문제점 분석 | 배포환경 개선 | 배포설정 개선

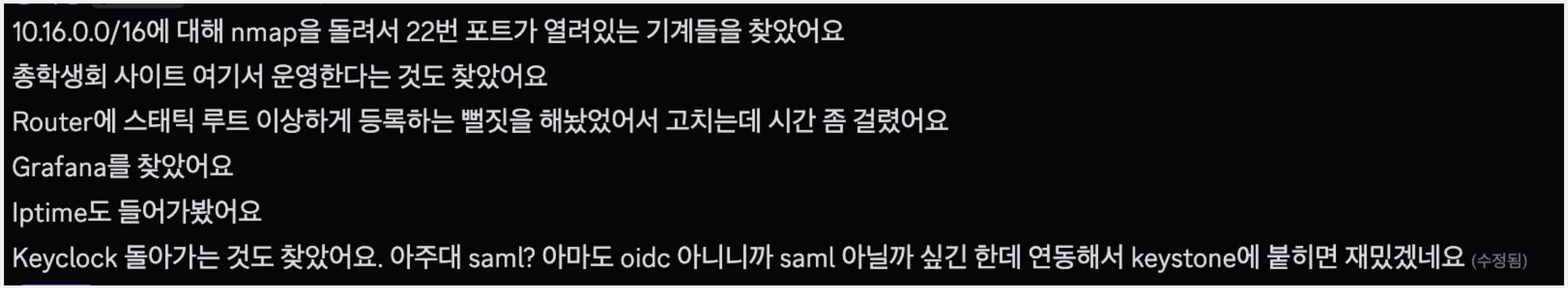

GW (iptime)
switch (L2)
compute
control
idle
Default Network (Untagged)



분리되어 구성되지 않은 네트워크 망 이원화 + 단일 네트워킹 장비

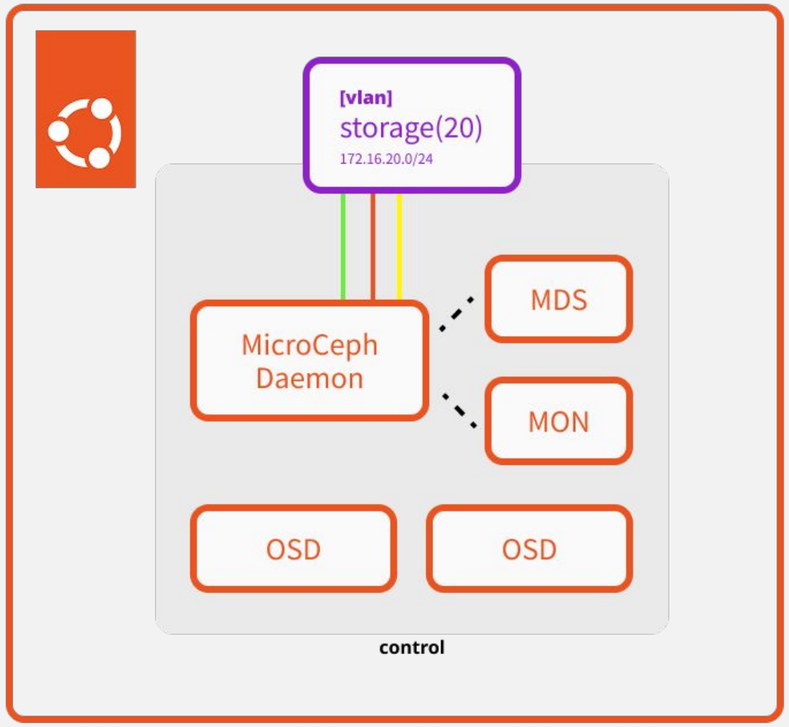
□ 내부망 전체 SPOF


compute
OVS
VM
control
OVS
idle



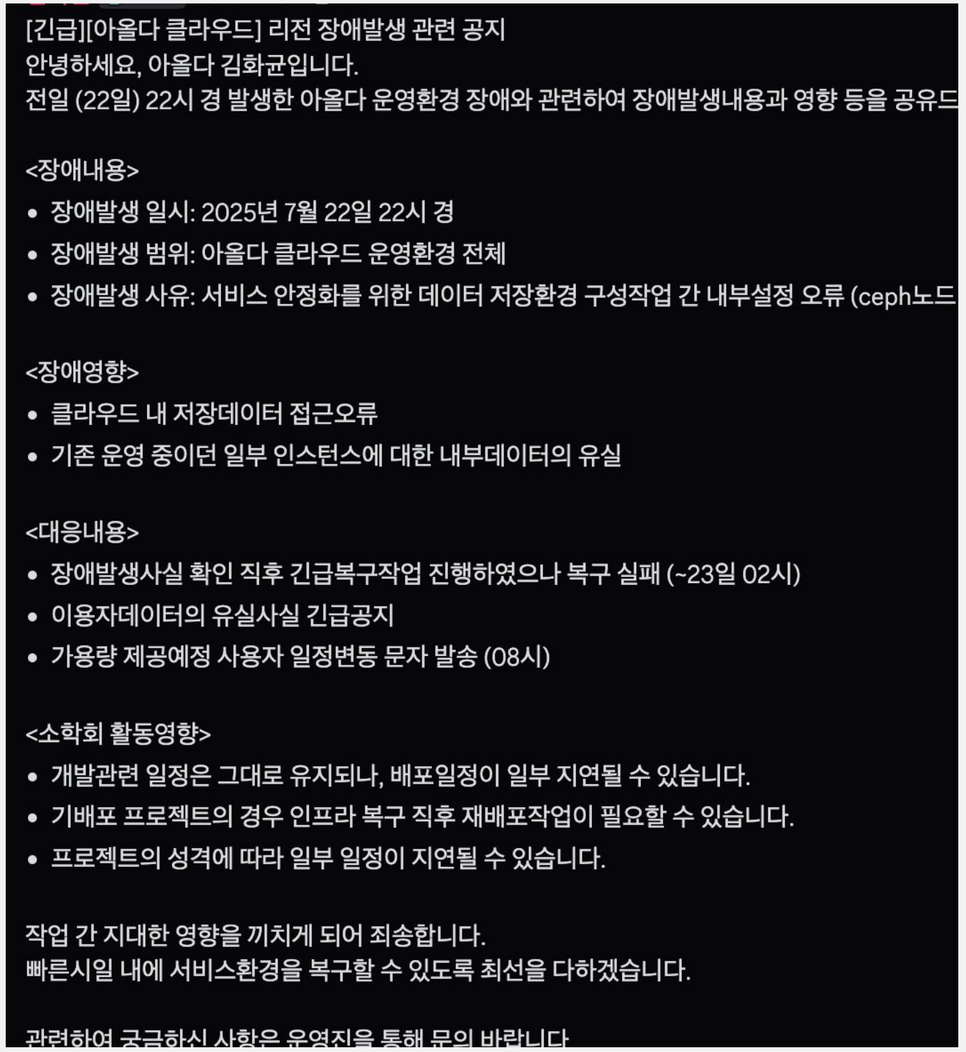
Provider / Management 네트워크 미분리

□ VM을 통한 운영노드 접근 가능



단일노드에 집중된 스토리지 구성

- Control 노드 장애 시
  전체데이터 유실



[긴급][아올다 클라우드] 리전 장애발생 관련 공지
안녕하세요, 아올다 김화균입니다.
전일 (22일) 22시 경 발생한 아올다 운영환경 장애와 관련하여 장애발생내용과 영향 등을 공유드

<장애내용>
- 장애발생 일시: 2025년 7월 22일 22시 경
- 장애발생 범위: 아올다 클라우드 운영환경 전체
- 장애발생 사유: 서비스 안정화를 위한 데이터 저장환경 구성작업 간 내부설정 오류 (ceph노드

<장애영향>
- 클라우드 내 저장데이터 접근오류
- 기존 운영 중이던 일부 인스턴스에 대한 내부데이터의 유실

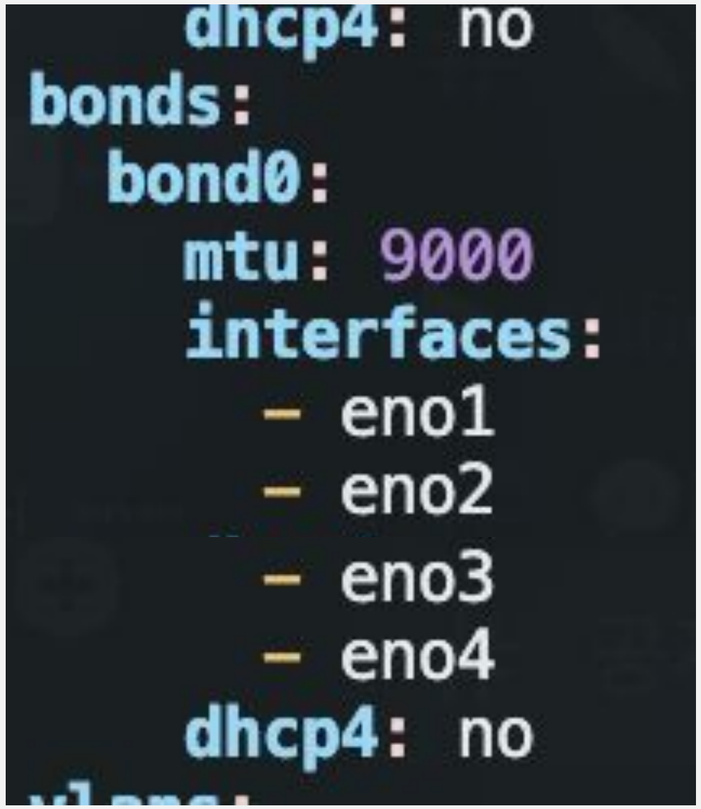
<대응내용>
- 장애발생사실 확인 직후 긴급복구작업 진행하였으나 복구 실패 (~23일 02시)
- 이용자데이터의 유실사실 긴급공지
- 가용량 제공예정 사용자 일정변동 문자 발송 (08시)

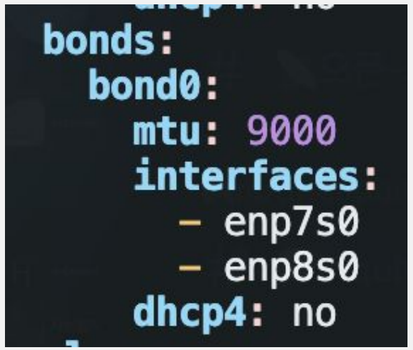
<소학회 활동영향>
- 개발관련 일정은 그대로 유지되나, 배포일정이 일부 지연될 수 있습니다.
- 기배포 프로젝트의 경우 인프라 복구 직후 재배포작업이 필요할 수 있습니다.
- 프로젝트의 성격에 따라 일부 일정이 지연될 수 있습니다.

작업 간 지대한 영향을 끼치게 되어 죄송합니다.
빠른시일 내에 서비스환경을 복구할 수 있도록 최선을 다하겠습니다.

관련하여 궁금하신 사항은 운영진을 통해 문의 바랍니다

A. 안정성 확보를 위한 물리망 분리

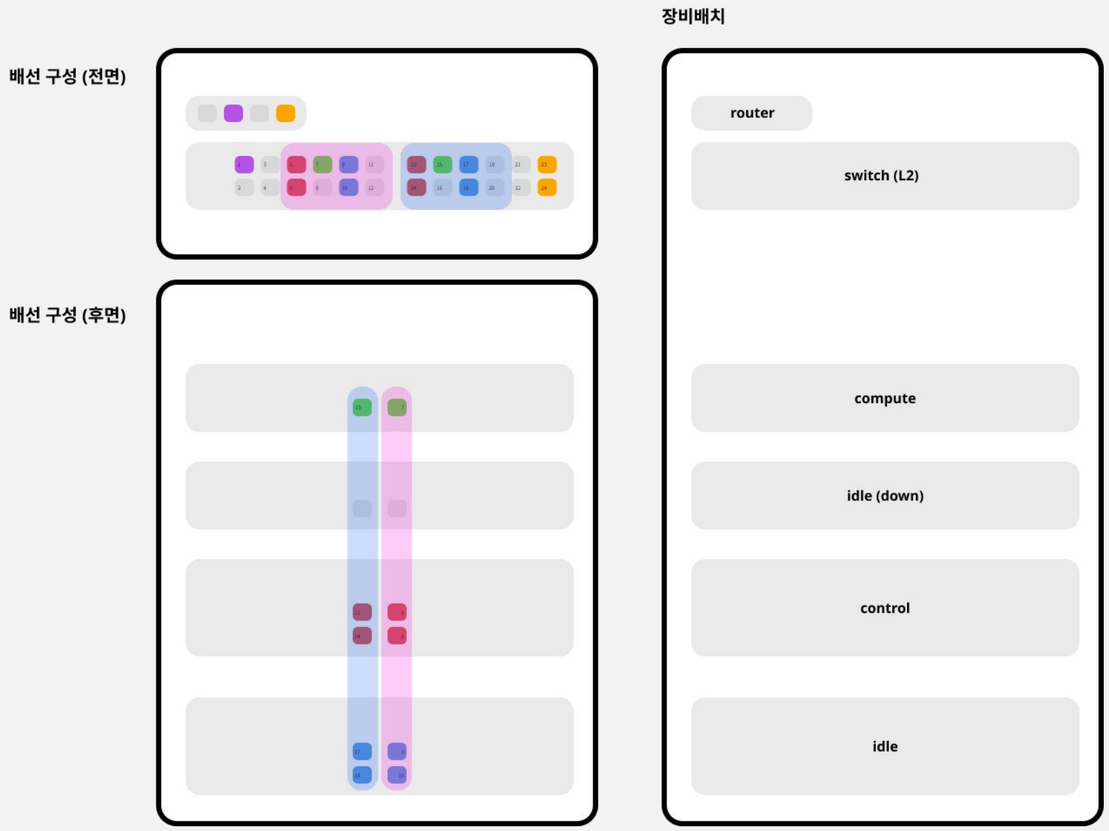




노드 당 2개 이상의 NIC로 네트워크 연결

□ 노드 NIC에 대한 네트워크 SPOF 제거

A. 안정성 확보를 위한 물리망 분리



물리 네트워크 연결구성 개선

- 내부망 SPOF 해결기반 마련

신규 L3 라우터 도입

- 논리적 망분리 및 보안설정 기반마련

B. 목적에 따른 논리망 분리

#기준

1

우선처리 서비스

2

외부노출 서비스

3

다중망 접근 서비스

B. 목적에 따른 논리망 분리

#적용

Before

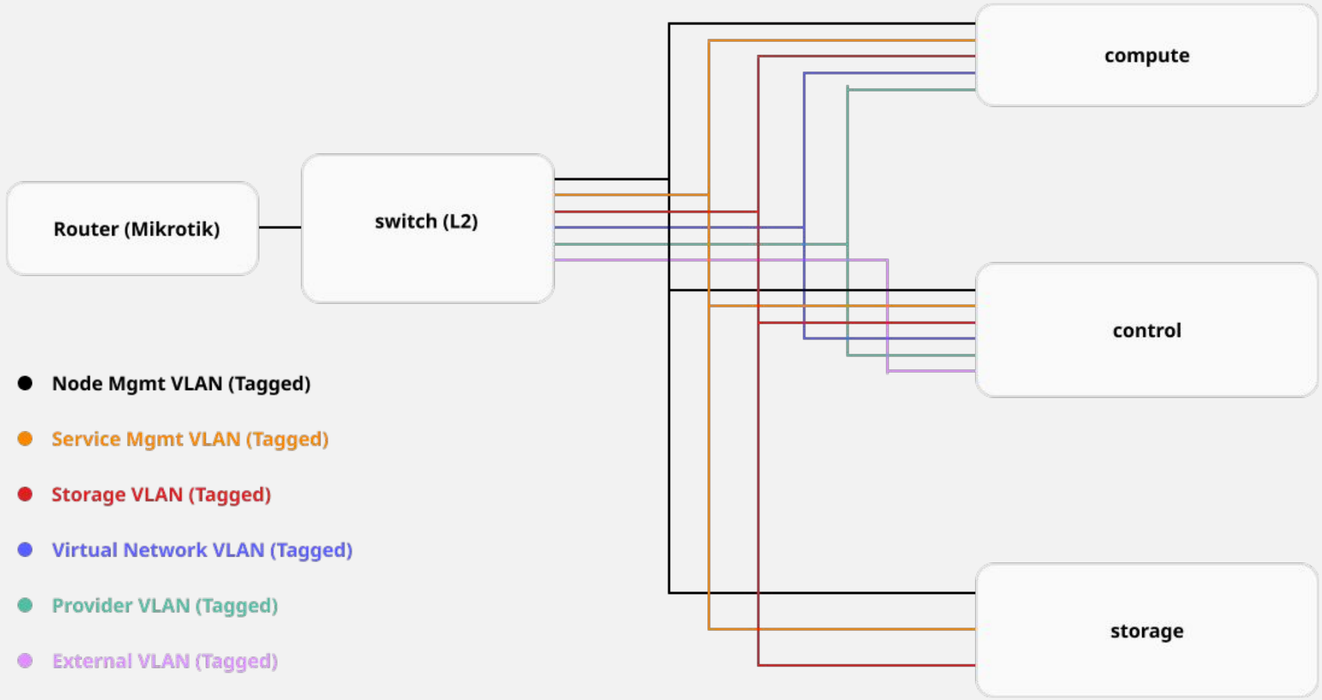
Default Network
(Untagged)

After

Node Mgmt VLAN
Service Mgmt VLAN
Storage VLAN
Virtual Network VLAN
Provider VLAN
External VLAN

B. 목적에 따른 논리망 분리

#적용



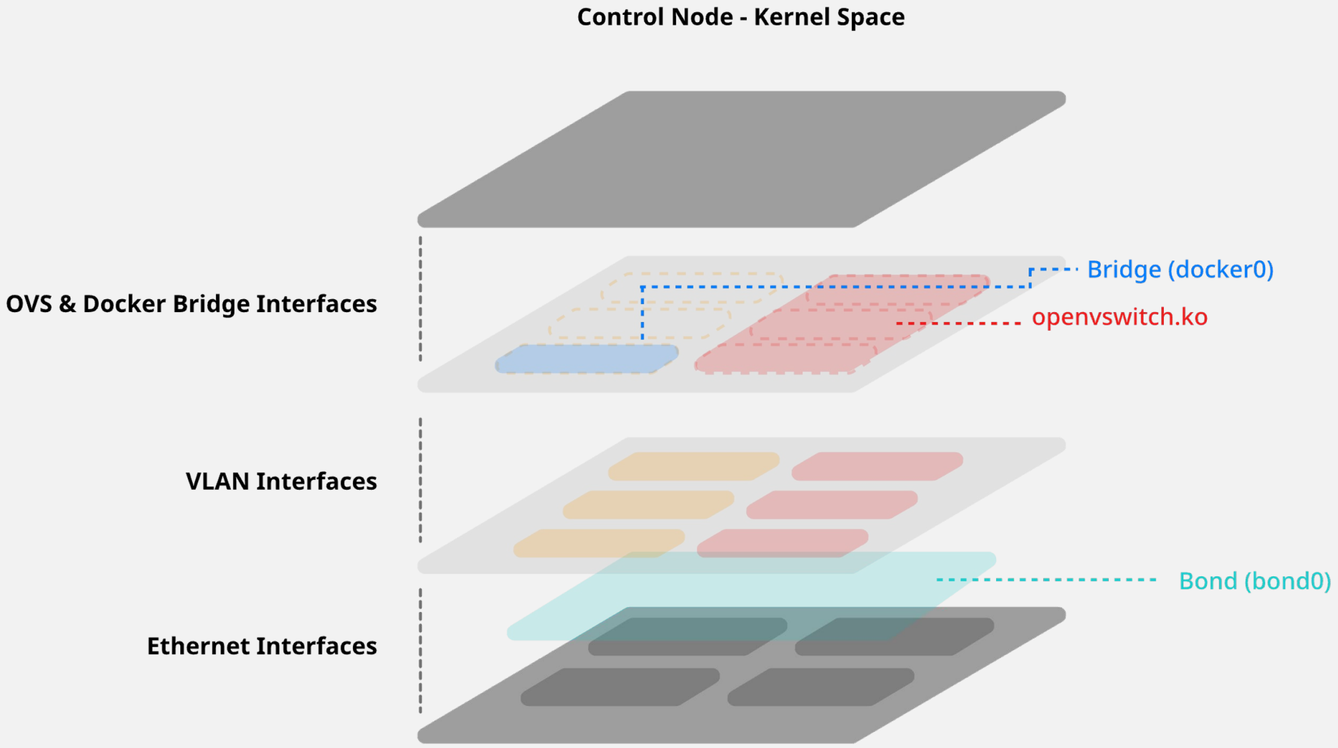

```
    dhcp4: no
  vlans:
    vlan-device:
      id: 100
      link: bond0
      mtu: 1500
    vlan-mgmt:
      id: 10
      link: bond0
      mtu: 1500
    vlan-storage:
      id: 20
      link: bond0
      mtu: 9000
  bridges:
```

```
    dhcp4: no
  vlans:
    vlan-device:
      id: 100
      link: bond0
      mtu: 1500
    vlan-mgmt:
      id: 10
      link: bond0
      mtu: 1500
    vlan-storage:
      id: 20
      link: bond0
      mtu: 9000
    vlan-vxlan:
      id: 30
      link: bond0
      mtu: 1500
```

```
  vlans:
    vlan-device:
      id: 100
      link: bond0
      mtu: 1500
    vlan-mgmt:
      id: 10
      link: bond0
      mtu: 1500
    vlan-storage:
      id: 20
      link: bond0
      mtu: 9000
    vlan-vxlan:
      id: 30
      link: bond0
      mtu: 1500
    vlan-provider:
      id: 40
      link: bond0
      mtu: 1500
    vlan-mgmt-d:
```

C. 논리망 기반 서비스환경 구성

#Issue

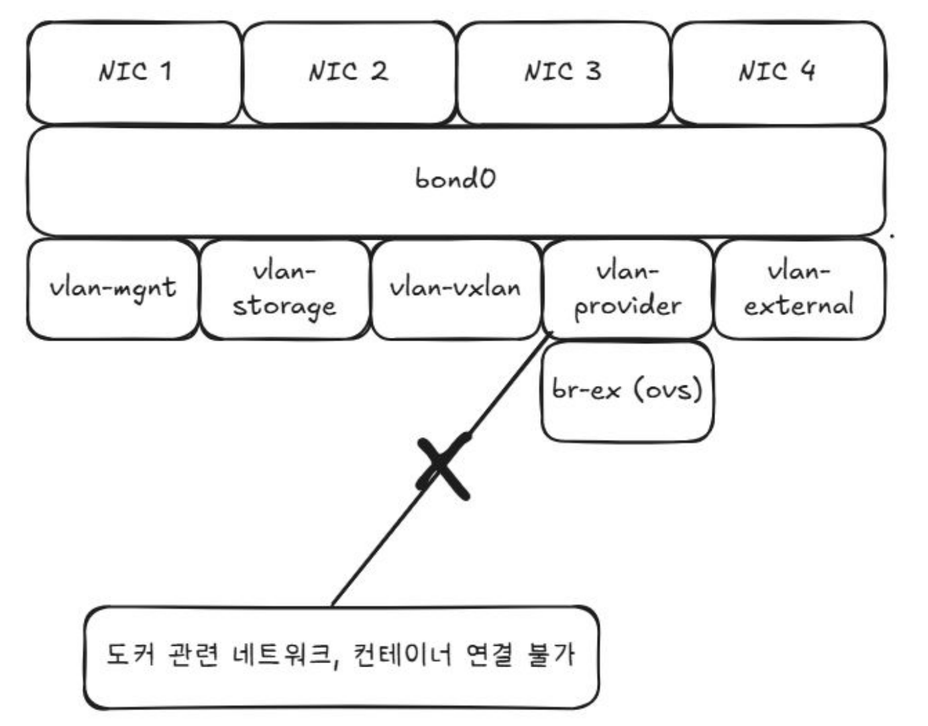


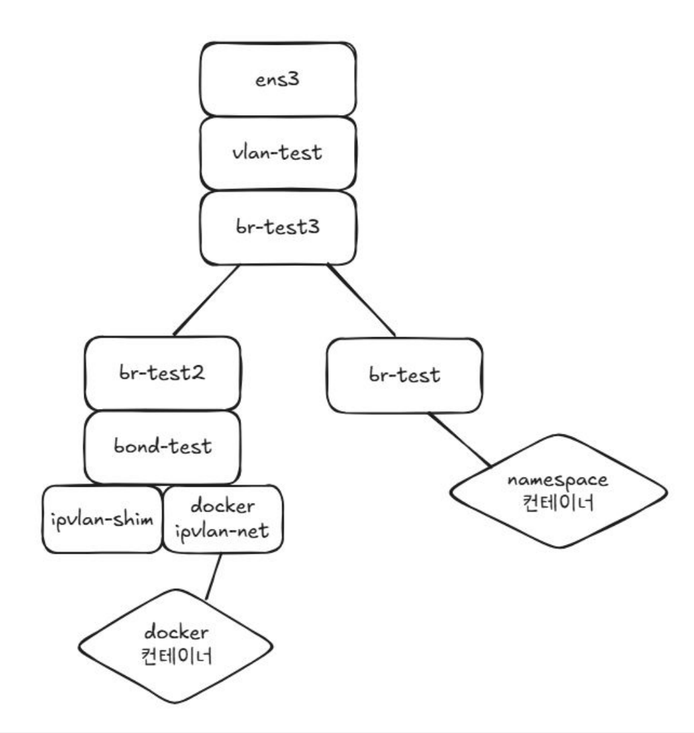
OVS의 인터페이스 점유

► Docker 인스턴스
외부네트워크 직접접근 불가

C. 논리망 기반 서비스환경 구성

#Solve1 – 인터페이스 중첩연결을 통한 강제연결





C. 논리망 기반 서비스환경 구성

#Solve1 – 인터페이스 중첩연결을 통한 강제연결

📌 도커 컨테이너 → bond-test

```
130e5e69f768:~# ping 192.168.0.103
PING 192.168.0.103 (192.168.0.103) 56(84) bytes of data.
64 bytes from 192.168.0.103: icmp_seq=1 ttl=64 time=0.206 ms
64 bytes from 192.168.0.103: icmp_seq=2 ttl=64 time=0.084 ms
64 bytes from 192.168.0.103: icmp_seq=3 ttl=64 time=0.066 ms
^C
--- 192.168.0.103 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2070ms
rtt min/avg/max/mdev = 0.066/0.125/0.206/0.057 ms
```

📌 도커 컨테이너 → br-test3

```
130e5e69f768:~# ping 192.168.0.100
PING 192.168.0.100 (192.168.0.100) 56(84) bytes of data.
64 bytes from 192.168.0.100: icmp_seq=1 ttl=64 time=0.198 ms
64 bytes from 192.168.0.100: icmp_seq=2 ttl=64 time=0.084 ms
64 bytes from 192.168.0.100: icmp_seq=3 ttl=64 time=0.076 ms
^C
--- 192.168.0.100 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2045ms
rtt min/avg/max/mdev = 0.076/0.119/0.198/0.055 ms
```

📌 도커 컨테이너 → ens3

```
130e5e69f768:~# ping 192.168.0.165
PING 192.168.0.165 (192.168.0.165) 56(84) bytes of data.
64 bytes from 192.168.0.165: icmp_seq=1 ttl=64 time=0.253 ms
64 bytes from 192.168.0.165: icmp_seq=2 ttl=64 time=0.082 ms
64 bytes from 192.168.0.165: icmp_seq=3 ttl=64 time=0.082 ms
^C
--- 192.168.0.165 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2052ms
rtt min/avg/max/mdev = 0.082/0.139/0.253/0.080 ms
```

📌 도커 컨테이너 → br-test

```
130e5e69f768:~# ping 192.168.0.101
PING 192.168.0.101 (192.168.0.101) 56(84) bytes of data.
64 bytes from 192.168.0.101: icmp_seq=1 ttl=64 time=0.081 ms
64 bytes from 192.168.0.101: icmp_seq=2 ttl=64 time=0.084 ms
64 bytes from 192.168.0.101: icmp_seq=3 ttl=64 time=0.082 ms
^C
--- 192.168.0.101 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2074ms
rtt min/avg/max/mdev = 0.081/0.082/0.084/0.001 ms
```

📌 도커 컨테이너 → namespace 컨테이너

```
130e5e69f768:~# ping 192.168.0.108
PING 192.168.0.108 (192.168.0.108) 56(84) bytes of data.
64 bytes from 192.168.0.108: icmp_seq=1 ttl=64 time=0.105 ms
64 bytes from 192.168.0.108: icmp_seq=2 ttl=64 time=0.199 ms
64 bytes from 192.168.0.108: icmp_seq=3 ttl=64 time=0.093 ms
^C
--- 192.168.0.108 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2055ms
rtt min/avg/max/mdev = 0.093/0.129/0.105/0.040 ms
```

테스트 결과

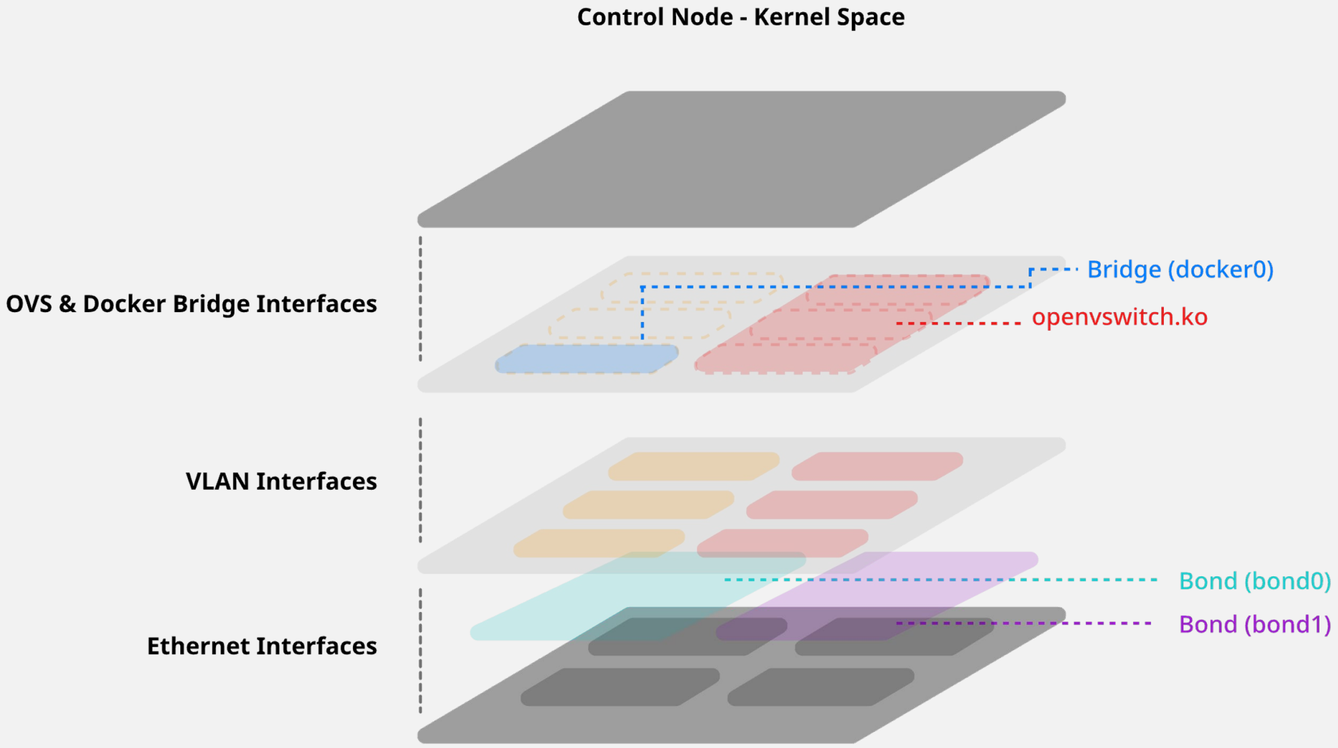
- 호스트 내부에서 통신이 잘 된다.

4. 한계점 및 개선 요소

- 여러 개의 브릿지, veth, 그 외 가상 인터페이스를 사용하다보니 이에 대한 오버헤드가 존재한다.
  - 처리가 느려질 수 있다.
  - 추후, 이를 개선해야할 여지가 있다.
- 오픈스택과 OVS가 없는 환경에서 테스트를 진행하였다.
  - 컨테이너간 통신에서는 잘 되는 것이 확인되었으나 인스턴스와 컨테이너간 통신이 잘 되는지 확인이 필요하다.
  - OVS가 브릿지를 인터페이스로 점유하였을 때, 문제없이 동작하는지 확인이 필요하다.
  - 결국, 오픈스택이 배포된 환경에서 테스트를 진행해볼 필요가 있다.
- 인스턴스 환경에서 테스트 해본 것으로 인스턴스 외부로 패킷이 나가고 들어오는 테스트를 진행해야 한다.
- Shim 인터페이스 등 CLI 환경에서 조작이 필요한 경우가 있다.
  - systemd로 등록이 필요할 것 같다.

C. 논리망 기반 서비스환경 구성

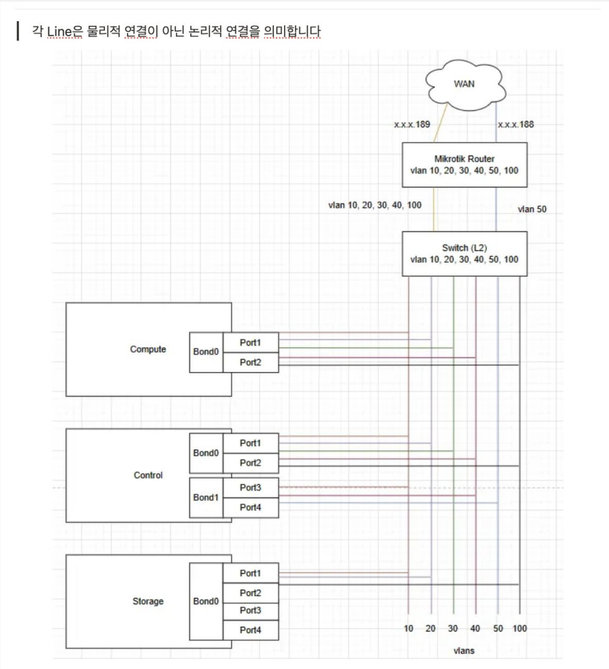
#Solve2 – Bond 분할구성




```
    dhcp4: no
bonds:
  bond0:
    mtu: 9000
    interfaces:
      - eno1
      - eno2
    dhcp4: no
  bond1:
    mtu: 1500
    interfaces:
      - eno3
      - eno4
    dhcp4: no
vlans:
```

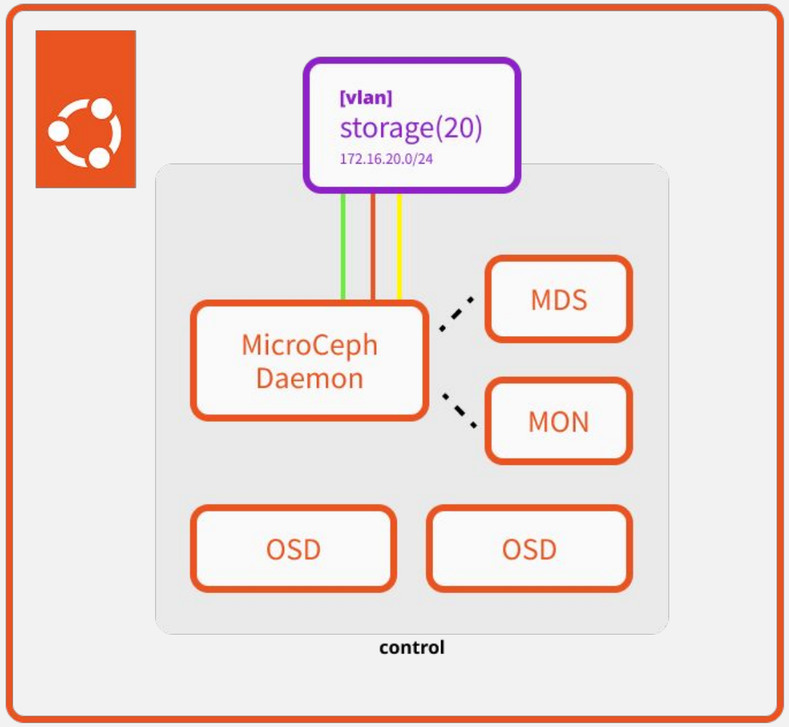
C. 논리망 기반 서비스환경 구성

#Solve2 – Bond 분할구성

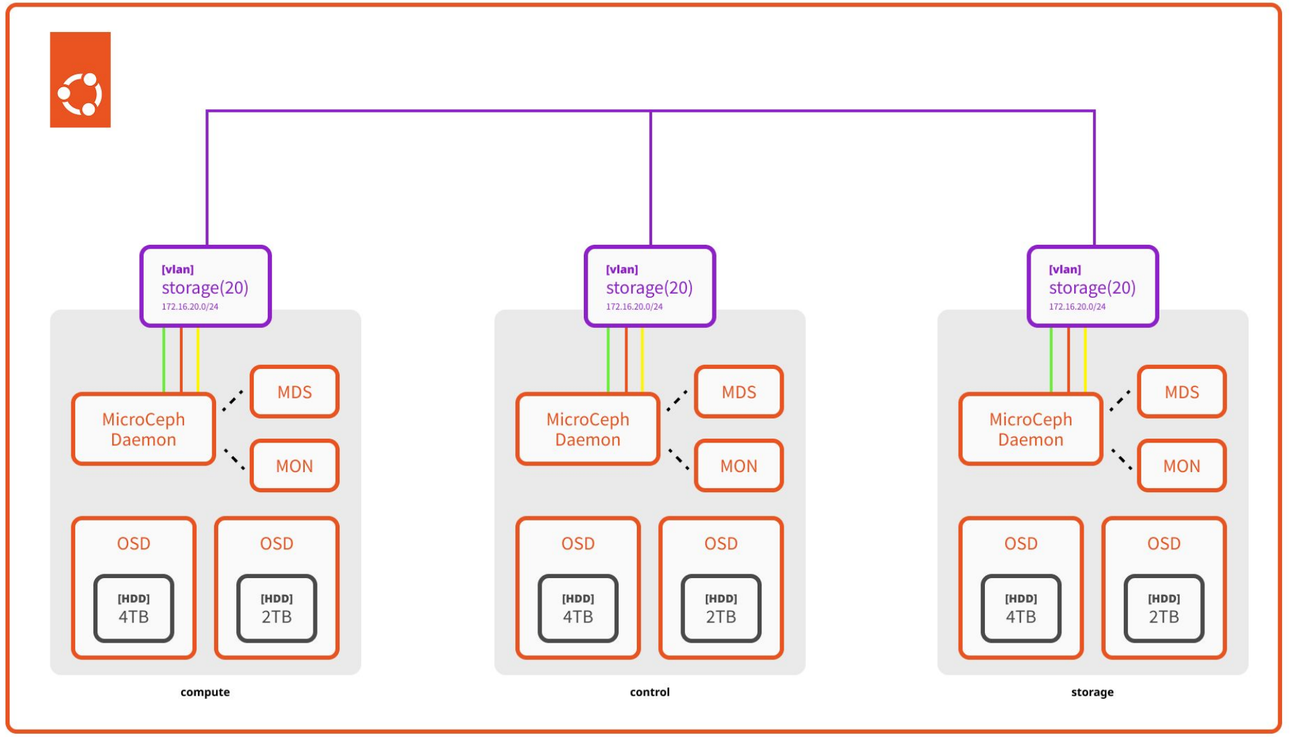


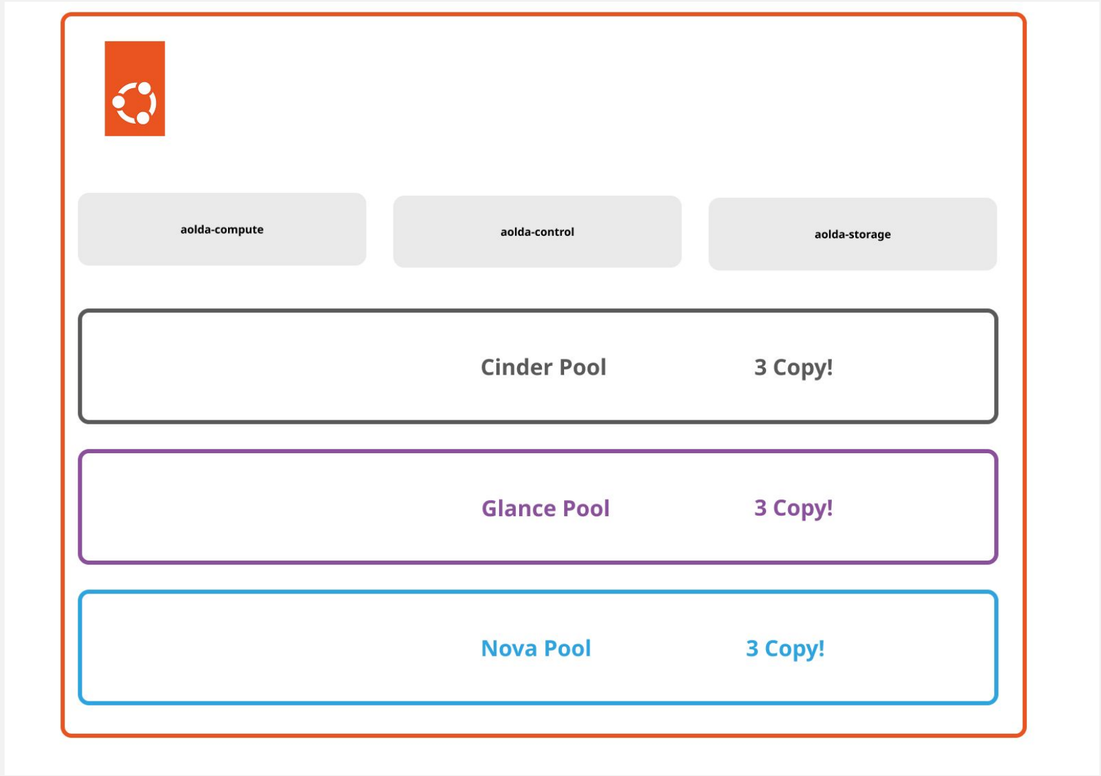
- APM(ITDA) 등 아올다 서비스 컨테이너는 Control 노드의 Bond1 인터페이스와 연결됩니다.
  - 아올다 서비스 컨테이너는 macvlan을 사용하여 컨테이너별 IP, MAC 주소를 갖도록 합니다.
  - 컨테이너가 개별 MAC 주소를 갖게 되면서 스위치를 통해 컨테이너와 노드간 통신이 가능해집니다.
    - 스위치 단에서 MAC주소를 보고 각 컨테이너 혹은 노드로 프레임을 전송합니다.
    - 물리 인터페이스 및 본딩 인터페이스를 분리하고 개별 MAC 주소를 가져 Hairpin 문제가 해결됩니다.

```
external:
  name: macvlan-external
  driver: macvlan
  driver_opts:
    parent: vlan-external-d
  ipam:
    config:
```



정보 유실이 되었기 때문에, 최대한 정보가 유실되지 않도록 하는 것이 목표!


MicroCloud internal subnet
Ceph public traffic
Ceph internal (cluster) traffic
MicroCeph service traffic
[vlan]
storage(20)
172.16.20.0/24
MicroCeph Daemon
MDS
MON
OSD
[HDD]
4TB
[HDD]
2TB
compute
control
storage



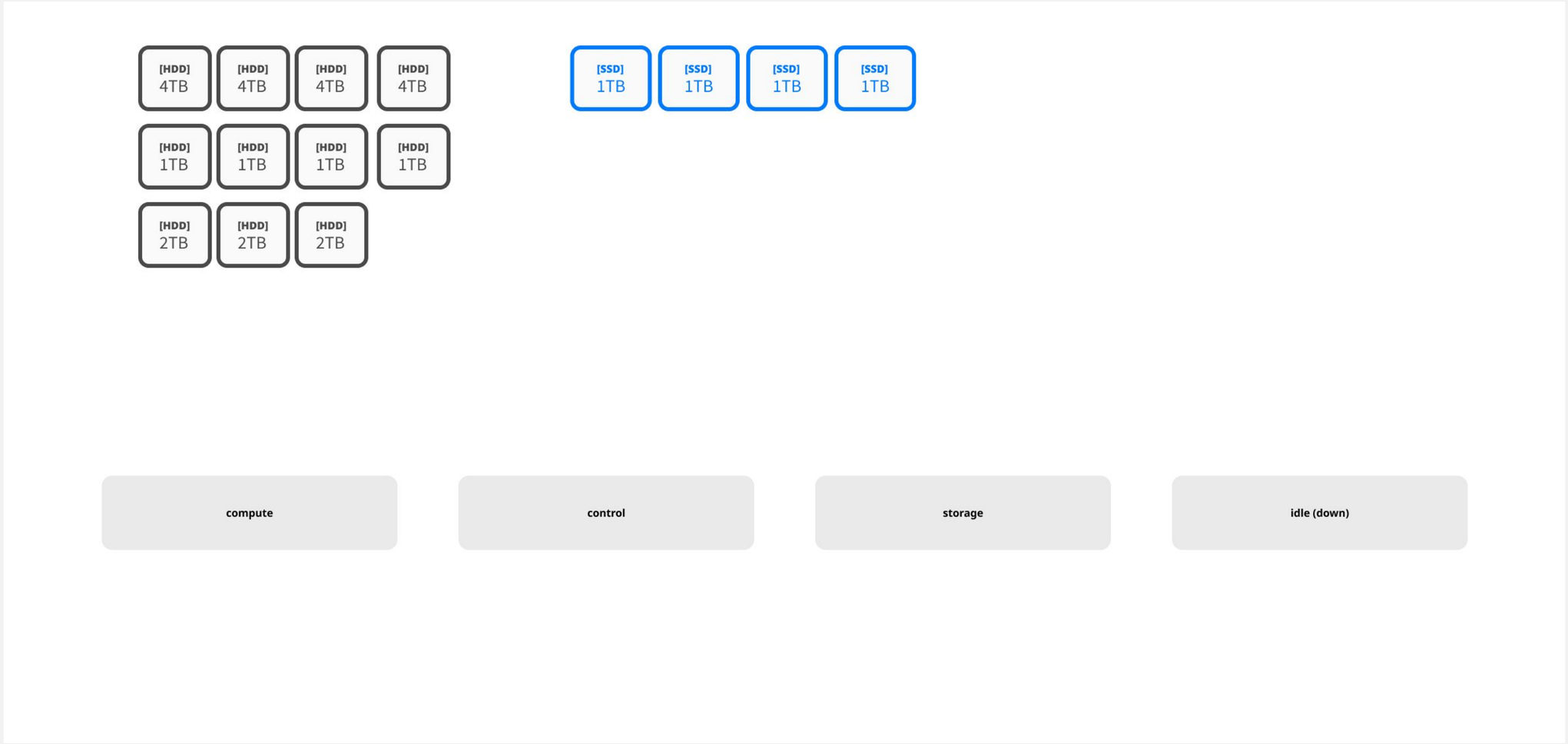
RULES

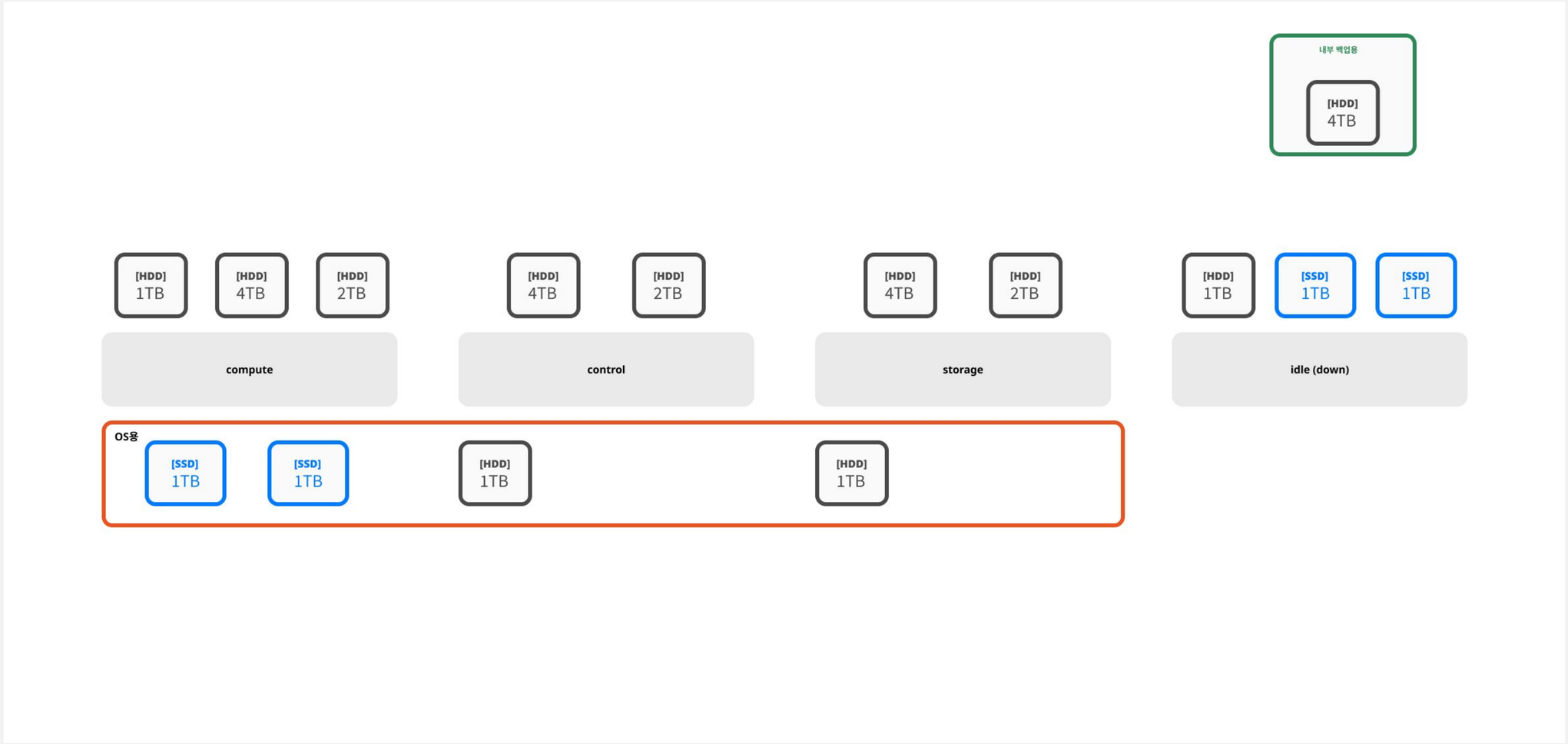
Rules define policy about how data is distributed across the devices in the hierarchy.

CRUSH rules define placement and replication strategies or distribution policies that allow you to specify exactly how CRUSH places object replicas. For example, you might create a rule selecting a pair of targets for 2-way mirroring, another rule for selecting three targets in two different data centers for 3-way mirroring, and yet another rule for erasure coding over six storage devices. For a detailed discussion of CRUSH rules, refer to CRUSH - Controlled, Scalable, Decentralized Placement of Replicated Data, and more specifically to **Section 3.2**.

최대한 OSD를 잘 분배해서 복제본이 노드마다 하나씩 존재할 수 있도록 물리적 인프라 구성 필요

microceph_auto_osd, microceph_auth_host

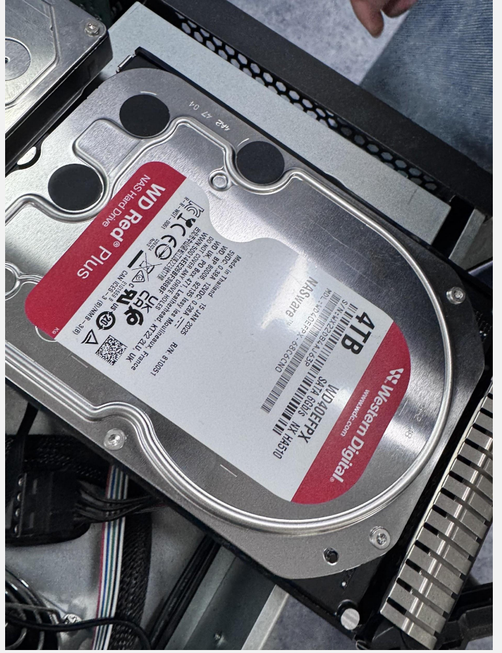

[HDD] 4TB
[HDD] 4TB
[HDD] 4TB
[HDD] 4TB
[HDD] 1TB
[HDD] 1TB
[HDD] 1TB
[HDD] 1TB
[HDD] 2TB
[HDD] 2TB
[HDD] 2TB
[SSD] 1TB
[SSD] 1TB
[SSD] 1TB
[SSD] 1TB
compute
control
storage
idle (down)


내부 백업용
[HDD]
4TB
[HDD]
1TB
[HDD]
4TB
[HDD]
2TB
[HDD]
4TB
[HDD]
2TB
[HDD]
4TB
[HDD]
2TB
[HDD]
1TB
[SSD]
1TB
[SSD]
1TB
compute
control
storage
idle (down)
OS용
[SSD]
1TB
[SSD]
1TB
[HDD]
1TB
[HDD]
1TB

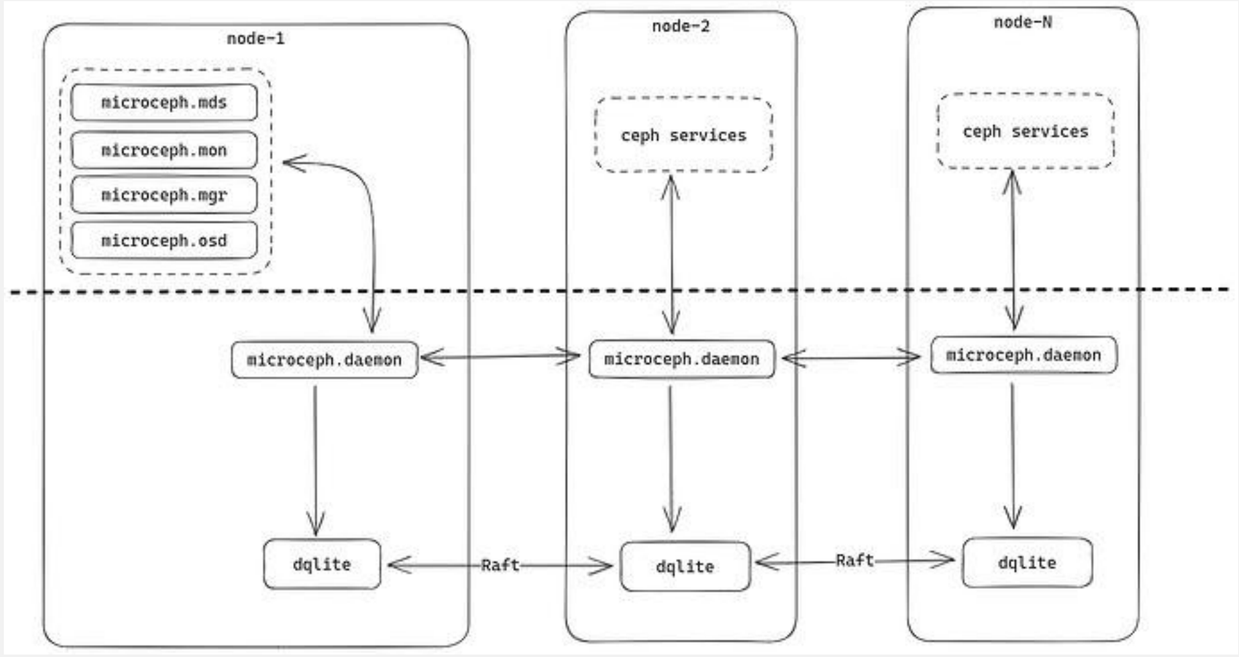
물리 인프라 구성


Fan Cage Handle


LFF Drives Box #2


WD Red Plus
4TB
Western Digital

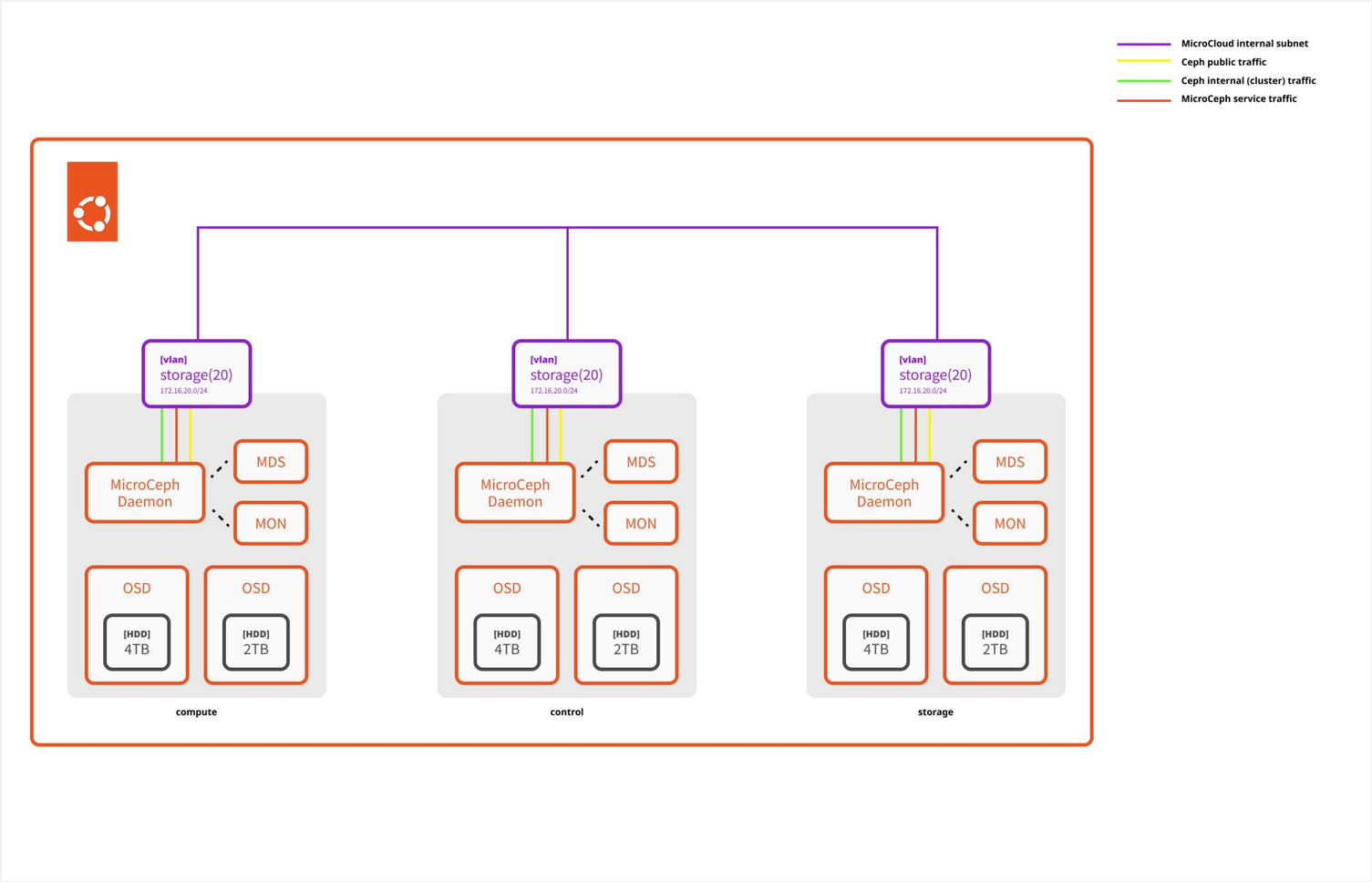

Hewlett Packard Enterprise
PM1633a 960GB SAS 12 Gbps
HPE Model No: VO000960JWDAT
Model: MZ-ILS960N
HPE P/N: 867887-002

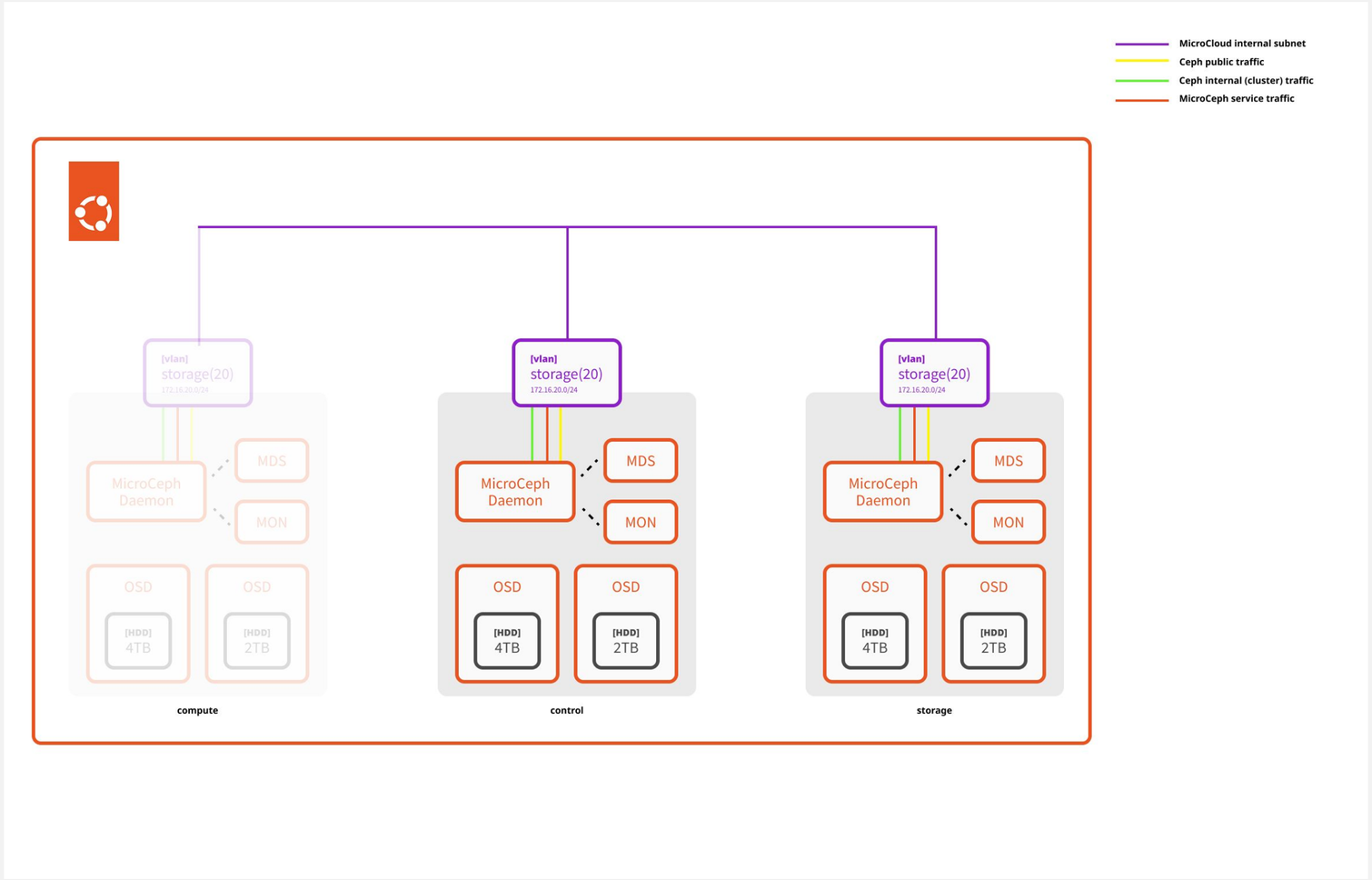


[스토리지] 5. Pool 설정 및 장애 상황 테스트	ANC (Aolda Next-infrast...	2025년 7월 26일 오후 7:36	찬주 이
[스토리지] 4. Cluster 네트워크 설정 및 RGW, CephFs 추가	ANC (Aolda Next-infrast...	2025년 7월 26일 오후 1:59	찬주 이
[스토리지] 3. RBD Client Cache	ANC (Aolda Next-infrast...	2025년 7월 26일 오후 12:42	찬주 이
[스토리지] 2. Microceph으로 Multi Node 클러스터 구축	ANC (Aolda Next-infrast...	2025년 7월 26일 오전 11:25	찬주 이

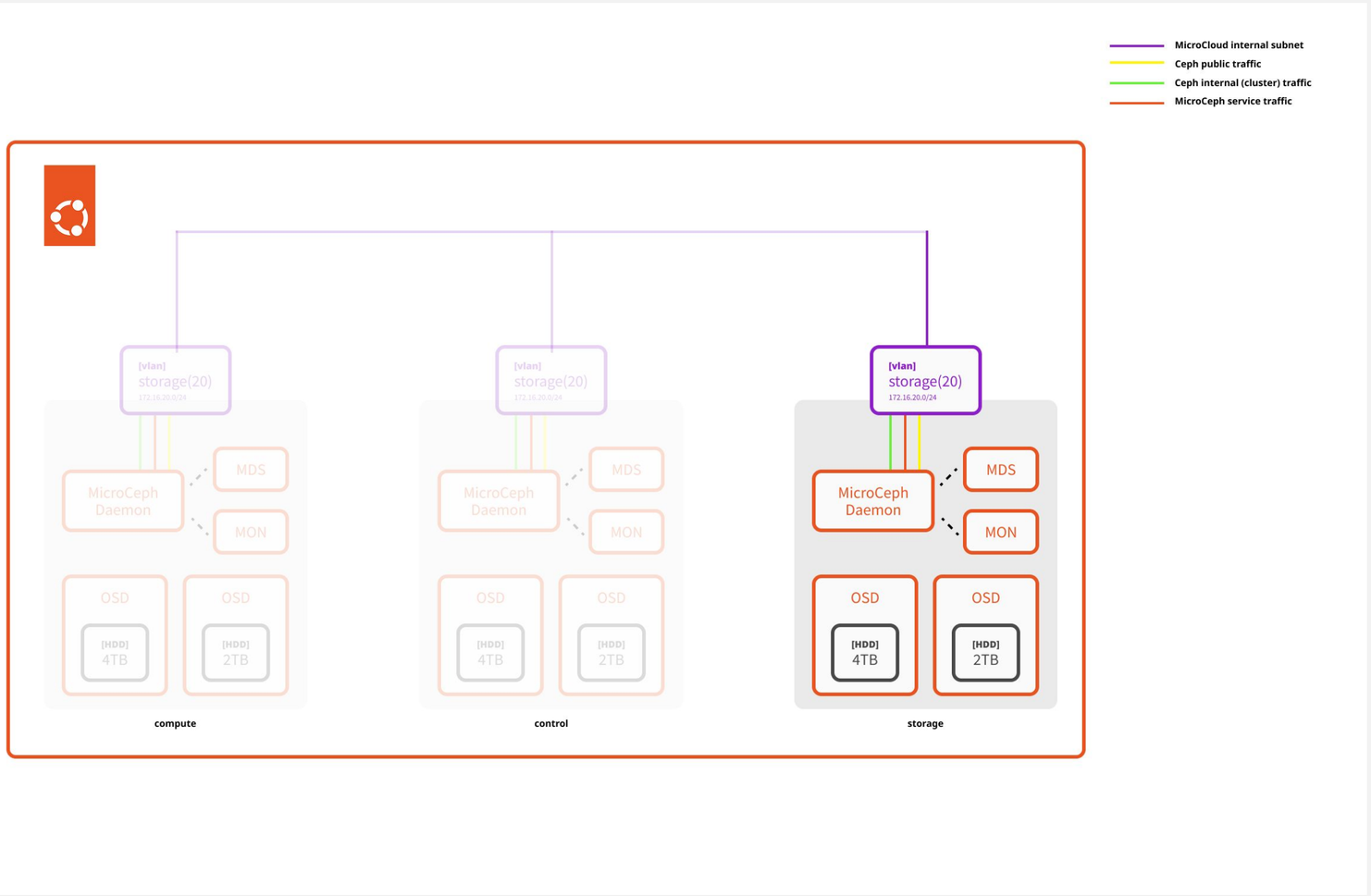
전체 클라우드가 그리 크지 않아 복잡한 설정값을 할 필요가 적고
MicroCeph이 설정값이 더 간편하다
 표준화된 설정으로 운영 리스크를 최소화하여
후대에 쉽게 이어질 수 있음

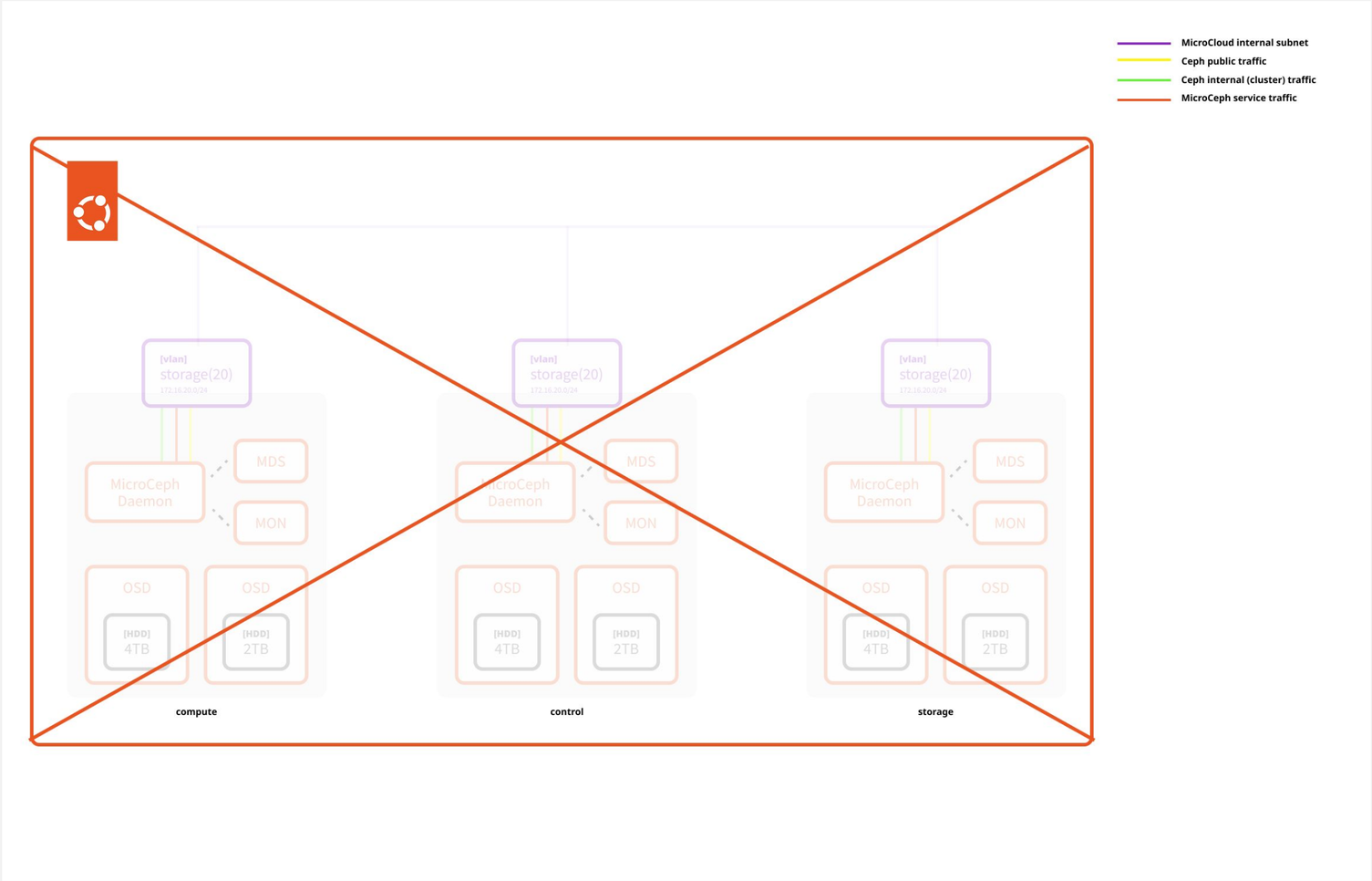
필요시 Ceph으로 마이그레이션을 할 수 있음


MicroCloud internal subnet
Ceph public traffic
Ceph internal (cluster) traffic
MicroCeph service traffic
[vlan]
storage(20)
172.16.20.0/24
MicroCeph Daemon
MDS
MON
OSD
[HDD]
4TB
[HDD]
2TB
compute
control
storage

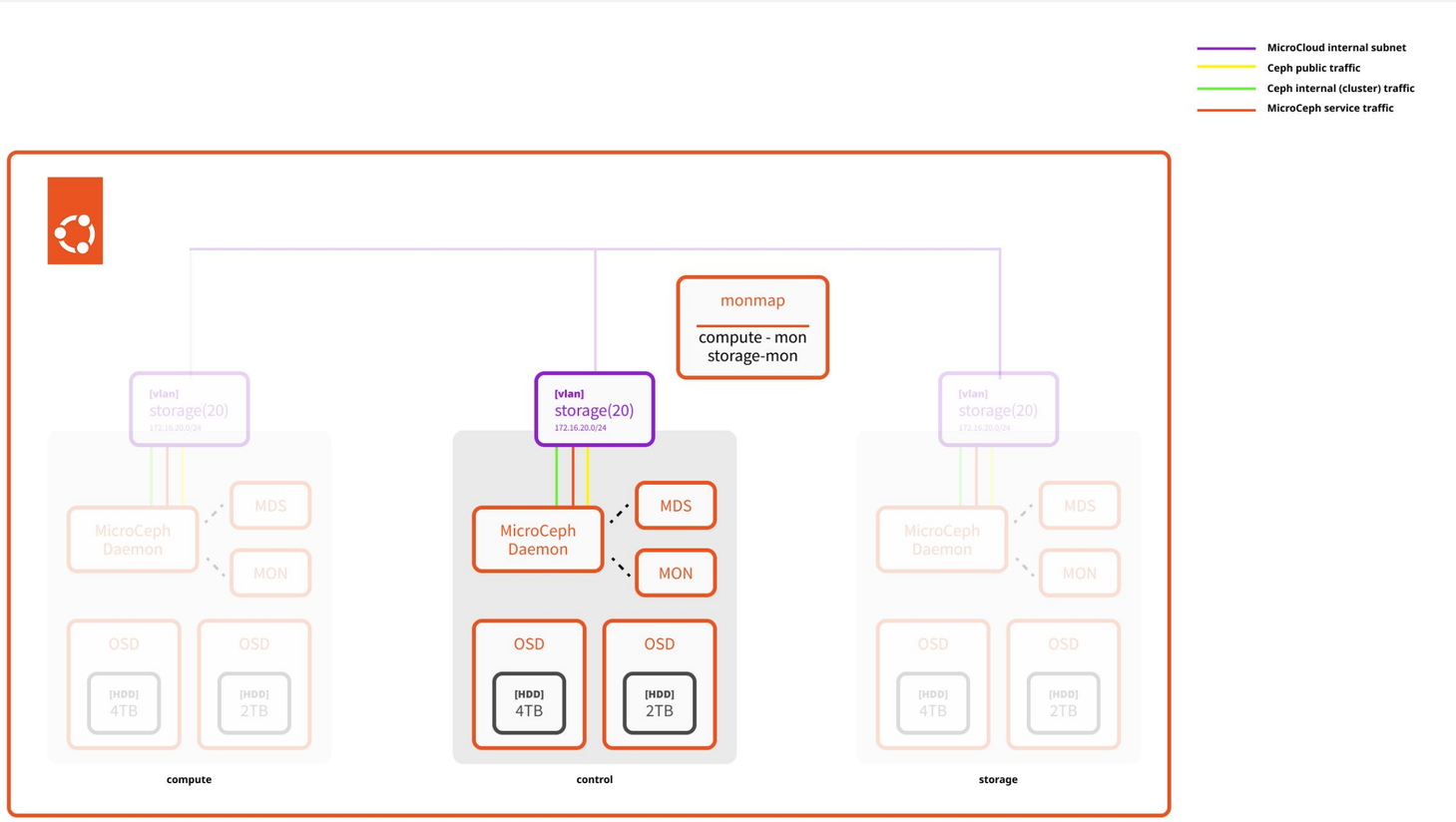


노드 하나 장애에 대해서는 아무런 문제 없음


MicroCloud internal subnet
Ceph public traffic
Ceph internal (cluster) traffic
MicroCeph service traffic
[vlan]
storage(20)
172.16.20.0/24
MicroCeph Daemon
MDS
MON
OSD
[HDD]
4TB
[HDD]
2TB
compute
control
storage

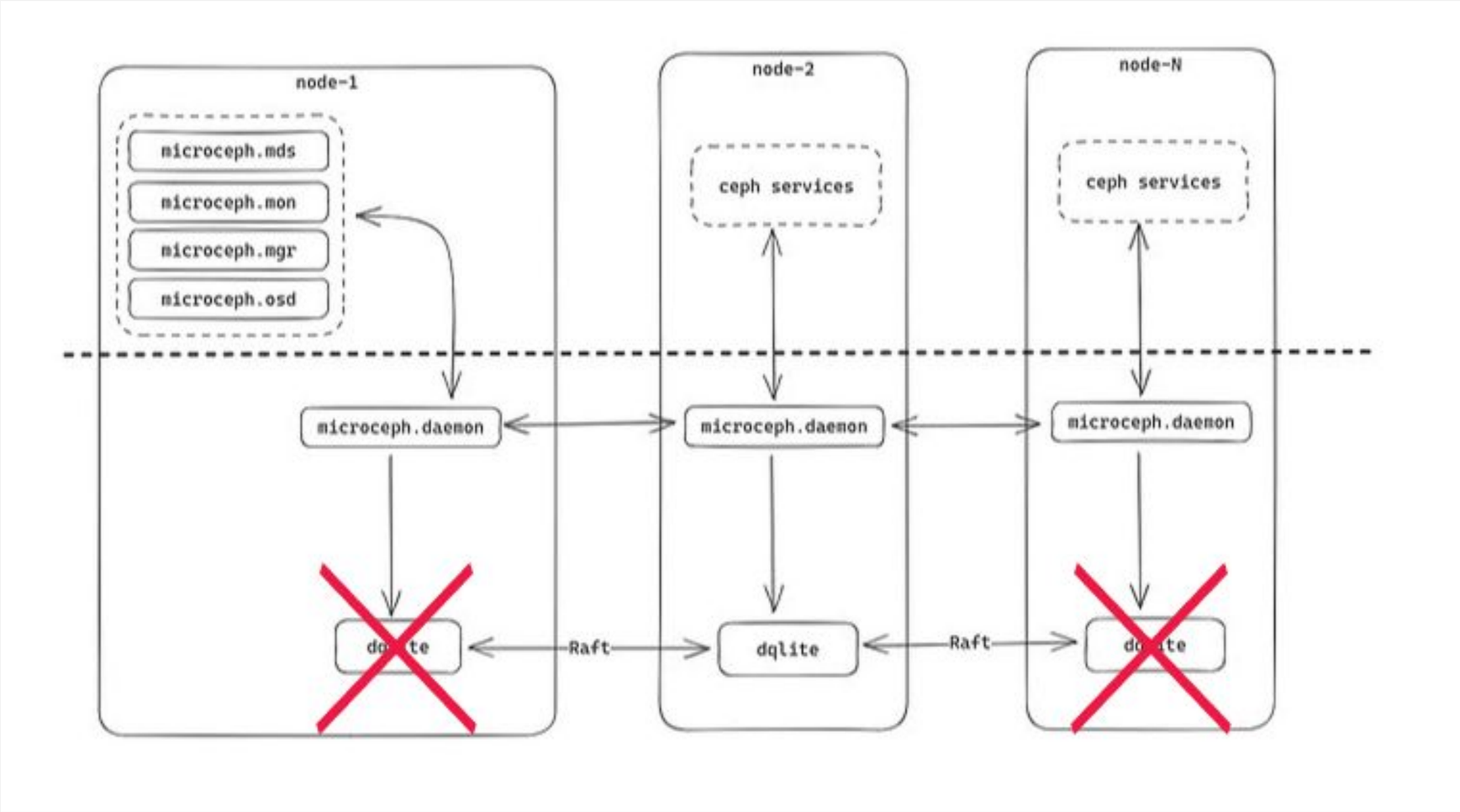


3개 MON 중 2개 노드 장애로 쿼럼 요구사항(과반수) 미충족, 이에 따라 MicroCeph가
데이터 보호를 위해 읽기/쓰기 전면 차단



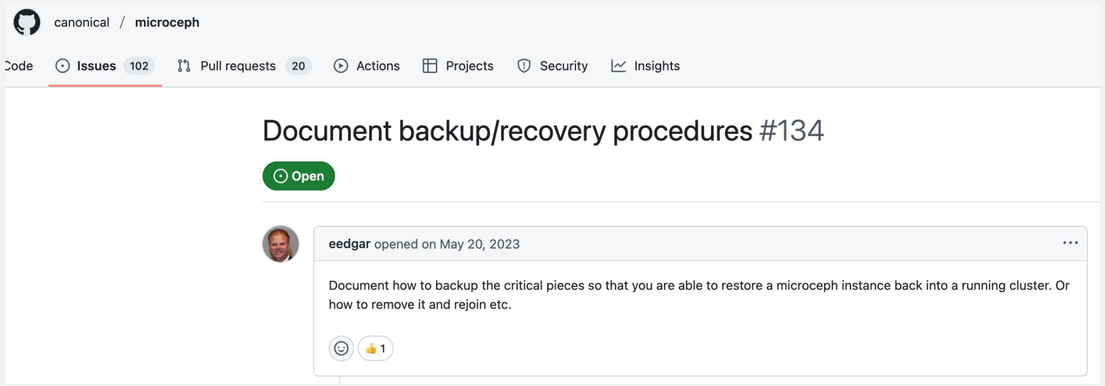
Ceph 클러스터 단일 노드 복구 절차

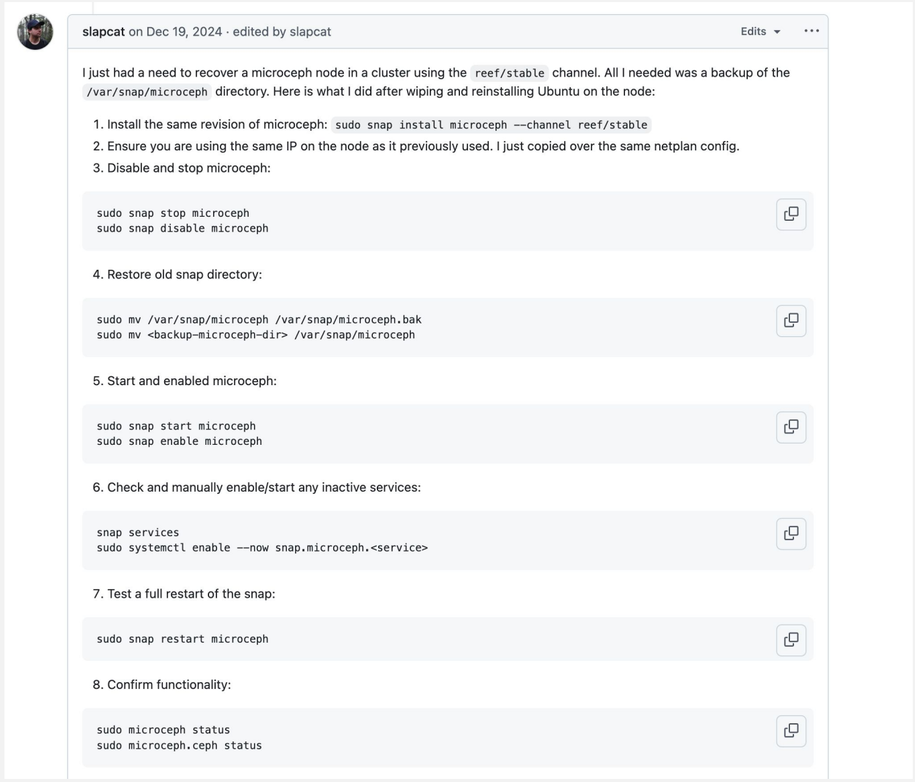
MON 재구성: monmap에서 장애 monitor 제거 후 단일 노드 쿼럼 구성
복제 정책 조정: Pool size를 1로 설정하여 단일 복제본 요구사항 충족
OSD 정리: CRUSH map에서 접근 불가능한 OSD 완전 제거
CRUSH 규칙 수정: chooseleaf_type을 'osd'로 변경하여 단일 호스트 운영 지원



dqlite 클러스터에 대해 2개 노드 장애로 쿼럼 요구사항(과반수) 미충족 문제 발생 □ 제어 플레인 오류
하지만 데이터 플레인은 정상적으로 데이터 복구 가능

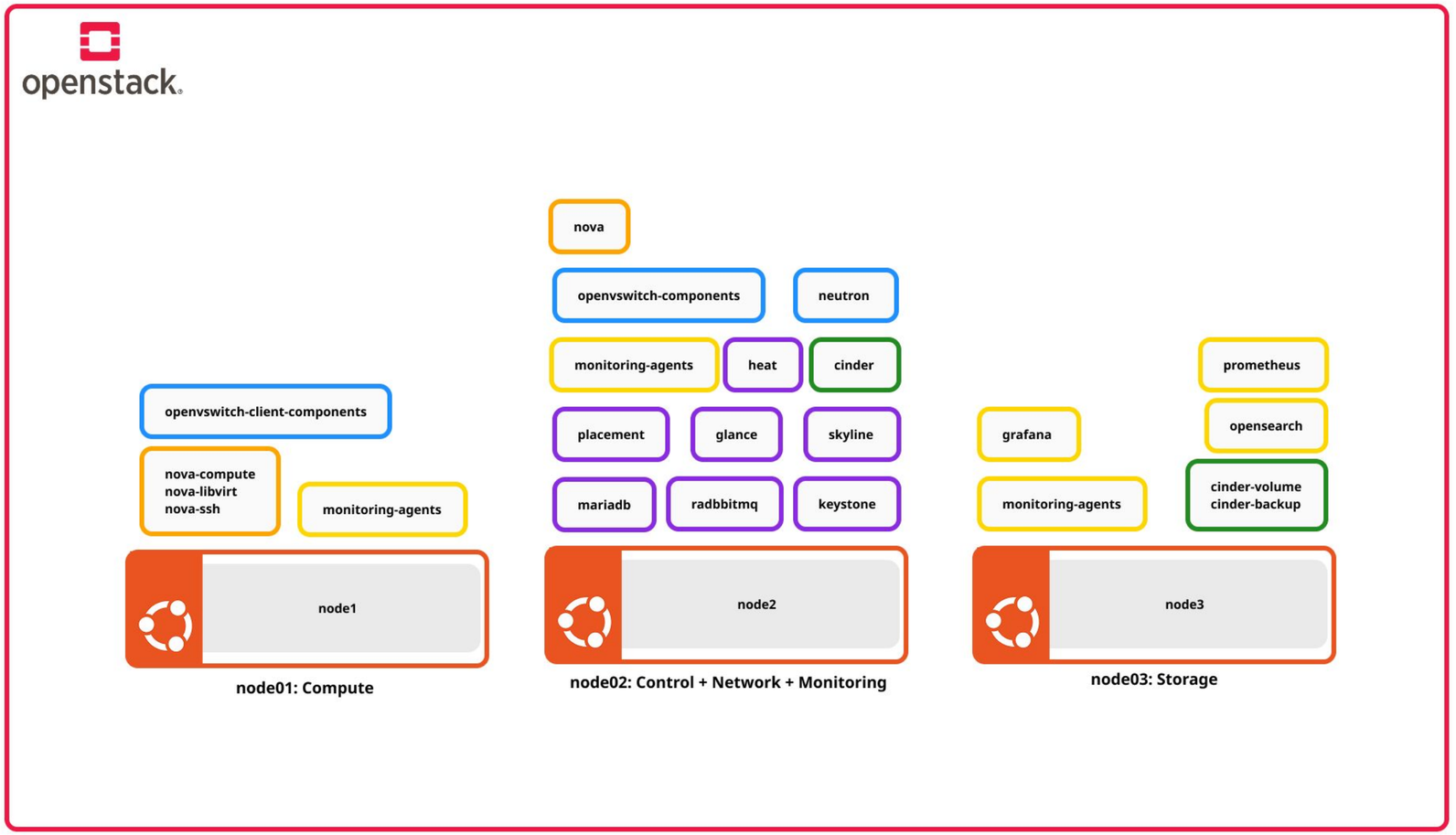
https://github.com/canonical/microceph/issues/134

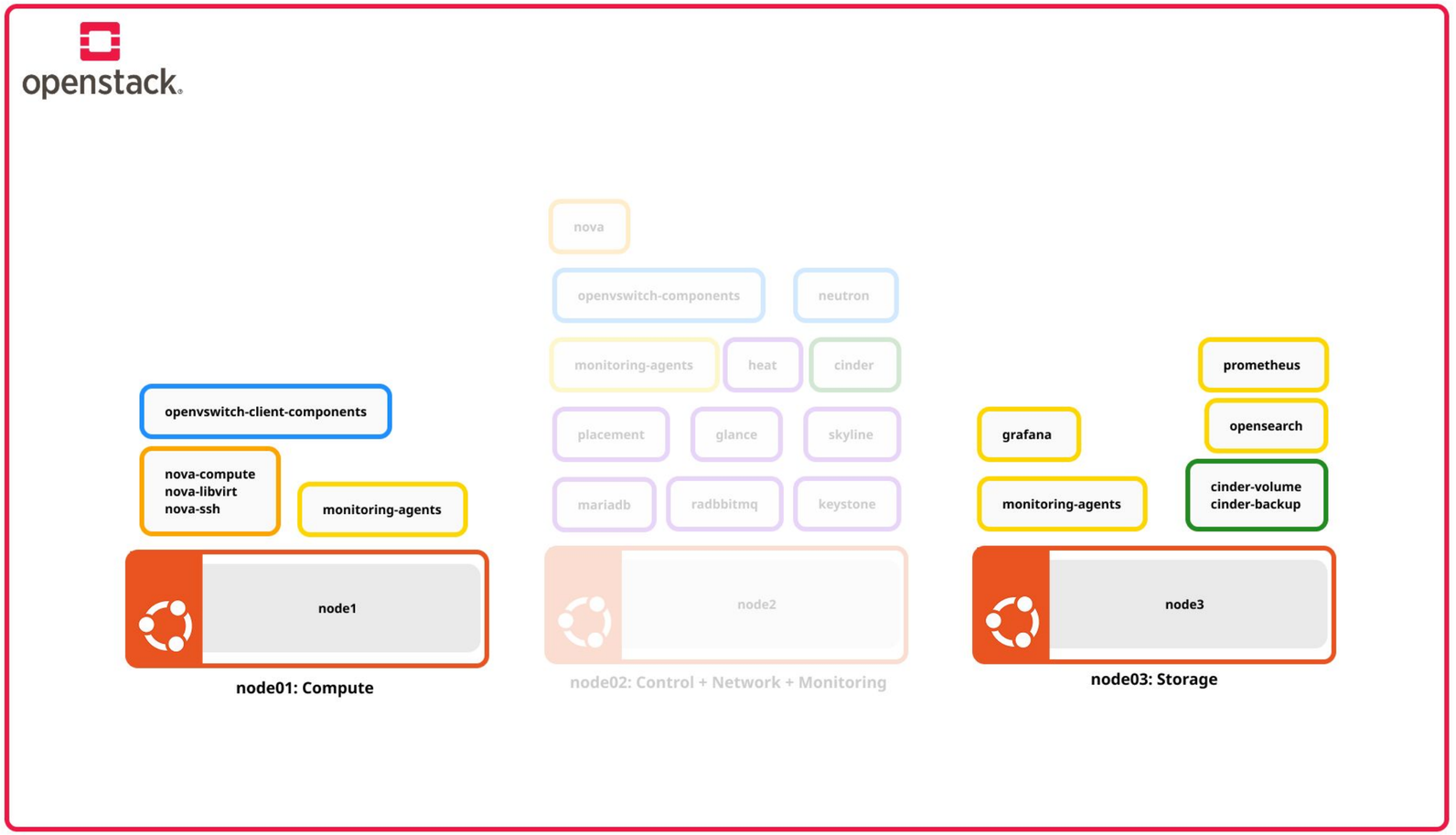


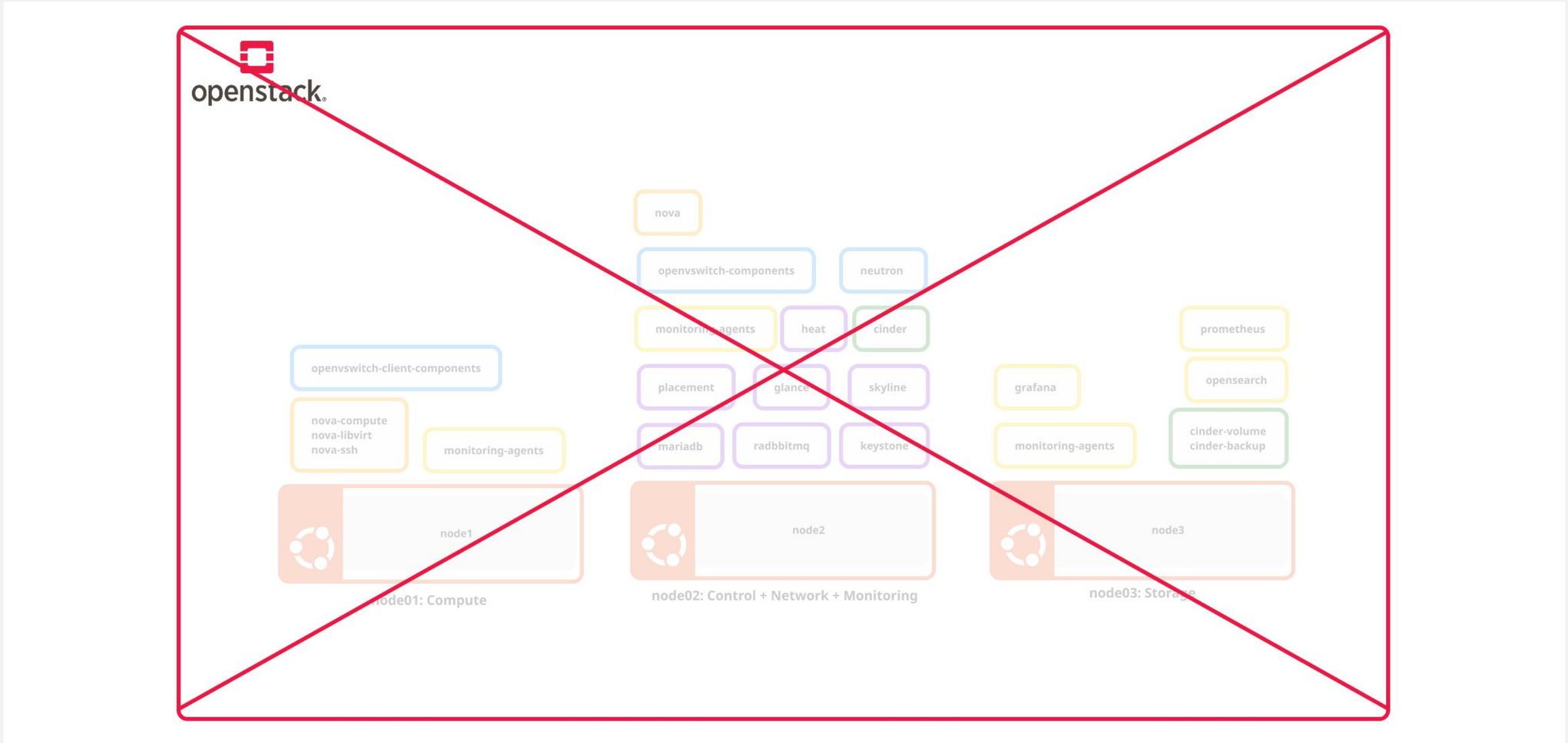


Snap 패키지를 통한 MicroCeph 설치 시

1. /var/snap/microceph 디렉터리 전체를 백업해두면, 노드 장애 후 동일한 IP와 microceph 버전으로 재설치 후 해당 디렉터리를 복원하여 클러스터에 재합류 가능
2.복구 과정에서 snap 서비스 중지/복원/재시작, 서비스 활성화, 상태 확인 등의 단계가 필요함.
3. IP가 변경되면 mon(모니터) 복구가 어려울 수 있음
4. 백업 시 symlink, 소켓 파일 등 특수 파일 타입도 함께 보존해야 함

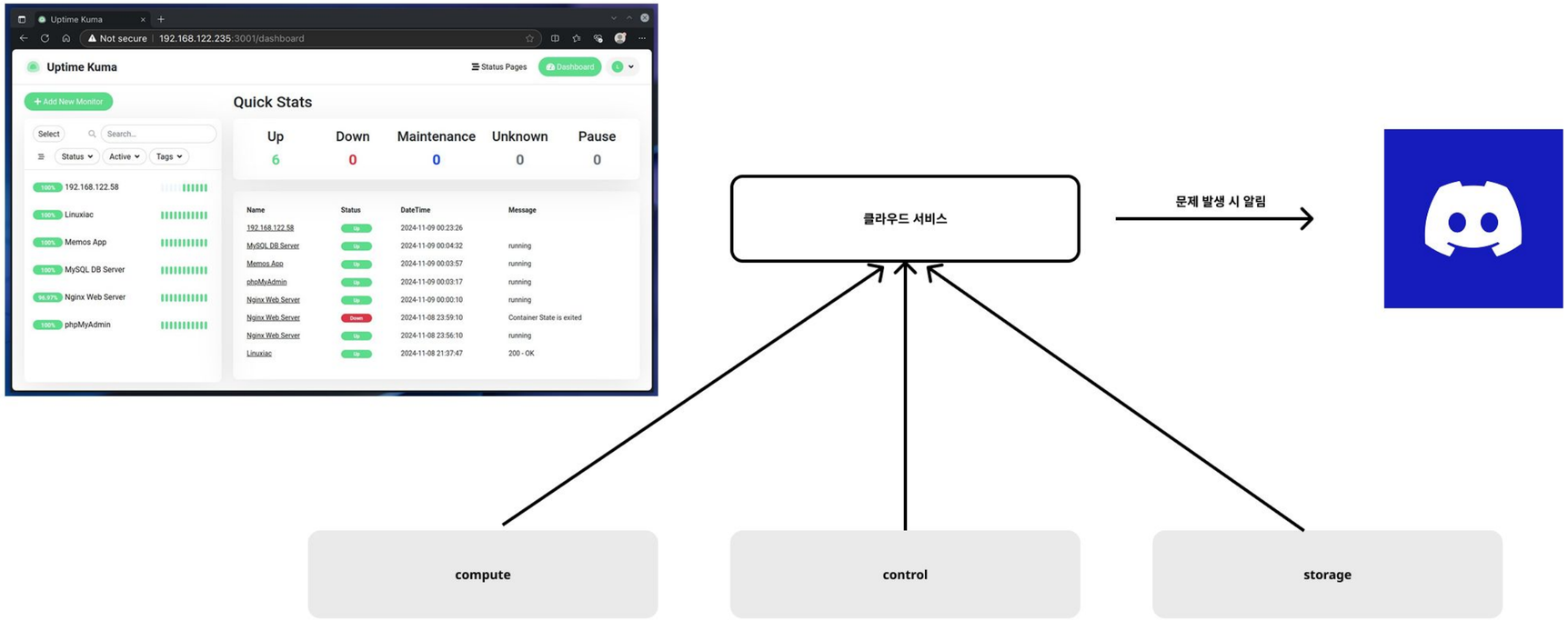

openstack.
openvswitch-client-components
nova-compute
nova-libvirt
nova-ssh
monitoring-agents
node1
node01: Compute
nova
openvswitch-components
neutron
monitoring-agents
heat
cinder
placement
glance
skyline
mariadb
radbbitmq
keystone
node2
node02: Control + Network + Monitoring
prometheus
opensearch
grafana
cinder-volume
cinder-backup
monitoring-agents
node3
node03: Storage


openstack.
openvswitch-client-components
nova-compute
nova-libvirt
nova-ssh
monitoring-agents
node1
node01: Compute
nova
openvswitch-components
neutron
monitoring-agents
heat
cinder
placement
glance
skyline
mariadb
radbbitmq
keystone
node2
node02: Control + Network + Monitoring
prometheus
opensearch
grafana
cinder-volume
cinder-backup
monitoring-agents
node3
node03: Storage



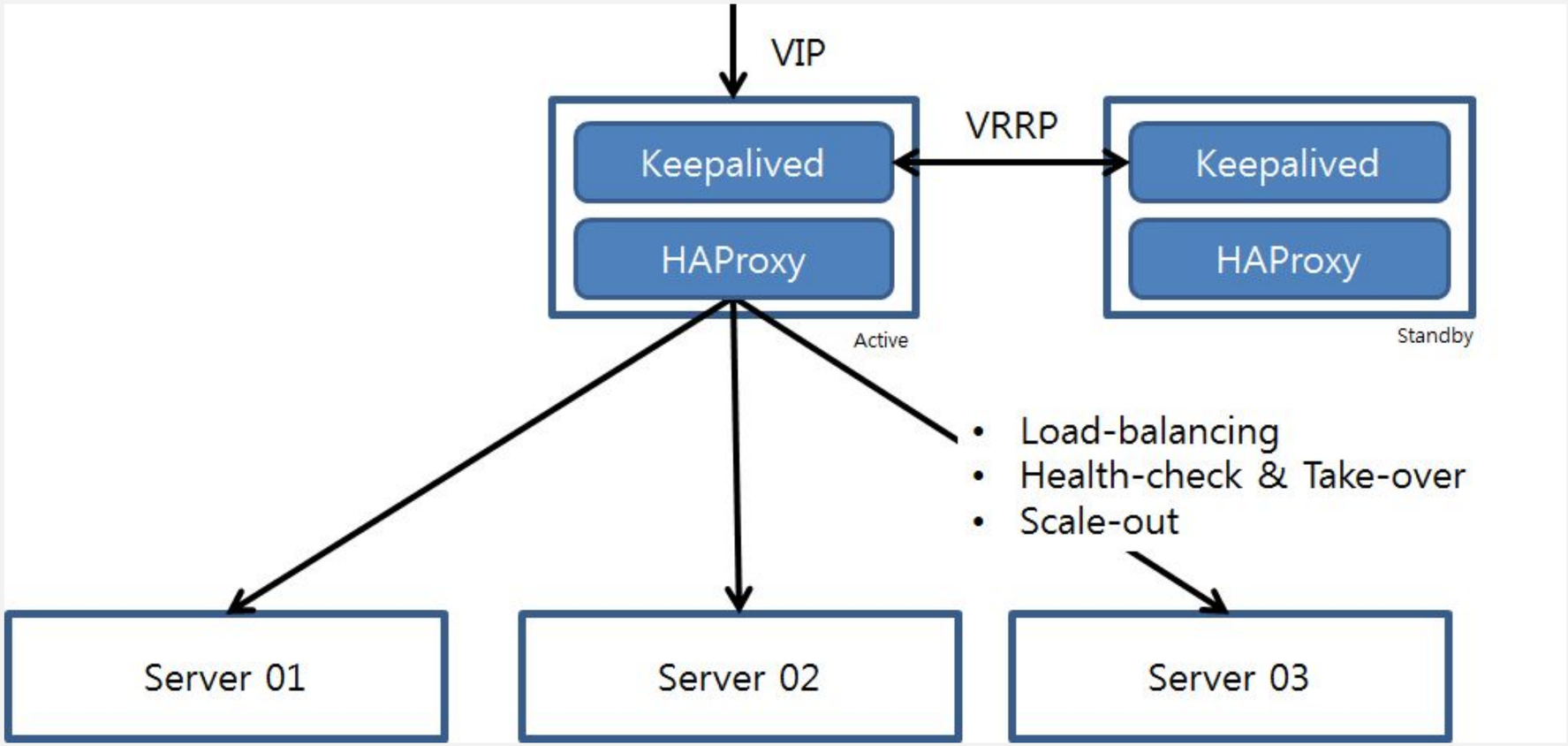
1. 베어메탈 노드에 문제가 생겼음을 관측할 수 없음

2. 컨트롤 노드 장애로 전체 시스템 마비



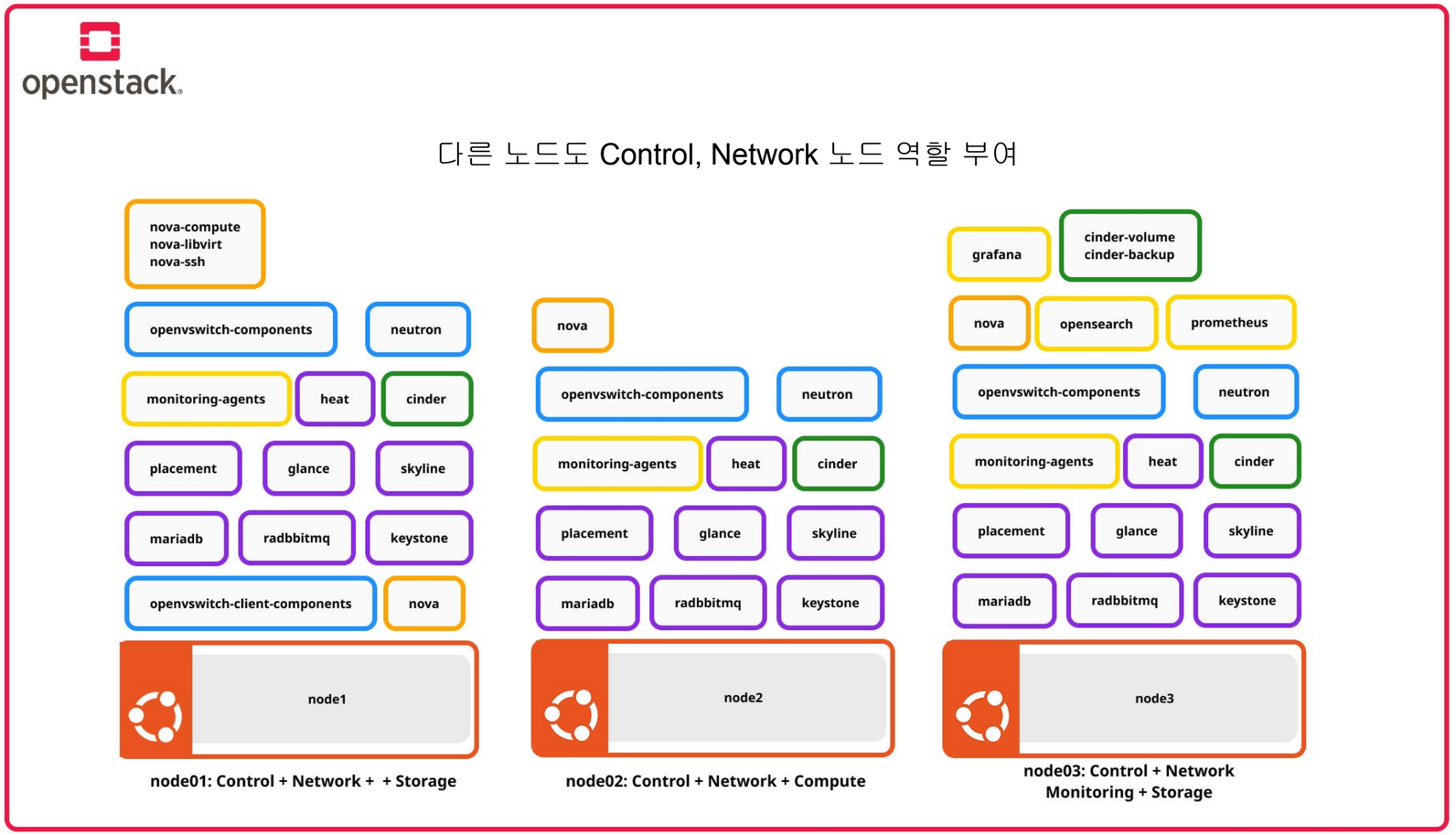
30초에 한번씩 하트비트

1. 베어메탈 노드에 문제가 생겼음을 관측할 수 없음
   □ 클라우드 서비스에서 노드 생존만 모니터링

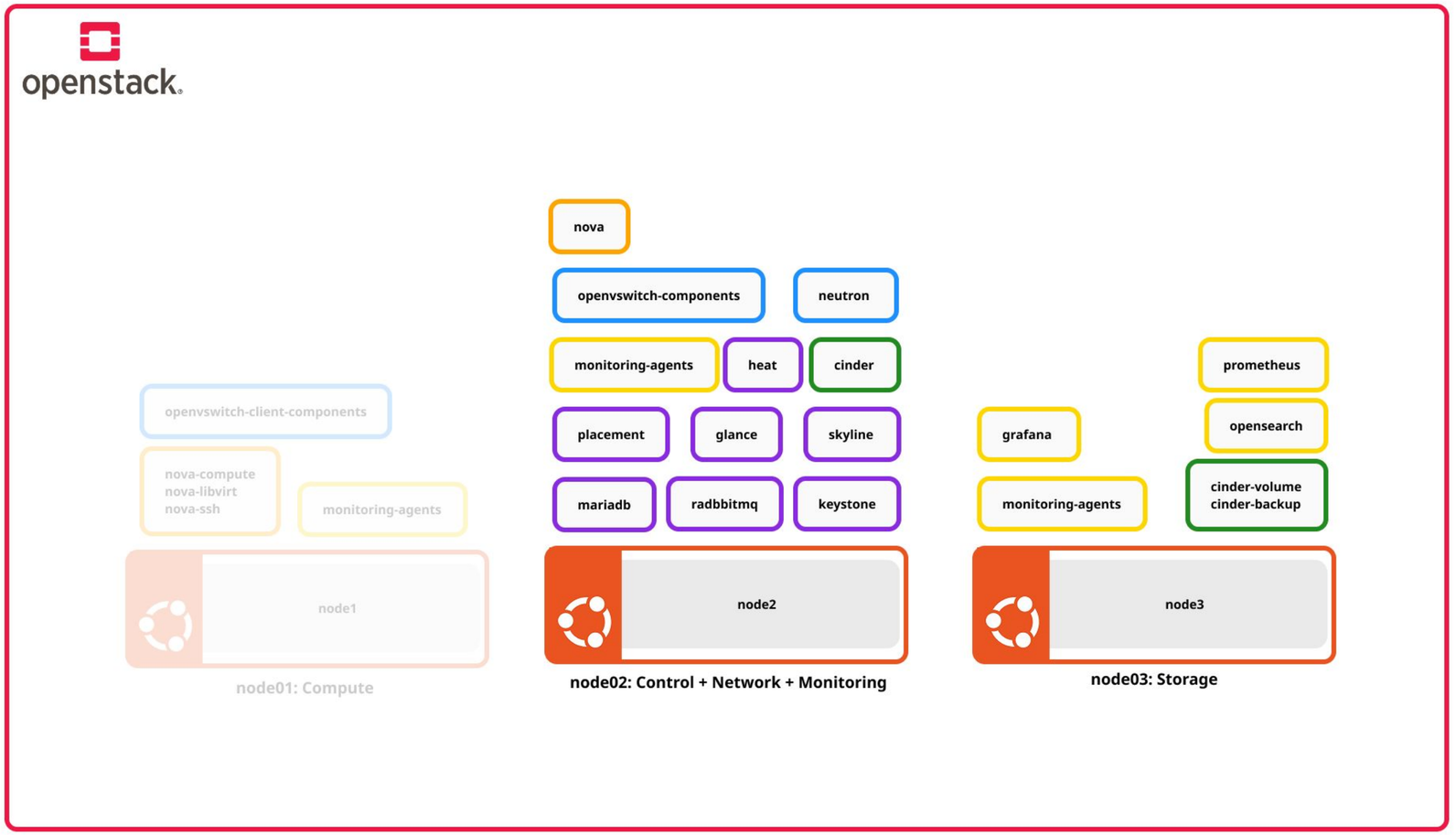


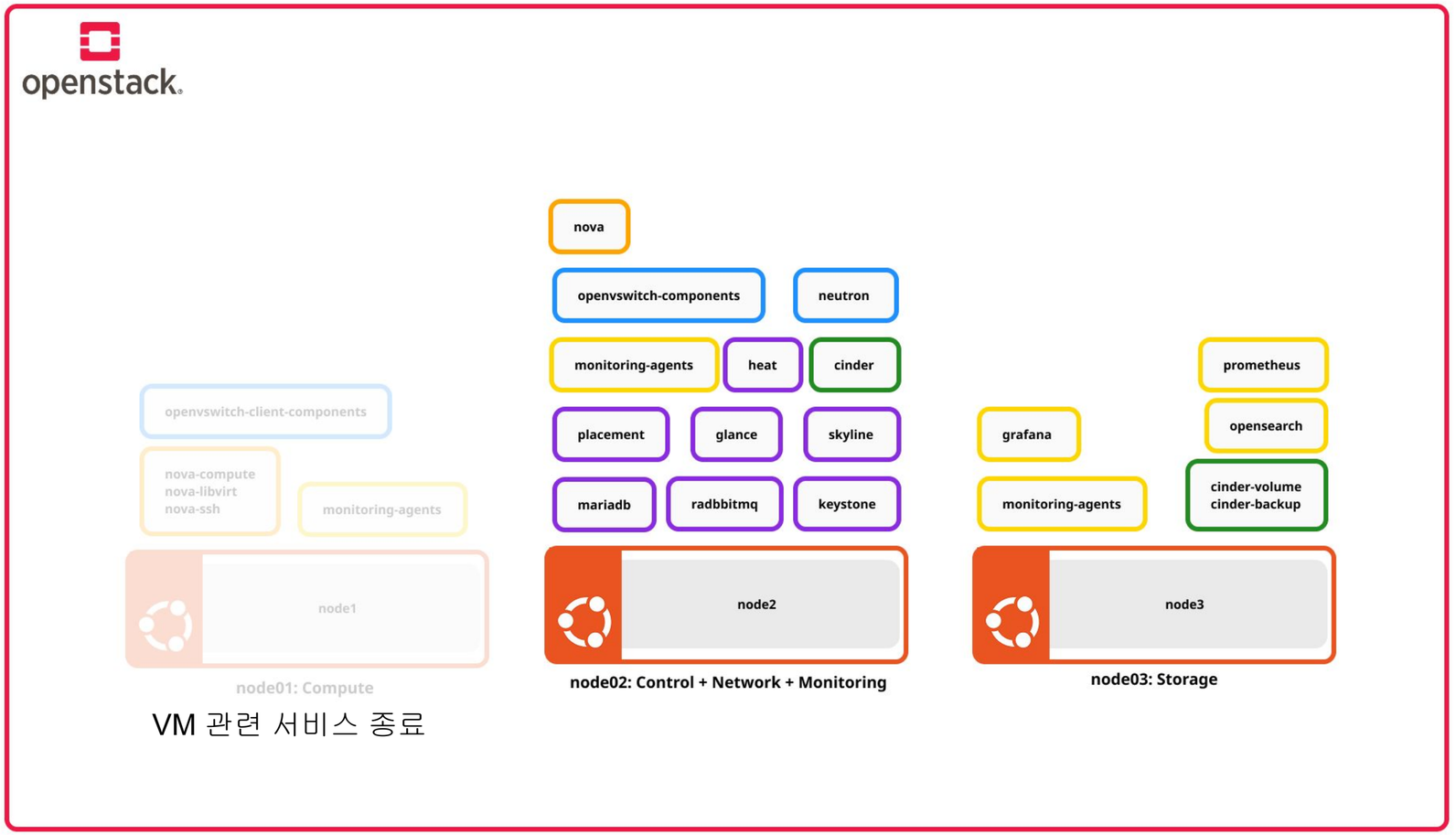
출처: https://dveamer.github.io/architecture/HAProxyAndKeepalived.html

2. 컨트롤 노드 장애로 전체 시스템 마비 □ 다중화

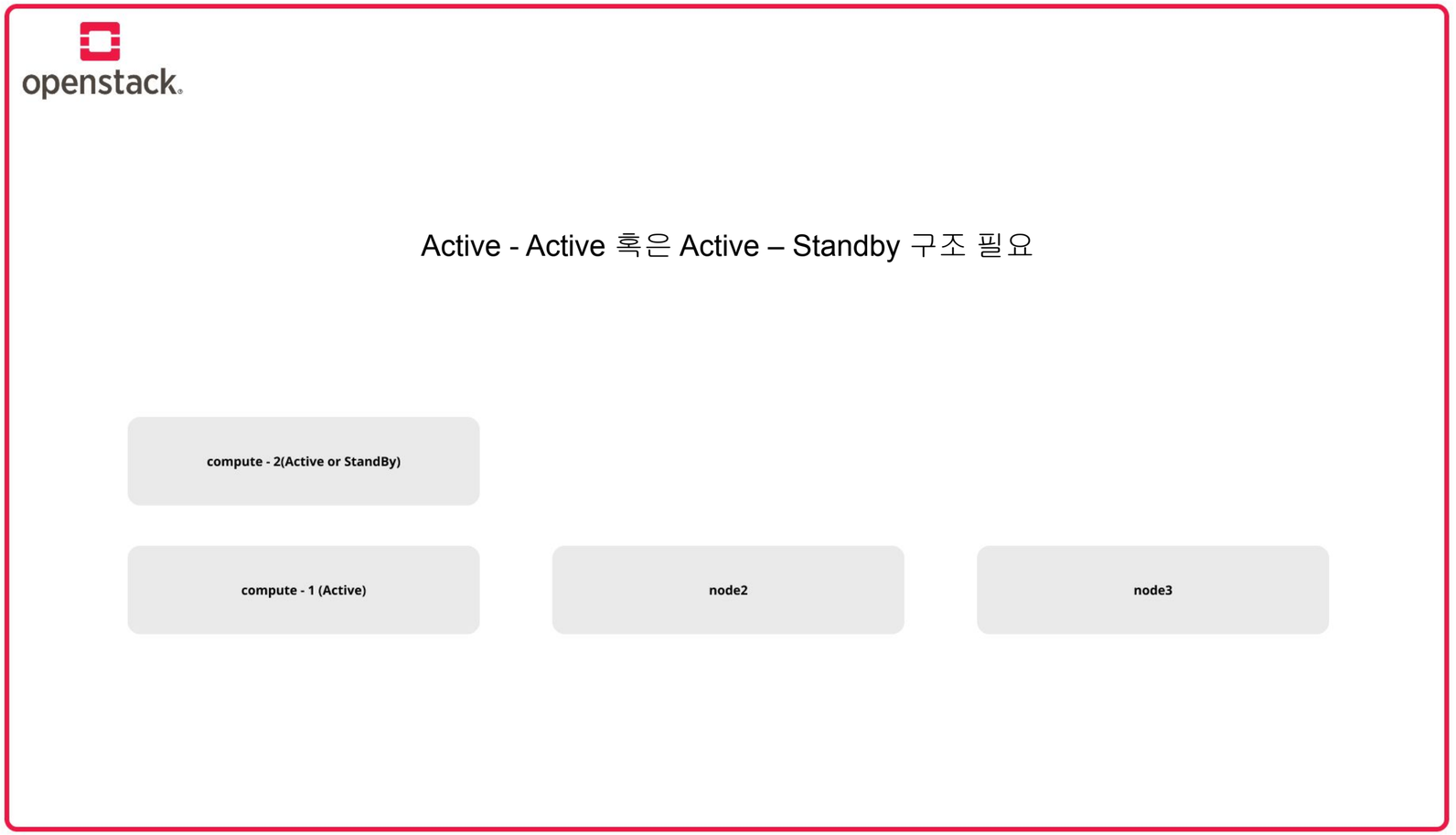


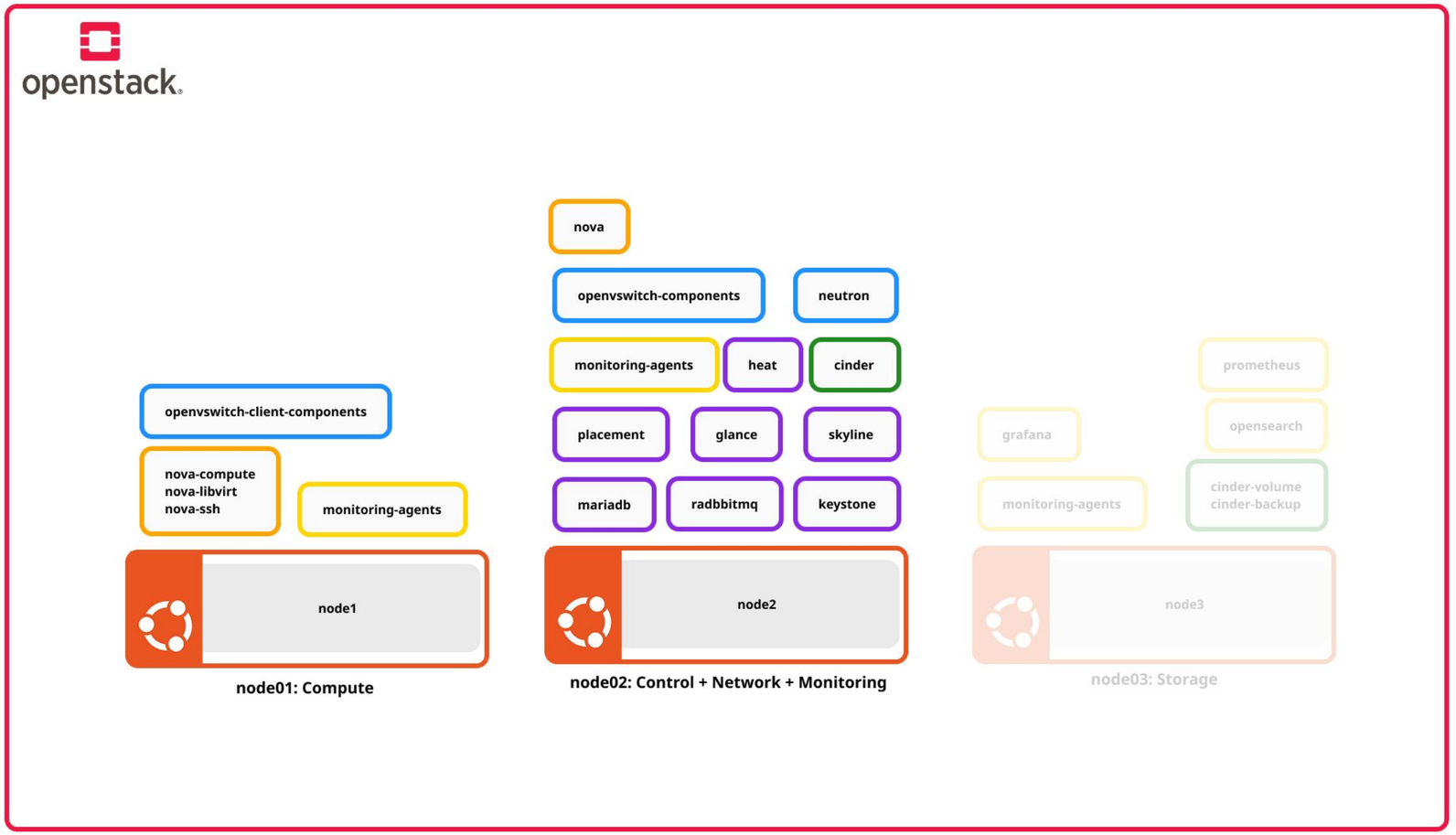
2. 컨트롤 노드 장애로 전체 시스템 마비 □ 다중화

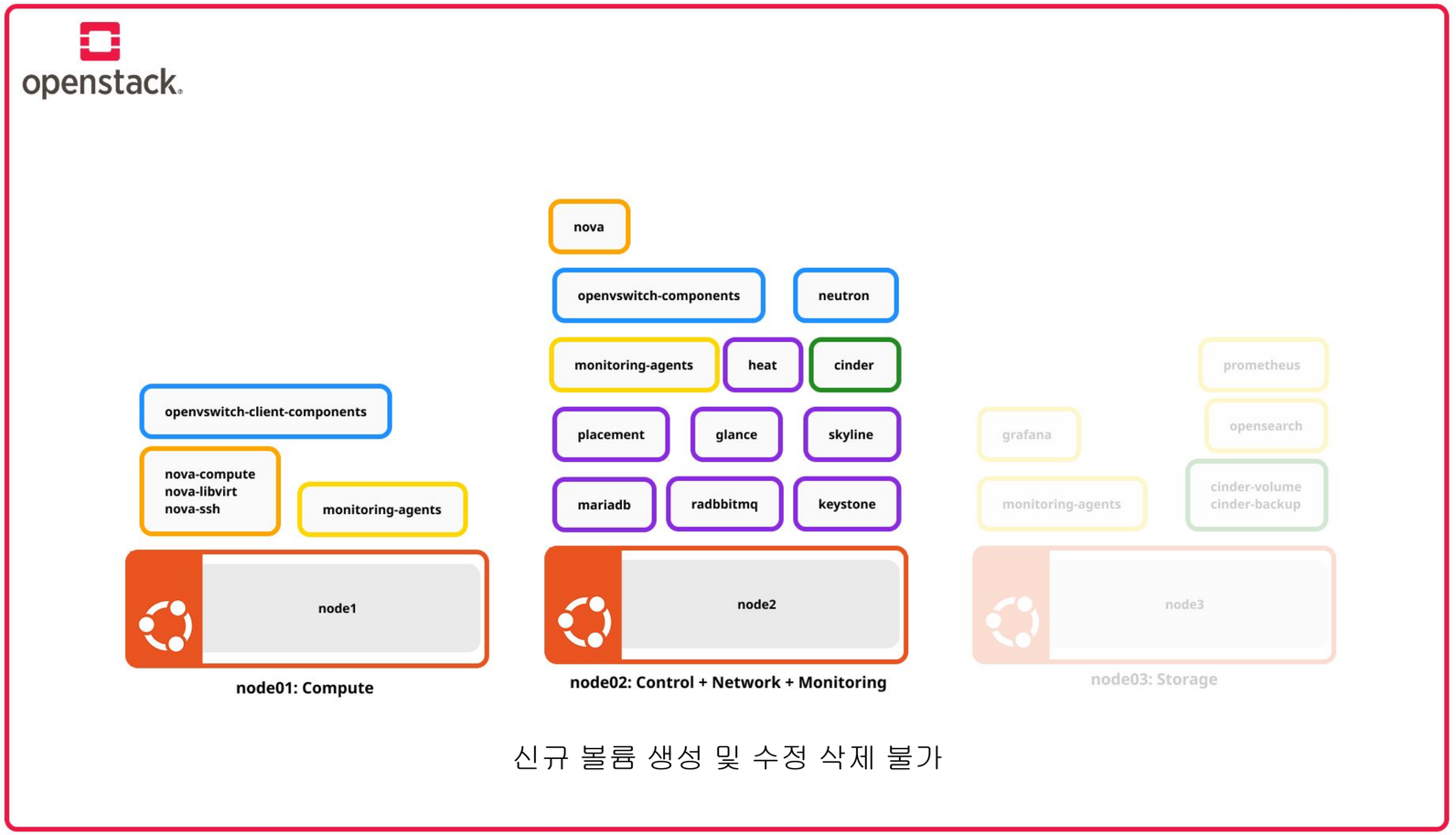

openstack.
openvswitch-client-components
nova-compute
nova-libvirt
nova-ssh
monitoring-agents
node1
node01: Compute
nova
openvswitch-components
neutron
monitoring-agents
heat
cinder
placement
glance
skyline
mariadb
radbbitmq
keystone
node2
node02: Control + Network + Monitoring
prometheus
opensearch
grafana
cinder-volume
cinder-backup
monitoring-agents
node3
node03: Storage


openstack.
openvswitch-client-components
nova-compute
nova-libvirt
nova-ssh
monitoring-agents
node1
node01: Compute
nova
openvswitch-components
neutron
monitoring-agents
heat
cinder
placement
glance
skyline
mariadb
radbbitmq
keystone
node2
node02: Control + Network + Monitoring
prometheus
opensearch
grafana
cinder-volume
cinder-backup
monitoring-agents
node3
node03: Storage

VM 관련 서비스 종료


openstack.
Active - Active 혹은 Active – Standby 구조 필요
compute - 2(Active or StandBy)
compute - 1 (Active)
node2
node3

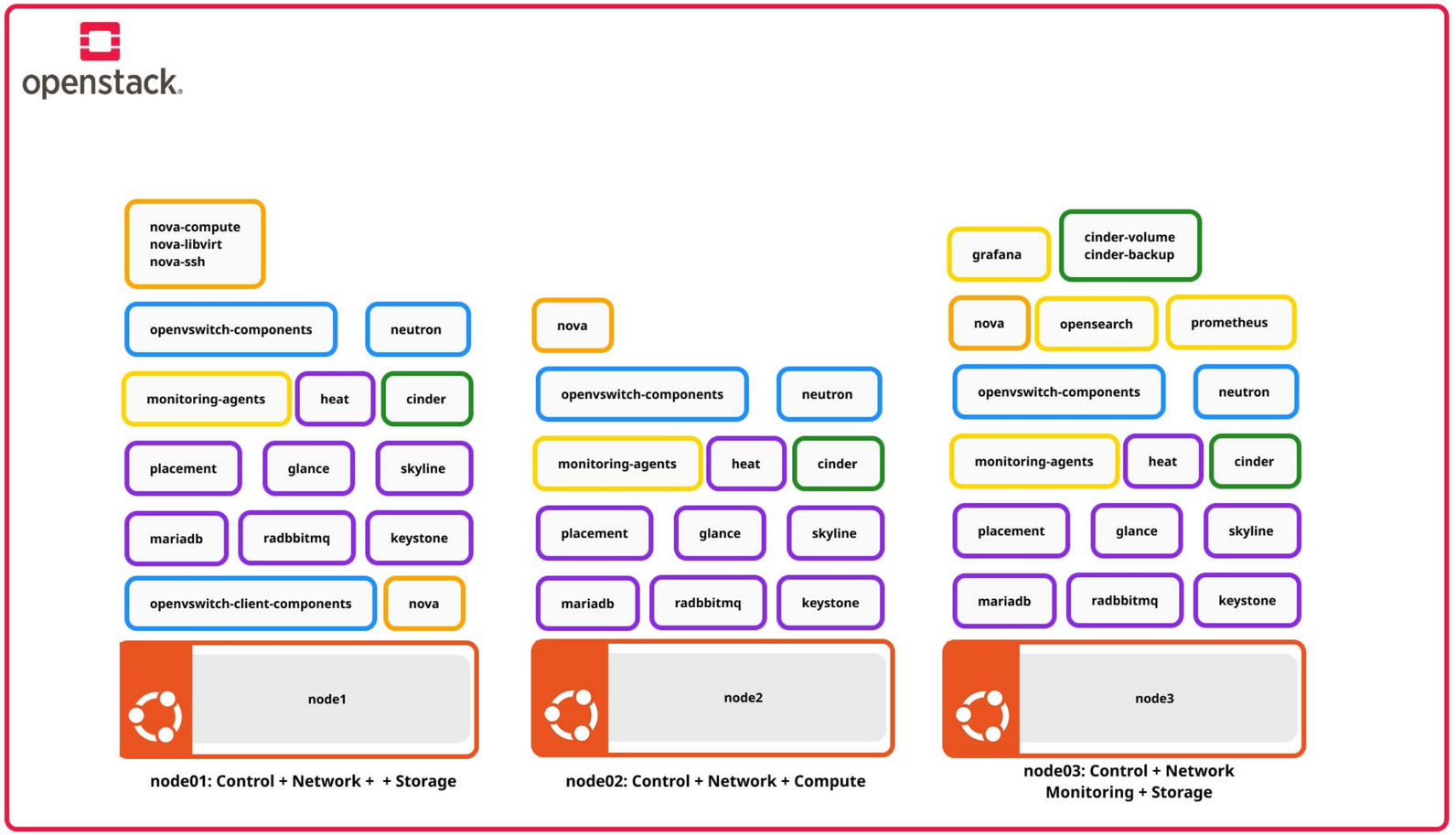

openstack.
openvswitch-client-components
nova-compute
nova-libvirt
nova-ssh
monitoring-agents
node1
node01: Compute
nova
openvswitch-components
neutron
monitoring-agents
heat
cinder
placement
glance
skyline
mariadb
radbbitmq
keystone
node2
node02: Control + Network + Monitoring
prometheus
opensearch
grafana
cinder-volume
cinder-backup
monitoring-agents
node3
node03: Storage

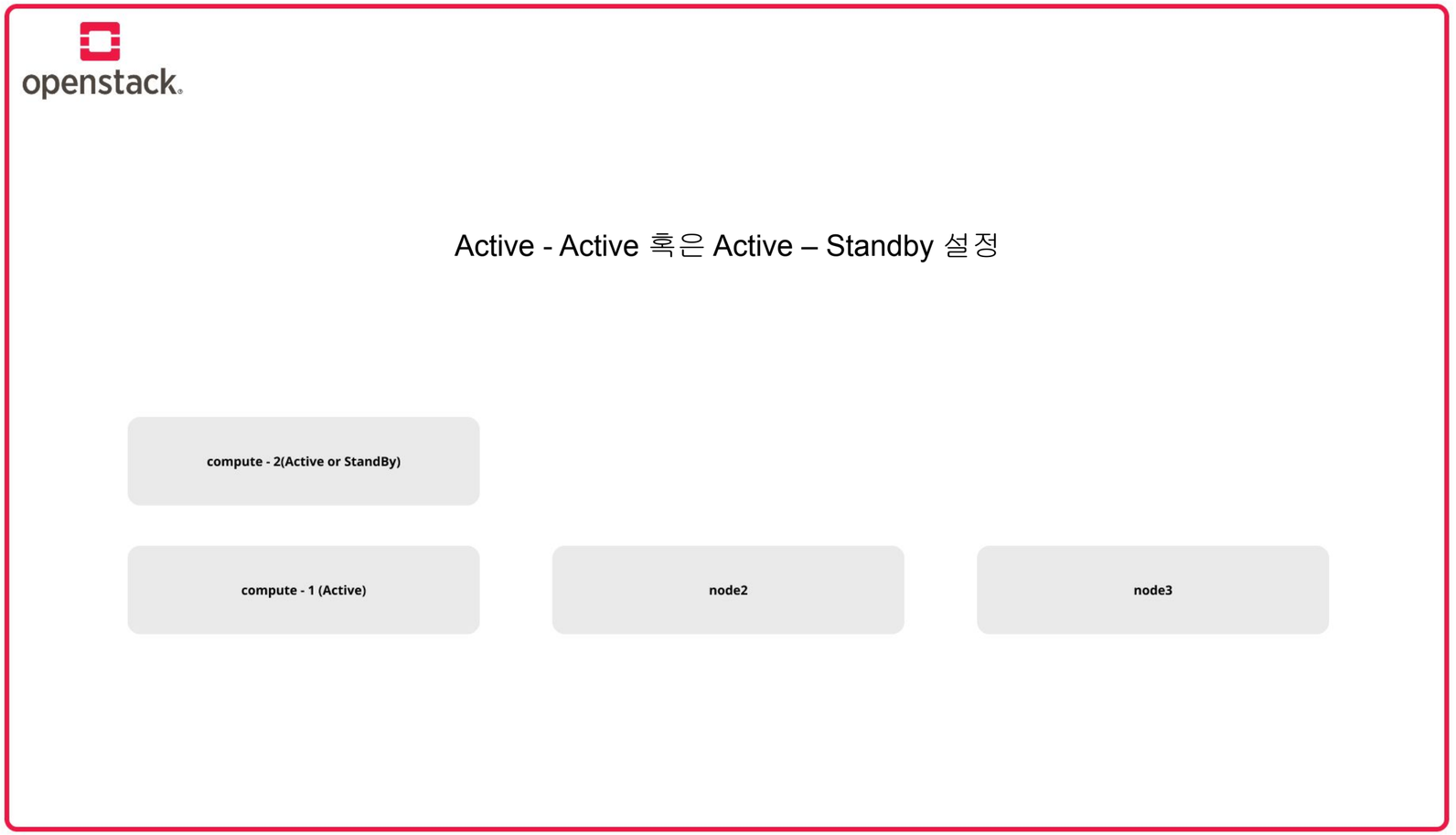

openstack.
openvswitch-client-components
nova-compute
nova-libvirt
nova-ssh
monitoring-agents
node1
node01: Compute
nova
openvswitch-components
neutron
monitoring-agents
heat
cinder
placement
glance
skyline
mariadb
radbbitmq
keystone
node2
node02: Control + Network + Monitoring
prometheus
grafana
opensearch
monitoring-agents
cinder-volume
cinder-backup
node3
node03: Storage
신규 볼륨 생성 및 수정 삭제 불가

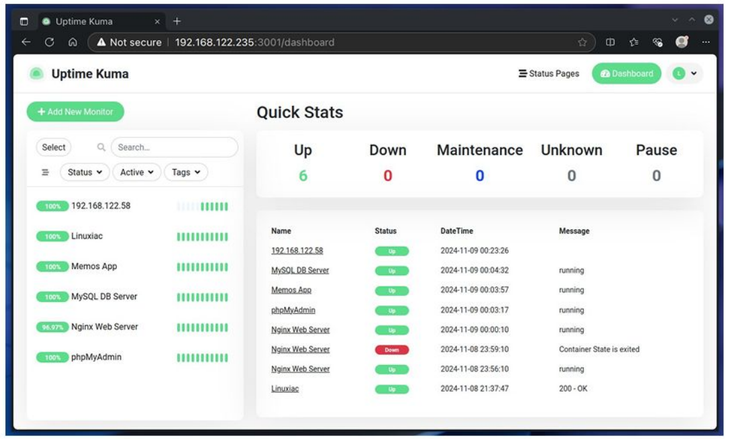
04

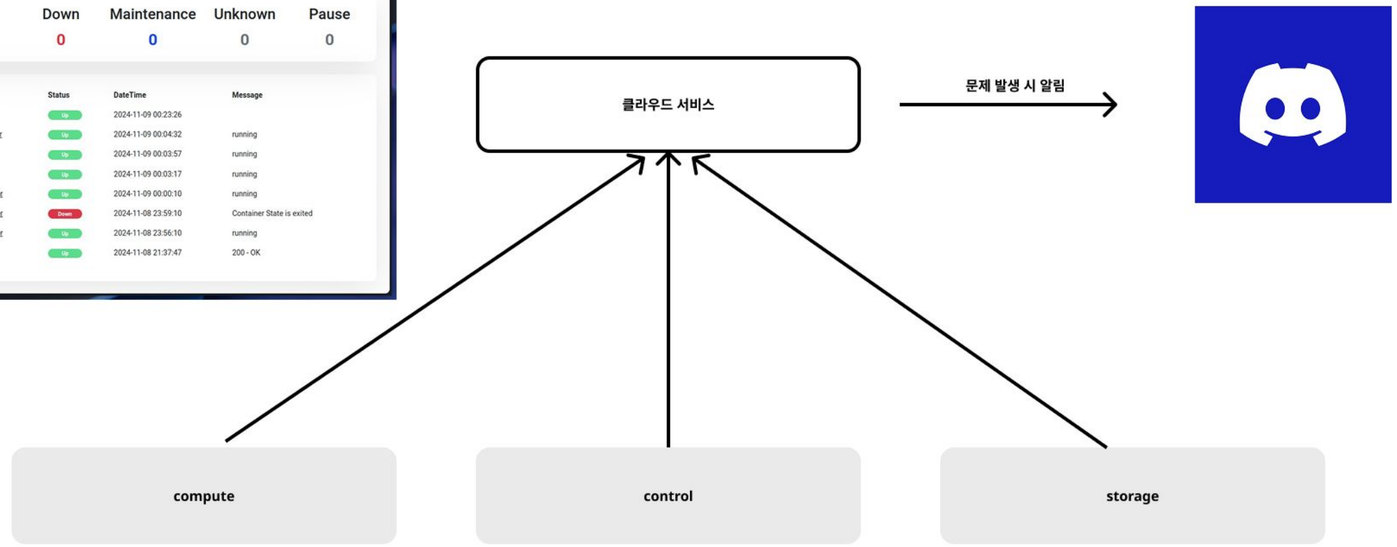
We will

앞으로의 아올다


openstack.
nova-compute
nova-libvirt
nova-ssh
openvswitch-components
neutron
monitoring-agents
heat
cinder
placement
glance
skyline
mariadb
radbbitmq
keystone
openvswitch-client-components
nova
node1
node01: Control + Network + + Storage
nova
openvswitch-components
neutron
monitoring-agents
heat
cinder
placement
glance
skyline
mariadb
radbbitmq
keystone
node2
node02: Control + Network + Compute
grafana
cinder-volume
cinder-backup
nova
opensearch
prometheus
openvswitch-components
neutron
monitoring-agents
heat
cinder
placement
glance
skyline
mariadb
radbbitmq
keystone
node3
node03: Control + Network
Monitoring + Storage


openstack.
Active - Active 혹은 Active – Standby 설정
compute - 2(Active or StandBy)
compute - 1 (Active)
node2
node3





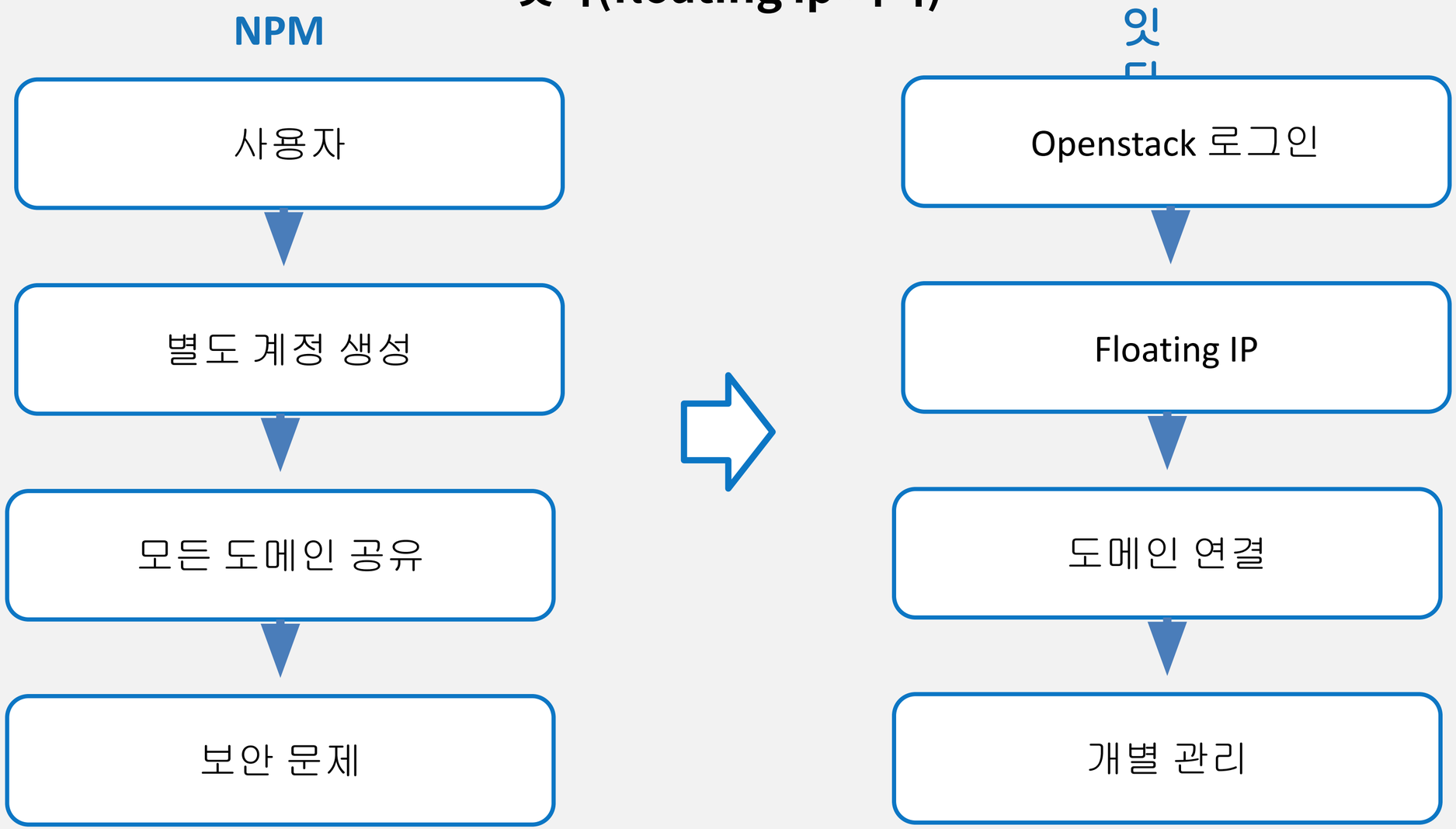
30초에 한번씩 하트비트

컨트롤 노드 3개로 SPOF 해소

컴퓨트 노드 이중화를 통한 VM 다운 타임 최소화

클라우드 서비스 기반 베어메탈 모니터링으로 노드 다운 시 알림

잇다(floating ip 처리)



운영과 사용성을 모두 고려한 클라우드의 확장

AMS
Aolda Monitoring System

- 문제가 생겼을 때 즉각 감지하고 대응할 수 있는 모니터링 시스템

ACC
Aolda Cloud Console

기존의 skyline 콘솔의 불편함

Aolda Cloud 기능 통합을 위한 자체 콘솔 제작

Aolda의 다음 단계

운영 <= 학습

- 서버 유지뿐 아니라,
  후속 인재들의 학습 기반이 중요
- 소학회가 이어지기 위해선
  운영 + 학습 환경이 병행되어야 함

학습용 테스트베드 구성

- VMWare 등
  하이퍼바이저를 설치해
  VM 기반 실습 환경 구성
- 운영용과 분리해 테스트 상황을 실습할 수 있는 시나리오 설계

모니터링 환경 구축

- 외부 클라우드 기반
  안정적 모니터링 환경 구축이 **목표**

우린 그냥 시작했을 뿐인데 

클라우드가
돌아가고 있습니다

랜딩페이지 QR 링크

감사합니다