

Ironic Baremetal Service	3
	4
	6
Inspection	가	6
Provision	7
Clean	8
Ironic(Stein)	(devstack)	8
Ironic	9
Provisioning	16
Cleaning	16
Deploy	User	17
Ironic Node	17
Ironic Node	19
Ironic Conductor	22
Ironic	24
Ironic Python Agent(IPA)	25
Ironic Console	26
Ironic	26
	27

Ironic Baremetal Service

Ironic 가

OpenStack Cloud

Compute(nova), Network(neutron), Image(glance),

OpenStack

, OpenStack Identity(keystone),

Object(swift)

PXE IPMI

, Compute

가

: <https://docs.openstack.org/ironic/latest/>

가

가

가

, Ironic

OpenStack

OpenStack

- Ironic Standalone(Bifrost): Ironic

- OpenStack (Nova): Ironic Nova

: Bifrost Standalone Ironic Ansible

Ironic OpenStack

가 가

가

Ironic

PXE IPMI

Ironic

Ironic

가

API,

Ironic

(: Kubernetes

)

Ironic

1.

2. **OpenStack**

Ironic OpenStack :

1. **Keystone**:

2. **Glance**:

3. **Nova**: Ironic

4. **Ironic**:

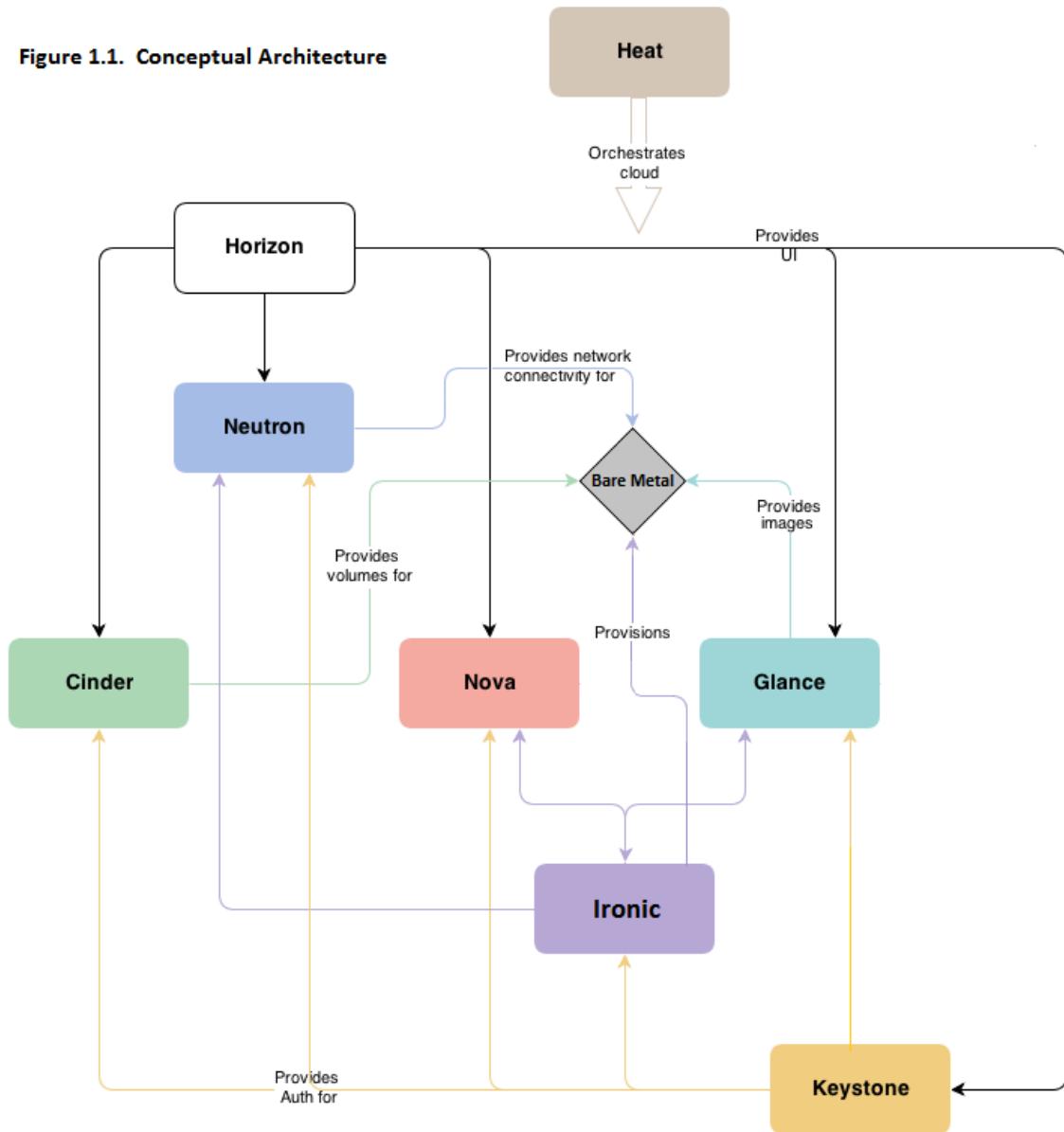
5. **Neutron**:

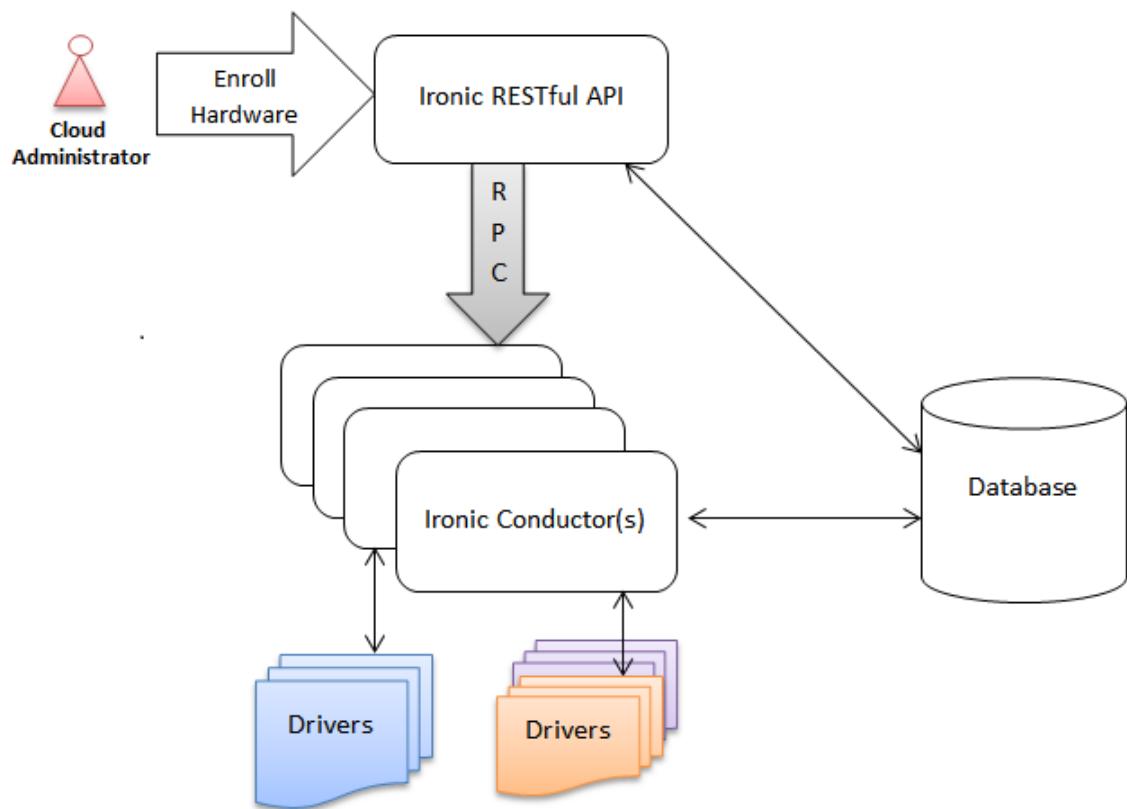
6. **Cinder**:

7. **Ceilometer**: IPMI

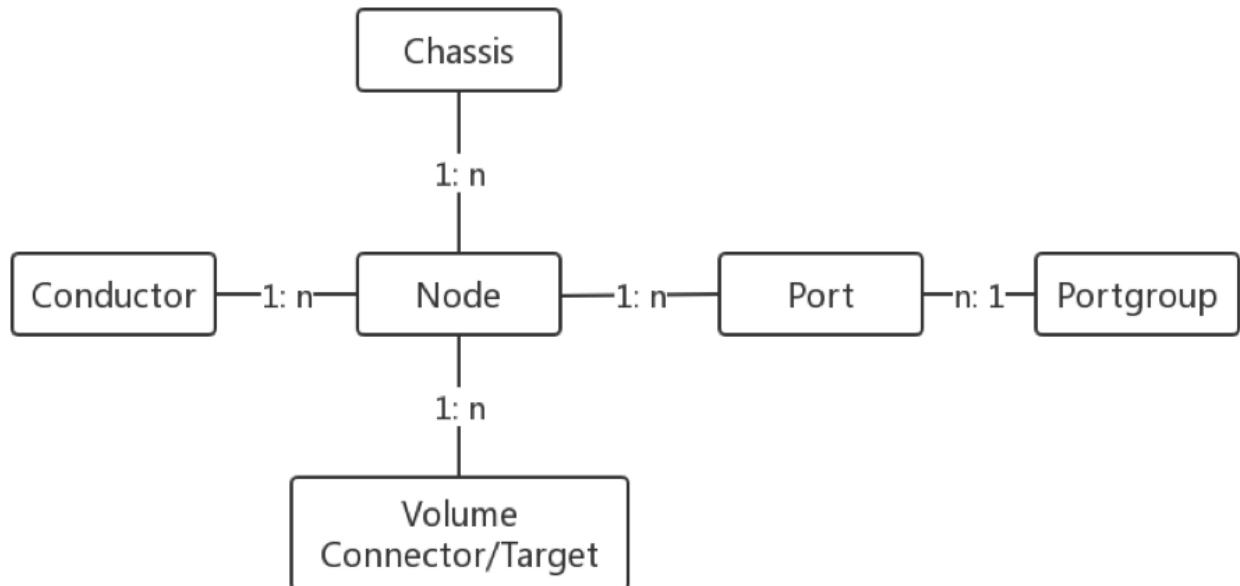
OpenStack

8. **Swift**:

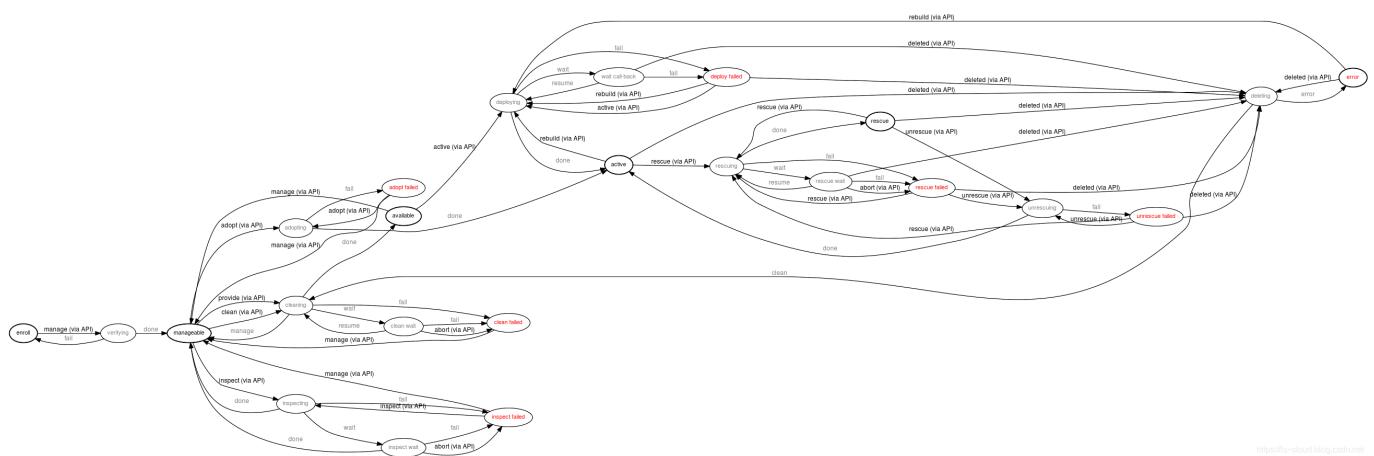
Figure 1.1. Conceptual Architecture<https://is-cloud.blog.csdn.net>



1. Ironic API: RESTful API
 2. Ironic Conductor:
 - 가, , , / , , , ,
 3. Drivers: , Ironic Conductor
 4. Ironic Inspector: “ ” , , , ,
 5. Ironic Python Agent(IPA): RAMDisk Python , RESTful API
Ironic Inspector Ironic Conductor , 가
 6. RAMDisk: “ ” , , 가
RAMDisk Ironic Python Agent , diskimage-builder
 7. diskimage-builder:
 8. python-ironicclient: Ironic CLI
 9. ironic-ui: Horizon
 10. DB:
 11. MQ:
 - : (out-of-band) (in-band) 가
 - **(out-of-band):** 가, , 가
 - **(in-band):** 가 (:) 가



1. **node**: , CPU, Ironic
 2. **chassis**: ,
 3. **port**: , MAC , LLDP
 4. **portgroup**:
 5. **conductor**: ironic-conductor
 6. **volume connector/target**:



Inspection

가

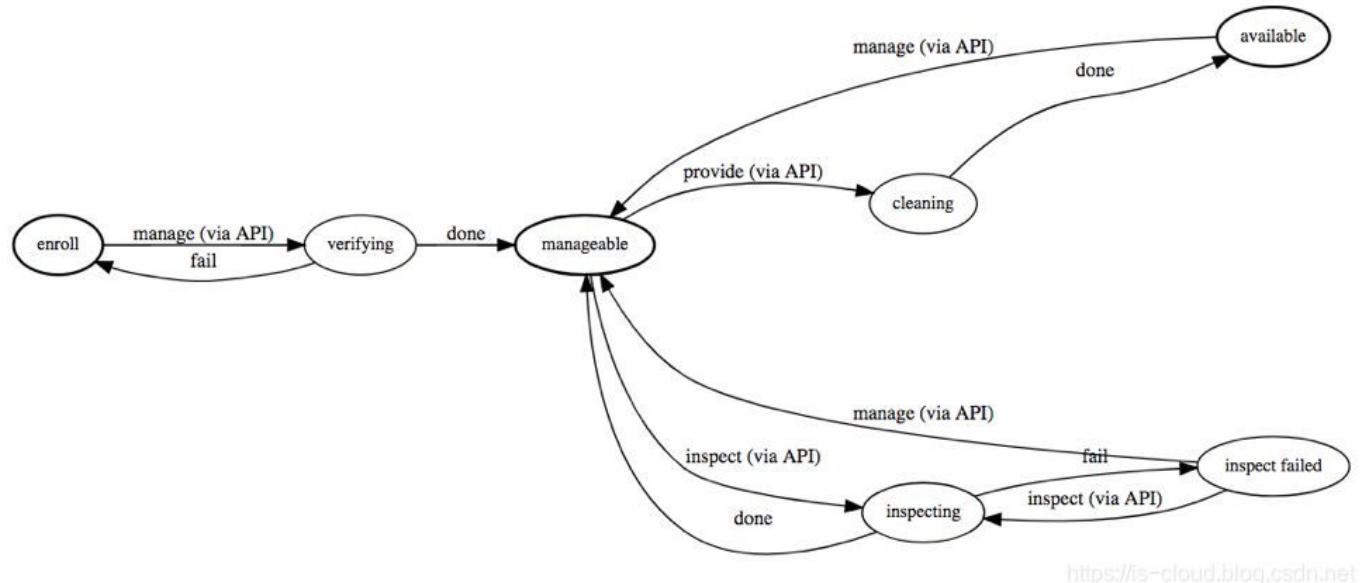
Ironic
Inspector

Ironic Inspector가

가 IPA(IPanic Python Agent) Ironic Inspector가 IPA가 . IPA
, Ironic Inspector . IPA가 . IPA
CPU Feature Flag, . IPA
Scheduler . IPA
LLDP(. IPA
, Chassis ID . IPA
Port ID . IPA

Inspection

: [] [] [] [] [] [] .



- enroll:

- verifying: 가 manage API , Ironic

- manageable: , 가 ,

- inspecting: Ironic , NIC , CPU, RAM,

- available: 가 manageable provide API

cleaning ,

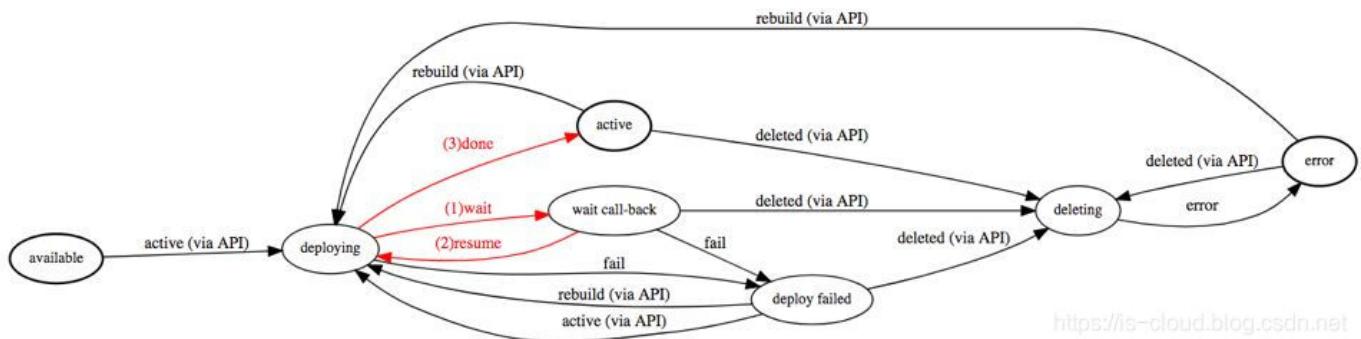
cleaning

(CPU, RAM, , NIC) 가

Provision

가
available , Provision
가
Provision : [] [] [] [] randisk
[].

Ironic-pyagent 가

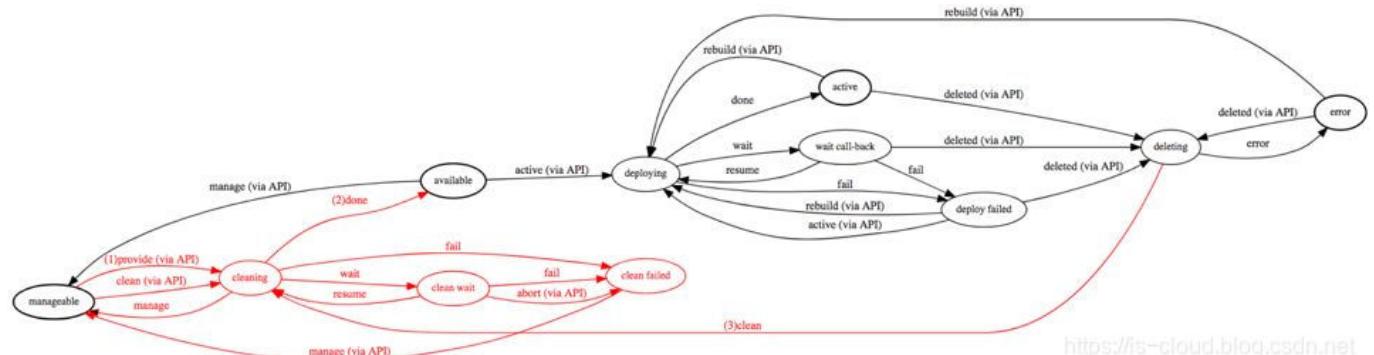


<https://is-cloud.blog.csdn.net>

- available: 가
active API (: ,
ironic-conductor
- deploying: 가
Ironic . PXE
ironic-pyagent
- wait callback: 가
Ironic
ironic-pyagent Ironic
Ironic . PXE
ironic-pyagent
- active: 가
Ironic . PXE
ironic-pyagent

Clean

Clean Ironic (. Clean
)
가
, RAID , BIOS



- Inspection
- Provision
- Provide
- Delete
- Cleaning
- Cleaning
- 가
- 가

Ironic(Stein) (devstack)

<https://docs.openstack.org/ironic/stein/contributor/dev-quickstart.html#deploying-ironic-with-devstack>

devstack

```
git clone https://git.openstack.org/openstack-dev/devstack.git -b
stable/stein
sudo ./devstack/tools/create-stack-user.sh
sudo su - stack
```

local.conf

```
[[local|localrc]]
HOST_IP=192.168.1.100

# Use TryStack(99cloud) git mirror
GIT_BASE=http://git.trystack.cn
#GIT_BASE=https://git.openstack.org

# Reclone each time
RECLONE=no

# Enable Logging
DEST=/opt/stack
LOGFILE=$DEST/logs/stack.sh.log
VERBOSE=True
LOG_COLOR=True
SCREEN_LOGDIR=$DEST/logs
LOGDAYS=1
```

```
# Define images to be automatically downloaded during the DevStack built
process.
DOWNLOAD_DEFAULT_IMAGES=False
IMAGE_URLS="http://download.cirros-cloud.net/0.3.4/cirros-0.3.4-x86_64-disk.
img"

# use TryStack git mirror
GIT_BASE=http://git.trystack.cn
NOVNC_REPO=http://git.trystack.cn/kanaka/noVNC.git
SPICE_REPO=http://git.trystack.cn/git/spice/sice-html5.git

# Apache Frontend
ENABLE_HTTPD_MOD_WSGI_SERVICES=False

# IP Version
IP_VERSION=4

# Credentials
ADMIN_PASSWORD=password
DATABASE_PASSWORD=password
RABBIT_PASSWORD=password
SERVICE_PASSWORD=password
SERVICE_TOKEN=password
SWIFT_HASH=password
SWIFT_TEMPURL_KEY=password

# Enable Ironic plugin
enable_plugin ironic https://git.openstack.org/openstack/ironic stable/stein

# Disable nova novnc service, ironic does not support it anyway.
disable_service n-novnc

# Enable Swift for the direct deploy interface.
enable_service s-proxy
enable_service s-object
enable_service s-container
enable_service s-account

# Cinder
VOLUME_GROUP_NAME="stack-volumes"
VOLUME_NAME_PREFIX="volume-"
VOLUME_BACKING_FILE_SIZE=100G

# Neutron
ENABLED_SERVICES+=,q-svc,q-agt,q-dhcp,q-l3,q-meta
# By default, DevStack creates a 10.0.0.0/24 network for instances.
# If this overlaps with the hosts network, you may adjust with the
# following.
NETWORK_GATEWAY=10.1.0.1
FIXED_RANGE=10.1.0.0/24
FIXED_NETWORK_SIZE=256
```

```

# Swift temp URL's are required for the direct deploy interface
SWIFT_ENABLE_TEMPURLS=True

# Create 3 virtual machines to pose as Ironic's baremetal nodes.
IRONIC_VM_COUNT=3
IRONIC_BAREMETAL_BASIC_OPS=True
DEFAULT_INSTANCE_TYPE=baremetal

# Enable additional hardware types, if needed.
#IRONIC_ENABLED_HARDWARE_TYPES=ipmi,fake-hardware
# Don't forget that many hardware types require enabling of additional
# interfaces, most often power and management:
#IRONIC_ENABLED_MANAGEMENT_INTERFACES=ipmitool,fake
#IRONIC_ENABLED_POWER_INTERFACES=ipmitool,fake
# The 'ipmi' hardware type's default deploy interface is 'iscsi'.
# This would change the default to 'direct':
#IRONIC_DEFAULT_DEPLOY_INTERFACE=direct

# Change this to alter the default driver for nodes created by devstack.
# This driver should be in the enabled list above.
IRONIC_DEPLOY_DRIVER=ipmi

# The parameters below represent the minimum possible values to create
# functional nodes.
IRONIC_VM_SPECS_RAM=1280
IRONIC_VM_SPECS_DISK=10

# Size of the ephemeral partition in GB. Use 0 for no ephemeral partition.
IRONIC_VM_EPHEMERAL_DISK=0

# To build your own IPA ramdisk from source, set this to True
IRONIC_BUILD_DEPLOY_RAMDISK=False

VIRT_DRIVER=ironic

# Log all output to files
LOGFILE=/opt/stack/devstack.log
LOGDIR=/opt/stack/logs
IRONIC_VM_LOG_DIR=/opt/stack/ironic-bm-logs

```

```

[root@localhost ~]# openstack compute service list
+-----+-----+-----+-----+-----+
| ID  | Binary          | Host           | Zone  | Status | State
| Updated At          |               |           |
+-----+-----+-----+-----+-----+
| 3  | nova-scheduler | localhost.localdomain | internal | enabled | up

```

2019-05-03T18:56:18.000000	6 nova-consoleauth localhost.localdomain internal enabled up
2019-05-03T18:56:22.000000	7 nova-conductor localhost.localdomain internal enabled up
2019-05-03T18:56:14.000000	1 nova-conductor localhost.localdomain internal enabled up
2019-05-03T18:56:15.000000	3 nova-compute localhost.localdomain nova enabled up
2019-05-03T18:56:18.000000	
-----+-----+-----+-----+-----+-----+	
-----+-----+	
[root@localhost ~]# openstack network agent list	
-----+-----+-----+-----+-----+-----+	
-----+-----+-----+-----+-----+-----+	
ID Agent Type Host	
Availability Zone Alive State Binary	
-----+-----+-----+-----+-----+-----+	
52f23bda-a645-4459-bcac-686d98d23345 Open vSwitch agent	
localhost.localdomain None :-) UP neutron-	
openvswitch-agent	
7113312f-b0b7-4ce8-ab15-428768b30855 L3 agent	
localhost.localdomain nova :-) UP neutron-l3-agent	
a45fb074-3b24-4b9e-8c8a-43117f6195f2 Metadata agent	
localhost.localdomain None :-) UP neutron-	
metadata-agent	
f207648b-03f3-4161-872e-5210f29099c6 DHCP agent	
localhost.localdomain nova :-) UP neutron-dhcp-	
agent	
-----+-----+-----+-----+-----+-----+	
-----+-----+-----+-----+-----+	
-----+-----+-----+-----+-----+	
[root@localhost ~]# openstack volume service list	
-----+-----+-----+-----+-----+-----+	
-----+-----+-----+-----+-----+	
Binary Host Zone Status	
State Updated At	
-----+-----+-----+-----+-----+-----+	
-----+-----+-----+-----+-----+	
cinder-scheduler localhost.localdomain nova enabled up	
2019-05-03T18:56:54.000000	
cinder-volume localhost.localdomain@lvmdriver-1 nova enabled up	
2019-05-03T18:56:53.000000	
-----+-----+-----+-----+-----+-----+	
-----+-----+-----+-----+-----+	
-----+-----+-----+-----+-----+	
[root@localhost ~]# openstack baremetal driver list	
-----+-----+-----+-----+-----+	
Supported driver(s) Active host(s)	

```
+-----+-----+
| fake-hardware | localhost      |
| ipmi          | localhost      |
+-----+-----+-----+
[root@localhost ~]# openstack baremetal node list
+-----+-----+-----+
|-----+-----+-----+
| UUID                         | Name   | Instance UUID | Power
| State | Provisioning State | Maintenance | 
+-----+-----+-----+
| adda54fb-1038-4634-8d82-53922e875a1f | node-0 | None      | power off
| available          | False   | 
| 6952e923-11ae-4506-b010-fd7a3c4278f5 | node-1 | None      | power off
| available          | False   | 
| f3b8fe69-a840-42dd-9cbf-217be8a95431 | node-2 | None      | power off
| available          | False   | 
+-----+-----+-----+
|-----+-----+-----+
```

```
[root@localhost ~]# openstack server create --flavor baremetal --image
cirros-0.4.0-x86_64-disk --key-name default --nic net-id=5c86f931-64da-4c69-
a0f1-e2da6d9dd082 VM1
+-----+
|-----+-----+
| Field          | Value
|-----+-----+
| OS-DCF:diskConfig | MANUAL
|-----+-----+
| OS-EXT-AZ:availability_zone | 
|-----+-----+
| OS-EXT-SRV-ATTR:host | None
|-----+-----+
| OS-EXT-SRV-ATTR:hypervisor_hostname | None
|-----+-----+
| OS-EXT-SRV-ATTR:instance_name | 
|-----+-----+
| OS-EXT-STS:power_state | NOSTATE
|-----+-----+
| OS-EXT-STS:task_state | scheduling
|-----+-----+
| OS-EXT-STS:vm_state | building
|-----+-----+
| OS-SRV-USG:launched_at | None
|-----+-----+
```

OS-SRV-USG:terminated_at	None
accessIPv4	
accessIPv6	
addresses	
adminPass	k3TgBf5Xjsqv
config_drive	
created	2019-05-03T20:26:28Z
flavor	baremetal (8f6fd22b-9bec-4b4d-
b427-7c333e47d2c2)	
hostId	
id	70e9f2b1-a292-4e95-90d4-55864bb0a71d
image	cirros-0.4.0-x86_64-disk (4ff12aca-
b762-436c-b98c-579ad2a21649)	
key_name	default
name	VM1
progress	0
project_id	cbf936fc5e9d4cfcaa1dbc06cd9d2e3e
properties	
security_groups	name='default'
status	BUILD
updated	2019-05-03T20:26:28Z
user_id	405fad83a4b3470faf7d6c616fe9f7f4
volumes_attached	

[root@localhost ~]# openstack baremetal node list

UUID	Power State	Provisioning State	Maintenance	Name	Instance UUID

```
-----+-----+-----+-----+
| adda54fb-1038-4634-8d82-53922e875a1f | node-0 | None
| power off    | available           | False    |
| 6952e923-11ae-4506-b010-fd7a3c4278f5 | node-1 | None
| power off    | available           | False    |
| f3b8fe69-a840-42dd-9cbf-217be8a95431 | node-2 | 70e9f2b1-
a292-4e95-90d4-55864bb0a71d | power off   | deploying | False
|
-----+-----+-----+-----+
```

```
[root@localhost ~]# openstack server list --long
```

```
+-----+-----+-----+-----+
-----+-----+-----+-----+
-----+-----+-----+-----+
-----+-----+-----+-----+
-----+-----+-----+-----+
| ID                           | Name | Status | Task State | Power
State | Networks                 | Image Name           | Image ID
| Flavor Name | Flavor ID          |                     | Availability Zone |
Host          | Properties |           |
+-----+-----+-----+-----+
-----+-----+-----+-----+
-----+-----+-----+-----+
-----+-----+-----+-----+
| 70e9f2b1-a292-4e95-90d4-55864bb0a71d | VM1   | ACTIVE | None      |
Running      | private=10.0.0.40 | cirros-0.4.0-x86_64-disk | 4ff12aca-
b762-436c-b98c-579ad2a21649 | baremetal | 8f6fd22b-9bec-4b4d-
b427-7c333e47d2c2 | nova      | localhost.localdomain | |
+-----+-----+-----+-----+
-----+-----+-----+-----+
-----+-----+-----+-----+
-----+-----+-----+-----+
```

```
[root@localhost ~]# openstack baremetal node list
```

```
+-----+-----+-----+
-----+-----+-----+-----+
| UUID                         | Name | Instance UUID
| Power State | Provisioning State | Maintenance |
+-----+-----+-----+
-----+-----+-----+-----+
| adda54fb-1038-4634-8d82-53922e875a1f | node-0 | None
| power off    | available           | False    |
| 6952e923-11ae-4506-b010-fd7a3c4278f5 | node-1 | None
| power off    | available           | False    |
| f3b8fe69-a840-42dd-9cbf-217be8a95431 | node-2 | 70e9f2b1-
a292-4e95-90d4-55864bb0a71d | power on   | deploying | False
|
-----+-----+-----+-----+
-----+-----+-----+-----+
```

```
[root@localhost ~]# ssh cirros@10.0.0.40
```

\$

Ironic OpenStack Nova

```
# nova.conf

[DEFAULT]
...
ompute_driver = ironic.IronicDriver
```

Ironic

Provisioning

Provisioning

DHCP PXE

```
# /etc/neutron/plugins/ml2/ml2_conf.ini

[ml2_type_flat]
flat_networks = public, physnet1

[ovs]
datapath_type = system
bridge_mappings = public:br-ex, physnet1:br-eth2
tunnel_bridge = br-tun
local_ip = 172.22.132.93
```

OvS Provider

```
$ sudo ovs-vsctl add-br br-eth2  
$ sudo ovs-vsctl add-port br-eth2 eth2
```

0vS

```
$ sudo systemctl restart devstack@q-svc.service  
$ sudo systemctl restart devstack@q-agt.service
```

```
$ neutron net-create sharednet1 \
    --shared \
    --provider:network_type flat \
    --provider:physical_network physnet1

$ neutron subnet-create sharednet1 172.22.132.0/24 \
    --name sharedsubnet1 \
    --ip-version=4 --gateway=172.22.132.254 \
    --allocation-pool
start=172.22.132.180,end=172.22.132.200 \
    --enable-dhcp
```

: DHCP

PXE

IP

Cleaning

ironic node clean	Cleaning	가	Cleaning
Provisioning	.	.	.

```
# /etc/ironic/ironic.conf

[neutron]
cleaning_network = sharednet1
```

Ironic

```
$ sudo systemctl restart devstack@ir-api.service
$ sudo systemctl restart devstack@ir-cond.service
```

Deploy User

Deploy	User	Disk Image Builder	.	DevStack
Deploy

disk-image-builder

```
$ virtualenv dib
$ source dib/bin/activate
(dib) $ pip install diskimage-builder
```

User Image

```

$ cat <<EOF > k8s.repo
[kubernetes]
name=Kubernetes
baseurl=http://yum.kubernetes.io/repos/kubernetes-el7-x86_64
enabled=1
gpgcheck=0
repo_gpgcheck=0
gpgkey=https://packages.cloud.google.com/yum/doc/yum-key.gpg
https://packages.cloud.google.com/yum/doc/rpm-package-key.gpg
EOF

$ DIB_YUM_REPO_CONF=k8s.repo \
  DIB_DEV_USER_USERNAME=kyle \
  DIB_DEV_USER_PWDLESS_SUDO=yes \
  DIB_DEV_USER_PASSWORD=r00tme \
  disk-image-create \
    centos7 \
    dhcp-all-interfaces \
    devuser \
    yum \
    epel \
    baremetal \
    -o k8s.qcow2 \
    -p vim,docker,kubelet,kubeadm,kubectl,kubernetes-cni

...
Converting image using qemu-img convert
Image file k8s.qcow2 created...

$ ls
dib  k8s.d  k8s.initrd  k8s.qcow2  k8s.repo  k8s.vmlinuz

```

Glance

```

# Kernel
$ openstack image create k8s.kernel \
    --public \
    --disk-format aki \
    --container-format aki < k8s.vmlinuz

# Initrd
$ openstack image create k8s.initrd \
    --public \
    --disk-format ari \
    --container-format ari < k8s.initrd

# Qcow2
$ export MY_VMLINUZ_UUID=$(openstack image list | awk '/k8s.kernel/ { print $2 }')
$ export MY_INITRD_UUID=$(openstack image list | awk '/k8s.initrd/ { print

```

```
$2 }')
$ openstack image create k8s \
    --public \
    --disk-format qcow2 \
    --container-format bare \
    --property kernel_id=$MY_VMLINUZ_UUID \
    --property ramdisk_id=$MY_INITRD_UUID < k8s.qcow2
```

Ironic Node

Ironic Ironic

(**Ironic**)가

```
$ ironic driver-list
+-----+-----+
| Supported driver(s) | Active host(s) |
+-----+-----+
| agent_ipmitool      | ironic-dev      |
| fake                | ironic-dev      |
| ipmi                | ironic-dev      |
| pxe_ipmitool        | ironic-dev      |
+-----+-----+
```

: , Set up the drivers for the Bare Metal service 가

Ironic

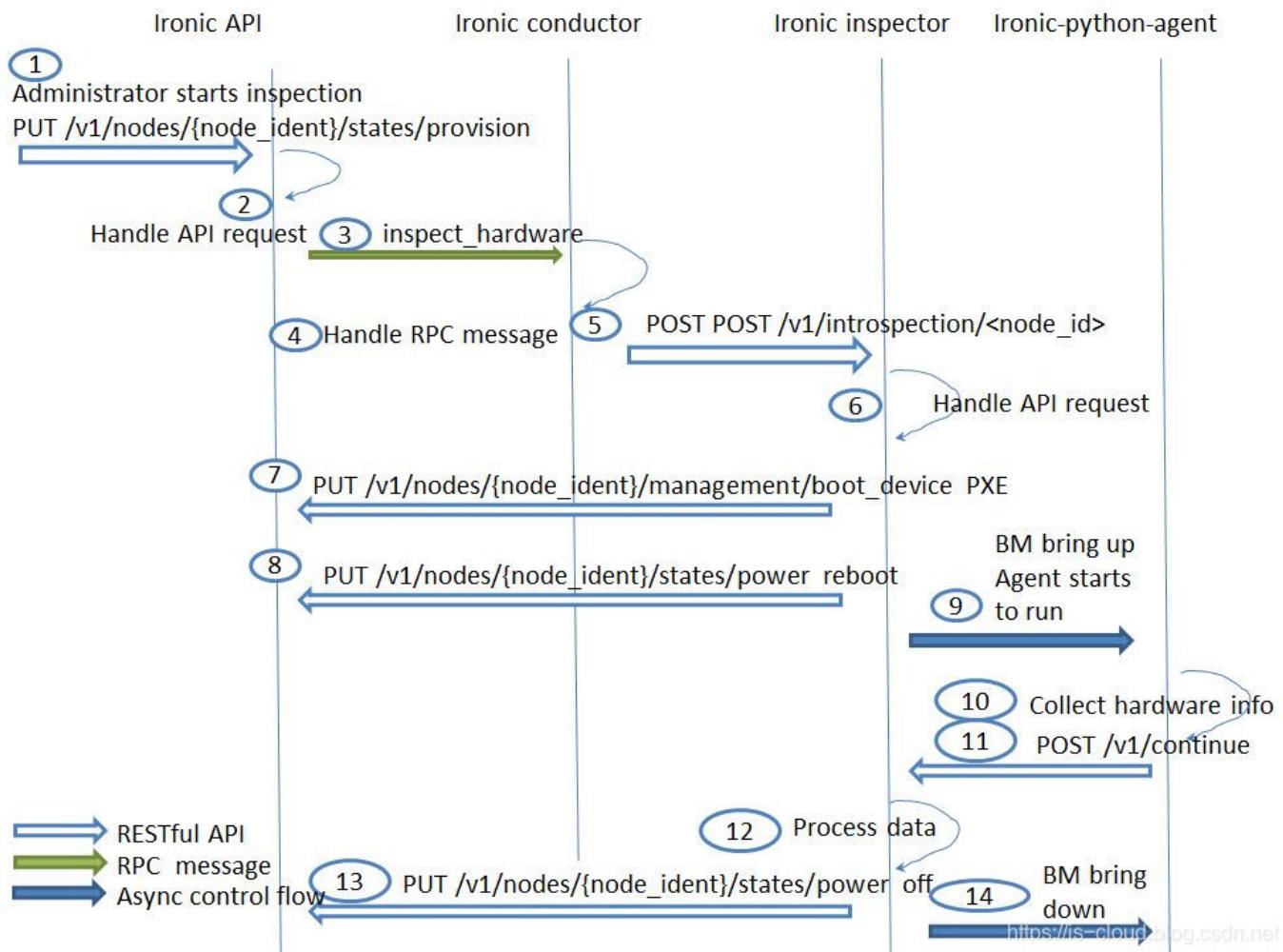
```
$ export DEPLOY_VMLINUZ_UUID=$(openstack image list | awk '/ipmitool.kernel/ \
{ print $2 }')
$ export DEPLOY_INITRD_UUID=$(openstack image list | awk \
'/ipmitool.initramfs/ { print $2 }')
$ ironic node-create -d agent_ipmitool \
    -n bare-node-1 \
    -i ipmi_address=172.20.3.194 \
    -i ipmi_username=maas \
    -i ipmi_password=passwd \
    -i ipmi_port=623 \
    -i ipmi_terminal_port=9000 \
    -i deploy_kernel=$DEPLOY_VMLINUZ_UUID \
    -i deploy_ramdisk=$DEPLOY_INITRD_UUID
```

```
$ export NODE_UUID=$(ironic node-list | awk '/bare-node-1/ { print $2 }')$ ironic node-update $NODE_UUID add \  
    properties/cpus=4 \  
    properties/memory_mb=8192 \  
    properties/local_gb=100 \  
    properties/root_gb=100 \  
    properties/cpu_arch=x86_64
```

: 가 Ironic Inspector
:

```
# /etc/ironic-inspector/dnsmasq.conf  
  
no-daemon  
port=0  
interface=eth1  
bind-interfaces  
dhcp-range=172.22.132.200,172.22.132.210  
dhcp-match=ipxe,175  
dhcp-boot=tag:!ipxe,undionly.kpxe  
dhcp-boot=tag:ipxe,http://172.22.132.93:3928/ironic-inspector.ipxe  
dhcp-sequential-ip  
  
$ devstack@ironic-inspector-dhcp.service  
$ devstack@ironic-inspector.service
```

Inspection



Ironic

```
$ ironic port-create -n $NODE UUID -a NODE PXE NIC MAC ADDRESS
```

Ironics

deploy	False	Cannot validate image information for node 8e6fd86a-8eed-4e24-a510-3f5ebb0a336a because one or more parameters are missing from its instance_info. Missing are: ['ramdisk', 'kernel', 'image_source']
inspect	True	
management	True	
network	True	
power	True	
raid	True	
storage	True	
<hr/>		

: Ironic boot deploy가 False

Irony 가

```
$ ironic --ironic-api-version 1.34 node-set-provision-state $NODE_UUID
manage
$ ironic --ironic-api-version 1.34 node-set-provision-state $NODE_UUID
provide
$ ironic node-list
+-----+-----+-----+-----+
|-----+-----+
| UUID | Provisioning State | Maintenance | Name | Instance UUID | Power
State | | |
+-----+-----+-----+-----+
|-----+-----+
| 0c20cf7d-0a36-46f4-ac38-721ff8bfb646 | bare-0 | None | bare-0 | None | power off
| cleaning | False | | |
+-----+-----+-----+-----+
|-----+-----+

```

: Ironic 가 cleaning available

Ironic Conductor

Conductor

. Conductor

가
Ironic

가) Conductor
Conductor

Conductor

Conductor
가 Conductor

Conductor

가 가
가 (: take-over

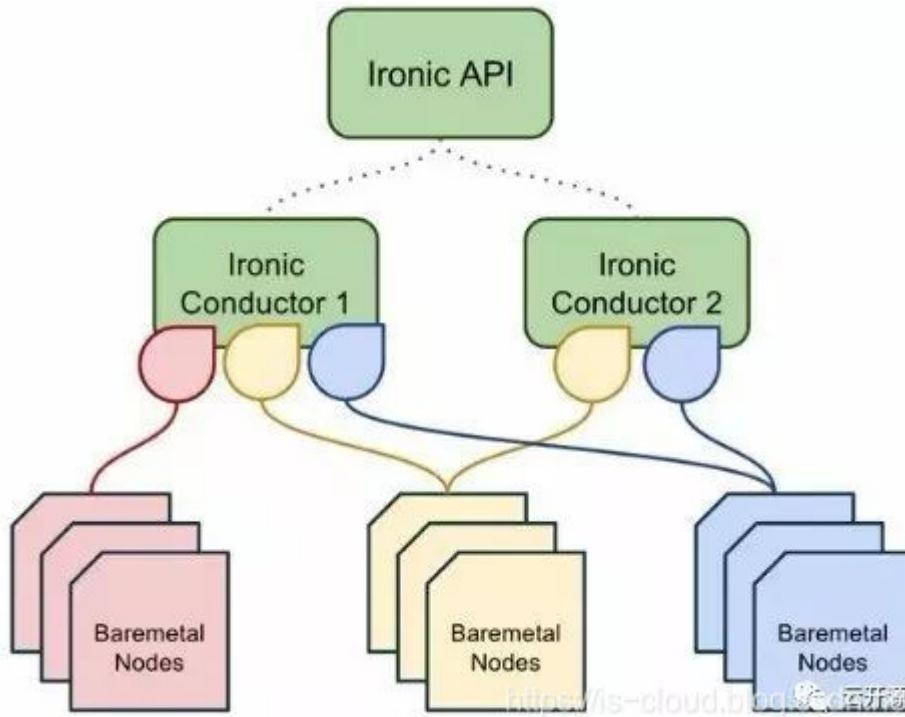
clean-up),

Ironic

가 (: take-over) 가

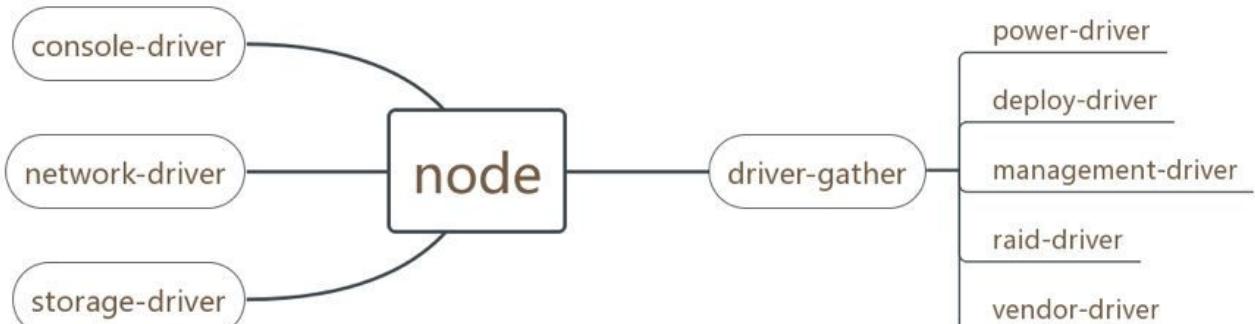
The diagram illustrates a distributed system structure. It features a central conductor node (shaded gray) connected to four other nodes (white circles) via curved arrows labeled 'conductor key1' through 'conductor key4'. Each of these four nodes is also connected to a 'node key' node (white circle) via straight arrows. Each 'node key' node is associated with a 'hash' value (represented by a cloud-like shape). The nodes are labeled as follows:

- Top node: 'node2' with 'hash' value.
- Top-right node: 'conductor1' with 'node key2' and 'hash' value.
- Middle-left node: 'node key1' with 'node1' and 'hash' value.
- Middle-right node: 'node key3' with 'node3' and 'hash' value.
- Bottom-left node: 'node key4' with 'node4' and 'hash' value.
- Bottom-right node: 'conductor2' with 'node key2' and 'hash' value.
- Left node: 'conductor3' with 'node key1' and 'hash' value.
- Right node: 'conductor4' with 'node key3' and 'hash' value.



Ironic

Ironic은 네트워크를 통해baremetal 노드를 관리하는OpenStack 서비스입니다. 네트워크 드라이버, 스토리지 드라이버, 콘솔 드라이버, 파워 드라이버, 배포 드라이버, 관리 드라이버, RAID 드라이버, 벤더 드라이버 등 다양한 드라이버를 지원합니다.



<https://is-cloud.blog.csdn.net>

1. **power:**
2. **boot:** ()
3. **deploy:**
4. **management:**
5. **raid:** RAID
6. **inspect:** 가
7. **console:**
8. **network:**
9. **storage:**

10. vendor:

Ironic

RESTful API

driver集合名称	driver名称	driver类别
agent_ipmitool	ipmi	power
	pxe	boot
	agent	deploy

Ironic Python Agent(IPA)

Ironic Python Agent, IPA
ramdisk IPA

initramfs IPA Ironic Conductor

- **ironic-agent.kernel:**
- **ironic-agent.initramfs:** IPA Ironic Conductor

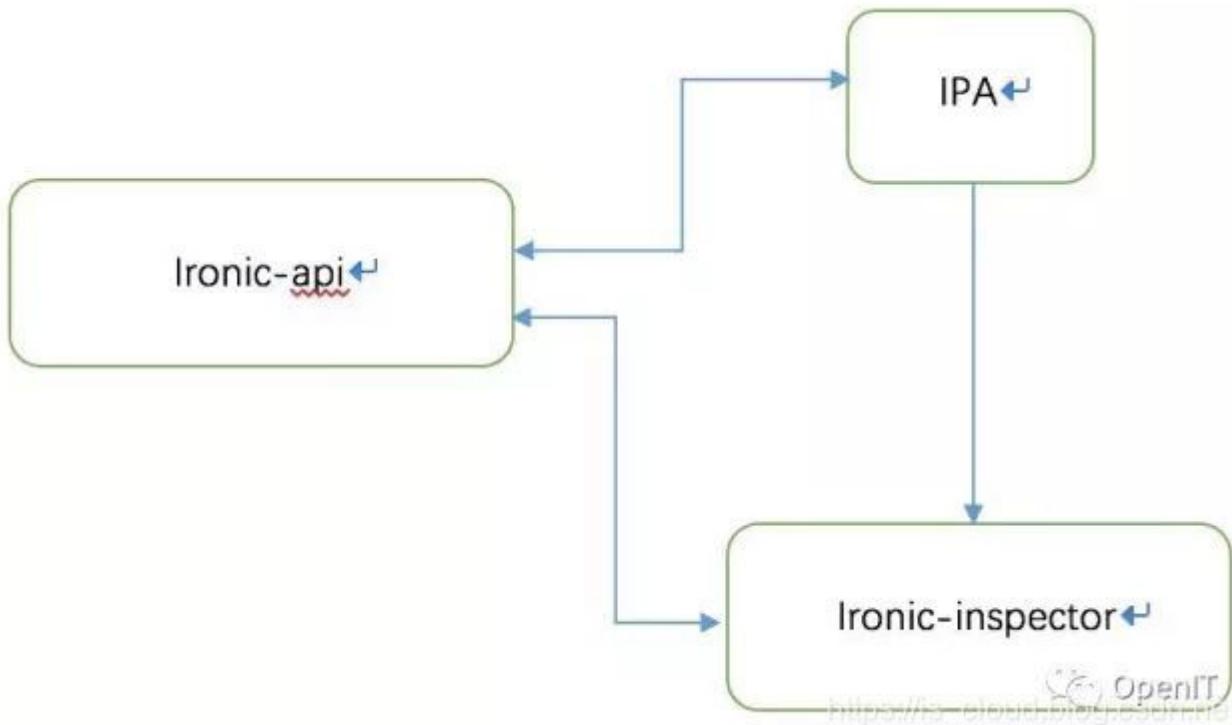
IPA :

-
-
-
-
- RAID

IPA :

```
disk-image-create -c ironic-agent ubuntu dynamic-login stable-interface-
names proliant-tools -o ironic-agent
```

: PXE	IPA	. IPA	ironic-inspector
, inspector	IPMI	IP/MAC	ironic-api
. ironic-api IPA	agent_*		Clean



Ironic Console

Ironic 가 :

1. **Shellinabox**
2. **Socat**

Shellinabox

Ajax HTTP

Socat Shellinabox

Shellinabox 가 , Ironic Conductor TCP

. Socat TCP Nova Serial Console Socat

Ironic Conductor . Socat `yum`
, Shellinabox EPEL

. Shellinabox RPM

Ironic Socat Socat (: `agent_ipmitool_socat`).
Shellinabox , 가

가

Ironic

Ironic

가

가

Disk

Image Builder :

- **Deploy Images**
- **User Images**

Deploy Images

tgt+iSCSI

IPA(Ironic Python Agent)

. IPA

deploy image

- ironic-deploy.vmlinuz (ironic-deploy.kernel)
- ironic-deploy.initramfs

```
disk-image-create ironic-agent centos7 -o ironic-deploy
```

User Images

user image

- my-image.qcow2
- my-image.vmlinuz
- my-image.initrd

```
image-create centos7 baremetal dhcp-all-interfaces grub2 -o my-image
```

- <https://www.cnblogs.com/jmilkfan-fanguiju/p/11825068.html>

From:

<https://atl.kr/dokuwiki/> - AllThatLinux!

Permanent link:

https://atl.kr/dokuwiki/doku.php/ironic_baremetal_service

Last update: **2024/10/11 06:11**

