elasticsearch
for developer

Life on Cluster

02_cluster.txt

Single node

NODE 1

master
data

Create index for single node

```
PUT /blog
{
  "settings": {
    "number_of_shards": 3,
    "number_of_replicas": 0
  }
}
```

Cluster health

GET _cluster/health

```json
{
    "cluster_name": "elasticsearch",
    "status": "green",
    "timed_out": false,
    "number_of_nodes": 1,
    "number_of_data_nodes": 1,
    "active_primary_shards": 3,
    "active_shards": 3,
    "relocating_shards": 0,
    "initializing_shards": 0,
    "unassigned_shards": 0,
    "number_of_pending_tasks": 0
}
```

n Nodes

NODE 1

NODE 2

master
data

master
data

Distributed data

Distributed data

blog
size: 7.10ki (7.10ki)
docs: 3 (3)

Info ▼   Actions ▼

★ **node01**
Info ▼   Actions ▼

1 **2**

● **node02**
Info ▼   Actions ▼

0

Cluster health

GET _cluster/health

```json
{
    "cluster_name": "kaidee",
    "status": "green",
    "timed_out": false,
    "number_of_nodes": 2,
    "number_of_data_nodes": 2,
    "active_primary_shards": 3,
    "active_shards": 3,
    "relocating_shards": 0,
    "initializing_shards": 0,
    "unassigned_shards": 0,
    "number_of_pending_tasks": 0
}
```

Replica = 1

NODE 1

NODE 2

master
data

master
data

Replicate data

NODE 1

S 1 S 2

S 3 S 4

NODE 2

S 1 S 2

S 3 S 4

Create index for replica

```
PUT /blog2
{
  "settings": {
    "number_of_shards": 4,
    "number_of_replicas": 1
  }
}
```

Replicate data

blog2
size: 460B (776B)
docs: 0 (0)
Info ▾ Actions ▾

node01
Info ▾ Actions ▾

0 1 2 3

node02
Info ▾ Actions ▾

0 1 2 3

Node with single responsibility

NODE 1

master

NODE 2

data

NODE 3

query

Create index

```
PUT /blog3
{
 "settings": {
  "number_of_shards": 4,
  "number_of_replicas": 1
 }
}
```
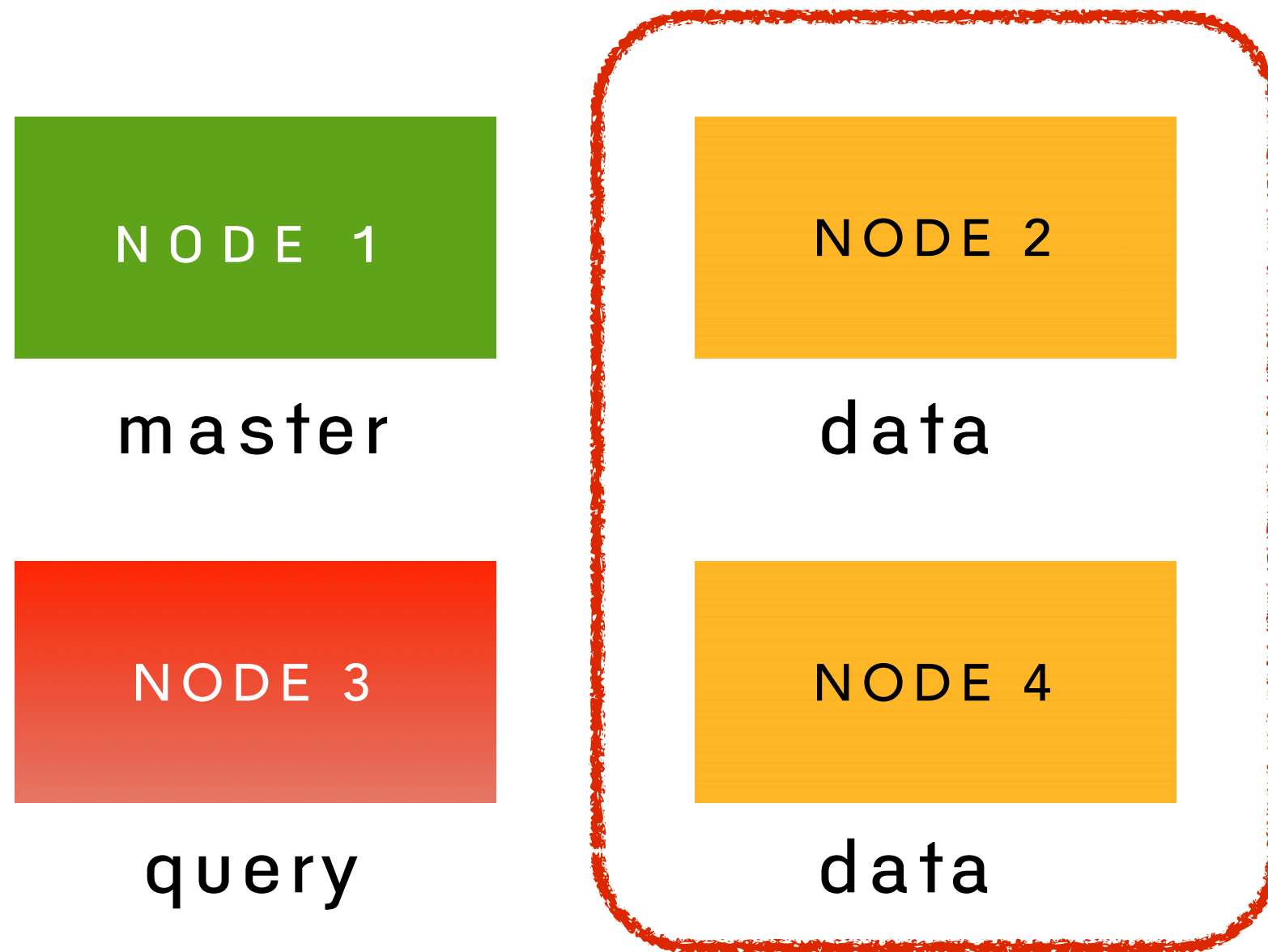
Replicate data

Cluster health

GET _cluster/health

```
 1  {
 2      "cluster_name": "kaidee",
 3      "status": "yellow",
 4      "timed_out": false,
 5      "number_of_nodes": 3,
 6      "number_of_data_nodes": 1,
 7      "active_primary_shards": 4,
 8      "active_shards": 4,
 9      "relocating_shards": 0,
10      "initializing_shards": 0,
11      "unassigned_shards": 4,
12      "number_of_pending_tasks": 0
13  }
```

Add data node

Replicate data

Cluster health

GET _cluster/health

```
 1 ▾ {
 2       "cluster_name": "kaidee",
 3       "status": "green",
 4       "timed_out": false,
 5       "number_of_nodes": 4,
 6       "number_of_data_nodes": 2,
 7       "active_primary_shards": 4,
 8       "active_shards": 8,
 9       "relocating_shards": 0,
10       "initializing_shards": 0,
11       "unassigned_shards": 0,
12       "number_of_pending_tasks": 0
13 ▴ }
```
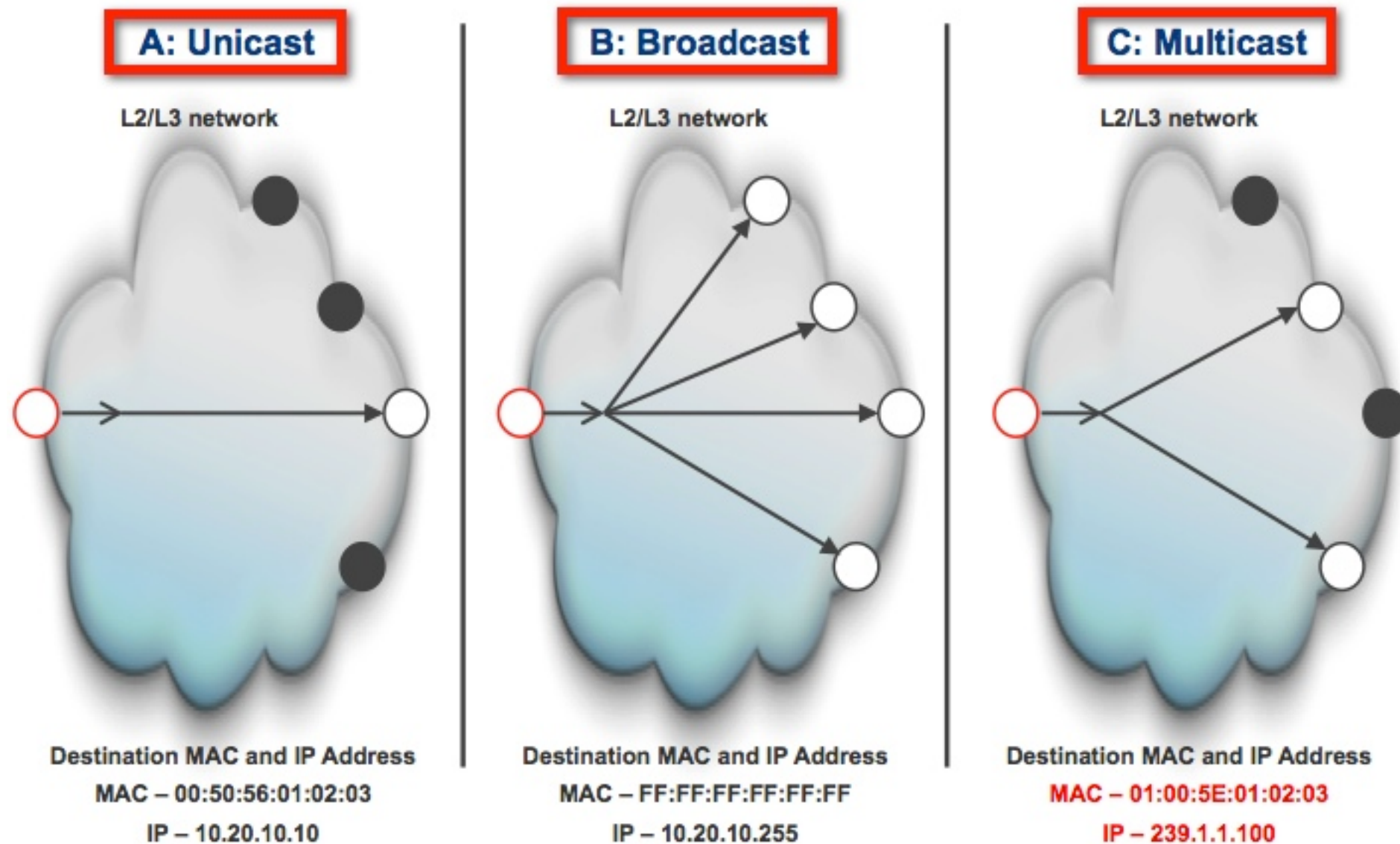
Zen Discovery

Multicast

Unicast

Multicast vs Unicast

A: Unicast

L2/L3 network

Destination MAC and IP Address
MAC – 00:50:56:01:02:03
IP – 10.20.10.10

B: Broadcast

L2/L3 network

Destination MAC and IP Address
MAC – FF:FF:FF:FF:FF:FF
IP – 10.20.10.255

C: Multicast

L2/L3 network

Destination MAC and IP Address
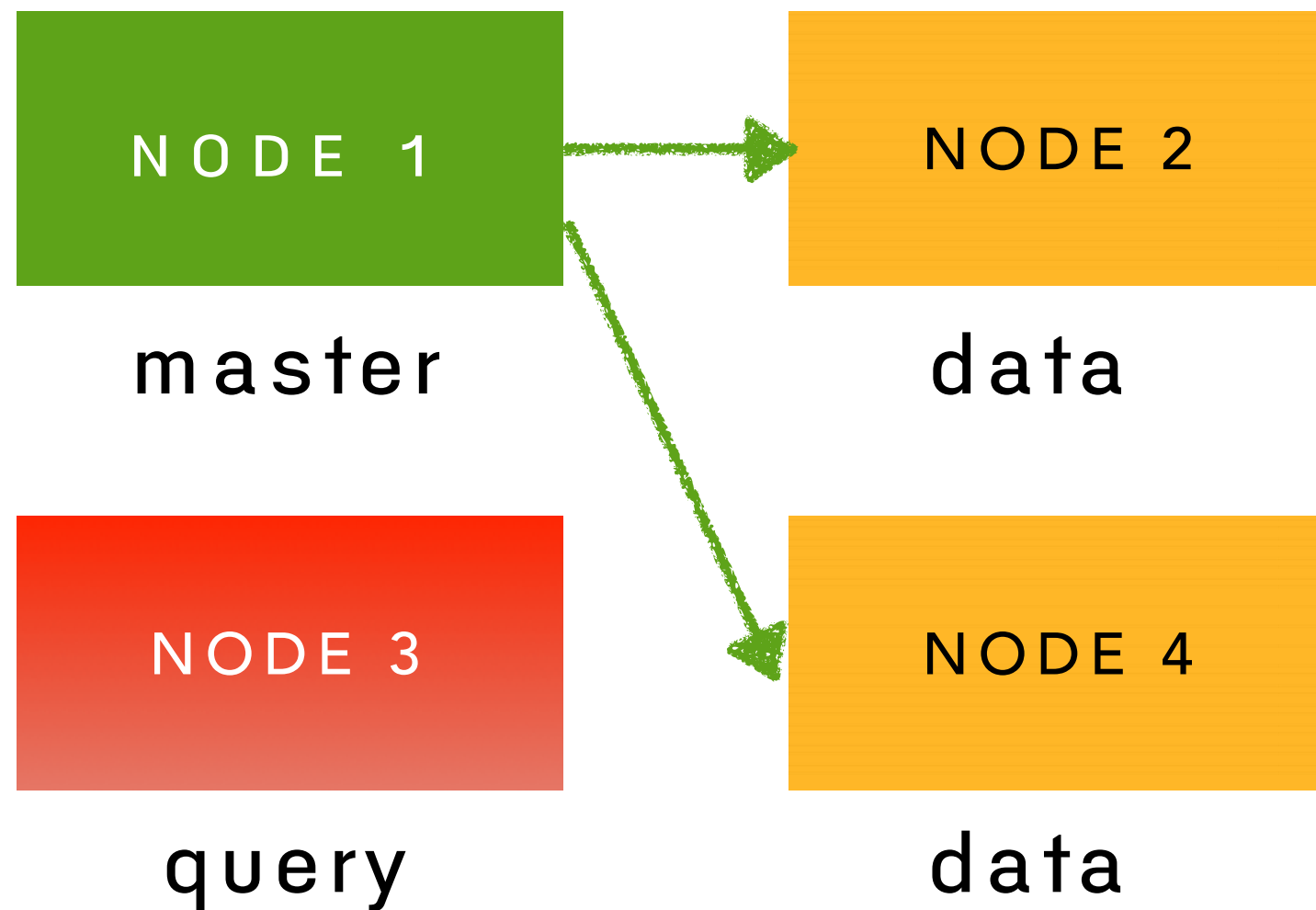MAC – 01:00:5E:01:02:03
IP – 239.1.1.100

Use unicast

#**elasticsearch.yml**

cluster.name: kaidee

discovery.zen.ping.multicast.enabled: false

discovery.zen.ping.unicast.hosts: ["node02:9201",

"node04:9203"]

Unicast
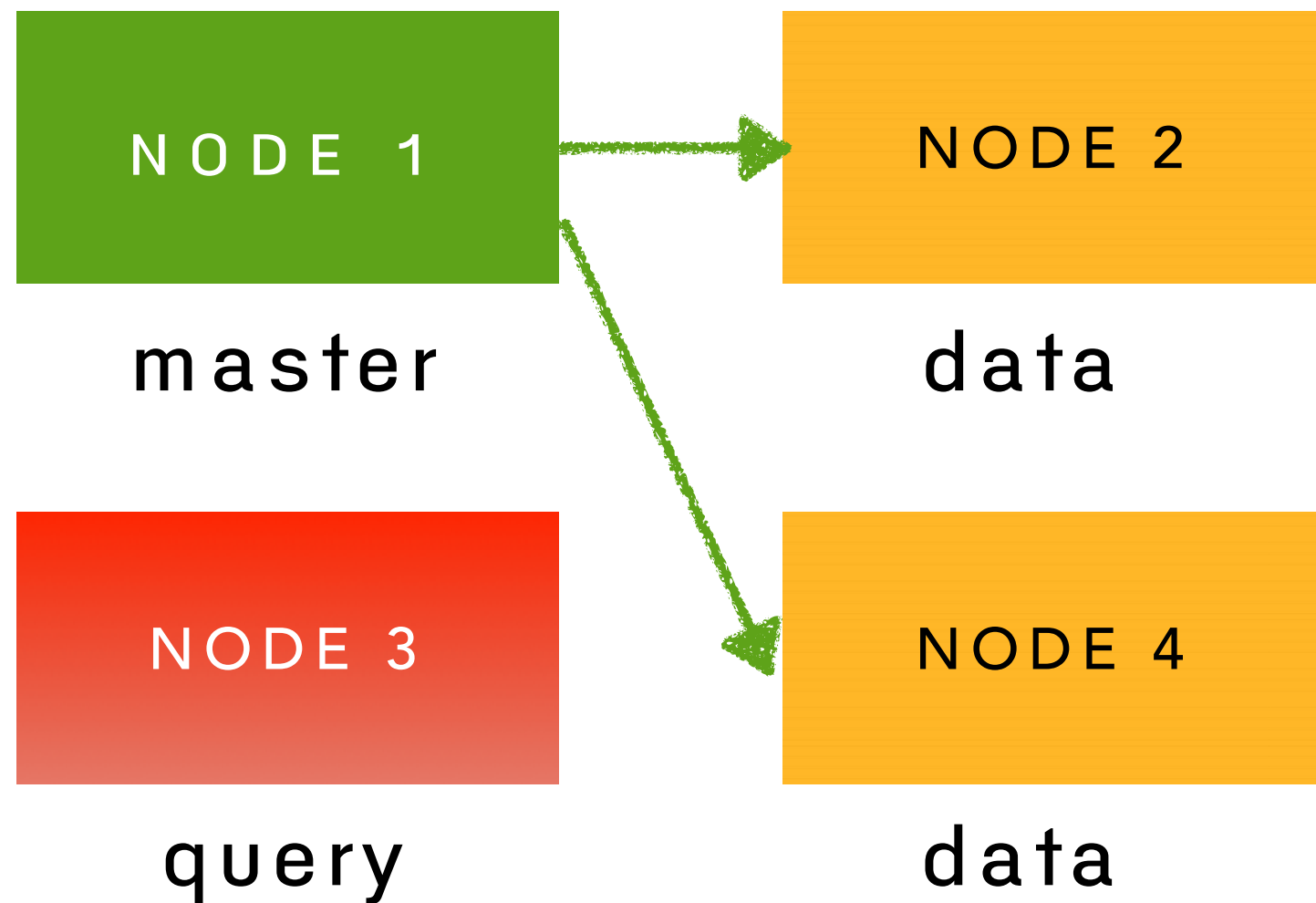
Master election

MINIMUM MASTER NODE $>=$ TOTAL NODE $/\ 2$

Topology

4 / 2 = 2

NODE 1
master

NODE 2
data

NODE 3
query

NODE 4
data

Minimum master node
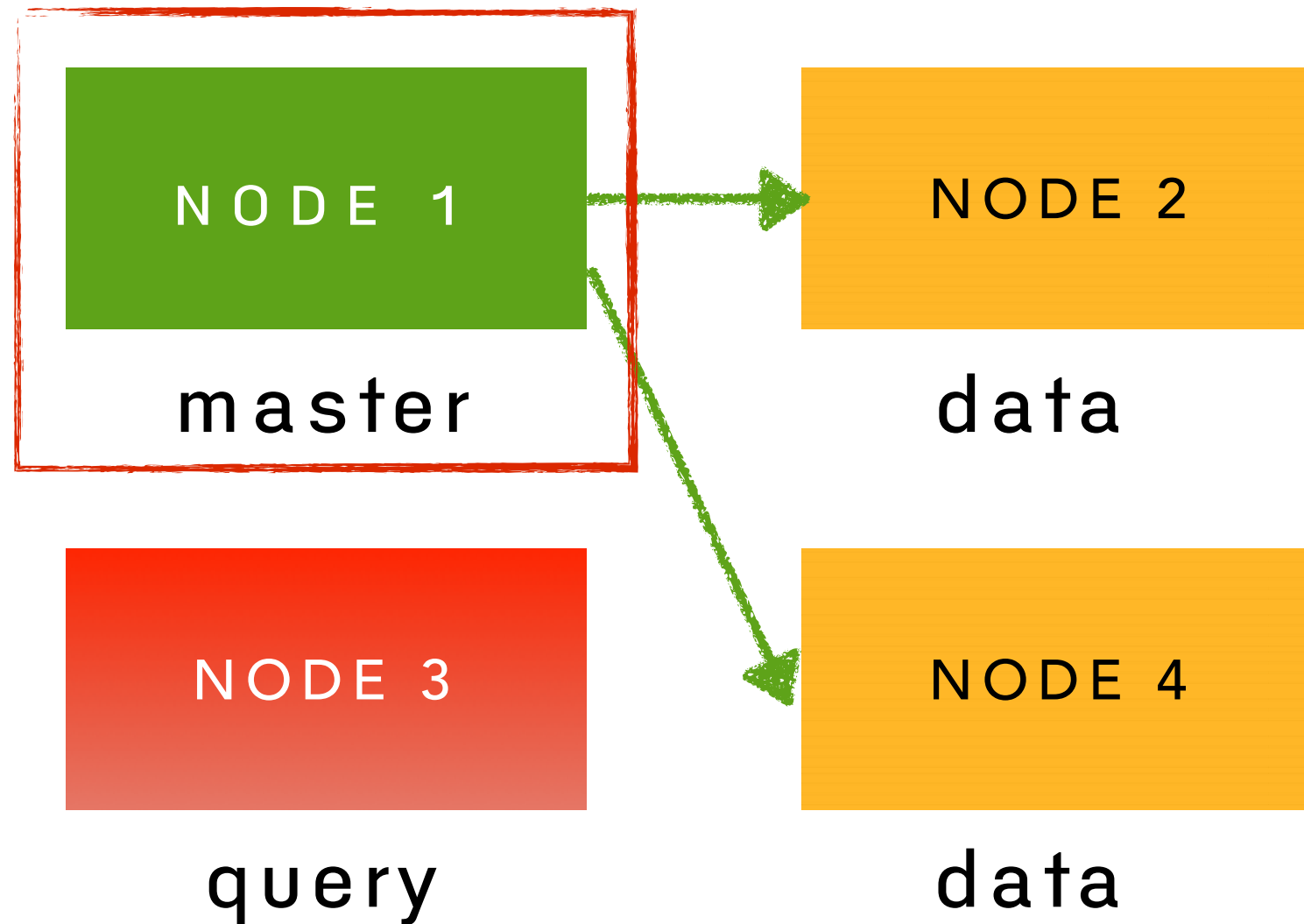
elasticsearch.yml

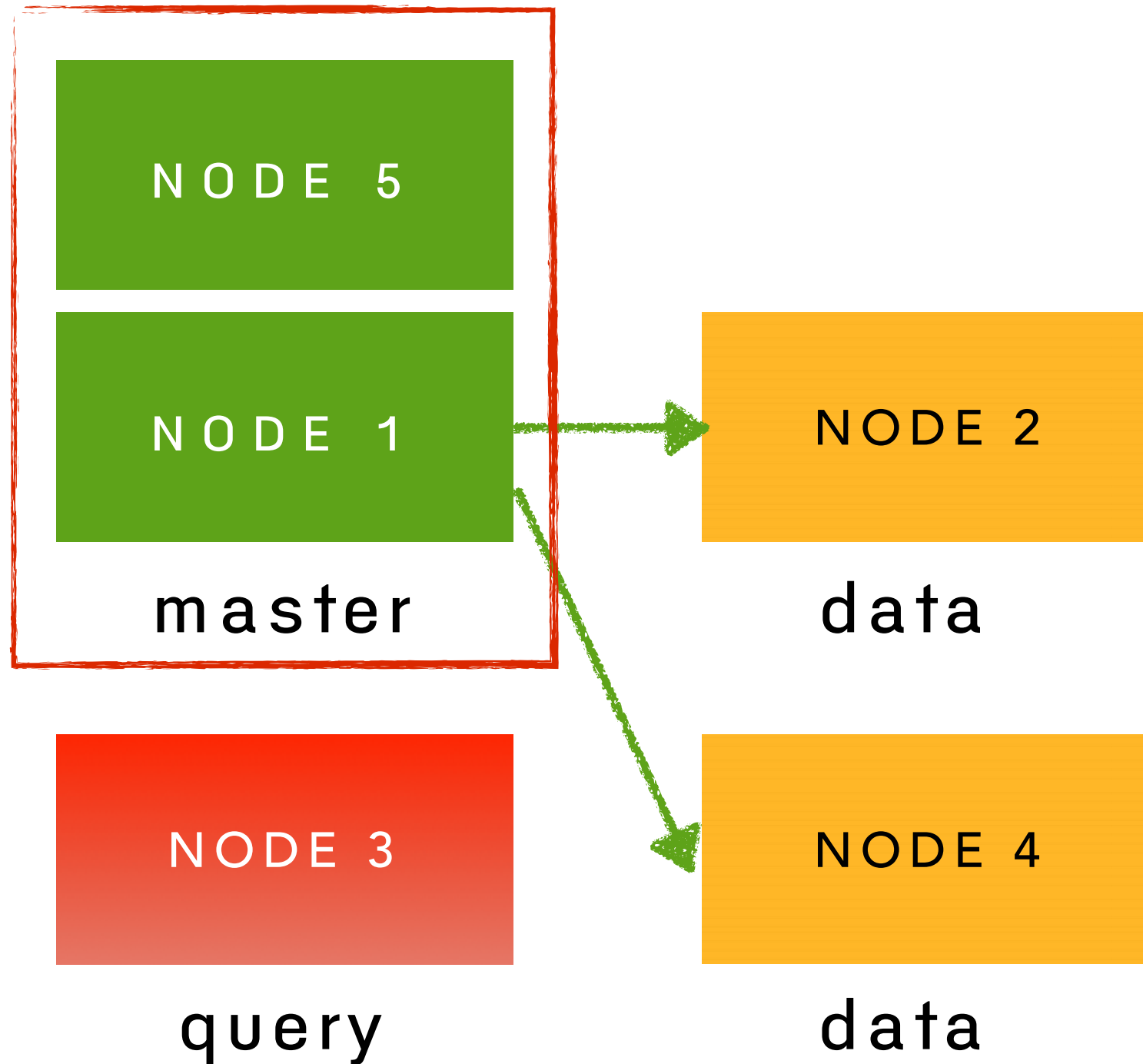cluster.name: kaidee

discovery.zen.ping.multicast.enabled: false

discovery.zen.ping.unicast.hosts: ["node02:9201",

"node04:9203"]

discovery.zen.minimum_master_nodes: 2

Not valid ?

NODE 1
master

NODE 2
data

NODE 3
query

NODE 4
data

Add master node

NODE 5

NODE 1

master

NODE 2

data

NODE 3

query

NODE 4

data

Restart node in cluster

อย่า shutdown หรือ stop service

Restart node in cluster

Need rolling restart node by node

Step by step

1. Stop rebalance for this node

```
PUT _cluster/settings

{

  "transient": {

    "cluster.routing.allocation.enable": "none"

  }

}
```

Step by step

2. Shutdown node

Step by step

3. Restart node to join in cluster

Step by step

4. Enable rebalance

```
PUT _cluster/settings

{

  "transient": {

    "cluster.routing.allocation.enable": "all"

  }

}
```