

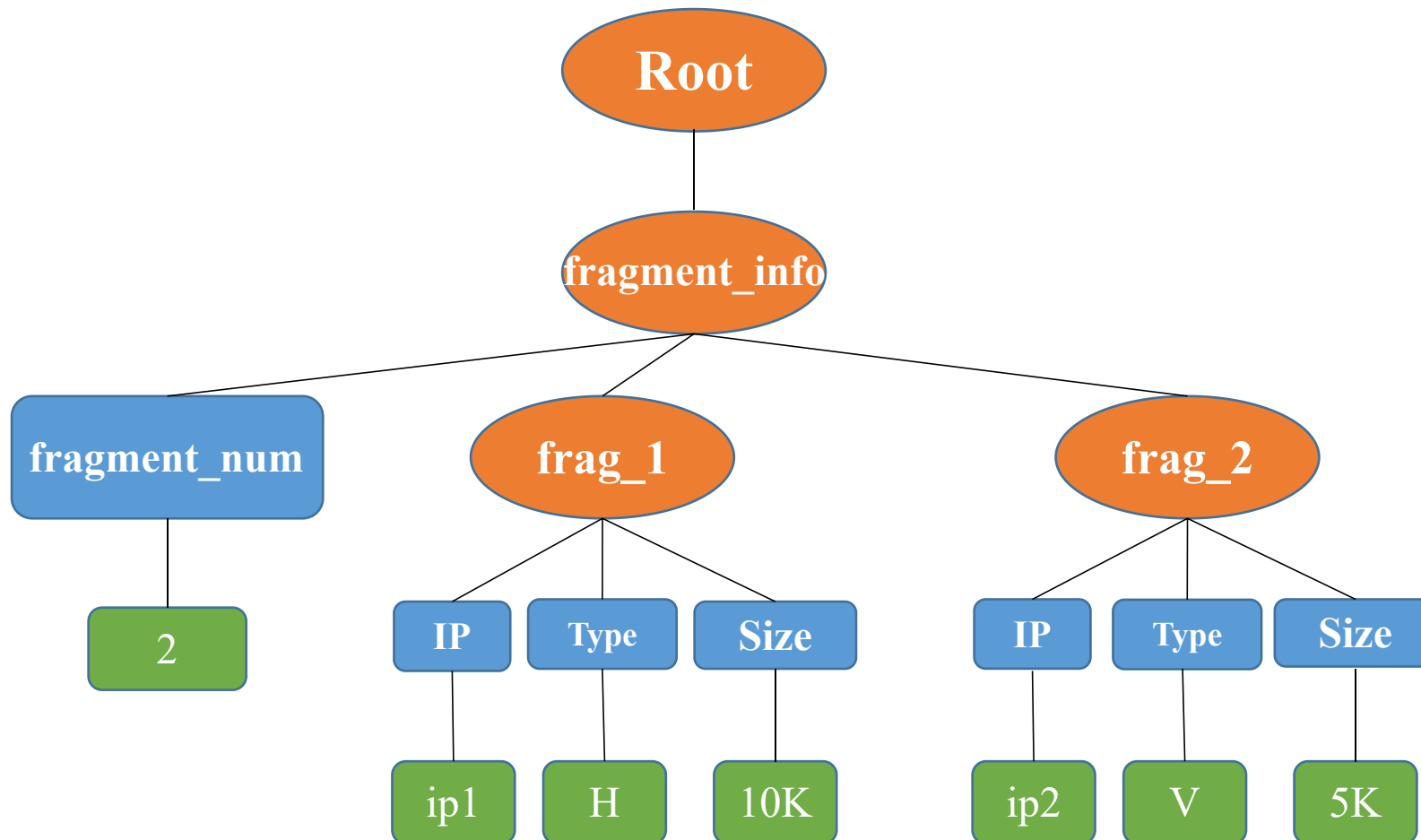
Tutorial For ETCD

What is ETCD?

- ETCD is a **strongly consistent**, distributed **key-value store** that provides a reliable way to store data that needs to be accessed by a **distributed system or cluster** of machines^[1].

[1] <https://etcd.io/>

What is ETCD?



/fragment_info

- fragment_num : 2

/frag_1

- IP : ip1

- Type : H (horizontal)

- Size : 10K

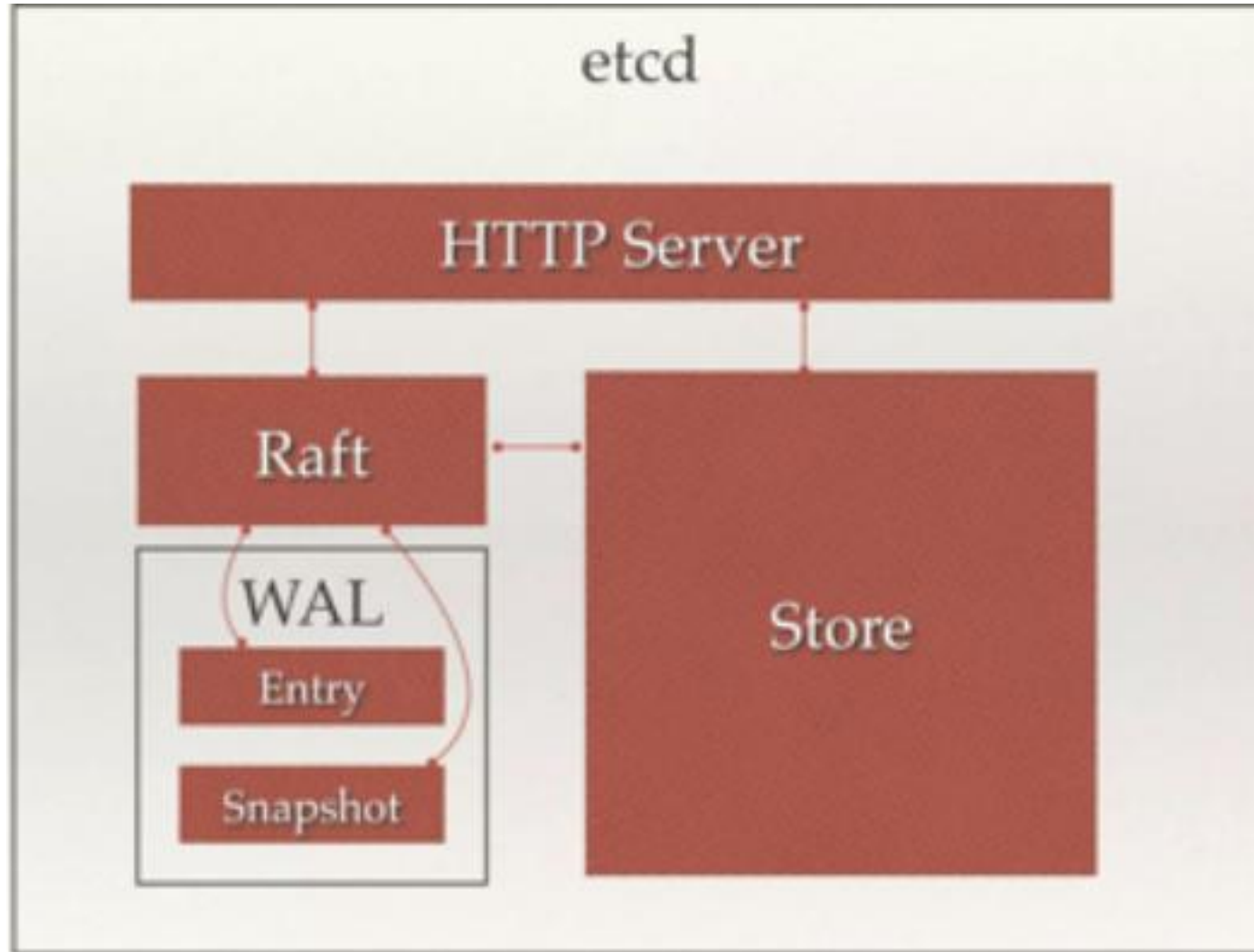
/frag_2

- IP : ip2

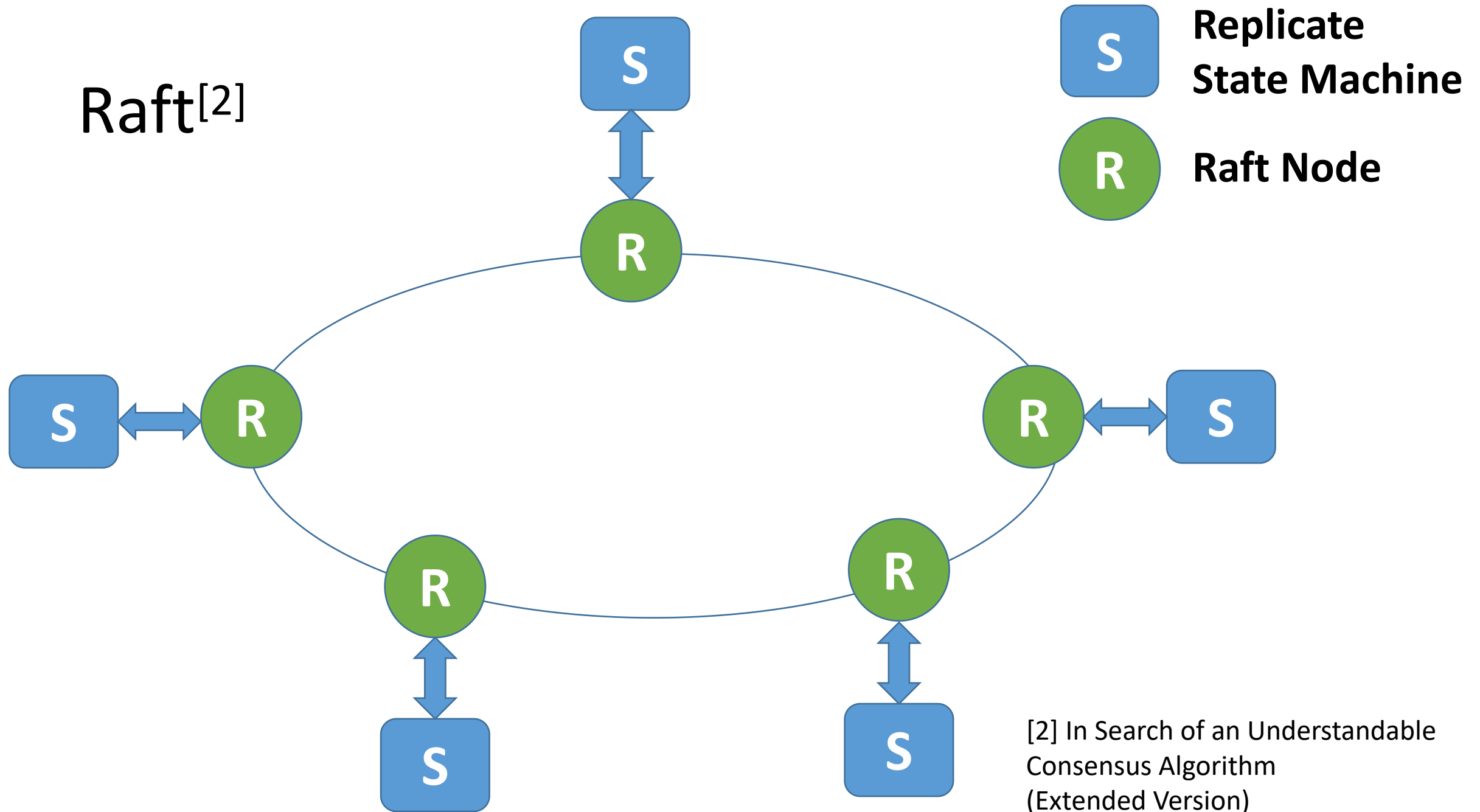
- Type : V (vertical)

- Size : 5K

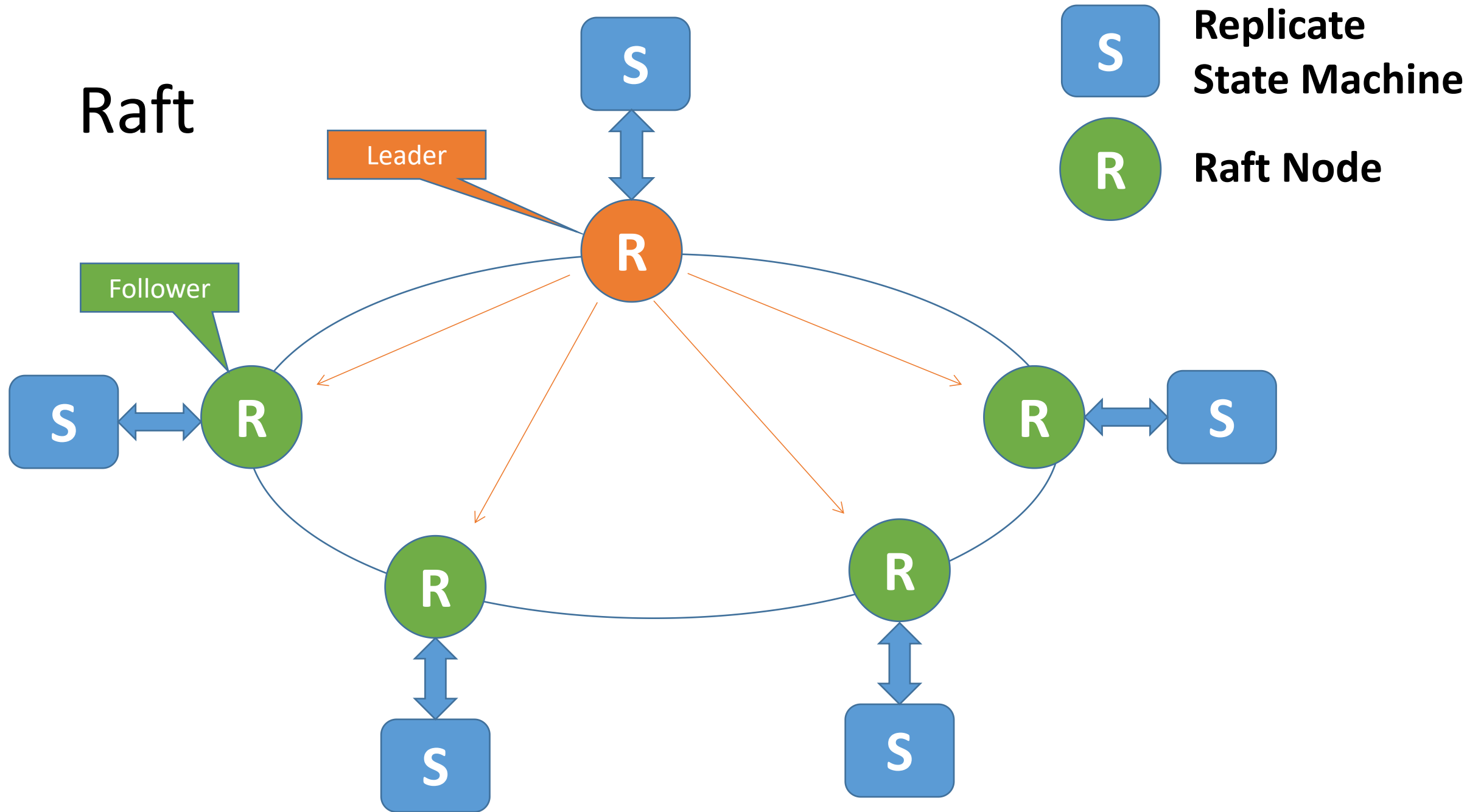
What is ETCD?



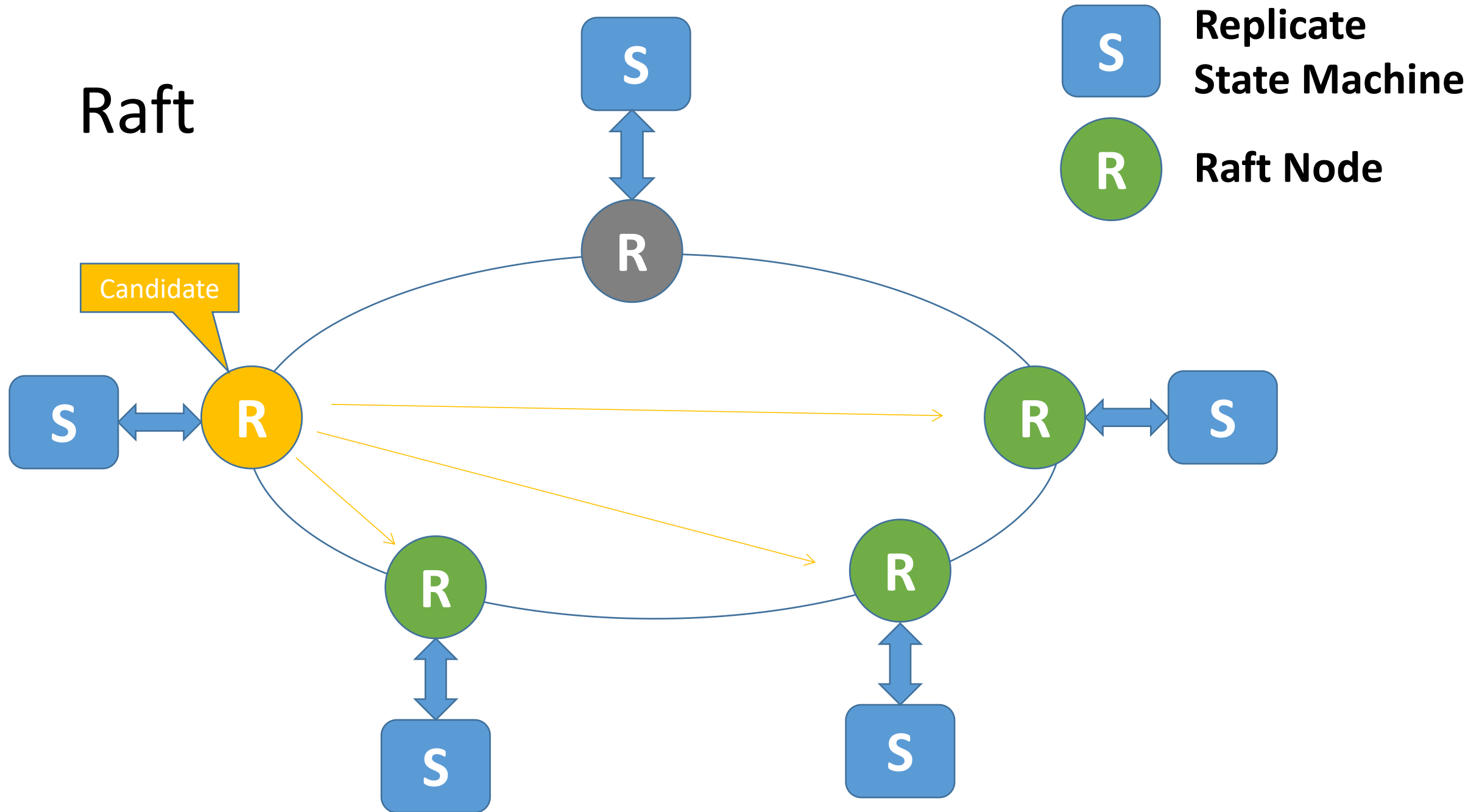
Raft^[2]



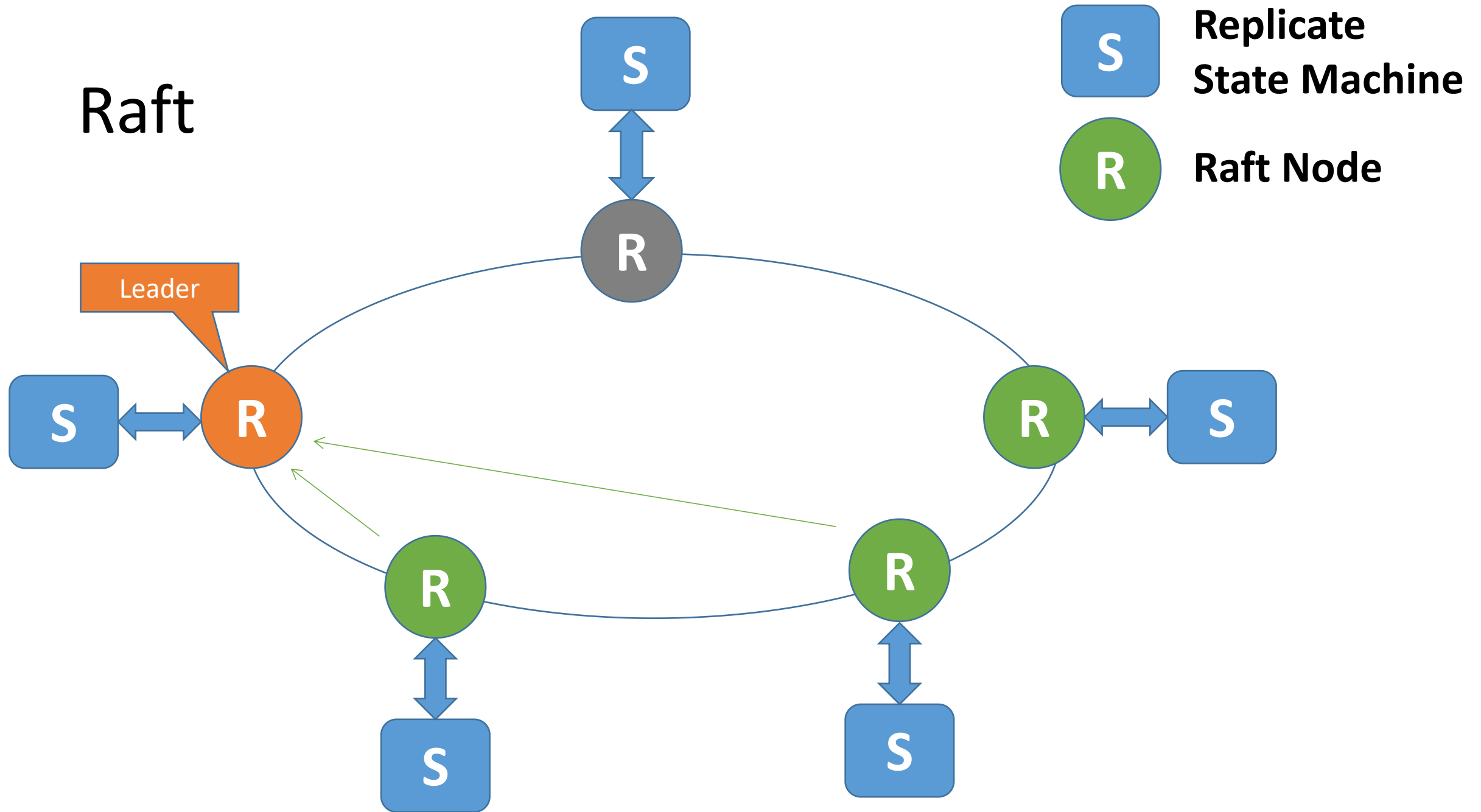
Raft



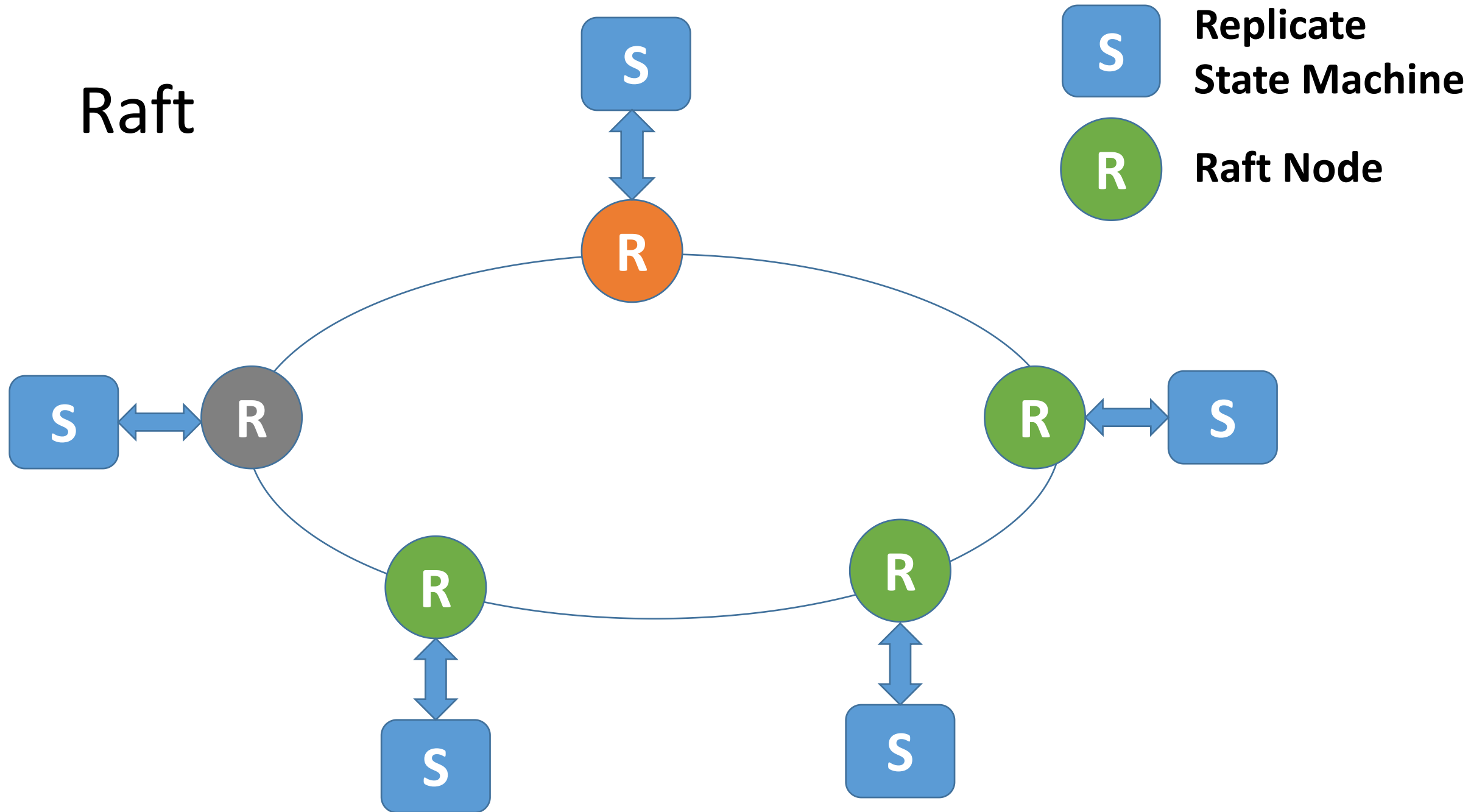
Raft



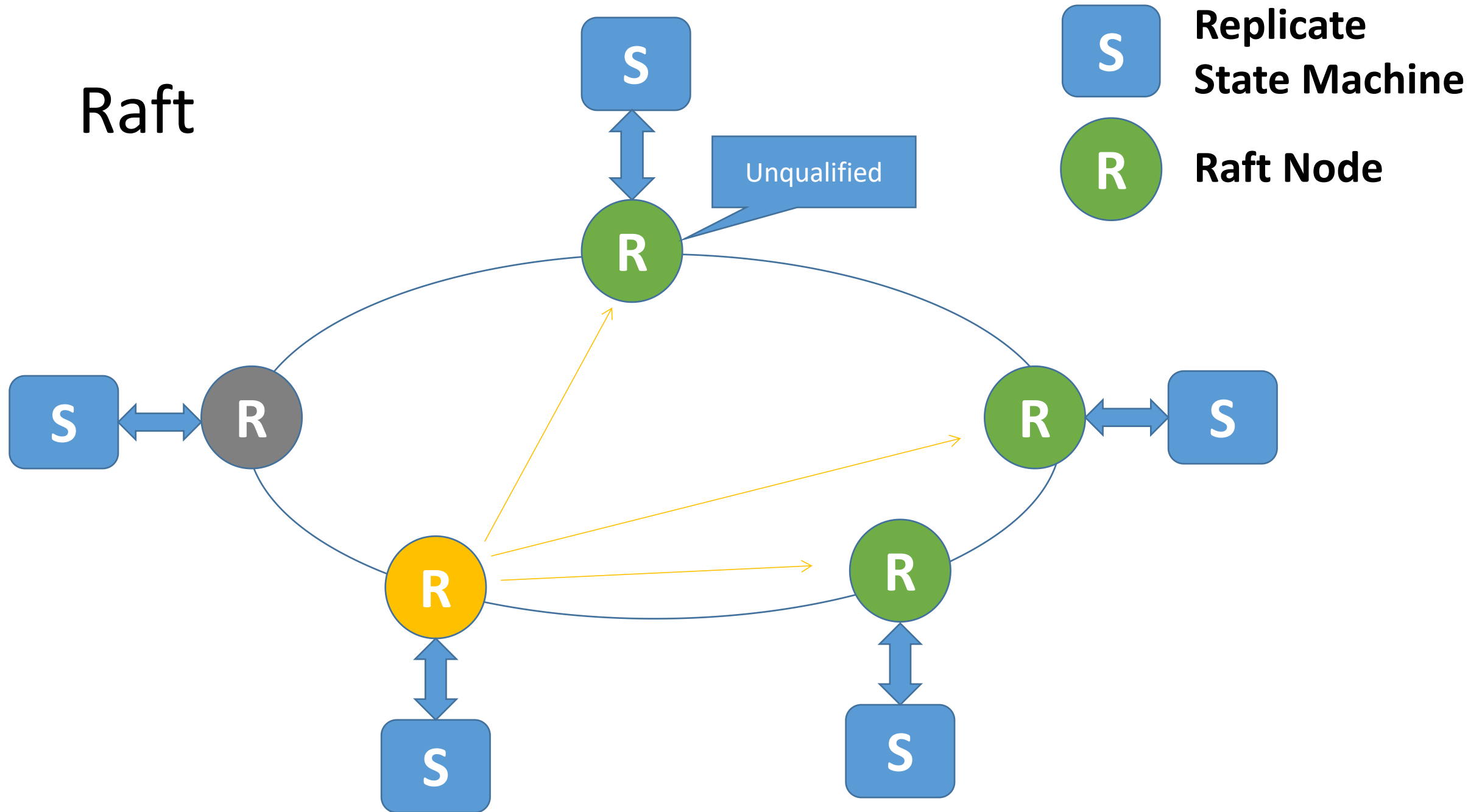
Raft



Raft




Raft



How to deploy it?

- <https://github.com/etcd-io/etcd/releases>

▼ Assets 8

 etcd-v3.4.13-darwin-amd64.zip	26.8 MB
 etcd-v3.4.13-linux-amd64.tar.gz	16.6 MB
 etcd-v3.4.13-linux-arm64.tar.gz	15.4 MB
 etcd-v3.4.13-linux-ppc64le.tar.gz	15.2 MB
 etcd-v3.4.13-windows-amd64.zip	16.5 MB
 SHA256SUMS	489 Bytes
 Source code (zip)	
 Source code (tar.gz)	

How to deploy it?

- Decompress

Name	Size (KB)	Last modified
..		
Documentation		2017-08-22 ...
etcd	16 738	2017-08-22 ...
etcdctl	14 305	2017-08-22 ...
README-etcdctl.md	33	2017-08-22 ...
README.md	5	2017-08-22 ...
READMEv2-etcdctl.md	7	2017-08-22 ...

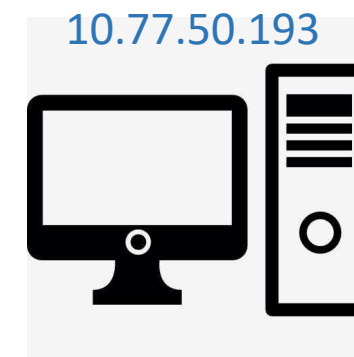
```
liuty@vulcan001:~/etcd/etcd-v3.2.6-linux-amd64$ ./etcd --version
etcd Version: 3.2.6
Git SHA: 9d43462
Go Version: go1.8.3
Go OS/Arch: linux/amd64
liuty@vulcan001:~/etcd/etcd-v3.2.6-linux-amd64$
```

```
liuty@vulcan001:~/etcd/etcd-v3.2.6-linux-amd64$ ./etcdctl --version
etcdctl version: 3.2.6
API version: 2
liuty@vulcan001:~/etcd/etcd-v3.2.6-linux-amd64$
```

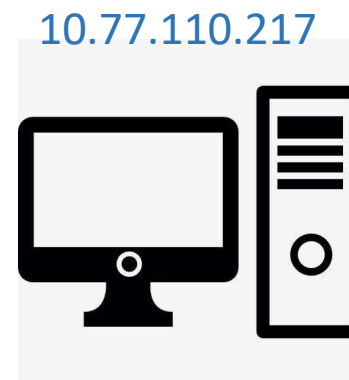
How to deploy it?

- Configuration

```
./etcd --name etcd01  
  --initial-advertise-peer-urls http://10.77.50.193:2380 \  
  --listen-peer-urls http://0.0.0.0:2380 \  
  --listen-client-urls http://0.0.0.0:2379 \  
  --advertise-client-urls http://10.77.50.193:2379 \  
  --initial-cluster-token etcd-cluster-1 \  
  --initial-cluster etcd01=http://10.77.50.193:2380, etcd02=http://10.77.110.217:2380  
  --initial-cluster-state new
```



```
./etcd --name etcd02  
  --initial-advertise-peer-urls http://10.77.110.217:2380 \  
  --listen-peer-urls http://0.0.0.0:2380 \  
  --listen-client-urls http://0.0.0.0:2379 \  
  --advertise-client-urls http://10.77.110.217:2379 \  
  --initial-cluster-token etcd-cluster-1 \  
  --initial-cluster etcd01=http://10.77.50.193:2380, etcd02=http://10.77.110.217:2380  
  --initial-cluster-state new
```



How to access data?

- By **etcdctl**

```
lty@DBIIR-PowerEdge-T640:~/etcd/etcd-v3.2.6-linux-amd64$ ./etcdctl
NAME:
  etcdctl - A simple command line client for etcd.

WARNING:
  Environment variable ETCDCTL_API is not set; defaults to etcdctl v2.
  Set environment variable ETCDCTL_API=3 to use v3 API or ETCDCTL_API=2 to use v2 API.

USAGE:
  etcdctl [global options] command [command options] [arguments...]

VERSION:
  3.2.6

COMMANDS:
  backup      backup an etcd directory
  cluster-health check the health of the etcd cluster
  mk          make a new key with a given value
  mkdir      make a new directory
  rm         remove a key or a directory
  rmdir      removes the key if it is an empty directory or a key-value pair
  get        retrieve the value of a key
  ls         retrieve a directory
  set        set the value of a key
  setdir     create a new directory or update an existing directory TTL
  update     update an existing key with a given value
  updatedir  update an existing directory
  watch      watch a key for changes
  exec-watch watch a key for changes and exec an executable
  member     member add, remove and list subcommands
  user       user add, grant and revoke subcommands
  role       role add, grant and revoke subcommands
  auth       overall auth controls
  help, h    Shows a list of commands or help for one command
```

How to access data?

- By **etcdctl**

```
lty@DBIIR-PowerEdge-T640:~/etcd/etcd-v3.2.6-linux-amd64$ ./etcdctl mkdir fragment_info
lty@DBIIR-PowerEdge-T640:~/etcd/etcd-v3.2.6-linux-amd64$ ./etcdctl ls
/fragment_info
```

```
lty@DBIIR-PowerEdge-T640:~/etcd/etcd-v3.2.6-linux-amd64$ ./etcdctl mkdir fragment_info/frag_1
lty@DBIIR-PowerEdge-T640:~/etcd/etcd-v3.2.6-linux-amd64$ ./etcdctl mkdir fragment_info/frag_2
lty@DBIIR-PowerEdge-T640:~/etcd/etcd-v3.2.6-linux-amd64$ ./etcdctl set fragment_info/fragment_num 2
2
```

```
lty@DBIIR-PowerEdge-T640:~/etcd/etcd-v3.2.6-linux-amd64$ ./etcdctl ls /fragment_info
/fragment_info/fragment_num
/fragment_info/frag_1
/fragment_info/frag_2
lty@DBIIR-PowerEdge-T640:~/etcd/etcd-v3.2.6-linux-amd64$ ./etcdctl get /fragment_info/fragment_num
2
```

/fragment_info

- **fragment_num** : 2

/frag_1

- **IP** : ip1

- **Type** : H (horizontal)

- **Size** : 10K

/frag_2

- **IP** : ip2

- **Type** : V (vertical)

- **Size** : 5K

How to access data?

- By **etcdctl**

```
lty@DBIIR-PowerEdge-T640:~/etcd/etcd-v3.2.6-linux-amd64$ ./etcdctl get /fragment_info/fragment_num
2
lty@DBIIR-PowerEdge-T640:~/etcd/etcd-v3.2.6-linux-amd64$ ./etcdctl set /fragment_info/frag_1/IP ip1
ip1
lty@DBIIR-PowerEdge-T640:~/etcd/etcd-v3.2.6-linux-amd64$ ./etcdctl set /fragment_info/frag_1/Type H
H
lty@DBIIR-PowerEdge-T640:~/etcd/etcd-v3.2.6-linux-amd64$ ./etcdctl set /fragment_info/frag_1/Size 10K
10K
```

/fragment_info

- **fragment_num** : 2

/frag_1

- **IP** : ip1

- **Type** : H (horizontal)

- **Size** : 10K

/frag_2

- **IP** : ip2

- **Type** : V (vertical)

- **Size** : 5K

```
(base) fanju@ubuntu:~/lty/etcd/etcd-v3.2.6-linux-amd64$ ./etcdctl ls /fragment_info/frag_1
/fragment_info/frag_1/IP
/fragment_info/frag_1/Type
/fragment_info/frag_1/Size
```


How to access data?

- By **curl**
 - `curl -s http://127.0.0.1:2379/v2/keys/fragment_info/frag_1?recursive=true`

```
lty@DBIIR-PowerEdge-T640:~/etcd/etcd-v3.2.6-linux-amd64$ curl -s http://127.0.0.1:2379/v2/keys/fragment_info/frag_1?recursive=true
{"action": "get", "node": {"key": "/fragment_info/frag_1", "dir": true, "nodes": [{"key": "/fragment_info/frag_1/IP", "value": "ip1", "modifiedIndex": 12, "createdIndex": 12}, {"key": "/fragment_info/frag_1/Type", "value": "H", "modifiedIndex": 13, "createdIndex": 13}, {"key": "/fragment_info/frag_1/Size", "value": "10K", "modifiedIndex": 16, "createdIndex": 16}], "modifiedIndex": 9, "createdIndex": 9}}
```

How to access data?

- By **curl**

```
{
  "action": "get",
  "node": {
    "key": "/fragment_info/frag_1",
    "dir": true,
    "nodes": [{
      "key": "/fragment_info/frag_1/IP",
      "value": "ip1",
      "modifiedIndex": 12,
      "createdIndex": 12
    }, {
      "key": "/fragment_info/frag_1/Type",
      "value": "H",
      "modifiedIndex": 13,
      "createdIndex": 13
    }, {
      "key": "/fragment_info/frag_1/Size",
      "value": "10K",
      "modifiedIndex": 16,
      "createdIndex": 16
    }
  ],
  "modifiedIndex": 9,
  "createdIndex": 9
}
```

How to access data?

- By **curl**
 - `curl http://127.0.0.1:2379/v2/keys/fragment_info/frag_1/Size`

```
lty@DBIIR-PowerEdge-T640:~/etcd/etcd-v3.2.6-linux-amd64$ curl -s http://127.0.0.1:2379/v2/keys/fragment_info/frag_1/Size  
{"action": "get", "node": {"key": "/fragment_info/frag_1/Size", "value": "10K", "modifiedIndex": 16, "createdIndex": 16}}
```

```
{  
  "action": "get",  
  "node": {  
    "key": "/fragment_info/frag_1/Size",  
    "value": "10K",  
    "modifiedIndex": 16,  
    "createdIndex": 16  
  }  
}
```

How to access data?

- By **curl**
 - `curl http://127.0.0.1:2379/v2/keys/fragment_info/frag_1/Size`

```
lty@DBIIR-PowerEdge-T640:~/etcd/etcd-v3.2.6-linux-amd64$ curl http://127.0.0.1:2379/v2/keys/fragment_info/frag_1/Size -XPUT -d value=12K
{"action": "set", "node": {"key": "/fragment_info/frag_1/Size", "value": "12K", "modifiedIndex": 17, "createdIndex": 17}, "prevNode": {"key": "/fragment_info/frag_1/Size", "value": "10K", "modifiedIndex": 16, "createdIndex": 16}}
```

```
{
  "action": "set",
  "node": {
    "key": "/fragment_info/frag_1/Size",
    "value": "12K",
    "modifiedIndex": 17,
    "createdIndex": 17
  },
  "prevNode": {
    "key": "/fragment_info/frag_1/Size",
    "value": "10K",
    "modifiedIndex": 16,
    "createdIndex": 16
  }
}
```

```
{
  "action": "get",
  "node": {
    "key": "/fragment_info/frag_1/Size",
    "value": "12K",
    "modifiedIndex": 17,
    "createdIndex": 17
  }
}
```

常用ETCD命令:

https://blog.csdn.net/warrior_0319/article/details/80223947

How to access data?

- By **curl API : libcurl**^[3]

```
#include <curl/curl.h>
```

- Get easy handle

```
CURL *easy_handle = curl_easy_init();
```

- Set easy handle

```
char result[MAX_LEN] = "";
```

```
curl_easy_setopt(easy_handle, CURLOPT_URL, "http://127.0.0.1/v2/keys/fragment_info/frag_1/Size");
```

```
curl_easy_setopt(easy_handle, CURLOPT_PORT, 2379);
```

```
curl_easy_setopt(easy_handle, CURLOPT_WRITEFUNCTION, &write_data);
```

```
curl_easy_setopt(easy_handle, CURLOPT_WRITEDATA, &result);
```

- Execute

```
curl_easy_perform(easy_handle);
```

[3]<https://curl.haxx.se/libcurl/c/libcurl-tutorial.html>

How to access data?

- By **libcurl**

```
#include <curl/curl.h>
```

- Get easy handle

```
CURL *easy_handle = curl_easy_init();
```

- Set easy handle

```
curl_easy_setopt(easy_handle, CURLOPT_URL, "http://127.0.0.1:2379/v2/keys/fragment_info/frag_1/Si
```

```
curl_easy_setopt(easy_handle, CURLOPT_POST, 1);
```

```
curl_easy_setopt(easy_handle, CURLOPT_POSTFIELDS, "12K");
```

```
curl_easy_setopt(easy_handle, CURLOPT_CUSTOMREQUEST, "PUT");
```

- Execute

```
curl_easy_perform(easy_handle);
```