Understand etcd with questions

Xiang Li, Alibaba
Wenjia Zhang, Google

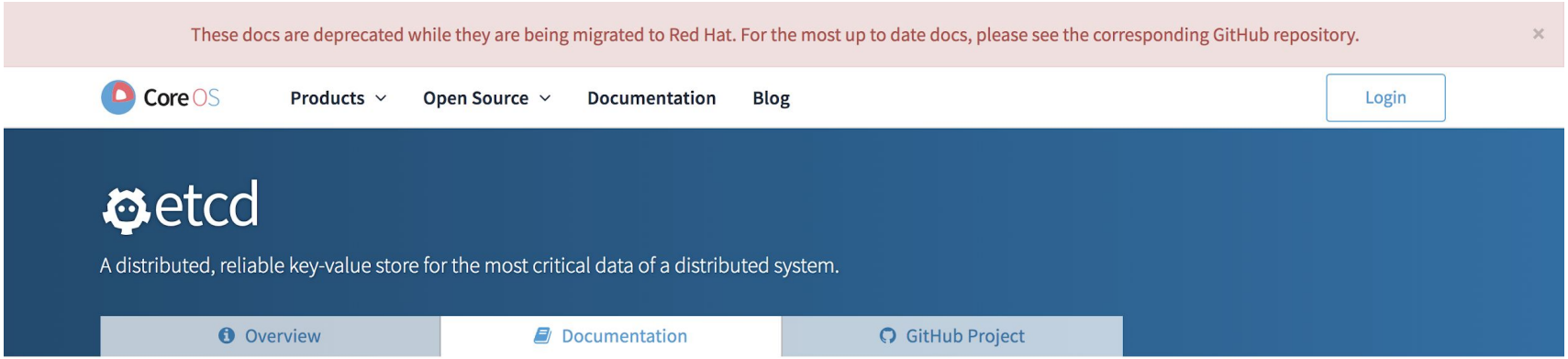
Q&A

Questions?

- Where is etcd official documentation?
- Why does Kubernetes use etcd?
- Why does etcd prefer odd number of members?
- What are WAL files?
- Does etcd work with cross-region deployments?
- How to safely upgrade/downgrade etcd?
- What is the database file?
- Why does etcd have data size limit?
- Why does etcd require compaction?
- Why is etcd sensitive to I/O latency?
- How to analyze etcd performance?

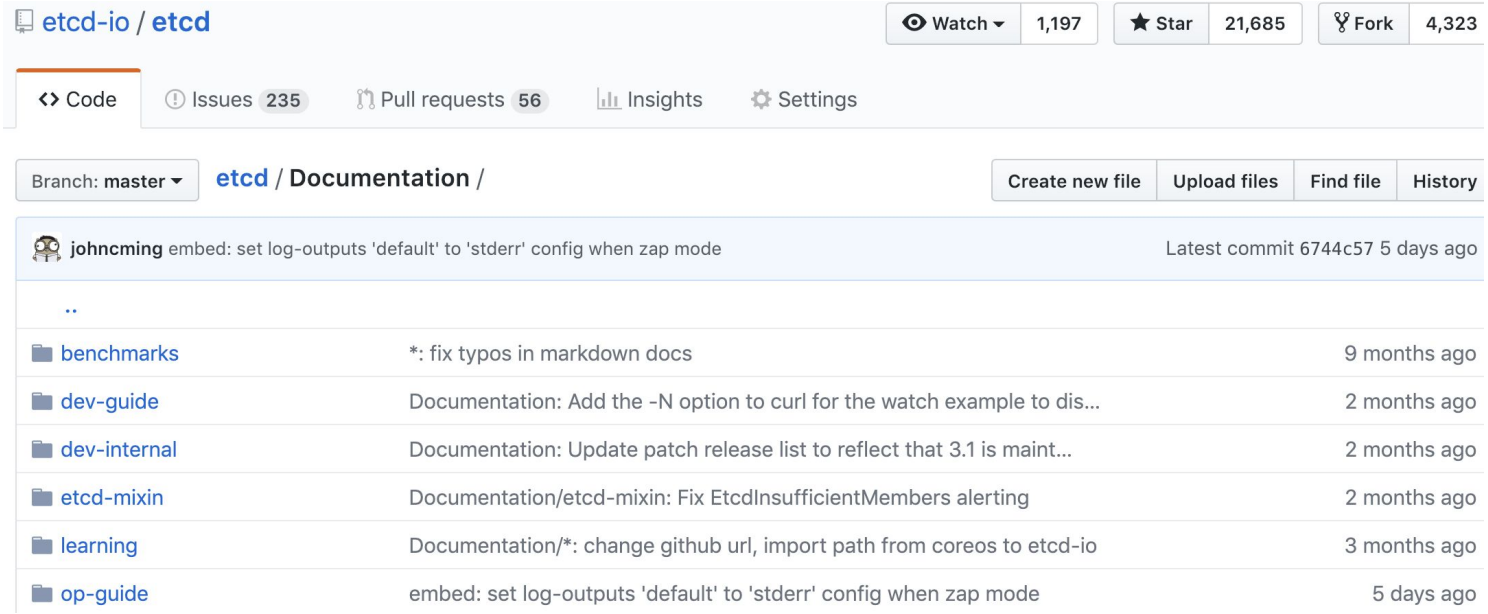
Where is etcd official documentation

https://coreos.com/etcd/docs/latest/



Where is etcd official documentation

https://github.com/etcd-io/etcd/tree/master/Documentation



Why does Kubernetes use etcd

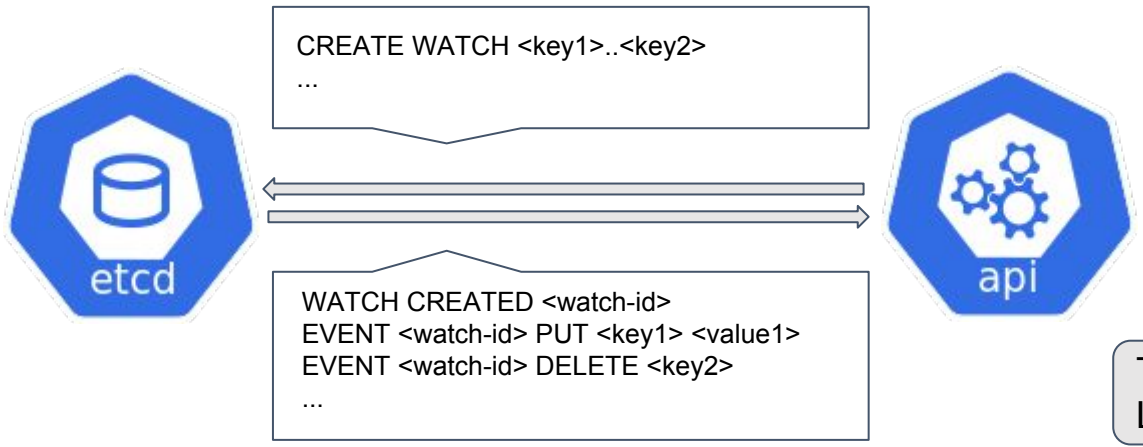
- High Availability
- Watch is the Key

Why does Kubernetes use etcd

- **High Availability**
  - Kuberentes - high availability
  - etcd clustering - high availability
  - etcd is THE data store for kubernetes control plane
- Watch-List is the Key

Why does Kubernetes use etcd

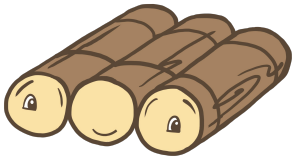
- High Availability
- **Watch is the Key**
  - etcd: Watch
  - Kubernetes: Watch



Thursday 4:30pm
Life of a k8s watch event

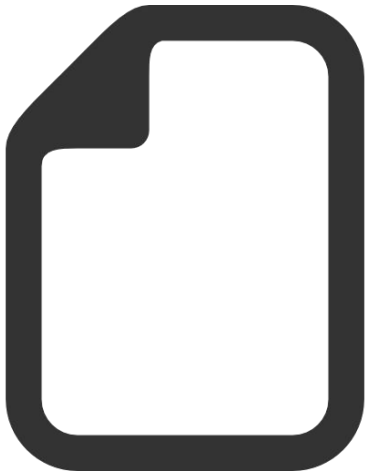
Why does etcd prefer odd number of members

Cluster Size	Majority	Failure Tolerance
1	1	0
2	2	0
3	2	1
4	3	1
5	3	2



What are WAL files

File



WAL

C1
C2
C3

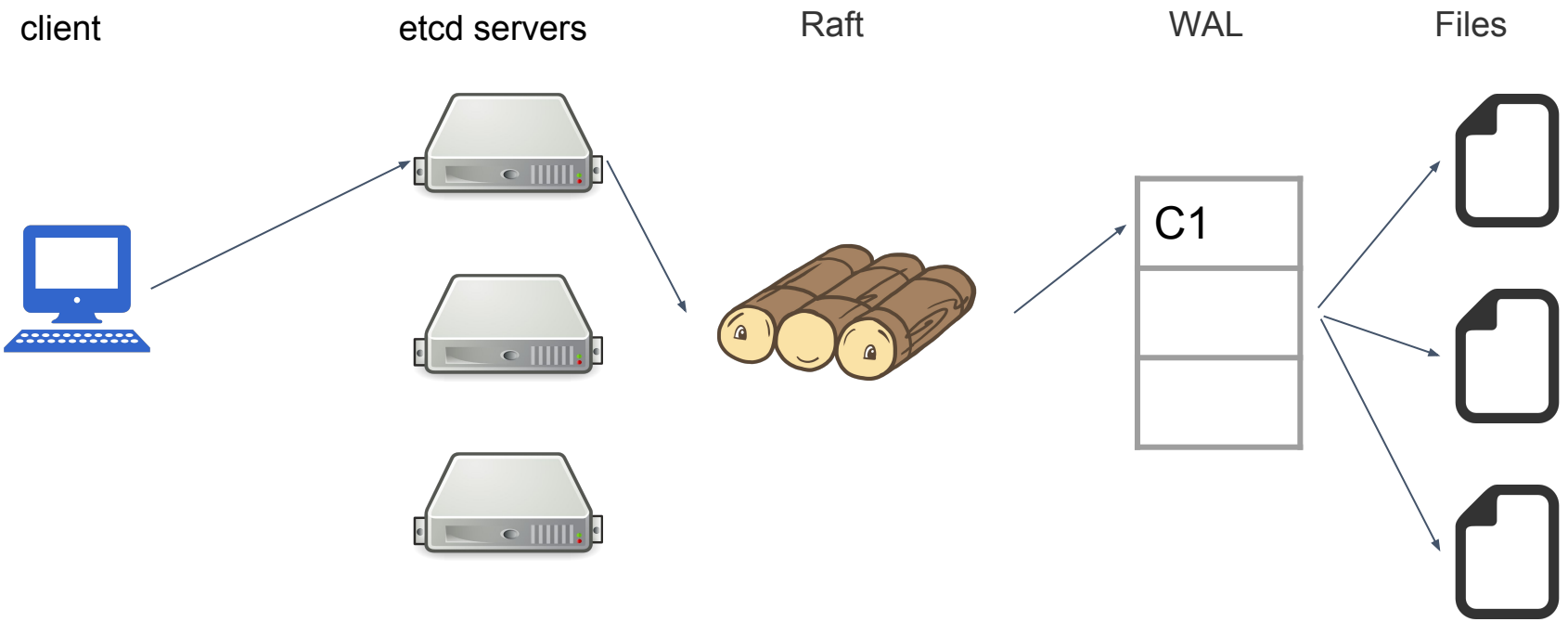
State modifications

C1: set foo = bar

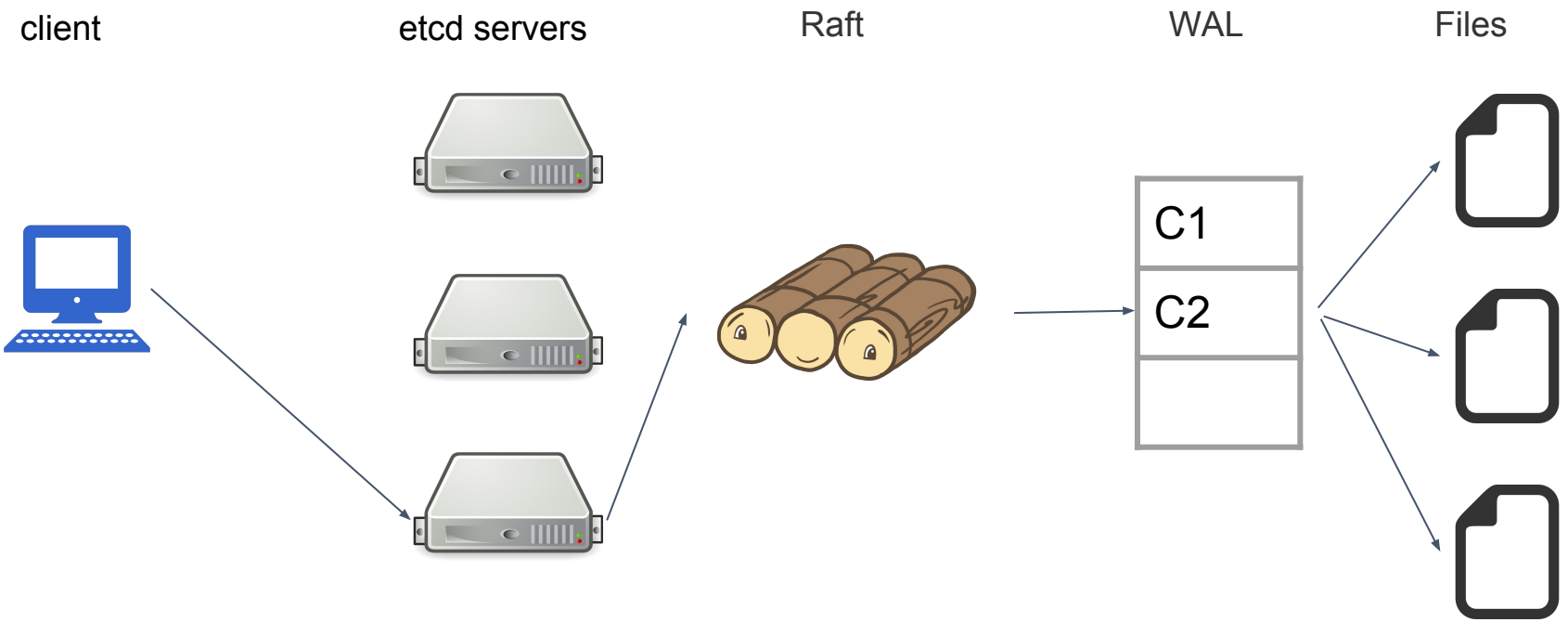
C2: set k8s = awesome

C3: set etcd = awesome

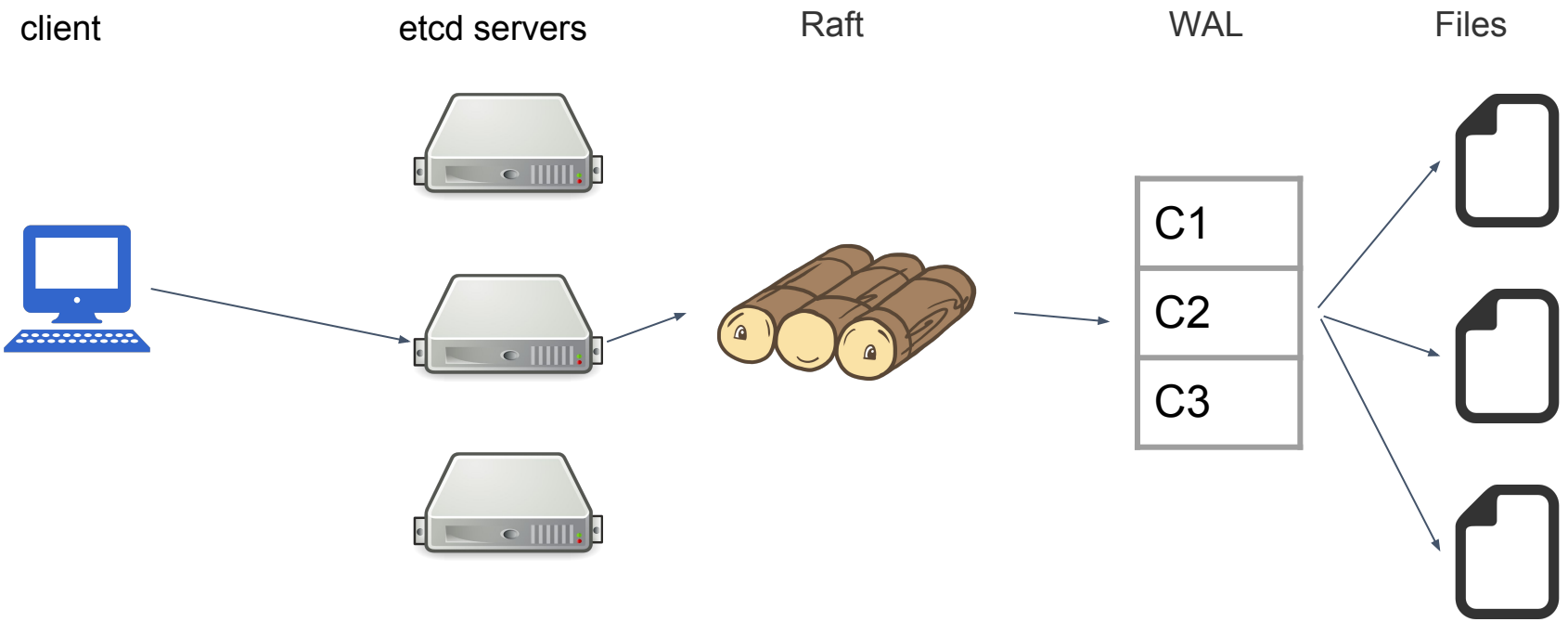
What are WAL files



What are WAL files



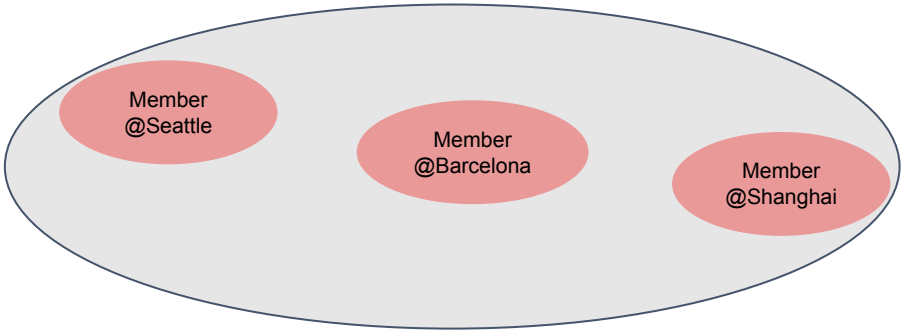
What are WAL files



Does etcd work with cross-region deployments?

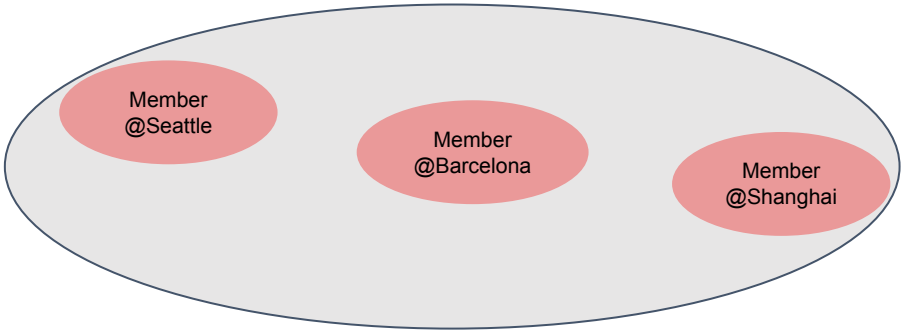
Yes!

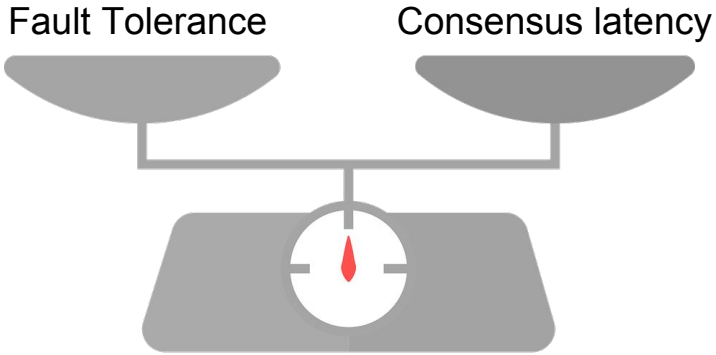
Does etcd work with cross-region deployments?



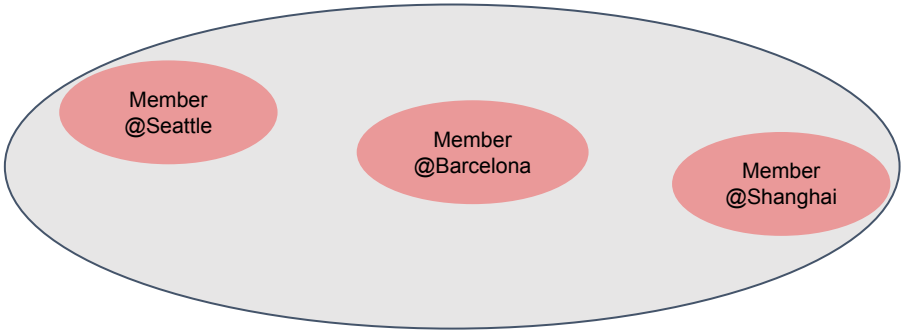
Fault Tolerance

Does etcd work with cross-region deployments?



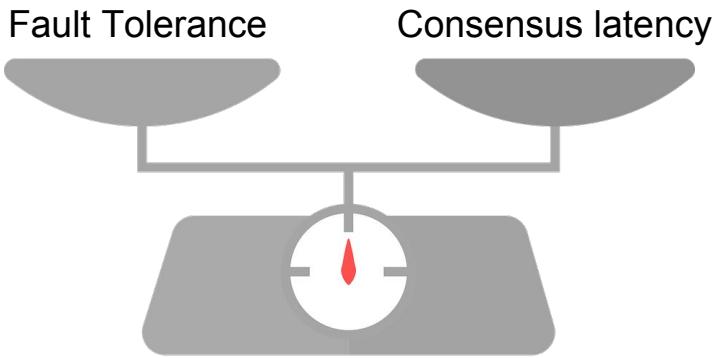


Does etcd work with cross-region deployments?



Tuning:
heartbeat interval and election timeout setting

Time | Snapshot | Disk | Networking



How to safely upgrade/downgrade etcd

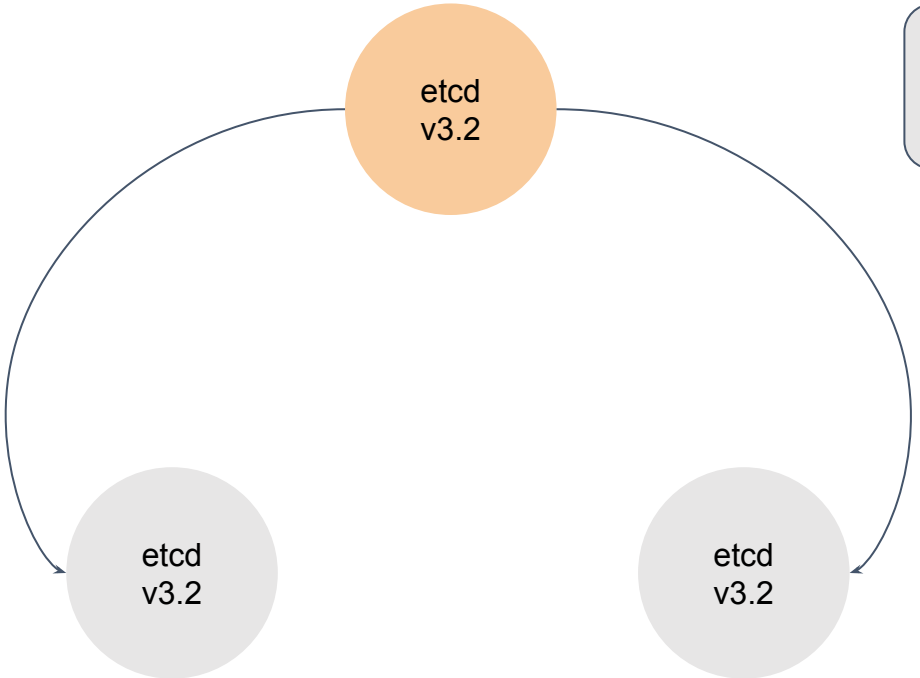
Before we begin…

```
$ etcdctl snapshot save backup.db
$ etcdctl --write-out=table snapshot status backup.db
+----------+----------+------------+------------+
|   HASH   | REVISION | TOTAL KEYS | TOTAL SIZE |
+----------+----------+------------+------------+
| fe01cf57 |       10 |          7 | 2.1 MB     |
+----------+----------+------------+------------+
```

How to safely upgrade etcd

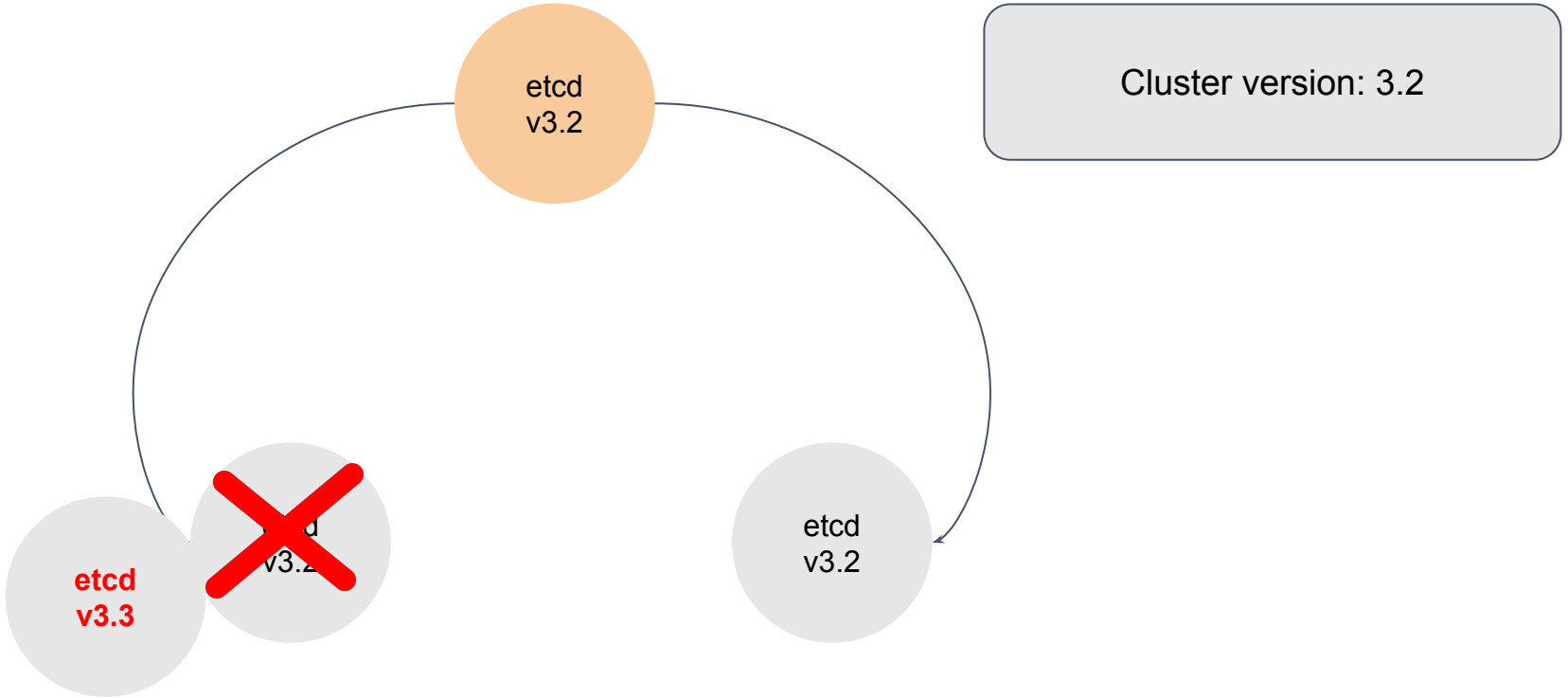
- Rolling update
- One-minor version upgrade

How to safely upgrade etcd

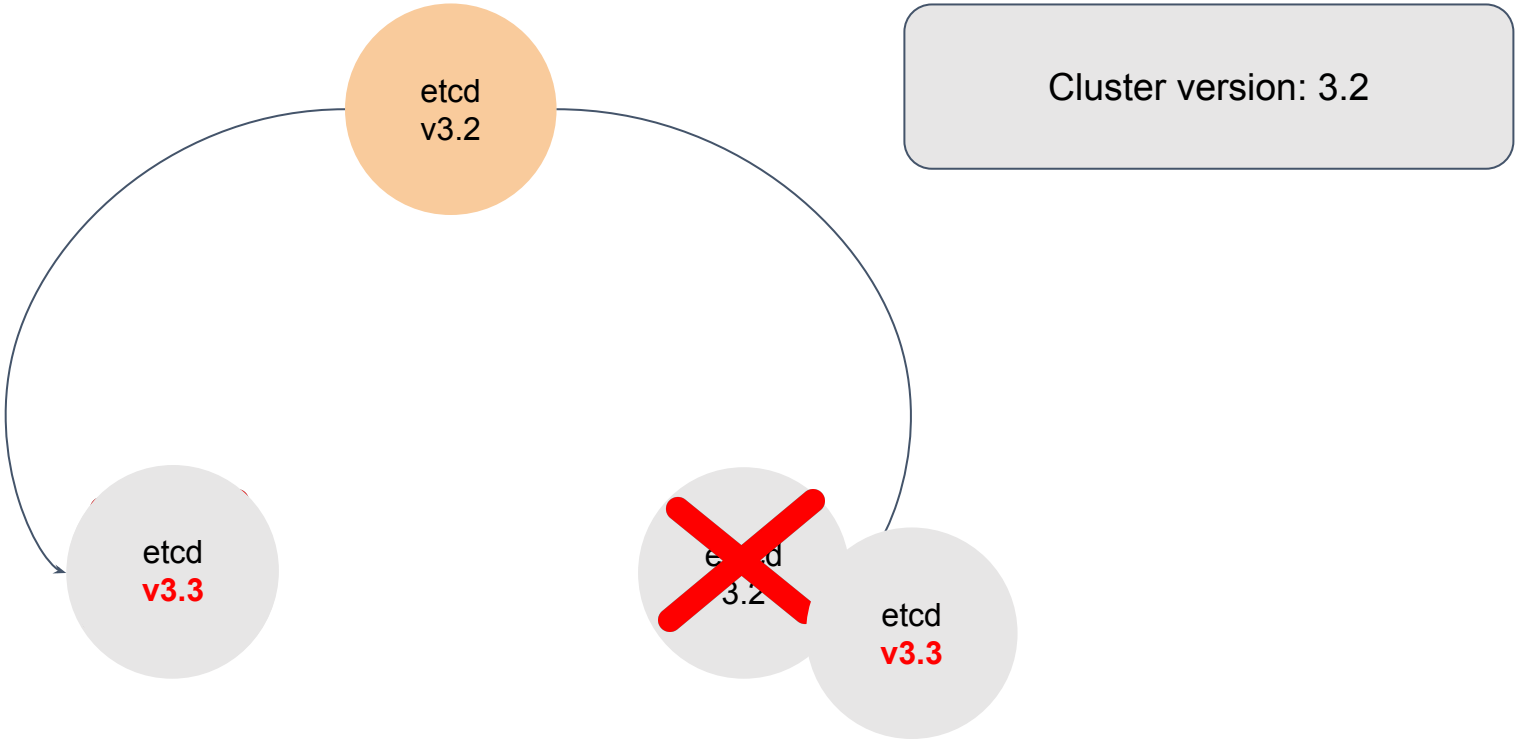


Cluster version: 3.2

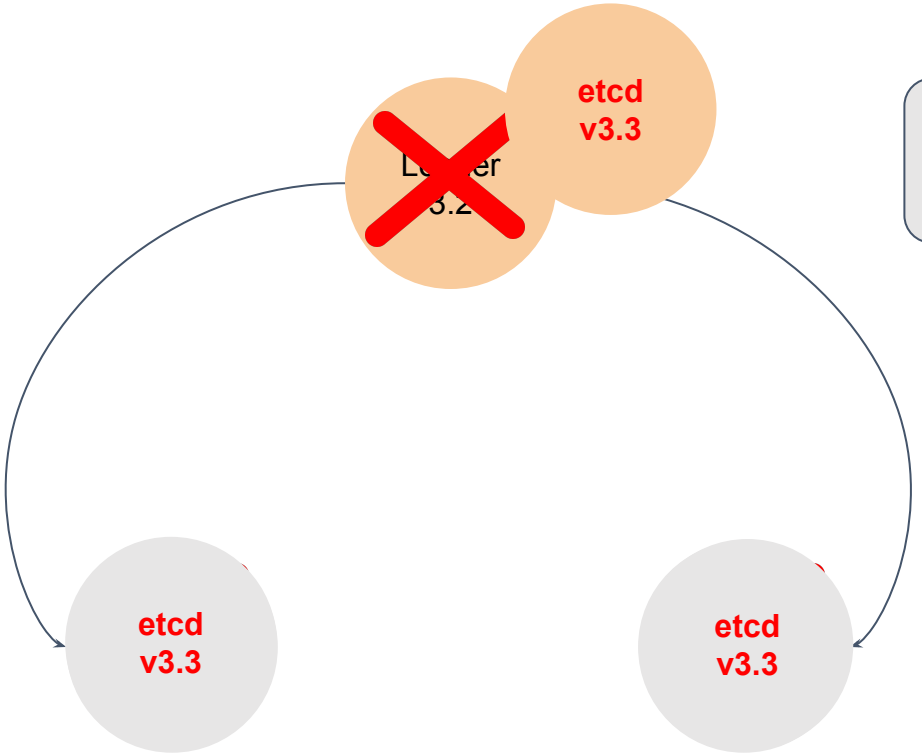
How to safely upgrade etcd



How to safely upgrade etcd



How to safely upgrade etcd



Cluster version: **3.3**

How to safely upgrade/downgrade etcd

Upgrade

https://github.com/etcd-io/etcd/tree/master/Documentation/upgrades

Downgrade

- Downgrade with downtime
- Downgrade with NO downtime: etcd v3.4 (2019), issues#9306

What is the database file

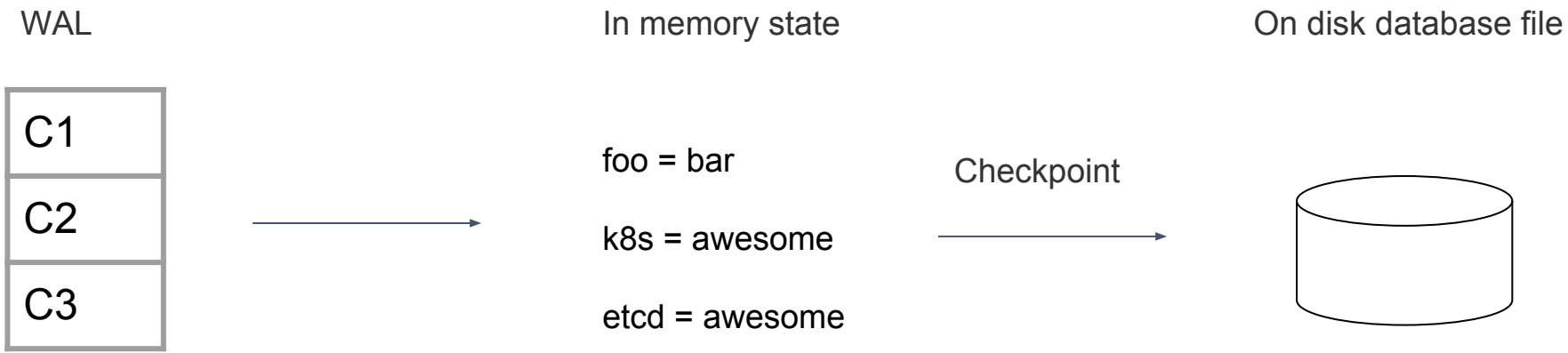
Restart == re-apply ALL entries in the WAL?

New member == get and re-apply ALL entries from existing members?

What is the database file

Restart == re-apply entries *after the checkpoint*

New member == get database file + get and re-apply entries *after the checkpoint*



Why does etcd have data size limit

New member == get database file

Mean Time To Recovery ~= Total data size / IO throughput

Why does etcd have data size limit

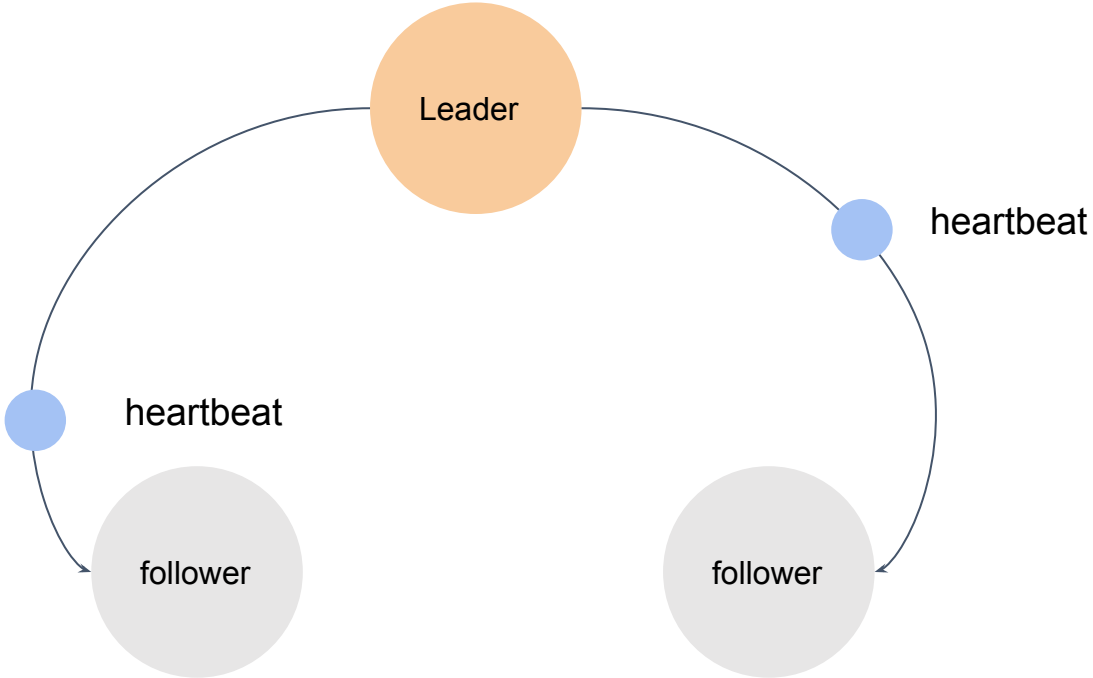
etcd data is mmap-ed

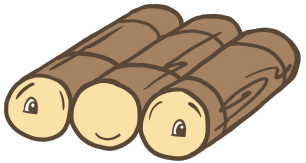
Data in memory for fast read

Why does etcd require compaction

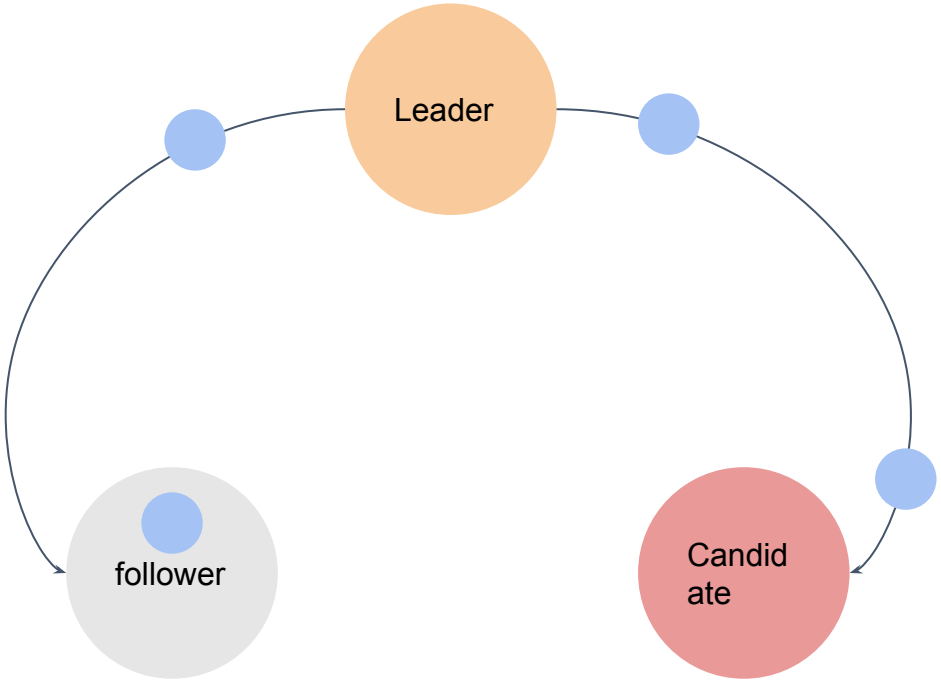
- Keep all versions of the keys
  - Configuration rollback
  - Reliable watch (similar to Kafka offset)
- Compaction removes the old versions of the keys

Why is etcd sensitive to I/O latency

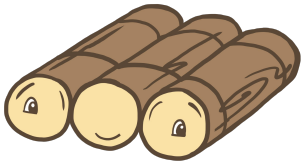




Why is etcd sensitive to I/O latency



heartbeat



How to analyze etcd performance

2 performance factors: latency and throughput

2 physical constraints: networking I/O and disk I/O

1 etcd benchmark tool: [etcd/tools/benchmark]

How to analyze etcd performance

Current benchmark results

https://github.com/etcd-io/etcd/blob/master/Documentation/op-guide/performance.md

Hardware configuration

https://github.com/etcd-io/etcd/blob/master/Documentation/op-guide/hardware.md#hardware-recommendations

How shall I help with etcd development

- Contact:
  - Email: etcd-dev@googlegroups.com
  - IRC: #etcd IRC channel on freenode.org
  - Community meeting: 11:00 PST Tuesday 01/08/2018, Monthly https://github.com/etcd-io/etcd#community-meetings
- Issues and PRs: https://github.com/etcd-io/etcd
- CONTRIBUTING! https://github.com/etcd-io/etcd/blob/master/CONTRIBUTING.md

Areas to contribute

- Test
- New documentation website: https://etcd.readthedocs.io/en/latest/
- Documentation of etcd Metrics
- Downgrade support https://github.com/etcd-io/etcd/issues/9306
- Non-voting member: https://github.com/etcd-io/etcd/issues/9161

Why etcd is slow

- Slow Disk I/O?
- Not enough Memory?
- Large value size?
- Large range queries?
- Old version of etcd?
- ?? help us to investigate!

etcd tools

- etcdctl
- etcd-dump-db
- etcd-dump-logs
- Auger

etcdctl


```
$ ETCDCTL_API=3 etcdctl
NAME:
        etcdctl - A simple command line client for etcd3.
USAGE:
        etcdctl
VERSION:
        3.3.0+git
API VERSION:
        3.3
COMMANDS:
        get                     Gets the key or a range of keys
        put                     Puts the given key into the store
        del                     Removes the specified key or range of keys [key, range_end)
        txn                     Txn processes all the requests in one transaction
        compaction              Compacts the event history in etcd
        alarm disarm            Disarms all alarms
        alarm list              Lists all alarms
        defrag                  Defragments the storage of the etcd members with given endpoints
        endpoint health         Checks the healthiness of endpoints specified in `--endpoints` flag
        endpoint status         Prints out the status of endpoints specified in `--endpoints` flag
        endpoint hashkv         Prints the KV history hash for each endpoint in --endpoints
        move-leader             Transfers leadership to another etcd cluster member.
        watch                   Watches events stream on keys or prefixes
        version                 Prints the version of etcdctl
        lease grant             Creates leases
        lease revoke            Revokes leases
        lease timetolive        Get lease information
        lease list              List all active leases
        lease keep-alive        Keeps leases alive (renew)
        member add              Adds a member into the cluster
        member remove           Removes a member from the cluster
        member update           Updates a member in the cluster
        member list             Lists all members in the cluster
        snapshot save           Stores an etcd node backend snapshot to a given file
        snapshot restore        Restores an etcd member snapshot to an etcd directory
        snapshot status         Gets backend snapshot status of a given file
        make-mirror             Makes a mirror at the destination etcd cluster
        migrate                 Migrates keys in a v2 store to a mvcc store
        lock                    Acquires a named lock
        elect                   Observes and participates in leader election
        auth enable             Enables authentication
        auth disable            Disables authentication
        user add                Adds a new user
        user delete             Deletes a user
        user get                Gets detailed information of a user
        user list               Lists all users
        user passwd             Changes password of user
        user grant-role         Grants a role to a user
        user revoke-role        Revokes a role from a user
        role add                Adds a new role
        role delete             Deletes a role
        role get                Gets detailed information of a role
        role list               Lists all roles
        role grant-permission   Grants a key to a role
        role revoke-permission  Revokes a key from a role
        check perf              Check the performance of the etcd cluster
        help                    Help about any command
```

https://github.com/etcd-io/etcd/tree/master/etcdctl

etcd-dump-db

etcd-dump-db

etcd-dump-db inspects etcd db files.

```
Usage:
  etcd-dump-db [command]

Available Commands:
  list-bucket    bucket lists all buckets.
  iterate-bucket iterate-bucket lists key-value pairs in reverse order.
  hash           hash computes the hash of db file.

Flags:
  -h, --help[=false]: help for etcd-dump-db

Use "etcd-dump-db [command] --help" for more information about a command.
```

```
$ ./etcd-dump-db -h
etcd-dump-db inspects etcd db files.

Usage:
  etcd-dump-db [command]

Available Commands:
  hash           hash computes the hash of db file.
  help           Help about any command
  iterate-bucket iterate-bucket lists key-value pairs in reverse order.
  list-bucket    bucket lists all buckets.

Flags:
  -h, --help               help for etcd-dump-db
      --timeout duration   time to wait to obtain a file lock on db file, 0 to
block indefinitely (default 10s)

Use "etcd-dump-db [command] --help" for more information about a command.
```

https://github.com/etcd-io/etcd/tree/master/tools/etcd-dump-db

etcd-dump-logs

etcd-dump-logs

etcd-dump-logs dumps the log from data directory.

```
Usage:

  etcd-dump-logs [data dir]
    * Data dir is where the snapshots and WAL logs are located. The structure of the data dir should look li
        - data_dir/member
            - data_dir/member/snap
            - data_dir/member/wal
                - data_dir/member/wal/0000000000000000-0000000000000000.wal

Flags:

  -entry-type string
        If set, filters output by entry type. Must be one or more than one of:
        ConfigChange, Normal, Request, InternalRaftRequest,
        IRRRange, IRRPut, IRRDeleteRange, IRRTxn,
        IRRCompaction, IRRLeaseGrant, IRRLeaseRevoke
  -start-index uint
        The index to start dumping
  -start-snap string
        The base name of snapshot file to start dumping
  -stream-decoder string
        The name and arguments of an executable decoding tool, the executable
        must process hex encoded lines of binary input (from etcd-dump-logs)
        and output a hex encoded line of binary for each input line
```

```
$ ./etcd-dump-logs --h
Usage of ./etcd-dump-logs:
  -entry-type string
        If set, filters output by entry type. Must be one or more than
one of:
        ConfigChange, Normal, Request, InternalRaftRequest,
        IRRRange, IRRPut, IRRDeleteRange, IRRTxn,
        IRRCompaction, IRRLeaseGrant, IRRLeaseRevoke
  -start-index uint
        The index to start dumping
  -start-snap string
        The base name of snapshot file to start dumping
  -stream-decoder string
        The name of an executable decoding tool, the executable must
process
        hex encoded lines of binary input (from etcd-dump-logs)
        and output a hex encoded line of binary for each input line
```

https://github.com/etcd-io/etcd/tree/master/tools/etcd-dump-logs

Auger

Auger

Directly access data objects stored in `etcd` by `kubernetes`.

Encodes and decodes Kubernetes objects from the binary storage encoding used to store data to `etcd`. Supports data conversion to `YAML`, `JSON` and `Protobuf`.

Automatically determines if etcd data is stored in `JSON` (`kubernetes` `1.5` and earlier) or binary (`kubernetes` `1.6` and newer) and decodes accordingly.

Why?

In earlier versions of `kubernetes`, data written to `etcd` was stored as `JSON` and could easily be inspected or manipulated using standard tools such as `etcdctl`. In `kubernetes` `1.6+`, for efficiency reasons, much of the data is now stored in a binary storage representation, and is non-trivial to decode-- it contains a enveloped payload that must be unpacked, type resolved and decoded.

This tool provides `kubernetes` developers and cluster operators with simple way to access the binary storage data via `YAML` and `JSON`.

```
$ build/auger -h
Inspect and analyze kubernetes objects in binary storage
encoding used with etcd 3+ and boltdb.

Usage:
  auger [command]

Available Commands:
  decode      Decode objects from the kubernetes binary key-value store encoding.
  encode      Encode objects to the kubernetes binary key-value store encoding.
  extract     Extracts kubernetes data from the boltdb '.db' files etcd persists
to.
  help        Help about any command

Flags:
  -h, --help   help for auger

Use "auger [command] --help" for more information about a command.
```

https://github.com/jpbetz/auger

Thanks!

Xiang Li Alibaba

Wenjia Zhang Google