



CockroachDB 中国社区大会





CockroachDB灾备介绍及应用

--杨亮

百度云数据库资深研发工程师

CockroachDB Contributor

近期业界故障案例



✓ GitLab误删数据

- 300GB生产环境数据被误删，数据库恢复后损失大约6个小时的数据

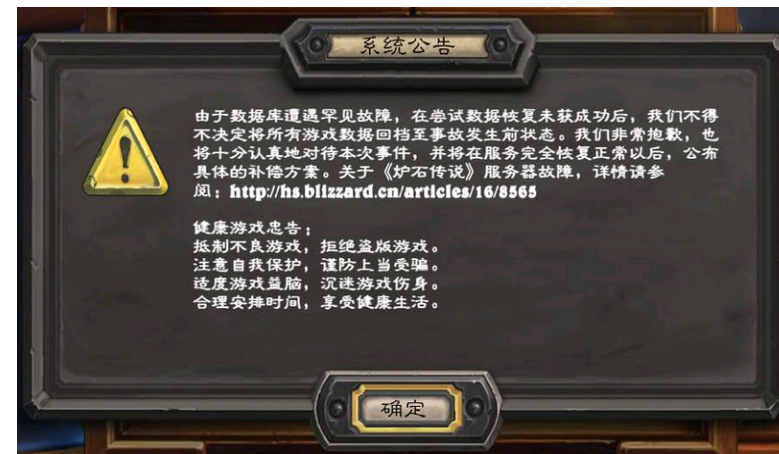


GitLab.com is currently offline due to issues with our production database.

We're working hard to resolve the problem quickly. Follow [GitLabStatus](#) for the latest updates.

✓ 炉石传说

- 供电意外中断，数据库恢复后数据损失30%



应用场景



✓ 数据库为什么需要数据备份功能？

- 灾备场景：误操作，硬件故障，软件故障，黑客攻击，病毒入侵，自然灾害
- 如果没有备份，数据损坏或者丢失时，业务损失将无法挽回。

✓ 高可用的CockroachDB为什么也需要备份功能？

- CockroachDB持续清理超过垃圾回收期的历史数据
- 单个Range损失半数以上副本时，将不可访问，例如三副本10节点的集群，宕机2节点的情形

Question



- 如何使用备份与恢复功能？
- 备份与恢复的大概原理？
- 备份数据如何存放？
- 备份数据在生成，转移和恢复过程中发生损坏和丢失问题，如何应对？
- 单点备份还是分布式备份？
- 如何应对单节点故障？
- 执行时是否需要停服？
- 如果备份和过程中想要暂停或者取消，如何处理？

用法



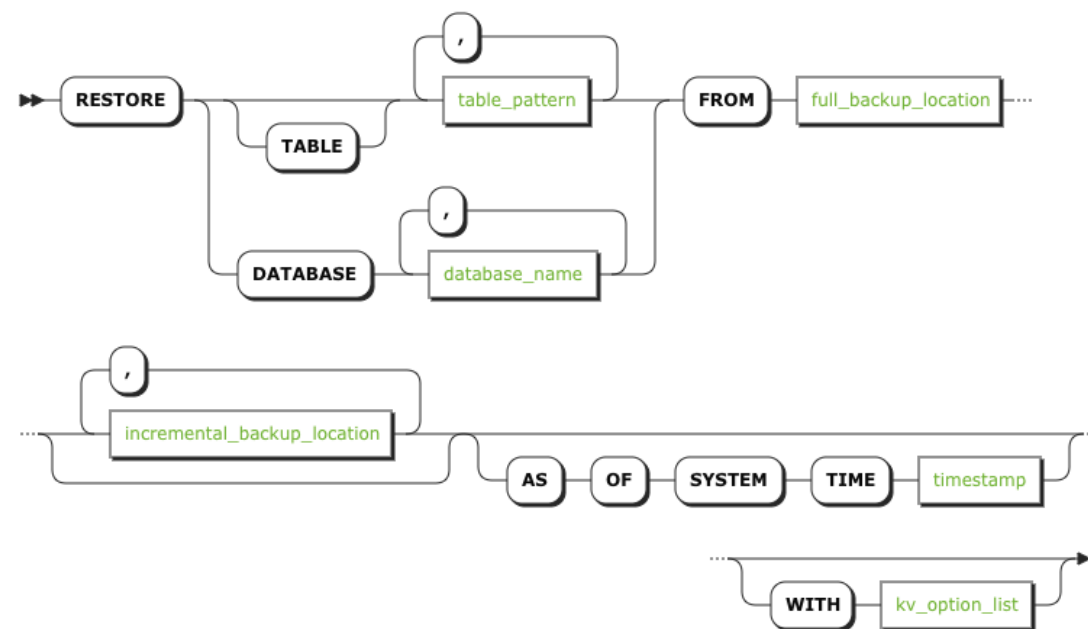
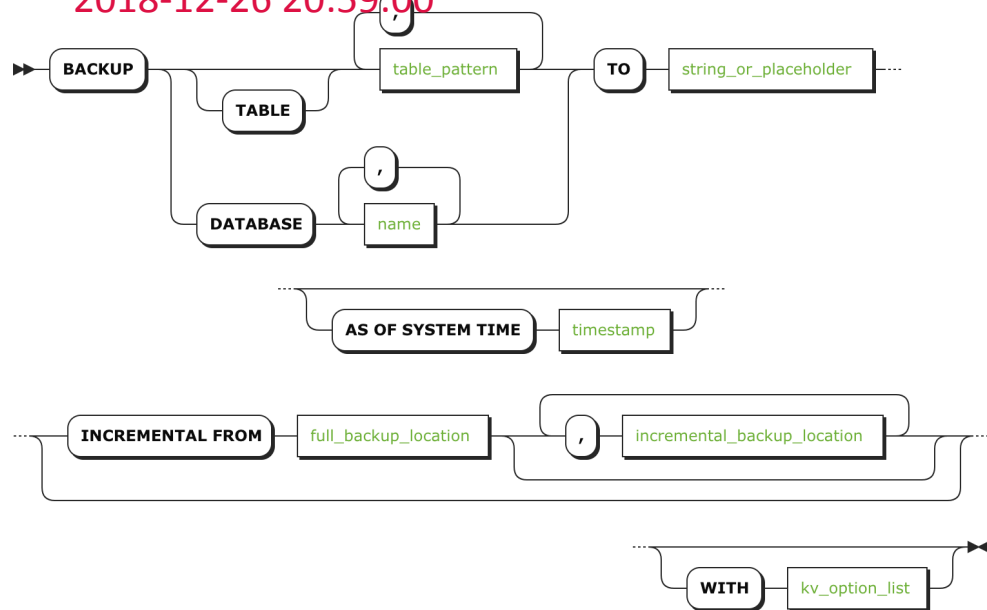
示例：

```
BACKUP DATABASE tb TO 'nodelocal:///var/inc' AS OF SYSTEM TIME '2018-12-26 23:59:00' INCREMENTAL
```

```
FROM 'nodelocal:///var/full'
```

```
RESTORE DATABASE tb FROM 'nodelocal:///var/inc' , 'nodelocal:///var/full' AS OF SYSTEM TIME
```

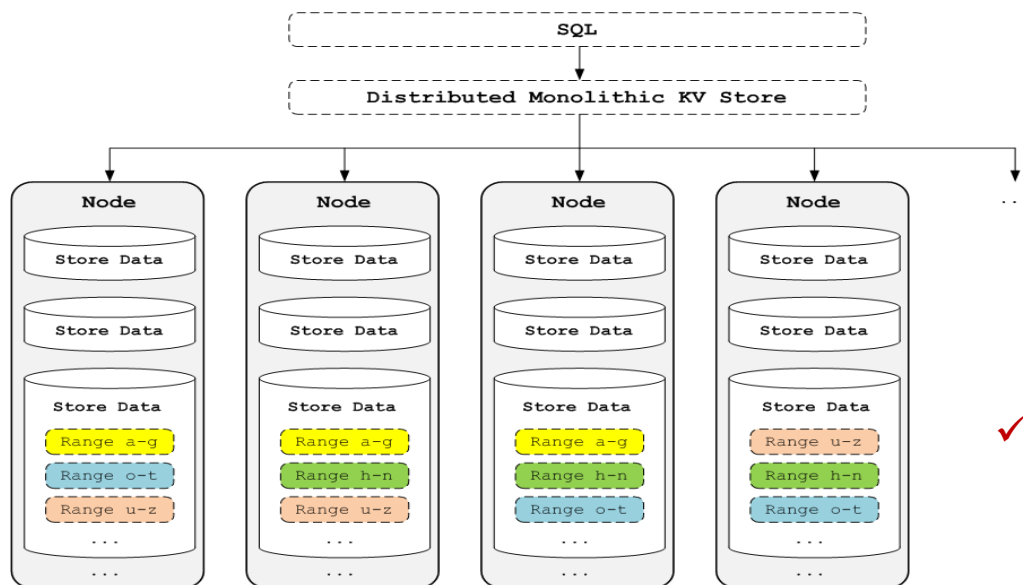
```
'2018-12-26 20:59:00'
```



数据存储



Range内部数据结构



CockroachDB架构

Key_1 : Value_1
Key_2 : Value_2
Key_3 : Value_3
...
Key_n : Value_n



✓ Key-Value编码：

- 前缀编码技术
- Key编码：Table ID, HLC Time(数据写入时间)，Index ID等
- Value编码：Checksum，行数据等

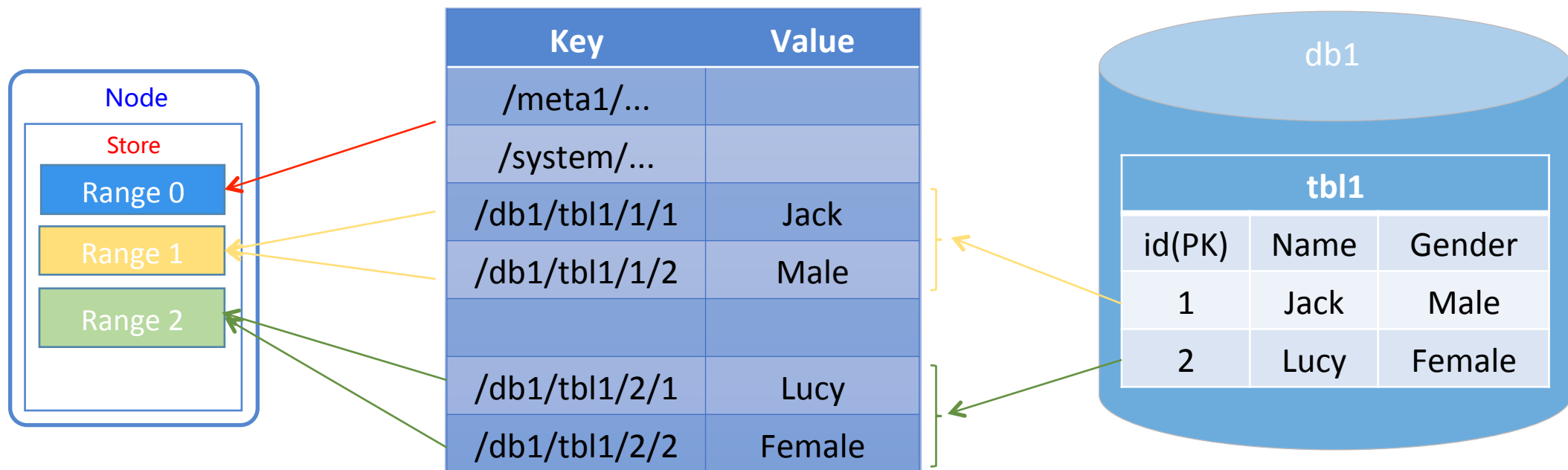
✓ CockroachDB数据管理特点：

- 以Range为单位分散存储和管理，
- 以Raft协议保障Range内部副本间数据一致性
- Range内部以Key-Value形式有序存储数据；
- 以两级路由有序管理所有Range；
- CockroachDB整个分布式KV层形似有序KV列表。

SQL-KV映射



Range , KV 列表和用户表之间的关联



有序KV列表，支持Column Family

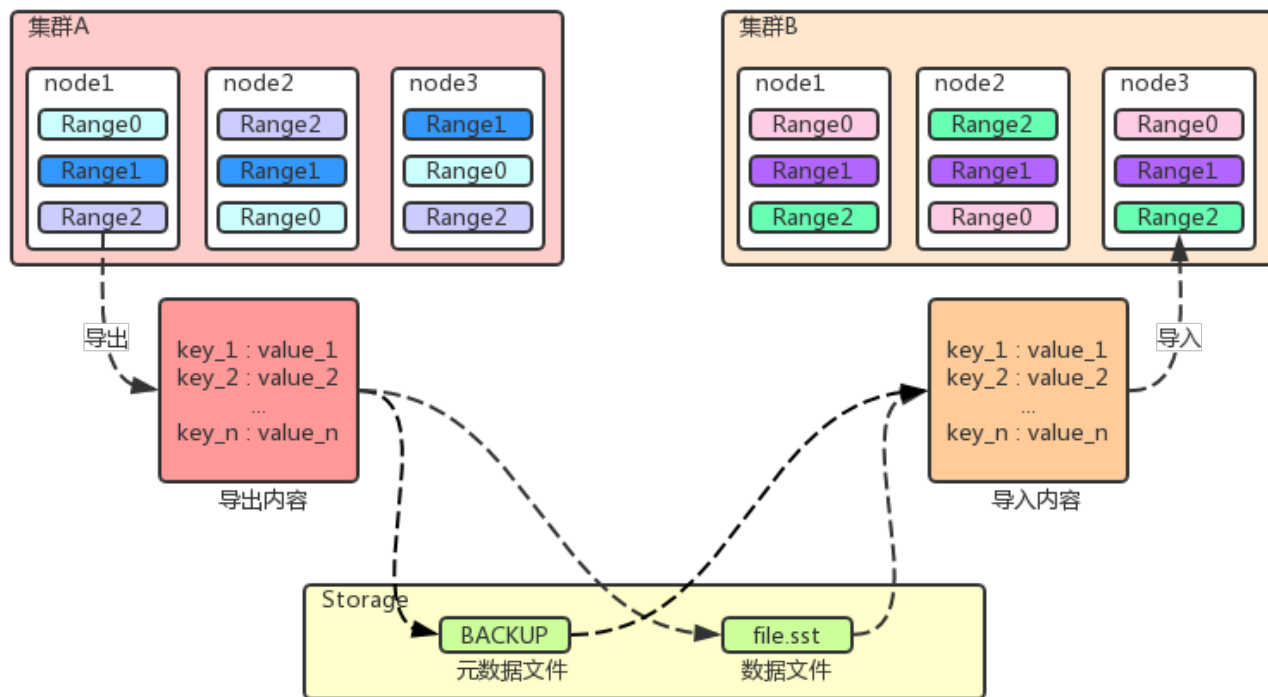
备份与恢复本质



备份：从分布式KV层导出指定Key范围KV记录的过程

恢复：将KV记录导入至KV层的过程，导入后KV层保持有序

备份与恢复过程



备份过程：BACKUP语句转化为对一组Key范围KV记录的导出操作，任务并行发送至指定Range处理，每个Range导出一个.sst文件，最后生成一个元数据文件BACKUP；

恢复过程：RESTORE语句转化为对一组Key范围KV记录的导入操作，预先分裂出新Range，并发读入数据，更新Key和Value，并写入Range

- ✓ **备份数据组成：**一组SSTable格式数据文件，一个备份数据元数据文件BACKUP（记录了备份数据的元信息，包括备份语句，库和表描述符，**数据文件的checksum**，Key范围等）

存储介质



存储介质	Scheme
Amazon S3	s3
Azure	azure
Google Cloud	gs
HTTP	http
NFS/Local	nodelocal
S3-compatible services	s3

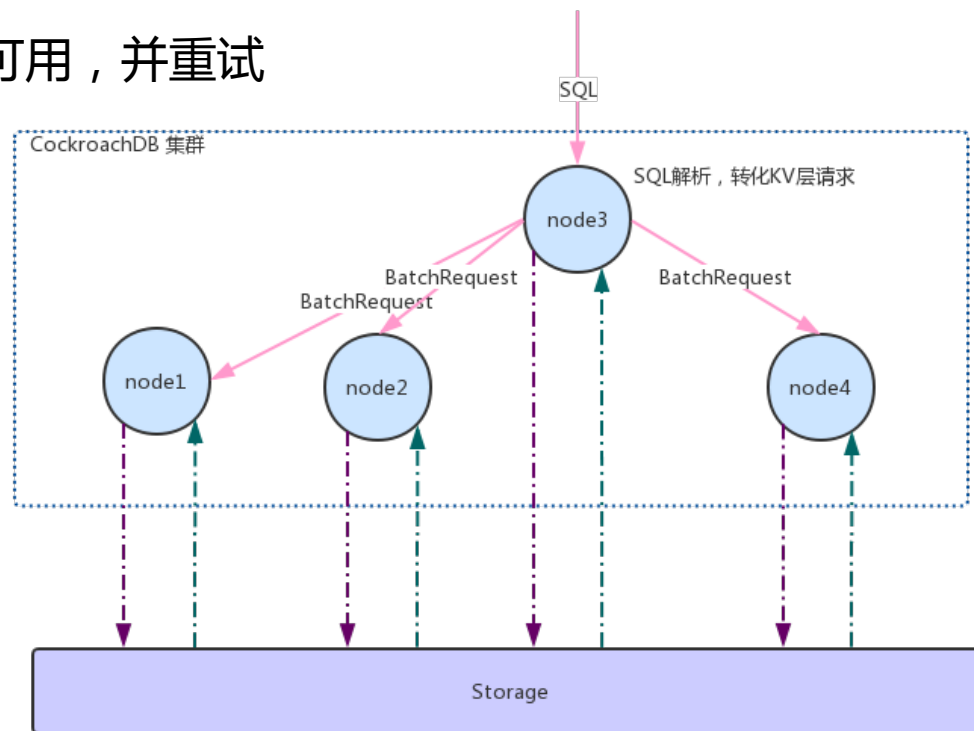
尚不支持国内厂商的云存储，用户可自建共享存储，例如NFS，HTTP。

我们将为在百度的分支版本上支持BOS，敬请期待！

可用性



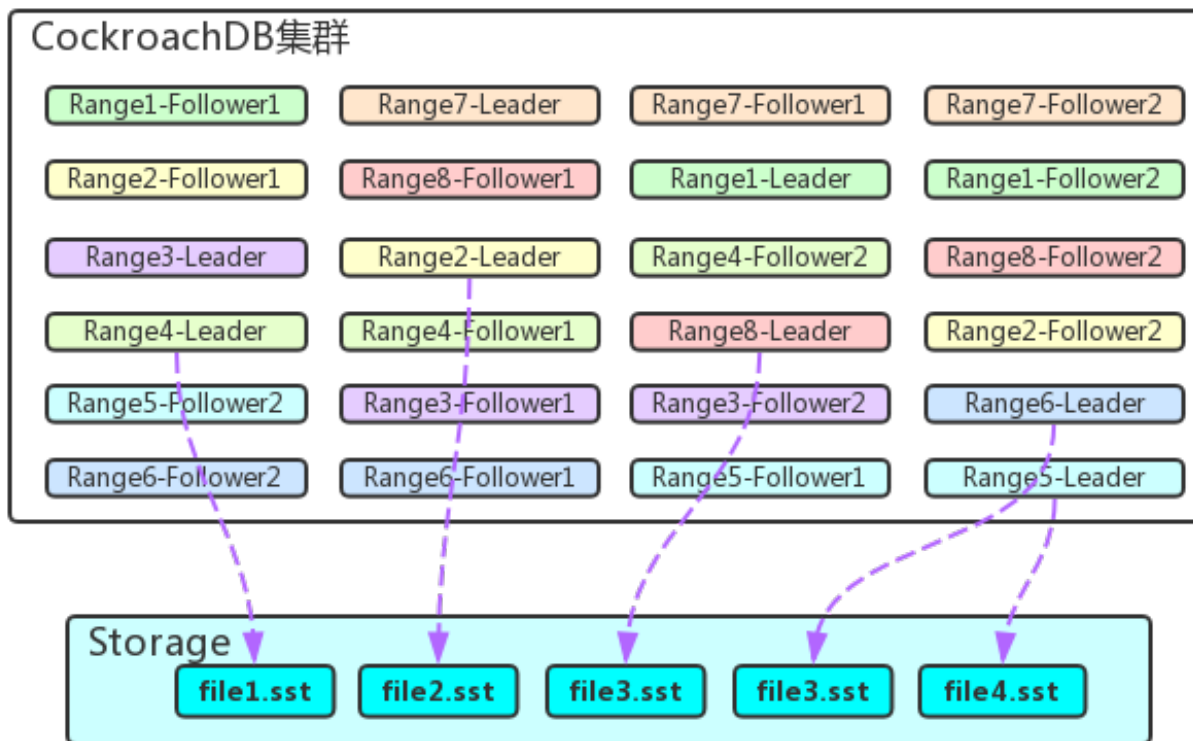
- ✓ Range Leader均衡分散，各节点负载均衡
- ✓ 单节点故障引起Range Leader副本丢失，
 - node1故障：请求会阻塞至Range恢复可用，并重试
 - node3故障：任务失败并终止
- ✓ 备份恢复过程中不停服，不影响业务使用



性能



- 流式备份，数据不落本地磁盘；
- Range Leader在集群中分散存储，备份和恢复都是分布式运行，Range级并发，性能随节点数增加而增长。



数据一致性

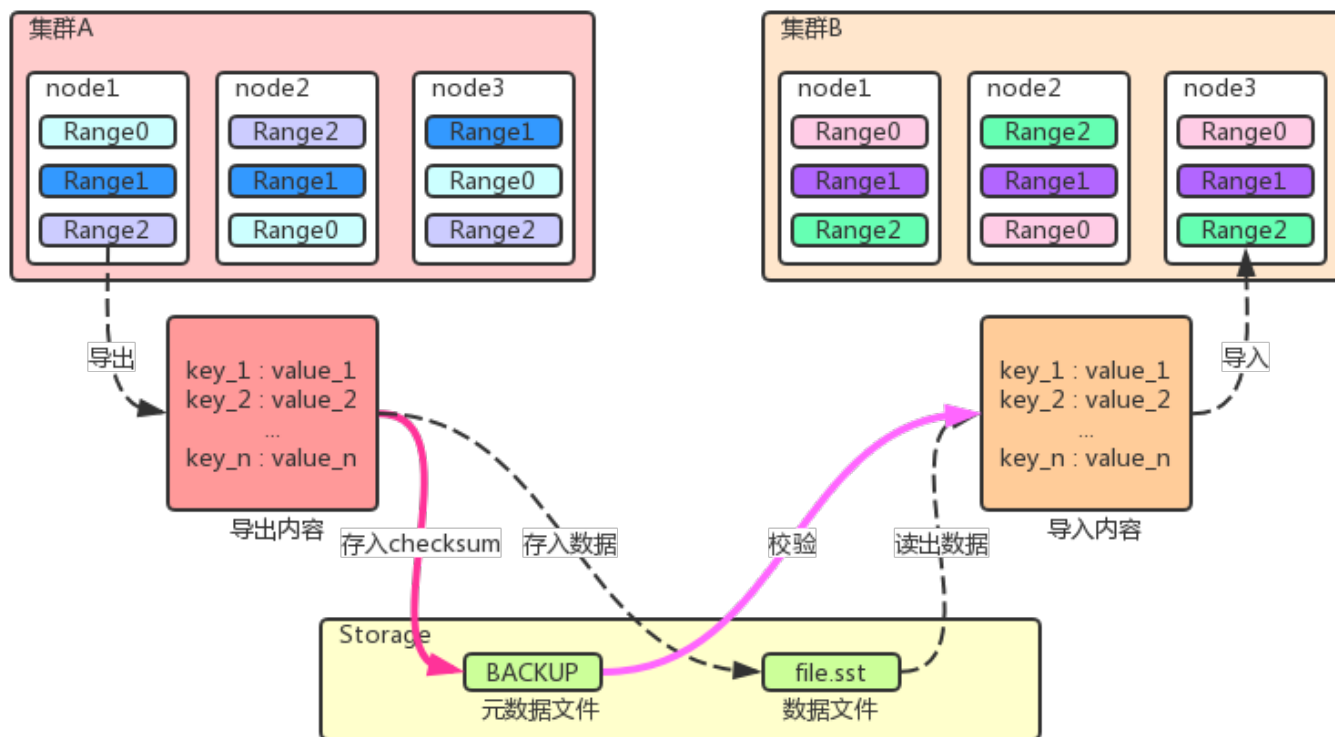


✓ 一致性保障机制：

- 备份数据的BACKUP文件保存了数据文件的checksum；
- 每一条KV记录都保存了该KV记录的checksum；

✓ 一致性校验过程：

- 恢复时，校验文件的checksum和KV记录的checksum，防止文件在生成，传输和读取过程中发生数据错误，人为篡改；



控制管理



CockroachDB以JOB形式管理BACKUP和RSTORE任务，提供JOB的查看，暂停，重启和取消功能，见右图语法。

» SHOW JOBS «

» PAUSE JOB job_id «

» RESUME JOB job_id «

» CANCEL JOB job_id «

» SHOW BACKUP location «

逻辑备份

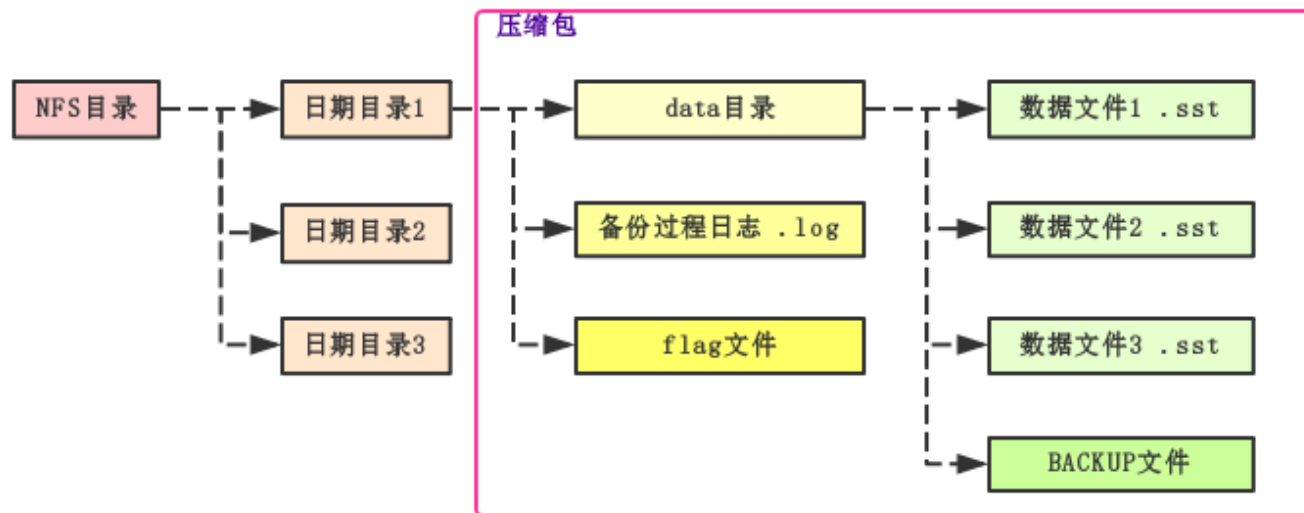


- ✓ CockroachDB提供类似于mysqldump的备份和恢复工具：
- `cockroach dump` 导出SQL语句格式的备份数据
 - `cockroach sql` 执行SQL语句恢复数据

建议



- ✓ 根据集群数据量和每日增量来设置全量与增量备份周期以及垃圾回收周期
- ✓ 平台建设：
 - 自建NFS共享存储，分目录压缩管理备份数据
 - WEB控制台，实现周期备份，恢复，流程控制等功能，
 - 自建小规模CockroachDB集群存储备份和恢复日志



了解更多备份恢复细节



[更多详细内容请查阅CockroachDB公众号文章：](#)

《CockroachDB Key-Value编码解析》

《CockroachDB备份与恢复》

《CockroachDB用户手册中文版》



Question



- 如何使用备份与恢复功能？
- 备份与恢复的大概原理？
- 备份数据如何存放？
- 备份数据在生成，转移和恢复过程中发生损坏和丢失问题，如何应对？
- 单点备份还是分布式备份？
- 如何应对单节点故障？
- 执行时是否需要停服？
- 如果备份和过程中想要暂停或者取消，如何处理？



Thanks

2019年01月12日

