BACKUP仅支持企业版用户，非企业版用户请使用cockroach dump

CockroachDB的BACKUP语句允许您创建群集的schema和数据的完整备份或增量备份，这些备份与给定时间戳一致。 备份可以带或不带revision history。

由于CockroachDB具有高容错能力，因此这些备份主要用于通过RESTORE进行灾难恢复（例如你的群集丢失了大部分节点）。 Isolated问题（例如小规模节点中断）不需要任何干预。

功能细节

备份目标

你可以全量备份你的表（包括索引）或视图；备份数据库会备份所有的表和视图。

BACKUP支持表级别备份，但不支持备份表的子集。

对象依赖关系

备份对象的同时需要备份它们的依赖对象。

| **对象** | **依赖** |
| --- | --- |
| Table with[foreign key](http://doc.cockroachchina.baidu.com/#develop/constraints/foreign-key/)constraints | 它依赖的表格，这个依赖可以在restore时删除 |
| Table with a [sequence](https://www.cockroachlabs.com/docs/stable/create-sequence.html) | New in v2.0:它使用的sequence; 这个依赖可以在restore时删除 |
| [Views](http://doc.cockroachchina.baidu.com/#develop/views/) | 视图的SELECT语句中使用的表 |
| [Interleaved tables](https://www.cockroachlabs.com/docs/stable/interleave-in-parent.html) | [interleaved hierarchy](https://www.cockroachlabs.com/docs/stable/interleave-in-parent.html#interleaved-hierarchy)所在的表 |

用户和权限

每次备份都会自动备份system.users，它存储了用户和密码。可以使用restore相关语句来恢复你的数据库用户。

已还原的表从目标数据库继承授予的权限，它们不会保留备份表中授予的权限，因为还原群集可能具有不同的用户。

表级别的权限需要在恢复完成后[授予用户](http://doc.cockroachchina.baidu.com/#develop/sql-statements/GRANT-%3Cprivileges%3E/)。

备份方式

CockroachDB 提供两种备份方式：全量备份和增量备份

全量备份

全量备份包含未复制的数据副本，它可以用来做集群恢复。这些文件大小与数据大小相同，并且需要比增量备份更多的资源。您可以从给定时间戳开始执行完整备份，并（可选）包括可用的[revision history](http://doc.cockroachchina.baidu.com/#develop/sql-statements/BACKUP/#backups-with-revision-history-new-in-v2-0).。

增量备份

增量备份比全量备份会更快并且备份文件更小，因为只包含对比指定的基本备份（需要包含一个全量备份，且可以包含许多增量备份）有变化的数据。您可以从给定时间戳或完整[revision history](http://doc.cockroachchina.baidu.com/#develop/sql-statements/BACKUP/#backups-with-revision-history-new-in-v2-0)获取增量备份。

请注意以下限制:

* 只能在基本备份的最新时间戳的垃圾回收期内创建增量备份。这是因为增量备份是通过查找在基本备份中的最新时间戳以来已创建或修改了的数据来创建增量备份--然而，时间戳数据会被垃圾回收过程删除。

你可以使用 ttlseconds[replication zone setting](http://doc.cockroachchina.baidu.com/#deploy/configure-replication-zones/)来配置垃圾回收周期

* 如果在全量备份后，又create/drop/truncate了新的table，则无法使用增量备份。你需要重新进行一次全量备份。

使用revision history备份（New in v2.0）

你可以通过revision history来创建全量或增量备份：

* 使用revision history进行全量备份时，你可以备份所有垃圾收集期内所做的每一项更改，直至并包括给定的时间戳。
* 使用revision history做增量备份时，你可以备份从上次备份至今的，且在垃圾回收期内的和对应时间戳内的所有变化。你可以使用 revision history进行增量备份，尽管你之前的全量或增量备份并没有使用 revision history。

你可以使用 ttlseconds [replication zone setting](http://doc.cockroachchina.baidu.com/#deploy/configure-replication-zones/)来配置垃圾回收周期。 使用revision history来备份，允许你在revision history内使用时间点恢复数据。

性能

BACKUP过程通过将任务分配到多个节点执行，以减少对集群性能的影响。每个节点只备份它存储的数据的指定部分的子集（该节点提供写入的数据，更多相关细节即将发布），不会存在两个节点备份相同数据的情况。

为了更好的性能，我们推荐在备份时指定最少10秒前的 [timestamp](http://doc.cockroachchina.baidu.com/#develop/data-types/TIMESTAMP/) ，例如：

> **BACKUP**...**AS** **OF** SYSTEM **TIME** '2017-06-09 16:13:55.571516+00:00';

这将降低backup过程可能与[其他语句/事务存在竞争而导致重试](http://doc.cockroachchina.baidu.com/#develop/transactions/#transaction-retries)的可能性，从而提高性能。不过，因为AS OF SYSTEM TIME返回历史数据，所以你读取的数据可能是旧的。

自动备份

我们建议每日自动备份集群。

你需要一个客户端来发送BACKUP语句到集群来自动备份。

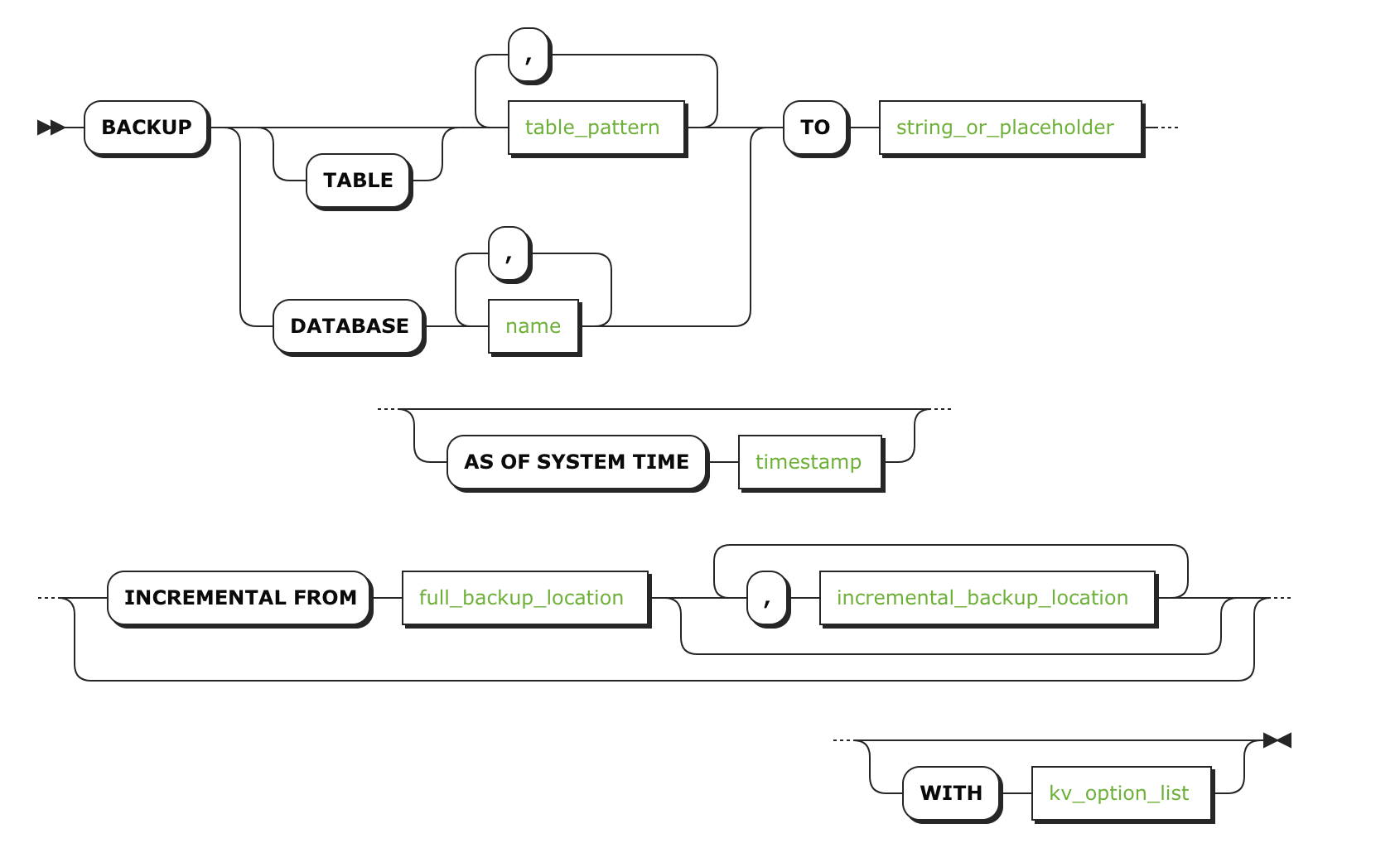
当备份完成后，客户端将收到BACKUP的回复。

查看和控制备份job

每当你启动备份时，CockroachDB将其注册为一个job，你可以通过 [SHOW JOBS](http://doc.cockroachchina.baidu.com/#develop/sql-statements/SHOW-JOBS/)查看。

当备份任务启动后，你可以通过 [PAUSE JOB](https://www.cockroachlabs.com/docs/stable/pause-job.html), [RESUME JOB](https://www.cockroachlabs.com/docs/stable/resume-job.html) 和 [CANCEL JOB](https://www.cockroachlabs.com/docs/stable/cancel-job.html)来控制对应job。

Synopsis



所需权限

只有 root 用户可以执行 BACKUP.

参数

| **Parameter** | **Description** |
| --- | --- |
| table\_pattern | 你想备份的table或者view |
| name | 你想备份的数据库名(备份数据库下所有table和view). |
| destination | 存储备份文件的URL地址  关于URL结构的更多细节，查看[备份文件URLs](https://www.cockroachlabs.com/docs/stable/#backup-file-urls). |
| AS OF SYSTEM TIME timestamp | 备份指定[timestamp](https://www.cockroachlabs.com/docs/stable/as-of-system-time.html)前存在的数据. timestamp 需要在集群最后一次垃圾回收之前 (默认是每25小时,每个表的该项都可单独[配置](http://doc.cockroachchina.baidu.com/#deploy/configure-replication-zones/#replication-zone-format)). |
| WITH revision\_history | New in v2.0: 创建具有完整revision history的备份，其记录了在垃圾回收期内对集群所做的每个更改，直至并包括给定的时间戳。 |
| INCREMENTAL FROM full\_backup\_location | 使用指定URLfull\_backup\_location存储的全量备份文件作为基础，创建一个增量备份.  **注意：**如果在全量备份后，又create/drop/truncate了新的table，则无法使用增量备份。你需要重新进行一次全量备份。 |
| incremental\_backup\_location | 创建增量备份，其中包括在提供的URL中列出的所有备份。  增量备份列表必须从最旧到最新排序. 最新的增量备份的时间戳必须在表的垃圾回收期内。  更多关于备份URL的细节，查看 [Backup File URLs](https://www.cockroachlabs.com/docs/stable/#backup-file-urls).  更多关于垃圾回收的细节，查看[Configure Replication Zones](http://doc.cockroachchina.baidu.com/#deploy/configure-replication-zones/#replication-zone-format). |

备份文件URLs

每次备份的路径必须唯一。备份路径的URL格式如下：

[scheme]://[host]/[path]?[parameters]

| **Location** | **scheme** | **host** | **parameters** |
| --- | --- | --- | --- |
| Amazon S3 | s3 | Bucket name | AWS\_ACCESS\_KEY\_ID, AWS\_SECRET\_ACCESS\_KEY |
| Azure | azure | Container name | AZURE\_ACCOUNT\_KEY, AZURE\_ACCOUNT\_NAME |
| Google Cloud | gs | Bucket name | AUTH (可选): 可以是默认的或隐式的 |
| HTTP | http | Remote host | N/A |
| NFS/Local | nodelocal | File system location | N/A |
| S3-compatible services 4 | s3 | Bucket name | AWS\_ACCESS\_KEY\_ID, AWS\_SECRET\_ACCESS\_KEY, AWS\_REGION, AWS\_ENDPOINT |

举例

根据我们在“性能”部分所描述，我们建议使用AS OF SYSTEM TIME指定从过去至少10秒开始备份。

备份一个表或视图

> **BACKUP** bank.customers \

**TO** 'gs://acme-co-backup/database-bank-2017-03-27-weekly' \

**AS** **OF** SYSTEM **TIME** '2017-03-26 23:59:00';

备份多个表

> **BACKUP** bank.customers, bank.accounts \

**TO** 'gs://acme-co-backup/database-bank-2017-03-27-weekly' \

**AS** **OF** SYSTEM **TIME** '2017-03-26 23:59:00';

备份整个数据库

> **BACKUP** **DATABASE** bank \

**TO** 'gs://acme-co-backup/database-bank-2017-03-27-weekly' \

**AS** **OF** SYSTEM **TIME** '2017-03-26 23:59:00';

通过Revision History备份（New in v2.0）

> **BACKUP** **DATABASE** bank \

**TO** 'gs://acme-co-backup/database-bank-2017-03-27-weekly' \

**AS** **OF** SYSTEM **TIME** '2017-03-26 23:59:00' **WITH** revision\_history;

创建增量备份

增量备份需要以之前创建过的全量备份为基础。

> **BACKUP** **DATABASE** bank \

**TO** 'gs://acme-co-backup/db/bank/2017-03-29-nightly' \

**AS** **OF** SYSTEM **TIME** '2017-03-28 23:59:00' \

INCREMENTAL **FROM** 'gs://acme-co-backup/database-bank-2017-03-27-weekly', 'gs://acme-co-backup/database-bank-2017-03-28-nightly';

使用Revision History创建增量备份New in v2.0）

> **BACKUP** **DATABASE** bank \

**TO** 'gs://acme-co-backup/database-bank-2017-03-29-nightly' \

**AS** **OF** SYSTEM **TIME** '2017-03-28 23:59:00' \

INCREMENTAL **FROM** 'gs://acme-co-backup/database-bank-2017-03-27-weekly', 'gs://acme-co-backup/database-bank-2017-03-28-nightly' **WITH** revision\_history;

See Also

* [RESTORE](http://doc.cockroachchina.baidu.com/#develop/sql-statements/RESTORE/)
* [Configure Replication Zones](http://doc.cockroachchina.baidu.com/#deploy/configure-replication-zones/)