列族是表中的一组列，它们在底层kv存储层被存为单个kv对。keys数量可以减少存储开销，更重要的是可以提高INSERT, UPDATE和 DELETE操作的性能。

本文解释CockroachDB是如何将列组织成列族的，以及提供了你可能想要手动覆盖默认行为的case。

默认行为

创建表时，所有列都存储为单个列族。

大多数情况，默认情况可以保证有效的kv存储和性能。然而，当频繁更新列和低频更新列组合在一起时，每次更新重写时都会更新到低频更新的列，特别是低频更新的列很大时，把它们分为不同的族是更高效的。

手动覆盖

在表创建时指定列族

在建表时，使用FAMILY关键字手动指定列族。

例如，假设我们要创建一个表来存储不可变的数据blob数据（data BYTES）和最后访问的时间戳（last\_accessed TIMESTAMP）。因为我们知道blob数据是不能更新的，我们使用FAMILY关键字来将它拆分到单独一个列族。

> **CREATE** **TABLE** test (

id INT **PRIMARY** **KEY**,

last\_accessed **TIMESTAMP**,

**data** BYTES,

FAMILY f1 (id, last\_accessed),

FAMILY f2 (**data**)

);

> **SHOW** **CREATE** **TABLE** users;

+-------+---------------------------------------------+

| Table | CreateTable |

+-------+---------------------------------------------+

| test | CREATE TABLE test (␤ |

| | id INT NOT NULL,␤ |

| | last\_accessed TIMESTAMP NULL,␤ |

| | data BYTES NULL,␤ |

| | CONSTRAINT "primary" PRIMARY KEY (id),␤ |

| | FAMILY f1 (id, last\_accessed),␤ |

| | FAMILY f2 (data)␤ |

| | ) |

+-------+---------------------------------------------+

(1 row)

属于主索引的列始终分配给第一个列族。 如果手动将主索引列分配给一个列族，则它必须是CREATE TABLE语句中列出的第一个列族。

添加列时指定列族

使用ALTER TABLE ... ADD COLUMN语句将列添加到表时，可以将列分配给新的或现有的列族。

* 使用CREATE FAMILY将一个新列分配给一个新列族。 例如，以下内容会将data2 BYTES列添加到上面的test表中，并将其分配给新的列族：

> **ALTER** **TABLE** test **ADD** **COLUMN** data2 BYTES **CREATE** FAMILY f3;

* 使用FAMILY将一个新列添加到已存在的列族。下面例子中，将会添加一列 name STRING到上面的test表，并将其分配给f1

> **ALTER** **TABLE** test **ADD** **COLUMN** name **STRING** FAMILY f1;

* 使用CREATE IF NOT EXISTS FAMILY为现有列族分配新列，如果族不存在，则分配到新的族。下面例子中，将新列分配给现有的f1，如果该列族不存在，则创建一个新列族，并将新列分配给它；

> **ALTER** **TABLE** test **ADD** **COLUMN** name **STRING** **CREATE** **IF** **NOT** **EXISTS** FAMILY f1;

与老版本的兼容性

使用beta-20160714版本会使你的数据与早于beta-20160629版本的版本不兼容。

See Also

* [CREATE TABLE](http://doc.cockroachchina.baidu.com/#develop/sql-statements/CREATE-TABLE/)
* [ADD COLUMN](http://doc.cockroachchina.baidu.com/#develop/sql-statements/ADD-COLUMN/)
* [Other SQL Statements](http://doc.cockroachchina.baidu.com/#develop/sql-statements/overview/)