## Retention policy management

<https://docs.influxdata.com/influxdb/v1.7/query_language/database_management/#retention-policy-management>

下面将会说到 how to create，alter，delete retention policies。

创建数据库的时候，Influxdb会自动的创建一个autogen rp，autogen是没有限制的数据保留策略。可以修改auto-creation 在configuration file，地址：

<https://docs.influxdata.com/influxdb/v1.7/administration/config/#metastore-settings>

创建rp

CREATE RETENTION POLICY <retention\_policy\_name> ON <database\_name> DURATION <duration> REPLICATION <n> [SHARD DURATION <duration>] [DEFAULT]



说明：

DURATION：决定保存数据的时长，最少一个小时，最大INF（infinite）就是没限制。

REPLICATION：这个是在集群的情况下用的（企业版）决定point在集群中的复本。默认是数据节点数。如果数据节点大于3，默认还是3. 为了确保数据可以马上查询到，一般设置小于集群数据节点。当数据节点很多大于3个的时候，指定replication是很有必要的。

Replication factors do not serve a purpose with single node instances.

SHARD DURATION：可选参数，指定一个shard group覆盖的time range。这里不支持INF（infinite），默认的shard group duration 是由 duration参数来决定的。对应关系



最小的hard group duration是 1h，如果尝试设置大于0s小于1h，influxdb自动的设置为1h。如果尝试设置为0s，influxdb根据上面的表格设置对应的默认值。

DEFAULT：这个参数有，就设置为数据库默认的rp。没有的话，就不是默认rp。

例子：

创建一个 RP

CREATE RETENTION POLICY "one\_day\_only" ON "NOAA\_water\_database" DURATION 1d REPLICATION 1

创建一个default RP

CREATE RETENTION POLICY "one\_day\_only" ON "NOAA\_water\_database" DURATION 23h60m REPLICATION 1 DEFAULT

成功的创建RP返回一个空的response。如果你尝试identical RP，就是和存在的RP一模一样，influxdb 不会返回error。如果你尝试创建一个RP，它的名字和已经存在的RP一样，属性不一样，influxdb 会返回error。

也可以在创建db的时候同时创建RP，语法如下

CREATE DATABASE <database\_name> [WITH [DURATION <duration>] [REPLICATION <n>] [SHARD DURATION <duration>] [NAME <retention-policy-name>]]

例如：

> CREATE DATABASE "NOAA\_water\_database" WITH DURATION 3d REPLICATION 1 SHARD DURATION 1h NAME "liquid"

创建一个liquid默认RP

修改RP

ALTER RETENTION POLICY <retention\_policy\_name> ON <database\_name> DURATION <duration> REPLICATION <n> SHARD DURATION <duration> DEFAULT

必须声明一个DURATION，REPLICATION， SHARD DURATION， DEFAULT中的一个。

例如：

先create 一个RP

CREATE RETENTION POLICY "what\_is\_time" ON "NOAA\_water\_database" DURATION 2d REPLICATION 1

然后修改RP

ALTER RETENTION POLICY "what\_is\_time" ON "NOAA\_water\_database" DURATION 3w SHARD DURATION 2h DEFAULT

删除RP

DROP RETENTION POLICY <retention\_policy\_name> ON <database\_name>

注意：删除一个RP，会永久的删除RP下面的measurement和data。所以谨慎操作。

例如：DROP RETENTION POLICY "what\_is\_time" ON "NOAA\_water\_database"

删除语句返回empty result，如果删除不存在的RP，也不会返回error。

RP的管理是通过query来实现的，在queries部分，

地址：<https://docs.influxdata.com/influxdb/v1.7/query_language/spec#queries>

alter\_retention\_policy\_stmt

drop\_retention\_policy\_stmt

create\_retention\_policy\_stmt

show\_retention\_policies\_stmt

创建默认的retention policy，会顶掉以前默认的autogen

> CREATE RETENTION POLICY "two\_hours" ON "test\_db" DURATION 2h REPLICATION 1 DEFAULT

> show retention policies on test\_db

name duration shardGroupDuration replicaN default

---- -------- ------------------ -------- -------

autogen 0s 168h0m0s 1 false

two\_hours 2h0m0s 1h0m0s 1 true

That query creates an RP called two\_hours that exists in the database test\_db. two\_hours keeps data for a DURATION of two hours (2h) and it’s the DEFAULT RP for the database test\_db

The replication factor (REPLICATION 1) is a required parameter but must always be set to 1 for single node instances.

Note: When we created the test\_db database in step 1, InfluxDB automatically generated an RP named autogen and set it as the DEFAULT RP for the database. The autogen RP has an infinite retention period.

With the query above, the RP two\_hours replaces autogen as the DEFAULT RP for the test\_db database.

## 使用PR

**显示RP**

SHOW RETENTION POLICIES [ON <database\_name>]

例如：

显示在test\_db库上的RP

> show retention policies on test\_db

name duration shardGroupDuration replicaN default

---- -------- ------------------ -------- -------

autogen 0s 168h0m0s 1 true

如果没有on clause 显示所有库的RP

**insert数据的时候使用 non-default retention policy**

普通的insert语句

insert lean,tag1=222,tag2=dfdf field1="value1",field2="value2"

使用 retention policy 时，语法有所变化。需要用 insert into

INSERT INTO [<database>.]<retention\_policy> <line\_protocol>

>INSERT INTO one\_day mortality bool=true

INSERT INTO oneday treasures,captain\_id=pirate\_king value=2

**查询的时候使用non-default PR**

> SELECT \* FROM "mydb"."one\_day"."mortality"

其中one\_day 是数据保留策略，mortality是measurement

查询non-default retenttion policy 需要用fully qualify measurement 如下语法：

"<database>"."<retention\_policy>"."<measurement>"