在CockroachDB Admin界面中的**Replication**仪表盘能够帮助用户监控集群中数据冗余相关的指标。用户可以登陆Admin界面，点击左手边导航栏的**Metrics**，选择**Dashboard** > **Replication**，来查看该仪表盘。

CockroachDB术语回顾

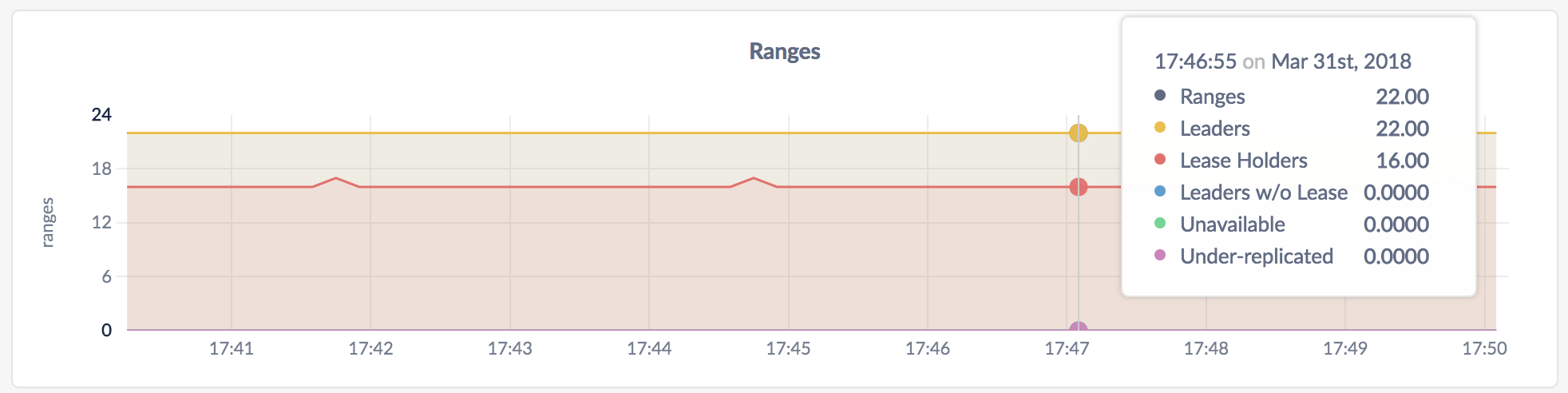
* **Range**：CockroachDB存储所有用户数据和绝大部分系统数据，将这些数据映射到一个巨大的键值对MAP当中。这个Key值空间被切分成连续的块，称为“Range”。每个Key值能够对应到一个Range。
* **Range Replica**：CockroachDB对每个Range做数据冗余（默认3份副本），并将每个副本尽可能地存储在不同的节点上。
* **Range Lease**：对于每个Range，只有一个副本持有“Range租约”，称为租约持有者，能够接受、协调该Range的所有读写请求。
* **Under-replicated Ranges**：在集群第一次初始化的时候，默认的少量起始Ranges只有一个副本。随着集群可用节点数量的增加，这些Range会复制到可用节点直到副本数量满足复制因子（默认为3）的要求。如果一个Range未达到足够的副本数量，则该Range称为“under-replicated”。
* **Unavailable Ranges**：如果集群节点上一个Range的绝大多数副本不可用，则整个Range不可用，无法处理相关的查询请求。

更多细节可以查看[Scalable SQL Made Easy: How CockroachDB Automates Operations](https://www.cockroachlabs.com/blog/automated-rebalance-and-repair/)

Replication dashboard

**Replication**仪表盘显示了以下不同指标的时间序列图：

Ranges



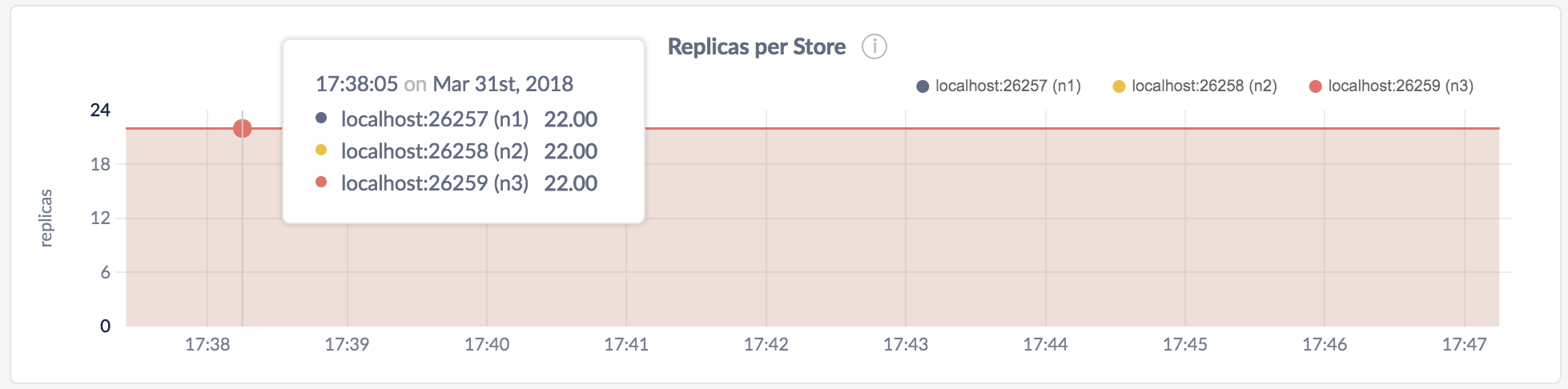
**Ranges**图向用户展示了Range状态相关的很多细节内容。

* 在节点视图中，**Ranges**图显示的是指定节点持有的Range的详细信息。
* 在集群视图中，**Ranges**图显示的是集群中每个节点持有的Range的详细信息。

将鼠标悬停在图表上时，会显示以下指标的值：

| **指标** | **简介** |
| --- | --- |
| Ranges | 该节点持有的Range数量 |
| Leaders | 拥有Leader的Range数量。如果一个节点上拥有Leader的Range数量与节点持有的Range数量在很长时间内都不匹配，则需要进行故障定位追踪。 |
| Lease Holders | 持有租约的Range数量。 |
| Leaders w/o Leases | 没有租约的Raft leaders数量。如果该指标的数值在很长时间内非零，则需要进行故障定位追踪。 |
| Unavailable | 不可用的Ranges数量，如果该指标的数值在很长时间内非零，则需要进行故障定位追踪。 |
| Under-replicated | 正在复制的Range数量。 |

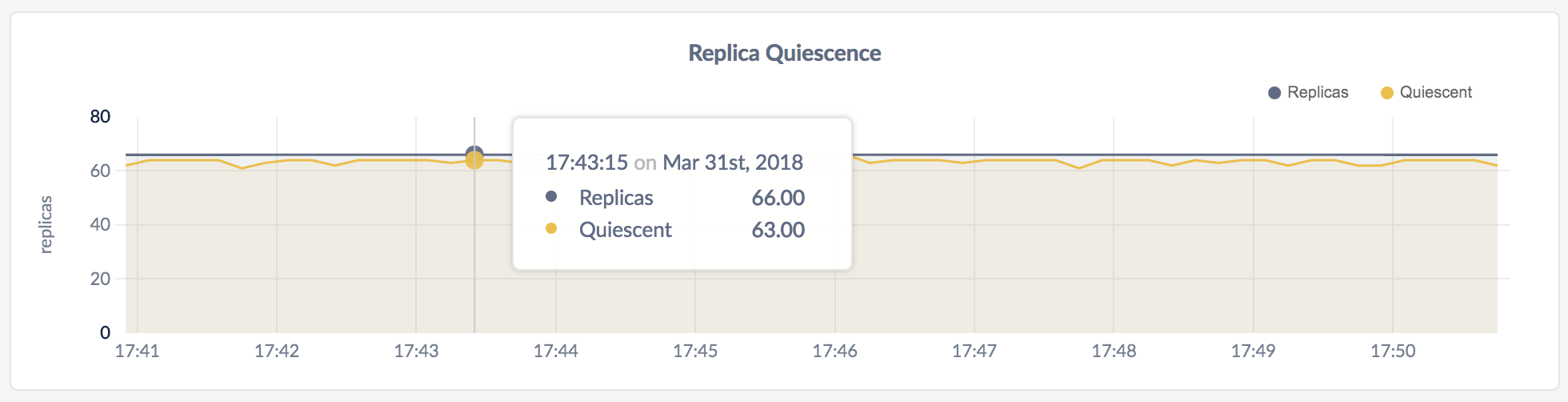
Replicas Per Store



* 在节点视图中，该时间序列图显示的是指定store上Range副本数量。
* 在集群视图中，该时间序列图显示的是每个store上Range副本数量。

用户可以[配置复制区域](http://doc.cockroachchina.baidu.com/" \l "deploy/configure-replication-zones)来设置副本的数量和位置，并通过Admin界面验证配置变更情况（详见[Fault tolerance and recovery](http://doc.cockroachchina.baidu.com/" \l "deploy/admin-ui/replication-dashboard/quick-start/explore-benefits/fault-tolerance-and-recovery)）。

Replica Quiescence

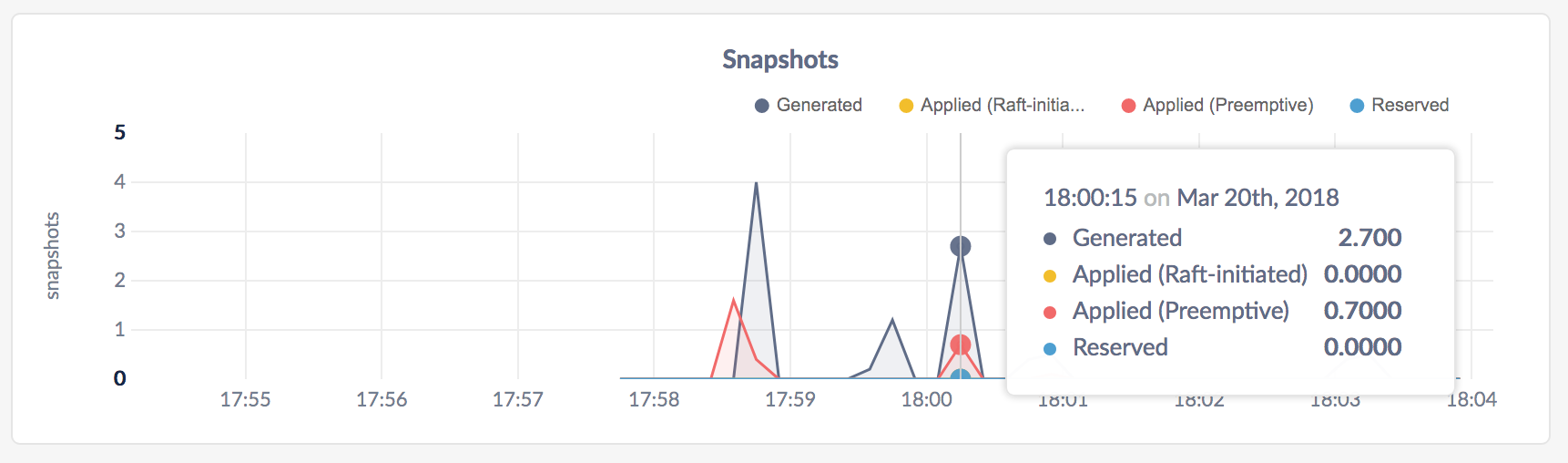


* 在节点视图中，该时间序列图显示的是指定节点的副本数量。
* 在集群视图中，该时间序列图显示的是集群中所有节点的副本数量。

将鼠标悬停在图表上时，会显示以下指标的值：

| **指标** | **简介** |
| --- | --- |
| Replicas | 副本的数量 |
| Quiescent | 一段时间未访问的副本数量 |

Snapshots



通常情况下Raft组里的节点会通过相互之间传递Raft日志消息的方式来保持同步。然而当一个节点需要同步的日志落后太多（例如节点下线，或是该节点为新节点），相比起发送所有引起Range变更的消息，集群会直接发送Range的快照，落后节点在应用快照后重新开始同步。大多数情况下这是一个主动推的过程，集群会在认为某个节点需要跟上同步进度的时候推送快照。而在少数情况下，节点会根据Raft协议请求快照同步。

| **指标** | **简介** |
| --- | --- |
| Generated | 每秒钟创建的快照的数量。 |
| Applied (Raft-initiated) | 每秒钟应用的快照的数量（由Raft发起的快照同步）。 The number of snapshots applied to nodes per second that were initiated within Raft. |
| Applied (Preemptive) | 每秒钟应用的快照的数量（集群提前预料到并主动推送快照，例如一个节点即将加入Raft组的时候）。 The number of snapshots applied to nodes per second that were anticipated ahead of time (e.g., because a node was about to be added to a Raft group). |
| Reserved | Throttling模式下快照接受窗口的大小。 The number of slots reserved per second for incoming snapshots that will be sent to a node. |

Other graphs

**Replication**仪表盘还提供了对于开发者有帮助的其他一些时间序列图：

* Leaseholders per Store
* Average Queries per Store
* Logical Bytes per Store
* Range Operations

在监控CockroachDB性能表现方面，强烈推荐使用**Ranges**、**Replicas per Store**和**Replica Quiescence**图。