



链滴

Prometheus 查询语法 PromQL

作者: [Gakkiyomi2019](#)

原文链接: <https://ld246.com/article/1602746680154>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)



Prometheus

PromQL (Prometheus Query Language) 是prometheus提供的一个函数式的表达式语言。可以实
查找数据也可以聚合时间序列数据进行图表展示，或者作为数据源以HTTP的方式提供给外部系统调
。

表达式语言数据类型

- **instant vector** 瞬时向量 - 它是指在同一时刻，抓取的所有度量指标数据。这些度量指标数据的ke
都是相同的，也即相同的时间戳。
- **range vector** 范围向量 - 它是指在任何时间范围内，抓取的所有度量指标数据。
- **scalar** 标量 - 一个简单的浮点值
- **string** 字符串 - 一个当前没有被使用的简单字符串

瞬时向量选择器

瞬时向量选择器可以只指定度量名称来获取该度量名的所有的即时向量。

已kong网关抓取各个服务的http状态码为例

`kong_http_status`

Load time: 112ms
Resolution: 14s
Total time series: 29

Element	Value
<code>kong_http_status{code="200",instance="kong.kong:8001",job="prometheus",service="auth-public"}</code>	2837
<code>kong_http_status{code="200",instance="kong.kong:8001",job="prometheus",service="sky-auth"}</code>	15
<code>kong_http_status{code="200",instance="kong.kong:8001",job="prometheus",service="sky-cmdb"}</code>	2584
<code>kong_http_status{code="200",instance="kong.kong:8001",job="prometheus",service="sky-dashboard"}</code>	1026
<code>kong_http_status{code="200",instance="kong.kong:8001",job="prometheus",service="sky-docs"}</code>	126

也可以进行标签筛选，数据埋点的时候我们会对采集的数据打上不同的tag标签

`kong_http_status{code="200",service="sky-cmdb"}`

```
kong_http_status{code="200",service="sky-cmdb"}
```

Load time: 114ms
Resolution: 14s
Total time series: 1

Execute - insert metric at cursor -

Graph Console

Element	Value
kong_http_status{code="200",instance="kong.kong:8001",job="prometheus",service="sky-cmdb"}	2616

可以采用不匹配的标签值也是可以的，或者用正则表达式不匹配标签。标签匹配操作如下所示：

- `=`: 精确地匹配标签给定的值
- `!=`: 不等于给定的标签值
- `=~`: 正则表达匹配给定的标签值
- `!~`: 给定的标签值不符合正则表达式

范围向量选择器

在瞬时向量的基础上加上时间范围，具体语法是在向量选择器末尾的方括号中填上持续时间，用来指每个结果范围向量元素提取多长时间值。

持续时间指定为数字，紧接着是以下单位之一：

- `s` - seconds
- `m` - minutes
- `h` - hours
- `d` - days
- `w` - weeks
- `y` - years

`kong_http_status{code="200",service="sky-cmdb"}[5m]`

```
kong_http_status{code="200",service="sky-cmdb"}[5m]
```

Load time: 114ms
Resolution: 14s
Total time series: 1

Execute - insert metric at cursor -

Graph Console

Element	Value
kong_http_status{code="200",instance="kong.kong:8001",job="prometheus",service="sky-cmdb"}	2639 @ 1595560421.126
	2639 @ 1595560481.126
	2639 @ 1595560541.126
	2639 @ 1595560601.126
	2639 @ 1595560661.126

偏移修饰符

这个`offset`偏移修饰符允许在查询中改变单个瞬时向量和范围向量中的时间偏移

这将返回一周前的瞬时向量:

```
kong_http_status{code="200",service="sky-cmdb"} offset 1w
```

内置函数

prometheus 内置了许多函数提供我们调用

abs()

`abs(v instant-vector)`返回输入向量, 所有样本值都转换为其绝对值。

ceil()

`ceil(v instant-vector)` 将`v`中所有元素的样本值舍入到最接近的整数。

delta()

`delta(v range-vector)`计算范围向量`v`中每个时间系列元素的第一个和最后一个值之间的差值, 返回有给定增量和等效标签的即时向量。 `delta`被外推以覆盖范围向量选择器中指定的全时间范围, 因此使样本值都是整数, 也可以获得非整数结果。

rate()

`rate(v range-vector)`计算范围向量中时间序列的每秒平均增长率。单调性中断(例如由于目标重启导致的计数器重置)会自动调整。此外, 计算推断到时间范围的末端, 允许错过刮擦或刮擦循环与范的时间段的不完美对齐。

sum()

对向量多个标签数据进行聚合

`<aggregation>_over_time()`

`avg_over_time(range-vector)`: 范围向量内每个度量指标的平均值。

`min_over_time(range-vector)`: 范围向量内每个度量指标的最小值。

`max_over_time(range-vector)`: 范围向量内每个度量指标的最大值。

`sum_over_time(range-vector)`: 范围向量内每个度量指标的求和值。

`count_over_time(range-vector)`: 范围向量内每个度量指标的样本数据个数

聚合函数

Prometheus支持以下内置聚合函数, 这些函数可用于聚合单个即时向量的元素, 从而生成具有聚合

的较少元素的新向量：

- **sum** (在维度上求和)
- **max** (在维度上求最大值)
- **min** (在维度上求最小值)
- **avg** (在维度上求平均值)
- **stddev** (求标准差)
- **stdvar** (求方差)
- **count** (统计向量元素的个数)
- **count_values** (统计相同数据值的元素数量)
- **bottomk** (样本值第k个最小值)
- **topk** (样本值第k个最大值)
- **quantile** (统计分位数)

操作符

除了能够方便的查询和过滤时间序列以外，还支持丰富的操作符，用户可以使用这些操作符进一步的事件序列进行二次加工。这些操作符包括：数学运算符，逻辑运算符，布尔运算符等等。

数学操作符

操作数可以是一个常数，也可以是一个查询表达式，比如：

`kong_http_status{service="sky-cmdb"} > 2000`

The screenshot shows a Prometheus query interface. At the top, a text input field contains the query: `kong_http_status{service="sky-cmdb"} > 2000`. To the right of the input, the following performance metrics are displayed: Load time: 122ms, Resolution: 14s, Total time series: 1. Below the input is an 'Execute' button and a dropdown menu with the text '- insert metric at cursor -'. Underneath are two tabs: 'Graph' and 'Console'. The 'Console' tab is active, showing a table with two columns: 'Element' and 'Value'. The table contains one row with the element `kong_http_status{code="200",instance="kong.kong:8001",job="prometheus",service="sky-cmdb"}` and the value 2639. At the bottom right of the console output, there is a 'Remove Graph' link.

Element	Value
<code>kong_http_status{code="200",instance="kong.kong:8001",job="prometheus",service="sky-cmdb"}</code>	2639

支持的所有数学运算符如下所示：

- **+** (加法)
- **-** (减法)
- ***** (乘法)
- **/** (除法)
- **%** (求余)
- **^** (幂运算)