# Effective management of high volume numeric data with histograms

Fred Moyer @Circonus
DataEngConf SF ‘18

@phredmoyer


- Engineer to Engineer @circonus
- Recovering C and Perl programmer
- Geeking out on histograms since 2015

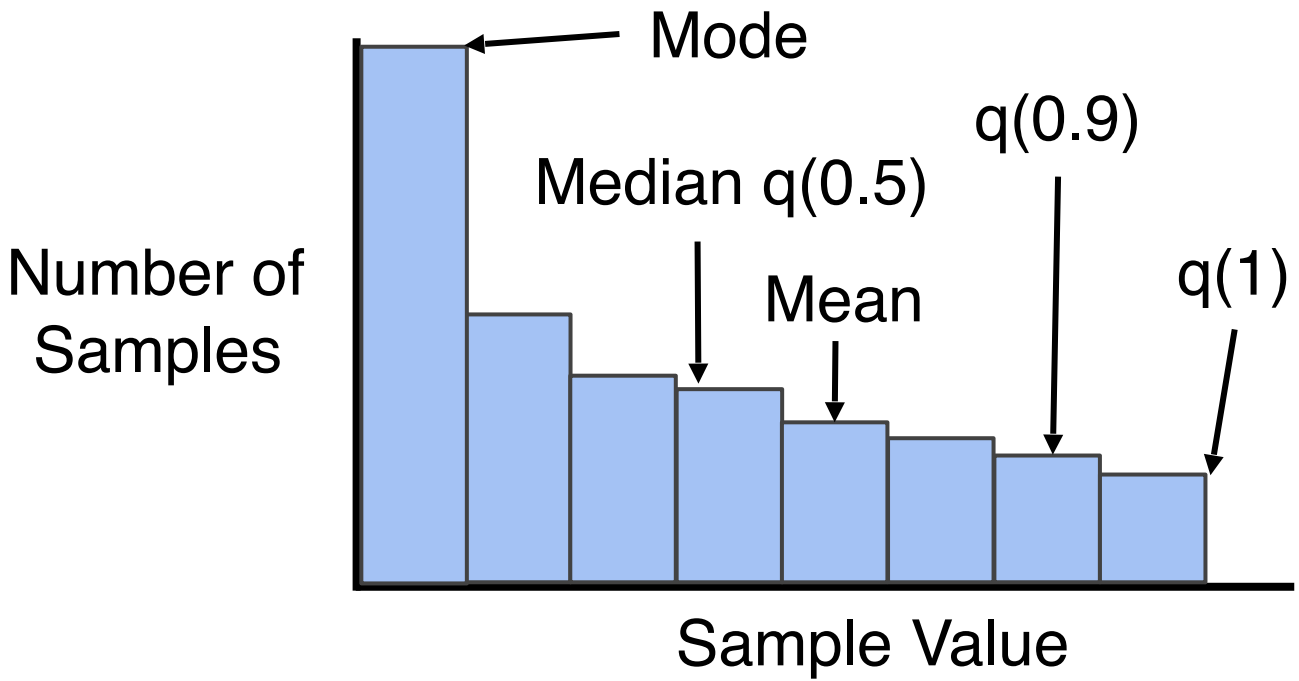
# Pain driven development


- Observability tools caused a telemetry firehose
- Existing monitoring systems got washed away
- Average based metrics gave limited insight

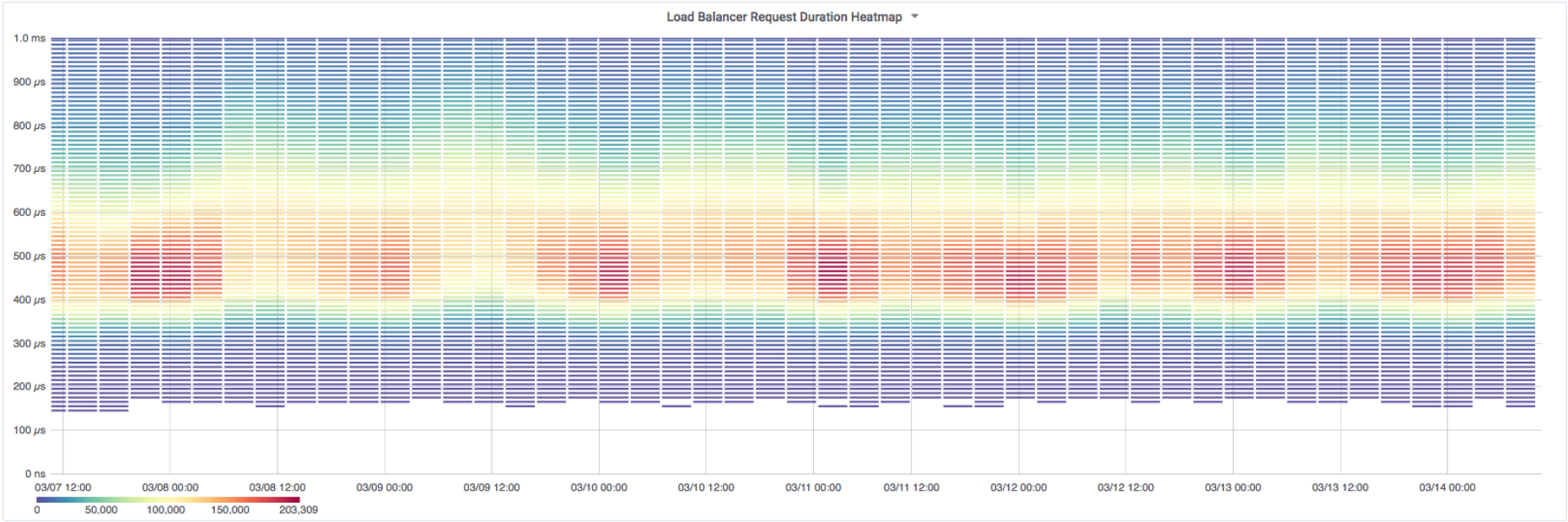
# “Effective Management”


- Performance AND scalability
- Avoid memory allocations, copies, locks, waits
- Persist data in size efficient structures

# Histogram Basics



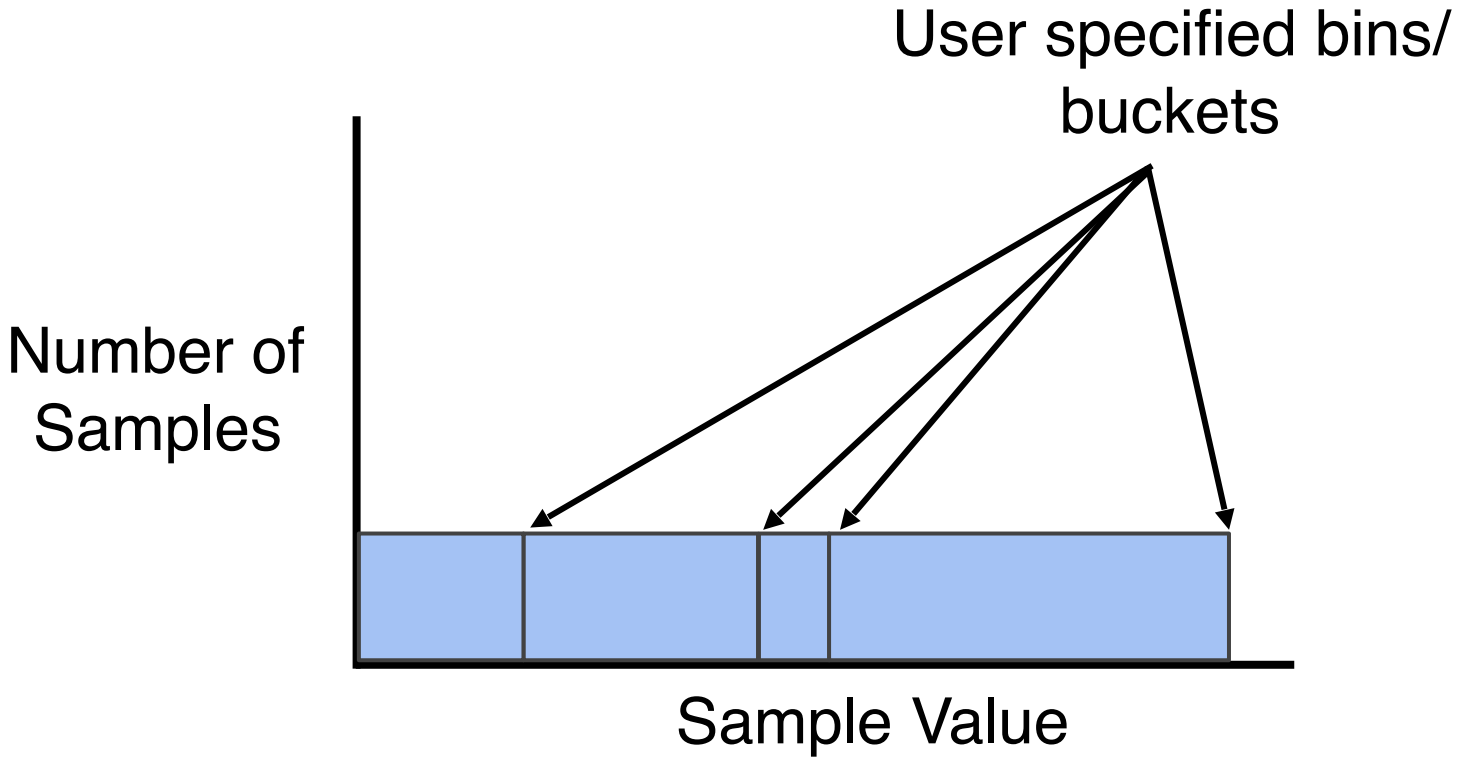
# Heatmap Basics



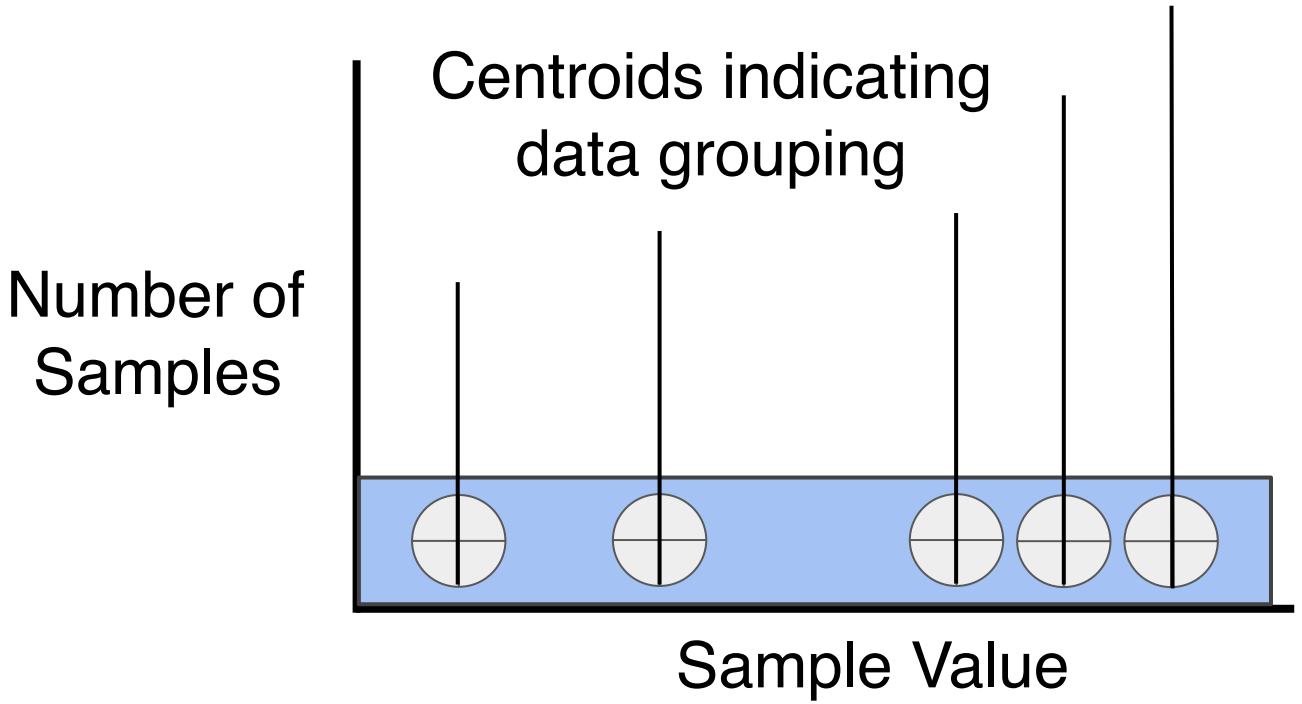
# Histogram Types


- Fixed Bucket
- Approximate
- Linear
- Log Linear
- Cumulative

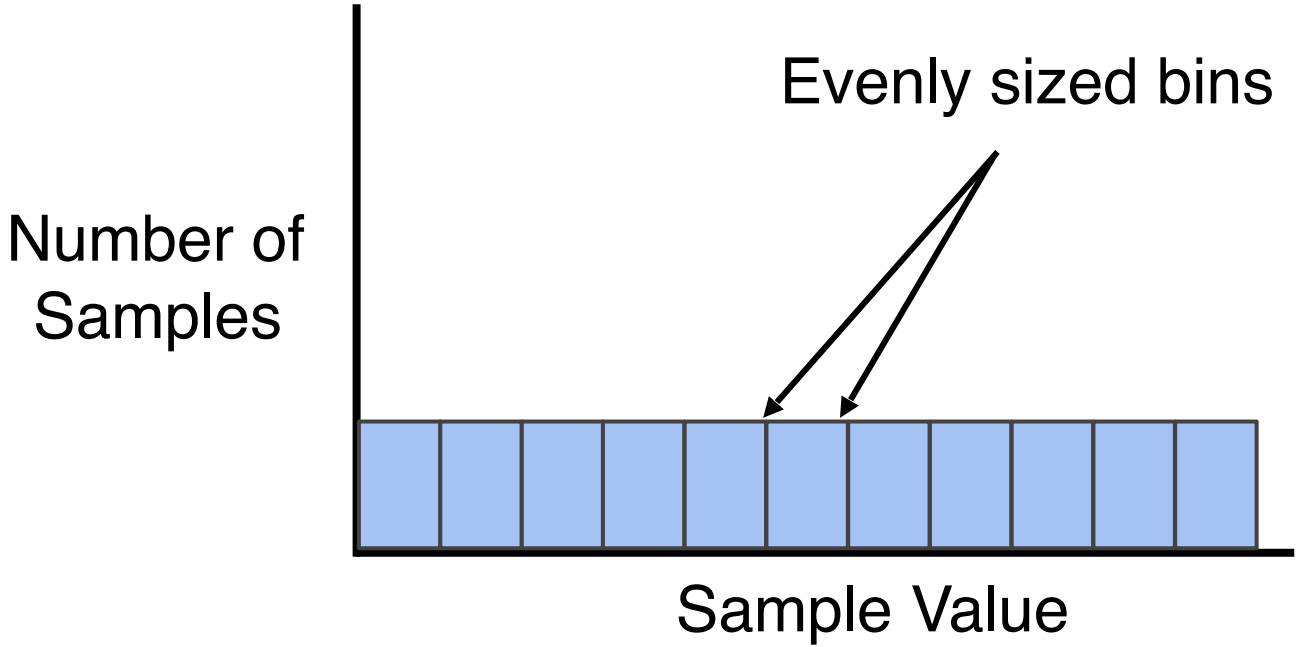
# Fixed Bucket



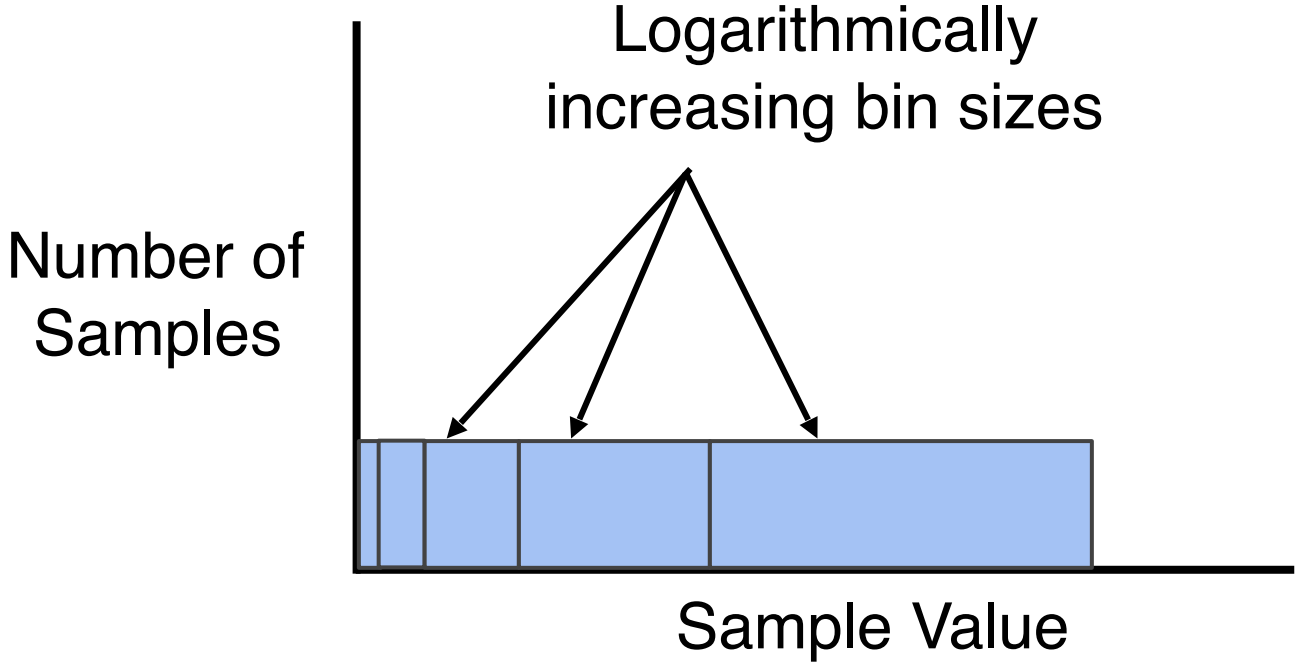
# Approximate



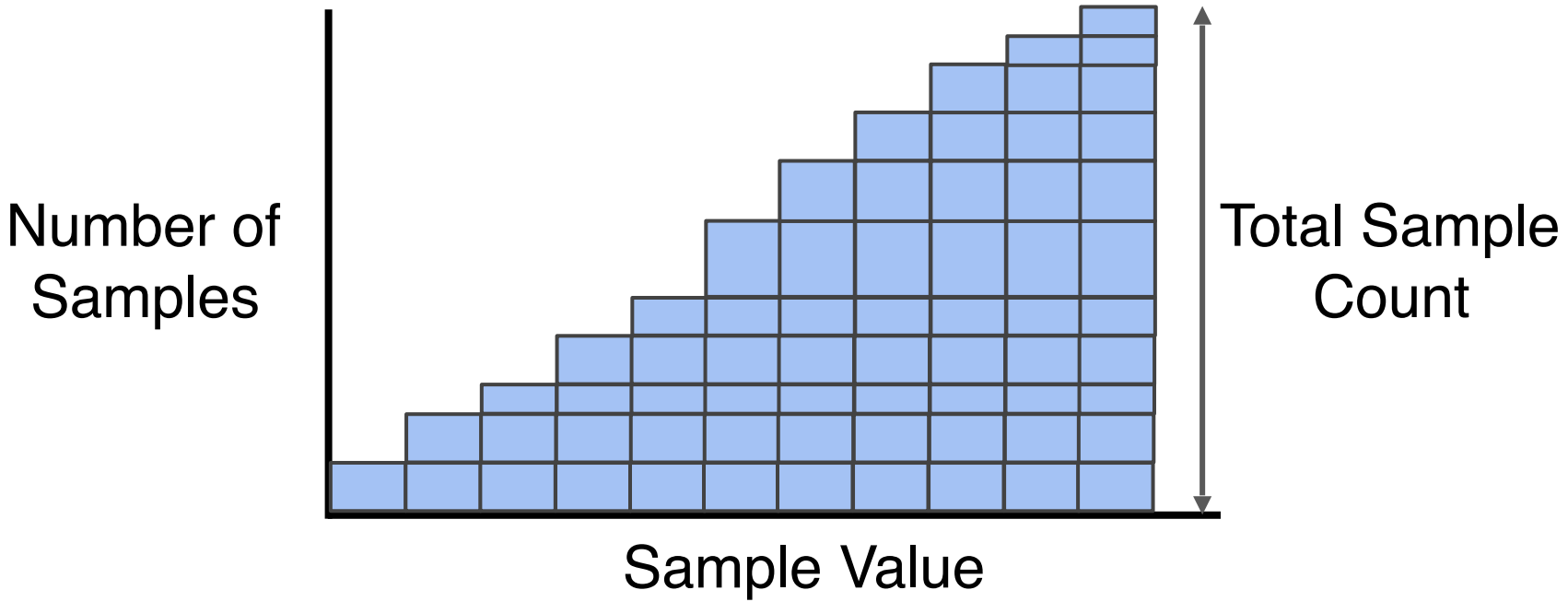
# Linear



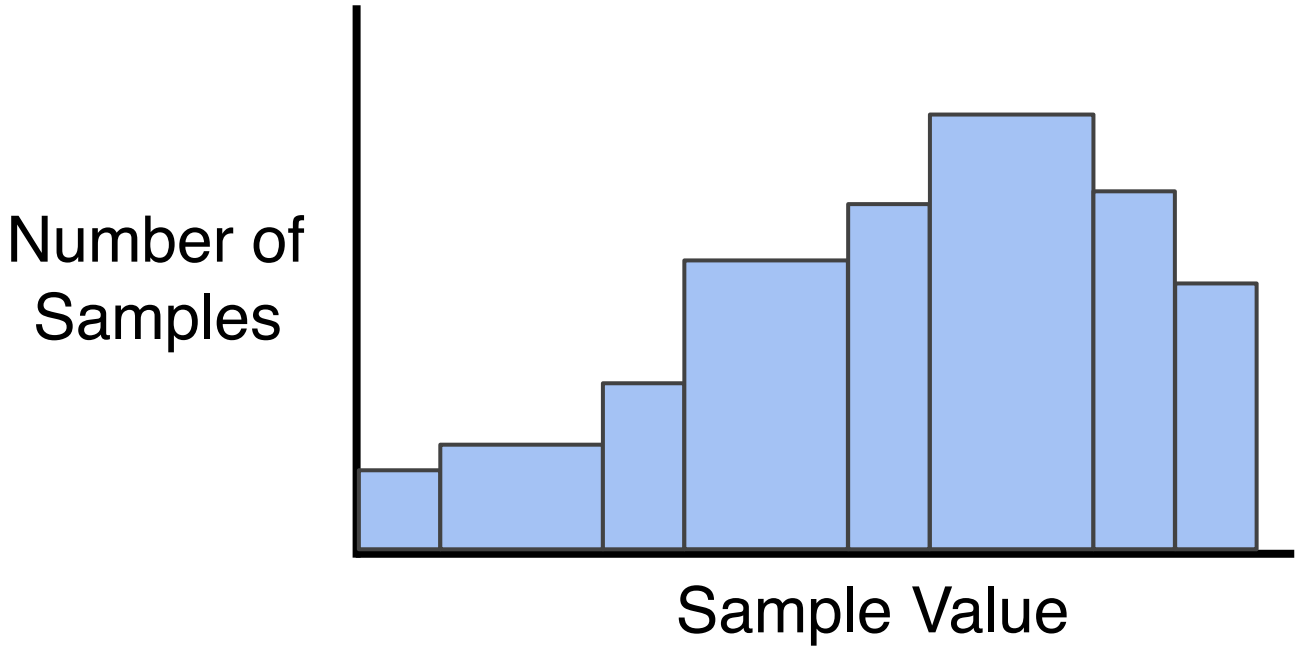
# Log Linear


Logarithmically
increasing bin sizes
Number of
Samples
Sample Value

# Cumulative



# Custom



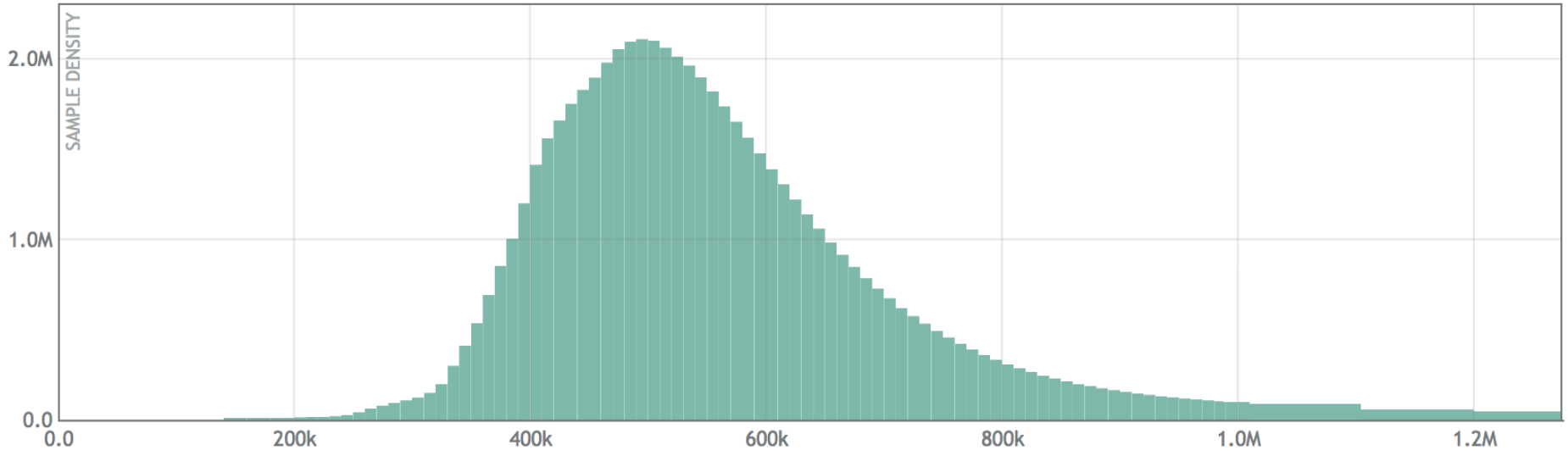
# Open Source Log Linear


C - github.com/circonus-labs/libcircllhist

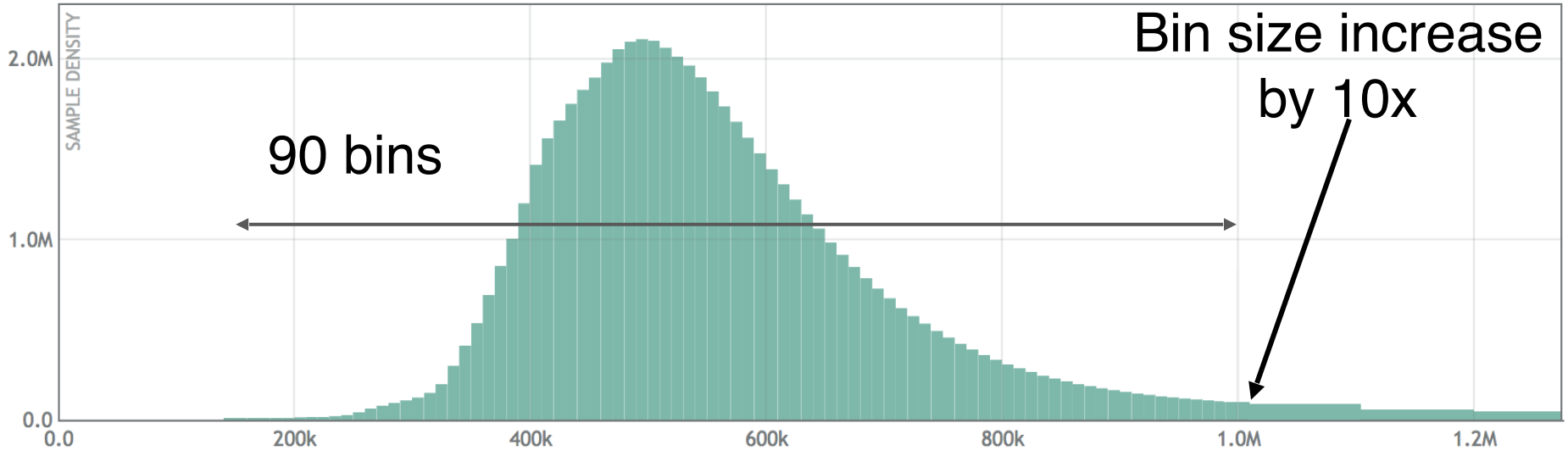
Go - github.com/circonus-labs/circonusllhist

# Open Source Log Linear

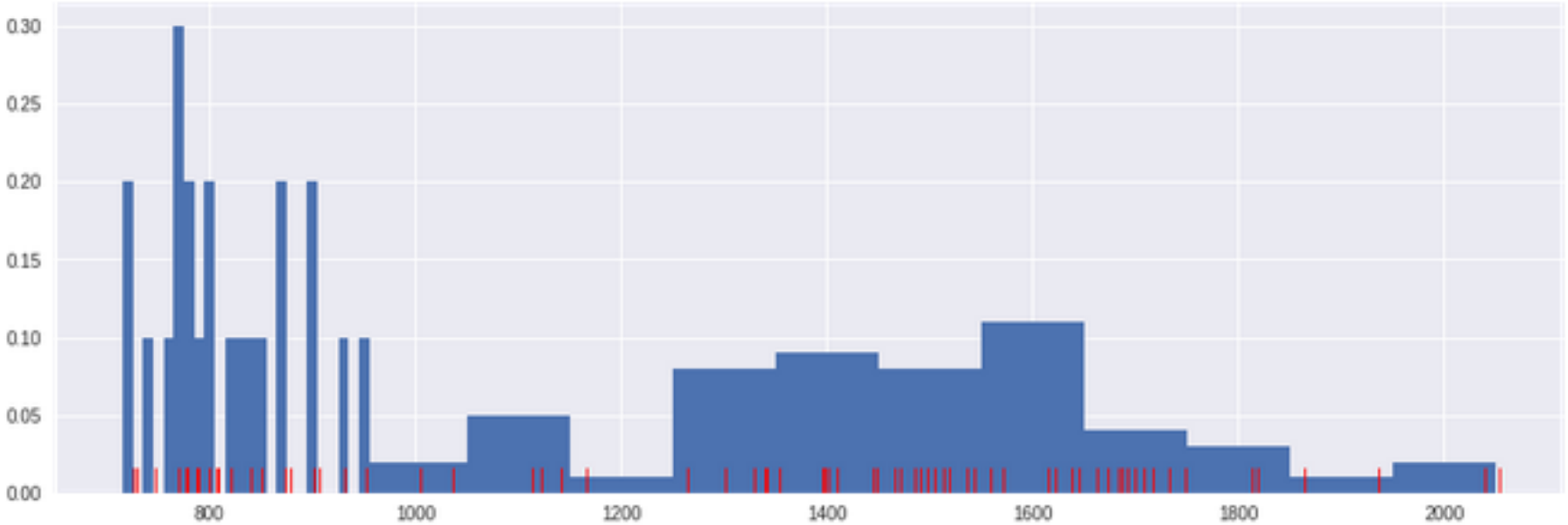
CIRCONUS



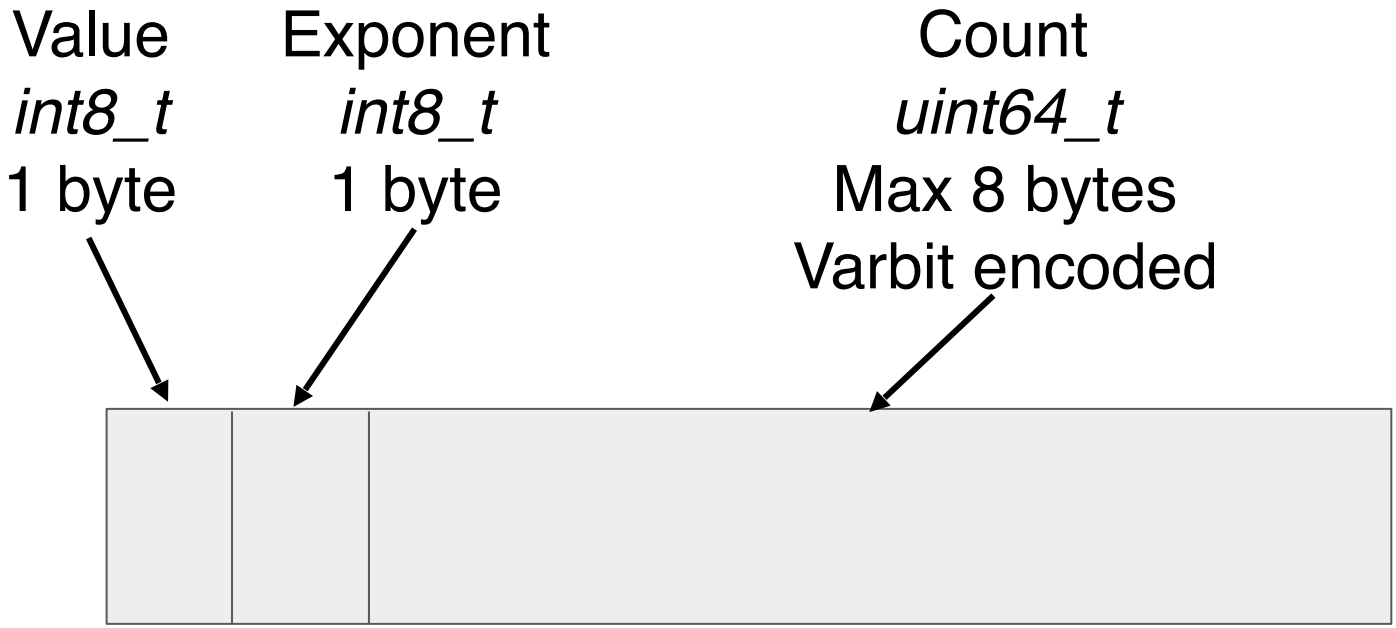
# Open Source Log Linear



# Open Source Log Linear



# Bin data structure



# Storage efficiency - 1 month

CIRCONUS

30 days of one minute histograms

30 days * 24 hours/day * 60 bins/hour * 300 bin span * 10 bytes/bin * 1kB/1,024bytes * 1MB/1024kB = 123.6 MB

# Storage efficiency - 1 year

CIRCONUS

365 days of five minute histograms

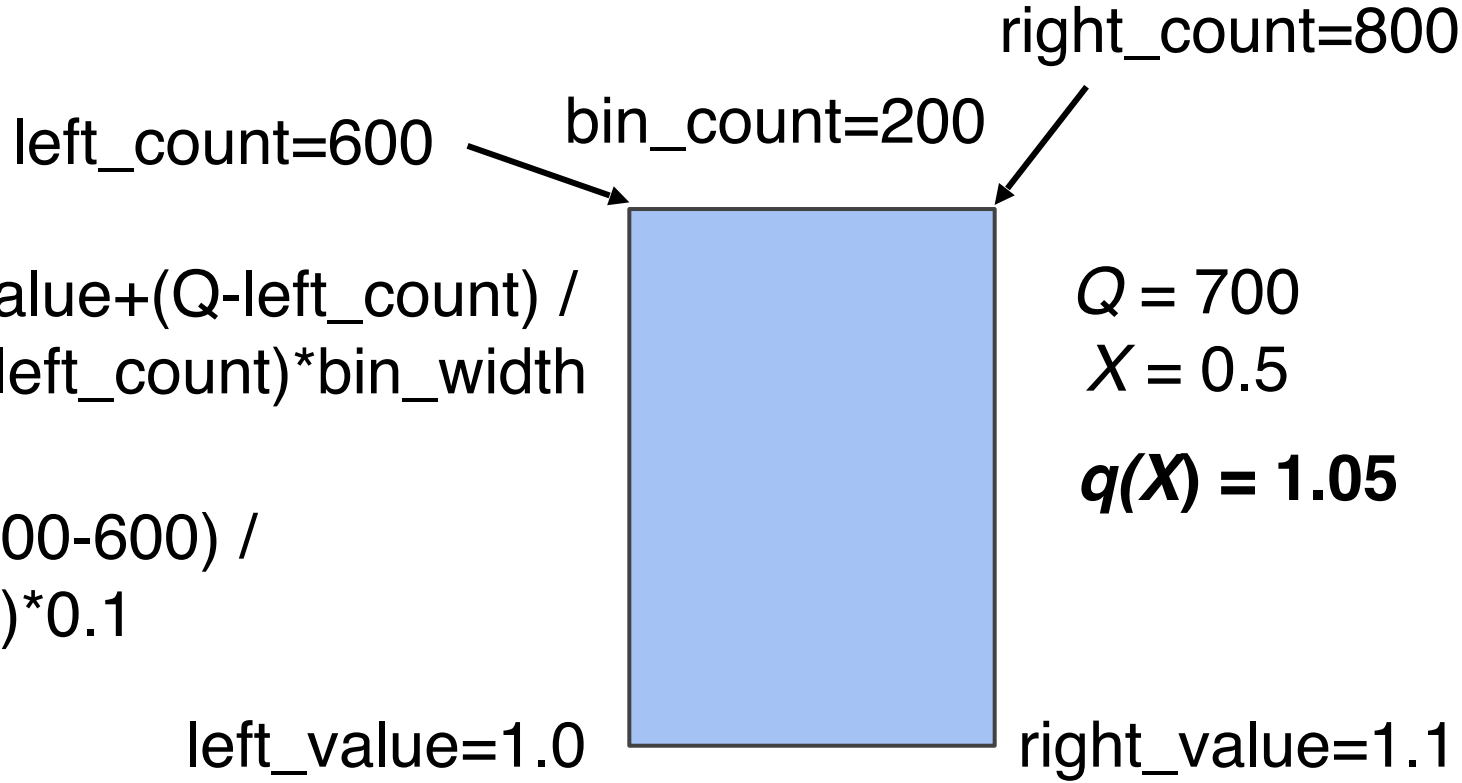
365 days * 24 hours/day * 12 bins/hour * 300 bin span * 10 bytes/bin * 1kB/1,024bytes * 1MB/1024kB = 300.9 MB

# Quantile calculation

CIRCONUS

1. Given a quantile *q(X)* where *0 < X < 1*
2. Sum up the counts of all the bins, *C*
3. Multiply *X * C* to get count *Q*
4. Walk bins, sum bin boundary counts until *> Q*
5. Interpolate quantile value *q(X)* from bin

# Linear interpolation



# Recap


- Several different types of histograms
- Highly space efficient
- O(1) and O(n) complexity calculating quantiles
- What other fun things can we do?

# Inverse Quantiles

CIRCONUS

- What's the 95th percentile latency?
  - q(0.95) = 10ms
- What percent of requests exceeded 10ms?
  - 5% for this data set; what about others?

# Inverse Quantile calculation

CIRCONUS

1. Given a sample value X, locate its bin
2. Using the previous linear interpolation equation, solve for Q given X

# Inverse Quantile calculation

CIRCONUS

$X$ = left_value+(Q-left_count) / (right_count-left_count)*bin_width

X**-left_value** = (Q-left_count) / (right_count-left_count)*bin_width

(X-left_value)**/bin_width** = (Q-left_count)/(right_count-left_count)

(X-left_value)/bin_width***(right_count-left_count)** = Q-left_count

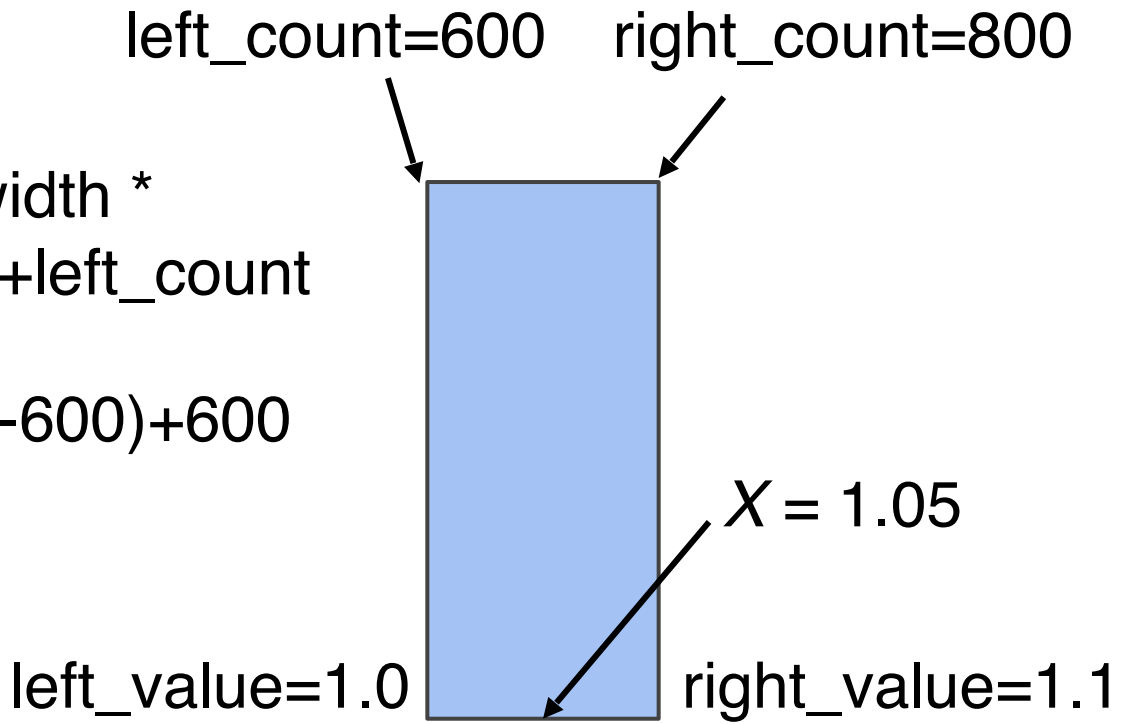
**Q** = (X-left_value)/bin_width*(right_count-left_count)**+left_count**

# Linear interpolation

Q =(X-left_value)/bin_width *
(right_count-left_count)+left_count

*Q* = (1.05-1.0)/0.1*(800-600)+600

***Q* = 700**

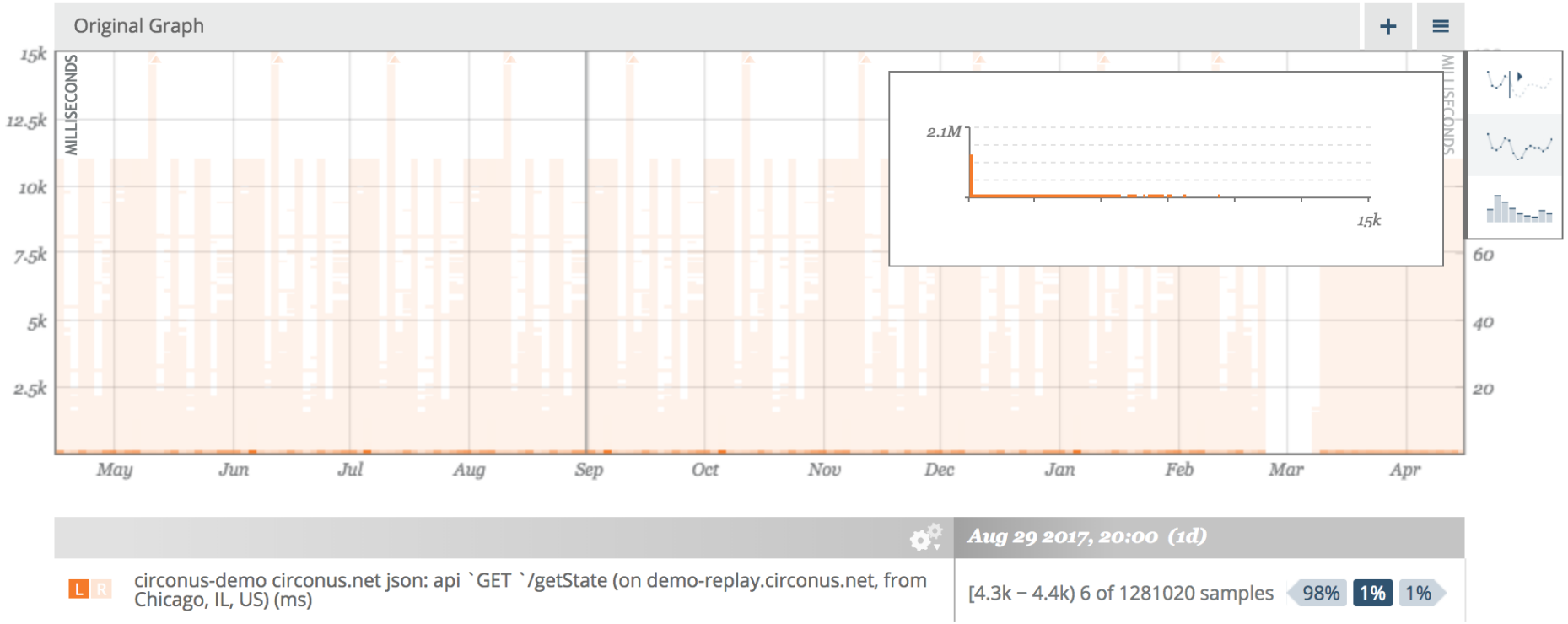


# Inverse Quantile calculation

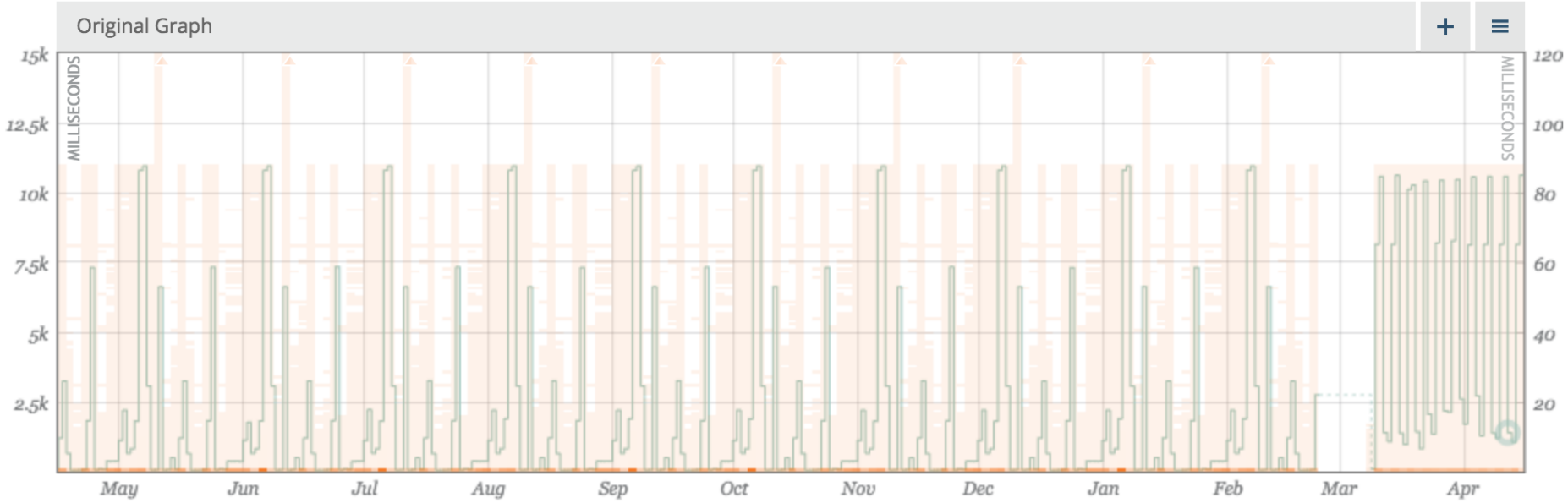
CIRCONUS

1. ~~Given a sample value *X*, locate its bin~~
2. ~~Using the previous linear interpolation equation, solve for *Q* given *X*~~
3. Sum the bin counts up to *Q* as $Q_{left}$
4. Inverse quantile $q_{inv}(X) = (Q_{total}-Q_{left})/Q_{total}$
5. For $Q_{left}=700$, $Q_{total} = 1{,}000$, $q_{inv}(X) = 0.3$
6. 30% of sample values exceeded *X*

# Quantiles - Heatmap

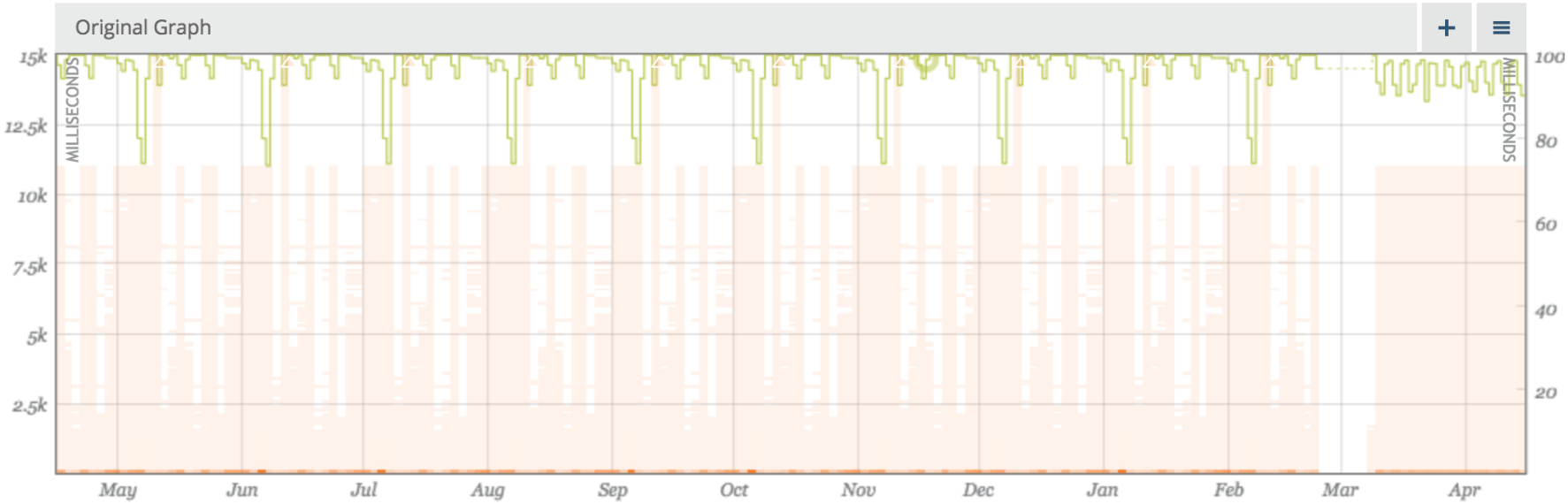


# Quantiles - q(0.9)



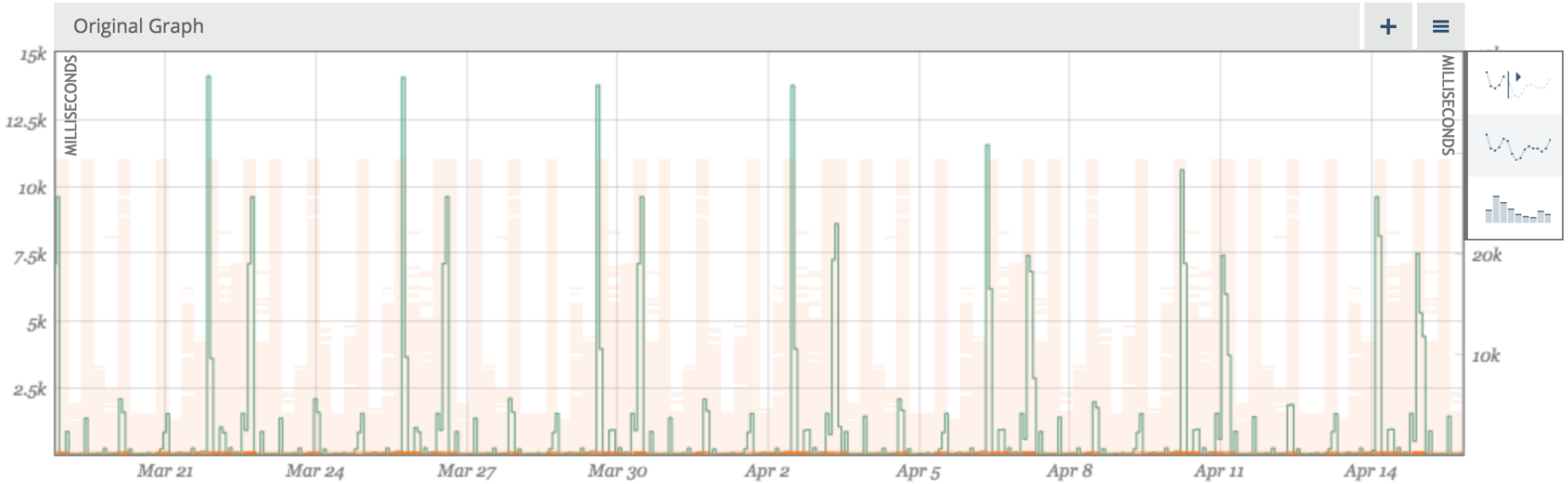

| | Apr 11 2018, 20:00 (1d) |
|---|---|
| L R 99th percentile | 1.3913050314464999k |
| L R circonus-demo circonus.net json: api `GET `/getState (on demo-replay.circonus.net, from Chicago, IL, US) | - |

# Inverse Quantiles - SLO



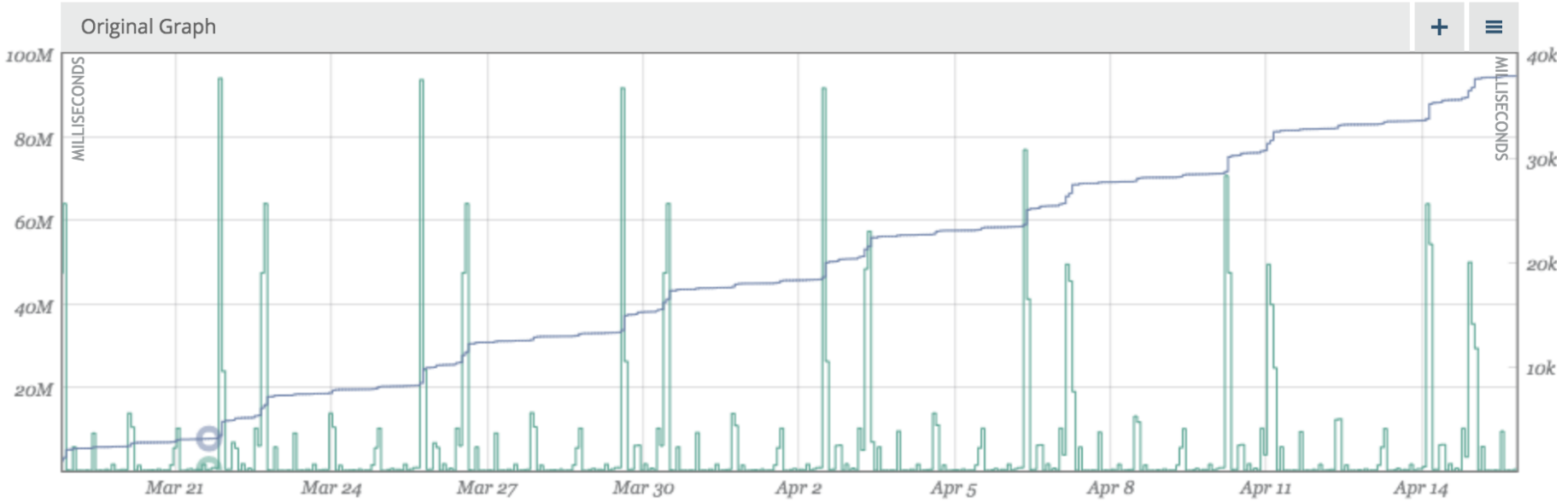

| | Nov 17 2017, 19:00 (1d) |
|---|---|
| Inverse quantile calculation for 500 milliseconds | 98.783298034185 |
| circonus-demo circonus.net json: api `GET `/getState (on demo-replay.circonus.net, from Chicago, IL, US) | [12k – 13k) 0 of 361939 samples 100% 0% 0% |

# Inverse Quantiles - SLO



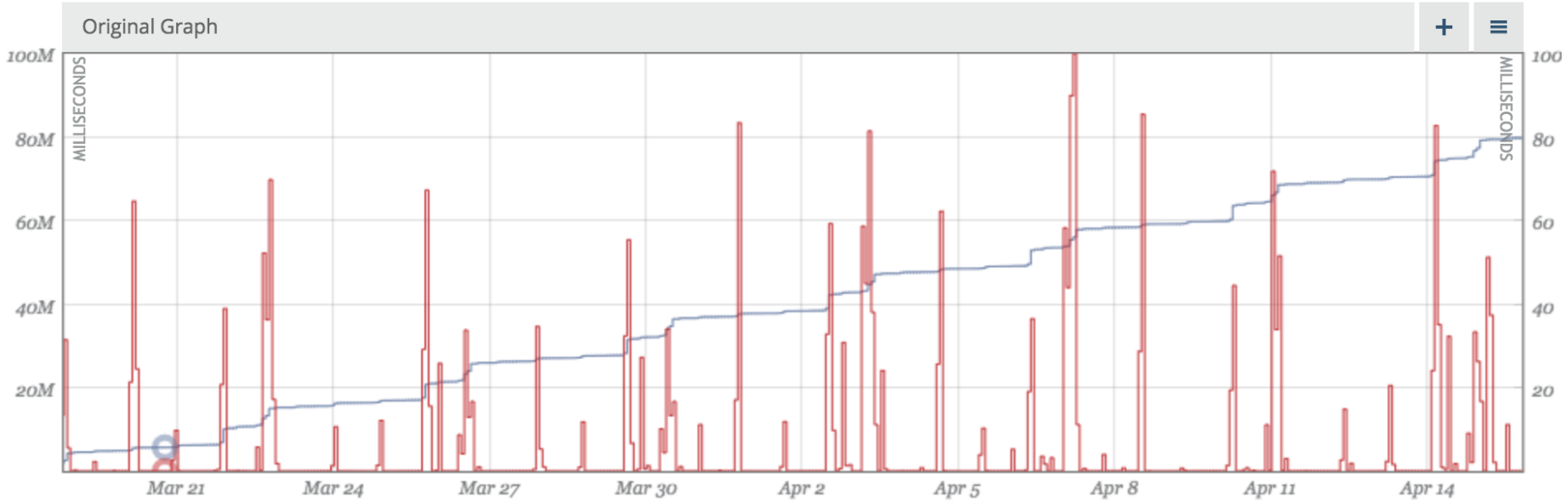

| | Mar 18 2018, 18:30 (1h 30M) |
|---|---|
| L R Inverse quantile calculation count of requests that violated 500 millisecond SLO | 18.959302325581k |
| L R circonus-demo circonus.net json: api `GET `/getState (on demo-replay.circonus.net, from Chicago, IL, US) | - |

# Inverse Quantiles - SLO




| | Mar 21 2018, 15:30 (1h 30M) |
|---|---|
| Inverse quantile calculation count of requests that violated 500 millisecond SLO | 0.14487179487179k |
| Cumulative number of requests exceeded the 200 millisecond SLO | 7.68593M |

# Anomalies




| | Mar 20 2018, 18:30 (1h 30M) |
|---|---|
| L R Cumulative number of requests exceeded the 500 millisecond SLO | 5.6637122222222M |
| L R SLO violation anomaly detection | 0 |

# Thank you!


Questions?

Bug me at the Circonus booth

Come to Office Hours

Tweet @phredmoyer or @circonus