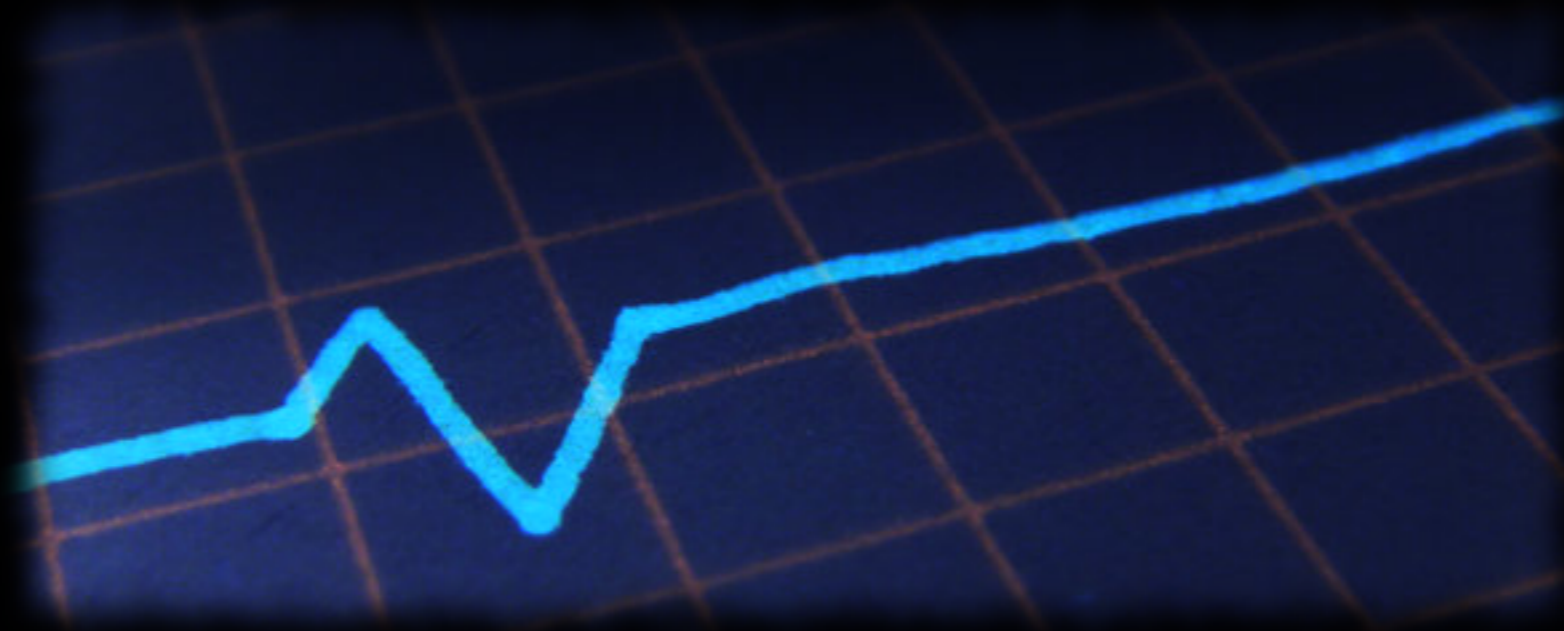


OpenTSDB

A Distributed, Scalable, Time Series Database

“Monitoring at an unprecedented level of granularity”



Benoît “tsuna” Sigoure
tsuna@stumbleupon.com

Where's my Paradigm Shift?



Get Real-Time Data from your Infrastructure



Working at Scale



No SPoF



HBase

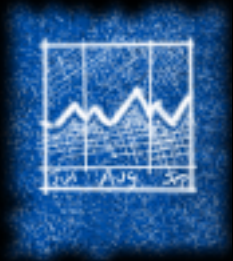
Distributed

Scalable

Reliable

Efficient



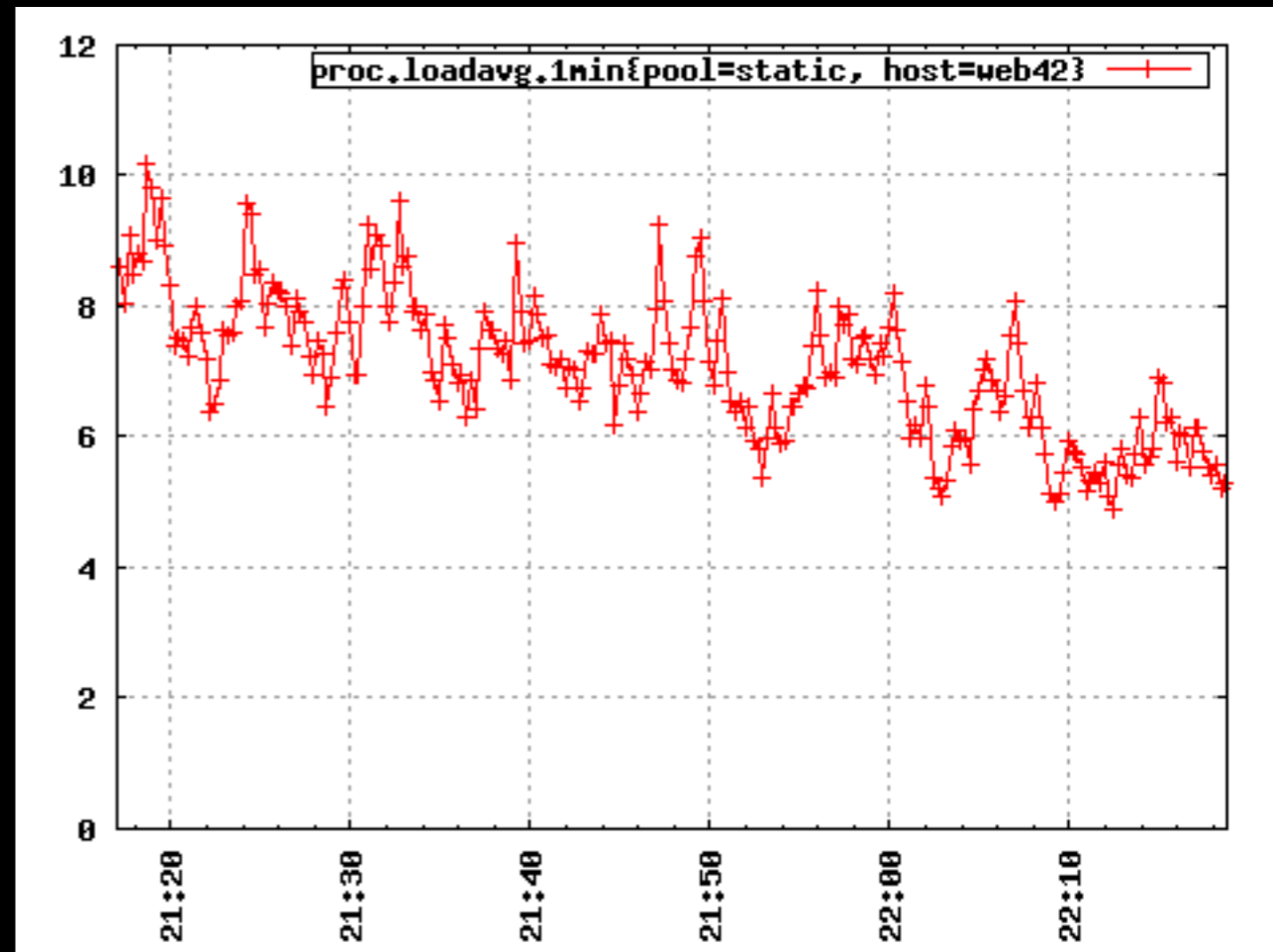


Design Goals

- Distributed storage of monitoring data
- No Single Point of Failure
- Pulling custom graphs must be trivial & fast
- Scale to:
 - Thousands of machines
 - Many billions of data points

Key concepts

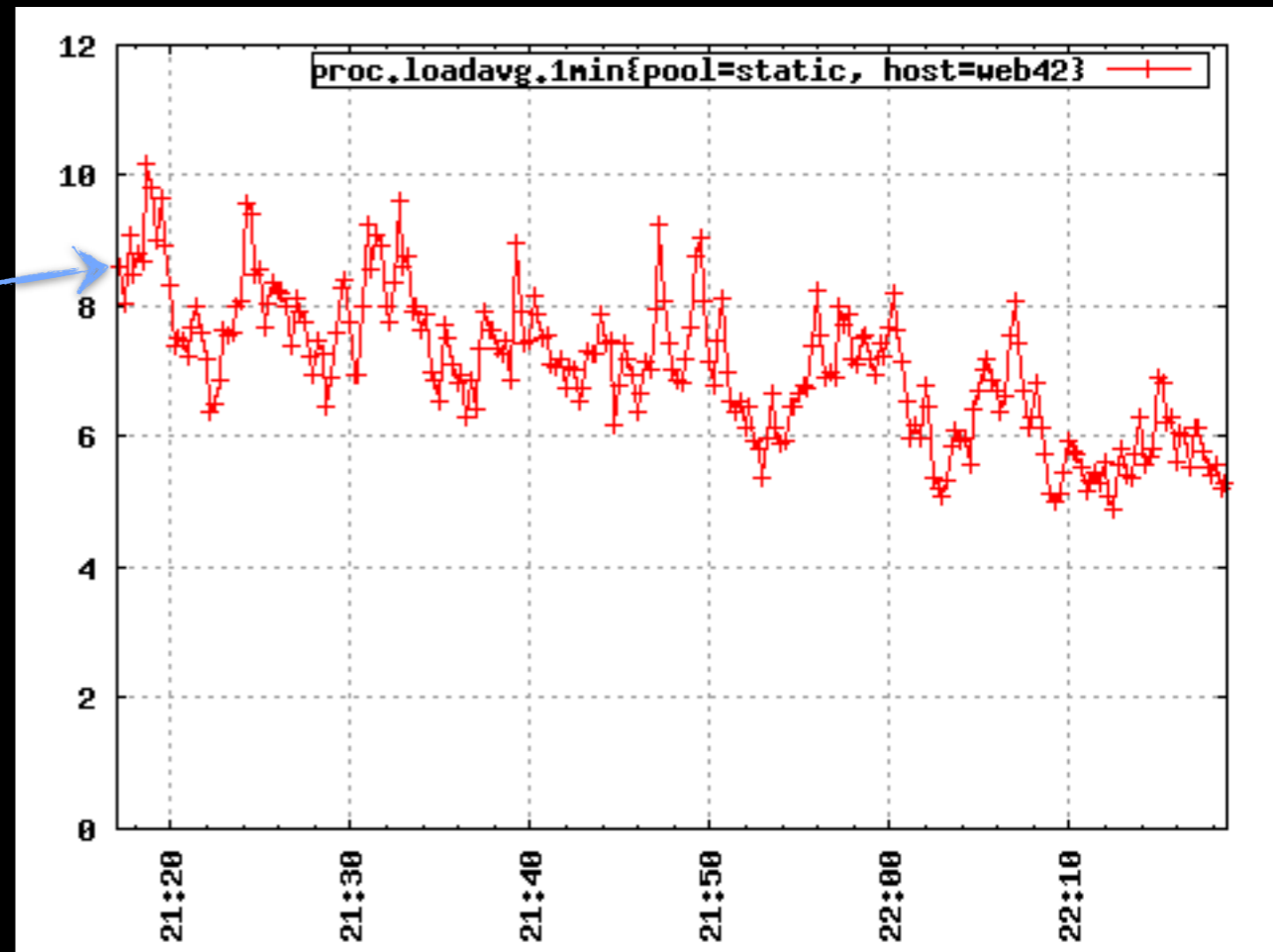
- Data Points
(time, value)
- Metrics
`proc.loadavg.1m`
- Tags
`host=web42 pool=static`
- Metric + Tags = Time Series



```
put proc.loadavg.1m 1234567890 0.42 host=web42 pool=static
```


Key concepts

- Data Points
(time, value)
- Metrics
`proc.loadavg.1m`
- Tags
`host=web42 pool=static`
- Metric + Tags = Time Series



```
put proc.loadavg.1m 1234567890 0.42 host=web42 pool=static
```

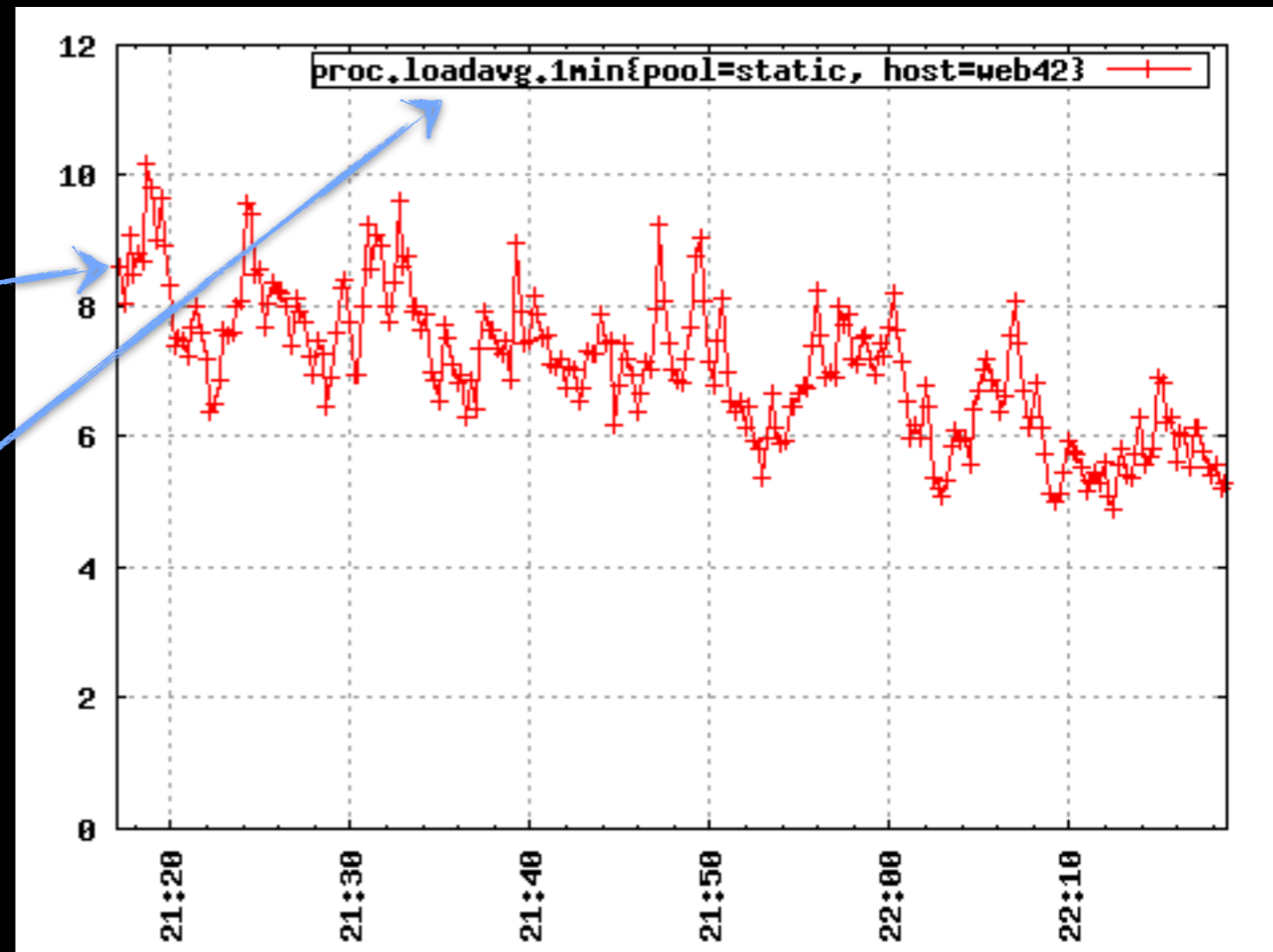
Key concepts

- Data Points
(time, value)

- Metrics
`proc.loadavg.1m`

- Tags
`host=web42 pool=static`

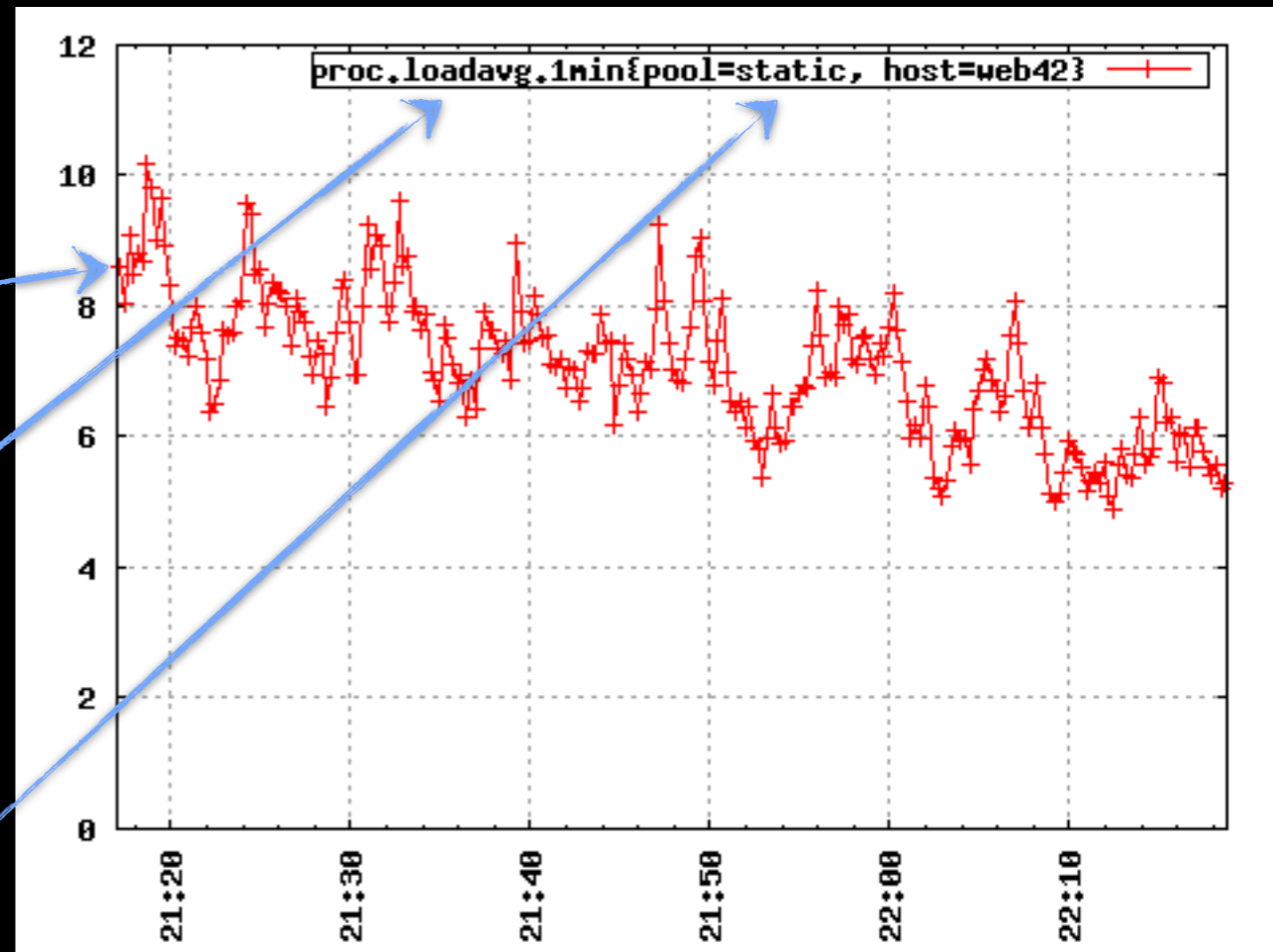
- Metric + Tags = Time Series



```
put proc.loadavg.1m 1234567890 0.42 host=web42 pool=static
```

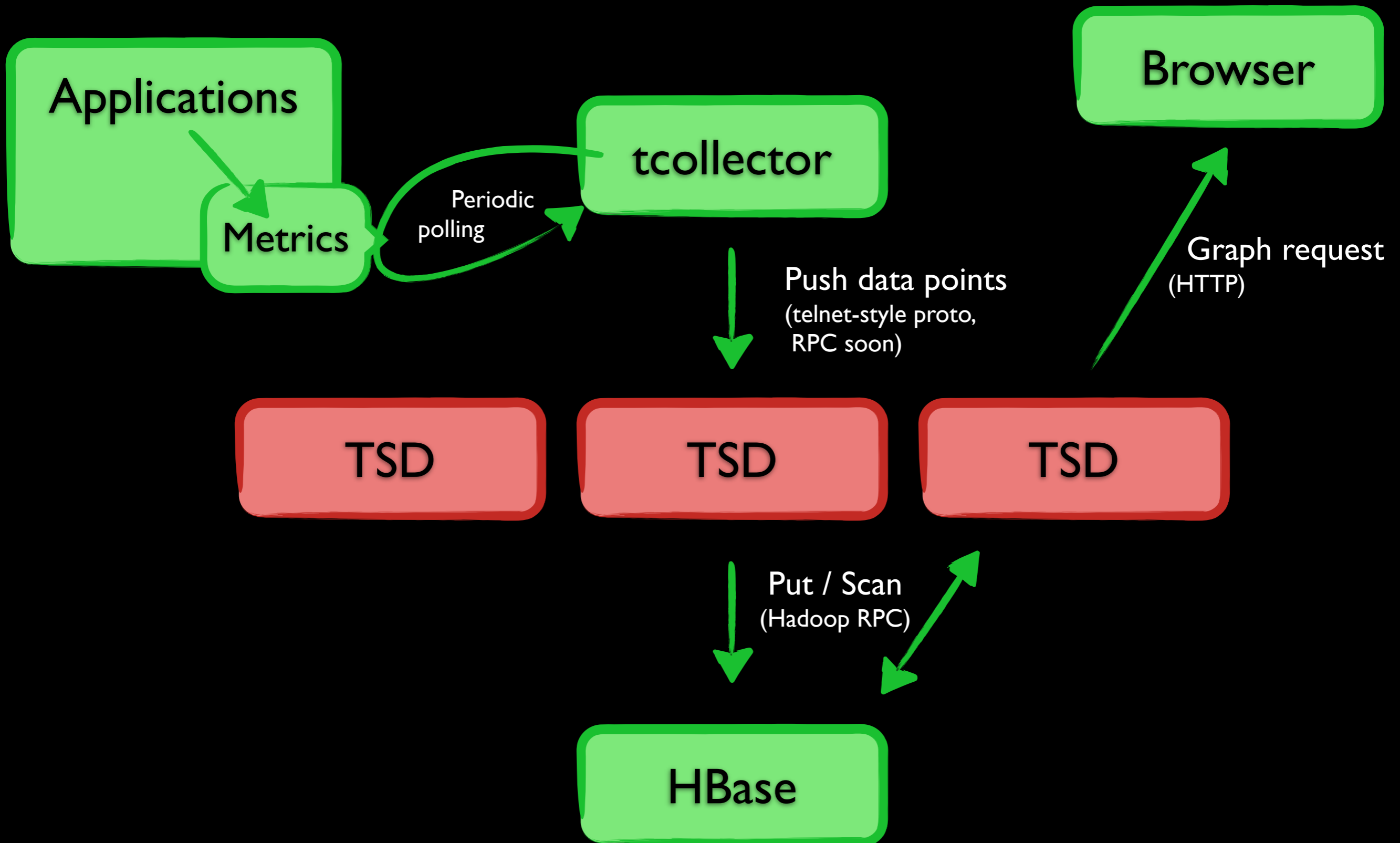
Key concepts

- Data Points
(time, value)
- Metrics
`proc.loadavg.1m`
- Tags
`host=web42 pool=static`
- Metric + Tags = Time Series



```
put proc.loadavg.1m 1234567890 0.42 host=web42 pool=static
```

The Big Picture™



12 Bytes Per Datapoint



4TB per year for 1000 machines

OpenTSDB @



150 Million Datapoints/Day

in a typical datacenter

Demo Time!

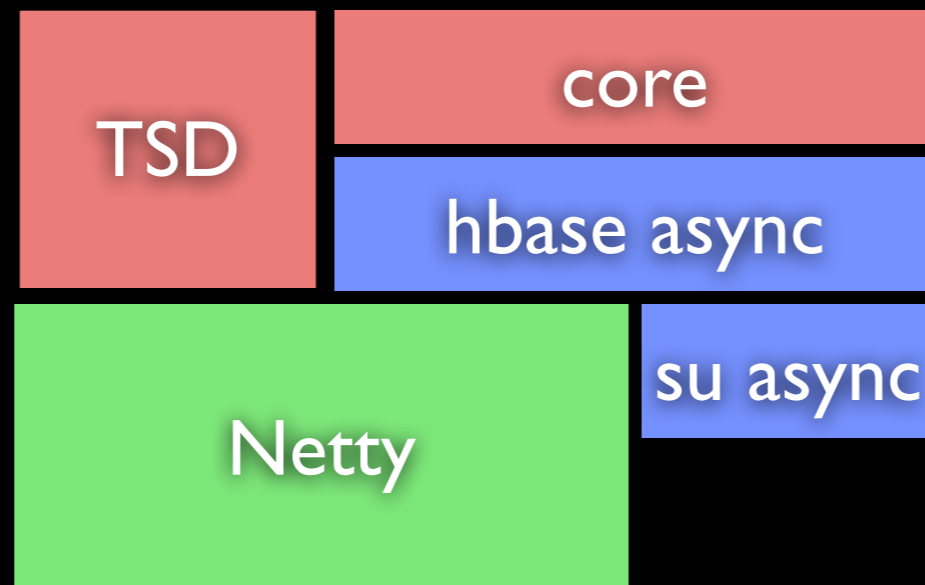


Set it up in 15 minutes

With zero prior experience

- JDK + Gnuplot 1 minute (1 command)
- Single-node HBase 4 minutes (3 commands)
- OpenTSDB 5 minutes (5 commands)
- Deploy tcollector 5 minutes

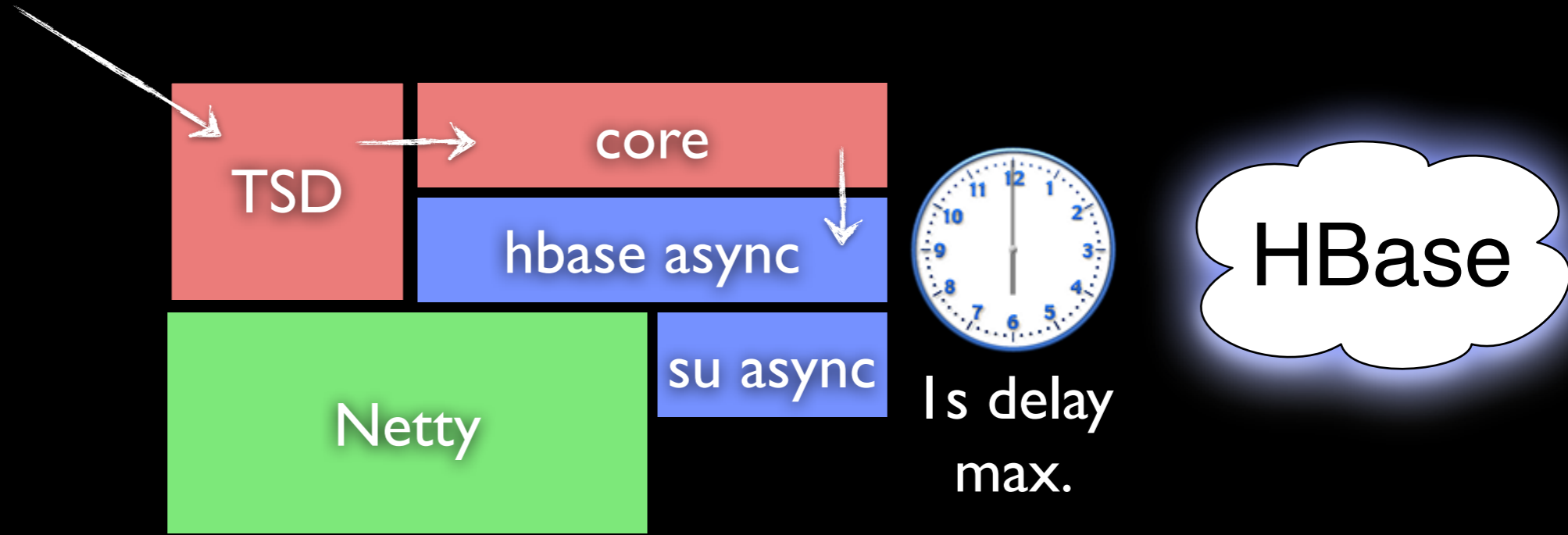
Under the Hood



Local Disk
(cache)

Under the Hood

```
put proc.loadavg.1m 1234567890 0.42 host=web42 pool=static
```



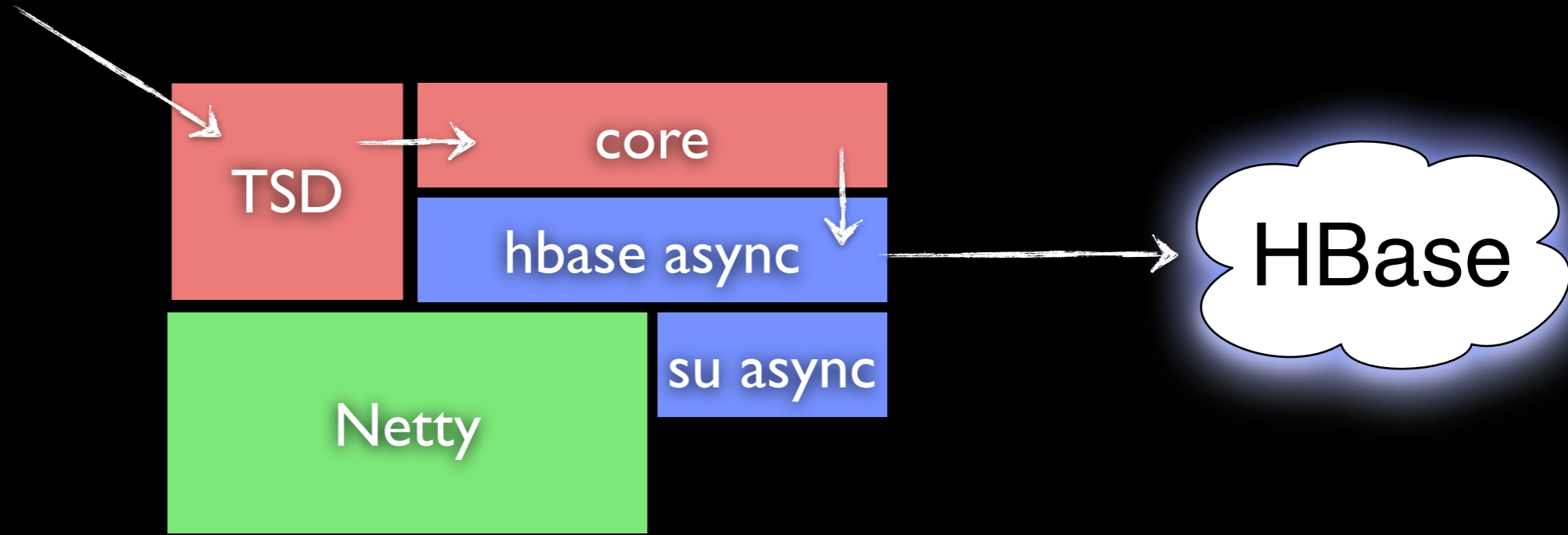
Local Disk
(cache)

Write Path

>2000 data points / sec / core

Under the Hood

```
put proc.loadavg.1m 1234567890 0.42 host=web42 pool=static
```

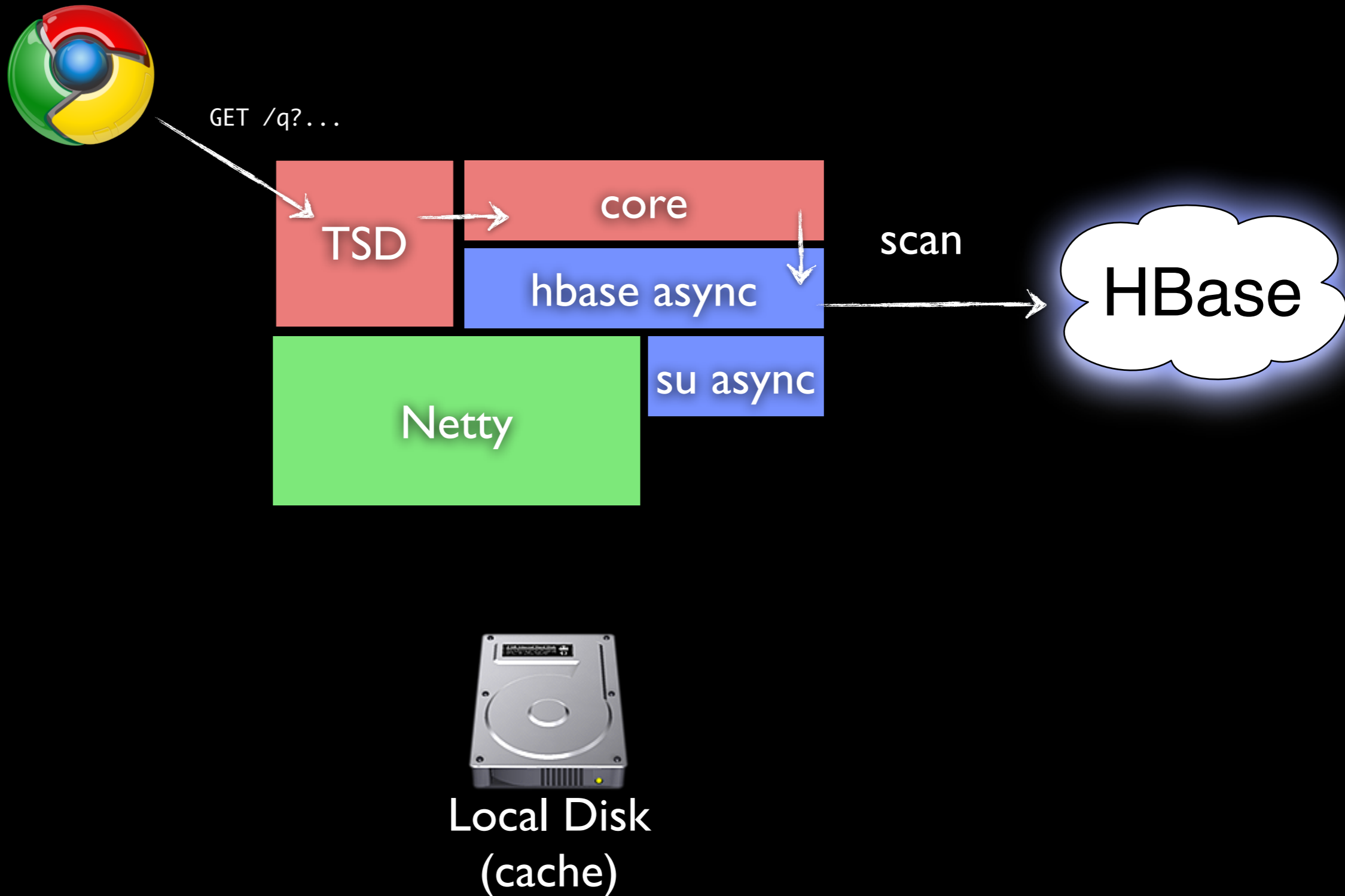


Local Disk
(cache)

Write Path

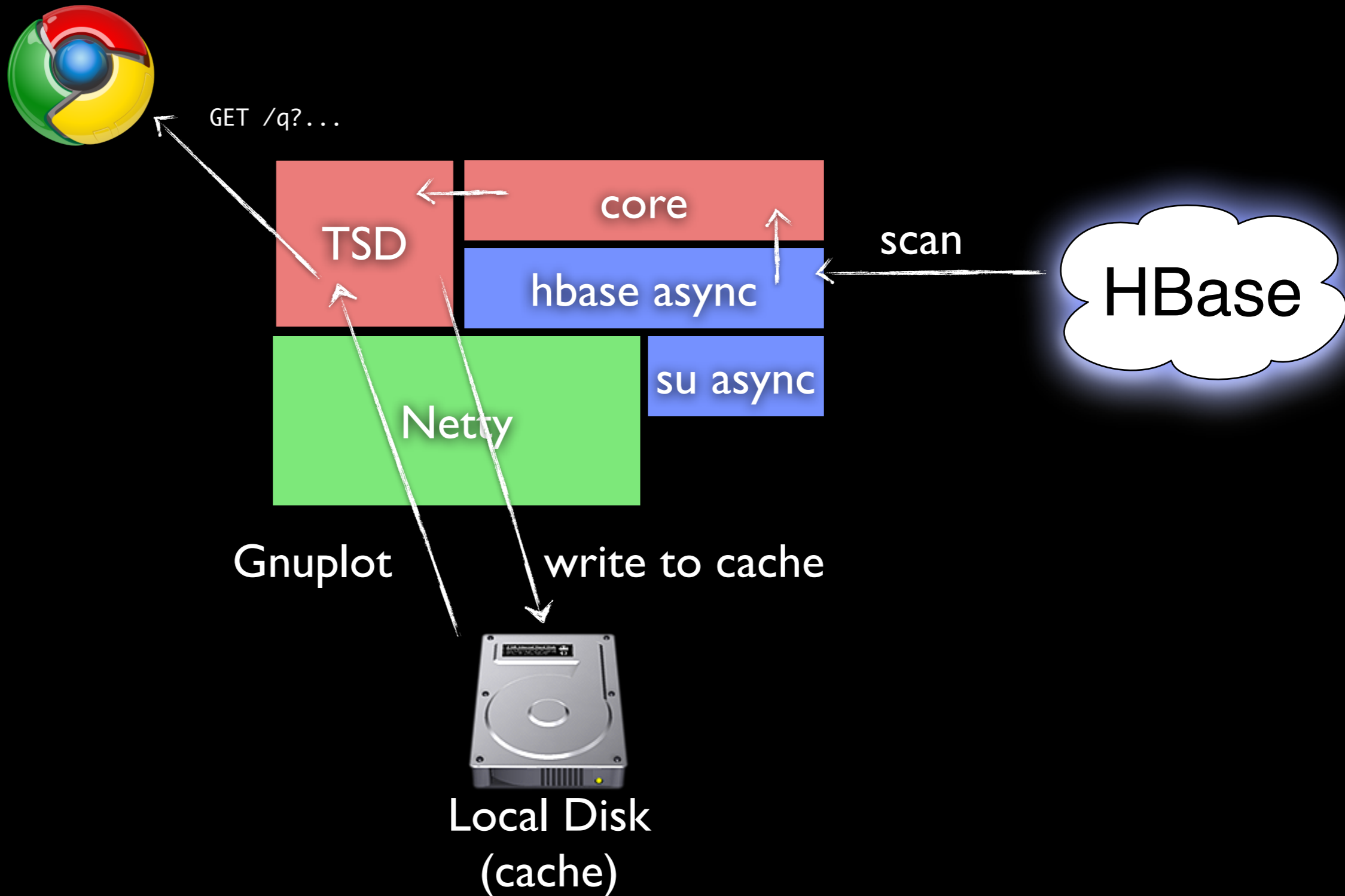
>2000 data points / sec / core

Under the Hood



Read Path

Under the Hood



Read Path

100% Natural, Organic Free & Open-Source



¿ Questions ?

opentsdb.net

Benoît “tsuna” Sigoure
tsuna@stumbleupon.com



Table:
tsdb-uid

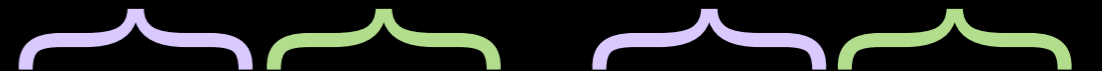
Inside HBase

Row Key	Column Family: name			Column Family: id					
	metrics	tagk	tagv	metrics	tagk	tagv			
<table border="1"><tr><td>0</td><td>0</td><td>1</td></tr></table>	0	0	1		host	static			
0	0	1							
<table border="1"><tr><td>0</td><td>5</td><td>2</td></tr></table>	0	5	2	proc.loadavg. .1m					
0	5	2							
host					<table border="1"><tr><td>0</td><td>0</td><td>1</td></tr></table>	0	0	1	
0	0	1							
proc.loadavg. 1m				<table border="1"><tr><td>0</td><td>5</td><td>2</td></tr></table>	0	5	2		
0	5	2							

0	5	2
---	---	---



0	0	1	0	2	8	0	4	7	0	0	1
---	---	---	---	---	---	---	---	---	---	---	---

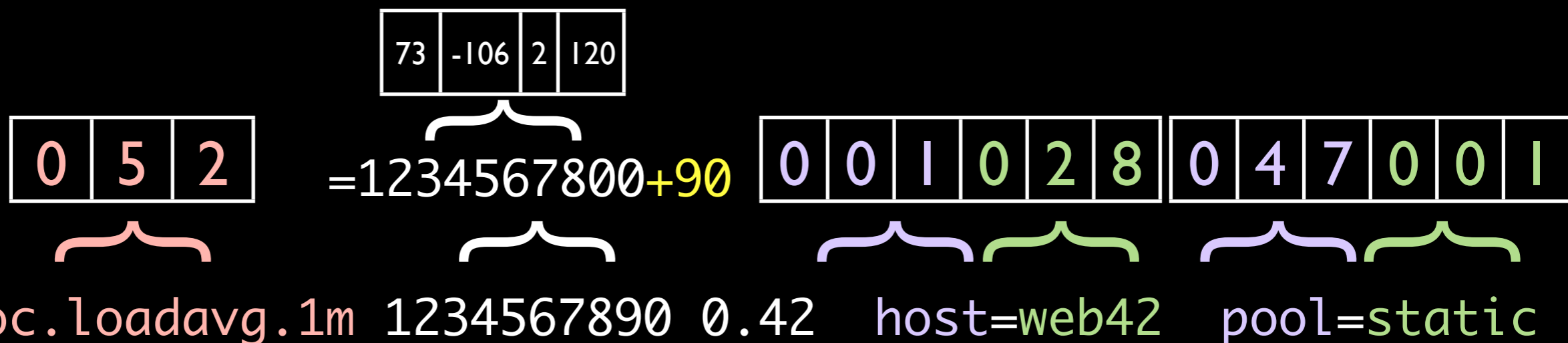


put proc.loadavg.1m 1234567890 0.42 host=web42 pool=static

Table:
tsdb

Inside HBase

Row Key	Column Family: t					
	+0	+15	+20	...	+90	+600
	0.69		0.51		0.42	
	0.99	0.72				

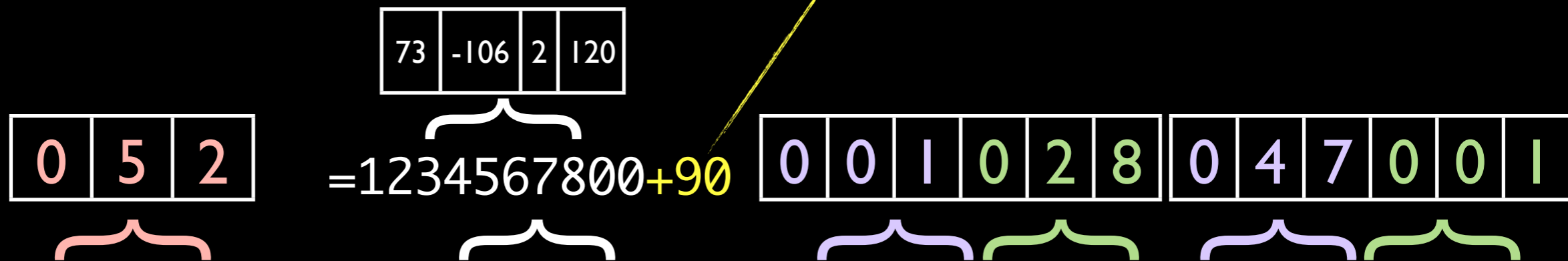


put proc.loadavg.1m 1234567890 0.42 host=web42 pool=static

Table:
tsdb

Inside HBase

Row Key	Column Family: t					
	+0	+15	+20	...	+90	+600
	0.69		0.51		0.42	
	0.99	0.72				



put proc.loadavg.1m 1234567890 0.42 host=web42 pool=static

Table:
tsdb

Inside HBase

Row Key	Column Family: t					
	+0	+15	+20	...	+90	+600
	0.69		0.51		0.42	
	0.99	0.72				

0 5 2

73 -106 2 120

=1234567890+90

0 0 1 0 2 8 0 4 7 0 0 1

put proc.loadavg.1m 1234567890 0.42 host=web42 pool=static

Table:
tsdb

Inside HBase

Row Key	Column Family: t					
	+0	+15	+20	...	+90	+600
	0.69		0.51		0.42	
	0.99	0.72				

0 5 2

73 -106 2 120

=1234567800+90

0 0 1 0 2 8 0 4 7 0 0 1

put proc.loadavg.1m 1234567890 0.42 host=web42 pool=static