

Prometheus - CTJB?



sasaniak @som_zlo



P R O M E T H E U S

Monitoring

- Nagios? Icinga?
- Zabbix?

Monitoring

- Nagios? Icinga?
 - RCE as a service? (NRPE)
- Zabbix?



What does Prometheus do?

It collects data

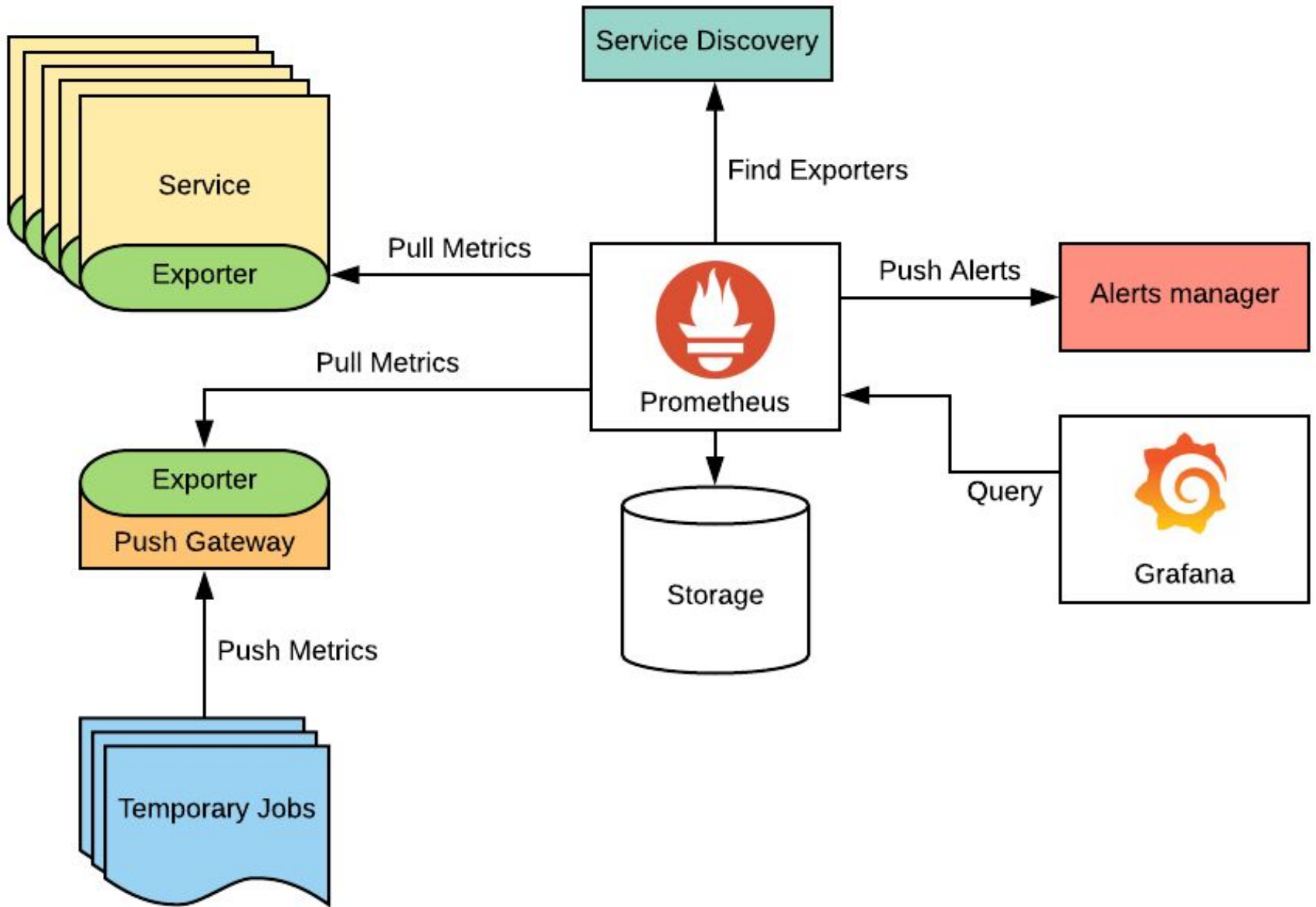
- A shit ton of data
- From everywhere

Encourages instrumentation

Has really nice graphs

What is instrumentation?





Data ingestion: protocol

Simple protocol:

- HTTP transport
- Plain text content (protobuf optional)
- Pull-based collection

Data ingestion: implementation

Very efficient implementation:

- Hundreds of 1000s of metrics/s per server
- Disk-efficient storage (esp. 2.0 +)
- Tunable retention
- Sane defaults!

Data ingestion: sources (I)

node_exporter

- Network, disk, cpu, ram, etc
- Add your custom metrics (text file)

push_gateway

- Cron jobs, short-lived services
- Data that has to be pushed

Data ingestion: exporters

Official

- Node/system metrics
- AWS CloudWatch
- Collectd
- Consul
- Graphite
- HAProxy
- Hystrix metrics
- JMX
- Mesos tasks
- MySQL server
- StatsD bridge

Unofficial

- CouchDB
- Django
- Memcached
- Meteor JS framework
- Minecraft module
- MongoDB
- Munin
- New Relic
- RabbitMQ
- Redis
- Rsyslog
- ...

Data ingestion: instrumentation

Language-specific libraries for instrumentation

Go, Java, Scala, Python, Ruby

Bash, Haskell, Node.js, .NET / C#

Already instrumented: etcd, kubernetes, ...

Roll your own! (it's easy)



Labels



Data processing

Powerful query language. Use it to:

- Browse data - interactive console
- Synthesise metrics from complex calculations
- Create nice graphs
- Wake you up at 3am

Query language: example

Source data:

<code>node_cpu{cpu="cpu0",instance="here.cz:9000",mode="idle"}</code>	16312937.7
<code>node_cpu{cpu="cpu0",instance="here.cz:9000",mode="iowait"}</code>	182080.66
<code>node_cpu{cpu="cpu0",instance="here.cz:9000",mode="system"}</code>	282463.23
<code>node_cpu{cpu="cpu0",instance="here.cz:9000",mode="user"}</code>	552748.8
<code>node_cpu{cpu="cpu0",instance="there.org:9100",mode="idle"}</code>	17914450.35
<code>node_cpu{cpu="cpu0",instance="there.org:9100",mode="iowait"}</code>	81386.28
<code>node_cpu{cpu="cpu0",instance="there.org:9100",mode="system"}</code>	47401.76
<code>node_cpu{cpu="cpu0",instance="there.org:9100",mode="user"}</code>	124549.65
<code>node_cpu{cpu="cpu1",instance="there.org:9100",mode="idle"}</code>	18005086.74
<code>node_cpu{cpu="cpu1",instance="there.org:9100",mode="iowait"}</code>	12934.74
<code>node_cpu{cpu="cpu1",instance="there.org:9100",mode="system"}</code>	44634.8
<code>node_cpu{cpu="cpu1",instance="there.org:9100",mode="user"}</code>	86765.05

Query language: example

```
sum by (instance, mode) (rate(node_cpu[1m]))
```

```
{instance="here.cz:9000",mode="idle"}           0.89222
{instance="here.cz:9000",mode="iowait"}        0.00911
{instance="here.cz:9000",mode="system"}       0.03444
{instance="here.cz:9000",mode="user"}         0.05799

{instance="there.org:9100",mode="idle"}       1.8464
{instance="there.org:9100",mode="iowait"}     0.0217
{instance="there.org:9100",mode="system"}     0.0211
{instance="there.org:9100",mode="user"}      0.107
```

Query language: example



Demo time!