

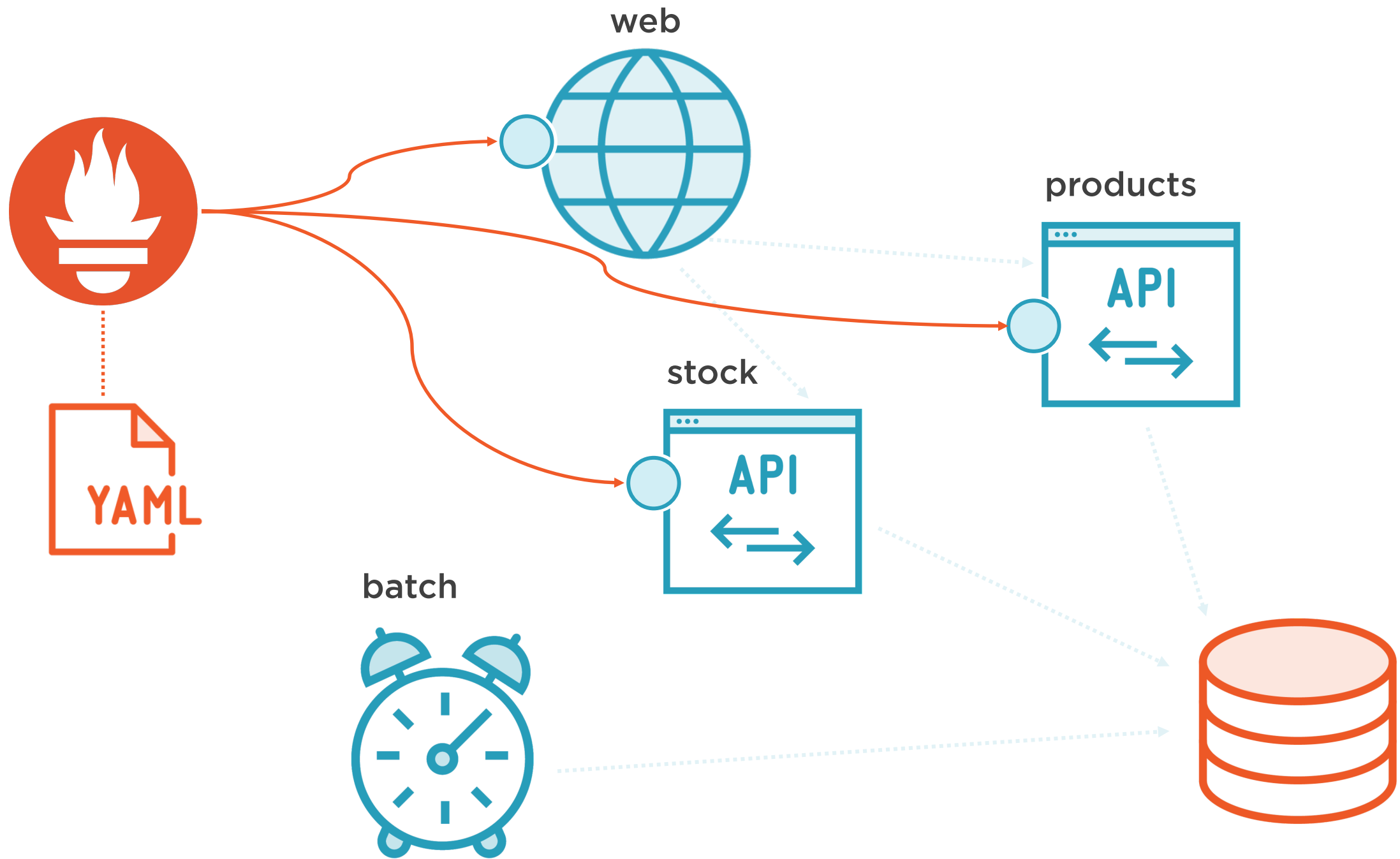
Pushing Metrics from Batch Jobs

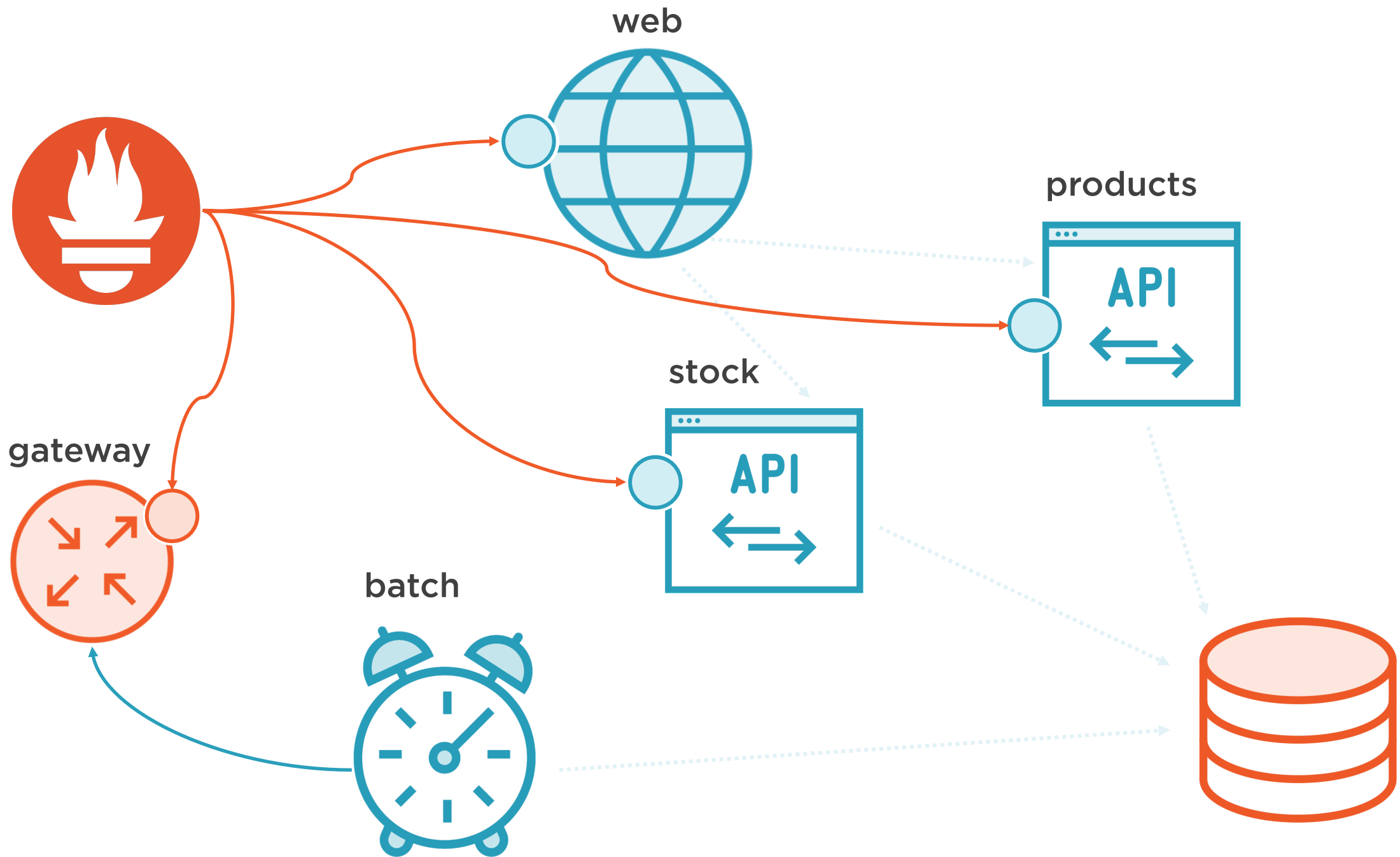


Elton Stoneman

CONSULTANT & TRAINER

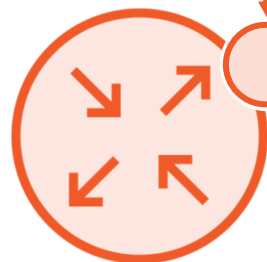
@EltonStoneman | blog.sixeyed.com







gateway



- Multiple instances
- Complicates aggregation
- Convert to server



- Context-specific
- Identify trends
- Use Node Exporter



- Ad-hoc running
- No context
- Minimal metrics

Demo



Using the Pushgateway with Prometheus

- Running the Pushgateway
- Pushing metrics with the HTTP API
- Scraping with Prometheus

Running the Pushgateway

docker-compose.yml

services:

pushgateway:

image: prom/pushgateway:v1.2.0

ports:

- "9091:9091"

networks:

- wb-net

Pushing Metrics with HTTP

metrics.txt

```
# TYPE test_counter Sample counter
```

```
test_counter 20.09
```

```
# TYPE test_gauge gauge
```

```
# HELP test_gauge Sample gauge
```

```
test_gauge{key="value"} 6
```

curl.sh

```
curl -X POST
```

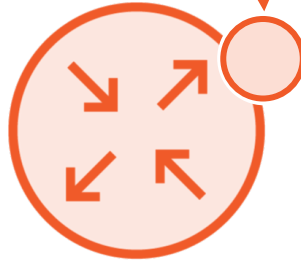
```
'http://localhost:9091/metrics/job/test/instance/test01'
```

```
--header 'Content-Type: text/plain'
```

```
--data-binary '@metrics.txt'
```



gateway



processed_total {instance="b1"} 10

processed_total {instance="b2"} 65

processed_total {instance="b3"} 4

processed_total 79

http_requests_in_progress
3400

process_cpu_seconds_total
443.21

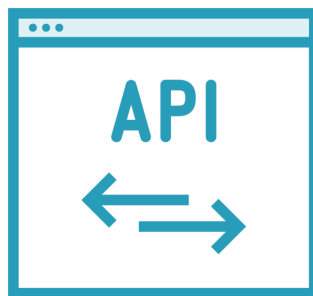
total_memory_bytes
6410859

batch

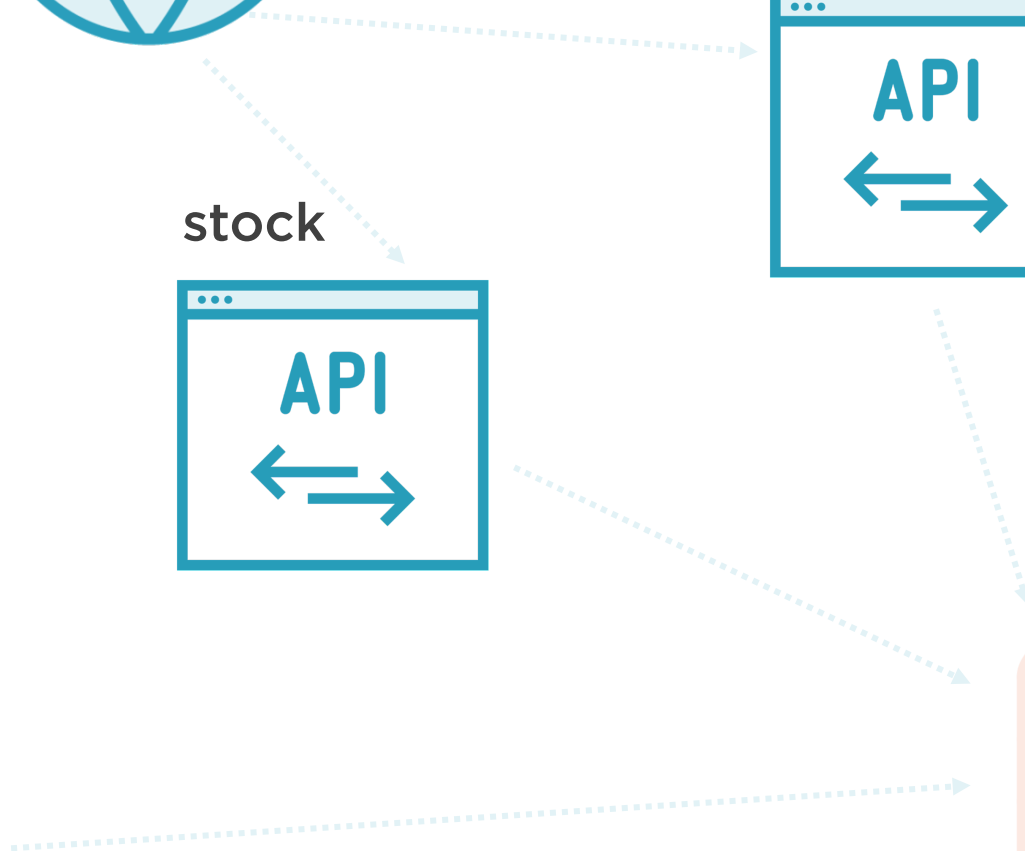
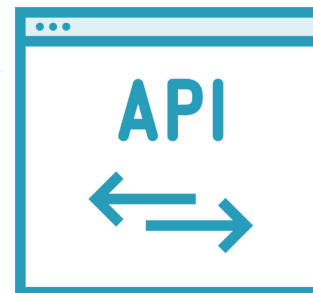


web

stock



products



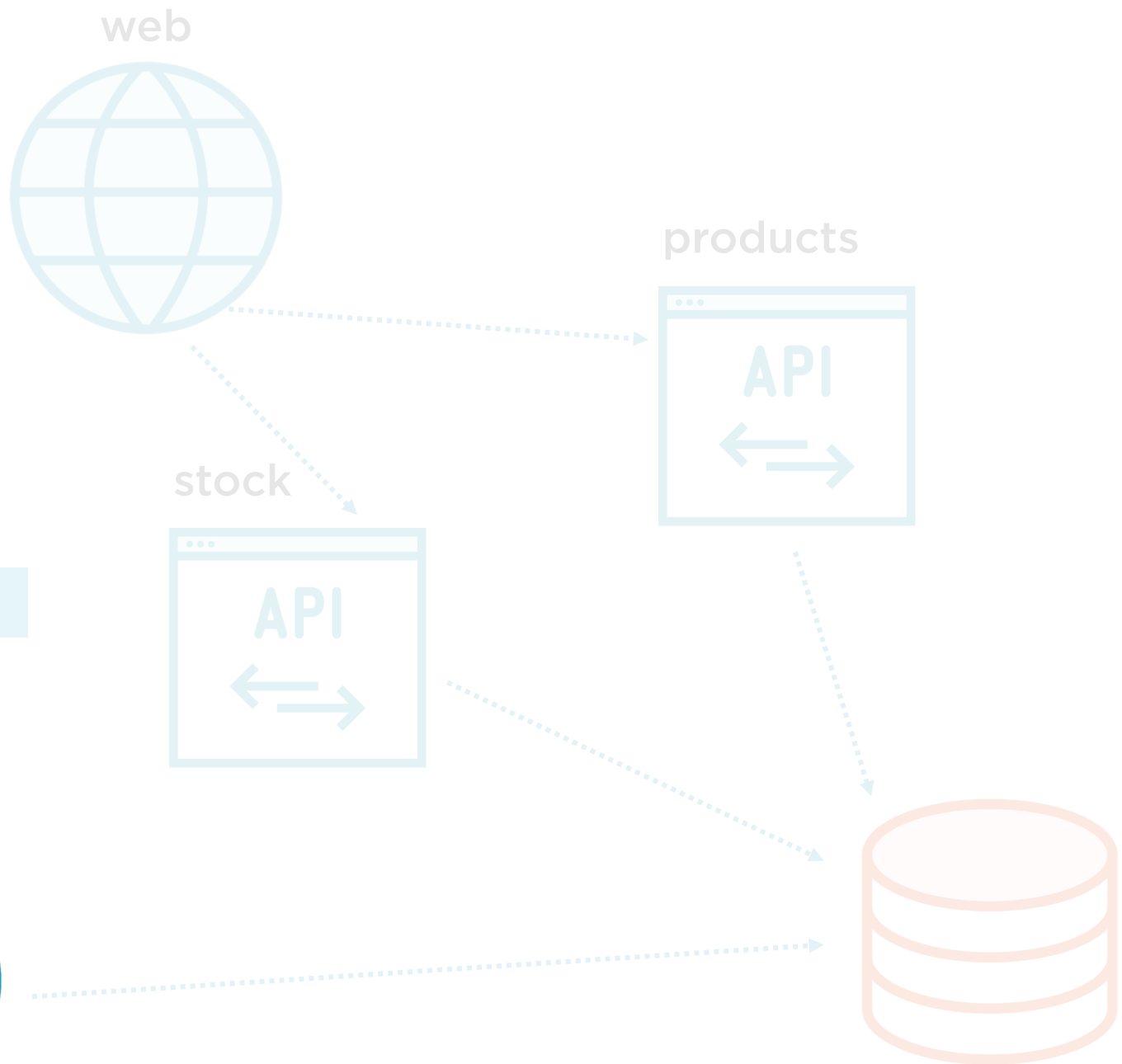
last_success_time
1600174805.15

last_failure_time
1580699045.21

job_duration_seconds
7.33

app_info {version="0.3.0"}

batch





Library

Package reference



Push

Send metrics at exit



Metrics

Record custom values

Declaring Metrics

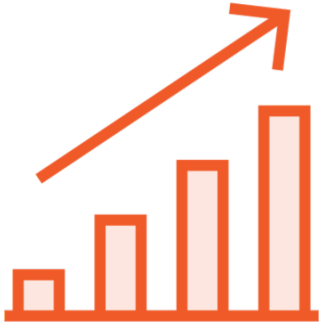
bad.example

```
success = new Gauge()
failure = new Gauge()

if (ok) {
    success.setToCurrentTime()
}
else {
    failure.setToCurrentTime()
}
```

good.example

```
if (ok) {
    success = new Gauge()
    success.setToCurrentTime()
}
else {
    failure = new Gauge()
    failure.setToCurrentTime()
}
```



Counter



Gauge



Summary



Histogram



Insufficient data

Demo



Adding metrics to the pricing batch job

- Using the Node.js client library
- Pushing metrics to the gateway
- Adding the info metric

Demo



Recording metrics for batch processes

- Gauges for success and failure
- Process duration gauge
- Scraping metrics in Prometheus

Referencing the Client Library

package.json

```
{  
  "name": "batch",  
  "dependencies": {  
    "prom-client" : "12.0.0"  
  }  
}
```


Wiring up the Metrics Endpoint

server.js

```
prom.collectDefaultMetrics();

server.get("/metrics", function(req, res, next) {
  res.end(prom.register.metrics());
});
```

Pushing Metrics at Exit

server.js

```
gateway.pushAdd({ jobName: 'bat'}, function(err, resp, body) {  
  if (err) {  
    console.log(err)  
  }  
  process.exit();  
})
```

Setting the Info Metric

server.js

```
var appInfoGauge = new prometheus.Gauge({  
  name: 'app_info',  
  help: 'Application info',  
  labelNames: ['version', 'node_version'],  
});  
appInfoGauge.labels('0.3.0', '10.19.0').set(1);
```

Setting Success and Failure Times

server.js

```
if (err) {  
  var errorGauge = new prometheus.Gauge({  
    name: 'batch_last_error_seconds'  
  });  
  errorGauge.setToCurrentTime(); }  
else {  
  var successGauge = new prometheus.Gauge({  
    name: 'batch_last_success_seconds'  
  });  
  successGauge.setToCurrentTime(); }
```

Setting the Duration

server.js

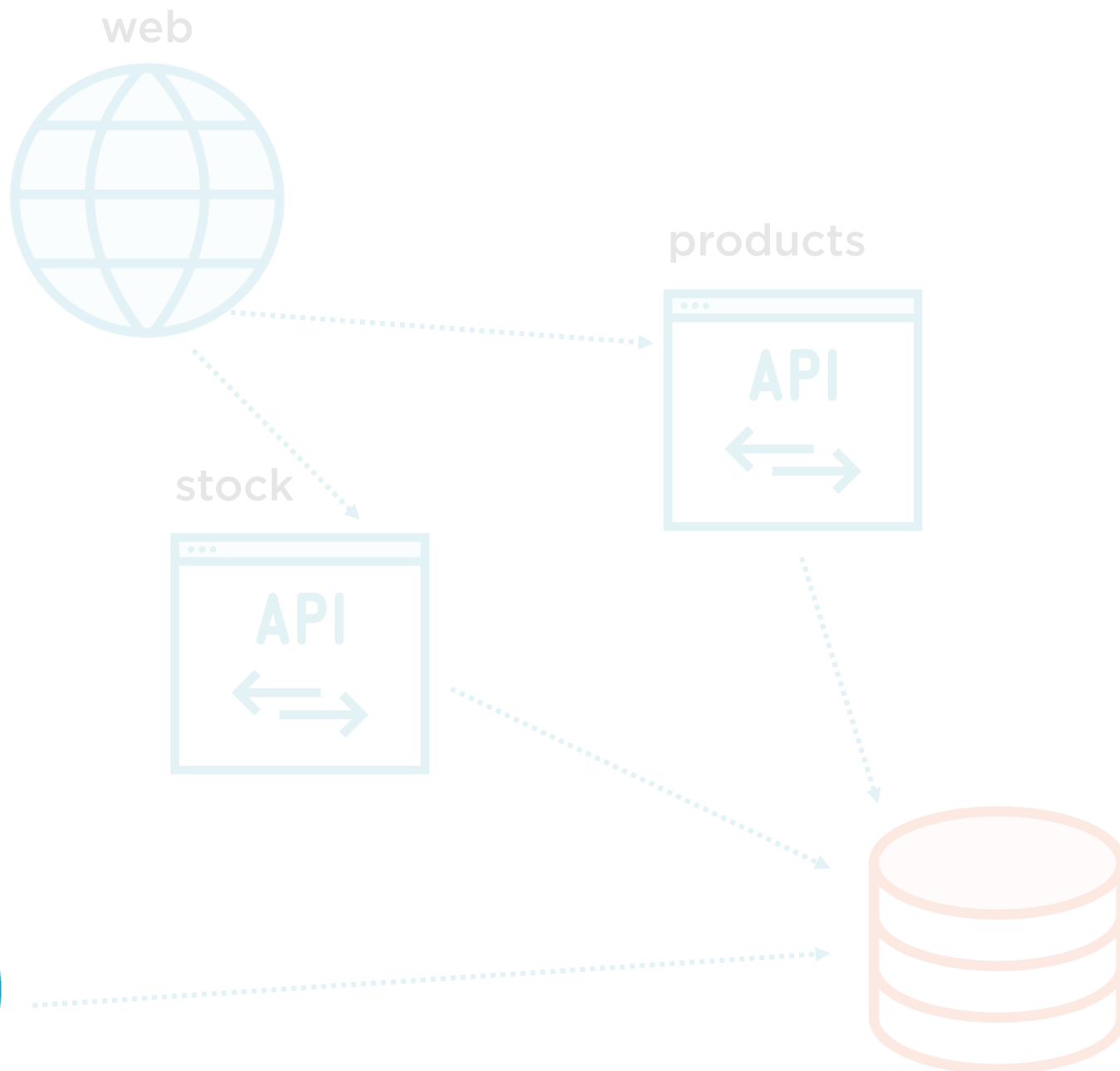
```
var durationGauge = new prometheus.Gauge({  
  name: 'batch_duration_seconds' });  
  
const end = durationGauge.startTimer();  
  
client.connect()  
  
client.query(sql, factor, (err, res) => {  
  end(); //...
```

last_success_time
1600174805.15

last_failure_time
1580699045.21

job_duration_seconds
7.33

batch



Last Success
1 hour ago

Last failure
3 days ago

Last duration
45 seconds

Summary



Metrics for batch processes

- Running the Pushgateway
- Pushing metrics in jobs
- Scraping the Pushgateway

Useful metrics

- Last success & failure timestamp
- Job duration
- Processing stats

Pushgateway alternatives

- Node Exporter
- Converting to server

Up Next:

Scraping Application Metrics with Prometheus
