inuits
OPEN SOURCE INNOVATORS

Julien Pivotto
@roidelapluie

Improved alerting with Prometheus and Alertmanager

November 8th, 2019

PromCon Munich

Important notes!

- This talk contains PromQL.
- This talk contains YAML.
- What you will see was built over time.

Context

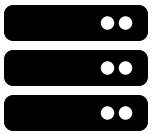
Message Broker

Message Broker in the Belgian healthcare sector

- High visibility
- Sync & Async
- Legacy & New
- Lots of partners
- Multiple customers

Monitoring

Technical



Business

Alerting

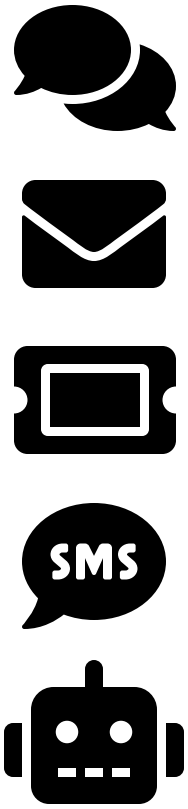
Alerts are not only for incidents.

Some alerts carry business information about ongoing events (good or bad).

Some alerts go outside of our org.

Some alerts are not for humans.

Channels



Time frames

Repeat every 15m, 1h, 4h, 24h, 2d
24x7, 10x5, 12x6, 10x7, never

Legal holidays

15m/1h repeat interval?

Updated annotations & value

Updated graphs

Constraints

- Alertmanager owns the notifications
- Webhook receivers have no logic
- Take decisions at time of alert writing

Challenges

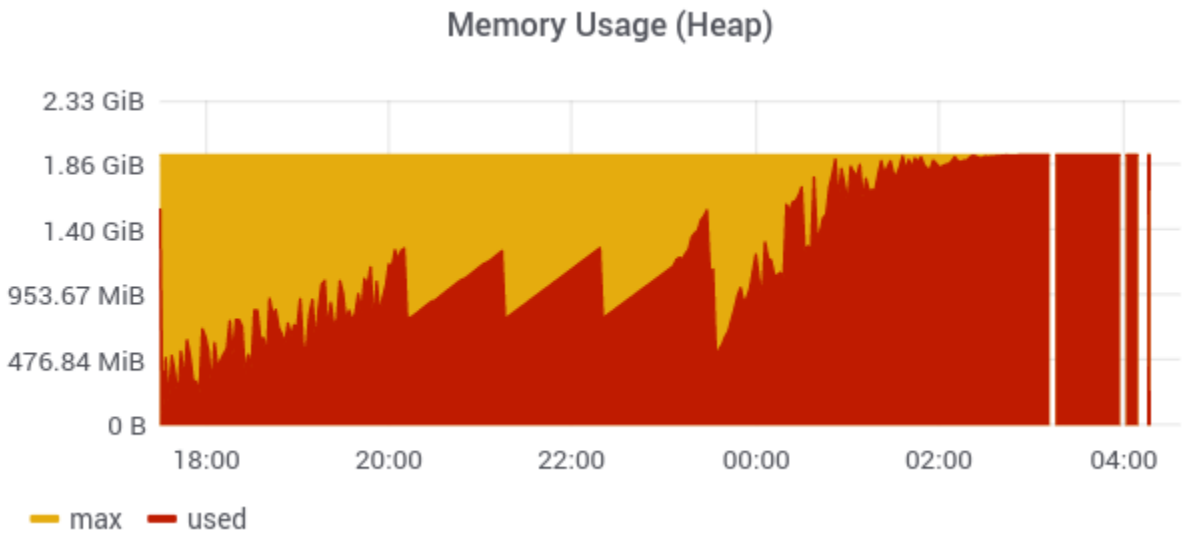
- Avoid Alertmanager reconfigurations
- Safe and easy way to write alerts
- Only send relevant alerts
- Alert on staging environments

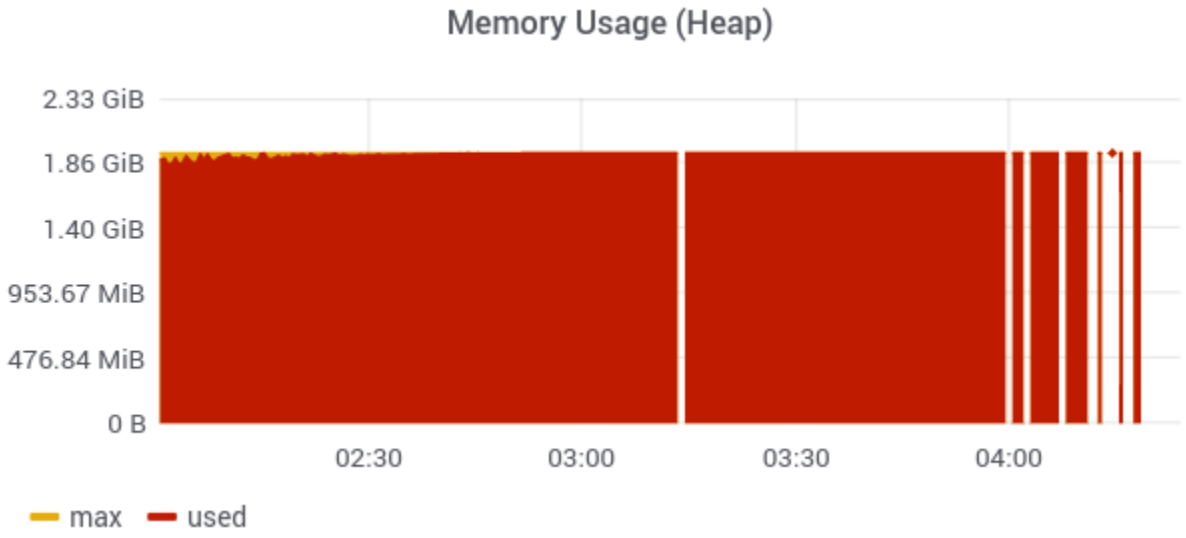
PromQL

Gauges

```
- alert: a target is down
  expr: up == 0
  for: 5m
```

Gauges


Memory Usage (Heap)
2.33 GiB
1.86 GiB
1.40 GiB
953.67 MiB
476.84 MiB
0 B
18:00
20:00
22:00
00:00
02:00
04:00
max
used


Memory Usage (Heap)
2.33 GiB
1.86 GiB
1.40 GiB
953.67 MiB
476.84 MiB
0 B
02:30
03:00
03:30
04:00
max
used

Gauges

Instead of:

```
- alert: a target is down
  expr: up == 0
  for: 5m
```

Do:

```
- alert: a target is down
  expr: avg_over_time(up[5m]) < .9
  for: 5m
```

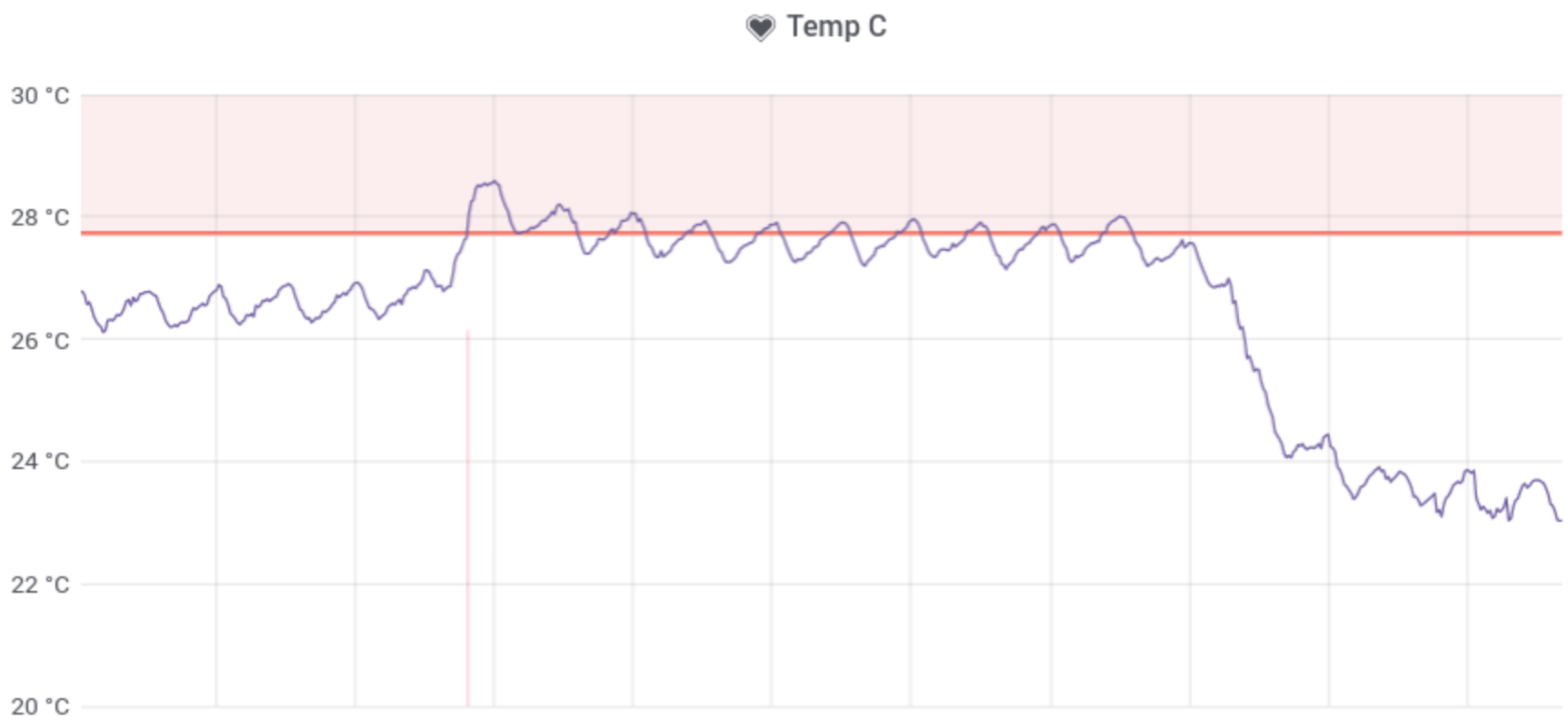
Hysteresis

Alert me if temperature is above 27°C

Hysteresis

```
- alert: temperature is above threshold
  expr: temperature_celcius > 27
  for: 5m
  labels:
    priority: high
```

Hysteresis



Hysteresis

Hysteresis is the dependence of the state of a system on its history.

Hysteresis

```
- alert: temperature is above threshold
  expr: |
    avg_over_time(temperature_celcius[5m])
    > 27
  for: 5m
  labels:
    priority: high
```

alternative: `max_over_time`

5m might be too short

if > 5m: when is it resolved?

Hysteresis

Alert me

- if temperature is above 27°C
- only stop when it gets below 25°C

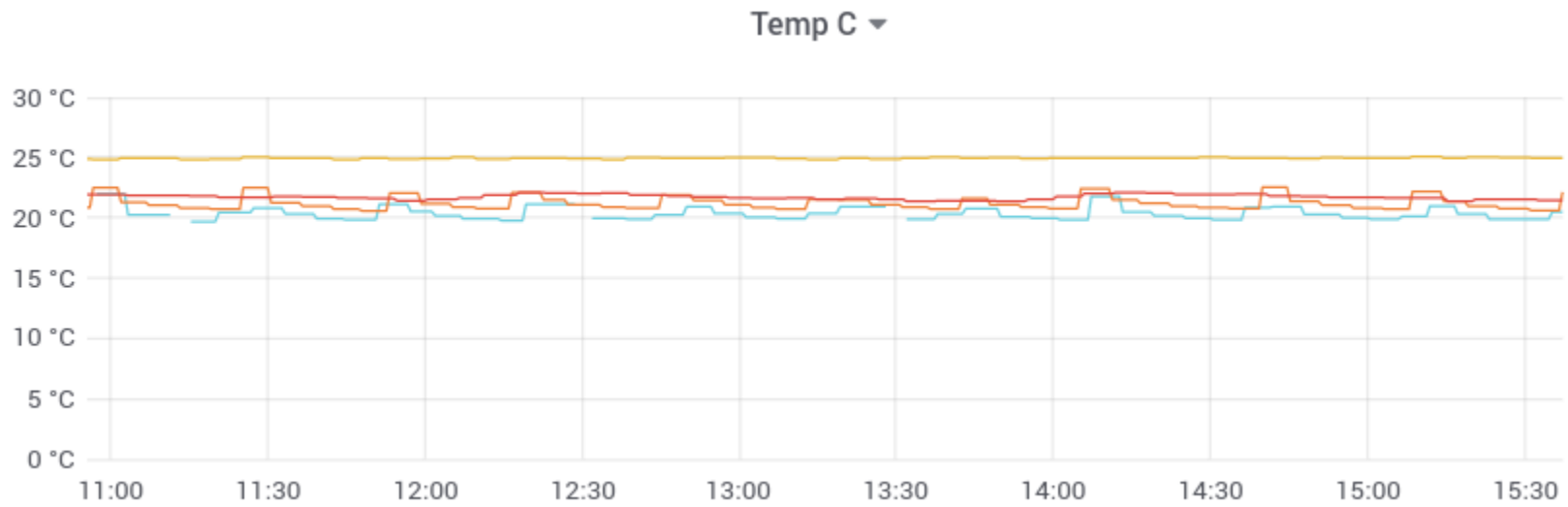
Hysteresis

```
(avg_over_time(temperature_celcius[5m]) > 27)
or (temperature_celcius > 25 and
count without (alertstate, alertname, priority)
  ALERTS{
  alertstate="firing",
  alertname="temperature is above threshold"
})
```

Computed threshold

```
temperature_celcius > 27
```

but...



Computed threshold

```
- record: temperature_threshold_celcius
  expr: |
    27+0*temperature_celcius{
      location=~".*ambiant"
    }
    or 25+0*temperature_celcius
```

Bonus: `temperature_threshold_celcius` can be used in grafana!

Computed threshold

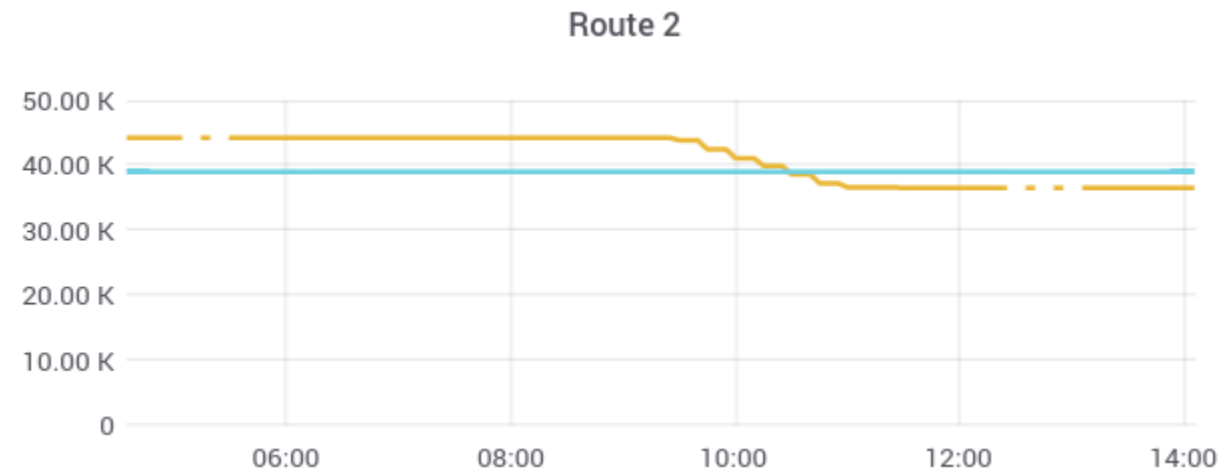
```
- alert: temperature is above threshold
  expr: |
    temperature_celcius >
    temperature_threshold_celcius
```

Note: put threshold & alert
in the same alert group

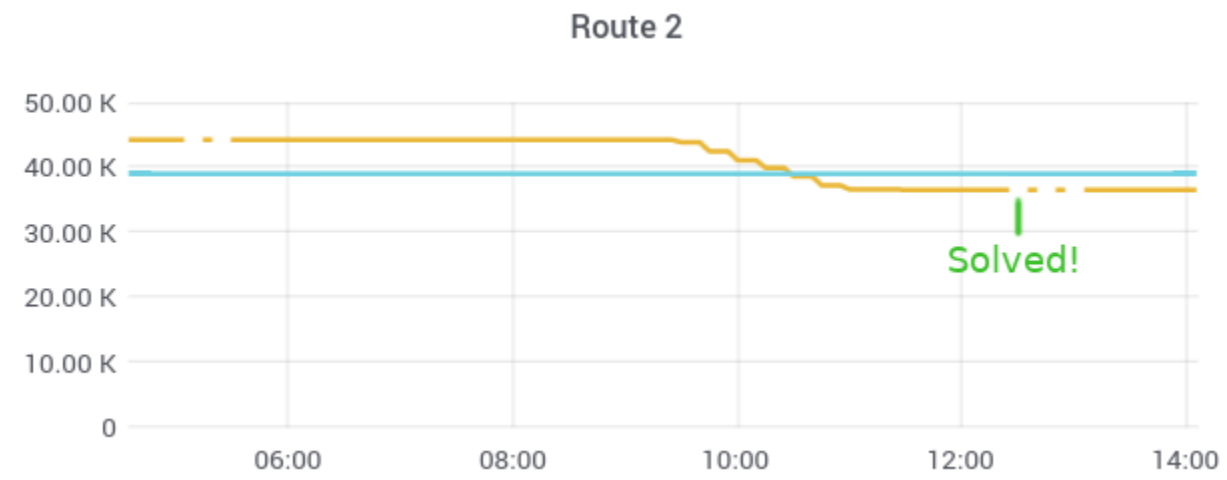
Absence

```
- alert: no more sms
  expr: sms_available < 39000
```

Absence


Route 2
50.00 K
40.00 K
30.00 K
20.00 K
10.00 K
0
06:00
08:00
10:00
12:00
14:00

Absence



No metric = No alert!

Metric is back = New alert!

Absence

```
- record: sms_available_last
  expr: |
    sms_available or
    sms_available_last
- alert: no more sms
  record: sms_available_last < 39000
- alert: no more sms data
  record: absent(sms_available)
  for: 1h
```

Configuration

Recipients

recipients:

name/channel

jpivotto/mail
opsteam/ticket
appteam/message
customer/sms
dc1/jenkins

Receivers

Alertmanager receivers

```
- name: "opsteam/mail"
  email_configs:
  - to: 'ops@inuits.eu'
    send_resolved: yes
    html: "{{ template \"inuits.html.tmpl\" . }}"
    text: "{{ template \"inuits.txt.tmpl\" . }}"
    headers:
        Subject: "{{ template \"title.tmpl\" . }}"
```

Hint: Subject can be a template.

Receivers

Alertmanager receivers

```
- name: "opsteam/mail/noresolved"
  email_configs:
  - to: 'ops@inuits.eu'
    send_resolved: no
    html: "{{ template \"inuits.html.tmpl\" . }}"
    text: "{{ template \"inuits.txt.tmpl\" . }}"
    headers:
        Subject: "{{ template \"title.tmpl\" . }}"
```

Same, but with `send_resolved: no`

Email: CC, BCC

Alertmanager receivers

```
- name: "lotsOfPeople/mail"
  email_configs:
  - to: 'a@inuits.eu,b@inuits.eu,c@inuits.eu'
    headers:
        To: a@inuits.eu
        CC: b@inuits.eu
        Reply-To: support@inuits.eu
```

`c@inuits.eu` is now BCC.

Who gets the alert?

Prometheus alert

```
- alert: Not  enough traffic
  expr: ...
  for: 5m
  labels:
    recipients: customer1/sms,opsteam/ticket
  annotations:
    summary: ...
    resolved_summary: ...
```

Who gets the alert?

Alertmanager routing

```
- receiver: "customer1/sms"
  match_re:
    recipient: "(.*,)?customer1/sms(,.*)?"
  continue: true
  routes: [...]
- receiver: "opsteam/ticket"
  match_re:
    recipient: "(.*,)?opsteam/ticket(,.*)?"
  continue: true
  routes: [...]
```

Resolved

Prometheus alert

```
- alert: Not  enough traffic
  expr: ...
  for: 5m
  labels:
    recipients: customer1/sms,opsteam/ticket
    send_resolved: "no"
```

Resolved

Alertmanager routing

```
- receiver: "customer1/sms"
  match_re:
    recipient: "(.*,)?customer1/sms(,.*)?"
  continue: true
  routes:
  - receiver: customer1/sms/noresolved
    match:
      send_resolved: "no"
```

Repeat interval

Prometheus alert

```
- alert: Not  enough traffic
  expr: ...
  for: 5m
  labels:
    recipients: customer1/sms,opsteam/ticket
    repeat_interval: 1h
```

Repeat interval

Alertmanager routing

```
- receiver: "customer1/sms"
  match_re:
    recipient: "(.*,)?customer1/sms(,.*)?"
  continue: true
  routes:
  - receiver: customer1/sms
    repeat_interval: 1h
    match:
      repeat_interval: 1h
```

Extra configurations

Some channels have specific `group_interval: 0s`.
Some channels always `send_resolved: no`.
Some recipients have aliases (ticket+chat).

Routes tree

Extract of amtool config routes show

```
– {recipient=~"^(?:(.*,)?jpivotto/mail(,.*)?)$"}  continue: true receiver: jpivotto/mail
   ├── {repeat_interval="15m",send_resolved="no"}  receiver: jpivotto/mail/noresolved
   ├── {repeat_interval="15m"}  receiver: jpivotto/mail
   ├── {repeat_interval="30m",send_resolved="no"}  receiver: jpivotto/mail/noresolved
   ├── {repeat_interval="30m"}  receiver: jpivotto/mail
   ├── {repeat_interval="1h",send_resolved="no"}  receiver: jpivotto/mail/noresolved
   ├── {repeat_interval="1h"}  receiver: jpivotto/mail
   ├── {repeat_interval="2h",send_resolved="no"}  receiver: jpivotto/mail/noresolved
   ├── {repeat_interval="2h"}  receiver: jpivotto/mail
   ├── {repeat_interval="4h",send_resolved="no"}  receiver: jpivotto/mail/noresolved
   ├── {repeat_interval="4h"}  receiver: jpivotto/mail
   ├── {repeat_interval="6h",send_resolved="no"}  receiver: jpivotto/mail/noresolved
   ├── {repeat_interval="6h"}  receiver: jpivotto/mail
   ├── {repeat_interval="12h",send_resolved="no"}  receiver: jpivotto/mail/noresolved
   ├── {repeat_interval="12h"}  receiver: jpivotto/mail
   ├── {repeat_interval="24h",send_resolved="no"}  receiver:jpivotto/mail/noresolved
   ├── {repeat_interval="24h"}  receiver: jpivotto/mail
   ├── {send_resolved="no"}  receiver: jpivotto/mail/noresolved
   └── {repeat_interval=""}  receiver: jpivotto/mail
```

How do we achieve it?

Config Management!

Our input

```
receivers:
  customer:
    email:
      to: [customer@example.com]
      cc: [service-management@inuits.eu]
      bcc: [ops@inuits.eu]
    sms: [+1234567890, +2345678901]
    chat:
      room: "#customer"
```

Configuration management

- Script that is deployed with AM
- Knows all the recipients
- Will validate alerts yaml
  - promtool
  - mandatory labels
  - validate `receivers` label
  - validate `repeat_interval` label

Not possible to write alerts that go nowhere by accident.

Time frame

Time frame

Prometheus alert

```
- alert: a target is down
  expr: up == 0
  for: 5m
  labels:
    recipients: customer1/sms,opsteam/ticket
    time_window: 13x5
```

Timezone

```
- record: daily_saving_time_belgium
  expr: |
    (vector(0) and (month() < 3 or month() > 10))
    or
    (vector(1) and (month() > 3 and month() < 10))
    or
    (
     (
      (month() %2 and (day_of_month() - day_of_week()
      > (30 + +month() % 2 - 7)) and day_of_week() > 0)
     or
      -1*month()%2+1 and (day_of_month() -
      day_of_week() <= (30 + month() % 2 - 7))
     )
    )
    or
    (vector(1) and ((month()==10 and hour() < 1) or (month()==3 and hour() > 0
    or
    vector(0)
- record: belgium_localtime
  expr: |
     time() + 3600 + 3600 * daily_saving_time_belgium
```

Belgian hour

```
hour(belgium_localtime)
```

`hour()` and other time-functions can take a timestamp as argument.

Holidays

```
- record: public_holiday
  expr: |
     vector(1) and
     day_of_month(belgium_localtime) == 25
     and month(belgium_localtime) == 12
  labels:
     name: Xmas
```

Easter

```
groups:
- name: Easter Meeus/Jones/Butcher Algorithm
  interval: 60s
  rules:
    - record: easter_y
      expr: year(belgium_localtime)
    - record: easter_a
      expr: easter_y % 19
    - record: easter_b
      expr: floor(easter_y / 100)
    - record: easter_c
      expr: easter_y % 100
    - record: easter_d
      expr: floor(easter_b / 4)
    - record: easter_e
      expr: easter_b % 4
    - record: easter_f
      expr: floor((easter_b +8 ) / 25)
    - record: easter_g
      expr: floor((easter_b - easter_f + 1 ) / 3)
    - record: easter_h
      expr: (19*easter_a + easter_b - easter_d - easter_g + 15 ) % 30
    - record: easter_i
      expr: floor(easter_c/4)
    - record: easter_k
      expr: easter_c%4
    - record: easter_l
      expr: (32 + 2*easter_e + 2*easter_i - easter_h - easter_k) % 7
    - record: easter_m
      expr: floor((easter_a + 11*easter_h + 22*easter_l) / 451)
    - record: easter_month
      expr: floor((easter_h + easter_l - 7*easter_m + 114) / 31)
    - record: easter_day
      expr: ((easter_h + easter_l - 7*easter_m + 114) %31) + 1
```

Easter

```
- record: public_holiday
  expr: |
    vector(1) and
    day_of_month(belgium_localtime-86400) == easter_day
    and month(belgium_localtime-86400) == easter_month
  labels:
     name: Easter Monday
- record: public_holiday
  expr: |
     vector(1) and
     day_of_month(belgium_localtime-40*86400) == easter_day
     and month(belgium_localtime-40*86400) == easter_month
  labels:
     name: Feast of the Ascension
```

Business hour

```
- record: business_day
  expr: |
    vector(1) and day_of_week(belgium_localtime) > 0
    and day_of_week(belgium_localtime) < 6
    unless count(public_holiday)
- record: belgium_hour
  expr: |
    hour(belgium_localtime)
- record: business_hour
  expr: |
    vector(1) and belgium_hour >= 8 < 18
    and business_day
```

Extended business hours

```
- record: extended_business_hour
  expr: |
     (vector(1) and belgium_hour >= 7 < 20
     and business_day)
- record: extended_business_hour_sat
  expr: |
     extended_business_hour
     or (vector(1) and belgium_hour >= 7 < 14
     and day_of_week(belgium_localtime) == 6
     unless count(public_holiday))
```

Thresholds depending on time

```
(sum(rate(http_requests_total{code=~"5.."}[5m])) by (vhost)
> 10 and on () business_hour)
or
(sum(rate(http_requests_total{code=~"5.."}[5m])) by (vhost) > 1
and sum(rate(http_requests_total{code=~"2.."}[5m])) by (vhost) < 1)
```

```
and on () business_hour
```

Day and night

```
- record: daylight
  expr: |
     vector(1) and belgium_hour >= 8 < 18
- record: extended_daylight
  expr: |
     vector(1) and belgium_hour >= 7 < 20
```

Alerts

```
- alert: Time Window - Night
  expr: absent(daylight)
  labels:
      recipient: none
- alert: Time Window - OBH
  expr: absent(business_hour)
  labels:
      recipient: none
- alert: Time Window - Extended Night
  expr: absent(extended_daylight)
  labels:
      recipient: none
```

Alerts

```
- alert: Time Window - Extended OBH with Saturday
  expr: absent(extended_business_hour_sat)
  labels:
      recipient: none
- alert: Time Window - Extended OBH
  expr: absent(extended_business_hour)
  labels:
      recipient: none
```

Alerts

At this point, we will have "meaningless" alerts at night and during business holidays.

Inhibition

Inhibition is a concept of suppressing notifications for certain alerts if certain other alerts are already firing.

Inhibition

Alertmanager inhibition

```
inhibit_rules:
- source_match:
    alertname: "Time Window - Night"
  target_match:
    time_window: 10x7
- source_match:
    alertname: "Time Window - OBH"
  target_match:
    time_window: 10x5
```

Inhibition

Alertmanager inhibition

```
- source_match:
    alertname: "Time Window - Extended Night"
  target_match:
    time_window: 13x7
- source_match:
    alertname: "Time Window - Extended OBH"
  target_match:
    time_window: 13x5
- source_match:
    alertname: "Time Window - Extended OBH with Saturday"
  target_match:
    time_window: 13x6
```

Alerts Relabeling

Per env recipients

Prometheus alert

```
- alert: a target is down
  expr: up == 0
  for: 5m
  labels:
    recipients_prod: customer1/sms,opsteam/ticket
    time_window_prod: 24x7
    recipients: opsteam/chat
    time_window: 8x5
```

Per env recipients

Prometheus config

```
alerting:
  alert_relabel_configs:
  - source_labels: [time_window_prod,env]
    regex: "(.+);prod"
    target_label: time_window
    replacement: '$1'
  - source_labels: [time_window_dev,env]
    regex: "(.+);dev"
    target_label: time_window
    replacement: '$1'
```

Repeat for other env, other labels.

Drop alert

Prometheus config

```
alerting:
  alert_relabel_configs:
  - source_labels: [time_window]
    regex: "never"
    action: drop
```

Be careful about the order (`time_window` can be mutated by relabeling).

Conclusion

Conclusion

- Alert-Writing experience is great with this
- Prometheus and PromQL can do a lot
- Config management fills the "gaps"
- With some effort, we have everything we wanted

Bonus

```
- name: normal rate
  interval: 120s
  rules:
    - record: request_rate_history
      expr: |
        sum(rate(http_requests_total[5m])) by (env)
      labels:
        when: 0w
```

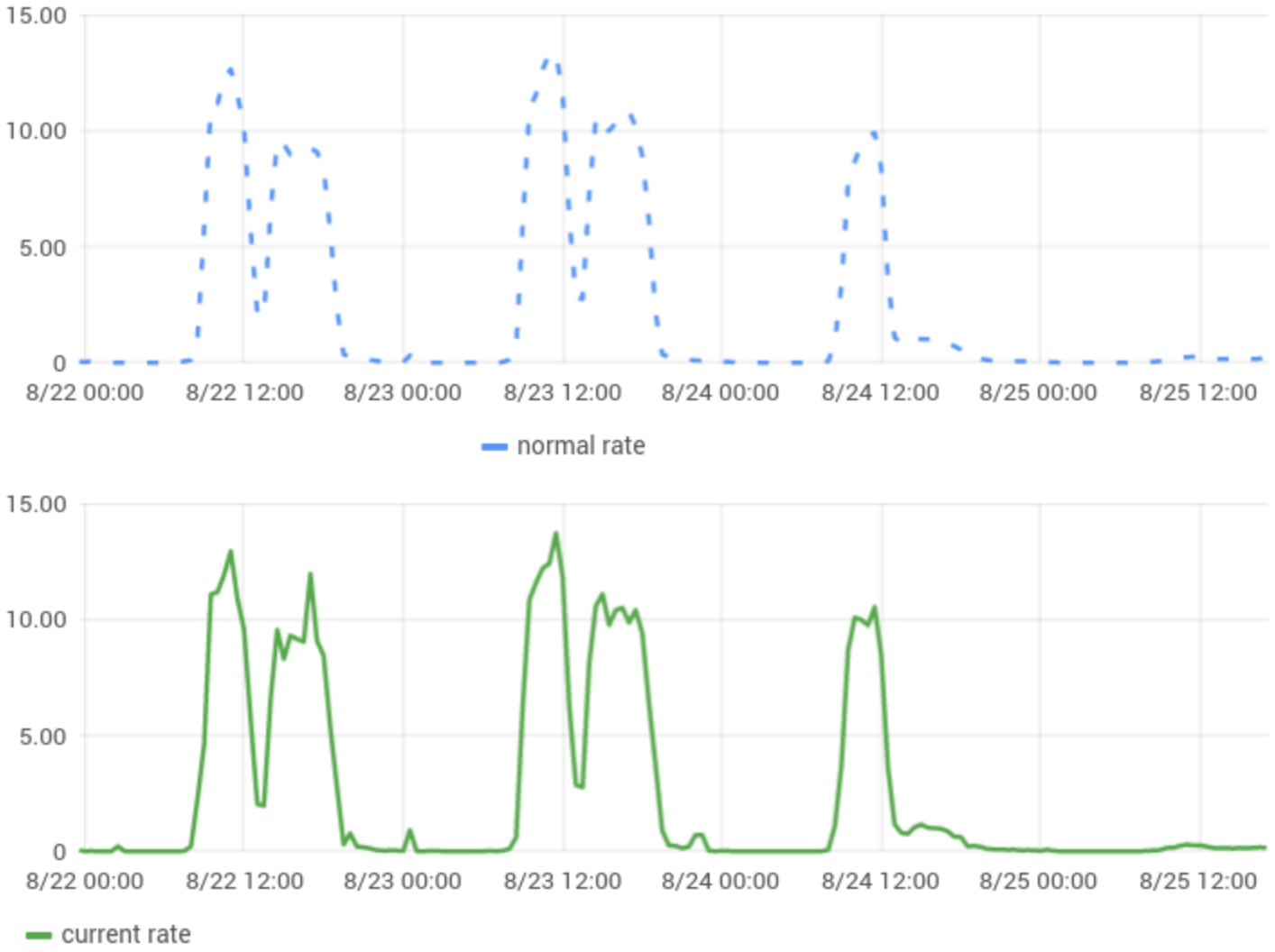
Bonus

```
- record: request_rate_history
  expr: |
    (
        sum(rate(http_requests_total[5m] offset 168h)) by (env)
        and on () (
            daily_saving_time_belgium offset 1w
            == daily_saving_time_belgium)
    ) or
    (
        sum(rate(http_requests_total[5m] offset 167h)) by (env)
        and on () (
            daily_saving_time_belgium offset 1w
            < daily_saving_time_belgium)
    ) or
    (
        sum(rate(http_requests_total[5m] offset 169h)) by (env)
        and on () (
            daily_saving_time_belgium offset 1w
            > daily_saving_time_belgium)
    )
  labels:
    when: 1w
```

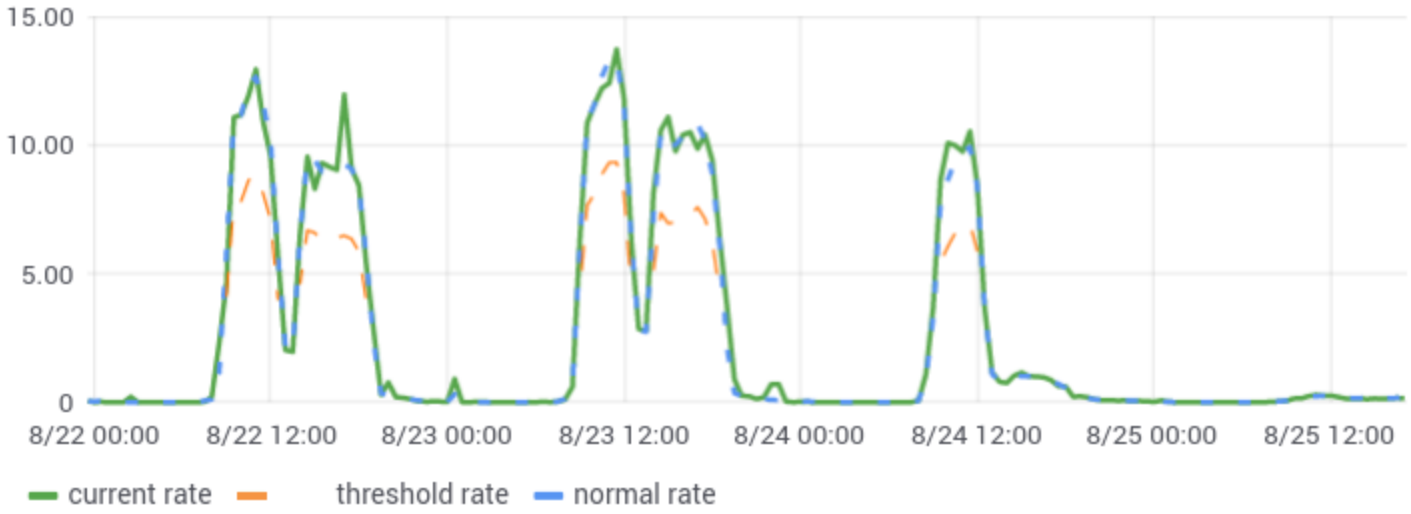
Bonus

```
- record: request_rate_normal
  expr: |
    max(bottomk(1,
      topk(4, request_rate_history) by(env)
    ) by(env)) by(env)
```

Bonus


15.00
10.00
5.00
0
8/22 00:00
8/22 12:00
8/23 00:00
8/23 12:00
8/24 00:00
8/24 12:00
8/25 00:00
8/25 12:00
normal rate
15.00
10.00
5.00
0
8/22 00:00
8/22 12:00
8/23 00:00
8/23 12:00
8/24 00:00
8/24 12:00
8/25 00:00
8/25 12:00
current rate

Bonus


15.00
10.00
5.00
0
8/22 00:00
8/22 12:00
8/23 00:00
8/23 12:00
8/24 00:00
8/24 12:00
8/25 00:00
8/25 12:00
current rate
threshold rate
normal rate

Bonus

Really hope we will get rid of DST in 2021!

inuits
OPEN SOURCE INNOVATORS

Essensteenweg 31
2930 Brasschaat
Belgium

Julien Pivotto
@roidelapluie
roidelapluie@inuits.eu

Contact:

info@inuits.eu
+32-3-8082105