

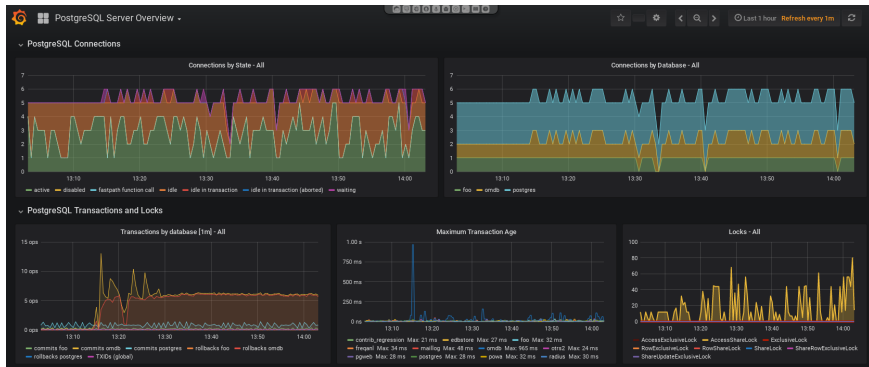


# PostgreSQL-Monitoring mit Prometheus und Grafana

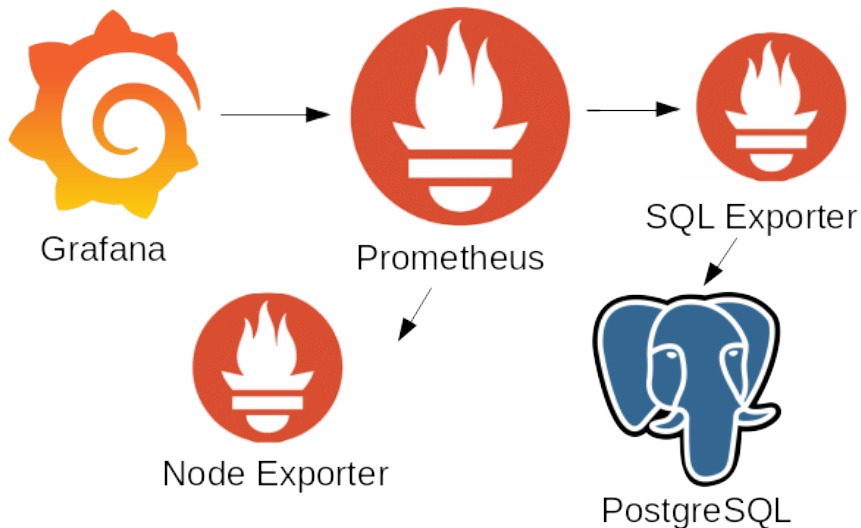
Christoph Berg

Berlin, 13. April 2018

- ▶ PostgreSQL: Datenbanksystem
- ▶ Monitoring:
  - ▶ Überwachung (*Wie voll ist die Platte*)
  - ▶ Performance-Analyse (*Wie voll war die Platte die Nacht über*)
- ▶ Prometheus: Timeseries-Datenbank, Exporter zum Datensammeln
- ▶ Grafana: Dashboards für Prometheus-Graphen



# Workflow



```
# SELECT datname, xact_commit, xact_rollback  
FROM pg_stat_database  
WHERE datname !~ 'template(0|1)'  
ORDER BY datname;
```

datname	xact_commit	xact_rollback
foo	8881	0
omdb	29149	18926
postgres	43780	1754

(3 Zeilen)

- ▶ aktuelle Werte
- ▶ keine historischen Daten

- ▶ Prometheus-Exporter für SQL-Queries, Ausgabe JSON-artig
- ▶ [https://github.com/credativ/sql\\_exporter](https://github.com/credativ/sql_exporter)
- ▶ <http://localhost:9237/metrics>

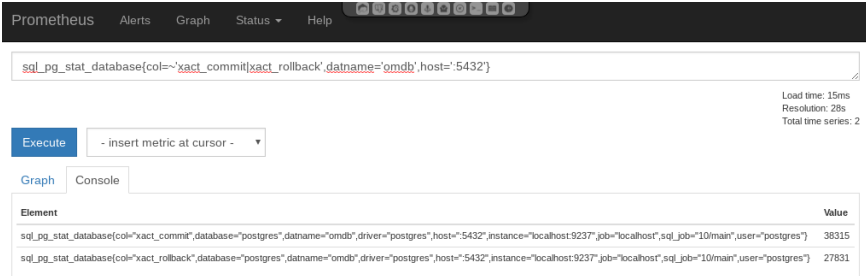
```
sql_pg_stat_database{col="xact_commit",  
  datname="omdb",  
  host=":5432",  
  sql_job="10/main",  
  user="postgres"} 29149
```

```
sql_pg_stat_database{col="xact_rollback",  
  datname="omdb",  
  host=":5432",  
  sql_job="10/main",  
  user="postgres"} 18926
```

- ▶ Timeseries-Datenbank
- ▶ pollt *Exporter*, z.B. alle 30s
- ▶ Abfragesprache mit Statistik-Funktionen (sum, rate)
- ▶ Ausgabe JSON, Rendering im Browser



- ▶ `sql_pg_stat_database{col=~'xact_commit|xact_rollback', datname='omdb', host=':5432'}`



The screenshot shows the Prometheus web interface. At the top, there are navigation links for Prometheus, Alerts, Graph, Status, and Help. Below this is a search bar containing the query: `sql_pg_stat_database{col=~'xact_commit|xact_rollback', datname='omdb', host=':5432'}`. To the right of the search bar, performance metrics are displayed: Load time: 15ms, Resolution: 28s, and Total time series: 2. Below the search bar is an 'Execute' button and a dropdown menu with the text '- insert metric at cursor -'. There are two tabs: 'Graph' and 'Console'. The 'Console' tab is active, showing a table with two columns: 'Element' and 'Value'. The table contains two rows of query results.

Element	Value
<code>sql_pg_stat_database{col="xact_commit", database="postgres", datname="omdb", driver="postgres", host=":5432", instance="localhost:9237", job="localhost", sql_job="10/main", user="postgres"}</code>	38315
<code>sql_pg_stat_database{col="xact_rollback", database="postgres", datname="omdb", driver="postgres", host=":5432", instance="localhost:9237", job="localhost", sql_job="10/main", user="postgres"}</code>	27831

# Prometheus: Graph



Prometheus Alerts Graph Status Help

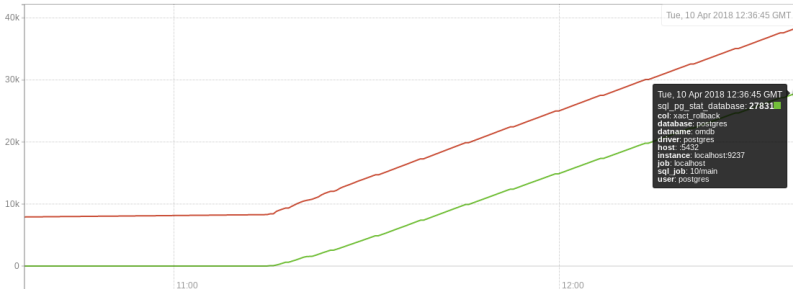
`sql_pg_stat_database(col=~'xact_commit|xact_rollback',datname='omdb',host=':5432')`

Load time: 15ms  
Resolution: 28s  
Total time series: 2

Execute - insert metric at cursor -

Graph Console

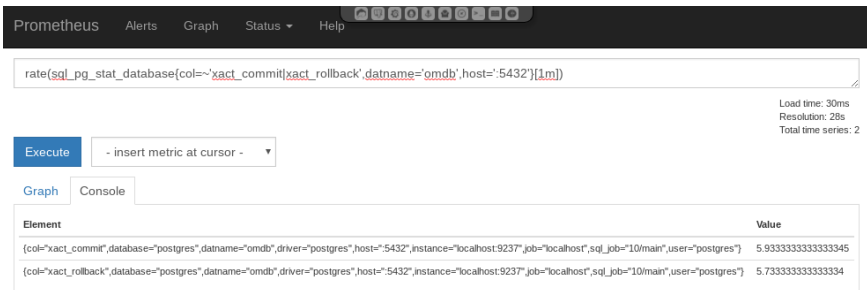
2h Until Res. (s) stacked



Tue, 10 Apr 2018 12:36:45 GMT  
sql\_pg\_stat\_database: 27831  
col: xact\_rollback  
database: postgres  
datname: omdb  
driver: postgres  
host: :5432  
instance: localhost:9237  
job: localhost  
sql\_job: 10/main  
user: postgres

■ sql\_pg\_stat\_database(col=~'xact\_rollback',database='postgres',datname='omdb',driver='postgres',host=':5432',instance='localhost:9237',job='localhost',sql\_job='10/main',user='postgres')  
■ sql\_pg\_stat\_database(col=~'xact\_commit',database='postgres',datname='omdb',driver='postgres',host=':5432',instance='localhost:9237',job='localhost',sql\_job='10/main',user='postgres')

- ▶ `rate(sql_pg_stat_database{col=~'xact_commit|xact_rollback', datname='omdb',host=':5432'})[1m]`
- ▶ Rate pro Sekunde, gemittelt über 1 min-Intervall



The screenshot shows the Prometheus web interface. At the top, there are navigation tabs: Prometheus, Alerts, Graph, Status, and Help. Below the navigation is a search bar containing the query: `rate(sql_pg_stat_database{col=~'xact_commit|xact_rollback',datname='omdb',host=':5432'})[1m]`. To the right of the search bar, performance metrics are displayed: Load time: 30ms, Resolution: 28s, and Total time series: 2. Below the search bar is an 'Execute' button and a dropdown menu with the text '- insert metric at cursor -'. Below that are two tabs: 'Graph' and 'Console'. The 'Console' tab is active, showing a table with two rows of results.

Element	Value
<code>{col="xact_commit",database="postgres",datname="omdb",driver="postgres",host=":5432",instance="localhost:9237",job="localhost",sql_job="10/main",user="postgres"}</code>	5.9333333333333345
<code>{col="xact_rollback",database="postgres",datname="omdb",driver="postgres",host=":5432",instance="localhost:9237",job="localhost",sql_job="10/main",user="postgres"}</code>	5.733333333333334

# Prometheus: Rate-Graph

Prometheus Alerts Graph Status Help

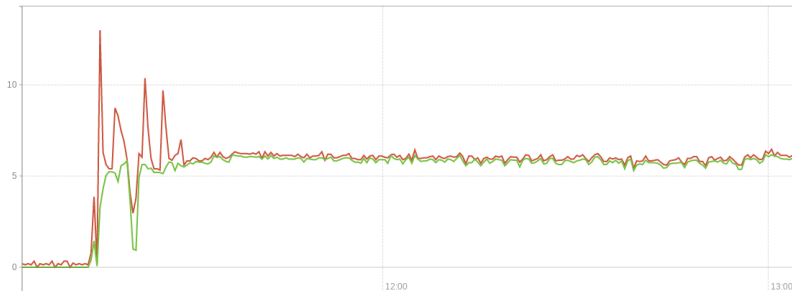
rate(sql\_pg\_stat\_database[col=~'xact\_commit|xact\_rollback',dbname='omdb',host='5432'])[1m]

Load time: 40ms  
Resolution: 28s  
Total time series: 2

Execute - insert metric at cursor -

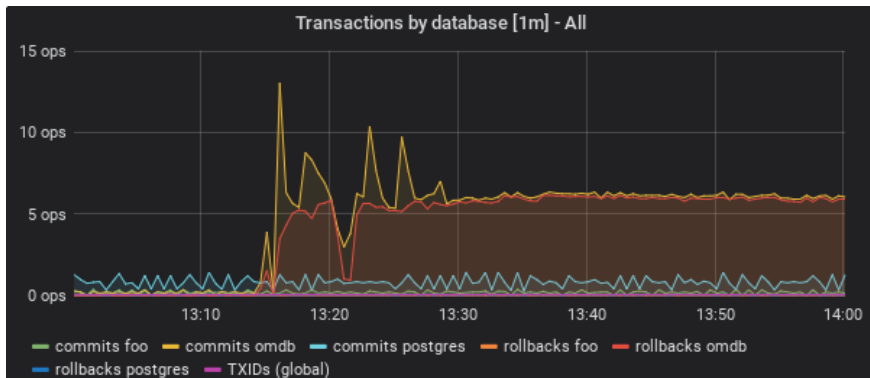
Graph Console

- 2h + << Until >> Res. (s) stacked



■ (col="xact\_rollback",database="postgres",dbname="omdb",driver="postgres",host="5432",instance="localhost:9237",job="localhost",sql\_job="10/main",user="postgres")  
■ (col="xact\_commit",database="postgres",dbname="omdb",driver="postgres",host="5432",instance="localhost:9237",job="localhost",sql\_job="10/main",user="postgres")

- ▶ Grafana zeigt Prometheus-Daten in Panels an
- ▶ umfangreich konfigurierbar
- ▶ Rendering im Browser




# Grafana: Dashboard




# Live Demo


► elephant-shed.io

PostgreSQL Appliance Dashboard 


Logged in as cbe - Logout




Home




DB Administration




Monitoring




Metrics




Log Reports




Backup




System Management



Shell



Documentation



Support

PostgreSQL Cluster

Cluster	Port	Data directory	Archiving	Full Backup	Incr Backup
8.2/main	5482	/var/lib/postgresql/8.2/main	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>





- ▶ [www.omdb.org](http://www.omdb.org)
- ▶ <https://github.com/credativ/omdb-postgresql>

```
SELECT page, count(*), sum(runtime) AS runtime
FROM access_log GROUP BY 1;
```

page	count	runtime
language	6	0.607174396514892
job	18	1.88291954994202
category	209	7.4515106678009
character	163	8.79396939277649
movie	8011	481.059208393097
country	10	0.430700302124024
person	12002	263.316899299622
date	5095	85.0427992343903

(8 Zeilen)

- ▶ /etc/prometheus-sql-exporter/omdb.yml
- ▶ update-prometheus-sql-exporter-config.service (Elephant Shed 1.2)

```
- name: "omdb"
  help: "Web hits in our application"
  scope: database
  database: omdb
  labels:
    - "page"
  values:
    - "count"
    - "runtime"
  query: >-
    SELECT page, count(*), sum(runtime) AS runtime
    FROM access_log GROUP BY 1;
```

▶ `rate(sql_omdb{host=":5432",col="count"}[1m])`

