

Artem Demchenkov

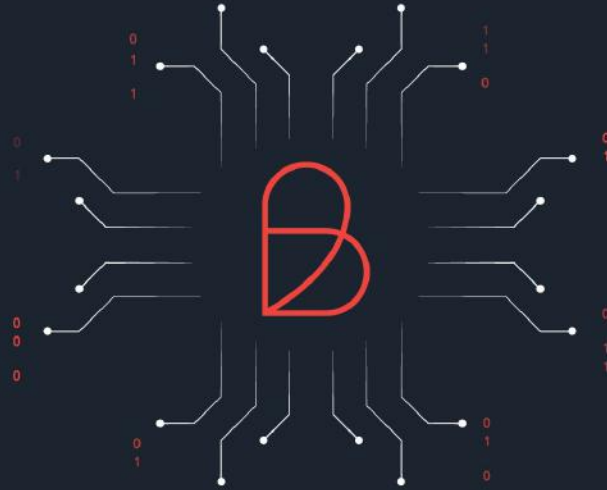


Illustration © Caterina Carraro/Billie

## Smart Data-driven Alerts (SDdA) and How They Make Life Easier



# Artem Demchenkov

- CTO & Co-Founder at Billie
- [ardemchenkov@gmail.com](mailto:ardemchenkov@gmail.com)
- <https://www.linkedin.com/in/artem-demchenkov-76b69934/>



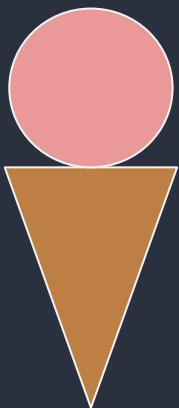


# Billie

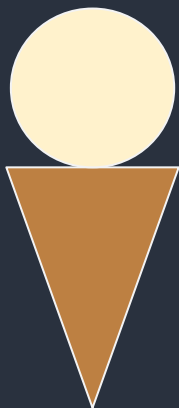
We are an innovative financial platform, focused on working capital financing.

Some numbers:

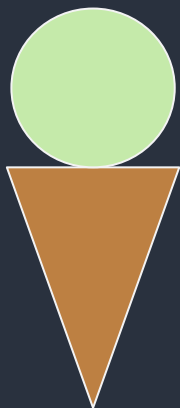
- Start: 01.12.2016
- Closed Beta: 01.05.2017
- Public Launch: 01.06.2017
- Team: 45
- Engineering Crew: 22



**Strawberry**



**Vanilla**



**Mint**



**Mango**



**Lime**



**Chili-Curry**

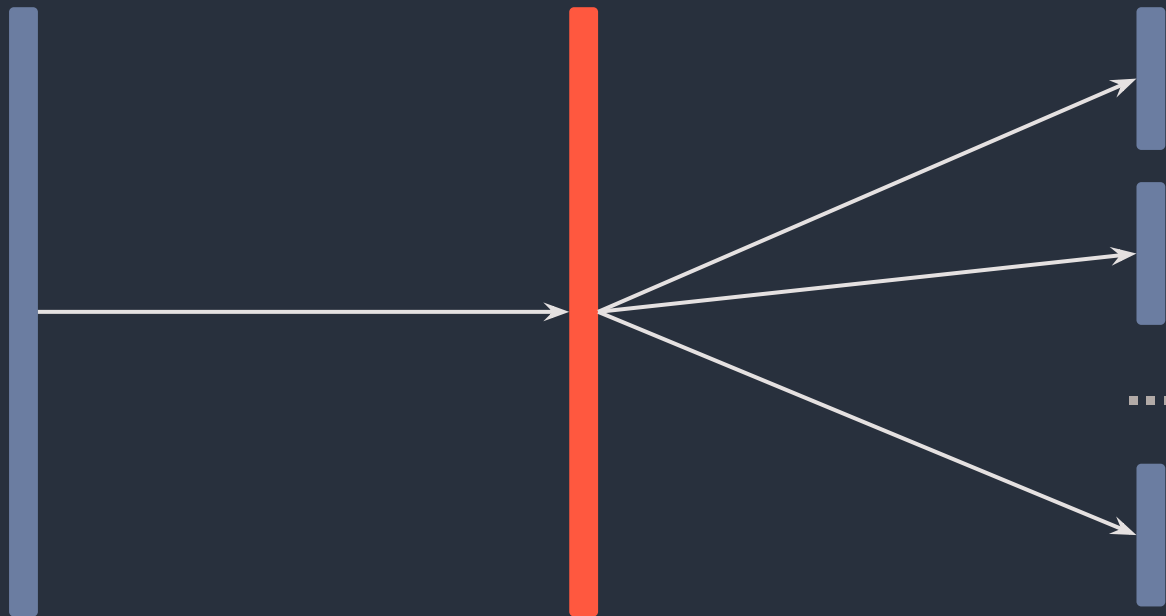


# Standard data-flow

Source of data

Storage

Consumers





# Why could possibly make data wrong?

- Bugs
- Out-of-sync versions of code
- Insufficient validation rules
- Manual intrusion into the database

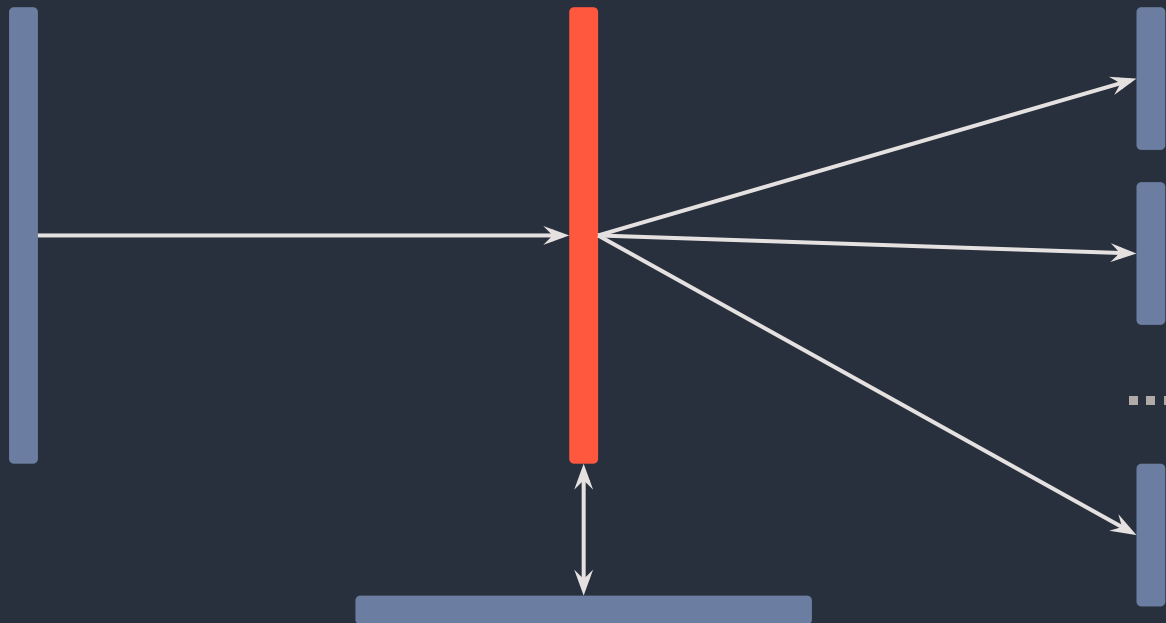


# Monitoring of Data as a Part of Standard Data-flow

Source of data

Storage

Consumers



Monitoring



# Why is it important to monitor data?

- Consistent INSERTs/UPDATEs
- Reliable SELECTs
- Sharing of data between services
- Predictable reports
- Real-time awareness





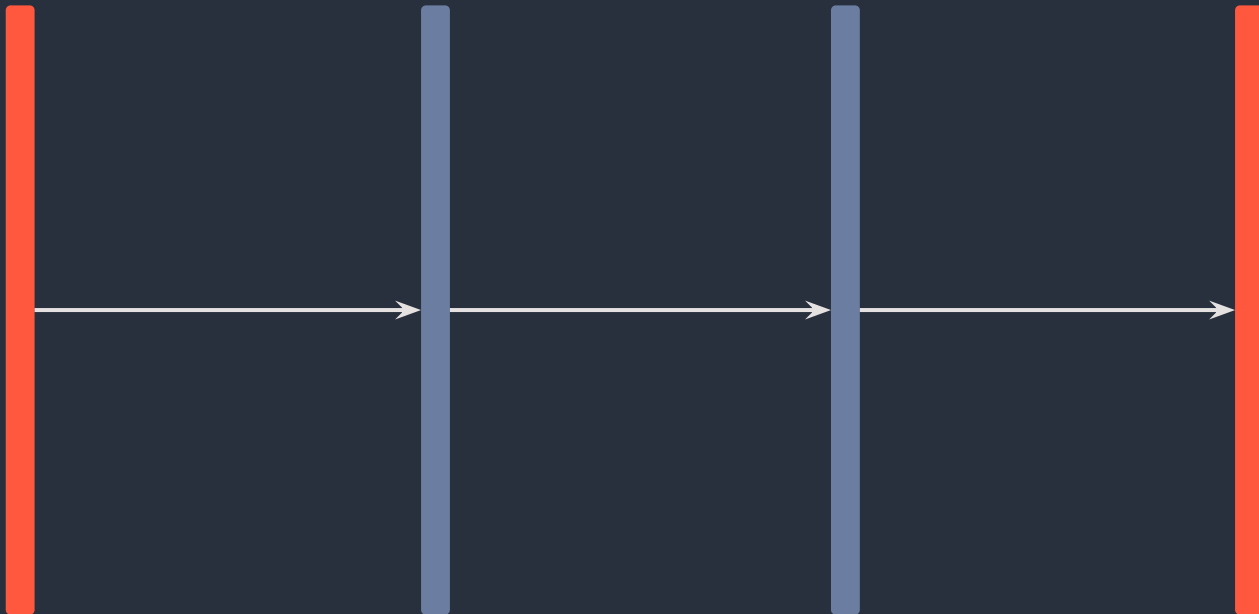
# Principal set up

Data source

Listener

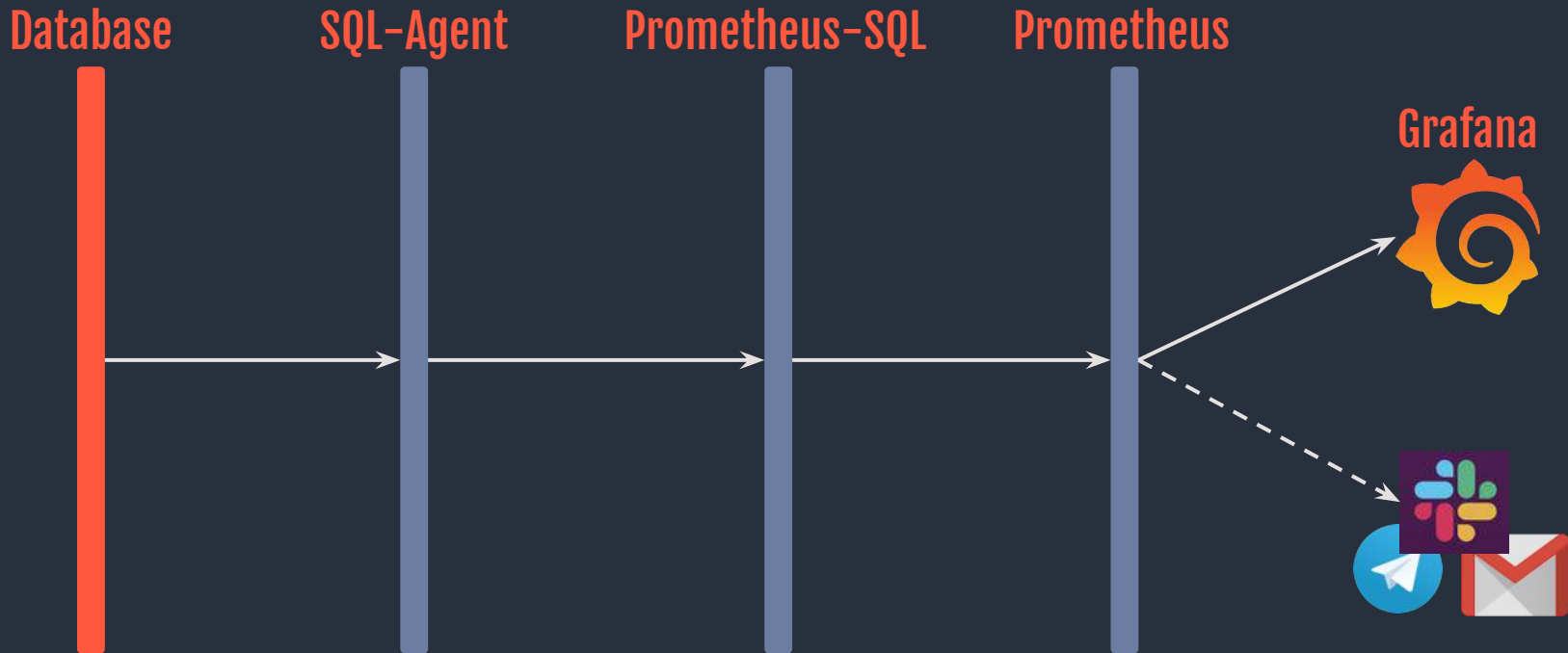
Decision maker

Alert receiver





# Detailed set up





Demo



# Preparation of a database structure

```
[mysql> DESCRIBE secret_keys;
```

Field	Type	Null	Key	Default	Extra
id	int(11)	NO	PRI	NULL	auto_increment
value	varchar(180)	NO		NULL	
created_at	datetime	NO		NULL	

```
3 rows in set (0.04 sec)
```



# Preparation of data

```
[mysql>
mysql>
mysql> INSERT INTO secret_keys (value, created_at) VALUES ("Key1", "2017-01-04 15:17:08"), ("Key2", "2018-01-04 15:17:08"), ("Key3", "2019-01-04 15:17:08");
Query OK, 3 rows affected (0.01 sec)
Records: 3 Duplicates: 0 Warnings: 0

mysql> SELECT * FROM secret_keys;\n
+----+-----+-----+
| id | value | created_at |
+----+-----+-----+
| 1  | Key1  | 2017-01-04 15:17:08 |
| 2  | Key2  | 2018-01-04 15:17:08 |
| 3  | Key3  | 2019-01-04 15:17:08 |
+----+-----+-----+
3 rows in set (0.01 sec)
```



# SELECT query

```
mysql>
mysql>
mysql> SELECT DATEDIFF(NOW(), created_at) as days FROM secret_keys ORDER BY id DESC LIMIT 1;
+-----+
| days |
+-----+
|    20 |
+-----+
1 row in set (0.00 sec)
```



# queries.yml

```
- secret_age:  
  data-source: alerts  
  sql: >  
    SELECT DATEDIFF(NOW(), created_at) as days  
    FROM secret_keys  
    ORDER BY id DESC LIMIT 1;  
  interval: 30s
```



# prometheus-sql.yml

defaults:

data-source: alerts

query-interval: 10s

query-timeout: 5s

query-value-on-error: -1

data-sources:

alerts:

driver: mysql

properties:

host: '127.0.0.1'

port: 3306

user: '%YOUR\_CREDENTIALS%'

password: '%YOUR\_CREDENTIALS%'

database: 'alerts'





# prometheus.yml

```
global:
  scrape_interval: 15s
  evaluation_interval: 15s
rule_files:
  - "simple.rules"
scrape_configs:
  - job_name: 'sql'
    static_configs:
      - targets: ['prometheus-sql:8080']
```



# simple.routes

```
# This one fires, when it is a time to update a key, but it is not yet expired
- alert: TimeToRefreshSecretKey
  expr: query_result_secret_age > 85 and query_result_secret_age < 91
  for: 5m
  labels:
    severity: warning
  annotations:
    summary: "It is a time to refresh a secret key"
    description: "It is a time to refresh a secret key"
```



# simple.routes

```
# This one fires, when it the key is expired and has to be refreshed immediately
- alert: RefreshSecretKeyNow
  expr: query_result_secret_age > 90
  for: 5m
  labels:
    severity: warning
  annotations:
    summary: "The secret key is expired! Refresh it immediately!"
    description: "The secret key is expired! Refresh it immediately!"
```



# Start containers

```
# SQL Agent
```

```
docker run -d \  
  -p 5000:5000 \  
  --name sqlagent \  
  --link mysql8:mysql8 \  
  dbhi/sql-agent
```

```
# Prometheus-SQL
```

```
docker run -d \  
  -p 8080:8080 \  
  --name prometheus-sql \  
  -v ${PWD}/queries.yml:/queries.yml \  
  -v ${PWD}/prometheus-sql.yml:/prometheus-sql.yml \  
  --link sqlagent:sqlagent \  
  dbhi/prometheus-sql \  
  -service http://sqlagent:5000 \  
  -config prometheus-sql.yml
```



# Start containers

```
# Prometheus
docker run -d \
  --name prometheus \
  -p 9090:9090 \
  -v ${PWD}/prometheus.yml:/etc/prometheus/prometheus.yml \
  -v ${PWD}/simple.rules:/etc/prometheus/simple.rules \
  --link prometheus-sql:prometheus-sql \
  prom/prometheus \
  --config.file=/etc/prometheus/prometheus.yml
```



# Alerts in Prometheus

A screenshot of the Prometheus Alerts page in a web browser. The browser's address bar shows the URL `localhost:9090/alerts`. The page has a dark navigation bar with links for `Prometheus`, `Alerts`, `Graph`, `Status` (with a dropdown arrow), and `Help`. The main content area has a white background and features the heading `Alerts`. Below the heading is a toggle switch labeled `Show annotations`. The alert list contains two entries, each in a light green row: `TimeToRefreshSecretKey (0 active)` and `RefreshSecretKeyNow (0 active)`.



# Alerts to Grafana





**End of Demo**





# Alerts to Slack



**Prometheus** APP 9:21 AM

[FIRING:1] RefreshSecretKeyNow @

The secret key is expired! Refresh it immediately!



**Prometheus** APP 09:23 AM

[RESOLVED] RefreshSecretKeyNow @

The secret key is expired! Refresh it immediately!



# Conclusion



Smart Data-driven Alerts is definitely a stable and reliable way to control your data



However it is your last line of defense and it makes a fuss only when a problem already exists.



# Thank you

- [ardemchenkov@gmail.com](mailto:ardemchenkov@gmail.com)
- <https://www.linkedin.com/in/artem-demchenkov-76b69934/>



## More in our "Engineering Corner" on Medium

Check it out. There's interesting



<https://medium.com/billie-finanzratgeber>