

Influxdb 을 통한 모니터링 시스템 구축

상황

1. 각 서버들의 리소스들을 실시간으로 모니터링
2. 과거 이력을 살펴볼수 있게 저장 및 관리
3. 오픈소스로 구현

해결방안

- 방안 1 : 상관없으니 RDBMS로 하자
- 방안 2 : hbase 을 이용하자
- 방안 3 : 파일을 통하여 로그를 적재하자

해결방안으로 찾은 방안이 Time series database(시계열 데이터베이스)

시계열 데이터베이스?

시계열 데이터베이스(TSDB)는 시계열 데이터, 시간별 지수화된 번호 배열(Datetime 또는 Datetime 범위)의 처리에 최적화된 소프트웨어 시스템이다.

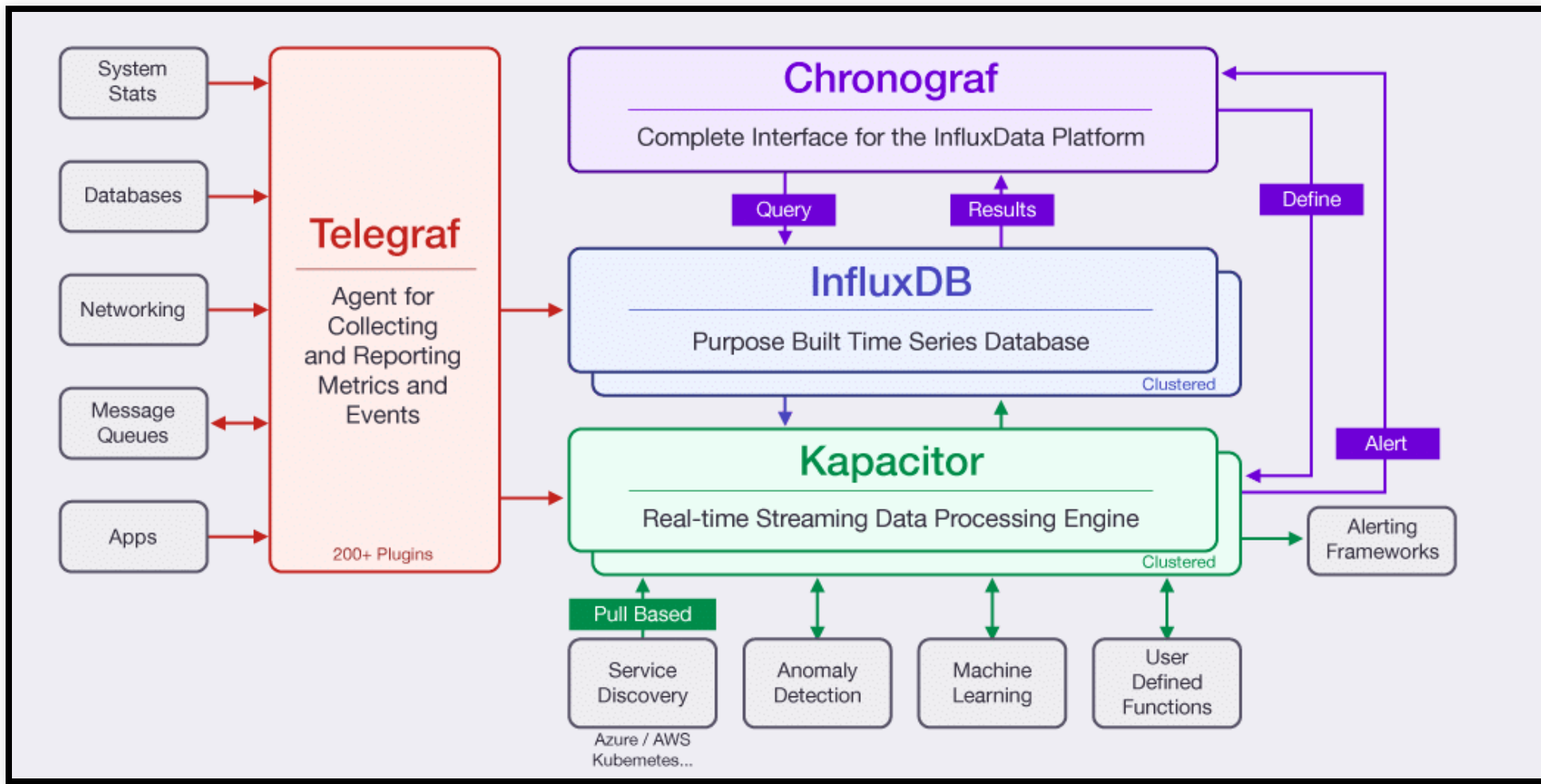
TSDB 순위

☐ include secondary database models

28 systems in ranking, February 2019

Rank			DBMS	Database Model	Score		
Feb 2019	Jan 2019	Feb 2018			Feb 2019	Jan 2019	Feb 2018
1.	1.	1.	InfluxDB	Time Series	15.77	+0.41	+5.18
2.	2.	2.	Kdb+	Time Series, Multi-model	5.41	+0.20	+2.50
3.	3.	4.	Graphite	Time Series	2.95	+0.10	+0.73
4.	4.	3.	RRDtool	Time Series	2.71	+0.10	-0.16
5.	5.	7.	Prometheus	Time Series	2.50	+0.17	+1.58
6.	6.	5.	OpenTSDB	Time Series	2.24	+0.11	+0.52
7.	7.	6.	Druid	Multi-model	1.49	+0.08	+0.55
8.	8.	19.	TimescaleDB	Time Series	0.88	+0.10	+0.86
9.	9.	8.	KairosDB	Time Series	0.52	+0.03	+0.07
10.	11.	9.	eXtremeDB	Multi-model	0.42	+0.06	+0.09

리소스 수집 시스템 구성도



docker-compose.yml 작성

```
version: '2.1'

services:
  influxdb:
    container_name: influxdb
    image: influxdb:latest
    ports:
      - '8086:8086'
    environment:
      - INFLUXDB_DB=alio
      - INFLUXDB_ADMIN_USER=root
      - INFLUXDB_ADMIN_PASSWORD=root
      - INFLUXDB_USER=telegraf
      - INFLUXDB_USER_PASSWORD=secretpassword
    networks:
      influx:
```


docker 생성 및 확인

```
docker-compose up -d
```

```
docker ps
```

telegraf.conf 가져오기

```
docker cp telegraf:/etc/telegraf/telegraf.conf ~/telegraf.conf
```

telegraf.conf 수정

```
[[outputs.influxdb]]
  ## The full HTTP or UDP URL for your InfluxDB instance.
  ##
  ## Multiple URLs can be specified for a single cluster, only ON
  ## urls will be written to each interval.
  # urls = ["unix:///var/run/influxdb.sock"]
  # urls = ["udp://127.0.0.1:8089"]
  urls = ["http://10.18.1.2:8086"]

  ## The target database for metrics; will be created as needed.
  database = "alio"

  ## If true, no CREATE DATABASE queries will be sent. Set to tr
  ## Telegraf with a user without permissions to create databases
  ## database already exists.
  # skip database creation = false
```

docker 에 복사

```
docker cp ~/telegraf.conf telegraf:/etc/telegraf/telegraf.conf
```

```
docker restart telegraf
```

influxdb 에 접속

```
docker exec -it influxdb /bin/bash
```

```
influx
```

데이터 확인

```
use alio
```

```
show measurements;
```

```
select * from cpu limit 100;
```

grafana 접속

<http://localhost:3000>

id : admin / pw : root

grafana.com 에서 가져오겠습니다.
<https://grafana.com/dashboards>

감사합니다.