

Stream metrics storage in Influx DB

- [Influx DB installation](#)
- [Influx DB setup](#)
- [DB structure](#)
- [Examples of data selection from Influx DB](#)

[Influx](#) is open source time series DB.

Influx DB installation

To install Influx DB on CentOS 7, do the following:

1. Create file /etc/yum.repos.d/influxdb.repo:

```
[influxdb]
name = InfluxDB Repository - RHEL $releasever
baseurl = https://repos.influxdata.com/rhel/$releasever/$basearch/stable
enabled = 1
gpgcheck = 1
gpgkey = https://repos.influxdata.com/influxdb.key
```

2. Execute the command

```
yum install influxdb -y
```

3. Enable UDP connection in /etc/influxdb/influxdb.conf file:

```
[[udp]]
  enabled = true
  bind-address = ":8089"
  database = "wcs_oam"
  retention-policy = "default"
```

4. Start Influx DB

```
systemctl start influxdb
```

The installation procedure for Influx DB on Debian / Ubuntu differs only in the method of adding the necessary repository.

Influx DB can be installed to the same server with monitoring backend server. By default, TCP port 8086 or UDP port 8089 is used to connect to Influx DB.

Influx DB setup

To configure Influx DB for metric storage do the following:

1. Set the following parameter in wcs_oam.properties file

```
metric_store=influx
```

2. Set metrics retention policy in init_tsdbs.properties file

```
influx_retention_interval=48h0m0s
```

By default, metrics are stored for 2 days (48 hours).

3. Launch DB setup script

```
./init_tsdb.sh
```

DB structure

The database to store metric values as time series contains fields with the following keys:

```
VIDEO_HEIGHT
VIDEO_WIDTH
VIDEO_RATE
VIDEO_SYNC
VIDEO_FPS
VIDEO_NACK
VIDEO_PLI
VIDEO_CODEC
AUDIO_SYNC
AUDIO_RATE
AUDIO_LOST
AUDIO_CODEC
```

Examples of data selection from Influx DB

Data can be selected from Influx DB for a stream, for which node and media session identifiers are known:

1. Enter Influx DB command line interface

```
influx
```

2. Connect to wcs_oam database

```
use wcs_oam
```

3. The command

```
show measurements
```

will show time series list, every of which corresponds to stream published to certain node in certain media session, for example

```
name: measurements
name
----
3-7ecbd270-123e-11e9-bb40-b96debd59887
3-93412000-123b-11e9-8357-3d4423e30d73
```

4. Select video bitrate values for stream on node 3 in media session 7ecbd270-123e-11e9-bb40-b96debd59887

```
select VIDEO_RATE from "3-7ecbd270-123e-11e9-bb40-b96debd59887"
```

Video bitrate values with timestamps will be shown

name: 3-7ecbd270-123e-11e9-bb40-b96debd59887

time VIDEO_RATE

```
-----  
1546839525823000000 28424000  
1546839525960000000 1002914  
1546839526169000000 727679  
1546839526358000000 662007  
1546839526575000000 645467  
1546839526770000000 633490  
1546839526967000000 583736  
1546839527162000000 622472  
1546839527365000000 593104  
1546839527563000000 666688  
1546839527796000000 638784  
1546839527999000000 637000  
1546839528192000000 647208  
1546839528772000000 421640  
1546839528775000000 587632  
1546839528805000000 740064  
1546839529031000000 753504  
1546839529232000000 767672  
1546839529431000000 931088  
1546839529643000000 1090696  
1546839529840000000 772440  
1546839530028000000 755744  
1546839530242000000 812624  
1546839530429000000 867240  
1546839530630000000 878008  
1546839530833000000 873528  
1546839531031000000  
850352  
...
```