

Хранение статистики потоков в SQL БД

- [Структура БД](#)
- [Установка и настройкаPostgresql](#)
- [Оптимизация работы с БД](#)

Для хранения статистики публикаций потоков бэкенд-сервер использует реляционную SQL БД. Рекомендуется использовать БД Postgresql. В целях отладки и тестирования может быть использована встроенная БД H2, но для промышленной эксплуатации эта БД крайне не рекомендуется.

Структура БД

SQL БД бэкенд-сервера содержит следующие таблицы. При запуске таблицы создаются автоматически:

Поле	Тип	Примечание
ALARM		
id	bigint	PRIMARY KEY
name	character varying(255)	UNIQUE CONSTRAINT
time	bigint	
type	bigint	
value	bigint	
metric_id	bigint	FOREIGN KEY REFERENCES metric(id)
node_id	bigint	FOREIGN KEY REFERENCES node(id)
METRIC		
id	bigint	PRIMARY KEY
enum_name	character varying(255)	UNIQUE CONSTRAINT
name	character varying(255)	UNIQUE CONSTRAINT
note	character varying(255)	
NODE		
id	bigint	PRIMARY KEY
host	character varying(255)	UNIQUE CONSTRAINT
NODE_PROFILE		
node_id	bigint	FOREIGN KEY REFERENCES node(id)
profile_id	bigint	FOREIGN KEY REFERENCES profile(id)
PROFILE		
id	bigint	PRIMARY KEY
name	character varying(255)	UNIQUE CONSTRAINT
rate	integer	
PROFILE_METRICS		
profile_id	bigint	FOREIGN KEY REFERENCES profile(id)
metrics_id	bigint	FOREIGN KEY REFERENCES metric(id)
PROFILE_RULE		
profile_id	bigint	FOREIGN KEY REFERENCES profile(id)
rule_id	bigint	FOREIGN KEY REFERENCES rule(id)
RULE		
id	bigint	PRIMARY KEY

name	character varying(255)	UNIQUE CONSTRAINT
type	bigint	
value	character varying(255)	
STREAM		
id	bigint	PRIMARY KEY
create_date	timestamp without time zone	
duration	bigint	
media_id	character varying(255)	
name	character varying(255)	
node_id	bigint	FOREIGN KEY REFERENCES node(id)
publisher_id	bigint	FOREIGN KEY REFERENCES stream(id)
STREAM_STATUS		
id	bigint	PRIMARY KEY
status	character varying(255)	
timestamp	timestamp without time zone	
stream_id	bigint	FOREIGN KEY REFERENCES stream(id)

Установка и настройкаPostgresql

1. Установите PostgreSQL (на примере CentOS 7) командой

```
yum install postgresql-server
```

2. Инициализируйте БД и запустите PostgreSQL

```
postgresql-setup initdb
systemctl enable postgresql
systemctl start postgresql
```

3. Войдите в консоль PostgreSQL

```
su - postgres
psql
```

4. Создайте базу данных и пользователя, дайте пользователю права на доступ к базе

```
CREATE USER wcoam WITH PASSWORD 'wcoam';
CREATE DATABASE wcoam;
\c wcoam
GRANT ALL PRIVILEGES ON ALL TABLES IN SCHEMA public TO "wcoam";
```

5. Выйдите из консоли

```
\q
exit
```

Оптимизация работы с БД

В сборках до [1.163](#), для ускорения работы PostgreSQL и снижения нагрузки на CPU, необходимо после успешного запуска WCS OAM создать следующие индексы:

```
CREATE INDEX ON stream(name, media_id, node_id);
CREATE INDEX ON stream(node_id, name, status);
CREATE INDEX ON stream(publisher_id);
```

Структура данных таблицы stream в БД wsoam в этом случае должна выглядеть следующим образом:

```
wsoam=# \d stream
          Table "public.stream"
  Column      |          Type          | Modifiers
-----+-----+-----
 id           | bigint                 | not null default nextval('stream_id_seq'::regclass)
 create_date  | timestamp without time zone |
 duration     | bigint                 |
 media_id     | character varying(255) |
 name         | character varying(255) |
 status       | character varying(255) |
 node_id      | bigint                 | not null
 publisher_id | bigint                 |
Indexes:
    "stream_pkey" PRIMARY KEY, btree (id)
    "stream_name_media_id_node_id_idx" btree (name, media_id, node_id)
    "stream_node_id_name_status_idx" btree (node_id, name, status)
    "stream_publisher_id_idx" btree (publisher_id)
Foreign-key constraints:
    "fkfvlh9nv3oa5gel1ocxqlm7bfl" FOREIGN KEY (node_id) REFERENCES node(id)
    "fkh48fb2gtjt8o6acfvln3o4v3" FOREIGN KEY (publisher_id) REFERENCES stream(id)
Referenced by:
    TABLE "stream_status" CONSTRAINT "fk3g764hn776vf7b7gl4ux8lvy" FOREIGN KEY (stream_id) REFERENCES stream(id)
    TABLE "stream" CONSTRAINT "fkh48fb2gtjt8o6acfvln3o4v3" FOREIGN KEY (publisher_id) REFERENCES stream(id)
```