

How to convert NoMa sqlite3 to PostgreSQL

NoMa sqlite3 to PostgreSQL Conversion

1.0 General Information

This document outlines steps to convert NoMa from sqlite3 to PostgreSQL.



- This procedure is for a system that was upgraded from a pre-7.1.x version of GroundWork Monitor with NoMa.
- These steps can only be performed on a system running 7.1.x.
- Prior to 7.1.x, the NoMa connection was hardcoded to use sqlite3. In 7.1.x the database selection in the `NoMa.yml` file is honored.



- Fresh installs of GroundWork Monitor version 7.1.1 do not need to perform this conversion.

2.0 Conversion Steps

Step 1 - Prerequisites

1. Back up existing NoMa database:

```
cp /usr/local/groundwork/noma/var/NoMa.db /tmp/NoMa.db
```

2. Source environment variables:

```
source /usr/local/groundwork/scripts/setenv.sh
```

3. Stop NoMa prior to database work:

```
service groundwork stop noma
```

Step 2 - Cleanup existing sqlite3

1. Compact the database:

```
sqlite3 /usr/local/groundwork/noma/var/NoMa.db  
sqlite> VACUUM
```

2. Check database integrity:

```
sqlite> PRAGMA integrity_check;
```

3. Remove any queued notifications:

```
sqlite> delete from tmp_active;
sqlite> delete from tmp_commands;
sqlite> delete from notification_stati;
sqlite> delete from escalation_stati;
```

4. Exit out of sqlite3:

```
CTRL+D
```

Step 3 - Migrating sqlite3 to PostgreSQL

1. Change directory to /usr/local/groundwork/core/migration/postgresql:

```
cd /usr/local/groundwork/core/migration/postgresql
```

2. Convert the existing NoMa database to PostgreSQL using:

```
/usr/local/groundwork/core/migration/postgresql/dump_sqlite_data_for_postgresql
/usr/local/groundwork/noma/var/NoMa.db | /usr/local/groundwork/postgresql/bin/psql -q -h
"$dbhost" -v ON_ERROR_STOP= -d noma
```

3. Validate the PostgreSQL database now has the data from the sqlite3 database:

- Log into psql command line, the default password is nomapass:

```
psql -U noma
```

- Validate data was migrated successfully by examining the notifications table:

```
select * from notifications;
```

- Run the following commands to set the appropriate sequences for the Postgresql NoMa database:

```
select setval('contacts_id_seq', (select MAX(id) from contacts));
select setval('contactgroups_id_seq', (select MAX(id) from contactgroups));
select setval('escalation_stati_id_seq', (select MAX(id) from escalation_stati));
select setval('escalations_contacts_id_seq', (select MAX(id) from escalations_contacts));
select setval('holidays_id_seq', (select MAX(id) from holidays));
select setval('notification_logs_id_seq', (select MAX(id) from notification_logs));
select setval('notification_methods_id_seq', (select MAX(id) from notification_methods));
select setval('notification_stati_id_seq', (select MAX(id) from notification_stati));
select setval('notifications_id_seq', (select MAX(id) from notifications));
select setval('timeframes_id_seq', (select MAX(id) from timeframes));
select setval('tmp_active_id_seq', (select MAX(id) from tmp_active));
select setval('tmp_commands_id_seq', (select MAX(id) from tmp_commands));
```

- Exit out of psql:

```
\q
```

4. If you were using sqlite3 after the 7.1.x upgrade, edit the /usr/local/groundwork/etc/NoMa.yml file to use postgresql instead of sqlite3:

```
db:
  type: postgresql
```

5. Start NoMa service:

service groundwork start noma