Management of D2000 Applications (Linux)

⚠ Management of D2000 applications must be performed as a `root` user.

The script <instancedir>/bin/d2app <command> [<appname>] is used for management of D2000 applications

<appname> is the name of the application, if not specified, the script will ask for the application name.

<command> is one of the options:

create

creates a D2000 application

⚠ The application name must be unique across the D2000 instance within the computer.

```
[root@localhost bin]# ./d2app create
Enter D2000 application name ('?' for list all): app1
Enter application description: My D2000 application
Enter application startup parameters: /Capp1
Enter D2000 server listen port: 3119
=======================
Ready to create new D2000 application:
           Application name: app1
    Application description: My D2000 application
         Startup parameters: /Capp1
            TCP listen port: 3119
            psql executable: /usr/pgsql-9.6/bin/psql
 Systemd postgresql service: postgresql-9.6.service
--
        D2000 instance name: d2000
     D2000 installation dir: /opt/d2000
=======================

Proceed (yY|nN)? [y]:
```

During creation, enter a description and any additional application startup parameters. If you need to run multiple applications at the same time, change the port so that each has a unique port.

After approval, the application will be created.

```
Proceed (yY|nN)? [y]: y
Creating configuration database...done
Creating logfile database...done
Application app1 created.

Create archive database (yY|nN)? [y]:
```

The script will offer the creation of an archive database for the application after approval fill in the name of the archive:

```
Create archive database (yY|nN)? [y]: y
Enter D2000 archive name ('?' for list all) [self]:
======================
Ready to create D2000 archive:
          Application name: app1
              Archive name: self
           psql executable: /usr/pgsql-9.6/bin/psql
--
        D2000 instance name: d2000
     D2000 installation dir: /opt/d2000
======================

Proceed (yY|nN)? [y]: y
Creating archive database...done
Archive database self created.
```

After approval, the archive database will be created.

A systemd service with a name in the form `d2000-<appname>.service` (in this case it is `d2000-app1.service`.) is automatically created for the application.

The application can be started with the script ./d2app start <appname>, see below.

delete

deletes the entire <appname> application and the archives that are part of the application

```
[root@localhost bin]# ./d2app delete app1
======================
Ready to delete D2000 application:
          Application name: app1
           psql executable: /usr/pgsql-9.6/bin/psql
--
        D2000 instance name: d2000
     D2000 installation dir: /opt/d2000
======================

Proceed (yY|nN)? [n]: y
Deleting archive "self"
Application "app1" was deleted.
```

start

starts the application <appname>

```
[root@localhost bin]# ./d2app start <appname>
<appname> started successfully.
```

stop

stops the application <appname>

```
[root@localhost bin]# ./d2app stop <appname>
<appname> stopped successfully.
```

restart

restarts (stops and starts) the application <appname>

```
[root@localhost bin]# ./d2app restart <appname>
<appname> restarted successfully.
```

status

gives the information on whether the <appname> application is running and whether it has auto-start set

```
[root@localhost bin]# ./d2app status app1
app1 is running. Autostart is enabled.
```

enable-autostart

enables autostart of the application <appname>

```
[root@localhost bin]# ./d2app enable-autostart app1
Autostart of app1 has been enabled.
```

disable-autostart

disables autostart of the application <appname>

```
[root@localhost bin]# ./d2app disable-autostart app1
Autostart of app1 has been disabled.
```

export-syscfg

exports configuration database to a file

```
[root@localhost bin]# ./d2app export-syscfg app1
Enter dump file name [app1.syscfg_dmp]:
Enter password for postgresql user dba:
Export to app1.syscfg_dmp finished.
```

export-logfile

exports log database to a file

```
[root@localhost bin]# ./d2app export-logfile app1
Enter dump file name [app1.logfile_dmp]:
Enter password for postgresql user dba:
Export to app1.logfile_dmp finished.
```

import-syscfg

imports configuration database from a file

```
[root@localhost bin]# ./d2app import-syscfg app1 app1.syscfg_dmp
Enter password for postgresql user dba:
Import from app1.syscfg_dmp finished.
```

import-logfile

imports monitoring database from a file

```
[root@localhost bin]# ./d2app import-logfile app1 app1.logfile_dmp
Enter password for postgresql user dba:
Import from app1.logfile_dmp finished.
```