

NORA inside a Docker

Install Docker on your host system

First, install docker, git and jq on your host system

```
sudo apt-get install docker.io git jq
```

There might be an issue with the DNS address in docker. To check this, run

```
docker run busybox nslookup google.com
```

If the host cannot be reached, grep the address of your DNS server

```
nmcli dev show | grep 'IP4.DNS'
```

create the file `/etc/docker/daemon.json` and insert the address as

```
{  
  "dns": ["yourDNSip", "8.8.8.8"]  
}
```

then restart docker

```
sudo service docker restart
```

go back to the busybox above and check if it works now.

Now, you should add your user to the docker group, otherwise you have not to *sudo* all NORA commands

```
sudo adduser <username> docker
```

!!! you have to re-login to make these changes apply !!!!!

Clone the git repository

NORA was historically named DPX. Note that this name might still often be used in the following. Clone the repository into to a temp folder, or to your favorite program place.

Let's say we clone to your home directory into ~/nora

```
git clone https://<yourname>@bitbucket.org/reisert/dpx.git ~/nora
```

If want to use NORA in a multi-user environment you might want to create a specific group (we usually use *dpxuser*) and change the group of that directory (recursive) accordingly. In any case, you have to set the *s*-flag and some ACLs recursively to that dir (*you might have to run this as sudo. In this case make sure to apply it to the correct ~/nora directory !!*)

```
find ~/nora -type d -exec chmod g+s {} +
```

If want to manage rights by ar group user

```
chgrp -R NORA ~/nora # if you manage your right via usergroup NORA./dp
```

nonetheless

```
setfacl -R -d -m g::rwx ~/nora
```

Initial configuration

Go into the nora directory and run ./install

This will create some local config files in the "conf" directory (for details see separate section below).

NORA has 3 modules

- **Frontend:** *webserver with image viewer*
- **DICOM node** to receive images from a PACS
- **Backend:** processing module (depending on your preferences matlab or nodejs and slurm or SGE based)

By default, all three modules are enabled. To run NORA in this default configuration edit main.conf and set at least MATLABPATH: <your-path-to-matlab>

If you are behind a proxy, it might also be necessary to set the proxy (maybe even with user/password)

```
DOCKER_http_proxy:" ... "
```

Also set the user Nora should act as in the main.conf

```
DPXUSER: " ... "
```

```
DPXGROUP: " ... "
```

Build the docker image

Your main function to control NORA is dpxcontrol. You can always run

```
./dpxcontrol
```

to get more help. To build the DOCKER image, first run

```
./dpxcontrol docker build
```

This will take a while. If everything went well, start all modules with

```
./dpxcontrol start
```

If something went wrong during installation / starts, there might be old 'zombie' containers. In this case you will be suggested docker commands to remove them. You can also start with --forceto autoremove old NORA containers.

To check the status, now use

```
./dpxcontrol status
```

If at least the first point, Docker, is running nicely, you should be able to log into the Webinterface. (see below)

Log into the Webinterface

Now, open a web-browser and go to localhost:81. Default login is

user: root

password: dpxuser

There are several options to create new users. You can either connect to an existing LDAP server, or just create users based on the internal user management of NORA (see Administration section for details).

Troubleshooting and testing

Most log files are written into

```
<path-to-nora>/var/syslogs/
```

These can also be seen from the *admin* dialog at the top of the webinterface. **If the daemon is not starting (or stopping again, red status) check the daemon.log for more info.** In case of a license error, maybe you have to forward your MAC adress to Docker (see main.conf for more info)

For other system parts, there are also some test functions

```
./dpxcontrol test [slurm | email | more_to_be_programmed]
```

Upgrade the database

When your daemon is running nicely, it might be necessary to upgrade the initial database with

```
./dpxcontrol matlab updatedb
```

Autostart on system startup

To automatically start NORA when your computer starts, you can for example add `<path-to-nora>/dpxcontrol start --force` to your `/etc/rc.local`

Slurm setup

A proper configuration of slurm is a science on its own. Ask google for more information. If you are running slurm inside docker (default), the `conf/slurm.conf` is used and you do not have to do too much. Otherwise, if docker is installed outside (this should be the correct practice) you have to it install via

```
sudo apt-get install slurm-wlm
```

on a deb-system and set

```
DOCKER_run_daemon_in_docker: 0,
```

in the `<path-to-nora>/conf/main.conf` file. For configuring slurm itself there is a configurator interface via html, which you usually find here

```
/usr/share/doc/slurmctld/slurm-wlm-configurator.easy.html
```

which generates a conf file `/etc/slurm-llnl/slurm.conf` where you have to replace the hostname by the machine you are running NORA on. Further, the partitions have to be specified. NORA has as default two partitions

(computing queues) `DPXproc` and `DPXimport`, which have to be specified in the `slurm.conf` like in this example:

```
# COMPUTE NODES
```

```
NodeName=hostname CPUs=8 State=UNKNOWN
```

```
PartitionName=DPXproc Nodes=hostname Default=YES MaxTime=INFINITE State=UP
```

```
PartitionName=DPXimport Nodes=hostname Default=YES MaxTime=INFINITE State=UP
```

Alternatively, if you have an existing SLURM running with predefined partitions, you can change the partitions (queues) in NORA's configuration in `<path-to-nora>/conf/main.conf` (`SLURM_QUEUES`)

Also consider

```
./dpxcontrol test slurm
```

for testing whether your slurm configuration is working.

General Configuration

All configuration files are located in `<path-to-nora>/conf` directory. The configuration files are

- `main.conf` - includes all major configurations, responsible for
 - Location of external programs (MATLAB etc)
 - DOCKER port forwarding
 - Daemon behaviour
 - Import behavior (dicoms etc)
 - Mysql information
 - signin/signup behaviour, LDAP configuration
 - Backup behavior
- `pacs.conf` - includes all information how to connect to external dicom nodes
- `routes.conf` - ip-addresses of DB, and routes of mount dirs (depending on hostnames different mount locations are possible)
- `slurm.conf` - slurm configuration
- `smail.conf` - mail configuration

Revision #3

Created 17 September 2020 12:46:45 by reiser tm

Updated 28 September 2020 09:29:16 by reiser tm