Fedbird - tutorial - EdgeLab setup

We go through the steps necessary to setup the Fedbird project in the EdgeLab. The following machines will be used as clients/servers:

Function	Type	IP
Database	virtual	172.25.16.4
Reducer	virtual	172.25.16.5
Combiner	virtual	172.25.16.6
Client1	AGX	172.25.16.2
Client2	AGX	172.25.16.3

On client and sever machines start all docker images inside `tmux` or any other terminal multiplexer.

Begin by opening a VPN session, see VPN - User Guides .

- Database
- Reducer
- Combiner
- Client1 and Client2

Database

Log into the database host machine.

1. In a `root_path` of your choice.
2. Clone the Fedn repo and checkout the right version

```
git clone https://github.com/scaleoutsystems/fedn.git
cd fedn
git checkout v0.2.3
```

3. In `root_path/fedn`start/build the database, skip the `--build` if the image already exists. Inside a terminal multiplexer (eg `tmux`)

```
docker-compose -f config/base-services.yaml up --build
```

Reducer

Log into the reducers host machine.

1. In a `root_path` of your choice.
2. Clone the Fedn repo in the same way as for the Database.
3. In `root_path/fedn` copy settings files

```
cp config/settings-reducer.yaml.template config/settings-reducer.
yaml
cp config/extra-hosts-reducer.yaml.template config/extra-hosts-
reducer.yaml
```

4. In `config/settings-reducer.yaml` set
   `host: 172.25.16.4` and
   `storage_hostname: 172.25.16.4`

5. In `config/extra-hosts-reducer.yaml` set
   `extra_hosts:`
   `combiner: 172.25.16.6`
6. In `root_path/fedn` start/build the reducer, skip the `--build` if the image already exists. Inside a terminal multiplexer (eg `tmux`)

```
docker-compose -f config/reducer-dev.yaml -f config/extra-hosts-
reducer.yaml up --build
```

Combiner

Log into the combiner host machine.

1. In a `root_path` of your choice.
2. Clone the Fedn repo in the same way as for the Database.
3. In `root_path/fedn` copy settings files

```
cp config/settings-combiner.yaml.template config/settings-combiner.
yaml
```

4. In `config/settings-combiner.yaml` set
   `discover_host: 172.25.16.5`
5. In `root_path/fedn` start/build the combiner, skip the `--build` if the image already exists. Inside a terminal multiplexer (eg `tmux`)

```
docker-compose -f config/combiner-dev.yaml up --build
```

Client1 and Client2

Log into the clients.

1. In a `root_path` of your choice.
2. Clone the FedBird repo

```
git clone https://github.com/aidotse/fedbird.git
cd fedbird
```

3. On one of the clients, or some other place, package the client code

```
tar -cvzf package/fedbird.tar.gz client/
```

4. Link or copy the `data` folder into the `root_path/fedbird` folder, on both the clients, for example

```
ln -s <path_to_fedbird_data> data
```

5. In `build_on_xavier.sh` change the build command to
   Clinet1:

```
docker-compose  -f roaster3.yaml -f extra-hosts-roaster3.yaml build
```

Clinet2:

```
docker-compose  -f farallon3.yaml -f extra-hosts-farallon3.yaml
build
```

6. In `run_on_xavier.sh` change the run command to
   Clinet1:

```
nvidia-docker run --gpus all --add-host=combiner:172.25.16.6  -v
$PWD/data/client2:/data fedbird-roaster3-client /bin/bash -c "fedn
run client -in fedn-network-xavier.yaml"
```

   Clinet2:

```
nvidia-docker run --gpus all --add-host=combiner:172.25.16.6  -v
$PWD/data/client1:/data fedbird-farallon3-client /bin/bash -c
"fedn run client -in fedn-network-xavier.yaml"
```

7. In `extra-hosts-roaster3.yaml` (for Client1) and `extra-hosts-farallon3.yaml` (for Client2) set
   `extra_hosts:`
   `combiner: 172.25.16.6`
8. Build the image on both clients

```
sh ./build_on_xavier.sh
```

9. Run the image on both clients

```
sh ./run_on_xavier.sh
```

Now the FedBird project should be ready to use. For training, inference, visualization and more see Fedbird - tutorial - training, inference, visualization and more