

Holo Fuel UI

Perry Kundert

April 15, 2019

Contents

1	Holo Fuel User Interface	1
1.1	Dashboard	1
2	Running the hApp	1
2.1	Requirements	1
2.2	Building	2
2.3	Starting DNA Agents	2
2.4	Running the holofuel-app GUI	2

1 Holo Fuel User Interface

Displays the Agent's account Nickname and Agent ID, and available balance (credit and balance, net of payables such as fees and outstanding funds promised but not yet received by the recipient).

1.1 Dashboard

Pending operations and current Transaction are the primary display. In the menu button, navigation to the Send and Request funds pages can be found.

2 Running the hApp

2.1 Requirements

Check out and build holofuel and n3h in your ~/src/ directory. First, build n3h:

```
$ cd ~/src
$ git clone git@github.com:holochain/n3h.git -b master
$ cd n3h
$ npm install
```

Next, build the HoloFuel DNA package ~/src/holofuel/dist/holofuel.dna.json:

```
$ cd ~/src
$ git clone git@github.com:Holo-Host/holofuel.git -b develop
$ make -C holofuel build install
```

2.2 Building

Obtain the `holofuel-app`, and Compile the project resources with `webpack`:

```
$ git clone git@github.com:mjbrisebois/holofuel-app.git
$ cd holofuel-app
$ npm run build
```

2.3 Starting DNA Agents

Run a "bootstrap" DNA agent: `##+BEGIN_EXAMPLE bash $ cd ~/src/holofuel-app $ make start-bootstrap-agent-dna ##+END_EXAMPLE bash`

This creates a file in your `~/src/holofuel` directory, used to start subsequent DNA agents connected via the "bootstrap" agent. In other tabs/windows, run:

```
##+BEGIN_EXAMPLE bash $ cd ~/src/holofuel-app $ make start-bootstrap-dna-1 ##+END_EXAMPLE
bash
```

And, in another tab/window (if desired):

```
##+BEGIN_EXAMPLE bash $ cd ~/src/holofuel-app $ make start-bootstrap-dna-2 ##+END_EXAMPLE
bash
```

2.4 Running the holofuel-app GUI

Run a simple HTTP server in the root directory, to make available the `~/src/holofuel-app/index.html` file at by navigating to `http://localhost:8000` with a web browser:

```
$ cd ~/src/holofuel-app
$ python -m SimpleHTTPServer
```

This will redirect you to `http://localhost:8000/#/3000/` (note the port 3000, assigned to the first "bootstrap" agent).

Any subsequent agents you started (above), will be assigned ports starting at 3100, eg. `http://localhost:8000/#/3100/`, `http://localhost:8000/#/3101/`, ...