

Pipes and Filters implementation

Alexander Gabriel

Setup

```
npm install
```

Start

Assuming rabbitmq-server is running on the system

```
npm start
```

Configuration

Pipes and Filters can be put together in the `config.yml`. There are three example filters implemented. You can put them together as you like. The only thing you need to do is write a filter and import it in the `index.mjs`. I didn't implement dynamic module imports, as this seemed very hacky to me. Would have been easier with `commonjs` modules, but i already started with ES6 modules.

Choices Made

I made all the parts of the system durable and didn't disable any acknowledgements as i wanted to get idempotent results and didn't want to lose any data. It totally depends on what the system should do, and how important the data is, the pipe handles. If it's time relevant streaming data maybe acknowledging the data or making the channels durable would be useless, as the data maybe arrive when nobody needs it anymore. In a case like this it would be better to just drop the data and process new data.

I used direct exchanges as there is always a fixed address a message should go to, at least in my case. The processing should always have the same order and so every filter talks "directly" (through the messagebus) with the next one in the row.