

Freaky Hours



Managing .odt files with node.js

Gabriele Falasca

*Freaky
Hours*

SOURCESENSE
Open Solutions for your Value

Who am I?



- Gabriele Falasca (aka gabrycaos)
- Frontend developer @Sourcesense
- Javascript enthusiast
- Open source lover
- Bassist during free time
- /gabrycaos on facebook
- @gabrycaos on twitter
- gabriele.falasca@sourcesense.com



.odt format

I don't need to explain what is it here :)

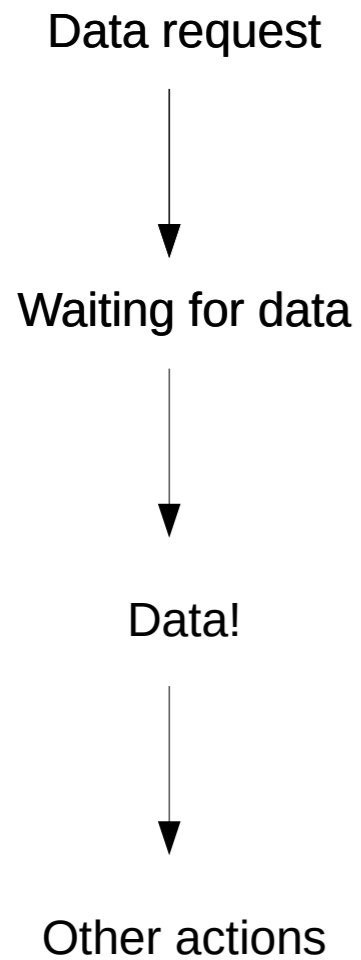
Node.js

Javascript V8 engine for writing server side applications.

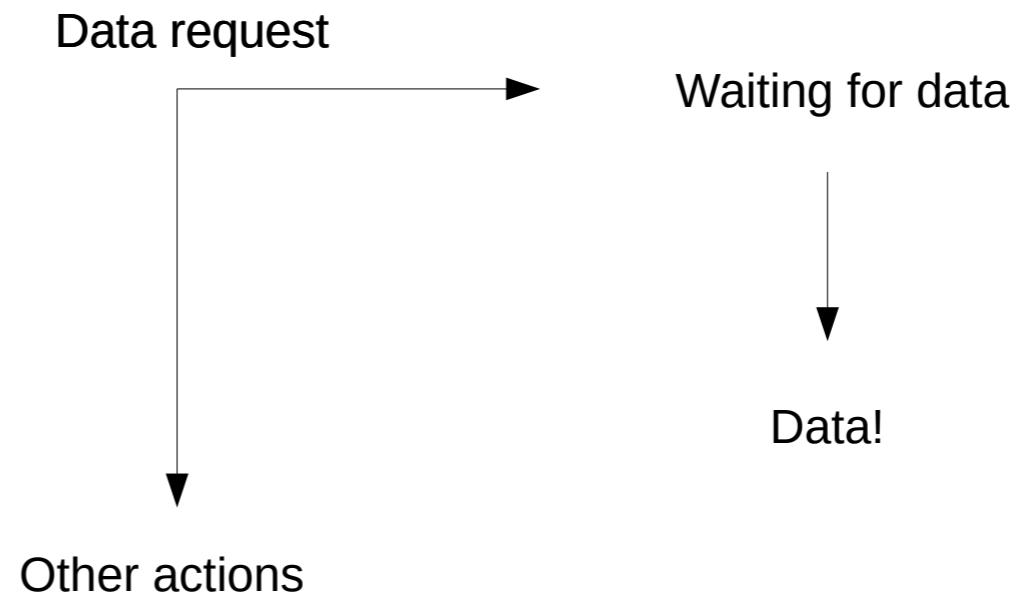


Node.js: async and event driven

Synchronous Flow:



Async Flow:



Node.js: async and event driven

```
// sync function
const data = getData(params)

// make something with data variable
---

// async function
getData(params, (data) => {
    // make something with data variable
})
```

Node.js: Stream

A sort of *data-flow* divided in *chunk*, that goes from a Producer to a Consumer, event driven.

A stream can be:

- **Readable,**
- **Writable,**
- **Duplex,**
- **Transform**

.odt files structure

- mimetype
- Thumbnails/
- **content.xml**
- Configurations2/
- manifest.rdf
- styles.xml
- settings.xml
- meta.xml
- META-INF

content.xml structure in the editor!

Let's see some code!

Let's see some code!

```
// import modules
import fs from 'fs';
import zlib from 'zlib';
import colors from 'colors';
import { parseString, Builder } from 'xml2js';
import archiver from 'archiver';
import AdmZip from 'adm-zip';
import Logo from './logo';
```

Let's see the rest of the code directly in the editor!

QUESTIONS?!

Demo code can be found on <https://github.com/gabrycaos/node-odt-managment-demo>