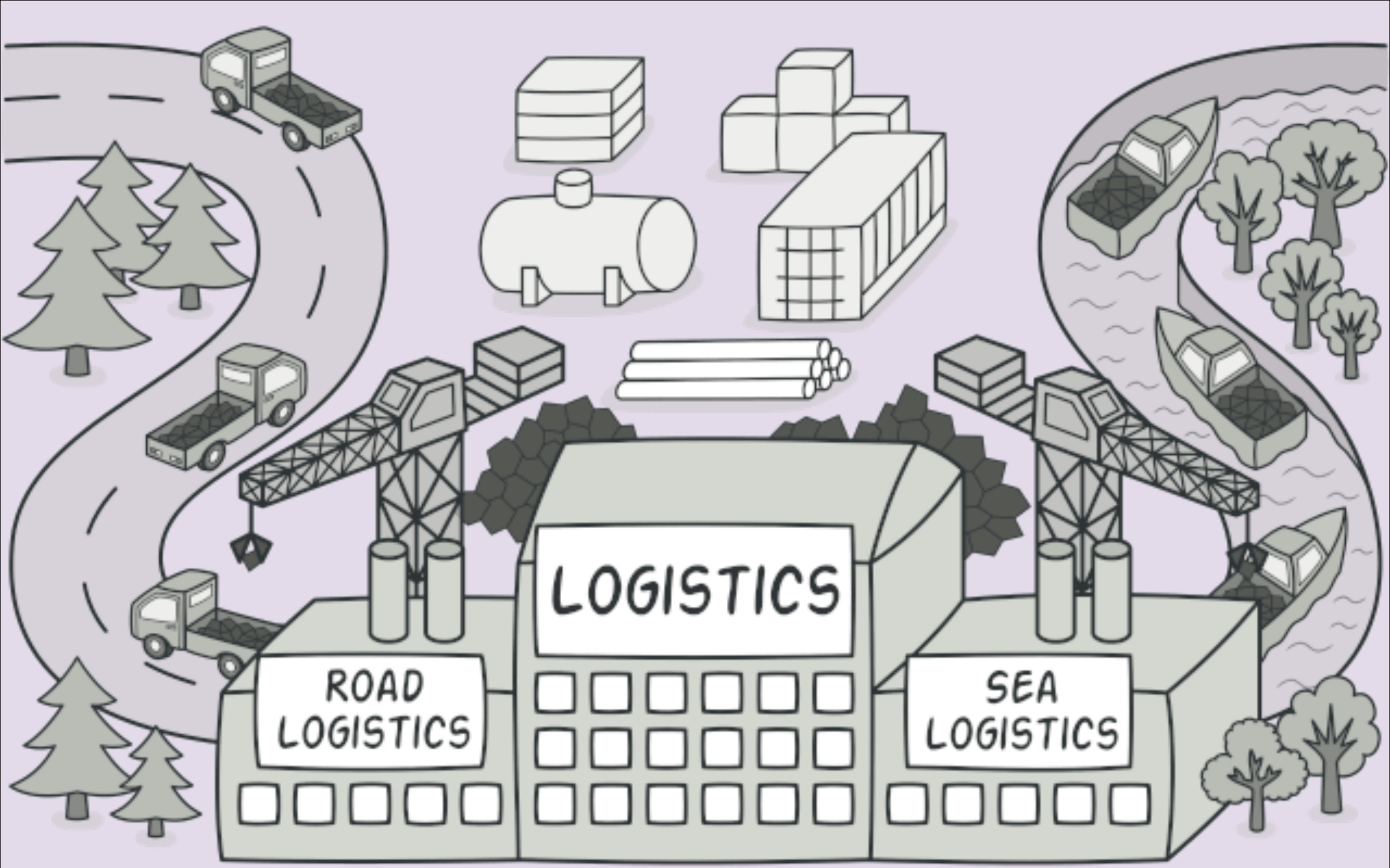
Builder Design Pattern

OGIF #6

Building on previous Factory pattern...


LOGISTICS
ROAD LOGISTICS
SEA LOGISTICS

DrinkFactory.createJager()?

“But I like my drink a bit more X and Y and Z”

Let’s build it out

Drink → DrinkBuilder

```
new DrinkBuilder()
    .name("JagerBomb")
    .addIngredient("Jagermeister", 30)
    .addIngredient("Redbull", 120)
    .mix();

new DrinkBuilder()
    .name("JagerGrenade")
    .addIngredient("Jagermeister", 30)
    .addIngredient("Tequila", 30)
    .addIngredient("Vodka", 30)
    .addIngredient("Redbull", 50)
    .mix();
```

The Builder class

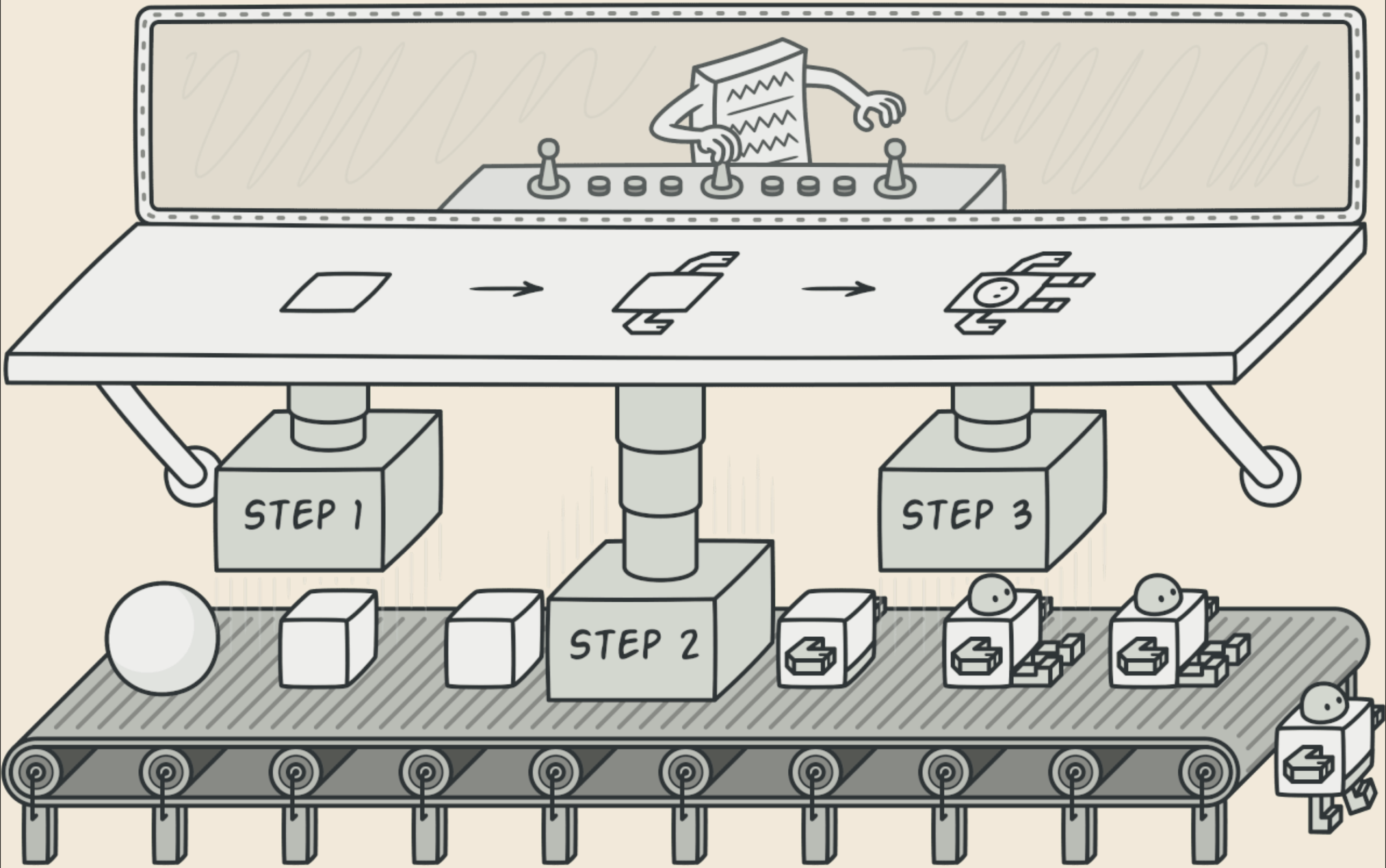
```
class DrinkBuilder implements Builder {
  private drink: Drink;
  private name = "";
  private ingredients = [];

  constructor() {}

  name(n: string) {
    this.name = n;
    return this;
  }

  addIngredient(name: string, ml: number) {
    this.ingredients.push({ name, ml });
    return this;
  }

  mix() {
    return new Drink(this.name, this.ingredients);
  }
}
```


STEP 1
STEP 2
STEP 3

Real life examples

Chained Method Calls

When a chain of method calls is too long t
from the object the methods are called on.

```
elements
    .addClass( "foo" )
    .children()
        .html( "hello" )
    .end()
    .appendTo( "body" );
```

```
wretch("anything")
  .get()
  .notFound(error => { /* ... */ })
  .unauthorized(error => { /* ... */ })
  .error(418, error => { /* ... */ })
  .res(response => /* ... */)
  .catch(error => { /* uncaught errors */ })
```

spotify-web-api-node / src / spotify-web-api.js

Code | Blame | 1662 lines (1551 loc) · 69.8 KB

```
117       */
118     getTrack: function(trackId, options, callback) {
119       return WebApiRequest.builder(this.getAccessToken())
120         .withPath('/v1/tracks/' + trackId)
121         .withQueryParameters(options)
122         .build()
123         .execute(HttpManager.get, callback);
124     },
```

Thank you.