JSON data class for REST calls

Phillip G. Bradford*

March 31, 2025

Abstract

These notes discuss a Javascript JSON data class: JsonContainer. ALl of our data-classes that store data from the APIs inherit from JsonContainer. All of these data classes use the class CardText.js to get their header-text. Specifically, CardText.js is a single-source for the cards header-text. THe Stacks APIs all have a single-source for the endpoints we use. This source is stored in the file RoutesAndEndpoints.js.

1 JsonContainer class

The class JsonContainer is the base-class for our data classes:

1. BurnBlockContainer
2. PoxCycle
3. StacksBlock
4. TenureTxContainer

The class JsonContainer is in Listing 2.

```
export default class CardText {
  static CARD_TITLE_BURN_BLOCK = "Burn Block";
  static CARD_TITLE_POX_BLOCK = "Pox Block";
  static CARD_TITLE_STACKS_BLOCK = "Stacks Block";
  static CARD_TITLE_TENURE_BLOCK = "Tenure Transaction";
}
```

Listing 1: The single-source of text for cards on the screen

*phillip.bradford@uconn.edu, phillip.g.bradford@gmail.com, University of Connecticut, Department of Computing, Stamford, CT USA

The class CardText in Listing 1 can be extended to hold other fields besides these CARD_TITLE...s. A key reaon for storing all card text in one common file is to make translation and updating easy.

```
class JsonContainer {
    constructor(data = {}) {
        this.setData(data);
    }

    setData(data) {
        if (typeof data !== "object" || data === null) {
            throw new Error("Data must be a valid JSON object");
        }
        this._data = { ...data }; // Clone
    }

    getData() {
        return { ...this._data };
    }

    toJson() {
        return JSON.stringify(this._data, null, 2);
    }
}
```

Listing 2: JsonContainer class

2 Classes derived from JsonContainer

Each of the classes BurnBlockContainer, PoxCycle, StacksBlock, and TenureTxContainer are derived from JsonContainer.

Indeed, any class that inherits from the class JsonContainer might have a constructor as seen in Listing 3.

```
export default class BurnBlockContainer extends JsonContainer {

        constructor(data = {}) {
            super(data);
            this.setData({...data,
        "card_title" : CardText.CARD_TITLE_BURN_BLOCK});
```

```
7           }
8   }
```

Listing 3: BurnBlockContainer inheriting from JsonContainer

The Listing 3 uses the JavaScript ... operator to split the object data so the new field "card_title" and its value may be added.

The data passed into the class BurnBlockContainer may be modified to surpress some fields from being presented. This will be on a block-type by block-type basis.

3 Notes

ES6 (ECMAScript 6) is ECMAScript 2015.

React is based on ES6.

References

[1] https://tc39.es/ecma262/