A Dynamic **Card** in React Bootstrap - Part 1

Phillip G. Bradford*

March 31, 2025

Abstract

The Card component in React bootstrap can be made dynamic. Card components play an important role for our block explorer. This dynamic card uses the *useState* function from *React hooks*. A stacks block is featured here.

1 Stacks block dynamic card component

To create a basic Card component for a *Stacks block*, do the next steps.

1. `npx` create-react-app card-example-dynamic

2. `cd` card-example-dynamic

   In Listing 1, we update the *App.js* file in the `src` directory. We have truncated the hash values to fit on the page.

```
1  import './App.css';
2
3  import DynamicCard from './components/DynamicCard';
4
5  const container =
6  {
7    "canonical": true,
8    "height": 733786,
9    "hash": "0x456ff10c1f3d870697fe8ea80ce27f01756ba4875515c47d...",
10   "block_time": 1741441821,
11   "block_time_iso": "2025-03-08T13:50:21.000Z",
12   "tenure_height": 187940,
13   "index_block_hash": "0xa9900a4323c92ae587f8e55622b8d46cb991...",
```

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

```
14    "parent_block_hash": "0xfeb3d1dd27e2555ea372669c9547b7887d3...",
15    "burn_block_time": 1741441223,
16    "burn_block_time_iso": "2025-03-08T13:40:23.000Z",
17    "burn_block_hash": "0x00000000000000000000079a14b7bcddfcb0c1...",
18    "burn_block_height": 886865,
19    "miner_txid": "0x0813472654ee73ed1686c1a72a3e7d28c655ded29d...",
20    "execution_cost_read_count": 0,
21    "execution_cost_read_length": 0,
22    "execution_cost_runtime": 0,
23    "execution_cost_write_count": 0,
24    "execution_cost_write_length": 0,
25    "txs": [
26        "0xc2f9bb6a6a73e08ce38da86757cd866e54381c2bde43141e70ab52...",
27        "0xd0c97bf834e9e87e6e0811a1e7fb4ccc9dbf1fc3ebccee83ab2ebc..."
28    ],
29    "parent_microblock_hash": "",
30    "parent_microblock_sequence": -1,
31    "microblocks_accepted": [],
32    "microblocks_streamed": [],
33    "microblock_tx_count": {}
34 };
35
36 export default function App() {
37    return (
38        <div className="App">
39            <header className="App-header">
40                <DynamicCardFull props={container} />
41            </header>
42        </div>
43    );
44 }
```

Listing 1: Update app.js in the *src* folder for a Stacks block

3. `npm` install react bootstrap react-bootstrap

4. Replace the code in App.js in the `src` directory with the code in Listing 1.

 Next we generate the component `<DynamicCardFull />`.

1. Create a subdirectory `components` of the `src` directory.

2. Create a file `BlockCard.js` in the `src/components` directory.

3. Put the content of Listing 2 in `DynamicCardFull.js`.

This code will host a *DynamicCardFull* in *App.js* on line 7. Specifically, Add `<DynamicCardFull />` between `<header className="App-header">` and `</header>`

This card is dynamic. It is dynamic by storing *state* based on
`const [open, setOpen] = useState(false);`
This is from *React hooks*.
Update `<DynamicCardFull />` with the following properties in the *container* object.
height="888217",
hash="0000000000000000000008567ffc2eb86897fd8db31b2dd0b71db42684178a0dd", etc.

In Listing 1, the basic updated version is,

```
<DynamicCardFull props={container} />
```

```jsx
import React, { useState } from "react";
import { Card, Button } from "react-bootstrap";
import "bootstrap/dist/css/bootstrap.min.css";

export default function DynamicCardFull ({ props }) {
  const [open, setOpen] = useState(false);

  return (
   <div className="card custom-card border-primary mb-3">
      <Card style={{ maxWidth: "700px", margin: "auto" }}
            className="container ms-0" >
      <Card.Body>
        <Card.Title><h3>{props['card_title']}</h3></Card.Title>
        <ul className="list-group list-group-flus no-bullets">
          <li class="list-group-item text-primary">
            <Button
              onClick={() => setOpen(!open)}
              className="mb-2"
            >
          </Button>
            <h6>{open ? "Hash tail: " + props['hash'].slice(-20)
                      : "Hash: " + props['hash']}</h6>
          </li>
          <ul className="list-group list-group-flus no-bullets">
              {Object.entries(props).map(([key, value]) => (
                  <li key={key}>
                      <h6>{key}: {JSON.stringify(value)}</h6>
                  </li>
              ))}
          </ul>
        </ul>
      </Card.Body>
      </Card>
      </div>
   );
};
```

Listing 2: Basic Dynamic Card for a Stacks block

Listing 2 shows a basic *dyamic* card from React-bootstrap. This card holds any block. It is purposely designed to hold any block so each block on the explorer looks similar. This is in the file DyanmicCardFull.js in the src/components directory.

References

[1] Ryan J. Blake: *Mastering Bootstrap 5*, Beginners Handbook, 2024.

[2] React Bootstrap: *React Bootstrap Cards*,
https://react-bootstrap.netlify.app/docs/components/cards/ , Accessed: 2025-03-15.