

Creating and Queuing Promises



Nathan Taylor

SOLUTION ARCHITECT

@taylonr www.taylonr.com



Promise States



Pending



Fulfilled



Rejected



A pending promise is a
promise that has not yet
settled



Promise

Object that represents the eventual completion (or failure) of an asynchronous operation, and its resulting value.

https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/Promise



A Promise Object

```
let temp = new Promise();
```

Understanding Promise States



Promise States

Settled

Resolved



If the associated promise has already been resolved, either to a value, a rejection, or another promise, this method [resolve] does nothing.

https://developer.mozilla.org/en-US/docs/Mozilla/JavaScript_code_modules/Promise.jsm/Promise



Rejecting a Promise



You have the power to
control how a promise is
settled



Waiting for All Promises to Resolve



Metadata Types

Job Type

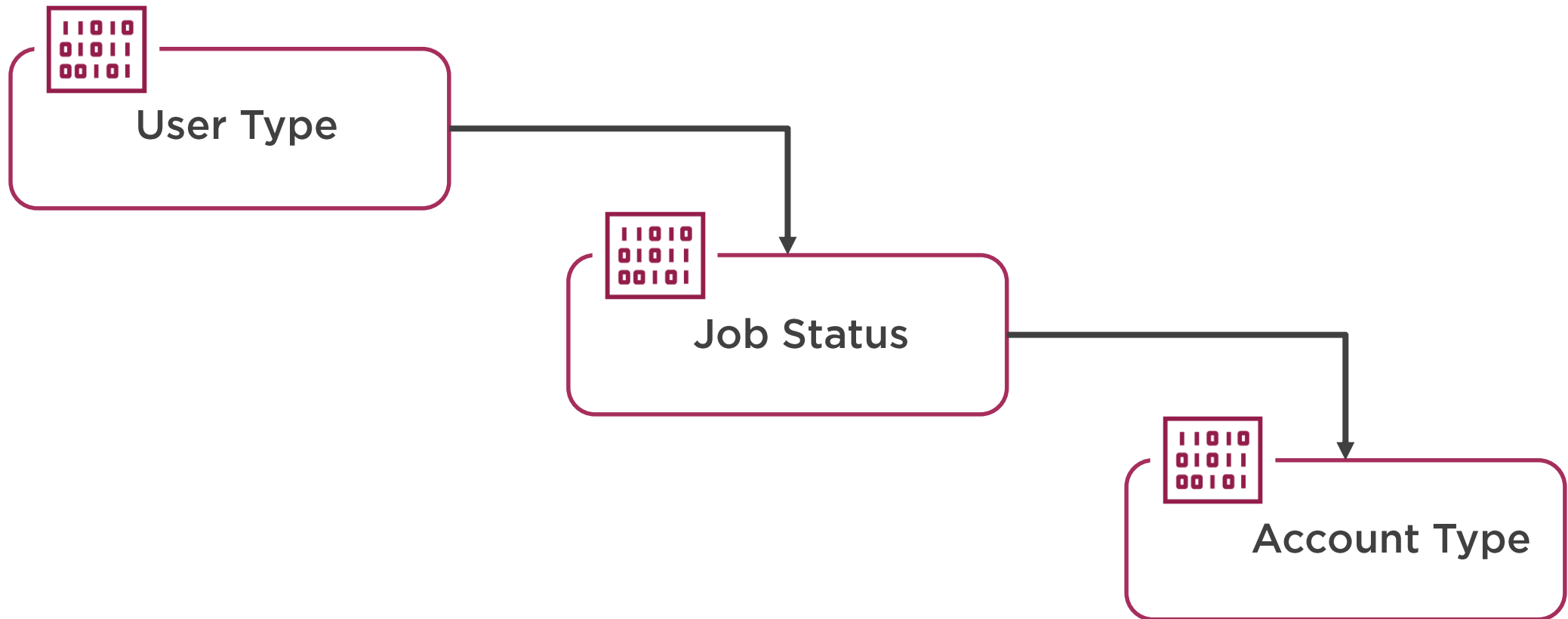
Job Status

Account Type

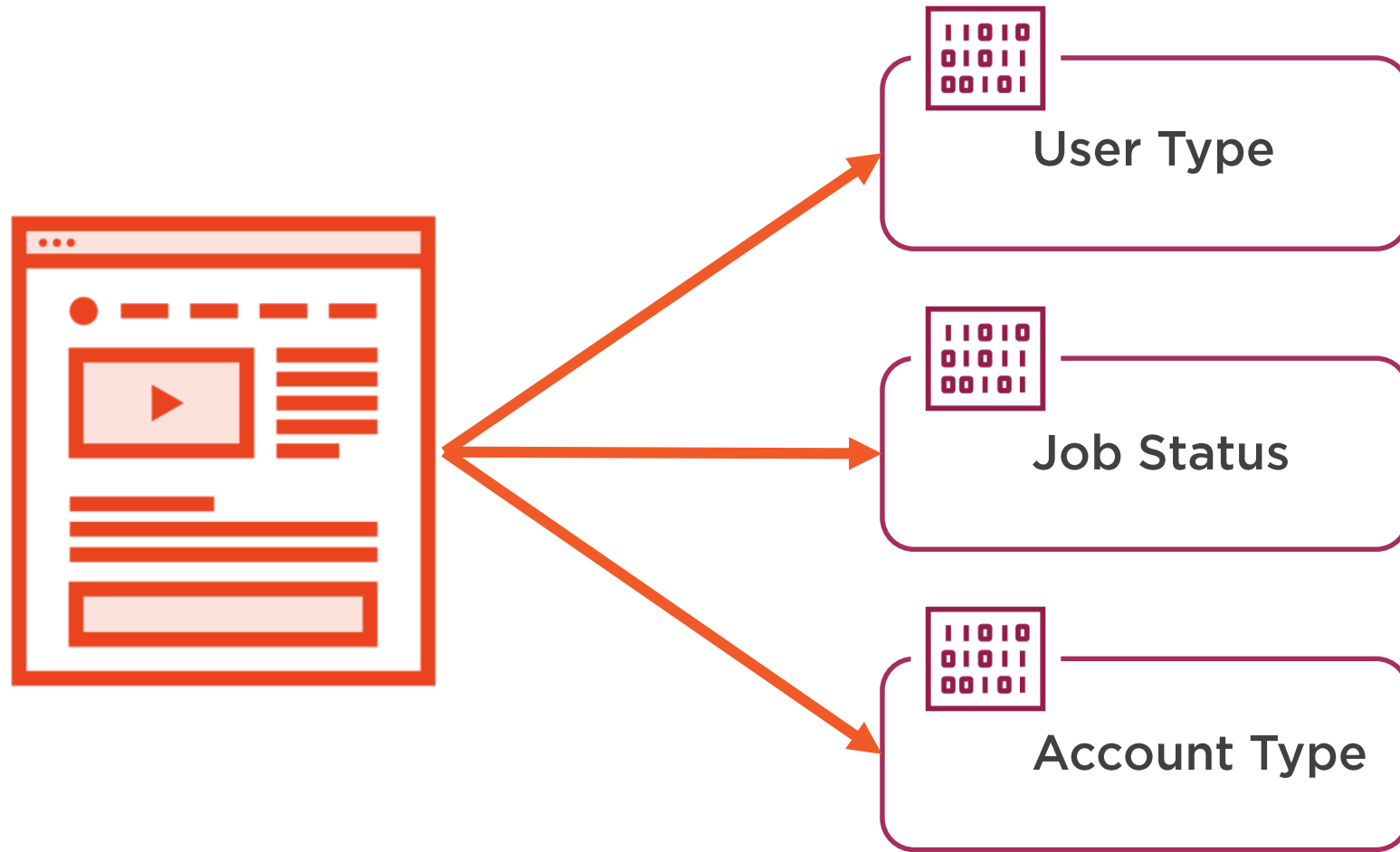


Loading Metadata

Sequential Loading



Power of Asynchronous Programming



Settling All Promises



How Is allSettled Different?

Different Data Returned



allSettled Returned Data

resolved.js

```
{  
  status: "fulfilled",  
  value: {}  
}
```

rejected.js

```
{  
  status: "rejected",  
  reason: {}  
}
```

How Is allSettled Different?

Different Data Returned

Don't *Need* a Catch



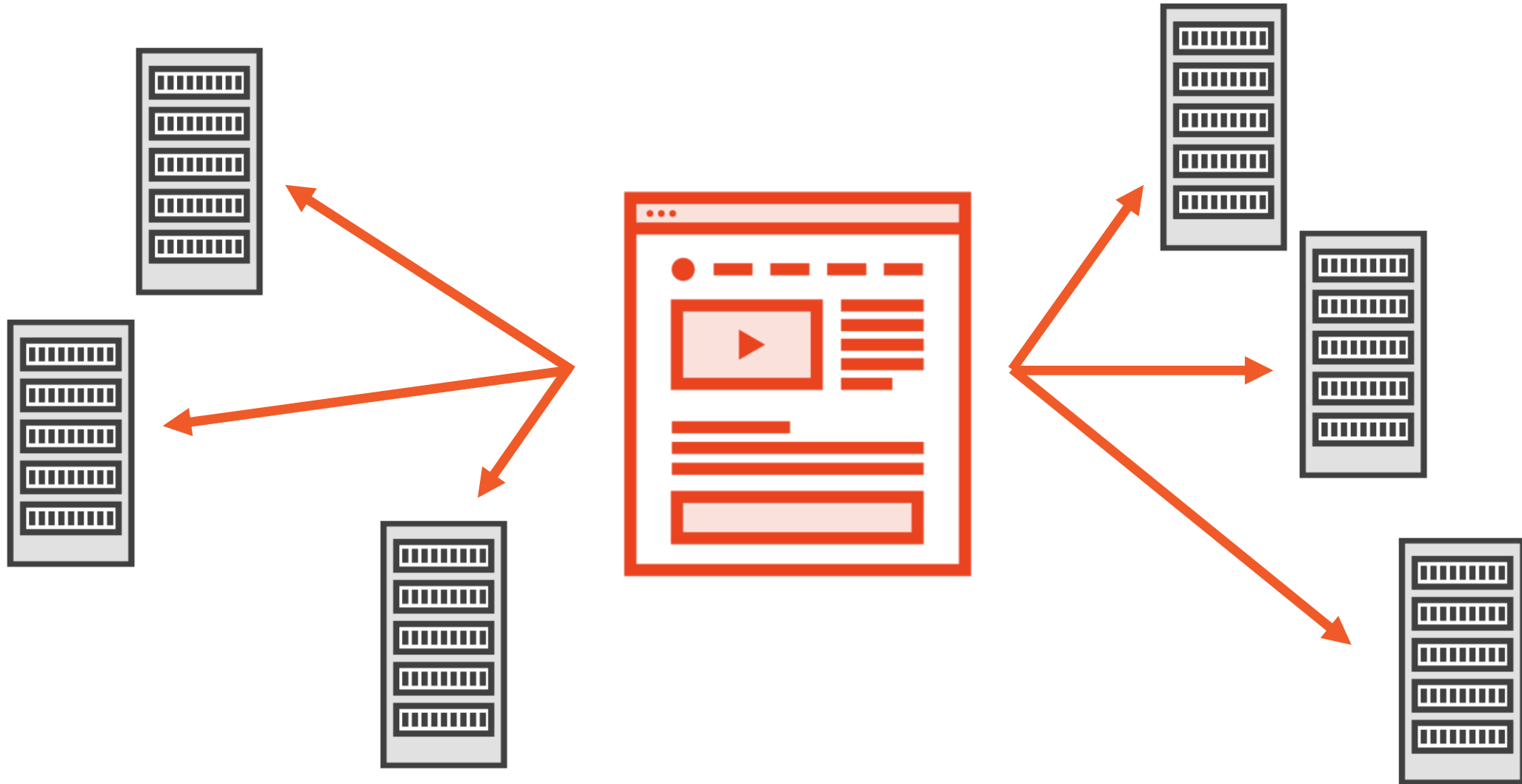
A catch block is
recommended



Racing Promises



Using the Closest Endpoint



Using the Closest Endpoint



“race” is a rare function



promise.js

```
let promise = new Promise();
```


promise.js

```
let promise = new Promise((resolve, reject) => {});
```

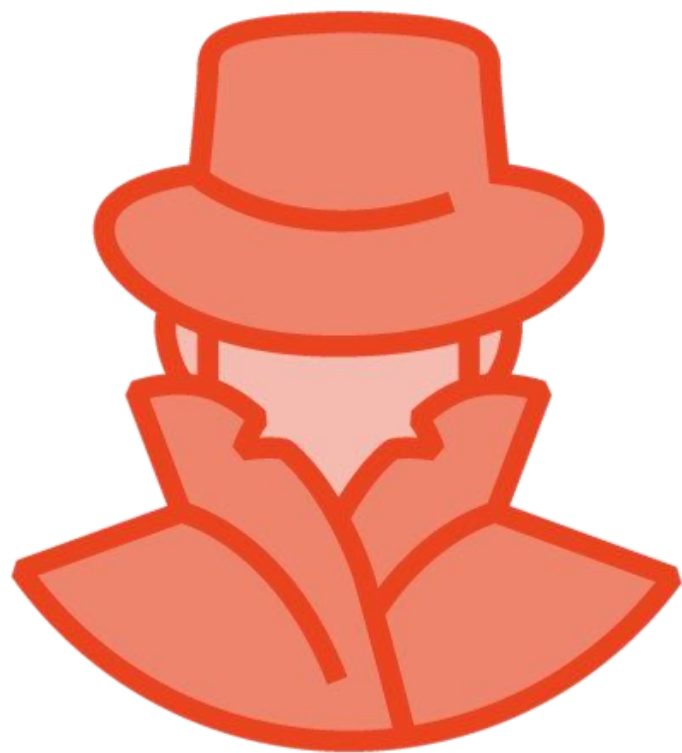
Ways to Queue Promises

all

allSettled

race





Secrets of async/await

