



NS2022

ACCELERATING INNOVATION

UxMedia: Browser-Native Px Rendering

*Logan Byam
Software Engineer @ Tridium*



NS2022
ACCELERATING INNOVATION



NS2022
ACCELERATING INNOVATION

UxMedia

What the heck is it?

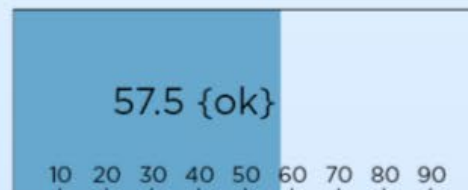
UxMedia!

This is a Px page.

It's rendered natively in the browser.

Just HTML, CSS, and JavaScript.

Subscriptions are
handled by BajaScript.



Station Does:

- Provide raw Px data
- Resolve ORDs / RPCs

Browser Does:

- Process and render Px data
- Bind data to widgets and update over time

Why UxMedia?

- Performance
- Developer Experience
- User Experience
- Future Growth



NS2022
ACCELERATING INNOVATION

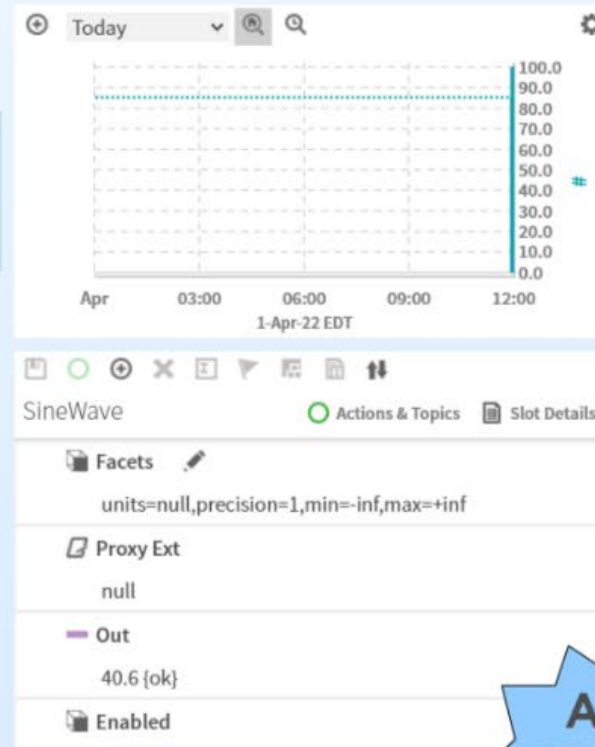
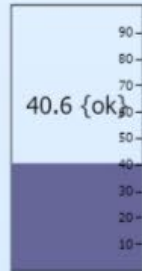
Should I Use UxMedia?

Compatibility

Buttons!

Labels!

Field Editors!



Graphs! Pictures! Web Widgets!

AND MORE!

Device Performance



Widgets Vs. Bindings

Lots And Lots And Lots And
Lots And Lots Of Widgets?

Or:

Display Name	Value
One	29.4
Really,	25.1
Really,	92.3
Really,	40.2
Really,	30.9
Big	61.2
Query?	87.8

Development Concerns

- Java still well-supported
- Web Development Tools from Tridium
- Browser-native experience

Where to learn more?

- Doc Developer → UxMedia
- <https://github.com/tridium>



NS2022

ACCELERATING INNOVATION

Let's code.



NS2022

ACCELERATING INNOVATION

Building User Interfaces with **spandrel**

*Logan Byam
Software Engineer @ Tridium*



NS2022
ACCELERATING INNOVATION

Prerequisites

- HTML/JS
- **bajaux**
- BajaScript
- Doc Developer → User Interface → Open Web Technologies

NS2022
ACCELERATING INNOVATION

spandrel

What the heck is it?



spandrel features

- Build widgets from other widgets
- Declarative and less procedural
- Easy event handling
- Automatic re-renders
- Development status! 🎉

spandrel and JSX

- Build out widgets and HTML together
- JSX is not React

Static HTML Widget

```
1 define([ 'bajaux/Widget' ], function (Widget) {  
2     return class extends Widget {  
3         doInitialize(dom) {  
4             dom.html('<span>Hello world!</span>');  
5         }  
6     };  
7 });  
8
```

Hello world!

Dynamic spandrel Widget

```
1 // from here, @jsx and the define() call will be omitted.
2
3 /** @jsx spandrel.jsx */
4 define([ 'bajaux/Widget' ], function (Widget) {
5   return class Greeting extends Widget {
6     doLoad(name) {
7       this.jq().html(`<span>Hi, ${ name }!</span>`);
8     }
9   };
10 });
```

Hi, Vikram!

Load Value

Read Value

Widgets as tags

```
1 const StringEditor = spandrel((string) => {  
2   return <input type="text" value={string}/>;  
3 });  
4  
5 return spandrel([  
6   <StringEditor value={'Hello world!'} />  
7 ]);  
8
```

Hello world!

Reading values

```
1 return class StringEditor extends spandrel((str) => {  
2   return <input type="text" value={ str } />;  
3 }) {  
4   doRead() {  
5     return this.jq().children('input').val();  
6   }  
7 };
```

Load Value

Read Value

Dynamic field editor lookup (4.12)

```
1 return spandrel(() => (  
2   <div style={{ display: 'grid', gap: '0.5em' }}>  
3  
4   </div>  
5 ));
```


Event handlers (4.12)

```
1 return spandrel(() => [  
2   <button type="button" className="ux-btn">  
3     Click me!  
4   </button>,  
5   <any value={baja.Date.today()} />  
6 ], { strategy: 'niagara' });
```

Click me!

1-Apr-22

Field Editors

```
1 return class FieldEditors extends spandrel((comp) => {  
2   const props = comp.getSlots().properties().toArray();  
3   return <table className="ux-table"></table>;  
4 }, { strategy: 'niagara' }) {};
```

Change Bound Component

Save

Slideshows are not production

- You need tests
- You need source control
- I'm just playing!
- Use **grunt-niagara** for test coverage
- <https://github.com/tridium>

How to get it?

- Out in 4.10, updates in 4.12
- **grunt-init-niagara** sets up everything including JSX
- Tutorials in Doc Developer → **bajaux**
- Live code examples in **docDeveloper** palette

A nighttime photograph of a city skyline with illuminated skyscrapers against a dark blue sky. The image is used as a background for the event title.

NS2022

ACCELERATING INNOVATION

Thanks!