Sencha
An Idera, Inc. Company

ARE YOU CHOOSING YOUR WEB COMPONENTS WISELY?

AN INTRODUCTORY GUIDE

By Kirti Joshi

Why should you read this guide?

Choosing components for your application can be tricky. Should you develop the desired components from scratch? Should you choose from available free open-source resources or go with an enterprise-backed vendor? Every situation is different—however, there are some prime considerations you should be aware of.

This guide discusses three key considerations when choosing your web components.

Web Components

Components for the web, promoting reusable, portable code have been around for a long time. Over the years, the evolution of modern JavaScript frontend frameworks like React and Angular have further strengthened component-based architecture by adding in core elements like a much nicer API, encapsulation, interoperability, composability and such. Although framework-based components are equipped with these essential enhancements, they only work really well within their ecosystem—lacking the flexibility to be reused outside the framework they were designed for.

Heralding the development of a new application development methodology (analogous to snapping Legos) is Web Components—a set of standard browser APIs that uses custom HTML elements to build framework-agnostic applications.

Web Components provide all the benefits of framework-based components plus offer much more:

✔ Flexibility

✔ Longer life span

✔ Little to no boilerplate overhead code

✔ Reusability within any front-end framework of choice or just plain JavaScript and HTML

3 Key Considerations
in Choosing Components

1 ADOPT COMPONENTS THAT INTEROPERATE AND SAVE DEVELOPMENT COSTS

Interoperability is a vital factor in choosing your web components; evaluate components as a collection.

To speed up application development, enterprises typically select components from a mix of open source options and enterprise-backed solutions—rarely are they developing the entire functionality from the ground up. When selecting components, it is crucial to evaluate them as an entire collection suite—ensuring they play well with each other. Not all critical components are created equal, so evaluating them holistically through a wider lens enables functionality that intertwines nicely. Carefully selected components that integrate together avoid application re-work later in the process, saving valuable time and money.

2 SELECT COMPONENTS WITH FLEXIBLE FEATURES AND LAYOUT

Uniform theming and layout are key to creating consistent, professional-looking applications

Web components should be designed to be flexible—ideally designed to cover all possible use cases. The selection process, as a result, should take into account the different scenarios in which the component could be used. For example, a component used in a data grid or a chart should behave similarly when used outside the grid or chart. Pay special attention to feature consistency and versatility when choosing components. Theming and layout are important parts of every web application as they provide the uniformity and professional appearance in all views. When selecting components, it is vital to check whether they support a certain design system (style guides, layouts) that your organization prefers to use. Theming or layout options should be made dependencies and commonly shared across multiple applications.

3 STANDARDIZE COMPONENTS ACROSS YOUR TEAM

Creating a standard component selection strategy across global teams enables faster time-to-market

Application development in medium to large organizations is spread across multiple teams and locations and to outsourced resources spread around the globe. In such cases, it is particularly important to establish standard internal methodologies to maintain coherent development practices. Blindly leaving web component selection in the hands of outsourcing teams without detailed requirements and clear alignment of scope will inevitably result in inconsistencies, missed expectations and eventually rework. Creating an internal methodology with a library of approved web components that interoperate well and fit in with the organization requirements might be a smarter move. Standardizing the component selection process isn't intended to thwart developer freedom, but as a means to ease the development process, encourage reusability, maintain brand consistency and consequently deploy applications quickly and at scale.

Preparing for the Future with Web Components

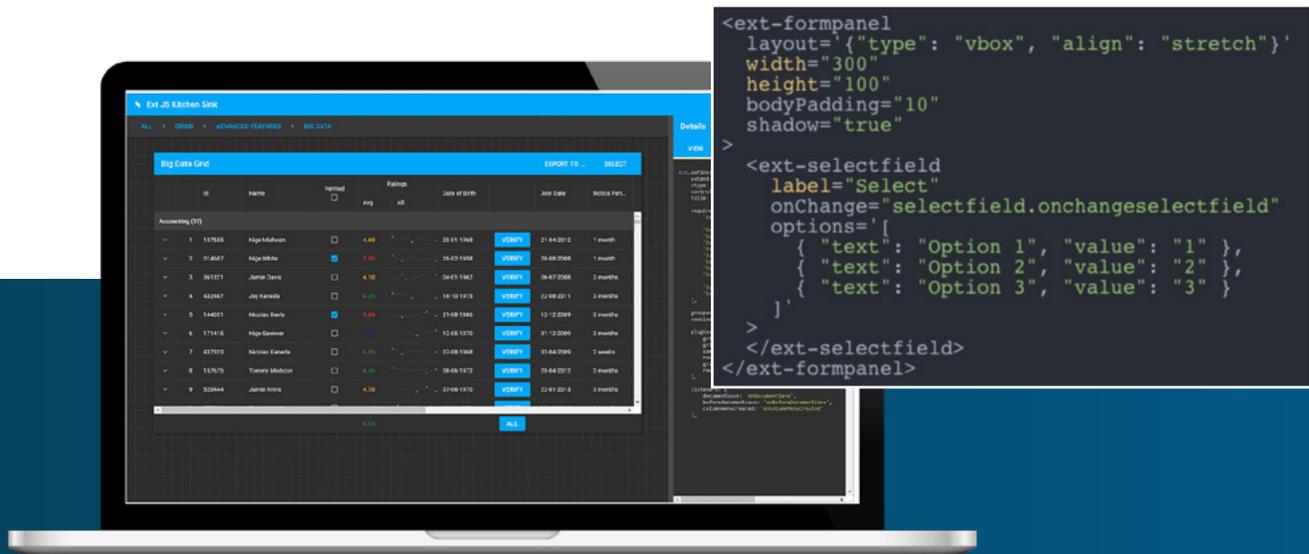## FLEXIBLE METHODOLOGY THAT ADAPTS TO CHANGE FASTER

The landscape of frontend development is pretty dynamic. With technological advancements driven by large corporations and the active and enthusiastic community, new tools, frameworks and libraries are introduced every few years. What is relevant and popular today maybe a thing of the past few years down the line. Switching to a new framework due to business decisions or maintainability issues can result in rebuilding entire applications and eventually turn into an expensive proposition.

Web component technology, built on web standards [governed by open standards World Wide Web Consortium (W3C)] enables organizations to adapt to change quickly—powering them to benefit from new frontend technologies without having to refactor or rewrite entire applications. Components built using web standards can be reused in the new framework seamlessly, saving significant time and resources. Web components are truly flexible, and though companies are rapidly providing support for some shortcomings in terms of browser support, overall they have a longer lifespan, little to no dependencies and a much lower barrier to adoption.

Easier Component Selection with Sencha ExtWebComponents

ExtWebComponents leverage Sencha Ext JS components adapting them to web standards. ExtWebComponents offer a central collection of **140+ feature-rich UI components** (such as complex grids, charts, buttons, forms, calendars, visualization and many more) that interoperate well with one another—plus, standard themes and layout styles for building consistent-looking applications. ExtWebComponents give you the freedom to develop your application with the framework of choice or with no framework at all—without cherry-picking components from different places or worrying about component consistency, interoperability or feature inadequacies. Make component selection easier for your development teams by equipping your global teams with the right components to build web applications faster!


```
<ext-formpanel
    layout='{"type": "vbox", "align": "stretch"}'
    width="300"
    height="100"
    bodyPadding="10"
    shadow="true"
>
    <ext-selectfield
        label="Select"
        onChange="selectfield.onchangeselectfield"
        options='[
            { "text": "Option 1", "value": "1" },
            { "text": "Option 2", "value": "2" },
            { "text": "Option 3", "value": "3" }
        ]'
    >
    </ext-selectfield>
</ext-formpanel>
```

START YOUR FREE 30-DAY TRIAL

Sencha
An Idera, Inc. Company

MORE HELPFUL LINKS:

See It in Action   Read the Getting Started Guides   View the webinar