Which Embedding Method is Right for You?

[Embedded analytics](#) integrates data visualizations from an [analytics platform](#) into another application or web portal. It combines software to appear as one cohesive platform rather than two separate products. Embedded analytics offers convenient, cost-efficient ways to build [scalable](#), accessible reporting, and integrate analytics into an application for end users.

This whitepaper provides an overview of four methods of embedded analytics:

1. Basic Embedding via iframe

2. [Web Components](#)

3. React SDK: Embedding Pre-built Dashboards and Visualisations

4. React SDK: Programmatic Embedding

To help you find the most suitable fit for your use case, we'll walk you through the differences in the level of integration with your application for each method, along with their advantages and disadvantages.

1. Basic Embedding via Iframe

Zero to Basic skill level required

It is the fastest and simplest method.

You create the dashboard in the analytics platform. To embed the dashboard, you copy an HTML snippet that represents a dashboard and paste it into your application or web page.

Integration With Your Application

Iframe has limited integration and flexibility.

- You can only embed the entire dashboard, not a single visualization.• It is not recommended for advanced embedding. The dashboard doesn't become a native part of your application and, therefore, you can't affect how the dashboard is going to be rendered. If you want to edit the dashboard, you need to edit it in the analytics platform first.

- Iframe offers a standard level of interactivity for end-users (drilling, export dashboard as PDF, interactive tooltips, etc.).

- The security level might not comply with your requirements.

This Method is Right for You, If:
- **You want the simplest embedded analytics solution and do not require deep integration between the analytics platform and your application.**

2. Web Components

Basic to Intermediate skill level required

- This method is flexible and easy to set up quickly.

- Web Components are supported by all modern browsers natively. You don't need to use any other ecosystem to be able to integrate data with your application.

- You create your dashboard and visualization in the analytics platform. After that, you can embed your pre-built analytics using Web Components.

Integration With Your Application

▌ Web Components offer a higher level of integration compared to iframe. The method sits between iframe and React SDK.

▌ It allows you to embed both the entire dashboard and a single visualization.

▌ You can't affect how the dashboard is going to be rendered.

▌ The application can listen to the changes in the dashboard, meaning the changes in the dashboard may be reflected in the application's content.

This Method is Right for You, If:
- **You don't want to use iframe due to its inflexibility or security requirements.**
- **You want simple embedding, but you only need to embed a single visualization (iframe only offers embedding of the entire dashboard).**
- **You want a simple but flexible method with slightly deeper integration than an iframe.**
- **If you use Angular, Vue, or any other non-React framework for the host application.**

3. React SDK: Embedding Pre-built Dashboards and Visualizations

Basic to Intermediate skill level required
Not much code is needed, but basic technical knowledge is required.

▌ This method uses live components.

▌ You can embed both entire dashboards and single visualizations.

| Dashboards and visuals are pre-built in the analytics platform, and they're embedded via code provided by SDK.

Integration With Your Application

| The analytics integrates with your application on a deeper level, becoming a native part of it.

| It offers more flexibility in how to code the analytics solution. There's a possibility to affect how the dashboard will be rendered. You can set the changes directly in the embedded solution, not in the analytics platform.

| Embedded visualizations or dashboards can trigger a change in your application or web portal (and the other way around). Despite some limitations, the developer can code such a level of interaction.

This Method is Right for You, If:
- **Your host application is already in React.**
- **You want to embed the whole dashboard and/or visualization on a deeper level, but you don't have special requirements.**

4. React SDK: Programmatic Embedding

Advanced skill level required

| This method uses either permanent components or raw data.

| You can only embed individual visualizations, not the entire dashboard.

| You can create completely custom visualizations. You can easily manage the visualization's characteristics via code (using React SDK, Python, or REST APIs) and define the behavior from A to Z. You don't need to sign into the analytics platform at all.
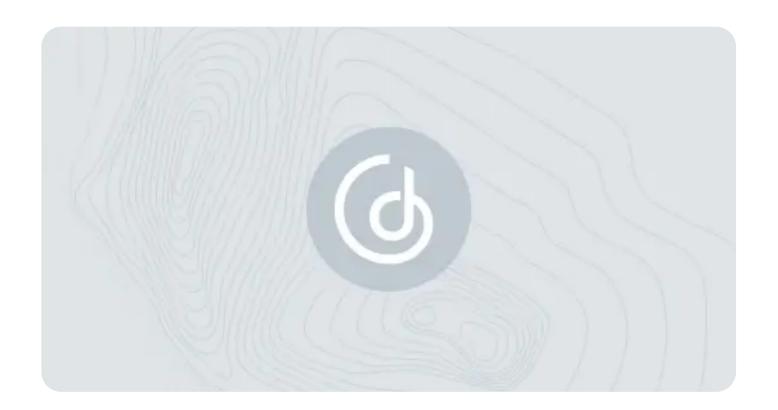
Integration With Your Application

| You can build whatever you want, as this method ensures native integration with the application. The visualization can appear in any shape you imagine and include any button you choose.

| You can reach context-aware visualization. You can code both the look and the behavior of the visualization, and adapt the visualization to the context of your application or website.

| As with the previous method, the visualization can trigger a change in your application. When working exclusively with code, you can set any behavior you want.

| End users can interact with every feature the developer codes. There are no limitations.

This Method is Right for You, If:
- **You need to meet specific requirements. You can create a custom-made, interactive data visualization and reach a high level of interactivity.**

This overview compares all of the embedding methods in terms of flexibility (integration) and ease of deployment.

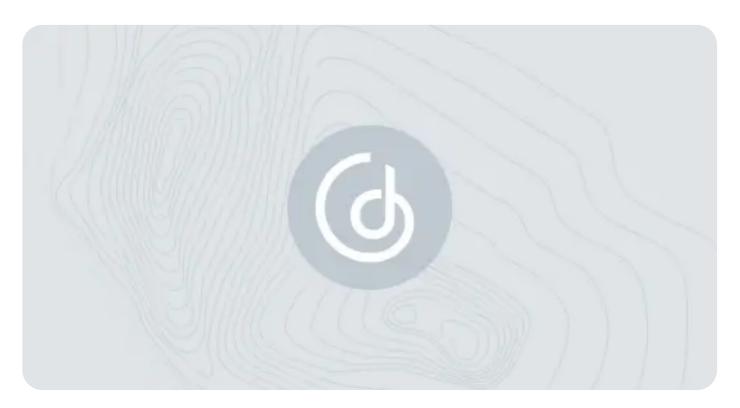## Advantages and Disadvantages of Each Embedding Method

	Advantages	Disadvantages
iframe	• Easy to apply with standard HTML code • Suitable for non-technical users	• Limited flexibility • Cannot support advanced customization or seamless integration • May not support all security requirements
Web Components	• Flexible and easy to set up quickly • Higher level of integration, despite working with minimum code • Web Components supported by all modern browsers natively • High level of security	• Limited API calls • Slightly limited interactivity for end users
React SDK: Embedding Pre-built dashboards and visuals	• Ease of use • Best-in-class integration • Additional flexibility for creating your analytical UI • High level of security	• Suite of pre-built visualizations is limited • Requires front-end development knowledge

React SDK: Programmatic Embedding	• Top-notch performance for skilled developers • High level of security • Highly interactive and customizable for end users	• Requires advanced front-end development knowledge • Time to deploy is average or slow (depends on the requirements)

What Kind of Embedding Is Right for You?

Deciding which method of embedding is right for you depends on your goals and the skillset of your developers. Do you want a simple solution ready within a week? Or do you need to tightly integrate your analytics into your application and create a dynamic and context-aware visualization? Use the flowchart below to help you find the most suitable option for your use case.

Ready To Learn More?

Learn about GoodData's offering for each embedding type in our video How to embed with GoodData. You can also check out our e-book comparing the best-embedded analytics tools.