Architecture Decision Record: chart library toolkit for data visualization using TypeScript and JSON

Primary Objective:
To select an advanced charting toolkit for creating interactive visualizations, focusing on financial data, scientific data, and government data using TypeScript and JSON. The library should provide robust features, flexibility, and be open-source.

Context and Requirements:

1. **Agile Development (High Priority)**: As a startup, quick iteration, prototyping, and flexibility in development are essential. The chart library must allow for fast development cycles.

2. **Chart Types (High Priority)**:

   - **Doughnut Chart**
   - **Radar Chart**
   - **Clustering Process Chart**
   - **Area Chart with Time Axis**
   - **Candlestick Chart**
   - **Nightingale Chart**
   - **Geo SVG Map**

   These chart types are specifically important for visualizing complex datasets, such as financial trends, scientific metrics, and geographical information.

3. **Free and Open Source (High Priority)**: The toolkit should be open-source to avoid licensing costs, provide transparency, and offer flexibility for customization.

4. **Low Importance Criteria**:

   - **Runtime Speed**: While performance is important, it is not a top priority for this decision.
   - **Scalability**: Although scalability is generally important, the immediate need is to build an MVP that may grow over time. Scalability concerns can be addressed later.
   - **Backwards Compatibility**: Not a primary concern for the initial build, as long as the library is modern and actively maintained.

Libraries Evaluated:

1. **Apache ECharts**
2. **Chart.js**
3. **ApexCharts**
4. **AG Charts**
5. **Highcharts**
6. **Carbon Charts**
7. **Layer Cake**
8. **D3.js**

1. **Apache ECharts**

Overview:
Apache ECharts is a powerful, flexible charting library for interactive, customizable visualizations. It provides support for a wide variety of charts and is particularly strong in complex, dynamic visualizations.

Strengths:

- **Advanced Interactivity**: ECharts excels in providing interactive charts, offering features like zooming, panning, and dynamic data updates.
- **Doughnut, Radar, Candlestick, Geo SVG Maps**: ECharts supports many of the required chart types, including doughnut, radar, candlestick, and geographical map visualizations.
- **Free and Open Source**: ECharts is an open-source library, which fits the budget-conscious nature of a startup and provides freedom to modify the code.
- **Flexibility and Extensibility**: Highly customizable, with extensive support for animations, custom visualizations, and advanced charting techniques.

Weaknesses:

- **Learning Curve**: ECharts, while powerful, can have a steeper learning curve due to its flexibility and extensive API.
- **Documentation Complexity**: The documentation is comprehensive but can be overwhelming for developers just starting out with it.

Verdict:
ECharts is highly suitable for the project due to its support for interactive charts, including all the required types like candlestick charts, radar charts, and geo maps. Its open-source nature aligns with the project's need for flexibility and cost-effectiveness.

---

2. **Chart.js**

Overview:
Chart.js is a simple, easy-to-use charting library for building common chart types. It’s known for its simplicity and ease of integration.

Strengths:

- **Ease of Use**: Chart.js is very simple to set up and use, with a minimal learning curve.
- **Open Source**: Chart.js is free and open-source, which is critical for reducing costs.
- **Common Chart Types**: It supports basic charts like doughnut, area, radar, and line charts, which cover most of the primary needs.

Weaknesses:

- **Limited Advanced Charts**: Chart.js does not natively support complex chart types like candlestick charts, geo SVG maps, or clustering process charts. While these features can be added through plugins or customization, it is not as straightforward as with other libraries.
- **Interactivity**: Although Chart.js supports basic interactivity (e.g., tooltips and hover effects), it does not offer as advanced features as ECharts or D3.js.

Verdict:
Chart.js is great for simple, quick projects, but its lack of support for complex chart types makes it unsuitable for a data-heavy application with advanced needs like candlestick charts and geo maps. It's a good choice for prototyping, but for the required chart types, more advanced tools are recommended.

---

3. **ApexCharts**

Overview:
ApexCharts is a modern charting library that provides a variety of chart types and focuses on interactive visualizations with an easy-to-use API.

Strengths:

- **Interactive Features**: ApexCharts offers interactive charts with tooltips, zooming, panning, and updates.
- **Support for Financial and Scientific Charts**: It supports a wide variety of chart types, including candlestick charts, radar charts, and area charts.
- **Ease of Use**: It has a straightforward API and is simple to integrate into a project.
- **Free and Open Source**: ApexCharts offers a free open-source version that is suitable for many use cases.

Weaknesses:

- **Complex Customization**: While it provides many features, the customization options are not as flexible as ECharts or D3.js for highly complex or custom charting needs.
- **Geo Maps**: ApexCharts does not natively support geo maps or clustering process charts, which are required for this project.

Verdict:
ApexCharts is a strong contender due to its ease of use and interactivity, but it falls short on certain advanced chart types, particularly the need for geo maps and clustering charts. It's a good option for simpler charts but lacks some required features.

---

4. **AG Charts**

Overview:
AG Charts is a commercial-grade charting library designed for performance and precision. It is highly suitable for creating financial, scientific, and business dashboards.

Strengths:

- **Advanced Chart Types**: AG Charts supports many advanced chart types, including candlestick charts, area charts, radar charts, and more. It also offers deep integration with other AG-Grid products.
- **High Performance**: It offers excellent performance, especially when dealing with large datasets.
- **Interactivity**: AG Charts supports a variety of interactive features like zooming, tooltips, and dynamic updates.

Weaknesses:

- **Not Fully Free**: While AG Charts offers a free version, the full-featured version is paid, which could be a barrier for startups looking to minimize costs.
- **Complexity**: While the library is feature-rich, it may be overkill for simpler projects and can require more setup and configuration compared to other options.

Verdict:
AG Charts is powerful and feature-rich but may not be the best fit due to its commercial nature and cost structure. Its suitability depends on whether the budget can accommodate paid versions or if open-source alternatives are preferred.

---

5. **Highcharts**

Overview:
Highcharts is a popular charting library known for its wide range of chart types and powerful customization options.

Strengths:

- **Comprehensive Chart Types**: Highcharts supports a wide variety of charts, including candlestick, radar, area, and geo maps.
- **Interactive and Dynamic**: Highcharts provides rich interactive features, including drill-downs, zooming, and panning.
- **Ease of Use**: It has a user-friendly API and good documentation, making it easy to get started.

Weaknesses:

- **Commercial License**: While Highcharts offers a free version for non-commercial use, the commercial license is expensive, which could be a significant drawback for startups.
- **Learning Curve**: Although not as steep as ECharts, the learning curve for Highcharts can still be challenging for beginners.

Verdict:
Highcharts is a feature-rich library, but its commercial licensing makes it less suitable for open-source, cost-sensitive projects. Its comprehensive charting options are a plus, but the licensing issue limits its appeal for this use case.

---

6. **Carbon Charts**

Overview:
Carbon Charts is a charting library developed by IBM, designed for creating visually appealing and highly customizable charts.

Strengths:

- **Customizability**: Carbon Charts allows for extensive customization of chart appearance and behavior.
- **Open Source**: It is free and open-source, which aligns with the project's requirement for budget-friendly solutions.
- **Support for Common Charts**: It supports common chart types like doughnut, radar, and area charts, though it lacks support for more advanced types like geo maps or candlestick charts.

Weaknesses:

- **Limited Advanced Chart Types**: It does not support geo maps, clustering process charts, or candlestick charts, which are essential for the project.
- **Smaller Ecosystem**: Carbon Charts has a smaller community and ecosystem compared to larger chart libraries like ECharts or Highcharts.

Verdict:
Carbon Charts is open-source and customizable but lacks support for the more complex chart types needed for this project. It is more suitable for simpler charting needs.

---

7. **Layer Cake**

Overview:
Layer Cake is a data visualization library designed for creating flexible, layered visualizations.

Strengths:

- **Customizable Layers**: It provides powerful layering options for complex visualizations.
- **Open Source**: It's free and open-source, making it a viable option for budget-conscious projects.

Weaknesses:

- **Limited Documentation**: Layer Cake lacks extensive documentation and community support, making it more difficult to work with compared to more established libraries.
- **Not Built for Charts**: Layer Cake is better suited for non-chart visualizations, so its out-of-the-box charting options are limited.

Verdict:
While interesting for unique visualizations, Layer Cake is not ideal for traditional charting requirements like candlestick charts or radar charts. It's more suited for custom visualizations outside the scope of standard charts.

---

8. **D3.js**

Overview:
D3.js is a powerful JavaScript library for creating data-driven visualizations through HTML, SVG, and CSS.

Strengths:

- **Unmatched Flexibility**: D3.js allows for creating virtually any type of custom visualization, making it highly powerful for advanced and interactive charts.
- **Extensive Features**: It supports all chart types required, including geo maps, clustering charts, and more.
- **Customizable**: The level of customization in D3.js is unparalleled, allowing developers to build highly tailored visualizations.

Weaknesses:

- **Steep Learning Curve**: D3.js has a steep learning curve and is more complex to integrate compared to other libraries.
- **Time-Consuming**: Building charts in D3.js can be time-consuming, especially for common charts like candlestick or doughnut charts.

Verdict:
D3.js is incredibly powerful for advanced, customized charts but is overkill for many typical use cases due to its steep learning curve and development time. It’s best for situations where the other chart libraries do not provide the required level of customization.

Conclusion

After evaluating the libraries based on the project’s needs, **Apache ECharts** stands out as the best option. It supports the full range of required charts, including geo maps, candlestick charts, and clustering charts. It’s open-source, feature-rich, and highly interactive, which aligns perfectly with the project's goals. While **D3.js** offers the most flexibility, its complexity and time investment make it less ideal for a startup looking to iterate quickly. **ApexCharts** and **Chart.js** are good alternatives for simpler projects but lack support for advanced chart types.