

AI-Enabled Big Data Visualization and Decision Support Using Responsive Web Technologies

Kennedy Ayanga

Department of Computer Science, Maseno University, P.O. Box 333 40105, Maseno, Kenya
Email ID: kennedy.ayangal@maseno.ac.ke

Abstract

Decision-makers increasingly rely on dashboards to understand large and fast-changing datasets. Building effective dashboards, however, still depends on analysts choosing the right charts, spotting the important patterns, and designing layouts that work on every device. Artificial intelligence (AI) can take on much of this work. Visualization recommendation systems suggest suitable charts, automated insight discovery surfaces notable patterns, and natural-language interfaces let users ask questions in plain words, which large language models (LLMs) now translate into visualizations. This article proposes an architecture for AI-enabled big data visualization and decision support built on responsive web technologies. It combines seven layers: responsive presentation, AI visualization recommendation, automated insight discovery, natural-language interaction, metadata-driven configuration, workflow-driven decision execution, and autonomous operations. Using a design-oriented approach grounded in ten studies published between 2016 and 2025, the article presents the architecture, the flow from data to decision, and responsive design strategies. It also maps how the reviewed evidence is distributed across the architecture. It illustrates the architecture with a retail inventory decision support case used by regional managers on phones and analysts on desktops. It argues that AI makes visualization faster and more insightful only when its outputs are configurable, verifiable, and connected to governed decisions.

Keywords: big data visualization, decision support, artificial intelligence, visualization recommendation, automated insights, natural language interfaces, large language models, responsive web design, metadata-driven architecture, enterprise workflow

1. Introduction

Visualization turns large datasets into patterns people can see and act on. In most organizations it is delivered through web dashboards viewed on desktops, tablets, and phones. Yet dashboard creation remains labor-intensive. Analysts must decide which fields to show, which chart types to use, and which patterns deserve attention, and then design layouts that work across screen sizes. As data grows in volume and variety, this manual process becomes a bottleneck. Important patterns can go unnoticed simply because no one thought to chart them.

AI offers help at several points. Visualization recommendation systems propose charts suited to the data and the user's goals (Wongsuphasawat et al., 2016; Hu et al., 2019; Dibia & Demiralp, 2019). Automated insight discovery searches large multidimensional datasets for notable patterns such as outliers, trends, and dominant contributors (Ding et al., 2019). Natural-language interfaces let users ask questions such as "show sales by region for last quarter" and receive a suitable chart (Narechania et al., 2021). LLMs extend this capability to a much wider range of questions (Maddigan & Susnjak, 2023). Responsive visualization research shows how charts can adapt to different screen sizes rather than simply shrinking (Hoffswell et al., 2020). Decision support requires more than generating charts. Dashboards must be configurable as business needs change, which a metadata-centric platform architecture supports (Basireddy, 2022a). Insights must lead to governed action through configurable workflows (Basireddy, 2022b). The services behind AI-enabled dashboards must stay responsive under changing load, which autonomous operations agents can help achieve (Sannidhanam, 2025).

This article asks: How can AI-enabled visualization be combined with responsive web technologies and enterprise architecture to deliver effective big data decision support? It contributes a mapping of the evidence base (Figures 1 and 2), a layered architecture (Figure 3), a data-to-decision flow (Figure 4), responsive design strategies, and an illustrative application.

2. Related Work

2.1 Visualization recommendation

Wongsuphasawat et al. (2016) introduced Voyager, which supports exploratory analysis by recommending charts that users can browse. The recommendations are based on the fields they select and on visualization design principles. This lets users explore data broadly without specifying each chart by hand. Hu et al. (2019) proposed VizML, which trains ML models on a large collection of dataset and visualization pairs to predict the design choices people make,

such as chart type and which axis a field goes on. Dibia and Demiralp (2019) presented Data2Vis, which uses a sequence-to-sequence neural network to translate data directly into visualization specifications. Together, these works show that chart design decisions can be learned and automated.

2.2 Automated insight discovery

Ding et al. (2019) introduced QuickInsights, which automatically searches multidimensional data for insights such as outstanding values, trends, correlations, and dominant contributions. It ranks these insights by significance and returns the most important ones quickly. It turns the question "what should I look at?" into a ranked list of candidate findings, each paired with a chart.

2.3 Natural-language interaction

Narechania et al. (2021) presented NL4DV, a toolkit that converts natural-language queries about a dataset into analytic specifications and visualizations. It identifies the fields, analytic tasks, and chart types implied by the question. Maddigan and Susnjak (2023) presented Chat2VIS, which uses LLMs to generate visualization code from natural-language requests. Their results show that LLMs can handle a wide variety of phrasings, including loosely specified ones, but that outputs need checking, because the model may misread a request or produce faulty code.

2.4 Responsive visualization

Hoffswell et al. (2020) studied how visualization designers adapt charts for different screen sizes. They identified techniques beyond simple rescaling: changing layout and aspect ratio, simplifying or changing encodings, reducing the number of data marks or labels, and adjusting annotations and interaction. They also found that authoring responsive visualizations remains laborious. Their work shows that responsive visualization is a design problem in its own right.

2.5 Metadata-driven configuration

Basireddy (2022a) proposes a metadata-centric platform architecture for dynamic configuration, intelligent process automation, operational agility, and sustainable enterprise modernization. In this architecture, business rules, process behaviors, and configurations are

held as metadata that can change without code changes. For AI-enabled dashboards, visualization specifications, key performance indicator definitions, thresholds, insight rules, and access settings can be managed as metadata. Business users can adjust dashboards quickly and consistently, and AI-generated specifications can be reviewed and stored in the same way.

2.6 Decision workflows

Basireddy (2022b) proposes configurable enterprise workflow architecture for digitizing risk management, in which identification, assessment, approval, escalation, and remediation are defined as configurable workflows with audit trails. Connecting dashboards to such workflows lets insights trigger governed actions, such as opening a case, requesting approval, or escalating an issue, with the supporting chart attached.

2.7 Autonomous operations

Sannidhanam (2025) examines autonomous AI agents for cloud infrastructure operations, which monitor systems, diagnose issues, and take corrective actions within defined limits under human oversight. AI-enabled dashboards depend on several backend services, including query engines, insight discovery, and language model services. Autonomous agents can keep these services responsive as usage changes.

2.8 Research gap

Research on recommendation, insight discovery, natural-language interfaces, and responsive design has largely focused on individual capabilities. Less attention has been paid to combining them with enterprise configuration, workflows, and operations into a complete decision support architecture. This article addresses that gap.

3. Research Methodology

The study follows a design-oriented research approach. The problem, that dashboard creation is manual and weakly connected to decisions, was identified from the literature. Requirements were derived from the reviewed studies: automated chart design, automated discovery of patterns, natural-language access, responsiveness across devices, configurability, governed action, and reliable operations. The architecture was designed from ten studies and

demonstrated through an illustrative retail case.

To make the evidence base transparent, each study was assigned to the architecture layer it primarily supports. Figure 1 shows the resulting distribution, and Figure 2 shows the studies by publication year. These figures describe the literature used in this article. They are not performance measurements. The architecture has not been evaluated empirically, which is left to future work.

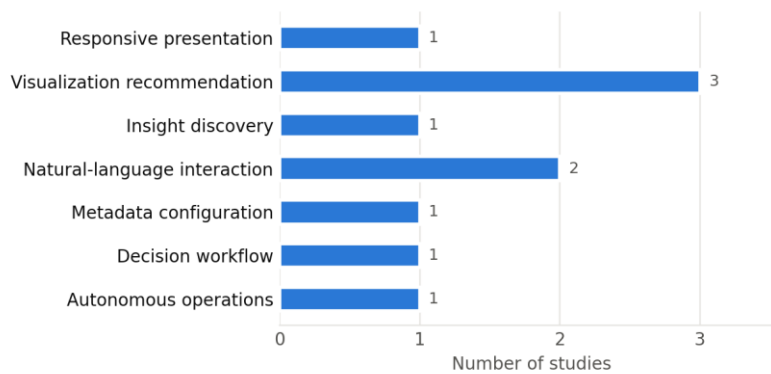


Figure 1. Number of reviewed studies primarily supporting each architecture layer (n = 10)

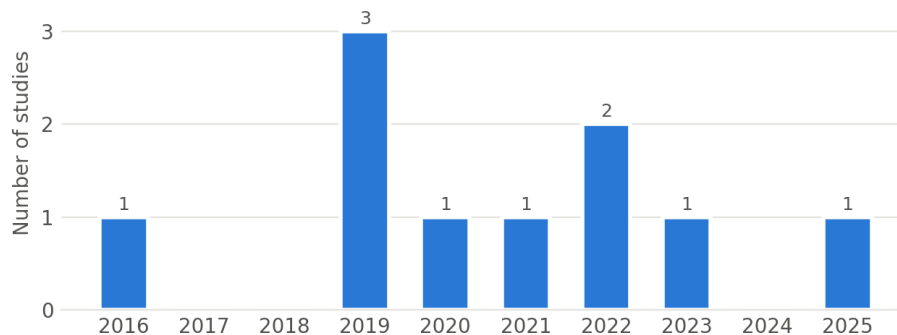


Figure 2. Reviewed studies by publication year (n = 10)

The distribution shows that research on AI visualization recommendation is the most developed strand, while each enterprise integration layer rests on a single study. This imbalance reflects a wider pattern. AI capabilities for visualization have been studied more thoroughly than the architecture needed to use them in organizations.

4. Architecture

The architecture has seven layers (Figure 3 and Table 1).

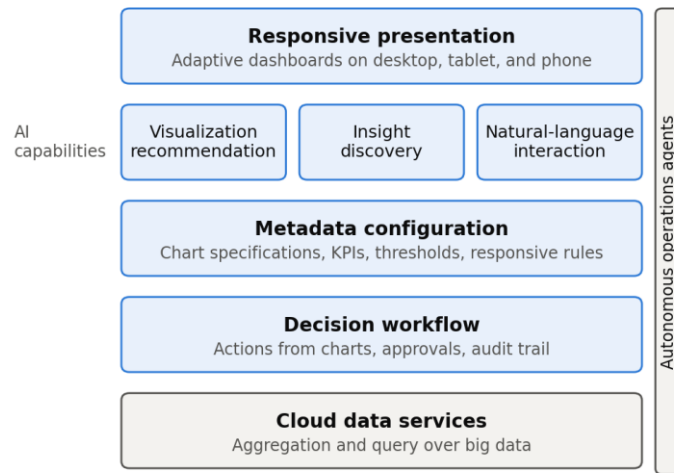


Figure 3. Layered architecture for AI-enabled big data visualization and decision support

Table 1. Layers of the AI-enabled visualization and decision support architecture

Layer	Function	Key techniques	Supporting studies
1. Responsive presentation	Display dashboards that adapt across devices	Responsive grid layouts, adaptive chart designs	Hoffswell et al. (2020)
2. Visualization recommendation	Propose suitable charts for data and tasks	Rule-based and learned recommenders	Wongsuphasawat et al. (2016); Hu et al. (2019); Dibia & Demiralp (2019)
3. Insight discovery	Surface notable patterns automatically	Ranked search for outliers, trends, contributors	Ding et al. (2019)
4. Natural-language interaction	Answer questions in plain language	Query-to-specification toolkits, LLM-based generation	Narechania et al. (2021); Maddigan & Susnjak (2023)
5. Metadata configuration	Store and manage dashboard definitions	Metadata-held specifications, KPIs, thresholds, rules	Basireddy (2022a)
6. Decision workflow	Turn insights into governed action	Configurable workflows triggered from charts	Basireddy (2022b)
7. Autonomous operations	Keep backend services responsive	Bounded monitoring and scaling agents	Sannidhanam (2025)

Responsive presentation. Dashboards are built on a responsive grid, such as that of Bootstrap, so that panels reflow across screen sizes. Following Hoffswell et al. (2020), charts adapt their design on small screens rather than simply shrinking. They show fewer categories, switch orientation, simplify labels, and move detail into expandable views.

Visualization recommendation. When users select data fields or open a new dataset, the recommendation layer proposes suitable charts. It uses design rules for common cases (Wongsuphasawat et al., 2016) and learned models for broader coverage (Hu et al., 2019; Dibia & Demiralp, 2019).

Insight discovery. On a schedule and on demand, the insight layer searches the data for significant patterns and ranks them (Ding et al., 2019). The top insights appear as cards, each with an explanatory chart and a short description.

Natural-language interaction. Users can type questions. Well-structured questions are converted into visualization specifications using toolkits such as NL4DV (Narechania et al., 2021), and more open-ended questions are handled by an LLM (Maddigan & Susnjak, 2023). Generated specifications are validated against the data schema before rendering, and the interface shows how the question was interpreted so users can correct it.

Metadata configuration. All dashboard definitions are stored as metadata (Basireddy, 2022a). These include chart specifications, KPI formulas, thresholds, insight rules, and user permissions. AI-generated charts that users choose to keep are saved in the same form after review. Business owners can then change dashboards without developers, and every change is versioned.

Decision workflow. Each chart and insight card offers actions that start configurable workflows, such as ordering stock, escalating an issue, or requesting approval (Basireddy, 2022b). The chart, its data query, and the insight description are attached as evidence.

Autonomous operations. Agents monitor query latency, insight job duration, and language model service performance. They scale services and restart failed jobs within approved limits and escalate other issues (Sannidhanam, 2025).

5. From Data to Decision

Figure 4 shows how the layers work together, from raw data to a governed decision.

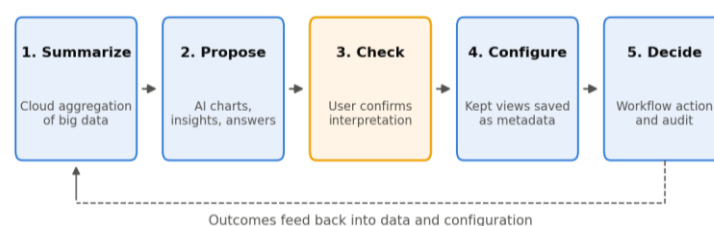


Figure 4. Data-to-decision flow in the AI-enabled visualization architecture

The flow has five stages. First, big data is summarized by cloud query services so that dashboards receive only the aggregates they need. Second, AI generates candidate views through recommendation, insight discovery, and natural-language questions. Third, a human checks these candidates. Users review what the system suggests, confirm how their question was interpreted, and choose which views to keep. Fourth, kept views and rules are stored as metadata, so they become part of the configured dashboard. Fifth, decisions made from the dashboard pass through configurable workflows, and outcomes are recorded. The human check in the third stage is essential. Learned recommenders and LLMs can misread requests or produce misleading charts (Maddigan & Susnjak, 2023), so AI output is treated as a proposal rather than a final answer.

Table 2 summarizes the AI capabilities and the checks applied to each.

Table 2. AI capabilities in visualization and their checks

Capability	Technique	Output	Human or system check
Chart recommendation	Rule-based and learned recommenders	Ranked candidate charts	User selects and adjusts
Insight discovery	Ranked search for significant patterns	Insight cards with charts	User confirms relevance before acting
Natural-language query	NL-to-specification toolkit or LLM	Chart answering the question	Schema validation, displayed interpretation, user correction
Specification generation	Neural and LLM-based generation	Visualization specification or code	Validation before rendering and before saving as metadata

6. Responsive Design Strategies

Drawing on Hoffswell et al. (2020), Table 3 sets out strategies for adapting AI-enabled dashboards to different screen sizes.

Table 3. Responsive strategies for AI-enabled dashboards

Element	Desktop	Tablet	Phone
Layout	Multi-column grid of charts and insight cards	Two-column grid	Single column, most important first
Charts	Full detail, many	Reduced categories	Top categories with "other"

	categories		group, horizontal bars for long labels
Labels and annotations	Full axis labels and annotations	Abbreviated labels	Key values as text above charts
Insight cards	Several cards visible	Scrollable row of cards	Top insight only, others on request
Natural-language input	Persistent query bar	Persistent query bar	Expandable query field, voice input where available
Actions	Buttons beside each chart	Buttons beside each chart	Action menu per card with large touch targets

These strategies are also stored as metadata (Basireddy, 2022a). Each chart specification includes rules for how it adapts at each breakpoint. When the recommendation layer proposes a chart, it proposes responsive variants too, which reduces the authoring effort that Hoffswell et al. (2020) identified as a key barrier.

7. Illustrative Case: Retail Inventory Decision Support

Consider a retail chain with several hundred stores and tens of thousands of products. It produces hundreds of millions of sales and inventory records each year. Regional managers review performance on phones while visiting stores, and central analysts work on desktops.

Insight discovery. Each morning, the insight layer scans the previous day's data (Ding et al., 2019). Its top-ranked insight shows that one product category in one region is selling far above its normal rate while inventory is falling. The insight card shows a bar chart of daily sales against the usual range and a projection of when stock will run out.

Mobile view. A regional manager opens the dashboard on a phone. The layout shows the top insight first, with the key figures as text above a simplified chart showing only the most affected stores (Hoffswell et al., 2020).

Natural-language question. The manager asks, "Which stores in my region will run out of this category this week?" The system converts the question into a chart specification (Narechania et al., 2021) and shows how it was interpreted: category, region, projected stock-out date, this week. The resulting ranked bar chart lists eight stores.

Decision workflow. The manager selects the eight stores and starts a stock transfer request from the chart (Basireddy, 2022b). The request, with the chart and data query attached, is routed to the supply planning team for approval.

Analyst follow-up. A central analyst reviews the case on a desktop. The recommendation layer suggests related views, such as the same category in neighboring regions and a comparison with last year's seasonal pattern (Wongsuphasawat et al., 2016; Hu et al., 2019). The analyst finds a similar early trend in a second region, saves one recommended chart to the regional dashboard as metadata (Basireddy, 2022a), and adds a threshold rule so that future spikes in the category raise an alert automatically.

Operations. As more managers open the dashboard during the morning peak, the operations agent increases query service capacity within its approved limits (Sannidhanam, 2025).

The case shows how AI capabilities reduce the effort of finding and charting what matters, how responsive design serves users on different devices, and how metadata and workflows turn insights into lasting configuration and governed action.

8. Discussion

8.1 Implications

For research, the architecture connects AI visualization research with enterprise concerns: configuration management, decision workflows, and operations. The evidence mapping in Figures 1 and 2 shows that the enterprise integration layers are the least studied, which points to a clear research opportunity. For practitioners, the architecture offers a practical path. Treat AI outputs as proposals that users confirm. Store dashboard definitions and responsive rules as metadata so that they can change without code. Connect every chart to actions through workflows. Design charts to adapt to small screens rather than shrink.

8.2 Limitations and future work

The architecture was demonstrated through an illustrative case and draws on a focused set of ten studies, and the evidence mapping reflects only these studies. Future work should implement and evaluate the architecture, measuring the time to reach insights, the accuracy of recommendations and natural-language interpretation, usability on different devices, and the proportion of insights that lead to decisions. Further research is needed on automatically generating responsive variants of charts, on validating LLM-generated visualizations, and on measuring how metadata-driven configuration affects the speed of dashboard change.

9. Conclusion

AI can make big data visualization faster, more insightful, and more accessible by recommending charts, discovering patterns, and answering questions in plain language. This article proposed a seven-layer architecture that combines these capabilities with responsive presentation, metadata-driven configuration, workflow-driven decisions, and autonomous operations. The data-to-decision flow places a human check between AI proposals and business action. Responsive strategies ensure that dashboards serve users on any device. The retail case showed the architecture in use, from an automated insight to an approved stock transfer. The central conclusion is that AI-enabled visualization supports better decisions when its outputs are verifiable, configurable, and connected to governed action.

References

1. Basireddy, S. (2022a). Pioneering a future-ready metadata-centric platform architecture for dynamic configuration intelligent process automation operational agility and sustainable enterprise modernization. *SAMRIDDHI: A Journal of Physical Sciences, Engineering and Technology*, 14(03), 415–429. <https://doi.org/10.18090/samriddhi.v14i03.27>
2. Basireddy, S. R. (2022b). Digitizing risk management through configurable enterprise workflow architecture. *SAMRIDDHI: A Journal of Physical Sciences, Engineering and Technology*, 14(4), 231–239.
3. Dibia, V., & Demiralp, Ç. (2019). Data2Vis: Automatic generation of data visualizations using sequence-to-sequence recurrent neural networks. *IEEE Computer Graphics and Applications*, 39(5), 33–46.
4. Ding, R., Han, S., Xu, Y., Zhang, H., & Zhang, D. (2019). QuickInsights: Quick and automatic discovery of insights from multi-dimensional data. In *Proceedings of the 2019 International Conference on Management of Data (SIGMOD)* (pp. 317–332).
5. Hoffswell, J., Li, W., & Liu, Z. (2020). Techniques for flexible responsive visualization design. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems* (pp. 1–13).
6. Hu, K., Bakker, M. A., Li, S., Kraska, T., & Hidalgo, C. (2019). VizML: A machine learning approach to visualization recommendation. In *Proceedings of the 2019 CHI Conference on*

Human Factors in Computing Systems (Paper 128, pp. 1–12).

7. Maddigan, P., & Susnjak, T. (2023). Chat2VIS: Generating data visualizations via natural language using ChatGPT, Codex and GPT-3 large language models. *IEEE Access*, 11, 45181–45193.
8. Narechania, A., Srinivasan, A., & Stasko, J. (2021). NL4DV: A toolkit for generating analytic specifications for data visualization from natural language queries. *IEEE Transactions on Visualization and Computer Graphics*, 27(2), 369–379.
9. Sannidhanam, A. H. (2025). Autonomous AI agents for cloud infrastructure operations. *Journal of Contemporary Science and Technology Management*, 1(01), 83–100. <https://jcstm.info/index.php/jcstm/article/view/20>
10. Wongsuphasawat, K., Moritz, D., Anand, A., Mackinlay, J., Howe, B., & Heer, J. (2016). Voyager: Exploratory analysis via faceted browsing of visualization recommendations. *IEEE Transactions on Visualization and Computer Graphics*, 22(1), 649–658.