A Composable Agentic System for Automated Visual Data Reporting

Péter Ferenc Gyarmati*
University of Vienna

Dominik Moritz†
Carnegie Mellon University

Torsten Möller‡
University of Vienna

Laura Koesten§
MBZUAI
Austrian Institute of Technology
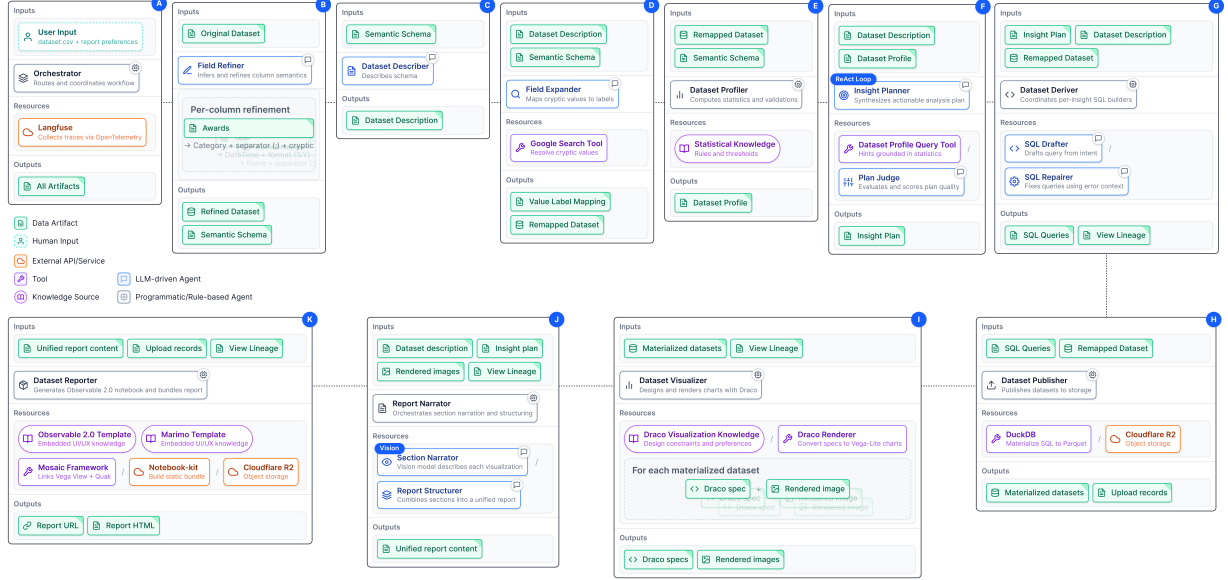University of Vienna

Figure 1: The architecture of our agentic system, a multi-stage pipeline that deconstructs report generation. This hybrid architecture externalizes critical logic to deterministic components like the `Dataset Profiler` and the rule-based `Dataset Visualizer` powered by Draco [21]. The system's granular artifacts are assembled into a dual output: an interactive Observable [17] report with Mosaic [10] for engaging reader exploration, and executable Marimo [1] notebooks that provide deep, per-insight provenance.

ABSTRACT

To address the brittleness of monolithic AI agents, our prototype for automated visual data reporting explores a Human-AI Partnership model. Its hybrid, multi-agent architecture strategically externalizes logic from LLMs to deterministic modules, leveraging the rule-based system Draco [21] for principled visualization design. The system delivers a dual-output: an interactive Observable [17] report with Mosaic [10] for reader exploration, and executable Marimo [1] notebooks for deep, analyst-facing traceability. This granular architecture yields a fully automatic yet auditable and steerable system, charting a path toward a more synergistic partnership between human experts and AI. For reproducibility, our implementation and examples are available at peter-gy.github.io/VISxGenAI-2025.

Index Terms: LLMs, Data Visualization, Human-AI Collaboration, Human-Data Interaction.

1 INTRODUCTION

The proliferation of autonomous agents is reshaping data analysis and some proclaim—despite no evidence—that they will replace human analysts. We explore a **Human-AI Partnership Model**,

---

*e-mail: peter.ferenc.gyarmati@univie.ac.at

†e-mail: domoritz@cmu.edu

‡e-mail: torsten.moeller@univie.ac.at

§e-mail: laura.koesten@mbzuai.ac.ae

where the AI serves as a powerful assistant that produces a complete, initial analysis, and the human expert acts as a strategist and supervisor. The goal is to move beyond the paradigm of a brittle, black-box agent toward a more explainable, steerable, and collaborative workflow.

This partnership model is motivated by a core challenge in data analysis. Even a technically flawless agent may fail if it perfectly executes the *wrong task*. Human intent is often ambiguous at the outset and evolves as insights emerge. A minor misalignment can render an agent's entire output useless, forcing costly and unstable regeneration that erodes user trust and agency. Genuine human–AI partnership requires systems explicitly designed to address this challenge: providing auditable, high-quality outputs while enabling domain experts to iteratively refine them. To investigate this model, we developed a prototype agentic system for automated visual data reporting[1]. The design of our implementation was guided by four high-level principles, adapting established guidelines for human-AI interaction [2] to prioritize human agency alongside automation [9].

Explainability & Trust. For an expert to trust and work with an AI, the system's reasoning should be transparent, and its data transformations should be fully traceable to their source. **Composability & Modularity.** Instead of relying on a single model's embedded knowledge, a robust system should be highly composable. It should leverage specialized tools for established best practices, externalizing critical knowledge into interpretable, often deterministic modules. This design improves future-readiness, as components can be updated without redesigning the core logic. **Granularity & Itera-**

---

[1]Our workshop challenge submission is available online.

tive Outputs. The analytical process is iterative. Its outputs should therefore be granular, deconstructed artifacts rather than an indivisible monolith. This enables "surgical modification" of a single component without a full, resource-intensive regeneration, supporting a more efficient and sustainable workflow. **Interoperability via Open Standards.** To facilitate seamless human intervention, a system should "speak the same language" as the expert community. Using common open standards makes its outputs useful and editable within a user's existing toolchain, facilitating collaboration.

2 SYSTEM DESIGN AND IMPLEMENTATION

Our prototype is a multi-agent system implemented using DSPy [12]. As Figure 1 shows, the main `Orchestrator` **A** coordinates a workflow of specialized, hybrid agents that combine LLM reasoning with deterministic logic. It accepts as input a dataset URI and high-level user intent, such as the overall goal and desired length of the report. To ensure full audibility, the `Orchestrator` collects and stores detailed execution traces of all agent interactions in Langfuse [13], an external observability service.

1. Data Understanding. The pipeline begins with data preparation: the `Field Refiner` **B** cleans the data and infers column semantics before the `Dataset Describer` **C** and `Field Expander` **D**, the latter augmented with a web search tool, create a semantic schema and resolve cryptic codes (e.g., 'BP' to 'Best Paper Award', 'C' to 'Conference Paper'). This phase culminates in the programmatic `Dataset Profiler` **E**, which computes a comprehensive statistical profile. This externalization of statistical truth into a deterministic module is a core tenet of our **Composability** principle.

2. Analysis and Materialization. With a clean dataset, the `Insight Planner` **F** operates within a Reasoning and Acting (ReAct) loop [19], using a dedicated tool to retrieve hints from the statistical profile and ground its planning in data. For each planned insight, the `Dataset Deriver` **G** then drafts and, if necessary, repairs a DuckDB SQL query [18], also outputting semantic roles for the generated fields. Adhering to our **Granularity** principle, the programmatic `Dataset Publisher` **H** executes each query and stores the resulting materialized dataset as a versioned Apache Parquet file in Cloudflare R2, an S3-compatible object store [3, 5].

3. Visualization. We avoid monolithic LLM-based chart design, a token-intensive approach whose recommendations only moderately align with human preferences and remain widely inconsistent [20]. Instead, we use LLM-generated metadata from prior stages to inform a deterministic visualization process. The semantic roles from the `Dataset Deriver` **G** and the analytical task from the `Insight Planner` **F** are used to programmatically construct a partial Draco specification that defines *what* to visualize [21]. The `Dataset Visualizer` **I** then delegates the task of determining *how* to visualize it to Draco's rule-based solver. Based on its computational analysis of the data's properties (e.g., cardinality, skew), Draco can recommend complex, appropriate charts like a faceted scatterplot with a symmetric logarithmic scale–a nuanced design decision difficult for a pure LLM approach. This provides another advantage: if a recommendation is suboptimal, its logic is transparent. We can systematically improve future results by adjusting Draco's knowledge base, a key benefit of a white-box system.

4. Reporting and Publication. The final stage packages all artifacts for human interaction. A vision-language model, coordinated by `Report Narrator` **J**, generates textual descriptions for each base64-encoded chart image and structures them into a cohesive narrative. The `Dataset Reporter` **K** then assembles these with granular artifacts produced by upstream agents. Because all these components are decoupled from presentation, they are modality-agnostic and can be composed into diverse formats, from an interactive web report to a static PDF. Our system demonstrates this flexibility by programmatically constructing a dual output from these same artifacts: JavaScript source for an interactive Observable 2.0 report [16], which a self-hosted Node.js service [15, 17, 8] builds into a static website, and a Python-based executable Marimo notebook [1] for deep, analyst-facing traceability. The Observable report uses Mosaic [10] to link each visualization with its Quak data table [14], an approach inspired by narrative visualization [11] that combines the AI's authored insights with interactive reader-driven exploration. The Marimo notebook, in turn, serves as an executable trace for a single insight, revealing the precise data transformation and visualization logic. This dual-output system provides distinct entry points for expert interaction and targeted modification.

By deconstructing the workflow into well-scoped tasks, our modular design allows for a "right-size" model strategy [4]. Smaller, faster models like Gemini 2.5 Flash [6] handle the majority of tasks, reducing overall latency, while the more powerful Gemini 2.5 Pro [7] is reserved for the most demanding functions: generating complex SQL **G** and authoring descriptive visual narratives **J**. This design makes such systems more future-ready, allowing models to be swapped individually and supporting the use of open-weight, self-hosted alternatives.

3 DISCUSSION

Testing on eight diverse datasets demonstrates our system's generalizability, producing meaningful reports without any domain-specific tuning[2]. The value of our Human-AI Partnership model is best illustrated by the distinct workflows it enables: systemic steerability for the analyst creating the report, and open-ended exploration for the reader consuming it. **Steerability for the Analyst.** An early report featured a weak insight, faceting a scatterplot by the highly skewed `GraphicsReplicabilityStamp` field[2]. Instead of fragile prompt engineering, our architecture supports a durable fix. Using execution traces, an analyst can pinpoint that the `Insight Planner` **F** was misled by a hint from its `Dataset Profile Query Tool`. The analyst then intervenes by modifying the tool's logic, adding a rule to only suggest fields for segmentation that exceed a statistical diversity threshold. This dataset-agnostic refinement permanently improves the system's knowledge base, preventing similar flawed insights in all future runs. **Explorability for the Reader.** The partnership extends to the reader. Our reports use Mosaic [10] to enable exploration beyond the AI's authored narrative. A static chart of downloads vs. citations is enhanced with a linked data table; readers can answer complex ad-hoc questions (e.g., "What about VIS papers in the last five years?") by cross-filtering with attributes like conference and year from the linked data table, which are *not* encoded in the main visualization.

This dual capability–deep, systemic refinement for the analyst and open-ended exploration for the reader–embodies our design philosophy. A successful partnership stems not from demanding perfection from the AI, but from ensuring its outputs are both deeply auditable and adaptable for the expert and flexibly explorable for the reader.

4 CONCLUSION

Our prototype demonstrates a Human-AI Partnership model where the agent acts as a transparent and auditable assistant, not a perfect oracle. Our design instantiates this philosophy through a curated, interoperable toolchain. We leverage Draco for principled, white-box visualization design. The final outputs then serve two distinct purposes: an interactive Observable report powered by Mosaic for reader-driven exploration, and deep, executable Marimo notebooks that provide the analyst with full traceability. By providing the full report source code, this granular architecture grants experts agency for targeted refinement. Designing for human agency is how we can transform AI into a genuine partner for trustworthy data-driven discovery.

---

[2]All discussed reports are available for exploration here.

REFERENCES

[1] A. Agrawal and M. Scolnick. marimo - an open-source reactive notebook for Python. https://github.com/marimo-team/marimo, aug 2023. Apache-2.0 License. 1, 2

[2] S. Amershi, D. Weld, M. Vorvoreanu, A. Fourney, B. Nushi, P. Collisson, J. Suh, S. Iqbal, P. N. Bennett, K. Inkpen, J. Teevan, R. Kikin-Gil, and E. Horvitz. Guidelines for Human-AI Interaction. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*, CHI '19, pp. 1–13. Association for Computing Machinery, New York, NY, USA, 2019. doi: 10.1145/3290605.3300233 1

[3] Apache Software Foundation. Apache Parquet. https://parquet.apache.org/. [Online; accessed August 17, 2025]. 2

[4] P. Belcak, G. Heinrich, S. Diao, Y. Fu, X. Dong, S. Muralidharan, Y. C. Lin, and P. Molchanov. Small Language Models are the Future of Agentic AI, 2025. doi: 10.48550/arXiv.2506.02153 2

[5] Cloudflare, Inc. Cloudflare R2 Documentation. https://developers.cloudflare.com/r2/. [Online; accessed August 17, 2025]. 2

[6] Google Cloud. Gemini 2.5 Flash — Generative AI on Vertex AI — Google Cloud. https://cloud.google.com/vertex-ai/generative-ai/docs/models/gemini/2-5-flash, 2025. [Online; accessed August 17, 2025]. 2

[7] Google Cloud. Gemini 2.5 Pro — Generative AI on Vertex AI — Google Cloud. https://cloud.google.com/vertex-ai/generative-ai/docs/models/gemini/2-5-pro, 2025. [Online; accessed August 17, 2025]. 2

[8] P. F. Gyarmati. observable-notebook-builder: REST API for building Observable notebooks using @observablehq/notebook-kit. https://github.com/peter-gy/observable-notebook-builder, 2025. MIT License. 2

[9] J. Heer. Agency Plus Automation: Designing Artificial Intelligence into Interactive Systems. *Proceedings of the National Academy of Sciences*, 116(6):1844–1850, 2019. doi: 10.1073/pnas.1807184115 1

[10] J. Heer and D. Moritz. Mosaic: An Architecture for Scalable & Interoperable Data Views. *IEEE Transactions on Visualization and Computer Graphics*, 30(1):436–446, 2024. doi: 10.1109/TVCG.2023.3327189 1, 2

[11] J. Heer and E. Segel. Narrative Visualization: Telling Stories with Data. *IEEE Transactions on Visualization & Computer Graphics*, 16(06):1139–1148, Nov. 2010. doi: 10.1109/TVCG.2010.179 2

[12] O. Khattab, A. Singhvi, P. Maheshwari, Z. Zhang, K. Santhanam, S. Vardhamanan, S. Haq, A. Sharma, T. T. Joshi, H. Moazam, H. Miller, M. Zaharia, and C. Potts. DSPy: Compiling Declarative Language Model Calls into Self-Improving Pipelines, 2023. 2

[13] Langfuse. Langfuse: Open source LLM engineering platform. https://github.com/langfuse/langfuse, 2025. MIT License. 2

[14] T. Manz. quak: A Scalable Data Profiler. https://github.com/manzt/quak, 2025. MIT License. 2

[15] Node.js contributors. Node.js javascript runtime. https://nodejs.org/, 2009. [Online; accessed August 17, 2025]. 2

[16] Observable, Inc. Observable Notebooks 2.0 Technology Preview. https://observablehq.com/notebook-kit. [Online; accessed August 17, 2025]. 2

[17] Observable, Inc. Observable Notebook Kit: A command-line tool for building static sites from Observable Notebooks. https://github.com/observablehq/notebook-kit, 2025. ISC License. 1, 2

[18] M. Raasveldt and H. Mühleisen. DuckDB: An Embeddable Analytical Database. In *Proceedings of the 2019 International Conference on Management of Data*, SIGMOD '19, pp. 1981–1984. Association for Computing Machinery, New York, NY, USA, 2019. doi: 10.1145/3299869.3320212 2

[19] Shunyu Yao and Jeffrey Zhao and Dian Yu and Nan Du and Izhak Shafran and Karthik Narasimhan and Yuan Cao. ReAct: Synergizing Reasoning and Acting in Language Models, 2023. 2

[20] H. W. Wang, M. Gordon, L. Battle, and J. Heer. DracoGPT: Extracting Visualization Design Preferences from Large Language Models. *IEEE Transactions on Visualization & Computer Graphics*, 31(01):710–720, jan 2025. doi: 10.1109/TVCG.2024.3456350 2

[21] J. Yang, P. F. Gyarmati, Z. Zeng, and D. Moritz. Draco 2: An Extensible Platform to Model Visualization Design. In *2023 IEEE Visualization and Visual Analytics (VIS)*, pp. 166–170, 2023. doi: 10.1109/VIS54172.2023.00042 1, 2