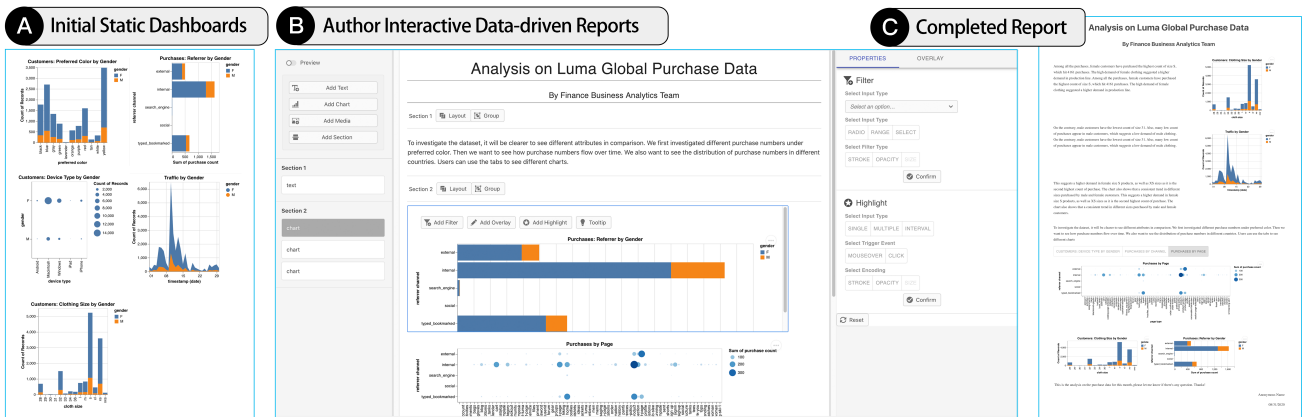# CODAS: Integrating Business Analytics and Report Authoring

Z. Zhang[2], S. Malik[1], S. Guo[1], J. Hoffswell[1], R. Rossi[1], F. Du[1], E. Koh[1]

[1] Adobe Research, San Jose, California, United States
[2]Information School, University of Washington, Seattle, Washington, United States

Figure 1: *An overview of the report authoring workflow: (A) Explore data using initial static dashboards from business analysts; (B) Author interactive data-driven reports using* CODAS*; (C) Share the final report with stakeholders using a shareable link.*

Abstract
Business analysts create rich dashboards to find data insights and subsequently communicate these findings with data-driven reports that combine visualization screenshots and descriptive text. Conventional analytics reports convey findings statically and passively, which suffers from limited interactivity and adaptability to data changes. There is therefore a need to facilitate authoring of interactive *reports in business analytics. To better support the needs of business analysts, we developed* CODAS: *a report authoring tool that allows analysts to transform dashboards into interactive, web-based reports through a no-coding user interface and a workflow that is compatible to business analysts' existing data analytics pipelines.* CODAS *supports authoring multiple levels of interactions, organizing story elements, and generating the final artifact. Through our case studies with two expert analysts, we discuss the usefulness of our system and report our findings on analysts' report authoring workflow. Our findings suggest that* CODAS *enables business analysts to create interactive, data-driven reports comfortably, and can complement their exisitng data analytics workflow without extra learning effort.*

1. Introduction

During exploratory data analysis (EDA), business analysts (BAs) create rich, interactive dashboards to better understand data, find trends, and explain insights. Though these dashboards are sometimes the final artifact of the analysis process, analysts often create secondary *data-driven reports* to present results to their stakeholders. The usual format for data-driven reports include illustrated documents and emails, blog posts, infographics, or presentation slides.

To facilitate this type of communication, some tools exist for assisting data analysts in authoring data-driven reports [noa16, Sof10c, Sof10a]. However, the features of these tools are often not sufficient to meet the needs of business analysts. In particular creating shareable interactive data stories is not well supported in tools like Vega [SMWH16, SWH14] and Idyll [CH18] that require technical expertise (i.e., programming skills) that most business analysts do not have. Other more accessible tools for business analysts like Tableau Stories [noa16, Sof10f] and PowerBI [Sof10a] are also

limited in integrating some of the key interactivity and narrative layouts required in report authoring in business analytics settings.

This paper explores how we can better integrate report authoring into analysts' data analysis pipeline to support the effortless curation of data-driven reports. To this end, we contribute CODAS: a web-based prototype for authoring data-driven reports for business analysts. CODAS primarily accepts Vega [SMWH16, SWH14] charts as input and allow business analysts to curate interactive data stories with a graphical user interface. Analysts can then add charts exported from other interactive visualization dashboards (e.g., [WQM*17, noa16, Sof10e]), arrange these charts together with other interactive components for supporting narrative storytelling, and edit the layout and style to produce the final report. Analysts can also add new interactions like highlights and tooltips to visualizations directly in CODAS. Finally, CODAS allows analysts to easily export a shareable version of the report for distribution.

We evaluate CODAS with two expert case studies to demonstrate the utility with respect to the reporting needs of analysts. The experts were generally positive about CODAS and praised its unique value in empowering analysts who regularly create and share interactive data stories. The experts also suggested potential future work such as the templatization of annotations and captions to better serve analysts when regularly updating reports with new data.

2. Related Work

Data-driven visual storytelling is a popular communication format in fields like journalism, blogs, and instructional videos [SH10], and has structural similarities to data-driven reports in business contexts. Lee et al. [LRIC15] outlined the authoring process of data-driven storytelling with three stages: (i) identifying facts or insights, (ii) creating a narrative, and (iii) presenting the story. Kosara et al. [KM13] note that while the first stage is well supported by commercial tools (e.g., Tableau and Power BI) and research systems (e.g., Voder [SDES18], Data Voyager 2 [WQM*17], and Lyra [SH14]), such tools fall short in supporting the second and third steps. Some recent advances aim to facilitate the second step by creating references between text and charts [MZJS18, BLE18, LZK*21], but mostly focus on cases where charts and texts are already prepared. In the process of report authoring, however, the content of charts and texts are dynamically changing, which makes it not feasible to apply existing automatic techniques for linking.

While most visual analytics tools lack support for creating narrative structures, practitioners have used a wide range of tools for their needs. For example, while Tableau and PowerBI provide interfaces to display many interactive charts, there is no hierarchy or ordering that is necessary for structuring a report. Tableau Stories allows people to extract charts, add captions, and reorder story pieces to create the narrative structure. However, Tableau Stories only supports stepper layouts, while more diverse types of text-chart layouts (e.g., linear, scrollytelling, etc.) are needed in data reporting. Thus, analysts often need to turn to document editing tools (e.g., MS Word, PowerPoint) for more nuanced report creation. Furthermore, researchers, analysts, and journalists are rapidly adopting statistical tools (e.g., Excel, SPSS) and computational notebooks (e.g., Jupyter) to create and share their analyses. However, these tools are limited at supporting both exploration (i.e., stage 1

of the data-driven storytelling process) and narrative explanation (i.e., stage 2) in a single document [RTH18, KRA*18].

There are also tools focused on improving the authoring experience of data-driven reports like Idyll Studio [CVTH21], Quadrigram [Sof10c], and Piktochart [Sof10b]. However, there remain gaps between the unique needs of business analysts and the capabilities of these tools. For example, most of these tools require analysts to recreate their charts directly in the respective tools. However, analysts often already have mature visual analytics tools they are familiar with and would like to use these existing approaches for creating and sharing relevant insights. In contrast, CODAS complements existing tools and the workflow used by analysts with a goal of optimizing their experiences with minimal changes. The goal of this work is not to have a competing system against commercial visual analytics tools (e.g., Tableau, Power BI), but to explore the idea of reusing charts from well-defined frameworks to make the data story authoring process more accessible to business analysts.

3. The CODAS System

CODAS is a web-based prototype to support report-authoring. Users start with exploratory analysis dashboards, then leverage CODAS by combining storytelling elements (like text, interactive charts, and other media), authoring multiple levels of interactions, organizing story elements, and generating the final artifact. Note that details of the views and live demo of the interactions introduced in this section can be found in the supplemental video.

3.1. Input and Preprocessing

We split the input into two parts: the dataset (as a CSV) and a set of Vega-Lite [SMWH16] charts (as JSON files). The system preprocesses the chart specifications by parsing the interactions and adjusting them to fit with our standard format. This step prepares the charts for annotations, tooltips, filters, highlighting, and responsive layouts. Note that we are using Vega-Lite because it is supported by a variety of data analytics tools and can be simply used as an input to demonstrate the capabilities of CODAS. Vega-Lite provides a declarative specification for quickly creating visualizations and many GUI tools exist to create Vega or Vega-Lite specifications (e.g., Lyra [SH14] and Data Voyager 2 [WQM*17]), which can then be imported into CODAS. While Vega-Lite is a straightforward way to showcase the functionality of our proposed system, we can easily adapt our approach to support other visualization languages.

3.2. Arranging the Storyline

CODAS contains a visual data storyline that shows an overview of the report in a condensed format. From this overview, users can add, remove, edit, and rearrange *story elements*, including text, charts, media (images and video), and section breaks (Fig. 1B, left panel).

Sections are containers for story elements and is the basic unit for defining interactions between elements. To make interaction design easier, we designed section-level interactions, which includes grouping interactions among charts and layout interactions between charts and text. From the storyline, users can drag-and-drop elements to rearrange their position within and between sections. Changes are rendered immediately in the main story panel.

3.3. Developing the Story

The *main story panel* comprises the main view for authoring the story narrative and allows users to edit individual story elements.

Text and media. We integrate an open-source "what you see is what you get" (WYSIWYG) system [Pur21] to provide a rich text editing capabilities including formatting, hyperlinking, and attachments. The narratives of a section are linked to the visualization in the same section, which helps creating various layouts of the report.

Charts. We use the *react-vega* [Sof10d] component to render visualizations in the main editing panel. We designed a toolbar to support users in authoring interactivity. All the authored interactions appear as a list of properties in the panel (Fig. 1B, right panel).

Authoring Chart Interactions. To maintain interactivity when transferring dashboard charts into data-driven reports, we developed a Vega specification parser and an easy-to-use model with a WYSIWYG toolbar that allows users to author chart-level interactions.

Adding Filters & Highlights. Filtering and highlighting are popular features preferred by business analysts. Users can add multiple filters or highlights to one chart, and the added filters will appear in the Properties panel as a list of properties. For each filter, CODAS provides three simple inputs : (1) an attribute in the dataset to filter, (2) the input type from a checkbox (for boolean values), range (for numeric values), selections or radio inputs (for nominal values), and (3) encoding type from size, stroke width, or opacity. Similarly, CODAS also provides three inputs for highlighting : (1) input type from single, multiple, or interval selection, (2) trigger event from mouseover or click, and (3) encoding type which is the same as filter. The supported filter and highlight aggregations are a subset of what Vega supports as query widgets.

Creating Overlays. Static charts with user-generated overlays are also popular in data stories. Analysts tend to add arrows, links, or self-defined annotations to highlight insights manually. The draft panel enables users to add, edit, and delete overlay annotations freely by providing PhotoShop-style components like text, pencil, line, rectangle, circle, selection tools, among others. After users have finished creating overlaying annotations, the CODAS system will adapt the changes and reflect the final artifact in the main panel.

Adding Tooltips. Similar to highlights, users can add tooltips to help the audience view exact values from the data. CODAS provides the tooltip types supported by Vega. Tooltips are generated based on different factors, including the encoding, the underlying data point, and the `tooltip` channel. Users can select from one of the three tooltip types: (1) *display data*, where the audience will see the relevant data shown in the chart, (2) *all data*, where the audience will see all the data associated with this datum in the dataset, (3) *value only*, where the audience will only see the current data value.

Layout Styles. Another category of interaction between charts and other elements is the layout. Different layout styles could emphasize different relationships between charts and story elements, link different components, or serve as navigational cues to other elements. Layout style takes effect as a section-level attribute, which means that all the elements in one section will be rendered according to the same layout parameters. The CODAS system supports four types of layouts that have been increasingly described in literature [VB18,SCBL21]: (1) *inline*, the default style that all elements will be in a linear mode; (2) *aside*, all charts will appear to the side of the paragraphs, which indicates a parallel relationship between texts and charts; (3) *aside-fixed*, a similar layout to *aside*, but with the charts fixed on the screen like a reference as text flows alongside; (4) *overlay-text*, where the charts appear as fixed backgrounds as the text paragraphs scroll down, which is typical when the charts are too complicated to be put aside, like a world map visualization.

Chart Grouping. The CODAS system supports another primary type of interaction among the charts, of which we use the term "grouping." Grouping also takes effect as a section-level interaction, which means all the charts in one section will be grouped accordingly. The CODAS system supports two types of grouping: (1) *toggled tabs*, an integrated group of visualization charts where users can use toggle buttons to switch between similar or related views; (2) *dashboards*, a resizable and repositionable set of coordinated views, which may contain different subsets of the same data and facilitate comparison across subsets. Furthermore, grouping and layout interactions could be applied together to one section. In this case, all the charts will be combined into a chart group, and it will become a new story element. The CODAS system will render the new element in different layout styles accordingly. By default, CODAS renders the assembled story elements in linear order.

Previewing and Rendering. Like most authoring and creation tools, we provide a preview function for business analysts. Analysts can preview the current artifact at any point during the authoring process. When the user is finished, they can export the data-driven report as a shareable link, which can be easily viewed in any native browser. CODAS uses a self-defined data structure that stored all needed interactivity, grouping, and layout styles to reproduce the data story in JSON files. The JSON file contains a sequence of specifications for different sections of the report in order. For each section, the specification records the layout of the section (e.g., inline, aside, aside-fixed and overlay-text), the content of the section (e.g., plain texts, media, and Vega-spec for the visualization charts), and the style of the section(e.g., width, height, the layout of the contents, etc.). For each visualization chart, there is also configuration on the interactions user can put on the charts (e.g., filtering and highlighting), and the style changes (e.g., stroke, size, opacity differences) after the interaction.

By downloading and sharing this file, the report can be restored in the system for collaborators to re-edit together with the original input data. This method could apply to situations where analysts iterate the reports alongside other team members. Also, with the data of each chart being separated from the JSON file, it gives a possibility of automatic data refreshing for periodic report generation by simply updating the external data file and applying the layout styles and chart specifications that are defined in previous reports.

4. Case Studies

To solicit feedback and reflections on the usability, potential usage, and future improvements of CODAS, we conducted two case studies with experienced business analysts. The analysts shared a dashboard that they wanted to create a report from ahead of time which was imported into CODAS. Each participant was trained on the features of the system, then asked to create a report using their own data and visualizations using a think-aloud protocol.

4.1. Scenarios

The first participant (P1) was an operations demand analyst responsible for managing and tracking spending across his organization and providing insights to the leadership team. P1 makes quarterly reports about target metrics, such as pacing and budget forecasting. His role also included consulting with other teams to optimize their analysis processes. He typically deliver the insights through static reports (e.g., PowerPoints, blogs), sometimes interactive PowerBI dashboard. His dataset consisted of spending and budget details for 791 teams in one organization. Each team had data from one quarter in 2020 about committed budget, actual spending, spending type, and metadata about the team, such as reporting executive and organization. The original dashboard he created with PowerBI consisted of five visualizations across three tabs, including a grouped bar chart, table summary display, and three bar charts.

The second participant (P2) was an analyst with seven years of experience in marketing insights. P2 supports multiple business units in meeting their visual reporting needs, from dashboards (e.g., Tableau, PowerBI, Analytics, SAS) to standalone reports (e.g., PowerPoints, Infographics, Word Documents). In this case, P2 had created a dashboard with Power BI to share the results of a customer satisfaction survey for a software company. The dataset consisted of over 18,000 customer responses about satisfaction related to a set of software products. Dimensions in the dataset included customer demographics such as gender, age, and job type (e.g., professional, student), as well as responses to ten five-point Likert questions about the product buying and usage experience. The shared dashboard consisted of ten visualizations: three bar charts, four donut/pie charts, a treemap, and two gauge charts.

4.2. Feedback and Takeaways

P1 and P2's overall feedback about CODAS was positive. P1 liked that it was easy to add story pieces (both chart and text) in the system and quickly arrange the sections to form a narrative. He constantly had the need of incorporating explainations to the visualization through annotations and highlights and the corresponding narratives. In particular, P1 explained that *"It's simple in a good way. The simplicity of adding things to the report and moving things around is really straightforward."* He also noted that CODAS would be useful in many use cases, especially when building regularly recurring reports: *"this would be really valuable to do once a quarter as a post-mortem to describe what the VP spent, with more granular insights [or] weekly business reports. I currently build everything in Excel, and put it into PowerPoint manually, for each spend leader."* Though we offered the ability to add other media, such as images or video, P1 said he would not use them for his reports.

P1 also emphasized that data is not static and generally changes over time, so analysts need to periodically update the reports; elaborating on this point, P1 explained that this updating is currently a manual process in his existing workflow: *"I have to refresh the spreadsheet in the dashboard every week."* In contrast, CODAS decouples the data from the visualizations so the data can be updated independently or automatically refreshed.

P2 similarly felt that CODAS was useful and showed a lot of potential. He emphasized the layout particular useful, as one of the pain point in creating report to him was arranging the flow of the story. In particular, P2 found the "dashboard" layout especially helpful, noting that even in dashboarding tools, it is difficult to flexibly resize and arrange charts. In terms of the final data story, P2 said it would be useful for replacing live presentations, for which he current uses PowerPoint or similar tools. Interestingly, P2 noted that it is particularly important to be able to directly edit the charts outside of the dashboard because the final report might not require the same amount of detail due to the additional text.

The main improvement suggested by P2 was templatized annotations. For example, P2 wanted to be able to author an annotation around the "maximum" peak, that could then be updated based on the data. P1 similarly wished to be able to templatize the captions in order to provide personalized insights across spender leaders and quarters. For example, he might want to call out the top spend categories for each audience member in a caption. To make this feature a reality, it would also be necessary to allow CODAS to compute calculated values, such as pacing based on the current week. Although we allowed users to author filters on the individual charts, P1 expressed the need for global (page-level) *and* section-level filters that would be applied to the charts.

As for the exporting options, P2 suggested that it could be more diverse, such as exporting to existing formats like PowerPoint, documents, and interactive PDFs. P2 also noted that it would be useful to allow for additional page layouts, such as horizontal scrolling. P1 similarly mentioned that he would like more design control over the page, such as selecting the background color and fonts or applying a theme. Lastly, P1 noted he would want the reporting feature to be more tightly integrated with the analysis process, in order to refine existing charts, create new ones, or preview and analyze the data.

5. Conclusion

We present CODAS: a web-based authoring tool that supports business analysts in creating data-driven reports alongside their existing data analysis workflow. While a commercial tool would require additional functionality to support a richer authoring experience, we focused on critical functions to support an end-to-end authoring workflow to validate our overall framework. There are several limitations in our current work that we plan to explore more in the future. Firstly, despite that we design the JSON structure exported from the system with an intention of separating input data and report specification to support data refreshing, our current system does not fully support this functionality due to the manual narrative input without linking of data and text. Additionally, our evaluation is currently limited to two expert reviews, we plan to run a more comprehensive evaluation of the system in the future. Future work should also explore the development of new intelligent features for authoring data-driven reports, such as automatic content generation for recurring reports, recommendation of visualization charts, etc.

References

[BLE18] BADAM S. K., LIU Z., ELMQVIST N.: Elastic documents: Coupling text and tables through contextual visualizations for enhanced document reading. *IEEE transactions on visualization and computer graphics 25*, 1 (2018), 661–671. 2

[CH18] CONLEN M., HEER J.: Idyll: A markup language for authoring and publishing interactive articles on the web. In *Proceedings of the 31st Annual ACM Symposium on User Interface Software and Technology* (2018), pp. 977–989. 1

[CVTH21] CONLEN M., VO M., TAN A., HEER J.: Idyll studio: A structured editor for authoring interactive & data-driven articles. In *The 34th Annual ACM Symposium on User Interface Software and Technology* (2021), pp. 1–12. 2

[KM13] KOSARA R., MACKINLAY J.: Storytelling: The next step for visualization. *Computer 46*, 5 (2013), 44–50. 2

[KRA*18] KERY M. B., RADENSKY M., ARYA M., JOHN B. E., MYERS B. A.: The story in the notebook: Exploratory data science using a literate programming tool. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems* (2018), pp. 1–11. 2

[LRIC15] LEE B., RICHE N. H., ISENBERG P., CARPENDALE S.: More than telling a story: Transforming data into visually shared stories. *IEEE computer graphics and applications 35*, 5 (2015), 84–90. 2

[LZK*21] LATIF S., ZHOU Z., KIM Y., BECK F., KIM N. W.: Kori: Interactive synthesis of text and charts in data documents. *IEEE Transactions on Visualization and Computer Graphics* (2021). 2

[MZJS18] METOYER R., ZHI Q., JANCZUK B., SCHEIRER W.: Coupling story to visualization: Using textual analysis as a bridge between data and interpretation. In *23rd International Conference on Intelligent User Interfaces* (2018), pp. 503–507. 2

[noa16] Tableau(Version. 9. 1). *Journal of the Medical Library Association : JMLA 104*, 2 (Apr. 2016), 182–183. 1, 2

[Pur21] PURI J.: Jpuri/react-draft-wysiwyg, Mar. 2021. original-date: 2016-09-04T18:33:34Z. 3

[RTH18] RULE A., TABARD A., HOLLAN J. D.: Exploration and Explanation in Computational Notebooks. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems* (New York, NY, USA, Apr. 2018), CHI '18, Association for Computing Machinery, pp. 1–12. 2

[SCBL21] SULTANUM N., CHEVALIER F., BYLINSKII Z., LIU Z.: Leveraging text-chart links to support authoring of data-driven articles with vizflow. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems* (2021), pp. 1–17. 3

[SDES18] SRINIVASAN A., DRUCKER S. M., ENDERT A., STASKO J.: Augmenting visualizations with interactive data facts to facilitate interpretation and communication. *IEEE transactions on visualization and computer graphics 25*, 1 (2018), 672–681. 2

[SH10] SEGEL E., HEER J.: Narrative visualization: Telling stories with data. *IEEE transactions on visualization and computer graphics 16*, 6 (2010), 1139–1148. 2

[SH14] SATYANARAYAN A., HEER J.: Lyra: an interactive visualization design environment. *Computer Graphics Forum 33*, 3 (2014), 351–360. 2

[SMWH16] SATYANARAYAN A., MORITZ D., WONGSUPHASAWAT K., HEER J.: Vega-lite: A grammar of interactive graphics. *IEEE transactions on visualization and computer graphics 23*, 1 (2016), 341–350. 1, 2

[Sof10a] SOFTWARE: Microsoft power bi. https://powerbi.microsoft.com/en-us/, Retrieved 2021-03-10. 1

[Sof10b] SOFTWARE: Piktochart. https://piktochart.com/, Retrieved 2021-03-10. 2

[Sof10c] SOFTWARE: Quadrigram: Data visualization & presentation tool. https://www.quadrigram.com/, Retrieved 2021-03-10. 1, 2

[Sof10d] SOFTWARE: React-vega. https://www.npmjs.com/package/react-vega, Retrieved 2021-03-10. 3

[Sof10e] SOFTWARE: Sisense: Advanced analytics & cloud data platform for data teams. https://www.sisense.com/product/data-teams/, Retrieved 2021-03-10. 2

[Sof10f] SOFTWARE: Tableau stories. https://help.tableau.com/current/pro/desktop/en-us/stories.htm, Retrieved 2021-03-10. 1

[SWH14] SATYANARAYAN A., WONGSUPHASAWAT K., HEER J.: Declarative interaction design for data visualization. In *ACM User Interface Software & Technology (UIST)* (2014). 1, 2

[VB18] VAIH-BAUR C.: Scrollytelling. In *Textspiele in der Wirtschaftskommunikation: Texte und Sprache zwischen Normierung und Abweichung*, Femers-Koch S., Molthagen-Schnöring S., (Eds.). Springer Fachmedien, Wiesbaden, 2018, pp. 343–346. 3

[WQM*17] WONGSUPHASAWAT K., QU Z., MORITZ D., CHANG R., OUK F., ANAND A., MACKINLAY J., HOWE B., HEER J.: Voyager 2: Augmenting visual analysis with partial view specifications. In *ACM Human Factors in Computing Systems (CHI)* (2017). 2