

RESEARCH STATEMENT

Jeffrey Michael Heer



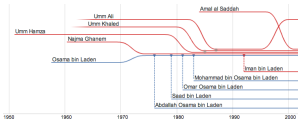
Mapping global open source software collaboration using our visualization tools.

The increasing scale and accessibility of digital data provides an unprecedented resource for informing governance, public policy, business strategy, and our personal lives. Yet, acquiring and storing this data is, by itself, of little value. We must make sense of data in order to derive value from it. Turning data into knowledge is a fundamental challenge for both computer systems and user interface research: it requires integrating analysis algorithms with human judgments of the meaning and significance of observed patterns.

My research seeks to enhance our collective ability to understand and communicate data through the design of new interactive systems for data visualization and analysis.

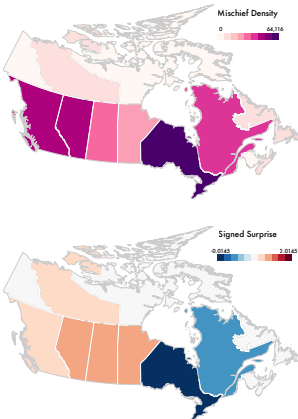
My group studies the perceptual, cognitive, and social factors affecting data analysis in order to enable people to more effectively work with data: to improve the efficiency and scale at which expert analysts work, and lower the threshold for non-experts.

VISUALIZATION TECHNIQUES



TimeNets show genealogical relations over time, including offspring, divorce, remarriage, polygamy, and age difference.

Our research on *visualization techniques* uses insights from studies of human perception and cognition to design visual representations and interaction techniques for data. My group has designed and evaluated novel visual analysis tools for diverse domains such as genealogy ([AVI'10](#)), open source collaboration ([MSR'11](#)), network science ([InfoVis'11](#), [AVI'12](#)), ant colonies ([JAE'13](#)), RNA sequence populations ([BioVis'13](#)), distributed database performance ([EuroVis'15](#)), and software runtime behavior ([EuroVis'16](#)).

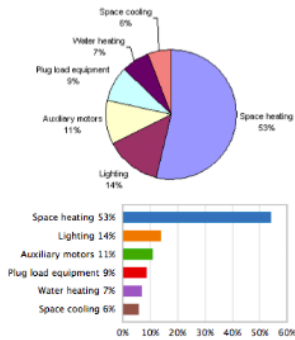


A density map of crimes in Canada (top) simply mirrors population density. A *Surprise Map* (bottom) uses models of population and sample variance to highlight areas with unexpected rates.

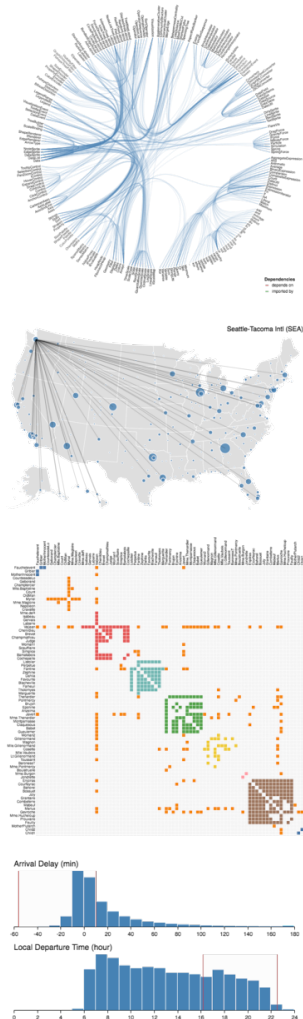
As a more detailed example, *Surprise Maps* ([InfoVis'16](#)) seek to de-bias common visualizations such as statistical maps, importing a measure from vision science named “Bayesian surprise.” Surprise Maps involve specifying a set of reference distributions (e.g., models of population density and expected variation, such as larger variances for smaller samples) and an initial prior distribution over how much we “believe” each distribution (e.g., equiprobable for an uninformed prior). As new data is observed, we perform updates by applying Bayes’ Rule. The resulting posterior distribution gives us a new set of “beliefs” that then can serve as our new prior. By comparing our prior and posterior beliefs we can quantify the amount of “surprise” the data has induced—or, in other words, to what degree our belief in the models was shifted by the observed data. By visualizing these surprise scores, we highlight regions of the data that could benefit from closer attention.

GRAPHICAL PERCEPTION & MODELING

Our research on *graphical perception* uses human-subjects experiments to evaluate visual encoding choices. We have demonstrated how crowdsourcing systems such as Amazon’s Mechanical Turk can be used to generate reliable and novel perceptual findings ([CHI'10](#)). For example, our experimental results ([CHI'10](#), [InfoVis'10](#)) challenge a decade-held assumption regarding the effects of aspect ratio on rectangular shape comparisons. These results suggest new guidelines for perceptually effective layout algorithms.



ReVision takes bitmap chart images as input (top), extracts the data table, and creates redesigned graphs (bottom).



Interactive visualizations made using the Vega language.
vega.github.io/vega

In related work, we have investigated perceptual principles and corresponding models for animated transitions ([InfoVis'07](#)), color palette design ([CHI'12](#), [EuroVis'13 Best Paper](#)), and chart sizing & aspect ratio ([InfoVis'06](#), [CHI'09 Best Paper](#), [CHI'10](#)). Recent work includes studies of “visual statistics” such as perceived correlation ([InfoVis'15](#)) and regression ([CHI'17](#)), as well as crowdsourcing methods for constructing perceptual models that can be used to guide automated design algorithms ([InfoVis'14](#)).

Moving beyond low-level perception, other empirical work ([InfoVis'10](#)) characterizes narrative devices employed to effectively “tell stories” with data. This work has received wide dissemination among journalists and designers, and has informed our subsequent work on design tools for crafting interactive narratives ([EuroVis'14](#)).

Extending our graphical perception research, we are also developing *computational models of chart perception*. *ReVision* ([UIST'11 Notable Paper](#)) uses computer vision methods to classify bitmap images of charts by their visualization type. Extraction procedures recover underlying data to enable redesign and indexing. More recent work ([EuroVis'17](#)) achieves automatic inference of visual encoding rules from a bitmap image via new state-of-the-art results for optical character recognition (OCR) in chart images. Ongoing work focuses on automatic computational interpretation of color encodings and cartographic projections.

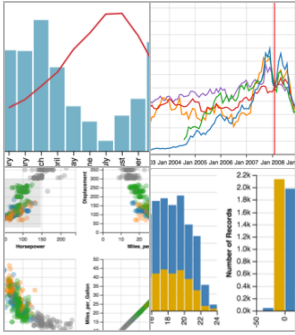
VISUALIZATION LANGUAGES & SYSTEMS

Improved visualization techniques are of little use if they never make it into the hands of designers and developers. Accordingly, my group has a long-standing interest in new tools that support visualization design. Our prior work includes the *Prefuse* ([CHI'05](#)), *Protopis* ([InfoVis'09](#), [InfoVis'10](#)), and *D3* ([InfoVis'11](#)) systems, each with significant real-world uptake. At the time of writing, *D3.js* was the 5th most popular project on GitHub.

Our current research focuses on *declarative languages for interactive visualization*. Long a staple of database query languages (SQL) and graphic design (HTML, CSS), declarative approaches to visualization have only more recently come to hold sway. In particular, declarative models for visual encoding (mapping data to visual elements) have become a dominant way of expressing visualizations, providing an appropriate balance of expressive power and usable, domain-specific constructs. Research highlights in this area include Wilkinson's *The Grammar of Graphics* (the basis of the popular *ggplot2* library for R) and the Stanford *Polaris* project (now Tableau Software, a multi-billion dollar company). While such tools use a declarative approach for static encodings, custom interaction design is either unsupported or achieved only via imperative callbacks.

The goal of the *Vega* project is to advance both visualization theory and tooling through higher-level languages that (1) simplify the creation of powerful, interactive visualizations and (2) support new applications for improved exploratory data analysis. This goal has led us to develop not a single system, but rather a stack of tools for interactive visualization.

At the foundation of this stack is the *Vega* visualization grammar ([UIST'14](#), [InfoVis'15](#)). Similar in spirit to how SQL provides a language for database queries, Vega provides a



Interactive statistical graphics created with Vega-Lite.

vega.github.io/vega-lite

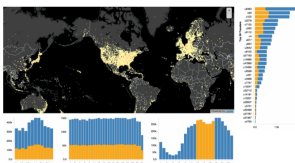
declarative language for describing visualizations. Vega specifications include the data transformations and visual encoding rules needed to express a rich space of visualizations. The Vega runtime parses JSON specifications to produce interactive web-based graphics, implemented using a reactive dataflow architecture. One unique aspect of Vega is its support for *declarative interaction design*: instead of the “spaghetti code” of event handler callbacks, Vega treats user input (mouse movement, touch events, etc.) as first-class streaming data to drive reactive updates to a visualization. Vega has been adopted by Wikipedia as a standard for adding interactive visualizations directly to wiki pages.

Though expressive and performant, Vega can still require verbose specifications: control flow is handled by the model, but the logic of event and data processing must be explicitly provided. As a result, Vega is not ideal for quickly authoring interactive plots in the midst of an analysis session. *Vega-Lite* (InfoVis’16 Best Paper) is a concise high-level language for rapidly creating interactive, multi-view visualizations, and compiles to complete Vega specifications. Vega-Lite introduces a *view algebra* for combining basic plots into more complex multi-view displays, and a new *selection* abstraction for declarative specification of interaction techniques. To the best of our knowledge Vega-Lite is the first example of a high-level *grammar of interactive graphics*. Vega-Lite has already been adopted in the Jupyter data science notebook community via the *Altair* project’s Python bindings.



The Voyager system for visual exploratory analysis integrates manual chart specification with recommendations based on the user’s current focus.

While Vega and Vega-Lite are useful in their own right, our larger research goal is to provide a convenient yet powerful means for writing *programs that generate visualizations*, ranging from design tools to chart recommender engines. *Lyra* (EuroVis’14) is a design environment in which users can craft custom visualizations via direct manipulation, realized as Vega specifications. *Voyager* (InfoVis’15, CHI’17) is an exploratory analysis interface that blends manual specification and automatic recommendation. Through multiple user studies, we found that judicious suggestions (e.g., that promote both data diversity and perceptually effective charts) can lead to more systematic coverage of a data set during early exploration. Underlying Voyager is *CompassQL* (SIGMOD’16), a query language that uses logic programming techniques to recommend visualizations given a partial Vega-Lite specification as input. Finally, *GraphScape* (CHI’17) is a directed graph model for reasoning about visualization similarity and sequence. Nodes represent Vega-Lite specifications. Edges represent atomic edits to transition from one chart to another, and are weighted by an estimated “cognitive cost” of interpreting the target chart having seen the source chart. GraphScape enables automatic search for alternative visualization designs and recommendation of chart sequence orderings for effective presentation.



Real-time visual querying of millions of data points within the *imMens* system.

In another project, we devised techniques for scaling visual encoding and interaction to the large data volumes now common in scientific and industrial practice. By binning data into overlapping, multivariate data tiles and leveraging parallel query processing, our *imMens* system (EuroVis’13) was the first to provide 50 frames-per-second interactive querying over summaries of billion+ element databases. In a follow-up experiment (InfoVis’14), we found that reducing interactive latency can significantly improve the rates of both observation and hypothesis formation during exploratory data analysis.



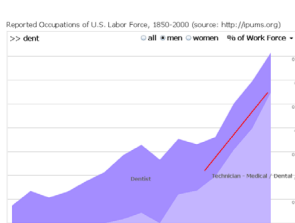
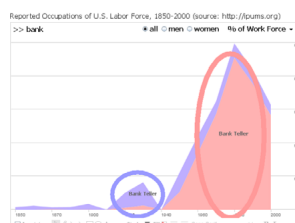
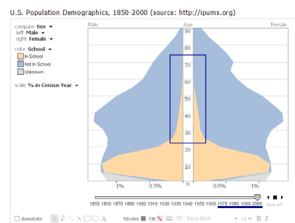
Visualizing topic model term distributions in *Termite*.

INTERACTIVE TEXT ANALYSIS & LANGUAGE TRANSLATION

Multiple projects in my group have targeted the intersection of visualization and natural language processing, leading to publications in visualization, HCI, NLP and machine learning venues. For example, we have investigated the strengths and weaknesses of topic models for large-scale text analysis, developing topic model visualization techniques ([AVI'12](#)), tools to track concept dissemination among academic disciplines ([CHI'12](#), [Poetics'13](#)), topic model evaluation methods grounded in surveys of domain experts ([ICML'13](#)), and tools for researchers such as political scientists to conduct robust topical analysis in the face of highly variable outputs across modeling runs ([NAACL'15](#)).

We have also contributed state-of-the-art classifiers for identifying medical terms in “messy” patient-authored text ([JAMIA'13](#), [JAMIA'14](#)). Working with an addiction specialist, we developed methods to analyze patterns of prescription opioid abuse, cessation, relapse, and recovery from posts to online medical forums ([CSCW'15](#)).

In addition, we have worked on interactive tools for natural language translation that combine human and machine translation (MT) approaches. We first conducted a study of “post-editing” of MT output ([CHI'13 Best Paper](#)), which in turn informed the design of novel translation interfaces that interleave human editing with adaptive machine suggestions ([UIST'14](#), [EMNLP'14](#)). Studies with professional translators found that our methods lead to improved translation quality over post-editing, in conjunction with online machine learning methods that refine translation models in the midst of a session. This work led to the founding of *Lilt*, a company providing language translation services.



Annotated views from social data analysis in *sense.us*.

1. The rise of adult education from the 1970s onward
2. Reversal of the dominant gender of bank tellers
3. Stratification of dentistry into dentists & hygienists

SUPPORTING THE DATA ANALYSIS LIFE-CYCLE

By deploying analysis tools, we can study how users apply them to make sense of data. For example, we have observed that analysis is often a social process, and our research on *collaborative visual analysis* explores how interfaces can catalyze social interpretation and deliberation. One such system is *sense.us*, a site for social exploration of 150 years of U.S. census data ([CHI'07](#), [CACM'09](#)). Our studies of system usage found that social features can improve hypothesis generation and spur new explorations by collaborators. This work informed a number of data sharing and visualization services, including IBM's Many-Eyes.com and Google's Fusion Tables. Subsequent research examined how to further structure collaboration to improve analysis outcomes. For example, we found that dividing analysis tasks into explicit stages and providing structured discussion tools can improve both the quantity and quality of findings ([CHI'11](#)), and can be used to successfully scaffold tasks to engage crowd workers in data interpretation ([CHI'12](#)).

In interview studies we also observed that professional analysts spend much of their time cleaning and reformatting data to make it suitable for analysis ([VAST'12](#)). Domain experts often spend more time manipulating data than they do exercising their specialty, while less technical users (*e.g.*, non-programmers) may be excluded. In response, we investigated new methods for *interactive data transformation*. With our *Wrangler* system

Year	extract
Reported crime in Alabama	
2004	
2005	
2006	
2007	
2008	
Reported crime in Alaska	
2004	
2005	
2006	
2007	
2008	
Reported crime in Arizona	

A user creates text extraction rules by example in *Wrangler*.

(CHI'11, UIST'11), users construct data transformation scripts in a direct manipulation interface. Wrangler uses programming-by-demonstration methods to automatically suggest applicable transforms. The result is not simply transformed data, but a reusable program that can be run on other platforms (e.g., Spark, MapReduce) to transform data at scale and be audited to review data provenance. The Wrangler project and subsequent work on visual data profiling (AVT'12) led us to found *Trifacta*, a software company that has received over \$50M in venture funding and now has over 100 employees.

CONTINUING RESEARCH

Going forward, my group is exploring a number of interconnected projects:

- *Empirical assessment of techniques for interactive machine learning.* Under what circumstances is it valuable to have a “human-in-the-loop” for feature engineering and model refinement? In what circumstances does it lead to wasted effort or overfitting?
- *Techniques for effectively visualizing uncertainty.* Following in the footsteps of our Surprise Maps work, we are investigating different visual encoding techniques that incorporate uncertainty, for example by intentionally making the values of highly uncertain estimates more difficult to discriminate. How might different depictions of uncertainty succeed (or fail) in influencing people’s decision making?
- *Methods for recommending perceptually-effective visualizations.* The primary approaches used for automated visualization design are now decades old, based on a simplistic model of user tasks and greedy optimization methods. Through both empirical study and new algorithmic approaches, we seek to contribute more robust and effective methods for guiding visualization design decisions.
- *Tighter coupling between visualization and database systems.* Our Vega language draws on both database and visualization methods, but currently focuses on client-side execution. We are now investigating how Vega query plans might be partitioned to automatically generate optimized, adaptive client-server applications that leverage scalable database backends. We are also interested in incorporating perceptual models, screen resolution, and predictions of likely user actions to further optimize performance, for example by choosing appropriate approximations (e.g., stop computation once some perceptual tolerance is reached) and pre-fetching methods (e.g., to speculatively query for data a user is likely to request in the near future).
- *New languages to create interactive documents.* Looking beyond visualization alone, we are exploring new markup languages for crafting reactive web documents with interactive content, with applications to data journalism, academic publishing, and e-textbooks.

For a complete list of projects, papers, and demos see <http://idl.cs.washington.edu>