A High-level Language for Interactive Data Visualization

Leila De Floriani, University of Maryland, College Park

For visualization tools to support exploratory data analysis, they must allow analysts to rapidly create and refine visualizations. Some current systems let developers build custom interactive visualizations for the web, but the time and effort required to use these tools can break the flow of an ad hoc analysis session. In contrast, tools such as Tableau (www.tableau.com) and ggplot2 (a visualization package for the R statistical computing language; ggplot2.org) support iterative and interactive construction of graphics using a high-level grammar for concise specification.

Existing grammars of graphics are effective for creating static charts but provide little if any support for constructing interactive visualizations. Analysts can, at most, enable a predefined set of common techniques—linked selections, panning and zooming, and so on—or parameterize their visualization with dynamic query widgets. For custom direct-manipulation interaction, they must turn to imperative event-handling callbacks, a stark departure from declarative visual-encoding grammars. Not only does this representational switch make interactions difficult to specify, it also impedes automated reasoning over the design space of interactive charts.

In “Vega-Lite: A Grammar of Interactive Graphics,” Arvind Satyanarayan, Dominik Moritz, Kanit Wongsuphasawat, and Jeffrey Heer introduce new formalisms for interaction techniques and multichart composition (IEEE Trans. Visualization and Computer Graphics, vol. 23, no. 1, pp. 341-350). Vega-Lite (vega.github.io/vega-lite) extends the traditional grammars of visual-encoding rules by adding a composition algebra for layered or multiview displays and a grammar of interaction. Interactions are specified as “selections” over data-points or intervals, which can then be used to drive conditional encoding logic (for example, to highlight only selected points), parameterize scale mappings (for example, an interval might represent a scale domain to be panned and zoomed), and filter data (for example, to perform brushing and linking across multiple views). Users need only specify the desired selection type’s semantics: the Vega-Lite compiler automatically synthesizes appropriate event-handling logic, producing a complete specification as output. The video at www.computer.org/computer-magazine/category/multimedia provides a more detailed description of Vega-Lite.

Satyanarayan and his coauthors received the Best Paper Award at the 2016 IEEE Information Visualization Conference. Vega-Lite is now being used to create new visualization systems (including the Voyager visualization recommendation browser) and enable rapid visualization in data-science environments such as Jupyter Notebooks (using, for example, the Altair Python API).

LEILA DE FLORIANI is a professor at the University of Maryland and Editor in Chief of IEEE Transactions on Visualization and Computer Graphics. Contact her at ldf4tvcg@umiacs.umd.edu.