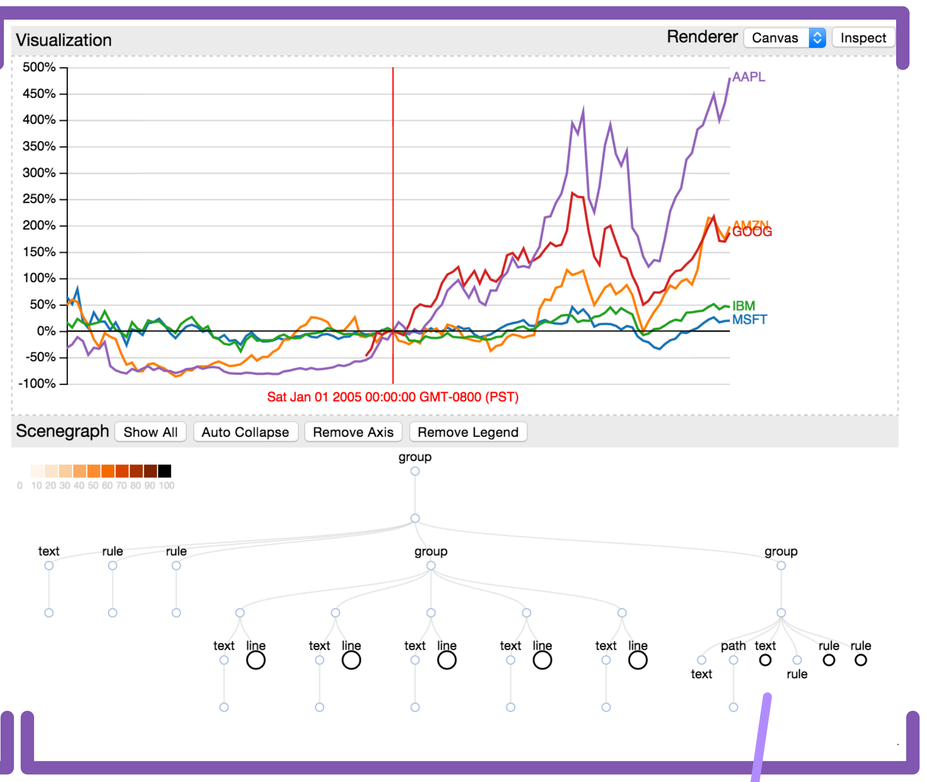
Visualizing Vega's Scenegraph and User Interaction

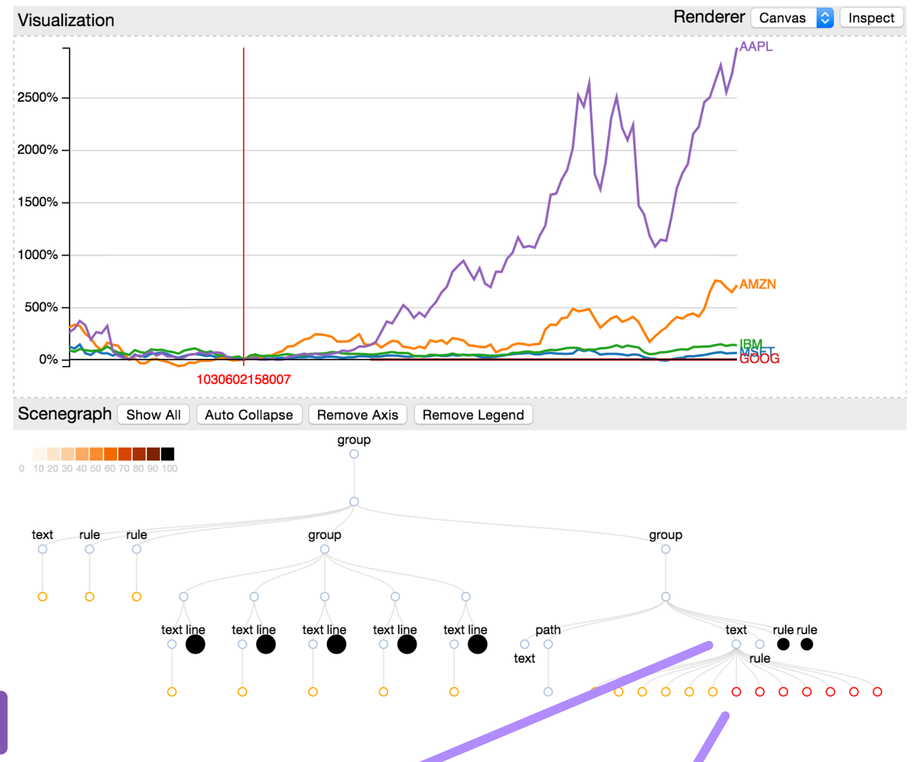
Jane Hoffswell

BACKGROUND: Vega is a declarative visualization language that enables **rapid iteration** of visualizations and supports **retargeting**, while allowing designers to **focus on visual encoding** instead of low-level implementation details. However, these advantages come at the cost of effective debugging due to obfuscation of the underlying program execution.

GOAL: To support debugging, this project augments the development process by visualizing the scenegraph in addition to the specification and visualization. End-user interaction and inspection of the visualization update the scenegraph to provide relevant context.

Visualization





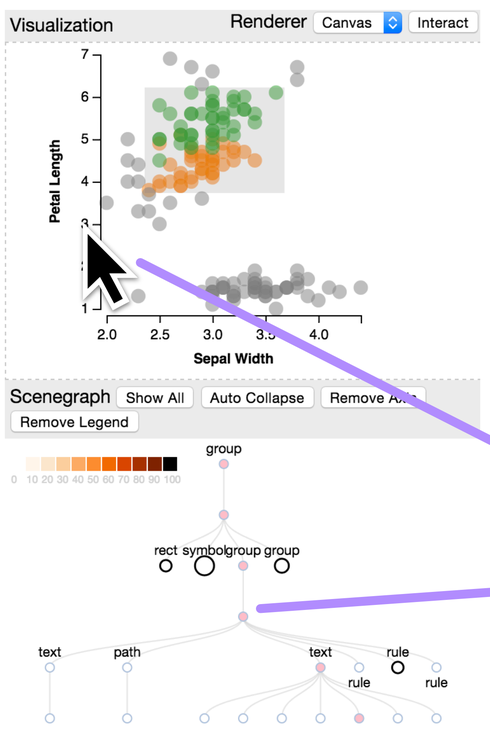
Specification

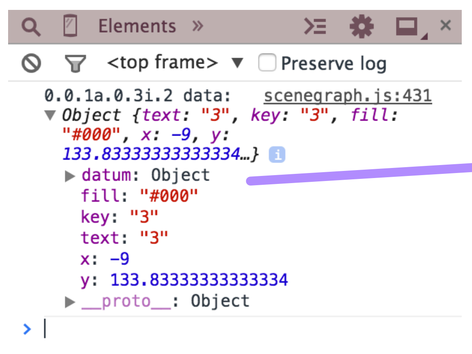
Scenegraph

End-User Interaction (above): As the user interacts with the **index chart above**, the scenegraph updates to show how the visualization is being changed. In this example, the lines are modified in their entirety and some of the axis marks are removed while others are modified.

Size Encoding: Selecting nodes in the scenegraph collapses/expands the contents. Collapsed nodes are identified by a thick black border and the size encodes the number of hidden descendants.

Color Encoding: Collapsed nodes show the percentage of descendants that have been modified in some way. Otherwise, the stroke color represents the status of the node (added, modified, or removed).





Inspection: When the user clicks the "3" label in the visualization, a path of nodes is highlighted in the scenegraph showing all nodes for which the selected point is within the node's bounds.

Scenegraph Data: Selecting nodes in the scenegraph prints the internal data to the JavaScript console.

Inspection (left): In "Inspection" mode, end-user interactions are disabled. The user can then select components of the visualization to see corresponding nodes in the scenegraph.

FUTURE WORK: These forms of user interaction enable inspection of the underlying structure. Though changes to the data are tracked and shown in the scenegraph, these data changes should be shown more explicitly so that the user can make more rapid comparisons. Also, brushing & linking to the specification would improve the user's ability to make modifications.