Shedding Light on Distributed System Executions

http://bestchai.bitbucket.org/shiviz/

Inspecting distributed system logs is tedious and error-prone:

Understand and debugging requires knowing how nodes communicated:

- Were events X and Y concurrent, or did one precede another?
- Did node A ever communicate with node B? When?
- Did node B ever communicate with node C? When?
- Can we effectively visualize series of communication?

ShiVector tool

- Automatically instruments a distributed system logging with ordering information (vector clocks)
- Transparently interposes on all socket communication and logging (via AspectJ)

ShiViz tool

- Concisely visualizes a ShiVector log of a distributed execution
- Lightweight (runs in a browser)
- Reveals the communication topology in the system (gives developers a new perspective on familiar console logs)

Ongoing work

- Visualizing multiple executions
- Mining behavior motifs from large logs
- Query/filter language for needle-in-a-haystack analysis
- Scaling visualization to long executions/many nodes

Jenny Abrahamson, Ivan Beschastnikh, Yurii Brun, Michael D. Ernst