Towards a Visualization Process Model for Online Visualization

Marco Angelini {angelini@dis.uniroma1.it},
Giuseppe Santucci {santucci@dis.uniroma1.it}
VisDis Research group, University of Rome 'La Sapienza'

Hans-Jörg Schulz (hjschulz@informatik.uni-rostock.de)
Heidrun Schumann (schumann@informatik.uni-rostock.de)
Institute for Computer Science, University of Rostock

INFOVIS poster session, VisWeek 2014, Paris (FRA)

MOTIVATIONS

Existing models follow a pipeline of operators.
They can be adjusted before or after being applied, but not on the fly.

Nowadays applications and datasets ask for:

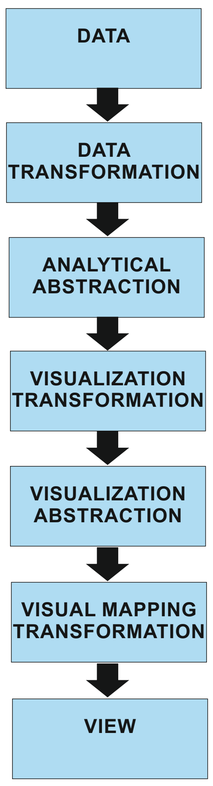
- supporting incremental data handling and incremental visualizations
- allowing the interaction with partial results adjusting parameters
- optimizing the visual analysis

OUR GOAL a new visualization model to address these needs that we term Online Visualization

BACKGROUND

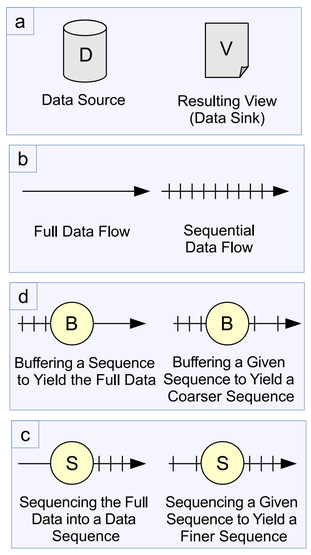
- Several disjoint visualization approaches deal with related aspects:
  - Out-of-core visualization (e.g., [1])
  - Streaming data visualization (e.g., [2])
  - Online dynamic graph drawing (e.g., [3])
  - Layered visualization (e.g., [4])
  - Progressive Visual Analytics (e.g., [5])
  - Parallel visualization (e.g., [6])
- These heterogeneous approaches can be subsumed under Online Visualization

COMMON VISUALIZATION MODELS (e.g. CHI)



THE ONLINE VISUALIZATION MODEL

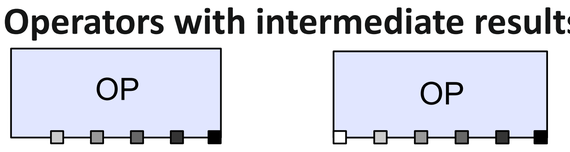
Enhanced Data Modeling



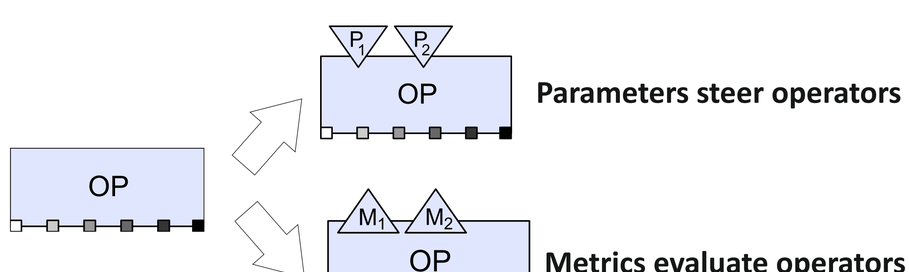
Enhanced Process Modeling

Monolithic Operators

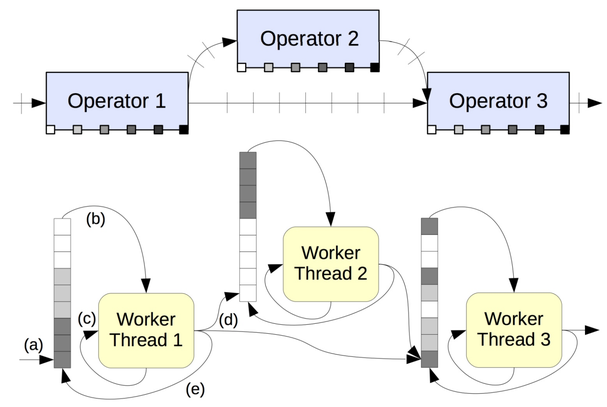
Operators with intermediate results



Metrics & Parameters



IMPLEMENTATION



Online Visualization can be implemented using priority queues and asynchronous threads[7]:

- (a) data chunks are added to the queue
- (b) retrieved by the worker thread that
- (c) iteratively computes better solutions
- (d) a good enough result is generated and passed on to subsequent threads.
- (e) If the result is not yet good enough to be final, it is again added to the queue

ONLINE VISUALIZATION INTERACTION & STEERING: A CASE STUDY

TASK (based on NTHSA FARS dataset): the user issues a query interacting with several sliders, setting a reference crash. The system computes similar crashes, plotting them on a density map

Common Visualization Model

- it is not possible to model and render intermediate results
- the slow response time makes the system not interactive system

Online Visualization Model

- it is possible to model and render intermediate results
- the quicker response time makes the system interactive

a: Mono **b:** Linear **c:** Non linear

METRICS VIEW 4

1 MODEL VIEW

VISUALIZATION VIEW
- It shows partial results immediately updated

2

3

SETTINGS VIEW
- changing parameters
- selecting metrics

a: Mono b: Linear c: Non Linear

Speed vs Accuracy

Comparison between two model's parametrizations:
- prioritizing speed
- prioritizing accuracy

Insights

Number of Active drunk people involved in crashes on Sunday's early hours: 0 (left), 1 (middle), 3 (right)

REFERENCES

[1] J. A. Cottam, A. Lumsdaine, and P. Wang, "Abstract rendering: Out-of-core rendering for information visualization," in Proceedings of the Conference on Visualization and Data Analysis (VDA'14).
[2] J. A. Cottam, "Design and implementation of a stream-based visualization language," Ph.D. dissertation, Indiana University, November 2011.
[3] Y. Frishman and A. Tal, "Online dynamic graph drawing," IEEE Transactions on Visualization and Computer Graphics, vol. 14, no. 4, pp. 727–740, 2008.
[4] H. Piringer, C. Tominski, P. Muigg, and W. Berger, "A multi- threading architecture to support interactive visual exploration," IEEE Transactions on Visualization and Computer Graphics, vol. 15, no. 6, 2009.
[5] C. D. Stolper, A. Perer, and D. Gotz. Progressive visual analytics: User-driven visual exploration of in-progress analytics. IEEE TVCG,20(12), 2014. to appear.
[6] H.T.Vo,J.L.D.Comba,B.Geveci,andC.T.Silva,"Streaming- enabled parallel data flow framework in the Visualization ToolKit," IEEE Computing in Science and Engineering, vol. 13, no. 5, pp. 72–83, 2011.
[7] J. Choo, C. Lee, and H. Park. PIVE: A per-iteration visualization environment for supporting real-time interactions with computational methods. Technical report, Georgia Institute of Technology, 2013.