



An Execution Engine for Physical Visualization Design

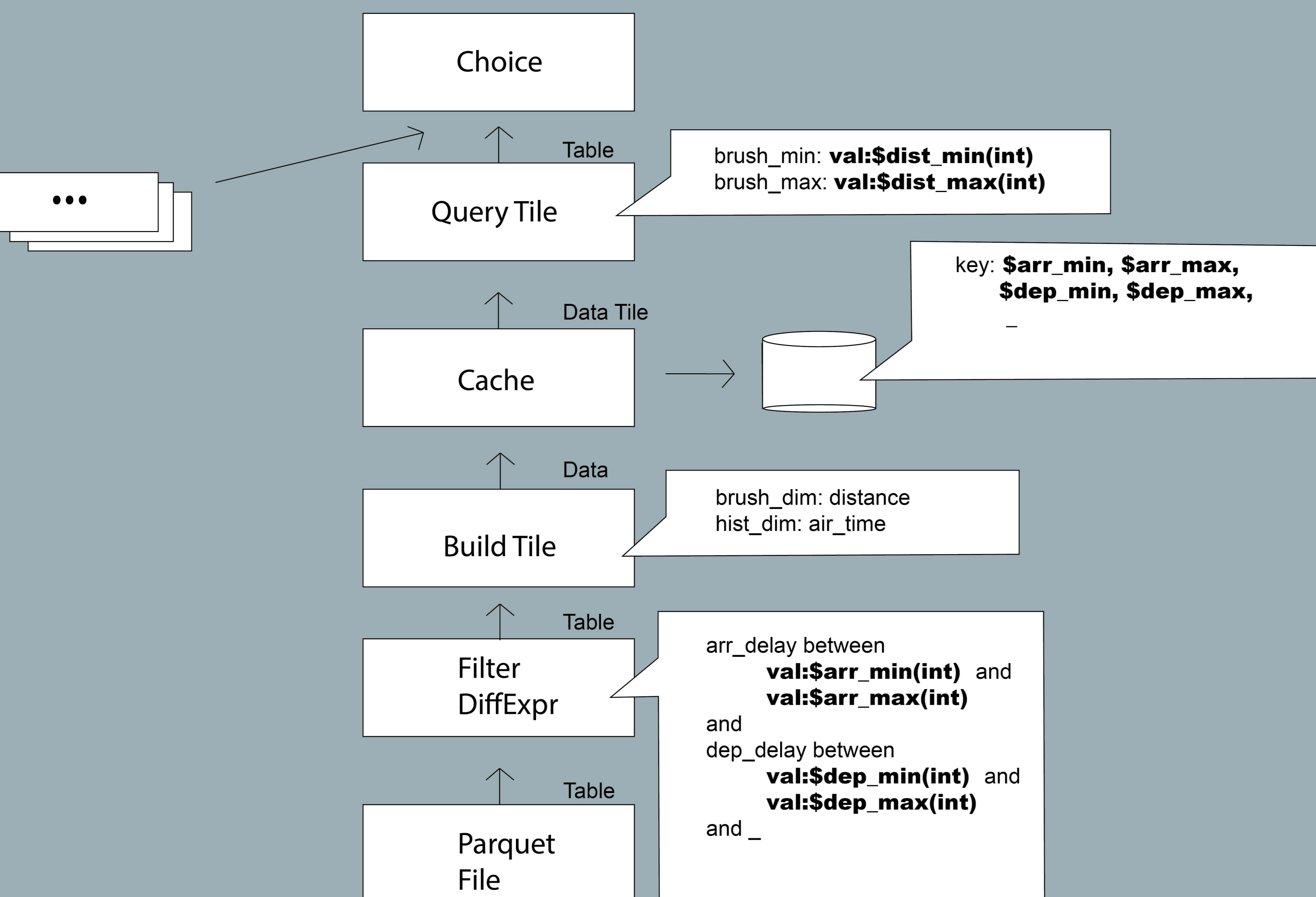


BY: JANISSA ABREU
COLUMBIA UNIVERSITY

INTRODUCTION:

Physical Visualization Design (PVD) is a system that allows a fast design of interfaces such as visualizations and interactions with user-designed choices. PVD is overall demonstrated as an optimizer for automating physical design; given a specification of an interface's query workload, storage constraints, and latency expectations for interactions.

METHOD:

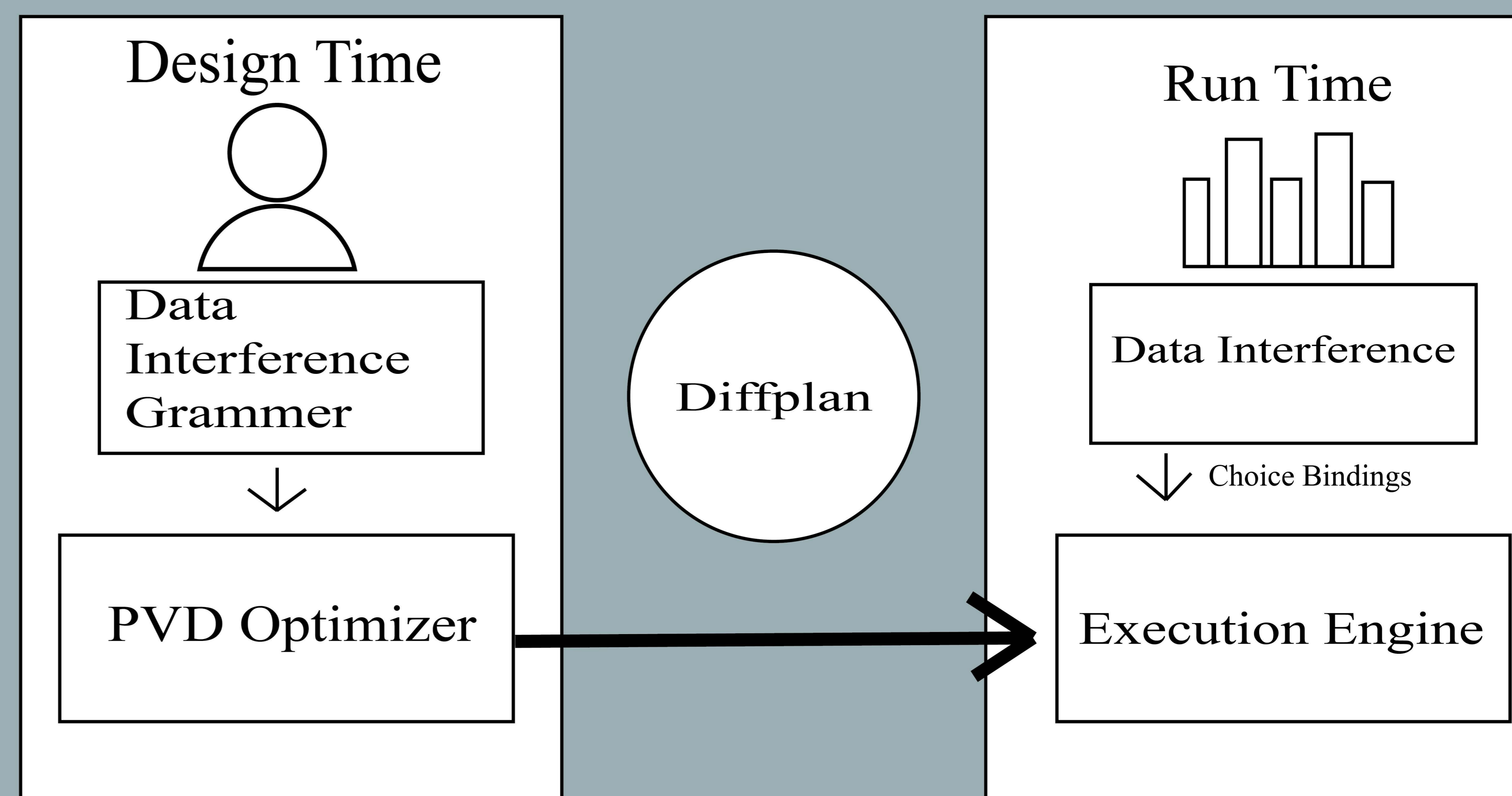


ANALYSIS:

A user makes choices known as “choice nodes” by interacting with a widget. As soon as a choice is made the query acts out the action of caching the changes made to the interface which builds the dataset based on the choice nodes. If a user makes more choice nodes their will need to be a new diffplan made and a new build tile will be created.

RESULTS:

Combined Diffplans with choice nodes through binding values, which are produced as the user interacts with the widgets in the interface.



FUTURE WORK:

- Fixing latency issues and implementing a new way for caching previously made datasets, since allocating memory was an issue.
- Changing the idea behind the caching system so it's not only used to save previous works the user has made.

REFERENCES/ ACKNOWLEDGEMENTS

- Ramjit, L., Kong, Z., Netravali, R., and Wu, E. Physical visualization design. In Proceedings of the 2020 ACM SIGMOD International Conference on Management of Data (New York, NY, USA, 2020), SIG-MOD '20, Association for Computing Machinery, p. 2809–2812.
- Mentor: Eugene Wu
- PHD students: Jeff Tao, Yiru Chen