

# Katz Centrality for Dynamic Graphs

## Bachelor's Thesis / Master Project

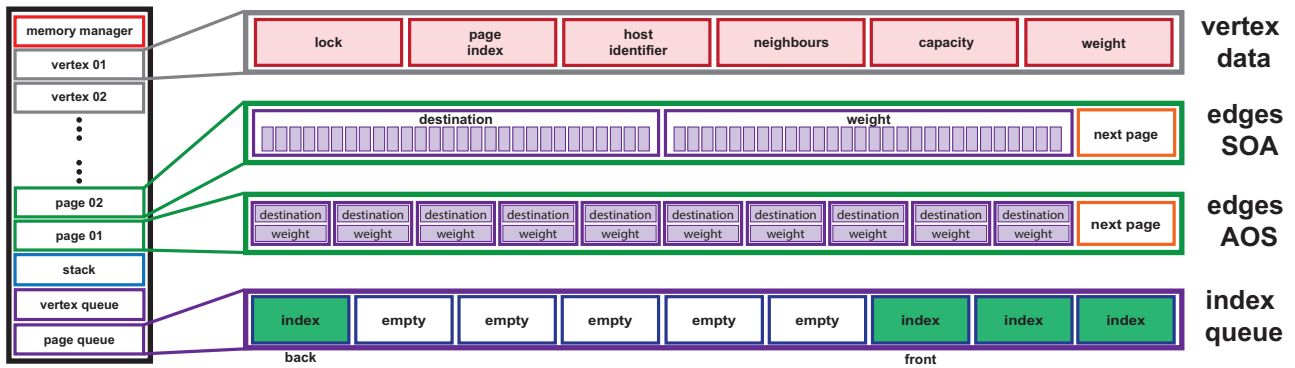


Figure 1: Visualization of the device memory layout as managed by *failGraph*.

### Description:

We are currently developing a dynamic graph framework for GPUs, called *failGraph*. As the pure representation of such a graph is only part of the appeal of graph processing, deriving interesting metrics from such a graph is typically the reason to manage a graph in the first place. The goal of this thesis is to implement *Katz Centrality* for *CSR*(Compressed Sparse Row) data structures as well as *failGraph*. *Katz Centrality* is a measure of centrality in a graph, which, unlike typical centrality measures that only consider shortest paths, is also influenced by the total number of walks between two vertices. Such implementations typically have a lot of optimization potential that can be exploited as much as is permitted by the time frame of the project. Optionally, one could exploit the dynamic setting to avoid re-computation after a graph update.

### Objective:

- Implement basic version of *Katz Centrality* for CSR on the CPU
- Implement algorithm on the GPU for CSR and *failGraph*
- (Optional) Exploit dynamic setting to avoid re-computation after a graph update

### Qualifications:

- Experience in C++
- Experience in CUDA, OpenCL or similar parallel computing framework
- Interest in massively-parallel compute architectures and graph processing

### Contact ICG:

Martin Winter  
 martin.winter@icg.tugraz.at