LSM-trie: An LSM-tree-based Ultra-Large Key-Value Store for Small Data (II)

NOTE: Your slides/presentation only need to cover background information necessary to answer the given questions (in a clear and well-organized manner). You are allowed to borrow contents from other resources, such as online slides, as long as you acknowledge them. The presentation should be mostly question-focused and proceed mostly in a Q&A format. Please include the questions in your slides. Don’t write detailed answers in the slides and read them to the class. Instead, use bullet points, graphs, or animations to explain your answers to the class.

In your Q&A report, use text to more thoroughly answer the questions. Include a short paragraph at the beginning of the report to summarize the paper.

(1) “The indices and Bloom filters in a KV store can grow very large.” Use an example to show that these metadata in LevelDB may have to be out of core.

(2) “Therefore, the Bloom filter must be beefed up by using more bits.” Use an example to show why the Bloom filters have to be longer?

(3) What’s the difference between SSTable in LevelDB and HTable in LSM-trie?

(4) “However, a challenging issue is whether the buckets can be load balanced in terms of aggregate size of KV items hashed into them” Why may the buckets in an HTable be load unbalanced? How to correct the problem?