Memory-Efficient GroupBy-Aggregate using Compressed Buffer Trees
Hrishikesh Amur (GT), Wolfgang Richter (CMU), David G. Andersen (CMU), Michael Kaminsky (Intel), Karsten Schwan (GT), Athula Balachandran (CMU), Erik Zawadzki (CMU)

NEED FOR MEMORY-EFFICIENCY
• Cost of DRAM
• Memory capacity per core decreasing [Lim, et al. ISCA’09]
• Future architectures including 3D-stacked DRAM have capacity constraints [Loh, et al., SHAW’12]

COMPRESSED BUFFER TREES
How to use compression to build memory-efficient and fast GroupBy-Aggregate?

GROUPBY-AGGREGATE
Sort vs. Hash-based GroupBy-Aggregate
• Google’s MapReduce and Hadoop use sorting as the GroupBy operator
• Hash-based grouping (common in RDBMS) performs better for many MapReduce applications. [Yu, Gunda, Isard, SOSP’09], [Li, et al. SIGMOD’11]

Can hashtables be compressed?

Cost of DRAM
Memory capacity per core decreasing [Lim, et al. ISCA’09]
Future architectures including 3D-stacked DRAM have capacity constraints [Loh, et al., SHAW’12]