## CHIUW 2017 - Sketching a Chapel Author: Chris Taylor

Traditional query support for Big Data applications has focused on static batched data sets. Recently, a family of algorithms called Sketching Algorithms, have gained popularity tackling technical challenges associated with querying streaming data sets. Sketching Algorithms provide approximate query results with mathematically provable error bounds. Chapel provides a straightforward, scalable, implementation and execution environment for Sketching Algorithms. This talk provides an introduction to Sketching Algorithms by way of a Chapel software library.

Sketching algorithms are referred to as "cardinality estimation" or "approximate counting" algorithms. For several application domains an approximate result is often sufficient. Sketching is an active area of research including the computation of summary statistics over frequency domains and streaming graph statistics. Sketches provide users the ability to interact with large data sets in real time. Users have the ability to interactively query large data sets in new and novel ways.

Sketching techniques rely on maintaining summary statistics, or state information, about a data stream. As an example, if a person were to flip coins and track the number of heads flipped in a row, another person provided that count information, could make a reasonable set of inferences about the stream of coin tosses. Sketching algorithms operate in a similar fashion.

Sketching algorithms describe the state of a stream by tracking summary statistics over hashed representations of each stream element. A hashed stream element is sliced into bins. For each bin, the longest run of zero values observed in the slice is maintained. The average value of each bin is also maintained. Some variations of the algorithm track arithmetic, geometric, or harmonic mean information about each bin. The bin state information is provided as input to a distinct value estimator. The distinct value estimator computes an estimated cardinality value from which a percent error can also be estimated.

Sketches exhibit properties associated with monoids, as a consequence of this relationship, a set of computed sketches can be joined under a union operation. This unique feature lends sketching computations to embarrassingly parallel computation. Chapel provides a vehicle to quickly prototype and implement scalable sketching solutions capable of execution in a differentiated (Cloud or HPC) compute environment.

In response to innovations in the Big Data application space, a software ecosystem is developing around Sketching algorithms. The Yahoo Data Sketches library provides Java users access to a robust set of sketch implementations for computing several streaming summary statistics. Druid is a Java-based query engine that provides a temporal storage solution and analytic engine for streaming data sets. Recently, the Druid team integrated Yahoo's sketching library into their analytic platform.

This talk will introduce Chapel implementations of the count-distinct (hyperloglog), streaming quantiles, streaming sampling, most frequent (heavy-hitters), and kth minimum value algorithms. A performance characterization of the Chapel library with respect to a Python library the same algorithms will be presented.