LUCATA PROGRAMMING ARCHITECTURE

The Lucata Pathfinder next generation computing platform not only powers faster analytics on existing Big Data graphs but also enables analytics on graphs that were previously too large to analyze cost-effectively. The Lucata Migratory Threads technology achieves breakthrough performance by removing the data access and processor utilization constraints of traditional computing approaches. You can leverage the tremendous performance improvements available with Lucata Pathfinder through three programming architectures – native, hybrid, and RedisGraph.

THE NATIVE LUCATA PROGRAMMING ARCHITECTURE

In the native programming architecture, your code runs solely on the Pathfinder platform. The native Lucata interface lets you analyze data at lightning-fast speeds. Your program must be written in C, C++, CILK, or GraphBLAS to run natively.

Execute your program through an interface running on your traditional computing environment. A high-performance data loader copies the data to be processed from your traditional computing environment system onto the Pathfinder. All processing is executed on the Pathfinder next generation computing environment. You can either run your own algorithms or take advantage of the pre-existing optimized Pathfinder algorithms, which include:

PageRank	Triangle Count
Connected Component	K-truss Subgraph
Breadth-First Search (BFS)	RandomAccess (GUPS)

THE HYBRID LUCATA PROGRAMMING ARCHITECTURE

In the hybrid architecture, your code runs on both your traditional computing environment and the Pathfinder platform. Pathfinder and your existing architecture work as co-processors. You can continue working in your existing development environment while, in the background, Pathfinder accelerates data-intensive workloads and allows you to process vastly larger datasets. Pathfinder provides massive benefits when analyzing graphs, performing sparse linear algebra, or processing large, sparse datasets.

Programming for Pathfinder is similar to programming for a GPU. Your program runs on the stationary cores in your existing traditional computing architecture and issues calls to run data-intensive processes on Pathfinder. Calls are documented in the Lucata SC-LCE API library. An automated data loader transfers data from your existing database and formats it for processing on Pathfinder.

CONTACT US

Contact Lucata now to learn more about the Pathfinder-S for high performance graph analytics.
Please email us at **info@lucata.com** or call us at **646 661-5252**.

You will modify your program to take advantage of Pathfinder by:

- Identifying the main data structures and distributing them in Pathfinder using the provided memory allocation functions

- Identifying time-consuming computational parts and parallelizing them using Cilk functions and the C/C++ helpers. If you use OpenMP, these pieces should be identified already.

- Using atomic operations, remote atomics, and intrinsics, as needed, to ensure the results are correct and efficient

THE REDISGRAPH PROGRAMMING ARCHITECTURE

In the RedisGraph programming architecture, you continue to run Redis in your traditional computing environment, while Pathfinder's custom version of RedisGraph enhances your experience behind the scenes. Lucata's custom implementation of RedisGraph provides higher performance and is much more scalable than RedisGraph running on traditional computing architectures. These enhancements reveal richer context and deeper insight into your data than ever before.

You run Redis and the RedisGraph module on your existing infrastructure with the same user interface you already know well. Whether you use the visual interface or command line to gather insights, Pathfinder will enhance your experience with higher performance graph processing.

Pathfinder works behind the scenes, intercepting GraphBLAS calls made by RedisGraph, to process graphs or sparse linear algebra that traditional architectures have issues processing. Pathfinder automatically distributes data and processing internally to optimize parallel performance.

CONCLUSION

Lucata Pathfinder provides breakthrough graph analytics performance for sparse Big Data graphs. Whether you use a commercial graph database or have created your own, there is a Lucata programming architecture that will enable you to take advantage of Pathfinder and take graph analytics to breakthrough levels of scale and performance.

0721/v1.0