TROVARES
seriously scalable

Trovares graph analytics: New tools for new challenges

Introduction

Graph analytics promises to make sense of a world increasing in data complexity. It's an approach that rethinks traditional relational database techniques. By mapping that data into a graph composed of nodes and edges (the relationships between the nodes), graph analytics is able to reveal insights beyond the data points themselves.

Existing analytic tools are challenged either by their lack of performance or their lack of scale. This data scaling problem is going to get worse – FAST. A growing army of machines from the Internet of Things (IoT) will produce more data much faster than humans have in recent decades. It's estimated that 50 billion or more IoT devices will join 7 billion humans producing data. The rule of thumb in technology is that more data equals more and better insights so this vast expansion of datasets should be a good thing. But, analytic tools already facing inherent limitations will be even more stretched and challenged to keep up with increasing data sizes.

Enterprise-class graph tools rooted in traditional data techniques (e.g. RDBMS, SQL) face difficulty managing these huge datasets. For a cybersecurity analyst facing an advanced persistent threat (APT), waiting days or even weeks to analyze a year's worth of network traffic could be catastrophic if malware strikes and results in financial loss.

The call to action is not simply to find new tools, but to bring new thinking to big data problems. Trovares brings a new approach honed over many years dealing with just these types of large datasets.

Trovares: New ideas, established track record

Trovares is a Seattle-based company creating a new generation of property graph analytics tools. The company's roots are in the High Performance Computing (HPC) space with major players from Cray guiding Trovares into a new era of enterprise problem-solving. Key technology designers have all worked in the HPC and high-performance analytics worlds.

- James Rottsolk, Co-founder and CEO: Co-founder of Cray Inc. and CEO from 1987 to 2006

- David Haglin, Co-founder and CTO: Chief Scientist Pacific Northwest Labs 2009 to 2016

- Gordon Matlock, Chief Strategy Officer: Led policy & technology initiatives at Pacific Northwest Labs

- Dick Russell, Director of Technical Marketing: Vector evangelist at Cray, helped create Northwest Institute for Advanced Computing

Trovares' early funding came from the Department of Defense (DoD). Trovares' charter was to abstract architectural approaches previously seen in HPC hardware into software. This allows users to take advantage of large multicore computing systems and their scalability in processing, memory, and storage capabilities. Instances of the software can run independently in the cloud for different users, problems, and data sizes. This utilization of HPC techniques makes this advanced technology available to new classes of users.

For instance, every enterprise is now concerned about cybersecurity as organized crime and state-sponsored actors test defenses constantly. Trovares brings a new solution to the hardest problems in these spaces and many others. The company is working with early adopter enterprises in several different application domains.

The challenge of graph analytics with truly huge datasets

Searching for patterns in graph-scale datasets involves processing a lot of data in parallel. As datasets increase in size, the tools must be able to handle the sheer scope of the dataset, the large number of ongoing memory accesses and also utilize the available processing power as efficiently as possible.

Trovares avoids many of the common parallelization pitfalls. Theoretically, exploiting parallelism should increase performance as multiple threads can run simultaneously. But, parallelism is still a relatively new programming practice in the enterprise space. Adding parallelism to existing applications and techniques only scratches the surface of the performance to be gained versus using parallelism as a central organizing construct in the program's construction. That approach is not easy to fully embrace for a variety of reasons (e.g. programmer experience with parallelism, uniprocessor versus multiprocessor system), but one of the most practical concerns is that a fully parallel tool may cause data interdependencies between threads, potentially introducing data integrity issues.

Many technical approaches hedge against this data integrity issue through processor locking mechanisms that essentially freeze data until cycles complete and interdependencies can be checked and/or corrected between threads. But, this locking approach robs performance as processors are idle and data from one thread is not available for another thread until cycles complete.

As datasets grow, this performance problem takes a larger and larger toll. Trovares has observed the performance gap becomes evident in the range of 250 million graph edges.

Performance also suffers from the way that a specific query or task propagates into parallelized multi-core, multi-processing systems. In simple terms, hunting for patterns in massive datasets requires breaking this huge search into smaller work units. Those smaller work units are then spread across threads that can be assigned to different processor cores and memory segments. Like parallelism, managing this process optimally is a complex undertaking. Performance can be robbed at several steps, from the syntax in the query to the way that work units are assigned.

Trovares can take advantage of all available cores in the system and drive them at near 100% utilization while ingesting or querying a graph.

The HPC world has always worked on massive problems (e.g. weather modeling, code cracking) that require that scale of computing. The Trovares approach draws on that history. It involves starting with a powerful modern programming language (C++) rather than Java or other popular enterprise tools. The main building blocks are:

- Lightweight synchronization to minimize the effect of locking mechanisms
- A run time system that keeps work flowing to all processors at scale

Lightweight synchronization ensures performance

Speed and scalability are related issues that require an architectural approach rather than just throwing more processor cores or memory at the problem. The Trovares approach uses parallelism inherent in modern architectures while minimizing the performance penalties produced by locking mechanisms. Locking mechanisms indicate a serial approach to data processing that trades off performance to achieve data integrity as the task's scale increases. Trovares achieves performance, scale and data integrity.

The key is a technique called "lightweight synchronization." Locking mechanisms are avoided through a message passing system that keeps different processor cores and threads coordinated without locking processes. All threads are independent and can move core to core to keep all processors active. Individual threads are dynamically allocated so the tool can scale to all available processor cores.

Optimization – an intelligent run-time system

Optimizing any individual pattern search is key to producing insights in a useful amount of time. Trovares achieves this optimization through an intelligent run-time system and Trovares Query Language.

The intelligent run-time system is built on a foundation of the open source Threading Building Block (TBB) layer, a C++ template library developed by Intel to spur multi-core multi-threaded processing. With this base layer, Trovares can break tasks into fine-grained lightweight work units. Those units can be created as needed rather than requiring the entire problem to be broken apart up front. TBB provides a "work-stealing" scheduler that ensures each core in a multi-core processor is fully utilized at all times.

Atop this TBB capability, Trovares designed its run-time system that maps the irregular workload produced by graph algorithms to the underlying tasking layer. The combination ensures that the workflow is optimized for the graph analysis problem at hand.

The Trovares Query Language (TQL) adds another way to optimize individual queries so they maximize this multi-core parallelized system. TQL is a subset of the declarative OpenCypher query language tuned to the run-time system. It arranges the order of operations to suit the problem. By formatting the queries properly, TQL can reduce the active dataset running on any one core for faster results. This speed advantage also allows for more complex queries and potentially better insights into the problem under study.

With lightweight synchronization and optimization techniques, Trovares is orders of magnitude faster than other tools, measured on 1 billion edge graphs. Essentially, Trovares can scale as large as the available memory space with no performance penalties. The result of these unique approaches is a property graph analytics tool that scales smoothly to the size and complexity of the dataset

A growing technology roadmap

Trovares continues to invest in its toolset and has an established technology roadmap that includes innovations such as:

- Integration with Machine Learning tools
- Integration with data visualization tools and user interfaces
- Further query enhancements
- Support for new data formats and algorithms

These innovations will maintain Trovares leadership in graph analytics and provide an evolutionary path for customers.

Trovares brings a long history in solving big problems. Trovares has been funded by government agencies with large datasets.

Big problems are now common in the enterprise space. Cybersecurity, fraud detection and medical research are three such problems with large opportunities for graph analytics tools. Trovares is bringing its expertise in lightweight synchronization and optimization to this market for the first time.