CASE STUDY

Analyzing Billions of Rows of Log File Data

Highlights:

- Performs ad-hoc queries of 7 billion row datasets
- Achieves millisecond lag time
- Provides minimal server & infrastructure costs

npm: The Package Manager for Node.js

npm, Inc. is the company that hosts and manages npm, the most widely used package manager for JavaScript. The npm registry hosts over a quarter million packages of reusable code – the largest code registry in the world. It is used daily by 4 million developers worldwide with 4.5 billion packages downloaded every month. Because of npm's keen focus on the long term success of the JavaScript community – including the open-source Node.js and npm projects, it is used by more than 30,000 companies, including Docusign, SiriusXM, Uber, and Visa, to manage and deploy their code packages.

npm & OmniSci

npm uses the OmniSci analytics database for exploring request log (log file) data. Request logs are a record of every request that the npm server has processed.

In production, npm runs OmniSci on an Amazon EC2 r3.8xlarge instance and receives approximately 700 million events per day which are bulk-loaded hourly. Currently they keep a total of 10 days worth of data - approximately 7 billion rows. The log file data contains information such as date/timestamp, JavaScript package name, node and npm version number, proxy cache server point of presence (PoP), region - even the npm commands issued.

npm has a particular challenge, namely it experiences a tremendous number of ad-hoc queries that cannot be predicted in advance such as requests from larger JavaScript package providers for information about how their packages are being utilized, trends in JavaScript development and changes to the versions of Node.js that developers use.

The same log file data that provides trends are also used for diagnostic purposes.

This may range from looking at all requests in a given data center to find a faulty node, filtering requests from a specific user agent or IP that for anomalous or failing requests. npm also looks for changes to regular usage patterns in the log files such as when a remote IP suddenly spikes, possibly indicating a problem, or simply a large new customer.

Why OmniSci

npm considered alternative log file analysis technology but found the price/performance attributes to be lacking. In OmniSci, npm found lightning fast response times without any requirement to index the data. Further, after experimenting with various open source data fabrics, npm found competing solutions couldn't scale or demanded more operational effort and hardware than npm's small but talented team could spare.

OmniSci delivered millisecond lag on multi-billion row datasets with a single server. As a result, npm is able to deliver against their varied objectives: superior performance, less administration and minimal server and infrastructure cost.

To find out if OmniSci would be a fit for your problem, reach out to us at sales@omnisci.com to schedule an appointment with our application and industry specialists.

OmniSci Overview

OmniSci is a next-generation database and visualization layer that harnesses the parallel power of GPUs to explore multi-billion row datasets in milliseconds. By leveraging the massive parallelism of commodity GPUs, OmniSci can query and visualize data up to 1000x faster than traditional systems and can render the results using the native graphics pipeline of the GPUs – resulting in immersive data exploration experiences.

"OmniSci lets us answer questions about our community and explore trends in all the different dimensions of our data in real time. We're excited about OmniSci Cloud, which gives us all that power in a convenient, scalable way."

— **Laurie Voss, Chief Operating Officer**

Learn More

To learn more about npm, visit npmjs.com.

For further information visit:
www.omnisci.com