gridgaintech.wordpress.com

Distributed Caching is Dead – Long Live…

By Nikita Ivanov

5-6 minutes

> In the last 12 months we observed a growing trend that use cases for distributed caching are rapidly going away as customers are moving up stack… in droves.

Let me elaborate by highlighting three points that when combined provide a clear reason behind this observation.

Databases Caught Up With Distributed Caching

In the last 3-5 years traditional RDBMSs and new crop of simpler NewSQL/NoSQL databases have mastered the in-memory caching and now provide comprehensive caching and even general in-memory capabilities. MongoDB and CouchDB, for example, can be configured to run mostly in-memory (with plenty caveats but nonetheless). And when Oracle 12 and SAP HANA are in the game (with even more caveats) – you know it’s a mainstream already.

There’s simply less reasons today for just caching intermediate DB results in memory as data sources themselves do a pretty decent job at that, 10GB network is often fast enough and much faster IB interconnect is getting cheaper. Put it the other way, performance benefits of distributed caching relative to the cost are simpler not as big as they were 3-5 years ago.

Emerging **“Caching The Cache” anti-pattern** is a clear manifestation of this conundrum. And this is not only related to historically Java-based caching products but also to products like Memcached. It’s no wonder that Java’s JSR107 has been such a slow endeavor as well.

Customers Demand More Sophisticated Products

In the same time as customers moving more and more payloads to in-memory processing they are naturally starting to have bigger expectations than the simple key/value access or full-scan processing. As MPP style of processing on large in-memory data sets becoming a new “norm” these customers are rightly looking for advanced clustering, ACID distributed transactions, complex SQL optimizations, various forms of MapReduce – all with deep sub-second SLAs – as well as many other features.

Distributed caching simply doesn’t cut it: it’s a one thing to live without a distributed hash map for your web sessions – but it’s completely different story to approach mission critical enterprise data processing without transactional data center replication, comprehensive computational and data load balancing, SQL support or complex secondary indexes for MPP processing.

Apples and oranges…

Focus Shifting to Complex Data Processing

And not only customers move more and more data to in-memory processing but their computational complexity grows as well. In fact, just storing data in-memory produces no tangible business value. It is the processing of that data, i.e. computing over the stored data, is what delivers net new business value – and based on our daily conversations with prospects the companies across the globe are getting more sophisticated about it.

Distributed caches and to a certain degree data grids missed that transition completely. While concentrating on data storage in memory they barely, if at all, provide any serious capabilities for MPP or MPI-based or MapReduce or SQL-based processing of the data – leaving customers scrambling for this additional functionality. What we are finding as well is that just SQL or just MapReduce, for instance, is often not enough as customers are increasingly expecting to combine the benefits of both (for different payloads within their systems).

Moreover, the tight integration between computations and data is axiomatic for enabling **"move computations to the data" paradigm** and this is something that simply cannot be bolted on existing distributed cache or data grid. You almost have to start form scratch – and this is often very hard for existing vendors.

And unlike the previous two points this one hits below the belt: there's simply no easy way to solve it or mitigate it.

So, what's next? I don't really know what the category name will be. May be it will be Data Platforms that would encapsulate all these new requirements – may be not. Time will tell.

At GridGain we often call our software end-to-end in-memory computing platform. Instead of one do-everything product we provide several individual but highly integrated products that address every major type of payload of in-memory computing: from HPC, to streaming, to database, and to Hadoop acceleration.

It is an interesting time for in-memory computing. As a community of vendors and early customers we are going through our first serious transition from the stage where simplicity and ease of use were dominant for the early adoption of the disruptive technology – to a stage where growing adaption now brings in the more sophisticated requirements and higher customer expectations.

As vendors – we have our work cut out for us.