Think of Latency as a Pseudo-permanent Network Partition

Thursday, August 12, 2010 at 8:21AM

Todd Hoff in CAP, Strategy

The title of this post is a quote from Ilya Grigorik's post Weak Consistency and CAP Implications. Besides the article being excellent, I thought this idea had something to add to the great NoSQL versus RDBMS debate, where Mike Stonebraker makes the argument that network partitions are rare so designing eventually consistent systems for such rare occurrence is not worth losing ACID semantics over. Even if network partitions are rare, latency between datacenters is not rare, so the game is still on.

The rare-partition argument seems to flow from a centralized-distributed view of systems. Such systems are scale-out in that they grow by adding distributed nodes, but the nodes generally do not cross datacenter boundaries. The assumption is the network is fast enough that distributed operations are roughly homogenous between nodes.

In a fully-distributed system the nodes can be dispersed across datacenters, which gives operations a widely variable performance profile. Because everything talks over TCP/IP, communication between servers is exactly the same, but what changes are characteristics of the channel like latency, bandwidth, and reliability. What also changes are locations of resources like disk storage. A datacenter in Europe, for example, wants to store data locally rather than ship it synchronously across the pond to a US datacenter. Fully distributed systems are very complex because they

must manage replication, availability, transactions, fail-over etc. in this very dynamic environment.

The notion is that latency is a kind of partition that requires eventual consistency to mask. The roundtrip time for a packet in a datacenter, for example, might be .5ms, and the roundtrip time from California to Europe back to California might be 150ms. So roughly 300 times more messages can be exchanged in a datacenter versus between datacenters. This implies that reads from different datacenters are unlikely to be consistent.

Imagine a write coursing down two network paths. The first path takes .5ms and the second takes 150ms. Without a strong consistency guarantee, as in a potentially performance/scaling/availability killing two-phase commit, a read will be inconsistent for those latency windows. Ilya writes:

> *Interestingly enough, dealing with network partitions is not the only case for adopting "weak consistency". The* PNUTS *system deployed at Yahoo must deal with WAN replication of data between different continents, and unfortunately, the speed of light imposes some strict latency limits on the performance of such a system. In Yahoo's case, the communications latency is enough of a performance barrier such that their system is configured, by default, to operate under the "choose availability, under weak consistency" model - think of latency as a pseudo-permanent network partition.*

Does this mean geographically disperse systems by their very nature must be eventually consistent? It's at least interesting think about as we slowly make our way towards truly distributed architectures.

Related Articles

Yahoo!'S PNUTS Database: Too Hot, Too Cold Or Just Right?
Consistency Across a WAN by MARK CALLAGHAN. *There are three solutions for providing consistency in a data service that operates across a wide area network (WAN). None of them are free.*
WAN Accelerate Your Way To Lightening Fast Transfers Between Data Centers
Numbers Everyone Should Know
Latency is Everywhere and it Costs You Sales - How to Crush it

---

Article originally appeared on High Scalability (http://highscalability.com/).

See website for complete article licensing information.