

AWS This is My Architecture Outline - WANdisco

The discussion will be technically focused, covering three main topics:

- What is unique about WANdisco?
- What is the biggest problem that we solve?
- How do we do it?

What does WANdisco do?

- WANdisco implements wide-area network distributed computing, which means:
- We solve the challenge of providing access to the same data from multiple locations or separate environments, guaranteeing that changes made in any location are available to all.
- This lets you implement disaster recovery, perform backup, ingest data in multiple physical locations, and support hybrid cloud environments where all your resources can contribute in an active manner. We call this active data replication, and do it at big data scale.

Here's how we do it (3 key features that make this work)

- Distributed consensus engine enforces agreement on intent rather than outcome
 - Each system agrees that it will make a change to the underlying storage before making that change. Because all environments execute these agreements in the same order, they all stay in lock-step, and are guaranteed to remain consistent.
- We implement that consensus engine in a way that works regardless of the distance between participants.
 - Distance equals latency. The speed of light down an optical fiber is probably slower than you expect. It's about 200,000 km/s. That means a round-trip latency between the US and Europe is quite large. Commercial network providers offer SLA guarantees of only 90ms between New York and London.
 - This degree of latency plays havoc with standard consensus protocol implementations. Anyone who's attempted to make Raft operate across a wide-area network can tell you that it just doesn't work.
 - WANdisco has solved this problem, which is the core of our technology that has been in the market for over 10 years. We call it DConE, or the Distributed Coordination Engine.
- WANdisco separates strongly consistent metadata changes from eventually available content
 - Another performance challenge that comes with big data environments is the sheer quantity of information. When replicating that data, you don't want the overhead of consensus to get in the way of content replication. Ideally, applications should continue to work as though they're just talking with their local cluster.
 - By separating the coordination of metadata from the replication of file content, applications can interact at local speed with their file or object store, but guarantee that those changes are consistent with the other environment to which they replicate.

What does this give someone using WANdisco?

- Our active replication lets you use the same content in multiple locations, guaranteeing that changes made anywhere are available everywhere.
- It works across different types of storage
 - WANdisco can replicate consistently between any Hadoop-compatible file system, including HDFS, Openstack Swift, Isilon, Amazon S3, a Netapp filer, and more.
 - So you could use HDFS in my on-premises cluster, and have that replicate consistently to S3 in AWS. I can spin up an Elastic Map-Reduce cluster to perform analytics on my S3 data, writing back to S3 like normal, but I then have those results available in my in-premises cluster. This is a true hybrid cloud deployment.
 - If I want to leverage Snowball for a bulk data upload, and WANdisco to continue to replicate while that device is in transit, I can do so.

Architecture Diagram

