Resilient Data Staging Through MxN Distributed Transactions

Jai Dayal, Jay Lofstead, Karsten Schwan, Ron Oldfield
jdayal3@gatech.edu, jhofst@sandia.gov, schwan@cc.gatech.edu raoldf@sandia.gov

Motivation

• Data staging techniques provide no guarantees about the data movement
• NoSQL-style eventual consistency not applicable for interactive online workflows
• Large number of resources increases potential for faults
• Database-style ACID transactions have not been applied to an MxN environment

Project Goals

• Bring ACID style guarantees to data staging
  • Atomicity allows us to ensure successful completion of our operations
  • Consistency allows us to ensure our data is up to date
  • Isolation shields operations from interfering with each other
  • Durability ensures that once our operations have completed, they are not lost in the face of system failures

Solution

• Distributed MxN transactions
  • Extend current distributed transaction (1xN) semantics
  • Distributed Transactions with many coordinated clients (M) and many coordinated servers (N)
• Must be scalable
  • Large number of clients and servers leads to high message volumes (MxN)
  • Too much overhead will reduce the gains associated with using data staging

Challenges

• HPC environments have unique characteristics
  • Operate at extreme scales
  • Extremely large data volumes
• Data staging systems hold data in volatile memory
  • Any performance requirements limit ability to delay computation to ensure correctness and completeness of our I/O operations.
• Online workflows require data guarantees
  • Data movement/processing complete prior to the next phase starting
  • Only correct (non-corrupted) data sets should be visible and processed
  • Data should not be removed from one queue prior to the successful insertion into the next (and the insert/delete done atomically)

Initial Implementation

• Dual Coordinators
  • Reduces problem to 1 to 1 coordination and thus reduces the volume of messages by avoiding all-to-all communication
  • Improves scalability
  • But, localized bottlenecks that may not scale
• 3 stages in a given transaction
  • Init Phase: client side initializes transactions and sub-transactions
  • Read/Write Phase: Clients perform read/write
  • Vote Phase: Clients and servers vote on success of operations
• Transactions and Sub-Transactions
  • I/O consists of many writes of many variables
  • Transaction: Groups operations in one output phase
  • Sub-transaction: represents one operation (or variable) in the overall transaction

Benefits

• Atomicity
  • Protocol extends upon traditional 2-Phase commit to operate in MxN environments
  • Provides guarantee that all operations have completed (atomic = all or none)
  • Correctness can be ensured by adding hashes (SHA-1, MD5, etc) to data
  • Applications are shielded from incomplete or erroneous data sets
• Durability, Consistency, Isolation
  • Future work
  • Durability: can be implemented by replicating operations on other nodes. Also possible to investigate an in memory RAID system or local SSD
  • Consistency: eventual consistency models fall short for HPC, as re-processing stale data yields no scientific insight.
  • Isolation: must ensure operations do not interfere with each other. Especially important as shared staging becomes more prominent

Fig. 1: Example Staging Area

Fig. 2: Logical Protocol

Fig. 3: Preliminary Results

Sandra National Laboratories is a multi-program laboratory managed and operated by Sandia Corporation, a wholly owned subsidiary of Lockheed Martin Corporation, for the U.S. Department of Energy’s National Nuclear Security Administration under contract DE-AC04-94AL85000.

References