Simplify High Performance Computing for Scientists using SOA

Jaison Paul Mulerikkal

Art of SOA
The art of service oriented approach to high performance scientific computing lies in breaking up big chunks of serial computations in to smaller atomic units (tasks) that can be parallelized to get it done fast in commodity hardware computational units (nodes).

Conjugate Gradient Solver (CGS)
CGS is a scientific application that involves iteration of two major mathematical operations - computation (Matrix-Vector Multiplication) and reduction (Vector-Vector operations), where tasks in each iteration can be called as “generation of tasks”. The value of r (residual - a function of above operations) determines the end of iteration.

Greatest Challenge for SOA for Scientific HPC
Because of the interdependency of parameters in scientific applications, the tasks become too small and number of tasks increases. So, the startup cost to begin tasks and the communication cost to send its results back to Client may outperform the computational advantages of distributing the work.

How Happy Scientists Would Program CGS?!
Client:
- Set Matrix and Vector as Data Service objects.
- Trigger “first generation” Matrix-Vector Multiply tasks.
- Trigger “first generation” r computation tasks.
- Query r from Data Service at intervals.
- When r within range, fetch final Vector from Data Service.

Service:
- Fetch (partial) Matrix and Vector from Data Service
- Compute (partial) Matrix and Vector multiplication.
- Update (partial segment of) Vector (at Data Service).

Data Service supports these operations and enforces a synchronization point as generation of task completes and upon reduction.

Client, Service and Manager
In SOA model, these atomic units of computations are considered as “services” which are at the disposal of the consumers who are called “clients” So, there are Clients and Services!
The Client should be able to access available resources (processing power, storage capacity and memory of available commodity hardware units) to get Services done in an orderly and effective manner. This is done by a Resource Manager.

Decouple task granularity (fineness of tasks) from startup and communication costs!

A Data Service to Implement Solution
Solution is to allow Service nodes to accept tasks, finish them one by one and update data back to a Data Service (rather than sending to Client) and start “next generation of tasks” on the basis of newly updated data synchronized at definite points. This highly efficient and fault-tolerant data-service around the Service and Management nodes of grid middleware shall provide a distributed space to access, modify and commit data based on programmer defined policies.

Why scientists are unhappy?
Scientists were making use of the exceptional computing abilities of high performance computing (HPC) in a grid environment for quite long time. But they are unhappy!

Designing HPC grid applications (writing programs!) for scientific problems is found to be too complicated for scientists (especially when they are not Computer scientists!).

What does make scientists happy?
Service Oriented Approach (SOA)! It is a computing paradigm that considers services as building blocks for applications. This model is easy to program (any scientist can do it) and fault tolerant (can restart the work, even if something go wrong!)

Supervisor: Peter Strazdins
Supported by: Platform Computing

Computer Systems Research Group
College of Engineering & Computer Science