Experiments with Wide Area Data Coupling Using the Seine Framework

L. Zhang, M. Parashar and S. Klasky
The Applied Software Systems Laboratory,
ECE/CAIP, Rutgers, The State University of New Jersey

HiPC 2006, Bangalore, India
December 18 – 21, 2006
Outline

• Introduction Problem Statement
• Related Work
• The Seine Approach
• The Fusion Simulation Project
• Wide-area Data Coupling Using Seine
• Conclusion
• Future Work
Coupled Scientific Simulations

• Motivation
  – Accurate solutions to realistic models of complex phenomena by modeling multiple interacting physical processes that comprise the phenomena being modeled

• Problem Statement
  – Coupling of multiple physical models and associated parallel codes that execute independently and in a distributed manner
    • Different (and possibly dynamic) distributions
    • Dynamic and complex communication and coordination patterns which depend on state of the phenomenon being modeled and are only known only at runtime

• Example
  – In plasma science, an integrated predictive plasma edge simulation couples an edge turbulence code with a core turbulence code through common grids at the spatial interface.
  – In geosciences, multiple physics models are coupled via shared boundaries between neighboring entities.
• Code Coupling in Scientific Computing
  – *Coupling between multiple physical models and associated parallel codes that execute independently and in a distributed manner*
  – *Multiple aspects:*
    – *algorithmic * numerical * computational*
  – *Our focus on computational aspects*

• Parallel Data Redistribution
  – *Transfer data from a parallel program running on M processors to another parallel program on N processors*
  – *Communication schedules, data flows*
The CPES Fusion Simulation Project (FSP)

GTC Runs on Teraflop/Petaflop Supercomputers

User monitoring

Visualization

40Gbps

End-to-end system with monitoring routines

Data archiving

Data replication

Large data analysis

Data replication

Post processing

User monitoring
Code Coupling and SciDAC CPES Project
Code Coupling and SciDAC CPES Project

• Requirements: Efficient, flexible, and scalable code coupling
  – Interaction/communication schedules between individual processors need to be computed efficiently, locally, and on-the-fly, without requiring synchronizations or gathering global information, and without incurring significant overheads on the simulation.
  – Data transfers are efficient and happen directly between individual processors of each simulation.

• Basic Approaches
  – Component-based approach
    • MCT, InterComm, PAWS, DDB, CUMULVS, etc.
  – PRMI-based approach
    • DCA, XCAT, SciRun, etc.
The Seine Approach

• The Concept
  – Provide a domain-specific virtual shared space abstraction to the application layer to facilitate dynamic and complex communication/coordination
  – Support intra-/inter-coupling for parallel scientific applications
  – Enable efficient and scalable implementation of the applications through domain-specific customization

• The Motivating Observations
  – Most parallel scientific applications have a notion of “space”
    • multi-dimensional geometric domain -- geometry-based space
    • other application aspects/degrees of freedom, such as temperature, energy, etc. -- temperature space, energy space, etc.
  – Formulations of the applications are based on a discretization of the “space”
    • e.g., mesh, grid, temperature levels, etc.
  – Interactions in the applications are based on the discretization and localized to a portion or sub-space of the global “space”
    • e.g., block coupling, model/code coupling, temperature swapping
Seine-based Parallel Data Redistribution Using Seine-Coupe

Conceptual view:  
Message passing approach  
to parallel data redistribution

Conceptual view:  
Seine-Coupe approach to  
parallel data redistribution
The Seine Architecture

- **The Seine Architecture**
  - A distributed directory layer
    - Uses domain-specific information to formulate a semantically specialized Distributed Hash Table (DHT)
    - Realizes the shared space abstraction
    - Facilitates dynamic and complex communication and coordination patterns
  - A storage layer
    - Provides local storage for objects/data in the shared space
  - A communication layer
    - Provides efficient peer-to-peer data transfers
  - Complements existing parallel programming systems (MPI, PVM, OpenMP, etc.)

- **Seine-based Prototypes**
  - Seine-Geo: supports coupled multiblock simulation
  - Seine-Coupe: supports code coupling and parallel data redistribution
  - Seine-Salsa: support asynchronous replica exchange
Seine-Coupe Operation
### Seine-Coupe Operators

<table>
<thead>
<tr>
<th>Operator</th>
<th>Usage</th>
</tr>
</thead>
</table>
| **init** | **init**(processor id, Seine runtime bootstrap server ip)  
/* init uses a bootstrap mechanism to initialize the Seine framework */ |
| **register** | **register**(object geometric descriptor)  
/* register registers a region with the Seine framework */ |
| **put** | **put**(object geometric descriptor, object data descriptor, tag)  
/* put inserts a geometric object into the Seine framework */ |
| **get** | **get**(object geometric descriptor, object data descriptor, tag)  
/* put retrieves a geometric object from the Seine framework */ |
Component View of Seine-Coupe

**User Component**

Port invocation:
- `init` \( (\text{my\_id}, \text{server\_id}) \);
- `register` \( (\text{geometric\ descriptor}) \);
- `put` \( (\text{geometric\ descriptor, object\ data\ descriptor}) \);
- `get` \( (\text{geometric\ descriptor, object\ data\ descriptor}) \);

**Seine-Coupe MxN Component**

- set up directory layer;
- [Directory layer] detects geometric relationships between regions registered;
- [Communication layer] socket data streaming to corresponding destination nodes;
- [Storage Layer] local memory copy from Seine storage to user component;

**Layer Diagram**

- Directory layer
- Communication layer
- Storage layer
An Sample Parallel Data Redistribution Scenario

Data redistribution between two direct-connected frameworks

* Note that we have also tested the Seine-Coupe based MxN component inside a direct-connected framework, assuming that components can be selectively active or inactive at any time.
Mock CPES Coupling Configuration

- Problem domain: a 3D toroidal ring
- Computation domain: a number of 2D poloidal planes
- Each plane contains a large number of particles
- Abstract 3D geometry-based space for data redistribution:
  - \textit{X-axis: particles; Y-axis: plane; Z-axis: variables associated with a particle}

8/16/32/64/128 procs on ewok cluster at ORNL.

Domain size varied as:
- (case1) 7,200x8x9,
- (case2) 14,400x8x9,
- (case3) 28,800x8x9

12 procs on frea cluster at CAIP.

Domain size varied as:
- (case1) 7,200x6x9,
- (case2) 14,400x6x9,
- (case3) 28,800x6x9

48 procs on frea cluster at CAIP.

Domain size varied as:
- (case1) 7,200x8x9/7,200x6x9,
- (case2) 14,400x8x9/14,400x6x9,
- (case3) 28,800x8x9/28,800x6x9
Experiment Results

Site M (XGC) per processor throughput

Site M (XGC) effective system throughput assuming 35% data transfer overlap

Site M (XGC) effective system throughput assuming 50% data transfer overlap
Experiment Results

[Graphs showing put cost with different data generation rates and Site M per processor throughput with different data generation rates]
Summary

• Seine provides an efficient and scalable framework for coupled simulations
  – *provides a simple but powerful abstraction based on semantically specialized shared spaces*
  – *addresses parallel data redistribution*

• Experiments using synthetic codes demonstrate that Seine can be potentially used for code-coupling within the fusion simulations project
  – *wide-area coupling and data-redistribution*

• We will continue to update the extend and tune Seine as the fusion codes develop and additional requirements become available.