Architecture and Performance of Runtime Environments for Data Intensive Scalable Computing

Student: Jaliya Ekanayake {jekanaya@cs.indiana.edu}

Advisor: Prof. Geoffrey Fox {gcf@indiana.edu}

*School of Informatics and Computing*

*Indiana University 919 E. 10th Street Bloomington, IN 47408 USA*

The deluge of data and the highly compute intensive applications found in many domains such as particle physics, biology, chemistry, and information retrieval, mandate the use of large computing infrastructures and parallel processing to achieve considerable performance gains. With the availability of many core CPU architectures and the advancements in virtualization techniques and cloud infrastructures, we can assume that the accessibility to computation power is no longer a barrier for the users who need to perform large scale data/compute intensive analyses.

Cloud technologies such as Google MapReduce, Hadoop, and Dryad have added new dimensions to parallel computing. Their support for handling large data sets, the concept of moving computation to data, and the better quality of services provided, simplify the implementation details of such problems over the traditional systems. However, applicability of these technologies to the diverse filed of parallel computing was not studied well.

Our initial research suggested that the cloud technologies are well applicable to most pleasingly-parallel problems, however, applications with complex communication patterns observe higher overheads when implemented using cloud technologies, and even with large data sets, these overheads limit the usage of cloud technologies for such applications. These observations raise questions: What applications are best handled by cloud technologies? How can we improve these programming models and the implementations? How would they operate on Cloud? These questions lead to the formulation of the following problem statement for my Ph.D.

“*Come up with an architecture and a prototype implementation of a new programming model based on MapReduce and data streaming techniques, and derive a performance model for the runtime by comparing it with other cloud technologies on both virtualized and non virtualized hardware platforms.”*

In pursuing the above goal, I have designed and developed CGL-MapReduce - a light-weight MapReduce runtime that incorporates several improvements to the MapReduce programming model wiz; (i) faster intermediate data transfer via pub/sub broker network, (ii) support for long running map/reduce tasks, (iii) efficient support for iterative MapReduce computations. When analyzing applications written in MapReduce programming model we can identify three basic execution units; (i) map-only, (ii) map-reduce, and (iii) iterative-map-reduce. Complex applications can be built by combining these three basic execution units under the MapReduce programming model. To derive a performance model for the MapReduce programming model, we have selected a series of data analysis applications and discussed their mapping to parallel architectures of different types and look at the performance of these applications. During the course of the research, we have implemented these applications using Hadoop, Dryad, and CGL-MapReduce and compared their performances. In addition, we have also evaluated the performance implications of MPI applications on virtualized resources and will be performing a similar analysis for cloud technologies as well.

Our findings reveal the usability of the MapReduce programming model for much data/compute intensive analyses. The Enhanced MapReduce runtime and the programming model we developed outperform other runtimes in most of the applications, and extend the usability of the MapReduce programming model for iterative MapReduce computations. The evaluations on Cloud environments reveal that the applications that are sensitive to latencies incur higher overheads while applications those are not susceptible to latencies, such as applications that perform large data transfers and/or higher communication/computation ratios show minimal total overheads when performed under Cloud environments.

My current research involves applying cloud technologies to different data/compute intensive applications, evaluating them on cloud infrastructures, and using the findings to improving the CGL-MapReduce implementation and its programming model. I plan to graduate in December 2009.

**Publications**

**Thesis Related Publications**

1. **Jaliya Ekanayake**, Geoffrey Fox, [High Performance Parallel Computing with Clouds and Cloud Technologies](http://grids.ucs.indiana.edu/ptliupages/publications/cloudcomp_submission.pdf) Technical Report June 12 2009. (Submitted for CloudComp 2009)
2. Geoffrey Fox, Seung-Hee Bae, **Jaliya Ekanayake**, Xiaohong Qiu, and Huapeng Yuan, [Parallel Data Mining from Multicore to Cloudy Grids](http://grids.ucs.indiana.edu/ptliupages/publications/CetraroWriteupJan09_v12.pdf), High Performance Computing and Grids workshop, 2008.
3. **Jaliya Ekanayake** and Shrideep Pallickara, [MapReduce for Data Intensive Scientific Analysis](http://www.cs.indiana.edu/~jekanaya/papers/eScience-final.pdf), Fourth IEEE International Conference on eScience, 2008, pp.277-284.
4. Shrideep Pallickara, **Jaliya Ekanayake**, Geoffrey Fox, [An Overview of the Granules Runtime for Cloud Computing](http://grids.ucs.indiana.edu/ptliupages/publications/GranulesOverview-eScience2008.pdf), Fourth IEEE International Conference on eScience, 2008, pp.412-413.
5. **Jaliya Ekanayake**, Shrideep Pallickara, and Geoffrey Fox, [A collaborative framework for scientific data analysis and visualization](http://www.ieeexplore.ieee.org/xpl/freeabs_all.jsp?isnumber=4543895&arnumber=4543948&count=86&index=52), [Collaborative Technologies and Systems(CTS), 2008](http://www.ieeexplore.ieee.org/xpl/RecentCon.jsp?punumber=4534816),pp. 339-346.
6. Shrideep Pallickara, **Jaliya Ekanayake** and Geoffrey Fox, [A Scalable Approach for the Secure and Authorized Tracking of the Availability of Entities in Distributed Systems](http://grids.ucs.indiana.edu/ptliupages/publications/NB-TrackingService.pdf) in the proceedings of Proceedings of the 21st IEEE International Parallel & Distributed Processing Symposium (IPDPS 2007). Long Beach, California.

**Other publications**

1. Srinath Perera, Chathura Herath, **Jaliya Ekanayake**, Eran Chinthaka, Ajith Ranabahu, Deepal Jayasinghe, Sanjiva Weerawarana,Glen Daniels [Axis, Middleware for Next Generation Web Services](http://csdl.computer.org/dl/proceedings/icws/2006/2669/00/26690833.pdf) on IEEE International Conference on Web Services (ICWS'06)
2. Eran Chinthaka, **Jaliya Ekanayake**, David Leake, CBR Based Workflow Composition Assistant, Accepted for publication, IEEE 2009 Third International Workshop on Scientific Workflows (SWF 2009).
3. Ajay Smitha and **Jaliya Ekanayake**, [Analysis of the Usage Statistics of Robots Exclusion Standard.](http://grids.ucs.indiana.edu/ptliupages/publications/IADISConferenceRobtoExclusion.pdf) In proceedings of the IADIS WWW/Internet 2006 Murcia, Spain 5-8 October 2006.