Abstract

The OpenFabrics Alliance has been the focal point for an open source project known as the OpenFabrics Interface (OFI) or libfabric. The desire for a new application-centric approach to developing fabric APIs was first expressed at a BoF held during SC13; this tutorial describes the API that the community of OFI developers designed to fulfill the challenges outlined at that BoF. The new library contains a set of fabric APIs that build upon and expand the goals and objectives of the original Verbs API but with a special emphasis on scalability, availability across a range of RDMA-capable networks and on serving HPC applications among others. Under active development by a broad coalition of industry, academic and national labs partners for two years, the new OFI API has matured to a point where it is ready for use by the SC15 community. This tutorial strengthens the engagement with the community of consumers are will benefit most directly from the new API by providing a high-level introduction, a fairly detailed look at the architecture and hands on experience with running simple applications over libfabric.

Detailed Description

Tutorial Goals

Under the auspices of the OpenFabrics Alliance, a new open source API known as the OpenFabrics Interface (OFI) is now ready for deployment by the HPC community. The recently released API is designed to meet the desire expressed by the HPC community during a BoF at SC'13 for a fabric API which is much more responsive to the actual needs of applications and middleware consumers. In response, a team consisting of middleware developers, API coders, and fabric vendors was formed to develop this new API. Guided by a concept described as "application-centric I/O", the API's fundamental architecture was driven from the outset by a set of requirements contributed by the consumers of network services, such as MPI and SHMEM middleware developers.

The key goal for this tutorial is to introduce this new network API to the HPC community and to those serving the HPC market with advanced networking technology. The second objective is to provide sufficient insight and detail into its technical details to enable an attendee to begin applying the API right away. The third objective is to accelerate the development of new implementations of the API, known as network providers, based on a range of existing network technologies.

Why the Topic is Relevant to SC15 Attendees

The genesis of this work lies in a BoF held during SC'13 entitled "Discussing an I/O Framework". The BoF was designed to assist the OpenFabrics Alliance in gauging the level of industry interest in developing an extensible, open source API aligned with application demand for high-performance fabric services. Active participants in the BoF included representatives from the MPI community, major network hardware vendors and key OEMs. The interest level at the BoF was taken by the OFA as a priori evidence of a high level of interest within the HPC community. Since the SC'13 BoF, a group of interested parties from academia, industry and government labs was formed under the auspices of the OpenFabrics Alliance with the express goal of developing and implementing a new API. The new API would be designed specifically to address the requirements of high performance parallel and distributed computing applications and middleware. The objectives would be to enable efficient application operation, scalability and performance and to do so in a way that is not tied to a specific underlying network technology. As expressed during the SC'13 BoF, these are topics of direct relevance to SC'15 attendees.

Targeted Audience

OFI is designed for consumers of network services who rely on network performance to enable scalability and application performance and efficiency. Hence, the tutorial is targeted at these consumers, notably the MPI community, PGAS language developers and the SHMEM community. Prior to beginning the development of OFI, extensive efforts were made to solicit these communities both to gauge interest and to provide requirements. In particular, input was sought from the national labs (LANL, ORNL, Sandia, LBL, ANL), as well as the MPI community (particularly the MPI Forum). Based on the feedback received during this requirements gathering phase, we believe that all of these consumers will find this tutorial important as they begin deploying the API. As an example, OFI is planned to be deployed as part of Argonne's Aurora acquisition, resulting from the DOE's CORAL procurement program. Beyond Aurora, OFI will be available on a variety of systems spawned as a result of the CORAL program. We expect that industries deploying commercial HPC systems such as those based on Aurora, such as the oil and gas industry, will also find this tutorial compelling.

In addition to consumers of the API, we expect to draw attendees from the vendors of networking technology who are engaged in developing the so-called provider software that enables the API to be deployed over various transports. This would include classical Ethernet as well as vendors developing advanced networks based on Ethernet such as RoCEv2.

And finally, the OpenFabrics Alliance has committed to actively marketing this emerging technology in order to ensure that we are successful in reaching these targeted audiences.

Content Level

Given that the OFI API is new, there isn't really yet such a person as an intermediate or advanced user. Therefore, although the tutorial will be technical, it is by definition 'beginner'. Nevertheless, as the tutorial progresses the material will likewise progress in content level. This is important in order to meet the audiences' need to emerge with a clear understanding of the motivation and architecture for the API as well as hands on exposure to working with it through some simple use cases.

Audience Prerequisites

The audience is expected to have significant background in either middleware written to an existing network API, or an understanding of sockets and sockets programming, or basic familiarity with RDMA-based networks such as InfiniBand, iWARP or RoCE. Detailed knowledge of programming to the sockets API will be very helpful but is not required.

General Description of the Tutorial Content

OFI is the result of a new approach to creating a network API which places an emphasis on understanding how middleware and applications typically prefer to access network services. Understanding these usage models was a key to defining the API.

The tutorial begins by describing the insights gained by focusing on middleware consumers of a new API and by describing the resulting motivations for creating a new API. To provide further insight into the design, we also describe the unusual partnership between HPC middleware and application developers, fabric vendors and network experts and describe how this partnership drove the API's development.

The second section comprises the bulk of the technical material and is devoted to exploring the major features of the API. These features appear as a set of four sets of services; a set of required control services, communication establishment services, completion services and the data transfer service.

Rounding out the students' understanding of OFI, the third section delves into the object model that forms the basis of the architecture.

The last section of the tutorial is designed to give students enough exposure to an implementation of the API to begin experimenting with it right away. To accomplish this, we use a fully featured sockets provider (implementation) that is included with the API specifically to allow easy experimentation with it using a standard laptop and a straightforward pingpong example.

Cohesive Content

The OpenFabrics Alliance is chartered to foster an inclusive, open source environment to foster collaboration among representatives of different companies and institutions to develop advanced network technology. Under the auspices of the OFA, the four presenters have been central figures in developing the new API; each has been an active participant in weekly teleconferences and bi-annual face-to-face meetings over the past two years, as well as active contributors to the code currently publicly available via GitHub. Thus the group of presenters begins as a cohesive team. We plan to increase that cohesion by devoting time during the team's weekly teleconferences to developing and refining this tutorial, ensuring that it represents the work product of the group as a whole.

Detailed Outline of the Tutorial

- Introducing OFI – Its motivation and the community that created it
  - How API design can limit scalability and performance
  - Interfaces co-designed with HPC developers and fabric vendors
  - Mapping application and middleware usages to API requirements
  - Highlight of features integrated into OFI
- Introduction to OFI architecture highlighting 4 main interface sets
  - Control Services
    - Discovery of available fabric services
    - Application desired capabilities versus vendor optimized usage models
  - Communication Services
    - Connection establishment
    - Scalable address resolution interfaces and services
  - Completion Services
    - Events queues versus completion counters
  - Data Transfer Services
    - Basic message transfers
    - Tag matching interface
    - Remote memory access (RMA) transfers
    - Atomic operations
    - Triggered operations
- OFI object model and relationships
  - Basic endpoint usage
  - Shared transmit and receive queues
  - Scalable endpoints
- Example code walk-through
  - Analysis of simple pingpong example using basic message transfers
  - Extending example to take advantage of MPI-focused tag matching interface
  - Example of scalable, SHMEM-focused RMA operations

Hands on Exercises

OFI has been implemented over several providers; key among them is a fully featured sockets provider, included primarily for purposes of experimenting with and understanding the API. The sockets provider (implementation) is designed for developmental purposes. During the hands on exercise, students will have an opportunity to explore this provider and through it to gain some insights into the design of the API.

A hands-on exercise will be given in the last 1-2 hours of the tutorial that makes use of a select number of developer focused examples. The hands-on exercise will include:

- Code walk-through of a simple pingpong example, demonstrating how an application bootstraps into the OFI framework.
- Code changes (via a second example) needed to switch from a simple message data transfer interface to an MPI-focused tag matching interface.
- Example code that highlights OFI's RMA features desired to implement a scalable SHMEM.

OFI and all samples run over both Linux and OS X platforms, including running within virtualized environments.

CURRICULUM VITAE
Jeffrey M. Squyres

Cisco Systems, Inc.
170 W. Tasman Dr., San Jose, CA 95134
+1 (408) 525-0971 / jsquyres@cisco.com

Educational Background

University of Notre Dame 1996-2004
Degree: Ph.D., Computer Science and Engineering, May 2004
University of Notre Dame 1994-1996
Degree: M.S., Computer Science and Engineering, May 1996
University of Notre Dame 1989-1994
Degree: B.S., Computer Engineering, May 1994
University of Notre Dame 1989-1994
Degree: B.A., English, May 1994

Relevant Professional Experience

MPI Forum 2009-Present
Position: Secretary
Cisco Systems, Inc. 2006-Present
Position: Technical Lead
Department: Server Access and Virtualization Technology Group
Indiana University 2005-2006
Position: Assistant Director, HPC
Department: Open Systems Laboratory, Pervasive Technologies Laboratory
Indiana University 2004-2005
Position: Post-Doctoral Research Associate
Department: Open Systems Laboratory, Pervasive Technologies Laboratory
Indiana University 2001-2004
Position: Research Associate
Department: Open Systems Laboratory, Pervasive Technologies Laboratory
University of Notre Dame 1996-2004
Position: Ph.D. candidate
Department: Computer Science and Engineering
Thesis Advisor: Dr. Andrew Lumsdaine
Thesis Research: Component architectures for high-performance MPI implementations

Short Courses Taught

IBM Benchmarking Center 2008
Course: Tuning Open MPI over InfiniBand
NCSA MPI Training 2007
Course: NCSA Open MPI Reference Training
Imperial College, London, England 2007
Course: Imperial College InfiniBand / Open MPI User-Level Training
IBM Cross-Training 2007
Course: IBM InfiniBand / Open MPI User-Level Training
Cineca, Italy 2007
Course: Cineca Open MPI User-Level Training
Cisco-Sponsored Open MPI Developer Meeting 2006
Course: Open MPI Developer Workshop
SC Tutorial 2004
Course: Taking Your MPI Application to the Next Level: Threading, Dynamic Processes, and Multi-Network Utilization

Recent Presentations / Invited Talks

1. Jeffrey M. Squyres, George Bosilca, et al., "Open MPI: State of the Union," Birds of a Feather Presentation, ACM/IEEE Supercomputing Conference, USA, November each year 2008 through 2015.

2. Jeffrey M. Squyres, "The State of Libfabric in Open MPI," OpenFabrics Alliance Workshop, Monterrey, California, USA, March 2015.

3. Jeffrey M. Squyres, "Experiences Building an OFI Provider for usNIC," OpenFabrics Alliance Workshop, Monterrey, California, USA, March 2015.

Relevant General Publications

1. "The Architecture of Open Source Applications, Volume 2," edited by Amy Brown and Greg Wilson. Chapter 15: "Open MPI," written by Jeffrey M. Squyres. June, 2012.

2. "Attaining High Performance Communication: A Vertical Approach," edited by Dr. Ada Gavrilovska. Chapter 11: "The Message Passing Interface," written by Jeffrey M. Squyres. September, 2009.

Relevant Software Packages

1. Open MPI, `http://www.open-mpi.org/`

2. Libfabric, `http://ofiwg.github.io/libfabric/`

3. Hardware Locality ("Hwloc"), `http://www.open-mpi.org/projects/hwloc/`

4. LAM implementation of MPI, `http://www.lam-mpi.org/`

Robert D. Russell

Associate Professor of Computer Science
Department of Computer Science
University of New Hampshire
Kingsbury Hall
33 Academic Way
Durham, NH 03824-2619
e-mail: rdr@unh.edu
phone: (603) 862-3778

PROFESSIONAL PREPARATION

Yale University	Math and Physics	B.A.	1965
Stanford University	Computer Science	M.S.	1967
Stanford University	Computer Science	Ph.D.	1972

PROFESSIONAL EXPERIENCE

1981-present	Associate Professor	University of New Hampshire
2000-present	Adjunct	InterOperability Laboratory, UNH
2005-2006	Visiting Professor	Technical University of Harburg-Hamburg, Germany
1997-1998	Associated Professor	Laboratory for Parallel Computing, ENS Lyon, France
1991-1992	Consultant	CERN, Geneva, Switzerland
1991	Lecturer	CERN School of Computing, Ystad, Sweden
1989-1990	Scientific Associate	CERN, Geneva, Switzerland
1984-1988	Summer Visitor	CERN, Geneva, Switzerland
1986	Instructor	Bell Labs, Andover, Massachusetts
1982	Lecturer	Computer Science Institute, Santa Cruz, California
1981-1982	Scientific Associate	CERN, Geneva, Switzerland
1975-1981	Assistant Professor	University of New Hampshire
1977	Scientific Associate	CERN, Geneva, Switzerland
1974-1975	Programming Specialist	Burroughs Corp., Goleta, California
1971-1974	Visiting Scientist	CERN, Geneva, Switzerland
1972	Lecturer	CERN School of Computing, Pertisau, Austria
1971	Acting Assistant Professor	Computer Science Department, UCLA
1969-1970	Instructor summer session	Computer Science Department, Stanford University
1966-1970	Research Assistant	Stanford Linear Accelerator Center, Stanford University
1966	Systems Programmer	IBM Scientific Center, Palo Alto, California
1964-1965	Systems Programmer	Yale University

RECENT RELATED PUBLICATIONS

1. “File Multicast Transport Protocol (FMTP)”. Li, J., Veeraraghavan, M., Emmerson, S., and Russell, R.D., To be presented at the **2nd International Workshop on Scalable Computing For Real-Time Big Data Applications (SCRAMBL’15)**, Shenzhen, Guangdong, China, 4-7 May, 2015.
2. “IBRMP: a Reliable Multicast Protocol for InfiniBand”. Liu, Q., and Russell, R.D., presented at the **22nd Annual Symposium on High-Performance Interconnects (HOTI 22)**, Mountain View, California, 26-28 August, 2014.
3. “An Efficient Method for Stream Semantics over RDMA”. MacArthur, P., and Russell, R.D., **Proceedings of the 28th IEEE International Parallel & Distributed Processing Symposium (IPDPS)**, Phoenix, Arizona, 19-23 May, 2014.

4. "A Performance Study of InfiniBand Fourteen Data Rate (FDR)". Liu, Q., and Russell, R.D., **Proceedings of the 22nd High Performance Computing Symposium (HPC 2014)**, Tampa, Florida, 13-16 April, 2014.
5. "A Performance Study to Guide RDMA Programming Decisions". MacArthur, P., and Russell, R.D., **Proceedings of the 14th IEEE International Conference on High Performance Computing and Communication (HPCC-2012)**, Liverpool, UK, 25-27 June, 2012.
6. "A General-Purpose API for iWARP and InfiniBand". Russell, R.D., **First Workshop on Data Center - Converged and Virtual Ethernet Switching (DC CAVES)**, Renato Recio, chair, Paris, France, 14 September, 2009.
7. "The Extended Sockets Interface for Accessing RDMA Hardware". Russell, R.D., **Proceedings of the 20th IASTED International Conference on Parallel and Distributed Computing and Systems (PDCS 2008)**, T.F. Gonzalez, ed., Orlando, Florida, 16-18 November, 2008, pp 279-284.
8. "Implementation and Evaluation of iSCSI over RDMA". Burns, E., and Russell, R.D., **Proceedings of the 5th IEEE International Workshop on Storage Network Architecture and Parallel I/O (SNAPI'08)**, Baltimore, Maryland, USA, 22 September, 2008.
9. "iSCSI: Past, Present, Future". Russell, R.D., invited talk published in **Proceedings of the 2nd JST CREST Workshop on Advanced Storage Systems**, San Francisco, California, USA, 16-17 December, 2005, pp 121-148.

UNIVERSITY TEACHING AT SENIOR/GRADUATE LEVEL

Courses I have designed and taught regularly

1. Storage Systems and Storage Area Networks
2. Operating Systems Programming
3. Operating Systems Kernel Design with Linux

Courses I have also taught

1. Introduction to Computer Networks
2. Compiler Construction

SYNERGISTIC ACTIVITIES

Member of the organizing and steering committee for, and session speaker at, the 2013 OpenFabrics Alliance User Day Event, designed to provide interaction among active users of RDMA technology and feedback to developers of the OpenFabrics Software for RDMA.

Invited speaker on RDMA at various conferences, workshops, and organizations.

In conjunction with the OpenFabrics Alliance, Inc., have developed and regularly presented a professional training course entitled "Writing Application Programs for RDMA using OFA Software".

Member of the OpenFabrics Interfaces Working Group (lists.openfabrics.org/mailman/listinfo/ofiwg), whose charter is to develop, test, and distribute:

1. An extensible, open source framework that provides access to high-performance fabric interfaces and services.
2. Extensible, open source interfaces aligned with ULP and application needs for high-performance fabric services.

Howard Pritchard

Los Alamos National Lab
Los Alamos, NM 87545 USA

505-667-7718
howardp@lanl.gov
https://github.com/hppritcha

SUMMARY

Research scientist with extensive experience designing and implementing network stack components for High Performance networks for distributed memory computer systems. This includes experience working closely with hardware engineers in defining differentiated hardware features for such networks.

EXPERIENCE

Los Alamos National Lab 2014 - present

Research Scientist

- Technical lead for the Open MPI effort at LANL. Responsibilities include enhancing Open MPI to be effective at scale on the DOE Trinity and Cori Intel KNL-based platforms. As part of this effort, co-leading an open source effort to implement a libfabric provider for the Cray XC interconnect. Additional responsibilities include mentoring student interns, serving as co–release manager for the Open MPI 1.9/2.0 release stream, and giving internal and external presentations about the Lab's Open MPI and libfabric activities.
- In conjunction with collaborators at LLNL, investigate approaches to better fault tolerance schemes for MPI applications.

Cray Inc. 2003 -2014

Software Architect

- Worked as part of a future technologies architecture group. Responsibilities included determining opportunities for using the Hadoop eco–system on Cray supercomputers.
- Participated in the Open Fabrics OpenFrameworks WG to develop a common set of APIs for applications to efficiently utilize evolving RDMA–capable network interface controllers. Gave company internal presentations about the Work Group's activities.
- Implemented a Nemesis Network Module to interface MPICH to the high speed internal network of Cray XE and XC computer systems. Gave presentations both internally and to customers about how best to use MPICH on Cray systems.
- Worked as part of a team responsible for the design and implementation of the Generic Network Interface (GNI) and Distributed Memory Application (DMAPP) components of the network stack for the Cray XC and XE network fabric.

- Participated in the co–design of two generations of high performance RDMA–capable net- work interface controllers for the Cray XC and XE networks, working closely with hardware engineers to define differentiated hardware features that could be effectively usedbysoftware to deliver superior performance.

EDUCATION

Ph.D. in **Chemistry**, California Institute of Technology, Pasadena, CA.
B.A. in **Chemistry**, Rice University, Houston, TX.

SELECTED PUBLICATIONS

- H. Pritchard, I. Laguna, K. Mohror, T. Gamblin, M. Schulz, and N. Davis, "A global Ex- ception Fault Tolerance Model for MPI", Workshop on Exascale MPI, SuperComputing '14.
- H. Pritchard, D. Roweth, D. Henseler, and P. Cassella, "Leveraging the Cray Linux Envi- ronment Core Specialization Feature to Realize MPI Asynchronous Progress on Cray XE Systems", Cray User Group Conference, May 2012.
- H. Pritchard, I. Gorodetsky, and D. Buntinas, "A uGNI–Based MPICH2 Nemesis Network Module for Cray XE Computer Systems", Proceedings of the 18*th* EuroMPI Users' Group Conference, (2011).
- J. Hursey, R. L. Graham, G. Bronevetsky, D. Buntinas, H. Pritchard, and D. Solt, "Run– Through Stabilization: An MPI Proposal for Process Fault Tolerance", Proceedings of the 18*th* EuroMPI Users' Group Conference, (2011).
- H. Pritchard, D. Gilmore, and M. Pagel, "Message Passing Toolkit (MPT) Software on XT3", Cray User Group Conference, May 2006.
- H. Pritchard, J. Nicholson, and J. Schwarzmeier, "Optimizing MPI Collectives for the Cray X1", Cray User Group Conference, May 2004.
- R. P. Weaver, M. L. Gittings, M. R. Clover, and H. Pritchard, "The Parallel Implementa- tion of RAGE: A 3-D Continuous Adaptive Mesh Refinement Shock Code", International Symposium on Shock Waves, (1999).

PATENTS

- "Extended Fast Memory Access in a Multiprocessor Computer System", D. Abts, R. Alver- son, E. Froese, H. Pritchard, S. Scott, Patent Application Ser. No. 20100318626.
- "Method and Apparatus for Deadlock Avoidance", E. Froese, E. Lundberg, I. Gorodet- sky, H. Pritchard, C. Geifer, R. Alverson, D. Roweth, Provisional Patent

SEAN HEFTY

18560 SW Hart Rd, Aloha, OR 97007, 503-356-1845, sean.hefty@intel.com

EDUCATION

Arizona State University, Tempe, AZ

Masters of Computer Science 1996

Focus on computer architecture and database systems

Purdue University, West Lafayette, IN

Bachelor of Science - Computer Science and Mathematics 1993

Honors and Distinction in Computer Science and Mathematics

TEACHING EXPERIENCE

Oregon Tech, Beaverton, OR

Adjunct Instructor 1996 - 2008

Developed syllabus and overall course structure

Administered all grades

Courses: Computer Architecture, Computer Networking,

Database Systems, Economics, Grammars, MIS, Oracle DBMS,

SQL, Systems Analysis, Theory of Computing, Visual Basic

PROFESSIONAL EXPERIENCE

Intel Corporation, Hillsboro, OR 1995 - present

Senior Software Engineer

Lead software developer and maintainer for Linux kernel and

Open Fabrics InfiniBand and iWarp software. Co-chair of Open Fabrics Interfaces Working Group.

OTHER

11 patents

Phi Beta Kappa

Statement Agreeing to Release the Notes for the SC15 digital copy

We agree to release the notes for the SC15 tutorial digital copy.

Request for Travel Support, if any

No travel support is requested.