

# XPRESS

---

## eXascale PRogramming Environment and System Software

### Products

### Publications

1. Priyanka Ghosh, Yonghong Yan, and Barbara Chapman. Support For Dependency Driven Executions Among OpenMP Tasks. In *2nd Workshop on Data-Flow Execution Models for Extreme Scale Computing (DFM 2012) in conjunction with PACT 2012*.
2. Alice Koniges, Jean-Luc Vay, Alex Friedman, Hartmut Kaiser, and Thomas Sterling. Position Paper: Consideration of Asynchronous Algorithms for Particle-Grid Simulations, *Examath 2013*, Washington DC.
3. Ghosh, Priyanka, Yonghong Yan, Deepak Eachempati, and Barbara Chapman. "A Prototype Implementation of OpenMP Task Dependency Support." *In OpenMP in the Era of Low Power Devices and Accelerators*, pp. 128-140. Springer Berlin Heidelberg, 2013.
4. A. Charif-Rubial, D. Barthou, C. Valensi, S. Shende, A. Malony, W. Jalby, "MIL: A language to build program analysis tools through static binary instrumentation," *Proc. of HiPC Conference*, pp. 206-215, 2013.
5. T. Heller, H. Kaiser, A. Schäfer, and D. Fey, Using HPX and LibGeoDecomp for scaling HPC applications on heterogeneous supercomputers, *Scala '13, Proceedings of the Workshop on Latest Advances in Scalable Algorithms for Large-Scale Systems*, Article No. 1, 2013.
6. A. Charif-Rubial, D. Barthou, C. Valensi, S. Shende, A. Malony, W. Jalby, "MIL: A language to build program analysis tools through static binary instrumentation," *IEEE International Conference on High Performance Computing (HiPC)*, pp. 206-215, 2013.
7. Stephen Olivier, Bronis De Supinski, Martin Schulz, and Jan F. Prins, "Characterizing and Mitigating Work Time Inflation in Task Parallel Programs", *Scientific Programming* 21(3-4), pp 123-136, 2013.
8. Liao, Chunhua, Yonghong Yan, Bronis R. de Supinski, Daniel J. Quinlan, and Barbara Chapman. "Early experiences with the OpenMP accelerator model." *In OpenMP in the Era of Low Power Devices and Accelerators*, pp. 84-98. Springer Berlin Heidelberg, 2013.
9. Allan Porterfield, Stephen L. Olivier, Sridutt Bhalachandra, and Jan F. Prins. Power Measurement and Concurrency Throttling for Energy Reduction in OpenMP Programs. In *9th Workshop on High-Performance, Power-Aware Computing (HPPAC13 Workshop)*, Boston, MA, USA, May 2013.
10. H. Kaiser and V. Amatya. HPX: A C++ Standards Compliant Runtime System For Asynchronous Parallel And Distributed Computing. In *C++Now!*, Aspen, Colorado, May 2013.
11. Kevin Huck, Sameer Shende, Allen Malony, Hartmut Kaiser, Allan Porterfield, Rob Fowler, and Ron Brightwell. An Early Prototype of an Autonomic Performance Environment for Exascale. In *International Workshop on Runtime and Operating Systems for Supercomputers (ROSS 2013)*, Eugene, Oregon, USA, June 10, 2013.

12. Xiaonan Tian, Rengan Xu, Yonghong Yan, Zhifeng Yun, Sunita Chandrasekaran, and Barbara Chapman, Compiling a High-level Directive-Based Programming Model for *GPGPUs 26th International Workshop on Languages and Compilers for Parallel Computing (LCPC2013)*, September 2013.
13. Munara Tolubaeva, Yonghong Yan, and Barbara Chapman, Compile Time Modeling of Off-Chip Memory Bandwidth for Parallel Loops *26th International Workshop on Languages and Compilers for Parallel Computing (LCPC2013)*, September 2013.
14. Matthew Anderson, Maciej Brodowicz, Abhishek Kulkarni, and Thomas Sterling. Performance Modeling of Gyrokinetic Toroidal Simulations for a many-tasking runtime system. In *The 4th international workshop on Performance Modeling, Benchmarking and Simulation of High Performance Computer Systems at SC13*, Denver, CO USA, November 2013.
15. Allan Porterfield, Rob Fowler, Sridutt Bhalachandra and Wei Wang. OpenMP and MPI Application Energy Measurement Variation. In *Proceeding of the 1st Workshop on Energy Efficient Super Computing*, Denver CO, USA, Nov 2013.
16. Antoine Tran Tan, Joel Falcou, Daniel Etiemble, Hartmut Kaiser, Automatic Task-based Code Generation for High Performance Domain Specific Embedded Language, *HLPP 2014: 7th International Symposium on High-Level Parallel Programming and Applications* (2014).
17. J.N. Brooks, A. Hassanein, A. Koniges, P.S. Krstic, T.D. Rognlien, T. Sizyuk, V. Sizyuk, and D. P. Stotler, "Scientific and Computational Challenges in Coupled Plasma Edge/Plasma-Material Interactions for Fusion Tokamaks", *Contrib. Plasma Phys.* 54, No. 4-6, 329 – 340, 2014.
18. J.-L. Vay, L. A. Drummond, A.E. Koniges, B. Godfrey, I.Haber, "Scalable Arbitrary-Order Pseudo-Spectral Electromagnetic Solver," Poster and two-page paper in *Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis (SC14)*, 2014.  
[http://sc14.supercomputing.org/sites/all/themes/sc14/files/archive/tech\\_poster/tech\\_poster\\_pages/post189.html](http://sc14.supercomputing.org/sites/all/themes/sc14/files/archive/tech_poster/tech_poster_pages/post189.html).
19. Kaiser, Hartmut; Heller, Thomas; Adelstein-Lelbach, Bryce; Serio, Adrian; Fey, Dietmar; HPX: A Task Based Programming Model in a Global Address Space, *Proceedings of the 8th International Conference on Partitioned Global Address Space Programming Models (PGAS '14)*, 2014, ISBN 978-1-4503-3247-7.
20. Qawasmeh, Ahmad, Abid M. Malik, and Barbara M. Chapman. "OpenMP Task Scheduling Analysis via OpenMP Runtime API and Tool Visualization." In *Parallel & Distributed Processing Symposium Workshops (IPDPSW)*, 2014 IEEE International, pp. 1049-1058. IEEE, 2014.
21. Xu, Rengan, Maxime Hugues, Henri Calandra, Sunita Chandrasekaran, and Barbara Chapman. "Accelerating Kirchhoff migration on GPU using directives." In *Proceedings of the First Workshop on Accelerator Programming using Directives*, pp. 37-46. IEEE Press, 2014.
22. Brandt, Steven R., Hari Krishnan, Gokarna Sharma, and Costas Busch. "Concurrent, parallel garbage collection in linear time." In *Proceedings of the 2014 international symposium on Memory management*, pp. 47-58. ACM, 2014.
23. Anilkumar Nandamuri, Abid M. Malik, Ahmad Qawasmeh, and Barbara M. Chapman. 2014. Power and energy footprint of openMP programs using OpenMP runtime API. In *Proceedings of the 2nd International Workshop on Energy Efficient Supercomputing (E2SC '14)*. IEEE Press, Piscataway, NJ, USA, 79-88.

24. Rengan Xu, Xiaonan Tian, Yonghong Yan, Sunita Chandrasekaran, and Barbara Chapman, *2014 International Workshop on Programming Models and Applications for Multicores and Manycores (PMAM 2014)*, February, 2014, Orlando, Florida, USA.
25. Munara Tolubaeva, Yonghong Yan and Barbara Chapman. Predicting Cache Contention for Multithread Applications at Compile Time, *16th Workshop on Advances in Parallel and Distributed Computational Models to be held in conjunction with IPDPS 2014*, May 2014.
26. Ryan Grant, Stephen L. Olivier, James Laros, Ron Brightwell and Allan Porterfield. Metrics for Evaluating Energy Saving Techniques for Resilient HPC Systems. In *10<sup>th</sup> Workshop on High-Performance, Power-Aware Computing (HPPAC14 Workshop)*, Phoenix, AZ, USA, May 2014.
27. Sridutt Bhalachandra, Allan Porterfield and Jan Prins. Using Dynamic Duty Cycle Modulation to Improve Energy Efficiency in High Performance Computing. To appear *11th Workshop on High-Performance, Power-Aware Computing (HPPAC 2015)*, Hyderabad, India, May 2015.
28. Allan Porterfield, Rob Fowler, Sridutt Bhalachandra, Barry Rountree, Diporup Deb Rob Lewis, Brian Blanton. Application Runtime Variability and Power Optimization for Exascale Computers. Submitted to *International Workshop on Runtime and Operating Systems for Supercomputers (ROSS 2015)*, Portland OR, USA, June 2015.
29. K. Huck, A. Porterfield, N. Chaimov, H. Kaiser, A. Malony, T. Sterling, R. Fowler, "An Autonomic Performance Environment for Exascale, Supercomputing Frontiers," *Supercomputing Frontiers*, invited talk, 2015. Journal paper submitted to the *Supercomputing Frontiers and Innovations*.
30. Brandt, Steven R., Hari Krishnan, Gokarna Sharma, and Costas Busch, "Concurrent parallel garbage collection in linear time," in Proceedings of the *2014 International Symposium on Memory Management*, pp. 47-58, ACM, 2014.

## Software Products

### APEX

We have developed a software of the APEX prototype with initial versions of OpenX. It is engineered to work with both HPX-3 and HPX-5, and integrates RCR Toolkit and TAU. Both RCR Toolkit and TAU were updated to work with APEX and support certain capabilities that APEX required. APEX has been ported to several platforms, including Edison system at NERSC. APEX is included in the XPRESS software releases. Ongoing work with APEX will interface with XPI, enhance LXX introspection, and implement more sophisticated policies, especially for multi-objective control. APEX is currently being evaluated with XPRESS project applications.

### HPX-3

- Hartmut Kaiser; Bryce Adelstein-Lelbach; Thomas Heller; Agustín Bergé; et.al.: HPX V0.9.10: A general purpose C++ runtime system for parallel and distributed applications of any scale, Mar 2015, <http://dx.doi.org/10.5281/zenodo.16302>.
- Hartmut Kaiser; Alexander Duchene; Steven R. Brandt: HPXPI V0.1: An implementation of XPI for HPX, Mar 2014, <http://dx.doi.org/10.5281/zenodo.16399>.

## HPX-5

The HPX-5 runtime system directly reflects the semantic constructs of the underlying ParallelX execution model, being comprised of an active global address space, lightweight user thread management, parcels and message-driven computation, distributed processes, local control objects, and continuations. HPX-5 has focused on understanding low-level details of efficient and scalable exascale runtime implementation, including interfaces and interactions with operating systems and with hardware architecture. The initial implementation of HPX-5 relied on a two-sided, parcel-based model of communication. However, a fast, low-latency, one-sided communication mode is the current default. HPX-5 supports dynamic, distributed processes with determination detection and a flat, byte-addressable, global address space using native RDMA transport. Our implementation also supports a software based active global address space (AGAS) that allows explicit movement of blocks within the global address space.

HPX-5 0.5 was released February 2015. New features for version 0.5.0 include:

- Replaced network infrastructure
- Instrumentation interface:
  - The new instrumentation system in hpx is designed to make it easy to collect performance-related information. Among other capabilities, the instrumentation allows us to trace parcel lifecycle events (create, send, receive, run, end), put with completion send and receive events, and some scheduler events.
- Commutative associative reduction LCO:
  - This is used to perform a commutative-associative reduction across multiple compute complexes. The LCO takes as parameter monoid: common operation would be min, max, +, \*, etc and a predicate which guards the LCO.
- SSSP application KLA support:
  - The idea of KLA within SSSP is to proceed with the active vertices within current k-level and process them asynchronously. Once that step is done (which is called superstep in KLA paradigm), we can start with the next k level and so on until the algorithm is finished.
- `hpx_call` to work with `hpx_parcel_send_through()`:
  - The hpx call interfaces were extended to work with `hpx_parcel_send_through()`, which delays parcel delivery until an LCO has been sent.

Improvements from the latest version include:

- Typed `hpx_bcast()` with variadic arguments.
- `hpx_run()` to support typed actions
- Setting up CI on BigRed2
- Updated the option parsing infrastructure.
- The API to allocate an array of LCO local to the calling locality.
- Experimental support for `--hpx-dbg-waitonsegv`.

## Kitten Lightweight Kernel

The Kitten lightweight kernel<sup>1</sup> is the basis for the development of the Lightweight eXascale Kernel (LXK) in the XPRESS project. Kitten continues to be actively developed as part of the XPRESS and Hobbes projects. As a research prototype, no formal releases of Kitten are made.

## Open MPI

University of Houston's research team are long-time contributors to the Open MPI software project. For the XPRESS project, they have added support to use HPX as a runtime layer for Open MPI. It currently uses the HPX runtime environment for startup and communication through the TCP/IP transport module, although some details with respect to the integration of memory allocators and the internal utilization of threads in Open MPI still have to be worked out. Support for other network transports, specifically InfiniBand and shared memory, are currently being actively worked on. Open MPI is also able to use the native SLURM startup through HPX.

## OMPTX

Chapman's research team developed OMPTX, an OpenMP runtime based on HPX which shares the same ABI as the newly open-sourced Intel OpenMP runtime. This runtime has been adopted in existing compilers from Intel, GNU, PathScale, and Clang, and support within OpenUH is underway. The ability to use OMPTX as a drop-in replacement for the Intel OpenMP runtime facilitates experimentation in running OpenMP programs with HPX.

The Intel OpenMP Runtime interface and a number of the compilers which use it support OpenMP 4.0, allowing us to experiment with implementation of newer OpenMP features within the proposed software infrastructure.

## Profugus

The Profugus mini-application (mini-app)<sup>2</sup> is a set of computational kernels extracted from the Exnihilo particle transport code package at Oak Ridge National Laboratory (ORNL)<sup>3</sup>. Exnihilo provides advanced transport solvers and associated front-ends that are applicable to a wide-range of nuclear technology applications including reactor neutronics, shielding, criticality safety, dosimetry, fusion component analysis, and others. The resulting problem-specific complexity in the various front-ends, the dependence on a wide range of third-party libraries (TPL), and, most importantly, export control makes Exnihilo a difficult platform to share among computer scientists and hardware engineers. To overcome these limitations and to provide a means to collaborate across fields, we have developed the Profugus mini-app. The goal of Profugus is to provide open-source kernels that effectively capture the algorithmic features of the full Exnihilo applications. In this manner more effective connections can be established between applications developers and Xpress researchers.

The extracted kernels from Exnihilo are a simplified PN (SPN) solver and Monte Carlo method. The common features for both kernels are:

---

<sup>1</sup> Kitten, <https://github.com/ktpedre/kitten>, 2015

<sup>2</sup> Profugus, <http://ornl-cees.github.io/Profugus>, 2015

<sup>3</sup> Exnihilo Overview, <http://www.casl.gov/Exnihilo.shtml>, 2015

- can be executed in both fixed-source and eigenvalue modes
- supports same suite of Trilinos linear solvers as Exnihilo
- XML-driven input allows easy generation of simplified reactor and shielding problems that effectively captures the solver complexity
- generic material cross section libraries are provided that capture the salient physics of reactor simulations

The SPN implementation in Profugus uses a 2nd-order finite volume discretization on three-dimensional regular Cartesian grids. It uses both Arnoldi and generalized Davidson eigensolvers through Trilinos' Anasazi package<sup>4</sup> and provides a native power iteration implementation. Numerous linear solvers are accessible through the Trilinos Stratimikos interface. The spatial parallelism is geometric domain decomposition.

The Profugus Monte Carlo kernel provides a complete random walk algorithm that uses simplified multigroup physics to model particle interactions. It contains a simplified tally (history scoring) system that supports both k-eigenvalue calculations and path-length flux. The geometry is consistent with the SPN front-end and provides a simple model for describing reactor-like configurations. The kernel provides a subset of the full Exnihilo Monte Carlo parallel options. In Profugus the problem state is replicated across all domains and particle histories are decomposed.

The Exnihilo Monte Carlo algorithm utilized MIMD (multiple instruction, multiple data) concurrency. These methods have worked very efficiently on traditional CPU-based hardware with message-passing based parallel communication (MPI). Next generation hardware, in which on node concurrency will be provided by heterogeneous devices (e.g. GPUs), will require more SIMT (single instruction, multiple thread) type algorithms. We have implemented a simplified Monte Carlo linear solver algorithm in Profugus to use a SIMT algorithm to test the applicability of these approaches<sup>5</sup>.

#### Software Kernels (in progress)

- Particle-in-cell (PIC) proxy code suite and run information available on NERSC used for modeling of Fusion Plasma, Accelerators, and High-Powered lasers
- Proxy Versions of Warp (Warpcore) with LLNL, LBL
- Proxy Version of Pseudo-Spectral Solver with LBL
- OpenMP and HPX translated versions of UCLA PIC codes
- GTC versions (with IU, PPPL, NERSC)
- Suite of proxy codes for data applications and image processing:
- MBO: Merriman-Bence-Osher (MBO) scheme for graph-based methods (with UCLA)
- Nystrom Method (with UCLA)

---

<sup>4</sup> Trilinos, <http://trilinos.org>, 2015

<sup>5</sup> We have tested device-based parallelism using the Trilinos Kokkos framework in Profugus

## Databases

### TAUdb

We use the TAUdb performance database to store performance results from experiments. However, the databases are primarily for internal use only.

## Technologies or Techniques

### APEX

The University of Oregon is developing an Autonomic Performance Environment for eXascale (APEX) that is integrated in the XPRESS OpenX software stack, primarily at the HPX runtime layer. APEX is designed to support performance introspection and adaptation through policy-based mechanisms. The introspection layer utilizes both event-based and asynchronous observation. It interfaces with the TAU Performance System for event-level measurement and the RCR Toolkit to capture lower-level information about the system resources, energy, and health. APEX conducts in situ analytics to process the information in real time. APEX policies are rules that decide on outcomes based on introspected state. Policies are registered at startup and can be triggered from introspection analysis or run periodically to look at APEX state. Callback functions are invoked in consequence to the policy rules. The main objective of APEX is to enable runtime adaptation across the OpenX stack by connecting online performance observation and decision analytics with policy-driven actuators.

## Presentations/Tutorials

1. Alice Koniges, and Mike Heroux. Using Application Proxies for Co-design of Future HPC Computer Systems and Applications,” Co-design Workshop, Dagstuhl, Germany, May 22, 2012.
2. Michael A. Heroux, Alice E. Koniges, David F. Richards, Richard F. Barrett, Thomas Brunner, Using Application Proxies for Co-design of Future HPC Computer Systems and Applications, SC12 Full Day Tutorial, November 2012.
3. A. Koniges, W. Lui, Y. He, D. Eder, A. Fisher, N. Masters, and R. W. Anderson, Multi-Material Plasma Modeling on HPC Platforms, Invited: 23rd International Conference on Numerical Simulation of Plasmas Beijing, China, September 13, 2013.
4. Tim Mattson, Alice Koniges, Simon McIntosh-Smith, [OpenCL: A Hands-On Introduction](#), SC13, SC14 Tutorial, November 18, 2013.
5. Alice Koniges. APPS on HPX, XSTACK PI Meeting, Boston MA May 27-28 2014.
6. S. Johnson (presenter), G. Davidson, T. Evans, S. Hamilton, T. Pandya, W. Joubert, and J. Larkin. A first strike at an OpenACC C++ Monte Carlo code. Porting Apps to Titan, GPU Technology Conference, San Jose, CA, March 17--20, 2015.

## Posters

1. A. Koniges, R. Gerber, D. Skinner, Y. Yao, Y. He, D. Grote, J-L Vay, H. Kaiser, and T. Sterling, [“Plasma Physics Simulations on Next Generation Platforms”](#), *55th Annual Meeting of the APS Division of Plasma Physics*, Volume 58, Number 16, November 11, 2013.
2. J.-L. Vay, L. A. Drummond, A.E. Koniges, B. Godfrey, I.Haber, “Scalable Arbitrary-Order Pseudo-Spectral Electromagnetic Solver,” Poster and two-page paper in Proceedings of

the International Conference for High Performance Computing, Networking, Storage and Analysis (SC14),2014.

[http://sc14.supercomputing.org/sites/all/themes/sc14/files/archive/tech\\_poster/tech\\_poster\\_pages/post189.html](http://sc14.supercomputing.org/sites/all/themes/sc14/files/archive/tech_poster/tech_poster_pages/post189.html)