

## Cornell Virtual Workshop Capabilities Brief

The Cornell University Center for Advanced Computing (CAC) is a leader in the development and deployment of Web-based training programs designed to: (1) enhance the computational science skills of researchers, (2) broaden the participation of underrepresented groups in the sciences and engineering, and (3) accelerate the adoption of new and emerging technologies. The *Cornell Virtual Workshop<sup>SM</sup>* learning platform was launched in 1995. Since then, over 87,000 unique visitors have accessed Cornell Virtual Workshop training modules on high-performance computing (HPC) topics ranging from parallel computing to visualization.

Besides HPC topics, CAC develops Virtual Workshops for the humanities, social sciences, sciences or engineering on any topics of interest. The client provides access to a Subject Matter Expert (SME) and CAC does the rest – from instructional design to the deployment of the Virtual Workshop on CAC servers. Usage data may be auto-generated for inclusion in funding agency reports.

A Virtual Workshop is made up of multiple, logically-related modules, comprised of full text discussion, short audio or video clips, graphical simulations, examples, exercises, and quizzes. Virtual Workshop modules are always available on the Web as a 24x7 option for users who want to study a topic on demand and at their own pace.

The Cornell Virtual Workshop learning platform is a proven training technology. Cornell has received numerous grants from the National Science Foundation (NSF), the Department of Defense (DOD), and private industry to develop Virtual Workshops on a wide variety of topics. For example, under an NSF grant, CAC developed the *Ranger Virtual Workshop* to train educators and students on how to effectively use the Ranger supercomputer [1]. CAC was selected by Dynamic Research Corporation to develop the *User Productivity Enhancement, Technology Transfer, and Training (PETTT) Virtual Workshop* as an online training resource for the users of DOD Supercomputer Resource Centers (DSRC).

Today, CAC is developing and deploying online training for XSEDE, the NSF's advanced cyberinfrastructure and services program. Twenty-seven Cornell Virtual Workshop modules are currently available through the XSEDE User Portal [2]. Most recently, CAC staff worked with a team of educators and technologists to convert Jim Demmel's *Applications of Parallel Computers* course into the Cornell Virtual Workshop format in order to broaden access to the popular Computer Science course taught at Berkeley [3]. Due to the success of this project, the conversion of other Computer Science courses are planned for the future in order to further enhance XSEDE educational offerings and broaden participation.

CAC is also developing the *Stampede Virtual Workshop* [4] and recently released two new technology modules: *Many Integrated Core (MIC)* (see Figure 1) and *Vectorization in Modern CPUs and the New Intel Xeon Phi*.

Cornell Virtual Workshops are available at all times to the entire scientific community – researchers, HPC practitioners, students, and educators – and they require no travel budget. By leveraging the content and learning methodologies developed over a decade of Virtual Workshop design, Cornell can bring new instructional content online quickly to accelerate new technology adoption.

[1] Mehringer, S., Woody, N., Dolgert, A., Lantz, S. & Stanzione, D. (2011). Maximizing Computational Learning for Faculty and Student Scientists: The Ranger Virtual Workshop. *TeraGrid Conference Proceedings*. Retrieved from: <http://www.cac.cornell.edu/about/pubs/RangerVirtualWorkshop.pdf>.

[2] XSEDE User Portal: On Demand Training (n.d.). Retrieved from <https://portal.xsede.org/web/xup/online-training>.

[3] Cornell Virtual Workshop: Applications of Parallel Computers (2013). Retrieved from <http://www.cac.cornell.edu/VW/apc/>.

[4] Stampede Virtual Workshop: TACC User Portal (n.d.). Retrieved from <https://portal.tacc.utexas.edu/stampede-virtual-workshop>.

Welcome shm7

<b>Home</b>	<b>Topics</b>	<b>Reference</b>	<b>Glossary</b>	<b>Help</b>	<b>Notebook</b>
<h2>Cornell Virtual Workshop</h2>					
<b>MIC</b>	<b>MIC Architecture</b>				
<a href="#">Introduction</a>	In this section, we provide an overview of the MIC architecture and describe several distinguishing features that are likely to be relevant to its performance when given different kinds of workloads.				
<a href="#">Goals</a>					
<a href="#">Prerequisites</a>					
<b>MIC Architecture</b>	As mentioned in the introduction, MIC (pronounced 'mike') is an acronym for "Many Integrated Core". This is the architecture used by the Xeon Phi coprocessors in Stampede. In essence, each MIC coprocessor is a fully functional Linux-based system running on a PCIe card featuring a specialized CPU that achieves very high floating point throughput and low power consumption through lots of simple cores derived from the x86 architecture. These cores support a modified x86 instruction set, contain wide vector units, and feature hardware threading (SMT) with four threads per core.				
<a href="#">* Processor</a>					
<a href="#">* Cache and Memory</a>					
<a href="#">* I/O and Networking</a>					
<a href="#">* File Systems</a>					
<b>Programming Paradigms</b>	Surrounding the processor is a limited amount of very fast GDDR5 memory. Each coprocessor in Stampede has 8GB of system memory. Communication with the host machine occurs via the PCIe bus. Viewed from afar, the Xeon Phi coprocessor behaves like an entirely separate compute node that happens to be physically contained within a host node:				
<a href="#">* Distributed Memory</a>					
<a href="#">* Shared Memory</a>					
<a href="#">* Hybrid</a>					
<a href="#">* Offload</a>					
<b>Running Code on the MIC</b>					
<a href="#">* Native Execution (M)</a>					
<a href="#">- Exercise: Interactive</a>					
<a href="#">- Exercise: OpenMP</a>					
<a href="#">* Symmetric Execution</a>					
<a href="#">- Exercise: MPI</a>					
<a href="#">- Exercise: Hybrid</a>					
<a href="#">* Offload</a>					
<b>Performance Considerations</b>					
<a href="#">* Vectorization</a>					
<a href="#">* Threading &amp; Affinity</a>					
<a href="#">* Memory Access</a>					
<a href="#">* Tuning</a>					
<a href="#">* Balancing</a>					
<b>Quiz</b>					

**Host with dual Intel Xeon "Sandy Bridge" (CPU)**

**PCIe card with Intel Xeon Phi™ (MIC)**

Access from network: ssh <host> (OS)  
ssh <coprocessor> (<μOS>)

Virtual IP service for MIC

Systems overview of a host and MIC coprocessor

In the next few sections we will describe more about the CPU architecture to understand how it is optimized for producing high floating point throughput, as well as its memory and I/O subsystems.

*Figure 1 – Stampede Virtual Workshop screenshot. The Many Integrated Cores (MIC) module features interactive exercises and quizzes focused on programming paradigms and running code on the MIC.*