OS/R PI Meeting Demonstrations

Date: May 23, 2016
Location: Gleacher Center
450 N Cityfront Plaza Dr. room 204
Chicago, Illinois

Argo:

We will showcase how different software components that form the Argo system (namely the Global OS, Node OS, Global Information Backplane and Argo Runtime) can be used in an integrated fashion. This particular demo showcases the working of the Argo system and various aspects of these software components by presenting a holistic, integrated approach for system-wide power management in the Argo framework. In the Argo project, we believe that in order to improve application performance, throughput and overall power utilization, power management strategies need to exist at all levels in the system stack, ranging from the node level (hardware), to the job or application level, and further to the system-level.

The demo will showcase users submitting jobs which are launched in their enclaves by Global OS. The monitoring software running on the system monitors sensors and publishes this information through BEACON, which various software components running on the system receive. The Global OS, based on some state of the system, decides to reduce power in an enclave and publishes a request via BEACON. This request is received by Node OS/R components such as the Node Resource Managers, which works with the Argobots software and higher-level applications to optimize the hardware resources on the system for reducing power.

Title: Argobots: A Lightweight Low-level Threading/Tasking Framework
Argobots is a lightweight, low-level threading and tasking framework. It provides high-level runtimes and domain-specific libraries with threading and tasking mechanisms so that they can build their solutions efficiently. In this demo, we will first show how applications or programming model runtimes such as Charm++, Cilk, MPI and PaRSEC can take advantage of Argobots and be written using the Argobots API. In order to demonstrate the ability of dynamic hardware resource management, we will also show how applications or high-level runtimes can adjust the number of cores (or hardware threads) that are represented as execution streams (ESs) in Argobots. We will show shrinking and expanding the number of ESs at run time using the Argobots interface and explain how it works as well as how the work units such as user-level threads (ULTs) and tasklets are managed and migrated when the events of shrinking and expanding occur.

Title: Demonstrating Node OS in the ARGO system
Argo NodeOS uses the concept of compute containers to provide an integrated control over individual hardware and software node resources. Building upon the Linux kernel, we provide partitioning of CPU and memory resources, add support for memory-mapped access to node-local, PCIe-attached NVRAM devices, and introduce a new scheduling class targeted at parallel runtimes supporting user-level load balancing. We will demonstrate integrated resource management and job launch capabilities on a complex application workflow consisting of a simulation and a multistage in situ visualization component.

Title: Demonstrating the BEACON and EXPOSE communication frameworks in ARGO

BEACON is a lightweight signaling framework that provides fast-path publish/subscribe interfaces to enclave and system-global entities with the required features. The BEACON demo will demonstrate how it functions as a distributed messaging bus for transferring event/control/response signals between publisher and subscriber entities. We will demonstrate the BEACON interface with the TAU Performance System where performance data from the memory, I/O, and communication subsystem will be passed to Beacon and available for other subscriber tasks.

EXPOSE is a lightweight layer for online analytics. It interfaces with BEACON to subscribe to certain monitored events and to publish analytics results, possibly for feedback control purpose. The EXPOSE demo will show various performance analytics being done with the data from the BEACON demo described above.

Title: Global OS/R : Providing a global view to the ARGO system

The Global OS forms the crux of the Argo system. The goal of Global OS is to provide a hierarchical view of an Argo system based on the enclave concept and a set of services distributed across this hierarchy implementing resource control and monitoring.

The demo will demonstrate the different features of Global OS, its sub-components and the interfaces it provides for machine configuration, deployment, monitoring, management and application launching across the system.

Hobbes:

Title: Parallel Programming Debugging in Runtime Systems Environments

Debugging is among the most challenging aspects of highly scalable parallel computing and is made more so by the inclusion of runtime systems for dynamic adaptive execution. In support of the Hobbes Project, a debugger is being developed based on the well-known gdb tool set but with important extensions for concurrent thread analysis and access of objects within the global address space. This demonstration will show a working instance of this debugger being used interactively with example buggy codes being performed within the context of HPX-5 runtime system that will itself be ported to the Hobbes OS.

X-ARCC:

Title: Components of the X-ARCC project

In this demonstration, we present three components of the X-ARCC project. First, we show how our power attribution model (XEMpower + MARC) can be used to accurately predict power usage for running applications in real time. Second, we demonstrate the impact of noise for various OS configurations as competing applications are started and stopped. Third, we show how our multinode resource scheduler, Taurus, works to improve the performance of modern managed languages in distributed environments.