BlackWidow Software

Don Mason
Cray Inc

ABSTRACT: *The BlackWidow Programming Environment software is a natural follow on to what was developed for the Cray X1/X1E. The OS and IO software is quite different. Cray has moved away from SGI's Irix ™ operating system to SUSE Linux Enterprise Server (SLES). The OS base is common with future scalar platforms. Other common components include the Lustre file system and application scheduling.*

KEYWORDS: BlackWidow, Linux

1. Introduction

The Cray BlackWidow vector processor to be delivered next year is a follow on to the Cray X1E system. The software for the Cray BlackWidow system addresses the following objectives:

- Deliver the highest possible performance from the Cray BlackWidow hardware to customer applications
- Provide a natural follow on for Cray X1 and Cray X1E customers
- Significantly move towards software commonality with other Cray platforms, especially the Cray XT
- Improve system reliability over previous platforms

Delivering hardware performance to customer applications is a long-standing Cray tenet. For the OS, this generally means staying out of the way of application execution. For the compilers it means taking full advantage of hardware capabilities in generated code.

Cray X1/X1E users will find the transition to the Cray BlackWidow system natural. The transition does require re-compilation, but few other changes. The same toolset available on the Cray X1 will be available on Cray BlackWidow.

Over the past few years Cray has been working towards platform software commonality; examples are in system administration, system support, programming environment tools and scheduling. When Cray BlackWidow software is released in 2007 Cray will have a high degree of commonality across platforms.

System reliability is requiring more interplay between hardware and software features. Over a period of time some hardware components will fail. By obtaining good information on the hardware failures the system software can contain the error and minimize impact.

2. Hardware Features Used by Software

Several new hardware features are significant to the system software.

Like the Cray X1, the Cray BlackWidow system uses a remote translation table to link the address spaces of several nodes into one. This means referencing off node memory is done with simple loads and stores, not message passing.

The Cray BlackWidow system includes an additional feature called the node translation table that removes the restriction that compute nodes must be scheduled contiguously. The benefit of this is that the operating system can place applications anywhere, eliminating the need to migrate applications to obtain the necessary contiguous space.

The Cray BlackWidow hardware is packaged with two four-processor nodes per board and up to 16 boards in a cabinet. Processors on a node share a common local memory. OpenMP is supported across the nodes processors as on the Cray X1/X1E. However, there is no multi-streaming processor (MSP).

New hardware features such as vector atomic operations enable the compiler to generate higher performance code.

An instantiation of the Linux operating system runs on each node, treating the node like an SMP.

Several new hardware features will improve overall system availability. There is memory protection that can be enabled to prevent the OS from writing into another OS's (nodes) memory space. With the new distributed OS architecture this feature is very important.

In general more fault information is available to system software. For example there are retry options for memory errors.

Another new feature, called a graduation timeout, protects against network errors by letting the operating system know if a network memory reference failed for some unknown reason; for example, something other than a standard packet handling error. The graduation timeout will allow confining the error to a single application and prevent taking down a system.

The I/O system for Cray BlackWidow is quite different. The Cray XT I/O subsystem is used to support Cray BlackWidow. Cray BlackWidow compute nodes are connected to the Cray XT using a hardware bridge blade that interconnects the Cray BlackWidow network (YARC) to the Cray XT's SeaStar network. A Cray BlackWidow node looks to the Cray XT like another Cray XT node and correspondingly a Cray XT node looks to Cray BlackWidow like another Cray BlackWidow node.

The transport between the Cray BlackWidow system and the Cray XT is portals. This allows higher-level I/O packages like Lustre to easily interoperate between the Cray XT I/O subsystem and the Cray BlackWidow. Portals is not used for on machine communication.

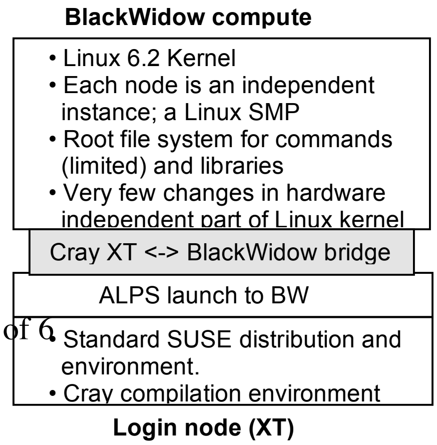
The benefits to the I/O architecture are many. It enables common peripheral support across different platforms including the Cray Eldorado system. It provides for future opportunities such as sharing Lustre files. And it is a basic building block for future architectures such as Cascade.

3. BlackWidow OS Architecture

The Cray X1/X1E operating system, UNICOS/mp is a monolithic OS that Cray has delivered to support systems of 4096 processors (SSPs). The Cray BlackWidow system is very different – it uses a distributed OS based on Linux.

There are several reasons that we made the change to a distributed Linux OS. The first was scalability. Scaling a monolithic OS becomes more and more difficult as "hot spots" arise and are resolved. A distributed OS allows scaling by replication.

A second reason is to align Cray's future OS direction. The BlackWidow OS, and a future scalar platform OS based on Linux compute nodes, can be developed from a common base. The model we are developing is shown in Figure 1.


BlackWidow compute
• Linux 6.2 Kernel
• Each node is an independent instance; a Linux SMP
• Root file system for commands (limited) and libraries
• Very few changes in hardware independent part of Linux kernel
Cray XT <-> BlackWidow bridge
ALPS launch to BW
• Standard SUSE distribution and environment.
• Cray compilation environment
Login node (XT)

The top box in Figure 1 represents one or more compute nodes that are connected to a set of Common IO and Login nodes shown in the bottom box.

Cray's future direction is to have a common IO and login subsystem that is utilized by different compute node types. This is true today for Cray XT Catamount compute nodes and will be true for both Cray BlackWidow and Cray Eldorado systems. The Application Level Placement Scheduler (ALPS) developed for BW will be used in the future to launch jobs for Linux based compute nodes on both scalar and specialized processor systems.

On the Cray BlackWidow compute node a Linux kernel runs, but with few of the services (and daemons) found on a typical Linux installation. Supported services include the Lustre client and an ALPS client used with program launch.

To facilitate scalability, only a limited number of commands are supported for compute nodes. These are generally for system administration and users do not log into compute nodes. Services that don't scale well such as demand paging are not supported. The limited commands and libraries are installed as part of a root file system during the boot process. Sites may add to the contents of the root file system depending on their requirements (e.g. for security support).

A goal for Cray is to minimize changes to the hardware independent parts of Linux. To date we have very few changes in the hardware independent parts of Linux and have make most modifications through drivers unique to the Cray BlackWidow hardware. Limiting hardware independent changes in the kernel is important to support of the product and tracking new Linux releases.

The service nodes (I/O and login) provide a standard SUSE user environment. For the Cray BlackWidow system, Cray compilers run on login nodes and the TotalView debugger is initiated from a login node (on Cray BlackWidow the main parts of the debugger run on service nodes with clients on compute nodes). And of course Linux OSTs and MDSs run on service nodes.

Users launch application on compute nodes using the ALPS aprun command, either interactively from login nodes or more commonly via script submitted with a batch job.

System Administration on the Cray BlackWidow system, called Mazama, provides an operator/administrator services for booting and dumping the system, examining logged information (various log sources are consolidated with a log manager), managing OS source images, and many other activities.

4. BlackWidow Software Features

Software features for the Cray BlackWidow system are nearly identical to what has been provided for the Cray X1/X1E.

For the programming environment, features include:

- C/C++ and Fortran compilers.
- gcc (new on the Cray BlackWidow system)
- UPC support
- Co-Array Fortran support
- OpenMP support on a Cray BlackWidow node
- Cray PAT and Cray Apprentice^2 performance analysis tools
- MPI based on MPICH2 (different MPI base than Cray X1's)
- shmem
- TotalView debugger
- gdb for single PE application
- Scientific libraries

Except as noted, all compilers and tools evolve from the Cray X1/X1E software. Cray BlackWidow compiler support will be in the next (5.6) release of the compilers. Many optimizations added for Cray BlackWidow compilers are applicable to the Cray X1/X1E systems.

gcc is used by Cray to compile commands and libraries for the Cray BlackWidow system. gcc will probably be most useful to customers porting open source C programs to the Cray BlackWidow. gcc does little vectorization.

The OS and IO software does not evolve from Cray X1/X1E technology, but most capabilities are retained. Features include:

- Linux kernel on all nodes
- Application Level Placement Scheduler (ALPS)
- ALPS interfaces to batch subsystems including PBSPro
- Mazama system administration
- Lustre parallel file system
- Process, project and job accounting
- I/O capabilities of the Cray XT system:
    - GbE and 10GbE networking support
    - RAID disk with fail-over support
    - TCP support including socket support on compute nodes
    - NFS support

The ALPS system, described in the next section is different from psched on the Cray X1/X1E systems although it performs some similar functionality.

Accounting is enhanced over the Cray X1/X1E using the Linux Comprehensive System Accounting (CSA) capabilities.

Currently there is no plan to support checkpoint/restart.

5. Application Level Placement Scheduler (ALPS)

ALPS is an innovative software suite designed to improve scalability of large systems. It does so by encapsulating several application services in a way that provides benefits similar to virtualization technologies being explored in the research community. Each component of an application is managed by an independent daemon process that provides services including signal distribution and output consolidation. An additional facility exists to allow tools such as debuggers and performance analyzers to gain access to the processes of an application.

Unlike the Cray X1 psched scheduler ALPS does not do batch scheduling. Third party packages such as PBSPro or LSF perform the scheduling function. ALPS exports an interface that allows schedulers and ALPS to exchange information on things like node availability.

6. Cray Platform Software Commonality

Cray's direction is to provide common software across platforms in the following areas:

- File systems
- Base OS
- System Administration
- Placement scheduling (ALPS)
- Programming Environment tools
- Communication libraries
- Hardware supervisory systems (HSS)

By the first release of Cray BlackWidow software Cray XT systems and Cray BlackWidow systems will be able to share access to Lustre files.

Cray BlackWidow compute and Cray XT service nodes will use a common Linux distribution. Cray is planning to offer compute node Linux on Cray XT systems and follow-ons from the common distribution.

The Mazama system administration package is to be the common administration tool for the future. At the time of Cray BlackWidow release partial Cray XT administrative functionality will be in the Mazama framework and over time more will be added. Our strategy is to not disrupt current Cray XT administrative functionality in a revolutionary way. Functionality that will be common includes alarm presentation and related information.

ALPS will be a common placement tool for Cray BlackWidow systems and Cray XT systems running compute node Linux. A user will be able to execute a script that launches applications to either or both platforms.

Programming Environment tools will also be common at the time of Cray BlackWidow release. These include the TotalView debugger, Cray PAT and Cray Apprentice[2].

All platforms will have the same MPI functionality based on MPICH2. Over time shmem functionality will be made common as well.

Finally the Hardware Supervisory System (HSS) will be common between the Cray BlackWidow system and future Cray XT based scalar platforms. HSS is the backbone support network that runs independent of the system fabric and controls machine hardware configuration and operation.

7. Implementation Tools and Status

Our major objective for the BlackWidow software implementation is to get all of the functionality checked out before the new hardware is available.

We developed two powerful simulation tools that enable us to achieve this objective. The first is called the Application Simulator. It is a performance accurate simulator utilized by compiler developers to checkout functionality and to evaluate performance improvements.

The Application Simulator simulates both the Cray X1/X1E system and the Cray BlackWidow. The Cray X1/X1E support helps us verify the accuracy of parts of the simulator for BlackWidow.

The second simulator, termed the Functional Simulator, is utilized to check out system software; e.g. the OS itself, ALPS, and Lustre. The programming environment test suites are also run on the Functional Simulator.

Figure 2 shows the Functional Simulator.

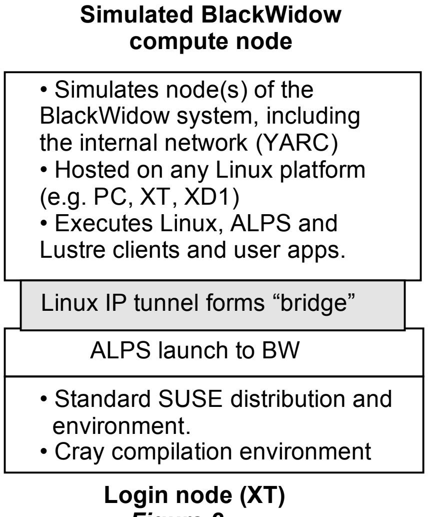

Simulated BlackWidow compute node
• Simulates node(s) of the BlackWidow system, including the internal network (YARC)
• Hosted on any Linux platform (e.g. PC, XT, XD1)
• Executes Linux, ALPS and Lustre clients and user apps.
Linux IP tunnel forms “bridge”
ALPS launch to BW
• Standard SUSE distribution and environment.
• Cray compilation environment
Login node (XT)

Figure 2

Note the similarity to Figure 1 that describes the actual Cray BlackWidow configuration. The connection between the simulated Cray BlackWidow system and the real XT service nodes is accomplished using a Linux IP tunnelling capability. ALPS uses IP to communicate between its host services and the ALPS client on the simulated node (as it does on the actual hardware). The Lustre client (actually the portals transport) requires an infrastructure to interact over the IP link.

Jobs are submitted to the Cray BlackWidow nodes via ALPS just as on the real system. Cray has run POP, GUPS and the HPCC suite on several simulated nodes. The simulator is fast enough that we’ve been able to run the simulator under the simulator!

At this stage of the project we are beginning to bring up hardware components. Almost all of the software functionality is complete.

Linux 2.6 has been running in simulation for over two years. ALPS has been running for the last year. We were able to launch and execute the aforementioned applications back in the fall of 2005. Hundreds of tests of all of the compilers have been run with high pass rates. We think we are ready for the hardware.

About the Author

Don Mason is the software project director for the BlackWidow project. He can be reached at dmm@cray.com.