Cray/SGI Joint Operating Systems Direction

Gabriel Broner
Cray Research / Silicon Graphics
655F Lone Oak Dr, Eagan, MN 55121, USA
E-mail: broner@cray.com

5/1/97

Abstract

After the merger Cray Research and Silicon Graphics decided to jointly develop a single operating system that will address the needs of the wide Cray/SGI product line. Cellular Irix is an evolution of Irix that incorporates features available in UNICOS and UNICOS/mk. Binary compatibility with Irix provides access to a wide range of applications, while source compatibility with UNICOS and UNICOS/mk provides ease of migration for existing Cray customers. The Cellular Irix system will support SN1.

Contents

---

1. Introduction: A Single Operating System

After the merger between Cray Research and Silicon Graphics, the two organizations analyzed their needs in operating systems. The groups jointly decided to proceed with a common path. A single operating system for the Cray and SGI product lines leverages the expertise existing on both sides to develop a better product. The joint system, Cellular Irix, offers features from Irix, UNICOS and UNICOS/mk. The system will be binary compatible with Irix, provide access to a large set of applications, and be source compatible with the Cray operating systems to provide an easy migration for existing Cray customers.

The decision to have a single operating system also has an impact in other areas; we will be able to reuse libraries, commands, compilers, etc. across the complete product line. At the same time, we expect a single operating system will simplify things for customers, who will only need to deal with and administer one system, and for ISVs, who would be able to target a wider range of machines with a single application port.

The rest of this paper covers the following. Section 2 talks about requirements for the operating system. Section 3 presents Cellular Irix. Section 4 describes the Cellular Irix roadmap and the intermediate deliverables. Section 5 covers the high-end features being incorporated into Cellular Irix. Section 6 describes the migration to SN1 for existing T3E customers.

2. Background: Operating System Requirements

An operating system that services the Cray/SGI product line has to address requirements coming from the various market segments it targets.

2.1 Requirements from the High End

An operating system for the high end, needs to be able to support large machines (typically from 64 to 4,000 CPUs) with good performance. The operating system has to properly scale for technical computing workloads, which typically are large CPU-intensive applications which also perform a large number of big-block I/O requests.

From a usability and ease of administration perspective, a large system (e.g. 1,000 CPUs) needs to present itself as a single coherent system (Single System Image, or SSI) and not as 1,000 dissimilar machines.

On a large system, hardware failures are expected to be relatively frequent. It is important for the operating system to be able to tolerate failures, such that any failure does not bring the system down.

There is also a series of features that have unique requirements for the high end. These include Checkpoint/Restart, Limits, Accounting, Scheduling, and Security among others.

2.2 Requirements from the Commercial World

The commercial world requires an operating system that will support "medium size" servers (from 4 to 64 CPUs.) Commercial applications are typically smaller than those in the supercomputing world, but

the demands they impose on the system may be higher, due to a more varied used of system services. Appropriately scaling these general purpose workloads is a challenging requirement for a system in the commercial world.

From a fault containment point of view, the requirements are also different. In the commercial world, a system being down may represent money being lost. For example, if a system servicing credit card transactions happens to be unavailable, customers may just switch to use a different credit card. In this example, money lost due to a server being down may never be recovered, and as a result, servers being up and available at all times is clearly a goal.

3. Cellular Irix

Cellular Irix is the operating system that addresses the needs of the Cray/SGI product lines. It is a scalable, single system image, distributed operating system. It also provides fault containment. Cellular Irix is an evolution of Irix, combined with features and expertise from UNICOS and UNICOS/mk.

3.1 The Architecture of Cellular Irix

A *cell* is an Operating System abstraction that represents a "piece" of the machine (a series of CPUs, their associated memories, etc.) Within a cell, the operating system looks pretty much like an SMP operating system (like Irix, or UNICOS). Across cells, the operating system behaves more like a distributed operating (like UNICOS/mk) where cells cooperate to provide the view of single system. (Figure 1 shows a multi-cell system.)

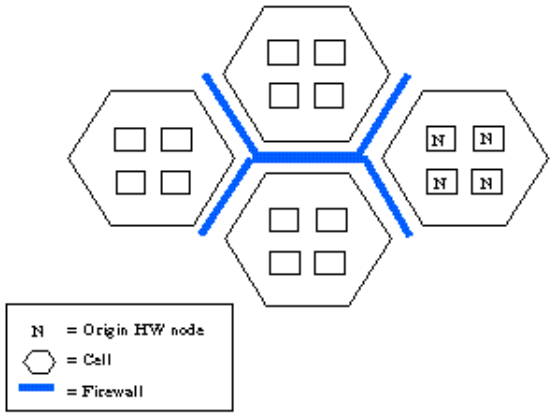

Figure 1 - A multi-cell Cellular Irix system

Cells can be of different *sizes*, and a particular machine can be configured in more than one way. For example a 128-CPU machine could be configured as 4 cells of 32 CPUs, or 8 cells of 16 CPUs, etc. Cell sizes and the number of cells have implications on *fault containment* and *scalability*. From a *fault containment* point of view, a failure in a cell brings down the complete cell. If cells are large, more of the machine will be lost in a failure. From a *scalability* point of view, within a cell, the system will scale like an SMP, and across cells, like a distributed OS. If cells are *large*, lock contention and memory latency within a cell will be high, while there will be less inter-cell traffic due to distributed protocols. On the other hand, if cells are *small*, SMP lock contention and memory latency will be small, but the

increased number of cells will imply higher inter-cell traffic. In general we expect a "medium ground" solution will satisfy the needs of each individual site in terms of fault containment and scalability. Just to give an idea of our current thinking, we imagine a 1,024-CPU machine would be 32 cells of size 32, and a 4,096 machine would be 64 cells of 64 CPUs.

From a *performance* point of view Cellular Irix behaves similarly to Irix when a system call is serviced locally within a cell. To achieve this level of performance, a series of techniques is being used to keep most operations local to a cell. Examples of these techniques are *aggressive caching*, and a *peer-to-peer* style of distribution.

4. Cellular Irix Deliverables

Cellular Irix is described above as a system that will offer full single system image, peer-to-peer distribution, and fault containment, to address the needs of the complete Cray/SGI product line. That is our target, and will continue to be our goal for the next few years. Nevertheless, since the final realization of Cellular Irix will take a few years, there will be a series of intermediate deliverables that will address the immediate needs of the different market segments independently.

Irix 6.5 is basically an SMP operating system. Compared to previous releases of Irix, this release offers increased SMP scalability to a larger number of CPUs. A less visible piece of work that went into this release is the *encapsulation* of subsystems, which will permit their distribution at the next release.

The first system with multiple cells, will be offered in two variants. The *Enterprise* variant addresses the more critical needs of the commercial market, while the *Supercomputer* variant addresses the needs of the technical computing market. These variants are in effect two possible configurations of the same release. The reason for offering two variants is that they will be available much sooner than the single version that addresses the needs of both markets, while the development effort is highly leveraged.

Figure 2 shows the Cellular Irix Roadmap. Irix 6.5 is planned for 4Q97, and it will be available as either a *single system*, to address the needs of the technical compute market, or as a *partitioned system* which will address the needs of the commercial market. The *Enterprise* and *Supercomputer* variants of Cellular Irix could be viewed as evolutions of these systems with additional capabilities. They are planned for 12/98.

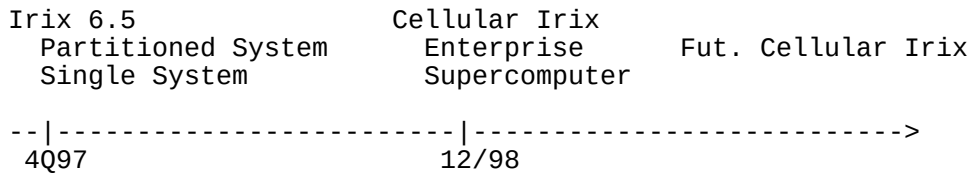

Figure 2 - The Cellular Irix Roadmap

4.1 Cellular Irix *Enterprise*

The Cellular Irix *Enterprise* variant addresses the needs of servers for the commercial market. The emphasis of this system is on *fault containment* over single system image. A multi-cell enterprise system will look to its users like a multi-host system. Each cell will have a unique host name, network

address, root file system, I/O devices, console, etc. From a user or application point of view a cell will look pretty much like an independent machine. The only element that will be distributed across cells will be the *file system* (and volume/disk manager), such that a file system could be equally accessible from all cells. Outside the distributed file system, the managing of multiple cells is done with an enhanced version of Irix Array software.

Compared to the final Cellular Irix picture, the Enterprise variant postpones having single system image, in order to have full fault containment for all cells.

4.2 Cellular Irix *Supercomputer*

The Cellular Irix *Supercomputer* variant addresses the needs of the high end technical market. It offers full *single system image*, which means that for users, applications, and administrators, a large machine running the Supercomputer variant will look like a single host. To provide single system image, *all subsystems are distributed*. However, not all subsystems are distributed in the same manner:

- The *file system* and *volume/disk manager* are fully distributed in a peer-to-peer fashion, such that file I/O issued on a given cell be processed on that cell.
- Other components are distributed in a more "basic" fashion. For example networks, ttys, pipes, System V message queues and semaphores, reside on a special cell, that we call the *golden* cell. Most requests to these subsystems are forwarded from the originating cell to the golden cell.

From a *fault containment* point of view, a failure on the golden cell is fatal and it brings the system down. A failure in any other cell can be contained to that cell.

The Supercomputer variant addresses the needs of large supercomputers, where the primary objective is to run large compute-intensive applications that perform I/O in large blocks to disk. Applications will "see" a single system, and will be able to equally access all system functions. The system will scale especially well for parallel I/O to disk.

Different from the Enterprise solution, the Supercomputer variant emphasizes *single system image*, while Enterprise emphasizes *fault containment for all cells*. The main reason is that the Supercomputer system is intended to run the large supercomputing applications that span cells, and will benefit from single system image. From an *administration* point of view, the Supercomputer solution will allow administering a large machine like a single machine, and not like a large collection of hosts. From a *fault containment* point of view, having a golden cell is a reasonable trade off: the golden cell will run mostly existing reliable non-distributed pieces of the operating system. Hardware failures will occur in the golden cell only in a small proportion (1/N, where N is the number of cells), so the system will have a good overall mean time between failures. The multithreaded nature of the golden cell will also allow for reasonable performance of non-distributed services.

Compared to the final Cellular Irix picture, the Supercomputer variant postpones full peer-to-peer distribution for all subsystems, and recovery for failures on all cells, to provide full single system image and full distribution of I/O to disk. This has been decided taking into account the use of a machine in the Supercomputer world.

The 12/98 Supercomputer release will support SN1.

5. High-end Features

Cellular Irix will provide a number of "high-end" features, currently available in UNICOS and UNICOS/mk. The following is the list of features that we plan to have for the 12/98 release:

- UNICOS and UNICOS/mk API and Commands
- MPP Application Support
- Synchronized Scheduling
- Political Scheduling
- Accounting
- Checkpoint/Restart
- Multi-Level Security
- User Database (UDB)
- Job Support
- Process/Job Limits
- System Monitoring
- Data Migration (DMF)
- Cray Tapes
- FFIO Libraries
- Cray ReelLibrarian
- Asynchronous I/O, Listio
- NQE
- DCE/DFS

6. Migration from T3E to SN1

Cellular Irix will appear to users and applications programmers similar to UNICOS, UNICOS/mk and Irix. The system will be binary compatible with Irix. For UNICOS/mk T3E applications, there will be source level compatibility, such that most T3E applications can be recompiled and run on SN1. Cellular Irix will support multi-threading within a cell, and the existing Cray MPP programming models across cells.

The administrator interface for Cellular Irix evolved from Irix, and as such it will be similar to today’s Irix administrator interface. At the same time, a number of administrative features have been added to provide admin functionality comparable to that of UNICOS and UNICOS/mk.

7. Conclusions

Cellular Irix represents the convergence in operating systems direction between the existing Cray and SGI product lines. Cellular Irix will be a single OS that addresses the needs of the various Cray/SGI markets. Source code, functionality, features, and expertise, come from Irix, UNICOS and UNICOS/mk. The system will be feature rich, and it will provide an easy migration for existing Cray and SGI customers. Intermediate Cellular Irix deliverables will provide immediate solutions for the various market segments, while work continues toward the final realization of our joint operating systems direction.