18

Spectrum

Information Systems Industry

Decision Resources, Inc.

Bay Colony Corporate Center
1100 Winter Street
Waltham, Massachusetts 02154

Telephone 617.487.3700
Telefax 617.487.5750

New Microprocessor Standards and Markets, Part I: Technology Assessment

Gordon Bell
Consultant to Decision Resources, Inc.

Business Implications

- RISC architectures offer performance improvement that is 2-3 times better than traditional CISC architectures. However, CISC-based Pentium adopts key RISC architectural concepts to compete on near equal footing with RISC microprocessors.
- Maintaining a microprocessor architecture is expensive over its 10-year life span. An uncompetitive architecture will not produce sufficient volume to pay back investment and support costs. Companies that develop an architecture solely for their own platforms will not be able to sustain a competitive position over the long term.
- Servers powered by multiple new high-performance microprocessors have nearly replaced minicomputers, and are threatening the mainframe market. Supercomputers are being attacked by these same micros, either to perform simple, scalar work or ganged together to perform massively parallel computations. The differences between workstations and personal computers are rapidly diminishing.
- Six similar microprocessors, Alpha, PA-RISC, Pentium, PowerPC, R4X00, and SPARC, will be implemented in a full range of high-powered, low-cost computer-based products, from handheld devices to massively parallel supercomputers. The key to their success will be the availability of new applications that will utilize the speed these micros offer.

In 1982, IBM galvanized the personal computer industry with the IBM PC built around Intel's 8088 microprocessor and Microsoft's DOS operating system. In doing so, it created a single computing environment for over 80% of the personal computers sold in the last decade. In contrast, the workstation (and server) industry, which also began in 1982, evolved around numerous microprocessor (chip) architectures and proprietary vendor-specific versions of the "industry standard" Unix operating system (i.e., Unicee), which resulted in proprietary, locked-in environments. Along with the computing environment, performance and price differences—personal computers have been slower and less expensive than workstations—have clearly distinguished these two markets. These distinctions are rapidly diminishing as a result of two recent developments:

- The introduction of high-performance Pentium chips from Intel and low-cost RISC chips from the leading workstation vendors (Digital Equipment, Hewlett-Packard, IBM, Mips Technologies [subsidiary of Silicon Graphics], and Sun Microsystems)
- Software that allows applications to run on workstations and personal computers, such as Microsoft's Windows NT and Sun's Windows application binary interface, named Wabi

The effect of these developments will be to make microprocessor architectures a high-tech commodity, with little or no differentiation based on software availability and compatibility. Part I of this briefing will examine the evolution of computing classes, the technology of microprocessor architectures, and the influence of the latter on the former. It will conclude with a discussion of the multitude of platforms and applications that these micros make possible, and their effect on other computer classes. In Part II, we will take an in-depth look at six high-performance microprocessors—Alpha, PA-RISC, Pentium, PowerPC, R4400, and SPARC—and the potential for each to prosper over the coming years.

Evolution of Computer Classes

Beginning with the first computers, companies established product classes marked by a hardware architecture, operating system, and market price range, as shown in Table 1. Each of the classes followed a similar pattern of development.

- *First*: New technology allowed a product segment to form with multiple, competing companies.
- *Second*: A decade of stability followed, during which many competitors prospered.
- *Third*: The market consolidated into a few vendors as the industry coalesced around 2-3 architectures.

At the same time IBM was introducing the IBM PC based on Intel's 8088 microprocessor and Microsoft's DOS operating system, Apollo Computer (now part of HP), Sun Microsystems, and others vendors introduced higher-priced workstations based on Motorola's 32-bit, 68000 (68K) architecture and chips. It is entirely plausible that had IBM also chosen the Motorola 68K for its PC instead of the Intel 8088, the distinction between PCs and workstations might never have occurred—in large part because Microsoft would not have been as memory-constrained as it was by the 8088 architecture. Apple's use of Motorola's 68K for its Macintosh personal computer, which first appeared in 1985, demonstrates a realistic direction in which early workstations and PCs might have evolved had IBM made this choice.

The mainframe and minicomputer industries were formed by companies that were vertically integrated, that is, systems were based on each company's standards for every layer of the system from hardware architecture to operating system to system utilities. In contrast, the PC industry is not at all vertically integrated. It is based instead on layers of single unitary standards, starting with Intel's X86 architecture and Microsoft's DOS and Windows operating systems. Languages, networks, databases, and generic and professional applications are all industry segments in a layered fashion. Thus, chips, boards, operating systems and other system software, databases, generic applications (e.g., word processors, E-mail, spreadsheets), and professional applications form the horizontal layers.

Table 1
Computer Classes

Class	Hardware and Software Architectures	When Established	Current Price Range	Other Vendors
Mainframes	IBM 360 and MVS/VM	1964	$500K-20MM	Amdahl, Fujitsu, Hitachi, NEC, and Unisys
Minicomputers	VAX and VMS AS/400 and AS/400 OS HP3000 and MPE	1978 1985	$20K-1MM	Data General
Supercomputers and Minisupercomputers	Cray and Unix[a] Convex and Unix	1976 1983	$2MM-32MM $300K-3MM	Fujitsu, Hitachi, IBM, and NEC
Personal Computers and Simple Network Servers	IBM PC (Intel X86) and MS DOS Apple Macintosh (Motorola 68K) and Macintosh OS	1982 1984	$1K-20K $1K-7K	Hundreds None
Workstations	Sun, HP, IBM, Digital, and SGI and Unicee	1983	$4K-100K	Intergraph, Sony, and others
Servers	Multimicroprocessor servers and Unicee/Windows NT	1990s	$20K-1MM	Auspex and NCR

[a] Originally a proprietary operating system.

Source: Gordon Bell.

Microprocessor Technology

RISC and CISC Architectures

By 1987, most major computer companies had begun to build microprocessor chip-sets using their own RISC (reduced instruction-set computer) architectures to replace the Motorola 68K. RISC was based on an idea created by John Cocke at IBM's T.J. Watson Research Center in the 1970s. Joel Birnbaum carried this concept to HP, which developed the first successful RISC microprocessor (PA-RISC) implemented in systems. Mips Technologies' R2000 and Sun Microsystems' SPARC RISC chips were based on Dave Patterson's and John Hennessy's work at Berkeley and Stanford, respectively. These products were followed by the IBM RS/6000 series RISC workstations, the architecture of the PowerPC chip. In 1992, Digital Equipment finally introduced its Alpha RISC architecture to replace its VAX and Mips R4000-based workstations. Digital simultaneously introduced software to cross-translate user programs to Alpha.

These five RISC architectures offer a performance gain that is approximately 2-3 times better than traditional CISC (Complex Instruction-Set Computer) architectures (e.g., IBM 360, DEC VAX, Motorola 68K, and Intel X86), although they must pay a penalty in increased program size. Since the first RISC chip was introduced, debate has swirled around the relative merits of RISC versus CISC architectures. Meanwhile, Intel has adopted key ideas from high-performance computers (e.g., supercomputers, RISC-based systems) in order to evolve its X86 architecture. Intel's latest implementation of this architecture, Pentium, operates at approximately the same performance level as Sun's SuperSPARC-40 chips. However, Pentium was introduced a full year after the SuperSPARC, and, thus, Sun will likely regain a 50% performance advantage in 1994.

Users, architects, and system implementors should stop debating the relative merits of RISC versus CISC and, instead, focus on performance and other features important to computer applications, such as the number of bits available to address memory. Figure 1 shows the performance curves from their introduction date projected to 1995 of DEC VAX and Intel X86 CISC processors, and DEC and Mips RISC microprocessors. The SPECmark89 benchmark is used as the common performance metric.[1] (Since technology improves exponentially, comparable shipment dates [e.g., first commercial release, steady-state production] must be taken into account to fully understand the performance of various microprocessors. In effect, performance must be discounted at a rate of 60% per year from comparable dates.)

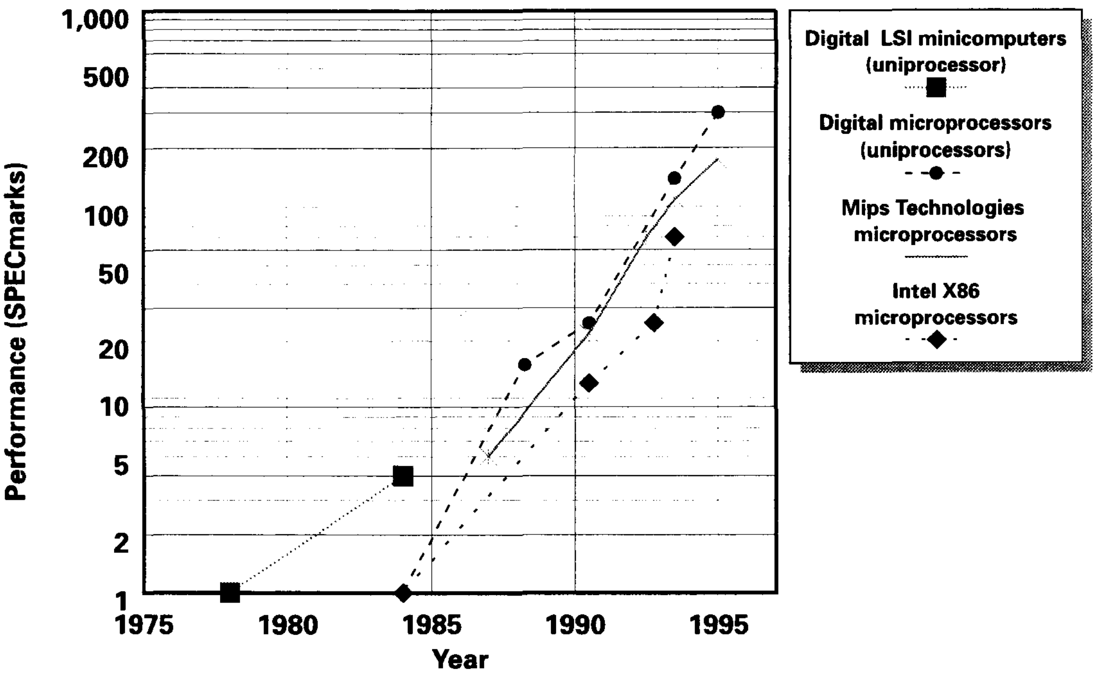
When factoring in the higher cost of VAX systems as shown in Table 1, Figure 1 illustrates clearly why Digital's multichip VAX was replaced by PCs, workstations, and servers (i.e., just workstations in a large box with many disks), powered by single-chip microprocessors. Higher-cost, multiple LSI (Large-Scale Integration) implemented minicomputers like the VAX simply cannot compete with microprocessors that have steeper performance evolution curves. Given that a six- or eight-processor mainframe provides only 50-100 SPECmarks per processor, then it too will be uncompetitive with microprocessor-based systems unless it either (1) becomes smaller and cheaper (e.g., becomes a low-cost server), or (2) provides at least an order of magnitude more computing power without raising the price. Ideally, it should do both by being fully scalable from one to thousands of processors.

In addition to the six microprocessor architectures already mentioned, many other architectures fulfill the needs of specific market niches. Some of these architectures are as follows:

- *ARM 610* (Advanced RISC Machines) and AT&T's *Hobbit*: Low-power microprocessors for personal digital assistants (PDAs), games, and control
- *AMD 29K, Intel i960, Motorola 68K,* and *NEC V-series*: CISC processors for computing, communications, and control devices
- *KSR 1*: The first scalable shared-memory multiprocessor computer
- *Transputer* (SGS-Thomson): For distributed controllers and parallel computing
- *TRON*: Research architecture standard introduced for Japan's next computer generation, but uncompetitive against U.S.-originated microprocessors

[1] SPECmark89 is a measure of processing speed relative to the VAX 11/780 (c. January 1978), combining both integer and floating-point performance. The VAX 11/780 is considered to be a well-balanced machine for both integer and floating-point calculations. SPECint92 and SPECfp92 are formed as the geometric mean of 6 integer and 14 floating-point benchmarks, respectively. The SPECmark benchmarks do not perform input/output operations.

Figure 1
Performance Curve for Several Minicomputers and Microprocessors



Source: Gordon Bell.

- *Intergraph Clipper*: A proprietary microprocessor for Intergraph workstations
- *Motorola 88K* and *Intel i860*: Microprocessors that found only limited application
- Digital signal processing chips and cores from a variety of vendors
- 4-, 8-, and 16-bit control computers and ASIC cores

The Importance of a Large Address Space

In 1993, just as in 1950, the main question about a processor architecture is the number of bits it has to address memory: this determines when programs will be perturbed to support larger memories. Computers need more address bits as time passes because computer memories expand in size with time, according to Moore's Law. (Moore's Law describes the improvement in semiconductor density, i.e., the number of transistors on a single chip doubles every 18 months or 60% growth per annum.) Not having enough address bits is the first mistake made by computer architects.

The first computers could address only a few thousand words. Minicomputers of the 1970s (e.g., the Digital PDP-11) addressed only 64KB. The VAX name was derived from its Virtual Address eXtension to the PDP-11. A program in the VAX's memory could address 4 gigabytes with its 32-bit addresses.

In the late 1970s, Intel followed in the tradition of not having enough bits to address memory: the 8-bit 8086 addressed 1MB, the 16-bit 286 addressed 16MB. Microsoft made a similar error in software by putting a program size limit of 640KB in the first release of DOS. A limit on the number of data items a computer can address has been the key factor in rendering computer architectures obsolete. However, beginning with the 386 and continuing through Pentium, Intel processors can now address 4GB of memory and 64TB (terabytes or trillion bytes) of virtual memory.

The first PCs had about 128KB of main memory using 64Kb memory chips, which required 17 address bits. Today, a typical personal computer has 4MB of main memory using 4Mb memory chips, requiring 22 address bits. This year, 16Mb memory chips will be available. Thus, as computers evolve, one more address bit is needed every 18 months—or 2 more bits every 3 years—just to specify the additional memory. Since a constant dollar amount goes toward memory of a fixed-price computer, memory size quadruples every 3 years.

Similar to Moore's Law is Moore's Speed Law, which states that processor speed doubles every 18 months. A computer design rule of thumb is that memory size must be correlated to processing speed in order to accommodate larger programs that are created to use this additional computing power. Again, chip architectures must increase address bits to keep up with expanding memory, which is required to take advantage of faster processors.

In the 1960s, Gene Amdahl and Richard Case of IBM posited that a 1 MIPS (millions of instructions per second) processor needs 1MB of memory. With larger scientific programs and more multiprogramming, a more realistic rule in today's general-purpose computing environments is that a 1 MIPS processor requires 8MB of memory. For example, a single dense, 3-D data-set of 1000, 8-byte floating-point grid points holds 1,000 x 1,000 x 1,000 x 8 bytes of data-elements, or 8GB requiring 33 address bits. Thus, a 500-MIPS processor would require 4GB of memory, or 32 bits to address the memory. While this may seem like a large number today, tens of gigabytes of memory would be required to randomly access each pixel or sound segment of a 1-hour multimedia presentation. Currently, most PCs and workstation hover around the 1 MIPS/1MB rule.

In addition to larger data-sets, a larger address space permits files to be directly "mapped" into a computer's address space so that a program can directly access data in the file. Accessing data in this fashion eliminates the overhead that would occur in locating a file's data item indirectly. Perhaps the greatest change in applications will occur when very large primary memories hold entire applications that previously would have been structured as database applications. Converting a database application to a straightforward application using conventional data structures and eliminating disk accesses can improve performance by 1 to 2 orders of magnitude. Furthermore, a several-gigabyte database requires only a gigabyte of memory, or thirty-two 256Mb memory chips, which should appear on the market by the end of the decade. For example, an MRP (manufacturing resource planning) application that requires 15 hours to process using the database approach takes only 45 seconds on a Digital Alpha when done in a direct-access fashion. A similar improvement can be realized for airline reservation applications.

Finally, a large address space will facilitate communication for parallel processing and allow a number of computers to operate together on a single application.

Measuring Performance

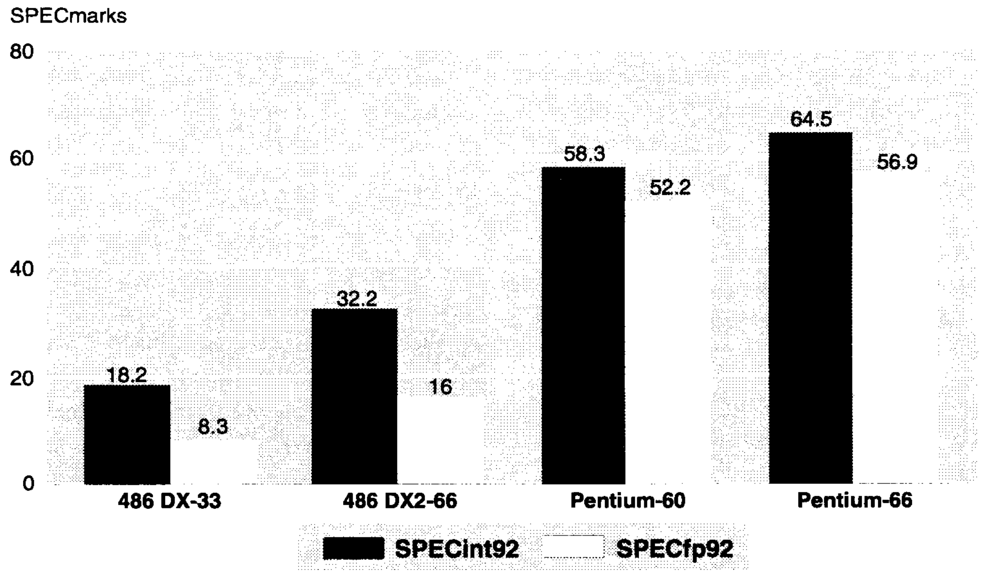
All modern chip architecture system implementations are complex. The designers of these architectures have made a multitude of trade-offs to favor certain applications and benchmarks, such as integer and floating-point SPECmarks. SPECmarks provide a good indication of how much performance is achievable from a microprocessor in a given system (personal computer, workstation, server, etc.) for a specific task. The integer measure shows how well a system may perform for Windows and generic business applications. Floating-point measures performance for technical applications including visualization, simulation of physical systems, mechanical CAD, and numeric analysis.

Intel X86 chips have historically favored integer performance over floating-point performance because they were used primarily in PCs for business applications. However, with the arrival of the Pentium chip, Intel's X86 architecture now has a more balanced integer and floating-point performance characteristic (Figure 2). In contrast, Digital, HP, and IBM engineers have provided exceptional floating-point performance for their respective RISC chips (which are implemented primarily in engineering and graphics workstations) to the detriment of integer performance. As shown in Figure 3, Mips' R4400 and Texas Instruments' SuperSPARC chips—also RISC designs—have achieved a balance more like the Pentium, albeit with substantially lower floating-point performance than their RISC competitors.

While it is difficult to accelerate integer performance for a given clock speed due to limited parallelism, floating-point applications often have a parallel structure, which allows them to be accelerated. In the extreme, the Cray-style vector architecture is best for highly parallel applications.

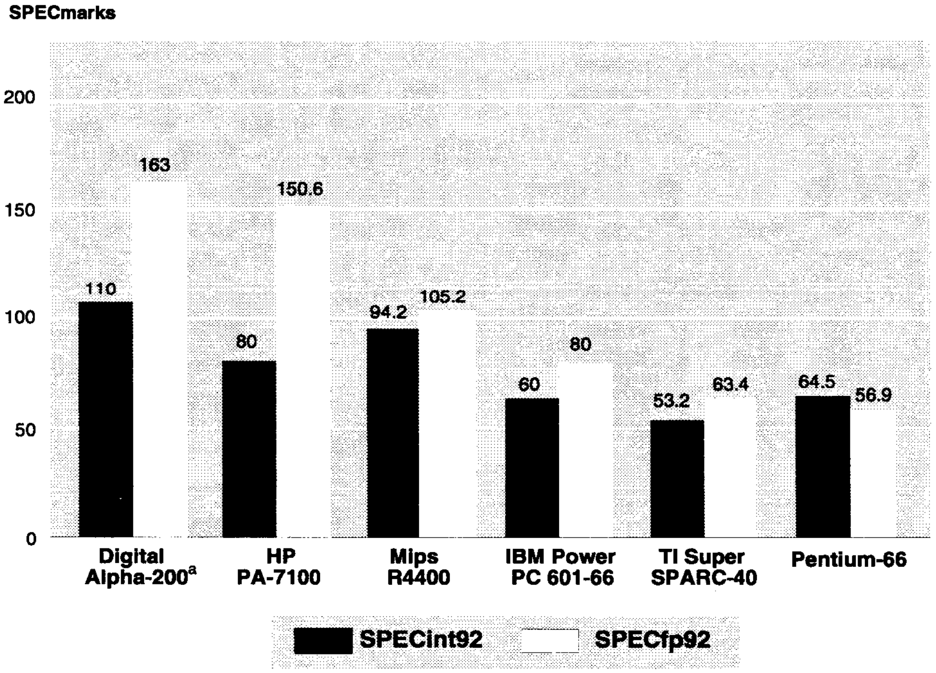
Modern microprocessor-based systems have more variations than in past systems (e.g., two caches instead of one) because there are more functional units and buses linking these units, all of which could potentially

Figure 2
SPECmark Comparison for Intel 486 and Pentium



Source: Intel.

Figure 3
SPECmark Comparison



[a] Alpha performance numbers are for workstation and deskside systems. Alpha implemented in the Digital 10 000 AXP mainframe-class server performs at 200 SPECfp92.

Source: Company data.

be a bottleneck to total throughput. Figure 4 shows the structure of a simple, microprocessor-based uniprocessor system, such as a workstation or low-cost server. Each line interconnecting the boxes represents a bus or collection of wires of a certain width and data rate. Bottlenecks may occur in any of the following areas:

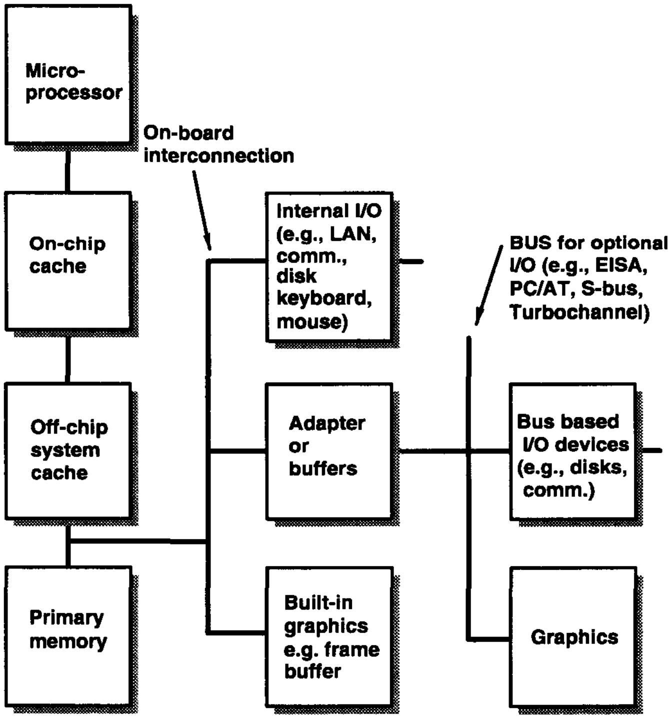
- Microprocessor performance design (e.g., poor floating-point performance) and/or design idiosyncrasies (e.g., performance of a single instruction, such as divide, that does not appear in benchmarks)
- Cache memory size and/or organization
- Translation buffers that map (locate) a processor's primary memory pages in various parts of the caches and memories
- I/O buses, including graphics, disks, and network communications
- Data rates between the processor, cache(s), and primary memory (e.g., in a supercomputer the data rate of the buses that feed a processor is 1.5-3 times the peak floating-point rate in order that an expression such as $C_i = A_j x + B_k$ can operate in a fully parallel [pipelined] fashion)
- Buses that carry data to the internal or external I/O and graphics

Given all of the factors that influence a specific application's performance on a specific system, it is important to remember that microprocessor benchmarks are only a performance indicator, not a guarantee. The only way to understand how well a specific task or set of benchmarks will perform on a particular system is to run the work. Even then, an application may run an order of magnitude slower when initially ported to a system than it will after it has been "tuned." This tuning requires arranging data to avoid cache conflicts, reduce I/O, and so on. When a system is multiprogrammed and I/O and scheduling conflicts occur, even more performance variations will occur.

Development and Implementation Costs

Maintaining a microprocessor architecture is very expensive. Maintaining a poor and uncompetitive architecture may cost a company market share. In today's

Figure 4
Hardware Structure of Typical Workstation or Low-Cost Server


Micro-processor
On-board interconnection
On-chip cache
Off-chip system cache
Primary memory
Internal I/O (e.g., LAN, comm., disk keyboard, mouse)
Adapter or buffers
Built-in graphics e.g. frame buffer
BUS for optional I/O (e.g., EISA, PC/AT, S-bus, Turbochannel)
Bus based I/O devices (e.g., disks, comm.)
Graphics

Source: Gordon Bell.

highly competitive environment, only those architectures that can achieve high volumes will survive long enough to recoup investment and turn a profit.

Microprocessors take about 3 years to develop and implement. Not only must the design team, which can number in the hundreds, develop the chip itself, it must also support manufacturing (e.g., diagnostics, test fixtures). In addition to the chip design, development of compilers and a unique operating system are required. The cost for maintaining an architecture requires a minimum of 300 man-years per microprocessor implementation, or roughly $50 million.

Companies that develop a chip architecture for implementation solely in their own systems (a vanity chip architecture, if you will) are at a competitive disadvantage. If only 50,000 chips are fabricated, the engineering development cost alone for each chip is $1,000. For a large company like Sun, which may fabricate as many as 500,000 chips, the development cost may work out to be only a few hundred dollars or less per chip. Similarly, if a company invests $500 million in a factory to fabricate chips (as Digital is doing), and is able to sell only 1 million chips, the factory amortization costs alone would be $500, independent of the design and manufacturing costs. Any company that wants to succeed in the microprocessor market must be prepared to maintain a technology position over at least the decade-life of a chip architecture, especially those based on a 32-bit or 64-bit architecture.

From a buyer's perspective, a vanity chip, such as Intergraph's Clipper, offers no real benefits. The company typically uses its architecture and applications software to lock users into expensive, low-performance systems, which are available from only one source—the company itself. Over the long haul, it is very difficult, if not impossible, for a company to be a competitive supplier in each horizontal layer of a system solution (chips, platforms, and applications). Users must decide whether the application software that is used to lock them into one platform is so superior as to be worth the cost. Usually, in time, competitive software products reduce or eliminate this advantage, rendering the system solution uncompetitive.

Many companies that used lock-in software to build proprietary hardware have been unsuccessful. Computervision, Daisy, and Valid each started out building vanity workstations (for extra revenue and higher margins) to run their CAD software. Each evolved for a time to use and sell workstations with their software to maintain revenue flow. All three soon became uncompetitive as they faced competition from both CAD companies and workstation companies. Eventually they abandoned their proprietary workstations for "off-the-shelf" workstations and became software-only companies. We are not aware of any company that has successfully developed and marketed chips, platforms, and application software for any length of time.

Chip Architectures and Software

Unix and Open Systems

In spite of open systems and cross-platform compatibility claims, once an architecture is chosen, data begins to accumulate so that it becomes difficult to switch to an alternative architecture. This is true because no two architectures are likely to interpret data, such as floating-point numbers or record layout, identically. Also, Pentium/X86 and Alpha are little-endian byte-ordered (numbering data from the least significant digit), whereas PowerPC, PA-RISC, and SPARC are big-endian byte-ordered. Windows NT functions in little-endian mode, while Unicee are predominantly big endian. However Alpha, R4400, and PowerPC can operate in either mode.

An architecture tends to capture user data hostage in what may be termed a "code and data museum" or a "legacy system." While the choice of an architecture does not appear critical in the short term, it is critical in the long term because it builds in significant "switching" costs when a new architecture is desired. However, having all NT software operate in little-endian mode will improve portability of software from one architecture to another.

With the advent of Unix, an operating system that can be easily "ported" to a variety of computer platforms, computer architecture becomes a somewhat less distinguishing characteristic. Unix allows all platforms, regardless of architecture, to "appear" to be open and accessible. Unfortunately, this is not completely true, simply because of differences among Unicee and variations in architectural details (such as endianness), compilers, and data types.

The original development of powerful RISC microprocessors by several vendors, including HP, Sun, and Mips (now part of Silicon Graphics), initiated a restructuring of the workstation industry aligned along chip architectures. However, partly to maintain product differentiation, partly through a lack of coordination, and partly because of the high NIH (not-invented-here) drive of U.S. engineers, these vendors, along with Digital and IBM, each created a unique and proprietary version of Unix for its workstation platform. Since each of their Unix versions was unique, it became difficult to create a single application program interface to facilitate the porting of applications to each of the six unique chip architectures. Furthermore, some of the architectures run several incompatible Unicee, developed by different OEM vendors.

Unix allows all platforms, regardless of architecture, to "appear" to be open and accessible. Unfortunately, this is not completely true, simply because of differences among Unicee and variations in architectural details (such as endianness), compilers, and data types.

Unix incompatibility was assured from its birth at AT&T's Bell Labs, which provided source language for universities to evolve at will. Bell Labs even prevented each company from using the name "Unix." In the early 1980s, UC/Berkeley created BSD 4.1 while AT&T simultaneously distributed System III (and System V). The race to ensure further Unix incompatibility began in the late 1980s when AT&T and Sun stated that they were going to converge System V and BSD 4.2 and take on the development and standardization role for Unix. Most of the other major Unix platform suppliers viewed this announcement as an attempt by Sun to gain a competitive advantage through its control of Unix; thus, they formed the rival Open Software Foundation (OSF) as a reaction. However, OSF evolved into a product development company for its members, negating its usefulness in setting standards. Only Digital uses the OSF/1 dialect. Furthermore, a subset of Unix, POSIX, is defined as a government procurement standard.

Unix development eventually fell under the auspices of the quasi-independent Unix Systems Laboratories (USL), which AT&T recently sold to Novell. OSF did reach agreement on a common set of Unix commands and utilities, as well as agreement on the use of X Windows. In April 1993, HP, IBM, Novell (USL), The Santa Cruz Operation (SCO), and Sun formed the Common Open Software Environment (COSE) alliance to create a level playing field for competing in the Unix market. The first agreement was the use of X Windows (which all but Sun had previously agreed on). In June, Digital endorsed the COSE process.

Microsoft's Portable Operating System

Microsoft's MS DOS and Windows, which utilizes DOS, have not been portable because they were written specifically for Intel's X86 architecture in machine language. Similarly, IBM's OS/2 is machine-dependent. This architectural dependence has allowed Intel to maintain a fundamental monopoly in the PC market with its X86 architecture. In the fall of 1992, Microsoft released the first beta-test version of its portable operating system, Windows NT. At first release, NT runs on the Alpha, X86, Intergraph, and Mips architectures, and work on porting it to the PowerPC and PA-RISC architectures is under way.

NT can provide a common, higher-level standard for developing applications, both on PCs and workstations. NT has caused the Unicee vendors to try again to standardize on a common user interface and common application program interface (API), which enables an application to be ported to an architecture without having to modify its source code. While a common API does not completely ensure application portability and availability, it does reduce the cost to support an application on a given platform. Unicee vendors, which maintain proprietary user and application interfaces, are threatened by the possibility of Microsoft gaining increased market share through control of a standard interface and as sole source provider of the operating system (NT). This threat was the real spur to the formation of COSE.

Ironically, Windows may turn out to be the only Unix application standard. COSE members agreed to provide Microsoft's Windows 32 Applications Binary Interface (WABI) as another application standard. WABI, in principle, provides any chip architecture that runs Unix access to shrink-wrap Windows applications written for the Intel architecture. For some Windows applications that spend most of their time calling and using the operating system functions, WABI may work with acceptable performance on non-Intel architectures. Application functions that call X86 binary code must be interpreted and, consequently, are likely to run comparatively slower on architectures other than the X86.

Sun introduced the first WABI product, for which it trademarked the name Wabi, in May 1993, which it is licensing to other Unix vendors. Wabi has been adopted by USL and SCO for their Unix versions. Significantly, Windows applications can run under Wabi without Windows itself. (Microsoft has expressed concern that Wabi may infringe on its copyrights, and may consider legal action against Sun.) However, Wabi runs on top of Unix, which requires 2-4 times the memory of a typical PC running DOS and Windows. This creates a cost disadvantage for Unix vendors that will have to be compensated for by significantly higher processor performance or lower prices.

NT will be an important product because it

- Provides an environment for running DOS and Windows applications,
- Is kernel based to provide an environment for hosting other operating systems, including POSIX and Unicee,
- Is platform neutral,
- Provides a separation between the hardware and the operating system software, allowing independent development of platforms and the operating system,
- Is multiprogrammed (like Unix),
- Can utilize multiple processors for both parallel processing and multitasking facilitated by multiple threads of control,
- Is fundamentally real-time with the ability to facilitate multimedia,
- Uses a 64-bit virtual address space, and
- Has adopted a 16-bit unicode byte (for Asian and other markets whose language character-sets require more than the 8 bits of ASCII).

However, the most important attribute of NT is that it is a single standard with one API, one user interface, one set of manuals, a single source code, and so on.

Application Software Accessibility

Users today are most likely to look first at the availability of the vertical application program they require (e.g., CASE, ECAD, desktop publishing tools) before deciding on the computer platform or vendor. Several technologies could affect the future direction of the microprocessor industry by allowing software code written for one architecture to be run on another architecture. That is, if applications can be written to be truly independent of an architecture, an arbitrary number of architectures can be supported. Most important, if the thousands of programs available for X86-based PCs can be run on RISC platforms, then RISC architectures can broaden their target markets significantly. The following key technologies may allow this application independence:

- *Microprocessors that are fast enough to interpret a common binary architecture standard,* such as Wabi. This implies that X86 binary instructions are either directly interpreted or compiled into the target machine for direct, fast execution. For example, Alpha and PA-RISC use this scheme for maintaining backward compatibility for previous applications and architectures.
- *Binary compilers or post-loaders that translate one binary format to another.* These are quite feasible since most RISC architectures are similar. Binary format translation can be used if a chip vendor decides to switch to another architecture. It will have to be used when a platform vendor switches architectures. Undoubtedly, this will occur when at least one to three of the current microprocessor architectures can no longer be justified in the market.
- *An architecture-neutral distribution format (ANDF)* that compiles all source programs into a format that can be rapidly loaded onto any machine for direct execution.
- *CDs and floppies of several hundred megabytes* to facilitate distribution of object files for all architectures. However, a software supplier must still maintain, compile, and test a program on each architecture and API.

Widespread Uses for Killer Micros

Undoubtedly, too many architectures exist in today's general-purpose microprocessor market. History has shown that only two or three architectures prevail for a given class of computers. However, killer micros of the 1990s, designed for high-volume personal computers and workstations, differ from past architectures because they are as powerful as their predecessors (mainframes and minicomputers) at a small fraction of the cost. This enormous price/performance advantage enables a single microprocessor architecture to be used for a wide variety of computer classes—

from hand-held PCs to powerful multiprocessor servers that substitute for mainframes to scalable massively parallel computers that substitute for supercomputers. Thus, a single, general-purpose microprocessor architecture can be used to build a range of chips for many computer products. The following list gives some examples.

- Portable PCs, ranging from palmtops (e.g., HP 100LX) to laptops (e.g., Apple PowerBooks)
- Desktop PCs used as networked workstations, including multiprocessor workstations (e.g., Sun's SPARCstation-10)
- Personal computer or workstation form-factor boxes in tower or desk-side configurations operating as file or computation servers
- Single processor and multiprocessor servers that are maintained by a central MIS organization (e.g., Compaq, NCR, Unisys, and nearly all system suppliers)
- Multiprocessor and multiple-computer servers that have built-in redundancy for database and server applications (e.g., NCR-Teradata, Pyramid, Stratus, Sun's SPARCcenter 2000, and Tandem)
- Massively parallel computers for technical and computational intensive tasks built from a large number (64-1,000) of processors (e.g., Convex, Cray, Intel, Meiko, NCUBE, Thinking Machines)

In addition, these architectures can be used to produce chips for a wide variety of applications that are inherently not general-purpose computers:

- Automobiles
- Printers, communication links, disks, controllers, etc.
- Industrial controller boards or systems and industrial robots
- Application-specific integrated circuit (ASIC) "cores" or "layouts" as part of a chip that carries out a particular application (e.g., modems, cellular phones, PDAs, video games)

Killer Applications

Computer users buy applications! Therefore, the microprocessor architectural story will be written by platform suppliers that use Microsoft's Windows NT operating system to provide applications not economically possible for the plethora of Unix-based platforms.

With the essential switch to larger than 32-bit virtual addresses, new applications become feasible. It is difficult to determine whether a "killer" application will develop to drive the need for faster processors and, hence, the demand for faster computers in the same way word processing and spreadsheets fueled the PC market. Historically, faster computers are almost instantly absorbed into the market. With the new high-performance microprocessors now on the market, just running existing applications faster will be useful, but we doubt that users will pay more simply to run existing word processing, spreadsheets, and e-mail applications. The question is, When is increased processor speed enough?

For the market to absorb the enormous processing power unleashed by new killer micros, new applications must create a need for this power. Fortunately, several generic applications await more processing power and have the potential to fill this role.

- *Multimedia:* Audio, images, video, and eventually high-resolution video will drive the demand for increased processing and secondary memories. Video pornography drove the early adoption of VCRs; CDs storing high-quality images are being used in the same fashion. Faster processing will enable video, thereby enabling interactive TV and VCRs on most desktops.
- *Video:* Video conferences, video phones, and video mail all require faster processors. Current 10-20 SPECmark computers are capable of playing back most video compression schemes. Nearly all video compression algorithms require 10 times more processing power to compress frames for transmission than to decompress frames for viewing. However, a 50-100 SPECmark computer should be capable of compressing video well enough for real-time video, although at a low-quality frame rate and resolution. Progress also continues in improving compression algorithms, which are currently processed by special chips.
- *Object-Oriented Programming:* While progress has been made in interpreting applications written with object-oriented programming (OOP) techniques, OOP applications are less efficient than applications using traditional programming techniques, in which any program can operate on any record without protection. OOP will facilitate interoperation of applications, but requires 2-10 times more running time, especially if programs are interpreted.

- *Speech Recognition:* The prediction of when we will be able to speak to a computer and have it understand us has been drawing closer as processing power has increased. For example, Kurzweil AI has a speech input product to facilitate generation of medical records in specific domains like pathology, radiology, and emergency medicine. Dragon Systems provides the technology for replacing the text or mouse command interface with spoken commands using a 486-based PC. Useful, more-general-purpose speech-input-based programs will begin to appear on computers that operate at 100 SPECmarks or more.
- *Rule-Based "Agent" Programs:* As the amount of information arriving at the desktop increases through electronic mail and other forms, it will be useful for computers to assist in the sorting, filtering, and digesting of this information.

For the next decade, technical users will be able to use and pay for increased processor power, resulting in shorter response times, larger and more detailed simulation and analysis, and more realistic visualization brought about by 3-D graphics and image processing. With more realistic models of everything from atoms to automobiles, virtual reality applications are possible (e.g., enabling a viewer to "walk through" a home). This will facilitate the design, manufacture, purchase, and maintenance of many goods, from new drugs to artificial limbs.

Killer Micros' Market Impact

Mainframes and Minicomputers. Table 1 showed the plethora of servers, ranging from a simple personal computer to a multiprocessor, that are enabled by microprocessors. These are the basis of why minicomputers have almost entirely disappeared, and why mainframes are being placed on the endangered species list. In essence, minis and mainframes—characterized by the water-cooled, ECL technology IBM 360—are obsolete, to be replaced by scalable servers built from scores or hundreds of killer CMOS microprocessors. These servers will reside in the "glass houses" that currently hold mainframes, and will be managed by MIS departments.

The notion of client/server computing was invented by the mainframe and minicomputer industry to hold off the attack of killer micros in the corporate and distributed computing world. Clearly down-sizing is occurring as users find out that a collection of microprocessor-based servers will go a long way toward carrying the computing load for a company or department in a highly distributed and incremental (scalable) fashion.

Proprietary minicomputers have essentially disappeared. Multiple micro-based Unix servers ($50K-1MM) are replacing them and mainframes. Million-dollar mainframes will be harder to displace since they act as "code museums" that hold corporate data and programs hostage.

The basic problem with mainframes is that their structures evolved in an ad hoc way to deal with computing, secondary memory, and communications. Today, no one would architect a large computer as an ad hoc network of special-purpose computers for computation, switching, I/O and disk control, and diagnostics. Amdahl, Fujitsu, IBM, Hitachi, NEC, and Unisys designers have hung on to the wrong technology and structure too long in hopes that users are too locked-in to downsize. Four computer structures will replace these expensive water-cooled, ECL-technology mainframes:

- *Mainframes* using CMOS technology to lower costs.
- *Limited scalable multiprocessors* (mPs), such as Sun's 2-20 processor SPARCcenter 2000, which are used just like traditional mPs. Others include mP workstations, and mP servers from Digital, IBM, HP, NCR, Sequent, Stratus, Tandem, etc.
- *Truly scalable multiprocessors,* the first of which has been delivered by KSR. It provides mainframe capability at lower cost per operation. However, its real advantage is scalability, which enables it to expand from a simple server to the price of a mainframe, while providing 50-100 times the processing power and I/O throughput.
- *Computer clusters* formed from a network of uniprocessor PCs and fixed-function servers. Faster and more reliable communications switches, such as faster LANs and ATM switches, allow the use of distributed, low-cost, powerful nodes to build scalable mainframes from high-volume, low-cost parts.

Supercomputers. Future generations of supercomputers, based on the Cray design formula of the fastest clocks and vector multiprocessors, can be built and justified for current supercomputer applications. However, microprocessor-based systems are fast claiming niches from the supercomputer market such that no company will soon be able to sustain the significant i nvestment required to design future supercomputers.

The worldwide market for traditional supercomputers is projected to be flat for 1993 (approximately $2 billion, plus the attached vector units that IBM sells as mainframes). This market must sustain investments by Convex, Cray Computer, Cray Research, Fujitsu, Hitachi, and IBM. If only 15% of the supercomputer revenues of these companies goes into R&D investment, this leaves just $300 million available for R&D for the entire industry. Cray Research alone spends about $150 million a year on R&D.

Microprocessors are taking market share from traditional supercomputers in two niches:

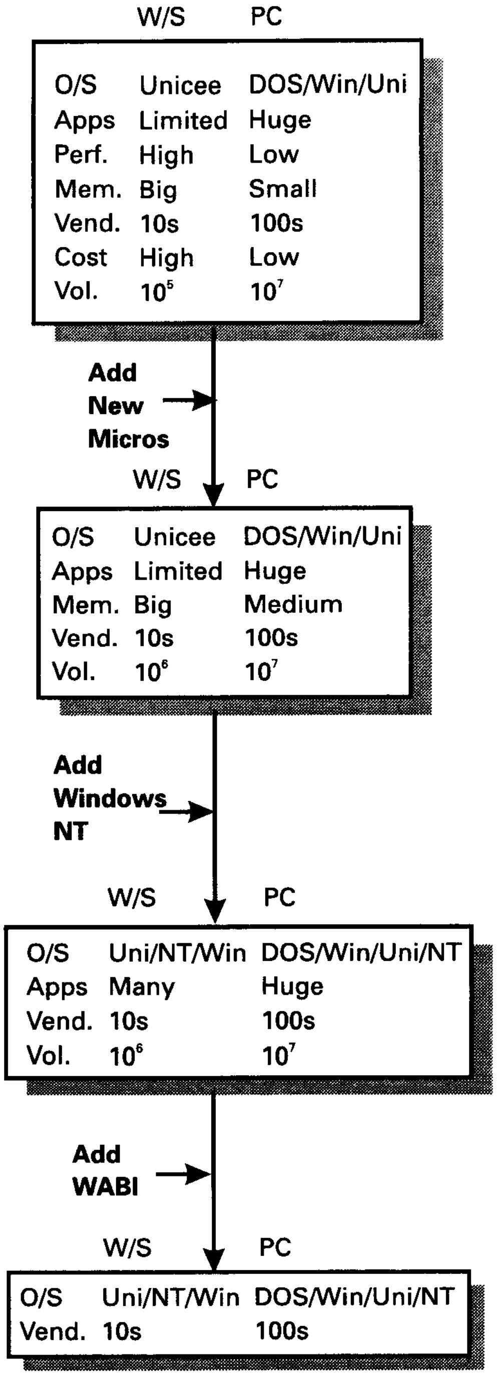
- Fast microprocessors that perform all the simple, scalar work, including compilation and visualization, at ⅓ to ½ the speed of a supercomputer, but at 1⁄10 or less the cost per operation.
- Gangs of microprocessors lashed together (i.e., massively parallel computers) performing highly parallel computations that do not require close communication.

All supercomputer vendors are designing massively parallel computers in addition to their traditional designs. The U.S. government is funding companies (e.g., Thinking Machines and Intel) that will compete with supercomputer companies, further accelerating their decline.

Workstations and Personal Computers. Users now are wary of being locked into a single architecture, preferring the freedom to choose the best platform at the time of purchase. Having the Windows NT operating system available on virtually every relatively indistinguishable workstation and PC platform provides the means for this freedom and, in the process, considerably reduces the distinction between PCs and workstations. Workstation vendors have announced their intention to use a Windows application binary interface to gain access to shrink-wrap Windows software, and Novell and Sun are pushing Unix for personal computers, particularly in the server market. Several software advances will continue this trend to application portability through higher-level languages and higher-level interfaces including 4GL languages such as SQL.

How well these moves will work is unclear. What is clear is that the distinction between high-end personal computers and low-end workstations has become a thing of the past, as shown in Figure 5. The result of this new dynamic will be to make all personal computer and workstation suppliers accessible to all

Figure 5
Differences Between Personal Computers and Workstations



	W/S	PC
O/S	Unicee	DOS/Win/Uni
Apps	Limited	Huge
Perf.	High	Low
Mem.	Big	Small
Vend.	10s	100s
Cost	High	Low
Vol.	10^5	10^7

	W/S	PC
O/S	Unicee	DOS/Win/Uni
Apps	Limited	Huge
Mem.	Big	Medium
Vend.	10s	100s
Vol.	10^6	10^7

	W/S	PC
O/S	Uni/NT/Win	DOS/Win/Uni/NT
Apps	Many	Huge
Vend.	10s	100s
Vol.	10^6	10^7

	W/S	PC
O/S	Uni/NT/Win	DOS/Win/Uni/NT
Vend.	10s	100s

Legend:

O/S	Operating systems	Cost	Cost of platform
Apps	Applications availability	Vol.	Volume of units shipped
Perf.	Performance	Uni	Unicee
Mem.	Memory capacity	Win	Windows
Vend.	Vendor base	NT	Windows NT

Source: Gordon Bell.

channels of distribution and vice versa. This dynamic will also increase the intensity of competition in both markets—and expand the size of the market accessible to vendors as well.

Conclusions

For the foreseeable future, six microprocessor architectures—Alpha, PA-RISC, X86, PowerPC, Mips R Series, and SPARC—are most likely to be implemented in everything from hand-held computers to massively parallel supercomputers. The immediate effect of having these relatively comparable chips from high-volume producers will be higher-powered, lower-cost machines across these computing classes. However, historically, a large number of architectures for a given computer class evolve to just two or three. For example:

- Eight 1960s mainframe architectures standardized on the IBM 360, with two others still in existence.
- Over 100 minicomputers evolved to essentially the VAX, HP3000, and AS/400 architectures. Today's minicomputers use multiple microprocessors. (Servers powered by microprocessors have essentially replaced the minicomputer.)
- Numerous attempts at building personal computers resulted in the Apple Macintosh and the IBM PC (i.e., Intel X86 and MS DOS) and compatibles, plus a few niche machines.
- Workstations have consolidated to five primary architectures with large PCs being used for similar applications.
- The $2 billion supercomputer market is fundamentally unable to profitably sustain six unique architectures.

It is difficult to predict how successfully the five RISC architectures can compete with the Intel X86 architecture in the PC arena, or whether personal computer vendors with Pentium-based systems will push into the higher-priced workstation market. The answer depends on whether new applications software will emerge to utilize the increased performance of these systems.

In addition to the effect on the PC/workstation marketplace, the ability to build powerful servers, which have essentially eliminated the VAX part of the minicomputer industry (but not yet IBM's costly AS/400, in part due to superior marketing including the FUD [fear, uncertainty, and doubt] factor), is almost certain to continue to affect the decline of—and possibly eliminate—the mainframe (i.e., IBM 360-based architecture). Mainframe designers have not utilized technology to build scalable structures from a small number of component types. The mainframe architecture is obsolete; large, scalable computers managed by central service providers will prevail. Already, application system designers are redesigning systems based on computers that are fully distributed and networked, powerful, low-cost, and microprocessor-based.

About the Author

C. ***Gordon Bell*** *is a computer industry consultant at large. He spent 23 years at Digital Equipment Corporation as vice president of research and development, where he was the architect of various minicomputers and time-sharing computers and led the development of Digital's VAX and the VAX environment. Bell has been involved in, or responsible for, the design of many products at Digital, Encore, Ardent, and a score of other companies. He is on the boards and technical advisory boards of Adaptive Solutions, Chronologic Simulation, Cirrus Logic, Kendall Square Research, Microsoft, Visix Software, University Video Communications, Sun Microsystems, and other firms.*

Mr. Bell is a former professor of Computer Science and Electrical Engineering at Carnegie-Mellon University. His awards include the IEEE Von Neumann Medal, the AEA Inventor Award, and the 1991 National Medal of Technology for his "continuing intellectual and industrial achievements in the field of computer design." He has authored numerous books and papers, including High Tech Ventures: The Guide to Entrepreneurial Success, *published in 1991 by Addison-Wesley. Mr. Bell is a founder and director of The Computer Museum in Boston, Massachusetts, and a member of many professional organizations, including AAAS (Fellow), ACM, IEEE (Fellow), and the National Academy of Engineering.*

Eric P. Blum, Research Program Manager

93-11-50

About Decision Resources

Decision Resources, Inc. (DR), is an international publishing and consulting firm that evaluates worldwide markets, emerging technologies, and competitive forces in the information technology, life sciences, and process industries. DR links client companies with an extensive network of technology and business experts through consulting, subscription services, and reports. For additional information, please contact Joan Smith by phone at (617) 487-3731 or by fax at (617) 487-5750.

This material, prepared specifically for clients of Decision Resources, Inc., is furnished in confidence and is not to be duplicated outside of subscriber organizations in any form without our prior permission in writing. The opinions stated represent our interpretation and analysis of information generally available to the public or released by responsible individuals in the subject companies. We believe that the sources of information on which our material is based are reliable and we have applied our best professional judgment to the data obtained. We do not assume any liability for the accuracy, comprehensiveness, or use of the information presented.