

The Evolution of Switch Intelligence: An AT&T Network Perspective

Carl E. Betta
John B. Lewis
Larry A. Russell

Since the mid-1970s, the 4ESS™ switch has been the principal switching system—first in the AT&T Long Lines network, and today in the AT&T network as a whole. Until now, its main call-processing engine has been the 1A processor. The 1A is an ultra-reliable central controller designed to support high-volume electronic switching of simple phone calls having few if any special service features, like those in AT&T 800 service. In recent years, however, rising call volumes and sophisticated new long-distance service features have created the need for much more processing capacity and “intelligence” in the 4ESS switch. To meet this need, the AT&T Network Services Division—which manages the AT&T network—chose AT&T Network Systems Group’s 1B processor to replace the 1A processor in 135 4ESS switches. The 1B processor more than doubles the call-handling capacity of the 4ESS switch while performing at least as reliably as the 1A processor. Not only that—the 1B processor can connect to a community of other processors, databases, and switch fabrics.

Introduction

Rising call volume and new long-distance service features have created the need for much greater processing capacity and “intelligence” in the AT&T 4ESS™ switch. To meet this need, the AT&T Network Services Division chose the AT&T 1B processor to replace the 1A processor, or central controller, in 135 4ESS switches.

Background.¹ The 1B processor is the direct descendant of the 1976-vintage 1A processor, the central controller within AT&T’s first fully stored-program-controlled (SPC) toll exchange, the 4ESS switch (Figure 1). Being SPC means its call processing is defined and executed from software—that is, from instructions and data stored in computer memory. Before 1965, when AT&T fielded the first SPC switch, a local exchange called the No. 1 ESS™ switch, call processing was controlled by hard-wired logic circuits usually made up of electromechanical relays.

While not as technologically colorful to some people as relay-based switching

machines, the “steam engines” of telephony, these computer-like SPC switches would have far greater flexibility and processing power. Although hardware changes would always be needed for some new switching tasks, the bulk of new services and switching features could now be developed in software.

Switch Exhaust. In 1976, when the 1A-processor-based 4ESS switch first came out, it was expected eventually to exhaust:

- Its supply of trunk terminations;
- The number of calls it could simultaneously serve without the chance of blocking newly arriving calls; or
- The number of new calls it could set up per hour.

Throughout the network, AT&T planners expected exhausts of all three resources—at different switches.

But the flexibility of SPC provided the opportunity to market new services based on more complex, more time-consuming call processing, and soon the limit on

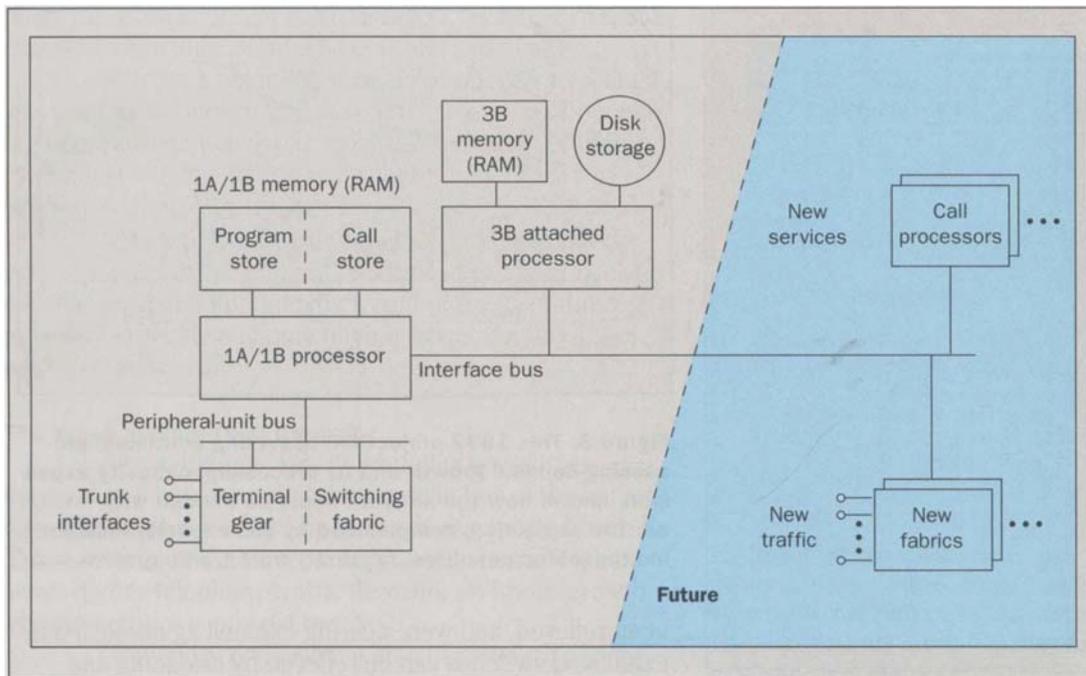


Figure 1. The current 4ESS-switch architecture with either the 1A or 1B processor is shown on the left in this block diagram. The 3B attached processor gave the 4ESS switch its first processing-capacity increase back in 1983. The interface bus, available only on the 1B processor, is a gateway not only to greater processing capacity but also to new switch fabrics.

Panel 1. Abbreviations, Acronyms, and Terms
 CPU—central processing unit
 NEBS—North American Equipment and Building Standards
 SPC—stored-program-controlled or stored-program control

call rate became dominant.

This call-rate limit is a *real-time* limit because it's reached when the processor has no more time to process calls. As the years passed, calls tended to use more real time for two reasons:

- Call processing overall became more complex to allow a greater choice of processing steps. This meant that more real time would be needed—because of this expanded repertoire—to do even the pre-existing, standard call-processing tasks.
- AT&T customers *chose* to place more-complex calls. This changed the call mix toward greater overall complexity.

Owing to this trend, processor real time limited the 4ESS-switch call rate so severely that in 1983, AT&T

decided to augment the 1A processor by adding a 3B20D computer, called the 3B attached processor² (Figure 1). The 3B attached processor, though, has been confined to off-loading certain administrative and signaling tasks from the 1A processor because of the 3B's limited real-time capacity as well as other reasons.

In this period, AT&T product and service planners didn't expect memory to exhaust ahead of any of the three other resources listed above—namely terminations, simultaneous calls, and real time. But in time, memory also became critical.

In the 1A processor, memory addresses are 22 bits long, so they can address up to about four million words of memory. This memory is divided into two halves:

- *Program store*, with a two-word fetch for efficiently feeding instructions to the processor, and
- *Call store*, which is both scratch-pad memory for calls in progress and "static" memory, used for call-routing tables, for example.

Large features typically have taken half a percent of program store, and a year's worth of features has taken from three to five percent of call store. When avail-

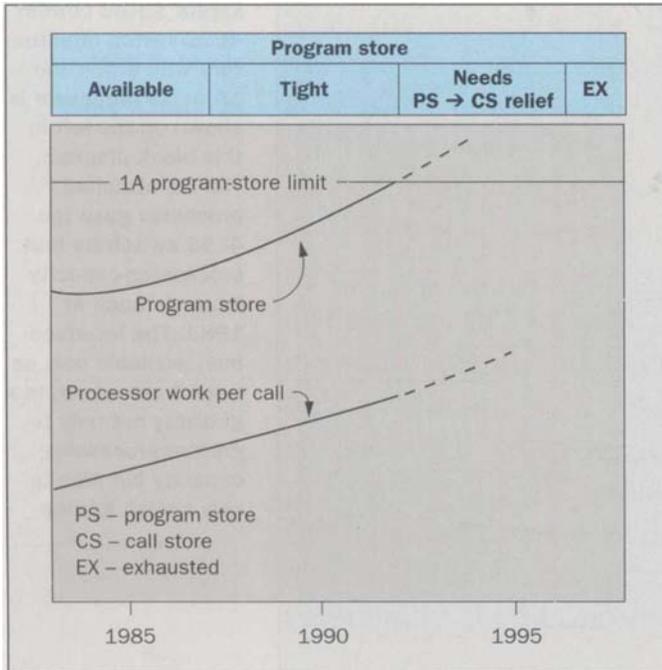


Figure 2. The average processor work per call is raised both by adding features in software and having customers use features in their calling. Large features have usually taken half a percent of program store, and a year's worth of features has recently taken three to five percent of call store. The curves show a long-term view of work per call and program-store usage. Planners saw the need for action as program store approached its limit.

able, however, call store can be in almost unlimited demand—as an alternative to network-based databases, for some features. A long-term view of real-time and program-store usage is shown in Figure 2.

Exhaust Relief. The two contending kinds of exhaust just before the 1B project, namely real-time and memory exhaust, have important differences. Real-time exhaust is highly dependent on the number of trunks served by the switch. However, even if one stops adding trunks just before real-time exhaust, the switch will still exhaust later as the calls on existing trunks use more real time.

Looking over the network, then, there were two kinds of switches: smaller switches that hadn't reached exhaust; and larger switches that had neared exhaust,

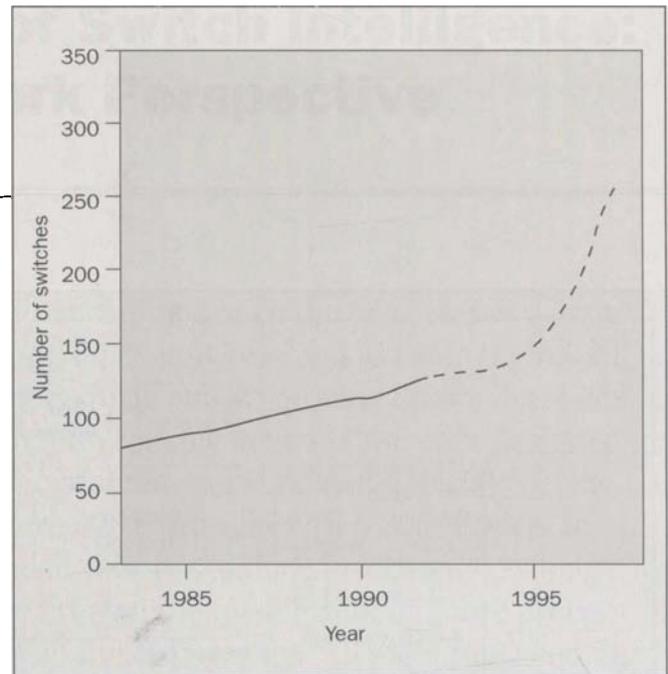


Figure 3. This 1992 projection, assuming continued processing-demand growth and no processing-capacity expansion, shows how the network could be flooded with switches. The steep rise, compounded by call-capacity loss and multimachine penalties, is mostly from traffic growth.

been relieved, and were nearing exhaust again. Real-time exhausted switches can be relieved by *deloading and rehoming*, which is moving some of their trunks to new or existing switches that have spare capacity, or by other means, such as adjusting network architecture to reduce the need for switching in the first place.

The network architecture was adjusted so areas that called each other a lot tended to be served from the same switch, but mostly switches were deloaded and trunks rehomed—a costly affair. On top of the labor expense, new terminal gear at the relief switch had to be bought while leaving costly terminal gear on the full switch *stranded*. In theory, the terminal gear could have been moved from the full switch to the relief switches; but this would have cost even more for the labor, called “grooming,” needed to free movable units of such gear. This is because, for reliability, the trunks to be rehomed are usually distributed across most of the terminal gear on the switch rather than being confined to a few units of such gear.

On the other hand, memory exhaust depends only weakly on the number of trunks and is barely relieved by adding switches to the network. Program-store usage doesn't depend at all on the number of trunks. So, memory exhaust threatens all switches at once. Some call-store capacity is freed by removing trunks and then, when needed, moving a few seldom-used

programs from program store to call store, but each trunk removed frees only about a dozen words of memory.

So from a planning standpoint, memory exhaust was a serious concern and real-time exhaust was becoming ever more expensive to relieve. The threat of memory exhaust was the single most compelling reason to begin the 1B project (Figure 2).

Although the threat of memory exhaust got the project started, a doubling of real-time capacity is its clearest near-term benefit; but why would one expect the cost to be any lower than simply buying twice as many 1A-equipped switches?

The Value of Real-Time Capacity

A new, higher-capacity switch processor, as opposed to filling the network with switches of the current size (Figure 3) is justified by the savings expected from it. Those savings are in the cost of growing the network to serve higher telephone-traffic demand. (Without growth, of course, there is no cost to save.)

Growth is costly even with a big enough processor, but it can get much costlier if the processor is too small. For an existing network, the minimum acceptable processor size depends mostly on the size of the processor demand and on the number of switches already in the network. Not all the cost of growth is related to the size of the switch processor, but what is related is stranding cost, per-switch cost, and the multimachine penalty.

Stranding Cost. As mentioned earlier, deloading and rehomeing leave valuable terminal gear unused, or *stranded*, while requiring new terminal gear to be bought to handle the same traffic at the relief switches. This stranding cost is a major reason for raising real-time capacity instead of simply buying more switches, because raising it can eliminate full switches and the need to deload and rehome. Furthermore, after the capacity has been raised, the terminations stranded by earlier processor exhausts can often be reused and some of the stranding cost recovered.

Per-Switch Cost. Each switch must have a processor, of course; but besides that, each time a new switch is established, a network provider also must pay:

- Capital dollars for all the other switching and transmission hardware, power, and properly conditioned floor space that is needed before the first call can be switched, and

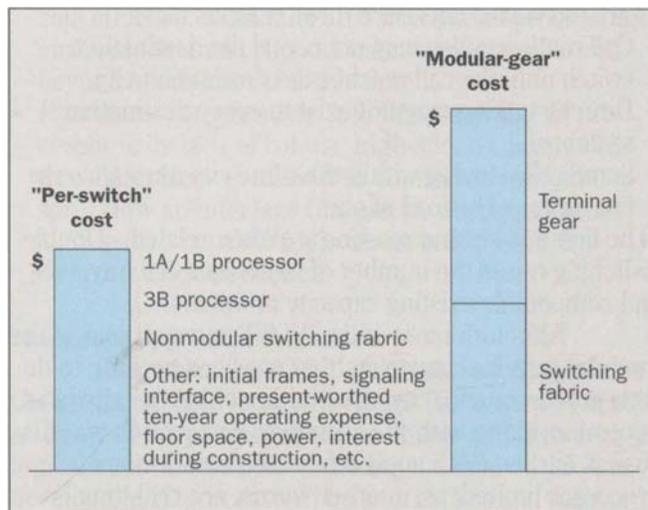


Figure 4. The bar graph shows the per-switch and modular-gear costs for a single, typical switch in the AT&T network. When switching cost was tabulated for the network as a whole in planning studies leading to the 1B, per-switch costs made up most of the part that depended on switch capacity.

- Annual expenses for operating and maintaining it. These added costs—above the cost of the processor itself—vary of course from site to site, but AT&T network 4ESS switches roughly fit the picture in Figure 4.

Not included in the per-switch cost is what can be the bulk of switching cost: the switching and terminal gear that is modular and bought as the switches grow in size. This modular-gear cost is the taller, right-hand bar in Figure 4. Surprisingly perhaps, as long as the average number of switches a call traverses stays about the same (for example, about two in the AT&T network), this gear is bought independently—for the most part—of the number of switches in the network as a whole. The small part of the modular cost that does depend on the number of switches is called the *multimachine penalty*, which is explained next.

The Multimachine Penalty. The *extra* cost of growing the network when switches are small and therefore numerous is known as the multimachine penalty. The cost penalty is for less-efficient trunking (called a trunk-splintering penalty) and for more double switching.

Double switching happens when a single city must be served by multiple switches because the switch-

es are too small. There are three reasons for it:

- Call-routing tables may not reveal the destination switch until the call reaches the destination city;
- Direct trunking may not exist to every destination switch; and
- Some calls staying within the same city must traverse two switches instead of one.

(The first and second reasons are often related.) Double switching raises the number of switches a call traverses and compounds existing capacity problems.

Although most of the AT&T network cost of the modular terminal and switching gear has nothing to do with processor capacity, the multimachine penalty is an exception, along with the stranding cost, which was discussed earlier. Both apply whenever switches are processor limited; or, in other words, are real-time or call-rate limited.

The Long-Term Outlook. The annual-cost profile with and without the 1B capacity expansion predicted for 1994 is shown in Figure 5. This is just the sum of the stranding costs, per-switch costs, and the multimachine penalties. The negative cost, or cost saving, is from the reuse of stranded terminations to serve growth traffic. There is a sizable advantage—again assuming growth continues—in raising 4ESS-switch real-time capacity above that of the 1A processor. This is true counting the cost of developing the 1B processor, which is included in the processor price.

Figure 5 also shows that with continued growth, even the 1B becomes too small, and the need for more processing capacity recurs. In the long term, other resources (such as the maximum number of terminations and simultaneous calls mentioned in the beginning) will also need to be expanded.

High-Level Network Requirements

Based on the preceding observations and other considerations, high-level requirements were developed that reflect AT&T network needs in a 4ESS-switch processor upgrade. These requirements were used to guide all facets of the project: writing detailed requirements, deployment planning, and so forth. Other papers in this issue, especially those by Hsu and Seese³ and Hillis et al.,⁴ describe these activities in more detail. These high-level requirements are as follows:

- *Preserve the embedded base of 4ESS-switch terminations*

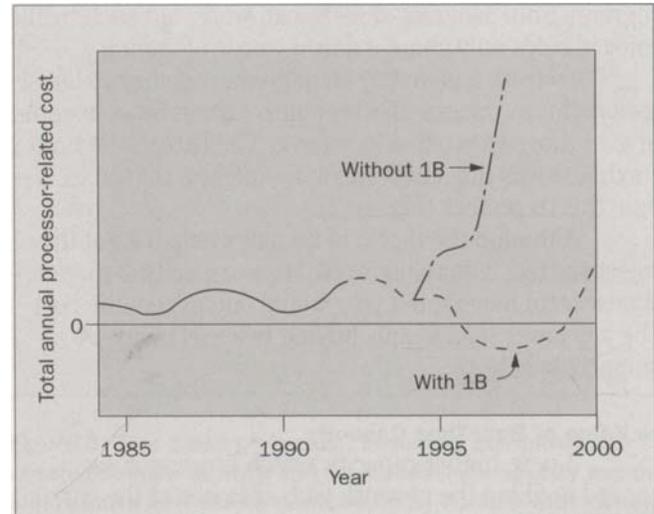


Figure 5. This annual-cost profile is a 1992 planning view with and without the 1B capacity expansion predicted for 1994. This profile only covers costs that depend on processor capacity: per-switch and stranding costs and multimachine penalties. (The large part of modular-gear cost that depends only on traffic growth is not included.) The dip on the right, a cost saving, is from the reuse of stranded terminations. With continued growth, of course, the 4ESS switch will again need more processing power, as well as other kinds of capacities, such as switch-fabric capacity.

and application software. By far, the largest AT&T investment in the 4ESS switch is in its terminal gear, office transmission-terminal equipment, and application software. As discussed earlier, the main reason for pursuing a switch-processor upgrade is to avoid either replacing existing switches or buying new ones. More importantly, more than 20 major 4ESS-switch application-software releases have been developed thus far to meet continually evolving customer and network needs. Such a large volume of software would be very costly to reproduce.

- *Maintain the same interfaces to other systems.* A typical switching system has interfaces to many other systems. These include other network elements, such as signal transfer points, which route common-channel signaling messages among switches and network databases. These signaling messages set up calls through other switches and also retrieve service information, such as 800-service routing, from centralized databases called

network control points. Switches also interface to the operations-support systems used to provision and maintain switches, collect billing information, and do other important jobs. To change these interfaces would require a large investment in software development, and it would create a difficult transition problem.

- **Ensure high reliability.** To satisfy customers' expectations, the AT&T network must continue to meet stringent reliability requirements. Generally speaking, the 4ESS switch must provide greater than "five nines" reliability; that is, the system must be up more than 99.999 percent of the time, which translates to an average unplanned downtime of less than three minutes per year. (Typical commercial computer systems are down for many hours per year.)
- **Maintain uninterrupted customer service during 1B deployment.** Nonstop operation, 24 hours a day, seven days a week is a hallmark of the AT&T network. Not only must interfaces to customers be maintained through the transition from the 1A to the 1B processor, but also the transition cannot interrupt a significant number of calls in progress. Accordingly, the transition must be made in a brief interval typical of yearly software upgrades (several minutes) late at night, when traffic volumes are low. Also, robust "back-out" procedures must be available, which allow the process to be reversed quickly if something goes wrong.
- **Complete deployment within one year to minimize dual feature development and testing.** No processing can be upgraded without affecting *some* software, both application software and especially lower-level software, which is hardware dependent. More importantly, even if there were no software changes, every new feature would have to be tested, and every existing feature would have to be regression tested in both processing environments. This is to ensure reliability and accounts for a large proportion of the development cost. Completing deployment in one year minimizes this cost because only one generic software release then incurs this "dual-development" penalty.
- **Significantly increase memory and real time to provide for several years of growth.** As discussed in the section before, this is the main reason for doing a switch-processor upgrade. But given the complexity and cost of such an upgrade, a small improvement that would suffice for only a couple of years simply would not be

cost effective. In addition, the new switch-processing environment must support memory expansion well beyond needs known at this time.

- **Provide an interface bus.** A limitation of the 1A processor is its lack of robust, high-speed communications interfaces to allow it to connect to new processors. How an interface bus can be used to expand 4ESS-switch capabilities is discussed later in the Future Evolution section.

Processor Upgrade Alternatives

Together, the AT&T Communications Services Group and AT&T Network Systems performed in-depth analyses of the preceding requirements and their technological implications before making the final decision to develop the 1B processor. The alternatives and their pros and cons follow.

Commercial Processors. At first glance, the use of a commercial processor to replace the 1A processor would seem to be the least-expensive way, both in development and capital, to meet 4ESS-switch processing-capacity needs. Commercial processors are available from several vendors, and stiff competition makes them inexpensive for the processing capacity and memory they provide. Use of existing processors that met AT&T requirements would have meant low or no development cost for the processor itself. Significantly, these processors are on a "technology curve," which promises continuing improvement. Unfortunately, commercial processors have several drawbacks, and they do not meet the following two major requirements:

- **Reliability.** Commercial processors typically have downtimes of several hours or more per year. The best generally available fault-tolerant processors have downtimes of 30 to 60 minutes per year. The 4ESS-switch requirement, as discussed earlier, is at least ten times better: less than three minutes of downtime expected per year.
- **Compatibility.** The 1A processor's instruction set is designed specifically for telecommunications applications and is unique to 4ESS and 1A ESS switches. Changing to a different instruction set would violate the application-software compatibility requirement, resulting in much higher development costs. Automatically "porting" the software to a new instruction set or emulating the 1A instruction set on a differ-

ent processor would provide questionable performance and probably unacceptable reliability.

The physical interfaces the 1A provides to the 4ESS-switch peripherals via the peripheral-unit bus and auxiliary-unit bus are also unique. Finally, the 4ESS switch must meet North American Equipment and Building Standards (NEBS) for central-office equipment, which include operation on 48-volt dc power and under extreme environmental conditions. Commercial processors typically do not meet these standards.

Custom Processor Based on Commercial CPUs.

Commercial central-processor-unit (CPU) chips, such as those produced by Intel* and Motorola,* have been successfully incorporated into highly fault-tolerant processor systems used in such products as the AT&T 5ESS® switch. As with commercial processors, though, the main problem with this approach is instruction-set incompatibility.

Adding Processors to the 4ESS Switch. Developers considered various ways to augment the 1A processor. These included, for example, off-loading call-processing functions to switch adjuncts, which connect to a switch through conventional signaling interfaces. Again, the most serious drawback of this approach is the lack of application-code compatibility. The high cost of these stand-alone systems and the performance of the available interfaces were also of concern. Moreover, the 1A real-time overhead in communicating with these systems is often greater than the real time saved in the 1A processor by off-loading call-processing functions.

1B Processor. Developing this processor was chosen as the only alternative that would meet all the major 4ESS-switch processor-upgrade requirements. Although the 1B makes extensive use of standard, commercially available components at the chip level, from the board level up it is a custom design—necessary for instruction-set compatibility. A custom design allows the 1B processor, like the 1A, to achieve extraordinarily high reliability through full duplication of the CPU and cross-CPU checking of the results of each instruction to detect errors. Unlike the 1A processor, the 1B has been designed with 32-bit buses to allow memory expansion. The 1B's design also includes a high-speed interface bus to which additional processors and other peripherals can be attached. Its high degree of application-software compatibility means that the interfaces to almost all operations-support systems are unchanged. A sophisticated deployment

plan, which allows rollback, has been developed to minimize the possibility of disrupting calls during retrofit.⁵

Future Evolution

As part of an overall 4ESS-switch plan, the 1B processor has been designed from the outset with evolution in mind.⁶ Early in the program, planners realized that the 1B, which has twice the 1A's processing power, could not meet AT&T switching needs indefinitely.

Long-Term Need. As discussed in the Introduction section, the 4ESS switch with the 1B processor today is limited by the number of terminations the switch can physically support (soon about 127,000). Eventually, the 1B will once again limit 4ESS-switch capacity because of rising feature complexity. More importantly, though, 4ESS-switch capacity expansion will be needed to meet growing traffic demands in the AT&T network.

One possible solution is to build a single-processor "1C" with greater capacity than the 1B. But the trend today in designing for higher performance is to rely not so much on faster electronics in a single processor as on parallelism and multiprocessing. Even if another two- or three-times improvement over the 1B could be achieved with a single-processor approach, there is a critical need to support a more flexible software environment, such as *object-oriented technology*. This new environment could continue to reduce service time-to-market and support sophisticated new services, such as multimedia services; but it requires processing capacity hundreds of times higher than that of the 1B processor.

Future Processing Architecture. Figure 1 shows a possible 4ESS-switch architecture to meet these needs. Key to this architecture is the interface bus, which provides a high-speed interface directly to the 1B processor, and which is not available on the 1A processor. The interface bus can be connected to new processors and fabrics to support emerging multimedia services.

By off-loading functions from the 1B to be supported by these processors—that is, call processing, signaling, recording, and switch administration—planners estimate that (with enough switch-fabric expansion) the capacity of the 4ESS switch can be expanded several fold.

This architecture allows complex 1B fabric-control and maintenance software (which has been costly to develop) to be retained. At the same time—with a new software environment in new, high-capacity processors—

it allows sophisticated features to be developed rapidly and then offered at high volume. Of great importance, techniques have been developed to allow this transition to take place gradually as more capacity is needed.

Summary

Today, network providers and their customers often take for granted the stored-program control of switching systems—the reliance on software to deliver ever-increasing numbers of sophisticated new features. But this software-based environment, like the rest of the computer industry, demands continuing increases in both real-time processing capacity and memory. The 1B processor is a major, essential step in meeting this need on the 4ESS switch. It will serve as a platform for continuing improvements to the 4ESS switch for many years.

Acknowledgment

The authors wish to acknowledge the valuable contributions of the hundreds of people in AT&T—especially those in the AT&T Network Services Division and the AT&T Network Systems Group—who have contributed to the planning, development, and deployment of the 1B processor.

*Trademarks

Intel is a registered trademark of Intel Corp.
Motorola is a registered trademark of Motorola Inc.

References

1. "Switching Systems," *Engineering and Operations in the Bell System*, Bell Telephone Laboratories, Murray Hill, New Jersey, First Edition, 1977, pp. 225-287.
2. J. W. James and M. Marques, "Beefing Up the Processing Power of the 4ESS Switching System," *AT&T Bell Laboratories Record*, February 1984, pp. 17-20.
3. J. C. Hsu and L. A. Seese, "The AT&T Switching Evolution Challenge," *AT&T Technical Journal*, Vol. 74, No. 3, May/June 1995, pp. 6-9.
4. M. A. Hillis, G. W. Minkus, C. L. DeCaluwe, M. N. Meyers, and E. T. Schafer, "An Improved Approach to Product Quality Through Testing," *AT&T Technical Journal*, Vol. 74, No. 3, May/June 1995, pp. 40-48.
5. G. S. Bullard, S. Golinski, B. D. Morrison, W. T. Myrick, R. W. Neal, Jr., R. L. Rose, P. J. Rutkowski, E. T. Schafer, and W. H. Thurmond, "1B Processor Deployment: Leading the Way to Flawless Execution," *AT&T Technical Journal*, Vol. 74, No. 3, May/June 1995, pp. 61-72.

6. T. W. Anderson, P. D. Carestia, J. H. Foster, and M. N. Meyers, "The Evolution of the 4ESS Switch," *AT&T Technical Journal*, Vol. 73, No. 6, November/December 1994, pp. 93-100.

(Manuscript approved April 1995)

Carl E. Betta is director of the AT&T Network Services



Division Project Planning Center in Holmdel, New Jersey. He is responsible for providing detailed project planning and management of activities in support of the AT&T Communications Services Group business unit's efforts that relate to the new AT&T Wireless Services unit, formerly McCaw Cellular. Mr. Betta holds a B.E.S. degree from Johns Hopkins University in Baltimore, Maryland and an M.S. from Purdue University in West Lafayette, Indiana—both in electrical engineering. He joined AT&T in 1976.

John B. Lewis is a member of technical staff in the Network Evolution Planning Division in the AT&T Network Services Division at Holmdel, New Jersey. He is responsible for switch capacity planning for the AT&T network. Mr. Lewis holds a B.S. degree from Worcester Polytechnic Institute in



Massachusetts, as well as a D.Eng. from Yale University in New Haven, Connecticut—both in electrical engineering. He joined AT&T in 1968.

Larry A. Russell is division manager of Network Evolution Planning in the AT&T Network Services Division at Holmdel, New Jersey. He is responsible for the introduction of new technology and capabilities into the AT&T network. Mr. Russell holds two degrees in electrical engineering—a B.S. from the Georgia Institute of Technology in Atlanta and an M.S. from Stanford University in Palo Alto, California. He joined AT&T in 1978.

