

Conceptual Approaches to Switching in Future Military Networks

GINO J. COVIELLO, MEMBER, IEEE, AND ROBERT E. LYONS, SENIOR MEMBER, IEEE

Abstract—This paper presents conceptual approaches to the synthesis of switching functions in future military networks. The special requirements of military communications are shown to favor the implementation of integrated networks which provide all services (voice, data, imagery) within a single unified network structure. Transparent and transactional methods are defined as contrasting approaches to the realization of the required switching functions. The conceptual approaches to switching in future military networks are presented in the context of cultural differences which appear to exist between the differing perceptions of the adherents of circuit switching and packet switching. Special consideration is devoted to the issue of software engineering and the impact of current trends in DoD procurement policy.

I. INTRODUCTION

SINCE its initial inception in 1961, the Defense Communications System (DCS) has continued to evolve as the common-user switched telecommunications system of the Department of Defense. Initial services were oriented toward voice communications, message traffic, and nonswitched point-to-point connectivity for a spectrum of special needs. In 1980 the Defense Communications Agency (DCA) will also provide a new packet-switched service, offering high-speed data communications for interactive computer users. In the future continuing evolution may provide connectivity for facsimile and imagery users, and an increasing level of service features, such as minimum cost routing, electronic mail and message service and multimedia conferencing.

A common element to each of the above evolutionary steps is an increasing level of complexity and sophistication in the so-called "switching function" aspect of telecommunications system design. Historically, the function of switching, particularly in the traditional circuit-switching concepts has simply involved locating an available path through a complex maze of switching and transmission elements and effecting an interconnection among transmission links to create a continuous path over which electrical signals can flow to provide an end-to-end connection. Today, and particularly in the future, the so-called switching function may encompass a broader scope of activities including, for example, storage, queueing, transformation of information signals from one format to another, and ADP-like processing. Thus, the communications switching machines of the future will become generalized to include functions not previously associated with switching. Some may prefer to refer to these functions as "nodal processing," but in

this paper the more traditional nomenclature of "switching" will still be used.

As plans and designs for future switching functions evolve, the status of emerging technology plays a critical role in determining the degree of complexity and sophistication that can be realistically implemented. The effects of this emerging technology are manifested in two ways:

- 1) the cost of implementing standardized approaches, concepts and functions can be dramatically reduced; and
- 2) the cost versus complexity balance can be so fundamentally altered that entirely new concepts and approaches will arise to replace traditional solutions.

Both of these effects are being demonstrated within current plans for the near-term DCS. For example, existing cost trends for LSI circuitry utilized in processing and switching equipments indicate the potential for simultaneous decreases in cost and increases in capabilities. On the one hand, this effect is allowing DCA to achieve higher survivability in its future networks by increasing the number of switches in the network and distributing them more closely to the user communities [1]. On the other hand, this same effect has allowed the DoD to implement a completely new type of switching network, i.e., the packet-switched AUTODIN II [2]. The objective of this paper is to focus primarily on the second aspect of the technology implications. Specific approaches will be explored to the utilization of technology in a way that could significantly enhance the capability of future implementations of communications switching functions to satisfy military needs and objectives.

II. SWITCHING ALTERNATIVES

To conduct any analysis of switching alternatives on a logical basis it is, of course, useful to delineate the spectrum of available or existing approaches. One way to achieve this objective would be to list, perhaps within a tree structure, all the possible choices. For example, one could start with circuit switching, categorize it as either analog or digital, and divide each of these according to the use of a space- or time-division matrix, or even further into the structure of the matrix (space-time-space, etc.). However, such a procedure would entail a level of detail that would obscure some of the major considerations to be addressed in this paper. It will be sufficient for this discussion simply to define two basic alternatives, namely, circuit and packet switching, and to further restrict our discussions to digital switching. This restriction is based on the assumption that even though analog transmission may be around for a long time in the future, current trends are universally moving towards implementing all switches based on digital designs. The discussions will not center on the final detailed

Manuscript received January 15, 1980.

G. J. Coviello is with the Defense Communications Agency, Code 101C, Washington, DC 20305.

R. E. Lyons is with the Computer Systems Division, Defense Communications Engineering Center, Defense Communications Agency, Reston, VA 22090.

designs of such switches, but rather on the philosophical—one could almost say cultural—approaches to thinking about switching functions within the overall context of general system design.

Recognizing the existence of two completely different switching approaches, circuit and packet switching, is a necessary but not sufficient step in the overall discussion. One also needs to address what relationships should exist between these approaches in meeting the demanding requirements of future needs and services. One fundamental question arises: should one continue to design and build separate circuit- and packet-switched networks, or should they be "integrated" in some sense? There are, as always, pros and cons to either approach. A number of papers have already addressed certain of the issues with respect to integrated networks such as efficiency and cost. However, a number of other issues dealing with operational requirements must be addressed before this issue can be resolved. Such a complex issue could never be settled within the context of a single paper such as this, but it is useful to highlight a few of these considerations to illustrate why the question is still open.

First of all, it is herewith postulated that a decision to build separate circuit- and packet-switching networks cannot be considered viable unless the decision also includes a credible rationale for allocating which telecommunication services should be provided by which network. As an example, consider facsimile service. The preponderance of developments today, commercial as well as military, are oriented towards satisfying this service on circuit-switched networks. Yet, there are a number of technical and operational considerations which would favor its satisfaction through packet switching. Thus, the questions arise as to whether one can get agreement among all communities (e.g., civil, tactical, strategic, NATO) on which network will carry facsimile traffic. Failing this, is the lack of communicability among these communities of sufficient military concern to warrant the additional cost of gateways between the networks in sufficient quantities to satisfy interoperability and survivability considerations? Voice services also present an important issue. Currently all voice is serviced by circuit switching. However, some studies have indicated significant cost advantages by the use of packetized voice [3]–[5]. There are a number of issues that have been raised regarding packetized voice [6], [7], and the viability of this approach is far from decided. If this approach becomes accepted, then the issue of interoperability between circuit-switched and packet-switched networks must likewise be addressed for voice communications.

Still another issue to be raised concerns the survivability of military communications services in a wartime situation. The key to enhancing survivability of telecommunications systems has traditionally focused on providing a high degree of redundancy and alternate routing in the system. Adding extra redundancy to a system over the minimum-cost design is generally necessary to achieve a given level of survivability, and this represents an important design tradeoff. In general, the percentage increase in cost to achieve survivability goes down as the network sizing (number and dispersal of users and total traffic offered) goes up, since the minimum cost design for large networks inherently results in a more distributed (i.e., redun-

dant) network. Thus the integrating of several smaller systems into a single larger system tends to improve the economics of achieving a desired level of survivability. This effect is more pronounced when one system is considerably smaller. Projected traffic for the DCS packet-switched network, AUTODIN II, is considerably smaller than for the circuit-switched network, AUTOVON, and it is highly doubtful whether the same level of survivability could be economically provided for AUTODIN II as compared to AUTOVON if they were designed as completely separate networks.

These considerations all point strongly to the desirability of integrating packet- and circuit-switching requirements into a common design. It must be strongly emphasized that the term "integration" is subject to many different interpretations, ranging from the selection of a single switching approach (e.g., packet or circuit switching) to service all requirements, to the employment of separate switching approaches for the different services but using common transmission facilities to provide connectivities for each subsystem. Several such approaches will be discussed in this paper. The most important conclusion to be drawn from this paper is that there is no single approach to integration that is universally optimum, but that we need to strive for an integrated design approach. What is meant here is that one should never consider designs for circuit or packet switching for voice, data, facsimile, etc., in isolation, but rather in a way that interactions and potentials for integration are fully exploited.

III. CULTURAL DIFFERENCES BETWEEN CIRCUIT AND PACKET SWITCHING

Looking to the long term future, there are a number of potential approaches that could achieve closer integration of circuit- and packet-switching capabilities. Such approaches have been postulated in a number of already published papers [3], [4], [8], [9]. This paper will approach that objective, not by offering yet another technique, but rather by developing certain insights into the fundamental characteristics of circuit and packet switching which, when considered in conjunction with available technology, puts the function of switching into a wholly different light. As will be shown, one of these approaches has the potential of significantly enhancing the interoperability, flexibility, and survivability of future military telecommunications.

As a first step in this development, it is necessary to establish an understanding of what is considered to be the most fundamental distinction between the implementations of current circuit- and packet-switches. This distinction centers on the concepts of *transparent* versus *transactional* flow of information transfer. Although circuit switch designers have traditionally employed what will be defined later as transparent concepts, and packet switch designers have used transactional concepts, it needs to be emphasized that this association has evolved as a result of certain cultural interpretations of the switching functions, and is not necessarily an intrinsic characteristic of the switching approach [10].

Before proceeding to the definition and discussion of the switching approaches in this paper, the reader should be forewarned of certain subtleties in the terms which will be used.

The terms "packet switching" and "circuit switching" are used to broadly describe network concepts which accomplish user-to-user transfer of information. On the other hand, the terms "transactional" and "transparent" are used to describe the basic implementation architectures of the switches existing in the network. In the following paragraphs the traditional circuit-switching technique will be used as an example to illustrate the transparent switching concept, and the traditional packet-switching technique will be used to illustrate the characteristics of transactional switching concepts. However, use of these illustrative examples should not be interpreted as implying that inviolate relationships exist between circuit switching and transparency on the one hand, and between packet switching and transactions on the other. In fact, a major objective of paper is to show that the transactional switch design approach is unique in that it can perform both circuit and packet switching functions, while the transparent approach can *only* handle circuit switching functions. This argument will now be developed in greater detail.

A rather general architecture for a circuit switch is illustrated in Fig. 1. The multiplex/demultiplex box interfaces the wide-band trunks of the transmission facility to the link-by-line terminations on the switch matrix. These lines can, of course, be either analog or digital in nature. The matrix provides, through space- or time-derived crosspoints, the capability to connect one incoming line to an outgoing line and achieves the function of information transfer. Control represents the "intelligence" of the switching function, and includes setting up and tearing down calls, routing, signaling/supervision, and other similar functions.

An inherent property of traditional circuit-switch designs is the complete transparency of the switch matrix, defined as follows. Since it cannot be known *a priori* which of the hundreds of thousands of lines terminating on the matrix will need to be connected during a specific call, the electrical standards and modulation modes of each line must be identical, or else information transfer would not be possible. Clearly, the specifications of "crosspoint" in the matrix must be compatible with those of the lines. To avoid perturbations in the timing characteristics of those lines, it is also clear that any delay in the matrix must be held constant, or at least within very narrow limits. These two characteristics lead to the requirement that the matrix itself must be transparent in terms of the modulated signal. Clearly, switching between dissimilar signals is often required but this is achieved by inserting "conversion boxes" (if such a conversion is possible) external to the matrix itself, to preserve the monolithic sameness of each matrix termination. Analog to PCM represents the most obvious example of converting all external trunks to either an analog signal into an analog matrix or to digital signals into a digital (PCM) matrix.

Having defined and discussed the aspects of transparency as related to common perceptions of circuit switching, let us now turn to a discussion of the transactional aspects of packet switching. First of all, let us examine the typical architecture of a packet switch as illustrated in Fig. 2. The figure indicates a number of processors (usually minicomputers or microprocessors) interconnected through a data bus which allows information to flow between the processors. Typically, the pro-

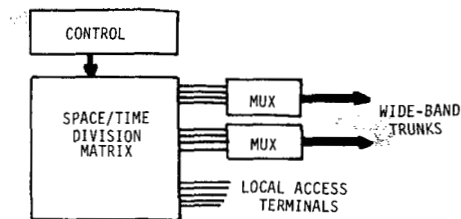


Fig. 1. Generalized circuit switch architecture.

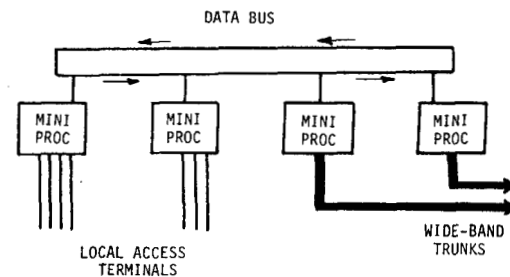


Fig. 2. Generalized packet switch architecture.

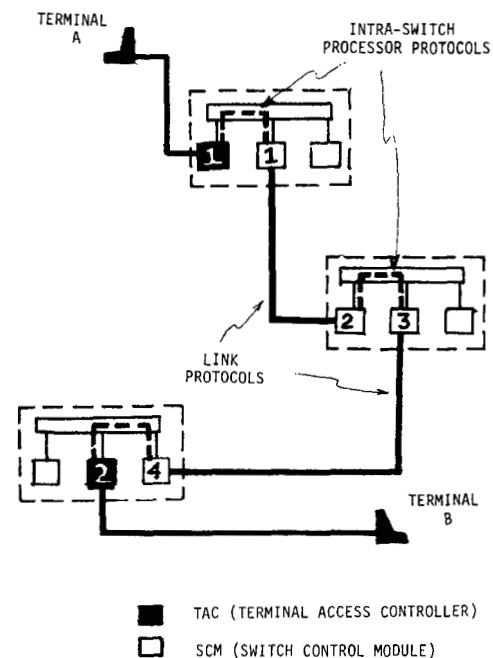


Fig. 3. Packet network structure.

cessors are programmed either to handle a number of single-channel terminals directly to provide local access connectivity, or to handle packets over a wide-band trunk interconnected to another switch. In the AUTODIN II system, for example, the local access processor is referred to as a TAC (Terminal Access Controller) and the trunk processor is referred to as an SCM (Switch Control Module) [2]. The TAC and SCM software-provided functions are both implemented on identical processor hardware, namely, the PDP-11/70. Let us now look at the operation of an illustrative end-to-end connection as shown in Fig. 3, tracing the information flow from terminal A to terminal B through the network. Terminal A may operate into TAC 1 either on a synchronous or asynchronous basis and could use a variety of standards for data transfer, e.g., ASCII characters. As opposed to circuit switching, these data do not con-

tinue to flow through the network uninterrupted, but rather they are transferred through a series of data blocks or packet exchanges which will be called "transactions." A basic definition of such a transaction between two terminals/processors includes not only the transfer of "information bits," but also transfer of signaling, control and addressing bits sufficient to define completely how the "sender" wants the "receiver" to handle each transaction. Clearly the ground rules of this data transfer must be understood without ambiguity by both ends of the communication. The rules for handling such transactions are commonly referred to as "protocols." Hence, terminal *A* and TAC 1 converse within a specific predetermined protocol. This information is then transferred to the SCM over the data bus in another "transaction" using another protocol. Since this transfer occurs within a single switch, and has no interaction with other elements of the network, the protocols governing this transaction are generally not published nor known outside of the switch design community, but they do indeed exist. These procedures continue throughout the network until the data are delivered to terminal *B*. Seven different transactions are required to complete the end-to-end transfer, namely:

- 1) Terminal *A*-TAC 1
- 2) TAC 1-SCM 1 (internal)
- 3) SCM 1-SCM 2
- 4) SCM 2-SCM 3 (internal)
- 5) SCM 3-SCM 4
- 6) SCM 4-TAC 2 (internal)
- 7) TAC 2-Terminal *B*.

It is important to highlight here that one could potentially utilize seven different sets of transfer protocols in this end-to-end transfer of raw information bits. It should be recognized that in order for terminal *B* to "understand" the message intended by terminal *A* there must be additional protocols that take care of these requirements: hence, some control bits need to be transferred between terminals *A* and *B*. However, these bits appear as information bits to the system and are not even recognized by the switch protocols. Our concern in this paper will be restricted only to the network data transfer protocols, although the reader should be aware that an entire hierarchy of protocols has been defined in relation to data communication [1]:

The most significant impact of the transactional over the transparent approach is not so much on the hardware and switch designs but rather on the manner in which one thinks of "switching." In essence, there exists a real cultural difference in the two approaches. The transparent (circuit-switched) network designer thinks primarily of maintaining an orderly end-to-end, i.e., terminal-to-terminal flow of information, and therefore strives to maintain a uniformity of standards throughout all trunks and switches in his whole network towards that end. On the other hand, the transactional (packet-switched) network designer can focus on the "parts" of the network (with a high degree of freedom and independence) as a method of achieving the "whole." For example, in Fig. 3, the protocols for the SCM 1-SCM 2 transfer could be quite different from those of the SCM 3-SCM 4 transfer without hindering the network performance. In fact, a single switch could simultaneously

be operating with four or five different protocols to other switches almost as easily as if all switches used the same protocol. Clearly, one still needs some semblance of end-to-end connectivity to ensure terminal *A* can talk to terminal *B*, but these particular protocols are implemented only at TAC 1-terminal *A* and TAC 2-terminal *B* interfaces and are "invisible" inside the network. This ability to solve problems on a piece-by-piece basis rather than on a total system basis affords the transactional approach some very interesting advantages in terms of flexibility, interoperability and survivability.

The reader has perhaps noticed that the use of the transactional approach to communications inherently involves a decentralization of control to the user. This is because some of the functions of communications which are provided within the switch hardware for transparent systems are assigned to terminal protocols for transactional systems. This decentralization creates a special problem for military systems: interoperability requires strict adherence to protocol standards among the interoperating community. Accordingly, the Department of Defense has established procedures for developing standard protocols and for enforcing compliance among network users. These standards apply to the specification of protocol functions and the software which implements them; it is not necessary to standardize on hardware. Actually, the ability to utilize different hardware implementations *can* be an advantage.

IV. THE DOMINANCE OF SOFTWARE

Another technological change which will have a profound impact on the design of future DoD telecommunications systems is the emergence of software as a dominant cost element. The orders-of-magnitude improvement in hardware performance per unit cost over the past decade or two has not been matched by an improvement in software performance per unit cost. Despite the recent establishment of software engineering as a viable technical discipline, software remains more of an art than a science. Software development and maintenance continue to be labor-intensive activities whose costs continue to rise with time even while hardware costs are falling dramatically. In April 1979, the Deputy Under Secretary of Defense for Research and Advanced Technology made the following statement before the Subcommittee on Research and Development of the House Committee on Armed Services: "It has been estimated that the DoD spends more than \$3 billion a year on computer software in direct costs only; this excludes payroll cost and general-purpose software which is estimated to add another \$1 to \$1.5 billion ... software development costs (are) now nearly 90 percent of total computer system costs" [12].

The Department of Defense, in recognition of the dominance of software costs over hardware costs for many major defense systems acquisitions, has revamped acquisition policies to achieve better control of software costs. Among these policy initiatives are the requirement to use standard high-order languages for major software projects [13], the attempt to develop a single high-order language specially designed for the requirements of embedded computing applications [14], and the attempt to develop a standard instruction set architecture for the computers to be used in future embedded applications. (An "embedded" computer is one which serves a

particular function within a larger defense system, in contrast to a general-purpose computer used for many generic functions in an office environment (such as payroll or scientific computing). A computer used as a switching machine at a communications node is an excellent example of an embedded computer application.)

If these DoD policy initiatives are successful, future switching machines will be acquired using a "software-first" procurement approach. The procurement environment will include a broad class of computers, generally of the minicomputer or microprocessor variety, all of which will be capable of executing the standard instruction set and the standard high-order language(s) efficiently. Major defense embedded computer acquisitions, including those to perform nodal switching functions, will be made competitively to a specification which will include a trusted software design provided in a standard high-order language and based on modern top-down structured programming design methods. The benefits of trusted software are most important for DoD applications, for the use of software verification techniques can greatly increase the reliability of software. Extreme reliability of software is required in a military environment where security must be provided. This point is developed further in a paper by Stillman and Defoire provided elsewhere in this issue [15].

Within the Defense Department, a High-Order Language Working Group was established to investigate the use of standard high-order languages as a means to bring the high cost of software under control. This group recommended to DoD management that the use of a single standard high-order language for DoD embedded computer applications is feasible and ought to be pursued. Further investigation developed the basic requirements for the standard language and determined that no then-existing language was adequate to meet all the requirements. Accordingly, contracts were awarded which led to the development of specifications for a new standard high-order language which utilized Pascal as a basis. The new language, now known as "Ada," was specified in detail in April of 1979 [14]. Following a year of shakedown, final detailed specifications were provided in May of 1980. In the meantime, many test applications have been coded in Ada and run by the use of rudimentary Ada translators. Work is currently under way to provide highly efficient Ada compilers on a wide variety of hardware elements normally used for DoD embedded computer applications.

It is not certain that the development of Ada will succeed in providing a single universal language for DoD embedded computer applications. That goal is extremely ambitious, for the scope of embedded computer applications is very broad indeed. Yet, it is clear that Ada will profoundly influence the methods of developing software and procuring hardware for future DoD embedded computer systems. Accordingly, work is under way to develop the software tools necessary to structure a total software development and maintenance environment based on Ada. The cost and complexity of this environment for development and maintenance provides additional motivation to the development of a single language concept, and, consequently, a single environment. In addition to the compiler and environment efforts, work is also under way to

investigate methods for validating the correctness of Ada programs. This work goes beyond the realm of the U.S. Department of Defense; Ada has all the attributes of an international standard high-order language for systems software development as well. There has been strong participation in the Ada effort by representatives of NATO countries and by the private sectors within these countries.

The emergence of new software engineering methods such as structured programming techniques and a standard high-order language is extremely timely for the synthesis of switching functions for future defense communications systems. These methods are most applicable to the transactional techniques discussed in the previous section. The implementation of transactional systems has traditionally been based on the use of standard, garden-variety computer hardware, programmed to perform its required functions in software modules which reduce to practice the protocols which define system performance. The methods of modular software design, top-down structured programming, software specification and validation using standard high-order languages, and software-first procurement are all highly relevant to the transactional approach. In contrast, the transparent approach has traditionally relied on special hardware development. Standardization of function and software design appears sufficient to ensure interoperability among transactional systems; for transparent systems, standardization at the hardware level appears to be a much more difficult goal to achieve.

In a word, the systems design of a transactional system is embodied in the specification of standard protocols. We are learning how to provide these specifications in a more efficient and flexible manner, as we learn how to manage software-dominated systems.

V. FAR-TERM APPROACHES TO INTEGRATION

In the longer range, many opportunities to exploit the integration of digital telecommunications services within a single switch are available. These opportunities are based on the merging of two technical disciplines, telecommunications engineering and computer science. Two fundamental technological changes having the potential to revolutionize future telecommunications systems have been discussed in the previous two sections: the emergence of a transactional (packet-oriented) communications method to complement the traditional transparent (circuit-oriented) method, and the emergence of software as the dominant cost element in the synthesis of nodal switching functions. This section provides a far-term approach to telecommunications systems integration which is in harmony with these two technological changes.

A total (integrated) communications service for the DoD should be flexible enough to provide both circuit- and packet-switched services. Fortunately, it is possible to overlay a transactional service on a transparent base and vice-versa. The configuration of Fig. 4 is, in fact, an example of a fundamentally transparent (circuit-switched) service which carries packet (transactional) traffic as an overlay. The other approach to providing a fundamentally transaction-oriented service with a transparent overlay is also possible. A virtual-circuit type packet-switched protocol is an example, wherein the terminal

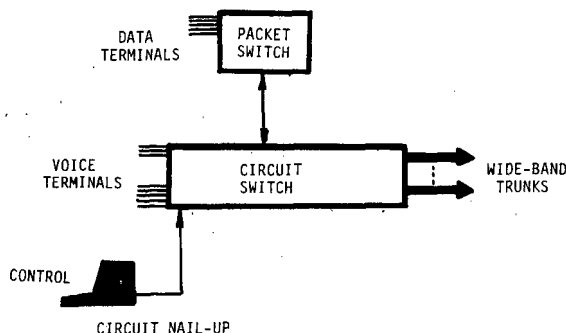


Fig. 4. Integrated switching facility.

devices, connected to a packet-switched network using a CCITT standard X.25 protocol [16], appear to be communicating on a transparent basis, although the underlying packet transport system is a transactional datagram service. The concept of using a transactional based system to perform so-called circuit switching functions appears to be a difficult one to accept by either the circuit- or packet-switching community. It is believed that much of this reluctance may be due to the historical association of packet switching with asynchronous transmission. But there is no fundamental reason for ruling out a synchronous transactional structure as a possibility. For illustration consider a standard TDM-T1 multiplex structure running at 8000 frames/s, each frame consisting of 8-bit samples from 24 voice channels plus 1 frame bit, yielding a 193-bit frame. If each end of such a T1 link were terminated onto a microprocessor one could easily envision such a link as performing synchronous file transfers of sequential frames, each of which contains 193 bits. Each such frame could contain a certain number of bits dedicated to protocols, bits for synchronous traffic (circuit-switched), and/or some bits dedicated to transfer of asynchronous packet-like traffic. This illustration shows that both circuit- and packet-switching can be performed within a transactional structure. The switch in this case, would be synthesized in a manner very similar to the packet switch illustrated in Fig. 2 and would not require matrix hardware. The real-time clock, built into any mini- or microprocessor, would be used to maintain synchronism of the transmission transfers. Viewed from outside the switch, one would not really be able to distinguish whether the switch was based on a transactional design or a more traditional (transparent) design. Although this approach is feasible and has been proposed earlier [9], the major question arises as to what advantages exist as a result of the transactional design. These will be discussed in the following paragraphs.

In the area of procurement, this approach accomplishes the following.

- 1) It allows common hardware to be utilized for both packet and circuit switching;
- 2) it permits an entire multiprocessor architecture (including processors and busses) to be purchased commercially off-the-shelf in open competition; and
- 3) it provides natural modularity in terms of processors procured, permitting continual upgrading by the addition of new processors, as needed, utilizing the then-current pricing and technology.

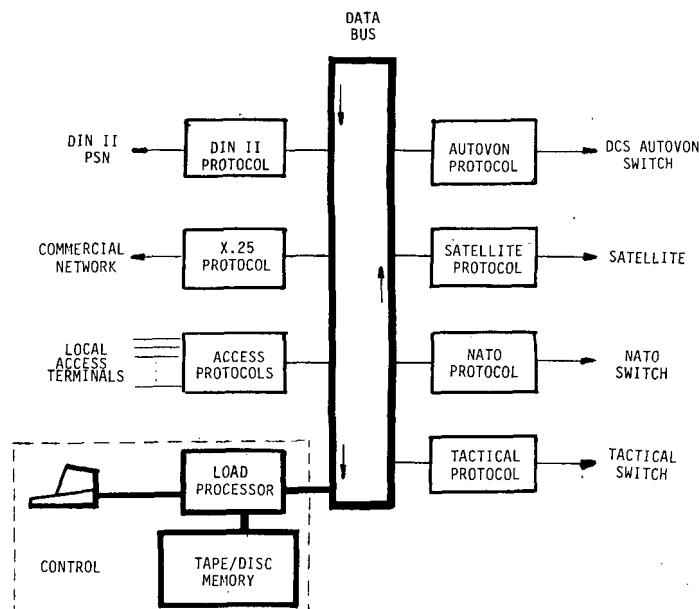


Fig. 5. Proposed node configuration.

With respect to achieving interoperability with other systems, the transactional approach has a major advantage over the transparent one in that dissimilar systems can be terminated on separate processors, each of which has been individually programmed to match that system as shown in Fig. 5. Internal translations between any two systems' processors can also be programmed and located in either processor. Hence, the typical nodal configuration shown in Fig. 5, which can interoperate with several different networks, is no harder to procure than a node which operates within a single system. What is needed, of course, is a separate software program to implement the protocols for each system to which an interface is desired; but once this software is written, it can be duplicated, transferred, and loaded onto any compatible processor in the system.

Clearly, the additional burden of translation between systems will require somewhat more powerful processing capability and will contribute to somewhat higher cost. However, one major point that should be made here is that although interoperability does require the additional programs to be written, it does not require special black-box hardware for each different network—all the hardware can be identical.

The transactional approach allows a separation between hardware and software. It is really the software that decides whether a processor will operate as a packet switch or as a circuit switch, or whether it will perform some other communications processing function. A second, less obvious point is that one could:

- 1) install a DCS site with a multiprocessor architecture without precommitment of its function;
- 2) include in the site a tape or disk drive which contains all the modular software for any switching functions that might be desired; and
- 3) provide for loading the functional switching structure and changing it at will within minutes by loading a new program into the appropriate processor.

Such an approach makes restoral actions much simpler, since each facility can now store a variety of software interface packages on tape instead of storing a variety of hardware black boxes for contingencies. Changeovers can also be effected quite rapidly from the operator's console by reading in a new tape in minutes instead of the hours or days that might be required to physically install and wire black boxes. It would even be possible to "install" a new function such as a packet switching nodal function, at a location where it did not exist previously. Thus, a functional mobility might be achieved from such an approach at a cost which could not even be approached by associating each switching or concentrating function with specific hardware.

Clearly, restoral and reconfiguration will not be as easy as the "genie-in-the-bottle" approach suggested here. Yet, in spite of this, the capability to approach such a goal is real, and, because these capabilities are essential for military communications, efforts toward achieving as many of these goals as possible should be included in DoD's future switching development efforts.

VI. NEAR-TERM APPROACH TO INTEGRATION

The concepts presented in this paper are clearly oriented toward the far-term future. A natural question arises as to whether any useful degree of integration is feasible in the near term. In this regard, it is interesting to note that current trends in the commercial digital switching field are moving in a direction that not only facilitates but actually promotes a higher degree of integrated design, particularly with respect to the use of common transmission facilities. A key feature of modern switches is the complete integration of time-division multiplexing (TDM) as an integral part of the switching internal hardware in contrast to the more traditional frequency division multiplexing (FDM). The impact of this feature will be discussed in the following paragraphs.

To achieve economy of scale, transmission facilities are generally structured on a multiplexed (FDM or TDM) hierarchy basis, while traditional circuit switches operate on a line-by-line (demultiplexed) basis. The interface between transmission and switching is generally provided by the tech control facility as illustrated in Fig. 6, consisting of two major elements, a patch panel and multiplexers. Together these elements have provided the means to assign individual demultiplexed lines manually to the circuit switch or data switch as applicable, and in many cases to patch over from one incoming trunk group to another outgoing trunk group to serve those dedicated communications that do not require any switching at all. Flexibility in adding or removing capacity and rerouting to circumvent trunks which have become degraded or destroyed is completely achievable at the patch panel, but is generally slow and cumbersome due to its inherently manual operation. Thus, it is seen that this current approach as implemented in the DCS already has a high degree of integration with respect to transmission facilities. In modern digital switch design, there is an economic advantage in performing time-division switching directly at the trunk-group level. Hence the functions of multiplexing and demultiplexing are inherently included within the scope of the switching function. A major impact of this design ap-

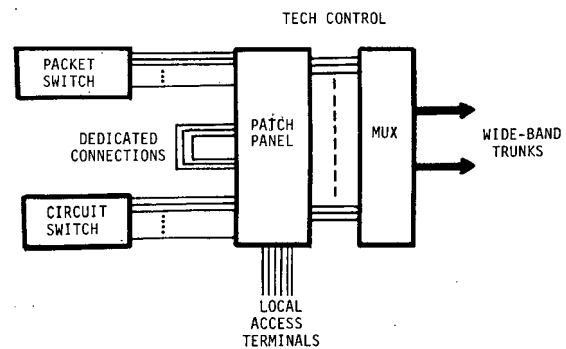


Fig. 6. Typical node configuration.

proach is the elimination of the necessity for separate tech control hardware, since it can now be provided by the switch itself.

A major design question for the future DCS is, to what extent should external tech control hardware be retained?

One option is to retain the separate functions in their entirety by including additional multiplexers and demultiplexers to break down the trunk groups coming out of the switch into the individualized channels appearing at the patch panel. This option has two significant drawbacks: the ability to reconfigure the trunk groups retains the slow and cumbersome operation inherent in the manual procedures, and the potential for manpower reduction associated with tech control automation is not fully achieved.

Another approach is to interconnect data switches and dedicated services through the circuit switch itself, thereby eliminating most of the external tech control functions. This approach is illustrated in Fig. 4. In this approach the patching functions can be performed by holding a connection indefinitely within the switching matrix, under command from the controller console. Under direction of this same controller, rapid restoral of degraded trunks can be achieved through semi-automated procedures at significantly greater speed. A drawback of this option is that the circuit switch sizing must be increased to handle the additional lines for data and dedicated services. However, with decreasing cost trends for switches, this added cost is likely to be substantially less than the cost of the tech control facility. The architecture of Fig. 4 maintains a high degree of system integration in the transmission facility and provides a rapid and flexible capability to alter the mix of different services with respect to their share of total capacity. A significant advantage of this approach is the fact that data communications and dedicated connections, which often represent very high priority services, are afforded a high degree of survivability since they can be restored rapidly.

Since the option of Fig. 4 is essentially available today with current technology, the near-term evolution of the DCS will attempt to exploit this level of integrated services.

VII. CONCLUSIONS

As the spectrum of communication services continues to expand, i.e., voice, data, facsimile, etc., the capability for integration of all services into a single network (at least at the transmission and control level) is important in order to maximize survivability of essential services. To address issues of in-

tegration, definitions and descriptions of transactional versus transparent switching concepts have been postulated. It has been shown that integration by overlaying of transactional services onto a transparent system is relatively straightforward and achievable in the near timeframe. This approach can significantly enhance the survivability of military networks by providing rapid alternative routing and restoration of damaged facilities, but does not, of itself, improve interoperability with other networks. Integration by overlaying transparent services onto a basically transactional network, on the other hand, can provide a high level of flexibility and interoperability with other networks. Although this approach requires considerable investigation to assess its feasibility, the potential flexibility it affords in dynamically reconfiguring future networks should provide the DoD with significant future benefits. Accordingly, research programs should study the feasibility of integrating all communications services within a single transactional network, so that the benefits of integrated communications can be enjoyed at an early date.

ACKNOWLEDGMENT

The authors gratefully acknowledge the many suggestions for improvement of this paper which they received from their colleagues at the Defense Communications Agency.

REFERENCES

- [1] T. M. Shimabukuro, "The next generation AUTOVON," this issue, pp. 1460-1466.
- [2] R. E. Lyons, "A total AUTODIN system architecture," this issue, pp. 1467-1471.
- [3] I. Gitman *et al.*, "Sensitivity of integrated voice and data networks to traffic and design variables," in *Conf. Rec. 6th Data Commun. Symp.*, Monterey, CA, Nov. 1979.
- [4] J. Gruber, "Delay related issues in integrated voice and data transmission," in *Conf. Rec. 6th Data Commun. Symp.*, Monterey, CA, 1979.
- [5] R. Fossum and V. Cerf, "Communications challenges for the 80's," *Signal*, Oct. 1979.
- [6] G. Coviello, O. Lake, and R. Redinbo, "System design implications of packetized voice," in *Conf. Rec. Int. Comput. Conf.*, Chicago, IL, June 1976.
- [7] G. Coviello, "Comparative discussion of circuit versus packet switched voice," *IEEE Trans. Commun.*, vol. COM-27, Aug. 1979.
- [8] K. Kummerle, "Multiplexor performance for integrated line and packet-switched traffic," in *Conf. Rec. ICCS 1974*, Stockholm, Sweden.
- [9] G. Coviello and P. Vena, "Integration of circuit/packet switching by SENET," in *Conf. Rec. 1975 Nat. Telecommun. Conf.*, New Orleans, LA.
- [10] I. Lebow, "Communications and computers—Two cultures or one?" *Signal*, Mar. 1980.
- [11] International Organization for Standardization, Reference Model of Open Systems Interconnection, Version 4, June 1979.
- [12] R. Davis (Deputy Under Secretary of Defense), "Statement on computer science before Subcommittee on Research and Development, Armed Services Committee," U.S. House of Representatives, 96th Congress, 1st Session, Apr. 5, 1979.
- [13] "Management of computer resources in major defense systems," DoD Directive 5000.29, Apr. 26, 1976, Section V-G.
- [14] "Preliminary Ada reference manual and rationale for the design of the Ada programming language," *SIGPLAN Notices*, vol. 14, no. 6, Parts A and B, June 1979.
- [15] R. Stillman and C. Defiore, "Technical issues in military data communications networks," this issue, pp. 1472-1477.
- [16] CCITT, Revised Recommendation X.25: Interface Between Data Terminal Equipment (DTE) and Data Circuit-Terminating Equipment (DCE) for Terminals Operating in the Packet Mode on Public Data Networks, Nov. 1979.

★

Gino J. Coviello (M'54), for a photograph and biography, see this issue, page 1443.

★

Robert E. Lyons (A'53-M'57-SM'66), for a photograph and biography, see this issue, page 1471.