IBM’s LU6.2: Implications for the future of corporate distributed processing

by BONNIE M. WEISS
Systems Strategies, Inc.
New York, New York

ABSTRACT

Logical Unit 6.2, along with the related Physical Unit 2.1, are enhancements to IBM’s Systems Network Architecture (SNA) that promise to revolutionize data communications in distributed processing environments. IBM is positioning LU6.2 as a converged solution for corporate distributed processing, and is gradually incorporating LU6.2 support into virtually all of its major products. This paper examines the phenomenon of LU6.2 and some of its likely effects on office information system communications and configurations. In light of the de facto standard nature of SNA and the fact that most IBM competitors have announced or pledged LU6.2 support, LU6.2 is viewed as not only an IBM communications architecture, but as the basis for integrated multi-vendor networks.
INTRODUCTION

Until recently, corporate communications networks have relied upon a mainframe-based central control structure to manage all data distribution functions throughout the network. However, the tremendous influx in recent years of departmental minicomputers and desktop microcomputers into office information systems has created a distributed processing environment, often consisting of widely disparate systems.

Since it involves communication between multiple processors, distributed processing is intimately related to connectivity; the ability to connect systems to each other to satisfy application processing requirements. Ease of connectivity is therefore the paramount concern of communications management when configuring a distributed processing environment. Related concerns include the ability to obtain maximum functionality, ease of use, and ease of expandibility and reconfiguration.

In the ideal distributed processing environment, the effort involved in achieving connectivity should be limited to the design and implementation of the applications to perform the required functions. The underlying communications architecture should be automatically compatible for interconnection purposes.

One approach to achieving this connectivity goal has been taken by IBM Corporation through enhancements to its proprietary communications architecture, Systems Network Architecture (SNA). These enhancements are provided by the strategic Logical Unit (LU) type known as LU6.2. The marketing term given to LU6.2 by IBM is Advanced Program-to-Program Communications (APPC).

BACKGROUND: OVERVIEW OF SYSTEMS NETWORK ARCHITECTURE

As all readers may not be familiar with SNA, it is pertinent at this point to provide a brief overview in order to give the proper historical perspective, as well as to elucidate the concepts needed for a basic understanding of LU6.2 before describing its features, benefits and implications.

SNA: Theory and Structure

IBM's master plan

SNA is IBM's master plan for communications among its products. It defines the structure, formats, rules, and controls for transmitting data through networks, and for managing and operating the networks. SNA was originally conceived in 1974 to provide resource-shared communications functions between mainframe computers and peripherals. Resource sharing was designed to reduce the cost of dedicated devices, transmission lines, and other equipment, by allowing different program applications to use the same facilities at different times. SNA continues to be a single strategic architecture that is constantly evolving to accommodate new technology and market demands.

De facto standard

IBM and IBM plug-compatible systems currently account for over 90 percent of the U.S. mainframe computer market, as well as large and growing shares of most other information processing markets. It is estimated that between 70 and 80 percent of major U.S. corporations implement SNA; this base is growing. SNA is currently regarded as a de facto standard for data communications in the United States. This situation is unlikely to change in the near future.

Seven-layer definition

SNA is structured as a seven-layer architecture. The layers and their functions are depicted in Figure 1. A layered architecture divides the communications process functionally; each layer performs specific functions to pass a message between two end points in the network. A message passes through all the layers from the top down in the sending device or node and then back up the layers in the reverse order on the receiving end. In certain cases a message may pass through some of the lower layers in both directions, each time it encounters an intermediate node.

The layered nature of SNA allows a great deal of flexibility and has facilitated SNA's evolution over the years as technology advances. For instance, it is possible to alter the communications process at one layer without affecting the others, as long as the way information is passed to and from the altered layer and its adjacent layers remains intact.

This flexibility has allowed IBM to slowly advance standardizations from the bottom layers upward. Upon its introduction, SNA immediately standardized the level just above Physical (Data Link) with its SDLC (Synchronous Data Link Control) protocol. LU6.2 is now standardizing the upper layers.

Relevant SNA Concepts

Network addressable units

In an SNA implementation, special program code segments called Network Addressable Units (NAUs) are used to repre-
Logical units

Logical Units represent end users to the network. An end user may be either an operator at a device or an application program. Multiple LUs may reside in one node; quantity depends on the type and function of the Physical Unit. LUs provide the interface through which end users gain access to network resources and manage information transmission between end users.

LU-LU sessions

LUs allow end users to communicate by establishing sessions. A session is a logical, two-way connection between two NAUs over a specific link for a specific period of time. Several types of sessions occur within SNA. This discussion is limited to LU-LU sessions.

There are currently seven LU-LU session types, and seven corresponding LU types, defined within SNA. LU and LU-LU session types are defined by the nature of the services they provide to their programs. Until recently, the definition of an LU type was also intimately related to where (i.e., in which type of device) it resided.

Two LUs may communicate with one another via an LU-LU session only if they are of the same type. This point is important to note here, as is the definition of LU Type 6 (LU6.0), a session between two application programs.

OVERVIEW OF LU6.2

LU6.2 is the strategic LU type designed by IBM as the basis for a converged solution for corporate distributed processing. LU6.2 and the related Physical Unit Type 2.1 (PU2.1) are designed to standardize all the SNA levels of a system below the application (user) level, thereby providing complete compatibility for inter-connection purposes at those levels.

A derivative of LU6.0, LU6.2 differs from the former and all other prior LU types in that it is conceived as a single, product-independent LU type. It provides a direct program-to-program interface between application programs residing on different processors. It therefore provides a base for implementing communications across a broad range of product types.

LU6.2 is supported by several PU types. Among them is the new type known as PU2.1. As LU6.2 is a derivative of LU6.0, PU2.1 is an extension of PU2.0. PU2.1 is designed to support the enhanced capabilities of LU6.2. It possesses superior capabilities over PU2.0, which give it extended connectivity ability. There are two aspects to this extended connectivity.

First, PU2.1 can connect a node to other network nodes in two ways. It can link to a mainframe in a hierarchical manner. Also, most significantly, it can connect to another PU2.1 node in a peer-to-peer relationship. The significance of this is that remote intelligent nodes, or peripheral nodes, can use PU2.1 to connect to one another directly, without mainframe intervention.

Second, PU2.1 allows multiple links, as well as parallel...
session support, an improvement in resource sharing and efficiency.

MAJOR LU6.2 FEATURES

The salient features of the LU6.2 architecture and their major benefits are:

Conversations

LU6.2 provides a significant improvement in resource sharing through the use of conversations. Two transaction programs communicate via a conversation, using a session between their associated Type 6.2 LUs to exchange data. Conversations use time-sliced session segments to share the communications link, creating a very efficient use of the session resource (see Figure 2).

LU6.2 provides for two types of conversations: basic and mapped. Basic conversations are implemented on all LU6.2 products, providing a basic universal interface for communications among them.

Mapped conversations are optional; they are intended for use by products that provide an interface for user-written application programs to communicate with one another. Mapped conversations provide a simpler interface for such programs than basic conversations.

The Protocol Boundary

LU6.2 provides a standardized interface to the SNA network for use by application programs, called the Protocol Boundary. The Protocol Boundary is rigidly defined and specified by the LU6.2 verbs. The LU6.2 verbs constitute a generic Application Program Interface (API) that facilitates a programmer's task when designing distributed transactions involving different product types. This API also provides a common specification for hardware designers who want to implement LU6.2 on their products. Through use of this approach, LU6.2's product-independent nature is supported.

Parallel Sessions

LU6.2 provides parallel session capability to allow many pairs of transaction programs in a distributed processing system to connect simultaneously. Parallel sessions allow multiple sessions to exist concurrently between LUs, facilitating more efficient use of network resources and increasing system throughput.

Primary LU Capability

In order for application programs to communicate without mainframe intervention, both ends of the session must be capable of initiation. In SNA this responsibility lies with the primary LU; every LU6.2 implementation can assume either the primary or secondary role in any given session. This supports the peer-to-peer nature of LU6.2 communications.

Commitment Control

Commitment control involves the ability to synchronize transactions across a network (i.e., to insure that changes are committed to all appropriate resources). In LU6.2 terminology, this capability is called syncpoint. Syncpoint is the highest level of resource synchronization defined by LU6.2; it also provides error-protection recovery services, or rollback support.

ARCHITECTED APPLICATIONS

LU6.2 is the keystone in IBM's long-term office systems communication strategy. It has built-in support for a series of architected applications to be implemented at the transaction services level. These currently include Document Interchange Architecture (DIA), Document Content Architecture (DCA), SNA Distribution Services (SNADS) and Distributed Data Management (DDM). New developments from IBM in this area can also be expected.

Document Interchange Architecture/Document Content Architecture

DIA and DCA have been developed to overcome the differing commands among diverse operating systems. DIA allows the interchange of documents and other information across a network. Transmitted documents can be in final or revisable form, and can be directed to multiple destinations. DIA also provides access to the processing and distribution services of the Distributed Office Support System (DISOSS).

DCA defines uniform formatting of documents to be interchanged in an office environment, providing document compatibility across the products that support DCA. Formatting controls are included in DCA, including such functions as pagination, highlighting, heading, and centering. As with DIA, documents can be either in draft or final form.

Distributed Office Support System

DISOSS is an application subset residing in the host that stores, retrieves, and distributes documents created by IBM.
products that support LU6.2. Examples of these include the 5520 Administrative System, Scanmaster 1, and Displaywriter. DISOSS allows remote users to access host services, such as the host library.

SNA Distribution Services

SNADS is an architecture for asynchronous distribution of information between users. SNADS provides delayed delivery services, allowing information to be forwarded through the network as paths between intermediate nodes become available. This eliminates the need for a complete end-to-end session between the origin and the destination of a transmission.

Distributed Data Management

DDM is the most recently announced architected LU6.2 application. It provides data connectivity for record-oriented files residing on systems that support it. With DDM, System/36 and System/38 users can access such files remotely. The files may reside on a remote System/36 or System/38, or in CICS/VS on the host. Some examples of DDM functions include copying a remote file onto a local file, accessing a remote keyed file as if it were local to read, write, update, or delete records, and reading a remote sequential file.

Low-Entry Networking

Before proceeding to a discussion of the implications of LU6.2, it is highly pertinent at this point to mention a most significant extension to PU2.1, formally unveiled by IBM in mid-June 1986, amidst a flurry of announcements. This extension, given the marketing term Low-Entry Networking (LEN), exploits LU6.2 by enhancing PU2.1’s capabilities. Under its original definition, PU2.1 allows an intelligent peripheral node to connect directly to an adjacent intelligent node, without mainframe intervention. The limitation is that the peer-to-peer connectivity does not extend beyond the adjacent link stations.

LEN extends PU2.1’s capabilities to allow PU2.1 nodes to handle intermediate routing of sessions not intended for themselves. LEN allows the configuration of LU6.2 networks consisting of interconnected systems of widely differing sizes in an arbitrary topology. True peer-to-peer networking is now possible. LEN also provides for dynamic routing within the network, as well as dynamic reconfiguration.

LU6.2: USER’S PERSPECTIVE

True Distributed Processing

LU6.2 promises many benefits to users, both in the short and long term. It will rid intelligent workstations of their current SNA identity crisis; they will no longer need to impersonate 3270 terminals to communicate on the network. Therefore, users can begin to realize the full processing power of their ubiquitous PCs. LU6.2 will allow microcomputers to conduct work sessions in real-time with the host, as well as with all other network systems, while retaining full standalone processing capabilities. When the session involves a link between two intelligent devices other than the mainframe, no host intervention is required. The result will be a net gain in overall system efficiency: improved throughput, more usable computing power, no dormant excess processing power, and more effective handling of peaks. In short, true distributed processing.

Programmability

The definition of the SNA upper layers and the standard program-to-program interface provided by the LU6.2 verbs will result in the “decoupling” of programs and devices. Program-to-program communications become independent of the environments (i.e., operating system, programming language, hardware type) of the individual programs. For example, a “C” language program running on a UNIX-based system can communicate with a COBOL program on an IBM MVS machine. The language, operating system, and physical location of the program are all transparent to the programmer and user. The verbs and syntax specified by LU6.2 will provide a universal “language” for user-written programs. It will be much easier for users to configure and maintain large networks and to write distributed application programs for those networks.

Expandibility

LU6.2 will offer the same lasting use of distributed transaction processing programs that IBM’s 360 operating system environment provided for batch processing programs. A capital investment in software, therefore, will be protected longer, and the cost of software maintenance support reduced.

LU6.2 will provide increased flexibility in distributing work across networks. Applications can be written for single or multi-machine environments, and value-added utility programs can be generated for many configurations. This will make expandable solutions easier to come by, and less costly.

Lower Design Costs

LU6.2 will eventually formalize the rules for creating new distributed systems. This will make everyone’s life simpler, as every system designer will not have to try to “reinvent the wheel.” This will lead to a decrease in cost for designing highly specialized systems, as less expertise will be required.

Configurability

The proliferation of PU2.1 nodes will greatly reduce the amount of host communications software required in LU6.2 network implementations. This is because many previously centralized control functions will be offloaded to the remote intelligent nodes, thus relieving the mainframe of some of the responsibility for communications control.
For example, in a traditional hierarchical network, centralized network control programs operating under ACF/NCP (Advanced Communication Function/Network Control Program) keep tabs on the actual physical location of every LU on the network. Therefore, if a user moves his terminal, an NCP regeneration is necessary. In an LU6.2 implementation such as a token-ring LAN, the NCP sees the virtual Logical Units but is transparent to their physical placement on the LAN. In such configuration, users will be able to move PC's as easily as one moves modular-plug telephones today. Eventually such LANs will have extremely powerful distributed processing capabilities. Figure 3 depicts just a few of the concurrent sessions which any PC on such a LAN may one day pick and choose from. Bridges between major data bases such as Cullinet's IDMS/DB and IBM's IMS/DB are possible within the same node in this context. The possibilities are many orders of magnitude more than was previously feasible (see Figure 3).

CURRENT STATUS AND LIMITATIONS

The LU6.2 future holds a great deal of promise. However, at the present time, LU6.2 is a technology in its infancy. Much development and implementation will be necessary before LU6.2 networks become a working reality in user sites. There is, however, much evidence that IBM is devoting a fair-sized chunk of its massive resources to get LU6.2 into the marketplace.

Hardware Support

As of this writing, IBM has announced LU6.2 support for CICS/VS, the System/36 and System/38, the System/88, Series 1, the 8100, and the IBM PC family, and has issued a statement of direction indicating future LU6.2 support for the 4700. IBM has also announced support for the 5520, Scanner 1, and Displaywriter under its DISOSS architecture. In mid-April, IBM announced a direct link for the token-ring network to the System/370 through the 3725 communications controller, a token-ring-to-System/36 connection via a PC/AT gateway, and a token-ring-to-token-ring bridge. IBM also announced software for the Series 1 that implements links to DISOSS, System/36 and System/38 for its PCs, providing document distribution and library services.

In mid-June, IBM announced the 3174 family of cluster controllers that directly links the token-ring network to IBM mainframes. The new controllers, which replace the 3274 models, provide attachment of 3270 system displays, printers, and workstations to IBM host processors via a local channel, remote link, IBM token-ring LAN gateway and IBM token-ring LAN. At the same time, IBM also announced a physical token-ring connection for the RT, and remote PC access to the token-ring and PC Network through NetBIOS.

The announcements state a range of availability dates stretching from the present out to mid-1987. Given the apparent level of activity, more announcements can be presumed to be imminent, and may even occur between the final edit of this paper and its publication.

Software Support

Quite aside from the issue of support in hardware, there is the matter of LU6.2 applications software. IBM has published the LU6.2 specifications, and is relying heavily on third-party vendors to fill the applications software gap. Third-party development is already underway, but there is a long way to go. 3270 applications software—literally millions of lines of mainframe source code—will have to be largely rewritten to take advantage of a distributed processing environment.

Some of the 3270 applications will likely never be upgraded, because the cost in time, risk, and dollars is unwarranted. The fact that IBM has made new 3270 emulation products announcements concurrent with LU6.2 announcements is evidence of its recognition that the hierarchical environment will exist for quite some time into the future. The first LU6.2 implementations will most likely be side-by-side with 3270 technology.

Network Management

Finally, there are the issues of network management and diagnostics. As corporate processing moves away from a central control structure, these matters become increasingly complex. IBM has taken some steps toward solving these problems with recent announcements, particularly that of LEN, but it still has quite a long way to go.

LU6.2 AS AN INTERNETWORKING STRATEGY

As it provides an environment-independent, program-to-program communications technology, LU6.2 is well-suited as the basis of an internetworking strategy. The LU6.2 specifications are public, and most of IBM's competitors have already
announced that they will support LU6.2. Third-party communica-
tions software vendors already have portable LU6.2 soft-
ware packages on the market.

With a portable software package, the SNA and LU6.2 pro-
gramming is pre-packaged in machine- and operating
system-independent modules that need only to be compiled
and linked. Implementation in hardware consists of a por-
tion. The design of such packages simplifies porting to diverse
operating environments.

The availability of such pre-packaged LU6.2 capability
greatly reduces the development cost and time-to-market of
LU6.2 products for the hardware vendor. This means that
even smaller and special-purpose hardware vendors will be
able to offer LU6.2 products within a fairly short period.
Integrated LU6.2 networks composed of a wide variety of
equipment from many different vendors is a very real scenario
of the not-too-distant future.

BIBLIOGRAPHY

1. IBM Distributed Data Management Architecture: General Information
   (GC21-95270-0).
2. IBM Distributed Office Support System/370 Version 3 Release 3: Inter-
   change Architecture Reference (SC30-376-4).
3. An Introduction to Advanced Program-to-Program Communication
   (APPC) (GG24-1584-0).
   Distribution Services (SC30-3986-2).
7. Document Content Architecture: Final-Form-Text Reference
   (SC23-0757-1).
8. eSNA/LU6.2 Advanced Program-to-Program Interface User Manual, Pre-
   liminary Release 4.0.
9. Networks and Architectures: IBM Systems Network Architecture (SNA),
10. A. E. Baratz, V. P. Gray, P. E. Green, Jr., U. M. Jaffe and D. P. Pozefsky,
    "SNA Networks of Small Systems." IEEE Journal on Selected Areas In
    Communications, SAC-3 (1985) 3.