10
8/14/89

M.L.R.

1

SANDIA REPORT

SAND89—0466 • UC—905
Unlimited Release
Printed July 1989

The Product Data Network and Distributed Data System: Node Configuration (U)

Ronald C. Hall

Prepared by
Sandia National Laboratories
Albuquerque, New Mexico 87185 and Livermore, California 94550
for the United States Department of Energy
under Contract DE-AC04-76DP00789

DISCLAIMER

This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

DISCLAIMER

Portions of this document may be illegible in electronic image products. Images are produced from the best available original document.

Issued by Sandia National Laboratories, operated for the United States Department of Energy by Sandia Corporation.

NOTICE: This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, nor any of their employees, nor any of their contractors, subcontractors, or their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government, any agency thereof or any of their contractors or subcontractors. The views and opinions expressed herein do not necessarily state or reflect those of the United States Government, any agency thereof or any of their contractors.

Printed in the United States of America. This report has been reproduced directly from the best available copy.

Available to DOE and DOE contractors from
Office of Scientific and Technical Information
PO Box 62
Oak Ridge, TN 37831

Prices available from (615) 576-8401, FTS 626-8401

Available to the public from
National Technical Information Service
US Department of Commerce
5285 Port Royal Rd
Springfield, VA 22161

NTIS price codes
Printed copy: A03
Microfiche copy: A01

SAND--89-0466

DE89 013996

SAND89-0466

The Product Data Network and Distributed Data System: Node Configuration

Ronald C. Hall

Product Data Systems Management
and Development Division 2825

Sandia National Laboratories
Albuquerque, New Mexico 87185

DISCLAIMER

This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

ABSTRACT

Prototype systems have been established which prove the concept and utility of supporting distributed access to shared Product Test Data from Sandia National Laboratories (SNL) and General Electric Neutron Devices (GEND) using existing network communications facilities. It is essential, however, that adequate control (audit and isolation) be present as intrinsic characteristics of this connectivity in the production environment. A plan has emerged to put into place a configuration which provides required network functionality, yet preserves the necessary control. This configuration supports further development of a Product Data Network containing a Distributed Data System. We anticipate a complete "black hat" assessment of the final plan and configuration which results from these efforts.

* This work performed at Sandia National Laboratories supported by the U. S. Department of Energy under Contract Number DE-AC04-76DO0789.

TABLE OF CONTENTS

1. Introduction and Acknowledgements

The purpose of this document is to specify configuration options which support required bidirectional functionality (file transfer, distributed database, and electronic mail) between nodes at the Sandia National Laboratories (SNL) and General Electric Neutron Devices (GEND) sites in an external network. Explicit recognition is given to the fact that adequate SRD-level control must be preserved over such a network. This precludes direct intersite terminal access to processors (both internal and external) in the SNL secure partition. It is also necessary to identify, isolate, and control placement of data on shared media which can be accessed from the external processors. This will be locally administered at the site where the storage is maintained. In the case of remote access, only generic accounts needed to support registered applications which accomplish the required functionality will exist on the external processors. No remote user sessions will be permitted on these processors, only proxy access between registered applications to communicate in the precisely defined manner necessary to carry out the allowable functions. Local users will have accounts on the external processors as well as the internal processors and will be able to initiate direct user sessions.

This work provides a basis for enhancement and integration of existing Product Test Data / Systems Evaluation (PTD/SE) VAXcluster applications. The basis will support local integration of functions and data distributed among similar nodes participating in a Product Data Network which could make weapon evaluation data available to all Nuclear Weapon Complex engineers having the proper need to know.

Acknowledgements:

The productive participation of individuals from various organizatiions in deriving, reviewing, and refining relevant configuration options and features is acknowledged. In particular, the concept of a Product Data Network and Distributed Data System to support PTD/SE applications has only progressed through the foresight of G. Carli, G. M. Ferguson, and E. J. Theriot (SNL Departments 2820, 2530, and 2640) in envisioning the advantages obtainable through employment of this emerging technology to meet projected business requirements. The early concept [1] was refined and specified as an achievable goal consistent with SNL policy through the cooperative effort of J. K. Sharp (2825) and E. J. Theriot (2640). H. M. Bivens and D. H. Jensen (both 2534) provided insight from the applications perspective, and provided related contributions regarding network and node configuration details.

R. M. Jansma and B. Stiefeld (both 2612) contributed ideas and effort directed towards meeting computer security reuqirements. The significant contributions of L. G. Pierson and M. O. Vahle (both 2647) in merging the overlapping requirements of separate networking configurations and constraints are appreciated. C. K. Haaker and M. O'Malley (both 2644) made contributory comments and suggestions through review of proposed VAXcluster configuration details for nodes to be included in the Product Data Network. The interest of W. D. Wilson (8230) and M. H. Pendley (8231) shown in their evaluation of concepts applicable to the Product Data Network is acknowledged. Their willingness to practically assess the final configuration is further appreciated. Thanks also to C. W. Wiltshire and his associates at GEND who have long shared a common vision of what the Product Data Network and Distributed Data System might offer, who participated in the early formation of concepts, and who persevered through the definition and implementation work for the existing prototype system. Finally, the early impetus by J. R. Yoder (2813) given to the idea of distributing PTD/SE data is acknowledged.

2. Goal and Objectives

The goal being addressed is identification of network and individual node configuration options which can be achieved to implement secure connectivity and functionality supportive of a Product Data Network and Distributed Data System. Towards this end, an intersite connectivity model is described in this section which specifies required functionality yet recognizes the absolute requirement for SRD-level control over the connectivity. A network topology to meet these requirements and afford this protection is diagrammed. Implementation of a prototype system which proves these concepts and protection is discussed. Finally, the threat is identified against which the protection must remain effective. It is required that the protection remain effective whether against hostile attack or inadvertent error by legitimate users of the system.

The prototype system described in this section is intended to demonstrate feasibility and provide a platform or test bed to prove the configuration options. The implementation is not intended to imply a sole source specification for the Product Data Network and Distributed Data System. Through use of standards such as Standard Query Language (SQL) and high degrees of normalization in the database definitions, maximum transportability and interfacability among various hardware and software components is being sought. Database software gateways can be constructed to interface various data orgainzations into the system. In fact, the prototype plans include such a gateway to provide access to VMS RMS files just as if the data resided in local database relations. Technology is also advancing in areas of hardware interfaces among network components. The VAX VMS systems and INGRES relational database management software which constitute the prototype are being employed as examples of system components which conform to the connectivity model and topology supportive of the goal stated above. They are not sole source components of such a system.

2.1 Intersite Connectivity Model

Figure 1 diagrams an intersite connectivity model which addresses a previously identified need to specify the nature of required network functionality [1: Table I, Item 3]. Although simple in concept, the model is significant in that it represents a concensus among various interests pertaining to the requirements for intersite connectivity. The model, which was derived through initial effort to implement the GEND/SNL prototype system, reflects the participation of users and applications interests, systems and computer security personnel, as well as participation from multiple sites. The model specifies a need for controlled, bidirectional access between remote sites for the purpose of accomplishing precise functionality: file transfer, distributed database, and electronic mail. Since secure machines are involved, the control must be at the SRD level. Applications involving shared access to SRD data are also envisioned for the future. Control implies both audit features and isolation of discrete networks. The

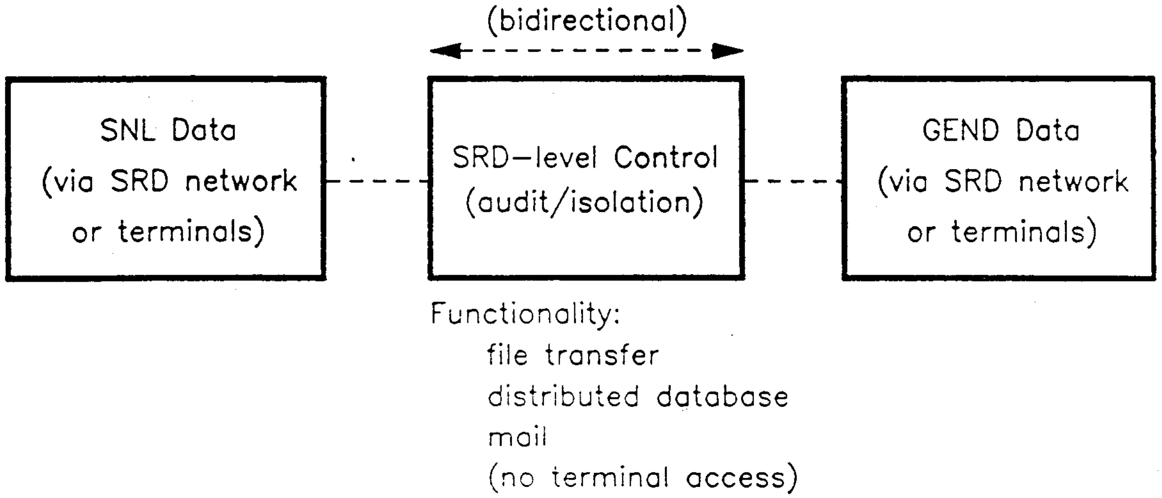


Figure 1. GEND/SNL Intersite Connectivity Model.

functionality is similar in that all required functions can be fully accommodated through registered applications which implement precisely defined, trusted protocols. No remote terminal access is required from one site to another. No "set host" or "submit to queue" functions need to be supported to meet these requirements. These limitations preclude the possibility that users at one site can place arbitrary instruction sequences into execution at another site. Terminal access is restricted to systems at the local site. The registered applications then provide any necessary access to shared information throughout the network. A topology to support this connectivity has been identified.

2.2 Network Topology Model

Figure 2 illustrates a topology through which the required network functionality can be realized in a controlled environment. The topology limits access to only that provided by the trusted (registered) applications. Internal data are not known or accessible from the external network. Shared data, on the other hand, are maintained on shared media which can be accessed from the external network as well as the internal network of the creating site. The external network includes local processors which have accounts for local authorized users, and also communications facilities (e.g. gateways and network links) which provide the intersite connectivity. These facilities implement the registered applications and shared resources such as distributed database definitions. Remote users can only access the shared data through the predefined functions and protocols which are supported in this manner throughout the external network. The line drawn through the external network in Figure 2 indicates that shared data are currently accessible only from GEND and SNL. Future enhancements are expected to extend this capability to include other sites, and future releases of the NWCNET software for the WBCN may be enhanced to provide such support. Hardware and software configurations which can be currently used to implement and protect these capabilities are described in later setions of this report. Configurations for specific nodes including both a communications (gateway) processor and external processors to support the user community are included as appendixes. These nodes are also identified in the following brief description of the existing prototype implementation.

2.3 Prototype Implementation

Figure 3 shows a distributed database (DDB) prototype system which has been constructed to join data from GEND with data from SNL on Organization 2500 and 2800 VAXclusters. The network links depicted in this diagram are considered to reside in the external network portion of the topology illustrated in Figure 2. The SAV06G gateway is an Organization 2600 communications machine which has no direct user accounts. This machine terminates the SNL end of the GEND ELSENET link. Proxy accounts do exist to support execution of INGRES/STAR and INGRES/NET componenets which function as a registered application to accomplish access to the distributed database. INGRES/STAR

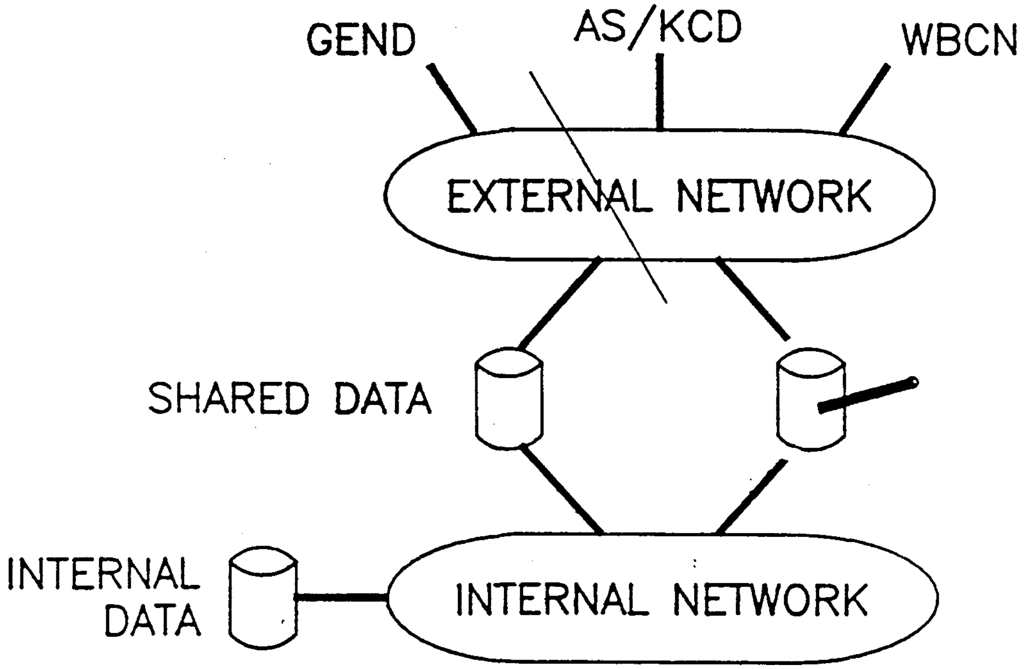


Figure 2. Network Topology Model.

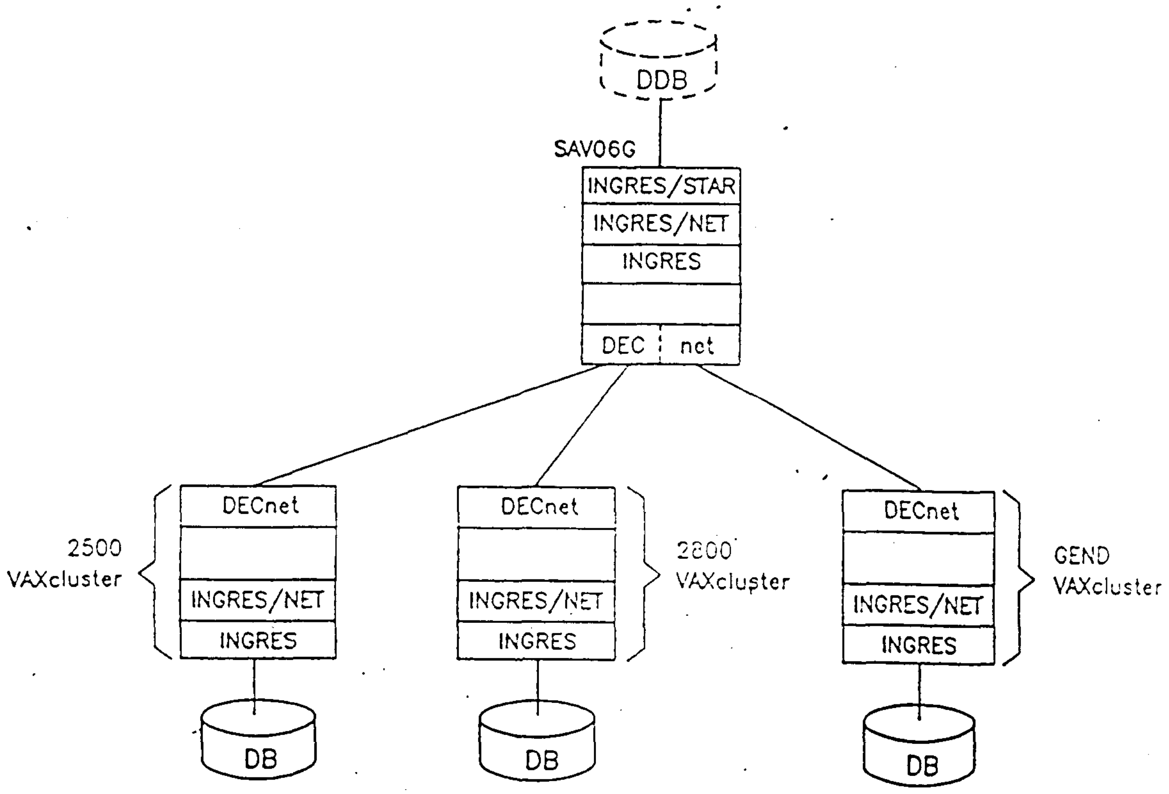


Figure 3. DDB Prototype.

has been used on SAV06G to define the distributed database. No physical data reside on the SAV06G storage. Instead, database links are defined into shared, external data on the worker machines (2500, 2800, and GEND VAXclusters). In this manner applications can be constructed on the worker machines which reference the distributed database on SAV06G. No further user access is required by this prototype, its function being limited to accessing authorized data through the distributed database defined on the SAV06G machine. Severed router software has been installed on the SAV06G gateway (indiated by the dashed line across DECnet). This limitation prevents requests for network connections from being routed through SAV06G unless they are recognized and authorized by the highest (application) layer in the networking protocol. This assures explicit limitation of logical links involving the connectivity through SAV06G to only those known processes which implement the registered applications. This prototype system will serve as the test bed to prove the options described and proposed in this report. Proof of the options will exist when it has been demonstrated that adequate protection is provided.

2.4 Threat Definition

The internal secure network already exists at SNL, and includes security features as described in applicable security plans for the systems. The threat against which additional protection must be established is unauthorized access or entry from the external network to the internal network. Gateway computers in the external network are trusted machines which contain no user accounts and run no user code. They pose no direct threat unless they are captured or subverted by users of worker machines residing in the external network. Therefore, the boundary which must be protected against is the set of processors in the external network which serve as worker machines to support user accounts. This includes local processors in the external network as well as remote external processors having network connections to the gateway facilities.

The meaningful threat against which protection must be assessed is the set of privileged and unprivileged accounts on external processors at remote sites, and the set of unprivileged accounts on external processors at the local site. This includes any default accounts, as well as surreptitious assumption of any account identity authorized by the system. Protection means that the control features (audit and isolation) of the system cannot be overidden by any means. It is expected and must be allowed that local privileged users of external processors will often be responsible for maintenance and administration of both the external and internal processors. They will typically have physical access to the facilities, software, and operations which support the machines. Local privileged users (e.g. system managers) of both external and internal processors at a site are not, therefore, included in the threat definition. In summary, the control and protection of the system must not be defeatable by any unprivileged user of external processors at any site, nor by any remote privileged user having access to processors at another site in the external network.

3. Node Hardware Configuration

Suitable configuration options must be specified for ongoing support of the prototype system described in the previous section. Once proven, these options will form the building blocks for the Product Data Network and Distributed Data System. Following are applicable hardware considerations.

3.1 VAXcluster

A VAXcluster [2] is an integration of VAX VMS systems that communicate over a high-speed communications path known as a Computer Interconnect (CI). A VAXcluster can support up to 16 active nodes through a single CI for the VAXcluster. The CI is a dual path bus that connects processor nodes and intellegent I/O subsystems within a 45 meter radius. The intellegent I/O subsystems are called Hierarchical Storage Controllers (HSCs), which provide shared access to DIGITAL Standard Architecture (DSA) disks. A star coupler is the common connection point for all nodes connected within the VAXcluster through the CI. The star coupler connects the CI cables and physically supports the cluster operations in a passive mode.

Individual VAXclusters can be constructed to conform to the node connectivity configuration diagrammed in Figure 4. As detailed in the appendixes to this report, the DDB prototype system of Figure 3 will be comprised of only the external processors in each of the illustrated VAXclusters. The SAV06G communications machine will be limited to external network connections. This configuration will establish isolation between shared storage and internal storage, i.e. it can be used to isolate internal storage from the external network. As shown in Figure 4, external processors are connected to the Product Data Network (PDN). Internal processors are conneted to the internal network at a site, e.g. the Central Computing Network (CCN) at SNL. There is no network connectivity between external and internal processors, including the absence of DECnet over the CI. Internal processors may have network connectivity to other internal processors, and external processors may be likewise connected to other external processors. These connections may be DECnet point-to-point circuits or local Ethernet taps depending on applicable requirements for connectivity.

3.2 Sh ·ed Storage

Shared storage will be implemented using an HSC with DSA disk units connected through the CI and star coupler. Shared storage will be available to all processors in the VAXcluster. Shared storage can be mounted from both internal and external processors.

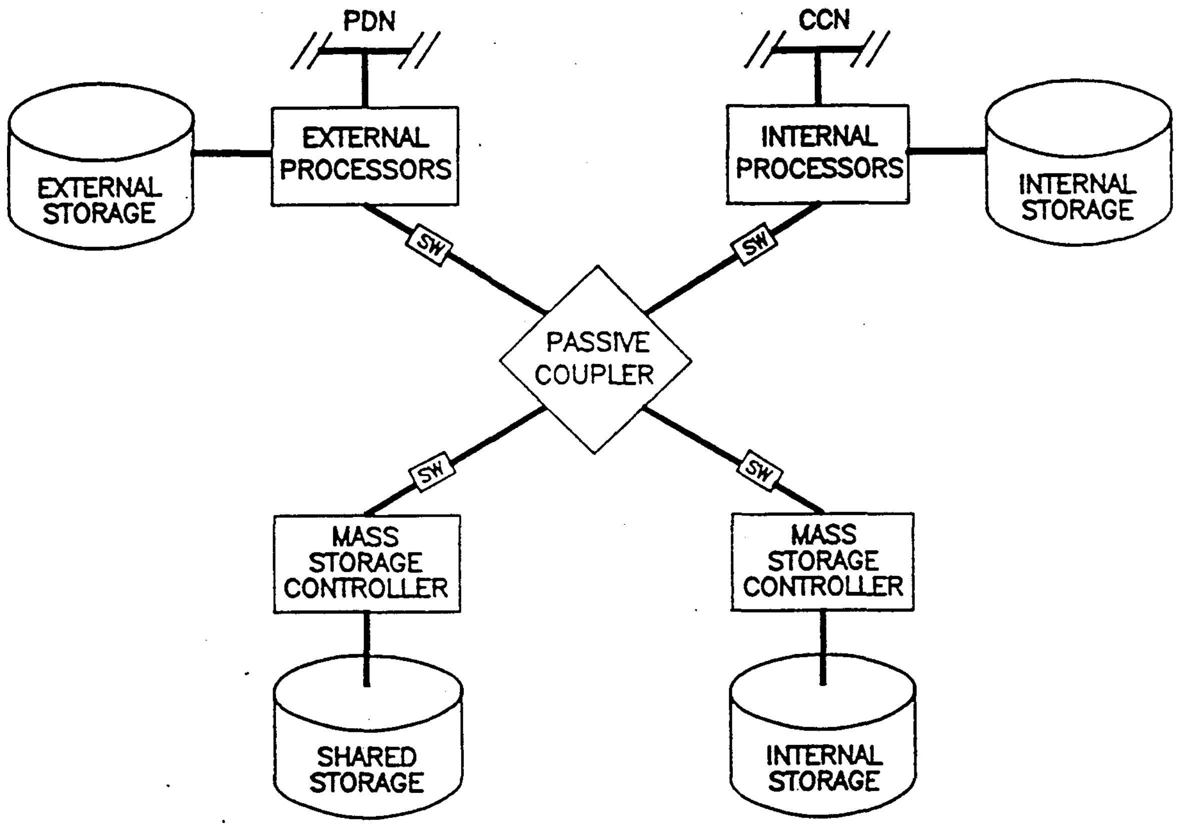


Figure 4. Intersite Connectivity Configuration.

3.3 Internal Storage

Two options are identified for internal storage, both of which are represented in the planned prototype configuration. First, an HSC and DSA disk units connected in a manner similar to that used for shared storage can have the disk units established in a "not available" device status at boot time for the external processors. This prevents mounting the devices from the external processors. The devices remain accessible from only the internal processors. The second option for supporting internal storage within the VAXcluster is through local disk devices (including MASSBUS, UNIBUS, and UDA units) directly connected to the processors. Such non-HSC disks which are local to VAXcluster processors can be established as cluster accessible by explicit declaration and action. These devices can be shared among internal processors having network connectivity through use of the VAX Distributed File System (DFS) software. Effective use of this option probably requires Ethernet connections (rather than point-to-point) in order to minimize the network overhead in supporting the shared access. As indicated in Figure 4, this option can also implement exclusive external storage. A possible application of such storage could be to provide a local system disk for an external processor, and then to share access to the device with other external processors in the cluster after they are booted.

3.4 DECnet Within the VAXcluster

DECnet [3] supports both the protocols necessary for communicating over a network of VAX VMS systems and the functions necessary for configuring, controlling, and monitoring the network. It is typically desirable to support DECnet communications within a VAXcluster, including using the CI as a circuit. As previously indicated, DECnet is not operational over the CI in the proposed configuration. Although other network connections may exist, the CI adapters will be limited to data buffering, node address translation, and protocol support for mass storage access. This restricts data communications through the star coupler (shown generally in Figure 4 as a "passive coupler") to only that necessary to support the Mass Storage Control Protocol (MSCP). This is the protocol used to communicate between processors and HSCs to access the shared DSA disks. This access does not require DECnet connectivity. The isolation provided is protected through the hardware in that the node address switches act as filters for the protocol messages. These switch filters (denoted by "SW" on the signal lines in Figure 4) prevent a mass storage response from being delivered into the memory of a processor unless it is the requester of the transfer. The request is not delivered to the subsystem unless it is for access to mass storage which is available to the processor. Responses are not received by any other processors in the cluster. This isolation is protected by a firmware EEPROM in the CI adapter itself. No modification or replacement of software on or from the external processors can establish a data path to or from the internal processors or internal storage. Other, network connectivity among the internal processors or among the external processors is a local VAXcluster management consideration. In no case, however, may network connectivity be established between external and internal processors.

4. Node Software Configuration

Just as suitable hardware configuration options are required as building blocks for the Product Data Network and Distributed Data System, software components must also be suitably configured to support the system. This includes installation of necessary software to provide the required functionality, and in some cases limitation or explicit control of software to secure the protection of the system. Applicable software considerations are addressed in this section.

4.1 VMS and DECnet

VMS and DECnet constitute the operating system and communications software to support network nodes within both the external and internal networks. DECnet interfaces are integral parts of VMS. Time-sharing and production environments can be simultaneously provided by VMS. Time-sharing accounts can accomplish work interactively and can also submit jobs into system queues. They can access other network nodes through proxy accounts or specific authorization and authentication carried out locally on the node being accessed. VMS and DECnet are well documented [4], and together with the wide variety of layered products available from Digital Equipment Corporation and compatible third-party software, can be used to achieve a very functionally rich computing environment. This environment can be further partitioned and secured through use of VMS Security Enhancement Software (SES) which is available to support multiple security levels within the same secure machine. The external network is a fully encrypted, secure network. Both classified and unclassified processing can occur on processors within this network. Although initial distributed applications are planned to be unclassified, it is appropriate to use VMS SES to partition SRD processors connected to the external network.

VMS and DECnet files reside on a device known as the system disk, which is the device from which the system is typically booted. VAXcluster nodes can be booted from any combination of common system disks and individual or local system disks. Clusters which boot from a single common system disk are said to be homogeneous, while clusters comprised of nodes with individual system disks are said to be heterogeneous. Various degrees of homogeneity or heterogeneity can be achieved through tailoring common and node-specific system procedure files to establish whatever startup, login, or other system environment is appropriate for each processor in the cluster. Separate procedures (files) can be used for each processor node, or, since node identifications are known at boot time, node-specific conditions can be evaluated by the software and appropriately accommodated when each processor is booted.

Hardware options for shared and internal storage for the external and internal networks were previously described in Section 3. Local disks cannot support common system files even if access to them is shared. The units must be physically connected to a processor in order to accomplish the system boot. There are three overlapping options which result for structuring system disks to accommodate a VAXcluster spanning the external and internal networks: a common system disk in shared HSC accessible storage, a common system disk in internal HSC accessible storage for internal processors only, and individual system disks physically connected to individual processors.

The requirements related to system disk allocation to support VMS and DECnet are achievement of the proper computing environment for each processor, and securing the necessary protections such that the boundary between the internal and external networks is not breached. The user authorization files accessed by each processor must be appropriate for the interactive community it supports. The batch environment and resource availability must be consistent with the work to be performed. Specific VMS requirements are described in the remainder of this section. Network software considerations are included in Section 5.

4.2 System Queues

Batch and print job processing is managed through queues which can be established on VAX VMS systems. This is the manner in which processor and I/O-device resources can be most effectively utilized. On a VAXcluster, processors can share device and computing resources. A cluster-wide job controller queue file can make queues available across a VAXcluster, and can enable jobs to execute on any queue from any cluster node. However, files queued for processing must be accessible to the cluster node on which they are being processed. There can be only one cluster-wide job controller queue file on a VAXcluster. If such a queue file is established, it must be available to all cluster nodes which participate in the cluster-wide queue scheme. Generic queues can be cluster-wide, or local to a particular processor. Generic queues are not specific to a particular resource. These facts have implications for deciding how to configure the system disks or for choosing which processors should participate in cluster-wide queue management. It is essential that jobs submitted from one network (external or internal) not be executed on processors which are connected to the other network. If the direction is from the external network to the internal network, then arbitrary instruction sequences can be placed into execution across the network boundary. If the direction is from the internal network to the external network, then a data path could be established which transends the network boundary.

Several options are available for establishing print and job queue configurations on a VAXcluster spanning the internal and external networks through the shared storage medium. In general, cluster-wide queue management should be restricted to either the internal or the external processors depending on relative need for this capability. Beyond that, several decisions need to be made in deriving the most productive configuration for a given facility. Applicable considerations relative to cluster printer queues include requirements for generic printer queues which are local to each processor, identification of specific printer device queues which should be assigned to local generic queues, and establishment of any needed non-local generic queues that will distribute print job processing across members of the cluster which participate in cluster-wide queue management. Configuration of batch queues involves comparable considerations except that no single-thread physical devices are involved. Considerations include establishment of local batch queues that conform to the processing requirements of each processor in the cluster, and then identification of any non-local generic queues needed for distribution of batch job processing through the cluster according to the cluster-wide queue management scheme. Appropriate procedures to initialize and start the queues must be tailored for each cluster node and executed at boot time.

4.3 Distributed Database

As described in Section 2, the INGRES relational database management system has been used to establish a prototype which demonstrates feasibility of the Product Data Network and Distributed Data System and provides a test bed to prove the configuration options. Once proven, these options will support continued development and implementation of the system. Installed components of this software include INGRES for VAX VMS [5], INGRES/NET [6] to accomplish the task-to-task communications necessary for distributed database access, and INGRES/STAR [7] to provide the distributed database management and definition functions. Together these software products constitute a registered application which communicates through the external network according to a specific and controlled protocol.

Using INGRES as a database management system, local databases can be established on shared media from either the internal or the external processors. A distributed database can then be defined using INGRES/STAR on an external processor such as a communications gateway in the external network. The distributed database contains references to the data stored in the local databases. Establishment of the distributed database is accomplished by the Distributed Database Administrator at a site, and does not require existence of user accounts on this machine. User access to the data is provided by application programs residing on other external processors which issue queries against the distributed database. INGRES/NET establishes the logical links necessary to cary out the task-to-task communication between INGRES and INGRES/STAR. General task-to-task communication is not required through the external network to accomplish this function, and is not supported in the existing prototype system.

4.4 User Applications

User applications, distinguished in this context from registered applications, are executed and perhaps even compiled and linked within sessions established by users. The official software version or release may be subject to formal configuration management and control, but ultimately the program can be executed directly from individual user accounts on user machines. These programs may issue requests for services which registered applications provide. The requests must conform to an established protocol in order for a valid response to result.

Applications developed using the INGRES Applications-By-Forms (ABF) software development tool [8] are examples of user applications which can access a distributed database through the registered application described above. SQL is a standard (protocol) which can be used to implement a controlled and protected interface between the users and the service. Similar interfaces support access to other required network functions such as file transfer and electronic mail. The DECnet File Access Listener (FAL) and Mail object (MAIL) are examples of software which accomplish these functions.

5. Network Configuration

Full network connectivity and functionality are supported in the internal network at SNL. This includes full-function DECnet between distributed VAX nodes in the internal network, as well as access to all Central Computing Network (CCN) services. Controlled connectivity must be provided in the external network to implement the functionality required by registered applications. Additionally, full DECnet connectivity can exist between external nodes in a VAXcluster, or to other local VAX nodes within the external network. No network connectivity is permitted between internal and external nodes, nor simultaneous connectivity to any other common node.

5.1 Connectivity

Nodes which have connectivity through a network communicate with each other over logical links. A DECnet logical link implies a connection or channel at the application level which is implemented through physical lines over which circuits operate between adjacent nodes. Routing nodes extend the connectivity by forwarding messages from one node to another when the two nodes exchanging the messages have no direct physical link or circuit between them. Connectivity between two nodes is established when sufficient circuits are defined and placed in an operational (ON) state such that continuity exists over some path of adjacent nodes between them. The connectivity allows an application program on one node to communicate with another application program on some other node within the context of a session. Multiple parallel sessions can be established over the same physical path.

A configuration database exists for each DECnet node, and contains information about the local node, remote nodes, local physical lines and circuits, monitoring and logging events, and local network objects which can be invoked. DECnet connectivity is established according to these parameters when the network software is started. Privileged access is required to alter these parameters or change the configuration in any way. Several restrictions are applicable for nodes which support DECnet connectivity within the external network. First, no lines or circuits will be defined for CI devices on SNL VAXclusters which span the internal and external networks. The CNDRIVER can even be removed from such systems to disable this functionality on the nodes. Additionally, in no case will a single configuration database or single network configuration file establish lines or circuits which are connected and operational (ON) to nodes in both networks. The network which is not being accessed will have lines and circuits established in a non-operational (OFF) state. Note that this restriction does not mean that a single processor cannot be alternatively connected within one network and then subsequently connected within the other. It does mean that a single processor cannot be configured and connected in both networks at the same time. Any node which establishes connectivity to another network must first shut down DECnet, must have appropriate DECnet node identifications for all networks, must have all queues stopped and deleted before a reconfiguration, and must have all user processes terminated during the transition. Any local internal storage connected to such a processor must be off line and made not accessible while the processor is connected to the external network.

5.2 Control

The intersite connectivity model described in Section 2 identifies certain controlled, bidirectional functionality as required features of the external network. Two aspects of control are identified: audit and isolation. Control in this sense does not refer to management or maintenance of the network. These capabilities are provided through use of the Network Control Program (NCP) used for DECnet network administration [9]. The control identified in the model refers to ensuring isolation such that the boundary between the internal and external networks will not be crossed except for legitimate access to shared storage. In addition, audit capabilities must exist such that connectivity within the external network is sufficiently monitored and recorded to resolve any access anomolies or unauthorized access attempts which occur. Initial applications are unclassified. However, since classified machines communicate over the connectivity and classified applications may be required in the future, control must be achievable and verifiable sufficient to obtain approval of required security plans covering SRD use of the link.

Several measures are available to provide necessary isolation. Many have already been described. In fact, the foregoing descriptions of hardware, software, and network connectivity options directly support the requirement for isolation. Isolation of the external and internal networks is implemented by not allowing DECnet connectivity between them. VAXcluster CI devices are not used for network circuits, even among like processors within the same network. Unused software which supports this interface (CNDRIVER) will be removed from the systems to further protect this isolating measure. Use of the severed router software on machines which terminate the SNL side of remote external circuits ensures that connect requests for arbitrary logical links are rejected before they would ever reach the network boundary. Terminal access from remote sites to external processors is not allowed. Proxy accounts with no interactive or batch access provisions will safeguard restriction of remote network access to these machines to only the registered applications allowed through the severed router. Finally, remote access can be further controlled by not having default DECnet accounts on external systems. Proxy access can be used for explicit control of all required functions within the external network. Code which implements these functions will have file protections established such that replacement or modification is not possible except from authorized privileged accounts which are local to the system being modified.

Most audit requirements applicable to external processors are not unique to the external network. Organization 2600 at SNL distributes and maintains certain VMS enhancements and utilities which include audit capabilities to support VAX systems connected to the internal network. These VAX systems are called SNL Type-1 VAX installations. It is anticipated that SNL external processors will be operated in a manner similar to the Type-1 internal processors. Of course central network access through the distributed computing gateway will not be provided for the external processors, but the SNL utilities and audit software for Type-1 VAX systems will be useful. Specifically, SNAPSHOT, SNLAUDIT, and NETWORK_ACCESS should be routinely run on the external processors. SNAPSHOT is needed to monitor the integrity of these systems, checking to be certain that various system information and software have not been altered. SNLAUDIT summarizes security alarms logged by the VMS AUDIT utility, including both file access activity and network access anomolies such as suspected breakin attempts. NETWORK_ACCESS summarizes DECnet loggin activity, identifying users and network objects involved in legitimate network access. Two additional audit requirements exist within the external network. These, which are more application specific, involve monitoring requests for network connections received at the severed router, and then journaling the application activity if the requested logical link is established. These requirements are identified here, but are described in the following discussion of registered applications.

5.3 Registered Applications

Task-to-task communication supported throughout the external network is intended to be limited to registered applications which implement verified required functionality. The manner in which the SNL side of the circuit to GEND is terminated on a trusted communications machine has been previously described. This machine, which can be called a gateway, implements a severed router to be sure that no request for a logical link is forwarded through the external network unless the connection is specifically authorized for a registered application. This authorization is verified by the severed router at the highest layer in the communications protocol. Implementation of communications networks is typically carried out according to a layered approach in order to simplify the design and interfaces. Each layer is intended to provide a communications facility to the next higher-level layer on the same node, and to corresponding-level layers on adjacent nodes. Figure 5 illustrates the ISO/OSI reference model for communications protocols described by the International Standards Organization. Interfaces for actual communications paths between two nodes are shown as solid lines. Protocols for virtual communications are shown as dashed lines. In the case of routing connect requests, protocols which serve the lower-level layers are severed. The result is that no network connection can be established unless it is seen and passed at the application level. This fact allows explicit control of network connectivity involving the gateway. Specifically, network connections can be restricted to those which are authorized

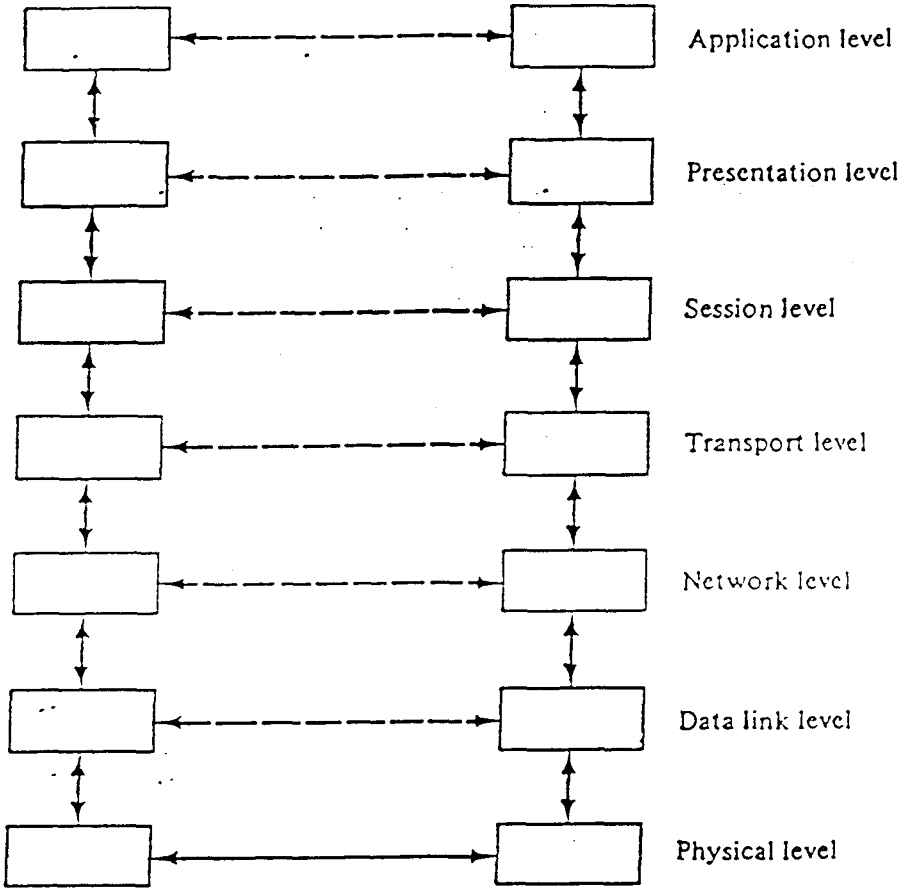


Figure 5. ISO/OSI Layers.

for registered applications on specified machines. Other requests are rejected. In addition, since control is obtained at the application level, audit can be implementated by logging the activity in whatever detail is appropriate. Only authorized requests for connections are processed on this machine, or forwarded on to other machines to establish logical links to registered applications which exist there. A permanent record of the connections can be made to provide an audit trail. Registered applications are installed and protected to be certain that only known and trusted functions are performed. They can be invoked by users across the network when connections to them are established through the severed router. Only this functionality is available to the network users, precluding the ability to place other instruction sequences into execution.

The need to journal transactions processed by the registered applications within the external network has been previously identified. Like monitoring of requests for connections, journaling is most practically and effectively implemented at the application level. Requirements for this activity vary from application to application. Application-specific transactions introduce variability into the nature and scope of information which should be recorded. Often only transactions which modify information are journaled. In some cases, however, there may be a need to record retrieval access as well. The application can record the source and result of journaled transactions, and can make the journal available for backup and recovery operations if necessary. At least two considerations are pertinent in establishing journals for applications implemented in the external network. First is the extent of journaling as it relates to the impact or result of each transaction. The second consideration involves the amount of detail necessary to sufficiently identify the source of a specific transaction. Journaling of results supports backup and recovery. Identification of the content and source of transactions is the important consideration as regards audit of network activity to provide logging and control. Monitoring can take the form of recording the precise text submitted from a source as a transaction. Alternatively, a transaction summary can be recorded which identifies only operations and objects for transactions which can be successfully parsed. The detail required depends on the application, but in all cases the source of a transaction should be recorded. The extent to which the original identity of a user is retained throughout the network is determined in part by the nature of the proxy accounts established to support exernal network access to registered applications. This has an impact on the detail available to the application in identifying a transaction source. If multiple users at a remote site are proxied into a single local account, then only that associated group of users will be identified as the source of a transaction arriving through this access path. This will also be the maximum level of control which can be specified in creating different levels of access, such as Access Control List (ACL) managed file transfers or granting different PERMIT access to distributed databases. Application-specific decisions must be made to achieve sufficient detail in source identification, while minimizing the extent of network proxy accounts which must be maintained throughout the external network. The prototype distributed database currently has a single proxy account (DDBUSER) to support retrieval only. The single account is acceptable for prototype work, but will not be sufficient for a production environment with differing access authorization requirements. File transfers and network mail are currently implemented on an individual user basis. The external network (including communications machines) must provide a capability to retain user identities necessary to support audit requirements and authorizations or restrictions for access.

5.4 Shared Storage

Hardware, software, and network components have been described which provide and protect the separation of internal and shared storage facilities. Only processors within the internal network can access internal storage. Access to shared storage is supported from both internal and external processors. The existence of shared storage is identified here to emphasize that it is the only commonality between the internal and external networks. No simultaneous network connectivity, terminal access, task-to-task sessions, or communications of any type is permitted between the networks except for the medium of shared storage.

6. "Black Hat" Assessment

It is necessary to practically assess the final configuration which is put into place to support the requirements identified in this report. Towards this end, Departments 2610 and 8230 have expressed an interest in carrying out a "black hat" assessment of the configuration. This assessment will result in contributions being made in several areas. Practical assessment of the configuration will help reveal any deficiencies which need to be corrected. Furthermore, periodic testing of protection features is a requirement for accreditation to operate in a classified mode. Since some of the systems involved include SRD machines, security testing of the network boundaries and the facility for shared storage will be required. It is important to categorize features of deficiencies which may be determined to exist so as to identify an appropriate means of resolution.

6.1 General vs. Implementation Deficiencies

Any deficiency which may be discovered could be unique to the configuration implementation, or could be characteristic of a general feature or system such as VAX VMS, DECnet, or the INGRES software. Any deficiency which relates to the configuration will be addressed within the context of the Product Data Network and Distributed Data System. General deficiencies, on the other hand, may have much broader implications and may need to be addressed from a broader perspective.

6.2 Correctable vs. Noncorrectable Deficiencies

Some deficiencies which may be identified may be correctable. Other deficiencies may not yield to direct solution. Alternative fixes or solutions should be identified and evaluated through the "black hat" assessment when they are known to exist. Alternative approaches which may be identified should be described and evaluated when deficiencies are considered to be noncorrectable.

6.3 Threat Assessment

The threat against which deficiencies should be evaluated is the threat defined in Section 2. This includes unprivileged users at any processor within the external network, and remote privileged users of machines at remote sites connected to the external network. Audit and isolation features of the external network must remain intact regardless of the actions of these users. A record of all network connections established must be maintained. All transactions must be journaled according to the audit specifications applicable for any registered applications. Internal storage must not be read or modified from the external network. It must not be possible to override access protections established for shared storage or gain unauthorized privileges for any process. No software execution or network connectivity may proliferate into the internal network.

6.4 Assessment of the Severed Router

The function of the severed router in limiting network activity to only previously authorized connections has been described in the discussion of registered applications. As one might imagine, the administrative task of maintaining and supporting such a system grows significantly with the number of users, processors, and network objects which must be served. For this reason it is important to assess the configuration both with and without the severed router in the external network. This comparison will demonstrate the real benefit resulting from the severed router. If the audit and isolation features of the external network even without the severed router are not vulnerable to the defined threat, then the severed router should not be included in a production environment. To do so would unnecessarily increase administrative burden, and would increase the likelihood of inappropriately rejected logical link requests when network participants and characteristics change.

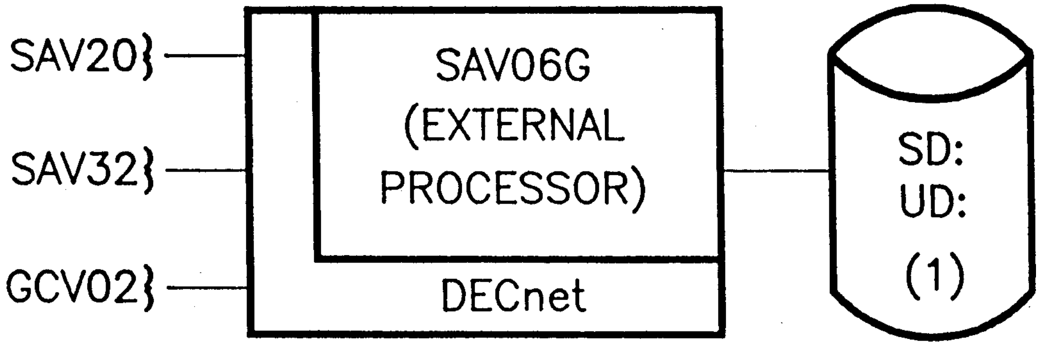
7. Conclusion

Configuration options have been identified and described for systems which will participate in a Product Data Network and Distributed Data System. These options are specified in a plan for enhancing the existing prototype which supports distributed access to shared Product Test Data from SNL and GEND. This plan includes a variety of options as specified in the appendixes to this report, all of which should be practically assessed. "Black hat" assessment of these configuration options is proposed, and is invited in response to the willingness of Departments 2610 and 8230 to carry it out.

Appendix I

SAV06G Communications Machine

SAV06G Communications Machine



Notes

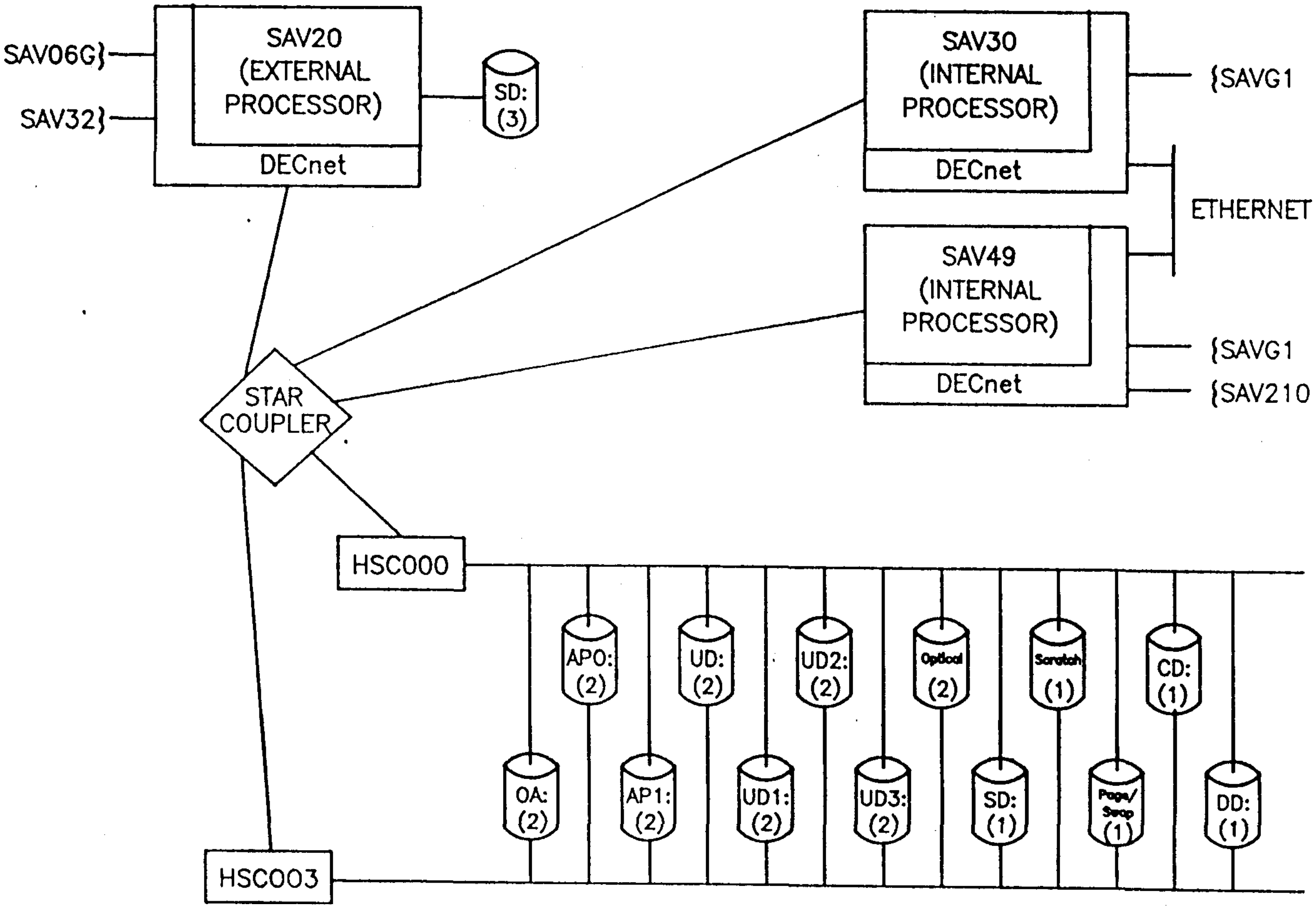
SD: system
UD: user and DDB definition

(1) local storage, entirely within external network.

Appendix II

2500 VAXcluster

2500 VAXcluster



Notes

OA: office automation
APn: applications
UDn: user
Optical: optical device

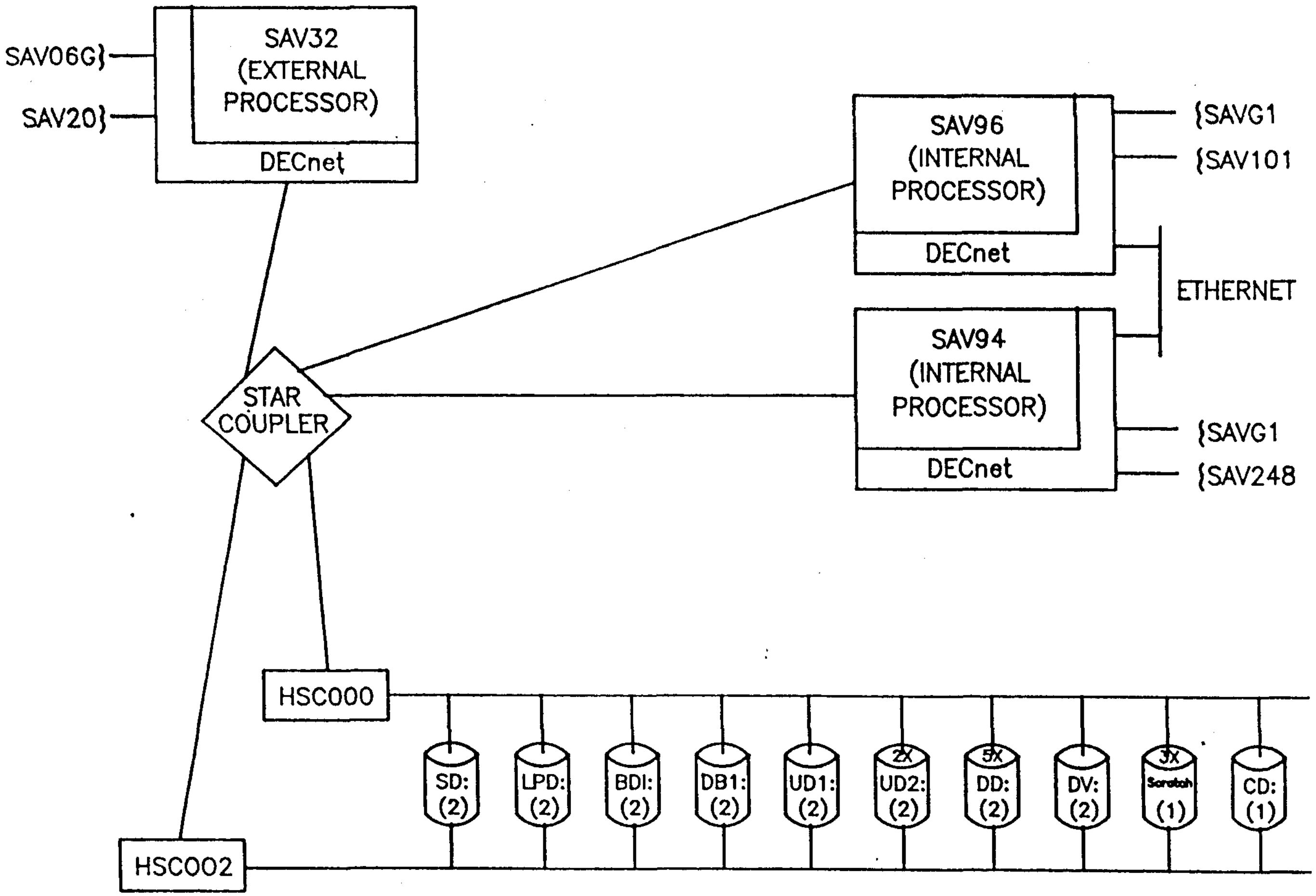
SD: system
Scratch: available
Page/Swap: system
CD: classified storage
DD: data

(1) "NOAVAILABLE" from SAV2Ø, accessible from internal network only.
(2) shared storage, accessible from internal and external networks.
(3) local storage, entirely within external network.

Appendix III

2800 VAXcluster

2800 VAXcluster



Notes

SD: system
LPD: layered products
BDn: backup
DBn: database

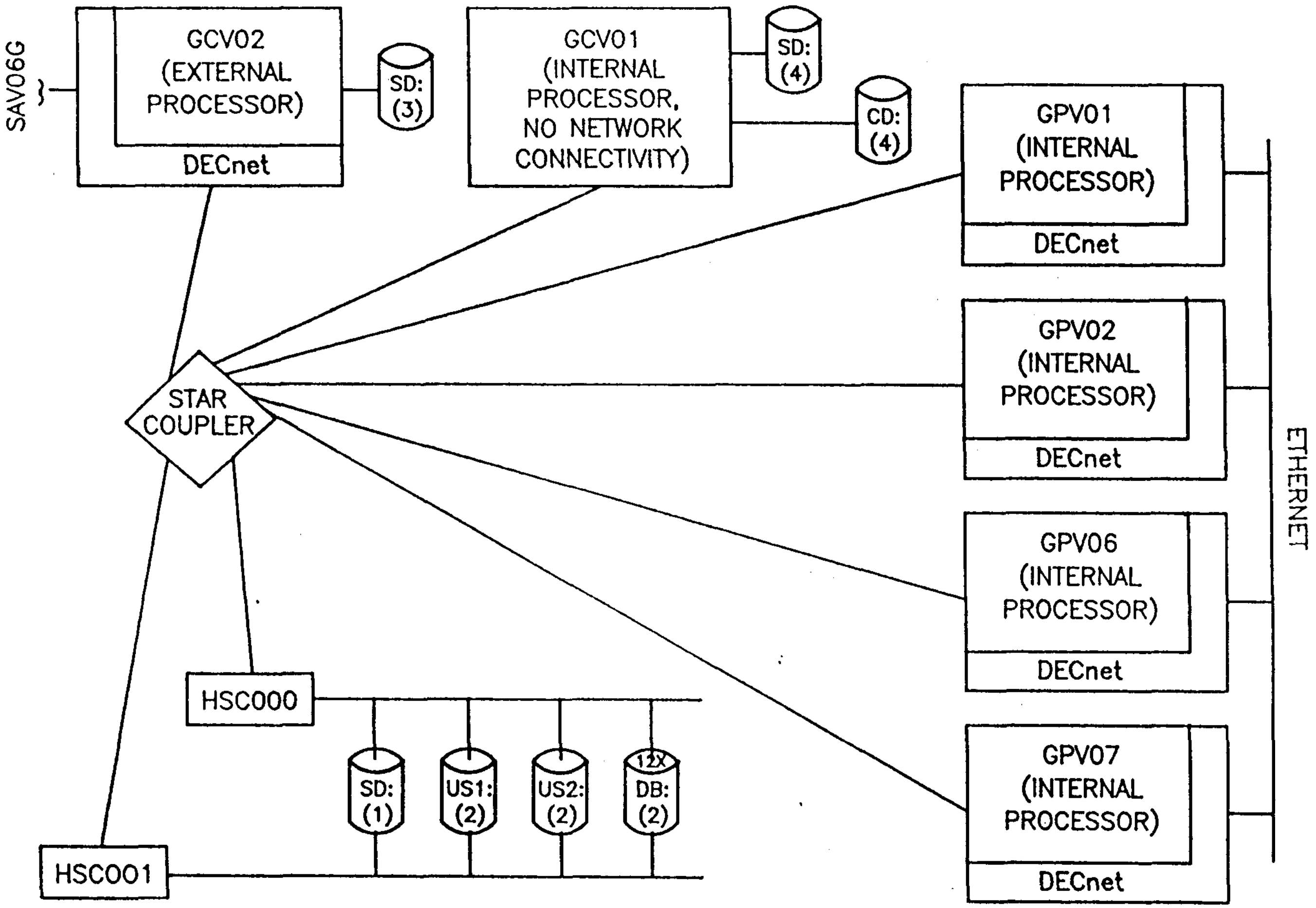
UDn: user
DD: data
DV: development data
Scratch: available
CD: classified storage

(1) "NOAVAILABLE" from SAV32, accessible from internal network only.
(2) shared storage, accessible from internal and external networks.

Appendix IV

GEND VAXcluster

GEND VAXcluster



Notes

SD: system
CD: classified storage
USn: user
DB: database

(1) "NOAVAILABLE" from GCVØ2, accessible from internal network only.
(2) shared storage, accessible from internal and external networks.
(3) local storage, entirely within external network.
(4) local storage, entirely within internal network.

REFERENCES

1. Hall, R. C., The Product Data Network and Distributed Data System: Conceptual Node Model, SAND88-0689, Sandia National Laboratories, Albuquerque, New Mexico, June, 1988.

2. Digital Equipment Corporation, Guide to VAXclusters, VAX/VMS Version 4, AA-Y513A-TE, Digital Equipment Corporation, Maynard, Massachusetts, September, 1984.

3. Digital Equipment Corporation, VAX/VMS Networking Manual, VAX/VMS Version 4.4, AA-Y512C-TE, Digital Equipment Corporation, Maynard, Massachusetts, April, 1986.

4. Digital Equipment Corporation, Introduction to the VAX/VMS Document Set, VAX/VMS Version 4.4, AA-Z101C-TE, Digital Equipment Corporation, Maynard, Massachusetts, April, 1986.

5. Relational Technology, Inc., INGRES Installation and Operations Guide, Release 5.0, VAX/VMS, Relational Technology, Inc., Alameda, California, August, 1986.

6. Relational Technology, Inc., INGRES/NET User's Guide, Release 5.0, VAX/VMS, Relational Technology, Inc., Alameda, California, April, 1978.

7. Relational Technology, Inc., INGRES/STAR Administrator's Guide, Release 5.0, VMS Version, Relational Technology, Inc., Alameda, California, 1987.

8. Relational Technology, Inc., INGRES/APPLICATIONS: Applications-By-Forms (SQL) User's Guide, Release 5.0, Relational Technology, Inc., Alameda, California, August, 1986.

9. Digital Equipment Corporation, VAX/VMS Network Control Program Reference Manual, VAX/VMS Version 4.4, AA-Z425C-TE, Digital Equipment Corporation, Maynard, Massachusetts, April, 1986.

Distribution:

2340	M. W. Callahan
2342	D. W. Arquette
2360	P. A. Longmire
2500	R. L. Schwoebel
2510	D. H. Anderson
2513	D. E. Mitchell
2520	N. J. Magnani
2526	R. C. Lincoln
2530	G. M. Ferguson
2534	H. M. Bivens
2534	D. H. Jensen
2540	G. N. Beeler
2560	J. T. Cutchen
2600	L. D. Bertholf
2610	D. C. Jones
2612	B. Stiefeld
2612	R. M. Jansma
2630	W. F. Mason
2640	E. J. Theriot
2640A	L. Stans
2643	L. D. Buxton
2644	S. K. Fletcher
2644	C. K. Haaker
2644	M. O'Malley
2645	W. D. Swartz
2647	M. O. Vahle
2647	J. M. Eldridge
2647	L. G. Pierson
2648	D. M. Darsey
2800	W. E. Alzheimer
2810	D. W. Doak
2813	J. R. Yoder
2820	G. Carli
2821	G. F. Quinlan
2825	J. K. Sharp
2825	O. H. Bray
2825	L. T. Davis
2825	D. S. Eaton
2825	R. C. Hall (20)
2825	L. Hernandez
2825	P. F. Martinez
2825	J. L. Orman
2825	L. M. Claussen
2825	N. H. Stevens
2826	A. J. Ahr
2830	G. R. Urish
2850	D. L. McCoy

Distribution:

7200	C. H. Mauney
7220	R. R. Prairie
7223	R. G. Easterling
7223	H. E. Anderson
7250	G. T. Merren
7260	J. A. Hood
8200	R. J. Detry
8230	W. D. Wilson
8231	M. H. Pendley
8270	R. C. Dougherty
GEND	C. W. Wiltshire
GEND	A. Quets
3141	S. A. Landenberger (5)
3151	W. I. Klein (3)
3154-1	C. L. Ward (8) for DOE/OSTI
8524	J. A. Wackerly