Functional Comparison of the Internet Protocol Suite and the OSI Protocol Suite

National Institute of Standards and Technology
Computer Systems Laboratory

February 3, 1994

Contents

List of Figures

List of Tables

1 Introduction

An important trend in Information Technology is the view that the real value of a computer is no longer perceived as being directly related to its inherent computational and memory capabilities; rather, the real value is in the machine's ability to tap an ever growing worldwide mesh of email targets, databases, and other remote resources. In this view, the capabilities of the local computer are secondary and, generally, only of concern when they are found to limit the user's ability to access, use, and manipulate the information mesh. This mesh can be thought of as the beginning of an evolving international information infrastructure — a comprehensive infrastructure will be much more pervasive than the existing mesh[1].

There is compelling evidence of the rapid rise in importance of the information mesh. One strong indicator is the explosive growth of the Internet. Another indicator is the careful integration of Internet access services into new operating system products such as Windows NT from Microsoft. A host of other indicators could be cited, but the Internet growth by itself shouts a clear message to even the casual observer of Information Technology:

> *The benefits of ubiquitous networked resources have been recognized by the public and private sectors worldwide; the need for the ubiquitous network infrastructure is urgent; the need is growing every day.*

In these circumstances, users and purchasers of Information Technology do not particularly care about the details of the underlying network technology; similarly, they don't generally care about the details of the local system that gives them access to the information mesh. What they do care about are the information *services* that can be brought to bear on their everyday problems. These problems exist in the "here and now" so typical users are almost exclusively interested in the services that are available here and now.

The urgency of the problems and the widespread realization of the benefits of large networks have combined to fuel the recent exponential growth in the Internet. This growth has occurred not because there is a consensus that the current Internet provides the ultimate scalable network environment — it most likely does not — but, rather because it is here now and its services are available now.

The rapid rate of growth in the Internet has allowed it to become entrenched in the sense that it is solidly established and widely regarded as a viable foundation for the continued evolution of an information infrastructure during the next 10 – 20 years[2]. This entrenchment is reinforced daily as more users jump on the Internet bandwagon to help solve their burning problems. From a strategic perspective, the entrenchment of Internet cannot be ignored; it is backed by a huge investment in

[1] The existing mesh does not currently service many large groups of potential users such as pre-college education and commerce.

[2] There are acknowledged deficiencies in Internet, e.g., network address space is too small, network management needs to be extended, and native directory services do not scale to worldwide needs. These deficiencies are being addressed and are not perceived by Internet supporters as reasons to move away from the Internet environment in the future.

equipment, software, and training. The large investment will inevitably lead to business decisions that strongly resist any attempt to change to a competing technology when such change is perceived as a discontinuous jump that seriously, or completely, devalues the existing investment without providing significantly superior services at an affordable price.

The entrenchment of Internet is of fundamental importance to NIST because it impacts long-standing strategic policy calling for a federal information infrastructure whose foundation is the OSI protocol suite. This strategic stance is becoming increasingly tenuous as the Internet grows while OSI standards and products continue to struggle through an agonizingly long gestation period. If OSI products, with competitive prices, had been widely available before the Internet Protocol Suite (IPS) became so solidly entrenched, the situation would, perhaps, be quite different. However, the reality is that the OSI suite has taken too long to develop and has not resulted in technical solutions that are universally perceived as significantly superior to those offered by the IPS. It can be argued that this perception is inaccurate but the arguments offered to date have only led to inconclusive religious wars between the backers of the OSI suite and the backers of the IPS.

In any event, even if OSI were generally perceived as superlative and economical, the entire installed base of IPS would not be replaced overnight; strategically speaking, IPS should be thought of as here to stay. Any competing technology must not only offer affordability and overwhelmingly superior services, but it must also be based on a clear migration scheme and must fit smoothly into a larger environment of coexistence with IPS. In this regard it is important to note that the OSI standardization effort began well over a decade ago, long before IPS became entrenched; at that time, there was no clear migration or coexistence problem. OSI-related technology is not, therefore, particularly sensitive to the problems of migration and coexistence.

Hence, NIST's current strategic policy calling for an exclusive infrastructure base of OSI has at least two serious potential problems stemming from the entrenchment of IPS. These problems are summarized in the following two assertions.

1. *Any technology that aspires to replace an entrenched technology must be affordable and clearly superior to the entrenched technology.* The question of the affordability of OSI is still largely unanswered since commercial products for some critical OSI components (e.g., X.500) are just emerging. The fact that the Internet user community is growing rapidly seems to indicate that there is a widely held belief that the current IPS satisfies the most important contemporary needs of users. Even if OSI provides superior services, users aren't generally willing to wait for OSI; they need the services now. Also, OSI would probably have to offer *overwhelmingly* superior services before IPS users would be induced to change to OSI.

2. *Any technology that aspires to replace an entrenched technology must provide reasonable migration paths and must address coexistence issues.* The OSI architecture does not generally acknowledge migration or coexistence.

A primary goal of this document is to provide information about the functionality of the IPS and OSI to foster a better understanding of their comparison as an aid to further understanding the OSI versus IPS debate. This document identifies technical differences between the two suites while emphasizing that in many cases, the two suites can be viewed as complementary.

A secondary goal of this document is to capture information from within the community of computer scientists working in the Systems and Network Architecture Division of the Computer Systems Laboratory at NIST. Over the past several years, this community has participated in efforts to expedite the development of OSI and foster its deployment in Government networks (including the Internet). In the course of pursuing these goals, the division has learned much about the relationship between OSI and IPS; this document summarizes some of this knowledge and provides a repository that can be updated as our knowledge and perspectives continue to evolve.

This document is by no means meant to be the definitive description of OSI, nor the IPS. Several text books have been written on these topics that cover the material in a more general fashion. In particular the following references are suggested for additional information:

- **OSI** - *Open Systems Interconnection, Computer Communications Standards and GOSIP Explained* [1] provides a comprehensive introduction to OSI. *The Open Book* [2] provides a critical examination of OSI standards and their implementations.
- **IPS** - *Internetworking with TCP/IP* [3] provides a through discussion of the IPS architecture and principal protocols. *The Matrix: Computer Networks and Conferencing Systems Worldwide* [4] describes the major components of the operational Internet.
- **OSI and IPS Comparison** - *Open Systems Networking, TCP/IP and OSI* [5] provides an insightful comparison of the protocol suites, their development processes, and history.

2 Comparison of Standardization Paradigms and Processes

2.1 The Internet Protocol Suite as a Standards Paradigm and Process

When discussing the paradigm and process in which IPS standards are developed it is interesting to note that the organizations tasked with their development have always had two dimensions of responsibility: the development of IPS technical standards and the oversight of the operational Internet. The discussion that follows only addresses those aspects related to IPS standards development.

The process by which IPS standards are developed has evolved over time as the scope and importance of the IPS, the Internet and related commercial markets have grown. Figure 1 depicts the current structure of the IPS standardization organizations. IPS standards development is primarily the responsibility of the Internet Engineering Task Force (IETF). The IETF consists of numerous working groups (WGs) that are formed to address specific, narrow focused issues in IPS development or the Internet's operation. As such, IETF working groups can be short lived; some may be created, accomplish their goal, and be disbanded in less than a year.

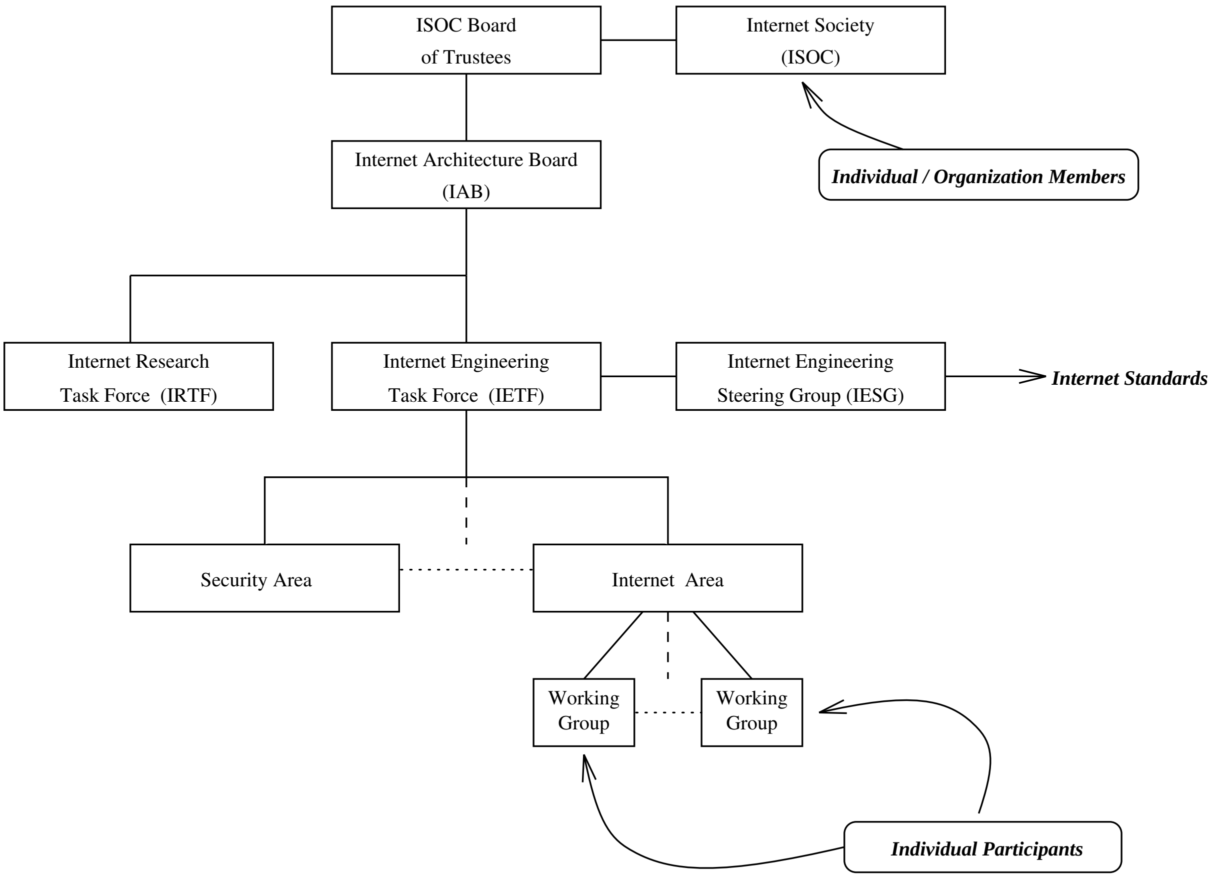

Figure 1: IPS Standardization Organizations

The primary goal of most IETF working groups is the technical development of IPS standards.

Individuals interested in the content and progression of IPS standards attend the 3 to 4 IETF meetings and participate in the deliberations of the working groups. All documents being developed by IETF working groups are required to be available on-line (i.e., via electronic mail or file transfer) and much of the substance of an IETF working group's technical deliberations occurs via electronic mail in between actual IETF meetings.

While the IETF working groups develop the technical specifications, the progression of these documents through the standards process is controlled by an oversight committee called the Internet Engineering Steering Group (IESG). For the purposes of such oversight, IETF working groups are organized into specific subject areas. Each area has one, or more, director(s) who together with the IETF chairperson comprise the IESG. The IESG makes all decisions regarding the progression of specifications through the IPS standards process. Currently there are 10 areas with a total of 12 area directors.

Looking outward, the IETF forms one half of the Internet Society's (ISOC) standards and research infrastructure. The ISOC is a professional society that is "*concerned with the growth and evolution of the worldwide Internet, with the way in which the Internet is and can be used, and with the social, political, and technical issues that arise as a result*". The Internet Research Task Force (IRTF) forms the other half of the infrastructure and is tasked with long term research issues related to the evolution of the Internet. The Internet Architecture Board (IAB), which reports to the ISOC board of trustees, is tasked with the broad oversight and coordination of all IETF and IRTF activities.

2.1.1 IPS Standards Process

In discussing the process by which standards are created it is interesting to examine: what gets standardized, what review and approval procedure must it follow, and who participates in these procedures. The Internet standards process is currently under revision. RFC-1310 [6] and its proposed revision [7] document the Internet standards process.

The primary means of Internet standardization is through *requests for comments* (RFCs). RFCs began as an informal means of documenting technical information for the original ARPA net. Today, the RFC mechanism is part of a formal Internet standards process.

What Gets Standardized: The subjects of RFCs are classified as either technical specifications (TS) or applicability statements (AS). The majority of RFCs are technical specifications that describe protocols, procedures, data formats, or conventions. While TS RFCs describe a general scope and intent for use, they typically do not specify their requirements for use in the Internet. Applicability statement RFCs describe how, and under what circumstances, TS RFCs are to be used to support a particular Internet capability. AS RFCs may additionally specify further restrictions (e.g., subfunctions to be supported, required operations values for parameters) on the use of a TS in particular situations. An AS RFC may also specify a *requirement level* (i.e., required, recommended, elective, limited use, or not recommended) for TS support in particular circumstances.

Not all RFCs are considered candidates for becoming Internet standards. RFCs that are not intended for standardization, or for which standardization activities at this time would be premature, are known as *off-track* RFCs. Off-track RFCs are classified into one of four maturity levels: prototype, experimental, informational, and historic.

Review and Approval Process: Those RFCs that are on the *standards track* move through a series of maturity levels that correspond to stages of the Internet standardization process. Table 1 summarizes the maturity levels of this process. To move from one maturity level to the next requires a *standards action*. A standards action is always requested by the source WG when it feels the specification has met the criterial for progression. All standards actions must be approved by the IESG. The following notes summarize the typical Internet standards review and approval process:

- The final decision on all standards actions rests with the IESG.
- Individuals participate in the working group deliberations (i.e., at IETF meetings and through EMail) to develop a specification for progression through the standards track. Emerging specifications are documented through a series of evolving Internet Drafts.
- Once there is a consensus within the IETF working group that a specification should be submitted to the standards track, the working group chair will recommend to the appropriate area director that an IESG *standards action* advance the current Internet Draft to Proposed Standard status.
- Once in the standards track, the ID will be published as an RFC and assigned an RFC number. With each revision of the document a new RFC will be published.
- At each step along the standards track, it is expected that the community will review and comment on the content, stability, and desirability of the specification.
- At each stage of progression the IESG shall consider the movement of the RFC through the standards track. The IESG may advance and RFC, leave it at its current level, or remand it to a lower level.
- The IESG may form independent technical review and analysis committees to advise them on the progression of RFCs that are controversial or particularly critical to the Internet.

  In the case where multiple specifications are competing to solve the same problem, the IESG may develop a set of additional criteria and schedules that must be met by proposals in order to advance through the standards track.
- A process of conflict resolution and appeal has been defined to address concerns regarding IESG decisions related to Internet standards. As the court of last resort, the IAB may be asked to adjudicate unresolved disputes.

Implementation and operational experience are integral parts of the Internet standards process. To become a Proposed Standard, neither implementation nor operational experience is usually

Maturity Level	Review Period
Internet Draft (ID)	Must be available for review for two weeks prior to submission to IESG.
Proposed Standard	A specification must remain at the proposed standard level for six months.
Draft Standard	A specification must remain at the draft standard level for six months.
Internet Standard	Final approved specification is issued an Internet standard (STD) number.

Table 1: Internet Standard RFC Maturity Levels

required, but such experience is highly desirable. For Draft Standard, at least two independent and interoperable implementations must have been developed, and sufficient operational experience gained to demonstrate the operational appropriateness of the specification. To become an Internet Standard, additional implementation and operational experience is expected.

Who Participates: In general there is no formal notion of *membership* within the general IETF community. Alternatively, one could consider that there are no restrictions on participation. The IETF is defined as "a loosely self-organized group of people who make technical and other contributions to the engineering and evolution of the Internet and its technologies".

IETF Working groups are comprised of any individuals, as opposed to organizations, who attend IETF meetings and participate in EMail discussions. The IETF oversight committees (i.e., the IESG and IAB) do have explicit members. Members of the IESG and IAB are selected by a nominating committee. The nominating committee is selected at random from a pool of volunteers from the IETF community. The ISOC board of trustees is responsible for approving appointments to the IAB. The IAB is responsible for approving appointments to the IESG.

2.2 The OSI Suite as a Standards Paradigm and Process

Figure 2 depicts the current structure of OSI standardization organizations. All OSI standards are developed in Joint Technical Committee 1 of the International Organization for Standardization (ISO) and the International Electronic Commission (IEC). While the scope of ISO/IEC/JTC1 is broadly defined as information technology, this discussion focuses on its development of OSI related standards.

Quite often ISO/IEC cooperates in the development of OSI standards with the CCITT[3]. CCITT is a United Nations treaty organization whose constituents are primarily comprised of large communications providers. Historically, the CCITT and ISO initiated work on OSI standards in parallel. The level of cooperation and coordination in OSI development between the CCITT and ISO has continually increased over time. Originally, this effort resulted in the development of coordinated but independent specifications (CCITT Recommendations and ISO Standards). Over time the coordination process has evolved through the production of *joint standards* (identical technical

[3]CCITT is an acronym for the French translation of the International Telegraph and Telephone Consultative Committee. This committee has since been renamed as the International Telecommunications Union-Telecommunications Standardization Sector (ITU-TS)

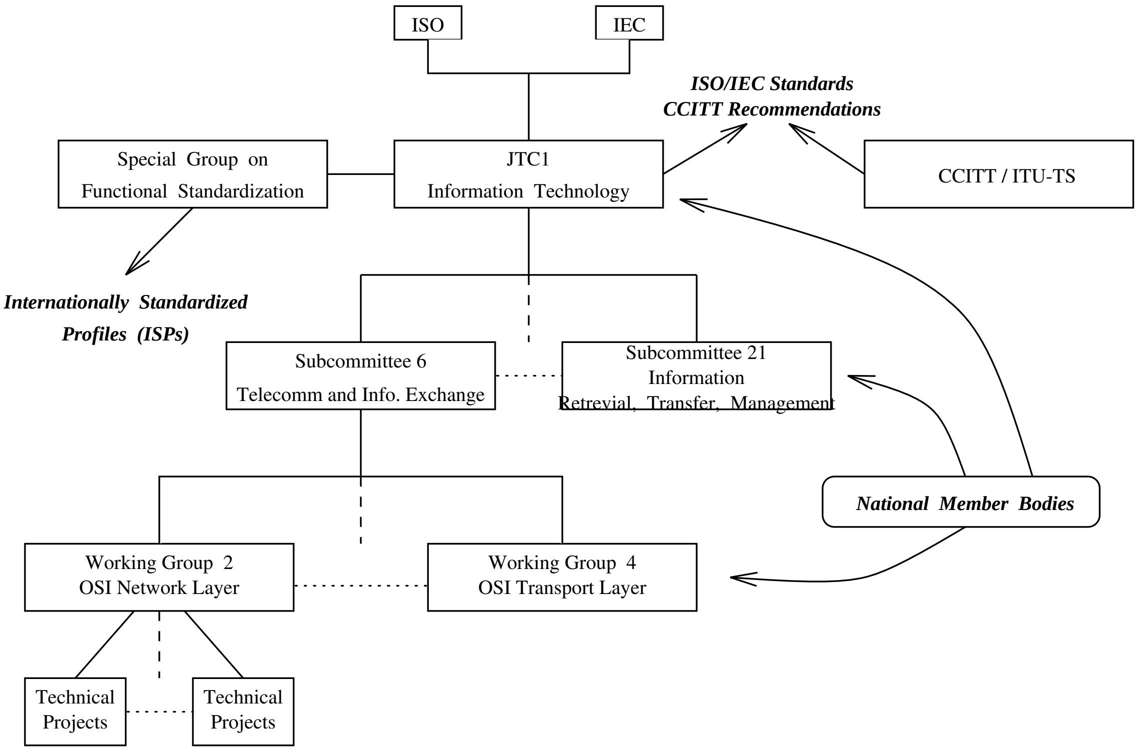

Figure 2: OSI Standardization Organizations

content, potentially different wording and format) and *joint text standards* (identical document used in both communities).

ISO/IEC/JTC1 consists of several subcommitties (SCs) that are assigned broad areas of scope and jurisdiction for the development of OSI standards. Subcommitties consist of several working groups (WGs) that are formed to address specific tasks or areas of responsibility. Within these (WGs) specific projects are created to address individual efforts (e.g., typically a new project is created for each standard to be developed). As an example, work on developing the OSI intra-domain routing protocol was developed as project 41 of ISO/IEC/JTC1 SC6/WG2 (SC6 has responsibility for most OSI lower layer standards, WG2 has more specific responsibility for the Network Layer).

The formal participants in ISO/IEC standardization are the national standards bodies of the countries that voluntarily participate. The American National Standards Institute (ANSI) is the ISO/IEC member representing the United States. When attending an ISO/IEC meeting, individuals are considered as members of a national delegation and formal actions (e.g., votes, formal statements of positions) are made by the national head of deligation (HOD).

2.2.1 OSI Standards Process

The OSI standards process is documented in the ISO/IEC directives on Procedures for the technical work of ISO/IEC JTC1 on Information Technology [8].

What Gets Standardized: The output of the OSI standardization process are classified as either ISO/IEC international standards or technical reports. These *OSI base standards* include specifications of reference models, naming and addressing schemes, services, protocols and data formats. ISO/IEC technical reports (TR) are used for informational, non-normative documents such as tutorials and architectural frameworks. Typically TRs do not contain specific conformance requirements.

In addition to base standard specifications, the OSI standards process contains an additional level of standardization known as *functional standards*. Functional standards, also known as profiles, contain selections of specific OSI standards along with additional refinements (e.g., selection of protocol options, specification of parameter values) appropriate for a particular user community. Several organizations develop OSI functional standards:

- ISO/IEC - develops International Standardized Profiles (ISP) of internationally recognized and recommended stacks.
- Government - OSI profiles (GOSIPs) specify OSI protocol combinations for use in national government networks. Numerous countries (e.g., US, UK, Canada) have GOSIP specifications. Profiles are also developed for government treaty organizations such as the European Economic Community (EEC).
- Commercial - profiles specify OSI stacks for industry and consortia user groups. The Industry Government Open Systems Specification (IGOSS) is a OSI profile developed by a consortia consisting of the US Government, Canadian Government, World Federations of MAP/TOP Users, and the Electrical Power Research Industry.

In the remainder of this section, only the issues associated with the development of base standards are discussed. Further analysis of the process of developing functional standards is beyond the scope of this document.

Review and Approval Process: Base standard specifications move through a series of maturity levels that correspond to procedural stages of the OSI standardization process. Table 2 summarizes the stages of this process. To move from one maturity level to the next requires an explicit voting process within the appropriate ISO committee. The following notes summarize the typical OSI standards review and approval process:

- Any national body (NB) may submit a new work item proposal (NP) for the creation of a new standard. NPs are ballotted and require support by a simple majority of voting members plus commitments by at least 5 national bodies to participate in the work.

  When a new project is approved a *project editor* is chosen also. The project editor typically performs all editing and procedural processing required for the development of a particular standard.

  Frequently, NPs are ballotted with a preliminary specification attached as a proposed working draft.

Maturity Level	Review Period
New Work Item Proposal	Creation of a new project requires a ballot at the subcommittee level.
Working Drafts	Preliminary drafts circulated for review and comment within a working group, subcommittee, or technical committee.
Committee Draft (CD)	CDs are submitted for 60 day ballot at the subcommittee level.
Draft International Standard (DIS)	DISs are submitted for 6 month ballot at the technical committee level.
International Standard (IS)	After passing DIS ballot, the revised text is issued as a full international standard.

Table 2: OSI International Standard Maturity Levels

- Working drafts are reviewed and refined within ISO working group communities on a informal basis until they achieve a level of maturity appropriate to initiate formal standards processing.

  A working group vote is required to register working draft specifications for processing as a committee draft.

- Upon registration as a committee draft, a specification is assigned a number that will remain fixed throughout the progression through the standards process. CDs are ballotted for 60 days.

- ISO/IEC balloting is a formal process by which member bodies submit votes of approval or disapproval for the specification and provide detailed comments for changes in text.

  During international meetings, the ballot results are reviewed and resolved (i.e., the participants in the meeting can agree to changes in the specification that respond to member body comments and allow those nations that voted NO, to subsequently change their vote to YES). Explicit voting criteria are defined for progression to each level of the standardization process.

- A CD that receives a two-thirds majority of approval votes can be advanced to draft international standard status. A specification at the DIS level is submitted for six month ballot period.

- A DIS that receives a two-thirds majority of approval votes can be advanced to full international standard status.

Who Participates: As noted above, the nominal participants in ISO/IEC standardization are national standards bodies. In practice the individual participants at international standards meetings are a subset of the computer and data communications professionals that participate in the corresponding national standards bodies.

In the United States, ANSI delegates its responsibilities as the ISO/IEC member to a hierarchy of accredited standards committees (ASCs) that corresponds, roughly to the ISO/IEC structure. For example, ASC X3 (Information Technology) has a general scope of responsibilities equivalent to that of JTC1. Subcommittee 3 of X3 (known as X3S3) is responsible for data communications issues addressed in JTC1/SC6. Task group 3 of X3S3 (known as X3S3.3) is responsible for the issues addressed in JTC1/SC6 working groups 2 and 4 (OSI Network and Transport layers).

Formally, the participants in ANSI's committees are its member organizations. Organizations may pay dues and send voting representatives to each level of ANSI ASCs. The process by which ANSI ASCs develop US positions and contributions for international representation is through explicit voting of their member organizations.

2.3 Summary of Major Differences in Paradigms and Processes

In comparing the obvious, documented attributes of the OSI and IPS standardization processes one sees more similarities than differences. In general terms of structure of organizations and evolution of specifications, the Internet standards track is roughly similar to that of ISO/IEC. The IETF has a scope of standards development responsibilities similar to that of JTC1. IETF areas are similar to ISO/IEC subcommittees. IETF working groups are similar to specific technical projects within ISO/IEC working groups.

In comparing the review and approval process and its participants, there are several note worthy differences. The ISO/IEC/ANSI process employs a consistent mechanism of explicit voting by member organizations. There is an open opportunity to participate at all levels of the ISO/IEC/ANSI process. The IETF process employs oversight committees to make all final standards decisions. In the past, much discussion has occurred on the "openness", "fairness" and potential legal liabilities of the IAB/IETF standards procedures. These discussions primarily focused on composition and operating procedures of these oversight committees.

While recent evolution of the Internet standards process has made some aspects of these procedures more explicit and open (e.g., the IETF nominating committee) there are still aspects of the process that are left informal and thus subject to question. The IETF credo of "rough consensus and running code" begins to weaken when the community is required to resolve contentious issues. Several recent experiences have indicated that the lack of formal decision processes (e.g., voting) renders the community impotent to make difficult decisions.

Other Observations about the Standards Processes

There are several, less obvious, comparisons that can be made between the two standards processes. These comparisons primarily focus on the culture of the respective communities and are thus more subjective than the preceding discussion.

The Internet standards process addresses both the issues of IPS standards development and the oversight of the operational Internet. This dual responsibility has its advantages and disadvantages.

On the positive side, the focus upon the viability and utility of standards for use in a specific deployed environment has always served as an important reality check for the IPS. Internet standards efforts tend to be more modest in their objectives, typically adding small incremental functionality to existing Internet capabilities rather than attempting to architect the most general, and long term solution possible. The Internet standards process has as part of its procedures, requirements for experimentation and demonstration of emerging specifications.

As the market for commercial uses of IPS technology continues to grow the assumption that the IPS and the Internet are synonymous begins to lose validity. The tight coupling between the development of IPS standards and the oversight of the operational Internet may be strained by large user groups that may want to develop IPS technology for use in private networks (i.e., not part of the Internet).

The OSI standards process has never had close ties to any particular operational environment. While this could be viewed as an absence of artificial constraints, this situation can also be attributed as the critical enabling factor in most negative perceptions of the OSI effort. In particular, the OSI community is often viewed as:

- being obsessed with academic models and architectures;
- attempting to address grand requirements for general applicability and scaling; and,
- failing to address the practical requirements imposed by current technology and deployed systems;

Critics often cite the tangible evidence of this view of the OSI process:

- the plethora of OSI standards;
- an agonizingly slow development process; and,
- the lack of viable commercial products and deployed systems.

The lack of a culture of development and experimentation associated with standards development may lead to decisions without practical foundations. For certain, there is some validity in these views of cause and effect in the OSI community.

There are additional cultural differences in the composition of the two communities and their motivations. The Internet and the IETF are still overwhelmingly dominated by North American concerns. The IPS represents a thriving commercial market place in which all participating organizations profit by the rapid development and deployment of new technology. Organizations voluntarily participate in the IETF activities that benefit from mutual cooperation with other vendors and ignore the activities that do not. Thus, to some extent, the risks in participating in IPS standardization are low and few organizations have motive to actively impede progress to protect their interests.

It can be argued that the OSI community is much more diverse than that of the IETF. The lack of common interests, such as the preservation and evolution of a common deployed networking infrastructure or a thriving marketplace, leads the community to domination by political compromises rather than technical selections[4]. This is complicated by the fact that given the initial objectives, scope, and applicability of the OSI effort organizations are perceived as having greater risks in the OSI standardization process. This results in less cooperation, more protectionist behavior and consequently more political compromise.

[4]One cynic described OSI as "the greatest triumph of form over function"

3 Comparison of Lower Layer Services

3.1 Lower Layer Services Viewed From the Standards

Issues of protocol/service design and selection at the lower layers are critical to the ability to build networking infrastructure. That is, ubiquitous interoperable connectivity at the lower layers provides the enabling infrastructure upon which network applications grow. There are many similarities between the IPS and OSI suites in the lower layer services and protocols. In fact there is a substantial history, in both camps, of using existing work from one suite as a model, or basis, for developments in the other. For example, the OSI internetwork protocol, CLNP, and transport protocols drew heavily from the design and experiences of the IPS Internet Protocol (IP) and Transmission Control Protocol (TCP). The design of recent IPS routing protocols have been strongly influenced by the design of their OSI counterparts.

While at a high level there are many similarities between the IPS and OSI lower layers, there are also some significant technical differences.

The remainder of this section provides a high level comparison of the lower layer services and protocols of OSI and the IPS.

3.1.1 Architectures and Protocols

The basic lower layer architecture and main protocols of OSI and the IPS are depicted in Figures 3 and 4, respectively. Several general observations can be made in comparing these architectures:

1. OSI has adopted the notion of standardizing both a description of the service offered by a layer and the specification of protocols that implement that service. Historically the IPS has not adopted this approach. The services offered to the users are specific to individual protocols.

   In some cases the distinction between protocol and service may not be particularly important in that only one OSI protocol has been developed to implement a specific service (e.g. CLNP and CLNS). In other cases a service specification provides for the formalization and generalization of layer services. As an example, applications operating over the OSI connection-oriented transport service (COTS) should be independent of the selection of which of the 5 protocols/classes is used to provide the service.

2. The IPS and OSI layering architectures are similar. As seen from applications both suites offer a reliable, or connection-oriented, and a unreliable, or connection-less, transport service to applications.

3. There are significant differences in the collection of specific protocols that are provided within the transport and network layers. At the transport layer the IPS has a single means, that is protocol, for providing each of the reliable and unreliable transport services. At the

Network layer, the IPS maintains a consistent inter-networking architecture by providing a single network service and protocol (IP).

At the Transport and Network layers, the OSI suite shows the results of political compromise within a standardization process. The 5 classes of the connection-oriented transport protocol and the two types of Network service represent compromises that were required to gain international standardization. Unfortunately, this proliferation of protocols has greatly complicated the development of OSI infrastructure and often results in islands of non-interoperability among OSI systems. Examples of the ramifications of this situation include:

(a) Several technical means of solving the CO/CL interworking and transport service bridge problems. See [2, 9] for further details.

(b) Various approaches to reflecting the potential lower layer diversity of OSI systems in the naming and addressing infrastructure.

(c) Further duplication of effort in the development of distinct supporting network services (e.g., security protocols, routing protocols) for the Connection-oriented and Connectionless network services.

While the US GOSIP attempts to mitigate these problems of the OSI Transport and Network layers by requiring support of TP Class 4 and CLNP in all systems (i.e., putting it on an architectural par with the IPS), the problems of protocol and service proliferation are still present.

It should be noted that, in isolation, there are some perceived benefits of the operation of TP classes 0-3 over CONS. In particular, in communications environments in which X.25 provides ubiquitous connectivity among systems the operation of TP class 0 or 2 directly over X.25 may be much more efficient than operating TP class 4 and CLNP over X.25.

In what follows, a more detailed comparison of the services of the lower layers is presented. The comparison focuses mainly on the OSI protocols that are most similar to those of the IPS. That is, we compare the protocol combination of TCP/IP with that of Transport Class 4 and CLNP. Beyond the discussion above, no further comparison will be made, specifically TCP/IP and particular transport stacks operating over the CONS are not compared.

3.1.2 Transport Services

Both the OSI and IPS offer two distinct types of transport service: a reliable or connection-oriented transport service (COTS) and an unreliable or connectionless transport service (CLTS). OSI maintains a distinct standard (ISO-8072) that describes the service offered (both COTS and CLTS) by the transport layer. The transport services provided in the IPS are defined by the specific protocols that are used: the Transmission Control Protocol (TCP) and the User Datagram Protocol (UDP).

Several comparisons have been made of TCP and TP4 and the COTS. Broadly speaking level the protocols/services are similar. The design of COTS and TP4 were greatly influenced by the lessons

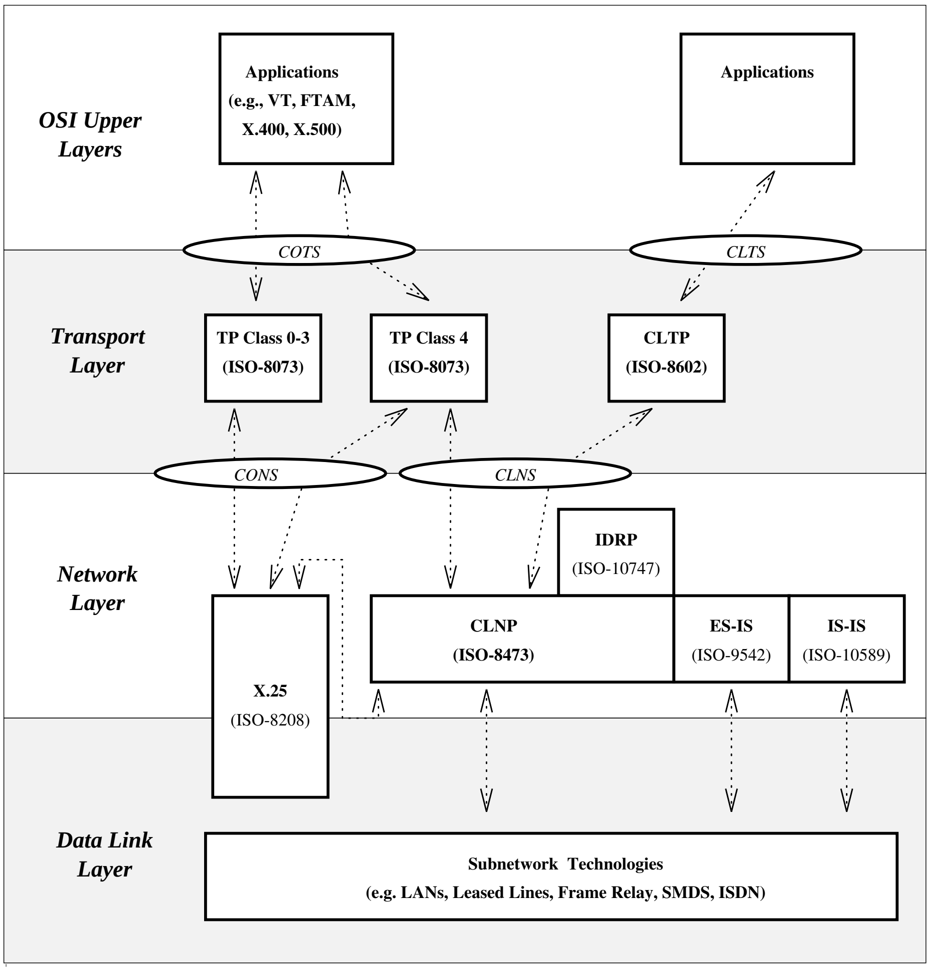

Figure 3: OSI: Lower Layer Architecture and Main Protocols

learned from TCP. There are some areas of difference that are briefly noted below. For a more detailed analysis consult [5].

1. **Packet versus Stream Service** - TCP offers a reliable *stream* service to its users. The service model and protocol mechanisms (e.g. acknowledgment, flow control) operate on unstructured sequences of octets. While this is sometimes viewed as a useful generality, it does place requirements on applications to devise their own means of marking data boundaries (i.e., packets), reconstructing them upon receipt and differentiating data and control information. The *record oriented* service model of TP4/COTS provides a natural solution to these problems.

   It should be noted that TP4's expedited data is a more useful out-of-band signalling mechanism than TCP's URGENT data mechanism.

2. **Connection Establishment/Release Semantics** - The ability to perform *passive opens* and *graceful closes* of connections are often cited as capabilities lacking in the OSI transport layer.

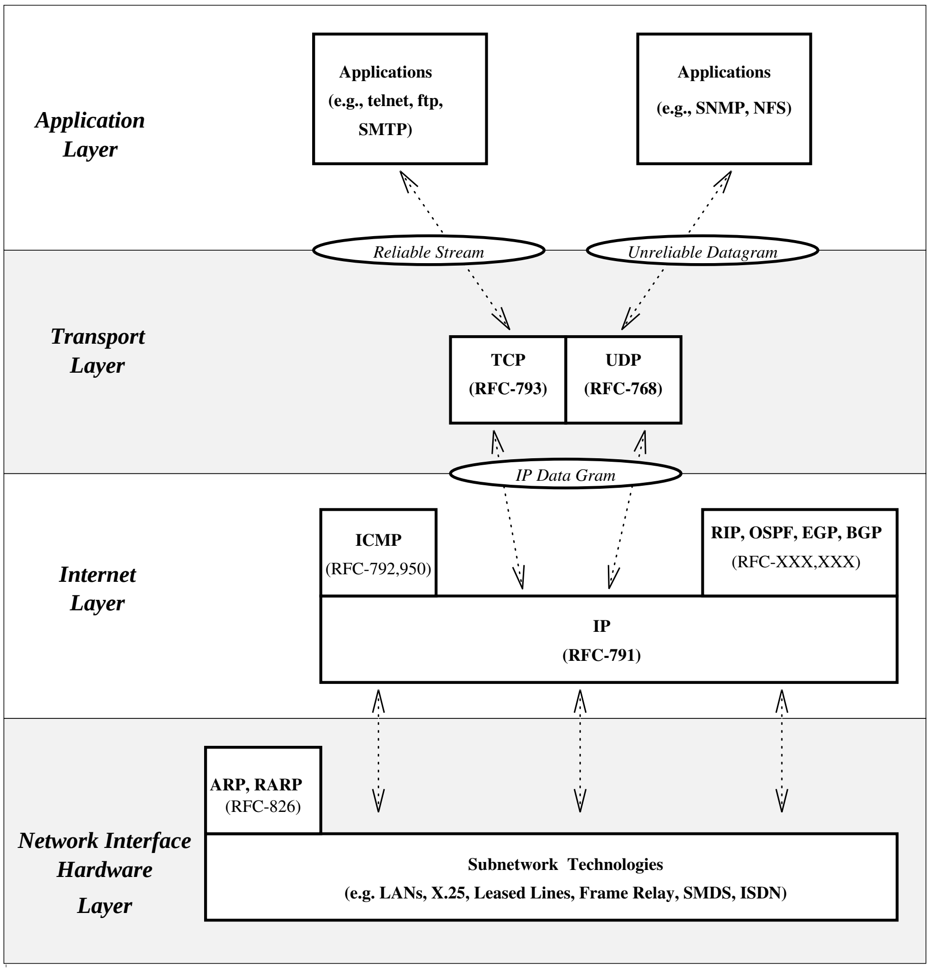

Figure 4: IPS: Lower Layer Architecture and Main Protocols

In practice these capabilities are of little practical significance to applications. They represent more of an application implementation convention than a significant technical difference in the protocols.

3. **Flow Control/Congestion Control** - The basic mechanisms and policies of flow control and congestion control in TCP and TP4 are very similar. There has been significant cross pollination in the state of the art in these areas. Up to date TP4 and TCP implementations employ very similar techniques in the areas of retransmission timer estimation and congestion control/recovery mechanisms. TP4/CLNP offers network layer feedback to the congestion control mechanisms that aren't present in TCP/IP.

   It should also be noted that there are significant differences in the process by which such techniques are standardized in the two communities. The TP4/CLNP flow and congestion control policies are formally documented in the agreements of the OIW and cited by GOSIP while the equivalent TCP documentation exists as published research papers and the BSD public reference implementation.

4. **Detection of Data Corruption** - The data corruption detection mechanisms of TCP and

> TP4 are often compared. TCP uses a weaker, but more efficient checksum algorithm but expands the semantics of the check to include components of the enclosing IP packet (i.e. the *pseudo-header*). The TP4 algorithm is stronger, particularly in detecting transposition errors, but is typically viewed as more computationally complex.

The services offered by the unreliable, or connectionless, transport protocols of both suites are so simple that they are effectively the same. Both protocols provide a simple framing and demultiplexing service for the transport service user.

3.1.3 Network Services

As stated earlier, this section will only address the comparison of the IPS and OSI internetworking protocols, IP and CLNP, and their associated routing and support protocols. Beyond the general architectural issues noted in section 3.1.1, no comparison will be made between the IPS IP and the connection-oriented network service of OSI.

In comparing IPS internetworking with that of OSI it is necessary to consider not only IP and CLNP but their supporting routing and control protocols. In what follows general references to IP and CLNP internetworking are meant to encompass all such protocols.

The IP and the CLNP internetworking protocols are very similar in most regards. The most fundamental difference in the two internetworking schemes is in their forms of Network addressing. The IPS network addressing scheme consists of 32 bit IP addresses that support two, relatively limited, levels of hierarchical routing. IP addresses define a routing architecture that consists of networks (in subnetted environments, "networks" are more properly described as sites, or organizations), subnets within the network, and hosts attached to the subnet. Given that the IPS assigns addresses to individual subnets, IP addresses are really *interface addresses*. That is, hosts have distinct addresses for each network interface. IP addresses are partitioned into 3 *classes*, A, B, and C, depending upon the number of bits allocated to the network portion of the address.

OSI network addresses, commonly called NSAPs, consist of up to 20 octets of addressing information. While, in the abstract NSAPs do not define a routing architecture, the OSI routing protocols have imposed some structure upon their use. Additional structure can be imposed during allocation and deployment to meet the scaling requirements of Internet routing (e.g., see RFC-1237 [10]).

The IPS is currently facing *the routing and addressing problem* in the Internet. The tremendous growth of the Internet combined with the limitations of the 32bit IP address have resulted in problems that threaten the viability of the existing infrastructure. The problem is two fold. First there is a simple problem of address depletion. IP is simply running out of code-points from which to assign network addresses. It is predicted that the Internet will exhaust the available Class B IP addresses by 1995.

The second critical problem is that of an explosion of routing information within highly connected backbone networks. The lack of ability to encode additional levels into the IP routing hierarchy results in IP addresses being treated as flat identifiers for the purpose of routing. While there are proposals to mitigate this problem using a technique known as Classless Interdomain Routing (CIDR), this is only a stop-gap measure.

It is generally recognized that the Internet will have to evolve to using a new internetworking protocol that alleviates this current problems. It is also recognized that the OSI internetworking protocols are capable of efficient operation at scales of deployment that meet the future growth predictions of the Internet.

The following additional observations can be made in comparing other aspects of OSI and IPS internetworking protocols:

1. **Mapping of Functions** - The IPS includes the Internet Control Message Protocol (ICMP) that provides support services (e.g., host redirects, error reports, echo functions) to the IP protocol. In CLNP internetworking these functions are provided by either CLNP itself or the ES-IS routing protocol.

2. **Subnetwork Address Resolution, Address Assignment, and Router Discovery** - The functionality provided by the ARP, RARP and the proposed router discovery protocols are all provided by the ES-IS protocol.

3. **Autoconfiguration** - OSI network addresses are large enough to allow embedding of commonly available globally unique identifiers (e.g. IEEE MAC address, or IP addresses) within NSAPs. IP addresses are not large enough to accommodate this.

4. **Embedding of subnetwork addresses** - Certain routing and configuration efficiencies can be had by embedding subnetwork addresses within network addresses. OSI NSAPs are large enough to allow one to embed subnetwork addresses (e.g. X.25 X.121 address, LAN MAC addresses, ISDN E.164 addresses).

5. **Routing protocols** - The new IPS intra-domain routing protocol, OSPF, drew heavily in its design from its OSI counterpart IS-IS. Likewise the OSI inter-domain routing protocol (IDRP) is a direct extension of the IPS border gateway protocol (BGP).

   In general these protocols are functional equivalents with specific tailoring for their appropriate suites. One significant area of comparison is the extensions specified for IS-IS and IDRP to allow a single, *integrated routing protocol* to carry information for multiple protocol suites. Both IS-IS and IDRP are capable of carrying routing information for IP in addition to that for CLNP. Future additions to these protocols may allow similar service to be performed for the AppleTalk and Novel IPX suites.

   While the utility of integrated routing is often debated, the capability offers an opportunity for economies of scale, both in terms of router and human resources, in environments that must support multiple protocols.

6. **Support of Mobility** - Given that OSI NSAPs are not required to be derived from specific subnetwork points of attachment, the OSI internetworking protocols can better support the mobility of systems without manual reconfiguration.

7. **Granularity of Intra-domain routing** - Both IP and CLNP support two levels of routing within a domain (i.e., site or campus). Both support the notion of partitioning a domain into areas. The routing protocols (i.e., IS-IS and OSPF) support routing within an area and between areas.

   In intra-area routing the IPS routes towards specific subnets and then relies on the ARP protocol for the final forwarding to a specific host. The CLNP routing protocols maintain information on individual hosts within an area. The ES-IS protocol is used to provide the host configuration information to the routers and perform subnetwork address resolution.

8. **Granularity of Inter-domain Routing** - Both the IPS and OSI inter-domain routing protocols provide for the policy based control of routing between domains. IDRP is a direct descendent and extension of BGP.

   The addition of the notion of hierarchical confederations to IDRP allows for better long term scaling of OSI inter-domain routing.

3.1.4 Subnetwork Services

Other than very general design considerations regarding the performance profiles of large classes of user traffic generated using OSI and the IPS, one can safely say that subnetwork services are independent of (i.e., do not belong to) either suite. In general, the design, development and standardization of subnetwork technologies is typically driven by issues that are not specific to any particular higher layer protocol suite.

This being said, the only areas in which OSI and the IPS can be compared is in their *mappings* onto specific subnetwork services and their suitability for operation in environments dominated by specific subnetwork technologies.

Given the drive of a thriving marketplace and a streamlined standardization process, the IPS typically adopts to new subnetwork technology faster than OSI. The OSI suite is typically encumbered by a lag while waiting for the basic technology to progress through international standardization and the complexity of mapping two network services (CONS and CLNS) onto the service. In the definition of subnetwork mapping or *encapsulation* schemes, the IPS has historically been much more in tune to addressing the issues of a multi-protocol environment.

The following specific observations can be made:

1. **Local Area Networks** - There are few significant differences in LAN operation.

2. **Point-to-Point Links** - OSI has failed to develop appropriate multi-protocol encapsulation schemes for operation on point-to-point links. Given the reality of multi-protocol environments, this has resulted in vendor-specific schemes to support multiple-protocols over HDLC LAPB. Recognizing this problem, the IGOSS (and implicitly US GOSIP V3) will adopt the IPS Point-to-Point Protocol (PPP) for operation on point-to-point links.

   The IPS also has devoted greater effort to accommodating the special characteristics of point-to-point links. PPP and other specifications document techniques to perform header compression on bandwidth constrained links, and procedures for dynamically established (i.e., dial-up) links.

3. **General Topology Wide Area Networks** - To some extent the OSI suite suffers from the existence of the CONS in this area. Historically in ISO, there has been a partitioning of the network technology space into CONS and CLNS camps. General topology wide area networks (i.e., primarily X.25) have typically been thought of as part of the CONS camp.

   The IPS, in contrast, has a single network service and interworking paradigm. As a result there has been a greater focus on tailoring the operation of IP and its supporting protocols to the WAN environment. Some developments in the areas of address resolution protocols and operation of OSPF offer WAN optimizations and functionality not found in the OSI CLNS based protocols.

4. **Fast Packet Services** - The IPS has been, in general, quicker to adopt emerging fast packet switching WAN services. Credible mappings of the IPS to Frame Relay, ISDN, SMDS, and ATM either exist or are under development. OSI will continue to play catch up in these areas.

In summary, the most important factor in the state of subnetwork services is the drive of the IPS marketplace and the IPS willingness to adopt any technology that seems commercially viable. OSI, while architecturally better suited in some areas (see discussion of addressing in section 3.1.3), continues to lag behind due to the lack of market incentives and concerns for purity in the process.

3.1.5 Lower Layer Security

Lower Layer Security protocols are relatively new for both the OSI and IPS suites. The OSI Reference Model defines a security architecture for the OSI suite that contains security services for both the transport and network layers. The IPS equivalent for lower layer security services is still in it's earliest stages of development and is at this time developing a security protocol for the network layer.

NIST, NSA and DISA developed the SDNS family of security protocols. The SDNS family of security protocols includes two protocols for providing security services at the transport and network layers, SP4 and SP3, respectively. The OSI and IPS suites have used these protocols and the services they provide as the basis for defining security protocols more relevant to the requirements of each suite. The following sections contain a brief description of the lower layer security protocols that have been developed (or in the process of being developed) for each suite.

OSI Lower Layer Security The OSI approach for lower layer security protocols is to use variants of the SDNS transport and network layer security protocols, SP4 and SP3, respectively. These are being progressed as security solutions for the OSI Transport and Network Layers. Currently the OSI Network Layer Security Protocol (NLSP) is an ISO Draft International Standard , and the Transport Layer Security Protocol (TLSP) is an International Standard.

TLSP extends the services of either the OSI connection oriented or connectionless transport layer protocols to support the following security services defined in ISO 7498 Addendum 2:

- Data Integrity,
- Data Confidentiality,
- Data Origination Authentication, and
- Access Control.

In the OSI reference model, the transport layer has the sole responsibility to provide reliable end-to-end communications between peer end-systems. TLSP services, therefore, nicely complement those provided by the transport protocols.

TLSP functionality is situated near the bottom of the transport layer as illustrated in Figure 1. It consists of a simple encapsulation/decapsulation protocol that protects transport PDUs within a cryptographically secure envelope.

TLSP used with ISO 8073 can protect PDUs on either a per connection basis or an end system oriented basis. If a TLSP entity operates on a per connection basis, a user can distinguish a different level of protection for each connection. Multiple transport connections of differing levels of protection can be multiplexed over the same instance of communications at the network layer. The end system oriented alternative applies the same level of protection to all PDUs, for all connections of the same class between two end systems.

TLSP used with ISO 8602 protects the confidentiality and integrity of individual transport PDUs independently of one another. A significant difference between TLSP used with ISO 8602 or ISO 8073 is that although the integrity mechanism is the same, ISO 8602 supports a limited connectionless integrity service (without recovery), while the connection oriented mechanisms of ISO 8073 allow a more robust integrity services, connection integrity with recovery.

NLSP consists of a simple encapsulation/decapsulation protocol similar to TLSP. NLSP may reside in either end systems or intermediate systems. The capability of NLSP to be used within intermediate systems, either between two intermediate systems or between an intermediate system and an end system, distinguishes it from TLSP.

NLSP has two modes of operation corresponding to either a connectionless or connection oriented service interface. The abstract interfaces and security services provided are dissimilar to the extent that the modes of operation should be viewed as two distinct protocols. Functionally, as shown in Figure 1, the connectionless mode of NLSP (NLSP-CL) resides within the Connectionless Network Protocol and the connection oriented mode of NLSP (NLSP-CO) resides at the top of the Network layer. The connectionless mode of NLSP (NLSP-CL) supports the following security services:

- Connectionless Integrity,
- Connectionless Confidentiality,

- Traffic Flow Confidentiality,
- Data Origination Authentication, and
- Access Control.

The connection oriented mode of NLSP (NLSP-CO) supports the following security services:

- Connection Integrity without Recovery,
- Connection Confidentiality,
- Traffic Flow Confidentiality,
- Peer Entity Authentication,
- Data Origination Authentication, and
- Access Control.

In either mode of operation the data Confidentiality, Traffic Flow Confidentiality, Data Origination Authentication, and Access Control services are essentially equivalent. NLSP-CL integrity is done on a PDU basis and is unable to detect additions or deletions of PDUs; only modification of a PDU is detectable. NLSP-CO can maintain an integrity sequence number space to protect against insertions and deletions.

IPS Lower Layer Security As stated above, the IPS lower layer security is in it's earliest stages of development. This development effort consists of defining a protocol to provide security services for IP version 4. Three proposals are being considered, SP3, I-NLSP, and swIPe.

SP3 as mentioned in the previous section is part of the SDNS set of security protocols and is the basis for most lower layer security efforts, including NLSP, I-NLSP, and swIPe. I-NLSP is an integrated security protocol based on the connectionless mode of ISO's NLSP that provides security services for both CLNP and IP. SwIPe is a stream-lined security protocol designed to provide services for IP Version 4, particularly on low bandwidth networks. All three protocols consist of a simple encapsulation/decapsulation mechanism that protects network layer data in a secure envelope. All three protocols are decoupled from security policy (what and when to protect) and any particular encryption and integrity mechanisms that might be available for use. The three protocols differ in their PDU formats, optional mechanisms, provided services, and functional complexity. It is unclear which protocol is the most appropriate for the security services needed.

Conclusion Both the OSI and IPS suites have developed functional specifications based primarily on the SDNS work. The OSI suite has more maturity associated with these specifications, but both suites have extremely little implementation and deployment experience. With both suites the solution seems to consist of a simple encapsulation/decapsulation mechanism that provides a common set of security services. Since the lower layer services that are provided by both suites are mostly common, the lower layer security service requirements should be essentially identical. Any differences would be the result of relatively minor issues.

3.2 Lower Layer Services: Current State of the Marketplace

The IPS represents a thriving, competitive marketplace and user community. Almost all vendors of host computers, networking components, and distributed application software offer support of IPS protocols. Some areas of the IPS marketplace have become highly price/performance competitive, with numerous vendors selling commodity products.

Vendor support of the OSI suite is improving. In the area of multi-protocol routers, most vendors support CLNP and its associated routing protocols (with the exception of IDRP, due to its recent emergence). The number and quality of the implementations indicate that the marketplace would and could support the deployment of a viable CLNP infrastructure. End system support of OSI lower layers is increasing also. Most major systems vendors support TP4/CLNP lower layers.

In all cases the state of OSI lower layer technology would be improved by more operational deployments. The lack of such experience and the lack of a competitive marketplace hinders the robustness and efficiency of most OSI implementations. Vendors that have made honest efforts to implement viable OSI lower layers have developed products whose quality and efficiency matches or exceeds that found in most IPS products. Unfortunately, the perception of OSI lower layer technology is often clouded by vendors who have made token implementation efforts so as to qualify on bidding government procurements. These implementations are often highly inefficient and defective.

3.3 Lower Layer Services: Current Deployment Status

The IPS TCP/UDP/IP and their supporting protocols provide wide spread connectivity to a large population of users. The extent of IPS deployment is often measured by the size of the Internet[5]. It should be noted that a large percentage of the IPS deployment occurs in private networks, that is, systems that are not part of the Internet.

The following numbers provide a rough characterization of the size of this deployment. These statistics are culled from various sources, by various means and should only be considered as rough characterizations of the size of the IPS deployment.

1. From advertisements into the Domain Name System we see the following statistics: Hosts $\approx$1.5 million, government hosts $\approx$100,000, name domains $\approx$25,000, active countries $\approx$70.

[5] There are various definitions of "the Internet" and ways of estimating its size (e.g., number of DNS registered hosts, number of hosts reachable by IP from the NSFNet backbone, number of hosts reachable through SMTP, number of attendees at the IETF meetings). The past chair of the IAB once defined inclusion in the Internet as: "You are part of the Internet if you think that you are!".

2. From statistics from major backbone networks there appear to be $\approx$10,000 sites reachable directly through use of the IP protocol. Several major transit networks of the Internet (e.g., NSFNet, ESNet, NSI) and various regional networks offer CLNP switching services. In many cases these are not production services but their existence is beginning to establish a base of OSI capable equipment and trained personnel.

3. Recent market surveys characterize the IPS as $\approx$25% of the government networking environment. The same survey characterizes OSI as $\approx$2% of the government networking environment.

The deployment status of OSI and the IPS cannot be characterized by system counts alone. There is a large human investment in technical training for the management and administration of IPS networks. The number of personnel that are fluent in the management and administration of OSI networks is probably proportional to the relative size of its installed base.

4 Comparison of Upper Layer Services and Applications

4.1 Electronic Mail

4.1.1 Service As Described in the Standards

IPS Standard Electronic Mail

The IPS electronic mail application is known as Simple Mail Transfer Protocol (SMTP). SMTP is used in conjunction with RFC-822. RFC-822 defines a commonly implemented message format used in Internet mail, USENET, BITNET, and other messaging systems.

The SMTP model contains two primary components: a sender-SMTP and a receiver-SMTP. To transfer a message, a sender-SMTP establishes a single full duplex transmission channel with the receiver-SMTP. The sender-SMTP sends ASCII commands across the channel, which are processed and replied to by the receiver-SMTP. If the two hosts are connected by the same transport service (e.g., TCP/IP) the transmission of mail is direct. Otherwise, the mail is relayed through a gateway between two or more networks (e.g., between the Internet and BITNET).

RFC-822 was initially written to convey ASCII text. The Multi-media Internet Mail Extensions (MIME) standard was developed to expand the message format to include non-ASCII text and multimedia attachments. MIME uses a two level (type/subtype) mechanism to identify attachments. The top level has eight types defined, including nested messages, image data, audio data, video data, private data (for bilaterally agreed types), and others. The subtype level further defines the type, for example, *image/g3fax*. MIME offers some services that are not present in many off-the-shelf X.400 products. These services include:

- rendering attachments sequentially or in parallel (e.g., audio and video),
- sending the same attachment in different forms and letting the recipient application choose the appropriate form,
- encapsulating partial messages as a content type, allowing large messages to be sent in fragment form, and
- indicating that a message is external, e.g., enclosing instructions to retrieve it using FTP.

SMTP contains many services similar to the X.400 IPM services. Two services included in X.400 which are not present in SMTP are the Message Store and acknowledgments. SMTP contains no protocol or entity dedicated to message storage. Also, SMTP provides no standardized acknowledgment. Most SMTP systems, however, use a *Postmaster* mailbox to generate automatic non-receipt notices.

OSI Standard Electronic Mail

The OSI standard for electronic mail is specified in the CCITT (Consultative Committee for International Telephone and Telegraph) X.400 Set of Recommendations. The Recommendations define a general purpose, store-and-forward messaging service. The system providing the OSI electronic mail service is called the MHS (Message Handling System).

Nearly all MHS services are provided by its two primary entities: the MTA (Message Transfer Agent), and the UA (User Agent). MTAs transfer messages from an originator to the recipients on a cooperating, store-and-forward basis. UAs interface with users, assisting with message submission and reception. Depending on the type of UA, functions relating to the creating, editing, reading, and storing of messages may be provided.

The advantage of dividing electronic mail functionality into two entities is that one MTA can transfer messages containing multiple syntaxes. This concept is realized at two levels: the message level and the body part level. Two standard message types are presently defined by the CCITT: the IPM (Interpersonal Message) and the EDIM (EDI Message). The main purpose of the interpersonal message is to convey memo-like information between two or more people. The main purpose of the EDI message is to convey business transactions between two or more computer applications. Because the users of the two message types are different (i.e., a person versus a computer application), a UA is defined for each message type. One MTA, however, can support both IPM-UAs and EDI-UAs.

Interpersonal messages and EDI messages may comprise multiple body parts. These body parts can contain any syntax, including text, FAX, voice, telex, and others. In addition to these predefined types, X.400 includes an extension mechanism for identifying body part types such as word processing documents, spreadsheets, and others. An MTA may convert between body parts (e.g., between IA5 text and telex). A UA can register with an MTA the body part types that may be delivered, and request the MTA to perform conversions.

X.400 supports two types of acknowledgement messages. The first is a delivery report. It is generated when an MTA delivers a message to a mailbox or gateway. The second is a receipt notification (for IPM messages) or an EDI notification (for EDI messages). It is generated when a user reads a message or when the user or UA takes some definitive action. The message originator can request either, neither, or both types of notification.

In addition to the UA and the MTA, the X.400 Recommendations define two optional entities: the MS (Message Store) and the AU (Access Unit). MSs provide a repository for messages, from which a UA can perform database style retrievals. An MS may also act on behalf of a UA to submit and accept delivery of messages. AUs link the MHS to other communications systems. For example, the PDAU (Physical Delivery Access Unit) enables MHS users to send messages which are transcribed and delivered to recipients served by a postal system.

When submitting a message to the MHS, an originator specifies the address of the recipient(s). An X.400 address, known as an O/R (Originator/Recipient) address, comprises a complex, hierarchical set of attributes. A valid O/R address must contain a country name, an ADMD (Administration Management Domain) name, and at least one of the following attributes: a PRMD (Private

Management Domain) name, an organization name, an organizational unit name, a personal name, or a domain defined attribute. In lieu of an O/R address, an originator may provide an X.500 Distinguished Name.

An O/R name may reference a Distribution List (DL) as well as a single recipient. DLs enable an originator to specify a group of recipients with a single O/R name. There is no syntactic difference between an O/R name identifying a DL and an O/R name identifying a single recipient.

4.1.2 Electronic Mail: Current State of Products

IPS Electronic Mail Products

Most vendors marketing the UNIX operating system include SMTP as part of the operating system. This is in contrast to X.400 products which are not typically bundled into operating systems packages. For this reason, SMTP is widely available from most major vendors and many small software houses. Public domain versions of MIME are also available. It is questionable, however, if MIME and Privacy Enhanced Mail (PEM) interoperate.

OSI Electronic Mail Products

CCITT released the first set of X.400 Recommendations in 1984. Following their four year publication cycle, the second set of Recommendations was released in 1988. The 1988 Recommendations added significant functionality to the MHS. The major enhancements included the Message Store and X.400 security services.

Although 1984 based X.400 products are available from many vendors, it was not until 1992 that the first commercial 1988 products were marketed. Recently, more 1988 implementations have become available; however, these implementations are minimally conformant. Most 1988 X.400 products do not include the X.400 security services, and most use proprietary means for storing messages (i.e., do not use P7, the standard Message Store protocol). In addition to commercial products, one public domain X.400 implementation is available. This implementation is called *PP*, and is based on the 1988 version of X.400.

Most X.400 implementations support interpersonal messaging. The number of implementations supporting EDI messaging (i.e., based on the X.435 Recommendation, published in 1990) is limited. Since the requirement for X.400 security is far greater with EDI messaging, due to the sensitive nature of the information being transferred, EDI messaging is acting as a catalyst for the implementation of X.400 security services.

To increase portability for X.400 products, X/Open and the APIA (Application Program Interface Association) have developed a set of APIs. These APIs define a standard interface to an API service, which is typically provided by an MTA. Using the APIs, non-mail applications, such as word processing applications, can be enhanced to include electronic mail functionality. Also, UA and MTA products can be procured from different vendors. Few X.400 products include these

APIs.

See section 4.4.3 for a recent market analysis of X.400 availability and interoperation.

4.1.3 Electronic Mail: Current State of Deployment

Most Federal agencies and private companies use some type of electronic mail product. To provide X.400 access to a group of users, the easiest and most commonly implemented solution is to install an X.400 gateway. This gateway converts X.400 messages to and from the proprietary format. The disadvantage of this solution is that the proprietary user interface is seldom modified to accommodate the additional X.400 services. As a result, some X.400 services are not made available to the users.

Recent NIST experience in working with other Federal agencies indicates that X.400 access is available in most agencies. Access, however, is generally limited within the agencies to a select group of users. These users are typically the technical personnel responsible for implementing X.400 within the agency, or users who have procured a mailbox on a public mail carrier.

Deployment of SMTP is far more widespread than X.400. SMTP is available from most major vendors and many small software houses, and is included as part of the UNIX operating system. Thus, most government agencies, universities, and private companies have access to SMTP and RFC-822 Internet mail.

4.1.4 Electronic Mail: Security Issues

X.400 security features were introduced in the 1988 X.400 Recommendations. They can be categorized into three general services. First, they allow various MHS entities to authenticate their identity. Authentication occurs at connection establishment time via the exchange of passwords or signatures. Second, they allow the detection of any modification of a message; this service is called *message integrity* and is provided through the use of a digital signature applied to the message content. Third, messages are protected against unauthorized disclosure. Message confidentiality is provided by encrypting the message content. An additional level of security can be provided through the use of security labels. Security labels can be used to control the submission, routing, and delivery of messages.

Many of the MHS security services require security capabilities within the UA, but not the MTA (e.g., the UA encrypts the message content). Some security services, however, require MTAs with security capabilities (e.g., if requested, an originating MTA must generate and return proof of submission to the originating user). Some of the MHS security services apply to the MS (Message Store) as well as to UAs and MTAs, such as services involving the exchange of security labels. In general, however, the MS is transparent to security features that apply between the originating and recipient UAs.

Many of the MHS security services rely on encryption techniques. These services are flexible regarding whether public-key or symmetric encryption techniques are used, and more specifically, which algorithms are used. Some services, such as the non-repudiation services, require a public-key encryption algorithm.

The X.400 security services require that every MTA handling the message conform to the 1988 Recommendations. MTAs based on the 1984 Recommendations are not capable of relaying the security extensions. Since the public mail carriers presently use 1984 based MTAs, X.400 security services cannot be used in conjunction with the X.400 services provided by the public mail carriers.

The IPS electronic mail security service, PEM (Privacy Enhanced Mail), is similar to the X.400 service. Its major capabilities comprise the equivalent of X.400's content integrity service using signatures and content confidentiality service using encryption. Although the service is not as rich as X.400's, the only major omission is a counterpart to X.400's non-repudiation of delivery service.

4.1.5 Electronic Mail: Coexistence and Convergence Issues

The requirement for a standard means to gateway X.400 and SMTP implementations is satisfied by the Internet Request For Comments 1327, titled "Mapping between X.400(1988)/ISO 10021 and RFC 822". RFC 1327 allows users of X.400 mail systems (1984 and 1988) to communicate with users of RFC 822 mail systems and vice versa. Specifically, it describes a set of mappings that enable interworking between systems operating the two mail applications.

A number of RFCs describing gateways between X.400 and SMTP mail systems have been developed. The earliest, RFC 987 (June 1986) is based on the 1984 version of X.400. RFC 1026 specifies a number of additions and corrections to RFC 987. RFCs 1138, 1148, and 1327 (the most current) deal with 1988 X.400. All the above RFCs map X.400 IP (Interpersonal Messages) to RFC 822 messages and vice versa. None of the RFCs specify mappings for mail security services.

Many gateways between X.400 and RFC 822 are currently available from vendors and public mail carriers.

ISO/IEC	CCITT	Title
7498	X.700	Management Framework
10040	X.701	Systems Management Overview
9595	X.710	Common Management Information Services (CMIS)
9596-1	X.711	Common Management Information Protocol (CMIP)
10165-1	X.720	Structure of Management Information (SMI)
10165-2	X.721	Definition of Management Information (DMI)
10165-4	X.722	Guidelines for the Definition of Managed Objects (GDMO)
10164-1	X.730	Object Management Function
10164-2	X.731	State Management Function
10164-3	X.732	Attributes for Representing Relationships
10164-4	X.733	Alarm Reporting Function
10164-5	X.734	Event Report Management Function
10164-6	X.735	Log Control Function
10164-7	X.736	Security Alarm Reporting Function
10164-8	X.737	Security Audit Trail Function

Table 3: ISO/CCITT Network Management Documents

4.2 Network Management

Over the last decade, industry has seen a rise in the numbers of large networks, consisting of mainly multi-vendor, heterogeneous devices. As these networks grow in complexity and organizations become increasingly dependent upon them for daily operations, the need to administer, operate, and maintain them in a reliable manner has become a key issue. As a result, two major sets of standards have emerged for managing networks.

One of these standards comes from the Internet cummunity and is defined by three RFCs:

RFC-1155 - Structure and Identification of Management Information for TCP/IP Networks

RFC-1157 - Simple Network Management Protocol

RFC-1213 - Management Information Base for Network Management of TCP/IP-based Internets

These RFCs are collectively referred to as SNMP and constitue the network management application services in the IPS.

The ISO/CCITT network management is defined by a set of jointly published International Standards and Recommendations. These documents are listed in Table 3. For the purposes of discussion, this set of standards are collectively referred to as CMIP. The services described in this set constitute the network management application services in the OSI suite.

In the following sections, these two sets of standards are compared and contrasted, paying particular attention to the services as described in the standards, the current state of products, the current state of deployment, and security issues.

4.2.1 Service As Described in the Standards

Both the IPS Management model and the OSI Management model utilize a client/server paradigm, where the servers contain managed resources and clients request services from the resources. Here, the client is known as the *manager* and the server is known as the *agent*.

The object model for CMIP is defined in much the same fashion as object oriented data models. In contrast, the IPS standards model definition is more aligned with a relational data model. This difference in definition has effects that are relevant to flexibility and extensibility.

In both SNMP and CMIP, the relevant standards can be combined into two major categories. Management Information is defined as the set of standards representing the structure of the information that is to be managed. Management Functions are defined as the service and protocol for exchanging management operations. CMIP defines a set of common tasks called System Management Functions, which are also grouped into this second category.

The following sections describe, in detail, the differences in management information, management functions, and underlying communications between the two management standards. Finally, we look at evolving work with regard to each of these standards.

Management Information

This section provides an overview of the structure of management information for SNMP and CMIP. A comparison of this information structure with regard to object-oriented methodologies (i.e. encapsulation, classes, inheritance, and polymorphism), naming, and data representation indicates that although SNMP and CMIP are similar in their overall object-oriented approach toward network management, they differ greatly in their object definition and the operations that may be performed upon these objects.

The IPS Structure of Management Information (SMI) standard provides a methodology for abstractly defining managed objects, which in this model are variables that represent atomic data elements. These managed objects are collected into a database, called the Management Information Base (MIB). Each managed object is described using a standard notation, Abstract Syntax Notation One (ASN.1), and contains the following:

- Name: unique identifier
- Syntax for the type
- Access: read-only, read-write, write-only, not-accessible

- Status: mandatory, optional, obsolete

The name of a managed object is an object identifier, which is a sequence of non-negative integer values, separated by dots, that traverse a tree. In the case of the Internet, objects are defined under the MGMT subtree:

```
mgmt OBJECT IDENTIFIER ::= { iso(1) org(3) dod(6) internet(1) 2 }
```

The unlimited depth and breadth of the tree allows an infinite number of names for managed objects. Managed objects are only templates; it is the instances of objects which are manipulated. Thus, both the object name and the instance identifier must be specified in order to manipulate an object. A side effect of this naming scheme is that variable names are only unique within a system. Although this problem could be alleviated by qualifying variable names with globally unique system names, the SMI does not provide a method for maintaining system name information.

In OSI management, a managed object is characterized by its operations, notifications, attributes, and behavior, all of which are defined at the object boundary. Similar objects are grouped into classes, which may be derived from previously defined classes using strict multiple inheritance. Although the OSI naming tree may appear to be similar to the IPS OBJECT IDENTIFIER registration tree, they should not be confused. Managed object classes and their properties represent types, not names, and are assigned OBJECT IDENTIFIERS from various arcs in the registration tree, depending upon the defining organization. Names are hierarchical, typed, and global; they are based on the X.500 distinguished name. This hierarchical nature defines a naming tree where each node represents an object instance. A name binding template defines a place for each object instance in this naming tree, ensuring global uniqueness for each object instance.

In the Internet SMI, the ASN.1 data types that may be used are limited to four out of the twenty-six defined; specifically, these types are INTEGER, OCTET STRING, OBJECT IDENTIFIER, and NULL. Although this simplicity allows an agent to support a subset of ASN.1 data types, it does not provide for some otherwise useful data types. Overloading of these basic data types would allow the additional functionality, but the manager and the agent must agree on the semantics on a variable by variable basis.

In OSI management, objects are again defined by ASN.1. However, CMIP does not place any limits on the ASN.1 data types that may be utilized. Particularly useful data types, such as those relating to character strings, time, sequences, and sets are available, taking advantage of the self-describing capabilities of ASN.1. The differences between the two methods represent a trade-off between simplicity and flexibility.

The Internet SMI allows MIB variables to be collected into lists and tables, but does not provide a formal mechanism for encapsulating related variables. Even though variables may be aggregated into tables or lists, the SMI does not specify that these variables must be related in any fashion, and, furthermore, the SNMP protocol does not contain any list or table oriented operations. It is, however, possible for a variable to encapsulate behavior, as the SNMP SET operation allows side-effects. For example, setting a variable to zero may have the effect of cycling the power on a particular device. In OSI management, objects are characterized by their behaviors, attributes, operations, and notifications

A principle advantage of the class concept in object-oriented paradigms is the ability to extend and combine previously defined interface specifications in creating new interfaces. In the IPS model, every MIB variable instance contains its class name, and the instance names are dependent upon their location in the MIB. A variable type cannot be reused to represent a similar variable, even though they may share many semantics. In the OSI model, managed object instances are independent of the class definition, and hence, may be reused in different places in the naming tree.

Inheritance allows specification reuse, which typically implies code reuse. SNMP variable definitions cannot be reused when defining other MIB variables. OSI management supports strict multiple inheritance as a means for defining new object classes that may be an extension of an existing class. A subclass must inherit all the attributes, actions, notifications, and behaviors of its superclass. Since these properties are assigned a globally unique identifier, it is possible for a subclass to inherit properties from more than one superclass.

Polymorphism provides a mechanism for migration and coexistence between differing versions of class definitions. In the Internet SMI, migration is handled by obsoleting old variable types in favor of newly created variable types. There is no mechanism for relating the new type to the old. In OSI management, allomorphism, which is a restricted form of polymorphism, is supported. Allomorphism allows an object instance of one class to be managed as if it were an instance of another class.

Management Functions

The IPS management protocol defines a set of functions that allow a manager to obtain and manipulate resources in an agent. The following protocol data units (PDUs) must be supported:

- Get_Request: Retrieves a MIB variable
- GetNext_Request: Retrieves, via traversal, the next MIB variable
- Get_Response: Response to a get, getnext, or set operation
- Set_Request: Change a MIB variable to a specified value
- Trap: Indication that something has happened

Typically a manager will generate an SNMP Get, GetNext, or Set operation and an agent will generate a Get_Response and a Trap operation. Although the Trap operation is defined, it is not heavily used. Most of the management operation is controlled by the manager, who will poll the agent for information. This poll-directed form of management is fundamental to SNMP. Occasionally, an agent will send a trap to a manager, which may in turn, cause the manager to obtain more information.

The OSI management protocol, CMIP, defines a larger set of PDUs than SNMP. The following PDUs are supported:

- Get: Retrieves management information
- Set: Conditionally modifies management information
- Action: Request that an action be performed
- Create: Create a managed object
- Delete: Delete a managed object
- Event-Report: Report an event about a managed object
- Cancel-Get: Cancel a previously issued Get operation

Each of these PDUs have a request form as well as a response form. Typically, a manager will generate CMIP Get, Set, Action, Create, Delete, and Cancel-Get request PDUs and an agent will generate the corresponding response PDUs. The managing system will generate an Event-Report request PDU. A CMIP Event-Report is somewhat similar to an SNMP trap in that both indicate that an important event has occurred. Unlike the SNMP trap, the CMIP Event-Report is heavily used in OSI management, in both confirmed and unconfirmed modes. The frequency and amount of information that a CMIP agent will pass in an Event-Report is configurable by the manager, allowing control over the use of network bandwidth for management operations.

CMIP scoping and filtering allow any portion of a MIB to be accessed by a single management operation. A filter specifies the conditions that must be met before an operation can continue and scope defines the set of managed object instances to which the filter will be applied. This powerful wildcarding can greatly reduce bandwidth requirements, but is done at a significant cost in agent size and complexity. Scoping and filtering is optional, and can be negotiated at association establishment. Both best-effort and atomic synchronization options are supported in CMIP. Best-effort implies that the managing system will attempt to perform the indicated operation on each managed object and attribute selected. Atomic implies that an operation will be performed if and only if it can be performed successfully on all the selected managed objects.

OSI defines several System Management Functions (SMFs) that provide capabilities in addition to those defined by CMIS/CMIP. Currently, there are standards defined for Object Management, State Management, Attributes for Representing Relationships, Alarm Reporting, Event Reporting, Log Control, Security Alarm Reporting, and Security Audit Trail Functions.

Management Communications

There is currently a difference of opinion as to whether network management protocols should run over connection-less versus connection-oriented services. SNMP is designed for use over a connection-less protocol for several reasons. SNMP proponents believe that a connection-less protocols allow management traffic to get through, even on congested networks. It is also believed that at least for sets and gets, a response from the agent is sufficient to guarantee reliability of delivery. On the other hand, CMIP proponents believe that management protocols should be run over a connection-oriented service. Here, reliable delivery is guaranteed. Some of the ramifications of a connection-less service are summarized below.

The SNMP protocol operates over a connection-less service via a well known user datagram protocol (UDP) port and the internet protocol (IP). The underlying connection-less service provides for signalled data corruption and a maximum packet size of 484 octets. However, data loss due to lost or discarded packets is not indicated. Data may be lost for a number of reasons, including an unavailable receiver, network congestion, or loss of connectivity. Additionally, the connection-less model does not afford the manager and agent an opportunity to negotiate such features as capabilities and security parameters.

Future Developments

Currently, SNMP seems to be the protocol of choice for managing the resources associated with TCP/IP networks. Recently, a new version of SNMP, deemed SNMPv2, has been making its way through the Internet standards process. This new version promises enhanced management capabilities, including Secure SNMP, bulk data transfer, primitive manager-manager capabilities, and better managed object definition.

ISO/CCITT also continues work in extending the current OSI standards and recommendations. Work is on-going in the area of additional SMFs for performance, security, accounting, and configuration management. Work is also progressing with regard to additional managed object definitions, management profiles, and ensembles.

4.2.2 Network Management: Current State of Products

The IPS management protocol has had widespread implementation by many vendors over the last several years. The cost to implement has been relatively low due to the deliberately simple design of the protocol and the MIB. However, this simplicity has led to limits in addressing, requirements for event reporting, and problems scaling to large networks. Although initial implementation costs may have been relatively low, future development may cause this to change.

The OSI management protocol has been relatively slow to be deployed, with only a few companies having products available. With the release of OmniPoint in the last year, it is possible that more vendors will implement ISO management. The overall generic design of this approach implies greater complexity and hence increased costs. However, future expansion should not be nearly as costly as the initial outlay.

4.2.3 Network Management: Current State of Deployment

The IPS and OSI network management protocols can and have been deployed over alternative transports. For example, there are several mixed protocol stack approaches defined, including CMIP for the Internet, CMIP over LLC, and SNMP over OSI. Each of these approaches exist today, and allow management applications to be deployed over existing transports.

4.2.4 Network Management: Security Issues

Security of management services differs greatly between SNMP and CMIP. In SNMP, security is handled via the community name string. When an SNMP agent receives a PDU request, it checks the community name string to determine whether or not the requested operation is allowed. Generally, the ability to read, modify, append, create, and delete objects may be required on a managed object basis. That is, there may be a requirement to give one user modify access to a particular object, and another create and delete access. Currently, this level of control is not supported in SNMP. However, a proposed Internet Standard, Secure SNMP, specifies both authentication and access control mechanisms. Additionally, since SNMP uses UDP to provide a connection-less service, it must rely on the security of UDP for network communications. As of yet, no security mechanisms have been defined and/or planned for UDP.

OSI management occurs over a connection-oriented service provided by OSI Association Control Service Element (ACSE). The ACSE standard provides a protocol for peer-entity authentication that can be negotiated at association establishment. This authentication is based on simple credentials including a user name, password, and optional time-stamp parameters, all of which may be encrypted. The authentication information is compared with a predefined set of usernames and passwords to authenticate the peer-entity. CMIP provides an access control parameter in operations that can be used to supplement the peer-entity authentication. A draft standard, ISO Objects and Attributes for Access Control, defines procedures for both the security of management and the management of security. Data confidentiality and data integrity are supported using cryptographic techniques via the Transport Layer Security Protocol (TLSP). It is envisioned that the Network Layer Security Protocol (NLSP) will provide additional network security measures for end-to-end connection-less and connection- oriented services in the future.

4.2.5 Network Management: Coexistence and Convergence Isssues

Coexistence and convergence strategies have been defined to allow IPS and OSI management applications to exist in a distributed environment. Some early attempts at coexistence brought about straight forward solutions such as mixed stacks, dual stacks, and common APIs.

In a mixed stack approach, management applications are deployed over alternative transports. For example, RFC-1006 [11] which allows OSI applications to run in the IPS environemnt, can be utilized to deploy OSI management, CMIP, in the Internet. In the opposite direction, RFC-1283 [12] defines a mechanism for deploying SNMP over OSI. Although both of these methods provide for simple coexistence, they do not provide a mechanism for managing resources defined by one set of standards with a protocol defined by the other set of standards.

Another method of achieving simple coexistence is through the use of dual protocol stacks. Here, implementations of both protocol stacks in the same device allow for the processing of both IPS and OSI management messages. This approach may be costly in simple devices, both in terms of implementation and performance degradation. One strategy, as pointed out in [13] would be to implement the dual stack in the manager, while implementing only the native management protocol in the agent.

A common application programming interface (API) has been defined to operate over a dual stack. The X/Open Management Protocol API (XMP) [14] defines a common set of verbs and data structures for interfacing to both SNMP and CMIP services. However, use of the XMP API requires that the programmer have significant knowledge of the underlying protocols and of the differences in naming, services, and error handling. In some cases, these problems can be less worrisome if the programmer uses only the set of services that are common to both protocols, but this approach limits the functionality of the management applications to their least common denominator.

More recent work has focused on true interworking of the IPS and OSI management protocols. This interworking can be achieved by means of MIB translations, protocol translations, and service emulation.

Translating between Internet and ISO/CCITT MIBs involves mapping a MIB defined using the IPS SMI to ISO GDMO and vice-versa. This process can be defined using either a direct, or "one-to-one", approach or an abstract, or "many-to-one" approach. In the direct translation method, each aspect of the original MIB is represented in the translated MIB, with some exceptions for differences in style. Translation procedures for this method have been documented in [15]. These procedures have recently been validated through the release of a automated public domain translator that will take as input any IPS style MIB and produce its GDMO equivalent. Additionally, a direct translation of the IPS MIB-II to ISO GDMO style format has been performed [16] using these translation procedures [15]. Translation procedures have also been defined in the reverse direction, that is, ISO GDMO to IPS SMI MIBs[17]. MIBs translated using both of these procedures may be used when it is necessary to manage resources using differing protocols, to provide uniformity in dual stack environments, and in support of proxy or gateway architectures.

Protocol translations provide a mapping between SNMP PDUs and CMIP PDUs. This mapping is relatively straight forward for simple operations, such as gets and sets on individual attributes. More complex operations, such as CMIS scoped gets/sets and SNMP GetNexts, cannot be directly translated. Rather, these services must be emulated. Protocol translation is most effective when combined with MIB translations and service emulations.

Service emulation provides a consistent view of the managed network, although the underlying protocols may differ. A key component involves mapping an incoming service request into a combination of requests and responses, using another protocol and MIB, to carry out and respond to the original request. Some examples of service mappings include simulating CMIP scoping and filtering, scheduled polling and conversion of SNMP poll results into simulated CMIP events, and appropriate error handling where services do not map [13].

These methods are most effective when used in conjunction with each other. To this end, a proxy architecture which permits an existing CMIP manager to manage resources on either an SNMPv1 or SNMPv2 agent, has been defined [9] using a combination of these approaches. Although it has not yet been specified, one could imagine a SNMP based manager managing resources on a CMIP-based agent via proxy mechanisms.

Although mixed stack and dual stack approaches are the simplest to implement, they should only be viewed as as short-term solutions to coexistence. A long-term solution must provide a uniform, end-to-end view of the managed network, regardless of the underlying protocols. Recent work has shown that this interworking can be achieved through a combination of MIB translations, protocol translations and service emulations.

4.3 File Transfer

4.3.1 Service As Described in the Standards

IPS Standard File Transfer

The IPS file transfer application is known as File Transfer Protocol (FTP). FTP is a simple and user-oriented protocol designed to transfer files between remote hosts. FTP was originated by the United States Department of Defense (DOD) in the MIL-STD-1780 document. It is a popular protocol with a mature implementation base.

FTP uses the IPS TELNET protocol to set up a command connection between the FTP user and server. The command syntax and its calling sequence are much less complicated than the FTAM counterparts, and, as a result, are easier to implement and use. A three-digit reply code followed by explanatory text not only indicates the status of the preceding request, but also provides sufficient diagnostic information to explain each transaction.

The services provided by FTP are login (USER, PASS, ACCT), close (QUIT), transfer of entire files (GET, PUT), and file directory capability. A limited number of file types (e.g., ascii, binary, image) can be transferred.

In comparision to FTAM, this protocol provides a relatively unsophisticated set of file access functions and limited file management. The capability for an FTP user to directly modify access permission to files on the remote host is not available. File access is limited to an entire file, rather than to a particular part of a file. File management functions, renaming and appending are simple. Furthermore, FTP combines file management and file transfer in one protocol machine while FTAM separates management and transfer as it relates to files.

OSI Standard File Transfer

The OSI File Transfer application is known as File Transfer, Access and Management (FTAM). FTAM is currently an International Standard in ISO, and is defined in the ISO-8571 documents. It has strong file access, file transfer, and file management capabilities.

FTAM supports filestore management functions, and file selection/creation mechanisms. FTAM has the following functional units: kernel, read, write, lock, limited file management, enhanced file management, restart, recovery, file access, and grouping. There are also file attributes and activity attributes. Different document types specify particular file structures of interest.

Further Comparison Between FTAM and FTP

FTAM requires much more user-supplied information to communicate about files than does FTP. This information comes in the form of parameters and attributes. In general, because FTAM is more generic than is FTP, it is potentially more difficult to implement, but FTAM allows greater flexibility to convey information about files. FTAM deals only with aspects of applications processes, and leaves data representation and data management facilities to other protocol machines.

One difference between FTAM and FTP is that FTAM may access, manipulate, and transfer portions of an entire file. This capability is not available within the FTP protocol. Also, FTAM modularizes operations into regimes of primitives which should properly be grouped together. For example, file management functionality is kept distinct from file transfer functionality. Therefore, an implementor could choose to implement either file management or file transfer, instead of having to implement all functionality, when only one limited capability is desired.

FTAM's detailed file access mechanisms are an important part of the services and protocol. FTAM recovery and restart functional units would provide the user the ability to restore and retransmit files in various stages of transfer. These capabilities are not generally available in FTP implementations.

FTP is simple and has a large existing user base; FTAM products are relatively immature and not widely deployed. However, FTP has extremely limited file access capability and incomplete data translation abilities.

4.3.2 File Transfer: Current State of Products

OSI File Transfer Products

Most current FTAM products only implement the simple file transfer and management profiles. Some products support FTAM gateways to FTP and other file transfer technologies.

Future evolution of FTAM specifications will address character set agreements, FTAM directory service requirements, filestore management, and run length compression. Lower priority work items are security functions, overlapped access, use of exchange formats over FTAM, and advanced adaptive compresssion.

Commercial FTAM implementations are available for most computing environments (e.g., MS-DOS, Unix). A public domain FTAM implementation is contained in the ISO Development Environmend (ISODE) distribution.

See section 4.4.3 for a recent market analysis of FTAM availability and interoperation.

IPS File Transfer Products

Commercial implementations of FTP are widely available for most common computing environments. Public domain implementations of FTP are readily available for MS-DOS and Unix environments.

4.3.3 File Transfer: Current State of Deployment

IPS File Transfer Deployment

As a core IPS host protocol, FTP is widely deployed throughout the Internet and private networks that employ IPS technology. The provision and use of *anonymous FTP* services is the primary mechanism for the public sharing of files in the Internet community.

OSI File Transfer Deployment

As a core OSI end system protocol, FTAM is commonly available on deployed end systems supporting OSI capabilities.

4.3.4 File Transfer: Security Issues

No specific security services, other than typical host accound access control procedures, are provided by either the FTP or FTAM protocol. FTAM does provide the distinct concepts of access passwords and filestore passwords within the protocol. FTAM may make use of authentication ACSE functions in support of access control.

4.3.5 File Tranfer: Coexistence and Convergence Issues

Application gateways have been developed to allow the FTAM and FTP protocols to interoperate. RFC-1415 [18] provides a technical specification of a FTAM-FTP gateway. The FTAM service available through such an application gateway is a subset of the full FTAM fuctionality.

A NIST project developed a prototype FTAM-FTP application gateway. A MITRE prototype FTAM-FTP gateway is included as part of the ISODE public domain software.

4.4 Remote Login (Virtual Terminal)

4.4.1 Service As Described in the Standards

IPS Standard Virtual Terminal

The IPS virtual terminal protocol, known as TELNET, was designed with scroll mode terminals in mind, although some page mode terminal properties can be negotiated. Scroll mode terminals do not have any local editing capability. When a key is hit, the corresponding character is sent over the line and then is displayed. As new lines are displayed, the old lines just scroll upward. Page mode terminals are typically CRT terminals that can display 25 lines of 80 characters each. The computer can move the cursor around the screen to modify selected portions of the display.

The TELNET protocol is concerned with setting up and manipulating two simplex data streams, one in each direction. The protocol has no concept of a data structure that must be kept identical at both ends by the protocol. As keys are hit on the terminal, a stream of 8-bit bytes is transmitted over the line. The terminal process must convert whatever character code the terminal outputs to the network standard, ASCII.

OSI Standard Virtual Terminal

The OSI Virtual Terminal (VT) application provides mechanisms to effectively insulate application processes from the specific characteristics of the terminals with which they communicate. This allows terminals to access applications running on a variety of systems, and vice versa, regardless of the supplier of the terminal or host system. Hence, VT facilitates intercommunication in a multivendor environment.

As its name implies, the VT service is provided by defining a virtual terminal which is a generic representation of all the operations possible on a user's terminal. This generic representation is then mapped onto the specific characteristics of the terminal.

The VT protocol provides a number of features, many of which are optional. Implementations of the protocol vary in the features they support. The VT standard and implementors' groups have defined profiles to support today's terminals and applications. A profile is a named subset of features of the VT protocol. A profile definition determines the types of applications you can run on a remote host, as well as the terminal types which can be used.

The VT standard defines two default profiles, one for each communication mode (synchronous and asynchronous). Other defined profiles include: Generalized Telnet, Forms, Paged Application, X.3, Telnet-88 and Scroll. A brief description of each profile follows:

Generalized Telnet - The Generalized Telnet profile supports services equivalent to those provided by IPS TELNET. This profile supports three basic services - data transfer, commands, and option negotiation. The profile was developed to facilitate the migration from the IPS

TELNET environment to the OSI VT environment and to support the development of gateways between networks using IPS TELNET and networks using OSI VT. Many users access UNIX or POSIX style systems where the native terminal support is TELNET. The Generalized Telnet profile has been specifically designed to emulate TELNET in this environment.

X.3 - Triple-X (X.3, X.28, X.29) is a group of standards which define network communications using a PAD meeting the CCITT X.3/ X.28/X.29 recommendations. A PAD interfaces between a start/stop terminal and a packet switch network. Like the Generalized Telnet profile, the X.3 profile provides a migration path between the Triple-X standards and the OSI Virtual Terminal and was developed to help support development of gateways to OSI VT networks. The X.3 profile user accesses applications via a PAD and the X.3 profiles supplies the user with a familiar set of X.3-like characteristics. A major attribute of the X.3 profile is the ability to set local terminal characteristics. For example, the user can determine very precisely the conditions which cause data to be forwarded to the remote host.

Telnet-1988 - This version of the TELNET profile does not mandate support for any options, but limits the options which an implementation may support. The Telnet-1988 profile does not include the IPS TELNET basic negotiation facility for expanded option support. The Generalized Telnet profile, which supports many options and the negotiation facility for expanded option support, provides a superior technical solution. For this reason, an erratum to Version 2 of GOSIP mandates support of Generalized Telnet. Both versions of Telnet are available from vendors, but they do not interoperate.

Scroll - The Scroll profile provides line-at-a-time interactions between a terminal and host system, allowing bi-directional (forward and backward) scrolling of the display. The profile includes the ability to switch local echo on or off and supports the "type-ahead" ability. A typical use for this profile is for applications where type-ahead may be advantageous and control over local echo is required, e.g., the type of application where a conventional teletypewriter device or "teletype-compatible" video device having "full-duplex" capability is often used. Implementors' Agreements for the Scroll profile are in progress but are not stable at this time.

Forms - The Forms profile supports the use of forms-based, field-oriented, data-entry applications between a terminal and a host system. The profile provides facilities for defining and using screen forms, defining field validation and field entry rules, and controlling and validating field entry. This profile also contains capabilities commonly required by other nonforms applications, such as word processors and text editors. The Forms profile is applicable to a wide range of applications utilizing screens. A typical application requires a complex field structure. The profile allows for the complex field structure and in addition, supports an unformatted paged operation where no fields are defined or where a single field covers the entire screen. It is also possible for the Forms profile to simulate scrolling.

Paged Application - The Paged Application profile reflects the functionality of existing block-mode terminals, in particular the IBM 3270 style. The profile contains a subset of the functionality of the Forms profile and represents a natural step in the migration to the Forms profile. As the name implies, the Paged Application profile is used in applications where data is displayed on a whole screen or paged basis. The Paged Application profile also supports applications using fields.

Similar profiles that are identified by several regions of the world are harmonized to produce one specification which can be used internationally. International Standardized Profiles (ISPs) have been developed for the Forms and Paged Application profiles. International Standardized Profiles for other profiles may be developed in the future. The OIW VT Special Interest Group (SIG) is currently progressing the Generalized Telnet Profile through the ISP process.

Version 2 of U.S. GOSIP, includes two categories of VT systems: simple systems and forms capable systems. Simple systems support the Generalized Telnet profile; forms capable systems support the Forms profile. Version 3 of GOSIP will reference the IGOSS, which specifies the following profiles for VT service: Generalized Telnet, Forms, X.3 and Paged Application profiles

4.4.2 Virtual Terminal: Current State of Products

IPS Virtual Terminal Products

Most vendors who market the UNIX operating system, include the Internet TELNET as part of the operating system. For this reason, the Internet TELNET is available from most major vendors and many small software houses. There is at least one public domain version of IPS TELNET, NCSA Telnet.

OSI Virtual Terminal Products

Today, users can procure VT implementations supporting several asynchronous mode profiles. One such profile is the Telnet-1988 profile. Telnet-1988 profile implementations are available primarily because the profile was originally specified in GOSIP Version 2. Vendors who implemented the Telnet-1988 profile are likely to include it as part of an asynchronous mode implementation. Vendors who market in Europe offer the Telnet-1988 profile, since it satisfies a U.K.GOSIP requirement.

In addition to the Telnet-1988 profile, several VT Generalized Telnet implementations are now available. The implementations fall into three categories: terminal servers, PC implementations, and host implementations. Of course, all VT profiles require a full OSI stack and suitable network to interconnect the terminal and host machines. In the coming years, users will increasingly see vendors supporting the Generalized Telnet profile. There are several reasons for this, with the most important reason being the Generalized Telnet profile is a direct competitor with and replacement for IPS TELNET. Vendors who are strong supporters of OSI will no doubt include the Generalized Telnet profile and a Generalized Telnet/ TELNET gateway in their VT products to aid in migration to OSI Virtual Terminal.

In the next couple of years, VT implementations are expected to include the following additional profiles:

- Forms profile
- Paged Application profile
- X.3 profile
- possibly, Scroll profile

Additional VT implementations supporting the Generalized Telnet profile may appear on the market. Vendors may upgrade their Generalized Telnet profile implementations by including additional options which have been defined for IPS TELNET. One attraction of the Generalized Telnet profile is that is has been specifically designed to allow future options of IPS TELNET to be automatically incorporated into the Generalized Telnet profile.

The following vendors have Generalized Telnet or Telnet-1988 implementations; those vendors with an "*" have or plan to have in the near future a Forms implementation: Alcatel TITN *, Control Data Corporation *, Cray, Data General *, Digital *, ICL *, McDonnel Douglas, Olivetti, Racal Data, Retix, Santa Cruz Operation, Sequent Computer Systems, Spider Systems *, Unisys, The Wollongong Group. Several of the vendors listed plan to have X.3 implementations in the future; only 3COM Corporation has an X.3 implementation available now. Vendors like Hewlett-Packard, IBM and SUN Microsystems also will have implementations when users start requesting OSI Generalized Telnet.

4.4.3 Virtual Terminal: Current State of Deployment

IPS Virtual Terminal Deployment

As mentioned above, every major vendor supplies some version of IPS TELNET with the operating system. This is especially true of UNIX operating system vendors. Most PC networking packages contain the IPS TELNET as well. Almost every government agency relies on the Internet and especially Telnet for remote login. This is also true of many universities. Most database services and electronic bulletin boards have IPS TELNET as one of the primary access methods.

OSI Virtual Terminal Deployment

OSI virtual terminal implementations are not widespread today. The NIST had access to five implementations and was aware of at least two others during the production of its "Guidelines for the Evaluation of Virtual Terminal Implementations". But availability of implementations does not imply deployment. Deployment of OSI virtual terminal products will increase only as users start to request the functionality.

The Aerospace Industries Association (AIA) conducted a GOSIP Demonstration Project which was described as a test readiness demonstration. The first phase of the demonstration was completed in April of 1992 and was aimed at evaluating the production readiness and capabilities of GOSIP

	Interoperability	Ease of Implementation	Product Availability	Vendor Support
X.400	High	High	High	High
FTAM	Moderate/High	Moderate	High	Moderate
VT	Unknown*	Low	Low	Low

*Insufficient experience and product availability

Table 4: AIA GOSIP Demonstration Project Results.

products. The project specifically targeted three GOSIP protocol suites: FTAM, MHS, and VT. Nine companies participated in the FTAM demonstration and testing activities, seven companies participated in direct company-to-company MHS PRMD testing and two companies participated in VT testing. The results of the demonstration are presented in the Table 4.

As can be seen in the chart that the AIA team found X.400 production ready in every sense. The team found FTAM production ready. However, FTAM is immature in the sense that it is not easy to install or configure; something the team believed was compounded by inexperienced vendor support. Finally, VT is not ready. Not enough products were available for a reasonable test. The only VT products tested were from the same vendor, so no determination was made as to contract readiness.

The second phase of the demonstration project involves testing X.500 and evaluating its production readiness and capabilities. The second phase should be complete by June 1993.

4.4.4 Virtual Terminal: Security Issues

For both IPS and OSI virtual terminal the only security feature that appears to be present is use of a password. Every system has the capability of requiring a user to enter a password before completing the remote login process. On some systems, even this protection has been disabled. There are no known implementations for either IPS or OSI virtual terminal service which include any other security services.

4.5 Transaction Processing

4.5.1 Service As Described in the Standards

IPS Standard Transaction Processing

There is no equivalent IPS Standard Distributed Transaction Processing.

OSI Standard Transaction Processing

The OSI Transaction Processing (TP) protocol is the first standardized Application protocol which allows distributed Application programs to perform Transactions spanning multiple applications on different computers and operating in an heterogeneous environment (e.g., with multiple vendors platforms). It became an International Standard in April 1992. Its main features rely on the properties attached to the notion of transaction.

A transaction is a logical set of operations characterized by the ACID (Atomicity, Consistency, Isolation, Durability) properties. The TP protocol makes use of the facilities of the OSI CCR (Commitment, Concurrency, Recovery) protocol to ensure the *ACID properties* of the distributed transaction. These ACID properties are defined as follows:

- Atomicity means that either the entire transaction will complete or none of it will. This ensures that the data base will not remain in an inconsistent state in the event of a failure.

  The notion of recreating the initial state of the atomic action is called "rollback".
- Consistency means that the outcomes of the operations are consistent with the application semantics.
- Isolation means that a user cannot have access to any partial results of the operations of the atomic action before the completion of the entire action.
- Durability means that the operations must endure any communication or application failure.

The CCR protocol is itself based on the "Two-Phase-Commitment" protocol, taken from database theory. Briefly, itt works as follows:

Phase 1: A superior node A asks a subordinate B if it can perform an atomic action, When B receives the request, it can agree or refuse to perform the action. B informs A of its choice. If B agrees, it sends an "offer to commit" message to A. In any case, B needs not perform the atomic action during this phase, although it may put a lock on its bound data (i.e., data which may be affected by the atomic action).

Phase 2: Node A receives the refusal or the "offer to commit" message from B. If B sent an "offer to commit" message, then A can decide whether it wants to follow through. In the meantime, B is in a doubt period because it does not know whether A wants to follow through with the commit. If A decides to follow through, then it sends a "commit" signal to B. B then performs the atomic action and sends a "done" signal to A when it is finished. If A decides not to follow through, it sends a "rollback" signal to B. When B receives the "rollback" signal, it drops the task. If it put a lock on its bound data earlier, it should remove the lock.

At every strategic moment in the lifetime of the transaction, TP records all the necessary log informations on a secure storage.

An Application Process Invocation uses the services of TP through a TPSUI (TP Service User Invocation). At a given instant, the TPSUI can dialogue with several remote TPSUIs through associations. A peer-to-peer relationship between two TPSUI is called a dialogue. An association is used for each dialogue. TP ensures the synchronization of the different dialogues during the lifetime of the transaction. A dialogue has direction: the TPSUI that initiates the dialogue is Superior while the recipient is Subordinate. Dialogue trees are formed when a subordinate TPSUI begins a new dialogue.

Dialogue Establishment Dialogues are established with an optionally confirmed service called TP_BEGIN_DIALOGUE. The parameters include the caller and recipient names, optional user data, functional units, and the choice of whether or not this dialogue begins a transaction. One of the functional units, the "Commit" functional unit, provides the Application Process Invocation with the choice of either requesting TP to ensure the ACID properties or not. (In this last case, namely a dialogue with a "NONE" coordination level, TP only manages the associations for the different dialogues; in the first case, namely a dialogue with a "commitment" coordination level, TP will manage the new dialogue branch as a Transaction branch). A confirmed begin-dialogue is non-blocking (i.e., the issuer of the begin-dialogue may immediately send data, prepare a transaction,or perform any other normal dialogue operation). With the "NONE" coordination level, an unreliable datagram method of transaction processing is also supported. It is possible to initiate a dialogue, send data and terminate the dialogue without receiving a reply of any type.

Chained-Unchained Transactions The Chain functional unit provides two modes of transactional supports: chained or unchained. A chained transaction begins at the same time as the supporting dialogue. A new transaction, typically of the same type, starts as soon as the previous transaction ends. With unchained transactions, a dialogue may begin without being in a transaction, or the transaction may begin at the same time as the dialogue (depending on the value of the "begin-Transaction" parameter of the TP_BEGIN_DIALOGUE request). When the transaction commits or rolls back, a new transaction does not begin. A transaction may begin subsequently on receipt of an explicit request (TP_BEGIN_TRANSACTION). The unchained transaction paradigm is useful for applications that may access resources in an unprotected manner to set up the transaction. It makes possible for an application to minimize its work in terms of transactions. An example of this is doing several dirty reads (i.e., accessing bound data without concurrence control protection) while not in a transaction, to determine the specific data to be modified during the course of a transaction. Once this has been determined, a transaction can be started, and reads and updates with locking can complete the work. In the same dialogue tree, it is possible to have some dialogue branches supporting a chained transaction, and others supporting an unchained transaction.

Dialogue Control One of the functional units provides the dialogue with two modes for controling the flow of data and TP services: polarized and shared.

- In the Polarized Control mode, a TPSUI must own a token to initiate action other than error notification, rollback and abrupt termination (abort) of the dialogue.
- In the shared mode, a TPSUI is not required to own a token to initiate an action.

Data Transfer TP does not define a specific data transfer mechanism. Because numerous data transfer mechanims are already standardized (or are in the process to be standardized) within OSI (e.g., remote data base access RDA, file transfer FTAM, remote procedure call RPC,...), it was inappropriate to bias the protocol in favor of a particular method. Instead, TP is flexible enough to provide the bracketing services for an arbitrary data transfer method or even multiple data transfer methods. Indeed TP provides only an abstract service, TP_DATA, to model the data transfer for the sequencing rules. This approach provides possibly the greatest level of flexibility with only one evident but big constraint: two products from different vendors, both written to use TP, may not interoperate if they each use different data transfer methods. The data transfer method is part of the information contained in the Application Context of the application.

Handshake mechanism The Handshake functional unit allows the TPSUIs to synchronize their processing with one another and possibly transfer control. TP_HANDSHAKE service is used by a TPSUI to request confirmation of processing to a known point by another TPSUI. The semantics of this service are determined by the TPSUIs and not by TP. TP_HANDSHAKE_AND_GRANT_CONTROL service is used by the TPSUIs not only to synchronize but also to transfer control, in the case of polarized control dialogue mode.

Heuristic Completion TP takes into account the fact that after a subordinate TPSUI has issued a TP_COMMIT request (i.e., at the end of the CCR first phase), an unacceptable long delay may occur before the final outcome is known (i.e., TP_COMMIT or TP_ROLLBACK indication issued by the superior TP node). Because of this delay, and before the final outcome is received from the superior, the TPSUI may choose to release its bound data into either initial or final state, or any other state. At the time the heuristic decision is made, enough information is written to stable storage to report or undo the possible inconsistency created by this decision. After the outcome is known, the TPSUI uses the TP_DONE service to report whether it set the bound data to the state matching the outcome or to a different state, or to report that it does not know. If possible inconsistency exists, TP attemps to report it to the other TPSUI higher in the transaction tree using the TP_HEURISTIC_REPORT indication. Because of the use of the presumed abort protocol in TP, and subsequent communication failures, this reporting may be lost; the reporting is therefore done on a best-effort basis. There is also no requirement for the TPSUI to take any specific action based on the reporting.

4.5.2 Transaction Processing: Current State of Products

IPS Transaction Processing Products

No IPS product can ensure distributed Transaction processing in a heterogeneous environment.

OSI Transaction Processing Products

Initial OSI TP products are beginning to emerge. Recent public TP inteoperability demonstrations included implementations from Retix, IBm and Transarc. Other major vendors active in this market consider OSI/TP as highly strategic and are on the verge of delivering their first release of OSI/TP implementation.

Currently, The TP standard is just undergoing a process to take into account a list of 30 defect reports, which should end in october 1993 after a period of a 3 months letter ballot.

Another important event will affect the life and the deployement of the OSI/TP protocol. The CCR protocol, which acquired the status of International standard in 1990, and which is an important part of TP, has two important amendments in the process. The first one consists of extensions dealing with the problem of optimizing the "Two-Phase Commitment" mechanism, and has currently the status of Working Draft (WD). The second one deals with several modifications concerning the "mapping" of TP on the Session layer, and has currently the status of Draft Amendment (DAm).

As a matter of fact the OSI/TP uses the version of CCR which corresponds to the second amendment (referenced as version 2).

The publication of International profiles for OSI/TP is expected at the end of 1993.

The transparency of OSI/TP to the transfer methods makes its use potentially universal. However each Application standard to be used jointly with OSI/TP has to define how it integrates into a distributed transactional context.

- RDA is the first standard which specifies how to be used in a context including OSI/TP.
- RPC standard is currently specifying an Application context to be used with OSI/TP.

In conclusion, although some implementations of OSI/TP have recently appeared, it is more reasonable to expect most products to appear at the end of 1994.

4.5.3 Transaction Processing: Current State of Deployment

No deployement can be discussed so far.

4.5.4 Transaction Processing: Security Issues

OSI Transaction Processing Security Services

Explicitly security mechanisms are not present in the OSI/TP specifications. The CCR protocol, which ensures the ACID properties, can be viewed as a form of security mechanism (especially with the Atomicity and Isolation properties). But as far as OSI/TP is transparent to the data transfer method used by the application, it belongs to the application itself to define the security level it needs, and to incorporate the corresponding mechanisms.

4.6 Remote Database Access

Remote Database Access (RDA) is an important application that involves the structure of query information sent to database servers from database clients. Basically, a user issues a request from an RDA client to an RDA server, which is connected to one or more databases. Currently the standard consists of a generic document and an SQL specialization document. There are also two "contexts" defined in RDA, a basic application context and a Transactions Processing (TP) application context.

SQL Access

The SQL Access Group is a non-profit corporation comprised of over 40 member companies, including almost all database vendors, tools vendors and some end users. X/Open works jointly with SQL Access on specifications, including their publication. The goal of the SQL Access Group is to accelerate multi-vendor database interoperability and application portability by (1) accelerating existing standards efforts, and (2) prototyping multi-vendor interoperability. SQL Access interfaces involve: (1) API (application programming interface) — an embedded SQL language application programming interface definition, (2) CLI (call-level interface) — a procedure call-oriented, dynamic SQL interface, and (3) FAP (formats and protocols) — a client-server communication protocol for SQL remote database access.

The FAP specifies the formats and protocols used for communication between a SQL Access client and server. It is based on ISO RDA with the following differences: (1) the addition of several extensions to support API features not supported by ISO RDA, and (2) omission of control dialogues for out-of-band cancel and status functions.

The FAP currently supports single-phase transaction commitment. Supported networks are (1) intially ISO OSI, due to ISO RDA origin, and (2) TCP/IP mapping (draft specification is complete). Current status is that a preliminary specification is available from X/Open, and for RDA for TCP/IP, a first draft will be available through X/Open soon. SQL Access FAP agreements form much of the basis for the RDA agreements in the OIW.

Members of SQL Access are: Digital Equipment Corp., Hewlett-Packard, Oracle, NCR (now AT&T), Teradata, Tandem, Microsoft, Sybase,MCC,Cincom, Sun Microsystems, Informix, Unisys, Fujitsu, X/Open, and Unify. A formal technology demonstration was given in July 1991 , using an 802.3 LAN (OSI) as backbone. This event demonstrated that heterogeneous database systems based on the RDA standard should be viable in the near future.

4.6.1 OSI Remote Database Access

RDA is an OSI Application Layer International Standard for heterogeneous remote database access. It is a generic standard with specializations: (1) ISO 9579-1 (Generic RDA), and (2) ISO 9579-2 (RDA SQL Specialization). The generic standard provides a general purpose mechanism. Specializations specify database type specific refinements (e.g., additional rules or constructs). Some

RDA characteristics are: (1) client-server model, (2) completely vendor-neutral, (3) standard SQL (SQL-89 or SQL-92 Entry Level), (4) standard error codes (database errors specified by SQL standards, and RDA errors fully specified by RDA standards), and (5) platform-neutral message syntax and encoding (uses ASN.1and Basic Encoding Rules, and completely and unambiguously specified). In RDA, the handling of SQL statements is fully specified. It is primarily a requestresponse protocol. Some requests have no response, or error responses only for performance reasons. Several request types are "non-blocking" (requests can be issued without waiting).

For RDA, multiple database requests may be submitted simultaneously, and a repetition count mechanism allows batching of input and output data values. There are two Application Contexts:(1) Basic Context — single-phase commitment (using RDA services), and (2) TP Context — two-phase commitment (using ISO TP services). There is flexible character set usage on a per-column basis.

In terms of associations, there is a basic underlying client-server connection. The association is managed by ACSE (Association Control Service Element). Associations are shared with ISO TP, if it is used. There is security authorization at the client-server level. Associations may be reused for multiple dialogues (amortize set-up costs).

In terms of dialogues, there is an end-user-level conversation between client and server, on a full duplex communications path. One dialogue maps to an underlying association. Security authorization occurs at the end-user level (authorization information required is specified by individual servers). Specialpurpose control dialogues are used for cancel and status functions on operations occurring on another dialogue to the server.

RDA services are: (1) dialogue management services, (2) resource handling services, (3) transaction management services, (4) control services (in- or out-of-band), (5) database language services, and "stored execution" services. Database language services provide for immediate execution of database requests, and multiple requests may be outstanding. "Stored execution" services are used to define database language that persists for the duration of the dialogue only.

An RDA application may be implemented in conjunction with the Basic Application Context or the TP Application Context. The Basic Application Context includes only the ACSE Application Service Element and provides a one-phase commit protocol. The TP Application Context provides a two-phase commit which allows updates at multiple remote sites in the same transaction. Initial RDA implementations, as well as GOSIP Version 3, will reference the Basic Application Context. Procurement categories for the TP Application Context will be specified in a later version of GOSIP.

Current focus in implementors agreements and products is on RDA Basic Application Context. Future focus will be on TP Application Context, on alternate abastract syntaxes for transferring SQL argument values and SQL result values, on requirements for RDA managed objects, and on any future RDA specializations.

RDA OIW Implementors' Agreements

The OIW RDA implementors agreements address interaction between an application program and a remote database server, in terms of: (1) limitations on values of parameters of the RDA protocol, (2) other restrictions on operations of an RDA client or server, and (3) profiles. Currently only the Basic Application Context of the RDA SQL Specialization is specified. Agreements on TP Application Context is a future work item. There are agreements on the dialogue initialization, dialogue termination, transaction management, cancel, status, resource handling, immediate execution Database Lanaguage (DBL), and stored execution DBL functional units.

SQL Access and RDA

The SQL Access Group has divided RDA into subsets in order to specify basic interoperability. The following RDA features are not supported: (1) control dialogues (used for out-of-band cancel and status), (2) R-DefineDBL, R-InvokeDBL and R-DropDBL, and (3) TP Application Context. All unsupported features are addressed by official RDA subsets (negotiated at start-up).

The following three SQL features require extensions to RDA: (1) dynamic SQL, (2) character-varying (VARCHAR) datatype, and (3) extended diagnostic information. All three extensions have been submitted for inclusion in a future RDA Addendum.

4.6.2 SQL Access and the IPS

The goal of an SQL Access over TCP/IP mapping is to (1) maximize potential target platforms (minimize memory footprint, avoid dependencies, maximize user/administrator acceptance), (2) minimize specification effort, and (3) use an approach easily extended to other network technologies. The approach used is to: (1) map SQL Access FAP directly onto TCP service interface (e.g., sockets, XTI, etc.), (2) retain RDA message contents and encoding, and (3) replace ACSE association setup with a single, simple message exchange. The approach is specified in terms of a mapping from Presentation and ACSE service primitives to TCP/IP service interface.

Prototyping RDA over TCP is a simple first step for basic interoperability using the direct mapping approach. Future extensibility is possible using RFC-1006 for more advanced features (e.g., two-phase commit). A server product is expected from Digital (client) soon, as well as base technology (source) products from Retix (supports OSI, TCP/IP (direct and RFC-1006)).

The Network Database Working Group within the IETF is chartered to define a standard interface among databases on TCP/IP networks. The Working Group will address the issue of database connectivity in a distributed environment which allows authorized users remote access to databases. It will be designed as a client/server model based on TCP/IP as its communication protocol. Issues identified with this approach are (1) management of multiple threads between clients and servers, (2) management of multiple servers, (3) management of data buffers, (4) data conversions, and (5) security.

The Working Group is independent from the SQL Access Group; however, due to possible overlapping interest, discussions are starting between the two activities with a possible goal of merging them in the future. There are two Internet Drafts available: Network Database Protocol and Network Database Implementation Information Internet Draft. No decision has been made by the Working Group at this time as to whether to use their approach (using RFC-1006) or the SQL Access approach (using a direct mapping) to map to TCP/IP. An issue was identified with getting public access to the SQL Access (and X/Open) documents required for a proper comparison.

4.6.3 RDA: Current State of Products and Deployment

RDA implementations are just starting to emerge. There is no equivalent functionality in the Internet world, although talks are going on. Some test code for RDA is under study at NIST, and SQL testing can be combined with RDA testing as a less expensive alternative to building a complete RDA tester. In RDA there is one-phase commit and two-phase commit. Each of these may involve separate complex issues.

DRDA

IBM has produced a Distributed Relational Database Architecture (DRDA) that addresses database interoperability. The initial focus of DRDA was to provide a vehicle for interoperation among IBM's four relational database managers. More recently, IBM has provided DRDA specifications to other companies, so DRDA can be used for multivendor interoperability. The DRDA specification defines a model for client/server interaction based on several other IBM architectures, including SNA LU 6.2.

Significant differences exist between DRDA and the SQL Access approach, involving language, catalog tables, message/data value encoding, and data conversion. Nonetheless, DRDA has and will continue to have an influence on the actions of SQL Access and the RDA activities. Some gateway products are emerging to convert between DRDA and these other approaches.

4.7 Directory Services

4.7.1 Service As Described in the Standards

IPS Standard Directory Services

In the traditional[6] IPS environment, directory services are usually described in terms of three basic components:

- the "WHOIS" service currently provided under the InterNIC Directory and Database Services contract;
- the "Host Table" service currently provided under the InterNIC contract; and
- the "Domain Name System" (DNS).

This section briefly describes each of these components in terms of the information available to a user and how it may be accessed. It is important to note that each component is a physically separate repository of information and that there is significant information overlap among them.

NIC "WHOIS" Service

The official specification of the WHOIS service is provided in RFC 954 [19]. Another commonly used name for this service is "NICNAME." For years this service was provided by a centralized[7] query/response database system running on a single server in California. Since the advent of the InterNIC contract in 1992, the service is provided by a combination of three servers; however, there is still a single server that provides a common point of access to all WHOIS servers. WHOIS continues to be regarded as a centralized database service because of the small number of servers and because there is a single "master" server that uses the other two servers as "slaves" while the two slave servers do not interact between themselves. The master server is ds.internic.net and the two slaves are rs.internic.net and nic.ddn.mil.

For the typical user, the WHOIS service is accessed via an interface program that is executed on the user's local host to programmatically connect to the master InterNIC Directory Services server. Each execution of the local interface program is designed to service a single query.

The WHOIS database contains information about the following types of objects.

---

[6] The adjective *traditional* is used here to denote the situation prior to the advent of large pilot projects in the Internet which aim to deploy the OSI Directory Services standard within the Internet. See 4.7.3 for more information about the use of the OSI Directory Services standard in the Internet environment.

[7] A database system is said to be *centralized* if it is supported on a single host machine. In contrast, a database which is fragmented across multiple network hosts, with access via a globally common set of application software, is called a *distributed* database. These are simplified definitions; for a careful discussion see [20].

- Network users: information such as organizational affiliation, phone numbers, and Internet email address is available for users who have registered with the WHOIS service.
- Internet host machines: information includes network address information, type of operating system, computer type, and the name of a registered user who is an administrative contact. Similar information is available for gateways, Terminal Access Controllers (TACs), and Packet Switching Nodes (PSNs).
- Hierarchical domains used for naming purposes within the Domain Name System [8].

The WHOIS database system, from a user's perspective, accepts simple query requests and performs a broad or constrained search of the WHOIS databases for relevant information. The query request consists of a full or partial name for a user, host, mailbox, organization, gateway, or TAC. Alias names, where applicable, are also allowed. Alternatively, the query request could specify a full or partial Internet address for a host, gateway, PSN, TAC, or Domain server. The request may also contain an optional parameter which constrains the search to only consider a specific category of name or address (e.g., search host names only). Each entry in the database is also assigned a unique label, called a *handle*, which may be specified instead of a name to completely constrain the search. An unconstrained search on a partial name performs a very broad search of the database for names in any category which begin with the given partial name; the result is a list of objects where each has a name, handle, or address which begins with or consists of the given search key. WHOIS also provides the capability to associate certain types of information; for example, a list of registered users can be associated with each registered host and TAC.

Because WHOIS is a centralized database service, it does not scale easily to serve the needs of a global user community. Problems dealing with the growth in size and use of the database have already led to breaking the database into three physical pieces that are logically integrated using ad hoc methods. Centralized database services also suffer from availability problems during peak usage periods or if the host is unavailable or unreachable for a prolonged period. In addition, turnaround time on data changes can be slow when database administration is centralized. The combined limitations of the traditional directory services for Internet Protocol Suite users have led to extensive pilot programs throughout the world that are experimenting with the use of the OSI Directory Service to provide a more flexible and scalable directory. The OSI Directory pilot programs are described in 4.7.3.

NIC Host Table Service

The official description of the Host Table service is contained in RFC 952 [21] and RFC-953 [22]. The former specifies the format of the Internet Host Table while the latter specifies the Hostname Server Protocol for accessing information in the Host Table.

[8] The WHOIS database contains information about the first and second tiers within the hierarchy; RFC-1032 specifies this information. Although stored in the WHOIS database, this information describes the logical structure of the Domain Name System (DNS) distributed namespace.

This service, like WHOIS, could be described as a centralized query/response system hosted on an InterNIC server. In this context, access would involve use of a local application program which establishes a TCP connection to port 101 at the service host. Another way that this service is accessed involves the use of FTP or the Hostname Server Protocol to copy the contents of the Host Table to a local system. Once copied, the table can be accessed using a text editor or library application routine.

Each Host Table entry is an ASCII text string containing six fields. The first three are:

1. *keyword* which indicates whether the entry pertains to a net, gateway, domain, or host;
2. *Internet Address* corresponding to the net, gateway, domain, or host[9];
3. *official name* of the network, gateway, host, or DNS domain.

The last three fields of a Host Table entry are optional and their use depends on the type of entry.

From a user's perspective, using the Hostname Server Protocol, the NIC Host Table service accepts simple query requests, performs a search of the Host Table for related entries, and then returns the related records. Query requests, as described in RFC-953 [22], may specify a hostname or host address. In these cases, the Hostname Server would return all host entries matching the requested name or address. A query may also specify various keywords (e.g., "ALL" or "DOMAINS" or "ALL–OLD") which result in the return of subsets of Host Table entries. These keywords could be used to copy the entire Host Table, or subsets of the table containing only DNS domain entries or gateway entries.

The problems associated with Host Table service mirror those of WHOIS: the service is not scalable to the needs of a global network infrastructure.

Domain Name System (DNS)

The offical specification for the DNS is contained in RFC-1035 [23]. A detailed discussion of basic concepts and capabilities is given in RFC 1034 [24]. Administration of the system is discussed in RFC-1032 and RFC-1033 [25, 26]. A brief description of requirements for the system is given in RFC-920 [27]. RFC-974 [28] focuses on mail routing and the DNS. A good reference on the origins and evolution of the DNS is [29]. This section provides an integrated overview of some of the information in these documents. Aspects of the DNS that are described as experimental or obsolete in RFC-1034 or RFC 1035 [24, 23] are not considered in this document [10].

Unlike Host Table and WHOIS, the DNS is a distributed database system. The contents of the database can be viewed from a network global perspective as being allocated among many DNS

[9] An alternate address is also included. Address information associated with a domain specifies the address of a Domain Name Server for the domain – such records exist in the Host Table only for the top level of the DNS domain hierarchy.

[10] This includes Resource Record types MB, MG, NULL, MINFO, MF, and MD.

servers. The basic unit of fragmentation is called a *zone*; each zone consists of a set of *Resource Records* which describe the set of hosts in that zone. The description of each host in a zone always includes the host's Internet address and may optionally include information about the machine model, operating system, *Well Known Services* [11], mail exchange relationships, and alias names.

The functional model of the DNS follows the client/server paradigm with clients being called *Resolvers* and servers being called *Name Servers*. A Name Server may or may not be capable of interacting directly with other Name Servers. All network communication associated with DNS utilizes a standardized message format described in RFC 1035 [23].

A typical DNS Resolver offers the following three functions.

1. Translation of a host name to the corresponding host address; historically, this function was designed to mimic the functionality of the Host Table service.
2. Translation of a host address to the corresponding host name.
3. General lookup of host information. In this case, the client provides a DNS domain name and Resource Record type. The DNS response consists of all DNS database records matching the given name and record type.

The DNS is a highly specialized database system. The specialization of the system provides one of its strengths and one of its weaknesses. Because it is specialized to handle a small number of record types and has limited search capability, it is able to provide high performance. The ability of the DNS to rapidly respond to queries is important because it promotes smooth operation of applications which use DNS to obtain routing information. However, specialization also limits its applicability. In general, DNS functionality is relatively difficult to extend [12] when compared to the flexibility inherent in the OSI Directory architecture. On the other hand, the OSI Directory architecture, while catering to scalability and flexibility, is vulnerable to performance problems. The comparison between DNS and the OSI Directory Service is continued in the next section.

OSI Standard Directory Services

The OSI Directory standardization effort is a joint activity of ISO and ITU-T. To date, this effort has produced a base standard, in 1988, which has been revised and extended in the new 1993 edition. The standard is published as the ISO/IEC 9594 series of standards and ITU-T X.500 series of recommendations. IGOSS and GOSIP requirements are based on the 1993 edition of the standard and include mandatory support for 1993 extensions such as access control and replication.

The OSI Directory is designed to be a highly scalable and extensible distributed database system which, like the DNS, follows the basic client/server paradigm of architecture. The client application programs are called Directory User Agents (DUA) while the servers are known as Directory Service

[11] Well Known Services include network services such as FTP, TELNET, and SMTP.
[12] The difficulty of extending DNS functionality is briefly described in RFC 1308.

Agents (DSA). Beyond this most basic similarity, however, the DNS and OSI Directory are very different.

While the DNS is primarily devoted to mapping between host names and addresses, the OSI Directory architecture is designed to provide a more general and easily extended database service. The OSI Directory has many standardized object classes including hosts, people, organizations, organizational units, countries, localities, and application processes. Most implementations allow local database administrators to easily add locally defined object classes. Each object class is described in terms of mandatory and optional standardized attributes. In most implementations, Object class descriptions can also be easily augmented to include locally defined attribute types[13]. The architecture of the OSI Directory is specifically designed to accommodate a broad spectrum of locally defined objects and attributes without the need to modify neither the standard nor server implementations. Draft IGOSS and GOSIP requirements for the OSI Directory require implementors to provide products that support the flexibility inherent in the standardized architecture.

Other important differences between DNS and the OSI Directory include the following.

- In DNS, data can be replicated from one Name Server to another, but the only unit of replication is an entire zone; also, there is no automated way to manage updating replicated information. In the OSI Directory the unit of replication is very flexible to allow a DSA to copy only the data that is most needed to serve its user community. Both, the OSI Directory and the DNS have an automated means of managing refresh of replicated data.
- In DNS, information held by each Name Server about the contents of other Name Servers must be manually managed and updated as the structure of the Domain hierarchy changes. In the 1993 OSI Directory, information in each DSA about the contents of other DSAs can be managed automatically.
- The query functions of the OSI Directory provide a richer interrogation service than is offered by the DNS. Specifically, the DNS only supports READ, while the OSI Directory supports READ, LIST, SEARCH, UPDATE, DELETE, and other operations.
- The security services supported by the 1993 OSI Directory include comprehensive standardized access control, authentication, and data integrity (See section 4.7.4 for more information on OSI Directory security). The DNS offers no security services.

As previously described, one positive aspect of the specialization of the DNS is that its performance is high. Many typical uses of the DNS would suffer significantly if DNS were not able to quickly perform host name→address translation. It is not clear if the performance of the OSI Directory will be adequate for all such uses. Implementors are currently working hard to improve the performance of the OSI Directory.

[13]In all existing implementations, locally defined attributes must use standardized syntax. However, this is not usually regarded as a serious constraint since standardized syntax includes most well–known data types.

4.7.2 Directory Services : Current State of Products

IPS Directory Services Products

Products needed to access and use the WHOIS and Host Table services are widely available. Likewise, DNS server and resolver products are widely available; a public domain implementation is also available.

OSI Directory Services Products

A comprehensive catalog of OSI Directory products is provided in RFC 1292. There are several commercial implementations and one dominant public domain implementation of the 1988 OSI Directory standard. RFC-1292 also lists numerous products that implement various subsets of the full functionality of the 1988 standard.

NIST is also aware of three companies expected to complete implementations of the 1993 edition of the standard by the end of 1994. These implementations include standardized access control and replication and it is anticipated that they will satisfy draft IGOSS and GOSIP requirements.

4.7.3 Directory Services : Current State of Deployment

IPS Directory Services Deployment

The WHOIS service is widely used by the IPS community. The Host Table service, despite its problems, continues to be used by legacy applications that have not been converted to use DNS.

DNS is widely deployed throughout the IPS community and is used extensively by some implementations of Internet electronic mail.

OSI Directory Services Deployment

The OSI Directory is filling Directory Services needs in a large number of countries. As a directory to locate people, it is provided in the U.S. as the White Pages Pilot Project, run by PSI, and in Europe under the PARADISE Project as a series of nation–wide pilots. The PARADISE project currently comprises over 500 servers collectively containing over 1,000,000 user entries.

The OSI Directory is also being used by the FOX Project in the United States to provide a service similar to WHOIS for people and networks, and to provide a pilot K-12 Educators directory. It is also being investigated for its ability to provide resource location facilities and to provide source location for Wide Area Information Service (WAIS) servers.

Within the Federal Government, there has been a recent push initiated by the present administration to speed up efforts to interconnect Govenment Agencies with a ubiquitous electronic mail service.

These efforts have resulted in the initiation of several large pilot projects using OSI electronic mail and directory services. In July 1993, OMB initiated a task force of high–level Government officals to ensure the continued progress of deployment of X.400 and X.500 within the Government.

An important aspect of current deployment is that the PARADISE, FOX, and White Pages pilot projects all use the IPS environment. For these pilots, the OSI Directory is used in conjunction with RFC-1006 which specifies a mechanism that allows OSI applications to run in the IPS environment. Many of the Federal Government pilot projects, in contrast, are primarily using the pure OSI stack environment.

4.7.4 Directory Services : Security Services

IPS Directory Services: Security Services

WHOIS, Host Table, and DNS specifications do not address security services in any form.

OSI Directory Services: Security Services

The 1988 OSI Directory Services standard provides standardized authentication services using passwords and digital signatures based on public key cryptography. Standardized object classes are also provided to support the infrastructure associated with public key certification authorities. Digital signatures provide authentication of requestor and data origin (with non–repudiation) while also incorporating a data integrity service. Generally, implementations of the 1988 standard include proprietary access control schemes.

Security extensions in the 1993 edition of the standard include standardized access control and improvements to the digital signature authentication service. The standardized access control service is based on the concept of an access control list and incorporates a model of organizational security policy that accommodates many forms of delegation of authority.

Current work on future extensions to the standard include an effort to define a standardized confidentiality service for the OSI Directory. This service will be based on the new Generic Upper Layer Security standard.

4.8 Ad hoc Information Services

This section describes other ad hoc services that are part of the IPS by for which there are no direct OSI counterparts.

There is an enormous volume of information available through the Internet. This information is in a variety of forms. Examples are:

- anonymous ftp archives: Freely accessible, these contain all manner of papers, documents, graphic images, software sources, and raw data.
- database access: A variety of databases are available for querying. The genome database and various white pages directories are examples.
- services: These consist primarily of access to proprietary services such as UPI news, NEXIS/LEXIS research services, and the Dow Jones information service. Often, these appear as "services" rather than as other forms of information so the purveyor can charge for access by providing accounts.
- Usenet newsgroups: Usenet is a collection of topically-named newsgroups, which have the character of special-interest discussion lists. There are over 2,000 newsgroups covering a very diverse set of topics, each with its own readership.

In the midst of such a cornucopia, it is often difficult, if not impossible, to locate and retrieve information on a particular topic – or to know if such information is even available. For this reason, the last several years have seen a number of resource discovery tools being designed and experimented with in the Internet. These tools typically include relatively simplistic protocols or, in some cases, merely use existing protocols, such as ftp; often the protocol consists of client-server transfers of ASCII messages. A few of the tools use more "formal" protocols (e.g., see the description of WAIS below).

The more complex and interesting part of resource discovery is information management. The designers of the tools make certain assertions on how to organize information so that it can best be presented to and used by the end user. In general, the designers recognize that users want to:

- browse the available information,
- find specific information of interest, and,
- retrieve the information.

It is interesting to note that, while many of the decisions made on information management are different, they are also complementary. In a number of cases below, the different tools have been integrated to a greater or lesser degree, providing a richer information environment (further discussion of this is beyond the scope of this paper).

The understanding of resource discovery in the Internet is evolving. A comprehensive and well-written overview is provided in "A Comparison of Internet Resource Discovery Approaches" (see References [30] – [31], Section 5.2). The article also contains a good bibliography. A representative sample of existing tools is described below. This sample illustrates the diversity of work, but is not all-inclusive.

None of the tools described is currently on the Internet Standards track; this awaits further evolution of ideas, experience and understanding. When and if one or more of these tools (or their descendents) is proposed as an Internet Standard, it will most likely be in recognition of its de-facto achievements in the Internet "marketplace."

None of these tools has an exact analogue in the OSI suite. On the other hand, most or all of them could operate equally well in OSI, since they make use of functionality that exists in both the IPS and OSI worlds. The key determinants for devloping these tools in the IPS environment are the need to manage information resources and the fact that this need has arisen as a result of the large and growing IPS-based Internet.

4.8.1 Archie

The primary purpose of archie is to provide a convenient way for users to browse the anonymous ftp archives available throughout the Internet. Archie servers maintain lists of the known anonymous ftp archives in the Internet, of which there are about 900. On a periodic basis, usually about once a month, each archie server will automatically contact each ftp archive and retrieve its directory listing.

Archie clients contact archie servers and issue queries against the database of directory listings. Queries can be of several types: exact match, substring match, and regular expression matching. All of these can be performed as both case sensitive and insensitive.

Servers return lists of file and directory names that match the queries, along with the anonymous ftp host on which each file or directory resides. A reasonable user interface, such as the XWindows-based client, xgopher, incorporates ftp so that the user can retrieve one or more of the files easily.

Archie appears to have three shortcomings.

- Much of the available anonymous ftp information is replicated across a number of ftp archive sites. The result is that a query will return references to duplicate information proportional to the extent of replication.
- The information that is maintained by the archie servers is in the form of file names. These are not always selected with an eye towards being the best query keys, given the file's contents.
- Often a query will match a directory name, not a file name. It is then left to the user to ftp to the host separately and retrieve a directory listing. Given the duplication of results cited above, this quickly becomes cumbersome.

State of Products

There are no known vendor products of archie. The original implementation was done at McGill University in Montreal, Canada. A small number of client and server implementations are available, estimated at five and three, respectively. However, in preparing this report, it was difficult to obtain current information on archie developments. A design specification was not available (e.g., through archie servers), although the most likely source of a specification, the archie server at McGill, was down due to hardware problems. There is no Usenet newsgroup active on archie.

State of Deployment

There are about 15 servers worldwide. There appears to be significant usage. No Usenet newsgroup is in evidence, but there may be a mailing list for interested parties.

4.8.2 Gopher

Gopher was initially intended as a campus-wide information system for the University of Minnesota. Thus, the emphasis is on providing a convenient structure to information of the scope and size necessary to support a campus community. But, since not all of the information is typically stored on on-campus systems, the design of the tool is oriented towards giving the user convenient access to the information without having to know precisely where it resides.

To the gopher user, "gopherspace" appears to resemble a file system, where the user is presented with a hierarchy of items. The service access paradigm is: the user issues a query on an item via a gopher client, and the contents of the queried item are returned from a gopher server. The user browses the hierarchy by querying named items from a client and receiving the contents of the named items from the server.

Items can be of several types. Directory items define the hierarchical structure of gopherspace. Querying a directory causes the server to return the directory contents, i.e., a directory listing. The items in the directory listing are each tagged with a type.

Directories contain a number of types of items, aside from other directory items. Querying a server with the identifying string of an ASCII or binary file causes the server to return the contents of the file.

There are three additional types of items that are of interest. An index-search item can be queried, with an optional search string. The index-search item represents a full-text index of the contents of a plain text document database. Querying the named index-search item causes the server to return a list of file items. The file items represent the results of a full-text search on the document database using the search string. The effect is of getting a constrained listing of the file names in the index-search item.

The second additional type is a text-based telnet session type. The result of querying this type is to access a service via a telnet session. The telnet session is usually to a host that supports some specific service, such as the NEXIS/LEXIS research service. Aside from the way in which the information to initiate the session is discovered, it is otherwise a normal telnet session.

The third additional type is a phonebook type. Similar to the index-search type, the phonebook type is queried with a search string representing the name of the user whose telephone number (and other white pages information) is desired. The current version of this type is a gateway into X.500.

The potential drawback to gopher is a lack of scalability. The difficulty lies in the fact that gopherspace encompasses both enormous breadth and amounts of information. As the information space increases, it will become more difficult for users to locate the information of interest via browsing, the prevalent mode of information location in a hierarchical information system.

To ameliorate this problem, a specialized index-server has been developed for gopher called veronica. Veronica collects lists of gopher items from gopher servers and allows users to perform keyword searches on the lists. The results appear to the user as a list of gopher-accesible items. Veronica is experimental, and itself seems apt to have scaling problems given the centralized model. However, it is also a desirable feature, given the scale of information that is becoming available.

State of Products

There are no known vendor products of gopher. Although it is widely deployed, as with all the resource discovery tools, it is still evolving. Client implementations are available for about 20 hardware/software platforms. Server implementations are available for about a half dozen platforms. Veronica server software is also available. All implementations are free.

State of Deployment

Given the distributed, independent nature of the gopher service, it is difficult to get an exact measure of deployment. From information available, there appear to be 500 or more gopher servers available in the Internet. A very rough estimate at the number of clients would be an order of magnitude greater than the number of servers. Judging by this and other factors, such as press articles and Usenet newsgroup activity, gopher appears to enjoy significant popularity.

Veronica index-servers are available at two or three sites. Generally, these sites refuse service requests due to high load on the servers (whether from veronica, gopher, or other reasons is not at all clear).

4.8.3 Wide Area Information Service

WAIS (pronounced "ways") is a full-text search and retrieval system. It allows users to deploy, search, and retrieve information from indexed databases. The information can be of many types: text, formatted documents, pictures, spreadsheets, graphics, sound, and video.

WAIS was originally developed by Thinking Machines, Apple Computer, Dow Jones and Company, and KPMG Peat Marwick. It uses the Z39.50-1988 information retrieval service and protocol developed by the National Information Standards Organization as the basis for information transfer.

Since the databases supported by WAIS are subject-matter oriented, a database of databases is also supported. This allows users to search for the appropriate database to access based on a specified subject. Currently, this is limited to a single centralized site, provided by Thinking Machines. At a minimum, scalability will require replicas of this database and possibly partitioning of the database across multiple servers.

State of Products

A company called WAIS, Inc., markets a WAIS product. The product is based on the public-domain version (now referred to as "freeWAIS" to differentiate to two).

In the public domain arena, there appear to be two server implementations for WAIS, one for Unix and one for VAX VMS. There are about 14 client implementations for a variety of hardware/software platforms.

State of Deployment

There are over 70 WAIS servers world-wide supporting over 300 databases. The databases contain a variety of information, including weather maps, the Bible, mailing list and news archives, and technical reports. There is an active Usenet newsgroup for WAIS discussion (comp.infosystems.wais).

4.8.4 Knowbot Information Service

The Knowbot Information Service (KIS), copyrighted by CNRI in 1990, provides a uniform interface to heterogeneous directory services that simplifies the task of locating users.

KIS appears to the user as a 'master directory' that contains email address information from several sources, including the DDN NIC WHOIS database, the RIPE WHOIS database, the Internet X.500 white pages, and MCI Mail. Most of these services are email registries themselves, but KIS provides a fairly convenient method to access all of them in one place.

State of Products

Fundamentally, KIS is not a protocol, client-server or otherwise, but a user interface to N different directory services. The only implementation appears to be that done by CNRI.

State of Deployment

There are only two sites that could be found currently operating this service. The most up-to-date documentation is an Internet-Draft dated December 1989 describing KIS as an experimental protocol and service. Thus, this service could be fairly characterized as not under active development.

4.8.5 Finger

"Finger" is the name of a UNIX command [14] which invokes a protocol known as "The Finger User Information Protocol" described in RFC 1288 [32].

Finger provides a specific set of information about the target person. For example, the command

```
finger ransom@osi.ncsl.nist.gov
```

returns the following information:

```
Login name: ransom                        In real life: Michael Ransom
Directory: /home/osi/npg/ransom           Shell: /bin/csh
On since May 25 09:43:42 on console
Plan:
Mail last read Fri Jun 4 15:01:06 1993
```

where the first three lines of information are taken from the UNIX operating systems information and the line(s) of information following the "Plan:" line are taken from a file named .plan which each user maintains. Limitations of the Finger service include:

- the user must already know which host to finger to find a specific person;
- since primarily UNIX machines run fingerd, people who reside on other types of operating systems are not locateable by this method;
- Finger is often disabled on UNIX systems for security purposes;
- if one wishes to be found on more than one system, it is necessary to make sure that all the .plan files are consistent; and
- there is no automated search mechanism to search the .plan files on a given host to, for example, find "everyone on osi.ncsl.nist.gov who works on X.500".

Finally, it is important to note that Finger is only available over TCP/IP.

[14] Finger is also implemented on a small number of other machines.

5 Comparison of Testing Paradigms and Techniques

5.1 Testing Within the OSI Environment

Following the introduction of the GOSIP requirement, NIST established a program for testing and registration of GOSIP products. This program focuses on mandatory conformance testing, and optional interoperability testing. The conformance program includes the following:

- Adoption of internationally used abstract test suites; continuous quality assessment and improvement of test suites.
- Establishment of a validation program for conformance test systems (i.e., Means of Testing (MOT)).
- Accreditation of conformance testing laboratories.
- Review and registration of successfully tested GOSIP products.

In addition, the GOSIP Testing Program has active links with European testing authorities, and developing links with Japanese testing organizations, with a view to eventual worldwide harmonization.

The GOSIP register currently contains around 200 products, from a wide range of suppliers. New entries are tested against an incrementally improving MOT base.

The GOSIP approach to Interoperability testing is to qualify and register Interoperability testing organizations. Two such organizations are currently registered - OSINET, based at the Corporation for Open Systems, and the SPAG PSI scheme. The DOD/DISA Joint Interoperability Test Center (JITC) has been licensed as a SPAG PSI operator. Contact COS or JITC for information on the OSINET and PSI registers, respectively.

5.2 Testing Within the IPS Environment

Historically, testing on the Internet has been ad-hoc and informal. The "TCP/IP Bakeoff" — a sort of interoperability testing tournament - is conducted, during which some implementations are tested against each other, using some informal accummulation of tests. As far as is known there are no specific codified Static and Dynamic interoperability criteria, so the level of interworking is not documented anywhere. However, the usual result of such a bakeoff is a media event, announcing the interoperable participants. It is not known whether any formal record is maintained of the level of testing conducted, and the extent of success of the participants.

In 1988 the Department of Defense established a mandatory conformance testing policy for all DOD procurements. This testing requirement has been waived in all subsequent procurements.

References

[1] Alan Lloyd Gary DIckson. *Open Systems Interconnection, Computer Communications Standards and GOSIP Explained*. Pretice Hall, 1992.

[2] Marshall T. Rose. *The Open Book, A Practical Perspective on OSI*. Pretice Hall, 1990.

[3] Douglas E. Comer. *Internetworking with TCP/IP, Volume I Principles, Protocols, and Architecture*. Pretice Hall, second edition, 1991.

[4] John S. Quarterman. *The Matrix: Computer Networks and Conferencing Systems Worldwide*. Digital Press, 1990.

[5] David M. Piscitellow A. Lyman Chapin. *Open Systems Networking, TCP/IP and OSI*. Addison-Wesley, 1993.

[6] A. Lyman Chapin. The internet standards process. RFC 1310, IAB, March 1992.

[7] IAB and IESG. The internet standards process – revision 2. Internet draft, IETF, September 1993.

[8] ISO/IEC. Procedures for the technical work of iso/iec jtc1 on information technology. Technical report, ISO/IEC JTC 1, Switzerland, 1992.

[9] Jim West Rachid Sigelmassi. A survey of solutions to the connecitonless/connection-mode and the osi/dod internetworking problems. Technical Report NCSL/SNA – 90/1, NIST, February 1990.

[10] R. Colella, E. Gardner, and R. Callon. Guidelines for OSI NSAP Allocation In The Internet. RFC 1237, Network Working Group, July 1991.

[11] M. Rose D. Cass. Iso transport services on top of the tcp: Version: 3. RFC 1006, IETF, May 1987.

[12] M. Rose. Snmp over osi. RFC 1283 (Obsoletes RFC1161), IETF, December 1991.

[13] Network Management Forum. Iso/ccitt and internet management coexistence - issue 1.0. Technical report, NMF.

[14] X/Open. Systems management: Management protocols api (xmp). X/Open Preliminary Specification P170, ISBN 1-872630-32-4, X/Open, July 1992.

[15] Network Management Forum. Translation of internet mibs to iso/ccitt gdmo mibs. Technical report, Network Management Forum, September 1993.

[16] Network Management Forum. Translation of internet mib-ii (rfc1213) to iso/ccitt gdmo mib. Technical report, Network Management Forum.

[17] Network Management Forum. Translation of iso/ccitt gdmo mibs to internet mibs, issue 1.0. Technical report, Network Management Forum, September 1993.

[18] R. Slaski J. Mindel. Ftp-ftam gateway specification. RFC 1415, IETF, January 1993.

[19] et al K. Harrenstien. Nicname/whois. RFC 954, SRI International, October 1985.

[20] G. Pelagatti S. Ceri. *Distributed Databases – Principles and Systems*. McGraw–Hill, Inc., 1984.

[21] et al K. Harrenstien. Dod internet host table specification. RFC 952, SRI International, October 1985.

[22] et al K. Harrenstien. Hostname server. RFC 953, SRI International, October 1985.

[23] P. Mockapetris. Domain name – implementation and specification. RFC 1035, Network Working Group, November 1987.

[24] P. Mockapetris. Domain name – concepts and facilities. RFC 1034, Network Working Group, November 1987.

[25] M. Stahl. Domain administrator's guide. RFC 1032, SRI International, November 1987.

[26] M. Lottor. Domain administrator's operations guide. RFC 1033, SRI International, November 1987.

[27] et al. J. Postel. Domain requirements. RFC 920, USC Information Sciences Institute, October 1984.

[28] C. Partridge. Mail routing and the domain system. RFC 974, CSNET CIC, BBN Laboratories, Inc, January 1986.

[29] K. Dunlap P. Mockapetris. Development of the domain name system. *Proceedings ACM SIGCOMM '88*, 1988.

[30] S. Fisher. Ietf groups define specs for plugging holes in snmp management. *CommunicationsWeek, No. 448*, 1993.

[31] A. Deutsch P., Emtage. Archie. *Connexions: The Interoperability Report, Vol. 6, No. 2,*, 1992.

[32] D. Zimmerman. The finger information protocol. RFC 1288, IETF, December 1991.