

ASURITE

Arizona State University
Rational Information Technology Environment

Version 1.65
October 19963





Table of Contents

I.	INTRODUCTION.....	1
II.	GENERAL CHARACTERISTICS.....	4
III.	CHARACTERISTICS OF INDIVIDUAL SERVICES	5
IV.	SPECIFIC SERVICES THAT WILL BE PART OF ASURITE.....	6
V.	PRODUCT ARCHITECTURE	8
VI.	SPECIFICATIONS FOR BASIC SERVICES	13
A.	TIME	13
B.	AUTHENTICATION	13
C.	AUTHORIZATION	14
D.	FILE SERVICE	16
E.	FINDER/NAVIGATOR SERVICE	18

I. INTRODUCTION

The Problem

Computers are getting continually more powerful and consistently less expensive. This has encouraged the pervasive use of information technology throughout the University. Despite our historical reliance on mainframe computing, the majority of ASU's investment in technology now sits on the desktop.

When viewed at the department level, this evolution toward distributed computing is a good thing. It allows the department to improve their operation without long delays waiting for external expertise. However, when technology is implemented in piecemeal fashion, units can find themselves unable to accomplish work that crosses organizational boundaries. This is usually because the various departments' computing environments have not been designed to work together. Computer people call this situation "Islands of Technology." For example, a faculty member may have a useful computing "island" for tracking records and grades of his students, yet find it impossible to send final grades to the University's student information system.

When faced with the obstacles presented by these isolated islands of technology, many organizations attempt to centrally control or coordinate technology implementation. Some organizations try to settle on a single vendor to insure that all pieces of technology will operate together. This approach is not practical for ASU because we have a high degree of autonomy within our various units; we have already made a large investment in a de facto heterogeneous computing environment; and even if we could all agree on a homogeneous set of technical products, we lack the massive budget required to replace significant portions of our technology.

So, we need a better strategy to address this problem. Departments need to answer three basic questions:

If I'm buying technology and want to maximize my ability to work cooperatively with others, what should I buy?

If I have limited budget but want to improve my technological situation incrementally, how can I evolve in the same direction as everyone else?

If some functions are going to be handled centrally for efficiency's sake, what tasks will be done for me and what tasks should I prepare to do for myself?

Approach

In a simplistic sense, the way to guarantee that we can cooperatively share technology is to identify standards. This approach has worked well in the area of audio music. We can all buy cassettes and expect them to work on any cassette player made by any manufacturer. The same is true for compact disks. One can plug a Sony amplifier into a Technics receiver with Bose speakers and easily construct a viable audio system. This is because the manufacturers adhere to standards.

Unfortunately, we don't yet have widely accepted standards for most aspects of computer technology. There are emerging standards that hold promise for improving our situation, but today there is no "plug and play" ability among all vendors. A significant obstacle to the ultimate success of any standard is the rapid pace of change among computer technologies.

ASU's Rational Information Technology Environment (ASURITE) is an information technology architecture that positions the university to take advantage of emerging standards. It also recognizes the need to accommodate budget constraints, moderate the pace of change and preserve the autonomy of the individual departments.

ASURITE describes a distributed style of computing that is constructed of modules. Each module performs a specific function and can be thought of as analogous to an audio component. The components are frequently called "servers." So, the computing environment at ASU will have several data servers which store and retrieve data. Several print servers will produce output at various locations. Mail servers will store and forward mail throughout the university, etc.

Some modules lend themselves to being implemented and supported centrally. For example, security can be best maintained by allowing a single method to gain access to computing resources. Customers would typically identify themselves during their first interaction with any computing resource and then be granted authority to all valid and appropriate services. One server can be maintained centrally rather than have multiple security checks with multiple procedures and passwords.

Departmental implementation and support is more appropriate for other modules, such as a database used only by a single department. But that departmental database may need to obtain some of its data from a central database, so the ability to interact with central services must be maintained.

ASURITE is an architectural framework which describes how all supported components will interact within such an environment. The architecture encompasses various styles of computing including client/server, distributed computing, cooperative processing and object orientation. It is intended to help ASU achieve flexibility, adaptability and efficiency in information technology, by putting processes on the right platforms, in the right location, and in a consistent manner.

ASURITE treats the individual as the focal point of a series of software “services” supporting the individual’s dual role as both **data producer** and **information consumer**. In general, services can exist on any combination of hardware and physical locations deployed in an “intelligent” network. It is primarily the desktop which invokes the services where and when needed to satisfy an individual’s need.

The desktop will become the focal point of the individual’s interaction with enterprise systems and data, as well as with collaborative groups, research data bases, etc. Data, voice/sound, graphics/images, and live video will converge on the desktop as the common denominator for synthesizing information from data. All applications will reside in a robust, intelligent network which presents the information consumer with a single system image. ASU systems and links to the external world will appear to be a single network composed of services and data, invoked by name regardless of the physical locations and technology used to provide those services and data.

Figure 1 provides an overview of the ASURITE as it will be implemented over time.

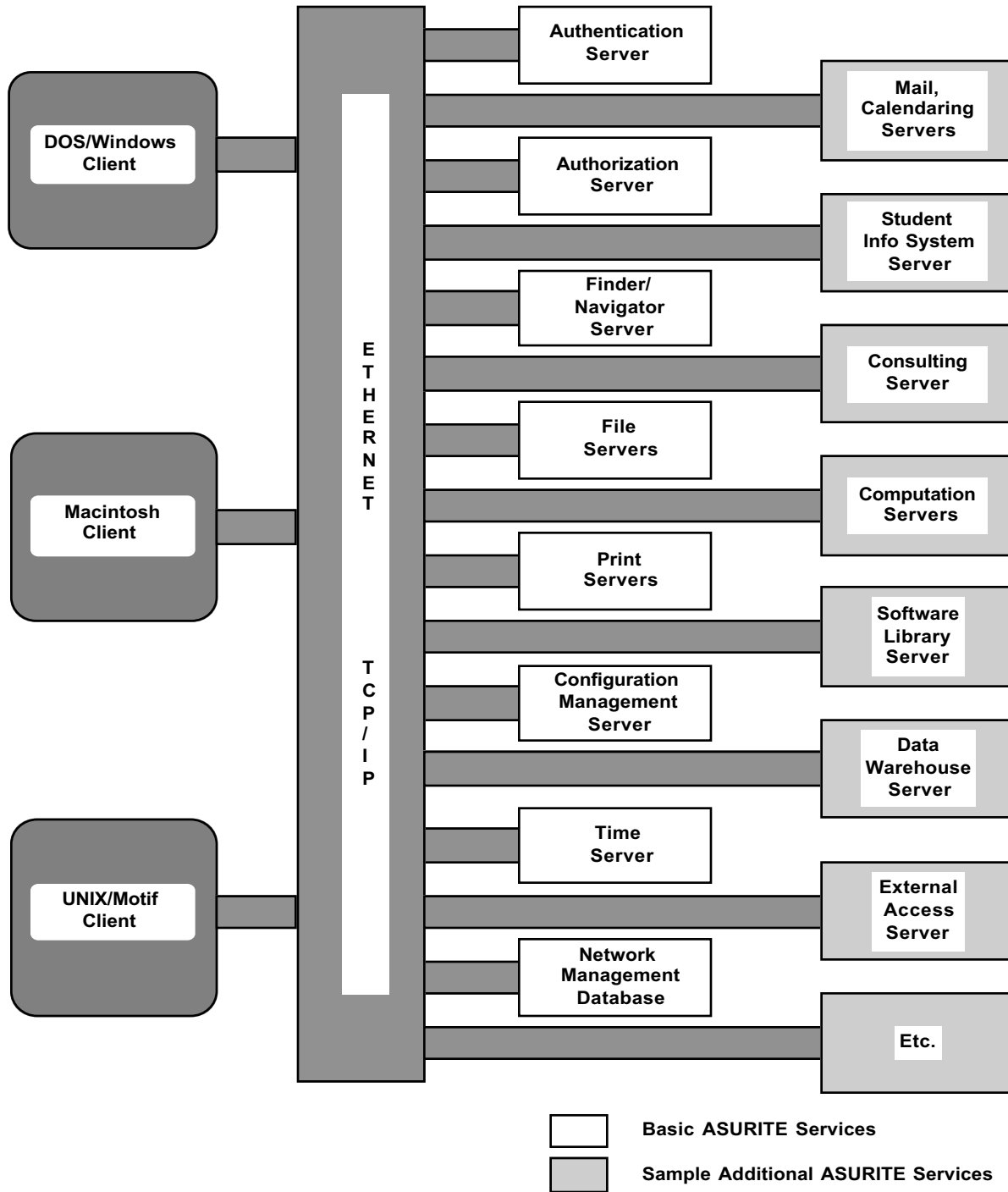


Figure 1. ASURITE Overview

II. GENERAL CHARACTERISTICS

In order to achieve the overall ASURITE architectural objectives, the following qualitative characteristics are established:

Adaptability - change as national & industry standards evolve, so we can enhance and incorporate new ways of doing essentially the same business function without major developmental impact.

Manageability - centrally manage or coordinate and monitor, including the orderly planning for capacity changes of various essential services.

Reliability - remain in continuous operation even if part of the system suffers failure, needs maintenance or upgrading, or is destroyed or damaged by a disaster.

Securability - provide different access to individuals based on the classification of data and the user's business function. This will require that all basic ASURITE services use standard (ASURITE) authentication and authorization services.

Extensibility - easily add new kinds of functionality to existing processes without major impact.

Scalability - increase or decrease size or capability in cost-effective increments without software impact or "spikes" in the unit cost of operations due to step functions in procuring additional resources.

Performance - fast response and high throughput.

Connectability - communications access to a variety of area, national, and international networks.

Consistency - relative stability of the person/machine interface over time.

Accessibility - university community members should be able to access and use ASURITE services wherever they are, provided that they have a properly configured "client" workstation.

III. CHARACTERISTICS OF INDIVIDUAL SERVICES

The following qualitative objectives are established for each individual service offered within ASURITE:

Like an extension to the desktop - how information is presented to, manipulated by, or provided by the user needs to be consistent across all applications no matter where the application is actually running -- on the desktop or on a network server. The user interface can be made more consistent by making it appear as if the information is completely under the control of the workstation software with which the individual user is already familiar. Just as the user can tailor the workstation software to satisfy her needs, so should she be able to tailor the interface for information from and to external systems.

Interoperable - (1) any supported service is available to any supported client no matter the particular brand of server or client hardware and software and (2) the interaction between clients and servers is transparent to the client, e.g., the client does not need to know where the service is coming from.

Incorruptible (virus-free) and as secure as practical - computer viruses are detected and prevented from spreading to servers and clients, and data and computer systems are protected from unauthorized use and tampering. Absolute guarantees, however, of virus prevention and security are not feasible.

Fault-tolerant - reduce the impact of hardware and software failures. Highly critical servers might have redundant processors and databases so recovery from a failure would be immediate and transparent to the user; other, less critical servers could have backup servers that could be put into operation within a few hours.

Disaster-tolerant - restore services in a timely manner when a disaster, such as a fire, destroys equipment.

Expandable - additional capacity can be added to meet the demands of more users or increased functionality without modification to user procedures.

Peer to the client - no master/slave relationship should exist between client and any server - the client is not controlled by the server. A client makes a request of a server and is prepared to receive a response (or a request to supply more data to the server) but is free to do other processes in the meantime.

Restricted in access as needed - since all services are technically accessible to any user on the network, individual service providers may limit access to their services if necessary; e.g., a departmental printer may be restricted to use by members of the department.

Non-interfering and non-conflicting with other services - any user can use any service or combination of services concurrently.

Appropriately interactive with the client - the client can monitor and alter its requests for services. The server should make the status of a request for service available to the client so the client/user can cancel or modify the request if needed; e.g., it should be possible to determine that a data base query is retrieving much more data than anticipated and to cancel that query if desired.

Optional - local (to the client) services are allowed; e.g. a local printer may be completely under the control of the local client.

IV. SPECIFIC SERVICES THAT WILL BE PART OF ASURITE

- A. The following services are considered basic to ASURITE, and must be in place before other services can be added:

Authentication and authorization - authentication is the verification process that confirms the identity of a person requesting service; this process must be done in a secure manner to prevent others from determining the method of verification. Authorization is the process that permits only those who have been granted permission to use a particular service to actually use that service.

Finder/navigator - permits users, clients, and servers to reference services, devices, and people by name rather than by physical location or network address. These services also permit transparent relocation of devices, servers, etc.

Time - synchronize date and time of day on all the servers and clients so that time-dependent processes are coordinated.

File management - provides for the storage, access, and security of data (e.g., text, images, and voice) particularly to facilitate the sharing, interchange and security of data. File management services include

backup and recovery services make duplicate copies of data in case the working copies are damaged and provide procedures to restore lost data from the backup copies, and

archiving services provide facilities to store and retrieve seldom used data on low-cost media, such as tape.

Print - provides for the transmission, temporary storage, and production of paper output of data (including text, plots, and images) from clients and other servers.

Configuration management - set of services to coordinate the software and hardware on the servers and clients and includes

notification of changes,

update by subscription,

coordination of non-optional upgrade of software, and

verification of hardware compatibility.

Network management data base and status - maintains data concerning the network configuration and operations.

- B. The following are examples of services or classes of services that will be added to the basic ASURITE services over time:

Collaboration support is a set of services that facilitate human communication between individuals, within a group working on a common effort, or among groups interested in a particular topic. These services include messaging, computer-facilitated conferencing, electronic mail, voice mail, calendaring, and groupware.

Enterprise applications are those computer applications that support the major administrative functions of the university.

Object catalog contains data about enterprise data (e.g., names, descriptions, and usage rules) and common processes using enterprise data (e.g., a complex query that extracts data from several data bases and puts them in a spreadsheet). An object catalog is an extension of the data dictionary concept.

On-line consulting is a repository of information and previously asked questions and answers to help users and support personnel solve hardware and software problems.

Computation servers handle resource-intensive calculations that are inappropriate for running on a local workstation.

Software library services provide for the distribution of shareware or site-licensed software and the lending of software for trial use.

Databases services make information available to any client and are provided by commercial, research, administrative, and other sources (e.g., the administrative data warehouse).

Scheduling of tasks services control processes that do not need to be run immediately, e.g. long reports or database backups.

External access services permit use of the servers from locations outside of the university-operated network, e.g. from Internet sites or via dial-in from home.

Problem reporting and tracking services receive and facilitate the resolution of system problems.

Approval and signature services permit the electronic (i.e., sans hardcopy) authorization of official documents, e.g., purchase requisitions, grades, and payroll.

V. PRODUCT ARCHITECTURE

The intent of this section is to list some specific products and standards that currently appear to comply with the general characteristics and services that form the core of ASURITE. The list of products and standards is incomplete because in some cases no product or standard exists that conforms to the ASURITE architecture. However, current products and standards will evolve and new ones will emerge to fill in the gaps.

An example of an emerging distributed computing standards isare the Distributed Computing Environment (DCE) ~~and Distributed Management Environment (DME)~~ standards originally provided by the Open Software Foundation (OSF). OSF is an independent company formed by a coalition of computer and network product suppliers. Recently, OSF and X/OPEN, an independent company that promotes international standardization, have combined to form a company called the Open Group. Its goal is to provide a set of open industry standards for distributed computing. Manufacturers that conform to these standards are assured of interoperable products. Because DCE ~~and DME also~~ provides extensive coverage of the services listed earlier, ASURITE will be relying heavily on products that use it them. At one time there was a companion product called the Distributed Management Environment (DME) that was to provide system and network management standards. DME has been disbanded because the computer and networking industries could not reach consensus on protocols. Instead there are several competing point solutions for some of the management areas and none for others.

Users view ASURITE from their desktop workstations, each with the individual's own preferred method of interaction. In acknowledgment of this, ASURITE will provide support to the general community for desktop Operating System-Graphical User Interface (OS-GUI) combinations that will interact with appropriate servers. The initial set ~~will~~ was be:

DOS 5 - Windows 3.1

Mac System 7.1

UNIX-Motif.

These OS/GUI are expected to evolve or be replaced over time. Possible successors include Windows NT and Windows for Workgroups for DOS/Windows and AOCF for the Macintosh. To keep from updating this document every six months as hardware, operating system and GUI versions come into the fore, a separate document, called ASURITE Recommended Client Configurations, will be available on line in the IT portion of the Web.

The primary network communications protocol set required to obtain ASURITE services will be Ethernet and TCP/IP. However, protocols in addition to TCP/IP, e.g., Appletalk, IPX, and DECNET, will also be supported on the university backbone for use by individual work clusters and departmental local area networks for the near term. It is expected that additional network capacity will be needed at ASU, particularly on the University Backbone, to support all of the new services, so ASURITE will be evolving to new, higher speed protocols as needed and funds allow. ATM is an eExamples of new network standards s being considered ~~are FDDI and ATM~~. In the near future higher speed protocols will be used primarily on the backbone and to connect high volume servers and workstations, but the vast majority of workstations will be connected via Ethernet.

With the large number of LAN's at ASU, it will be important that ASURITE coexist with the LAN's and interoperate with LAN services such as file sharing, local mail, and print services. Client workstation connectivity via Ethernet provides access to both LAN services and ASURITE services even though the LAN might be using a non-TCP/IP protocol. The ASURITE file service, initially, will use the Andrew File System (AFS) which at present can only be directly used by UNIX clients. However, Network File System (NFS) software can be installed on Mac and DOS/Windows workstations and access AFS files via a NFS/AFS translator provided as part of the ASURITE file service. To the user the AFS files appear in the directory as if they resided on the workstation and can be moved to a LAN server or the work-station by simple drag and drop of the file icons. LAN vendors are expected to incorporate interoperability functions in the future so intermediary services, such as the NFS/AFS translator, are not required.

The following tables summarize the ASURITE standards and related products for basic services, client workstations, and communications. These form the components of ASURITE, but, of course, must inter-

act with each other to form a cohesive architecture. Figures 2 and 3 following the tables are included to foster some insight into these complex relationships for a couple of example processes.

Table 1: Basic Services Product Architecture

Service	Future Standard	Current Initial Product	Future Product
Authentication	OSF/DCE	Kerberos v. 4	Kerberos v. 5
Authorization	OSF/DCE	native OS access control	DCE Access Control List
File	OSF/DCE	Andrew File System (Transarc)	DCE Distributed File System
Time	OSF/DCE	Internet, Network Time Protocol	DCE Distributed Time Service
Finder/Navigator	OSF/DCE X.500	X.500 Internet Domain Name Service	X.500
Print	OSF/DME	TCP/IP LPR/LPD HP OpenSpool Palladium	HP OpenSpool DME Print Service
Data Base Management	SQL Access Group	Sybase Informix Oracle	Sybase Informix Oracle
Data Base Transaction Management	SQL Access Group X/Open-XA	Encina	Encina
Configuration Management	OSF/DME		
Network Management Database	OSF/DME	HP OpenView/Sybase	HP OpenView/Sybase
Email	X.400 SMTP/ MIME	SMTP/ MIME POP3/IMAP MS Exchange	X.400 SMTP/IMAP
Calendaring	None yet	MS Exchange	

Table 2: Client Product Architecture

Function	Current Initial Product	Future Product
Operating System / Graphical User Interface	DOS 5.0 /Windows 3.1, Windows 95 , Mac System 7.5 + , UNIX/Motif	Future versions compatible with current initial products
Data Base Access	Sybase Open Client, Microsoft ODBC, Borland IDAPI	SQL Access Group standard
Communications card	Ethernet (10 or 100 Mbit)	Ethernet (10 or 100 Mbit); FDDI or ATM on high volume workstations
Communications software	TCP/IP	TCP/IP

Table 3: Network Architecture

Function	Current Initial Product	Future Product
Wiring	Broadband coax with parallel some fiber optic campus backbone to building router, Broadband coax to floor closet, Twisted pair to workstation	Fiber optic backbone and to high volume workstations, Twisted pair to most workstations
Low-level communication protocol	Ethernet with limited FDDI on backbone	FDDI backbone and to high volume workstations in short term, ATM or other long term; Ethernet to most workstations,
Transmission protocol	TCP/IP (see Note)	TCP/IP

Note: The backbone network also supports DECNET, IPX, and Appletalk for use by departmental LAN's, but access to the ASURITE services requires TCP/IP.

Figure 2 depicts clients accessing enterprise data bases. For example, an application running on a DOS workstation running Windows is accessing enterprise databases. Some conversions are required to turn the application's query into an ASURITE standard form. The DDE, a Windows inter-application communication mechanism, is converted by ClearAccess to SQL commands understood by Sybase's Open Client software. The Macintosh client uses DAL (the Apple Data Access Language) in place of DDE and Open Client to construct standard SQL commands. The query is sent to the enterprise database server using an ASURITE standard network transport protocol. At the server the query is converted from DAL or Open Client SQL to the specific SQL dialect supported by the DBMS before action is taken in the database.

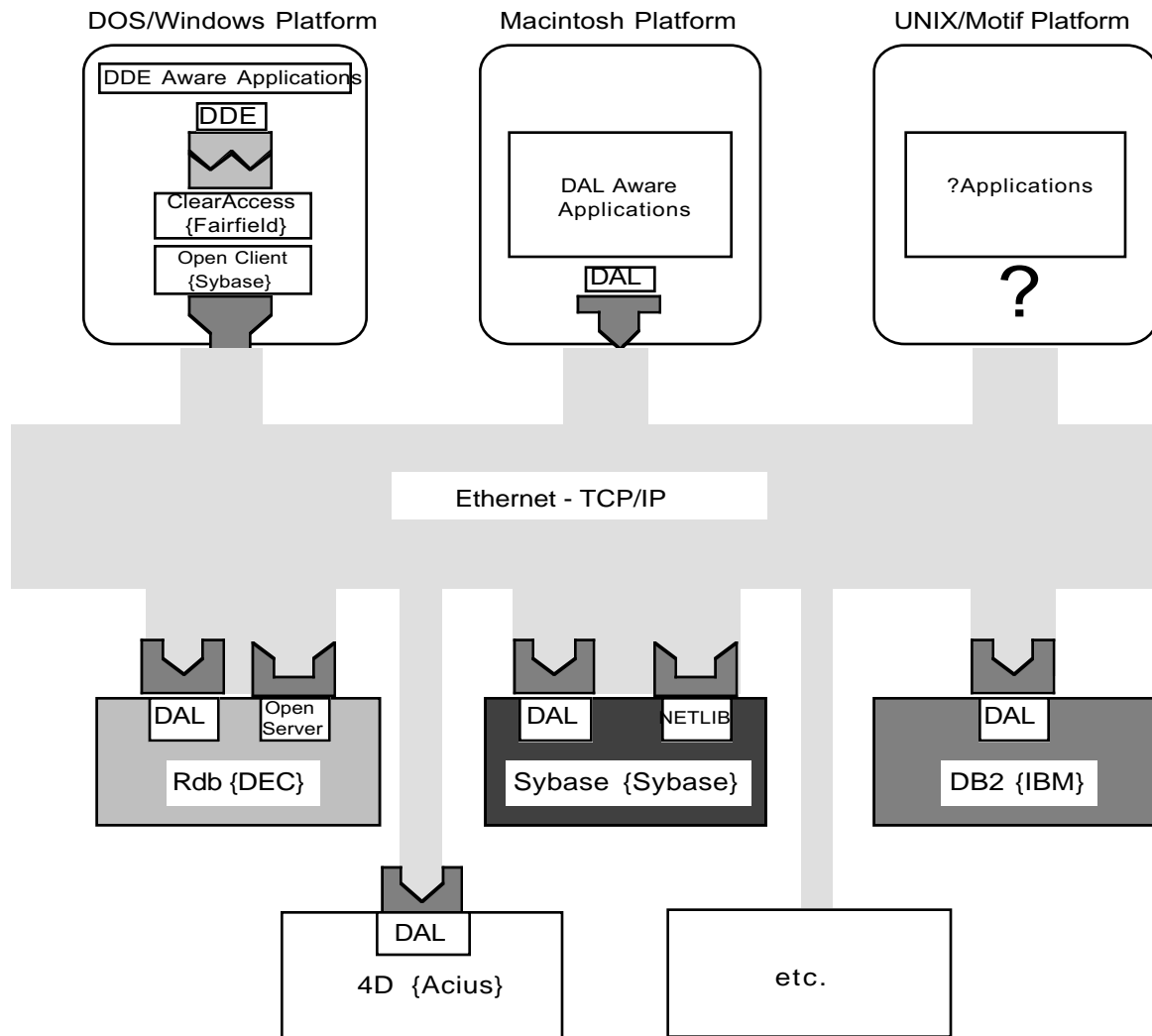


Figure 2. Database Access Environment

Figure 3 shows some of the components involved in a workgroup computing environment. When sending a mail message to an associate, the user may only know the associate's name and not the appropriate mail server. In this case, the directory server is consulted to determine the correct mail server and its address. Even before the mail server can be consulted, the sender's identity and authority to do so must be established using the Kerberos authentication server. The use of each of these services wraps user requests in an envelope appropriate to the service and the envelope is in turn wrapped in the ASURITE standard network protocol. Also depicted in this figure is a possible file system configuration for the workstation. The file system shown has local private files, remote private files, local and remote files shared on a peer-to-peer basis, and remote shared files in a client-server setting.

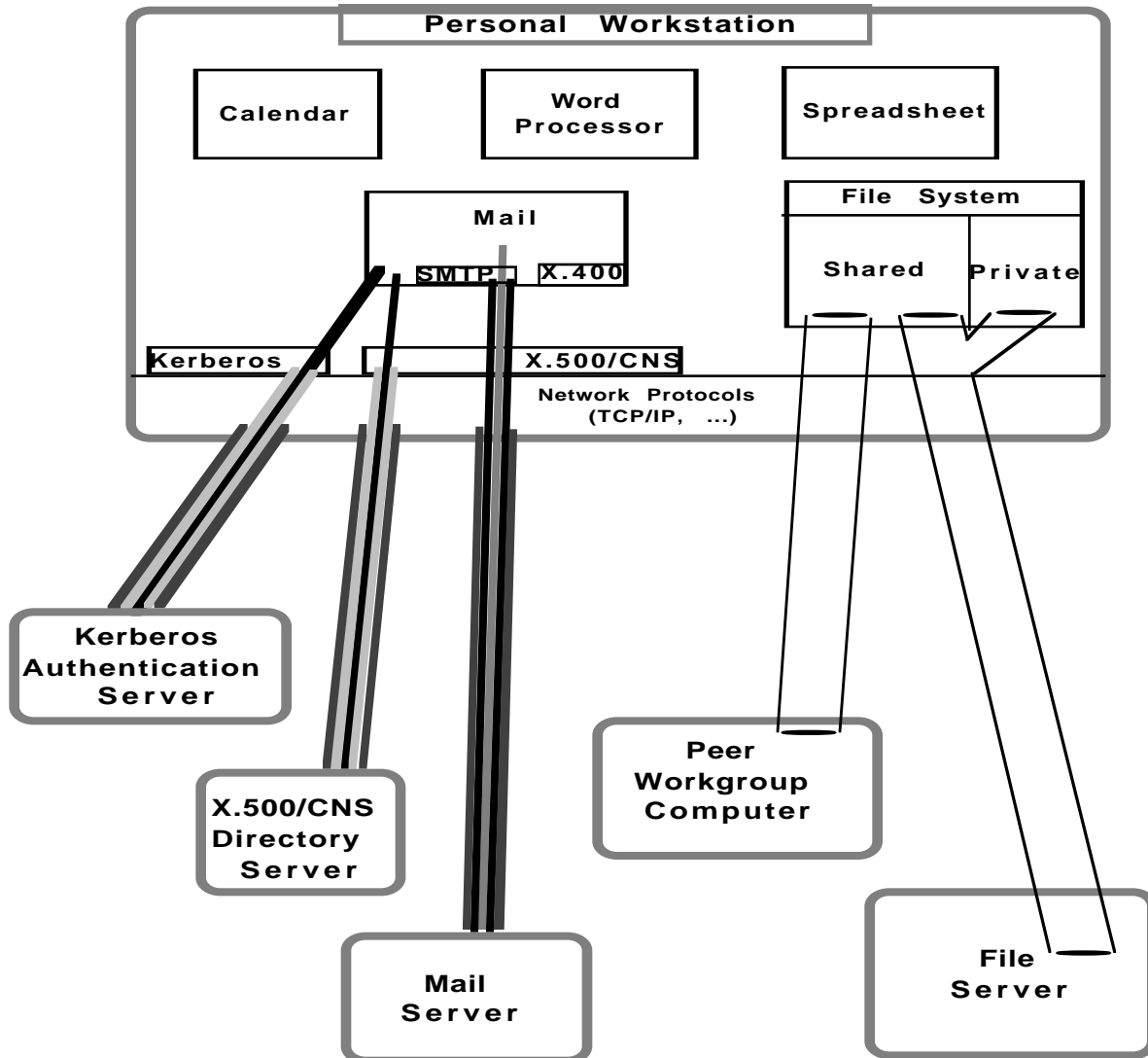


Figure 3. Workgroup Computing Environment

VI. SPECIFICATIONS FOR BASIC SERVICES

The purpose of this section is to provide a more complete description of the basic services, the standards selected for the services, how the services work using that standard, and some implementation considerations.

A. TIME

The ASURITE time service will initially use the Internet Network Time Protocol (NTP) and will convert to the Distributed Time Service (DTS) when it becomes available from OSF and interoperates with other ASURITE services.

Purpose of function

In a networked environment, many distributed applications need a single time reference to determine event sequencing, to schedule activities, and to measure and report event occurrences. Clocks on computers can drift away from the correct time at different rates. If time-dependent components of a distributed application obtain time from clocks on different computers, and the clocks are not synchronized, then applications may give incorrect results. A distributed time service is required to synchronize and standardize the system clocks of the computer systems in a distributed environment. One of the significant users of the time service is the authentication service.

Why this particular standard

NTP will be interoperable with the OSF DCE's Distributed Time Service (DTS).

NTP is a distributed client/server application.

NTP is built on the Internet Protocol (IP) and User Datagram Protocol (UDP) which have low overhead due to the connectionless transport mechanism.

Overview of service function

Two hosts are required to establish a NTP connection. NTP can operate in one of five modes: Symmetric active, Symmetric passive, Client, Server, and Broadcast.

In what may be the most common client/server modes a client sends an NTP message to one or more time servers, which process the replies as received. A server interchanges addresses, overwrites certain fields in the message, recalculates the checksum and returns the message immediately. Information included in the NTP message allows the client to determine the server time with respect to local time and adjust the local time accordingly. In addition, the message includes information to calculate the expected time keeping accuracy and reliability, so that inferior data can be discarded and only the best from possibly several servers can be selected. Quoted from RFC 1129. In the symmetric modes the client/server distinction disappears, the host announces its willingness to synchronize and be synchronized by the peer in an active or passive fashion. In the broadcast mode, the host only announces its willingness to synchronize with all of the peers, but not to be synchronized by any of them.

What is required to implement it?

NTP is generally implemented as part of the UNIX operating system. It is available on the VMS platform through the MultiNet TCP/IP software. PC and Mac implementations will be part of the Kerberos software for those platforms.

The ASU master NTP server will also be a Radio Timecode Receiver so local time will be derived from GMT provided by a the national time service. Also, in order to provide better performance and availability, a few second tier time servers will be implement throughout the campuses to serve client workstations.

B. AUTHENTICATION

The ASURITE authentication service will use Kerberos.

Purpose of function

The purpose of the authentication service is to provide a secure method of verifying the identity of users and servers and to pass user identification in a way that guarantees the user's identity to other services. Transmission of such information across the network must be done in a manner that prevents detection of information that could lead to the compromising of data and service integrity.

By having one authentication service, the other services are relieved of individually providing that function and there is a single authentication occurrence for the user. An authenticated user can access any other service without further authentication.

Why this particular standard

Kerberos was originally developed for the Athena project at MIT and has been adopted by OSF as the authentication standard in the DCE. Kerberos is the most widely accepted authentication standard and is being adopted on heterogeneous platforms. It is commercially available on many UNIX platforms and is available on several non-UNIX platforms as well.

Overview of service function

To use any service within the ASURITE environment a user will first go through a login process which invokes the Kerberos authentication service. The user identifies herself and provides a password known only to her and the authentication service. All transmissions of the password and other sensitive data in the authentication process are in an encrypted form using the National Institute of Standards and Technology Data Encryption Standard (DES). The authentication service not only verifies that the user provided the correct password but also establishes a dialogue (via the passing of "tickets") with the client workstation and with any other service desired by the user to verify to the other service that the user has gone through the authentication process. The authentication service also provides to other services the user's identity and access groups to which the user belongs; such data are maintained in a registry database as part of the overall authentication service.

What is required to implement it - server and client portions

Kerberos software is commercially available for UNIX platforms; sources for Mac and Windows client software are being identified and the software tested.

Procedures for updating the registry database which contains authentication information need to be established.

Kerberos tickets contain an expiration time. Thus workstations and servers must agree on the "correct" time, so a network-wide distributed time service must be available.

Since Kerberos client software depends on the directory service to locate a Kerberos server, at least a limited directory service needs to be available.

There needs to be at least 3 Kerberos servers, each one placed in a secure location and distributed on the network to optimize reliable client access. These locations need to be identified.

Scale factors for Kerberos need to be investigated to determine the number of service areas required.

Mechanisms for distributing client software need to be established.

C. AUTHORIZATION

The authorization function is dependent on the authentication function discussed elsewhere in this document. For purposes of discussion in this section, the Kerberos authentication system is assumed to provide authentication for the ASURITE authorization function.

Purpose of Function

Authorization encompasses a broad range of functions that limit access to particular objects (e.g., files, directories or databases) or services to only those who have [been granted] permission to do so. The functionality encompassed under this ASURITE function include: actual run-time authorization checks and requesting, granting and enabling authorization (sometimes called access management). This function can be very complex in a heterogeneous distributed computing environment if advanced planning has not accounted for the heterogeneity. In ASURITE, authentication is a centralized function but authorization should be distributed where possible because the numbers and ownership patterns of ASURITE protected objects and services are too complex for centralized control of each object.

Is There a Standard?

There are two widely used ways to accommodate low-level run-time authorization checks: use the Sun Open Network Computing (ONC) approach or use the Open Software Foundation (OSF) Distributed

Computing Environment (DCE) approach. The ONC approach, while an industry standard, is a proprietary approach and does not extend to all workstation hardware or operating system bases in general use at ASU. The DCE approach is an industry standard proposed for adoption by all ASURITE supported workstations and servers. However, implementation of DCE for all ASU hardware or operating systems does not yet exist and those that do are immature. Nevertheless, the functionality provided by the DCE for run-time authorization is the most extensible and, in the future, will provide the precise functionality needed for ASURITE. Until all vendors supply a DCE authorization function, the ASURITE goal is to provide services as close to DCE functionality as possible.

How Authorization Works

Authorization has, of necessity, a distributed aspect because the objects are themselves distributed. In ASURITE, as in DCE, the goal is to put control of access to individual objects in the hands of the users as much as possible. On-the-other-hand, another goal is to minimize the administrative overhead of ASU distributed computing. In the case of authorization, the overhead can be reduced by centralizing some of the related functionality. The way this will be accomplished in ASURITE is to distribute the access control to the servers and workstations at which the controlled objects reside but maintain a central master registry and update the distributed access control information as appropriate. When a Kerberos authenticated request for a service or access to an object controlled by ASURITE authorization is made, the authorization system only has to do local checks to make sure that the requester is authorized to get the requested service or access. As can be seen from this introduction, there are two aspects of authorization: (1) run-time authorization and (2) authorization management.

Run-time Authorization

Run-time authorization is dependent on the existence of an Access Control List (ACL) for objects and services. ACL's contain user names (unique ASURITE wide) and a list of rights the user has in accessing the object. ACL's may also contain group names in addition to user names. Users may be assigned to groups as part of the authentication function. When a user makes a request to access an object, the authentication process (Kerberos) passes to the run-time authorization process on the object server a list of all the groups to which the user belongs. If the user or any of the groups to which the user belongs has the right to access the requested object in the way requested then the access is granted, otherwise the access is denied. The access rights of the requester are checked by the run-time authorization system by looking in the object's ACL for the user name or groups contained in the authenticated request.

Authorization Management

When creating objects, such as files, the objects get a default ACL composed of information derived from the containing object and the creator. For example, the creator of a file gets all relevant accesses to the file (e.g., read, write, delete) while a subset of these access rights is given to groups that have access to the directory in which the file resides. Users who have the right to modify an ACL can add, delete or modify users' or groups' access rights to that object. All of this happens locally (to the object) with code residing on the workstation or server where the object resides.

Authorization management at the local level, as in the above case, works fine when the number of objects and the number authorized users are small. But for ASURITE services there will be thousands of objects and tens of thousands of users, and authorization management will be a major effort. There are basically three steps in creating the ACL's for an object:

- (1) A potential user submits a request for access to an object.
- (2) The appropriate authority checks the validity of the request and approves (or denies) the request.
- (3) The request is enabled by adding it to the appropriate ACL.

Traditionally each step requires action by a person or persons to process and approve the request. One of the goals of the ASURITE authorization process is to grant authorization requests without requiring human intervention. Details of this automated process need to be determined, but the following is a possible scenario:

- (1) The potential user completes an electronic form and submits it via electronic mail.
- (2) The authorization management software checks the validity of the request by one of the following methods:

- a) Checking the appropriate ASU databases (e.g., student database or human resources database) to determine if the requester's ASU status automatically qualifies the person for access to the requested object.
 - (b) Electronically sending the request to the appropriate authority for the requested object, and receiving approval electronically.
- (3) Necessary information is sent by the authorization management system to the appropriate server(s) to update the ACL for the requested object.

Implementation Issues

Implementation issues relate to customer and provider objectives, and to open issues for run-time authorization checks and for obtaining authorization.

The customer (user) wants run-time authorization to be fast and non-intrusive. She also wants obtaining authorization to be easy to obtain, e.g., paperless and fast with no need to go to various offices, with self-registration and approval status obtainable electronically. Service providers want re-authentication to be possible during run-time for critical services, distributed approval, service provider definable approval processes and rules with a wide range of criteria, and the ability to require audit trails (journaling) for specified services. Availability of the authorization functions should be continuous for run-time authorization checks and obtaining authorizations should be available, at least, during business hours.

Open issues:

ASURITE Kerberos must exist and ASU-wide user and group name creation provided.

Obtaining ACL software for server platforms (DCE or POSIX 1003.6).

Obtaining work-flow software for electronic authorization.

Obtaining software to access enterprise databases to determine user status for privilege policies.

Writing or obtaining software for database triggers to initiate authorization and de authorization procedures.

Writing or obtaining software to allow service providers to specify access policies.

Determining transition policies and procedures.

NOTE: This discussion has used the term "name" for user, server and group information kept in the Kerberos registry database, kept in ACL's, and passed in service requests. This is a minor simplification in that the names are translated by the various authentication and authorization functions into unique user identifiers which make little sense to a human.

D. FILE SERVICE

Purpose of Function

In a distributed computing environment, the file services play a very large role. The file system and the services that are part of it are one of the most visible services that users see. The file system stores data and programs that users expect to have available whenever they are logged on. They expect to have secure, fast, hassle-free access to a very large, pervasive, robust file store.

The following is a list of objectives for the ASURITE File Service from the user perspective:

Location transparency - users should not need to know the physical location of files and can access all files they are authorized to access from anywhere in the network.

File visibility - the user should be able to see all files and directories (and only those files and directories) they are authorized to access from anywhere in the network.

Desktop extension - files and directories should appear in "native client workstation" format.

Fast access - file access should only be dependent on network speed and efficiency (i.e., the remote file access protocol should be efficient).

Hassle free - (1) second authentication should not be required to for access to private files; (2) file service changes (e.g., authorization for changes) should be simple.

Interoperable - all ASURITE supported file services should interoperate. This does not imply application interoperability (e.g., WordPerfect and MSWord need not interoperate).

Large file store - there should be no inherent file size limit. This does negate the need for user quota restrictions.

Security - only those authorized to view, write, delete, etc. a file or directory should be able to do so. This implies integration with the ASURITE authentication system.

Recovery - file services should provide file backup, restore, and archiving facilities. This is expanded on below.

Temporary storage - temporary storage should be available. This type of storage may be needed for execution of a program for intermediate data or for a few days for other purposes. The second cause is primarily an authorization function and implies a simple method of getting that authorization. The first cause is a capacity planning function but may be approached as an authorization function on a case-by-case basis.

Import and export - sending and receiving files from other file system types. This implies that some translators may have to be written or acquired.

Multiple storage media - storage media should be transparent to the user and the file service should not limit the use of different storage media for such things as backup, archiving or multi-media data.

The backup of the file system to overcome system failures and subsequent restoration of the files is important to all ASURITE users. Of similar importance is the efficient utilization of the system, especially in the area of archiving (i.e., the off loading and subsequent restoration of seldom used files). The following objectives for ASURITE File Service backup and archiving services have been identified:

Backup and archiving of enterprise servers, departmental servers and end-user client workstations should be automatic after initial configuration.

Backup and archiving of servers and workstations should be configurable by date and some measure of local disk capacity utilization.

Restoration of server and workstation files and directories should be through a simple to use and understand GUI interface. Restoration access should be by name, date or range of dates, etc.

There should be an electronic way (e.g., e-mail or work-flow forms) to make initial requests for backup and archiving which includes configuration and timing information. Changes to such configurations should also be accomplished electronically.

Special one time requests for backup or archiving should be supported.

Off-site disaster recovery should be supported.

The following objectives for system support of ASURITE File Services beyond backup/archiving/restore have been identified:

Capacity planning guidelines should be available. Programmatic support for capacity planning in ASURITE is impossible or difficult since, in general, this is a system specific function and ASURITE is a heterogeneous environment.

Space management (e.g., quotas) should be available for permanent disk space. It is usually unavailable for temporary storage.

Performance tuning tools should be available. On a workstation or server system level this depends on the specific system. On a global ASURITE-wide level this means that the system should support the transparent relocation of a user's files even while a file is being used.

Distributed File System Standards

If all the ASU workstations and servers are seen as contributing to the ASURITE file system, the result is a huge set of files that are potentially difficult to navigate around and manage. There are only two widely used file systems that provide good navigability and that permit effective management for a file system of this size: the Andrew File System (AFS) from Carnegie Mellon University and the Network File System (NFS) from Sun Microsystems.

NFS is more widely used than AFS but has security and functionality drawbacks. NFS's functionality drawbacks are rooted in the NFS principle of stateless servers. In this approach, the server retains no information on client access to files between operations (e.g., reads or writes). This principle makes it difficult to provide concurrent access restrictions and adds significant network overhead. NFS's security problems can be overcome with a variant of NFS called Secure NFS which integrates Kerberos authentication and NFS. This variant is not widely available.

AFS is the core technology used for the OSF Distributed File System (DFS). While not as widely used as NFS, AFS (herein-after called DFS) provides for integrated authentication by Kerberos and authorization using Access Control Lists (ACL's) and has less network usage due to extensive caching and client state retention. DFS implementations for all the ASURITE workstation and server platforms exist or are in development. DFS also has interoperability with NFS and several LANs as part of its definition.

Implementation Issues

It is expected that considerable time will be required to migrate all existing ASURITE workstations and servers to the ASURITE distributed file system. For this reason and because it provides better service, DFS has been chosen as the target ASURITE file system. The migration path to DFS begins with using commercial versions of AFS until DFS versions become available. The use of NFS and LAN file systems will not interfere with this migration but at some point exclusive use of these file systems will not be supported by ASURITE.

In addition to the DFS availability issue and the system and user support issues discussed above, there are some LAN coexistence issues that need to be resolved. The following issues are being investigated:

Which LAN client workstations can also access ASURITE File Services?

How can and should LAN file servers be supported for backup and archiving in ASURITE?

E FINDER/NAVIGATOR SERVICE

The ASURITE finder/navigator service will use the OSF/DCE Cell Directory Service for the university network and both X.500 and the Internet Domain Name Service for connections to non-university networks.

Purpose of function

The purpose of the finder/navigator service is to provide a simple way to locate the people, services or devices on the ASURITE network and to communicate electronically with the people or to make use of the services and devices. Users can refer to, or search for, people, services and devices by name or other characteristics, and the finder/navigator service will provide appropriate location information. The location information may be network addresses, e-mail servers, physical location, type of service or any of these for servers that can provide more detailed information. The finder/navigator service hides the complexity of the physical and logical network topology from the user.

Another purpose of the finder/navigator service is to facilitate management of resources on the network. To maintain high performance standards, network resources, such as e-mail servers and file servers, may need to be relocated on the network. By updating the finder/navigator service the new locations are made available to all users without any action on their part.

Why this particular standard

X.500 is becoming the standard for national and international directory services. The National Science Foundation has funded an X.500 directory service accessible via the Internet, and that service is now available on a limited basis, and usage is growing rapidly. So the ASURITE finder/navigator service has to interface with X.500 for off-campus directory services.

However, the Internet currently does not use X.500 to translate node names to IP addresses; the Domain Name Service (DNS) is used for that purpose. So the ASURITE finder/navigator service also needs to interface with DNS.

The Cell Directory Service, which is part of the OSF/DCE set of standards, provides finder/navigator services within DCE cells and can link to both DNS and X.500 for services outside the cell. It also uses other DCE standards being adopted by ASURITE, such as Kerberos and the Distributed File System.

Overview of service function

The Cell Directory Service operates in a client/server mode. When a user or application program on the client workstation makes a request for finder/navigator services, the CDS software on the client (called the CDS clerk) handles the request. In its simplest form, the clerk sends the request to a CDS server which finds the request information in its database and returns the information to the clerk. The clerk in turn passes the information to the user or client application.

However, for performance and reliability reasons, there are multiple CDS servers with each one containing only a portion of the finder/navigator database. If a CDS server cannot find the request information it so informs the CDS clerk and gives the clerk the location of other CDS servers that may have the requested information. The clerk then tries the other CDS servers. Once a request has been fulfilled, the clerk retains the information so the next request for the same information can be fulfilled without accessing the CDS servers.

The CDS servers contain information about people, services and devices within a single cell. If information is needed from another cell, an X.500 directory server or a Domain Name Server, then a Global Directory Agent acts as an intermediary between the CDS clerk and the foreign directory services. If the target directory is X.500 or DNS then the GDA translates between the CDS clerk and the target directory. If information from another cell is needed, then the GDA points the CDS clerk to a CDS server in the other cell.

Implementation Issues

- The number of DCE cells needs to be determined.

- The number and location of CDS servers per cell needs to be determine.

- The finder/navigator service needs to be coordinated with the authorization service so the finder/navigator databases can be easily updated without duplication of effort.