

EXHIBIT F

Part 1 of 2



US005678042A

United States Patent [19]

[11] Patent Number: **5,678,042**

Pisello et al.

[45] Date of Patent: **Oct. 14, 1997**

[54] **NETWORK MANAGEMENT SYSTEM HAVING HISTORICAL VIRTUAL CATALOG SNAPSHOTS FOR OVERVIEW OF HISTORICAL CHANGES TO FILES DISTRIBUTIVELY STORED ACROSS NETWORK DOMAIN**

[75] Inventors: **Thomas Pisello, De Bary; David Crossmier, Casselberry; Paul Ashton, Oviedo, all of Fla.**

[73] Assignee: **Seagate Technology, Inc., Scotts Valley, Calif.**

[21] Appl. No.: **590,528**

[22] Filed: **Jan. 24, 1996**

4,914,571	4/1990	Baratz et al.	395/600
4,987,531	1/1991	Nishikado et al.	395/600
5,001,628	3/1991	Johnson et al.	395/600
5,077,658	12/1991	Bendert et al.	395/600
5,133,075	7/1992	Risch	395/800
5,163,131	11/1992	Row et al.	395/200
5,175,852	12/1992	Johnson et al.	395/600
5,216,591	6/1993	Nemirovsky et al.	395/200
5,220,562	6/1993	Takada et al.	370/85.13
5,247,670	9/1993	Matsunaga	395/650
5,271,007	12/1993	Kurahashi et al.	395/600
5,287,453	2/1994	Roberts	395/200
5,287,461	2/1994	Moore	305/275
5,295,244	3/1994	Dev et al.	395/200
5,325,527	6/1994	Cwikowski et al.	395/650
5,403,639	4/1995	Belsan et al.	395/600
5,448,727	9/1995	Annevelink	395/600
5,506,986	4/1996	Healy	395/600
5,537,585	7/1996	Blickenstaff et al.	395/600

Related U.S. Application Data

[62] Division of Ser. No. 153,011, Nov. 15, 1993, Pat. No. 5,495,607.

[51] Int. Cl.⁶ **G06F 7/00; G06F 7/06; G06F 12/00; G06F 17/30**

[52] U.S. Cl. **395/610; 395/607; 395/200.03; 395/200.01; 395/800; 395/650; 395/280**

[58] Field of Search **395/600, 200.03, 395/200.11, 200.13, 800, 440, 444, 200.06, 200.12, 200.17, 200.18, 182.02, 182.04, 182.05, 182.06, 610, 607, 650**

[56] References Cited

U.S. PATENT DOCUMENTS

4,141,006	2/1979	Braxton et al.	340/505
4,710,870	12/1987	Blackwell et al.	364/200
4,805,134	2/1989	Calo et al.	395/600
4,897,841	1/1990	Gang, Jr.	370/85.13

Primary Examiner—Jack B. Harvey

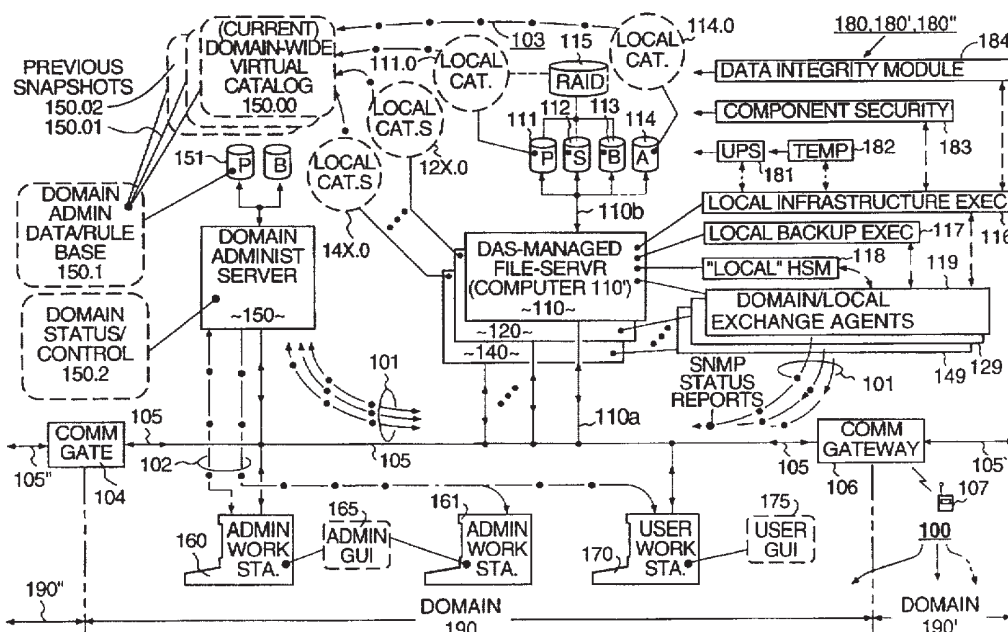
Assistant Examiner—Raymond N. Phan

Attorney, Agent, or Firm—Fliesler, Dubb, Meyer & Lovejoy

[57] ABSTRACT

A network management system includes a domain administrating server (DAS) that stores a virtual catalog representing an overview of all files distributively stored across a network domain currently or in the past. The current and historical file information is used for assisting in auditing or locating files located anywhere in the domain. The current file information is used for assisting in transferring files across the domain. The domain administrating server (DAS) also includes a rule-base driven artificial administrator for monitoring and reacting to domain-wide alert reports and for detecting problematic trends in domain-wide performance based on information collected from the network domain.

4 Claims, 4 Drawing Sheets



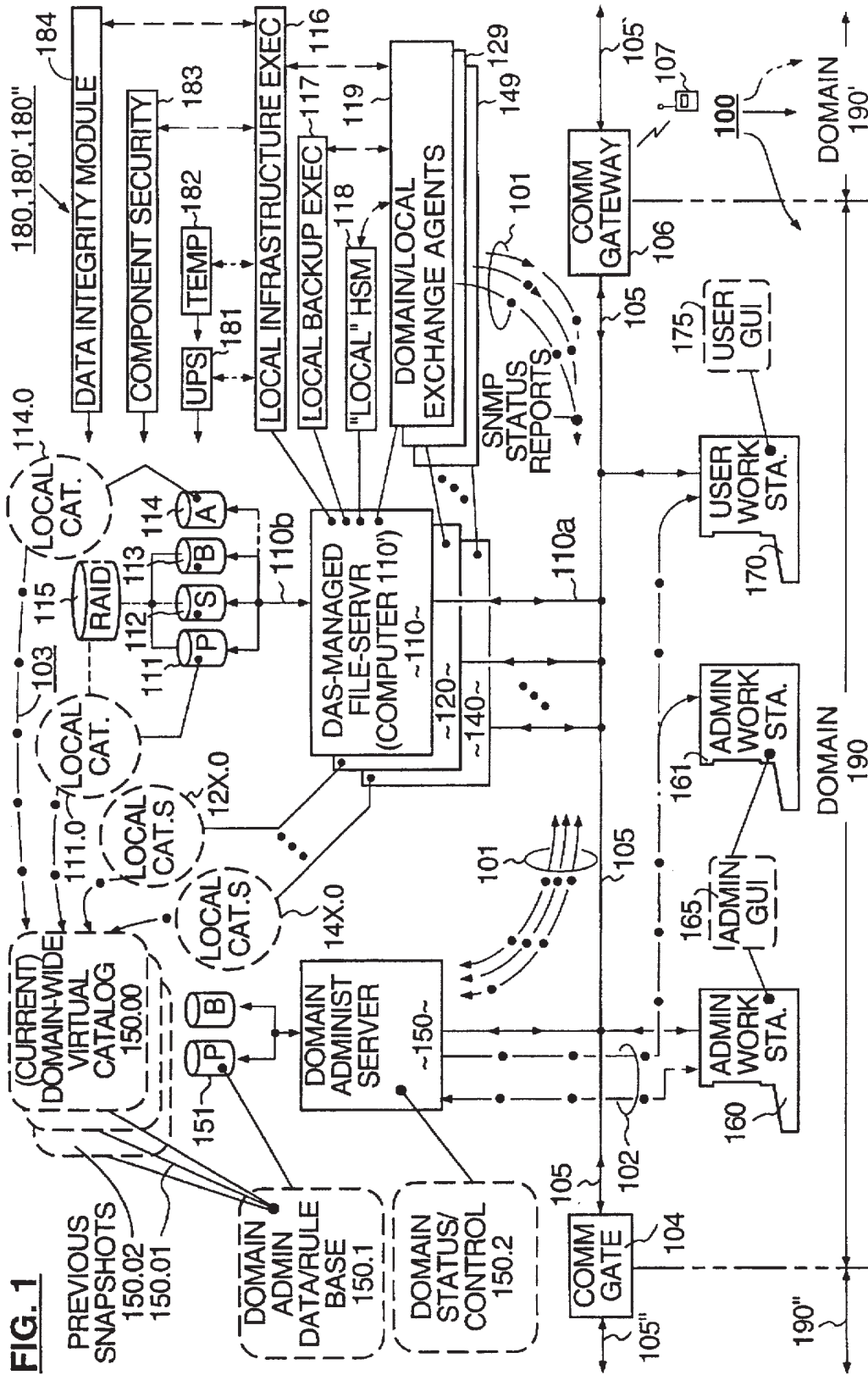


FIG. 1

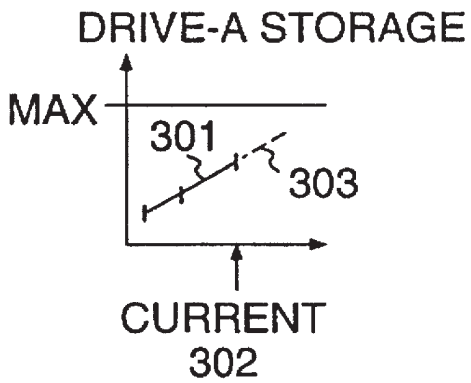
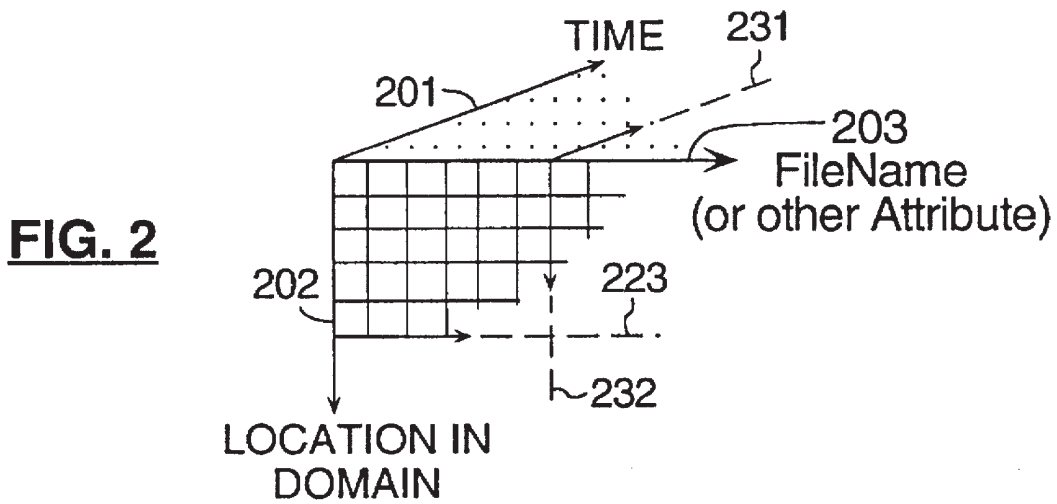


FIG. 3A

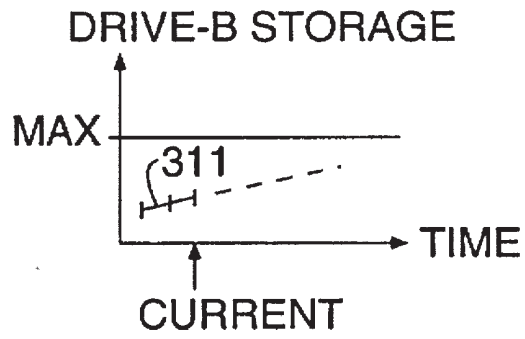


FIG. 3B

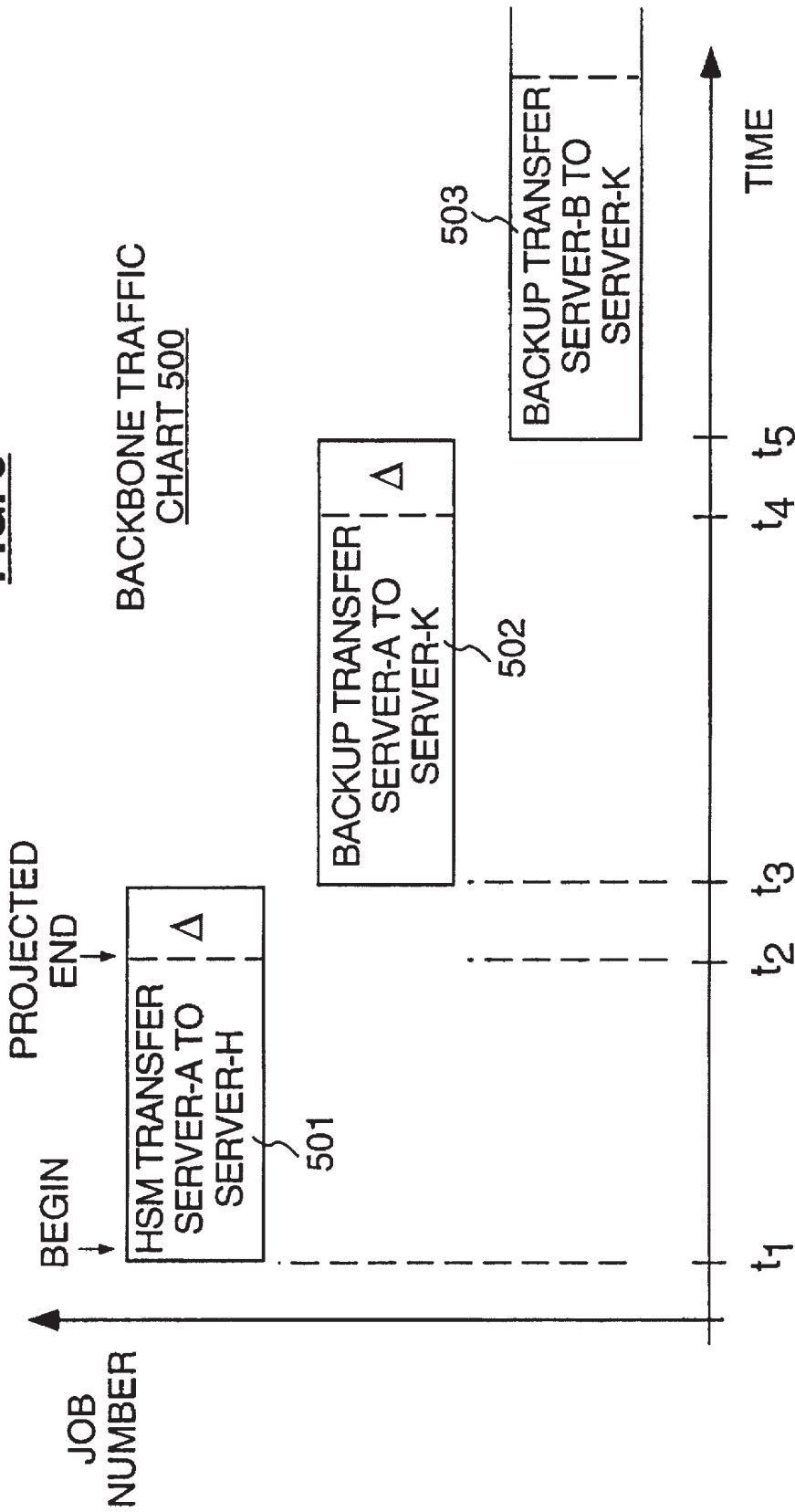


FIG. 4A



FIG. 4B

FIG. 5



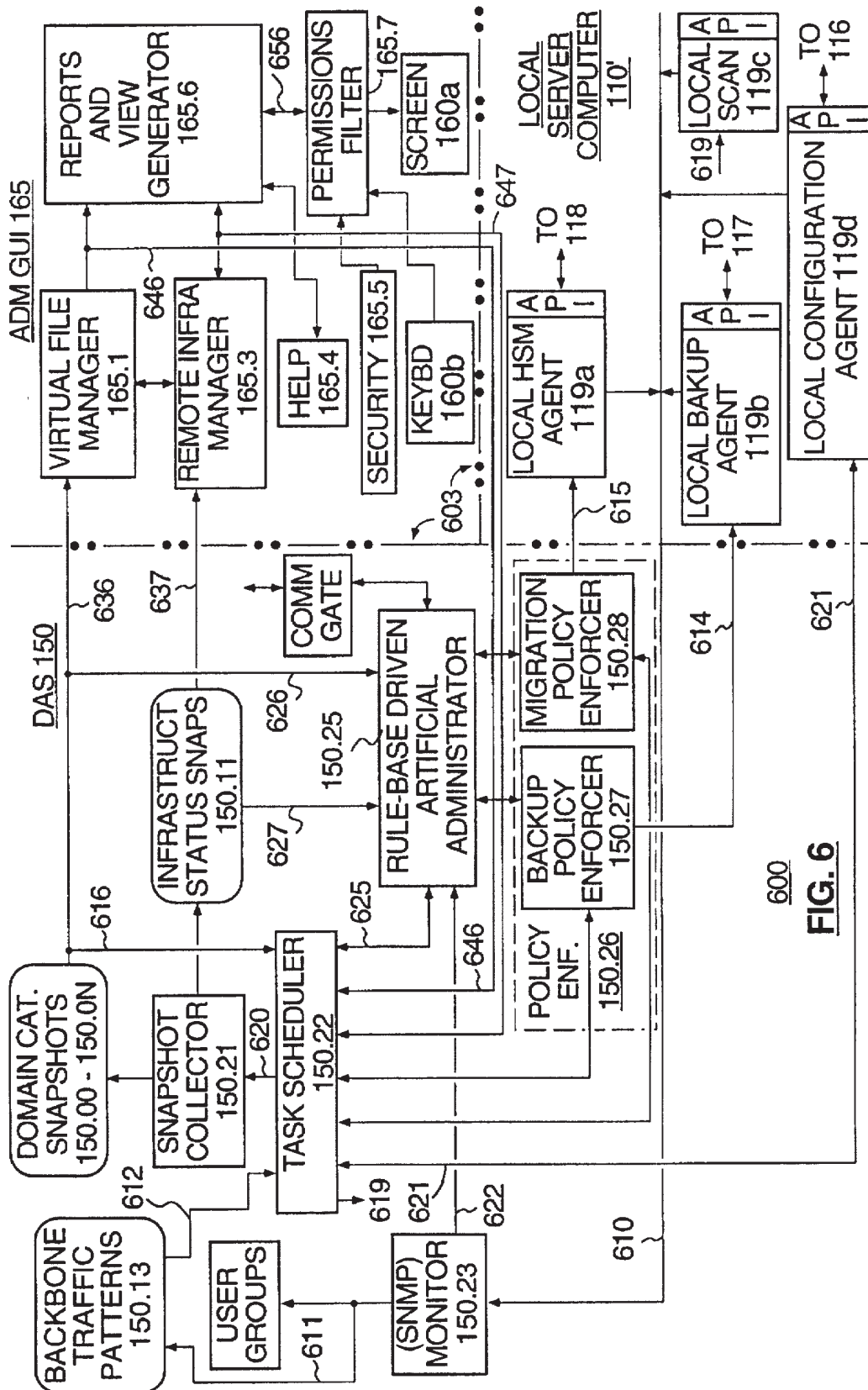


FIG. 6

**NETWORK MANAGEMENT SYSTEM
HAVING HISTORICAL VIRTUAL CATALOG
SNAPSHOTS FOR OVERVIEW OF
HISTORICAL CHANGES TO FILES
DISTRIBUTIVELY STORED ACROSS
NETWORK DOMAIN**

This application is a division of Ser. No. 08/153,011, filed Nov. 15, 1993, now U.S. Pat. No. 5,495,607.

BACKGROUND

1. Field of the Invention

The invention relates generally to the field of computerized networks. The invention relates more specifically to the problem of managing a system having a variety of file storage and file serving units interconnected by a network.

2. Cross Reference to Related Applications

The following copending U.S. patent application(s) is/are assigned to the assignee of the present application, is/are related to the present application and its/their disclosures is/are incorporated herein by reference:

(A) Ser. No. 08/151,525 filed Nov. 12, 1993 by Guy A. Carbonneau et al and entitled, SCSI-COUPLED MODULE FOR MONITORING AND CONTROLLING SCSI-COUPLED RAID BANK AND BANK ENVIRONMENT, and issued Dec. 17, 1996 as U.S. Pat. No. 5,586,250.

3. Description of the Related Art

Not too long ago, mainframe computers were the primary means used for maintaining large databases. More recently, database storage strategies have begun to shift away from having one large mainframe computer coupled to an array of a few, large disk units or a few, bulk tape units, and have instead shifted in favor of having many desktop or mini- or micro-computers intercoupled by a network to one another and to many small, inexpensive and modularly interchangeable data storage devices (e.g., to an array of small, inexpensive, magnetic storage disk and tape drives).

One of the reasons behind this trend is a growing desire in the industry to maintain at least partial system functionality even in the event of a failure in a particular system component. If one of the numerous mini/micro-computers fails, the others can continue to function. If one of the numerous data storage devices fails, the others can continue to provide data access. Also increases in data storage capacity can be economically provided in small increments as the need for increased capacity develops.

A common configuration includes a so-called "client/server computer" that is provided at a local network site and has one end coupled to a local area network (LAN) or a wide area network (WAN) and a second end coupled to a local bank of data storage devices (e.g., magnetic or optical, disk or tape drives). Local and remote users (clients) send requests over the network (LAN/WAN) to the client/server computer for read and/or write access to various data files contained in the local bank of storage devices. The client/server computer services each request on a time shared basis.

In addition to performing its client servicing tasks, the client/server computer also typically attends to mundane storage-management tasks such as keeping track of the amount of memory space that is used or free in each of its local storage devices, maintaining a local directory in each local storage device that allows quick access to the files stored in that local storage device, minimizing file fragmentation across various tracks of local disk drives in order to

minimize seek time, monitoring the operational status of each local storage device, and taking corrective action, or at least activating an alarm, when a problem develops at its local network site.

Networked storage systems tend to grow like wild vines, spreading their tentacles from site to site as opportunities present themselves. After a while, a complex mesh develops, with all sorts of different configurations of client/server computers and local data storage banks evolving at each network site. The administration of such a complex mesh becomes a problem.

In the early years of network management, a human administrator was appointed for each site to oversee the local configuration of the on-site client/server computer or computers and of the on-site data storage devices.

In particular, the human administrator was responsible for developing directory view-and-search software for viewing the directory or catalog of each on-site data storage device and for assisting users in searches for data contained in on-site files.

The human administrator was also responsible for maintaining backup copies of each user's files and of system-shared files on a day-to-day basis.

Also, as primary storage capacity filled up with old files, the human administrator was asked to review file utilization history and to migrate files that had not been accessed for some time (e.g., in the last 3 months) to secondary storage. Typically, this meant moving files that had not been accessed for some time, from a set of relatively-costly high-speed magnetic disk drives to a set of less-costly slower-speed disk drives or to even slower, but more cost-efficient sequential-access tape drives. Very old files that lay unused for very long time periods (e.g., more than a year) on a "mounted" tape (which tape is one that is currently installed in a tape drive) were transferred to unmounted tapes or floppy disks and these were held nearby for remounting only when actually needed.

When physical on-site space filled to capacity for demounted tapes and disks, the lesser-used ones of these were "archived" by moving them to more distant physical storage sites. The human administrator was responsible for keeping track of where in the migration path each file was located. Time to access the data of a particular file depended on how well organized the human administrator was in keeping track of the location of each file and how far down the chain from primary storage to archived storage, each file had moved.

The human administrator at each network site was also responsible for maintaining the physical infrastructure and integrity of the system. This task included: making sure power supplies were operating properly, equipment rooms were properly ventilated, cables were tightly connected, and so forth.

The human administrator was additionally responsible for local asset management. This task included: keeping track of the numbers and performance capabilities of each client/server computer and its corresponding set of data storage devices, keeping track of how full each data storage device was, adding more primary, secondary or backup/archive storage capacity to the local site as warranted by system needs, keeping track of problems developing in each device, and fixing or replacing problematic equipment before problems became too severe.

With time, many of the manual tasks performed by each on-site human administrator came to be replaced, one at a time on a task-specific basis, by on-site software programs. A first set of one or more, on-site software programs would

take care of directory view-and-search problems for files stored in the local primary storage. A second, independent set of one or more, on-site software programs would take care of directory view-and-search problems for files stored in the local secondary or backup storage. Another set of one or more, on-site software programs would take care of making routine backup copies and/or routinely migrating older files down the local storage migration hierarchy (from primary storage down to archived storage). Yet another set of on-site software programs would assist in locating files that have been archived. Still another set of independent, on-site software programs would oversee the task of maintaining the physical infrastructure and integrity of the on-site system. And a further set of independent, on-site software programs would oversee the task of local asset management.

The term "task-segregation" is used herein to refer to the way in which each of the manual tasks described above has been replaced, one at a time by a task-specific software program.

At the same time that manual tasks were being replaced with task-segregated software programs, another trend evolved in the industry where the burden of system administration was slowly shifted from a loose scattering of many local-site, human administrators—one for each site—to a more centralized form where one or a few human administrators oversee a large portion if not the entirety of the network from a remote site.

This evolutionary movement from local to centralized administration, and from task-segregated manual operation to task-segregated automated operation is disadvantageous when viewed from the vantage point of network-wide administration. The term "network-wide administration" is used here to refer to administrative tasks which a human administrator located at a central control site may wish to carry out for one or more client/server data storage systems located at remote sites of a large network.

A first major problem arises from the inconsistency among user interfaces that develops across the network. In the past, each local-site administrator had a tendency to develop a unique style for carrying out man-to-machine interactions. As a result, one site might have its administrative programs set up to run through a graphical-user interface based on, for example the Microsoft Windows™ operating environment, while another site might have its administrative programs running through a command-line style interface based on, for example the Microsoft DOS 6.0™ operating system or the AT&T UNIX™ operating system. A network-wide administrator has to become familiar with the user interface at each site and has to remember which is being used at each particular site in order to be able to effectively communicate with the local system administering software programs. Inconsistencies among the interfaces of multiple network sites makes this a difficult task.

Another problem comes about from the task-segregated manner in which local administrative programs have developed over the years. A remote human administrator (or other user) has to become familiar with the local topology of each network site when searching for desired files. In other words, he or she has to know what kinds of primary, secondary, backup and archive storage mechanism are used at each site, how they are connected, how data files migrate through them, and which "file manager" program is to be used to view the files of each type of storage mechanism.

More specifically, if a file cannot be found in the directory of a primary storage device located at a particular network site, the administrator has to switch from the primary storage viewing program to a separate, migration-tracking program

to see if perhaps the missing file has been migrated to secondary or archive storage at that site. The administrator may have to switch to a separate, backup-tracking program to see if a file that is missing from primary and secondary storage might be salvaged out of backup storage at the same or perhaps a different site. Sometimes, the administrator may wish to see a historical profile of a file in which revisions have been made to the file over a specified time period. A separate file-history tracking program at the site might have to be consulted, if it exists at all, to view such a historical profile.

If a file cannot be found at a first site then perhaps a copy might be stored at another site. To find out if this is the case, the administrator has to log out of the first site, log-in to the system at a next site and repeat the above process until the sought after data is located or the search is terminated.

Each switch from one site to a next, and from one independent file-managing program to another disadvantageously consumes time and also introduces the problem of inconsistent user interfaces.

A similar set of problems is encountered in the overseeing of lower-level infrastructure support operations of a networked data storage system. Included in this category are the scheduling and initiation of routine file backup and file migration operations at each site, the tracking of problems at each site and so forth.

A method and system for integrating all the various facets of system administration on a network-wide basis is needed.

SUMMARY OF THE INVENTION

The invention overcomes the above-mentioned problems by providing a network management system having virtual catalog overview function for viewing of files distributively stored across a network domain.

A network management system in accordance with the invention comprises: (a) a domain administering server (DAS) coupled to a network-linking backbone of a network domain for scanning the network domain to retrieve or broadcast domain-wide information, where the domain administering server (DAS) has means for storing and maintaining a domain-wide virtual catalog and for overseeing other domain-wide activities, and where the domain-wide virtual catalog contains file identifying information for plural files distributively stored in two or more file servers of the network domain; and (b) one or more workstations, coupled by way of the network-linking backbone to the domain administering server for accessing the domain-wide information retrieved by the domain administering server.

A method in accordance with the invention comprises the steps of: (a) interrogating the local catalog of each data storage device in a network composed of plural data storage devices linked to one another by a network-linking backbone, (b) retrieving from each interrogated local catalog, file identifying information identifying a name, a storage location and/or other attributes of each file stored in the interrogated device; and (c) integrating the retrieved file identifying information collected from each local catalog into a domain-wide virtual catalog so that each file stored on the network can be identified by name, location and/or another attribute by consulting the domain-wide virtual catalog.

BRIEF DESCRIPTION OF THE DRAWINGS

The below detailed description makes reference to the accompanying drawings, in which:

FIG. 1 is a block diagram showing a centralized domain management system in accordance with the invention;

FIG. 2 is a perspective view of a multi-dimensional viewing window for visualizing domain-wide activities spatially, temporally and by file attributes;

FIGS. 3A-3B show a set of trend analysis graphs that may be developed from the domain-wide, virtual catalog snapshots obtained by the system of FIG. 1;

FIGS. 4A-4B show side-by-side examples of pie charts showing used-versus-free storage space on respective storage drives DRIVE-A and DRIVE-B within the domain of FIG. 1;

FIG. 5 a job scheduling chart for minimizing traffic congestion on the network-linking backbone; and

FIG. 6 shows a logical flow map between various data and control mechanisms distributed amongst the domain administrating server (DAS), an administrative workstation, and a given server computer.

DETAILED DESCRIPTION

FIG. 1 is a block diagram of a networked enterprise system 100 in accordance with the invention.

Major components of the networked enterprise system 100 include: a network-linking backbone 105, a plurality of DAS-managed file-servers 110, 120, . . . , 140, operatively coupled to the backbone 105; and a domain administrating server (DAS) 150 also operatively coupled to the backbone 105.

The network-linking backbone 105 can be of any standard type used for forming local-area or wide-area digital data networks (or even metropolitan wide networks). Examples of standard backbones include Ethernet coaxial or twisted pair cables and token ring systems.

One or more communication gateways 104, 106 can link the illustrated backbone 105 to additional backbones 105', 105". The communications gateways 104, 106 may be of the wired type (e.g., high-speed digital telephone lines) or a wireless type (e.g., microwave or satellite links). As such the overall communications network -105'-104-105-106-105'-etc., can extend over long distances and pass through many geographic sites. Examples include communication networks which interlink different offices of a large building complex, or those which interlink multiple buildings of a campus, or those which interlink campuses of different cities or those that interlink transcontinental or global sites.

For purposes of administration, it is convenient to call the overall communications network -105'-104-105-106-105'-etc., and the resources connected to it, an "enterprise". It is convenient to subdivide the enterprise into a plurality of nonoverlapping "domains" The domains are logical subdivisions but may follow physical subdivisions. Examples of such subdivisions include but are not limited to: (a) subdividing a building-wide enterprise into floor-wide domains, one for each floor; (b) subdividing a corporate-wide enterprise into department-wide domains, one for each department of the corporate structure (e.g., accounting, marketing, engineering, etc.); (c) subdividing a multi-city enterprise according to the different cities it services; and so forth.

A block diagram of a first domain 190 within an enterprise system 100 in accordance with the invention is shown in FIG. 1. The enterprise system 100 can be composed of the one illustrated domain 190 or may have a plurality of like-structured or differently-structured domains connected to the illustrated first domain 190.

The aforementioned network-linking backbone 105 and plural file servers 110, 120, . . . , 140 are included within the first domain 190. The domain administrating server (DAS)

150 is also included within the first domain 190 as are a plurality of administrative workstations 160, 161, etc., and a plurality of user workstations 170, 171 (not shown), etc., which also connect to the network-linking backbone 105.

Although not shown, it is to be understood that numerous other data input and/or output devices can be connected to the network-linking backbone 105, including but not limited to: so-called "dumb" terminals which do not have a non-volatile mass storage means of their own, printers, label-makers, graphical plotters, modems, data acquisition equipment (analog-to-digital converters), digital voice and/or image processing equipment, and so forth. File-servers 110, 120, . . . , 140 may be used for storing or outputting the data created or used by these other data input and/or output devices.

Each file server 110, 120, . . . , 140 has associated with it: (1) a respective, local server computer 110', 120', . . . , 140'; (2) a set of one or more nonvolatile data storage devices (e.g. 111-114); and (3) a respective infrastructure 180, 180', . . . , 180" for supporting operations of the local server computer (e.g., 110') and its associated data storage devices (e.g. 111-114).

It is to be understood that communications gateway 106 can be used to link the first domain 190 to a variety of other structures, including a subsequent and like-structured second domain 190'. Similarly, communications gateway 104 can be used to link the first domain 190 to a variety of other structures, including a preceding and like-structured third domain 190". Data can be transferred from one domain to the next via the communications gateways 104, 106.

In addition to being able to communicate with other domains, each communications gateway 104, 106 can link via telephone modem or by way of a radio link to remote devices such as an administrator's home computer or an administrator's wireless pager (beeper) 107 and send or receive messages by that pathway.

The internal structure of the first of the DAS-managed file servers, 110, is now described as exemplary of the internal structures of the other DAS-managed file servers, 120, . . . , 140. The term "DAS-managed" indicates, as should be apparent by now, that each of file servers 110, 120, . . . , 140 is somehow overseen or managed by the Domain Administrating Server (DAS) 150. Details of the oversight and/or management operations are given below.

The first DAS-managed file server 110 includes a client/server type of computer 110' which is represented by box 110 and referred to herein as the "local server computer 110'". Server computer 110' is understood to include a CPU (central processing unit) that is operatively coupled to internal RAM (random access memory) and/or ROM (read-only memory). Examples of client/server type computers that form the foundation for server computer 110' include off-the shelf tower-style computers that are based on the Intel 80486™ microprocessor and come bundled with appropriate client/server supporting hardware and software.

The local server computer 110' of the first DAS-managed file-server 110 has a network interface port 110a that operatively couples the server computer 110' to the network-linking backbone 105 and a mass-storage port 110b that operatively couples the server computer 110' to one or more of: a primary mass storage means 111, a slower secondary storage means 112, a backup storage means 113, and an archived-data storage and retrieval means 114.

The primary storage means 111 can be a high speed Winchester-type magnetic disk drive or the like but can also include battery-backed RAM disk and/or non-volatile flash-

EEPROM disk or other forms of high-performance, non-volatile mass storage.

The secondary storage means 112, if present, can include a slower WORM-style optical storage drive (Write Once, Read Many times) or a "floptical" storage drive or other secondary storage devices as the term will be understood by those skilled in the art. (Secondary storage is generally understood to cover mass storage devices that have somewhat slower access times than the associated primary storage but provide a savings in terms of the cost per stored bit.)

The backup storage means 113 can include magnetic disk drives but more preferably comprises DAT (Digital Audio Tape) drives or other forms of tape drives or other cost-efficient backup storage devices. A backup copy of each file held in primary or secondary storage (111, 112) is preferably made on a periodic basis (e.g., nightly or every weekend) so that a relatively recent copy of a given file can be retrieved even in the case where the corresponding primary or secondary storage means (111, 112) suffers catastrophic failure; e.g., a head crash or destruction.

The archived-data storage and retrieval means 114 typically comes in the form of an archive create/retrieve drive and an associated set of removable tapes or removable disk cartridges. Most if not all of the associated set of removable archive tapes and/or removable archive disk cartridges are not physically mounted to the archive create/retrieve drive (as indicated by the dashed connection line) and are thus not immediately accessible to the server computer 110'. They can be mounted when requested and thereafter accessed.

Note: The above description is intended to be generic of the types of nonvolatile mass storage means 111-114 that might be connected to the mass-storage port 110b of the server computer 110'. In theory, each server computer can have all of the primary (P), secondary (S), backup (B) and archive (A) storage means (111-114) connected to its mass-storage port 110b. Due to cost and performance considerations however, a typical set-up will instead have one or more "groups" of server computers to which primary but not secondary storage means is connected. Each such server computer will be referred to as a primary file server. A second set of server computers will have secondary but not primary storage means connected to them and will be each referred to as a secondary or "HSM" file server and will each service a particular "group" of primary file servers. A secondary file server is sometimes also referred to as a "storage server".

Backup storage means (e.g., a tape cartridge drive) may be provided either on a one-for-one basis for each server computer or one server computer might be dedicated for generating backup tapes/disks for a pre-assigned group of primary and/or secondary file servers.

Archive storage can be similarly configured on a one-for-one basis for each server computer or one server computer might be dedicated for creating and retrieving archive tapes/disks for an associated group of primary and/or secondary file servers.

The data files of the primary, secondary and backup storage means 111-113 can be organized conventionally or distributed redundantly across a plurality of drives in accordance with a practice known as RAID (Redundant Array of Inexpensive Data-storage drives). A detailed description of the intricacies involved in managing a RAID system may be found in the above-cited patent application, SCSI-COUPLED MODULE FOR MONITORING AND CONTROLLING SCSI-COUPLED RAID BANK AND BANK ENVIRONMENT, which application is incorporated herein

by reference. As such these will not be detailed here. In brief, each file is distributively stored across two or more storage drives so that failure of a single drive will not interfere with the accessibility or integrity of a stored file. The dashed symbol 115 for a RAID bank indicates the possibility of file distribution across redundant drives.

The above-cited application also details the intricacies involved in maintaining an infrastructure 180 for supporting various operations of the data storage devices 111-113 of a given server computer, and as such these will not be detailed here either. In brief, the infrastructure 180 of the server computer 110' preferably includes an uninterruptible power supply means (UPS) 181 for supplying operational power to the local data storage devices 111-113 and to the local server computer 110'. A local temperature control means 182 (e.g. cooling fans) may be included in the infrastructure 180 for controlling the temperatures of the local devices 110', 111-113. A local component security means 183 (e.g. a locked, alarmed cabinet) may be provided for assuring physical security of one or more of the local components 110', 111-113 (and also, if desired, of the archived-data storage means and tapes 114). A local data path integrity checking module 184 may be further included within the local infrastructure 180 for assuring proper interconnections by cable or otherwise between units 110' and 111-113 so that data is properly transferred from one to the other.

A local infrastructure support program 116 is preferably loaded into the local server computer 110' for monitoring and managing one or more of the local infrastructure components 181-184 coupled to it and its associated data storage units 111-114.

A local backup execution program 117 is also preferably installed in the local server computer 110' for routinely making, or at least requesting, backups of various data files held in the local primary and secondary storage means 111-112. (Aside: As will be understood from the below discussion, a disadvantageous traffic congestion condition may develop on the network-linking backbone 105 as a result of many primary file servers all trying to backup their files at one time to a shared backup server. To avoid this, backup making is preferably controlled on a domain-wide basis by a backup-scheduler and policy-enforcer which is contained in the box numbered 150.2 and which will be described in more detail below. The local backup execution program 117 sends requests to the backup scheduler/policy enforcer and receives execution commands or permissions from the scheduler/policy enforcer 150.2. These backup commands/permissions are issued on the basis of a rule base 150.1 that tries to minimize traffic congestion and balance workloads along network-linking backbone 105 by appropriate scheduling.)

A "local" hierarchal storage migration (HSM) control program 118 may also be installed in the local server computer 110' for managing the migration of less-often used files from primary storage 111 to secondary storage 112. As explained above, a typical set-up will have one dedicated, HSM file server providing migration services to a designated "group" of primary file servers. Domain 190 can have plural, nonoverlapping "groups" of primary file servers and in such a case, each group will have its own dedicated, HSM file server. When dedicated HSM file servers are used, the hierarchal storage migration (HSM) control program 118 will typically reside only inside the dedicated HSM file servers. (As will be understood from the below discussion, if two HSM servers try to perform their migration operations at the same time, it is possible that such operations will lead to excessive traffic congestion on the shared network-linking

backbone 105. As such, migration of files between primary and secondary storage is preferably controlled on a domain-wide basis by a migration-scheduler and policy-enforcer which is contained in the box numbered 150.2 and which will be described in more detail below. The local hierarchical storage migration control program 118 sends requests to the migration scheduler/policy enforcer and receives execution commands or permissions from the scheduler/policy enforcer 150.2. These migration commands/permissions are issued on the basis of a rule base 150.1 that tries to minimize traffic congestion and balance workloads along network-linking backbone 105 by appropriate scheduling.)

A plurality of domain/local exchange agent programs 119 are also preferably loaded in the server computer 110 for cooperatively interacting with the domain administrating server 150 as will be explained shortly. Note that a filled circle, with a line extending from it to a corresponding software structure, is used to indicate that the software structure is installed on the particular server computer.

Each of the primary (P), secondary (S), backup (B) and archive (A) storage means 111-114 has a local catalog defined within it for identifying the name, location and other attributes of each file stored therein. The local catalog will also typically store information describing each directory full of files or full of subdirectories that is defined therein, and each volume full of directories that is defined therein.

The file-locating information in the local catalog may include a name (ServerName) given to the associated server computer. The file-locating information may also include a set of partition definitions: (a) for partitioning the physical storage media of a given server computer into a set of logical "volumes", (b) for assigning a unique name (VolumeName) to each volume, (c) for indicating the number of files (VolumeFileCount) stored in each volume, (d) for indicating the total storage capacity (VolumeSizeInBytes) of the volume, the amount of used space (VolumeActiveBytes), and the amount of free space (VolumeInactiveBytes).

which the file is located. The second item is a logical volume name assigned to a logical partition of the local storage means. For Microsoft DOS™ based machines, the volume names of hard drive partitions typically take on a name such as C:, D:, E:, and so forth. The remaining items in the pathname define a path through directories and subdirectories as will be understood by those skilled in the art. The last item is the name of the file being sought.

Aside from storage location and FileName the other attributes indicated in the local catalog may include but are not limited to: (1) File Size (e.g. in number of bytes); (2) File Chronology in terms of Creation date and time, latest Modify or revision date and time, latest read-only Access date and time, and latest Archive date and time; (3) File User information in terms of who is the "Owner" or original creator of the file, who was the LastModifier of the file, who has read/write/execute permission for this file, and so forth.

Yet further attributes may link the particular file to other data structures and provide system-level control and indicator bits such as the following list of Novell-defined attributes: System, ReadOnly, ExecuteOnly, Subdirectory, Archive, Shareable, Compress, Salvageable, Purgeable, Migrated, Indexed, ReadAudit, WriteAudit, ImmediatePurge, RenameInhibit, DeleteInhibit, CopyInhibit, ImmediateCompress, CompressInhibit and Uncompressable. File attributes for other standard network operating systems such as UNIX and Microsoft WindowsNT™ are also contemplated.

The local catalog may be arranged according to many well-known organizations including a tree organization which starts at a root directory and defines a path name from the root directory through subdirectories to a desired file. The below Table 1 gives an example of the kind of information that might be stored in a subdirectory of a local catalog.

TABLE 1

Path=AcctServr Cvolume Accounts New 4 Files						
File_name	File_size (KBytes)	Last_Rev (yyymmdd hh:mm)	By	First_Ver (yyymmdd hh:mm)	Owner	
dave.doc	1546	931004 09:15	tom	921224 16:45	dave	
dave.do1	1297	931105 11:23	tom	921224 12:25	dave	
tom.doc	1881	930906 09:15	dave	910115 09:45	tom	
paul.doc	1965	931107 11:23	tom	921224 12:25	paul	

Volumes are typically subdivided logically into a root directory and a plurality of subdirectories. The file-locating information in the local catalog will usually include a name (DirectoryName) given to each such subdivision and a count of the number of included files (FileCount).

Because each directory can be logically subdivided into subdirectories wherein a desired file might be found, the file-locating information in the local catalog will usually define a "pathname" for uniquely identifying each file according to the path followed from a root point to the subdirectory that holds the file. A typical pathname has its branches separated by the backslash symbol ("\") and takes on the form:

Path=: ServerName\VolumeName\RootDirectory\
Subdirectory\Subdirectory\...\Subdirectory\FileName
The first item in the pathname is the name of the physical server computer that controls the storage drive or drives in

Note that the information in the local catalog (Table 1) is only for the files stored on the local storage medium (e.g., the primary storage means 111) and does not cover files stored in other storage media, either at the same network site or elsewhere in the domain.

A dashed bubble at 111.0 is used in FIG. 1 to represent the contents of the local catalog for the primary storage means 111 of the first DAS-managed file-server 110. It is to be understood that if there is a secondary storage means 112 present within first file-server 110, such a second storage 112 will have its own local catalog 112.0. A bubble for the secondary storage local catalog 112.0 is not shown in FIG. 1 due to space limitations. Similarly, the backup storage means 113, if present, will have its own local catalog 113.0 (not shown) and the archive storage means 114, if present, will have its own local catalog 114.0 (not shown).

Additionally, if files are distributed across plural drives in accordance with RAID technology (115), each local catalog may be organized to identify the locations within each of the plural drives where redundant information for a given file is stored.

Although not fully shown, it is to be understood that the second DAB-managed file-server 120 has an internal structure that is generically similar to that of the first file-server 110. The physical components of the second file-server 120 may be located at a different network site from that of the first file server 110 and the characteristics of the components in the second DAS-managed file-server 120 may differ substantially from those of the first file-server 110. This is not to say that the second DAB-managed file-server 120 cannot be alternatively located at the same site and/or have substantially similar components as those of the first file-server 110. Rather it is to indicate that the second DAS-managed file-server 120 is to be thought of as having its own independent structure and that this structure, in its specifics, may be similar to or different from that of the first file-server 110.

More specifically, it is to be understood although not shown, that the second DAS-managed file server 120 has one or more of its own primary storage means 121, secondary storage means 122, backup storage means 123 and archive storage means 124. A RAID structure 125 (not shown) may or may not be provided within DAS-managed file server 120. Each of storage means 121-124, if present in the second DAS-managed file-server 120, has a corresponding local catalog 121.0-124.0 (not shown).

The combination of local catalogs 121.0-124.0 associated with the second DAS-managed file-server 120 is represented by dashed bubble 12X.0 in FIG. 1. (In similar vein, the combination of the earlier-described local catalogs 111.0-114.0 associated with the first DAS-managed file-server 110 will be referred to as 11X.0.)

For purpose of example, it will be assumed that the second DAS-managed file server 120 is located at a second site which is quite remote from the location site of the first DM-managed file server 110. Hence the combination of local catalogs 121.0-124.0 associated with the second DAS-managed file-server 120 will be referred to as the second site local catalogs 12X.0.

In addition to having its own bank of storage devices 121-124, the second file server 120 has its own infrastructure support system 180'. Infrastructure support system 180' is similar to that of the earlier described system 180 with the exception that the second infrastructure support system 180' is located at the second site together with the remainder of second file-server 120. Although not shown, the corresponding UPS, temperature control, component security, and data-path integrity-check modules of the second infrastructure 180' will be respectively referenced as 181', 182', 183' and 184'.

It is to be understood that a local infrastructure support program 126 is installed in server computer 120' of the second DAS-managed file-server 120 just as infrastructure support program 116 is installed in server computer 110' of the first site. A symbol for infrastructure support program 126 is not shown in FIG. 1 in order to avoid illustrative clutter. It is to be similarly understood that the second-site server computer 120' may have a local backup execution program 127 (not shown) installed therein, and a local hierarchical storage migration control program 128 (not shown) installed therein.

Furthermore, the second file server computer 120' has its own set of domain/local exchange agent programs 129

installed therein. These domain/local exchange agents 129 are used for cooperatively exchanging messages between the domain administrating server (DAS) 150 and portions of second-site server computer 120' as will be explained below.

5 A symbol for exchange agent programs 129 is shown in FIG. 1 just behind the symbol for domain/local exchange agents 119.

It is to be understood that physical communication signals between any two or more of the DAS-managed file servers 110, 120, . . . , 140 and the domain administrating server 150 travel along the network-linking backbone 105. Sets of dash-dot lines are included in FIG. 1 for showing the logical flow of certain communications.

In particular, a first bundle of such dash-dot lines 101 is drawn to represent a first flow of communications between the domain administrating server 150 and the domain/local exchange agents 119, 129, . . . , 149 (the last one belongs to server computer 140'). The first logical communications flow 101 includes: catalog snapshot data requests and responses; migration and backup scheduling requests/commands; SNMP (Simple Network Management Protocol) reports of status or alerts; and other types of information as will be explained later.

A second such bundle of dash-dot lines 102 represents a second flow of logical communications 102 which take place between the domain administrating server 150 and administrative or user workstations 160, 161, . . . , 170. The second logical communications flow 102 includes requests for information on domain-wide status and trends as will be explained below.

A third bundle of such dash-dot lines 103 represents a third logical flow of communications which take place between the local catalogs 11X.0, 12X.0, . . . , 14X.0 of respective systems 110, 120, . . . , 140 and a soon-to-be described domain-wide virtual catalog 150.00 (current snapshot) that is defined in the primary storage 151 of the domain administrating server 150.

As should be already apparent, any additional number of DAS-managed file servers similar to above-described file servers 110 and 120 can be connected to the network-linking backbone 105 and each such additional server can be located at a site remote from the sites of the other servers or at a same site. For purposes of illustration, the third DAS-managed file server 140 is shown as the last in a chain of such file servers 110, 120, . . . , 140. Like the above-described other file servers 110 and 120, the third file server 140 is understood to have its own infrastructure 180", its own set of data storage means 141-144 (not shown) and its own set of installed programs 146-149 (not shown except for last item). The last named item, 149, represents the domain/local exchange agent programs 149 of system 140 and a symbol for this collection of field agent programs is shown in FIG. 1 lying beneath the symbol for 129. The combination of local primary, secondary backup and archive catalogs for system 140 are represented by bubble 14X.0. The third logical communications flow 103 collects the contents of catalogs 11.X, 12.X, . . . , 14.X and integrates them into a soon-described, domain-wide virtual catalog 150.00.

The domain administrating server (DAS) 150 has a client/server type of computer 150' similar to those of already described servers 110, 120, . . . , 140 with the exception that the domain administrating server 150 is not intended to store individually-owned user files for use by individual network users. Instead, the mass storage means (e.g. 151) of the

domain administrating server 150 is used primarily for storing information related to domain-wide activities.

A first domain-wide activity that is supported by the domain administrating server (DAS) 150 is the maintenance of a current-snapshot of a domain-wide "virtual" catalog 150.00. Although the non-volatile data storage means 151 of the domain server 150 does not contain all the data stored in all the various data storage means 111, 112, . . . , 143, 144 of the remainder of the domain 190, the virtual catalog 150.00 is constructed to create an impression of such a case. Hence the name domain-wide "virtual" catalog is given to the data structure represented by dashed bubble 150.00.

The information of the domain-wide virtual catalog (current snapshot) 150.00 is held in a domain administrating data/rule base 150.1. The database portion of this construct 150.1 is preferably of a relational database type so that entries can be conveniently updated and searched according to a variety of known database update and query schemes. One embodiment of the invention uses a relational-style database package bearing the tradename MDBS IV™ which is available from Micro Data Base Systems Inc. of Lafayette, Ind. (U.S. postal zip code 47902).

A domain-wide status-monitor and control program 150.2 is installed in the domain administrating server 150. One of the domain-wide status monitoring functions of program 150.2 is to: (1) periodically scan the domain 190 and interrogate each DAS-managed file-server 110, 120, . . . , 140 regarding the contents of each local catalog 111.0, 112.0, . . . , 144.0 that is associated with each of data storage device 111, 112, . . . , 144 in the network domain 190; (2) to collect the file identifying information stored at a given scan time in each such local catalog by way of the network-linking backbone 105, and (3) to integrate the collected information into the domain-wide virtual catalog 150.00 so that each user file stored in the domain 190 during a given scan-period can be identified by time-of-scan, file-name, location or other relevant attributes simply by consulting the domain wide virtual catalog 150.00.

Each time a new scan of the domain 190 is carried out, and new information is collected, the older information which was collected by previous scans is preferably retained and re-labeled as belonging to the appropriately-dated previous scan rather than being discarded. A historical collection is thereby created. There will be some point, of course, when it will be desirable or necessary to discard older data. An example is where the used-space of storage means 151 begins to reach maximum capacity. In such a case, non-essential older data should be archived or discarded to make room for more recent data.

The data of each DAS scan is referred to as a "snapshot". After a number of time-spaced snapshots are taken, the domain administrating data/rule base 150.1 defines a historical record for every file in the domain 190. FIG. 1 shows two previous snapshots 150.01 and 150.02 behind current snapshot 150.00. There, of course, can be many more.

Each snapshot 150.00, 150.01, 150.02, etc., of the domain-wide virtual catalog should, of course, include information indicating the time of the respective domain-wide scan. The stored snapshot information should also indicate which physical server provided the file-status information and which local catalog served as the source of each file-identifying entry.

Consider the below Table 2 which shows an example of what might be displayed to an inquiring administrator or user when the domain administrating data/rule base 150.1 of the domain server is queried for virtual catalog information.

TABLE 2

Listing= Virtual_Domain.Catalog		99999999999999999999 Files			
(Snapshot Period: 900101 to 931130)					
File_Source	File_Name	File_Size (KBytes)	Last_Rev (yyymmdd hh:mm)	By	...
Acct111.0 . . .	dave.doc	1546	931004 09:15	tom	. . .
Acct111.0 . . .	dave.doc	[1544]	931003 17:35	paul	. . .
Acct111.0 . . .	dave.doc	[1543]	931002 14:22	dave	. . .
Acct111.0 . . .	dave.do1	1297	931105 11:23	tom	. . .
Acct111.0 . . .	paul.doc	1965	931107 11:23	tom	. . .
Acct112.0 . . .	tom.doc	1881	930906 09:15	dave	. . .
Acct112.0 . . .	tom.do2	0000	930906 09:15	dave	. . .
AcctBak.0 . . .	dave.doc	1544	931003 11:59	paul	. . .
AcctBak.0 . . .	dave.doc	[1543]	931002 11:59	dave	. . .
AcctBak.0 . . .	dave.doc	[1541]	931001 11:59	tom	. . .
AcctBak.0 . . .	dave.do1	1281	931104 11:59	tom	. . .
AcctBak.0 . . .	tom.doc	1872	930905 11:59	dave	. . .
AcctBak.0 . . .	paul.doc	1953	931106 11:59	tom	. . .
AcctArc.1 . . .	dave.doc	1530	911011 23:55	tom	. . .
AcctArc.1 . . .	dave.do1	1260	921101 23:56	tom	. . .
AcctArc.1 . . .	tom.doc	1850	910901 23:57	dave	. . .
AcctArc.1 . . .	paul.doc	1940	901101 23:58	tom	. . .
AcctArc.2 . . .	tom.do2	1776	900906 09:15	dave	. . .
Mktg121.0 . . .	dave.doc	1544	920704 09:15	tom	. . .
Mktg121.0 . . .	dave.do1	127	931105 11:23	tom	. . .

The query-results header of TABLE 2 is constructed to resemble the result of a Microsoft-DOS DIR *.* list command, with a first exception being that it names a directory that happens to be very large (99999999999999999999 files). A second exception is that it defines a snapshots-taking period (e.g., 900101 to 931130).

The first column of Table 2 shows the file source pathname as beginning with a file-server name (e.g., Acct111.0\ . . . or Mktg121.0\ . . .) rather than simply with a volume or root directory name. The second column shows the file name. Due to space limitations, only the ServerName is shown, but it is understood that the contents of column one can be expanded out to show a full pathname including VolumeName and one or more related directories.

Note that the same file name may appear multiple times in the listing of Table 2, even with identical path names (e.g., "dave.doc"). The difference lies in the date of creation or revision fields. List rows with square brackets around their size field do not have corresponding files still existing within the domain. Rather they are historical shadows that may be used for auditing the trail of who last modified the file, when, and to what extent did file size grow or shrink.

In addition to historical shadows, existing versions of a file may be distributed through out the domain. One version of a file with a same file-name might appear in the primary storage of a particular file server while a second version appears in backup storage and a third version appears in archive storage. Some copies of the same file may be distributed among different file servers.

In the example of TABLE-2, a first file named "dave.doc" is stored in the primary storage server of the accounting department (source=Acct111.0\ . . .) with the current version being the one most recently revised and historical shadows being indicated by square brackets (e.g., "[1544]") about their file size entries. Another same-named file is stored in the backup storage server (source=AcctBak.0\ . . .) and yet another same-named file is stored in the archive storage server (source=AcctArc.1\ . . .) and still another version of the same-named file is stored in the primary storage of the marketing department (source=Mktg121.0\ . . .). The size-zero file entry for tom.do2 in the Acct112.0 server is merely

a place marker indicating the actual file has been moved to archive (to AcctArc.2).

The domain-wide catalogs-snapshot information (150.00, 150.01, 150.02, etc.) that is stored within the domain administering data/rule base 150.1 can be used to quickly generate the above TABLE 2 using a database query and thereby give an administrator or user information about a given file or set of files even if they are distributed amongst different file servers and in different kinds of storage (e.g., primary, secondary, backup and archive). The database structure of the domain-wide catalogs-snapshot information (150.00, 150.01, 150.02, etc.) provides a quickly-accessible, multi-dimensional view of what is happening across the domain both spatially and temporally as well as by file attributes.

The searchable database fields preferably include: ScanTime (which is the time period for which the collected data was current); FileName; PathName (the storage path-name including searchability by ServerName); FileSize; FileChronology (in terms of for example, Creation date and time, latest Modify or revision date and time, latest read-only Access date and time, and latest Archive date and time); FileUser information (in terms of who is the "Owner" or original creator of the file, who was the LastModifier of the file, who has read/write/execute permission for this file, and so forth); and Netware attributes such as the Novell-defined attributes: System, ReadOnly, ExecuteOnly, Subdirectory, Archive, Shareable, Compress, Salvageable, Purgeable, Migrated, Indexed, ReadAudit, WriteAudit, ImmediatePurge, RenameInhibit, DeleteInhibit, CopyInhibit, ImmediateCompress, CompressInhibit and Uncompressable.

FIG. 2 illustrates this concept graphically in terms of a three-dimensional cube. The orthogonal XYZ axes are labeled 201-203. Axis 201 represents time of snapshot. Axis 202 represents location within the domain either in physical, global terms or within a type of storage media such as primary (P), secondary (S), backup (B) and/or archive (A). Axis 203 represents variation in file name and/or other file attributes.

Line 231 runs along the 201/203 plane in the 201 direction and represents a listing of file attribute versus time. By way of simple example, line 231 may represent a listing of all versions over time of a particular file having a specified name (e.g., dave.doc). A database query can be written to search the domain-wide virtual catalog snapshots, 150.00, 150.01, 150.02, etc., and generate a listing of all files located along line 231 as will be understood by those skilled in the art. Wildcard searches of the form, for example, "LIST dav?.do*" can broaden the scope of the search and list operation to cover files whose names conform to a particular pattern.

Line 232 runs along the 202/203 plane in the 202 direction and represents a listing of file attribute versus location. By way of simple example, line 232 may represent a listing of all copies of files having a specified name (e.g., dave.doc) that have been stored ever or during a specified time period in any one or more of: primary (P), secondary (S), backup (B) and archive (A) storage. A database query can be written to search the domain-wide virtual catalog snapshots, 150.00, 150.01, 150.02, etc., and generate a listing of all files located along line 232, or only those in a selected subset of the P, S, B and A storage categories as will be understood by those skilled in the art.

Line 223 runs along the 202/203 plane in the 203 direction and represents a listing of location versus file attributes. By way of simple example, line 223 may represent a listing of all files stored at a particular time in a specified domain

location (e.g., archive storage) that satisfy the name search criteria, "LIST day?.do*" where the symbol "?" represents a single-character wild card and "*" represents an N-character wild card.

The database information contained in the domain-wide virtual catalog snapshots, 150.00, 150.01, 150.02, etc., may be sorted according to any number of other schemes to provide the system administrator with all sorts of information including, but not limited to: (1) locating the latest version of a particular file regardless of where it is stored within the domain; (2) determining which files have been properly backed up in a recent time period and which have not, (3) reviewing the performance of the hierarchical storage management system (HSM) at each site and (4) viewing a historical snapshot of a file as various versions of it are migrated from primary to secondary storage and then perhaps to archive.

In the above listing of Table 2 for example, plural instances of the document named "dave.doc" appear. It can be determined from the listing that the latest version of "dave.doc" was stored in the primary storage server Acct111.0 of the Accounting Department on Oct. 4, 1993, while a backup copy was stored on the AcctBak.0 server the day before and yet another version was placed in an archive tape or disk volume named AcctArc.1 on Oct. 1, 1991. Yet another copy of the document "dave.doc" is found in the Marketing Department on a primary storage server named Mktg121.0, and it was last modified by a user named "tom" on Jul. 4, 1992.

Since the contents of the domain-wide virtual catalog snapshots, 150.00, 150.01, 150.02, etc., are held in a database 150.1 whose contents can be searched according to all type database queries (e.g., using a Standard Query Language or SQL) the types of information that can be gleaned is up to the system administrator's creativity. A system administrator might wish to simply track the progress of a particular file according version number (e.g., latest modification dates and times) or he or she may wish to study the migration patterns of certain classes of files from primary (P) to secondary (S) to archive (A) storage and from any of those to backup (B) storage. A search primarily on file name and secondarily on volume location and dates will reveal a historical view of where and when various versions of like-named files have been stored to secondary, backup and archive media.

Depending on the fields chosen as primary, secondary or tertiary search fields, if any, a system administrator can easily obtain views of various domain-wide activities, including but not limited to: (1) the amount of total primary storage available at each file server site—this is most quickly determined by looking at the total active bytes in the collective VolumeSize entries of the servers on-site, (2) the amount of used versus free storage space available in the primary storage of each file server, (3) the size, age and types of files that have been most recently moved to secondary storage at each file server by the local hierarchal storage management (HSM) programs 118, 128, . . . 148, (4) the size, type and age of all files that have been most recently moved from primary storage to backup storage across the domain 190 by the respective local backup execution programs 117, 127, . . . , 147 of the system, (5) the size, type, and age of files that have been archived to various archive volumes across the domain, (6) the distribution of files having particular file names or file types across primary, secondary, backup and archive storage as seen across the domain 190, including the date of last revision and storage into each type of storage media or the date of last read-only access, and so forth.