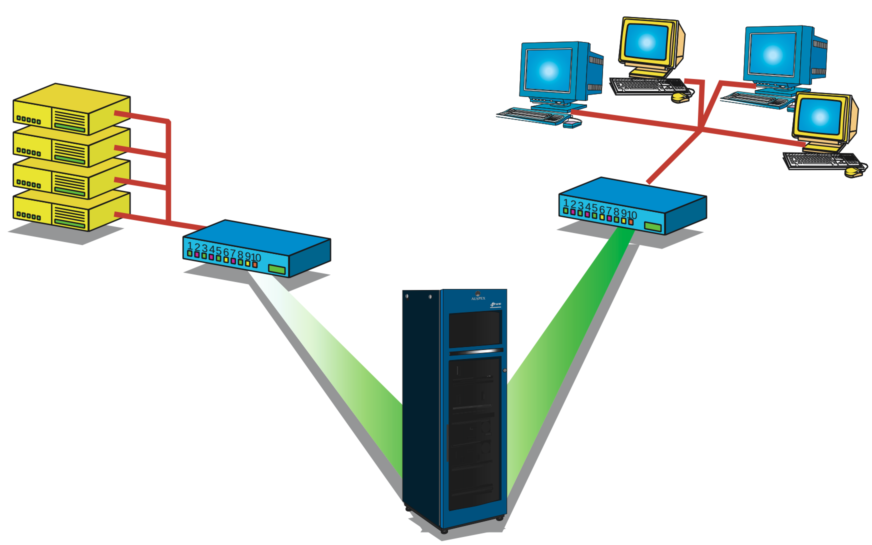
Auspex NS2000

Optimally deploying NAS in a large scale software development environment.

Software Development Network & Storage Architecture Guide



Auspex Technical Report

AUSPEX

Auspex NS2000

Optimally deploying NAS in a large scale software development environment.

The Importance of Computing Infrastructure Design in Computer Aided Software Engineering (CASE)

Situation Analysis - Software Development process, objectives and challenges

Case Study - BillSoft Inc. develops a wireless billing system

Designing a hierarchical, functionally specialized network and storage architecture

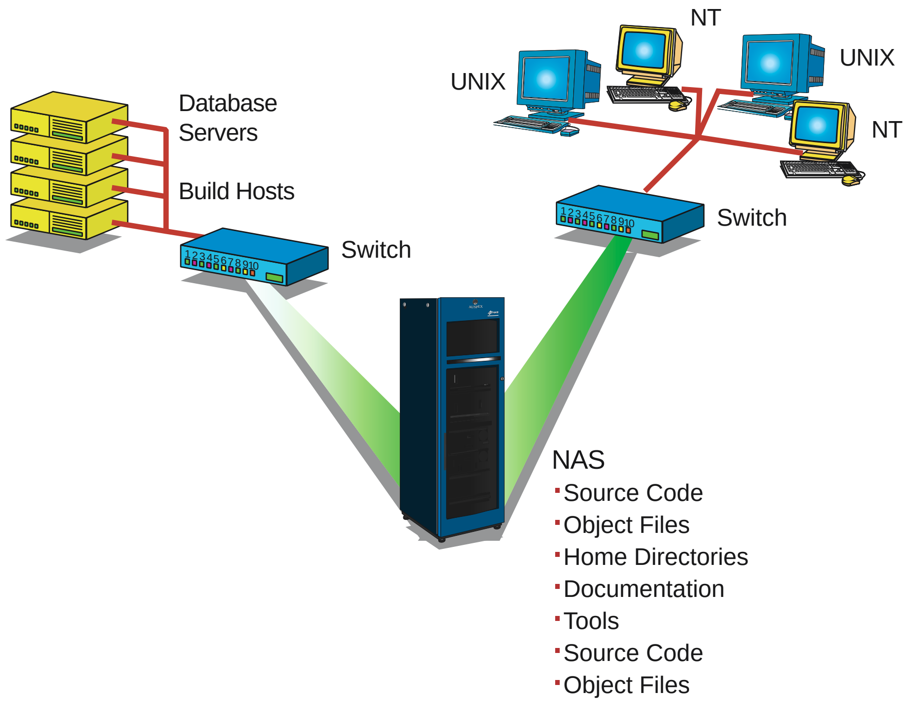
Why NAS is preferred to DAS or SAN for CASE architecture

Unique advantages of Auspex NAS solutions

Conclusion and testimonial quotes

Glossary of Terms

Software Development Network & Storage Architecture Guide



AUSPEX

Table of Contents

Table of Figures

Tables

Introduction to the Importance of Computing Infrastructure Design in Computer Aided Software Engineering (CASE)

1

History of Software Development

The sharing of data among a group of developers and networks did not exist.

The birth of software development is in the not too distant past. In the 1950's, programmers, mostly on mainframes, worked independently on an isolated machine to achieve the output of their work. Most programs and software were limited in their scope of customers and functional or hardware variants. Customers were typically limited to internal users across the company and the software was designed to perform a few, specific tasks, executed on one hardware platform. A simple development process was quite adequate. The development cycle was flat and linear – a single developer worked on a single body of code (or a small group of files) which no other developer touched, and usually the modifications would not directly affect the execution of other pieces of software. The software environment was highly cohesive, as all parts of the software product were available on one platform with the source code on a single disk. The sharing of data among a group of developers and networks - did not exist. This scenario has long since vanished.

Throughout the 70's and 80's, defining a software process became a necessity as software systems' complexity increased. The classic software process – the "Waterfall Model" – required the completion of each phase before proceeding to the next phase. The Waterfall Model worked well during the 70's and 80's, as requirements, once gathered and identified, did not change. The dawn of the Internet age and telecommunication services in the early nineties, however, created new complexities for software development shops utilizing the classic Waterfall model, because during the development stage, the customer changed the requirements frequently. The shifting sands of requirements and design documents are the products of real time feedback from end users and other marketing data. Changes in customer habits, available technology or legislation from state and federal governments in the second quarter of 2001 may obsolete a requirements document drafted in the first quarter of that same year. The paradigm, of completing a 'step' with a high degree of confidence that the decisions of that step would not change, shifts to that of a continuous and parallel process of refinement of requirements gathering, software design and release management.

Software Development occurs in virtually all industries

Virtually every industry is involved with some level of software development.

In addition to the growing demands for software, the number of industries utilizing proprietary software solutions has increased as well. Virtually every industry is involved with some level of software development. This includes companies that develop custom software for internal use, shrink-wrapped software, software for IT infrastructure, and software tools. The key goal of all companies engaged in software development is a ***process*** that enables the fast creation, modification and maintenance of the highest quality software products. Critical components of the process include software development methodology and tools, human resources, and a network and storage architecture. These are collectively known as Computer Aided Software Engineering (CASE). In the words of Dr. Carma

McClure, Vice President of Research at Extended Intelligence, Inc., "Although tools are an important part of CASE, the CASE technology consists of more than software tools. CASE is redefinition of the entire software environment." Weaknesses in any of these areas directly impact productivity, quality, and a company's ability to bring new software products to market profitably. This report focuses on the importance of the network and storage architecture and how it can be optimized to meet the requirements and goals for the overall software development process. A well designed network and storage architecture maximizes the productive capabilities of the human resources and their tools.

The importance of designing an optimal network and storage architecture in software development environments

Developers and testing staff must access data quickly.

The pressure to deliver a high quality software product with an impressive catalog of enhancements and bug fixes is standard in today's software development company. The summation of software process design, backups, metrics and managing multiple development environments are challenging companies as never before. Within this environment, developers and testing staff must access data quickly in a manner that does not impede the progress of the release cycle, or the progress of another development or testing group.

Design information, software components and all versions of developed code must be managed, available, and quickly accessible by developers and build machines; successful software projects depend on it. Advances in development tools and methodology provide development organizations with the ability to speed the development life cycle while maximizing quality as a result of more effective process management, better developer collaboration and round-the-clock development across geographically dispersed locations.

Advances in the way software is developed increasingly creates new demands on the network and storage architecture.

Such advances in the way software is developed increasingly creates new demands on the network and storage architecture that supports the development organization and its processes. It is not uncommon to find software development environments with computers and storage devices that have simply grown into massive, complex networks that are costly to manage. Bottlenecks and costly system administration inefficiencies now characterize hardware systems that once supported the needs of its users.

These problems are exacerbated as the system is expanded to accommodate more users and increased data capacities. Analysis of these issues typically point to network data flow inefficiencies caused by a storage architecture that is based on the distributed server model (as shown in Figure 1). Consequently, many companies are choosing to redesign their infrastructure, starting with the storage architecture. The resulting design is a multi-level, functionally specialized network and storage architecture, which is based on a Network Attached Storage (NAS) system. This new system enables quick access and sharing of any file from any desktop, and provides a new network and computing infrastructure that is simpler, thereby making it easier to manage and expand over time. Such an approach helps to optimize data flow for every stage of the development process while minimizing the cost and complexity of the overall network, storage and computing resources.

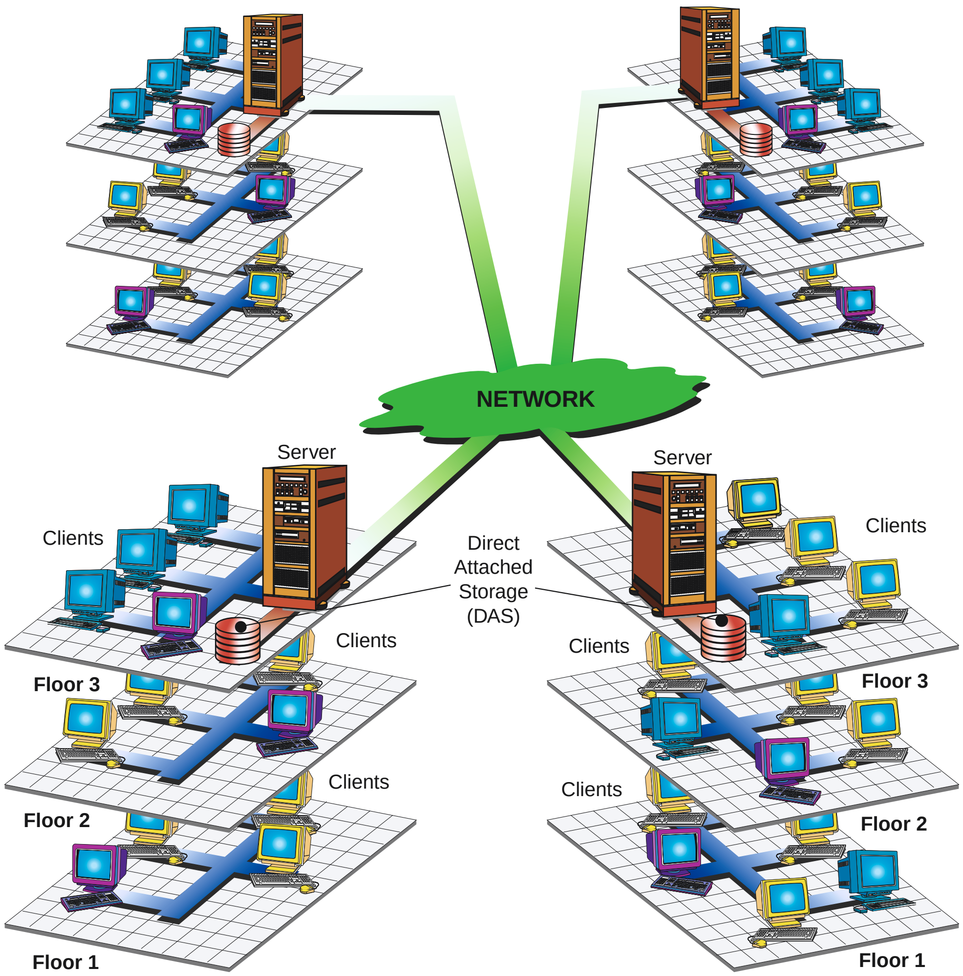


Figure 1 – A cluster of UNIX and NT servers with scattered critical data.

Drawbacks of the distributed server topology include:

- Within the workgroup: Applications run on a general-purpose work group server that doubles as a file server. Pressing the work group server into double duty limits the number of clients per server and impacts I/O performance and therefore the productivity of the group.
- System administration: The inability to manage data from a central point for all work groups creates excessive administration overhead. Multiple data servers create greater risk of reduced productivity due to having multiple points of potential failure.
- Network: Inefficient access to data (i.e., cross-mounting of file systems between servers) between work groups directly impacts productivity.
- Backup/Archive/Data Retrieval: These processes negatively impact file service performance as the work group server becomes taxed as a result of performing file service, backup, and compute service. Additionally, there is no single point of disk backup. Multiple tape drives or backup across the network is necessary which increases administration complexity and degrades network performance.
- Downtime
- Slowed developer productivity
- Difficulty in supporting developer collaboration
- Costly upgrades

It is important to consider data traffic.

For companies initiating an infrastructure analysis and subsequent redesign, it is important to consider data traffic at all stages of the software development life cycle. This includes data volumes; the need to share data during certain process steps; whether remote data access is required; the expected growth of projects, data, and developers, and the ability to tolerate downtime. From that understanding, important choices must be made regarding:

- Network topology (network media, protocols, switching scheme, interconnects)
- Computing platforms (for clients, builds, database processing)
- Storage architecture model: Direct Attached Storage (DAS), Network Attached Storage (NAS), Storage Area Network (SAN).

A thorough discussion of NAS, DAS and SAN storage architectures and a specific decision criteria for an enterprise to use in making appropriate deployment choices for a particular application can be found in a companion Auspex report titled ***A Storage Architecture Guide.***

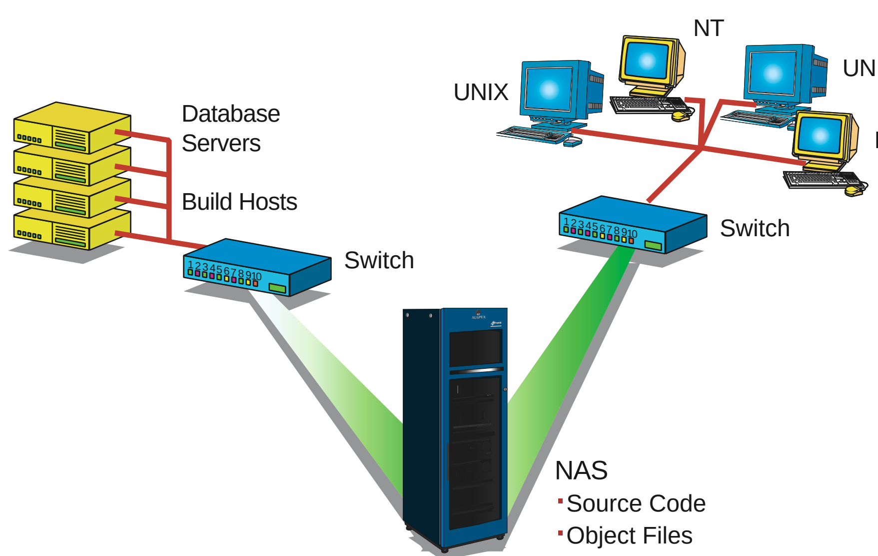
A multi-level, functionally specialized network and storage architecture for optimal work flow

A NAS based architecture has unique advantages.

After analyzing their current environment and requirements, increasingly, companies design and implement development environments based on a multi-level functionally specialized network and storage architecture. The heart of this model comprises a NAS data server, which is designed to enable a simplified, yet efficient network topology. This NAS-based model is chosen for its critical advantages, which include secure file sharing, consistent availability, performance, scalability, and manageability as compared to DAS and SAN-based models. A multi level topology based on NAS is shown in Figure 2.

A NAS-based architecture provides key advantages over other models:

Figure 2 – A multi-level functionally specialized hierarchical storage architecture provides an optimal balance of availability, collaboration, scalability, performance and centralized manageability for maximum developer productivity at the lowest cost.



- A NAS storage architecture enables the consolidation and centralization of data thereby eliminating the problems associated with a distributed data model as illustrated in Figure 1.
- Data is easily shared across multiple computers and heterogeneous platforms.
- System expansion does not add extensive complexity to the network and its maintenance since storage capacity and number of clients are highly scalable.
- A NAS-based infrastructure enables companies to choose best of breed components designed specifically for their function (data serving, data base processing, application execution).
- For remote locations with distances in excess of LAN technologies, NAS solutions provide the only option for the centralized management of the data structures and applications of large-scale software development projects

There are seven key benefits to developers, managers and administrators of a NAS implementation in large-scale software development.

1. Decreased build times
2. Support of parallel development
3. Decreased network overhead
4. Servers are easily tunable and processes that are bottlenecks are easily identified
5. Ease of administration of centralized data and backup processes
6. Higher data availability
7. Cost Savings:
    - Reduced cost of server and LANs
    - Reduced backup maintenance overhead
    - Third party applications are not needed to support data sharing among UNIX and NT clients
    - Reduced total storage (data reuse)
    - Reduced system administration costs

A large telecommunications company restructures their storage architecture

By 1995, driven by an increasing demand for the software associated with their networking and mobile products, a leading telecommunications company had grown a network and computing infrastructure that comprised 700 desktop computers supported by 45 general purpose servers. Five system administrators were tasked with ensuring developer access to 500GB of data (home directories, documentation, tools, and source code) which was distributed amongst the servers. As data capacity requirements grew, additional disk drives and servers were added to meet the demand. However, the distribution of data across multiple servers became increasingly difficult to manage, impeded the developers' access to data, and created more risk of inaccessible data. All of the administrators were in a constant reactive mode in an effort to provide maximum data availability for the user community. Expecting continued growth, it was decided to assess the existing system and redesign the infrastructure in a way that would best resolve their current challenges while meeting their needs for future expansion. Key objectives included:

- ***Speed developers and build engines' access to data.*** With the existing system, speed of datas I/O across the network was impeded as a result of the inefficiencies of accessing data that is stored across multiple servers. I/O performance was further limited due to the general-purpose servers having to perform all other tasks (i.e., database processing, application execution, backup, and system management) in addition to data I/O.

- ***Maximize access to data.*** The developers' ability to access files that were stored across multiple servers required the cross mounting of file systems. However, this creates a significant risk where any one failed server can cause other network servers to hang, thereby interrupting the work of the developer community.
- ***Enable shared access to data.*** A single file should be able to be seen by every desktop without the need for copies or cross mounting of file systems.
- ***Simplify administration.*** In addition to the efforts required to keep 45 servers up and running, each server had its own tape drive for backups, which made the backup process extremely time consuming and fraught with risk. Addition capacity was disruptive and continued to add unwanted complexity to the infrastructure.
- ***Reduce infrastructure costs.*** Adding storage capacity and network connectivity by adding additional servers to the network was costly since the purchase of general-purpose servers included the cost of the CPU. In effect, multiple CPUs were being purchased when all that was needed was additional storage capacity and physical network connections. Support costs were also on the rise due to the need for substantial system administration expertise as the system grew.

This company's analysis led them to a solution that effectively addressed all of their objectives by consolidating all 45 servers into a single NAS-based server. Data I/O performance was improved as a result of the NAS server being dedicated solely to serving data across the network. All developers were able to see any file from any desktop without the need to cross-mount file systems. System administration was dramatically simplified, thereby enabling the administrators to focus on investigating new tools and system optimization techniques instead of reacting to the latest system failure. Overall infrastructure costs per developer were reduced as the system was expanded to support a 300% increase in data volume during the course of a four year period.

Over this four year period, build servers were co-located with the Auspex NetServer and the system operated with very high availability 24 hours per day. In 1999 over 9000 builds occurred flawlessly with each build taking 1-2 hours each. Today over 2000 programmers use this system.

Auspex pioneered functional specialization in computer storage design

The multi-level, functionally specialized storage architecture is a natural extension of the Auspex concept of functional specialization.

The *multi level, functionally specialized storage architecture* is a natural extension of the Auspex concept of ***functional specialization*** implicit in its patented Functional Multiprocessing (FMP) architecture. As an Auspex NS2000 separates and assigns processing resources to specific tasks within the server, a *multi-level hierarchical network and storage architecture model* separates and assigns processing resources across the entire network. This model provides many advantages over the traditional two level flat hierarchical two-level infrastructures for all critical requirements:

- ***Availability*** - The NAS server improves reliability/data availability, flexibility, speed and manageability by insulating file service requests from the general purpose operating system that allows files to be served even in the event of an O/S failure.
- ***Scalability*** – As the site grows, storage and network capacity are easily expanded without sacrificing performance. Additionally, storage is dynamically allocated where needed within the NAS system.

- ***High Performance file sharing*** – Optimized for large file read intensive workloads, the Auspex NS2000 quickly delivers segments of the code tree to developers when requested in a securely locked UNIX and NT environment.
- ***Manageability*** - The unique FMP design of the NS2000 improves manageability by offering both conventional Unix / NT system level tools and the powerful Auspex Control Point™ software.
- ***Cost effectiveness*** – Consolidating server storage on NAS devices reduce capital acquisition and operating costs. Increased uptime results in fewer lost development hours.

The importance of understanding software development technology and terminology.

Those readers who already understand the basic terms and concepts of how the software development process works can skip Chapter 2. However, a large scale software development environment is unique in that it includes a broad range of technologies, ranging from browsers, PCs, client-server models, Local Area Networks (LANs), Wide Area Networks (WANs), many types of servers, and computer storage architecture. Therefore, it was felt that it was important to include an overview on the software development process and its relationship to the many technologies involved. Further definitions of all terms used in this report can be found in the Glossary.

A large scale software development environment is unique in that it includes a broad range of technologies.

Other information sources

This paper augments the concepts discussed in three other Auspex reports titled**:**

- ***A Storage Architecture Guide***
- ***An Internet Storage Architecture Guide***

Both reports are available on the Auspex home page at www.auspex.com and discuss alternative NAS products that are available today including the Auspex NS2000.

Additionally, a thorough review of the Auspex Net Server 2000 Series is available in the Auspex report *titled* ***Auspex NS2000 Product Guide*** which is also referenced from the Auspex home page at www.auspex.com. The Product Guide also contains a complete Glossary of Terms and many Figures and Tables that attempt to clearly explain the technical topics involved in the advanced parallel design of the Auspex NS2000 and how this efficient architecture provides industry unique advantages of "parallel" backup and "read/write" anywhere data sharing within a NAS system.

Another valuable reference article is titled *Network-Attached Storage, A Compelling Story for Storage Consolidation.* It was authored by International Data Corporation and can be downloaded from the Solutions section at www.auspex.com.

Situation Analysis - Software Development Process, Objectives and Challenges

2

Customer demands drive process complexity

With the explosion of the Internet and Wireless Technologies, implementing a process for software development is becoming increasingly complex due to the demands consumers place on the products they use. The customer variables that must be addressed by a single software product have increased far beyond the scope of the 1950's. A software product may have multiple customers, which requires that development efforts occur on multiple platforms. Additionally, each customer may have variants of functional design and bug fixes unique to that single environment. And with more customers, more bugs are discovered, thereby increasing a development company's workload. Adding to the workload, many customers began requiring a customized User Acceptance Testing (UAT) period. Such demands prompted development companies to break up software into releases (see Figure 3), which enabled a more effective development cycle given the new demands on the end product. However, along with the move to improve the development process came unique development management issues:

A software product may have multiple customers, which requires that development efforts occur on multiple platforms.

- Defining the scope of changes
- Defining a project plan for each release.
- Assigning developers to each enhancement or bug fix which occur in parallel with other modifications
- Gathering Metrics for Lines of Code (LOC) counts, Change Sets, Software Modifications per Functional area, etc.
- Managing Source Code to insure re-createabiliy of current releases and traceability of modifications
- Handling merges
- Creating different integration environments based on Quality Assurance (QA) and UAT feedback
- Designing and implementing a network and computing infrastructure to support a process and developer community that continue to demand new levels of performance regarding data access, sharing and processing

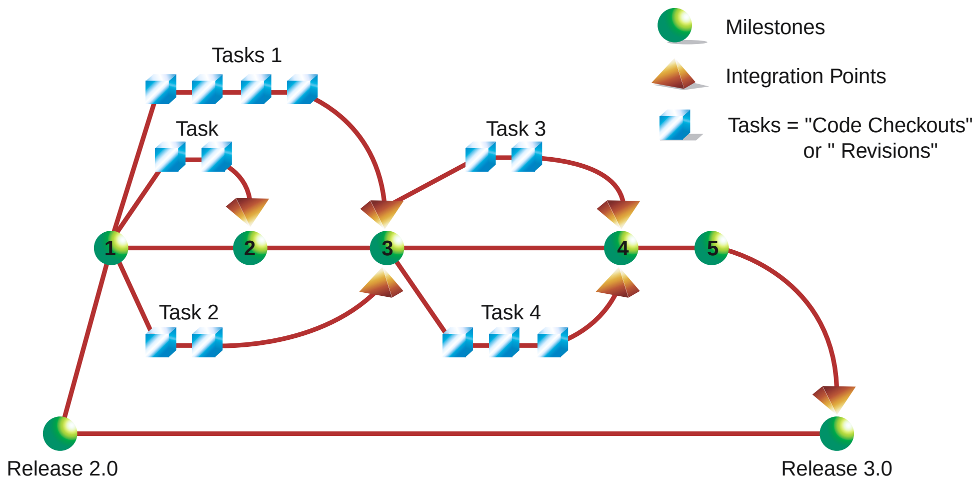


Figure 3 – The software release cycle can be complex with many code revisions, integration points and milestones.

Parallel development

Enhancements and long fixes often affect the same files.

The management of merging code modifications into a single, cohesive integration environment for the modern Software Company is a very specialized area critical to a company's achieving its business goals. In fact IDC's 1998 Global IT Survey identified one of the top business priorities of CEO - increasing productivity through improvements in the infrastructure. If there are numerous enhancements and bug fixes to be addressed in a single code release; however, many of these modifications affect the same physical files of code and demand the very best in computing infrastructure for optimum productivity.

Lost developer productivity while waiting for a co-developer next door to complete his efforts on a single file is not an effective utilization of human resources. The development environment's implementation must allow maximum productivity for all developers. This suggests a need to break the code into many smaller files to increase parallel development, then the need for a powerful NAS file system is greater.

A linear model is not efficient.

A linear model of development access is blocked for other developers to the code base (usually a few files) until a single bug fix or enhancement is complete. This is not efficient. As this inefficiency was recognized with development efforts occurring simultaneously, many effective configuration management tools were utilized for this purpose. Each tool, having strengths and weaknesses, shares one thing in common: storage of the physical source code, and all the revisions made during a particular release cycle. The more developers hitting the source code (read/writes for modification or unit test builds) during peak development hours increases the stress on the network and storage devices. A non-optimized network greatly impedes progress and slows access to the source code archives. This means less progress in a single release cycle.

Multiple development environments

Since many software products contain variants of functionality and target hardware due to market demands and customer requests, duplicate integration, test and quality assurance environments are necessary. The differing environments must remain as small as possible to minimize duplication of storage and retrieval network traffic. Certain modules are uniform throughout all variants of a software product and a duplicate builds for these modules must be avoided.

Keeping development efforts isolated

It is important to avoid storage duplication and complex merges.

When development efforts are isolated until an integration environment is defined, many development organizations discover that the time to market window will not allow for enhancements originally included in a release. Backing out changes is time consuming, as additional testing, and a separate integration environment is required. Configuration management tools that support this flexibility place high demands on network and storage.

Minimization of complex merges

A software process designed to eliminate complex merges will rely heavily on branching and parallel development strategies. Paired with knowledge gathered from the metrics process, management of a code base during a release cycle in a manner that minimizes complex merges requires multiple development environments and, therefore, increased disk and network utilization.

The importance of software development quality assurance and productivity metrics

Storage and retrieval of data must be transparent to developers.

In addition to the challenges of parallel development and release management, companies selling large software products to a customer base, with a diverse need of functionality and target platforms, are embracing quality assurance and metrics gathering as a standard part of their software process. Data such as lines-of-code counts and change sets, (pairing multiple check-ins of separate bodies of code as a single logical "unit"), assist software companies in tracking the history of its release cycles and in identifying bottlenecks in the physical architecture of its software products. The importance of identifying these bottlenecks and their underlying reasons is a tedious task that requires historical data that is normally stored in a database (e.g., Oracle, Sybase, etc.). Acquiring this data during an ongoing release cycle places demands on network and storage devices. Again, fast and accurate storage and retrieval of data must be achieved in a manner transparent to the development community.

Gathering metric data transparently

The metrics necessary for intelligent decisions with regard to partitioning a software product's architecture must be gathered during each release cycle – *transparently*. Certain questions need to be answered for effective metrics gathering: Why were changes introduced? What changes were introduced in a particular release? Which tasks put the most demand on network and storage resources? Can these tasks be broken up in a logical manner to speed the development efforts? How complex were the merge issues? What areas of a software product are the most troublesome with regard to parallel development? The answers to these questions lead to effective tracing of changes within each release and the identification of bottlenecks in the software architecture. The challenge is to gather metrics transparently such that development productivity is not degraded.

Adding "process" transparently

The challenge is to minimize disruption to the development staff.

With the goal of improving productivity, software organizations continue to modify critical aspects of the overall development process. Examples of such modifications include the implementation of a configuration management system, defect tracking or metrics gathering, hardware re-configurations, and the use of new developer tools. Certain modifications may necessitate the relocation of source code and require a training period, which impacts productivity in the short term. Therefore, the challenge is to minimize disruption to the development staff by making the process change as transparent as possible.

Centralized data management is critical for larger software projects

As software products grow, the complexity of its architecture increases exponentially. Management of the actual code base and the storage of differing functionality may (and usually does) become inefficient as new functionality is added. How is a company able to determine when a re-organization of its product at the architectural level is economically justified? Hard and fast documentation must be provided to upper level management to show the need for internal development efforts that are not perceived to add functionality.

A development team may be drowning in bugs all of which are related to a particular functional area, but if this information is not known and articulated in a manner that a non-developer (i.e., the Chief Financial Officer) can understand, the funds for an architectural re-organization often falls on deaf ears. When appropriate data is shown in a manner that details a proposed re-organization's effect of bug minimization in a particular functional area, or the reduced time of testing (and hence reduced costs and a faster time to market), funds expended on internal development efforts are understood to add value to the company's development environment.

Fast build times

A fast turn around time for a software development company's compilation, integration and testing efforts cuts the time-to-market window. Additionally, more features can be included in each release with fast "compile-integrate-test" turn around times.

Fast backups are key to productivity

Any downtime per day for a company's development staff is *expensive*. Salary and contract rates are increasing throughout the world as companies compete for high-tech talent. Efficient utilization of human resources is maximized with a backup and storage procedure that guarantees security and re-createability of a company's development environment and software product in a transparent manner with minimal downtime.

Central management of source code is critical

How are all these issues to be managed? The challenge of implementing a software process is significant; however, even the best software process solution may not overcome adverse issues of a development environment's network design and storage-and-retrieval implementation. Does the actual storage of a development environment's data (source code) affect the time-to-market window? Does it affect the number of enhancements (or bugs) completed in a single release? Are software builds adversely affected? Yes to all of these questions. With the wide variety of platforms and operating systems, and differing functionality requested from users, the process of managing software enhancements and the hardware required to implement this processes is intertwined.

The pressure to deliver a high quality product with an impressive catalog of enhancements and bug fixes is standard in today's software development company. The summation of software process design, backups, metrics and managing multiple development environments are straining networks as never before. Within this environment, developers and testing staff must access data quickly and in a manner that does not impede the progress of the release cycle or the progress of another development or testing group.

An effective network and storage implementation reduces wait time for development and testing staff.

The software process today, therefore, transcends the sequential list of tasks of the "Waterfall Model." With multiple activities occurring simultaneously in a development environment, the network and storage design enhances a company's ability to deliver products to clients in a timely manner. An effective network and storage implementation reduces "wait time" for development and testing staff, and ultimately the end users of the software produced. During a standard day in a software company, multiple builds occur for each active development environment. Testers are accessing releases to test. Metrics data is gathered at the time of "check-ins" to the source code archive. A finely tuned network and storage infrastructure takes all these questions into account. The network and storage implementations must work as one.

The synergy of both Software Configuration Management Systems (SCM) and NAS

As is illustrated in Chapter 3 example of a hypothetical software company with growing pains, it will be seen that there are synergistic benefits from implementing both

Software Configuration Management System (SCM) such as available from Rational Software and other vendors and a NAS networked storage architecture. Both strategies provide different benefits to address the problems of growth in Software Development.

SCM and NAS are synergistic.

There is usually one of a number of reasons that are responsible for the addition of programming staff to a development project:

1. Revenue opportunities from more features
2. Revenue opportunities from faster bug fixes
3. Revenue opportunities from accelerating product release

These business objectives often result in the addition of more programmers to a project. In turn, as more developers are added to the project there are increased strains on the development process in three areas:

1. Increased unit test builds
2. Increased need for development work space i.e., storage capacity
3. Increased need to fix bugs

Both Software Configuration Management Systems (SCM) and the NAS model of storage can directly address these problems in a synergistic way. With SCM, version control saves on disk capacity since it relieves the need to copy all modules (even those that are not updated) for a new software version. Version control uses pointers to unchanged modules so that they do not need to be copied which would waste disk space. In conjunction with version control, NAS brings the benefits of centralized storage management and improved system availability. In addition, NAS provides for the timely delivery of software modules during development with full read/write locking protection to a common data image. These benefits are discussed further in the balance of this report.

Case Study - BillSoft Inc. Develops a Wireless Billing System

3

In this chapter we illustrate the importance of designing and implementing an optimal network and storage architecture for software development by considering the practical (although hypothetical) case of a software development organization - BillSoft Inc.

BillSoft competes in the category of Wireless Billing, developing billing software which runs on five different hardware platforms. BillSoft's product provides three basic areas of functionality for the customer:

Back end data collection. Software is installed at the switch, which monitors and gathers data of actual wireless phone usage.

Bill consolidation. Data from the back end is downloaded daily. At a pre-determined time of the month, the bills are created and a batch job runs, mailing out invoices.

Front-end customer support. The front end is the piece of BillSoft's product that contains a GUI and functionality to interpret and alter data downloaded from the bill consolidation portion of BillSoft's product. Customer support representative's work with the front-end software assisting customers with billing discrepancies

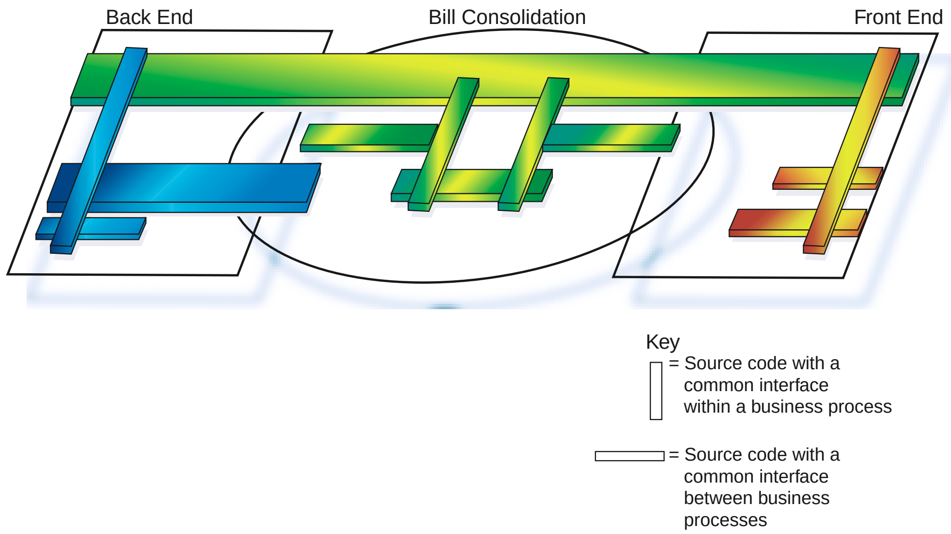
BillSoft's development environment

BillSoft's development staff are partitioned into the three basic functional areas.

Variants of functionality and target hardware are the norm in today's software development company. With many different customers running BillSoft's billing software on multiple developer platforms with three (3) different functional variants, the development environment is complex. BillSoft's product commands the need for seventy-five (75) software professionals not counting Quality Assurance and Testing staff. As part of the development strategy, these members of BillSoft's development staff are partitioned into the three basic functional areas of the product.

The architecture of the back end contains seventeen (17) modules. The bill consolidation contains nine (9), yet six of these modules are highly coupled with the front and back ends. The front end is much more complex due to the GUI features required and commands the bulk of BillSoft's development efforts with twenty-nine (29) functional areas that interface with multiple modules of the consolidation and back end areas. This interrelationship is shown in Figure 4.

Figure 4 – Software Modules are logically dependent on each other both within and between major business processes.



The actual product has thirty-two (32) functional areas. Each release cycle averages seven (7) major enhancements for the base product and two (2) major enhancements for the product's variants. Toss in a per-release average of thirty-five (35) bug fixes (out of the many hundreds that were not approved), and managing a single release becomes a full time job for multiple development managers.

BillSoft's project management demands a network and hardware infrastructure that is highly available, flexible and reliable.

BillSoft's project management demands a network and hardware infrastructure that is highly available, flexible and reliable. Due to the millions of lines of code in development with hundreds of developers running dozens of builds during the day – physical storage needs increase dramatically as enhancements and bug fixes occur in a single release cycle.

BillSoft's network and storage infrastructure impedes progress

BillSoft duplicated their original network configuration twice.

Eighteen months ago, BillSoft was adequately maintaining an entire product with a network configuration comprised of a single application and file server with local storage (DAS) on client machines. As more customers purchased service provider agreements, development staff was added to meet the need for enhancements and bug fixes. To accommodate the additional staff and growing demand for storage space, development workspace and unit test builds, BillSoft duplicated their original network configuration twice using hubs and switches to link workgroups (see Figure 5). While this approach met the need for increased storage space and more developer workstations on the network, BillSoft found that the developers were increasingly having to deal with interruptions, particularly during peak network traffic times (see Figure 6) when more developers hit the source code with read/writes for modification or unit test builds.

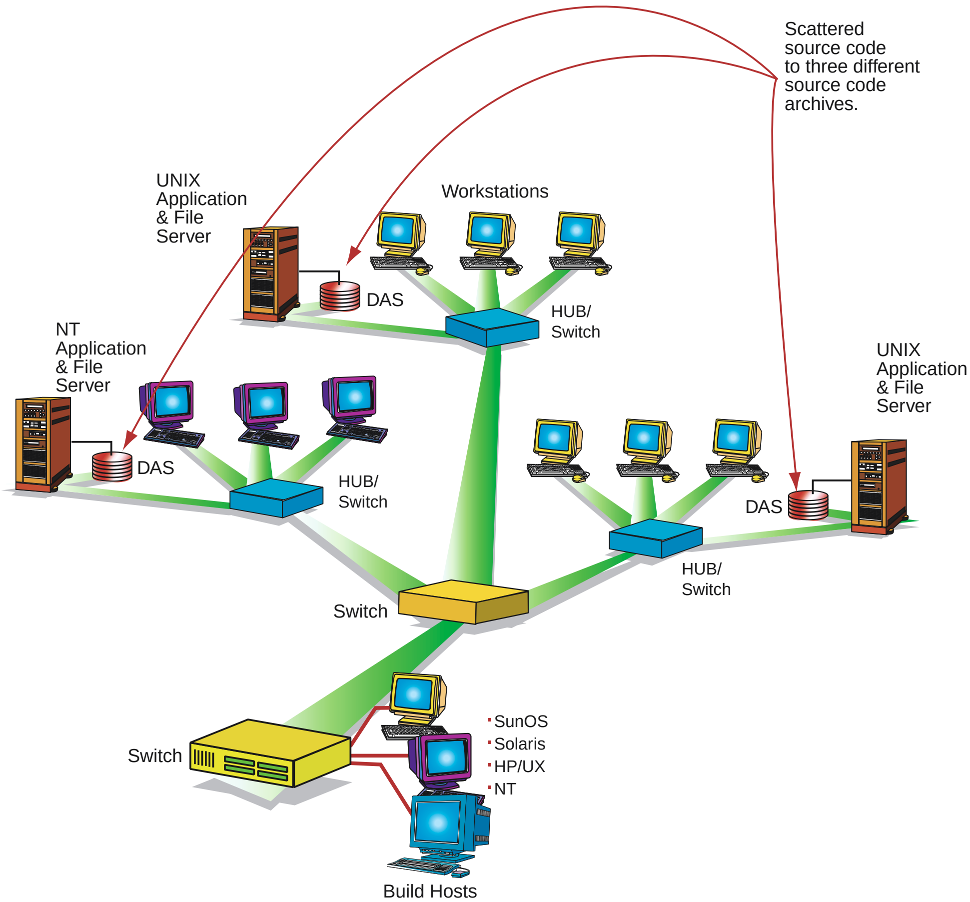


Figure 5 – Before NAS deployment, critical data is scattered in three different source code archives on three different networks.

BillSoft initiates an analysis and redesign of their IT infrastructure

Analysis pointed to a need for process improvements.

With increasing customer and competitive pressures to manage larger, more complex projects, BillSoft analyzed all aspects of their development process in an effort to optimize their development environment. The results of their analysis pointed to a need for process improvements and an IT infrastructure that would eliminate problems while providing the bandwidth necessary to support peak network traffic.

BillSoft's objectives for their infrastructure included:

- ***Data Sharing***

  The infrastructure must improve the development activities of accessing the code repository (check in/out) and multiple builds for different environments, especially during peak network traffic times.

- ***Consolidation***

  Creating multiple environments in a manner that minimized disk space usage. Each build environment is currently a copy of BillSoft's product on a disk (or multiple disks). The new infrastructure must consolidate common areas of BillSoft's product.

- ***Centralized Management***

  With the creation of multiple development environments, parallel development became intensive. As multiple files were scattered over various file systems, maintaining an accurate list of source files and their versions became increasingly difficult. The source code in the new infrastructure was to be centralized and managed by an SCM team.

- ***Backup***

  With multiple bodies of code scattered throughout the network on different disks and multiple development environments, undergoing activities simultaneously, network traffic from backups must be minimized and a process to expand the environment must be implemented in a manner that is harmonious with current system management policies. BillSoft's old environment required too many variations to be effective of backup and restore strategies.

Figure 6 – Network traffic in software development varies greatly depending on the time of day.

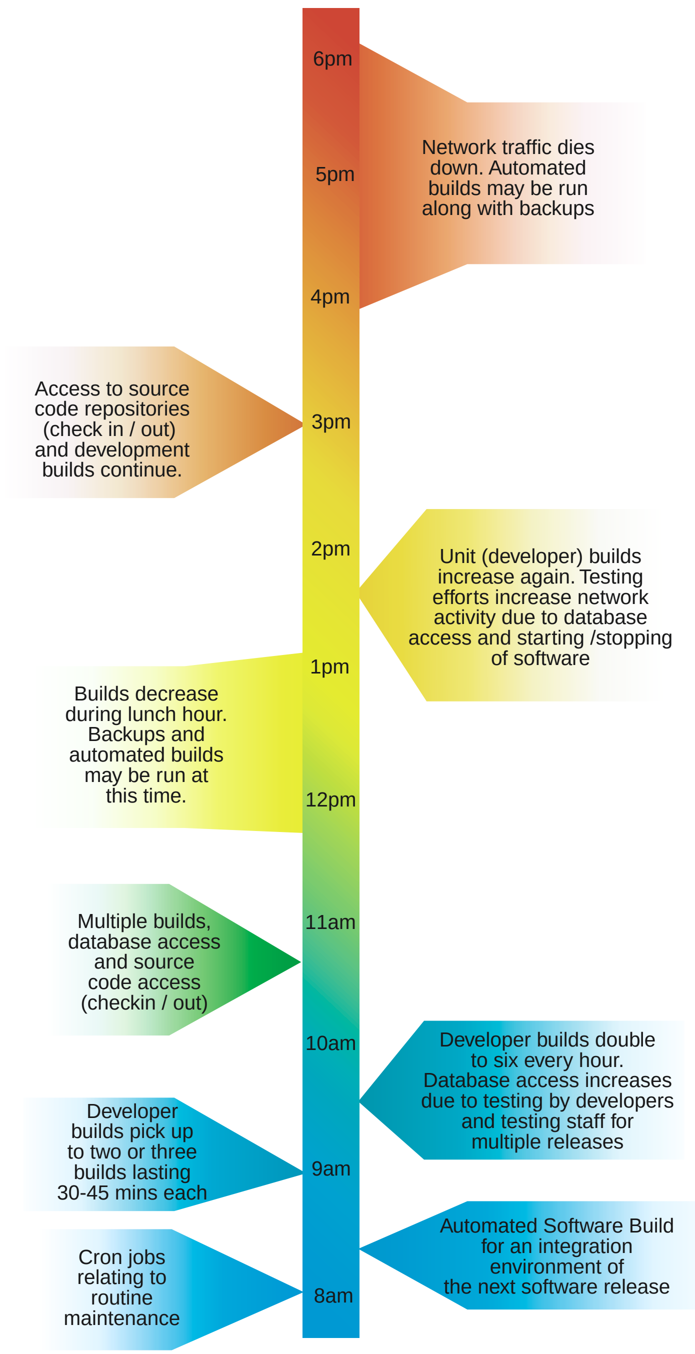


Figure 6 shows that a large variation of network traffic can vary depending on the time of day. As the work day begins, network traffic increases geometrically impeding the overall progress of BillSoft's development and testing staff. Certain builds should be run during the day are postponed to the early morning hours. By implementing an optimized storage and retrieval system, these builds could occur during regular business hours increasing turn-around time for the company as a whole.

BillSoft implements a NAS-based network and storage infrastructure

BillSoft's analysis confirmed that their DAS-based network and storage architecture was a key cause of their data flow bottlenecks. Furthermore, they determined that continued growth of the development environment with such a model would certainly increase infrastructure and administration costs, but not necessarily enable greater productivity. Their next step was to test a NAS-based solution against their redesign objectives and plans for future growth. Ultimately, BillSoft redesigned their network and storage infrastructure around a NAS-based solution. Chapter 4 describes in detail the rationale for such an approach along with specific benefits to the system administrators, network administrators, developers and the company.

Designing a Hierarchical, Functionally Specialized Network and Storage Architecture

4

The NAS model would help BillSoft realize their objectives.

BillSoft's decision to implement a NAS-based network and computing infrastructure was based on their findings that this model would best help them achieve their objectives. To provide the reader with an understanding of the rationale that is behind many software companies' decision to implement such a model, this chapter describes the structure of the NAS model and its benefits in a software development environment.

In designing an IT infrastructure to support a software development organization's workflow, two key goals stand out:

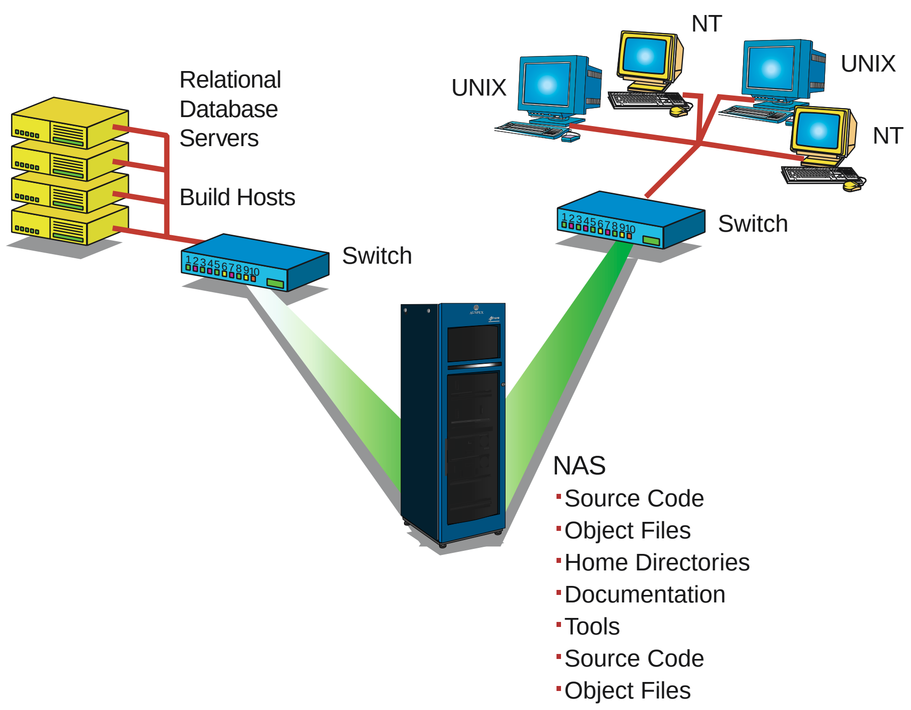
- Increase developer productivity by providing tools, methodology and a computing infrastructure that allows them to apply their skills and knowledge without having to wait for files or another process to complete.
- Minimize system administration costs while meeting the needs of the developer community as projects increase in size and complexity.

Achieving these goals requires careful analysis and design of the network topology AND the storage architecture; both are inextricably linked.

Functional specialization is essentially a "fit for function" approach

The concept of a hierarchical, functionally specialized network and storage architecture (see Figure 7) is based on a "fit for function" approach where each component (i.e., data servers, data base and build machines, network components) is deployed for a specific function, allowing choice of the right components and features for each function. Hierarchical design facilitates changes. In a network design, modularity allows creating elements that can be replicated as the network is expanded, thereby facilitating easy network growth. As each element in the network design requires change, the cost and complexity of making the upgrade is contained to a small subset of the overall network.

Figure 7 – Hierarchical, functionally specialized network and storage architecture.



Benefits to systems administration and the developer community

The modular nature of the hierarchical model provides benefits that include:

- Cost savings (due to data centralization). From a hardware perspective, cost saving factors include the need for fewer data servers, reduced total storage space, fewer network components, and the ability to add specific "modules" to expand system capacity (vs. complete servers or storage subsystems)
- Network simplicity and ease of understanding
- Easy network and capacity expansion. Upgrading the system to utilize technological advances as they become available is cost effective and more controllable. Commodity-priced compute servers and workstations can be replaced quickly and easily without disrupting or forcing change to the overall infrastructure
- Improved fault or bottleneck isolation
- Ability to sustain optimal performance of all components of the total system-data I/O performance is not compromised by compute intensive operations, backup or number of users. For example, segregating the compute function (i.e., software builds) from the data I/O function gives users the freedom to select best of class components
- Gives the choice to developers, managers and administrators the options of upgrading or replacing the infrastructure with the latest compute technology, including the choice of whether they write code on UNIX or NT platforms. This feature also supports the increasingly popular "telecommuting,", as may be the case when a developer has an NT computer at home and a UNIX computer in the office. The choice of platforms creates far more productivity when compared to a computing infrastructure that includes general purpose servers, where compute functions and data I/O functions are combined, resulting in performance degradation of one or both functions

A NAS-based hierarchical, functionally specialized network model is especially suitable for large scale storage consolidation in software development projects

NAS consolidates storage from numerous UNIX and Windows environments.

NAS is ideal for applications like software development where multiple activities occur across the network in parallel. NAS enables the developer community to work during build cycles without degrading performance of data access.

Software development environments often ***consolidate*** storage from numerous and distributed UNIX and NT servers onto a reliable NAS platform such as the Auspex NS2000. This has the advantages of reducing operating costs by centralizing data management functions, and improving performance and availability by consolidating environments where user data could be spread over 100 or more application servers. In these highly decentralized environments server node failures could lock users out of critical data where the critical data is not adequately protected – given its value. Robert Gray of International Data Corporation (IDC) discusses these cost and management advantages in a white paper titled ***Network Attached Storage A Compelling Story for Storage Consolidation***. A hyperlink to this IDC paper can be found by accessing the electronic version of this report at http://www.auspex.com.

In addition, there are many supporting references to the advantages of server and storage consolidation in today's enterprise infrastructure. Some selective quotes from noted industry analysts support this finding.

> *"Server consolidation results in a 34% decrease in time spent to manage a network in large scale environments."* - Strategic Research Corporation
>
> *"Today, the main reason for the trend toward server consolidation is the promise of a simpler infrastructure that is easier to support and has a lower TCO. Our latest research based on our most recent TCO model, shows that enterprises can save a significant amount of money by consolidating file servers…"*
>
> *"…Server consolidation projects are a great opportunity for the central IS organization to show value, provide capacity planning, and control hardware and software standards and upgrades. For many enterprises, server consolidation can result in impressive savings"*
> - Gartner Group

In a 1997 InfoWeek article – 80% of the 250 senior IT managers surveyed had or planned to centralize their environment because it promised:

- Easier management – fewer upgrades, streamlined backups, fewer points of failure
- Lower costs – lower maintenance., fewer system admins, less downtime
- Improved security – better control of systems and data
- Easier disaster recovery – data is more easily protected, backup procedures more secure, centralized data allows backups to be run over direct-attach channels, providing the dedicated bandwidth required for regularly scheduled backups of large amounts of data.
- Increased performance – applications receive data more quickly, network infrastructure can be better optimized
- Simpler storage strategies since adding capacity is less expensive and less burdensome

Availability benefits of a NAS based software development architecture

Mistakenly, a distributed data model's spreading of data across multiple servers is sometimes seen as optimal because a system failure is not seen as catastrophic. However this architecture actually exposes an organization to *multiple points of failure*, effectively increasing risk. In most environments, users require multiple instances of *data* to create *information.*

An excellent example of this is found in large-scale software development environments using the distributed server model. With this architecture, a programmer is prevented from doing a compile unless the source, libraries, headers, and objects are all available. Similarly, a customer service representative may not be able to handle a customer's issue if billing records are available, but payment records aren't. In these cases, loss of any one set of data results in *loss of information*. Distributing data increases the chances of *information unavailability*.

Consolidation to a NAS server lowers the risk of failure.

Consolidation to a NAS server lowers the risk of failure and provides higher levels of information availability, thereby making developers, managers and administrators more productive while reducing the system support burden.

Data sharing benefits of a NAS based software development

NAS products offer standardized, reliable, and integrated file locking.

NAS is the only form of storage that supports both NFS and CIFS network file system protocols in a manner suitable for sharing file storage between UNIX and NT clients. Since NAS products offer standardized, reliable, with integrated file locking for both the NT and UNIX environments, software developers now have the ability to share information and code between UNIX and NT clients, and/or develop UNIX applications from an NT client. It is important to note that about 3 out of 4 NAS users today share data between UNIX and NT, making this type of application a key differentiator between DAS and NAS.

Improved backup performance of NAS based software development

The NS2000 minimizes backup windows.

Clearly, managing one or just a few systems is less complex than managing multiple systems. According to Strategic Research Corporation, companies that have adopted a consolidated approach to managing client/server data save $1,700,000 per year on average. As impressive as those numbers are, they don't tell the whole story. Issues such as **backups,** which are transparent to users unless something goes wrong, are at the core of what system managers do. The Auspex NS2000 excels in this area. Transferring data directly from disk to tape, the Storage Processor can backup an entire NetServer during off-hours. Incremental backups only take an hour or two, although this is highly dependent on the amount of data and the amount of changes since the last backup. For 24x7 environments, an online backup feature ensures that backups get run and do not impact performance.

Compare the NS 2000 NetServer's capabilities to those of smaller servers and you can quickly see the difference. With many small servers to administer, you can backup to multiple tape drives and create an unmanageable situation or you can backup over the network to a central system, which is *very slow and burdens the network*. Faced with these choices, administrators in distributed environments often gamble and perform backups infrequently, significantly exposing users to data loss. If several hundred users lose a week's worth of data, the $1,700,000 figure will seem insignificant compared to the loss associated with development interruption, and the resultant impact on time to market.

Lower Total Cost Ownership (TCO)

The NAS model enables the consolidation of many servers, which in addition to the benefits:

- Reduced costs from consolidation (reduced resources, more economical storage capacity management, reduced requirements for spares).
- Reduced costs from common administration (fewer errors from reduced complexity, faster, less complex backup procedures).

Examples of customer and analyst comments about the lower TCO of NAS are:

"Server consolidation results in a 34% decrease in time spent to manage a network in large scale environments." - Strategic Research Corporation

"Today, the main reason for the trend toward server consolidation is the promise of a simpler infrastructure that is easier to support and has a lower TCO. Our latest research based on our most recent TCO model, shows that enterprises can save a significant amount of money by consolidating server…Server consolidation projects are a great opportunity for the central IS organization to show value, provide capacity planning, and control hardware and software standards and upgrades. For many enterprises, server consolidation can result in impressive savings". - Gartner Group April, 1998

Server consolidation projects are a great opportunity for the central IS organization to show value.

Reduced downtime related costs

According to a Strategic Research Corporation study, centralized data costs $2.00/MB over its useful life; distributed data costs $7.00/MB. Storing data across many small servers means you're paying for **infrastructure**-type items again and again: packaging, power supplies, fans, backplanes, boards, cables, removable media (CD, floppy, or tape). Once you get to medium capacity levels (200-300 GB), the apparent cost advantage of buying smaller systems has disappeared. With consolidation, infrastructure items are leveraged across much larger capacities, resulting in an overall lower cost/megabyte.

Easy scaling with NAS as the software project grows

With the distributed model, requirements for more processing power or more storage result in a need for additional servers. This creates increased system complexity which makes it more difficult to manage and adds unnecessary costs. A meaningful portion of new investment goes for infrastructure-type components, and not for processing and storage capabilities. A large NAS server can be easily upgraded. Capacities can increase significantly without creating additional system management burden. Additional functionally specific processor components can be added as loads increase which allows more cost-effective system expansion over time.

With the NAS model, as storage capacity needs grow, more disks are simply added. In distributed environments, additional storage needs quickly result in additional system purchases. Buying disks will always be less expensive (on average about *1/2 the price* of purchasing a new small system).

With the NAS model, as storage capacity needs grow, more disks are simply added.

Eventually, most every environment requires more advanced technology. There are two options:

1. Out with the old and in with the new, or
2. Update the existing.

Obviously, #2 is preferable because it is less costly and causes much less disruption for the users. The NAS model provides the benefits of much faster upgrades: simplicity, cost-effectiveness, and the extension of an asset's useful life. With the distributed model, even if in-box upgrades are available they are often times more complex than the consolidated model. Often, smaller systems are not capable of being upgraded and must be replaced. Result: large capital outlay, significant administrative effort, and lost productivity for users.

On average, a large NS2000 style enterprise class NAS server with good scalability should be useable for <u>twice as long</u> as a smaller, less scalable server.

Why NAS is Preferred to DAS or SAN for CASE Architectures

5

At the heart of the network: the data storage system

There are three architectures to consider for CASE.

There are three data storage architectures that companies must consider as they work to design an optimal network and storage architectures for software development: Direct Attached Storage (DAS), Network Attached Storage (NAS), and Storage Area Networks (SAN). Significant confusion currently exists over when to choose DAS, NAS or SAN for a particular application, however, DAS, NAS and SAN are complimentary and not mutually exclusive.

Networks are now faster than storage channels

During the past five years the transfer rate for leading edge Direct Attached Storage (DAS) interconnects has increased fivefold from 20 Mbytes/sec for Fast/Wide SCSI-2 to 100Mbytes/sec for Fibre Channel. Over this same period, however, the transfer rate for leading edge networking interconnects has increased ten-fold from 12.5Mbytes/sec for 100baseT Ethernet to 128Mbytes/sec for Gigabit Ethernet. In other words, network data rates have not only caught up, but have surpassed direct attached storage (DAS), and are no longer two times slower — as they were five years ago. This has shifted the bottleneck from the network to the server and its direct attached storage.

Analysts predict a major shift from DAS to NAS

Only recently, as DAS vendors were involved in the never-ending task of supporting all flavors of UNIX, NT, SCSI and FC for their storage products, both Dataquest and IDC began projecting explosive growth for NAS products as a percentage of the total storage market. These projections are based on four key factors:

1) The fact that there are strong standards for NAS resulting in simpler installation and lower cost of management.
2) The fact that increasing network speed has begun to equalize the performance gap that used to exist between NAS and DAS for many applications.
3) The fact that true data sharing between heterogeneous clients is possible with NAS and not with DAS.
4) The move to re-centralize storage to reduce management costs.

Dataquest has predicted that the commanding DAS storage market share (95%) of today will be eclipsed by NAS over the next five years, and IDC projects that specifically designed NAS products will grow fivefold to $5B by 2002, from $1B in 1998.

Current confusion over NAS and SAN

For years, server vendors have implemented a variety of specialized hardware and software schemes to encourage the sale of storage with their processors. General-purpose DAS vendors have followed the same strategy. Not wanting to support Network Attached Storage, where it would be easier for competitors to make inroads due to the clear NFS/CIFS standards, general purpose server vendors and general purpose storage vendors have developed their own *proprietary* visions of networked storage. These visions are alternatively called Storage Networks (SNs) or Storage Area Networks (SANs). The vendors developed these *proprietary* visions in order to bring the benefits of network attached storage to their users without losing control of the storage and networking sale to NAS vendors.

SAN does not meet two of three key objectives.

The SAN initiative, therefore, is a loose configuration of vendors desiring to promulgate the weak standards of the past while talking about bringing the benefits of networking to storage architecture.

Key benefits envisioned for the SAN model include:

- ❑ LAN and server-free backup
- ❑ Storage resource pooling / sharing
- ❑ Easy storage resource management
- ❑ Data sharing
- ❑ Interoperability of heterogeneous servers and storage

However, it is important to note that the SAN model does not provide two of these critical capabilities: data sharing and Interoperability of heterogeneous servers and storage.

Data sharing and Interoperability of heterogeneous servers and storage

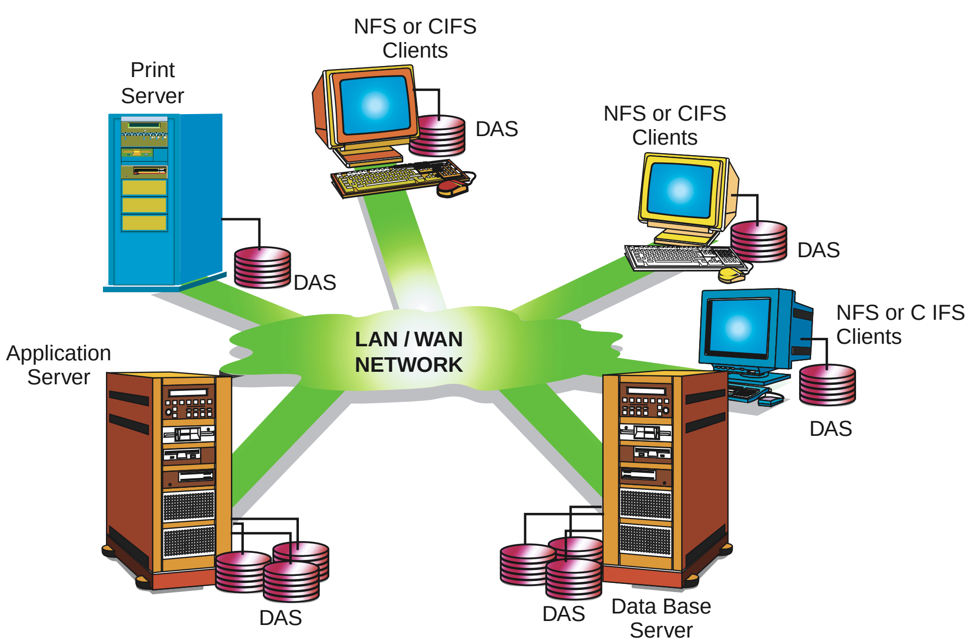
NAS meets all objectives.

Only the NAS model provides all of these capabilities, which makes it the choice for the primary data storage and delivery system. Most industry analysts agree that all three storage architectures serve different and complementary roles in the enterprise storage architecture. In addition, a majority of analysts and NAS vendors currently believe that NAS and SAN will co-exist and deliver their own unique benefits to networked storage.

DAS

Today, greater than 95% of all computer storage devices such as disk drives, disk arrays and RAID systems (Redundant Array of Independent Disks) are directly attached to a client computer through various adapters with "standardized" software protocols such as SCSI, Fibre Channel and others. This type of storage is alternatively called "captive storage," "server attached storage" or "direct attached storage" (DAS) as illustrated in Figure 8.

Figure 8 – DAS storage architecture.



DAS is an appropriate choice today for both the very low end PC and very high-end high-performance mainframe applications. It is also appropriate for certain compute intensive and high performance OLTP database applications. However, in a software development environment, a DAS-based network and storage architecture is not considered an effective model for software development due to certain drawbacks:

Backup - Many distributed sites don't perform backups on a consistent basis because of the time and performance impact.

Security - The more data servers in a case environment, the less the ability to physically secure all data. With server consolidation this problem is solved.

Data Access - Utilizing multiple, general-purpose work group servers (as illustrated in fig. 1) to perform multiple tasks (database processing, application processing, file serving, backup) limits the number of clients per server and impacts I/O performance, thereby slowing productivity. In addition, a DAS-based network topology often leads to file system cross mounting which also contributes to poor data access.

Support - It will take more system administrators to manage multiple systems versus one or very few systems. Experienced support engineers are a scarce and expensive resource.

Upgrades and system expansion - Performing software or hardware upgrades to multiple systems multiplies the time involved and multiplies the risk of doing one of the upgrades incorrectly. Due to this model's lack of modularity, expansion of network or storage capacity can create additional, unwanted complexity and/or force forklift upgrades.

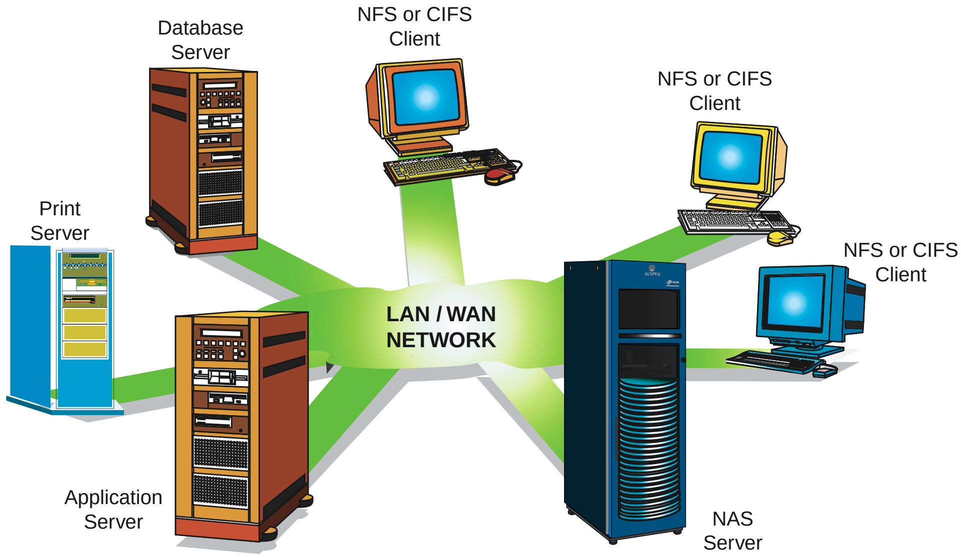
Fault or bottleneck isolation – This is difficult due to multiple tasks (builds, backups, etc.) being handled by general-purpose servers.

NAS

NAS is based on strong standards.

Unlike the SCSI/Fibre Channel/UNIX/NT situation described above, ***network standards on the other hand are strong standards*** and driven by system considerations. A strong network standard therefore can be thought of as the standards committee saying — "You will do it this way." There are two true network standards for accessing remote data that have been broadly implemented by virtually all UNIX and NT system vendors. Developed and put into the public domain by Sun Microsystems, NFS (Network File System) is the standard for UNIX. Developed by IBM and Microsoft, CIFS (Common Internet File System) is the standard for all flavors of the Windows operating system. As a result of these broadly accepted standards for network data access, storage devices that serve data directly over a network (called Network Attached Storage or NAS devices) are far easier to connect and manage than DAS devices. In addition, NAS devices support "true file sharing" between NFS and CIFS computers, which together account for the vast majority of all computers sold. Figure 9 shows a typical NAS topology.

Figure 9 – Network Attached Storage (NAS) storage architecture.



SAN

SANs today are all proprietary.

A SAN is defined as a network storage architecture that includes 2 or more disk/tape subsystems connected to 2 or more servers

Instead of putting the storage directly on the network, the emerging SAN concept puts a network in between the storage subsystems and the server as shown in Figure10. So in reality a SAN adds network latency to the DAS storage model. At this time, the creation of SAN standards is only in the formative stages and realistically years from being finalized. Leading storage vendors however, have announced proprietary SANs that are still largely visions. EMC has announced a proprietary Enterprise Storage Network (ESN), and Compaq has announced a proprietary ENSA (Enterprise Network Storage Architecture). As was the case with UNIX and SCSI, SAN is likely to become a variety of similar architectures that are not based on strong standards. This creates major roadblocks to successful integration and data sharing between heterogeneous platforms.

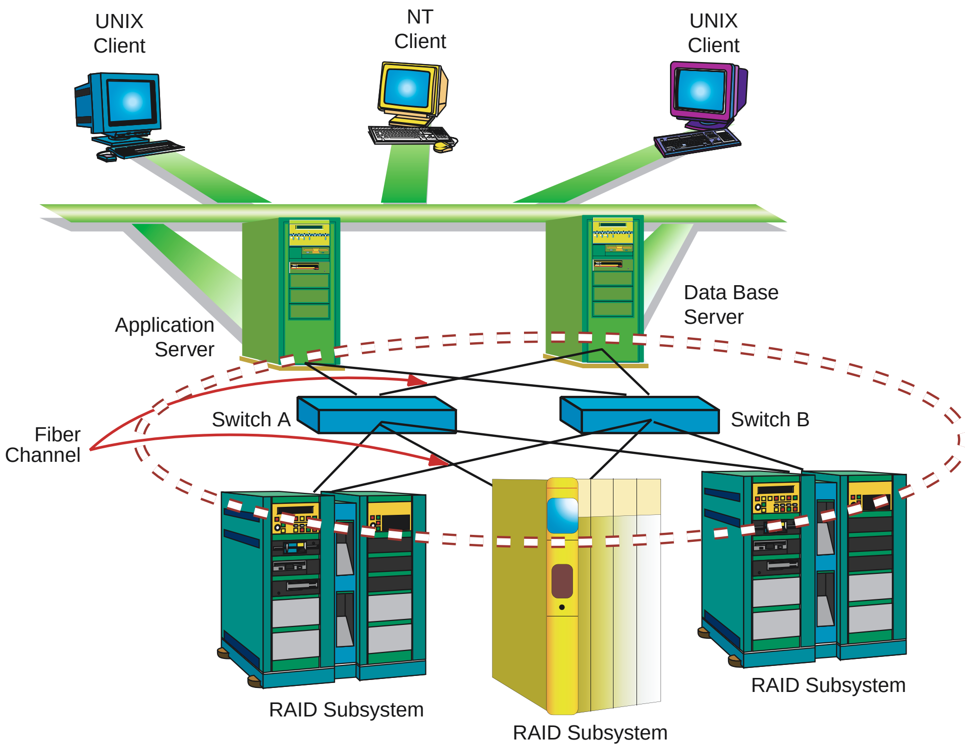


Figure 10 – SAN storage architecture.

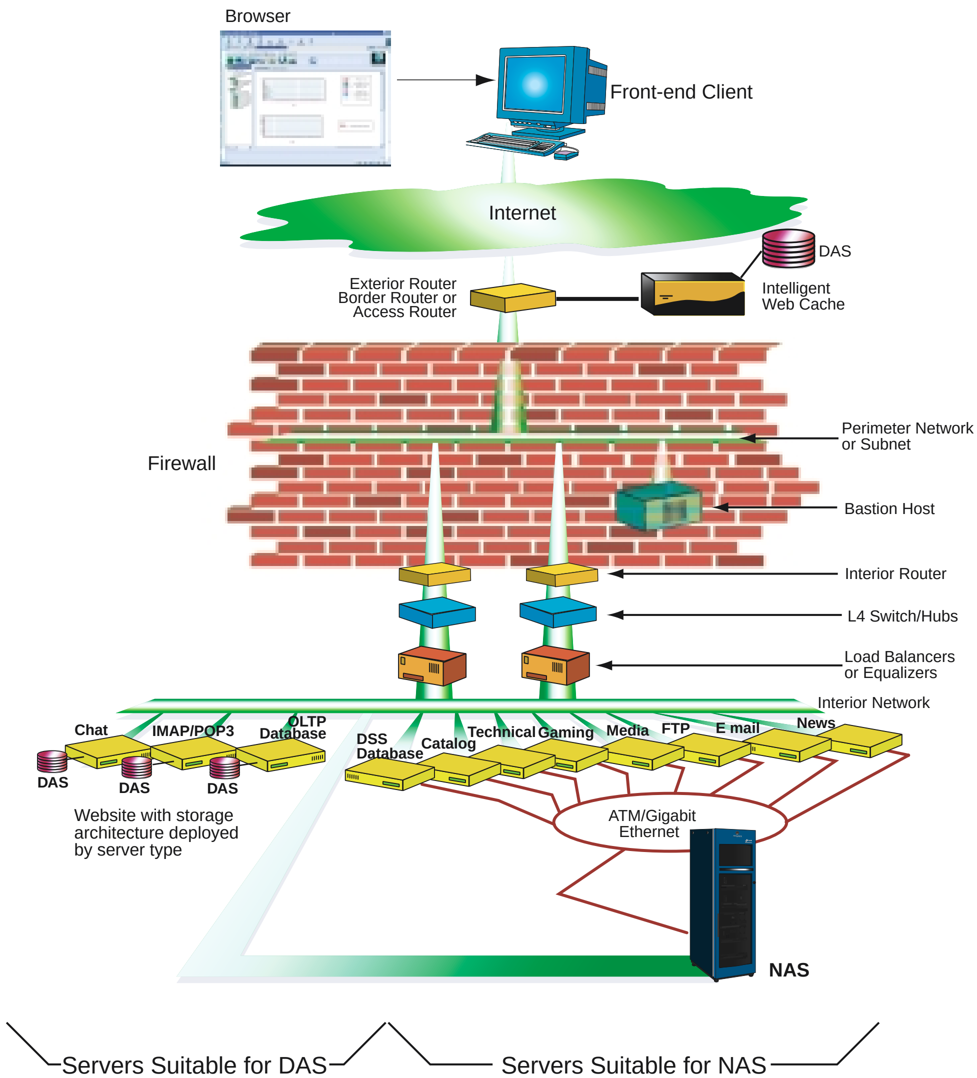
Because of its current lack of standards, SAN is available only in *proprietary* configurations and the long-term interoperability of these schemes is questionable. Although the SAN vision currently involves many benefits currently available on NAS, there are certain synergies with existing enterprise operational and management tools that have led early adopters to experiment with test deployments. Although the long-term SAN vision is for interoperability among heterogeneous servers and storage products, it is advisable to implement early SAN applications in a homogeneous environment with one of the available *proprietary* solutions from EMC, or Compaq. SAN is also appropriate for applications that do not require true data sharing, a feature which is not likely to be available until SAN standards evolve to the level of those available with NAS today. SAN is also appropriate for applications where the well known Fibre Channel (FC) and Fibre Channel Protocol for SCSI (FCP) security risks can be managed and where performance bottlenecks arising from Fibre Channel node and link congestion can be avoided.

Internet and VPN/WAN compatibility

With Turbo Copy, file systems or individual files can be replicated remotely over the Internet.

NAS is the only real choice for centrally managed data over WANs. This is because a long distance networking protocol like TCP/IP is required for accurate data replication over WANs. The NS2000's software option, TurboCopy™ provides such a technique. With TurboCopy, file systems or individual files can be replicated remotely over the Internet or other long distance WANs. SAN is today based on Fibre Channel, which has an effective limitation to campus distances. When configuring applications in a software development environment, it is important to understand the workload characteristics before making the storage architecture decision This topic is covered in more detail in the ***Storage Architecture Guide***, which is available for download from www.auspex.com.

Figure 11 – Internet applications where high performance of network data serving is critical.



Network Attached Storage (NAS) has migrated from LANs to WANs

First pioneered by Auspex, Network Attached Storage (NAS) evolved from the networking industry. The NAS concept migrated from LAN networks within enterprises to web based WAN networks such as Intranets, Extranets and the Internet itself. In these sites, NAS servers such as the Auspex NS2000 coexist with DAS as backend storage for Web servers attached to the Internet.

NAS is the best choice for UNIX and Windows NT data sharing applications.

With networking, there are strong standards for connectivity, data security and load balancing. In the Auspex NS2000 Series, the file system resides in the NS2000 as opposed to Web Servers with DAS where the file system competes for resources with the application server's CPU. Web Servers with DAS, NAS and SAN are appropriate for an enterprise depending on the application being supported. NAS is the best choice for UNIX and Windows NT data sharing applications, such as software development, audio and video streaming, video applications, on-line gaming applications, consolidated file serving applications, technical and scientific applications, Internet and Intranet applications and certain types of Decision Support (DSS) applications. In particular, the Auspex NS2000 is highly effective for FTP serving since the protocol is native to the server.

Unique Advantages of Auspex NAS Solutions

6

The most advanced design available in Network Attached Storage

The Auspex NS2000 architecture is considered the most advanced design available for the specific task of serving files over networks with market leading performance, consistent data availability and robust security.

The NS2000 is the most advanced NAS product design.

1. It is a ***modern parallel architecture*** introduced in 1999 and is the only choice in parallel hardware and software design among network file serving alternatives.
2. ***Very high availability*** is provided by a robust product design with customers experiencing 99.99+% availability for systems currently in production. This results in average Auspex NS2000 NetServer unscheduled downtime of less than thirty minutes of unscheduled downtime per year. Full environmental monitoring and hot swap capability is also provided.
3. ***Network support*** options include 10/100BaseT Ethernet, ATM, Gigabit Ethernet and Ethernet Trunking.
4. It provides ***full function data sharing*** between UNIX and Windows and any other NFS or CIFS client.
5. It offers ***scalable performance*** through multiple CPU architecture, hardware RAID, real time operating system (kernel) design, contiguous file system and superior network and system management tools.
6. ***Backup windows and disaster recovery are optimized*** with the Auspex NDMP **Turbo-Replicator** and efficient parallelism. Data can be replicated in parallel on each of three I/O Nodes. This provides for a total data replication performance of 200 GB+ per hour on a fully configured system. Data can also be replicated to remote locations for disaster protection, using Auspex TurboCopy.
7. The ***full suite of UNIX network and system management tools*** are available in addition to Auspex custom ***Control Point Software***.
8. Auspex offers ***robust factory dial-in capability*** that provides world class functionality for remote diagnosis of problems when they occur.
9. It offers ***easily scaleable capacity*** from the smallest department to the largest enterprise as shown in Figure 12.

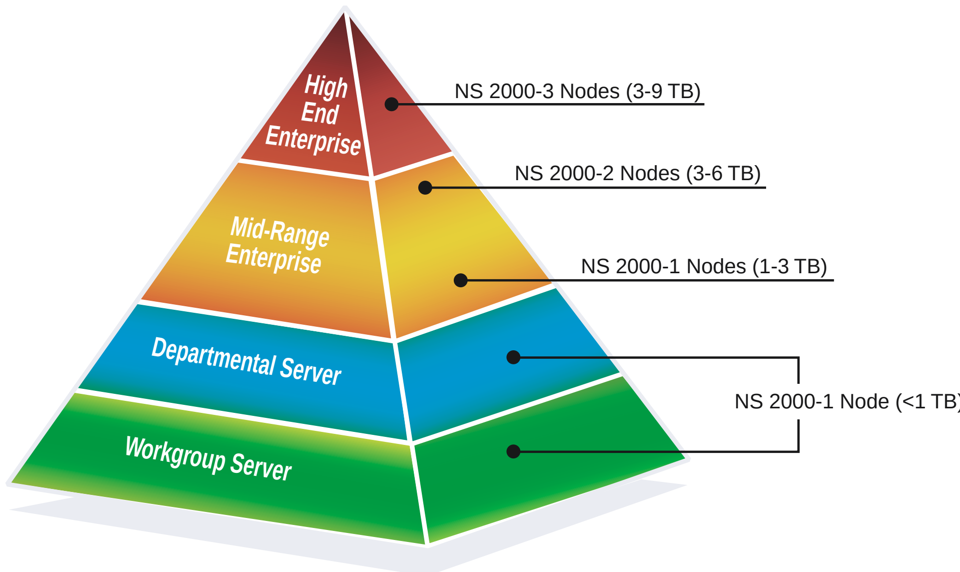


Figure 12 – The Auspex NS2000 product line scales from workgroup to data center.

Auspex is widely considered by customers and analysts alike to have the highest level of expertise in both NAS and networking.

The highest level of expertise available among NAS vendors

Being the originator of NAS, Auspex is widely considered by customers and analysts alike to have the highest level of expertise in both NAS and networking technology. Since the topic of NAS is new to many customers, Auspex provides the best public information available on optimizing the flow of accurate information and support on both a pre and post sales basis. The Auspex sales and system engineering teams have at their disposal experts in each of the important topics discussed in this report if needed. As with any IT architecture decision, probably the most important issue is the selection of a vendor/partner with the best "total" solution. This means not only choosing a vendor who remains at the forefront of technology with the most advanced parallel architecture, but also making sure the vendor can supply the most knowledgeable support personnel to assist in the design and implementation of optimal systems.

FMP is the key to the NS2000 advanced design.

Parallel hardware and software design of the Auspex NetServer 2000

The Auspex architecture is based on a parallel processing design as shown in Figure 13. The Functional Multiprocessing (FMP) architecture used in Auspex NetServer 2000 is unique to Auspex. FMP assigns a specific processor to handle each of the following processes: network processing, storage processing, file system processing and system and network management processing. In comparison, the Network Appliance architecture uses only one processor for all these processes. The FMP building block makes Auspex architecture highly scalable, easier to expand, and better performing than other alternatives.

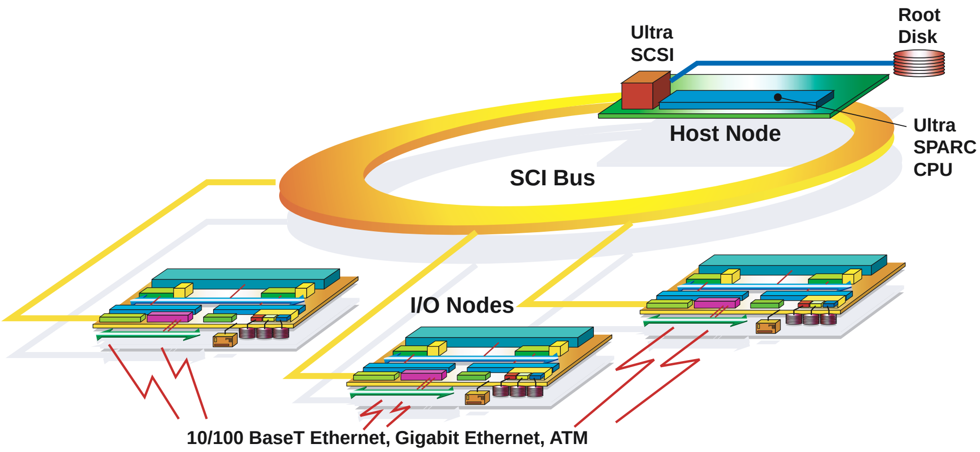


Figure 13 – System block diagram of the Auspex NS2000 parallel processing architecture.

A ***Host Node*** serves as the control center of Auspex NS 2000 and manages intelligent ***I/O Node*** modules that are connected via a high speed link based on the Scaleable Coherent Interface (SCI) standard designed especially for parallel processing designs. The product line supports up to three nodes, allowing efficient centralized storage and network management. This translates to lower resource allocation, easier maintenance and administration, and smaller footprint.

The Host Node provides efficient centralized storage and network management.

I/O Node building blocks

Each Auspex NetServer 2000 I/O node has two Intel Pentium processors. The Control Node has one UltraSparc processor for system management. This means a total of seven processors (six for I/O processing and one for host management) for a fully configured system. Each I/O node manages three shelves of four drawers, each with seven drives per drawer or 3 X 4 X 7 = 84 drives per node distributed to three RAID controllers. In the case of 36GB drives this equals 3TB per node or 9TB for a fully configured system. This results in a level of scalability that is not available in other enterprise storage options.

The I/O Nodes provide truly scalable performance.

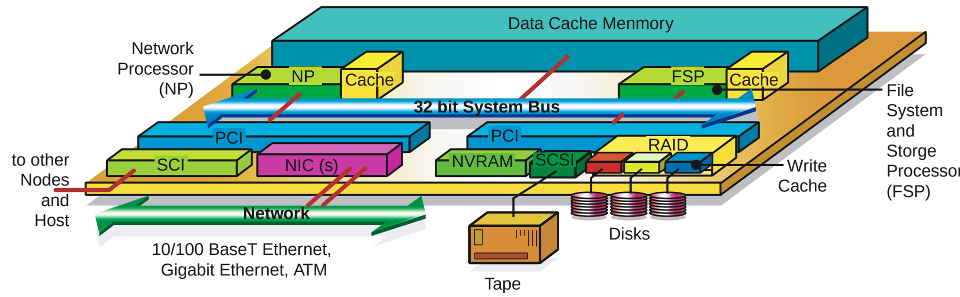
The internal hardware architecture of each I/O node is based on dual Pentium processors that reside on the bus with 1GB of DRAM memory that is used for read cache and program files. About 95% or more of the DRAM memory (950MB) is available for read cache. The NVM write cache is battery-backed SRAM that serves as a cache for fast writes.

Writes execute at memory speed for write back cache applications. An option exists for write through caching where the write goes to disk instead of cache as required by Oracle and other applications. Each node also contains a PCI add-on board with another 128MB of NVM that serves as file systems cache for journaling logs and fast restore. This results in a level of system performance and reliability that is not available in other enterprise storage options.

Auspex NS2000 Functional Multiprocessing (FMP) parallel architecture

Although all hardware is based on the industry-standard Intel high-volume architecture, each processor is assigned a specific function instead of operating symmetrically as in a typical Intel-based SMP system. Each I/O node consists of industry-standard dual Intel processors, dual PCI busses, associated PCI cards and ECC memory as shown in Figure 14.

Figure 14 – I/O Node block diagram illustrating the Auspex patented Functional Multiprocessing (FMP) design.



The FMP architecture improves system availability.

One I/O node processor is called the Network Processor (NP) and manages highly reliable customized software that controls all network protocol and caching functions. The other I/O node processor is called the File and Storage Processor, (FSP) which also executes highly reliable, customized software that handles file system processing and storage processing. The customized software on both processors works closely together in a configuration called the DataXpress kernel. FMP system software consists of a unique custom messaging system that enables efficient network and storage processing on the I/O nodes and efficient system and data management on the host node. It allows the nodes to each efficiently perform their assigned functions. In addition, the FMP architecture improves system availability by isolating the I/O nodes from unplanned outages of the general purpose Solaris OS; I/O processing can continue even when the host node is down.

For more information about Auspex functionally parallel architecture, see the ***NS2000 Product Guide*** at http://www.auspex.com.

Auspex NetServer 2000 Host Node and system management

The Auspex NS 2000 Host I/O node processor runs the standard Solaris OS. This allows all management and control functions typically expected in a data center UNIX environment. Auspex ControlPoint is management control software that provides additional features like non-disruptive local backup on each I/O node. ControlPoint is a Java-based Graphical User Interface (GUI) program that runs in standard web browsers and allows simple and effective remote control of Auspex NS 2000.

Why Auspex chose an FMP over an SMP architecture

SMP systems are suited for compute-intensive applications.

Symmetric Multiprocessing (SMP) is a type of computer architecture that provides fast performance by making multiple processors available to simultaneously execute multiple software programs. SMP systems are suited for compute-intensive applications. In SMP, any processor executes any program or process. A variety of specialized operating systems are available to support SMP architectures.

In contrast, Auspex's patented FMP architecture, each processor executes a predefined set of programs or processes. In a highly predictable environment, like I/O and network processing, this architecture can provide superior performance and scalability characteristics.

Auspex NS 2000 links multiple I/O nodes with a host node for data and system management. This further distributes the work to many processors working in parallel. The efficient Scaleable Coherent Interface (SCI) interconnect allows the multiple nodes of an Auspex NetServer 2000 system to act as one. This provides a superfast network for message exchange between computer nodes of the Auspex system.

SMP machine performance degrades more quickly in performance than an FMP design.

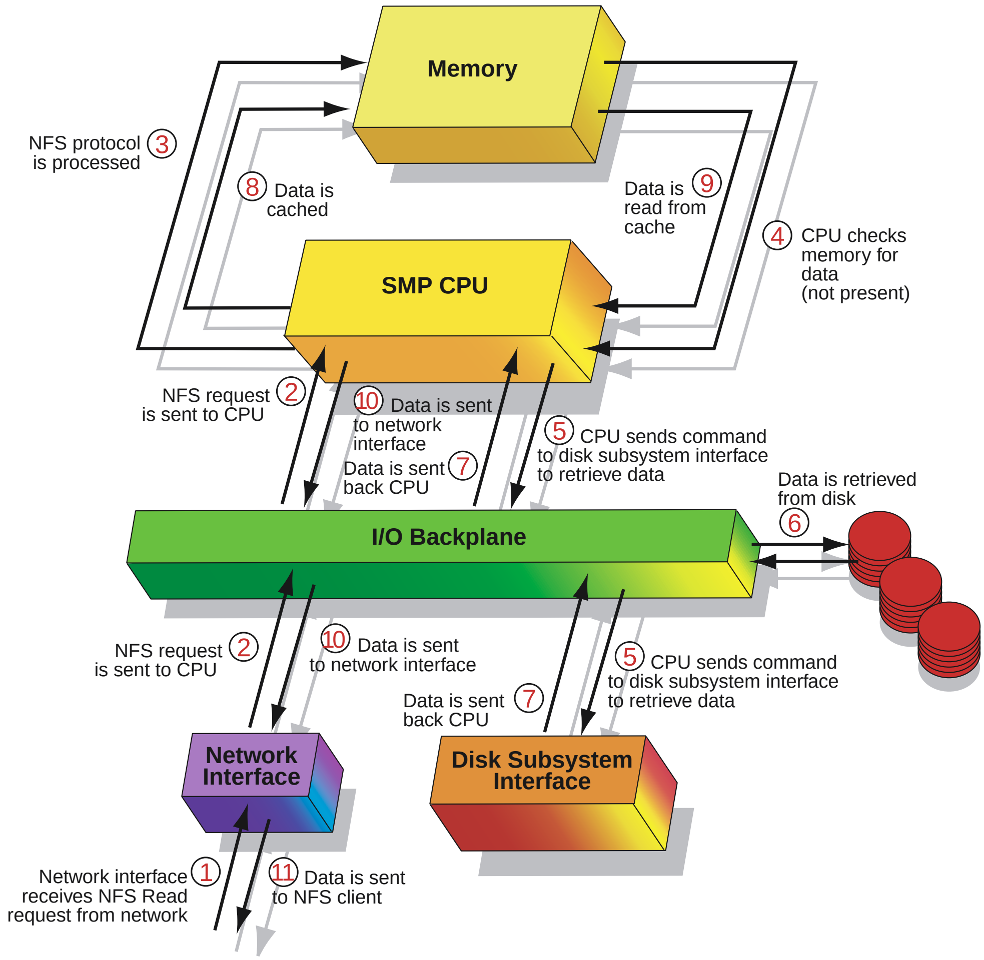
Symmetric Multiprocessing (SMP) systems are much more complicated than single-process operating systems because the operating system must allocate resources to competing programs, or processes, in a reasonable manner. The more processes an operating system must support, the more complex are the scheduling algorithms and the more time the processors spend task (or context) switching and running scheduling programs to determine what to do next. Also, SMP machine performance degrades more quickly than an FMP design. This is because as an SMP system gets busier, the processors must spend more time in scheduling work and less time performing work as in the FMP design of Auspex NS 2000. This makes an 8-processor SMP system more expensive, while yielding only marginally improved performance than a 4 processor SMP for heavy workloads. Not only does the NS2000 distribute different functions to the multiple CPU s in each I/O node, but also additional functional specificity is achieved by distributing different functions to the host node compared to the I/O node. In addition, the NS2000 design allows multiple I/O nodes to function in parallel. This efficient parallel distribution of work illustrates the advantages of the NS2000 parallel architecture compared to SMP architecture (See Table 1).

This NS2000 distribution of work, both within nodes and between nodes, results in higher performance compared to SMP machines especially at higher I/O workloads. At the I/O node level, this is due to reduced task switching and reduced time spent running scheduling routines and reduced processing time on each I/Os shown in Figures 15 and 16. By taking advantage of parallel processing, Auspex NS 2000 avoids the bottlenecks that result from scheduling complexities with heavy workloads in an SMP environment as shown in Table 1. The Auspex design also provides greater predictability and consistency in file service performance compared to an SMP design. This arises from the greater predictability of the time to complete each program on each node and each processor because of the greatly reduced task switching and scheduling overhead compared to SMP.

Architecture	**NS2000 (FMP)**							**SMP**
Processing Nodes	**One Host Node**	**Three FMP I/O Nodes**						**One Node**
		Network Processors			**File System and Storage Processors**			
		NP1	**NP2**	**NP3**	**FSP1 1**	**FSP2 2**	**FSP3 3**	
Network Processing	No	Yes	Yes	Yes	No	No	No	Yes
File System Processing	No	No	No	No	Yes	Yes	Yes	Yes
Storage Processing	No	No	No	No	Yes	Yes	Yes	Yes
Management software	Yes	No	No	No	No	No	No	Yes
Peripheral management	Yes	No	No	No	No	No	No	Yes
Complex scheduling	No	No	No	No	No	No	No	Yes

Table 1 – The NS2000 distributes processing functions not only among processors within nodes but also among processors between nodes. Symmetric Multiprocessing (SMP) computers perform all functions in one node.

Figure 15 – General Purpose servers take eleven operations for each I/O.



Data Backup/Restore

Advantages of the NS2000 parallel backup

Because of its modern parallel architecture, the Auspex NS2000 provides users with simplified scaleability of storage capacity, processors, network connections and performance. In addition, this design provides major advantages in reducing backup windows due to parallel backup of data on each I/O Node. The importance of backup windows is illustrated by the fact that this is the single most important storage concern in a recent survey of 80 enterprises by ITCentrix.

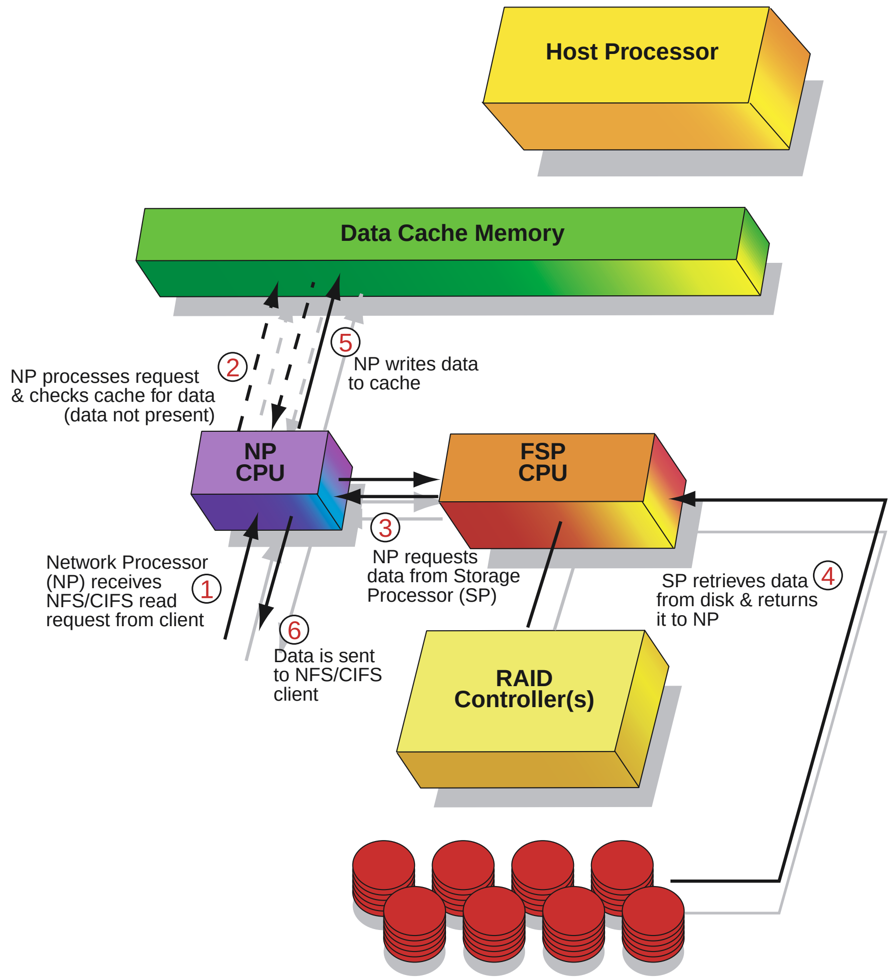


Figure 16 – The NS2000 is more efficient due to its FMP architecture with only six operations for each I/O.

Parallel backup performance of the NS2000

In evaluating backup performance in computer systems, the performance bottleneck is always the transfer rate of the tape drives them. Auspex has a major advantage over competitive designs such as Network Appliance and EMC in this respect since tape drives can be attached to each node. This allows backup to occur in parallel for an entire NS2000 system. With the NS2000, both block (BTE) and File (FTE) backup performance for a 9TB system can be accomplished at the rate of 195 Gbytes/hr for BTE and 186 Gbytes/hr for FTE with the maximum number of tape drives configured locally to each I/O Node for parallel operation. Although backup speed depends on file size, compression, and the type of RAID applied to the file system, a 2:1 compression ratio is typical. Assuming this compression ratio a maximum configured NS2000 system can be backed up completely in 11.8 hours for RAID 1 and 20 hours for RAID 5 at the block level. Similarly, a complete system can be backed up completely in 12.4 hours for RAID 1 and 21 hours for RAID 5 at the file level. A more normal scenario is to do an incremental file backup and only backup changed files. Assuming a 24% hit ratio a maximum configured three I/O Node NS2000 system can be incrementally backed-up up in 3.1 hours for RAID 1 and 5.2 hours for RAID 5. This is shown in Table 2.

Table 2 – Backup performance to tape.

Configuration for Backup	RAID 1 4.5TB useable	RAID 5 7.7TB useable
Backup BTE archive = 195 GB/hr	11.8 hrs	20.0 hrs
Level 0 (FTE) = 186 GB/hr	12.4 hrs	21.0 hrs
Incremental FTE assuming 25% hit ratio	3.1 hrs	5.2 hrs

Third Party UNIX-based backup tools

The Auspex NS2000 system supports a range of third party, UNIX-based backup tools, including products from Legato and Veritas. NeTservices allows backup of Windows data using these UNIX-based, enterprise-level backup products.

NDMP backup with device and file level acceleration

Auspex NS2000 system software provides a Network Data Management Protocol (NDMP) server with two data moving engines and interfaces that are designed to dramatically accelerate system backup. NDMP is a standard protocol that can be implemented on any server or backup device. NDMP hides the unique interfaces from third party backup software which allows this software to execute on any NDMP compliant system on the network (such as the NS2000 I/O Node, and control backups on the NS2000 using standard commands).

The NS2000 uses snapshot along with the ***Block DataXceleration Engine (BDX)*** which provides the capability to stream blocks of data from disk to tape, and provides the foundation for extremely rapid image backup. NDMP can also use the ***File DataXceleration Engine (FDX)*** which provides a similar interface that acts at the file system level and provides the basis for rapid file-by-file or directory backup. Data passes directly from disk to tape if the tape is attached to the same I/O Node or from disk on one FSP across the SCI network to tape on another FSP if the tape is attached to another I/O node. These engines in conjunction with NDMP are robust data management tools, which enable high-performance backup and restore of file systems, directories, and individual files along with UNIX and Windows security information, including NT ACLs, SID, etc.

NDMP compliant products from Veritas, Legato and other companies, can be used for high-performance backups of data on Auspex servers. Auspex NS2000 system software also provides UNIX commands that are capable of backup and restore of UNIX and Windows data along with associated security information.

Device level and file level snapshots

To create a device level snapshot all data is first flushed to disk and the volume is made momentarily quiescent to ensure stability. Once a snapshot occurs, any time a data block is changed a copy of the original is made and saved in a separate designated partition. Both types of snapshots preserve both UNIX and Windows data along with associated security information. Users can specify the interval at which these snapshot copies are made.

A fast backup method using mirroring

Back up of the software development environment can be achieved in an expedient manner with the use of Netserver's mirroring capabilities and local backup devices. Through an automated script, administrators can:

1. Lock the database;
2. Copy the database(s) to the NFS mounted file system in a backup area;

3. Detach a mirror of the stored data on the NetServer; and,
4. Unlock the database and backup the detached filesystems to local tape on NetServer.
5. With backup completed, the members are simply re-attached to the mirror. This scenario decreases database lock times and ensures data reliability.

Auspex NS2000 Servers and the Internet

Ever since its formation in 1987, Auspex has been delivering network server and storage solutions to companies with mission-critical profiles similar to today's Internet environment: 24x7 uptime, high-performance while handling thousands of operations per second, and multiple-terabyte capacities. With the NS2000 Enterprise Storage Server, introduced in January 1999, Auspex has increased its value to mission-critical customers, particularly those building Internet businesses.

Auspex has taken the basic Network Attached Storage model, and added significant capabilities. First, the NS2000 is based on the patented DataXpress™ architecture. DataXpress is unique in that it is the only *system of any kind* that can specifically scale Internet data throughput. Also, the NS2000's system software, called NetOS™ is tuned for DataXpress and scales gracefully as the system is expanded. With DataXpress and NetOS, as Internet loads inevitably increase, networks, processors, controllers, and disks can be added to meet the higher demands. Without these Auspex technologies, additional storage units would need to be acquired and installed. Beyond cost issues, the benefits of having a centralized Network Attached Storage system with a single, master copy of content would be diminished. Since the Internet and data/load growth are synonymous, the combination of DataXpress and NetOS allows a technology manager to deploy the right solution for today's needs with confidence that it is equipped to handle tomorrow's.

The NS2000 generates industry leading read performance.

Second, the Internet is primarily a read environment. This is why Auspex Network Attached Storage is optimized for reads. How? The NS2000's FastFLO file system utilizes a technique called contiguous block allocation. This means that when a file is written to disk, the file system ensures that the blocks are organized contiguously, or side-by-side. On read operations, disk head efficiency is greatly optimized since the head can move in one direction versus hopping around the spindle to access the different blocks. Using this approach, the NS2000 can generate industry leading read performance – 75MB/sec to a Network Attached Storage server.

Through its support of fragmentation, the NS2000 with FastFLO maximizes storage efficiency.

Third, all Internet data is not created equally and handle a wide variety of different types of data. A typical E-commerce site might include product images, text descriptions, and video/audio-based marketing content, each file a very different size. The NS2000 accounts for these differences through technology in the FastFLO file system. The standard block size within FastFLO is 4KBytes. With large files, disk I/Os are minimized and data throughput is increased since FastFLO can access data in 4KB chunks. But I/O optimization isn't the issue with very small files; it's disk utilization. To minimize wasted storage, FastFLO supports *fragments*, or pieces of a block. Specifically, a 4KB block can be sliced into (8) 512 byte fragments. With fragment support, a 500-byte file would occupy only one 512-byte fragment on disk. Without fragments, a 500-byte file would consume a full 4KB block – wasting 3.5KB of space. For a file system with thousands or even millions of small files, significant disk space would be rendered useless. Through its support of fragmentation, the NS2000 with FastFLO maximizes storage efficiency and minimizes disk costs. The FastFLO file system's dynamic read algorithms and robust block and fragment model helps an Internet site increase throughput and improve disk utilization.

FTP will remain the choice for those wanting to transfer medium to large-size files.

Fast FTP Support

By streaming FTP data directly to requesting clients, the need to have a UNIX host intervene in the actual data transfer is eliminated, reducing complexity and increasing transfer speed of FTP files to the Internet. Since many ISPs and Intranet sites currently provide FTP services as well as HTTP services, and until WebNFS becomes ubiquitous, FTP will remain the choice for those wanting to transfer medium to large-size files. The use of this unique feature of NAS servers like the Auspex NS2000 will provide better data throughput and greater scalability for sites, which provide FTP services.

Auspex NAS solutions certified interoperable with 3rd party applications and peripheral hardware

While Auspex's core products are NAS-based file servers, optimization of the total network and storage architecture solution is the goal. Achieving maximum functionality and performance in application environments, such as software development, requires adherence to industry standards and the integration and certification of 3rd party products. This addresses all areas of the total solution and includes:

- Network "trunking" technology (e.g. Cisco Fast EtherChannel, "Trunking")
- Backup & Restore (support for NDMP, Legato, Veritas, ATL tape drives and libraries)
- Application software (certification of Rational ClearCase configuration management system)
- Database[1]

[1] The NS2000 is Oracle certified.

Conclusion and Testimonial Quotes

7

Summary of a multi-level, functionally specialized storage architecture for software development

This report has explained the inner workings of the software development industry and provided recommendations for optimizing the design of network and storage architectures to support the software development life cycle. Facing the challenge to speed development while producing high quality software products, software development organizations increasingly depend on their network and computing infrastructure to provide the bandwidth and performance required to store and deliver data reliably and on-demand to all developers across the enterprise.

A multi-level, NAS based, functionally specific infrastructure model is proposed.

We have reviewed how large scale software development sites function in terms of engineering processes. Many organizations have found serious weaknesses with various implementations of a distributed compute/distributed data infrastructure, particularly as projects and system capacities grow. Continuing with such an infrastructure results in expensive network complexities, data flow bottlenecks, and an inability of the developers to effectively collaborate using shared data. A multi-level, NAS based, functionally specific infrastructure model is proposed for optimal local and wide area data flow. Key benefits of such an infrastructure model directly address the goals of the business, developers, and system administrators.

Benefits to the business

- Better time to market of quality products
- Lower TCO
- More accomplished developers and staff with greater job satisfaction

Benefits to the developers

- Data is always available when and where it is needed
- Easier collaboration
- Ability to accomplish more without distracting inefficiencies of the infrastructure

Benefits to the system administrators

- Simplified system management as a result of centralized management
- Scalability, ease of expansion, capacity planning
- Easier troubleshooting and resolution of problems

Testimonials from customers for the use of NAS in software development

There exist many success stories with the use of NAS products like the Auspex NS2000 in the software development industry. Some real life quotes from software professionals and system administrators speak volumes to illustrate the problems solved by installing Auspex NetServers in complex software development environments. Real customer examples are often the best reason to deploy enterprise class NAS products such as the Auspex NS2000:

After we installed NAS the job turnaround time was reduced to 5 minutes.

Data I/O performance is critical to builds and access to code

- "The communications industry market window is so short. Before we installed NAS, our jobs took 30 minutes to 2 hours. It was taking all night to catch up. After we installed NAS the job turnaround time was reduced to 5 minutes. My users' culture now expects that response." [2]
- The second challenge was performance. "It was taking so long to run the builds because the systems that were performing NFS file services were little workstation class systems well known for not providing adequate NFS serving."

Distributed data – a system management nightmare

The system was a spider web of cross-mounts.

- "All the HP code was kept on the HP server, and anything that needed to access that code was cross-mounted off the HP server. Multiply that by about 30 different UNIX systems and you end up with this massive cross-mounted network. It was very unmanageable and some of the systems were not optimized for file serving; HP's and IBM's systems are notoriously bad network file servers. And some of them were very old systems because we still support extremely old versions of UNIX. One other problem determination: because everything was cross-mounted to everything else, trying to figure out the source of a problem was a tiresome task. All it took was for some engineer to keep some little piece that was necessary for the build on his UNIX desktop in his office, turn off his UNIX system and go home — and the build would break that night. It was horrible to figure out who or what was causing the problem each time. The system was a spider web of cross-mounts and it took hours to chain through it and determine where the problem originated."
- "By centralizing all the data on the Auspex, we were able to get all that cross mounting under control, but we were also able to cut the build times in half from 14 hours down to about 7 hours."

Backup – a critical part of the infrastructure

"Auspex simplified our backup procedure."

- The third challenge related to backing up the system. "We are a software company. Our assets are software. With the previous system, all of our assets were spread all over the place. Even worse, we didn't know how well each individual workstation was being backed up because it might be on a developer's desk, or it might be in the lab, and the responsibilities for backing up the systems were not clearly established. Some had local drives attached to them, others were backed up by some network backup technology, but there was no consistency across the board on how that was done. There were even systems that fell between the cracks and were not being backed up regularly. Between these three problems, network storage was a nightmare. It was very clear that we needed to centralize our data storage and protect our software assets."
- "By having everything in the same place on the Auspex, we have also simplified our back up procedure. There is just one place to check the backup and only one backup solution that we have to understand. It hugely simplifies our ability to per form recovery of missing data as well."

[2] Software Development Manager (from IDC's Network Attached Storage: A Compelling Story for Storage Consolidation. This report can be downloaded from the Solutions section of www.auspex.com.

Availability

- "The requirement we had for a new server was a single, highly available, fast file server that we could move all the code to. We wanted to know where everything is; we wanted assurance that the server was going to be up all the time; and we wanted a performance level that would enable us to shorten our build time."

Data sharing – UNIX/NT

- "Since the R&D workload is widely varied and can be significant, we are migrating the R&D side to the NS2K where it will be handled more effectively. Since the NS2K has the bilingual file serving capability, to serve out NFS and CIFS, it is ideal for this application which has growing storage needs."

System scalability and parallel architecture

- "Capacity is important for me, and the NS2K provides the necessary capacity and scalability. I can easily see the day when I will have 9 terabytes on a single file server."
- "We brought in the original Auspex server because several engineers in our R&D group had experience with Auspex and preferred to keep working with Auspex. For the NS2K, however, we did a more involved evaluation against the other popular vendors in the market. We decided to buy the NS2K because when Auspex redesigned the architecture they implemented Solaris on the host processor, allowing the host processor to be migrated to the new generation of chips, and therefore increase speed and support more memory."
- "The NS2K has made it possible for us to deploy our key software on the host processor and monitor the system with our own application service management product, which we have been wanting to do. With the NS2K, now we can plug the file server into our nerve center and monitor the system.
- "The other server options on the market did not offer a host processor functionality at all. They did not have a general purpose UNIX operating system that we could plug into; at least not that was exposed to us, the end user. They had control programs, but they were proprietary to that vendor and would not support our product."
- "In terms of competitiveness, Auspex gives us a significant advantage since we are doing the builds on the Auspex, the server performance has become a critical component in meeting our product development deadlines. I cannot over-emphasize the importance of the nightly builds actually completing overnight. Just like everyone else in this industry, we are developing in Internet time. We are rolling new products out the door and anything we can do to speed the development time — so the developers are not waiting for a build to complete — is going to ensure that we get to market that much faster. And these days getting to market on time with a feature set and product is everything."
- "All the assets we have is in our people and the code they write. What is important to us is having fast and highly available access to that code so these expert developers do not have difficulty accessing the data they need to produce our next generation of (code). Whether they are completing a build, restore or redesign, they need access — and they need it fast."

Auspex gives us a significant advantage since we are doing the builds on the Auspex.

"The Auspex server helps provide an environment that keeps our engineers happy."

- "Like any corporation, we also want to attract talent to our company. In a highly competitive high tech employer market, one of the attractions to keep engineers is a high performance network. The engineers want the network and file servers to be like the telephone — always there whenever they want it. And that keeps them much happier than a negative environment in which the system is down all the time; you're constantly doing restores; and you're always waiting for something. It is hard to keep good people in that kind of environment and if we don't keep good people, that is our intellectual asset going out the door. The Auspex server helps provide an environment that keeps our engineers happy."
- "The truth is that there are several options out there if you are only looking for a fast server, but we needed more than that. We need support from a server vendor that understands what we are trying to accomplish with our business. Every time I've talked to Auspex, I've had the feeling the support organization understands that we are running mission critical applications on this system. This is something that the other vendors just don't seem to understand."

Reboot Mentality

- "There is this mentality with some of the other vendors: if the server hangs, reboot it. It's the Windows mentality. If you have a PC, you are used to the idea that if it locks up, you just hit control-alt-delete and reboot it. Many of the server vendors have adopted this same outlook."
- "In fact, some vendors sell how fast they reboot as a feature," he continued. "They are proud of how fast their server will reboot. And to me that's missing the point. They are not understanding that I don't want to reboot. I want to be able to get by until I can schedule a change to the system. I don't want to have to do really fast reboots in the middle of the day."

"Auspex stays up and running."

- "It's the difference between understanding the enterprise and its mission critical applications. Instead of a departmental level mentality where you walk around and tell everyone to go to lunch while you reboot the system. With the amount of corporate data we have, we need a server like Auspex that has the capacity to handle it and the stability to stay up and running. Rebooting while everyone goes to lunch just doesn't fly anymore."

Glossary of Terms

10BaseT
Ethernet with a data transfer rate of 10 Mbits/sec.

100BaseT
Also known as Fast Ethernet with a data transfer rate of 100 Mbits/sec.

ATM
Asynchronous Transfer Mode. A suite of network protocols providing low-level services spanning local and wide-area networks. ATM is intended to provide the switching and multiplexing services necessary to carry voice, data and video and multimedia traffic using fixed 53-byte cells. Standards are being defined to allow ATM to emulate traditional LANs (LANE).

b
Abbreviation for "bit" where 8 "bits" comprise a byte.

B
Abbreviation for byte or the equivalent of one character in text.

Bridge
A bridge is a device that connects two local-area networks (LANs), or two segments of the same LAN. The two LANs being connected can be alike or dissimilar. For example, a bridge can connect an Ethernet with a Token-Ring network. Unlike routers, bridges are protocol-independent. They simply forward packets without analyzing and re-routing messages. Consequently, they're faster than routers, but also less versatile.

Browser
Browser is short for Web browser, a software application used to locate and display Web pages. The two most popular browsers are Netscape Navigator and Microsoft Internet Explorer. Both of these are graphical browsers, which means that they can display graphics as well as text. In addition, most modern browsers can present multimedia information, including sound and video, though they require plug-ins for some formats.

Bug Fix
A request to correct software functionality that does not behave according to the design.

Cache
Cache (pronounced cash) can be either a reserved section of main memory or an independent high-speed disk storage device. Two types of caching are commonly used in personal computers: memory caching and disk caching. Disk caching can dramatically improve the performance of applications, because accessing a byte of data in RAM can be thousands of times faster than accessing a byte on a hard disk. When data is found in the cache, it is called a cache hit, and the effectiveness of a cache is judged by its hit rate. Many cache systems use a technique known as smart caching, in which the system can recognize certain types of frequently used data. The strategies for determining which information should be kept in the cache constitute some of the more interesting problems in computer science. Caching is applied to the Internet whenever a proxy server or web cache is placed at the Internet boundary. The same principles apply as in personal computer caching – that is to store frequently requested data closer to the requester whether the user is on the internal network or over the web.

CASE

Computer Aided Software Engineering. Software which assists in the design and implementation of other software.

Change Set

A grouping of source code modifications that comprise the entire work for a bug fix or enhancement.

CIFS

Common Internet File System. A connection-oriented, network file-sharing protocol developed by IBM and Microsoft as part of LAN Manager. CIFS is the native file sharing protocol for systems running Windows for Workgroups, Windows 95/98 and Windows NT. Sometimes referred to as SMB.

Client Server

Client Server Architecture is a network architecture in which each computer or process on the network is either a client or a server. Servers are powerful computers or processes dedicated to managing disk drives (file servers), printers (print servers), or network traffic (network servers). Clients are PCs or workstations on which users run applications. Clients rely on servers for resources, such as files, devices, and even processing power. Another type of network architecture is known as a peer-to-peer architecture because each node has equivalent responsibilities. Both client/server and peer-to-peer architectures are widely used, and each has unique advantages and disadvantages. Client-server architectures are sometimes called two-tier architectures.

Control Point™

Auspex's proprietary management control software.

CPU

Central Processing Unit. Can refer to either a processor chip such as Sun's SPARC or Intel's Pentium, or to a processor chip or chips and support circuitry on a CPU board.

DataXpress

Communication among the NS2000's multiple hardware processors and software processes are handled by DataXpress, a low-overhead message-passing kernel executing on each processor.

ECAD

Electrical Computer Aided Design

EM-Net

The NS2000 Environmental Monitoring Network that connects to all chassis in an NS2000 system and reports a variety of control information to the Host Node.

Enhancement

A request to add or modify functionality of a software product, which would require an update to the design, document.

Ethernet

A Local Area Network (LAN) protocol developed by Xerox in cooperation with Digital Equipment and Intel in 1976. Ethernet supports a star or bus topology and supports a data transfer rate of 10 megabits per second or 10 Mbps. The Ethernet specification formed the basis of the IEEE 802.3 standard, which specifies the physical and lower software layers. Ethernet uses the CSMA/CD access method for handling simultaneous demands and is one of the most widely implemented LAN standards.

FastFLO

The NS2000 proprietary file system that is optimized for providing high performance and consistent file services.

Fiber Channel

An ANSI standard designed to provide high-speed data transfers between workstations, servers, desktop computers and peripherals. Fibre channel makes use of a circuit/packet switched topology capable of providing multiple simultaneous point-to-point connections between devices. The technology has gained interest as a channel for interconnect. Fibre channel can be deployed in point-to-point, arbitrated loop (FC-AL), or switched topologies. Fibre channel nodes log in with each other and the switch to exchange operation information on attributes and characteristics. This information includes port names and port IDs and is used to establish interoperability parameters.

FSP

The File System and Storage Processor refers to one of the two Intel Pentium processors on an I/O Node of an NS2000 system. This processor runs highly optimized microcode that manages all file system and storage processing of the I/O Node and communicates with other I/O Nodes and the Host Node. See also Network Processor (NP).

FTP

FTP is an abbreviation of File Transfer Protocol, the protocol used on the Internet for sending files.

Functional specialization

A hierarchical, functionally specialized network and storage architecture is based on a “fit for function” approach where each component (i.e., data servers, data base and build machines, network components) are deployed for specific functions, allowing choice of the right components and features for each function.

Gateway

A gateway is a combination of hardware and software that links two different types of networks. An email gateway for example allows users on two different types of email systems to exchange emails.

Gigabit Ethernet

Gigabyte Ethernet is a standard of the IEEE 802.3 committee which will provides a mechanism for conveying Ethernet format packets at GB/s speeds. The goals of the gigabit Ethernet effort include: preserve the CSMA/CD access method with support for 1 repeater, use the 802.3 frame format, provide simple forwarding between Ethernet, fast Ethernet and gigabit Ethernet, support both fiber and copper (if possible), and accommodate the proposed standard for flow control. At the time of this writing it appears that Fibre channel will be adopted to provide the physical layer for the first implementations of gigabit Ethernet.

Gigabyte

1024 Megabytes.

GUI

An acronym referring to a Graphical User Interface that is the screen presented to a user in any computer application.

HTTP

HTTP is an abbreviation for HyperText Transfer Protocol, the underlying protocol used by the World Wide Web. HTTP defines how messages are formatted and transmitted, and what action Web servers and browsers should take in response to various commands. For example, when you enter a URL in your browser, this actually sends an HTTP command to the Web server directing it to fetch and transmit the requested Web page. The other main standard that controls how the World Wide Web works is HTML, which covers how Web pages are formatted and displayed. HTTP is called a stateless protocol because each command is executed independently, without any knowledge of the commands that came before it. This is the main reason that it is difficult to implement Web Sites that react intelligently to user input. This shortcoming of HTTP is being addressed in a number of new technologies, including ActiveX, Java, JavaScript and cookies.

Hub

A hub is a common connection point for devices in a network. Hubs are commonly used to connect segments of a LAN. A hub contains multiple ports. When a packet arrives at one port, it is copied to the other ports so that all segments of the LAN can see all packets. A passive hub serves simply as a conduit for the data, enabling it to go from one device (or segment) to another. So-called intelligent hubs include additional features that enable an administrator to monitor the traffic passing through the hub and to configure each port in the hub. Intelligent hubs are also called manageable hubs. A third type of hub, called a switching hub, actually reads the destination address of each packet and then forwards the packet to the correct port.

Internet

Internet refers to a global network connecting millions of computers. As of 1999, the Internet has more than 200 million users worldwide, and that number is growing rapidly. More than 100 countries are linked into exchanges of data, news and opinions.

Intranet

A network based on TCP/IP protocols (an Internet) belonging to an organization, usually a corporation, accessible only by the organization's members, employees, or others with authorization. An Intranet's Web Sites look and act just like any other Web Sites, but the firewall surrounding an Intranet fends off unauthorized access. Like the Internet itself, Intranets are used to share information. Secure Intranets are now the fastest-growing segment of the Internet because they are much less expensive to build and manage than private networks based on proprietary protocols.

IP

The IP (Internet Protocol) is the underlying protocol for routing packets on the Internet and other TCP/IP-based networks. IP is an internetwork protocol that provides a communication standard that works across different types of linked networks for example Ethernet, FDDI or ATM. In an internetwork, the individual networks that are joined are called subnetworks or subnets. IP provides a universal way of packaging information for delivery across heterogeneous subnet boundaries. See also TCP Transmission Control Protocol.

LAN

Local area networks or LANs are networks of computers that are geographically close together; this usually means within the same building. Most LANs are confined to a single building or group of buildings. However, one LAN can be connected to other LANs over any distance via telephone lines and radio waves. A system of LANs connected in this way is called a wide-area network (WAN).

LANs connect workstations and personal computers. Each node (individual computer) in a LAN has its own CPU with which it executes programs, but it is also able to access data and devices anywhere on the LAN. This means that many users can share expensive devices, such as laser printers, as well as data. Users can also use the LAN to communicate with each other, by sending e-mail or engaging in chat sessions. There are many different types of LANs and Ethernet LANs is the most common for PCs. Most Apple Macintosh networks are based on Apple's AppleTalk network system, which is built into Macintosh computers. LANs are capable of transmitting data at very fast rates, much faster than data can be transmitted over a telephone line; but the distances are limited, and there is also a limit on the number of computers that can be attached to a single LAN.

LOC
Acronym for the number of "Lines of code."

MCAD
Mechanical Computer Aided Design

Merge
The activity of including multiple change sets that are separated into a single release or integration milestone.

Metrics
Gathering objective data such as line of code counts, change set information, identifying areas of a piece of software that were modified and the developer who made the modifications.

MIB
Management Information Base is a set of standards for detailed system information that is reported to a control console for SNMP compliance. It's intent is to provide common metrics for heterogeneous computer systems.

MTBF
Mean Time Between Failure. A key component of the availability equation, AVAILABILITY = (MTBF – MTTR) ÷ MTBF. Example: A server that on average fails once every 5,000 hours and on average takes 2 hours to diagnose, replace faulty components and reboot would have an availability rating of (5,000 – 2) ÷ 5,000 = 99.96%.

MTTR
Mean Time To Repair. Includes the time taken to diagnose the failure, replace or repair faulty component(s) and reboot the system. See MTBF.

NAS
Network Attached Storage.

NDMP
NDMP is a standard protocol for network-based backup of network-attached storage. NDMP hides the unique interfaces from third party backup software which allows this software to execute on any NDMP compliant system on the network (such as the NS2000 Host Node, and control backups on the NS2000 using standard commands.

NIC
Network Interface Cards (or NICs) in the NS2000 support 10/100BaseT Ethernet, Gigabit Ethernet or ATM. There are from one to three on each I/O Node.

NFS
Network File System. NFS is an ONC application-layer protocol for peer-to-peer, distributed, file system communication. NFS allows a remote file system (often located on a file server) to be mounted transparently by client workstations. The client cannot perceive any functional difference in service between remote and local file systems (with trivial exceptions). NFS is the most popular ONC service, has been licensed to

over 300 computer system vendors, runs on an estimated 10 million nodes and is a de facto UNIX standard. See also VFS, ONC, and NFSv3.

NFSv3

NFS version 3. References to NFS generally imply NFS version 2 protocol. NFS version 3 is an update to the NFS protocol. Significant among the many changes made for NFSv3 are the adoption of a safe asynchronous write protocol and the use of block sizes up to 64 KB. Other protocol changes are intended to improve the overall network and client efficiency and provide improved support for client-side caching.

NFS ops/s

NFS operations per second. Typical NFS operations include: lookup, read, write, getattr, readlink, readdir, create, remove, setattr, and statfs.

Node

See Fibre Channel.

NP

The Network Processor (NP) refers to one of the two Intel Pentium processors on an I/O Node of an NS2000 system. This processor runs highly optimized microcode that manages all network processing of the NS2000 I/O Node and communicates with other I/O Nodes and the Host Node. See also File System and Storage Processor (FSP).

NTFS

A term that refers to Windows NT file system.

NVM

Non volatile memory is a term used to refer to battery backed up DRAM so that data will not be lost in the event of power failure.

NVRAM

Non-volatile random access memory such as static RAM will not lose data in the event that power is lost to the chip.

Operating System

The operating system is the most important software program that runs on a computer. The Operating System (OS) performs basic tasks such as recognizing input from a keyboard, sending output to the display screen, keeping track of files and directories on the disk and controlling peripheral devices such as disk drive and printers or a mouse. The OS acts as a traffic cop and schedules the various programs that the computer executes. The OS is also responsible for security, ensuring that unauthorized users do not access the system. Operating systems can be classified as follows: 1) Multi-user – allows two or more users to run programs at the same time.
2) Multi-processing – supports running a program on more than one CPU.
3) Multi-tasking – allows more than one program to run concurrently.
4) Multi-threading – allows different parts of a single program to run concurrently.
5) Real Time – Usually a stripped down OS that responds to input instantly.

Packet

A packet is a piece of a message transmitted over a packet-switching network. One of the key features of a packet is that it contains the destination address in addition to the data. In IP networks, packets are often called datagrams.

Parallel Development

Development activities, which occur simultaneously for either multiple change sets or releases and are physically separate from each other. Parallel development efforts are usually merged into an integration environment.

Parallel Development

Parallel processing refers to when a single computer simultaneously uses more than one CPU to execute a program. Ideally parallel processing makes a program run faster because there are more CPUs running it. In practice, it is often difficult to divide a program so that separate CPUs can execute different portions without interfering with each other. Among NAS vendors, only the Auspex NS2000 effectively overcomes this problem by designing each I/O Node with two processors each performing separate portions of the network file-serving task. In addition the NS2000 links multiple I/O Nodes together by a highly efficient Scalable Coherent Interface (SCI) interconnect which allows the multiple nodes to act as one system. See also Functional Multiprocessing (FMP).

PCI

The Peripheral Channel Interconnect is an ANSI standard for an I/O bus used predominantly in PC design.

PDU

Power Distribution Unit or Power Shelf in the NS2000. A cabinet model contains from three to seven power supplies and a stack model contains up to three power supplies. Both types of PDUs can be N+1 redundant. See also N+1.

Router

A Router is a device that connects any number of LANs. Routers use headers and a forwarding table to determine where packets go, and they use ICMP to communicate with each other and configure the best route between any two hosts. Very little filtering of data is done through routers. Routers do not care about the type of data they handle. See also ICMP.

SCI

Scalable Coherent Interface is an ANSI standard (#1596-1992) that is the modern equivalent of a processor-memory-I/O bus and a Local Area Network combined and made parallel to support distributed multiprocessing. The SCI interconnect has very high bandwidth, very low latency and a scalable architecture. This allows building large high performance systems and is used by Convex/HP supercomputers, Sun Clusters, Sequent, Auspex and others. Network latency has been measured at 150 times less than previous network connections for efficient and fast communication between computer nodes.

SCSI

Small Computer System Interface. An intelligent bus-level interface that defines a standard I/O bus and a set of high-level I/O commands. The SCSI busses in the NS2000 are used to connect multiple peripheral devices such as disk drives tape drives. Each SCSI device has an intelligent SCSI controller built into it. There are currently many flavors of SCSI defined by different bus widths and clock speeds. The seven major variations of SCSI are SCSI 1, SCSI 2 (Fast / Narrow), SCSI 2 (Fast / Wide), Ultra SCSI (Fast / Narrow), Ultra SCSI (Fast / Wide) – also called SCSI 3, Ultra 2 SCSI (Narrow), Ultra 2 SCSI Wide. Single ended SCSI is used when the peripheral device is close to the point of attachment as in the NS2000 method of attaching disk drives. Differential SCSI provides for reliable operation over greater distances and is used in the NS2000 for tape drive connections.

SE

System Engineer(s) perform a variety of technical pre and post sales services for customers and prospects.

Snapshot
A term that refers to a copy of a file system at a certain point in time. Snapshots are used for backup and recovery.

SMP
Symmetric Multi-Processing. A computer architecture in which processing tasks are executed in parallel on multiple, identical, general-purpose CPUs that share a common memory. SMP computer systems usually have modified operating systems that can themselves execute concurrently. The SMP architecture offers high computational throughput, but not necessarily high I/O throughput. See FMP.

SNMP
Simple Network Management Protocol. SNMP is a protocol used for communication between simple, server-resident SNMP agents that respond to network administration requests from simple-to-sophisticated SNMP manager tools running on remote workstations.

Software Design
The activity of planning the behavior and functionality of a software product.

Software Integration
The activity of merging multiple change sets into a single release or milestone.

Solaris 2.x
Sun's UNIX operating system based on System V release 4.

SPARC
Scalable Processor Architecture. SPARC International's specification for the Reduced-Instruction-Set-Computer (RISC) CPUs found in systems sold by Sun Microsystems, Auspex, etc.

TB
A Terabyte (TB) equals 1024 Gigabytes.

TCO
Total Cost of Ownership is a financial calculation of all costs of owning and operating computer equipment.

TCP
Transmission Control Protocol or TCP is a transport layer component of the Internet's TCP/IP protocol suite. It sits above IP in the protocol stack and provides reliable data delivery services over connection-oriented links. TCP uses IP to deliver information across a network and makes up for the deficiency of IP providing a guarantee of reliable delivery services that IP does not. TCP messages and data are encapsulated into IP datagrams and IP delivers them across the network.

UAT
User Acceptance Testing. UAT is the period of time a client runs proprietary test scenarios on a new release of software prior to installing the software into their production environment. Bug fix requests may be issued during this stage.

URL
URL is an abbreviation of Uniform Resource Locator, the global address of documents and other resources on the World Wide Web. The first part of the address indicates what protocol to use, and the second part specifies the IP address or the domain name where the resource is located.

Variant
A customized version of software for a particular client with regard to functionality or a customized build for the client's hardware.

VPN
VPN is an abbreviation for virtual private network, a network that is constructed by using public wires to connect nodes. For example, there are a number of systems that enable you to create networks using the Internet as the medium for transporting data. These systems use encryption and other security mechanisms to ensure that only authorized users can access the network and that the data cannot be intercepted.

WAN
Wide Area Networks or WANs are networks of computers that are geographically dispersed and connected by radio waves, telephone lines or satellites.

Waterfall Model
A process of software development which required that each stage in the process be complete in order for the next stage in the development process to begin.

Web
A system of Internet servers that support specially formatted documents. The documents are formatted in a language called HTML (HyperText Markup Language) that supports links to other documents, as well as graphics, audio, and video files. This means you can jump from one document to another simply by clicking on hot spots. Not all Internet servers are part of the World Wide Web. There are several applications called Web browsers that make it easy to access the World Wide Web. Two of the most popular are Netscape's Navigator and Microsoft's Internet Explorer.

Web Site
A site (location) on the World Wide Web. Each Web site contains a home page, which is the first document users see when they enter the site. The site might also contain additional documents and files. Each site is owned and managed by an individual, company or organization.

Corporate Headquarters
2800 Scott Boulevard
Santa Clara, CA 95050
USA
408.566.2000 **phone**
800.735.3177 **fax**

http://www.auspex.com **Web**

Japan
Auspex K.K.
ATT Shinkan, 6K
2-11-7 Akasaka
Minato-ku
Tokyo 107-0052
Japan

+81 3 3586 1751 **phone**
+81 3 3586 1778 **fax**

United Kingdom
Auspex Systems, Ltd
Portland House
Aldermaston Park
Church Road
ALDERMASTON
Reading RG7 4 HR
United Kingdom

+44 1189407300 **phone**
+44 1189407333 **fax**

France
Auspex Systems, SA
40, Boulevard Henri Cedex
France

+33 1 42 04 9150 **phone**
+33 1 42 04 9151 **fax**

Germany
Auspex Systems, GmbH
Ingolstaedter Stasse 22
80807 Muenchen
Germany

+49 89 350151-0 **phone**
+49 89 350151-50 **fax**

Auspex Asia Pacific
#06-02 International Building
360 Orchard Road
Singapore
238869

+65 732 9770 **phone**

Auspex Worldwide

AUSPEX

www.auspex.com