

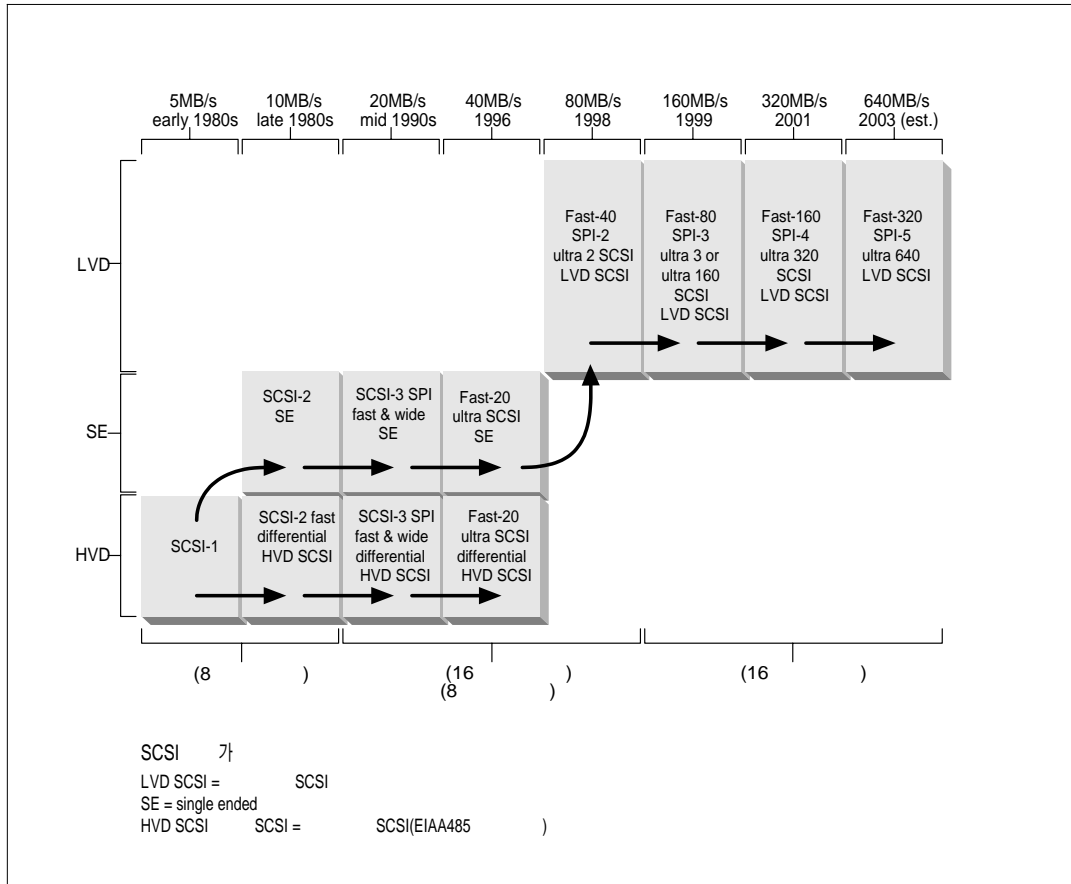
1

- SCSI SAN
- SAN ?
- SAN
- NFS SMB NAS
- SAN NAS
- ?

SAN NAS ?

컴퓨팅이 시작할 때부터 사람들은 컴퓨팅 자원을 공유하고 싶어했다. 1961년 버로프사 (Burroughs Corporation)가 이런 요구를 만족시키기 위해 멀티 프로그래밍과 가상 메모리를 개발했다. 제휴사인 슈가트(Shugart)도 사람들이 디스크를 쉽게 사용하고 공유하는 방법에 관심이 있음을 알고 1979년에 SCSI(Small Computer System Interface)의 전신인 SASI (Shugart Associates System Interface)를 만들었다. 그리고 썬(SUN)사의 기술팀도 더 나은 파일 공유 방식을 원하는 사람들을 위해 1980년대 초 NFS를 개발해 1984년에 발표하였는데 이는 나중에 유닉스 커뮤니티의 일반적인 파일시스템 공유 방식이 되었다. 1984년 사이텍(Sytec)이 IBM에서 사용하기 위해 개발한 NetBIOS는 SMB 프로토콜의 기반이 되고, SMB는 윈도우 환경에서의 주된 파일 공유 방법인 CIFS이 되었다.

SAN(Storage area network)과 NAS(Network attached storage)는 새로운 개념이 아니다. SAN은 SCSI가 발전한 형태이고, NAS는 NFS와 CIFS가 발전한 형태이다. [그림 1-1]를 보면 그 역사를 이해하기 쉬울 것이다.



[1-1] SCSI

SCSI에서 SAN까지

앞서도 언급했듯이, SCSI는 1979년에 슈가트에 의해 정의된 SASI(the Shugart Associates System Interface)가 발전한 것이다. 슈가트와 NCR은 NCR에서 개발한 인터페이스에 기능을 추가하고 SASI에 대한 문서를 개선하기 위해 함께 노력했다. 1982년에 ANSI 작업 그룹이 X3T9.3이 SASI에 근거를 둔 SCSI(Small Computer System Interface)에 대한 제안을 기안해 여러 회사와 사람들의 연구를 거쳐 SCSI는 1986년에 정식으로 ANSI 표준이 되었다. 많은 얼마 후, SCSI에 공통 명령어 셋(Common Command Set)을 통합함과 동시에 다른 기능을 강화한 SCSI-2에 대한 연구가 진행되었고, 1990년 7월에 승인되었다. SCSI-2

가 스토리지 장치와 중소 규모의 컴퓨팅 장치의 산업 표준이 되었음에도 불구하고, 모든 사람이 전통적인 SCSI를 좋다고 생각하는 것은 아니다. 이것은 구리선 기반의 병렬 SCSI 케이블이 갖는 물리적 특성 및 전기적 특성 때문이다(이러한 종류의 케이블에 기반을 둔 SCSI 시스템을 현재 병렬 SCSI라고 부르는데, 이것은 SCSI 신호가 병렬로 된 수십 개의 전도체 쌍을 통해 전달되기 때문이다). 1990년 이후로 SCSI가 오랫동안 사용되었음에도 불구하고 병렬 SCSI에는 다음과 같은 제약 사항이 아직도 남아있다.

- 병렬 SCSI는 하나의 버스에 장치를 16개만 접속할 수 있다.
- 병렬 SCSI를 이용해 한 스토리지에 컴퓨팅 장치를 두 대 연결할 수는 있지만 일반적으로 사용하지 않는다.
- 다중 전도체 병렬 SCSI 케이블에 있는 개별 전도체 사이의 혼선 및 외부 소스로부터의 전기적인 간섭 때문에, 병렬 SCSI는 케이블 길이에 제한을 받는다. 이러한 제한은 SCSI-파이버-SCSI 변환을 통해 다소 극복할 수는 있지만, 많은 소프트웨어와 하드웨어 벤더가 이를 지원하지 않는다.
- SCSI 체인에 부착되는 각 장치가 사용가능한 전체 길이를 짧게 만든다.

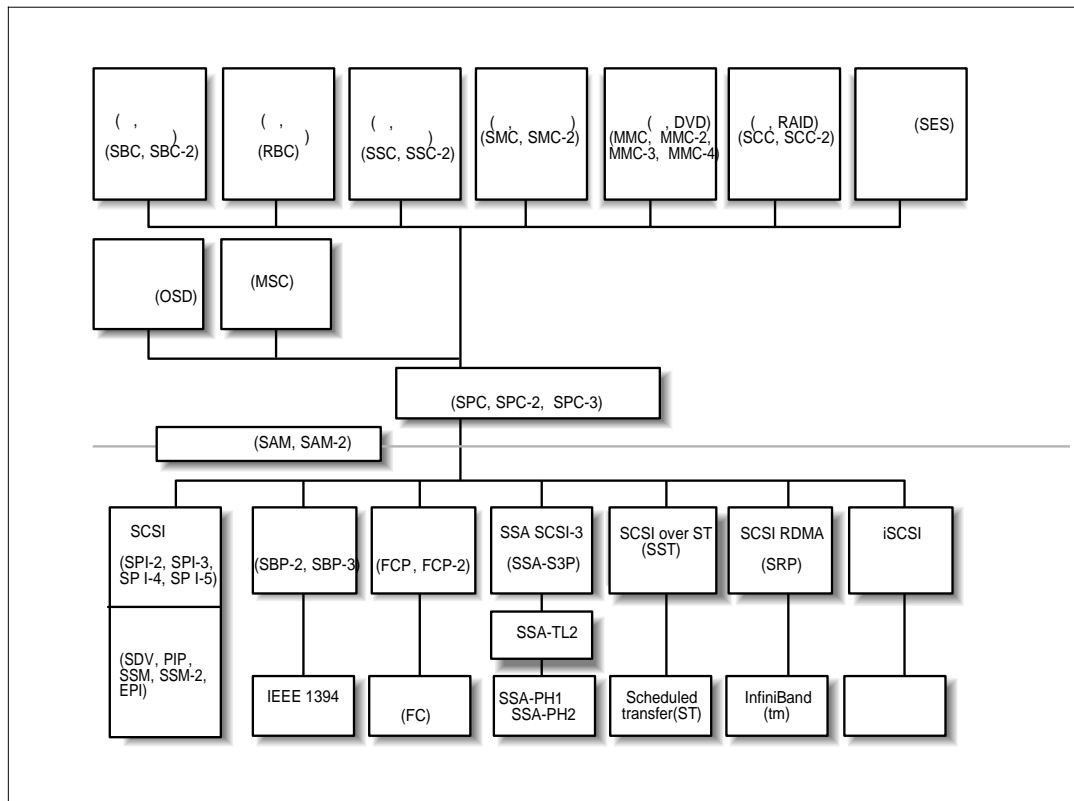
이러한 문제점을 해결하기 위해 물리 계층을 변환할 필요가 있는데, 현 시점에서의 가장 분명한 해답은 파이버이다. 병렬 SCSI와 달리 파이버 케이블은 전송 특성을 변경하지 않고도 수백 미터 떨어진 곳에 전송할 수 있기 때문에, 병렬 SCSI의 전기적인 특성과 관련된 문제를 모두 해결할 수 있다. 그리고 연결 문제도 해결하는데, 이는 루프에 있는 각 장치마다 자체의 전송 레이저가 있어 추가 장치의 전체 버스 길이를 늘릴 수 있기 때문이다.

문제는 병렬 상태의 전도체에서 동작하게 설계된 프로토콜을 하나의 전도체에서 전송하게 하는 방법이다. 우선 필요한 작업은 SCSI의 규격을 다양한 레벨로 나누는 것이다(네트워크 프로토콜 개발에서 얻은 교훈). 각 레벨마다 할당된 기능을 수행하면서 적절한 명령어 셋에 따라 상위나 하위 레벨과 통신한다. 이것이 SCSI-3 규격의 시작인데, 이렇게 다양한 레벨로 분류한 것을 문서 하나로 구성한 SCSI-2 규격을 20개 이상의 문서로

1) 이러한 간략한 SCSI 발달 과정은 X10 위원회의 의장인 존 로마이어(John Lohmeyer)의 공이다.

된 SCSI-3 규격으로 확장시킨다(그림 1-2에 있는 각 블록이 SCSI-3 규격의 문서를 나타낸다).

일단 이와 같이 분류되면 SCSI 상위 레벨에서 SCSI 물리 계층을 분리할 수 있고, 물리 계층이 분리되면 물리 계층에 의한 제약 사항을 확인할 수 있다. 여러분은 [그림 1-2]에 있는 SCSI-3 구조도를 통해 이러한 분리를 확인할 수 있는데, 이것은 SCSI-3 구조 정의를 담당하는 T10 위원회가 제공해 준 것이다. [그림 1-2]는 직렬 SCSI, 파이버 채널, SSA, SCSI 오버 ST, SCSI 오버 VI 등 5가지의 SPI(SCSI Parallel Interface)를 보여준다. 비교적 최근에 이 목록에 추가된 것이 iSCSI인데 이것은 직렬 SCSI 트래픽을 전달하기 위한 전송 계층으로 IP를 사용한다.



[1-2] SCSI-3



SPI 중 가장 보편적인 대안은 파이버 채널 프로토콜(Fibre Channel Protocol)이다. 이것은 SPI와 달리 하나의 경로로 신호를 보내고 다른 하나로 받는 직렬 프로토콜이고, 병렬 SCSI에 비해 다음과 같이 다양한 장점을 제공한다.

장치간의 거리는 더 이상 중요하지 않다. 하나의 파이버 케이블에 한 종류의 신호만 전송하는 단일 모드(Single-mode) 파이버로 10km 떨어진 장치를 연결할 수 있으며, ATM과 같은 교환 기술을 사용하면 이론적으로 거리의 제한을 거의 없앨 수 있다(호스트와 스토리지 사이의 먼 거리로 인한 대기 시간 때문에 이것을 온라인 스토리지 장치에 적용하는 것이 현재는 곤란하다. 그래서 ATM 교환을 사용하는 사람들 대부분 실시간 데이터 액세스가 아니라 원격 복사를 위해 이를 사용하고 있다).

단일 기가비트 파이버 채널 연결의 속도가 현재는 가장 빠른 병렬 SCSI 제품보다 느리지만, 더 넓은 대역폭을 사용하기 위해 다중 파이버 채널 연결을 만들 수 있다. 또한 파이버 채널은 오늘날 설치된 대다수의 SCSI보다 빠르며, 일반적으로 20 혹은 40MB/s에 달한다. 게다가 현재는 2기가비트 파이버 채널도 사용할 수 있다.

병렬 SCSI 카드보다 직렬 SCSI 카드(파이버 채널 호스트 버스 어댑터(HBA))에 연결할 수 있는 장치의 수가 백만 배 이상 많다. 병렬 SCSI는 16개의 장치를 연결할 수 있지만, 파이버 채널 구조는 1천 6백만 개까지 수용할 수 있다.

하나의 스토리지 장치를 1천 6백만 개의 컴퓨터에 간단히 연결할 수 있어 컴퓨터가 디스크나 테이프 같은 자원을 공유할 수 있게 해준다. 유일한 문제점은 컴퓨터 사용자에게 공유 방법을 가르쳐 주어야 한다는 것뿐이다. 현재 대부분의 제품이 단일 구조에서 연결할 수 있는 장치의 수를 제한하고 있다. 그리고 아직은 1천 6백만 개 범위 내에서 몇 개까지 연결할 수 있는지 명확하지 않다.

SAN이란?

여기에서 필자가 생각하는 SAN의 정의는 다음과 같다.

SAN은 파이버 채널이나 iSCSI와 같은 직렬 SCSI 프로토콜을 이용해 통신하는 두 개 이상의 장치이다.

이러한 정의는 SAN이 그다지 대단한 것이 아님을 의미한다. 이 정의에 의하면 스토리지 트래픽만 전송하는 LAN은 SAN이 아니다. LAN(혹은 NAS)과 SAN의 차이는 사용되는 프로토콜에 있다. LAN이 iSCSI 프로토콜을 사용해 스토리지 트래픽을 전송한다면 SAN이라고 할 수 있다. 하지만 전용 LAN을 통해 전통적인 LAN 기반 백업을 보내는 LAN은 SAN이라고 할 수 없다. 이것도 SAN이라고 칭하는 사람도 있지만, 필자는 이것인 혼동된다고 생각해 그렇게 부르지 않는다. 그러한 네트워크는 특별한 목적에 사용되는 전용 LAN 이상이 될 수 없어 그냥 스토리지 LAN 혹은 백업 네트워크라 부른다. 스토리지 LAN은 스토리지 장치 트래픽을 제거하는 데 유용한 툴이고, SAN은 데이터를 전송하기 위해 직렬 SCSI 프로토콜(예를 들어, 파이버 채널이나 iSCSI)을 사용하는 네트워크이다.

SAN은 NAS가 아니다. 앞서 언급한 것처럼, SAN은 SCSI 프로토콜을 사용하고 NAS는 NFS 나 SMB/CIFS 프로토콜을 사용한다(이 장의 후반부에 SAN과 NAS 비교 부분 참조). DAFS (Direct Access File System)는 전송시 파이버 채널도 지원하는 NFS 유사 프로토콜을 통해 파일 공유를 지원함으로써 SAN과 NAS가 밀접한 관계임을 보증한다(DAFS는 부록 A 참조).



NAS

SAN

요약하자면, SAN은 직렬 SCSI 프로토콜(예를 들어 파이버 채널이나 SCSI)을 통해 통신하는 두 개 이상의 장치로, 전통적인 병렬 SCSI보다 다음과 많은 장점이 있다.

- 파이버 채널(그리고 iSCSI)을 몇 개 묶어 하나로 보이는 연결로 구성해, 병렬 SCSI보다 훨씬 빠르게 통신할 수 있다. 단일 파이버 채널 연결도 이 글을 쓰는 시점에는 각 방향으로 2Gb/s 속도로 전송할 수 있어 전체적으로는 4Gb/s로 전송할 수 있다.
- 단일 파이버 채널 SAN에는 1천 6백만 개의 장치를 연결할 수 있다(이론상).
- SAN에 연결된 컴퓨터뿐만 아니라 다른 컴퓨터도 SAN에 연결된 임의의 장치에 쉽게 액세스할 수 있다.

SAN에서 SCSI로, 그리고 파이버 채널을 이용한 SCSI-3와 iSCSI로의 발달을 살펴보았으므로 이제 SAN이 가장 많이 사용되고 성공적으로 사용되는 분야를 알아보겠다. SAN은 백업과 복구 방식을 상당히 바꾸어 놓았다. 필자는 지금은 사용하지 않는 방법과 SCSI를 이용하는 방법을 모두 설명할텐데, 이것은 SAN이 유명해진 이유를 포괄적으로 이해하는 데 도움이 될 것이다.

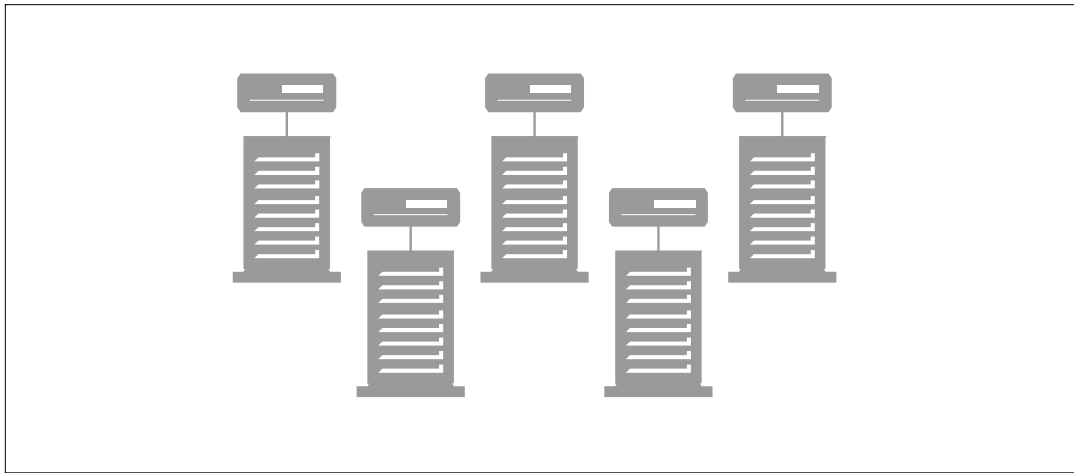
SAN 이전의 백업과 복구

오래 전에는 데이터 센터에 하나의 테이프 드라이브로도 충분한 소형 서버가 있었다. 이러한 데이터 센터는 [그림 1-3]과 같은 모양으로 백업 시스템을 설계하였고, 대부분의 시스템이 자체 테이프 드라이브를 가지고 있었는데 이 테이프 드라이브는 해당 시스템을 백업하기에 충분할 만큼 컸다(사실은 다른 시스템을 백업할 수 있을 정도였다). 완전히 자동화된 백업하기 위해 필요한 것은 몇 가지 셸 스크립트를 작성하고 아침에 테이프를 바꾸는 것뿐이었다.

당시에는 몇 가지 이유 때문에 대역폭이 크게 중요하지 않았다. 그 첫 번째 이유가 백업할 것이 많지 않았다는 것인데, 충돌할 우려가 있는 단일 10Mb 허브만으로 구성된 환경에서도 선로를 통해 전송할 데이터가 많지 않아 문제가 되지 않았다. 그리고 두 번

째 이유는 시스템마다 자체 테이프 드라이브가 있어 LAN으로 전송할 데이터가 많지 않았기 때문이다.

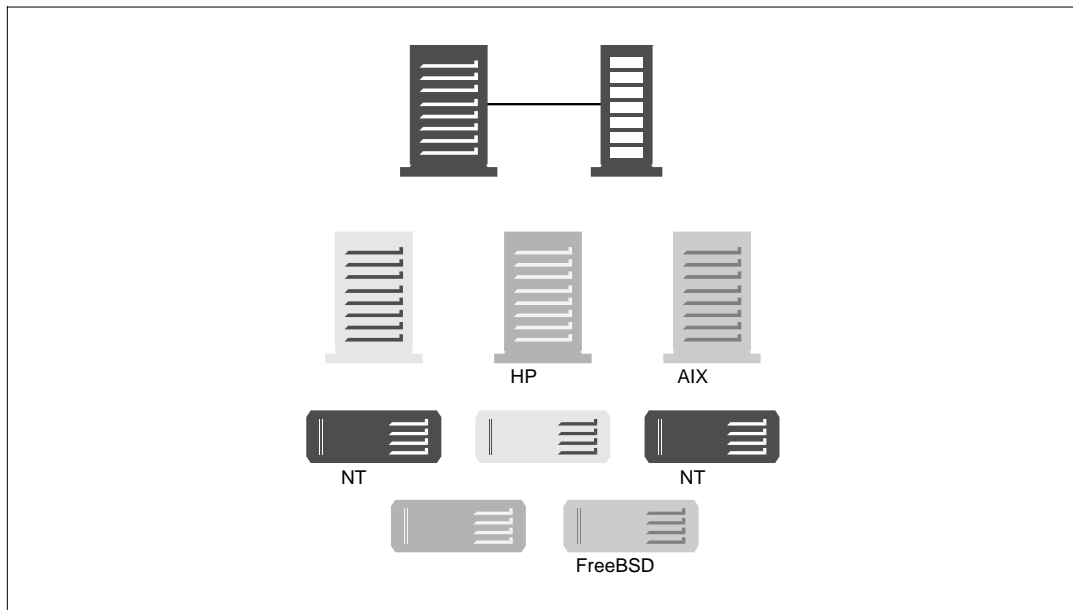
많은 회사와 개별 사용자가 점차 이 시스템의 능력보다 그 이상을 요구하게 되어 수많은 테이프를 바꾸는 작업에 지치고 시스템이 테이프의 용량을 넘어서기 시작했다. 그래서 이들은 더 나은 무언가를 요구하게 되었다.



[1-3]

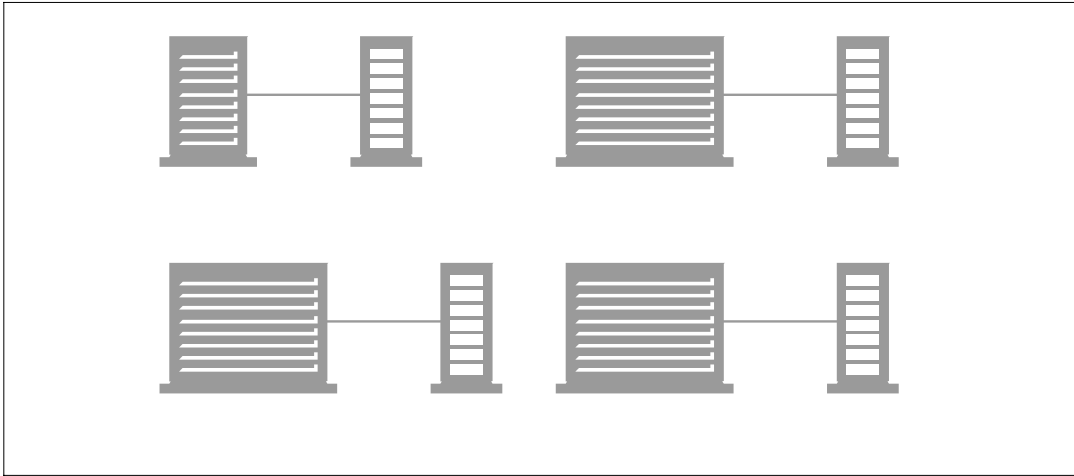
초기의 일부 혁신자들이 중앙 집중형 백업 서버의 개념을 도입하였으며, 이것을 테이프 스택커(stacker)와 연결함으로써 편리성이 증진되었다. 여기서는 5,000 ~ 10,000달러 정도 하는 백업 소프트웨어와 5,000 ~ 10,000달러 정도 하는 하드웨어만 있으면 문제를 해결할 수 있었다. 시스템 사용자 모두가 각 백업 클라이언트에 적절한 소프트웨어를 설치하기만 하면 네트워크를 통해 중앙 백업 서버에 백업할 수 있었다. 이는 심지어 이러한 소프트웨어 패키지가 클라이언트 소프트웨어를 여러 플랫폼에 포팅하면 [그림 1-4]에 있는 모든 시스템이 구동하는 운영체제에 상관없이 백업 서버에 백업할 수 있다는 것을 의미한다.

그런데 다른 문제가 발생했다. 사람들은 클라이언트 소프트웨어 일부만 구매하면 백업 문제가 모두 해결되는 것으로 생각했지만, 시스템이 방대해지고 주어진 네트워크의 시스템의 수가 늘어나면서 그 모든 시스템을 하루 밤새에 네트워크를 통해 백업하는 것이 어려워졌다. 물론, 기가비트 이더넷이 그랬던 것처럼 공유 네트워크부터 스위치 네트워크와 사적 VLAN까지 업그레이드하면 고속 이더넷(Fast Ethernet(100Mb))과 그 뒤를 잇는 이더채널(Etherchannel)류의 기술들(400Mb)이 많은 도움이 되기는 했지만, 일부 시스템은 너무 커 네트워크를 통해 백업하는 것이 불가능했고, 특히 100Gb에서 1Tb에 이르는 레코드와 파일이 있는 대용량의 데이터베이스 서버의 설치는 더 불가능했다.



[1-4]

일부 백업 소프트웨어 회사는 미디어 서버(media server)를 도입해 이 문제를 해결하려고 했다. [그림 1-5]는 중앙 백업 서버가 여전히 모든 백업을 제어하고, 100Mb 혹은 1000Mb 네트워크를 통해 여러 클라이언트를 백업하고 있음을 보여준다. 그러나, 미디어 서버를 지원하는 백업 소프트웨어는 테이프 라이브러리를 각 대용량 데이터베이스 서버에 부착할 수 있기 때문에, 각 서버가 네트워크를 통해 데이터를 전달하지 않고 자체 로컬 테이프 드라이브로 백업할 수 있다.



[1-5] SAN

미디어 서버가 대역폭 문제는 해결했지만 고비용과 비효율성이라는 결과를 초래했다. 각 서버마다 풀 백업을 할 수 있을 만큼 충분히 큰 테이프 라이브러리가 필요했는데, 그 라이브러리는 데이터베이스 서버의 크기에 따라 \$50,000 ~ 500,000달러 이상의 비용이 들었다. 또한 이러한 크기의 많은 서버가 매일 밤 풀 백업을 할 필요가 없기 때문에 비효율적이다. 만약 데이터베이스 소프트웨어가 증분 백업을 수행할 수 있다면, 일주일이나 한 달에 한 번만 풀 백업을 해도 되고, 이는 해당 달 중 나머지 기간 동안 라이브러리에 있는 테이프 드라이브 대부분이 사용되지 않음을 의미한다. 심지어 전통적인 풀 백업을 수행하지 않는 제품도 이러한 문제를 가지고 있다. 이러한 제품은 여러 증분 백업으로부터 적절한 파일을 매우 자주 읽어 테이프 셋에 기록함으로써 가상 풀 백업을 생성하게 된다. 이렇게 풀 백업을 생성하는 방식은 경우에 따라 상당히 많은 테이프 드라이브가 필요하기도 한다.

또 다른 고려 사항은 라이브러리 크기(특히 라이브러리가 포함하는 드라이브의 수)가 백업이 아닌 복구에 의해 달라진다는 것이다. 예를 들어, 한 회사가 600Gb 데이터베이스를 백업해야 한다고 해보자. 그 회사는 테이프 복구가 결코 필요치 않을 것이라 확신할 수 있게끔 모든 수단을 강구하겠지만 실제로 재난에 부딪히면 복구 작업이 필요하다. 복구 요구 사항은 3시간인데, 테이프에서 읽어오는 복구가 필요한데 그 요구 사항이 변하지 않았다면 3시간 안에 복구가 이루어져야 한다. 그래서 그 회사는 그 조건을 맞추려고 150,000달러의 드라이브 라이브러리 10개를 구매하였다. 물론, 데이터베이스를 3시

간 안에 복구할 수 있다면 3시간 안에 백업할 수 있다는 의미이므로, 하루에 약 21시간 정도는 150,000달러의 라이브러리를 사용할 수 없다.

SAN

일부 백업 소프트웨어 벤더는 단일 라이브러리를 다중 호스트와 연결해 비용 문제를 해결하려고 했다. 만약 다중 SCSI 연결 대용량 라이브러리를 구매했다면, 각각을 다른 호스트에 연결해 드라이브는 아니지만 테이프 라이브러리를 공유할 수 있다. 이 기능은 로보틱스²⁾를 공유해 비용을 절감할 수 있었지만 앞서 다룬 비효율성 문제는 완전히 해결하지 못했다.

실제 필요한 것은 드라이브를 공유하는 방식이었다. 테이프 드라이브를 공유하면 디스크 드라이브도 공유할 수 있었다. 만약 다음과 같이 된다면 어떨까?

- 대용량 데이터베이스 서버는 로컬 부착 테이프 드라이브로 백업할 수 있고 그 테이프 드라이브가 로컬 부착 테이프 드라이브로 백업하면 다른 대용량 서버가 사용할 수도 있다면?
- 그 데이터베이스를 사용하는 서버의 CPU를 통해 데이터를 전달하지 않으면서 그 디스크를 백업한 다른 서버가 대용량 데이터베이스 서버의 디스크를 볼 수 있다면?
- 디스크와 테이프 드라이브가 다른 서버의 CPU를 거치지 않고 데이터를 디스크에서 테이프로 직접 전달하는 방식으로 연결된다면?

파이버 채널과 SAN은 이런 가정을 모두 가능하게 해 주었는데 이는 앞으로 나올 장에서 다룰 것이다. SAN은 백업되는 서버의 크기에 상관없이 백업을 훨씬 더 용이하게 하도록 도와준다. 그래서 SAN은 기존의 병렬 SCSI나 LAN 기반 백업만 사용해서는 할 수 없었던 것을 수행하고 있다.

2) 감수자주 : 로보트 팔을 이용하는 자동 라이브러리 장치

NFS와 SMB에서 NAS로

NAS는 서버 메시지 블록(SMB)과 네트워크 파일시스템(NFS)이라는 두 가지 주 영역에 근간을 두고 있다. 매우 흥미롭게도 이 두 파일 공유 프로토콜은 컴퓨터분야에서 경쟁이 가장 심한 회사에서 개발했다. 그 두 경쟁사는 마이크로소프트와 썬인데 둘 다 1984년에 등장했다(IBM과 마이크로소프트사는 SMB를, 썬은 NFS를 개발했다).

SMB/CIFS

윈도우 탐색기로 네트워크 환경으로 가 다른 컴퓨터에 있는 드라이브를 읽고 드라이브에 기록할 수 있다면 SMB 프로토콜을 사용했을 것이다.³⁾ 1984년 IBM 기술 레퍼런스에서 처음으로 언급된 SMB 프로토콜은 원래 네트워크 네이밍과 브라우징 프로토콜을 의미하였다. 그러나 얼마 지나지 않아 마이크로소프트가 이를 파일 공유 프로토콜의 의미로 채택해 사용하였다. SMB 프로토콜 여러 버전이 해를 거치면서 출시되었고, SMB 프로토콜은 이제 모든 마이크로소프트 윈도우 운영체제(윈도우 3.1, 95, 98, Me, NT, 2000, XP)와 IBM OS/2 시스템을 위한 일반적인 파일 공유 프로토콜이 되었다. 그리고 마이크로소프트가 최근에 그 이름을 CIFS(Common Internet File System)로 변경하였다.

다른 마이크로소프트 애플리케이션처럼 CIFS는 단순화를 위해 설계되었다. 다른 사용자가 여러분의 시스템에 있는 드라이브에 액세스하는 것을 허용하려면 드라이브 아이콘 위에서 오른쪽 버튼을 클릭해 나온 메뉴에서 공유를 선택한 후, 읽기 전용이나 쓰기 전용을 선택하고 액세스를 제어하는 패스워드만 정하면 된다. 드라이브(즉, C:\)를 전체를 혹은, 일부만(즉 C:\MYMUSIC) 공유할 수도 있다.

3) 필자는 윈도우 데스크탑에 필요한 NFS 클라이언트를 살 수 있는 경우에만 '아마도'라는 말을 사용한다.

필자는 CIFS가 원래는 성능을 목적으로 설계된 것은 아니라고 했다. 나중에 다시 다룰 테지만, CIFS는 원래 다중 사용자 액세스를 염두해 두고 성능의 문제는 접어둔 채 설계되었다. 하지만 마이크로소프트와 다른 회사가 최근 CIFS 성능을 크게 향상시키고 있다.

CIFS의 인기가 많은 회사들이 대규모, 중앙 집중형 CIFS 서버를 설치해 수백 명 혹은 수천 명의 PC 클라이언트와 드라이브를 공유하게 만들었다. 이렇게 된 가장 일반적인 이유는 중요한 파일의 스토리지 장치를 중앙 집중화할 수 있기 때문이다. 중요한 것은 무엇이든지 네트워크 드라이브에 저장하도록 권장받는데, 많은 사람들이 데스크탑을 백업하지 않기 때문이다. 하지만 운영 스태프들은 CIFS 서버를 백업한다.

CIFS 서버의 다른 유형은 삼바를 구동하는 유닉스/리눅스 박스를 들 수 있는데, 이 이름은 SMB 프로토콜에서 따온 것이다. 이 시스템도 수백 혹은 수천의 PC 클라이언트와 드라이브를 공유할 수 있고, 드라이브를 공유하는 다른 PC와 다르지 않다. 리눅스와 삼바 솔루션은 모두 무료로 사용할 수 있어 상당한 인기를 얻고 있다.

요약해보면, SMB는 CIFS로 발전해 윈도우 기반 데스크탑과 랩탑간의 파일 공유를 위한 주요 방식으로 사용되고 있다. 이 CIFS가 가진 제약 사항은 나중에 자세히 다룰 것이다.

NFS

유닉스 프로젝트처럼 네트워크 파일시스템(NFS)은 썬에 있던 일부 엔지니어들이 소규모 프로젝트로 시작했다가, 1984년에 일반인에게 소개된 후 모든 유닉스 벤더가 기본 애플리케이션에 추가하였다. NFS를 통해 유닉스 박스는 서로 공유한다는 조건만 있으면, 다른 유닉스 박스에 있는 파일시스템도 읽고 쓸 수 있다.

NFS는 여러 방식으로 이용된다. 가장 일반적인 용도는 중앙 NFS 서버가 모든 홈 디렉토리를 포함하게 하는 것이다. 유닉스에서 홈 디렉토리는 사용자가 파일을 저장하는 곳이다. 사실, 적절하게 구성된 파일시스템에서는 사용자가 자신의 홈 디렉토리에 파일을

저장해야 한다(/tmp에 있는 파일처럼 분명히 폐기할 파일은 제외). 그러므로 모든 홈 디렉토리가 NFS 파일서버에 있으면 그 회사의 모든 데이터가 한 곳에 있는 것이 된다. 이러한 장점은 앞에서 다룬 CIFS 서버도 제공한다. 그러므로 PC 시스템은 써드 파티 소프트웨어를 사용해 NFS를 통해 공유된 유닉스 서버에 있는 디렉토리에 액세스할 수 있다.

NFS는 처음부터 단순화를 위해 설계됐을 수도 있지만, 윈도우 파일을 공유하기에는 그리 간단하지 않다.⁴⁾ 유닉스의 각 구현물에는 명령어 라인에서 파일시스템을 공유하는데 사용되는 명령어와 그 파일시스템을 영구적으로 공유할 수 있는 파일이 있다. 예를 들어, BSD 파생물은 보통 /etc/exports와 exportfs 명령어를 사용하지만, 시스템 V 파생물은 /etc/dfs/ dfstab와 share 명령어를 사용한다. 모든 유닉스 명령어를 사용하는 것처럼 파일시스템을 공유하려면 그 명령어의 문법을 익혀야 하는데, 이는 그 문법도 exports 파일에서 사용되기 때문이다. 물론, 명령어의 문법은 유닉스 버전마다 다르다.

NFS가 지닌 또 다른 문제는 원래 TCP가 아니라 UDP를 기반으로 했다는 것이다(NFS v3가 TCP를 사용할 수 있다 하더라도 NFS 트래픽 대부분이 아직도 UDP를 통해 구동된다). UDP와 NFS가 지닌 가장 큰 문제는 재전송을 처리하는 방법과 관련 있다. NFS는 여섯 개의 IP 조각으로 쪼개진 8K 패킷을 전달하는데(1,500바이트의 일반 프레임을 기반으로 한다), 전송 중 한 조각이라도 잃어버리면 여섯 조각을 모두 다시 전송해야 한다. 그리고 UDP도 흐름 제어 장치가 없어, 서버가 받을 수 있는 것보다 더 많은 데이터를 요청해 과도한 UDP 재전송으로 네트워크가 문제를 일으키기 쉽다.

이것이 바로 NFS v3가 TCP에서 NFS를 허용하는 이유이다. TCP는 잘 구축된 흐름 제어 장치를 지니고 있고, 문제가 발생하면 UDP처럼 NFS 패킷 전체를 재전송하는 것이 아니라 잃어버린 패킷만 재전송하면 되기 때문이다. UDP와 TCP 중 어느 것을 사용하느냐에 따라 서버의 관리를 훨씬 더 복잡하게 만들 수도 있다.

4) 유닉스와 관련 있는 다른 어려운 문제에 대해 그랬던 것처럼, 유닉스 운영자들은 그리 신경 쓰지 않는 것 같다. 아마 NFS를 쉽게 스크립트할 수 있기 때문에 NFS가 CIFS보다 구성하기 쉽다고 주장할지도 모른다.

NAS : NFS CIFS

NFS와 CIFS 서버는 나온 후 모두 여러 해 동안 상당한 인기를 누려왔다. 그렇다고 그 중 하나가 완벽하다는 것은 아니다. 실제 세계에서는 NFS와 CIFS 관리와 관련 있는 많은 문제가 제기되고 있다.

NFS와 CIFS의 첫 번째 문제는 서버가 보통 NFS나 CIFS 전용 서버라서 다른 시스템만 표현한다는 것이다. 패치 관리만으로는 어려운 문제가 발생할 수 있다. 또한, 사용자가 많은 환경에서는 NFS나 CIFS를 보장한다는 것이 모든 사용자가 NFS나 CIFS 서버를 관리하는데 필요한 전문 지식을 지니고 있음을 의미하지는 않기 때문에 문제가 발생할 수 있다.

두 번째 문제는 유닉스만 또는 윈도우만을 위한 환경이 거의 없다는 것이다. 즉, 대부분의 환경에 이 두 가지가 혼합되어 있다. 이러한 환경에서는 다음과 같은 문제에 봉착하게 된다. 윈도우 데스크탑에 PC NFS 소프트웨어를 로딩하는 데 추가 비용이 드는가? PC에 CIFS 서비스를 제공하기 위해 유닉스 박스에 있는 삼바를 이용하는가? 만약 그렇다면, NT-savvy 관리자가 삼바를 운영하는 방법을 알고 있을까? 혹은 Unix-savvy 관리자가 윈도우 ACL을 이해할까? 그 답은 그런 환경에서는 보통 유닉스 NFS 서버와 NT CIFS 서버를 같이 지닌다는 것이다. 그래서 지금은 두 개의 관리 박스가 사용되고 있다.

마지막 문제는 성능과 관련 있다. 전형적인 NFS나 CIFS 서버의 뒤에 위치한 디스크는 사용자가 로컬 디스크만큼 빠르게 사용할 수 없다. 그래서 대용량 또는 많은 파일을 저장하는 경우에는 비용이 많이 든다. 그래서 사용자에게는 파일을 로컬에 저장하도록 권장하고 있는데, 이 점이 네트워크 마운트 드라이브를 사용하려는 의도를 포기하게 한다.

NFS와 CIFS의 성능 문제가 제기된 것은 둘 다 추가된 기능이기 때문이다. 유닉스는 60년대 후반에, DOS는 80년대 초반에 개발되었다. 이 두 그룹은 오랜 시간이 지난 뒤에야 파일을 더 쉽게 공유하는 방법이 필요하다는 것을 깨닫고 NFS와 CIFS를 개발했다. 이 프로토콜들이 더 빠르게 동작하게 하기 위해 커널의 일부를 변경해야 하기는 하지만 이들은 CPU와 네트워크 자원에 견줄 수 있는 간단한 애플리케이션이다. 이러한 문제를 모두 해결하기 위해 스크래치하기 시작해 네트워크 접속 스토리지가 탄생하였다.

NAS

조금만 생각해보면, NAS 산업은 임의의 유닉스 또는 윈도우 시스템을 박스 밖에서도 수행할 수 있게 하는 박스 판매에 기본을 두고 있다는 것을 알 수 있다. 그렇다면 NAS 벤더가 그렇게 성공을 거둔 이유는 무엇일까? 그리고 많은 사람들이 앞으로 NAS가 지배적으로 사용될 거라고 예측하는 이유는 무엇일까? 그 이유는 NFS와 CIFS를 사용하면 서 직면한 문제를 이 벤더들이 잘 해결해 주었기 때문이다.

벤더가 NFS나 CIFS 서버를 더 쉽게 관리하려고 시도했다. 그들은 hot-swappable RAID 어레이가 달린 패키지 박스를 만들었는데, 이 어레이가 가용성을 향상시켰고 교정 유지보수에 필요한 시간도 줄여 주었다. 또 다른 참신한 개념은 NFS와 CIFS 서비스를 제공한 단일 서버로서, 그 일을 수행하는 데 필요한 전체 서버의 수를 감소시켜 주었다는 것이다. 게다가 NFS와 CIFS 모두를 통해 같은 디렉토리를 마운트할 수 있게 되었다. NAS 벤더도 NFS와 CIFS 파일시스템을 더 쉽게 공유할 수 있게 하기 위해 사용자 인터페이스를 설계했다. 이렇게 몇 가지 면에서 NAS가 이전 제품보다 관리가 더 쉽다.

NAS 벤더들은 NFS와 CIFS가 지닌 성능 문제도 해결했다. 사실, 일부 NFS는 로컬 디스크보다 더 빠르게 동작하게 하기도 한다.

첫 번째 NAS 벤더인 오스팩스(Auspex)는 일반적인 NFS⁵⁾ 구현물에서는 각 NFS 요청이 호스트의 CPU를 거쳐야 하기 때문에 성능이 떨어진다고 보고, 그 해결책으로서 각 기능별프로세서가 있는 커스텀 박스를 생성했다. 여기서 호스트 프로세서(HP)는 그 시스템을 부팅하는 경우에만 사용되고, 네트워크 프로세서(NP)와 스토리지 장치 프로세서(SP)가 NFS 요청을 담당한다. 그 결과 다른 NFS 서버가 어떤 프로세서보다 훨씬 빨라지게 되었다.

다음으로 주요한 NAS 벤더인 네트워크 어플라이언스(Network Appliance)는 그 문제는 유닉스 커널과 파일시스템과 관련된 것이라 보았다. 그들이 제시한 해결책은 커널을 3.5인치 플로피디스크에 적합한 것으로 축소한 다음, NFS 하부 시스템을 완전히 다시 기록해 더 효율적으로 만드는 것과 NVRAM 사용을 목적으로 설계된 최적의 파일시스템을

5) 초기의 모든 NAS 벤더가 NFS로 시작하였다가 나중에 CIFS 서비스를 추가하였다.

포함하는 것이었다. 이들은 자신의 서버가 로컬 부착 디스크보다 빠르다는 것을 보여주는 벤치마킹 결과를 발표한 첫 번째 NAS 벤더였다.

NAS 파일러는 파일을 전송하기 위해 NFS나 CIFS 프로토콜을 사용하지만 SAN은 SCSI-3 프로토콜을 사용해 장치를 공유한다(다음절에 나오는 SAN과 NAS의 비교 참조).

네트워크 접속 스토리지의 정의를 다음과 같이 정리하면서 이 절을 마무리하겠다.

네트워크 접속 스토리지(NAS)는 NFS, CIFS 또는 DAFS를 통한 파일 공유를 전용으로 하는 컴퓨터(또는 장치)이다.

SAN 대 NAS

[표 1-1]을 통해 NAS와 SAN의 공통점과 차이점을 정리할 수 있을 것이다.

[1-1] SAN NAS

	SAN	NAS
프로토콜	직렬 SCSI-3	NFS/CIFS
공유	로우 디스크와 테이프 드라이브	파일시스템
공유 항목의 예	/dev/rmt/Ocbr /dev/dsk/c0t0d0s2\\.\Tape0	\\filer\C\directory\filename.doc /nfsmount/directory/filename.txt
허용	다른 서버가 동일한 로우 디스크나 테이프 드라이브(일반적으로 최종사용자에 의해 확인되지 않은)에 액세스하는 것을 허용	다른 사용자가 동일한 파일시스템이나 파일에 액세스하는 것을 허용
대체	로컬 디스크와 테이프 드라이브를 SAN으로 대체하면, 수백 개의 시스템이 디스크와 테이프 드라이브를 공유할 수 있다.	네트워크 공유 파일시스템을 제공하는 유닉스 NFS 서버와 NT CIFS 서버를 대체한다.

여러분은 이 책에서 SAN과 NAS가 백업이나 복구와 어떤 연관이 있는지를 알 수 있을 것이다. SAN은 여러분이 현재 지닌 백업과 복구 시스템의 가치를 높이는 탁월한 방법이고, 다른 방법보다 시스템을 쉽게 백업할 수 있게 도와준다. 필자는 NAS 어플라이언스와 그 어플라이언스가 백업과 복구 시스템에서 야기할 수 있는 문제를 이야기하고자 한다. 그리고 상용의 백업과 복구 제품을 구매할 수 없는 독자를 위해 그 박스뿐만 아니라 NDMP(NAS 파일러의 백업을 지원하는 방식) 관련 사항도 상세히 다룰 것이다.

나에게 적합한 기술은?

이 책에서 SAN과 NAS를 다루는 이유 중 한 가지는 많은 사람들이 SAN 대신 파일러를 사용할 수 있다고 보기 시작했기 때문이다. 전에는 파일러를 "NFS in a box"로 간주했지만 이제 대용량 데이터베이스와 중요한 제품 데이터를 호스팅하는 데 사용한다. 그러면 어느 것이 사용자에게 적합할까? 이 장의 마지막 절에서는 각 구조에 대한 찬성과 반대의 견해를 다루어 이 질문에 대한 답을 여러분이 스스로 찾을 수 있도록 하겠다.



NAS

앞에서 설명한 것처럼, NAS 파일러는 여러 가지 이유로 많은 사람들에게 인기를 누려왔다. 다음에 NAS의 장점 몇 가지를 요약한 것이다.

SAN이 NAS보다 강력하고 파이버 채널보다 TCP/IP 기반에서 구동하는 NFS와 CIFS가

SCSI-3보다 클라이언트에 더 많은 오버헤드를 요구한다는 것은 단일 호스트가 NAS보다 SAN 기반 디스크에서 작업을 더 많이 할 수 있다는 것을 의미한다. 그러나 대부분의 애플리케이션에서는 파일러가 사용하는 최대 처리량보다 적은 양이 필요하다.

NAS 백업은 어떨까요?

필자는 한때 NAS가 단점이 더 많다고 있다고 생각했다. 왜냐하면 시스템 관리 요구 사항은 상당히 감소한 반면, 백업과 복구라는 특정 영역의 문제가 발생했기 때문이다(이 문제는 백업 전문가라도 처리하기 힘든 것이었다). 대부분의 NAS 파일러는 대개 운영체제의 스트립 다운(또는 상당히 사용자화 된) 버전과 관련이 깊어 일반적인 백업과 복구 소프트웨어는 응용하기 어려웠다. 일부 예외적인 경우를 제외하고, 파일러에 필요한 클라이언트 소프트웨어를 백업 벤더로부터는 살 수 없었다. 이 문제는 점점 해결되고 있지만, 사용자가 NAS 어플라이언스를 백업하는 유일한 방법은 `rdump`를 사용하거나 NFS 마운트를 통해 백업하는 것밖에 없던 때가 있었다.

네트워크 데이터 관리 프로토콜(NDMP, 뒷장에서 자세히 다룸)이 도래된 후에도 그 문제는 처음에는 별로 해결되는 것 같지 않았다. 이 프로토콜이 하는 일은 일반적으로 테이프 드라이브를 파일러에 로컬 장착해서 그 서버의 데이터를 테이프 드라이브로 백업하는 것이었다. 이것으로 자동화가 상당히 축소되기는 했지만, 소프트웨어 벤더가 NDMP 지원을 더디게 하는 것에 영향을 주지는 않았다. 왜냐하면 벤더들은 NDMP를 자신의 클라이언트 소프트웨어와 경쟁 관계로 보았기 때문이다.

그러나 최근에 많은 변화가 나타났다. 주요 백업 소프트웨어 벤더가 NDMP를 모두 지원해 파일러와 다른 백업 서버간에 테이프 라이브러리를 공유하는 데 SAN 기술을 사용할 수 있게 되었다. 네트워크를 통해 파일러를 백업하는 경우라도 호스트 CPU로부터 TCP/IP 처리를 오프로드하는 기가비트 NIC는 데이터를 더 빠르고 쉽게 전송할 수 있게 했다. 또한 점보 프레임도 일부 벤더에게 도움이 되었다.

파일러의 백업과 복구가 더 이상 어렵지 않게 된 또 다른 이유는 일부 NAS 벤더가 유닉스나 NT 시스템에서 사용가능한 많은 옵션에 상응하는 데이터 보호 옵션(때로는 더 사용하기 간단한)을 도입했다(예를 들어, 내장 스냅샷, 미러링, 복제). 이러한 이유로 그 만큼의 가치를 인정받을 수 있는 NAS를 필자는 최근에 더 존경하게 되었다.

NAS

SAN이 지닌 단점 한 가지는 대부분의 애플리케이션이 장치에 다중 호스트 액세스를 제공하면서 동시에 파일에 다중 호스트 액세스를 지원해야 한다는 것이다. 만약

SAN으로 연결된 시스템이 같은 파일을 읽고 기록하려고 한다면, SAN이나 클러스터 기반 파일시스템이 필요하다. 그런데 이런 파일시스템은 대부분 고가이고 비교적 신 기술을 사용한다. 반면, 파일러는 1984년 이후로 사용하는 기술로 파일을 다중 호스트 액세스하는 것을 제공한다.

NAS

파이버 채널과 패브릭 기반 SAN을 정확히 이해하지 못하는 사람도 있다. 그래서 SAN을 사용하기 위해서는 상당한 학습이 필요하다. 하지만 NAS는 파일러를 구현하기 위해서는 NAS 벤더가 제공한 비교적 간단한 매뉴얼만 읽어보면 된다. 그리고 그 매뉴얼의 설명도 그리 장황하지 않다. 파이버 채널의 경우에는 파이버 채널에 관한 것을 우선 읽고, 그 다음에 HBA 매뉴얼, 스위치 매뉴얼 그리고 SAN 관리 소프트웨어 등 이것과 함께 첨부되는 매뉴얼을 읽어야 이해하기 쉽다.

가

SAN과 NAS를 모두 관리하는 사람이라면 이 말에 모두 동의할 것이다. SAN은 여러 벤더에서 가져온 하드웨어 조각으로 구성되고 HBA, 스위치나 허브, 디스크 어레이를 포함하고 있기 때문에, 사용해 보지 않은 사용자에게는 낯설 것이다. 이와 달리, 파일러는 사용자가 지니고 있는 네트워크 인프라 구조를 그대로 사용한다. 유일하게 필요한 새로운 벤더는 파일러 자체를 만든 제조업자뿐이다. SAN은 폐일 가능성이 있는 많은 구성요소가 있는데 이러한 문제를 해결할 수 있는 툴이 거의 없고, 핑거 포인팅의 가능성이 더 많다. 그러므로 NAS 기반 네트워크의 유지보수가 더 쉽다.

가

파일러는 현재 사용 중인 네트워크 인프라 구조를 효율적으로 사용할 수 있어 SAN을 구현하는 것보다 비용이 더 저렴하다. SAN을 사용하는 경우에는 SAN과 연결된 각 호스트를 지원하는 파이버 채널 HBA, 각 호스트를 지원하는 허브나 스위치에 있는 포트, 하나 이상의 디스크 어레이, 그리고 이 모두를 연결할 수 있는 적절한 케이블을 구입해야 한다. 심지어 NAS 트래픽을 위한 별도의 LAN을 설치하는 경우에도 필요한 구성요소를 사는 비용이 SAN보다 싸다.

가

모든 NAS 벤더가 이 옵션을 제공하는 것은 아니지만, 일부 파일러는 파일시스템을 자동으로 복제해 다른 위치에 있는 파일러로 옮길 수 있다. 그리고 아주 낮은 대역폭의 네트워크 연결로도 가능하다. 물론, 이 작업은 몇몇 씨드 파티 패키지 중 하나를 구입해 SAN으로 할 수도 있지만 어떤 파일러에는 이 기능이 내장되어 있어서 훨씬 저렴하고 더 신뢰할 만 하다.

많은 사람들은 SAN이 과대 선전한 것을 비판하고 있다. SAN 벤더들이 내놓은 너무 많은 시스템이 서로 호환이 되지 않고, 지금도 너무 많은 소프트웨어가 쏟아져 나오는 실정이다. 그래서 SAN 벤더들은 파이버 채널 표준을 놓고 고민하고 있다. 오늘날 성공적으로 구현된 SAN이 많지만 그렇지 않는 것도 많다. 만약 문제있는 벤더로부터 구입한 장비를 연결하려고 하면 제대로 동작하지 않을 수도 있다. 이와 달리, 파일러는 완전한 상호 운영이 가능하고, 기반으로 둔 표준은 이미 오래 전부터 사용하던 것이다.

파일러도 단점이 없는 것은 아니다. 다음은 이 책을 쓰는 시점까지 발견한 단점이다. 이 책이 출간될 때도 있을지는 모르겠지만 독자에게 도움을 주고자 여기에 제시한다.

일부 NAS 벤더가 제공한 스냅샷과 오프 사이트 복제 소프트웨어가 SAN을 사용해서는 얻기 힘든 훌륭한 복구 가능성을 제공하기는 하지만, 파일러는 어떤 시점에서 반드시 테이프를 백업해야 하는데 이 백업한 것이 변경될 수도 있다. 그 이유 중 하나는 테이프를 전체 백업하는 것이 다른 애플리케이션보다도 I/O 시스템에 부하를 더 많이 줄 수 있기 때문이다. 즉, 매우 큰 용량의 파일러를 테이프를 백업하는 것은 시스템에 상당한 부하를 생성시킨다. 많은 파일러가 백업과 복구 속도를 상당히 개선하기는 했지만, 로우 데이터를 테이프를 처리하는 경우에는 SAN으로 처리하는 것이 아직은 더 빠르다.

NAS가

지금까지의 파일러에 대한 백업과 복구 옵션은 파일 기반인데, 이것은 백업과 복구 소프트웨어가 여러분이 하고자 하는 대로 파일시스템을 넘나든다는 것을 의미한다. 수 백만 개의 소규모 파일을 생성하는 애플리케이션도 있지만, 수 백만 개의 소규모 파일을 복구하는 것은 백업과 복구 시스템이 수행할 일 중 가장 어려운 임무가 될 것이다. 실제로 데이터를 복구하는 것보다 i노드를 생성하는 데 시간이 더 많이 소요되는데, 이것이 대부분의 주요 백업/복구 소프트웨어 벤더들이 파일 레벨의 복구 가능성을 유지하면서 로우 디바이스를 통해 파일시스템을 복구할 수 있는 소프트웨어를 개발한 이유이다. 불행히도, 오늘날의 파일러는 아직 이 문제를 해결하지 못하고 있다.

SAN

대부분의 애플리케이션이 데이터 전송 능력 이상의 부담을 파일러에 부여하지 않을 것이라는 것은 논쟁의 여지가 있지만, 이론적으로 SAN이 NAS보다 많은 데이터를 전송할 수 있다는 것은 중요하다. 만약 여러분의 애플리케이션이 많은 양의 일을 처리해야 한다면, 둘 다를 벤치마킹해야 한다. 어떤 환경에서는 NAS가 SAN보다 더 빠르고 저렴한 대안을 제공할 수 있지만 다른 환경에서는 SAN이 유일한 옵션이 될 수 있기 때문이다. 그러므로 구입하기 전에 시스템 사양을 반드시 테스트해야 한다.

SAN

많은 사람들이 SAN을 사용하기로 하고 NAS는 거들떠보지도 않기도 한다. 그들은 SAN이 고가이고 최신식 기술을 대변한다는 것을 알고 있다. 그러나 SAN만이 제공하는 이점을 누리기 위해 그것이 지닌 단점은 기꺼이 감수해야 하는데 우선 SAN의 장점은 다음과 같다.

SAN

NFS와 CIFS 모두 네트워크를 통해 로우 디바이스를 이용할 수 없고 파일만 이용할 수 있다. 그러므로 애플리케이션이 로우 디바이스에 액세스해야 한다면 NAS로는 불가능하다.

SAN

복잡하다고 보는 것을 유연성과 관련짓기는 사람도 있다. 그들은 구매한 파일시스템이나 볼륨 매니저를 가지고 기능을 활용하기도 하는데, 이는 NAS에서는 사용할 수 없는 기능이다. 특히 `ufs`, `NTFS`, `vxfs`로 구성된 경우에는 더 그렇다.

SAN

위에서 언급한 것처럼, SAN에서 더 빠르게 동작할 수 있는 애플리케이션이 있다. 만약 애플리케이션이 가장 고속의 파일러에서 사용가능한 것보다 더 많은 작업량을 소화하기 원한다면, 여러분이 사용할 수 있는 유일한 대안은 SAN뿐이다.

SAN

SAN을 사용하면 대규모 백업과 복구를 훨씬 더 쉽게 처리할 수 있다. 그래서 대규모 NAS 환경에서도 테이프 라이브러리를 공유하고 LAN 기반 백업을 수행하기 위해 SAN 기술을 활용한다.

다음 목록은 SAN 기술을 사용하면서 직면하는 SAN의 단점이다.

SAN

이 문제는 앞서서도 이미 다루었다. 아마 몇 년 후에는 벤더들이 적절한 표준에 관해 논쟁할 것이고, SAN 관리 소프트웨어는 완벽하게 상호운영이 가능한 SAN 장비를 사용해야 가능한 일을 모두 처리할 수 있을 것이다. 필자는 그 날이 오기를 학수고대하는 바이다.

SAN

파이버 채널, 중재 루프, 패브릭 로그인, 장치 가상화의 개념을 이해하기가 쉬운 것

은 아니다. 반면, NFS와 CIFS의 개념은 이보다 훨씬 간단하다.

SAN 가

SAN 비용이 점점 감소하고는 있지만, 파이버 채널 HBA는 표준 이더넷 NIC보다 비용이 더 많이 든다. 즉, 규모의 경제성 문제이다. 대부분의 사람들은 파이버 채널보다 이더넷을 필요로 한다.

사용자가 어떤 스토리지 구조를 선택할 것인가는 사용자의 환경에 달려 있다. 그러면 비용, 복잡성, 유연성, 로우 작업량 중 어느 것이 더 중요할까? 여러분의 스토리지 요구 사항에 대해 NAS와 SAN 벤더와 이야기해보고, 비용이 얼마만큼 드는지 비교해 보아야 할 것이다. 만약 SAN의 비용과 복잡성이 그리 심각하게 문제되지 않는다면 필자는 둘 다 벤치마킹해보라고 권하고 싶다.

벤치마킹하는 동안에는 각 벤더의 도움을 꼭 받아야 한다. 각 시스템의 적절한 구성이 적절한 성능과 직결되기 때문이다. 아마 원하는 것을 한 번만에 구하지 못할 것이다. 비용이 들더라도 벤더에게 테스트용 NAS나 SAN을 설치하게 하자. 지불한 돈만큼의 충분한 대가를 받을 것이고, 특히 한 번도 구성해 본적이 없다면 더욱 가치가 있을 것이다.

이 책의 나머지 장도 여러분에게 많은 도움이 되기를 바라며 다음 장으로 넘어가 보자.