

Oracle® Hierarchical Storage Manager and StorageTek QFS Software

파일 시스템 복구 설명서

릴리스 6.0

E56780-02

2015년 3월

Oracle® Hierarchical Storage Manager and StorageTek QFS Software

파일 시스템 복구 설명서

E56780-02

Copyright © 2011, 2015, Oracle and/or its affiliates. All rights reserved.

본 소프트웨어와 관련 문서는 사용 제한 및 기밀 유지 규정을 포함하는 라이선스 계약서에 의거해 제공되며, 지적 재산법에 의해 보호됩니다. 라이선스 계약서 상에 명시적으로 허용되어 있는 경우나 법규에 의해 허용된 경우를 제외하고, 어떠한 부분도 복사, 재생, 번역, 방송, 수정, 라이선스, 전송, 배포, 진열, 실행, 발행, 또는 전시될 수 없습니다. 본 소프트웨어를 리버스 엔지니어링, 디스어셈블리 또는 디컴파일하는 것은 상호 운용에 대한 법규에 의해 명시된 경우를 제외하고는 금지되어 있습니다.

이 안의 내용은 사전 공지 없이 변경될 수 있으며 오류가 존재하지 않음을 보증하지 않습니다. 만일 오류를 발견하면 서면으로 통지해 주시기 바랍니다.

만일 본 소프트웨어나 관련 문서를 미국 정부나 또는 미국 정부를 대신하여 라이선스한 개인이나 법인에게 배송하는 경우, 다음 공지 사항이 적용됩니다.

U.S. GOVERNMENT END USERS: Oracle programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, delivered to U.S. Government end users are "commercial computer software" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, shall be subject to license terms and license restrictions applicable to the programs. No other rights are granted to the U.S. Government.

본 소프트웨어 혹은 하드웨어는 다양한 정보 관리 애플리케이션의 일반적인 사용을 목적으로 개발되었습니다. 본 소프트웨어 혹은 하드웨어는 개인적인 상해를 초래할 수 있는 애플리케이션을 포함한 본질적으로 위험한 애플리케이션에서 사용할 목적으로 개발되거나 그 용도로 사용될 수 없습니다. 만일 본 소프트웨어 혹은 하드웨어를 위험한 애플리케이션에서 사용할 경우, 라이선스 사용자는 해당 애플리케이션의 안전한 사용을 위해 모든 적절한 비상-안전, 백업, 대비 및 기타 조치를 반드시 취해야 합니다. Oracle Corporation과 그 자회사는 본 소프트웨어 혹은 하드웨어를 위험한 애플리케이션에서의 사용으로 인해 발생하는 어떠한 손해에 대해서도 책임지지 않습니다.

Oracle과 Java는 Oracle Corporation 및/또는 그 자회사의 등록 상표입니다. 기타의 명칭들은 각 해당 명칭을 소유한 회사의 상표일 수 있습니다.

Intel 및 Intel Xeon은 Intel Corporation의 상표 내지는 등록 상표입니다. SPARC 상표 일체는 라이선스에 의거하여 사용되며 SPARC International, Inc.의 상표 내지는 등록 상표입니다. AMD, Opteron, AMD 로고, 및 AMD Opteron 로고는 Advanced Micro Devices의 상표 내지는 등록 상표입니다. UNIX는 The Open Group의 등록상표입니다.

본 소프트웨어 혹은 하드웨어와 관련문서(설명서)는 제3자로부터 제공되는 콘텐츠, 제품 및 서비스에 접속할 수 있거나 정보를 제공합니다. 사용자와 오라클 간의 합의서에 별도로 규정되어 있지 않는 한 Oracle Corporation과 그 자회사는 제3자의 콘텐츠, 제품 및 서비스와 관련하여 어떠한 책임도 지지 않으며 명시적으로 모든 보증에 대해서도 책임을 지지 않습니다. Oracle Corporation과 그 자회사는 제3자의 콘텐츠, 제품 및 서비스에 접속하거나 사용으로 인해 초래되는 어떠한 손실, 비용 또는 손해에 대해 어떠한 책임도 지지 않습니다. 단, 사용자와 오라클 간의 합의서에 규정되어 있는 경우는 예외입니다.

차례

머리말	5
설명서 접근성	5
이 문서 사용을 위한 필요 조건	5
규약	5
사용 가능한 설명서	6
1. 소개	7
1.1. 오류 및 복구 시나리오	7
1.2. 권장 준비 사항	8
2. 상황 안정화	11
2.1. 아카이브 및 재활용 프로세스 중지	11
2.1.1. 아카이브 중지	11
2.1.2. 재활용 중지	13
2.2. 아카이브되지 않은 데이터 보존	14
2.2.1. 아카이브되지 않은 파일 백업	14
2.3. 구성 및 상태 정보 보존	15
2.3.1. Oracle HSM 구성 저장	15
3. Oracle HSM 구성 복원	19
3.1. 백업 복사본 및/또는 SAMreport에서 구성 복원	19
3.1.1. 백업 파일 또는 SAMreport에서 구성 복원	19
3.2. 백업 정보 없이 구성 복원	23
4. 파일 시스템 복구	25
4.1. 파일 시스템 다시 만들기	25
4.1.1. 백업 구성 파일 및 sammkfs 명령을 사용하여 파일 시스템 다시 만들기	25
4.2. 디렉토리 및 파일 복원	29
4.2.1. samfsdump(qfsdump) 복구 지점 파일에서 파일 및 디렉토리 복원	29
4.2.1.1. 복구 지점 파일에서 손실된 파일 시스템 복원	29
4.2.1.2. 아카이브된 파일 다시 스테이징(필요한 경우)	31
4.2.1.3. 손상된 파일 식별 및 교체 복사본 찾기	31
4.2.1.4. 복구 지점이 생성된 후 아카이브된 누락 파일 찾기	33

4.2.1.5. 손상 및/또는 누락된 파일 복원	36
4.2.2. 복구 지점 파일 없이 아카이브 매체에서 파일 및 디렉토리 복원	36
5. 손실 및 손상된 파일 복구	41
5.1. 복구 지점 파일을 사용하여 파일 복구	41
5.2. 아카이버 로그 항목을 사용하여 파일 복구	43
5.2.1. 손실 및 손상된 정규 파일 복원	43
5.2.2. 손실 및 손상된 세그먼트화된 파일 복원	46
5.2.3. 손실 및 손상된 볼륨 오버플로우 파일 복원	51
5.3. 손상된 아카이브 복사본 복구	53
6. 마무리	57
6.1. 아카이브 파일 시스템을 일반 작업으로 복원	57
6.1.1. 아카이브를 사용으로 설정	57
6.1.2. 재활용을 사용으로 설정	58
6.2. 새 구성 정보 보존	60
6.2.1. 새로 복원된 Oracle HSM 구성 저장	60
A. 아카이버 로그 이해	63
B. 장비 유형 용어집	65
B.1. 권장 장비 및 매체 유형	65
B.2. 기타 장비 및 매체 유형	66
C. 제품 접근성 기능	69
용어집	71
색인	83

머리말

이 문서에서는 Oracle Hierarchical Storage Manager(이전의 StorageTek Storage Archive Manager)를 사용하여 손상되거나 손실된 파일, 파일 시스템 및 아카이빙 솔루션을 복구하는 시스템 관리자, 스토리지/네트워크 관리자 및 서비스 엔지니어의 요구를 다룹니다.

설명서 접근성

오라클의 접근성 개선 노력에 대한 자세한 내용은 <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc>에서 Oracle Accessibility Program 웹 사이트를 방문하십시오.

오라클 고객지원센터 액세스

지원 서비스를 구매한 오라클 고객은 My Oracle Support를 통해 온라인 지원에 액세스할 수 있습니다. 자세한 내용은 <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info>를 참조하거나, 청각 장애가 있는 경우 <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs>를 방문하십시오.

이 문서 사용을 위한 필요 조건

이 문서에서는 사용자가 이미 Oracle Solaris 운영 체제, 스토리지 및 네트워크 관리에 익숙하다고 간주합니다. 관련 작업, 명령 및 절차에 대한 자세한 내용은 Solaris 설명서와 매뉴얼 페이지 및 스토리지 하드웨어 설명서를 참조하십시오.

규약

이 문서에서는 다음 텍스트 규약이 사용되었습니다.

- 기울임꼴 유형은 책 제목 및 강조를 나타냅니다.
- 고정폭 유형은 터미널 창에 표시되는 명령 및 텍스트와 구성 파일, 셸 스크립트 및 소스 코드 파일의 내용을 나타냅니다.
- 고정폭 굵게 유형은 사용자 입력과 명령줄 출력, 터미널 표시 또는 파일 내용에 대한 중요 변경 사항을 나타냅니다. 파일 또는 표시의 관련 부분을 특히 강조하는 데 사용될 수도 있습니다.
- 고정폭 굵게 기울임꼴 유형은 터미널 표시 또는 파일에서 변수 입력 및 출력을 나타냅니다.
- 고정폭 기울임꼴 유형은 터미널 표시 또는 파일에서 기타 변수를 나타냅니다.
- ... (3개의 점으로 구성된 말줄임표)는 예와 관련이 없어 간결성 또는 명확성을 위해 생략된 파일 내용이나 명령을 나타냅니다.
- 예에서 라인 끝에 있는 /(백슬래시)는 다음 라인이 동일 명령의 일부가 되도록 줄바꿈을 이스케이프합니다.
- [-] (하이픈으로 구분된 값을 둘러싸는 대괄호)는 값 범위를 구분합니다.
- 명령 구문의 [] (대괄호)는 선택적 매개변수를 나타냅니다.

- `root@solaris:~#` 및 `[hostname]:root@solaris:~#` 는 Solaris 명령 셸 프롬프트를 나타냅니다.
- `[root@linux ~]#` 은 Linux 명령 셸 프롬프트를 나타냅니다.

사용 가능한 설명서

*Oracle Hierarchical Storage Manager and StorageTek QFS Software File System Recovery Guide*에서는 <http://docs.oracle.com/en/storage/#sw>에서 사용 가능하며 여러 권으로 구성된 *Oracle HSM* 고객 설명서 라이브러리의 일부입니다.

Oracle Solaris 운영 체제 설명서는 <http://docs.oracle.com/en/operating-systems/>에서 제공됩니다.

이 문서에서는 하드웨어 실패, 잘못된 구성, 사람의 실수 또는 시설 및 장비의 물리적 파손으로 인해 손실되거나 손상된 Oracle Hierarchical Storage Manager and StorageTek QFS Software, 파일 및 파일 시스템을 복구하기 위해 수행해야 하는 단계를 설명합니다. 제대로 구성된 Oracle HSM 파일 시스템은 매우 안정적입니다. 그러나 복구 중 수행해야 하는 단계 및 성공 가능성은 준비 정도에 따라 달라집니다. 따라서 이 소개는 복구 프로세스의 개요부터 시작합니다. 그런 다음 Oracle이 권장하는 데이터 및 파일 시스템 보호 조치에 대한 검토로 이동합니다. 마지막으로 마련한 준비 및 현재 사용 가능한 리소스에 따라 수행 가능한 복구 단계를 설명합니다.

1.1. 오류 및 복구 시나리오

파일 시스템 실패의 범위 및 필요한 복구 작업의 특성은 기본적인 문제의 특성에 따라 달라집니다. 예를 들면 다음과 같습니다.

- 서버 호스트가 실패할 경우 Oracle HSM 소프트웨어 및 파일 시스템 구성은 손실되더라도 파일 시스템 데이터 및 메타 데이터는 그대로 있지만 구성 정보가 복원될 때까지 액세스할 수 없습니다.

기본적인 하드웨어 문제가 해결되고 운영 체제가 복원되면 소프트웨어를 다시 설치하고 백업 복사본에서 구성 파일을 복원할 수 있습니다. 이 상황에서는 [3장. Oracle HSM 구성 복원 \[19\]](#)에 설명된 절차를 따르십시오.

- 관리자가 부주의로 하나 이상의 구성 파일, 라이브러리 카탈로그, 스크립트 또는 *crontab* 항목을 삭제하거나 손상시킬 경우 하나 이상의 파일 시스템에 대한 액세스가 일부 또는 모든 소프트웨어 기능과 함께 손실될 수 있습니다.

백업 복사본에서 구성 파일을 복원합니다. [3장. Oracle HSM 구성 복원 \[19\]](#)에 설명된 절차를 따르십시오.

- 독립형(비아카이브) QFS 파일 시스템의 데이터에 대한 디스크 캐시를 제공하는 디스크 또는 RAID 그룹에서 오류가 발생할 경우 디스크 캐시에 있는 모든 파일이 손실됩니다.

하드웨어 문제가 해결되면 손실된 파일을 QFS 백업 복사본에서 복원합니다. [5.1절. “복구 지점 파일을 사용하여 파일 복구” \[41\]](#)를 참조하십시오.

- 아카이브 파일 시스템의 데이터에 대한 디스크 캐시를 제공하는 디스크 또는 RAID 그룹에서 오류가 발생할 경우 디스크 캐시에 있는 모든 파일이 손실됩니다.

하드웨어 문제가 해결되면 아카이브 복사본 또는 Oracle HSM 백업 파일에서 파일을 복원합니다. [5.1절. “복구 지점 파일을 사용하여 파일 복구” \[41\]](#) 및 [5.2절. “아카이브 로그 항목을 사용하여 파일 복구” \[43\]](#)를 참조하십시오.

- 파일 시스템 메타 데이터를 저장하는 디스크에서 오류가 발생할 경우 파일 시스템이 손실되고 더 이상 데이터에 손쉽게 액세스할 수 없습니다.

하드웨어 문제가 해결되면 메타 데이터를 백업 파일에서 복원합니다. 아카이브 파일 시스템에 대한 메타 데이터가 백업되지 않은 경우 아카이버 로그 파일의 백업 복사본에서 재구성할 수 있습니다. [5장. 손실 및 손상된 파일 복구 \[41\]](#)를 참조하십시오.

- 관리자가 부주의로 Oracle HSM 파일 시스템을 호스팅하는 디스크 분할 영역을 포맷하거나 기존 Oracle HSM 분할 영역에 대해 `sammkfs` 명령을 실행할 경우 파일과 메타 데이터가 모두 손실됩니다.

백업 파일에서 메타 데이터를 복원하거나 아카이빙 파일 시스템의 아카이버 로그에서 메타 데이터를 재구성합니다. 데이터는 아카이브 매체 또는 백업 파일에서 복원할 수 있습니다. [5장. 손실 및 손상된 파일 복구 \[41\]](#)를 참조하십시오.

1.2. 권장 준비 사항

Oracle Hierarchical Storage Manager and StorageTek QFS 설치 및 구성 설명서에서 Oracle은 초기 구성 중에 다음과 같은 구성, 파일 시스템 및 데이터 백업 단계를 수행할 것을 권장합니다.

- Oracle HSM 아카이빙 파일 시스템에 중요한 데이터를 저장합니다.

파일 데이터 복사본을 2개 이상 아카이브하십시오. 하나 이상의 복사본을 자기 테이프와 같은 이동식 매체에 아카이브합니다.

가능한 경우 아카이브 파일 시스템의 디스크 캐시와 물리적 장치를 공유하지 않는 독립된 파일 시스템에서 디스크 아카이브를 구성합니다.

- 파일 시스템 메타 데이터는 중복성이 높은 미러 스토리지에 저장합니다.
- 복구 지점 파일을 사용하여 Oracle HSM 파일 시스템을 정기적으로 백업합니다.

복구 지점 파일에는 파일 시스템 메타 데이터 및 선택적으로 데이터가 저장되므로 해당 파일 또는 전체 파일 시스템을 복원할 수 있습니다.

Oracle Hierarchical Storage Manager 소프트웨어가 설치된 경우 `samfsdump` 명령을 실행하여 복구 지점 파일을 생성할 수 있습니다. QFS 파일 시스템 소프트웨어만 있는 경우 `qfsdump` 명령을 사용하십시오. `dump` 명령은 명령줄 또는 Oracle HSM Manager 그래픽 사용자 인터페이스에서 실행할 수 있습니다.

둘 중 한 명령을 사용하면 메타 데이터가 백업됩니다. 둘 중 한 명령을 `-u` 옵션과 함께 사용하면 메타 데이터뿐 아니라 데이터도 백업됩니다. `-u` 옵션은 이동식 매체에 아카이브되지 않는 파일 시스템을 보호하는 데 주로 유용합니다.

- Oracle HSM 메타 데이터 복구 지점 파일을 자동으로 저장하도록 호스트를 구성합니다. Solaris `crontab` 파일에 항목을 생성하거나 Oracle HSM Manager의 예약 기능을 사용하십시오.
- Oracle HSM 아카이버 로그 정보를 자동으로 저장하도록 호스트를 구성합니다. Solaris `crontab` 파일을 생성합니다.

Oracle Hierarchical Storage Manager 소프트웨어로 아카이브된 파일마다 파일 시스템에 있는 파일의 이름과 위치(경로), 복사본이 포함된 아카이브(*tar*) 파일의 이름, 아카이브 파일이 포함된 이동식 매체 볼륨 및 매체에 있는 아카이브 파일의 위치가 아카이버 로그 파일에 기록됩니다.

- 구성 파일, *crontab* 항목 및 사용자 정의 파일 시스템 관리 스크립트(있을 경우)의 백업 복사본을 저장합니다.
- Oracle HSM 복구 정보의 안전한 스토리지 위치를 선택합니다.

Oracle HSM 파일 시스템 호스트에 마운트할 수 있는 독립된 파일 시스템을 선택합니다.

선택한 파일 시스템이 물리적 장치, 논리 볼륨, 분할 영역 또는 LUN을 아카이빙 파일 시스템과 공유하지 않도록 합니다. 보호해야 하는 파일 시스템에 재해 복구 리소스를 저장하지 않습니다.

상황 안정화

중요한 파일 시스템 실패 또는 잠재적 데이터 손실로부터 복구해야 할 때마다 첫번째 단계에서는 영향을 받는 시스템을 안정화하고 추가 손실에 대한 가능성을 최소화하며 가능한 경우 진단 정보를 보존해야 합니다. 이 장에서는 이를 위해 수행할 작업에 대해 설명합니다.

- [아카이브 및 재활용 프로세스 중지 \[11\]](#)(있는 경우)
- [아카이브되지 않은 데이터 보존 \[14\]](#)
- [구성 및 상태 정보 보존 \[15\]](#).

2.1. 아카이브 및 재활용 프로세스 중지

아카이빙 파일 시스템이나 많은 수의 손실된 파일을 복원해야 할 경우 먼저 파일 시스템의 아카이브 및 재활용 프로세스를 중지해야 합니다. 상황을 해결하고 이상적으로는 모든 것을 정상으로 복원할 때까지 아카이브를 안정화하고 격리시키기를 원할 것입니다. 그렇지 않으면 지속되는 아카이브 및 재활용 작업이 상황에 따라 악화될 수 있습니다. 아카이브 및 스테이징 프로세스를 통해 손상된 파일이 전파될 수 있습니다. 재활용 프로세스 중에 유효한 데이터의 나머지 복사본만 삭제될 수도 있습니다.

따라서 가능하면 언제나 아래에 나열된 예방 조치를 수행하십시오.

- [아카이브 중지 \[11\]](#)
- [재활용 중지 \[13\]](#).

복구 작업이 완료되면 아래의 변경 작업을 되돌리고 일반적인 파일 시스템 동작을 복원할 수 있습니다.

2.1.1. 아카이브 중지

1. 파일 시스템 메타 데이터 서버에 *root*로 로그인합니다.

```
root@solaris:~#
```

2. 텍스트 편집기에서 */etc/opt/SUNWsamfs/archiver.cmd* 파일을 열고 첫번째 *fs(file-system)* 지시어로 스크롤합니다.

이 예에서는 *vi* 편집기를 사용합니다.

```
root@solaris:~# vi /etc/opt/SUNWsamfs/archiver.cmd
```

```
# Configuration file for Oracle HSM archiving file systems
#-----
# General Directives
archivemeta = off
examine = noscan
#-----
# Archive Set Assignments
fs = samqfs1
logfile = /var/adm/samqfs1.archive.log
all .
    1 -norelease 15m
    2 -norelease 15m
fs = samqfs2
logfile = /var/adm/samqfs2.archive.log
all .
...
```

3. 모든 파일 시스템에서 아카이브를 중지하려면 *archiver.cmd*에서 첫번째 *fs* 지시어 바로 앞에 *wait* 지시어를 삽입합니다. *archiver.cmd* 파일을 저장하고 편집기를 닫습니다.

이 예에서는 *wait* 지시어를 *samqfs1* 파일 시스템에 대한 지시어 바로 앞에 삽입하여 아카이브를 위해 구성된 모든 파일 시스템에 적용되도록 합니다.

```
root@solaris:~# vi /etc/opt/SUNWsamfs/archiver.cmd
...
#-----
# Archive Set Assignments
wait
fs = samqfs1
logfile = /var/adm/samqfs1.archive.log
all .
    1 -norelease 15m
    2 -norelease 15m
    3 -norelease 15m
fs = samqfs2
...
:wq
root@solaris:~#
```

4. 파일 시스템 하나에서만 아카이브를 중지하려면 해당 파일 시스템의 *fs* 지시어 바로 뒤에 *wait* 지시어를 삽입합니다. *archiver.cmd* 파일을 저장하고 편집기를 닫습니다.

이 예에서는 *samqfs1* 파일 시스템에서 아카이브 작업을 중지합니다.

```
root@solaris:~# vi /etc/opt/SUNWsamfs/archiver.cmd
```

```

...
#-----
# Archive Set Assignments
fs = samqfs1
wait
logfile = /var/adm/samqfs1.archive.log
all .
    1 -norelease 15m
    2 -norelease 15m
    3 -norelease 15m
fs = samqfs2
...
:wq
root@solaris:~#

```

5. [재활용 중지 \[13\]](#)를 수행합니다.

2.1.2. 재활용 중지

1. 파일 시스템 메타 데이터 서버에 *root*로 로그인합니다.

```
root@solaris:~#
```

2. 텍스트 편집기에서 */etc/opt/SUNWsamfs/recycler.cmd* 파일을 엽니다.

이 예에서는 *vi* 편집기를 사용합니다.

```

root@solaris:~# vi /etc/opt/SUNWsamfs/recycler.cmd
# Configuration file for Oracle HSM archiving file systems
#-----
logfile = /var/adm/recycler.log
no_recycle tp VOL[0-9][2-9][0-9]
library1 -hwm 95 -mingain 60

```

3. *-ignore* 매개변수를 *recycler.cmd* 파일의 각 재활용 지시어에 추가합니다. 그런 다음 파일을 저장하고 편집기를 닫습니다.

아카이브 세트가 아니라 라이브러리에 의해 재활용을 구성하지 않으면 *recycler.cmd* 파일에 재활용 지시어가 포함되지 않습니다. 하지만 지금 확인해 보십시오.

이 예에서는 테이프 라이브러리 *library1*에 대해 재활용 지시어가 한 개 있습니다.

```

root@solaris:~# vi /etc/opt/SUNWsamfs/recycler.cmd
# Configuration file for Oracle HSM archiving file systems
#-----
logfile = /var/adm/recycler.log
no_recycle tp VOL[0-9][2-9][0-9]

```

```
library1 -hwm 95 -mingain 60 -ignore
:wq
root@solaris:~#
```

4. 하나 이상의 아카이브 파일 시스템을 손실 또는 손상으로부터 복구하려는 경우 계속하기 전에 [아카이브되지 않은 파일 백업 \[14\]](#)을 수행합니다.
5. 파일 시스템을 서버 문제, 손실 또는 손상으로부터 복구하려는 경우 계속하기 전에 [Oracle HSM 구성 저장 \[15\]](#)을 수행합니다.
6. 디렉토리 및 파일을 복원해야 하는 경우 [Oracle HSM 구성 저장 \[15\]](#)할지 또는 바로 [5장. 손실 및 손상된 파일 복구 \[41\]](#)로 이동해야 하는지 결정합니다.

2.2. 아카이브되지 않은 데이터 보존

아카이브되지 않은 파일이 손상된 아카이브 파일 시스템의 디스크 캐시에 남아 있을 수 있습니다. 아카이브에는 이러한 파일의 복사본이 없습니다. 따라서 가능하다면 지금 복구 지점 파일에 백업해야 합니다. 다음과 같이 하십시오.

2.2.1. 아카이브되지 않은 파일 백업

1. 파일 시스템 메타 데이터 서버에 *root*로 로그인합니다.

```
root@solaris:~#
```

2. 복구 지점을 위한 안전한 스토리지 위치를 선택합니다.

이 예에서는 처음 구성할 때 복구 지점에 대해 생성한 디렉토리 아래에 하위 디렉토리 *unarchived/*를 생성합니다. */zfs* 파일 시스템에는 복구할 파일 시스템 */samqfs1*과 공통되는 장치가 없습니다.

```
root@solaris:~# mkdir /zfs1/samqfs_recovery/unarchived/
root@solaris:~#
```

3. 파일 시스템의 루트 디렉토리로 변경합니다.

예제에서는 마운트 지점 디렉토리 */samqfs1*로 변경합니다.

```
root@solaris:~# cd /samqfs1
root@solaris:~#
```

4. 디스크 캐시에 남아 있는 아카이브되지 않은 파일을 백업합니다. *samfsdump -u -f recovery-point* 명령을 사용합니다. 여기서 *recovery-point*는 출력 파일의 경로와 파일 이름입니다.

samfsdump 명령에 *-u* 옵션을 사용하면 아카이브되지 않은 데이터 파일이 백업됩니다. 이 예에서는 복구 지점 파일 20150325를 원격 디렉토리 */zfs1/samqfs_recovery/unarchived/*에 저장합니다.

```
root@solaris:~# samfsdump -u -f /zfs1/samqfs_recovery/unarchived/20150325
root@solaris:~#
```

5. 파일 시스템을 서버 문제, 손실 또는 손상으로부터 복구하려는 경우 계속하기 전에 [Oracle HSM 구성 저장 \[15\]](#)을 수행합니다.
6. 디렉토리 및 파일을 복원해야 하는 경우 [Oracle HSM 구성 저장 \[15\]](#)할지 또는 바로 [5장. 손실 및 손상된 파일 복구 \[41\]](#)로 이동해야 하는지 결정합니다.

2.3. 구성 및 상태 정보 보존

Oracle HSM 소프트웨어 및 파일 시스템을 복원하기 위해 필요한 모든 구성 파일 및 스크립트의 백업 복사본을 안전하게 저장해 두었다면 실패한 시스템의 현재 상태를 보존하는 것이 좋습니다. 남아 있는 구성 파일 및 스크립트에는 전체 구성이 마지막으로 백업된 이후 구현된 변경 사항이 포함되어 있을 수 있습니다. 이 경우 시스템을 실패 이전의 상태로 거의 정확하게 복원한 것과 단순히 가깝게 복원한 것의 차이로 나타납니다. 로그 및 추적 파일에는 파일을 복원하고 실패의 원인을 규명하는 데 도움이 되는 정보가 포함됩니다. 이러한 이유로 다른 작업을 수행하기 전에 남아 있는 모든 것을 보존해야 합니다.

2.3.1. Oracle HSM 구성 저장

1. 가능한 경우 파일 시스템 메타 데이터 서버에 *root*로 로그인합니다.

```
root@solaris:~#
```

2. *samexplorer* 명령을 실행하고 SAMreport를 생성하며 백업 구성 정보를 보관하는 디렉토리에 보고서를 저장합니다. *samexplorer path/hostname.YYYYMMDD.hhmmz.tar.gz* 명령을 사용합니다. 여기서 *path*는 선택한 디렉토리의 경로이고 *hostname*은 Oracle HSM 파일 시스템 호스트의 이름이고 *YYYYMMDD.hhmmz*는 날짜 및 시간 기록입니다.

기본 파일 이름은 */tmp/SAMreport.hostname.YYYYMMDD.hhmmz.tar.gz*입니다. 이 예에서는 이미 SAMreport를 저장하기 위한 디렉토리인 */zfs1/sam_config/*가 있습니다. 따라서 이 디렉토리에 보고서를 생성합니다(아래의 명령은 한 라인으로 입력하며 백슬래시로 이스케이프되어 있음).

```
root@solaris:~# samexplorer /zfs1/sam_config/explorer/server1.20150325.1659MST.tar.gz
Report name:      /zfs1/sam_config/explorer/samhost1.20150325.1659MST.tar.gz
Lines per file:   1000
Output format:    tar.gz (default) Use -u for unarchived/uncompressed.

Please wait.....
Please wait.....
Please wait.....

The following files should now be ftp'ed to your support provider
```

```
as ftp type binary.
```

```
/zfs1/sam_config/explorer/samhost1.20150325.1659MST.tar.gz
```

3. Oracle HSM 구성 파일을 가능한 수만큼 독립된 파일 시스템에 복사합니다. 여기에는 다음이 포함됩니다.

```
/etc/opt/SUNWsamfs/
mcf
archiver.cmd
defaults.conf
diskvols.conf
hosts.family-set-name
hosts.family-set-name.local
preview.cmd
recycler.cmd
releaser.cmd
rft.cmd
samfs.cmd
stager.cmd
inquiry.conf
samremote           # SAM-Remote server configuration file
family-set-name      # SAM-Remote client configuration file
network-attached-library # Parameters file
scripts/*            # Back up all locally modified files
/var/opt/SUNWsamfs/
```

4. 내역기 카탈로그를 비롯하여 남아 있는 라이브러리 카탈로그를 모두 백업합니다. 카탈로그마다 `dump_cat -V catalog-file` 명령을 사용합니다. 여기서 *catalog-file*은 카탈로그 파일의 경로와 이름입니다. 출력을 새 위치의 *dump-file*로 재지정합니다.

`dump_cat` 파일의 출력을 통해 `build_cat` 명령을 사용하여 교체 시스템에서 카탈로그를 재작성합니다. 예제에서는 `library1`의 카탈로그 데이터를 독립 NFS 마운트 파일 시스템인 `zfs1`에 있는 디렉토리의 `library1cat.dump` 파일에 덤프합니다. 아래의 명령은 단일 라인으로 입력됩니다. 줄바꿈은 백슬래시 문자로 이스케이프됩니다.

```
root@solaris:~# dump_cat -V /var/opt/SUNWsamfs/catalog/library1cat > /zfs1/sam
_config/20150325/catalogs/library1cat.dump
```

5. Oracle HSM 설치 및 구성 중 수정한 시스템 구성 파일을 독립된 파일 시스템에 복사합니다. 이러한 파일에는 다음이 포함될 수 있습니다.

```
/etc/
syslog.conf
system
vfstab
```



```

/kernel/drv/
    sgen.conf
    samst.conf
    samrd.conf
    sd.conf
    ssd.conf
    st.conf
/usr/kernel/drv/dst.conf

```

6. Oracle HSM 구성의 일부로 생성한 모든 사용자 정의 셸 스크립트 및 *crontab* 항목을 독립된 파일 시스템에 복사합니다.

예를 들어, 복구 지점 만들기를 관리하기 위해 *crontab* 항목을 만든 경우 지금 복사본을 저장합니다.

7. 현재 설치된 소프트웨어의 개정 레벨을 기록하는 *readme* 파일을 생성합니다. Oracle Oracle HSM, Solaris 및 Solaris Cluster를 포함합니다(해당되는 경우). 다른 복구 정보와 함께 독립된 파일 시스템에 파일을 저장합니다.
8. 가능할 경우 다운로드한 Oracle Oracle HSM, Solaris 및 Solaris Cluster 패키지의 복사본을 독립된 파일 시스템에 저장합니다.

패키지를 즉시 사용 가능하게 하면 필요할 경우 소프트웨어를 빠르게 복원할 수 있습니다.

9. Oracle HSM 서버 호스트의 손실로부터 복구하는 경우 [3장. Oracle HSM 구성 복원 \[19\]](#)로 이동합니다.
10. 하나 이상의 Oracle HSM 파일 시스템을 복원해야 하는 경우 [4장. 파일 시스템 복구 \[25\]](#)로 이동합니다.
11. 디렉토리 및 파일을 복원해야 하는 경우 [5장. 손실 및 손상된 파일 복구 \[41\]](#)로 이동합니다.

Oracle HSM 구성 복원

이 장에서는 Oracle Hierarchical Storage Manager and StorageTek QFS Software 및 파일 시스템 구성의 일부 또는 전체가 손실되거나 손상된 경우 복구하는 프로세스를 설명합니다. 서버 호스트가 실패할 경우 Oracle HSM 소프트웨어 및 파일 시스템 구성은 손실되더라도 파일 시스템 데이터 및 메타 데이터는 그대로 있지만 구성 정보가 복원될 때까지 액세스할 수 없습니다. 이러한 상황에서 성공은 남아 있는 파일 및 디렉토리에서 정보를 복원할 수 있는 능력과 철저한 재해 준비 여부에 따라 달라집니다.

- 백업 복사본 및/또는 SAMreport에서 구성 복원 [19]
- 백업 정보 없이 구성 복원 [23].

3.1. 백업 복사본 및/또는 SAMreport에서 구성 복원

Oracle Hierarchical Storage Manager and StorageTek QFS 설치 및 구성 설명서에서 권장하는 절차를 따른 경우, 아래의 절차를 사용하여 Oracle HSM 소프트웨어 및 파일 시스템 구성을 복구할 수 있습니다.

3.1.1. 백업 파일 또는 SAMreport에서 구성 복원

1. 서버 호스트 오류 이후에 구성을 복원하는 경우 하드웨어 문제를 해결하고 운영 체제를 재설치하며 필요에 따라 소프트웨어를 재설치합니다.
2. 루트 파일 시스템의 최신 백업 복사본이 있는 경우 루트 파일 시스템을 복원하고 여기서 중지합니다.
3. 그렇지 않은 경우 파일 시스템 서버 호스트에 *root*로 로그인합니다.

```
root@solaris:~#
```

4. 필요한 파일 시스템을 마운트합니다. 백업 Oracle HSM 구성 파일이 저장된 파일 시스템 및 데이터 파일의 디스크 아카이브 복사본이 포함된 모든 파일 시스템을 마운트합니다.

이 예에서는 Oracle HSM 서버 Solaris 구성 파일의 복사본을 독립된 파일 시스템 *zfs1*의 *sam_config* 하위 디렉토리에 유지 관리합니다. 그래서 마운트 지점을 생성합니다. *zfs1*을 마운트합니다. *zfs1* 파일 시스템에 있는 최신 복사본에서 *vfstab* 파일을 복원합니다. 필요한 마운트 지점을 생성합니다. 그런 다음 파일 시스템을 마운트합니다.

```
root@solaris:~# cp /etc/vfstab /etc/vfstab.back
root@solaris:~# mkdir /zfs1
```

```

root@solaris:~# mount -F zfs /net/remote.example.com/zfs1/ /zfs1
root@solaris:~# cp /zfs1/sam_config/20140127/etc/vfstab /etc/vfstab
root@solaris:~# mkdir /diskvols
root@solaris:~# mkdir /diskvols/DISKVOL1
root@solaris:~# mkdir /diskvols/DISKVOL2
...
root@solaris:~# mount /diskvols/DISKVOL1
root@solaris:~# mount /diskvols/DISKVOL2
...
root@solaris:~#

```

5. Oracle HSM 구성 파일의 백업 복사본을 사용할 수 있는 경우 구성 손실 이전 날짜의 최신 복사본을 찾습니다.

이 예에서는 Oracle HSM 구성 파일의 복사본을 독립된 파일 시스템 `/zfs1`의 `sam_config` 하위 디렉토리에 유지 관리합니다. 따라서 최근 파일을 쉽게 찾을 수 있습니다.

```

root@solaris:~# ls /zfs1/sam_config/20140127/etc/opt/SUNWsamfs/
archiver.cmd      defaults.conf     mcf              recycler.cmd     stager.cmd
cfg_backups       diskvols.conf     mgmt_sched.conf  releaser.cmd     startup
csn               inquiry.conf      notify.cmd       scripts          verifyd.cmd
root@solaris:~# ls /zfs1/sam_config/20140127/etc/opt/SUNWsamfs/scripts
archiver.sh  log_rotate.sh  nrecycler.sh  recycler.sh  save_core.sh
sendtrap     ssi.sh
root@solaris:~# ls /zfs1/sam_config/explorer/
server1.20140430.1659MST.tar.gz  server1.20140114.0905MST.tar.gz
server1.20110714.1000MST.tar.gz

```

6. SAMreport가 Oracle HSM 구성 손실 이전에 생성된 경우 최근 보고서를 찾습니다.
7. QFS 파일 시스템이 현재 마운트된 경우 이 파일 시스템을 마운트 해제합니다.
8. 각 누락된 구성 파일에 대해 사용 가능한 백업 파일을 복원하려는 서버의 필요한 위치에 복사합니다.

이 예에서는 백업 복사본에서 모든 Oracle HSM 구성 파일 및 스크립트를 복원합니다 (아래 명령은 한 라인으로 입력하며 줄바꿈이 백슬래시로 이스케이프되어 있음).

```

root@solaris:~# cp /zfs1/sam_config/20140127/etc/opt/SUNWsamfs/* /
/etc/opt/SUNWsamfs/
root@solaris:~# cp /zfs1/sam_config/20140127/etc/opt/SUNWsamfs/scripts/* /
/etc/opt/SUNWsamfs/scripts/
root@solaris:~# cp /zfs1/sam_config/20140127/etc/opt/SUNWsamfs/startup/* /
/etc/opt/SUNWsamfs/startup/
root@solaris:~# cp /zfs1/sam_config/20140127/etc/opt/SUNWsamfs/cfg_backups/* /
/etc/opt/SUNWsamfs/cfg_backups/
root@solaris:~# cp /zfs1/sam_config/20140127/etc/opt/SUNWsamfs/csn/* /
/etc/opt/SUNWsamfs/csn/

```

9. 구성 파일의 백업 복사본을 사용할 수 없는 경우 사용 가능한 최근 SAMreport에 포함된 정보를 사용하여 다시 만듭니다. 보고서에서 내용을 복사하고 텍스트 편집기에 붙여 넣은 다음 보고서에 나타난 파일 및 경로에 저장합니다.

SAMreport 파일에는 보고서가 생성된 당시에 있었던 Oracle HSM 파일의 전체 텍스트가 포함되어 있습니다. 또한 파일이 있었던 디렉토리도 나열합니다.

이 예에서는 *server1.20140127.SAMreport* 파일에서 Oracle HSM 마스터 구성 파일(*mcf*) 정보를 검색합니다. *cat* 명령의 출력을 *grep* 명령 및 정규 표현식 패턴 *//etc//opt//SUNWsamfs//mcf*에 연결합니다(아래 명령은 한 라인으로 입력하며 줄바꿈이 백슬래시로 이스케이프되어 있음).

```
root@solaris:~# cat /zfs1/sam_config/explorer/server1.20140127.SAMreport | / grep //etc//
opt//SUNWsamfs//mcf
...
----- /etc/opt/SUNWsamfs/mcf -----
server1# /bin/ls -l /etc/opt/SUNWsamfs/mcf
-rw-r--r--  1 root    root        1789 Feb  4 09:22 /etc/opt/SUNWsamfs/mcf

# Equipment      Equipment  Equipment  Family      Device      Additional
# Identifier      Ordinal    Type        Set          State        Parameters
#-----
samqfs1           100        ms          samqfs1      on
/dev/dsk/c1t3d0s3 101        md          samqfs1      on
/dev/dsk/c1t4d0s5 102        md          samqfs1      on
root@solaris:~#
```

grep 명령의 출력을 복사하여 vi 편집기에 붙여넣고 올바른 이름과 위치에 파일을 저장합니다.

```
root@solaris:~# vi /etc/opt/SUNWsamfs/mcf
# Equipment      Equipment  Equipment  Family      Device      Additional
# Identifier      Ordinal    Type        Set          State        Parameters
#-----
samqfs1           100        ms          samqfs1      on
/dev/dsk/c1t3d0s3 101        md          samqfs1      on
/dev/dsk/c1t4d0s5 102        md          samqfs1      on
:wq
root@solaris:~#
```

10. 2.3.1절. “Oracle HSM 구성 저장” [15] 절차 중 저장한 덤프 파일 데이터에서 라이브러리 카탈로그를 복원합니다. 각 카탈로그에 대해 *build_cat catalog-dump-file catalog-file* 명령을 사용합니다. 설명:

- *catalog-dump-file*은 *dump_cat* 명령으로 생성한 파일의 경로 및 이름입니다.

- *catalog-file*은 복원된 카탈로그 파일의 경로 및 이름입니다.

이 예에서는 */zfs1/sam_config/20140513/catalogs/library1cat.dump* 파일의 데이터를 사용하여 *library1*에 대한 카탈로그를 재작성합니다.

```
root@solaris:~# build_cat /zfs1/sam_config/20140513/catalogs/library1cat.dump / /var/opt/SUNWsamfs/catalog/library1cat
```

11. 하드웨어 실패 이후 시스템을 복구하는 경우 [4장. 파일 시스템 복구 \[25\]](#)로 이동합니다.
12. 부주의로 삭제되거나 잘못 편집된 하나 이상의 구성 파일을 교체하는 경우 및 하드웨어 또는 파일 시스템 변경 사항이 발생하지 않은 경우 *sam-fsd* 명령을 실행하여 구성 파일에서 오류를 검사합니다.

*sam-fsd*는 Oracle HSM 구성 파일을 읽는 초기화 명령입니다. 오류가 발견되면 실행을 중지합니다.

```
root@solaris:~# sam-fsd
```

13. *sam-fsd* 명령이 *mcf* 파일에서 오류를 찾을 경우 파일을 편집하여 오류를 해결하고 이전 단계에 설명된 대로 다시 검사합니다.

아래의 예에서는 *sam-fsd*가 장치에서 지정되지 않은 문제를 보고합니다.

```
root@solaris:~# sam-fsd
Problem in mcf file /etc/opt/SUNWsamfs/mcf for filesystem samqfs1
sam-fsd: Problem with file system devices.
root@solaris:~#
```

14. *sam-fsd* 명령이 오류 없이 실행될 경우 구성 파일이 올바른 것입니다. 다음 단계로 진행하십시오.

이 예는 오류가 없는 출력의 일부입니다.

```
root@solaris:~# sam-fsd
Trace file controls:
sam-amld      /var/opt/SUNWsamfs/trace/sam-amld
               cust err fatal ipc misc proc date
               size    10M  age 0
sam-archiverd /var/opt/SUNWsamfs/trace/sam-archiverd
               cust err fatal ipc misc proc date module
               size    10M  age 0
sam-catserverd /var/opt/SUNWsamfs/trace/sam-catserverd
               cust err fatal ipc misc proc date module
               size    10M  age 0
...
```

```

Would start sam-archiverd()
Would start sam-stagealld()
Would start sam-stagerd()
Would start sam-amld()
root@solaris:~#

```

15. Oracle HSM 소프트웨어가 *mcf* 파일을 읽고 그에 따라 재구성하도록 지정합니다. *samd config* 명령을 사용합니다.

```

root@solaris:~# samd config
Configuring SAM-FS
root@solaris:~#

```

16. *mcf* 파일에서 *samd config* 명령이 오류를 보고한 경우 오류를 수정합니다. 그런 다음 이전 단계를 반복합니다.
17. 영향을 받은 파일 시스템을 다시 마운트합니다.
18. 파일 시스템 작업을 모니터링합니다.

3.2. 백업 정보 없이 구성 복원

백업 파일 또는 SAMreport가 없다면 사용 가능한 모든 정보를 동원하여 구성을 재구성합니다. 그런 다음 새 구성에 따라 진행합니다. 지침은 *Oracle Hierarchical Storage Manager and StorageTek QFS* 설치 및 구성 설명서를 참조하십시오.

파일 시스템 복구

이 절에서는 전체 Oracle HSM 파일 시스템이 손상되거나 손실된 경우 사용하는 복구 프로세스를 설명합니다. 절차는 관련 파일 시스템의 유형 및 마련된 백업 유형 및 복구 준비에 따라 달라집니다. 그러나 수행해야 하는 두 가지 기본적인 작업이 있습니다.

- [파일 시스템 다시 만들기 \[25\]](#)
- [디렉토리 및 파일 복원 \[29\]](#).

시작하기 전에 Oracle HSM 메타 데이터 서버의 손실로부터 복구하는 경우 [3장. Oracle HSM 구성 복원 \[19\]](#)에 설명된 대로 [3장. Oracle HSM 구성 복원 \[19\]](#)을 완료한 다음 계속 진행하십시오. 이 장의 절차에서는 파일 시스템 손실 이전에 Oracle HSM 소프트웨어가 정상적으로 설치 및 구성되어 있다고 간주합니다.

4.1. 파일 시스템 다시 만들기

파일 및 디렉토리를 복구하려면 먼저 둘 곳이 있어야 합니다. 따라서 복구 프로세스의 첫 번째 단계는 빈 대체 파일 시스템을 만드는 것입니다. 다음과 같이 하십시오.

4.1.1. 백업 구성 파일 및 `sammkfs` 명령을 사용하여 파일 시스템 다시 만들기

1. 파일 시스템 메타 데이터 서버에 `root`로 로그인합니다.

```
root@solaris:~#
```

2. 파일 시스템이 현재 마운트된 경우 마운트 해제합니다. `umount mount-point` 명령을 사용합니다. 여기서 `mount-point`는 파일 시스템이 마운트된 디렉토리입니다.

이 예에서는 `/samqfs1` 파일 시스템을 마운트 해제합니다.

```
root@solaris:~# umount /samqfs1
root@solaris:~#
```

3. `/etc/opt/SUNWsamfs/mcf` 파일을 텍스트 편집기에서 엽니다. 하드웨어 구성을 확인합니다. 하드웨어를 변경했어야 하는 경우 적절히 파일을 편집하고 변경 사항을 저장합니다.

이 예에서는 2개의 실패한 디스크 장치에 대한 장비 식별자를 교체한 장치의 식별자로 바꿉니다. 장비 서수는 변경되지 않고 그대로 유지됩니다.

```

root@solaris:~# vi /etc/opt/SUNWsamfs/mcf
# Equipment      Equipment  Equipment  Family    Device  Additional
# Identifier      Ordinal   Type       Set       State   Parameters
#-----
samqfs1           100       ms         samqfs1   on
/dev/dsk/c1t3d0s3 101       md         samqfs1   on
/dev/dsk/c1t4d0s5 102       md         samqfs1   on
# Tape library
/dev/scsi/changer/c1t2d0 800       rb         lib800    on      .../lib800_cat
/dev/rmt/0cbn      801       li         lib800    on
/dev/rmt/1cbn      802       li         lib800    on
:wq
root@solaris:~#

```

4. *mcf* 파일에서 오류를 검사합니다. *sam-fsd* 명령을 사용합니다.

sam-fsd 명령은 Oracle HSM 구성 파일을 읽고 소프트웨어를 초기화합니다. 오류가 발견되면 실행을 중지합니다.

```

root@solaris:~# sam-fsd

```

5. *sam-fsd* 명령이 *mcf* 파일에서 오류를 찾을 경우 파일을 편집하여 오류를 해결하고 이전 단계에 설명된 대로 다시 검사합니다.

아래의 예에서는 *sam-fsd*가 장치에서 지정되지 않은 문제를 보고합니다. 장비 식별자 필드의 오타일 수 있습니다.

```

root@solaris:~# sam-fsd
Problem in mcf file /etc/opt/SUNWsamfs/mcf for filesystem qfsms
sam-fsd: Problem with file system devices.

```

대개 이러한 오류는 부주의한 타이핑 실수의 결과입니다. 여기에서는 *mcf* 파일을 편집기에서 열면 두번째 *md* 장치인 장치 102에 대한 장비 이름의 슬라이스 번호 부분에서 0 대신에 문자 *o*를 입력했다는 것을 알 수 있습니다.

```

root@solaris:~# vi /etc/opt/SUNWsamfs/mcf
...
qfsms           100       ms         qfsms      on
/dev/dsk/c0t0d0s0 101       md         qfsms      on
/dev/dsk/c0t3d0s0 102       md         qfsms      on

```

오류를 해결하고 파일을 저장한 후 다시 검사합니다.

```

root@solaris:~# vi /etc/opt/SUNWsamfs/mcf

```

```

...
qfsm      100      ms      qfsm      on
/dev/dsk/c0t0d0s0 101      md      qfsm      on
/dev/dsk/c0t3d0s0 102      md      qfsm      on
:wq
root@solaris:~# sam-fsd

```

6. *sam-fsd* 명령이 오류 없이 실행될 경우 *mcf* 파일이 올바른 것입니다. 다음 단계로 진행하십시오.

이 예에서는 *sam-fsd*가 오류 없이 실행됩니다.

```

root@solaris:~# sam-fsd
Trace file controls:
sam-amld      /var/opt/SUNWsamfs/trace/sam-amld
...
Would start sam-archiverd()
Would start sam-stagealld()
Would start sam-stagerd()
Would start sam-amld()
root@solaris:~#

```

7. Oracle HSM 소프트웨어가 *mcf* 파일을 읽고 그에 따라 재구성하도록 지정합니다.

```

root@solaris:~# samd config
Configuring SAM-FS
root@solaris:~#

```

8. 교체 파일 시스템을 만듭니다. *sammkfs family-set-name* 명령을 사용합니다. 여기서 *family-set-name*은 파일 시스템의 이름입니다.

이 예에서는 파일 시스템 *samqfs1*을 다시 만듭니다.

```

root@solaris:~# sammkfs samqfs1
Building 'samqfs1' will destroy the contents of devices:
/dev/dsk/c0t0d0s0
/dev/dsk/c0t3d0s0
Do you wish to continue? [y/N]yes
total data kilobytes      = ...
root@solaris:~#

```

9. 필요한 경우 파일 시스템에 대한 마운트 지점 디렉토리를 다시 만듭니다.

이 예에서는 */samqfs1* 디렉토리를 다시 만듭니다.

```

root@solaris:~# mkdir /samqfs1

```

```
root@solaris:~#
```

10. 운영 체제의 `/etc/vfstab` 파일을 백업합니다.

```
root@solaris:~# cp /etc/vfstab /etc/vfstab.backup
root@solaris:~#
```

11. 텍스트 편집기에서 `/etc/vfstab` 파일을 엽니다. `/etc/vfstab` 파일에 복원하는 파일 시스템에 대한 마운트 매개변수가 포함되지 않은 경우 마운트 매개변수를 복원해야 합니다.

이 예에서는 Oracle HSM 서버가 교체 호스트에 설치됩니다. 따라서 파일에는 복원하는 파일 시스템 `samqfs1`에 대한 마운트 매개변수가 포함되지 않습니다.

```
root@solaris:~# vi /etc/vfstab
#File
#Device      Device  Mount      System  fsck  Mount      Mount
#to Mount    to fsck  Point      Type    Pass  at Boot    Options
#-----
/devices      -       /devices   devfs   -     no         -
/proc         -       /proc      proc    -     no         -
...
```

12. 가능하다면 마운트 매개변수를 복원해야 하는 경우 원래 `/etc/vfstab` 파일의 백업 복사본을 열고 필요한 라인을 현재 `/etc/vfstab` 파일에 복사합니다. 변경 작업이 완료되었으면 파일을 저장하고 편집기를 닫습니다.

이 예에서는 백업 복사본 `/zfs1/sam_config/20140127/etc/vfstab`가 있습니다. 따라서 백업 복사본에서 `samqfs1` 파일 시스템에 대한 라인을 복사하고 현재 `/etc/vfstab` 파일에 붙여 넣습니다.

```
root@solaris:~# vi /zfs1/sam_config/20140127/etc/vfstab.20140127
#File
#Device      Device  Mount      System  fsck  Mount      Mount
#to Mount    to fsck  Point      Type    Pass  at Boot    Options
#-----
/devices      -       /devices   devfs   -     no         -
/proc         -       /proc      proc    -     no         -
...
samqfs1      -       /samqfs1   samfs   -     yes        stripe=1,bg
:q

root@solaris:~# vi /etc/vfstab
#File
#Device      Device  Mount      System  fsck  Mount      Mount
#to Mount    to fsck  Point      Type    Pass  at Boot    Options
```

```
#-----
/devices - /devices devfs - no -
/proc - /proc proc - no -
...
samqfs1 - /samqfs1 samfs - yes stripe=1,bg
:wq
root@solaris:~#
```

13. 파일 시스템을 마운트합니다.

이 예에서는 파일 시스템 *samqfs1*을 마운트합니다.

```
root@solaris:~# mount /samqfs1
root@solaris:~#
```

14. 이제 [디렉토리 및 파일 복원 \[29\]](#)을 시작합니다.

4.2. 디렉토리 및 파일 복원

기본 파일 시스템을 다시 만들었으면 디렉토리 및 파일 복원을 시작할 수 있습니다. 두 가지 가능한 방식이 있습니다.

- **samfsdump(qfsdump) 복구 지점 파일에서 파일 및 디렉토리 복원 [29]**은 정기적으로 복구 지점을 만들고 안전하게 저장한 경우 최상의 옵션입니다.

이 방식은 파일 시스템 메타 데이터를 복원하므로 파일 시스템을 즉시 완전한 작동 상태로 되돌립니다. 아카이브 파일 시스템은 아카이브 매체의 데이터에 즉시 액세스하고 사용자가 파일에 액세스할 때 즉시 또는 필요에 따라 파일을 디스크 캐시에 다시 스테이징할 수 있습니다. 파일은 원래 속성으로 복원됩니다.

복구 지점에 메타 데이터는 물론 데이터도 포함된 경우 이 방식은 타사 응용 프로그램으로 백업되지 않은 독립형(비아카이브) 파일 시스템을 복원할 수 있는 유일한 방법이기도 합니다.

- **복구 지점 파일 없이 아카이브 매체에서 파일 및 디렉토리 복원 [36]**에서는 복구 스크립트 및 Oracle HSM *star* 유틸리티를 사용합니다.

4.2.1. samfsdump(qfsdump) 복구 지점 파일에서 파일 및 디렉토리 복원

가능하면 언제나 사용 가능한 최근 복구 지점 파일을 기준으로 파일 시스템 복구 작업을 수행해야 합니다. 이 방식은 Oracle HSM 파일 시스템 실패로부터 복구하기 위한 가장 빠르고 신뢰할 수 있으며 철저하고 가장 덜 수고스러운 방법입니다. 따라서 복구 지점 파일이 있는 경우 다음과 같이 수행하십시오.

4.2.1.1. 복구 지점 파일에서 손실된 파일 시스템 복원

1. 파일 시스템 메타 데이터 서버에 *root*로 로그인합니다.

```
root@solaris:~#
```

2. 아직 그렇게 하지 않은 경우 2.1절. “아카이브 및 재활용 프로세스 중지” [11]의 절차를 사용하여 아카이브 및 재활용을 중지합니다.
3. 사용 가능한 최근 복구 지점 파일을 찾습니다.

이 예에서는 독립된 파일 시스템 `/zfs1`에서 잘 알려진 위치인 `samqfs1_recovery` 하위 디렉토리에 파일 시스템 `samqfs1`에 대한 날짜별 복구 지점 파일을 만들었습니다. 따라서 최신 파일 `20140324`를 쉽게 찾을 수 있습니다.

```
root@solaris:~# ls /zfs1/samqfs1_recovery/
20140321    20140322    20140323    20140324
root@solaris:~#
```

4. 다시 만든 파일 시스템에 대한 마운트 지점 디렉토리로 변경합니다.

이 예에서는 다시 생성된 파일 시스템이 `/samqfs1`에 마운트됩니다.

```
root@solaris:~# cd /samqfs1
root@solaris:~#
```

5. 현재 디렉토리에 상대적인 전체 파일 시스템을 복원합니다. `samfsrestore -T -f recovery-point-file -g logfile` 명령 또는 QFS 전용 명령 `qfsrestore -T -f recovery-point-file -g logfile`을 사용합니다. 설명:
 - `-T`는 명령이 종료될 때 처리된 파일과 디렉토리 수 및 오류 및 경고 수를 포함한 복구 통계를 표시합니다.
 - `-f recovery-point-file`은 선택한 복구 지점 파일의 경로와 이름을 지정합니다.
 - `-g logfile`은 복구 지점이 만들어질 때 온라인이었던 디렉토리 및 파일 목록을 만들고 `logfile`로 지정된 파일에 목록을 저장합니다.

아카이브 파일 시스템을 복원하는 경우 이 파일은 아카이브 매체에서 파일을 자동으로 스테이징하는 데 사용할 수 있으므로 디스크 캐시는 복구 지점이 만들어진 시점과 동일한 상태입니다.

이 예에서는 파일 시스템 `samqfs1`을 복구 지점 파일 `/zfs1/samqfs1_recovery/20140324`에서 복원합니다. `/root/20140324.log` 파일에 온라인 파일을 기록합니다(아래의 명령은 한 라인으로 입력하며 줄바꿈이 백슬래시로 이스케이프되어 있음).

```
root@solaris:~# samfsrestore -T -f /zfs1/samqfs1_recovery/20140324 /
-g /root/20140324.log
samfsdump statistics:
      Files:                52020
    Directories:            36031
    Symbolic links:         0
```

```

Resource files:      8
File segments:      0
File archives:      0
Damaged files:      0
Files with data:    24102
File warnings:      0
Errors:             0
Unprocessed dirs:   0
File data bytes:    0

root@solaris:~#

```

6. 독립형(비아카이브) 파일 시스템을 복원한 경우 복구 지점 파일에 저장되었던 파일 시스템 메타 데이터 및 파일 데이터가 복원됩니다. 여기에서 중지합니다.
7. 그렇지 않은 경우 [아카이브된 파일 다시 스테이징\(필요한 경우\) \[31\]](#)을 수행합니다.

4.2.1.2. 아카이브된 파일 다시 스테이징(필요한 경우)

1. 대부분의 경우 파일 시스템 복구 이후 아카이브 매체에서 디스크로 파일을 다시 스테이징하지 마십시오. 사용자가 필요에 따라 파일에 액세스하여 스테이징하도록 하십시오.

이 방식은 사용자 필요에 따라 스테이징 우선 순위를 자동으로 지정하고 일정 시간 동안 오프라인이었을 수 있는 파일 시스템의 가용성을 극대화합니다. 필요한 파일만 즉시 스테이징되므로 전체 스테이징 노력이 일정 기간에 걸쳐 분산됩니다. 이 방식은 드라이브와 같은 파일 시스템 리소스가 높은 우선 순위의 작업(예: 새로운 파일 아카이브 및 긴급하게 필요한 사용자 데이터 스테이징)에 항상 사용될 수 있도록 합니다.

또한 이 방식은 복구와 관련된 관리 노력을 줄여줍니다.

2. 오류 이전에 디스크 캐시에 있던 파일을 다시 스테이징해야 하는 경우 `/opt/SUNWsamfs/examples/restore.sh logfile` 명령을 사용합니다. 여기서 *logfile*은 `samfsrestore(qfsrestore)` 명령의 `-g` 옵션을 사용하여 만든 로그 파일의 경로와 파일 이름입니다.

`restore.sh` 스크립트는 로그 파일에 나열된 파일을 스테이징합니다. 이러한 파일은 `samfsrestore(qfsrestore)` 복구 지점 파일이 만들어질 때 온라인 상태였던 파일입니다.

수천 개의 파일을 스테이징해야 하는 경우 로그 파일을 여러 작은 파일로 분할하는 것을 고려해 보십시오. 그런 다음 각 파일에 대해 하나씩 `restore.sh` 스크립트를 실행하십시오. 그러면 스테이징 노력이 일정 기간에 걸쳐 분산되고 아카이브 및 사용자가 시작한 스테이징에 대한 간섭을 줄일 수 있습니다.

3. 이제 [손상된 파일 식별 및 교체 복사본 찾기 \[31\]](#)를 수행합니다.

4.2.1.3. 손상된 파일 식별 및 교체 복사본 찾기

`samfsrestore` 프로세스는 테이프에서 해당 파일 시스템을 찾아 파일 시스템 내의 적절한 위치에 복원할 수 있도록 파일 시스템 메타 데이터의 복사본을 복구 지점 파일에서 복원합니다.

다. 하지만 복구 지점 파일은 파일 시스템이 손실되기 전에 생성됩니다. 따라서 불가피하게 메타 데이터 중 일부는 복구 지점이 생성된 후 변경된 데이터 위치를 가리키는 것이 보통입니다. 파일 시스템은 이러한 파일의 레코드를 포함하고 있지만 그 내용을 찾을 수 없으므로 해당 파일에 각각 *damaged* 플래그를 설정합니다.

손상된 파일에 대한 데이터가 실제로 손실되는 경우도 있지만 복원된 메타 데이터가 단지 오래된 경우도 있습니다. 단지 복원된 메타 데이터가 현재 위치를 기록하지 않아 복원된 파일 시스템이 복구 지점 생성 후 아카이브된 파일의 데이터를 찾지 못할 수도 있습니다. 이 경우 직접 데이터를 찾고 복원된 메타 데이터를 업데이트하여 파일의 손상을 해제할 수 있습니다.

누락된 데이터를 찾고 메타 데이터를 업데이트하며 파일을 손상 해제하려면 아카이버 로그를 사용하십시오. 다음과 같이 하십시오.

1. 아직 그렇게 하지 않은 경우 파일 시스템 메타 데이터 서버에 *root*로 로그인합니다.

```
root@solaris:~#
```

2. 사용 가능한 최근 아카이버 로그 파일을 찾습니다.

서버에서 아카이버 로그를 여전히 사용할 수 있는 경우 최근 정보가 포함되어 있을 수 있습니다. 그렇지 않은 경우 백업 복사본을 사용해야 합니다.

이 예에서는 아카이버 로그 파일 *samqfs1.archiver.log*가 서버의 */var/adm/* 하위 디렉토리에 있습니다. 또한 독립된 파일 시스템 */zfs1*에서 잘 알려진 위치 *samqfs1_recovery/archlogs* 하위 디렉토리에 날짜별 아카이버 로그 파일 복사본이 있습니다. 따라서 최근 파일인 *samqfs1.archiver.log* 및 최근 백업인 *20150324*가 모두 있습니다.

```
root@solaris:~# dir /var/adm/*.archiver.log
samqfs1.archiver.log
root@solaris:~# dir /zfs1/samqfs1_recovery/archivlogs
20150322    20150323    20150324
root@solaris:~#
```

3. 새로 복원된 파일 시스템에서 손상된 파일을 식별합니다. *sfind mountpoint -damaged* 명령을 사용합니다. 여기서 *mountpoint*는 복구된 파일 시스템이 마운트된 디렉토리입니다.

이 예에서는 */samqfs1* 디렉토리에서 검색을 시작하고 6개의 손상된 파일을 찾습니다.

```
root@solaris:~# sfind /samqfs1 -damaged
./genfiles/ay0
./genfiles/ay1
./genfiles/ay2
./genfiles/ay5
./genfiles/ay6
./genfiles/ay9
```



```
root@solaris:~#
```

4. 아카이버 로그의 최근 복사본에서 손상된 각 파일과 관련된 항목을 검색합니다. `grep "file-name-expression" archiver-log` 명령을 사용합니다. 여기서 `file-name-expression`은 손상된 파일과 일치하는 정규 표현식이고 `archiver-log`는 검사하는 아카이버 로그 복사본의 경로와 이름입니다.

이 예에서는 정규 표현식 `genfiles//ay0`을 사용하여 최신 로그 파일에서 `genfiles/ay0` 파일에 관련된 항목을 검색합니다.

```
root@solaris:~# grep "genfiles//ay0 " /var/adm/samqfs1.archiver.log
```

5. 파일에 대한 항목을 찾을 경우 매체 유형, 볼륨 일련 번호 및 데이터 파일이 아카이브된 아카이브(`tar`) 파일의 위치를 메모합니다. 또한 파일 복원 방법에 영향을 주는 파일 유형도 메모합니다.

이 예에서는 `genfiles/ay0` 파일에 대한 항목을 찾습니다. 로그 항목은 LTO(`li`) 볼륨 `VOL012`를 사용하여 2015년 3월 4일 9:49 PM에 아카이브되었음(A)을 표시합니다. 파일은 16진수 위치 `0x78(78)`의 테이프 아카이브 파일에 저장되어 있습니다. 이 파일은 정규 파일이며 `f` 유형입니다.

```
root@solaris:~# grep "genfiles//ay0 " /var/adm/samqfs1.archiver.log
```

```
A 2015/03/04 21:49:15 li VOL012 SLOT12 allsets.1 78.1 samqfs1 7131.14 8087 genfiles/
ay0 f 0 51
```

```
root@solaris:~#
```

아카이버 로그 항목의 필드에 대한 자세한 설명은 [부록 A. 아카이버 로그 이해 \[63\]](#)를 참조하십시오.

6. 현재 아카이버 로그 복사본에서 손상된 파일에 대한 항목을 찾지 못할 경우 복구 지점 파일이 만들어진 이후 만들어진 백업 아카이브 로그를 사용하여 검색을 반복합니다.

아카이버 로그는 자주 교체됩니다. 따라서 여러 아카이버 로그 복사본을 보존하고 있을 경우 현재 아카이버 로그 기간 이전에 만들어진 아카이브 복사본을 사용하여 손상된 파일을 복구할 수 있습니다.

7. [복구 지점이 생성된 후 아카이브된 누락 파일 찾기 \[33\]](#)를 수행합니다.

4.2.1.4. 복구 지점이 생성된 후 아카이브된 누락 파일 찾기

`samfsrestore` 프로세스는 테이프에서 해당 파일 시스템을 찾아 파일 시스템 내의 적절한 위치에 복원할 수 있도록 파일 시스템 메타 데이터의 복사본을 복구 지점 파일에서 복원합니다. 하지만 복구 지점 파일은 파일 시스템이 손실되기 전에 생성됩니다. 이 파일이 생성된 후에 생성 및 아카이브된 파일의 메타 데이터를 이 파일에 포함할 수 없습니다.

대개 일부 파일은 마지막 복구 지점이 만들어진 이후 및 파일 시스템 손실 이전에 아카이브됩니다. 이러한 파일에 대한 메타 데이터는 복구 지점 파일에 없으므로 파일이 손상되더라도 `samfsrestore`가 파일을 복구할 수 없습니다. 하지만 파일 데이터는 아카이브 매체에 상주

하므로 메타 데이터를 다시 생성하고 아카이브 로그를 사용하여 파일 시스템의 적절한 위치에 파일을 복구할 수 있습니다.

1. 아직 그렇게 하지 않은 경우 파일 시스템 메타 데이터 서버에 *root*로 로그인합니다.

```
root@solaris:~#
```

2. 사용 가능한 최근 아카이버 로그 파일을 찾습니다.

서버에서 아카이버 로그를 여전히 사용할 수 있는 경우 최근 정보가 포함되어 있을 수 있습니다. 그렇지 않은 경우 백업 복사본을 사용해야 합니다.

이 예에서는 아카이버 로그 파일 *samqfs1.archiver.log*가 서버의 */var/adm/* 하위 디렉토리에 있습니다. 또한 독립된 파일 시스템 */zfs1*에서 잘 알려진 위치 *samqfs1_recovery/archlogs* 하위 디렉토리에 날짜별 아카이버 로그 파일 복사본이 있습니다. 따라서 최근 파일인 *samqfs1.archiver.log* 및 최근 백업인 *20150324*가 모두 있습니다.

```
root@solaris:~# dir /var/adm/*.archiver.log
samqfs1.archiver.log
root@solaris:~# dir /zfs1/samqfs1_recovery/archivelogs
20150322    20150323    20150324
root@solaris:~#
```

3. 아카이버 로그의 최근 복사본에서 복구 지점이 만들어진 이후에 만들어진 항목을 검색합니다. *grep "time-date-expression" archiver-log* 명령을 사용합니다. 여기서 *time-date-expression*은 검색을 시작할 날짜 및 시간과 일치하는 정규 표현식이고 *archiver-log*는 검사하는 아카이버 로그 복사본의 경로와 이름입니다.

이 예에서는 2015년 3월 24일 2:02 AM에 파일 시스템이 손실되었습니다. 마지막 복구 지점 파일은 2015년 3월 23일 2:10 AM에 만들어졌습니다. 따라서 정규 표현식 *^A 2015//03//2[45]*를 사용하여 3월 23일 또는 24일에 기록된 아카이브 파일의 최신 로그 파일을 검색합니다.

```
root@solaris:~# grep "^A 2015//03//2[34]" /var/adm/samqfs1.archiver.log
```

4. 복원되지 않은 파일의 아카이브된 복사본에 대한 항목을 찾을 경우 경로, 이름, 파일 유형, 매체 유형 및 위치 정보를 메모합니다.

파일 유형은 정규 파일의 경우 *f*, 이동식 매체 파일의 경우 *R*, 세그먼트화된 파일의 데이터 세그먼트의 경우 *s*로 나열됩니다. 매체 유형은 2자리 문자 코드입니다([부록 B. 장비 유형 용어집 \[65\]](#) 참조).

백업 복사본을 찾으려면 복사본을 저장하는 매체 볼륨의 볼륨 일련 번호가 필요합니다. 복사본이 마그네틱 테이프와 같이 순차 액세스 매체에 저장된 경우에도 아카이브(*tar*) 파일의 시작 위치를 나타내는 16진수 값을 메모합니다. 복사본이 아카이브 디스크와 같

이 랜덤 액세스 매체에 저장된 경우 볼륨 일련 번호에 대한 *tar* 파일의 상대 경로와 파일 이름을 메모합니다. 마지막으로 파일이 세그먼트화된 경우 세그먼트 길이를 메모합니다.

아래의 예에서 아카이버 로그 항목은 마지막 복구 지점이 만들어진 이후에 다음 파일이 아카이브되었음을 표시합니다.

```
root@solaris:~# grep "^A 2015//03//2[34]" /var/adm/samqfs1.archiver.log
A 2015/03/23 10:43:18 li VOL002 all.1 111.1 samqfs1 1053.3 69 genfiles/hops f 0 0
A 2015/03/23 10:43:18 li VOL002 all.1 111.3 samqfs1 1051.1 104 genfiles/anic f 0 0
A 2015/03/23 13:09:05 li VOL004 all.1 212.1 samqfs1 1535.2 1971 genfiles/genA0 f 0 0
A 2015/03/23 13:09:06 li VOL004 all.1 212.20 samqfs1 1534.2 1497 genfiles/genA9 f 0 0
A 2015/03/23 13:10:15 li VOL004 all.1 212.3f samqfs1 1533.2 6491 genfiles/genA2 f 0 0
A 2015/03/23 13:12:25 li VOL003 all.1 2.5e samqfs1 1532.2 17717 genfiles/genA13 f 0 0
A 2015/03/23 13:12:28 li VOL003 all.1 2.7d samqfs1 1531.2 14472 genfiles/genA4 f 0 0
A 2015/03/23 13:12:40 li VOL003 all.1 2.9c samqfs1 1530.2 19971 genfiles/genA45 f 0 0
A 2015/03/23 21:49:15 dk DISKVOL1/f2 all.1 2.2e9 samqfs1 1511.2 8971 socfiles/spcC4 f 0 0
A 2015/03/23 21:49:15 dk DISKVOL1/f2 all.1 2.308 samqfs1 1510.2 7797 spcfiles/spcC5 f 0 0
A 2015/03/23 14:01:47 li VOL013 all.1 76a.1 samqfs1 14.5 10485760 bf/dat011/1 S 0 51
A 2015/03/23 14:04:11 li VOL013 all.1 76a.5002 samqfs1 15.5 10485760 bf/dat011/2 S 0 51
A 2015/03/23 14:06:24 li VOL013 all.1 1409aa4.1 samqfs1 16.5 184 bf/dat011/3 S 0 51
A 2015/03/23 18:28:51 li VOL036 all.1 12d.1 samqfs1 11731.1 89128448 rf/rf81 f 0 210
A 2015/03/23 18:28:51 li VOL034 all.1 15f.0 samqfs1 11731.1 525271552 rf/rf81 f 1 220
root@solaris:~#
```

다음 정보를 메모합니다.

- 8개의 정규(유형 *f*) 파일이 LTO(*li*) 매체에서 아카이브됨(A): *VOL002* 볼륨의 *0x111* 위치에 *genfiles/hops* 및 *genfiles/anic*, *VOL004* 볼륨의 *0x212* 위치에 *genfiles/genA0*, *genfiles/genA9* 및 *genfiles/genA2*, 그리고 *VOL003* 볼륨의 *0x212* 위치에 *genfiles/genA13*, *genfiles/genA4* 및 *genfiles/genA45*가 있습니다.
- 2개의 정규(유형 *f*) 파일이 디스크(*dk*) 매체에서 아카이브됨(A): *DISKVOL1* 볼륨의 아카이브 파일 *DISKVOL1/f2*에 *spcfiles/spcC4* 및 *spcfiles/spcC5*가 있습니다.
- 1개의 세 부분으로 세그먼트화된(유형 *s*) 파일이 LTO(*li*) 매체에서 아카이브됨: *bf/dat011*이며, 두 세그먼트는 *VOL013* 볼륨의 *0x76a* 위치에서 시작하고 한 세그먼트는 *1409aa4* 위치에서 시작합니다. 세그먼트 /1은 10485760바이트, 세그먼트 /2는 10485622바이트, 그리고 세그먼트 /3은 184바이트 길이입니다.
- 1개의 정규(유형 *f*) 볼륨 오버플로우 파일이 LTO(*li*) 매체에서 아카이브됨(A): *rf/rf81*이며, *VOL036* 볼륨의 *0x12d* 위치에서 시작하고 *VOL034* 볼륨의 *0x15f* 위치에서 계속됩니다.

아카이버 로그 항목의 필드에 대한 자세한 설명은 [부록 A. 아카이버 로그 이해 \[63\]](#)를 참조하십시오.

5. 복구 지점 파일이 만들어진 이후 만들어진 백업 아카이브 로그를 사용하여 검색을 반복합니다.

아카이버 로그는 자주 교체됩니다. 따라서 여러 아카이버 로그 복사본을 보존하고 있을 경우 현재 아카이버 로그 기간 이전에 만들어진 아카이브 복사본을 사용하여 손상된 파일을 복구할 수 있습니다.

6. 이제 [손상 및/또는 누락된 파일 복원 \[36\]](#)을 수행합니다.

4.2.1.5. 손상 및/또는 누락된 파일 복원

매체 볼륨 및 매체에서 아카이브(*tar*) 파일의 위치를 안다면 누락되거나 손상된 파일 복원은 *tar* 파일에 액세스하고 필요한 데이터 파일을 추출하기만 하면 가능합니다. 아카이브 파일이 아카이브 디스크 장치에 상주하는 경우 *tar* 파일이 파일 시스템 마운트 지점 아래에서 랜덤 액세스 가능한 디렉토리에 상주하므로 간단합니다. 하지만 *tar* 파일이 테이프와 같은 대용량 순차 액세스 매체에 상주하는 경우 복잡해집니다. 대개 아카이브 파일이 랜덤 액세스 디스크 장치에 스테이징될 때까지 아카이브 파일에서 필요한 데이터 파일을 추출할 수 없습니다. 아카이브 파일은 클 수 있으므로 이 작업은 복구 상황에서 시간이 오래 걸리고 까다로울 수 있습니다. 따라서 아래의 절차에서는 아카이브 파일을 메모리로 읽고 디스크에서 읽는 것처럼 사용 가능하게 만드는 Oracle HSM 명령 *request*를 사용합니다.

가능한 많은 손상 및 누락된 정규 파일을 복원하십시오. 각 파일에 대해 다음과 같이 수행하십시오.

1. 볼륨에 걸쳐 있지 않은 정규 파일 복구를 시작합니다. [5.2.1절. “손실 및 손상된 정규 파일 복원” \[43\]](#) 절차를 수행합니다.
2. 그런 다음 세그먼트화된 파일을 복구합니다. [5.2.2절. “손실 및 손상된 세그먼트화된 파일 복원” \[46\]](#) 절차를 수행합니다.
3. 그런 다음 볼륨에 걸쳐 있는 정규 파일을 복원합니다. [5.2.3절. “손실 및 손상된 볼륨 오버플로우 파일 복원” \[51\]](#) 절차를 수행합니다.
4. 복사본이 있으며 누락되고 손상된 파일을 모두 복원한 후 *wait* 지시어를 *archiver.cmd* 파일에서 제거하여 아카이브를 다시 사용으로 설정합니다. *-ignore* 매개변수를 *recycler.cmd* 파일에서 제거하여 재활용을 다시 사용으로 설정합니다.

파일 시스템이 가능한 원래 조건과 가까워졌습니다. 여전히 손상되거나 누락된 파일은 복구할 수 없습니다.

5. 복사본이 있는 누락 파일과 손상된 파일을 모두 복원한 경우 [6.1절. “아카이브 파일 시스템을 일반 작업으로 복원” \[57\]](#)으로 이동합니다.

4.2.2. 복구 지점 파일 없이 아카이브 매체에서 파일 및 디렉토리 복원

복구 지점 파일을 사용하지 않고 아카이브 매체에서 직접 파일 시스템을 복구할 수 있습니다. 다음과 같이 하십시오.

1. 옵티컬 매체에서 파일을 복원하려는 경우 여기서 중지하고 오라클 고객 지원 센터로 문의하십시오.
2. 파일 시스템에 대한 NFS(네트워크 파일 시스템) 공유를 사용 안함으로 설정합니다.
3. 아카이브 및 재활용을 사용 안함으로 설정합니다. [2.1절. “아카이브 및 재활용 프로세스 중지” \[11\]](#)에 설명된 방법을 사용합니다.

4. 복구 프로세스의 배타적 사용을 위해 테이프 드라이브를 예약합니다. `samcmd unavail drive-equipment-number` 명령을 사용합니다. 여기서 `drive-equipment-number`는 `/etc/opt/SUNWsamfs/mcf` 파일의 드라이브에 지정된 장비 서수 번호입니다.

`samcmd unavail` 명령은 아카이브, 스테이징 및 릴리스 프로세스에서 드라이브를 사용하지 못하도록 합니다. 이 예에서는 `804` 드라이브를 예약합니다.

```
root@solaris:~# samcmd unavail 804
root@solaris:~#
```

5. `/opt/SUNWsamfs/examples/tarback.sh` 파일을 `/tmp`와 같은 대체 위치에 복사합니다.

`tarback.sh` 파일은 지정된 매체 볼륨 세트에서 파일을 복원하는 실행 스크립트입니다. 스크립트는 각 볼륨의 각 아카이브(`tar`) 파일에 대해 `star -n` 명령을 실행합니다. 테이프의 백업 복사본에 파일 시스템의 해당하는 파일이 없거나 복사본이 파일 시스템의 해당하는 파일보다 최신인 경우 `star -n`은 복사본을 복원합니다.

이 예에서는 스크립트를 `/tmp`에 복사합니다.

```
root@solaris:~# cp /opt/SUNWsamfs/examples/tarback.sh /tmp/tarback.sh
root@solaris:~#
```

6. 텍스트 편집기에서 `tarback.sh` 파일의 복사본을 엽니다.

이 예에서는 `vi` 편집기를 사용합니다.

```
root@solaris:~# vi /opt/SUNWsamfs/examples/tarback.sh
#!/bin/sh
# script to reload files from SAMFS archive tapes
STAR="/opt/SUNWsamfs/sbin/star"
LOAD="/opt/SUNWsamfs/sbin/load"
UNLOAD="/opt/SUNWsamfs/sbin/unload"
EQ=28
TAPEDRIVE="/dev/rmt/3cbn"
# BLOCKSIZE is in units of 512 bytes (e.g. 256 for 128K)
BLOCKSIZE=256
MEDIATYPE="lt"
VSN_LIST="VSNA VSNB VSNC VSNZ"
...
```

7. Oracle HSM 유틸리티 `star`, `load` 및 `unload`가 비표준 위치에 설치된 경우 `tarback.sh` 파일의 복사본에서 기본 명령 경로를 편집합니다.

이 예에서는 모든 유틸리티가 기본 위치에 설치되어 있으므로 편집이 필요하지 않습니다.

```
root@solaris:~# vi /opt/SUNWsamfs/examples/tarback.sh
#!/bin/sh
# script to reload files from SAMFS archive tapes
STAR="/opt/SUNWsamfs/sbin/star"
LOAD="/opt/SUNWsamfs/sbin/load"
UNLOAD="/opt/SUNWsamfs/sbin/unload"
...
```

8. *tarback.sh* 파일 복사본에서 변수 *EQ*를 찾습니다. 복구용으로 예약한 드라이브의 장비 서수 번호로 값을 설정합니다.

이 예에서는 *EQ=804*를 설정합니다.

```
root@solaris:~# vi /opt/SUNWsamfs/examples/tarback.sh
#!/bin/sh
# script to reload files from SAMFS archive tapes
STAR="/opt/SUNWsamfs/sbin/star"
LOAD="/opt/SUNWsamfs/sbin/load"
UNLOAD="/opt/SUNWsamfs/sbin/unload"
EQ=804
...
```

9. *tarback.sh* 파일 복사본에서 변수 *TAPEDRIVE*를 찾습니다. 장치에 대한 원시 경로(큰 따옴표로 묶음)로 값을 설정합니다.

이 예에서 *804* 장치에 대한 원시 경로는 */dev/rmt/3cbn*입니다.

```
root@solaris:~# vi /opt/SUNWsamfs/examples/tarback.sh
#!/bin/sh
# script to reload files from SAMFS archive tapes
STAR="/opt/SUNWsamfs/sbin/star"
LOAD="/opt/SUNWsamfs/sbin/load"
UNLOAD="/opt/SUNWsamfs/sbin/unload"
EQ=804
TAPEDRIVE="/dev/rmt/3cbn"
...
```

10. *tarback.sh* 파일 복사본에서 변수 *BLOCKSIZE*를 찾습니다. 원하는 블록 크기의 512바이트 단위 숫자로 값을 설정합니다.

이 예에서는 LTO-4 드라이브에 대해 256KB 세그먼트 크기를 원합니다. 따라서 *512*를 지정합니다.

```
LOAD="/opt/SUNWsamfs/sbin/load"
UNLOAD="/opt/SUNWsamfs/sbin/unload"
EQ=804
BLOCKSIZE=512
```

```
TAPEDRIVE="/dev/rmt/3cbn"
BLOCKSIZE=512
...
```

11. *tarback.sh* 파일 복사본에서 변수 *MEDIATYPE*을 찾습니다. [부록 B. 장비 유형 용어 집 \[65\]](#)에서 드라이브가 지원하는 매체 유형에 대해 나열되는 2자리 문자의 매체 유형 코드로 값을 설정합니다. 매체 유형은 큰 따옴표로 묶습니다.

이 예에서는 LTO-4 드라이브를 사용합니다. 따라서 *li*를 지정합니다.

```
EQ=804
TAPEDRIVE="/dev/rmt/3cbn"
BLOCKSIZE=512
MEDIATYPE="li"
...
```

12. *tarback.sh* 파일 복사본에서 변수 *VSN_LIST*를 찾습니다. 값으로 파일의 백업 복사본을 포함할 수 있는 테이프를 식별하는 VSN(볼륨 일련 번호)의 공백으로 구분된 목록을 제공합니다. 목록은 큰 따옴표로 묶습니다.

이 예에서는 *VOL002*, *VOL003*, *VOL004*, *VOL013*, *VOL034* 및 *VOL036* 볼륨을 지정합니다.

```
EQ=804
TAPEDRIVE="/dev/rmt/3cbn"
BLOCKSIZE=512
MEDIATYPE="lt"
VSN_LIST="VOL002 VOL003 VOL004 VOL013 VOL034 VOL036"
...
```

13. *tarback.sh* 파일 복사본을 저장합니다. 편집기를 닫습니다.

```
EQ=804
TAPEDRIVE="/dev/rmt/3cbn"
BLOCKSIZE=512
MEDIATYPE="lt"
VSN_LIST="VOL002 VOL003 VOL004 VOL013 VOL034 VOL036"
...
:wq
root@solaris:~#
```

14. */tmp/tarback.sh* 스크립트를 실행합니다.

```
root@solaris:~# /tmp/tarback.sh
```

15. 필요에 따라 사용자 및 그룹 소유권, 모드, 확장 속성 및 ACL(액세스 제어 목록)을 복원된 파일마다 다시 생성합니다.

/tmp/tarback.sh 스크립트로는 이 유형의 메타 데이터를 복원할 수 없습니다.

16. */tmp/tarback.sh* 스크립트를 실행했으며 파일 복구를 완료한 경우 [6.1절. “아카이브 파일 시스템을 일반 작업으로 복원” \[57\]](#)으로 이동합니다.

손실 및 손상된 파일 복구

이 장에서는 개별 파일을 파일 시스템에 복원하기 위한 절차를 설명합니다. 다음 항목을 다룹니다.

- 복구 지점 파일을 사용하여 파일 복구 [41]
- 아카이브 로그 항목을 사용하여 파일 복구 [43]
- 손상된 아카이브 복사본 복구 [53].

5.1. 복구 지점 파일을 사용하여 파일 복구

복구 지점 파일은 손실되거나 손상된 파일을 복구하는 가장 빠르고 완전하고 신뢰할 수 있으며 가장 덜 수고로운 방법입니다. 따라서 복구 지점 파일이 사용 가능한 경우 다음과 같이 수행하십시오.

1. 파일 시스템 메타 데이터 서버에 *root*로 로그인합니다.

```
root@solaris:~#
```

2. 아직 그렇게 하지 않은 경우 2.1절. “아카이브 및 재활용 프로세스 중지” [11]의 절차를 사용하여 아카이브 및 재활용을 중지합니다
3. 대상 파일 시스템에서 복구된 파일을 보관할 임시 복구 디렉토리를 만듭니다.

이 예에서는 다시 만든 파일 시스템인 */samqfs1*에 대한 마운트 지점 아래에 임시 디렉토리 *restore*를 만듭니다.

```
root@solaris:~# mkdir /samqfs1/restore
```

4. 아카이버가 임시 디렉토리에서 아카이브하지 않도록 합니다. *archive -r -n directory* 명령을 사용합니다. 설명:
 - *-r -n*은 지정된 디렉토리 안이나 아래에 있는 파일의 아카이빙을 재귀적으로 사용 안 함으로 설정합니다.
 - *directory*는 임시 복구 디렉토리의 경로 및 디렉토리 이름입니다.

```
root@solaris:~# archive -r -n /samqfs1/restore
```

5. 임시 복구 디렉토리로 변경합니다.

```
root@solaris:~# cd /samqfs1/restore
```

6. 사용 가능한 최근 복구 지점 파일을 찾습니다.

이 예에서는 독립된 파일 시스템 `/zfs1`에서 잘 알려진 위치인 `samqfs1_recovery` 하위 디렉토리에 파일 시스템 `samqfs1`에 대한 날짜별 복구 지점 파일을 만들었습니다. 따라서 최신 파일 `20150324`를 쉽게 찾을 수 있습니다.

```
root@solaris:~# dir /zfs1/samqfs1_recovery/
20150321    20150322    20150323    20150324
root@solaris:~#
```

7. 복구해야 하는 파일이 복구 지점 파일에 있는지 확인합니다. `samfsrestore -t -f recovery-point` 명령의 출력에서 필요한 파일을 검색합니다. 설명:

- `-t`는 목차를 표시합니다.
- `-f recovery-point-file`은 선택한 복구 지점 파일의 경로와 이름을 지정합니다.

이 예에서는 `genw445` 파일을 복구하려고 합니다. 따라서 `samfsrestore -t` 명령을 복구 지점 파일 `/zfs1/samqfs1_recovery/20150324`와 함께 실행합니다. 검색을 단순화하기 위해 `samfsrestore -t`의 출력을 Solaris `grep` 명령 및 정규 표현식 `"genw445"`에 연결합니다(아래 명령은 한 라인으로 입력하며 줄바꿈이 백슬래시로 이스케이프되어 있음).

```
root@solaris:~# samfsrestore -t -f /zfs1/samqfs1_recovery/20150324 | /
grep "genw445"
./genfiles/genw445
root@solaris:~#
```

8. 파일의 inode 정보를 현재 디렉토리에 복원합니다. `samfsrestore -f recovery-point file` 명령을 사용합니다. 설명:

- `-f recovery-point-file`은 선택한 복구 지점 파일의 경로와 이름을 지정합니다.
- `file`은 복구 지점 파일이 복구하고자 하는 파일에 대해 나열하는 정확한 경로와 이름을 지정합니다.

이 예에서는 `./genfiles/genw445`를 복구 지점 파일 `/zfs1/samqfs1_recovery/20150324`에서 복구합니다(아래 명령은 한 라인으로 입력하며 줄바꿈이 백슬래시로 이스케이프되어 있음).

```
root@solaris:~# samfsrestore -f /zfs1/samqfs1_recovery/20150324 /
./genfiles/genw445
root@solaris:~#
```

9. 파일이 올바르게 복원되었는지 확인합니다. `sls -D file` 명령을 사용합니다. 여기서 `file`은 임시 복구 디렉토리에 대한 파일의 상대 경로와 이름을 지정합니다.

이 예에서는 `genfiles/genw445` 파일이 임시 디렉토리 `/samqfs1/restore/`로 복구되었습니다.

```

root@solaris:~# sls -D genfiles/genw445
genfiles/genw445:
  mode: -rw-r--r--   links:   1  owner: data           group: samqfs1
  length:   14975  inode:   25739.1
offline; archdone;
copy 1: ---- Mar  4 11:55 8ae.1 xt 000000
copy 2: ---- Mar  4 15:51 cd3.7f57 xt 000000
  access:      Mar  4 11:55  modification: Mar  4 21:50
  changed:      Mar  4 11:50  attributes:   Mar  4 21:50
  creation:      Mar  4 11:50  residence:    Mar  4 21:50
root@solaris:~#

```

10. 파일이 올바르게 복원되면 파일 시스템에서 올바른 위치로 이동합니다.

이 예에서는 *genw445* 파일을 임시 작업 디렉토리 */samqfs1/restore/genfiles/*에서 */samqfs1/genfiles/* 안의 원래 위치로 이동합니다.

```

root@solaris:~# mv -f genfiles/genw445 /samqfs1/genfiles/genw445
root@solaris:~#

```

11. 모든 누락된 파일이 복구될 때까지 이 절차를 반복합니다.

12. 복구 절차를 완료합니다. [6.1절. “아카이브 파일 시스템을 일반 작업으로 복원” \[57\]](#)으로 이동합니다.

5.2. 아카이브 로그 항목을 사용하여 파일 복구

일정한 개수 이상의 파일을 복구할 때 아카이브 로그를 사용하는 것은 언제나 번거롭고 노력이 많이 드는 프로세스입니다. 따라서 가급적이면 복구 지점으로 필요한 파일을 복원할 수 없는 경우에만 이 절의 절차를 사용하십시오.

아카이브 매체에서 파일을 복구하는 프로세스는 기본적으로 모든 경우에 동일하지만 파일 유형에 따라 세부 사항이 달라질 수 있습니다. 따라서 복원하는 파일 유형에 알맞은 절차를 선택하십시오.

- [손실 및 손상된 정규 파일 복원 \[43\]](#)
- [손실 및 손상된 세그먼트화된 파일 복원 \[46\]](#)
- [손실 및 손상된 볼륨 오버플로우 파일 복원 \[51\]](#).

매체에서 복사본을 복구할 때 예상하는 정확한 위치로 파일을 복원하지 못할 수 있습니다. 파일은 아카이브 복사본이 만들어진 시점의 해당 위치로 복원됩니다. 따라서 이후에 이동된 파일은 손실된 당시에 있었던 디렉토리로 복원되지 않습니다.

5.2.1. 손실 및 손상된 정규 파일 복원

복구해야 하는 각 파일에 대해 다음과 같이 수행하십시오.

1. 아직 그렇게 하지 않은 경우 파일 시스템 메타 데이터 서버에 *root*로 로그인합니다.

```
root@solaris:~#
```

2. 아직 그렇게 하지 않은 경우 [2.1절. “아카이브 및 재활용 프로세스 중지” \[11\]](#)의 절차를 사용하여 아카이브 및 재활용을 중지합니다
3. 복원하는 파일 시스템의 루트 디렉토리로 변경합니다.

Oracle HSM 아카이브 파일은 파일 시스템 루트 디렉토리에 상대적인 복사본을 저장합니다. 따라서 원래 위치로 복원하려면 루트 디렉토리에서 복원하는 것이 좋습니다.

이 예에서는 *samqfs1* 파일 시스템의 루트로 변경합니다.

```
root@solaris:~# cd /samqfs1
root@solaris:~#
```

4. 정규 파일이 마지막으로 아카이브된 기간 동안의 아카이버 로그를 가지고 있는 경우 파일에 대한 최근 항목을 찾습니다.

첫번째 예에서는 정규(유형 *f*) 파일 *genA0*의 항목을 찾습니다.

```
A 2015/03/03 13:09:05 li VOL004 all.1 212.1 samqfs1 1535.2 1971 genfiles/genA0 f 0 0
```

두번째 예에서는 정규(유형 *f*) 파일 *spcC4*의 항목을 찾습니다.

```
A 2015/03/03 21:49:15 dk DISKVOL1/f2 all.1 2.2e9 samqfs1 1511.2 8971
socfiles/spcC4 f 0 0
```

5. 필요한 파일의 로그 항목을 찾으면 매체 유형, 매체의 볼륨 일련 번호 및 파일 시스템의 루트 디렉토리에 대한 파일의 상대 경로와 이름을 메모합니다.

첫번째 예에서는 *genA0* 파일이 LTO(*li*) 테이프 볼륨(VSN(볼륨 일련 번호)이 *VOL004*)에 있습니다. 이 파일은 원래 파일 시스템 디렉토리 */samqfs1/genfiles/*에 저장되었습니다.

```
A 2015/03/03 13:09:05 li VOL004 all.1 212.1 samqfs1 1535.2 1971 genfiles/genA0 f 0 0
```

두번째 예에서는 *spcC4* 파일이 디스크 아카이브(*dk*)(볼륨 일련 번호가 *DISKVOL1*)에 있습니다. 이 파일은 원래 파일 시스템 디렉토리 */samqfs1/socfiles/*에 저장되었습니다.

```
A 2015/03/03 21:49:15 dk DISKVOL1/f2 all.1 2.2e9 samqfs1 1511.2 8971 socfiles/spcC4
f 0 0
```

6. 필요한 파일이 마그네틱 테이프와 같이 순차 액세스 매체에 있는 경우에도 아카이브 (*tar*) 파일의 시작 위치를 나타내는 16진수 값을 메모합니다.

이 예에서는 *genA0* 파일이 0x212(212) 위치에서 시작하는 테이프에 있습니다.

```
A 2015/03/03 13:09:05 li VOL004 all.1 212.1 samqfs1 1535.2 1971 genfiles/genA0 f 0
0
```

7. 필요한 파일이 아카이브 디스크와 같이 랜덤 액세스 매체에 저장된 경우 볼륨 일련 번호에 대한 *tar* 파일의 상대 경로와 파일 이름을 메모합니다.

이 예에서는 디스크 볼륨이 *DISKVOL1*인 루트 디렉토리 바로 아래의 *f2* 하위 디렉토리에 *spcC4* 파일이 있습니다.

```
A 2015/03/03 21:49:15 dk DISKVOL1/f2 all.1 2.2e9 samqfs1 1511.2 8971 socfiles/
spcC4 f 0 0
```

8. 복원하는 파일이 디스크 매체에 아카이브된 경우 누락되거나 손상된 파일의 아카이브 복사본을 디스크 볼륨의 *tar* 파일에서 추출합니다. *star -xv -f tarfile file* 명령을 사용합니다. 설명:

- *tarfile*은 아카이브 파일 이름입니다.
- *file*은 복원해야 하는 파일의 파일 시스템 루트 디렉토리에 대한 상대 경로와 이름입니다.

star 명령은 지정된 파일을 아카이브 파일에서 복원하는 GNU *tar*의 향상된 Oracle HSM 버전입니다.

이 예에서는 데이터 파일 *socfiles/spcC4*를 *tar* 파일 *DISKVOL1/f2*에서 추출합니다. 파일이 */samqfs1/socfiles/spcC4*에 복원됩니다.

```
root@solaris:~# star -xvf DISKVOL1/f2 socfiles/spcC4
```

9. 필요한 파일을 디스크 아카이브에서 복원한 경우 필요한 파일이 모두 복원될 때까지 계속 [손실 및 손상된 정규 파일 복원 \[43\]](#)합니다.
10. 복원하는 파일이 마그네틱 테이프와 같이 이동식 매체에 아카이브된 경우 복원된 파일 시스템에서 임시 아카이브 파일을 보관할 디렉토리를 만듭니다.

이 예에서는 */samqfs1/tars* 디렉토리를 만듭니다.

```
root@solaris:~# mkdir /samqfs1/tars
```

11. 아카이브된 복사본을 보관하는 아카이브 파일에 대한 *tar* 헤더의 시작 위치에 매체를 두고 매체에서 메모리로 아카이브를 읽습니다. *request -m media-type -v volume-serial-number -p 0xposition path/requestfile* 명령을 사용합니다. 여기서
- *-m media-type*은 [부록 B. 장비 유형 용어집 \[65\]](#)에 나열된 2자리 매체 유형 코드 중 하나를 지정합니다.

- `-v volume-serial-number`는 매체 볼륨을 식별하는 6자리 영숫자 코드를 지정합니다.
- `-p 0xposition`은 아카이버 로그 항목에서 메모한 16진수 시작 위치를 지정합니다.
- `path`는 임시 복구 디렉토리에 대한 경로입니다.
- `requestfile`은 `request` 명령이 매체에서 읽는 인 메모리 `tar` 파일에 사용할 이름입니다.

이 예에서는 LTO(1i) 볼륨 `VOL012`의 `0x78` 위치에서 시작되는 요청 파일 `/samqfs1/tars/currentrequest`를 생성합니다.

```
root@solaris:~# request -m li -v VOL012 -p 0x78 /samqfs1/tars/currentrequest
```

12. 누락되거나 손상된 파일의 아카이브 복사본을 이전 단계에서 만든 인 메모리 `tar` 파일에서 추출합니다. `star -xv -f requestfile` 명령을 사용합니다. 설명:

- `requestfile`은 인 메모리 `tar` 파일의 이름입니다.
- `file`은 복원해야 하는 파일의 파일 시스템 루트 디렉토리에 대한 상대 경로와 이름입니다.

`star` 명령은 요청 파일(아카이브 파일의 인 메모리 복사본)에서 지정된 파일을 복원하는 GNU `tar`의 향상된 Oracle HSM 버전입니다.

이 예에서는 데이터 파일 `genfiles/genA0`을 요청 파일 `tars/currentrequest`에서 추출합니다. 파일이 `/samqfs1/genfiles/genA0`에 복원됩니다.

```
root@solaris:~# star -xvf tars/currentrequest genfiles/genA0
```

13. 필요한 파일 속성을 설정합니다.

`samfsdump` 또는 `qfsdump` 복구 지점 파일을 사용하지 않고 `tar` 파일에서 파일을 복원할 경우 원본 파일 속성이 손실됩니다. 기본 속성값을 사용하여 해당 파일에 대해 `.inodes` 파일을 처음부터 만들어야 합니다.

14. 이 절차를 반복합니다. 필요한 파일이 모두 복구될 때까지 [손실 및 손상된 정규 파일 복원 \[43\]](#)하십시오.
15. 필요할 경우 [손실 및 손상된 세그먼트화된 파일 복원 \[46\]](#)하거나 [손실 및 손상된 볼륨 오버플로우 파일 복원 \[51\]](#)합니다.
16. 그렇지 않으면 복구 절차를 마칩니다. [6.1절. “아카이브 파일 시스템을 일반 작업으로 복원” \[57\]](#)으로 이동합니다.

5.2.2. 손실 및 손상된 세그먼트화된 파일 복원

세그먼트화된 파일 복원은 정규 파일 복원과 상당히 유사합니다. 하지만 파일 자체보다 개별 세그먼트를 복구하게 됩니다. 따라서 파일을 복원하려면 세그먼트를 단일 파일로 리어셈블링한 다음 결과를 다시 세그먼트화해야 합니다. 복구해야 하는 각 파일에 대해 다음과 같이 수행하십시오.

1. 아직 그렇게 하지 않은 경우 파일 시스템 메타 데이터 서버에 `root`로 로그인합니다.

```
root@solaris:~#
```

2. 아직 그렇게 하지 않은 경우 2.1절. “아카이브 및 재활용 프로세스 중지” [11]의 절차를 사용하여 아카이브 및 재활용을 중지합니다.
3. 세그먼트화된 파일이 마지막으로 아카이브된 기간 동안의 아카이버 로그가 있을 경우 세그먼트화된(유형 s) 파일에 대한 항목을 검색합니다. 필요한 파일의 세그먼트에 대한 최신 항목을 선택합니다.

```
A 2015/03/03 14:01:47 li VOL013 all.1 76a.1 samqfs1 14.5 10485760 bf/dat011/1 s 0 51
A 2015/03/03 14:04:11 li VOL013 all.1 2476f.5002 samqfs1 15.5 10485760 bf/dat011/2 s 0 51
A 2015/03/03 14:06:24 li VOL013 all.1 1409aa4.1 samqfs1 16.5 184 bf/dat011/3 s 0 51
```

4. 세그먼트에 대한 최신 항목을 찾은 후에는 다음 세부 정보를 메모합니다.
 - 매체 유형
 - 파일을 저장하는 매체 볼륨의 볼륨 일련 번호
 - 세그먼트가 포함된 아카이브(*tar*) 파일의 16진수 시작 위치
 - 파일 시스템의 루트 디렉토리에 대한 세그먼트화된 파일의 상대 경로 및 이름
 - 파일에 있는 세그먼트 수

이 예에서는 *dat011* 파일이 3개의 세그먼트(1, 2 및 3)로 나뉩니다. 3개의 세그먼트는 모두 하나의 LTO(*li*) 테이프 볼륨(볼륨 일련 번호 *vol013*)에 있는 3개의 아카이브 파일에 저장됩니다. 3개의 아카이브 파일은 0x76a(76a), 0x2476f(2476f) 및 0x1409aa4(1409aa4) 위치에서 시작됩니다.

```
A 2015/03/03 14:01:47 li VOL013 all.1 76a.1 samqfs1 14.5 10485760 bf/dat011/1 s 0 51
A 2015/03/03 14:04:11 li VOL013 all.1 2476f.5002 samqfs1 15.5 10485760 bf/dat011/2 s 0 51
A 2015/03/03 14:06:24 li VOL013 all.1 1409aa4.1 samqfs1 16.5 184 bf/dat011/3 s 0 51
```

5. 복원하는 파일 시스템의 루트 디렉토리로 변경합니다.

Oracle HSM 아카이브 파일은 파일 시스템 루트 디렉토리에 상대적인 복사본을 저장합니다. 따라서 원래 위치로 복원하려면 루트 디렉토리에서 복원하는 것이 좋습니다.

이 예에서는 *samqfs1* 파일 시스템의 루트로 변경합니다.

```
root@solaris:~# cd /samqfs1
```

6. 복원된 파일 시스템에서 임시 아카이브 파일을 보관할 디렉토리를 만듭니다.

이 예에서는 */samqfs1/tars* 디렉토리를 만듭니다.

```
root@solaris:~# mkdir /samqfs1/tars
```

7. 하나 이상의 파일 세그먼트의 아카이브된 복사본을 보관하는 각 아카이브 파일의 시작 위치에 매체를 두고 매체에서 메모리로 아카이브를 읽습니다. `request -m media-type -v volume-serial-number -p 0xposition path/requestfile` 명령을 사용합니다. 여기서
 - `-m media-type`은 **부록 B. 장비 유형 용어집 [65]**에 나열된 2자리 매체 유형 코드 중 하나를 지정합니다.
 - `-v volume-serial-number`는 매체 볼륨을 식별하는 6자리 영숫자 코드를 지정합니다.
 - `-p 0xposition`은 아카이버 로그 항목에서 메모한 16진수 시작 위치를 지정합니다.
 - `path`는 임시 복구 디렉토리에 대한 경로입니다.
 - `requestfile`은 `request` 명령이 매체에서 읽는 인 메모리 `tar` 파일에 사용할 이름입니다.

이 예에서는 두 개의 요청 파일을 만들어야 합니다. 첫번째 요청 파일인 `/samqfs1/tars/request76a`는 LTO(1i) `VOL013`의 `0x76a` 위치에서 시작되는 아카이브 파일을 로드합니다. 이 아카이브에는 처음 두 세그먼트가 모두 포함됩니다. 두번째 요청 파일인 `/samqfs1/tars/request1409aa4`는 이 경우 동일한 볼륨(세그먼트는 라이브러리의 어느 볼륨에나 상주할 수 있음)에 있는 `0x1409aa4` 위치에서 아카이브 파일을 로드합니다.

```
root@solaris:~# request -m li -v VOL013 -p 0x76a /samqfs1/tars/request76a
root@solaris:~# request -m li -v VOL013 -p 0x1409aa4 /
/samqfs1/tars/request1409aa4
```

8. 누락되거나 손상된 파일에 대한 백업 복사본의 각 세그먼트를 이전 단계에서 만든 인 메모리 `tar` 파일에서 추출합니다. `star -xv -f requestfile segment` 명령을 사용합니다. 여기서 `requestfile`은 인 메모리 `tar` 파일의 이름이고 `segment`는 복원해야 하는 파일에 대한 각 세그먼트의 파일 시스템 루트 디렉토리에 대한 상대 경로와 이름입니다.

`star` 명령은 요청 파일을 사용하여 가리키는 아카이브 파일에서 지정된 파일을 복원하는 GNU `tar`의 향상된 Oracle HSM 버전입니다.

이 예에서는 데이터 파일 `bf/dat011`의 세 세그먼트 중 두 개를 요청 파일 (인 메모리 `tar` 파일) `tars/request76a`에서 추출하고, 한 개를 요청 파일 `tars/request1409aa4`에서 추출합니다. 파일이 세 개의 분리된 조각으로 `/samqfs1/bf/dat011/` 디렉토리에 복원됩니다.

```
root@solaris:~# star -xvf tars/request76a bf/dat011/1
root@solaris:~# star -xvf tars/request76a bf/dat011/2
root@solaris:~# star -xvf tars/request1409aa4 bf/dat011/3
```

`/samqfs1/bf/dat011`의 내용을 나열하면 각 복원된 세그먼트에 대해 순차적으로 번호가 지정된 하나의 파일을 보게 됩니다.


```

root@solaris:~# ls /samqfs/bf/dat011
total 40968
-rw-rw---- 1 root other 10485760 Mar  5 17:06 1
-rw-rw---- 1 root other 10485760 Mar  5 17:06 2
-rw-rw---- 1 root other      184 Mar  5 17:07 3
root@solaris:~#

```

9. 복원된 세그먼트를 하나의 세그먼트화되지 않은 임시 파일로 리어셈블링합니다.

이 예에서는 `/samqfs1/bf/dat011/` 디렉토리의 세 세그먼트를 연결하여 `/samqfs1/bf/dat011file` 파일을 만듭니다.

```

root@solaris:~# cat /samqfs/bf/dat011/1 /samqfs/bf/dat011/2 /
/samqfs/bf/dat011/3 > /samqfs/bf/dat011file
root@solaris:~#

```

`/samqfs1/bf/dat011`의 내용을 나열하면 세그먼트가 포함된 디렉토리과 함께 새로운 파일이 나타납니다.

```

root@solaris:~# ls -l /samqfs/bf/dat011*
drwxr-xr-x 2 root root      4096 Mar  5 17:06 dat011
-rw-rw---- 1 root other 20971704 Mar  5 17:14 dat011file
root@solaris:~#

```

10. 세그먼트 및 해당 세그먼트가 포함된 디렉토리를 제거합니다.

```

root@solaris:~# rm -r /samqfs/bf/dat011/
root@solaris:~#

```

11. 세그먼트화된 파일의 원래 경로와 이름을 사용하여 빈 파일을 만듭니다. `touch file` 명령을 사용합니다. 여기서 `file`은 원래 경로와 파일 이름입니다.

이 예에서는 복원하는 세그먼트화된 파일의 원래 이름인 빈 파일 `/samqfs/bf/dat011`을 만듭니다.

```

root@solaris:~# touch /samqfs/bf/dat011
root@solaris:~#

```

12. 새로 만든 빈 파일에서 Oracle HSM 세그먼트 속성을 설정합니다. `segment -l segment-length file` 명령을 사용합니다. 여기서 `segment-length`는 아카이버 로그 항목에서 메모한 세그먼트 길이이고 `file`은 세그먼트화된 파일의 원래 경로와 이름입니다.

이 예에서는 아카이버 로그가 `dat011` 파일에 대한 세그먼트 길이를 `10485760`(파일이 세번째 세그먼트에서 끝나므로 매체의 데이터 길이가 세그먼트 길이보다 작음)으로 표시합니다.

```
A 2015/03/03 14:01:47 li VOL013 all.1 76a.1 samqfs1 14.5 10485760 bf/dat011/1 S 0 51
A 2015/03/03 14:04:11 li VOL013 all.1 76a.5002 samqfs1 15.5 10485760 bf/dat011/2 S 0
51
A 2015/03/03 14:06:24 li VOL013 all.1 1409aa4.1 samqfs1 16.5 184 bf/dat011/3 S 0 51
```

따라서 빈 파일에 대한 세그먼트 길이를 10485760으로 설정합니다.

```
root@solaris:~# segment -l 10485760 /samqfs/bf/dat011
root@solaris:~#
```

13. 세그먼트화되지 않은 임시 파일을 빈 세그먼트화된 파일에 복사합니다.

이 예에서는 *dat011file*을 *dat011*에 복사합니다.

```
root@solaris:~# cp /samqfs/bf/dat011file /samqfs/bf/dat011
root@solaris:~#
```

sls -2K samqfs/bf/dat011 명령을 사용하여 세그먼트를 나열하면 다음과 같이 나열됩니다. 따라서 파일이 복원되었습니다.

```
root@solaris:~# sls -2K /samqfs/bf/dat011
-rw-rw---- 1 root other      20971704    Mar  5 17:12 samqfs/bf/dat011
-----
----- SI {3,0,0,0}
-rw-rw---- 1 root other      10485760    Mar  5 17:12 samqfs/bf/dat011/1
-----
----- SS
-rw-rw---- 1 root other      10485760    Mar  5 17:12 samqfs/bf/dat011/2
-----
----- SS
-rw-rw---- 1 root other        184      Mar  5 17:12 samqfs/bf/dat011/3
-----
----- SS
```

14. 기타 필요한 파일 속성을 설정합니다.

samfsdump 또는 *qfsdump* 복구 지점 파일을 사용하지 않고 *tar* 파일에서 파일을 복원할 경우 원본 파일 속성이 손실됩니다. 기본 속성값을 사용하여 해당 파일에 대해 *.inodes* 파일을 처음부터 만들어야 합니다.

15. 이제 파일이 복원되었습니다. 세그먼트화되지 않은 임시 파일을 삭제합니다.

이 예에서는 *dat011file*을 삭제합니다.

```
root@solaris:~# rm /samqfs/bf/dat011file
root@solaris:~#
```

16. 모든 필요한 파일이 복구될 때까지 이 절차를 반복합니다.

17. 복구 절차를 완료합니다. 6.1절. “아카이브 파일 시스템을 일반 작업으로 복원” [57]으로 이동합니다.

5.2.3. 손실 및 손상된 볼륨 오버플로우 파일 복원

볼륨 오버플로우 파일은 매체 볼륨에 걸쳐 있는 정규 파일입니다. 따라서 볼륨 오버플로우 파일 복원은 다른 정규 파일 복원과 상당히 유사합니다. 하지만 아카이브에서 데이터 파일을 추출하기 전에 여러 볼륨에 상주하는 아카이브 파일의 섹션을 디스크의 단일 아카이브 파일로 합쳐야 합니다. 따라서 복구해야 하는 각 파일에 대해 다음과 같이 수행하십시오.

1. 아직 그렇게 하지 않은 경우 파일 시스템 메타 데이터 서버에 *root*로 로그인합니다.

```
root@solaris:~#
```

2. 아직 그렇게 하지 않은 경우 [2.1절. “아카이브 및 재활용 프로세스 중지” \[11\]](#)의 절차를 사용하여 아카이브 및 재활용을 중지합니다.
3. 볼륨 오버플로우 파일이 마지막으로 아카이브된 기간 동안의 아카이버 로그를 가지고 있는 경우 파일에 대한 최근 항목을 찾습니다. 매체의 볼륨 일련 번호, 파일에 대한 각 섹션의 길이, 파일 시스템의 루트 디렉토리에 대한 파일의 상대 경로와 이름 및 파일의 섹션 수를 메모합니다.

이 예에서는 */samqfs1/rf/rf81* 파일이 *VOL036* 및 *VOL034*의 두 볼륨에 상주하는 *f* 유형의 정규 파일이고 *0* 및 *1*의 두 섹션을 가지므로 볼륨 오버플로우임을 알 수 있습니다.

```
A 2015/03/03 18:28:51 li VOL036 all.1 12d.1 samqfs1 11731.1 89128448 rf/rf81 f 0
210
A 2013/08/23 18:28:51 li VOL034 all.1 15f.0 samqfs1 11731.1 525271552 rf/rf81 f 1
220
```

4. 복원하는 파일 시스템의 루트 디렉토리로 변경합니다.

Oracle HSM 아카이브 파일은 파일 시스템 루트 디렉토리에 상대적인 복사본을 저장합니다. 따라서 원래 위치로 복원하려면 루트 디렉토리에서 복원하는 것이 좋습니다.

이 예에서는 *samqfs1* 파일 시스템의 루트로 변경합니다.

```
root@solaris:~# cd /samqfs1
```

5. 진행하기 전에 파일 시스템에 복구하는 파일 크기의 두 배 이상의 파일을 수용할 만큼 충분한 여유 공간이 있는지 확인합니다.

이 예의 파일인 *rf/rf81*의 경우, 파일에 대한 두 섹션의 크기를 기준으로 약 1.2GB의 여유 공간이 필요합니다($2 \times (89128448 + 525271552) = 1228800000$ 바이트).

6. 복원된 파일 시스템에서 임시 아카이브 파일을 보관할 디렉토리를 만듭니다.

이 예에서는 */samqfs1/tars* 디렉토리를 만듭니다.

```
root@solaris:~# mkdir /samqfs1/tars
```

7. 하나 이상의 파일 세그먼트의 아카이브된 복사본을 보관하는 각 아카이브 파일의 시작 위치에 매체를 두고 매체에서 메모리로 아카이브를 읽습니다. `request -m media-type -v volume-serial-number -p 0xposition path/requestfile` 명령을 사용합니다. 여기서
- `-m media-type`은 [부록 B. 장비 유형 용어집 \[65\]](#)에 나열된 2자리 매체 유형 코드 중 하나를 지정합니다.
 - `-v volume-serial-number`는 매체 볼륨을 식별하는 6자리 영숫자 코드를 지정합니다.
 - `-p 0xposition`은 아카이버 로그 항목에서 메모한 16진수 시작 위치입니다.
 - `path`는 임시 복구 디렉토리에 대한 경로입니다.
 - `requestfile`은 `request` 명령이 매체에서 읽는 인 메모리 `tar` 파일에 사용할 이름입니다.

이 예에서는 두 개의 요청 파일을 만듭니다. 첫번째 요청 파일, `/samqfs1/tars/requestVOL036`은 LTO(*li*) `VOL036`의 `0x12d` 위치에서 시작하는 아카이브 파일을 로드합니다. 두번째 요청 파일인 `/samqfs1/tars/requestVOL034`는 LTO(*li*) `VOL034`의 `0x15f` 위치에서 시작되는 아카이브 파일을 로드합니다.

```
root@solaris:~# request -m li -v VOL036 -p 0x12d /samqfs1/tars/requestVOL036
root@solaris:~# request -m li -v VOL034 -p 0x15f /samqfs1/tars/requestVOL034
```

8. 만든 각 인 메모리 `tar` 파일을 아카이브 파일의 섹션으로 디스크에 저장합니다. `dd if=requestfile of=archive_section` 명령을 사용합니다. 여기서 `requestfile`은 인 메모리 `tar` 파일의 경로와 이름이고 `archive_section`은 아카이브 파일에 대한 각 섹션의 경로와 이름입니다.

이 예에서는 요청 파일(인 메모리 `tar` 파일), `tars/requestVOL036` 및 `tars/requestVOL034`를 `tars/archive_part1` 및 `tars/archive_part2`로 저장합니다.

```
root@solaris:~# dd if=tars/requestVOL036 of=tars/archive_part1
root@solaris:~# dd if=tars/requestVOL034 of=tars/archive_part2
root@solaris:~#
```

9. 섹션을 하나의 아카이브 파일로 리어셈블링합니다.

이 예에서는 두 섹션 `tars/archive_part1` 및 `tars/archive_part2`를 연결하여 하나의 아카이브 파일 `/tars/archive_complete`를 만듭니다.

```
root@solaris:~# cat tars/archive_part1 tars/archive_part2 > /
tars/archive_complete
root@solaris:~#
```

10. 누락되거나 손상된 볼륨 오버플로우 파일의 백업 복사본을 이전 단계에서 만든 아카이브(`tar`) 파일에서 추출합니다. `star -xv -f tarfile file` 명령을 사용합니다. 여기서

*tarfile*은 아카이브 파일의 이름이고 *file*은 복원해야 하는 볼륨 오버플로우 파일의 파일 시스템 루트 디렉토리에 대한 상대 경로와 이름입니다.

star 명령은 요청 파일을 사용하여 가리키는 아카이브 파일에서 지정된 파일을 복원하는 GNU *tar*의 향상된 Oracle HSM 버전입니다.

이 예에서는 볼륨 오버플로우 파일 *rf/rf81*을 *tar* 파일 *tars/archive_complete*에서 추출합니다.

```
root@solaris:~# star -xvf tars/archive_complete rf/rf81
```

11. 기타 필요한 파일 속성을 설정합니다.

samfsdump 또는 *qfsdump* 복구 지점 파일을 사용하지 않고 *tar* 파일에서 파일을 복원할 경우 원본 파일 속성이 손실됩니다. 기본 속성값을 사용하여 해당 파일에 대해 *.inodes* 파일을 처음부터 만들어야 합니다.

12. 이제 볼륨 오버플로우 파일이 복원되었습니다. 임시 파일을 삭제합니다.

이 예에서는 *dat011file*을 삭제합니다.

```
root@solaris:~# rm tars/archive_*
root@solaris:~#
```

13. 모든 필요한 파일이 복구될 때까지 이 절차를 반복합니다.
14. 복구 절차를 완료합니다. [6.1절. “아카이브 파일 시스템을 일반 작업으로 복원” \[57\]](#)으로 이동합니다.

5.3. 손상된 아카이브 복사본 복구

손상된 아카이브 복사본은 디스크 캐시로 다시 스테이징할 수 없는 파일의 복사본입니다. 때로는 간헐적인 하드웨어 관련 I/O 문제로 인해 파일이 단순히 스테이징을 실패하고 쉽게 해결할 수도 있습니다. 그렇지 않은 경우 손상된 복사본은 손상되고 데이터를 복구할 수 없습니다. 이러한 경우 유일한 옵션은 대체 복사본에서 파일을 복구하는 것입니다.

손상된 복사본을 식별하고 관리하려면 다음과 같이 수행하십시오.

1. 손상된 아카이브 복사본이 있는 파일을 식별합니다. *sfind mountpoint -any_copy_d* 명령을 사용합니다. 여기서 *mountpoint*는 복구된 파일 시스템이 마운트된 디렉토리입니다.

이 예에서는 */samqfs1* 디렉토리에서 검색을 시작하고 손상된 복사본이 있는 3개의 파일을 찾습니다.

```
root@solaris:~# sfind /samqfs1 -any_copy_d
./genfiles/ab09
./genfiles/ab11
./genfiles/ay12
```

```
root@solaris:~#
```

2. 찾은 각 파일에 대해 손상된 복사본을 식별합니다. `sls -D file` 명령을 사용합니다. 여기서 `file`은 확인할 경로와 파일 이름입니다.

손상된 복사본은 `D`로 플래그가 표시됩니다. 이 예에서는 `/samqfs1/genfiles/ab09`의 **copy 2** 및 `/samqfs1/genfiles/ab11`의 **copy 1**이 손상되었습니다.

```
root@solaris:~# sls -D /samqfs1/genfiles/ab09
/samqfs1/genfiles/ab09:
mode: -rw-r----- links: 1 owner: root group: other
length: 306581 admin id: 0 inode: 11748.11
project: system(0)
copy 1: ---- Mar 11 13:52 76f.421bc li VOL011
copy 2: ---D Mar 31 14:02 286.1324f li VOL021
access: Mar 11 13:50 modification: Mar 11 13:50
changed: Mar 11 13:50 attributes: Mar 11 13:50
creation: Mar 11 13:50 residence: Mar 11 13:50
root@solaris:~# sls -D /samqfs1/genfiles/ab11
/samqfs1/genfiles/ab11:
mode: -rw-r----- links: 1 owner: root group: other
length: 380051 admin id: 0 inode: 1460.1
project: system(0)
copy 1: ---D Mar 01 10:21 431.21bc6 li VOL024
access: Mar 01 10:21 modification: Mar 01 10:21
changed: Mar 01 10:21 attributes: Mar 01 10:21
creation: Mar 01 10:21 residence: Mar 01 10:21
root@solaris:~#
```

3. 대체 복사본이 있는 경우 손상된 복사본을 아카이브 해제합니다. `unarchive -c CopyNumber file` 명령을 사용합니다. 여기서 `CopyNumber`는 복사본 번호를 나타내는 정수이고 `file`은 손상된 파일의 경로와 파일 이름입니다. 여기에서 중지합니다.

손상된 복사본을 아카이브 해제할 경우 Oracle HSM는 나머지 복사본에서 스테이징하고 다음에 아카이버 프로세스가 실행될 때 추가 아카이브 복사본을 만듭니다. 이 예에서는 `/samqfs1/genfiles/ab09`의 다른 손상되지 않은 복사본이 있으므로 손상된 복사본인 **copy 2**를 아카이브 해제합니다.

```
root@solaris:~# unarchive -c 2 /samqfs1/genfiles/ab09
root@solaris:~#
```

4. 다른 복사본이 없는 경우 손상된 복사본을 손상 해제합니다. `undamage -c CopyNumber file` 명령을 사용합니다. 여기서 `CopyNumber`는 복사본 번호를 나타내는 정수이고 `file`은 손상된 파일의 경로와 파일 이름입니다.

때로는 간헐적인 하드웨어 관련 I/O 오류로 인해 파일이 스테이징을 실패합니다. 손상 플래그를 지우고 다시 스테이징하면 문제가 해결될 수 있습니다. 이 예에서는 **/samqfs1/genfiles/ab11**의 복사본이 하나만 있습니다.

```
root@solaris:~# undamage -c1 /samqfs1/genfiles/ab11
```

5. 복사본 스테이징을 시도합니다. `stage -c CopyNumber -I file` 명령을 사용합니다. 여기서 `CopyNumber`는 복사본 번호를 나타내는 정수이고 `file`은 파일의 경로와 파일 이름입니다.

선택적 `-I(immediate)` 매개변수는 스테이징 작업을 대기열의 맨 위로 올립니다.

```
root@solaris:~# stage -c 1 -I /samqfs1/genfiles/ab11
```

6. 스테이징을 성공할 경우 여기에서 중지합니다.
7. 스테이징을 다시 실패할 경우 Oracle HSM는 손상 플래그를 다시 설정합니다. 손상된 복사본에 대한 `sls -D` 명령 출력에서 주요 inode 번호를 메모합니다.

이 예에서는 **/samqfs1/genfiles/ab11** 파일의 inode 번호가 1460입니다.

```
root@solaris:~# sls -D /samqfs1/genfiles/ab11
/samqfs1/genfiles/ab11:
  mode: -rw-r-----  links:   1  owner: root group: other
  length:   380051  admin id: 0  inode:   1460.1
  project: system(0)
  copy 1: ---D Mar 01 10:21      431.21bc6 li VOL024
  ...
root@solaris:~#
```

8. 가능한 원인을 찾습니다. 먼저, Oracle HSM `/var/adm/sam-log` 파일에서 손상된 복사본이 있는 파일의 inode에 해당하는 스테이징 관련 메시지를 찾습니다.

검색은 다양한 방법으로 수행할 수 있습니다. 이 예에서는 Solaris `cat` 명령을 사용하여 로그 파일의 내용을 나열하고 출력을 `grep` 및 inode 번호와 일치하는 정규 표현식에 연결합니다. 두 가지 메시지를 찾았습니다. 둘 다 I/O 오류를 나타내고 하나는 테이프 드라이브 중 하나인 장비(`eq`) 서수 번호 804를 명시적으로 나타냅니다.

```
root@solaris:~# cat /var/adm/sam-log | grep "inode 1460"
Mar 11 15:35:44 server1 genu-20[8899]: Stage request canceled for inode 1460
  (eq 804): I/O error.
Jan 11 15:35:44 server1 samfs[8894]: /sam4 inode 1460.1 - Archive copy 1 marked
  damaged: I/O error
```

9. `/var/adm/sam-log` 파일에서 특정 Oracle HSM 장비 서수 번호를 나타낼 경우 장치 로그 `/var/opt/SUNwsamfs/devlog/drive-equipment-number`를 검사합니다. 여

기서 *drive-equipment-number*는 */var/adm/sam-log* 파일에 나열된 서수 번호입니다.

10. 문제가 특정 드라이브와 관련된 것으로 보일 경우 *samcmd unavail drive-equipment-number* 명령을 사용하여 스테이징 프로세스에서 해당 드라이브를 사용하지 못하도록 합니다. 그런 다음 복사본을 손상 해제하고 다시 스테이징을 시도합니다.

```
root@solaris:~# samcmd unavail 804
root@solaris:~# stage -c 1 -I /samqfs1/genfiles/ab11
root@solaris:~# undamage -c1 /samqfs1/genfiles/ab11
root@solaris:~#
```

11. 스테이징을 다시 실패하거나 결함이 있는 드라이브가 하나가 아닌 것으로 보이는 경우 [5.2절. “아카이버 로그 항목을 사용하여 파일 복구” \[43\]](#)에 설명된 대로 *request* 및 *star* 명령 또는 *tar* 및 *dd*와 같은 Solaris 유틸리티를 사용하여 복사본 복구를 시도합니다.
12. 그래도 파일을 복구할 수 없고 데이터 값이 이를 보증하는 경우 데이터 복구 서비스를 받습니다. Oracle StorageTek 테이프 매체에 대한 지원이 필요한 경우 Oracle StorageTek 엔터프라이즈 테이프 데이터 복구 서비스를 받습니다. My Oracle Support(support.oracle.com)에 로그인합니다. 서비스 요청을 열고 요청 범주 아래에 있는 목록에서 테이프 드라이브 모델을 선택한 다음 하위 범위 아래에 있는 목록에서 Media Issues(매체 문제)를 선택합니다.
13. 복구할 수 없는 파일일 경우 손상된 복사본을 아카이브 해제합니다. *unarchive -c CopyNumber file* 명령을 사용합니다. 여기서 *CopyNumber*는 복사본 번호를 나타내는 정수이고 *file*은 손상된 파일의 경로와 파일 이름입니다.

```
root@solaris:~# unarchive -c 1 /samqfs1/genfiles/ab11
root@solaris:~#
```

14. 로그 파일에 나타난 모든 드라이브 또는 매체 문제를 해결합니다.
15. 이전 단계에서 아카이브, 스테이징 및 재활용 프로세스를 사용 안함으로 설정할 경우 지금 다시 사용으로 설정합니다. [6.1절. “아카이브 파일 시스템을 일반 작업으로 복원” \[57\]](#)으로 이동합니다.
16. 그렇지 않은 경우 여기서 중지합니다.

구성 및 데이터 복구를 완료했으면 다음 두 가지 마지막 작업을 수행해야 합니다.

- [아카이브 파일 시스템을 일반 작업으로 복원 \[57\]](#)
- [새 구성 정보 보존 \[60\]](#).

6.1. 아카이브 파일 시스템을 일반 작업으로 복원

아카이브 및 재활용을 사용 안함으로 설정한 경우 지금 다시 사용으로 설정하십시오.

- [아카이브를 사용으로 설정 \[57\]](#)
- [재활용을 사용으로 설정 \[58\]](#).

6.1.1. 아카이브를 사용으로 설정

1. 파일 시스템 메타 데이터 서버에 *root*로 로그인합니다.

```
root@solaris:~#
```

2. 텍스트 편집기에서 */etc/opt/SUNWsamfs/archiver.cmd* 파일을 열고 복구 작업을 시작할 때 파일에 추가했던 첫번째 *wait* 지시어로 스크롤합니다.

이 예에서는 *vi* 편집기를 사용합니다.

```
root@solaris:~# vi /etc/opt/SUNWsamfs/archiver.cmd
# Configuration file for Oracle HSM archiving file systems
#-----
# General Directives
archivemeta = off
examine = noscan
#-----
# Archive Set Assignments
wait
fs = samqfs1
logfile = /var/adm/samqfs1.archive.log
all .
    1 -norelease 15m
    2 -norelease 15m
fs = samqfs2
logfile = /var/adm/samqfs2.archive.log
```

```
all .
...
```

3. 아카이브를 사용으로 설정하려면 복구 작업을 시작할 때 파일에 추가했던 모든 *wait* 지시어를 삭제합니다. 파일을 저장하고 편집기를 닫습니다.

이 예에서는 추가했던 하나의 *wait* 지시어를 제거합니다.

```
root@solaris:~# vi /etc/opt/SUNWsamfs/archiver.cmd
...
#-----
# Archive Set Assignments
fs = samqfs1
logfile = /var/adm/samqfs1.archive.log
all .
    1 -norelease 15m
    2 -norelease 15m
    3 -norelease 15m
fs = samqfs2
...
:wq
root@solaris:~#
```

4. **재활용을 사용으로 설정 [58]**을 수행합니다.

6.1.2. 재활용을 사용으로 설정

1. 파일 시스템 메타 데이터 서버에 *root*로 로그인합니다.

```
root@solaris:~#
```

2. 텍스트 편집기에서 */etc/opt/SUNWsamfs/recycler.cmd* 파일을 열고 복구 작업을 시작할 때 파일에 추가했던 첫번째 *-ignore* 매개변수로 스크롤합니다.

이 예에서는 *vi* 편집기를 사용합니다.

```
root@solaris:~# vi /etc/opt/SUNWsamfs/recycler.cmd
# Configuration file for Oracle HSM archiving file systems
#-----
logfile = /var/adm/recycler.log
no_recycle tp VOL[0-9][2-9][0-9]
library1 -hwm 95 -mingain 60 -ignore
```

3. 복구 작업을 시작할 때 추가했던 모든 *-ignore* 매개변수를 제거합니다. 그런 다음 파일을 저장하고 편집기를 닫습니다.

이 예에서는 Oracle HSM 구성에 하나의 라이브러리인 *library1*만 있습니다.

```

root@solaris:~# vi /etc/opt/SUNWsamfs/recycler.cmd
# Configuration file for Oracle HSM archiving file systems
#-----
logfile = /var/adm/recycler.log
no_recycle tp VOL[0-9][2-9][0-9]
library1 -hwm 95 -mingain 60 -ignore
:wq
root@solaris:~#

```

- 수정된 구성 파일에 오류가 있는지 확인합니다. *archiver -lv* 명령을 사용하여 *archiver.cmd* 파일을 확인한 다음 초기화 명령 *sam-fsd*를 실행합니다. 오류를 수정합니다.

이 예의 구성 파일은 올바릅니다.

```

root@solaris:~# archiver -lv
Reading '/etc/opt/SUNWsamfs/archiver.cmd'.
1: #-----
2: # General Directives
3: archivemeta = off
4: examine = noscan
5: #-----
5: # Archive Set Assignments
7: fs = samqfs1
...
.sort: path
root@solaris:~# sam-fsd
Trace file controls:
sam-amld      /var/opt/SUNWsamfs/trace/sam-amld
...
Would start sam-archiverd()
Would start sam-stagealld()
Would start sam-stagerd()
Would start sam-amld()
root@solaris:~#

```

- 복원된 구성 파일을 사용하여 Oracle HSM 소프트웨어를 재구성합니다. *samd config* 명령을 사용합니다.

아카이브 및 재활용 프로세스가 재개됩니다.

```

root@solaris:~# samd config

```

- 하나 이상의 파일 시스템을 서버 문제, 손실 또는 손상으로부터 복구하려는 경우 **새로 복원된 Oracle HSM 구성 저장 [60]**을 수행합니다.

7. 그렇지 않은 경우 여기서 중지합니다.

6.2. 새 구성 정보 보존

복구 작업 과정에서 Oracle HSM 구성을 변경한 경우 지금 구성을 다시 백업해야 합니다.

6.2.1. 새로 복원된 Oracle HSM 구성 저장

1. 파일 시스템 메타 데이터 서버에 *root*로 로그인합니다.

```
root@solaris:~#
```

2. *samexplorer* 명령을 실행하고 SAMreport를 만듭니다. 백업 구성 정보가 포함된 디렉토리에 저장합니다. *samexplorer path/hostname.YYYYMMDD.hhmmz.tar.gz* 명령을 사용합니다. 여기서 *path*는 선택한 디렉토리의 경로이고 *hostname*은 Oracle HSM 파일 시스템 호스트의 이름이고 *YYYYMMDD.hhmmz*는 날짜 및 시간 기록입니다.

기본 파일 이름은 */tmp/SAMreport.hostname.YYYYMMDD.hhmmz.tar.gz*입니다. 이 예에서는 이미 SAMreport를 저장하기 위한 디렉토리인 */zfs1/sam_config/*가 있습니다. 따라서 이 디렉토리에 보고서를 만듭니다.

```
root@solaris:~# samexplorer /zfs1/sam_config/explorer/server1.20140430.1659MST.tar.gz
Report name:      /zfs1/sam_config/explorer/samhost1.20140430.1659MST.tar.gz
Lines per file:   1000
Output format:    tar.gz (default) Use -u for unarchived/uncompressed.

Please wait.....
Please wait.....
Please wait.....

The following files should now be ftp'ed to your support provider
as ftp type binary.

/zfs1/sam_config/explorer/samhost1.20140430.1659MST.tar.gz
```

3. 구성 파일을 다른 파일 시스템으로 복사합니다.

```
/etc/opt/SUNWsamfs/
mcf
archiver.cmd
defaults.conf
diskvols.conf
hosts.family-set-name
hosts.family-set-name.local
preview.cmd
recycler.cmd
```

```

releaser.cmd
rft.cmd
samfs.cmd
stager.cmd
inquiry.conf
samremote           # SAM-Remote server configuration file
family-set-name      # SAM-Remote client configuration file
network-attached-library # Parameters file
scripts/*            # Back up all locally modified files
/var/opt/SUNWsamfs/

```

4. 내역을 통해 유지 관리되는 데이터를 비롯하여 라이브러리 카탈로그 데이터를 모두 백업합니다. 카탈로그마다 `/opt/SUNWsamfs/sbin/dump_cat -V catalog-file` 명령을 사용합니다. 여기서 *catalog-file*은 카탈로그 파일의 경로와 이름입니다. 출력을 새 위치에 있는 *dump-file*로 재지정합니다.

이 예에서는 *library1*에 대한 카탈로그 데이터를 독립된 NFS 마운트 파일 시스템 *zfs1*의 디렉토리에 있는 *library1cat.dump* 파일에 덤프합니다.

```

root@solaris:~# dump_cat -V /var/opt/SUNWsamfs/catalog/library1cat > /
/zfs1/sam_config/20140513/catalogs/library1cat.dump

```

5. Oracle HSM 설치 및 구성 중 수정했던 시스템 구성 파일을 복사합니다. 이러한 파일에는 다음이 포함될 수 있습니다.

```

/etc/
syslog.conf
system
vfstab
/kernel/drv/
sgen.conf
samst.conf
samrd.conf
sd.conf
ssd.conf
st.conf
/usr/kernel/drv/dst.conf

```

6. Oracle HSM 구성의 일부로 만들었던 모든 사용자 정의 셸 스크립트 및 *crontab* 항목을 선택한 하위 디렉토리에 복사합니다.

예를 들어, 복구 지점 만들기를 관리하기 위해 *crontab* 항목을 만든 경우 지금 복사본을 저장합니다.

7. Oracle HSM, Solaris 및 Solaris Cluster(적용 가능한 경우)를 포함한 현재 설치된 소프트웨어의 개정 레벨을 기록하고 *readme* 파일 정보의 복사본을 선택한 하위 디렉토리에 저장합니다.

8. 필요한 경우 소프트웨어를 신속하게 복원할 수 있도록 새로 다운로드한 Oracle Oracle HSM, Solaris 및 Solaris Cluster 패키지의 복사본을 선택한 하위 디렉토리에 저장합니다.

부록 A

아카이버 로그 이해

다음 표는 아카이버 로그의 각 필드를 정의합니다.

필드	일반적인 값	의미
1	A	아카이브 작업: A(A rchived), R(R e-archived) 또는 U(U narchived)
2	2014/03/23	아카이브 작업 날짜이며, yyyy/mm/dd 형식입니다.
3	18:42:06	아카이브 작업 시간이며, hh:mm:ss 형식입니다.
4	mo	아카이브 매체 유형입니다. 매체 유형에 대한 자세한 내용은 매뉴얼 페이지의 mcf(4)를 참조하십시오.
5	0004A	VSN입니다. 이동식 매체 카트리지의 경우, 볼륨 일련 이름입니다. 디스크 아카이브의 경우, 디스크 볼륨 이름 및 아카이브 tar(1) 파일 경로입니다.
6	arset0.1	아카이브 세트 및 복사본 번호입니다.
7	9a089.1329	매체(tar 파일)에서 아카이브 파일의 물리적 시작 위치 및 아카이브 파일 내의 파일 오프셋이며, 16진수 형식입니다.
8	samfs1	파일 시스템 이름입니다.
9	118.51	Inode 번호 및 생성 번호입니다. inode 번호는 재사용되므로 생성 번호가 고유성을 위해 inode 번호와 함께 사용됩니다.
10	162514	파일이 한 볼륨에만 쓰여진 경우 파일의 길이입니다. 파일이 여러 볼륨에 쓰여진 경우 섹션의 길이입니다.
11	t0/fdn	파일 시스템의 마운트 지점에 대한 파일의 상대 경로 및 이름입니다.
12	f	파일의 유형: d(directory - 디렉토리), f(file - 파일), l(symbolic link - 심볼릭 링크), R(Removable-media file - 이동식 매체 파일), I(segment Index - 세그먼트 인덱스) 또는 s(data Segment - 데이터 세그먼트)
13	0	오버플로우된 파일 또는 세그먼트의 섹션입니다. 파일이 오버플로우된 파일인 경우, 값은 0이 아닌 값입니다. 기타 모든 파일 유형의 경우, 값은 0입니다.
14	56	파일이 아카이브된 드라이브의 장비 서수입니다.

다음 샘플은 아카이버 로그 파일의 샘플 라인을 보여줍니다.

```
A 2014/03/23 18:42:06 mo 0004A arset0.1 9a089.1329 samfs1 118.51 162514 t0/fdn f 0 56
A 2014/03/23 18:42:10 mo 0004A arset0.1 9aac2.1 samfs1 189.53 1515016 t0/fae f 0 56
A 2014/03/23 18:42:10 mo 0004A arset0.1 9aac2.b92 samfs1 125.53 867101 t0/fai f 0 56
A 2014/03/23 19:13:09 lt SLOT22 arset0.2 798.1 samfs1 71531.14 1841087 t0/fhh f 0 51
A 2014/03/23 19:13:10 lt SLOT22 arset0.2 798.e0e samfs1 71532.12 543390 t0/fhg f 0 51
A 2014/03/24 13:30:24 dk DISK01/d8/d16/f216 arset4.1 810d8.1 qfs2 119571.301 1136048 t1/fileem f 0 0
A 2014/03/24 13:30:25 dk DISK01/d8/d16/f216 arset4.1 810d8.8ad qfs2 119573.295 1849474
t1/fileud f 0 0
A 2014/03/24 13:30:25 dk DISK01/d8/d16/f216 arset4.1 810d8.16cb qfs2 119576.301 644930 t1/fileen f 0
0
A 2014/03/24 13:30:25 dk DISK01/d8/d16/f216 arset4.1 810d8.1bb8 qfs2 119577.301 1322899 t1/fileeo f 0
0
```


장비 유형 용어집

마스터 구성 파일(*mcf* [75])의 *Equipment Type* 필드 값은 Oracle Hierarchical Storage Manager and StorageTek QFS Software 내에서 장치 및 장치 구성을 식별합니다. 장비 유형은 2자리 문자 코드로 지정됩니다. 이 용어집에서는 샘플로 작업하거나 기존 *mcf*(자세한 내용은 *mcf*(4) 매뉴얼 페이지 참조)를 해석할 때 빠른 참조를 위한 코드를 나열합니다.

편의상 코드는 세 섹션으로 나눈 다음 알파벳순으로 나열되어 있습니다.

- [권장 장비 및 매체 유형 \[65\]](#)
- [기타 장비 및 매체 유형 \[66\]](#)

B.1. 권장 장비 및 매체 유형

이 절에서는 일반적으로 필요한 일반 장비 코드(*rb*, *tp* 및 *od*) 및 네트워크 연결 라이브러리 인터페이스와 Oracle HSM 내역을 식별하기 위한 코드 등 모든 장비 코드를 설명합니다.

일반 장비 코드인 *rb*, *tp* 및 *od*는 모든 SCSI 연결 라이브러리, 테이프 드라이브 및 옵티컬 디스크 장치에 대한 선호 장비 유형 코드입니다. 일반 장비 유형을 지정할 경우 Oracle HSM에서 SCSI 공급업체 코드를 기준으로 올바른 유형을 자동으로 설정할 수 있습니다.

gxxx

여기서 xxx는 [0-127] 범위의 정수이며, *ma* 디스크 캐시 패밀리 세트의 일부인 스트라이프된 디스크 장치 그룹입니다.

hy

Oracle HSM 내역기이며, 매체 카탈로그를 유지 관리하지만 연관된 하드웨어가 없는 선택적 가상 라이브러리입니다. 내보낸 매체를 추적하는 데 사용됩니다.

ma

하나 이상의 전용 *mm* 디스크 장치에서 파일 시스템 메타 데이터를 유지 관리하는 고성능 QFS 파일 시스템입니다. 파일 데이터는 별도의 *md*, *mr* 또는 *gxxx* 데이터 장치에 상주합니다.

md

ma 파일 시스템에 대한 파일 데이터 또는 *ms* 파일 시스템에 대한 데이터 및 메타 데이터를 저장하는 디스크 장치입니다. *md* 장치는 파일 데이터를 작은 4KB DAU(디스크 할당 단위) 및 큰 16KB, 32KB 또는 64KB DAU로 저장합니다. 기본 DAU는 64KB입니다.

mm

고성능 *ma* 파일 시스템에 대한 파일 시스템 메타 데이터를 저장하는 디스크 장치입니다.

mr

ma 파일 시스템에 대한 파일 데이터를 저장하는 디스크 장치입니다. *mr* 장치는 파일 데이터를 완전히 조정 가능한 8-65528KB 범위에서 8KB의 배수인 큰 DAU(디스크 할당 단위)로 저장합니다. 기본 DAU는 64KB입니다.

ms

파일 데이터를 저장하는 동일한 장치에서 파일 시스템 메타 데이터를 유지 관리하는 Oracle HSM 파일 시스템입니다.

od

모든 SCSI 연결 옵티컬 디스크입니다. Oracle HSM는 SCSI 공급업체 코드를 사용하여 적절한 장비 유형을 자동으로 설정합니다.

rb

모든 SCSI 연결 테이프 라이브러리입니다. Oracle HSM는 SCSI 공급업체 코드를 사용하여 적절한 장비 유형을 자동으로 설정합니다.

rd

SAM-Remote 의사 장치입니다. 마스터 구성 파일(*mcf*)에서 해당 *Equipment Identifier* 필드에 의사 장치의 경로를 포함해야 합니다(예: */dev/samrd/rd2*). 해당 *Family Set* 필드에 SAM-Remote 서버의 호스트 이름을 포함해야 합니다.

sc

SAM-Remote 클라이언트 시스템입니다. 마스터 구성 파일(*mcf*)에서 해당 *Equipment Identifier* 필드에 SAM-Remote 클라이언트에 대한 클라이언트 구성 파일의 경로를 포함해야 합니다. 해당 *Family Set* 필드에 서버의 패밀리 세트 이름을 포함해야 합니다. *Additional Parameters* 필드에 클라이언트의 라이브러리 카탈로그 파일에 대한 전체 경로를 포함해야 합니다.

sk

네트워크 연결 테이프 라이브러리에 대한 Oracle StorageTek ACSLS 인터페이스입니다. 마스터 구성 파일(*mcf*)에서 해당하는 *Equipment Identifier* 필드에는 ACSLS 인터페이스에 대한 매개변수 파일의 경로가 포함되어야 합니다. 자세한 내용은 *stk(7)* 매뉴얼 페이지를 참조하십시오.

ss

SAM-Remote 서버입니다. 마스터 구성 파일(*mcf*)에서 해당 *Equipment Identifier* 필드에 SAM-Remote 서버 구성 파일의 경로를 포함해야 합니다. 해당 *Family Set* 필드에 서버의 패밀리 세트 이름을 포함해야 하며, 이는 클라이언트에서 *mcf*의 *Family Set* 필드에 사용된 이름과 일치해야 합니다.

tp

모든 SCSI 연결 테이프 드라이브입니다. Oracle HSM는 SCSI 공급업체 코드를 사용하여 적절한 장비 유형을 자동으로 설정합니다. 하지만 [li \[67\]](#) 및 [ti \[68\]](#)와 같이 더 구체적인 장비 코드를 사용할 경우 일관성 있게 사용해야 합니다. 예를 들어, *mcf* 파일에서 *li*(LTO) 테이프 장비를 지정할 경우 [archiver\(아카이버\) \[71\].cmd](#) 파일에서 *tp* 장비와 동일한 장비를 참조할 수 없습니다.

B.2. 기타 장비 및 매체 유형

이 절에 나열된 장비 유형도 지원됩니다.

대부분의 경우 일반 장비 유형 *rb*, *tp* 및 *od*를 사용하여 SCSI 연결 라이브러리, 테이프 드라이브 및 광 디스크 장치를 식별하는 것이 좋습니다. 일반 장비 유형은 Oracle HSM가 SCSI 공급업체 ID를 사용하여 하드웨어를 동적으로 식별하도록 합니다. 아래의 유형 코드는 한 매체 유형에서 다른 매체 유형으로 마이그레이션하는 경우 필수적이며 경우에 따라 관리 목적에 유용할 수 있습니다. 하지만 이러한 코드를 마스터 구성 파일(*mcf*)에서 사용하면 일정 시점에서 실제 하드웨어와 더 이상 일치하지 않는 정적 장비 구성이 하드 코딩됩니다.

ac

Sun 1800, 3500 또는 L11000 테이프 라이브러리입니다.

at

Sony AIT-4 또는 AIT-5 테이프 드라이브입니다.

cy

Cygnnet 옵티컬 디스크 라이브러리입니다.

d3

StorageTek D3 테이프 드라이브입니다.

dm

Sony DMF 라이브러리입니다.

ds

DocuStore 또는 Plasmon 옵티컬 디스크 라이브러리입니다.

dt

DAT 4mm 테이프 드라이브입니다.

e8

Exabyte X80 라이브러리입니다.

fd

Fujitsu M8100 128트랙 테이프 드라이브입니다.

h4

HP SL48 또는 SL24 라이브러리입니다.

hc

Hewlett Packard L9-/L20-/L60 시리즈 라이브러리입니다.

i7

IBM 3570 테이프 드라이브입니다.

ic

IBM 3570 매체 교환기입니다.

il

IBM 3584 테이프 라이브러리입니다.

li

LTO-3 이상의 테이프 드라이브입니다.

lt

DLT(Digital Linear Tape), Super DLT 또는 DLT-S4 테이프 드라이브입니다.

me

Metrum 라이브러리입니다.

mf

IBM 다기능 옵티컬 드라이브입니다.

mo

5.25인치 지우기 가능 옵티컬 드라이브입니다.

o2

12인치 WORM 드라이브입니다.

ov

Overland Data Inc. Neo Series 테이프 라이브러리입니다.

pd

Plasmon D-Series DVD-RAM 라이브러리입니다.

q8

Qualstar 42xx, 62xx 또는 82xx 라이브러리입니다.

s3

StorageTek SL3000 라이브러리입니다.

s9

Oracle StorageTek 97xx 시리즈 라이브러리입니다.

se

StorageTek 9490 테이프 드라이브입니다.

sf

StorageTek T9940 테이프 드라이브입니다.

sg

StorageTek 9840C 이상 테이프 드라이브입니다.

sl

Spectra Logic 또는 Qualstar 테이프 라이브러리입니다.

st

StorageTek 3480 테이프 드라이브입니다.

ti

StorageTek T10000(Titanium) 테이프 드라이브입니다.

vt

Metrum VHS(RSP-2150) 테이프 드라이브입니다.

wo

5.25인치 옵티컬 WORM 드라이브입니다.

xt

Exabyte (850x) 8mm 테이프 드라이브입니다.

부록 C

제품 접근성 기능

저시력, 맹인, 색맹 또는 기타 시각 장애를 가진 사용자는 명령줄 인터페이스를 통해 Oracle Hierarchical Storage Manager and StorageTek QFS Software (Oracle HSM)에 액세스할 수 있습니다. 이 텍스트 기반 인터페이스는 스크린 리더와 호환되며, 모든 기능은 키보드를 사용하여 제어됩니다.

용어집

이 용어집에서는 Oracle HSM 소프트웨어 및 파일 시스템과 관련된 용어에 초점을 맞춥니다. 업계 표준 정의는 <http://www.snia.org/education/dictionary/>에서 Storage Networking Industry Association이 제공하는 사전을 참조하십시오.

커널	기본적인 운영 체제 기능을 제공하는 프로그램입니다. UNIX 커널은 프로세스를 만들고 관리하며, 파일 시스템 액세스 기능, 일반적인 보안 및 통신 기능을 제공합니다.
addressable storage(주소 지정 가능한 스토리지)	Oracle HSM 파일 시스템을 통해 사용자가 참조하는 온라인, 니어라인, 오프사이트 및 오프라인 스토리지를 포함하는 스토리지 공간입니다.
admin set ID(관리 세트 ID)	공통 특성을 공유하는 사용자 및/또는 그룹의 스토리지 관리자 정의 세트입니다. 관리 세트는 대개 여러 그룹의 사용자가 관여하고 여러 파일 및 디렉토리에 걸친 프로젝트에 대한 스토리지를 관리하기 위해 만들어집니다.
archival media(아카이브 매체)	아카이브 파일이 쓰여지는 매체입니다. 아카이브 매체에는 이동식 테이프 또는 마그네틱-옵티컬 카트리지를 위한 아카이브를 위해 구성된 디스크 파일 시스템이 모두 포함됩니다.
archival storage(아카이브 스토리지)	아카이브 매체에 만든 데이터 스토리지 공간입니다.
archive set(아카이브 세트)	아카이브 세트는 아카이브할 파일 그룹을 식별하며 파일은 크기, 소유권, 그룹, 디렉토리 위치와 관련한 공통 기준을 공유합니다. 아카이브 세트는 여러 파일 시스템 그룹에 걸쳐 정의할 수 있습니다.
archiver(아카이버)	이동식 카트리지에 파일 복사를 자동으로 제어하는 아카이브 프로그램입니다.
associative staging(연관 스테이징)	그룹의 한 멤버가 스테이징된 경우 관련 파일의 그룹을 스테이징합니다. 파일이 동일한 디렉토리에 있고 함께 자주 사용되는 경우 파일 소유자는 Oracle HSM 연관 스테이징 파일 속성을 설정하여 연관시킬 수 있습니다. 그런 다음 그룹의 한 파일이 응용 프로그램에서 액세스될 때 그룹의 파일이 오프라인인 경우 Oracle HSM는 전체 그룹을 아카이브 매체에서 디스크 캐시로 스테이징합니다. 그러면 모든 필요한 파일을 동시에 다시 사용할 수 있게 됩니다.
audit (full)(전체 감사)	VSN을 확인하기 위해 카트리지를 로드하는 프로세스입니다. 마그네틱-옵티컬 카트리지의 경우, 용량 및 공간 정보가 확인되고 자동화 라이브러리의 카탈로그에 입력됩니다. volume serial number(VSN, 볼륨 일련 번호) [81] 를 참조하십시오.
automated library(자동화된 라이브러리)	작업자 개입 없이 이동식 매체 카트리지를 자동으로 로드 및 언로드하도록 설계된 로봇으로 제어되는 장치입니다. 자동화 라이브러리에는 하나 이상의 드라이브와 카트리지를 스토리지 슬롯 및 드라이브에서 이동하는 전송 메커니즘이 포함됩니다.

backup(백업)	의도치 않은 손실을 막기 위한 용도의 파일 모음 스냅샷입니다. 백업에는 파일의 속성 및 연관된 데이터가 모두 포함됩니다.
block allocation map(블록 할당 맵)	디스크에서 각 사용 가능한 스토리지 블록을 나타내고 블록이 사용 중인지 또는 사용 가능한지 나타내는 비트맵입니다.
block size(블록 크기)	하드 디스크 또는 마그네틱 테이프 카트리지와 같은 블록 장치에서 가장 작은 주소 지정 가능한 데이터 단위의 크기입니다. 디스크 장치에서는 대개 512바이트인 섹터 크기와 같습니다.
cartridge(카트리지)	마그네틱 테이프 또는 옵티컬 매체와 같은 데이터 스토리지 매체에 대한 컨테이너입니다. volume(볼륨) [81] , 테이프 또는 매체 조각이라고도 합니다. volume serial number(VSN, 볼륨 일련 번호) [81] 를 참조하십시오.
catalog(카탈로그)	자동화 라이브러리에서 이동식 매체 볼륨의 레코드입니다. 각 자동화 라이브러리에 대해 하나의 카탈로그가 있고, 사이트에는 모든 자동화 라이브러리에 대해 하나의 내역기가 있습니다. 볼륨은 volume serial number(VSN, 볼륨 일련 번호) [81] 를 사용하여 식별 및 추적됩니다.
client-server(클라이언트-서버)	한 사이트의 프로그램이 다른 사이트의 프로그램에 요청을 보내고 응답을 기다리는 분산된 시스템의 상호 작용 모델입니다. 요청 프로그램을 클라이언트라고 합니다. 응답을 충족시키는 프로그램을 서버라고 합니다.
connection(연결)	안정적인 스트림 전달 서비스를 제공하는 두 프로토콜 모듈 사이의 경로입니다. TCP 연결은 한 컴퓨터의 TCP 모듈에서 다른 컴퓨터의 TCP 모듈로 확장됩니다.
data device(데이터 장치)	파일 시스템에서 파일 데이터가 저장되는 장치 또는 장치 그룹입니다.
DAU	disk allocation unit(DAU, 디스크 할당 단위) [73] 를 참조하십시오.
device logging(장치 로깅)	Oracle HSM 파일 시스템을 지원하는 하드웨어 장치에 대한 특정 오류 정보를 제공하는 구성 가능한 기능입니다.
device scanner(장치 스캐너)	모든 수동으로 마운트된 이동식 장치의 유무를 정기적으로 모니터링하고 사용자 또는 기타 프로세스가 요청할 수 있는 마운트된 카트리지의 유무를 감지하는 소프트웨어입니다.
direct access(직접 액세스)	니어라인 파일을 아카이브 매체에서 직접 액세스할 수 있고 디스크 캐시로 검색할 필요가 없도록 지정하는 파일 속성(스테이징 안함)입니다.
direct attached library(직접 연결 라이브러리)	SCSI 인터페이스를 사용하여 서버에 직접 연결된 자동화 라이브러리입니다. SCSI 연결 라이브러리는 Oracle HSM 소프트웨어에 의해 직접 제어됩니다.
direct I/O(직접 I/O)	큰 블록이 정렬된 순차적 I/O에 사용되는 속성입니다. <i>setfa</i> 명령의 <i>-D</i> 옵션은 직접 I/O 옵션입니다. 이 옵션은 파일 또는 디렉토리에 대해 직접 I/O 속성을 설정합니다. 디렉토리에 적용되면 직접 I/O 속성이 상속됩니다.

directory(디렉토리)	파일 시스템 내에서 다른 파일 및 디렉토리를 가리키는 파일 데이터 구조입니다.
disk allocation unit(DAU, 디스크 할당 단위)	<p>Oracle HSM 파일 시스템에서 쓰여진 데이터의 크기에 상관 없이 각 I/O 작업이 소비하는 최소 연속 공간 크기입니다. 따라서 데이터 할당 단위는 제공된 크기의 파일을 전송할 때 필요한 최소 I/O 작업의 수를 결정합니다. 디스크 장치의 block size(블록 크기) [72] 배수이어야 합니다.</p> <p>디스크 할당 단위는 선택한 Oracle HSM 장치 유형 및 사용자 요구 사항에 따라 달라집니다. <i>md</i> 장치 유형은 이중 할당 단위를 사용합니다. 처음 8번 쓰기에 대한 DAU는 4KB이고 후속 쓰기에 대해서는 사용자 지정 16, 32 또는 64KB이므로 작은 파일은 작은 블록에 기록되지만 더 큰 파일은 더 큰 블록에 기록됩니다. <i>mr</i> 및 striped group(스트라이프 그룹) [80] 장치 유형은 [8-65528]KB 범위 내에서 8의 증분으로 조정할 수 있는 DAU를 사용합니다. 따라서 파일은 큰 균일 크기의 파일 크기에 근접하는 큰 균일 블록으로 쓰여집니다.</p>
disk buffer(디스크 버퍼)	Sun SAM-Remote 구성에서 클라이언트에서 서버로 데이터 아카이브에 사용되는 서버 시스템의 버퍼입니다.
disk cache(디스크 캐시)	온라인 디스크 캐시와 아카이브 매체 사이에서 데이터 파일을 만들고 관리하는 데 사용되는 파일 시스템 소프트웨어의 디스크 상주 부분입니다. 개별 디스크 분할 영역 또는 전체 디스크를 디스크 캐시로 사용할 수 있습니다.
disk space threshold(디스크 공간 임계값)	관리자가 정의하는 디스크 캐시 사용률의 최대 또는 최소 레벨입니다. 릴리서는 이와 같이 미리 정의된 디스크 공간 임계값을 기준으로 디스크 캐시 사용률을 제어합니다.
disk striping(디스크 스트라이핑)	여러 디스크에 걸쳐 파일을 기록하는 프로세스로, 액세스 성능이 높아지고 전체적인 스토리지 용량이 증가합니다. striping(스트라이핑) [80]을 참조하십시오.
drive(드라이브)	이동식 매체 볼륨 사이에 데이터를 전송하기 위한 메커니즘입니다.
Ethernet(이더넷)	패킷 스위칭 로컬 영역 네트워크 기술입니다.
extent array(익스텐트 어레이)	파일에 지정된 각 데이터 블록의 디스크 위치를 정의하는 파일 inode 내의 어레이입니다.
family device set(패밀리 장치 세트)	family set(패밀리 세트) [73]를 참조하십시오.
family set(패밀리 세트)	디스크 모음이나 자동화된 라이브러리 내의 드라이브와 같은 독립된 물리적 장치의 논리적 그룹입니다. 또한 storage family set(스토리지 패밀리 세트) [80]를 참조하십시오.
FDDI	FDDI(Fiber-distributed Data Interface)는 최고 200km(124마일)까지 범위를 확장할 수 있는 로컬 영역 네트워크의 데이터 전송 표준입니다. FDDI 프로토콜은 토큰 링 프로토콜을 기반으로 합니다.

Fibre Channel(광 섬유 채널)	장치 사이에 고속 직렬 통신을 지정하는 ANSI 표준입니다. 광 섬유 채널은 SCSI-3에서 버스 아키텍처 중 하나로 사용됩니다.
file system(파일 시스템)	파일 및 디렉토리의 계층적 모음입니다.
file-system-specific directives(파일 시스템 특정 지시어)	<i>archiver.cmd</i> 파일에서 전역 지시어 다음에 오는 아카이버 및 릴리서 지시어는 특정 파일 시스템에 따라 다르고 <i>fs</i> =로 시작됩니다. 파일 시스템 특정 지시어는 다음 <i>fs</i> = 지시어 라인이 오거나 파일의 끝에 도달할 때까지 적용됩니다. 여러 지시어가 파일 시스템에 영향을 미칠 경우 파일 시스템 특정 지시어는 전역 지시어보다 우선합니다.
ftp	FTP(File Transfer Protocol)는 두 호스트 사이에 파일을 전송하기 위한 인터넷 프로토콜입니다. 보다 안전한 대체 프로토콜은 sftp [79] 를 참조하십시오.
global directives(전역 지시어)	모든 파일 시스템에 적용되고 첫번째 <i>fs=</i> 라인 앞에 나타나는 아카이버 및 릴리서 지시어입니다.
grace period(유예 기간)	quota(할당량) [77] 에서 파일 시스템이 지정된 사용자, 그룹 및/또는 admin set ID(관리 세트 ID) [71] 에 속하는 총 파일 크기가 쿼터에 지정된 soft limit(소프트 한계) [79] 을 초과하도록 허용하는 시간입니다.
hard limit(하드 한계)	quota(할당량) [77] 에서 지정된 사용자, 그룹 및/또는 admin set ID(관리 세트 ID) [71] 가 소비할 수 있는 스토리지 리소스의 절대 최대 용량입니다. soft limit(소프트 한계) [79] 를 참조하십시오.
high-water mark(고수위)	<ol style="list-style-type: none"> 1. 아카이브 파일 시스템에서 Oracle HSM 파일 시스템이 이전에 아카이브된 파일을 디스크에서 삭제하면서 릴리서 프로세스를 시작하는 디스크 캐시 사용률입니다. 제대로 구성된 고수위는 파일 시스템이 새 파일 및 새로 스테이징된 파일에 사용 가능한 공간을 항상 충분히 유지하도록 합니다. 자세한 내용은 <i>sam-releaser</i> 및 <i>mount_samfs</i> 매뉴얼 페이지를 참조하십시오. low-water mark(저수위) [75]와 비교하십시오. 2. 아카이브 파일 시스템의 일부인 이동식 매체 라이브러리에서 리사이클러 프로세스를 시작하는 매체 캐시 사용률입니다. 재활용은 현재 데이터의 일부 가득 찬 볼륨을 비워 새 매체로 교체하거나 다시 레이블을 지정할 수 있도록 합니다.
historian(내역기)	Oracle HSM 내역기가 <i>/etc/opt/SUNWsamfs/mcf</i> 파일에 정의된 자동화 매체 라이브러리에서 내보낸 볼륨의 카탈로그입니다. 기본적으로 Oracle HSM 파일 시스템 호스트의 <i>/var/opt/SUNWsamfs/catalog/historian</i> 에 위치합니다. 자세한 내용은 Oracle HSM <i>historian</i> 매뉴얼 페이지를 참조하십시오.
hosts file(hosts 파일)	hosts 파일에는 공유 파일 시스템의 모든 호스트 목록이 포함됩니다. 파일 시스템을 Oracle HSM 공유 파일 시스템으로 초기화하는 경우 파일 시스템이 만들어지기 전에 hosts 파일 <i>/etc/opt/SUNWsamfs/hosts.fs-name</i> 을 만들어야 합니다. <i>sammkfs</i> 명령은 파일 시스템을 만들 때 hosts 파

	일을 사용합니다. <i>samsharefs</i> 명령을 사용하여 나중에 hosts 파일의 내용을 바꾸거나 업데이트할 수 있습니다.
indirect block(간접 블록)	스토리지 블록의 목록을 포함하는 디스크 블록입니다. 파일 시스템에는 최대 세 레벨의 간접 블록이 있습니다. 첫번째 레벨 간접 블록은 데이터 스토리지에 사용되는 블록 목록을 포함합니다. 두번째 레벨 간접 블록은 첫번째 레벨 간접 블록 목록을 포함합니다. 세번째 레벨 간접 블록은 두번째 레벨 간접 블록 목록을 포함합니다.
inode	인덱스 노드입니다. 파일을 기술하기 위해 파일 시스템에서 사용되는 데이터 구조입니다. inode는 이름 이외의 파일과 관련된 모든 속성을 기술합니다. 속성에는 소유권, 액세스, 권한, 크기 및 디스크 시스템의 파일 위치가 포함됩니다.
inode file(inode 파일)	파일 시스템에 상주하는 모든 파일에 대한 inode 구조를 포함하는 파일 시스템의 특수 파일(<i>.inodes</i>)입니다. inode 길이는 512바이트입니다. inode 파일은 파일 시스템의 파일 데이터에서 분리된 메타 데이터 파일입니다.
LAN	로컬 영역 네트워크입니다.
lease(임대)	지정된 시간 동안 파일에서 작업을 수행할 클라이언트 호스트 권한을 부여하는 기능입니다. 메타 데이터 서버는 각 클라이언트 호스트에게 임대를 부여합니다. 파일 작업을 계속 수행할 수 있도록 필요에 따라 임대를 갱신할 수 있습니다.
library catalog(라이브러리 카탈로그)	catalog(카탈로그) [72] 를 참조하십시오.
library(라이브러리)	automated library(자동화된 라이브러리) [71] 를 참조하십시오.
local file system(로컬 파일 시스템)	Sun Cluster 시스템의 한 노드에 설치되고 다른 노드에서 별로 사용되지 않는 파일 시스템입니다. 또한 서버에 설치된 파일 시스템입니다.
low-water mark(저수위)	아카이브 파일 시스템에서 Oracle HSM 파일 시스템이 릴리서 프로세스를 중지하고 이전에 아카이브된 파일을 디스크에서 삭제를 중지하는 디스크 캐시 사용률입니다. 제대로 구성된 저수위는 파일 시스템이 최상의 성능을 위해 가능한 많은 파일을 캐시에 유지하고 새 파일 및 새로 스테이징된 파일에 사용 가능한 공간을 유지하도록 합니다. 자세한 내용은 <i>sam-releaser</i> 및 <i>mount_samfs</i> 매뉴얼 페이지를 참조하십시오. high-water mark(고수위) [74] 와 비교하십시오.
LUN	Logical Unit Number(논리 장치 번호)의 약어입니다.
mcf	Master Configuration File(마스터 구성 파일)의 약어입니다. 파일 시스템 환경에서 장치 사이의 관계(토폴로지)를 정의하는, 초기화 시 읽는 파일입니다.
media recycling(매체 재활용)	활성 파일이 적은 아카이브 매체를 재활용하거나 재사용하는 프로세스입니다.

media(매체)	테이프 또는 광 디스크 카트리지입니다.
metadata device(메타 데이터 장치)	파일 시스템 메타 데이터가 저장되는 장치(예: SSD(Solid State Disk) 또는 미러 장치)입니다. 파일 데이터 및 메타 데이터를 별도의 장치에 두면 성능이 향상될 수 있습니다. <i>mcf</i> 파일에서 메타 데이터 장치는 <i>ma</i> 파일 시스템 내에서 <i>mm</i> 장치로 선언됩니다.
metadata(메타 데이터)	데이터에 대한 데이터입니다. 메타 데이터는 디스크에서 파일의 정확한 데이터 위치를 찾는 데 사용되는 인덱스 정보입니다. 파일, 디렉토리, 액세스 제어 목록, 심볼릭 링크, 이동식 매체, 세그먼트화된 파일 및 세그먼트화된 파일의 인덱스에 대한 정보로 구성됩니다.
mirror writing(미러 쓰기)	단일 디스크 실패로부터 손실을 막기 위해 별도의 디스크 세트에 두 개의 파일 복사본을 유지하는 프로세스입니다.
mount point(마운트 지점)	파일 시스템이 마운트되는 디렉토리입니다.
multireader file system(다중 읽기 파일 시스템)	다중 호스트에 마운트될 수 있는 파일 시스템을 지정할 수 있는 단일 쓰기, 다중 읽기 기능입니다. 여러 호스트가 파일 시스템을 읽을 수 있지만, 하나의 호스트만 파일 시스템에 쓸 수 있습니다. 다중 리더는 <i>mount</i> 명령에서 <i>-o reader</i> 옵션으로 지정됩니다. 단일 쓰기 호스트는 <i>mount</i> 명령에서 <i>-o writer</i> 옵션으로 지정됩니다. 자세한 내용은 <i>mount_samfs</i> 매뉴얼 페이지를 참조하십시오.
name space(이름 공간)	파일, 속성 및 스토리지 위치를 식별하는 파일 모음의 메타 데이터 부분입니다.
nearline storage(니어라인 스토리지)	액세스하려면 먼저 로봇 마운트가 필요한 이동식 매체 스토리지입니다. 니어라인 스토리지는 일반적으로 온라인 스토리지보다 저렴하지만 액세스 시간이 다소 길어집니다.
network attached automated library(네트워크 연결식 자동화된 라이브러리)	StorageTek, ADIC/Grau, IBM, Sony 등의 공급업체에서 제공한 소프트웨어 패키지를 사용하여 제어되는 라이브러리입니다. QFS 파일 시스템은 자동화된 라이브러리용으로 특별히 설계된 Oracle HSM 매체 교환기 데몬을 사용하여 공급업체 소프트웨어와 상호 작용합니다.
NFS	네트워크 파일 시스템(Network File System)의 약어로 이기종 네트워크 환경에서 원격 파일 시스템에 대한 투명한 액세스를 제공하는 파일 시스템입니다.
NIS	네트워크 정보 서비스(Network Information Service)의 약어로 네트워크의 시스템 및 사용자에게 대한 주요 정보를 포함하는 분산된 네트워크 데이터베이스입니다. NIS 데이터베이스는 마스터 서버 및 모든 슬레이브 서버에 저장됩니다.
offline storage(오프라인 스토리지)	로드를 위해 작업자 개입이 필요한 스토리지입니다.

offsite storage (오프사이트 스토리지)	서버와 원격으로 떨어져서 재해 복구에 사용되는 스토리지입니다.
online storage (온라인 스토리지)	디스크 캐시 스토리지와 같이 즉시 사용 가능한 스토리지입니다.
Oracle HSM	<ol style="list-style-type: none"> 1. Oracle Hierarchical Storage Manager에 대한 일반 약어입니다. 2. 아카이브용으로 구성되고 Oracle HSM 소프트웨어에서 관리되는 QFS [77] 파일 시스템을 설명하는 형용사입니다.
partition (분할 영역)	장치의 일부분 또는 광자기 카트리지의 한쪽 면입니다.
preallocation (미리 할당)	파일을 쓰기 위해 디스크 캐시에 연속 공간을 예약하는 프로세스입니다. 미리 할당은 제로 크기 파일에만 지정할 수 있습니다. 자세한 내용은 <i>setfa</i> 매뉴얼 페이지를 참조하십시오.
pseudo device (의사 장치)	연결된 하드웨어가 없는 소프트웨어 부속 시스템 또는 드라이버입니다.
QFS	Oracle HSM QFS 소프트웨어 제품을 가리키며, 그 자체로 또는 Oracle Hierarchical Storage Manager에서 제어하는 아카이빙 파일 시스템으로 사용할 수 있는 고성능 고용량 UNIX 파일 시스템입니다.
qfsdump	samfsdump(qfsdump) [79] 를 참조하십시오.
qfsrestore	samfsrestore(qfsrestore) [79] 를 참조하십시오.
quota (할당량)	지정된 사용자, 그룹 또는 admin set ID(관리 세트 ID) [71] 가 소비할 수 있는 스토리지 리소스 용량입니다. hard limit(하드 한계) [74] 및 soft limit(소프트 한계) [79] 를 참조하십시오.
RAID	RAID(Redundant array of independent disks)입니다. 여러 독립된 디스크를 사용하여 파일을 안정적으로 저장하는 디스크 기술입니다. 단일 디스크 실패로부터 데이터 손실을 막을 수 있고, 내결함성 디스크 환경을 제공할 수 있으며, 개별 디스크보다 높은 처리량을 제공할 수 있습니다.
recovery point (복구 지점)	<p>Oracle HSM 파일 시스템에 대한 메타 데이터의 특정 시점 백업 복사본을 저장하는 압축 파일입니다.</p> <p>사용자 파일의 의도치 않은 삭제부터 전체 파일 시스템의 재해 손실에 이르기까지 모든 데이터 손실 발생 시 관리자는 파일 또는 파일 시스템이 영향을 받지 않은 마지막 복구 지점을 찾아 파일 또는 파일 시스템의 마지막으로 알려진 양호한 상태로 거의 즉시 복구할 수 있습니다. 그런 다음 관리자는 해당 시점에 기록된 메타 데이터를 복원하고 메타 데이터에 나타난 파일을 아카이브 매체에서 디스크 캐시로 스테이징하거나 사용자 및 응용 프로그램이 액세스할 때 필요에 따라 파일 시스템에서 파일을 스테이징할 수 있습니다.</p>
recycler (리사이클러)	만료된 아카이브 복사본이 차지하고 있는 카트리지의 공간을 확보하는 Oracle HSM 유틸리티입니다.

regular expression (정규 표현식)	파일 이름 및 구성 파일과 같이 다른 문자열을 검색, 선택 및 편집하기 위해 마련된 표준화된 패턴 일치 언어의 문자열입니다. Oracle HSM 파일 시스템 작업에서 사용되는 정규 표현식 구문에 대한 자세한 내용은 Oracle HSM Solaris <i>regex</i> 및 <i>regcmp</i> 매뉴얼 페이지를 참조하십시오.
release priority (릴리스 우선 순위)	파일 시스템의 파일이 아카이브된 후 릴리스되는 우선 순위입니다. 릴리스 우선 순위는 등록 정보의 다양한 가중치를 곱한 다음 결과를 더하여 계산됩니다.
releaser (릴리서)	더 많은 디스크 캐시 공간을 사용할 수 있도록 아카이브된 파일을 식별하고 디스크 캐시 복사본을 릴리스하는 Oracle HSM 구성 요소입니다. 릴리서는 상한 및 하한 임계값에 따라 온라인 디스크 스토리지의 용량을 자동으로 조절합니다.
remote procedure call (원격 프로시저 호출)	RPC [78] 를 참조하십시오.
removable media file (이동식 매체 파일)	마그네틱 테이프 또는 옵티컬 디스크 카트리지와 같이 이동식 매체 카트리지에 상주하는 위치에서 직접 액세스할 수 있는 특수한 사용자 파일 유형입니다. 아카이브 및 스테이지 파일 데이터를 쓰는 데도 사용됩니다.
robot (로봇)	스토리지 슬롯과 드라이브 사이에서 카트리지를 이동하는 automated library (자동화된 라이브러리) [71] 구성 요소입니다. transport (전송) [80]라고도 합니다.
round-robin (라운드 로빈)	전체 파일이 순차적으로 논리 디스크에 쓰여지는 데이터 액세스 방식입니다. 단일 파일이 디스크에 쓰여질 때 전체 파일이 첫번째 논리 디스크에 쓰여집니다. 두번째 파일은 그 다음 논리 디스크에 쓰여지는 방식으로 이어집니다. 각 파일의 크기는 I/O의 크기를 결정합니다. disk striping (디스크 스트라이핑) [73] 및 striping (스트라이핑) [80]을 참조하십시오.
RPC	Remote Procedure Call(원격 프로시저 호출)의 약어입니다. NFS에서 사용자 정의 네트워크 데이터 서버를 구현하기 위해 사용되는 기본 데이터 교환 메커니즘입니다.
SAM	Oracle Hierarchical Storage Manager 제품의 이전 이름인 Storage Archive Manager에 대한 일반 약어입니다.
SAM-QFS	<ol style="list-style-type: none"> 1. 이전 버전의 Oracle Hierarchical Storage Manager 제품에 대한 일반 약어입니다. 2. 아카이브용으로 구성되고 Oracle HSM 소프트웨어에서 관리되는 QFS [77] 파일 시스템을 설명하는 형용사입니다.
SAM-Remote client (SAM-Remote 클라이언트)	여러 의사 장치를 포함하는 클라이언트 데몬으로 구성된 Oracle HSM 시스템이며, 자체 라이브러리 장치를 가질 수도 있습니다. 클라이언트는 하나 이상의 아카이브 복사본에 대해 아카이브 매체용 SAM-Remote 서버에 의존합니다.

SAM-Remote server(SAM-Remote 서버)	전체 용량 Oracle HSM 스토리지 관리 서버 및 SAM-Remote 클라이언트 사이에 공유할 라이브러리를 정의하는 SAM-Remote 서버 데몬입니다.
samfsdump(qfsdump)	제공된 파일 그룹에 대한 제어 구조 덤프를 만들고 모든 제어 구조 정보를 복사하는 프로그램입니다. 일반적으로 파일 데이터는 복사하지 않습니다. <i>-u</i> 옵션을 사용하면 이 명령은 데이터 파일도 복사합니다. Oracle Hierarchical Storage Manager 패키지가 설치되지 않은 경우 이 명령을 <i>qfsdump</i> 라고 합니다.
samfsrestore(qfsrestore)	제어 구조 덤프에 대한 inode 및 디렉토리 정보를 복원하는 프로그램입니다. 또한 samfsdump(qfsdump) [79] 를 참조하십시오.
SAN	Storage Area Network(스토리지 영역 네트워크)의 약어입니다.
SCSI	Small Computer System Interface(소형 컴퓨터 시스템 인터페이스)의 약어로, 디스크 및 테이프 드라이브나 자동화된 라이브러리와 같은 주변 장치에 흔히 사용되는 전기 통신 사양입니다.
seeking(검색)	랜덤 액세스 I/O 작업 중에 디스크 장치의 읽기/쓰기 헤드를 특정 디스크 위치에서 다른 위치로 이동하는 것입니다.
sftp	Secure File Transfer Protocol의 약어로 ftp [74] 를 기반으로 하는 ssh [79] 의 보안 구현입니다.
shared hosts file(공유 호스트 파일)	공유 파일 시스템을 만들 때 시스템은 hosts 파일에서 메타 데이터 서버의 공유 hosts 파일로 정보를 복사합니다. samsharefs -u 명령을 실행할 때 이 정보를 업데이트합니다.
Small Computer System Interface(소형 컴퓨터 시스템 인터페이스)	SCSI [79] 를 참조하십시오.
soft limit(소프트 한계)	quota(할당량) [77] 에서 지정된 사용자, 그룹 및/또는 admin set ID(관리 세트 ID) [71] 가 무한대 기간 동안 채울 수 있는 스토리지 공간의 최대 용량입니다. 파일은 최대 하드 한계까지 소프트 한계에서 허용하는 것보다 많은 공간을 사용할 수 있지만 쿼터에 정의된 짧은 grace period(유예 기간) [74] 동안만 가능합니다. hard limit(하드 한계) [74] 를 참조하십시오.
ssh	Secure Shell의 약어로 안전한 원격 명령줄 로그인 및 명령 실행을 허용하는 암호화된 네트워크 프로토콜입니다.
staging(스테이징)	니어라인 또는 오프라인 파일을 아카이브 스토리지에서 온라인 스토리지로 다시 복사하는 프로세스입니다.
Storage Archive Manager	Oracle Hierarchical Storage Manager 제품의 이전 이름입니다.

storage family set (스토리지 패밀리 세트)	단일 논리 장치로 통칭되는 디스크 세트입니다.
storage slots (스토리지 슬롯)	드라이브에서 사용 중인 아닐 때 카트리지를 저장하는 자동화된 라이브러리 안의 위치입니다.
stripe size (스트라이프 크기)	쓰기가 다음 스트라이프 장치로 이동하기 전에 할당할 디스크 할당 단위(DAU)의 수입니다. <i>stripe=0</i> 마운트 옵션이 사용되는 경우 파일 시스템은 스트라이프 액세스가 아닌 라운드 로빈 액세스를 사용합니다.
striped group (스트라이프 그룹)	<i>mcf</i> 파일에서 하나 이상의 <i>gxxx</i> 장치로 정의되는 파일 시스템 내의 장치 모음입니다. 스트라이프 그룹은 하나의 논리적 장치로 취급되고 언제나 디스크 할당 단위(DAU)와 동일한 크기로 스트라이프됩니다.
striping (스트라이핑)	파일이 인터레이스 방식으로 논리 디스크에 동시에 쓰여지는 데이터 액세스 방식입니다. Oracle HSM 파일 시스템은 스트라이프 그룹을 사용하는 "하드 스트라이핑"과 <i>stripe=x</i> 마운트 매개변수를 사용하는 "소프트 스트라이핑"의 두 유형의 스트라이핑을 제공합니다. 하드 스트라이핑은 파일 시스템이 설정될 때 활성화되며 <i>mcf</i> 파일 안에 스트라이프 그룹이 정의되어야 합니다. 소프트 스트라이핑은 <i>stripe=x</i> 마운트 매개변수를 통해 활성화되며 파일 시스템 또는 개별 파일에 대해 변경될 수 있습니다. <i>stripe=0</i> 을 설정하면 사용 안함으로 설정됩니다. 파일 시스템이 동일한 수의 요소를 갖는 다중 스트라이프 그룹으로 구성되는 경우 하드 및 소프트 스트라이핑을 둘 다 사용할 수 있습니다. round-robin(라운드 로빈) [78] 을 참조하십시오.
SUNW.qfs	Oracle HSM 공유 파일 시스템을 지원하는 Solaris Cluster 리소스 유형입니다. <i>SUNW.qfs</i> 리소스 유형은 공유 파일 시스템의 메타 데이터 서버(MDS)에 대한 페일오버 리소스를 정의합니다.
superblock (수퍼 블록)	파일 시스템의 기본적인 매개변수를 정의하는 파일 시스템의 데이터 구조입니다. superblock은 스토리지 패밀리 세트의 모든 분할 영역에 쓰여지고 세트에서 분할 영역의 멤버십을 식별합니다.
tar	Tape archive의 약어입니다. 아카이브 이미지에 사용되는 표준 파일 및 데이터 기록 형식입니다.
TCP/IP	Transmission Control Protocol/Internet Protocol. 호스트간 주소 지정 및 경로 지정, 패킷 전달(IP) 및 응용 프로그램 지점간의 데이터 전달(TCP)을 담당하는 인터넷 프로토콜입니다.
timer (타이머)	사용자가 소프트웨어 한계에 도달하는 시간부터 사용자에게 부여된 하드 한계 사이에 경과된 시간을 추적하는 쿼터 소프트웨어입니다.
transport (전송)	robot(로봇) [78] 을 참조하십시오.
vfstab file (vfstab 파일)	<i>vfstab</i> 파일에는 파일 시스템에 대한 마운트 옵션이 포함됩니다. 명령줄에 지정된 마운트 옵션은 <i>/etc/vfstab</i> 파일에 지정된 마운트 옵션보다 우선하지만, <i>/etc/vfstab</i> 파일에 지정된 마운트 옵션은 <i>samfs.cmd</i> 파일에 지정된 마운트 옵션보다 우선합니다.

volume overflow(볼륨 오버플로우)

시스템에서 단일 파일을 여러 **volume(볼륨) [81]**에 걸쳐 분산시킬 수 있는 기능입니다. 볼륨 오버플로우는 개별 카트리지의 용량을 초과하는 매우 큰 파일을 사용하는 사이트에 유용합니다.

volume serial number(VSN, 볼륨 일련 번호)

1. 테이프 또는 디스크 스토리지 볼륨에 지정된 일련 번호입니다. 볼륨 일련 번호는 최대 6자리의 영숫자 대문자로 구성될 수 있으며, 문자로 시작되어야 하고 테이프 라이브러리 또는 분할 영역과 같이 제공된 컨텍스트 내에서 고유하게 볼륨을 식별해야 합니다. 볼륨 일련 번호는 볼륨 레이블에 쓰여집니다.
2. 느슨하게 사용될 경우 특정 스토리지 **volume(볼륨) [81]**, 특히 이동식 매체 **cartridge(카트리지) [72]**입니다.

volume(볼륨)

1. 스토리지 매체에서 단일의 액세스 가능한 논리적 스토리지 영역이며, 대개 **volume serial number(VSN, 볼륨 일련 번호) [81]** 및/또는 볼륨 레이블로 주소가 지정됩니다. 스토리지 디스크 및 마그네틱 테이프 카트리지는 하나 이상의 볼륨을 포함할 수 있습니다. 사용을 위해 볼륨은 파일 시스템의 지정된 에 마운트 **mount point(마운트 지점) [76]**됩니다.
2. 단일 논리적 볼륨을 포함하는 마그네틱 테이프 **cartridge(카트리지) [72]**입니다.
3. 랜덤 액세스 디스크 장치에서 파일 시스템, 디렉토리 또는 파일은 테이프와 같은 순차 액세스, 이동식 매체 카트리지인 것처럼 구성되고 사용됩니다.

WORM

Write-Once-Read-Many의 약어입니다. 한 번만 쓸 수 있지만 여러 번 읽을 수 있는 매체에 대한 스토리지 분류입니다.

색인

人

설명서

가용성, 6, 6, 6

표기 규약, 5

ㅈ

접근성

제품, 69

