(19)대한민국특허청(KR)
(12) 공개특허공보(A)

(51) 。Int. Cl.
G06F 15/16 (2006.01)

(11) 공개번호 10-2006-0060549
(43) 공개일자 2006년06월05일

(21) 출원번호 10-2005-0095461
(22) 출원일자 2005년10월11일

(30) 우선권주장 10/999,380 2004년11월30일 미국(US)

(71) 출원인 마이크로소프트 코포레이션
미국 워싱턴주 (우편번호 : 98052) 레드몬드 원 마이크로소프트 웨이

(72) 발명자 엘러, 알렉시스 제이
미국 98052 워싱턴주 레드몬드 원 마이크로소프트 웨이마이크로소프트 코포레이션 내
크리스티안센, 닐 알.
미국 98052 워싱턴주 레드몬드 원 마이크로소프트 웨이마이크로소프트 코포레이션 내
틴드, 라빈더 에스.
미국 98052 워싱턴주 레드몬드 원 마이크로소프트 웨이마이크로소프트 코포레이션 내

(74) 대리인 주성민
이중희
백만기

심사청구 : 없음

(54) 원격 파일을 국부적으로 캐싱하기 위한 방법 및 시스템

요약

원격 객체를 국부적으로 캐싱(cache)하기 위한 방법 및 시스템이 설명된다. 객체에 액세싱하기 위한 요청이 수신된다. 객체가 캐싱되어 있는지가 판정된다. 객체가 캐싱되어 있고 요청이 새로운 객체를 생성하거나, 기존 객체를 수정하거나, 디렉토리를 열기 위한 것이 아니면, 요청은 국부 파일 시스템에 향하게 된다. 그렇지 않으면, 요청은 원격 파일 시스템을 향하게 된다. 어떤 객체가 요청되는지에 대한 정보는 원격적으로 저장된 객체를 캐싱 정책에 지정된 바에 따라 국부 서버에 복사하는데 이용될 수 있다.

대표도

도 6

색인어

파일 관리자, 파일 시스템, 필터, 사용자 모드, 커널 모드

명세서

도면의 간단한 설명

도 1은 본 발명이 통합될 수 있는 컴퓨터 시스템을 나타내는 블럭도.

도 2는 본 발명의 다양한 양태에 따른 본 발명이 동작할 수 있는 예시적인 시스템 컴포넌트의 배치를 도시하는 블럭도.

도 3은 본 발명의 다양한 양태에 따른 본 발명이 동작할 수 있는 다른 예시적인 시스템 컴포넌트의 배치를 도시하는 블럭도.

도 4는 본 발명의 다양한 양태에 따른 본 발명이 동작할 수 있는 다른 예시적인 시스템 컴포넌트의 배치를 도시하는 블럭도.

도 5는 본 발명의 다양한 양태에 따른 본 발명이 구현될 수 있는 예시적인 환경을 나타내는 블럭도.

도 6은 본 발명의 다양한 양태에 따른 본 발명이 구현될 수 있는 예시적인 시스템 컴포넌트의 배치를 나타내는 블럭도.

도 7은 본 발명의 다양한 양태에 따른 본 발명이 구현될 수 있는 예시적인 시스템 컴포넌트의 배치를 나타내는 블럭도.

도 8은 본 발명의 다양한 양태에 따라 I/O 동작의 방향을 국부 캐시로 바꿔야하는지를 판정할 때 일어날 수 있는 액션들을 포괄적으로 나타내는 순서도.

도 9는 본 발명의 다양한 양태에 따라 I/O 동작의 방향을 국부 캐시로 바꿔야하는지를 판정할 때 행해질 수 있는, 도 8의 블럭(835)에 대응하는 액션을 포괄적으로 나타내는 순서도.

도 10은 본 발명의 다양한 양태에 따라 원격 객체가 수정될 때 행해질 수 있는 액션들을 포괄적으로 나타내는 순서도.

도 11은 본 발명의 다양한 양태에 따른 본 발명이 구현될 수 있는 또 다른 예시적인 시스템 컴포넌트의 배치를 나타내는 블럭도.

<도면의 주요 부분에 대한 부호의 설명>

210: API

215: I/O 관리자

220: 필터 관리자

230: 필터

235: 파일 시스템

발명의 상세한 설명

발명의 목적

발명이 속하는 기술 및 그 분야의 종래기술

본 발명은 일반적으로 컴퓨터에 관한 것으로, 보다 구체적으로 파일 시스템에 관한 것이다.

예를 들어, 서버 팜(server farm)과 같은 데이터센터 내의 컨텐츠 서버의 세트는 다양한 위치에 있는 수많은 클라이언트들에게 컨텐츠를 서비스할 수 있다. 이전에, 주어진 서버 팜에 의해 호스트되는 전체 컨텐츠 세트는 상대적으로 작았으므로,

과도한 비용 없이 팜 내의 각각의 컨텐츠 서버에 전송 및 저장될 수 있었다. 그러나, 이제는 서버 팜으로부터 이용가능한 총 컨텐츠가 종종 수백 기가바이트를 초과한다. 그들이 요청된 어떤 컨텐츠를 서비스할 수 있기 위해, 서버용 대용량 하드 드라이브를 구매하고 그 하드 드라이브에 데이터센터의 모든 컨텐츠를 제공하고 그 컨텐츠를 최신 상태로 유지시키는 것은 저장 및 전송 비용 면에서 비싸다. 이는 컨텐츠를 서비스하기 위해 2개 이상의 데이터센터를 이용할 때 특히 그러하다.

컨텐츠의 서브세트는 컨텐츠 서버 하드 디스크 상에 캐싱되어 있는 반면, 전체 컨텐츠 세트는 분리된 전용 저장소에 저장될 수 있도록, 원격 데이터를 국부적으로 캐싱하는 효율적인 방법 및 시스템이 요구된다. 이상적으로, 이러한 방법 및 시스템은 데이터를 요청하는 어떤 어플리케이션에 대해서도 통상적으로 또는 완전히 투명할 것이다.

발명이 이루고자 하는 기술적 과제

간략하게 요약하면, 본 발명은 원격 데이터를 국부적으로 캐싱하기 위한 방법 및 시스템을 제공한다. 객체에 액세스하기 위한 요청이 수신된다. 객체가 캐싱되어 있는지가 판정된다. 객체가 캐싱되어 있고, 요청이 새로운 객체를 생성하거나 기존 객체를 수정하거나 디렉토리를 열기 위한 것이 아니면, 이 요청은 국부 파일 시스템을 향하게 된다. 그렇지 않으면, 이 요청은 원격 파일 시스템을 향하게 된다.

본 발명의 한 양태에 있어서, 필터는 요청을 모니터링하고, 캐싱 서비스에게 국부 및 원격 파일 시스템 상에서 액세스되어 있는 객체의 이름을 보고한다. 캐싱 서비스는 이 정보에 정책을 적용시켜 어떤 원격 객체가 국부적으로 캐싱되어 있는지 및 국부적으로 캐싱된 어느 객체가 제거되는지를 판정한다.

본 발명의 또 다른 양태에서, 필터는 국부 및 원격 파일 시스템에의 요청을 모니터링하고, 필터는 정책을 적용시켜 어느 원격 객체가 국부적으로 캐싱되어 있는지 및 국부적으로 캐싱된 어느 객체가 제거되는지를 판정할 수 있다.

본 발명의 또 다른 양태에서, 필터는 객체가 원격으로 수정되었다는 통지를 수신하고, 국부적으로 캐싱된 그 객체의 사본을 삭제한다.

다른 양태들은 다음의 상세한 설명이 도면에 관련하여 취해질 때 명백해질 것이다.

발명의 구성 및 작용

도 1은 본 발명이 구현될 수 있는 적절한 컴퓨팅 시스템 환경(100)의 예를 나타낸다. 컴퓨팅 시스템 환경(100)은 단지 적절한 컴퓨팅 환경의 한 예이며 본 발명의 사용 또는 기능의 범위에 제한을 가하도록 의도된 것은 아니다. 컴퓨팅 환경(100)은 예시적인 운영 환경(100)에 도시된 컴포넌트들 중의 임의의 하나 또는 조합에 관하여 임의의 종속성 또는 요구사항을 갖는 것으로 해석되어서는 안된다.

본 발명은 많은 다른 범용 또는 특수목적 컴퓨팅 시스템 환경들 또는 구성들과 함께 동작될 수 있다. 본 발명과 함께 사용하기에 적합할 수 있는 잘 알려진 컴퓨팅 시스템, 환경, 및/또는 구성의 예로는, 퍼스널 컴퓨터, 서버 컴퓨터, 핸드헬드 또는 랩탑 장치, 멀티프로세서 시스템, 마이크로프로세서-기반 시스템, 셋 탑 박스, 프로그램가능한 가전제품, 네트워크 PC, 미니컴퓨터, 메인프레임 컴퓨터, 상기의 시스템 또는 장치 중의 임의의 것을 포함하는 분산형 컴퓨팅 환경 등이 포함되지만, 이에 한정되지 않는다.

본 발명은 컴퓨터에 의해 실행되는, 프로그램 모듈과 같은 컴퓨터 실행가능 명령어의 일반적인 관점에서 기술될 수 있다. 일반적으로, 프로그램 모듈은 특정 태스크를 수행하거나 특정 추상 데이터 유형을 구현하는 루틴, 프로그램, 오브젝트, 컴포넌트, 데이터 구조 등을 포함한다. 본 발명은 또한 통신 네트워크를 통해 링크된 원격 프로세싱 장치에 의해 태스크를 수행하는 분산형 컴퓨팅 환경에서 실행될 수 있다. 분산 컴퓨팅 환경에서, 프로그램 모듈은 메모리 저장 장치를 포함하는 국부 및 원격 컴퓨터 저장 매체 내에 위치할 수 있다.

도 1을 참조하면, 본 발명을 구현하기 위한 예시적인 시스템은 컴퓨터(110)의 형태의 범용 컴퓨팅 장치를 포함한다. 컴퓨터(110)의 컴포넌트들로는, 프로세싱 유닛(120), 시스템 메모리(130), 및 시스템 메모리를 포함하는 다양한 시스템 컴포넌트를 프로세싱 유닛(120)에 연결시키는 시스템 버스(121)가 포함될 수 있지만, 이에 한정되는 것은 아니다. 시스템 버스(121)는 다양한 버스 아키텍처 중의 임의의 것을 사용하는 국부 버스, 주변 버스, 및 메모리 버스 또는 메모리 컨트롤러를 포함하는 몇 가지 유형의 버스 구조 중의 임의의 것일 수 있다. 예로서, 이러한 아키텍처는 산업 표준 아키텍처(ISA) 버스,

마이크로 채널 아키텍처(MCA) 버스, 인핸스드 ISA(Enhanced ISA; EISA) 버스, 비디오 일렉트로닉스 표준 어소시에이션(VESA) 국부 버스, 및 메자닌(Mezzanine) 버스로도 알려진 주변 컴포넌트 상호접속(PCI) 버스를 포함하지만, 이에 한정되는 것은 아니다.

컴퓨터(110)는 통상적으로 다양한 컴퓨터 판독가능 매체를 포함한다. 컴퓨터 판독가능 매체는 컴퓨터(110)에 의해 액세스될 수 있는 임의의 이용가능한 매체일 수 있으며, 휘발성 및 비휘발성 매체, 분리형(removable) 및 비분리형(non-removable) 매체를 둘다 포함한다. 예로서, 컴퓨터 판독가능 매체는 컴퓨터 저장 매체 및 통신 매체를 포함할 수 있지만, 이에 한정되는 것은 아니다. 컴퓨터 저장 매체는 컴퓨터 판독가능 명령어, 데이터 구조, 프로그램 모듈 또는 다른 데이터와 같은 정보의 저장을 위한 임의의 방법 또는 기술로 구현되는 휘발성 및 비휘발성, 분리형 및 비분리형 매체를 둘다 포함한다. 컴퓨터 저장 매체는 RAM, ROM, EEPROM, 플래쉬 메모리 또는 기타 메모리 기술, CD-ROM, DVD(digital versatile disk) 또는 기타 광 디스크 저장장치, 자기 카세트, 자기 테이프, 자기 디스크 저장장치 또는 기타 자기 저장장치, 또는 컴퓨터(110)에 의해 액세스될 수 있고 원하는 정보를 저장하는 데 사용될 수 있는 임의의 기타 매체를 포함하지만, 이에 한정되지 않는다. 통신 매체는 통상적으로 반송파 또는 기타 전송 메커니즘 등의 변조된 데이터 신호에 컴퓨터 판독가능 명령어, 데이터 구조, 프로그램 모듈, 또는 다른 데이터를 구현하며, 임의의 정보 전달 매체를 포함한다. "변조된 데이터 신호"라는 용어는 신호 내의 정보를 인코딩하도록 하나 또는 그 이상의 특성을 설정 또는 변경시킨 신호를 의미한다. 예로서, 통신 매체는 유선 네트워크 또는 직접 유선 접속 등의 유선 매체와, 음향, RF, 적외선 및 기타 무선 매체 등의 무선 매체를 포함하지만, 이에 한정되지 않는다. 상술한 것들의 임의의 조합 또한 컴퓨터 판독가능 매체의 범위 내에 포함되어야 한다.

시스템 메모리(130)는 ROM(131) 및 RAM(132) 등의 휘발성 및/또는 비휘발성 메모리의 형태의 컴퓨터 저장 매체를 포함한다. 시동중과 같은 때에 컴퓨터(110) 내의 구성요소들 간에 정보를 전송하는 것을 돕는 기본 루틴을 포함하는 기본 입출력 시스템(133; BIOS)은 일반적으로 ROM(131)에 저장된다. RAM(132)은 일반적으로 프로세싱 유닛(120)에 즉시 액세스할 수 있고 및/또는 프로세싱 유닛(120)에 의해 현재 작동되는 프로그램 모듈 및/또는 데이터를 포함한다. 예로서, (한정하고자 하는 것은 아님) 도 1은 오퍼레이팅 시스템(134), 어플리케이션 프로그램(135), 기타 프로그램 모듈(136), 및 프로그램 데이터(137)를 도시한다.

컴퓨터(110)는 또한 다른 분리형/비분리형, 휘발성/비휘발성 컴퓨터 저장 매체를 포함할 수 있다. 단지 예로서, 도 1에는 비분리형 비휘발성 자기 매체로부터 판독하거나 그 자기 매체에 기록하는 하드 디스크 드라이브(140), 분리형 비휘발성 자기 디스크(152)로부터 판독하거나 그 자기 디스크에 기록하는 자기 디스크 드라이브(151), 및 CD-ROM 또는 기타 광 매체 등의 분리형 비휘발성 광 디스크(156)로부터 판독하거나 그 광 디스크에 기록하는 광 디스크 드라이브(155)가 도시되어 있다. 예시적인 운영 환경에서 사용될 수 있는 다른 분리형/비분리형, 휘발성/비휘발성 컴퓨터 저장 매체는 자기 테이프 카세트, 플래쉬 메모리 카드, DVD(Digital versatile disk), 디지털 비디오 테이프, 고체 RAM, 고체 ROM 등을 포함하지만 이에 한정되지 않는다. 하드 디스크 드라이브(141)는 일반적으로 인터페이스(140)와 같은 비분리형 메모리 인터페이스를 통해 시스템 버스(121)에 접속되고, 자기 디스크 드라이브(151) 및 광 디스크 드라이브(155)는 일반적으로 인터페이스(150)와 같은 분리형 메모리 인터페이스에 의해 시스템 버스(121)에 접속된다.

앞서 기술되고 도 1에 도시된 드라이브 및 그 관련 컴퓨터 저장 매체는 컴퓨터(110)를 위한 컴퓨터 판독가능 명령어, 데이터 구조, 프로그램 모듈 및 기타 데이터의 저장을 제공한다. 도 1에서, 예를 들어, 하드 디스크 드라이브(141)는 오퍼레이팅 시스템(144), 어플리케이션 프로그램(145), 기타 프로그램 모듈(146), 및 프로그램 데이터(147)를 저장하는 것으로 도시된다. 이들 컴포넌트는 오퍼레이팅 시스템(134), 어플리케이션 프로그램(135), 기타 프로그램 모듈(136), 및 프로그램 데이터(137)와 동일할 수도 있고 다를 수도 있다. 오퍼레이팅 시스템(144), 어플리케이션 프로그램(145), 다른 프로그램 모듈(146), 및 프로그램 데이터(147)는 최소한 다른 복사본(different copies)임을 나타내기 위하여 다른 번호를 부여하였다. 사용자는 일반적으로 마우스, 트랙볼, 또는 터치 패드라 불리우는 포인팅 장치(161) 및 키보드(162)와 같은 입력 장치를 통해 컴퓨터(110)에 명령 및 정보를 입력할 수 있다. (도시되지 않은) 기타 입력 장치는 마이크로폰, 조이스틱, 게임 패드, 위성 안테나, 스캐너 핸드헬드 PC 또는 기타 기록용 태블릿의 접촉 감지형 스크린 등을 포함할 수 있다. 이들 입력 장치 및 그외의 입력 장치는 시스템 버스에 연결된 사용자 입력 인터페이스(160)를 통해 종종 프로세싱 유닛(120)에 접속되지만, 병렬 포트, 게임 포트 또는 유니버설 시리얼 버스(USB)와 같은 기타 인터페이스 및 버스 구조에 의해 접속될 수 있다. 모니터(191) 또는 다른 유형의 디스플레이 장치는 또한 비디오 인터페이스(190) 등의 인터페이스를 통해 시스템 버스(121)에 접속된다. 모니터 외에도, 컴퓨터는 또한 출력 주변 인터페이스(195)를 통해 접속될 수 있는 스피커(197) 및 프린터(196) 등의 기타 주변 출력 장치를 포함할 수 있다.

컴퓨터(110)는 원격 컴퓨터(180)와 같은 하나 이상의 원격 컴퓨터로의 논리적 접속을 이용한 네트워크 환경에서 동작할 수 있다. 원격 컴퓨터(180)는 퍼스널 컴퓨터, 서버, 라우터, 네트워크 PC, 피어(peer) 장치, 또는 기타 공통 네트워크 노드일 수 있으며, 비록 도 1에는 메모리 저장 장치(181)만이 도시되어 있지만, 컴퓨터(110)에 관하여 상술한 구성요소 중 다

수 또는 모든 구성요소를 일반적으로 포함한다. 도 1에 도시된 논리적 접속은 근거리 통신망(LAN; 171) 및 원거리 통신망(WAN; 173)을 포함하지만, 그 외의 네트워크를 포함할 수도 있다. 이러한 네트워크 환경은 사무실, 기업 광역 컴퓨터 네트워크, 인트라넷 및 인터넷에서 일반적인 것이다.

LAN 네트워크 환경에서 사용되는 경우, 컴퓨터(110)는 네트워크 인터페이스 또는 어댑터(170)를 통해 LAN(171)에 접속된다. WAN 네트워크 환경에서 사용되는 경우, 컴퓨터(110)는 일반적으로 인터넷 등의 WAN(173)을 통해 통신을 구축하기 위한 모뎀(172) 또는 기타 수단을 포함한다. 내장형 또는 외장형일 수 있는 모뎀(172)은 사용자 입력 인터페이스(160) 또는 기타 적절한 메커니즘을 통해 시스템 버스(121)에 접속될 수 있다. 네트워크 환경에서, 컴퓨터(110)에 관하여 도시된 프로그램 모듈 또는 그 일부분은 원격 메모리 저장 장치에 저장될 수 있다. 예로서 (한정하고자 하는 것은 아님), 도 1은 메모리 장치(181)에 상주하는 원격 어플리케이션 프로그램(185)을 도시한다. 도시된 네트워크 접속은 예시적인 것이며, 컴퓨터들간의 통신 링크를 구축하는 그 외의 수단이 사용될 수 있다고 이해되어야 한다.

예시적인 필터 및 그것의 배치

Windows®NTFS (Windows®NT File System), FAT, CDFS, SMB 리다이렉터 파일시스템(SMB redirector file system) 또는 WebDav 파일 시스템과 같은 기초 파일 시스템을 구비한 마이크로소프트사의 Windows®XP 오퍼레이팅 시스템과 같은 현대의 오퍼레이팅 시스템을 구비한 하나 이상의 파일 시스템 필터 드라이버는 사용자 I/O 요청을 수신하는 I/O 관리자와 파일 시스템 드라이버 사이에 삽입될 수 있다. 일반적으로, 필터 드라이버(여기서 종종 단순히 "필터"라고 언급됨)는 바이러스 방지 소프트웨어, 파일 시스템 할당 공급자, 파일 복사기 및 암호화/압축 제품을 통해 파일 시스템 I/O (요청 및 데이터)를 전달하는 것과 같은 사용자가 원하는 태스크를 포함하는 다양한 파일 관련 컴퓨팅 태스크를 수행함으로써, 기초 파일 시스템을 강화시키는 프로세스 또는 컴포넌트이다.

예를 들어, 파일 복사 제품이 파일 시스템 레벨 모니터링을 수행하는 반면, 바이러스 방지 제품은 바이러스 서명을 찾기 위해 특정 파일 유형(.exe, .doc 등)에의 및 그로부터의 I/O를 관찰하는 필터를 제공한다. 다른 유형의 파일 시스템 필터 드라이버는 시스템 (사용자가 본래의 상태로 돌아갈 수 있도록, 파일 시스템이 변경되자마자 파일 시스템을 백업시키는) 복구, 디스크 할당 시행, 열린 파일 백업, 삭제된 파일 삭제 취소, 파일 암호화 등에 관한 것이다. 그러므로, 파일 시스템 필터 드라이버를 설치함으로써, 컴퓨터 사용자는 실제 오퍼레이팅 시스템 또는 파일 시스템 드라이버 코드를 바꾸지 않고서 컴포넌트를 업그레이드, 대체, 삽입 및 제거할 수 있는 방식으로, 자신이 원하고 필요로 하는 파일 시스템 특징을 선택할 수 있다.

도 2는 본 발명의 다양한 양태에 따른 본 발명이 동작할 수 있는 예시적인 시스템 컴포넌트의 배치를 도시하는 블럭도이다. 컴포넌트는 하나 이상의 어플리케이션(205), 어플리케이션 프로그래밍 인터페이스(210; API), 입출력(I/O) 관리자(215), 필터 관리자(220), 파일 시스템(225) 및 하나 이상의 필터(230-232)를 포함한다.

어플리케이션(205)은 API(210)를 통하여 (예를 들면, 함수/메소드 호출에 의하여) I/O 관리자(215)에게 파일 시스템을 요청할 수 있다. I/O 관리자(215)는 각 요청을 만족시키기 위하여 어떤 I/O 요청 또는 요청들이 발행되어야 하는지 판정하고 각 I/O 요청을 필터 관리자(220)에게 송신할 수 있다. I/O 관리자(215)는 또한 어플리케이션(205)에게, 파일 시스템 요청들에 관련된 동작들이 진행되거나, 완료되거나, 중단됨에 따른 데이터를 반환할 수 있다.

한 구현에서, 필터는 (예를 들면, 초기화 절차에서) 인스턴스화(instant)될 때 필터 관리자(220)의 등록 메커니즘에 등록한 객체 등을 포함한다. 효율성을 위하여, 각 필터는 전형적으로 이 필터가 프로세싱할 때 관심을 가질 수 있는 파일 시스템 요청만을 등록시킬 것이다. 이를 위하여, 등록의 일부로서, 각 필터는 필터 관리자(220)에게 이 필터가 관심 있어 하는 I/O 요청 유형들(예를 들면, 생성, 판독, 기록, 닫기, 이름바꾸기 등)을 통지한다. 예를 들면, 암호화 필터는 판독 및 기록 I/O에는 등록하지만, 데이터가 암호화되거나 복호화될 필요가 없는 다른 I/O에는 등록하지 않을 수 있다. 마찬가지로, 할당 필터는 객체 생성 및 객체 기록에만 관심이 있을 수 있다.

필터가 관심 있어 하는 I/O 요청 유형을 지정하는 것 외에도, 또한 필터는 자신이 각각의 I/O 유형 마다 프리-콜백(pre-callback)으로 통지되어야 할지 포스트-콜백(post-callback)으로 통지되어야 할지의 여부도 지정할 수 있다. 프리-콜백은 I/O 요청에 관련된 데이터가 I/O 관리자(215)로부터 파일 시스템(225)으로 전달될 때 호출되는 한편, 포스트-콜백은 I/O 요청 완료 시에 I/O 요청에 관련된 데이터가 파일 시스템(225)으로부터 I/O 관리자(215)에게로 전달될 때 호출된다.

각 I/O 요청으로부터, 필터 관리자(220)는 필터(230-232)가 이용하기에 적절한 균일한 포맷으로 된 데이터 구조를 생성할 수 있다. 이하, 이러한 데이터 구조를 때로는 콜백 데이터라고 칭할 것이다. 그 후 필터 관리자(220)는 콜백 데이터를 호출하여, 필터 관리자(220)가 수신한 I/O 유형에 대한 콜백을 수신하도록 등록한 각 필터에게 콜백 데이터를 전달한다. 필터 관리자(220)가 수신한 I/O 유형에 대한 콜백을 수신한다고 등록된 임의의 필터는 종종 등록된 필터라고 칭해진다.

전형적으로, 필터 관리자(220)는 특정 유형의 I/O 요청에 관련된 콜백 데이터를 각 등록된 필터에게 등록된 필터가 정렬된 순서로 전달한다. 예를 들면, 필터(230 및 232)가 모든 판독 I/O 요청에 대한 콜백을 수신한다고 등록되고 이러한 요청들을 처리함에 있어서 필터(230)가 필터(232) 보다 먼저 처리되도록 정렬된다면, 판독 I/O를 수신한 이후에, 필터 관리자(220)는 먼저 콜백 데이터를 호출하고, 필터(230)에게 콜백 데이터를 전달하고 필터(230)가 이 콜백 데이터를 처리한 후에, 필터 관리자(220)는 콜백 데이터를 호출하여 (조금이라도 수정됐으면) 콜백 데이터를 필터(232)에게 전달할 수 있다.

필터는 하나 이상의 볼륨에 첨부될 수 있다. 즉, 필터는 단지 하나 이상의 볼륨들에 관련된 I/O에 대한 콜백 데이터가 호출되어 이를 수신하도록 등록될 수 있다.

필터는 자신만의 I/O 요청을 생성할 수 있고 이 요청은 그 후에 다른 필터들에게 전달될 수 있다. 예를 들면, 바이러스 방지 필터는 파일을 열리기 전에 읽어보기를 원할 수 있다. 필터는 I/O 요청을 더 전달할 것을 중지할 수 있고 파일 관리자에게 I/O 요청에 대한 상태 코드(예를 들면, 성공 또는 실패)를 보고하라고 명령할 수 있다. 필터는 데이터를 메모리에 저장하고 이 데이터를 디스크에 영속(예를 들어, 저장)시킬 수 있다. 일반적으로, 필터가 커널 모드 또는 사용자 모드 프로세스에 의해 수행될 수 있는 임의의 액션 집합을 수행하도록 생성될 수 있고, 반응적(reactive)이고/이거나(예를 들면, 필터는 행동하기 전에 I/O 요청을 수신할 때까지 기다림) 선행적(proactive)일[예를 들면, 자신만의 I/O 요청을 개시하고 I/O 관리자(215)에 의해 처리되는 I/O 요청과 동기적으로 다른 액션들을 수행함] 수 있다.

한 실시예에서, 본 발명의 다양한 양태에 따른 본 발명이 동작할 수 있는 시스템 컴포넌트의 또 다른 예시적인 배치를 도시하는 블럭도인 도 3에 도시된 바와 같이 필터는 스택을 이루는 방식으로 배치될 수 있다. 이 실시예에서, 필터(305-307) 각각은 I/O 요청을 처리하고 (수정되거나 수정되지 않은) 이 요청들을 스택의 다른 필터 또는 다른 컴포넌트에게 전달할 수 있다. 예를 들면, 어플리케이션(205)들 중 하나로부터 수신된 판독 요청에 대한 응답으로, I/O 관리자(215)는 I/O 요청을 발행하고 이 요청을 필터(305)에 송신할 수 있다. 필터(305)는 I/O 요청을 검사하고 필터(305)가 I/O 요청에 관심이 없다고 판정한 후, 변경되지 않은 I/O 요청을 필터(306)에게 전달할 수 있다. 필터(306)는 필터(306)가 I/O 요청에 기초하는 소정의 액션을 수행할 것이라고 판정한 후, (변경되거나 변경되지 않은) I/O 요청을 필터(307)에게 전달할 수 있다. 필터(307)는 필터(307)가 I/O 요청에 관심이 없다고 판정하고 I/O 요청을 파일 시스템(235)에 전달할 수 있다.

파일 시스템(235)이 I/O 요청을 서비스한 이후에, 이 시스템은 필터(307)에게 결과를 전달한다. 전형적으로, 결과는 I/O 요청이 진행된 순서와 반대로[예를 들면, 처음에는 필터(307)로, 그 다음에는 필터(306)로, 그 다음에는 필터(305)의 순으로] 전달된다. 필터(305-307) 각각은 결과를 검사하여 필터가 결과에 관심있는지 여부를 판정할 수 있으며, (변경되거나 변경되지 않은) 결과를 다른 필터 또는 컴포넌트에게 전달하기 이전에 이 결과에 기초하는 액션들을 수행할 수 있다.

본 발명의 다른 실시예에서, 본 발명의 다양한 양태에 따른 본 발명이 동작할 수 있는 다른 예시적인 시스템 컴포넌트의 배치를 도시하는 블럭도인 도 4에 도시된 바와 같이, 필터는 스택을 이루는/관리되는 방식으로 배치될 수 있다. 이러한 구성에서, 몇몇의 필터들은 필터 관리자에 관련되어 있는 반면 다른 필터들은 필터 관리자에 관련되어 있지 않다. 필터 관리자(220)는 다른 필터들[예를 들면, 필터(305 및 307)]을 가지고 있는 스택에 배치된다.

필터는 본 발명의 취지 및 범위를 벗어나지 않으면서 다수의 다른 구성들로 구현될 수 있다는 것을 쉽게 알 수 있다. 몇몇의 실시예에서, 필터는 어플리케이션과 파일 시스템 간의 I/O를 검사하고 I/O를 변경하거나, 완료하거나, 중단하거나, 이에 기초하는 다른 액션들을 수행할 수 있는 임의의 객체를 포함한다. 이러한 필터는 사용자 모드 또는 커널 모드에서 실행될 수 있으며, 다른 컴포넌트들의 일부일 수 있다.

도 2로 돌아가면, 파일 시스템(235)은 어플리케이션(205)이 실행되는 기기 또는 기기들에 국부적으로 또는 원격으로 배치될 수 있는 하나 이상의 볼륨을 포함할 수 있다.

원격 컨텐츠를 국부적으로 캐싱하기

도 5는 본 발명의 다양한 양태에 따른 본 발명이 구현될 수 있는 예시적인 환경을 나타내는 블럭도이다. 이 환경은 파일 서버(505), 컨텐츠 서버(511-513) 및 클라이언트(520-523)를 포함하며, (도시되지 않은) 다른 컴포넌트들도 포함할 수 있다. 다양한 엔티티들은 인트라 네트워크 및 인터넷(515)을 포함하는 다양한 네트워크를 통해 서로 통신할 수 있다.

파일 서버(505)는 클라이언트가 이용할 수 있는 데이터센터로부터의 모든 객체의 세트(예를 들어, 디렉토리, 파일, 다른 컨텐츠 등)를 포함할 수 있다. 클라이언트에게 컨텐츠를 제공할 때, 컨텐츠 서버(511-513)는 이러한 객체에 액세스할 수 있다. 예를 들어, 컨텐츠 서버는 인터넷(515)을 포함하는 네트워크를 통해 클라이언트에게 컨텐츠를 서비스하는 웹 서버 어플리케이션을 호스팅할 수 있다. 컨텐츠 서버는 파일 서버(505)로부터의 객체에 액세스한 후 클라이언트에게 그 객체를 제공할 수 있다. 컨텐츠 서버는 파일 서버(505)로부터 획득한 객체를 캐싱할 수도 있고 하지 않을 수도 있다. 예를 들어, 자주 액세스되고 상대적으로 크기가 작은 객체는 캐싱되는 반면, 종종 액세스되지 않는 대용량 객체는 캐싱되지 않을 수 있다. 컨텐츠 서버는 메인 메모리(예를 들어, RAM) 및/또는 디스크와 같은 비활성 메모리 내에 객체를 캐싱할 수 있다. 어느 객체가 컨텐츠 서버에 캐싱될지는 컨텐츠 서버에 포함된 캐싱 컴포넌트[즉, 캐싱 컴포넌트(525-527) 중 하나]에 의해 판정된다.

한 구현에서, 컨텐츠 서버는 그들이 파일 서버(505)로부터 캐싱한 어떤 캐싱된 객체도 수정하지 않는다. 대신, 어플리케이션이 원격 객체를 수정하거나 새로운 객체를 생성할 것을 요청하면, 컨텐츠 서버는 파일 서버(505)에 그 요청을 전송한다. 이는 파일 서버(505) 상에 가장 최신의 컨텐츠 사본이 보유되도록 한다.

각각의 캐싱 컴포넌트(525-527)는 다음에 보다 상세하게 설명되어 있는 필터 및 캐싱 서비스를 포함하는 다양한 서브컴포넌트를 포함할 수 있다.

도 6은 본 발명의 다양한 양태에 따른 본 발명이 구현될 수 있는 예시적인 시스템 컴포넌트의 배치를 나타내는 블럭도이다. 이 시스템은 컨텐츠 서버 어플리케이션(605), 필터(610), 원격 파일 시스템(615), 캐싱 서비스(620) 및 국부 파일 시스템(625)을 포함하며, (도시되지 않은) 다른 컴포넌트들도 포함할 수 있다.

컨텐츠 서버 어플리케이션(605)은 원격 파일 시스템(615) 및 국부 파일 시스템(625)으로부터의 객체에 독립적으로 액세스할 수 있는 (도시되지 않은) 다양한 컴포넌트들을 포함할 수 있다. 몇몇 실시예에서, 컨텐츠 서버 어플리케이션(605)의 어떤 단일 컴포넌트도 컴포넌트 모두가 어떤 객체를 요청하는지 또는 이러한 객체가 어디에 (예를 들어, 국부적으로 또는 원격으로) 상주해 있는지를 모른다. 어떤 원격 객체가 요청되는지를 모니터링하기 위해, 필터(610)는 원격 파일 시스템(615)으로의 및 그로부터의 I/O를 모니터링할 수 있다. 주기적으로, 필터(610)는 원격 파일 시스템(615)으로부터 액세스되는 객체의 이름을 포함하는 리스트를 캐싱 서비스(620)에게 송신할 수 있다.

캐싱 서비스(620)는 그 후 필터(610)가 송신한 그 리스트를 이용하여, 국부 파일 시스템(625)에 원격 파일 시스템으로부터의 어떤 객체를 캐싱해야 할지를 판정한다. 이러한 판정은 관리자가 세운 정책 또는 객체가 액세스된 횟수, 객체의 크기 또는 임의의 다른 캐싱 규칙을 포함할 수 있는 그 정책과 유사한 것에 기초하여 내려질 수 있다.

한 실시예에서, 캐싱 서비스(620)는 사용자 모드에서 실행된다. 또 다른 실시예에서, 캐싱 서비스(620)는 커널 모드에서 실행된다. 또 다른 실시예에서, 캐싱 서비스(620)의 기능은 필터(610)에 의해 수행된다. 이 실시예에서, 필터(610)와 별개의 캐싱 서비스(620)는 불필요하다.

시스템 관리자 또는 그와 유사한 것은 캐싱 서비스(620)에게 객체를 어디에 캐싱할지를 알려주기 위한 레지스트리 키 또는 다른 구성 데이터를 생성한다. 예를 들어, 시스템 관리자는 네트 워크 공유 \\SERVER\SHARE 로부터 캐싱될 객체가 디렉토리 C:\CACHE 내에 배치되어야 함을 나타낼 수 있다. \\SERVER\SHARE의 서브 디렉토리 내의 객체가 캐싱되면, 유사한 디렉토리 구조를 유지하기 위해 객체의 임의의 상위 디렉토리도 C:\CACHE 내에 생성될 수 있다. 예를 들어, 캐싱 서비스(620)가 \\SERVER\SHARE\DOCUMENTS\COMPANY.HTML가 캐싱되어야 한다고 판정하면, DOCUMENTS라는 디렉토리가 C:\CACHE 내에 생성되고, 객체 COMPANY.HTML이 그 디렉토리 내에 배치될 수 있다.

어느 원격 객체가 액세스되는지를 모니터링하는 것 이외에, 필터(610)는 또한 원격 객체로 향해있는 요청의 방향을 국부적으로 캐싱된 객체로 바꾸도록 하는데 이용될 수도 있으며, 이는 다음에 보다 상세하게 설명된다.

도 7은 본 발명의 다양한 양태에 따른 본 발명을 구현할 수 있는 예시적인 시스템 컴포넌트의 배치를 나타내는 블럭도이다. 시스템은 컨텐츠 서버 어플리케이션(605), I/O 관리자(215), 필터(610), 리다이렉터(710), 원격 파일 시스템(615) 및 국부 파일 시스템(625)을 포함하며, 또한 (도시되지 않은) 다른 컴포넌트도 포함할 수 있다.

컨텐츠 서버 어플리케이션(605)의 임의의 컴포넌트가 원격 파일 시스템(615) 상에 있는 객체에 액세스할 것을 요청하면, 컴포넌트는 I/O 관리자(215)에게 객체의 이름(예를 들어, UNC 이름)을 알려준다. 임의의 리다이렉터가 그 이름에 대응하는 객체가 상주해 있는 곳을 아는지를 판정하기 위해, I/O 관리자(215)는 하나 이상의 리다이렉터[예를 들어, 리다이렉터(710)]와 상호작용한다. 이와 달리, 리다이렉터가 그 이름에 대응하는 객체가 어디에 상주해 있는지를 아는지 판정하기 위해, I/O 관리자(215)는 후에 각각의 이용가능한 리다이렉터와 상호작용하는 단일 컴포넌트와 상호작용할 수 있다. 리다이렉터가 응답하면, 리다이렉터는 객체가 상주해 있는 원격 서버와의 세션을 구축하는데 이용된다. 컨텐츠 서버 어플리케이션(605)에게 원격 객체를 액세스하기 위한 절차는 국부 객체를 액세스하기 위한 절차와 동일할 수 있다. 몇몇 구현에서, 컨텐츠 서버 어플리케이션(605)은 객체가 원격으로 또는 국부적으로 배치되어 있는지를 모를 수 있다.

도 6의 캐싱 서비스(620)가 실행되기 시작하면, 캐싱 서비스(620)는 원격 파일 시스템 이름(예를 들어, 네트워크 공유)을 국부 캐시 디렉토리 이름으로 매핑시킬 수 있다. 예를 들어, 이 매핑은 \\SERVER\SHARE가 C:\CACHE로 매핑되고 다른 원격 파일 시스템 이름이 다른 국부 캐시 디렉토리로 매핑됨을 나타낼 수 있다. 원격적으로 배치된 객체에 대한 요청의 방향을 국부 캐시 디렉토리로 바꿀 때를 판정하기 위해, 필터(610)는 이 정보를 저장해 둔다. 예를 들어, 필터는 매핑 테이블에 이 정보를 저장할 수 있다.

필터(610)가 원격 객체를 열기 위한 요청을 수신하면, 필터(610)는 요청이 국부 파일 시스템(625)으로 매핑되어야 하는지를 판정할 수 있다. 이것을 수행하기 위해, 필터(610)는 그 객체가 국부 파일 시스템(625) 상에 캐싱되어 있는지를 판정할 수 있다. 예를 들어, 필터(610)가 \\SERVER\SHARE\A.TXT이란 이름의 객체를 열기 위한 요청을 수신하면, 필터(610)는 매핑 테이블을 관찰하여 이 객체의 접두어에 대한 매핑이 있는지를 알아낼 수 있다. 이 경우에, 필터(610)는 \\SERVER\SHARE가 C:\CACHE로 매핑되어있음을 판정할 수 있다. 필터(610)는 그 후 객체 A.TXT가 C:\CACHE 내에 존재하는지를 판정할 수 있다. 존재하면, 필터(610)는 I/O 관리자(215)에게 그 요청의 방향을 국부 파일 시스템(625)으로 바꾸도록 명령할 수 있으며, 이는 다음에 보다 상세하게 설명된다.

도 8은 본 발명의 다양한 양태에 따라 I/O 동작의 방향을 국부 캐시로 바꿔야할지를 판정할 때 일어날 수 있는 액션들을 포괄적으로 나타내는 순서도이다. 블럭(805)에서, 프로세스가 시작된다.

블럭(810)에서, 필터가 생성 동작을 수신한다. 생성 동작은 객체를 생성하거나 이미 존재하는 객체를 열 수 있다. 다른 동작들은 필터에 의해 무시되고, 리다이렉터로 전달될 수 있다.

몇몇 구현에서, 어플리케이션은 디렉토리 내의 객체가 변경될 때를 통지받길 요청할 수 있다. 이러한 구현에서, 원격 서버가 변경이 일어났다는 통지를 송신하면, 필터는 도 10에 관련하여 더욱 상세하게 설명된 다른 액션을 취할 수 있다.

블럭(815)에서, 생성 동작이 쓰기 액세스를 위해 원격 객체를 열려고하는지가 판정된다. 액세스를 위해 원격 객체를 열려고하는 것이면, 브렌치는 블럭(830)으로 진행되며, 그렇지 않으면, 브렌치는 블럭(820)으로 진행된다. 생성 동작이 원격 객체를 수정하려고 하는 것이면(예로 이동), 필터는 리다이렉터로 동작을 전달하고, 이에 따라 원격 컨텐츠가 갱신된다. 이에 따라 국부적으로 캐싱된 컨텐츠가 아닌 본래 컨텐츠가 갱신된다.

블럭(820)에서, 동작이 디렉토리를 열려고 하는 것인지가 판정된다. 디렉토리를 열려고 하는 것이면, 브렌치는 블럭(830)으로 진행되고, 그렇지 않으면, 브렌치는 블럭(825)으로 진행된다. 컨텐츠 서버 어플리케이션이 디렉토리 정보를 요청하면, 전형적으로 국부적으로 캐싱되어 있는 객체들만이 아니라 디렉토리 내의 모든 객체들의 이름을 볼 수 있어야 한다. 원격 서버의 디렉토리를 평상시처럼 열 수 있도록 함으로써, 필터는 요청자가 원격 디렉토리 내의 모든 객체들을 포함하는 디렉토리 리스팅을 획득하게 할 수 있다. 디렉토리를 열 때 또는 열고 나서, 컨텐츠 서버 어플리케이션은 그 디렉토리 상에서 다른 방식(예를 들어, 디렉토리를 삭제하거나 변경 통지를 등록)으로 동작할 수 있다. 이러한 다른 동작들도 원격 서버에 송신될 수 있지만, 필터의 방향이 생성만을 향하도록 바뀌는 몇몇 실시예에서처럼, 이러한 다른 동작들은 필터의 상호작용 없이 송신될 수 있다.

블럭(825)에서, 생성 동작이 새로운 객체의 생성을 요청하는 것인지가 판정된다. 새로운 객체의 생성을 요청하는 것이라면, 브렌치는 블럭(830)으로 진행되며, 그렇지 않으면, 브렌치는 블럭(835)으로 진행된다. 생성 동작이 원격 객체를 생성하려고 하는 것이라면(예 브렌치), 필터는 그 동작을 리다이렉터에게 전달하며, 이에 따라 원격 컨텐츠가 갱신된다. 이는 객체를 국부 캐시에 생성시키지 않고 원격 파일 시스템 상에 생성시킨다. 블럭(815-825)에 관련된 판정들은 본 발명의 취지 및 영역을 벗어나지 않는 임의의 순서로 수행될 수 있음을 인식할 것이다. 한 실시예에서는, 가장 먼저 블럭(825)에 관련된 판정이, 다음에 블럭(815)에 관련된 판정이, 그리고 마지막으로 블럭(820)에 관련된 판정이 내려진다.

블럭(835)에서, 동작이 국부적으로 또는 원격 서버에 매핑되며, 이는 도 9에 관련하여 보다 상세하게 설명된다.

도 9는 본 발명의 다양한 양태에 따른, I/O 동작의 방향을 국부 캐시로 바꿔야하는지를 판정할 때 행해지는, 도 8의 블럭(835)에 대응하는 액션을 전반적으로 나타낸 순서도이다. 블럭(905)에서, 프로세스가 시작한다.

블럭(910)에서, 객체가 필터에 의해 유지되는 매핑 테이블에 의해 매핑되는지가 판정된다. 객체가 필터에 의해 유지되는 매핑 테이블에 의해 매핑되면, 브렌치는 블럭(915)으로 진행되고, 그렇지 않다면, 브렌치는 블럭(935)으로 진행된다. 객체가 매핑 테이블에 포함되어 있는 임의의 네트워크 공유 디렉토리의 자손(descendant)이면, 객체는 매핑 테이블에 의해 매핑된다.

블럭(915)에서, 객체가 캐싱되어 있는지가 판정된다. 캐싱되어 있으면, 브렌치는 블럭(920)으로 진행되며, 그렇지 않으면, 브렌치는 블럭(935)으로 진행된다. 한 실시예에서, 필터는, 매핑 테이블로부터 캐시 디렉토리를 획득하고 객체를 열려고 함으로써 객체가 캐싱된다고 판정할 수 있다. 또 다른 구현에서는, 캐싱된 객체가 메모리에 유지되고, 객체를 열려고 하지 않고도 객체가 캐싱되어있는지가 판정될 수 있다. 객체가 캐싱되어있는지를 판정하기 위한, 본 발명의 취지 및 영역을 벗어나지 않으면서 이용되는 다른 수많은 방식이 있음을 인식할 것이다.

블럭(920)에서, I/O가 새로운 이름으로 리파싱(reparse)된다. 반드시, I/O의 방향은 국부 파일 시스템의 캐싱된 객체로 바뀌게 된다. 몇몇 실시예에서, 이것은 I/O 관리자에게 STATUS_REPARSE를 반환하고 I/O 관리자에게 국부적으로 캐싱된 객체의 이름을 제공함으로써 이루어질 수 있다.

블럭(925)에서, I/O가 국부 파일 시스템으로 송신된다. I/O 관리자는 I/O가 어느 객체로 향해있었는지를 나타내는 정보를 저장하였다가, 향후에 필터에 의해 리파싱되거나 처리되지 않고서도 그 I/O에 관련된 임의의 다른 동작이 국부 파일 시스템에 직접 송신될 수 있게 한다.

리파싱 메커니즘이 없는 구현에서, 필터는 최근에 매핑된 I/O에 관련된 다음의 I/O들을 모니터링하여, 이러한 I/O들을 국부 파일 시스템에 향하도록 할 수 있다.

블럭(930)에서, 액세스된 객체가 캐싱 서비스에 통지된다. 앞서 설명된 바와 같이, 객체가 액세스될 때마다 통지를 송신하는 대신, 액세스된 객체의 리스트는 정기적으로 캐싱 서비스에 송신될 수 있다. 예를 들어, 캐싱 서비스가 국부 파일 시스템으로부터 캐싱된 객체를 언제 삭제해야 하는지를 판정할 수 있도록, 국부 캐싱된 객체에의 액세스에 대한 통지가 캐싱 서비스에 송신될 수 있다.

블럭(935)에서, I/O가 원격 파일 시스템에 송신된다. 이것은 앞서 설명된 바와 같이 리다이렉터 컴포넌트를 통해 행해질 수 있다.

블럭(940)에서, 객체가 원격으로 액세스되었다는 것이 캐싱 서비스에 통지된다. 또 다시, 앞서 설명된 바와 같이, 객체가 액세스될 때마다 통지를 송신하는 대신, 액세스된 객체의 리스트가 정기적으로 캐싱 서비스에 송신될 수 있다. 예를 들어, 캐싱 서비스가, 언제 원격 객체를 획득하고 그것을 국부 파일 시스템에 캐싱해야 하는지를 판정할 수 있도록 하기 위해, 원격 객체에의 액세스에 대한 통지가 캐싱 서비스에 송신될 수 있다.

블럭(945)에서, 프로세스가 반환된다.

도 10은 원격 객체가 본 발명의 다양한 양태에 따라 변경될 때 일어날 수 있는 액션들을 전반적으로 나타내는 순서도이다. 블럭(1005)에서, 프로세스가 시작된다.

블럭(1010)에서, 필터는 원격 서버에 포함되어 있는 객체가 변경되었다는 통지를 수신한다. 수정되었다면, 블럭(1015)에서, 그 객체가 국부적으로 삭제된다. 이것은 진부한(stale) 컨텐츠가 국부 캐시로부터 제거되었다는 것을 보증한다. 또한, 원격 객체가 다시 요청되면, 캐시에서 그것을 더 이상 찾을 수 없으므로, 필터는 그 요청을 원격 파일 시스템으로 전송할 것이다.

블럭(1020)에서, 이 통지가 컨텐츠 서버로 전송되고, 그 후 컨텐츠 서버는 객체의 가장 최신 복사본을 캐싱하기 위해 원격 객체로부터 그 객체의 가장 최신 복사본을 획득하려고 할 수 있다.

블럭(1025)에서, 프로세스가 반환된다.

도 11은 본 발명의 다양한 양태에 따른 본 발명이 구현될 수 있는 또 다른 예시적인 시스템 컴포넌트의 배치를 나타내는 블럭도이다. 시스템은 컨텐츠 서버 어플리케이션(605), I/O 관리자(215), 필터(1105), 리다이렉터(710), 원격 파일 시스템(615), 국부 파일 시스템(625)을 포함하며, (도시되지 않은) 다른 컴포넌트들도 포함할 수 있다.

몇몇 오퍼레이팅 시스템에서, 오퍼레이팅 시스템은 파일 시스템의 디렉토리에 대한 리파스 포인트(reparse point)를 제공할 수 있다. 리파스 포인트는 파일 시스템의 디렉토리에 관련된 데이터의 집합이다. 리파스 포인트의 데이터는 디렉토리에 관련된 캐싱된 객체가 존재하는 디렉토리 및 객체를 획득할 수 있는 원격 디렉토리를 나타낼 수 있다. 리파스 포인트에 관련된 디렉토리 또는 그것의 임의의 자손에 액세스하기 위한 동작이 수신되면, 리파스 포인트에 관련된 데이터와 함께 STATUS_REPARSE가 I/O 관리자(215)에게 반환된다. 리파스 포인트는 파일 시스템에 의해 유지되어, 파일 시스템을 올라가고(remount) 내려간(dismount) 후에도 영속될 수 있다.

국부 파일 시스템(710)이 STATUS_REPARSE로 응답하면, 필터(1105)는 객체가 국부적으로 캐싱되어 있는지를 판정하는데 리파스 포인트에 관련된 데이터를 이용할 수 있다. 객체가 국부적으로 캐싱되어 있으면, 필터는 I/O 관리자(215)가 I/O 동작의 방향을 국부적으로 캐싱된 객체로 바꾸도록 명령한다. 객체가 국부적으로 캐싱되어있지 않으면, 필터는 I/O 관리자(215)가 리다이렉터(615)를 통해 객체를 원격으로 획득하도록 명령할 수 있다.

도 8에 관련하여 설명된 바와 같이 원격 파일 시스템에 어떤 I/O 동작(예를 들어, 객체를 수정하고, 디렉토리를 열고, 객체를 생성하는 것)을 자동으로 전달하기 위한 조건이 도 11의 필터(1105)에도 적용된다. 이외에, 도 9의 블럭(930 및 940)에 관련된 보고 액션도 도 11의 필터(1105)에 의해 수행될 수 있다.

리파스 포인트를 이용함으로써, 하나 이상의 섀도우 디렉토리(shadow directory)가 국부 파일 시스템에 생성될 수 있다. 리파스 포인트는 국부 캐시 디렉토리를 나타내는 각각의 디렉토리 및 객체가 발견될 수 있는 원격 디렉토리에 관련될 수 있다. 이외에, 컨텐츠 서버 어플리케이션(605)은 (볼륨 이름 대신) 네트워크 이름을 이용하여 하나 이상의 섀도우 디렉토리를 통해 객체를 획득하도록 명령될 수 있다. 이것은 컨텐츠 서버 어플리케이션(605)이, 객체가 국부적으로 배치되어 있다고 잘못 판정되는 것을 피하기 위해 행해진다. 보안 위임(security delegation)은 또한 자격증(credential)이 원격 기기에 전달되게 할 수 있다.

상기 상세한 설명으로부터 알 수 있는 바와 같이, 원격 데이터를 국부적으로 캐싱하기 위한 방법 및 시스템이 제공된다. 본 발명이 다양한 수정물 및 상이한 구조를 가질 수 있지만, 여기에 예시된 특정한 실시예가 도면에 도시되고 앞서 상세하게 설명되어 있다. 그러나 본 발명을 개시된 그 특정 형태로 제한되도록 의도된 것은 아니며, 오히려, 본 발명은 본 발명의 취지 및 영역에 속하는 모든 수정물, 상이한 컴포넌트 및 등가물을 포함한다는 것을 이해해야 한다.

발명의 효과

본 발명은 원격 데이터를 국부적으로 캐싱하기 위한 방법 및 시스템을 제공한다.

(57) 청구의 범위

청구항 1.

원격 서버에 저장되어 있는 객체에 액세스하기 위한 요청을 수신하고,

상기 요청을 상기 원격 서버에 전송할지 또는 상기 요청의 방향을 국부 파일 시스템으로 바꿀지를 판정하기 위해, 상기 원격 서버로의 요청을 모니터링할 기회를 갖는 필터를 사용하여 상기 요청을 검사하고,

상기 객체가 캐싱(cache)되어 있으면, 상기 요청의 방향을 상기 국부 파일 시스템으로 바꾸는 컴퓨터 실행가능 명령어를 갖는 컴퓨터 판독가능 매체.

청구항 2.

제1항에 있어서, 상기 객체에 액세스하기 위한 요청이 상기 객체를 생성하기 위한 요청을 포함하면, 상기 요청을 상기 원격 서버에 전송하는 컴퓨터 실행가능 명령어를 더 포함하는 컴퓨터 판독가능 매체.

청구항 3.

제1항에 있어서, 상기 객체에 액세스하기 위한 요청이 상기 객체를 수정하기 위한 요청을 포함하면, 상기 요청을 상기 원격 서버에 전송하는 컴퓨터 실행가능 명령어를 더 포함하는 컴퓨터 판독가능 매체.

청구항 4.

제1항에 있어서, 상기 객체가 디렉토리를 포함하면, 상기 요청을 상기 원격 서버에 전송하는 컴퓨터 실행가능 명령어를 더 포함하는 컴퓨터 판독가능 매체.

청구항 5.

제1항에 있어서, 캐싱 서비스에게 상기 객체가 요청되었으며 상기 국부 파일 시스템에 포함되어 있지 않다는 것을 통지하는 컴퓨터 실행가능 명령어를 더 포함하는 컴퓨터 판독가능 매체.

청구항 6.

제5항에 있어서, 상기 원격 서버로부터 상기 객체를 획득하고 그것을 상기 국부 파일 시스템에 저장하는 컴퓨터 실행가능 명령어를 더 포함하는 컴퓨터 판독가능 매체.

청구항 7.

제1항에 있어서, 캐싱 서비스에게 상기 객체가 요청되었으며 상기 국부 파일 시스템에 포함되어 있다는 것을 통지하는 컴퓨터 실행가능 명령어를 더 포함하는 컴퓨터 판독가능 매체.

청구항 8.

제7항에 있어서, 상기 객체가 요청되는 빈도에 기초하여 상기 국부 파일 시스템에 상기 객체를 보관하는 컴퓨터 실행가능 명령어를 더 포함하는 컴퓨터 판독가능 매체.

청구항 9.

제1항에 있어서, 상기 필터는 커널 모드에서 실행되는 컴퓨터 판독가능 매체.

청구항 10.

제1항에 있어서, 상기 객체를 열고 닫고, 상기 필터를 사용해 입출력 동작을 검사하지 않으면서 상기 객체를 여는 것과 닫는 것 사이에 상기 객체에 영향을 주는 상기 입출력 동작의 방향을 제시하는 컴퓨터 실행가능 명령어를 더 포함하는 컴퓨터 판독가능 매체.

청구항 11.

제1항에 있어서, 상기 객체가 변경되었다는 통지를 수신하고, 상기 국부 파일 시스템 상에 있는 상기 객체의 캐싱된 복사본을 삭제하는 컴퓨터 실행가능 명령어를 더 포함하는 컴퓨터 판독가능 매체.

청구항 12.

제11항에 있어서, 상기 필터가 상기 통지를 수신하고, 상기 객체의 캐싱된 복사본을 삭제하는 컴퓨터 판독가능 매체.

청구항 13.

컴퓨팅 환경에서,

리파스 포인트(reparse point)에 관련된 객체에 액세스하기 위한 요청을 수신하는 단계- 상기 리파스 포인트는 국부 디렉토리 및 원격 디렉토리의 이름을 포함함 -;

상기 객체가 상기 국부 디렉토리에 캐싱되어 있는지를 판정하는 단계; 및

상기 객체가 상기 국부 디렉토리에 캐싱되어 있으면, 상기 국부 디렉토리를 통해 상기 객체에의 액세스를 제공하는 단계

를 포함하는 방법.

청구항 14.

제13항에 있어서, 상기 국부 디렉토리는 국부 파일 시스템에 저장되어 있고, 상기 원격 디렉토리는 원격 파일 시스템에 저장되어 있는 방법.

청구항 15.

제14항에 있어서, 상기 리파스 포인트는 상기 국부 파일 시스템 상에 영속되는 방법.

청구항 16.

제14항에 있어서, 상기 국부 파일 시스템이 상기 객체에 액세스하기 위한 시도를 수신하면, 상기 국부 파일 시스템은 상기 리파스 포인트를 반환하는 방법.

청구항 17.

제16항에 있어서, 상기 객체가 상기 국부 디렉토리에 캐싱되어 있는지를 판정하는 단계는, 필터가 상기 국부 파일 시스템이 반환한 상기 리파스 포인트를 검사하는 단계를 포함하는 방법.

청구항 18.

제17항에 있어서, 상기 객체가 상기 국부 디렉토리에 캐싱되어 있는지를 판정하는 단계는, 상기 필터가 상기 리파스 포인트로부터 상기 국부 디렉토리를 획득하고 상기 국부 디렉토리 내의 상기 객체를 열려고 시도하는 단계를 더 포함하는 방법.

청구항 19.

제18항에 있어서, 상기 국부 디렉토리 내의 상기 객체를 열려고하는 시도가 성공하면, 상기 요청의 방향을 상기 국부 디렉토리로 바꾸는 단계를 더 포함하는 방법.

청구항 20.

제18항에 있어서, 상기 국부 디렉토리 내의 상기 객체를 열려고하는 시도가 실패하면, 상기 요청의 방향을 상기 원격 디렉토리로 바꾸는 단계를 더 포함하는 방법.

청구항 21.

제17항에 있어서, 상기 필터는 상기 국부 파일 시스템 상의 객체에 액세스하기 위한, 사용자 모드 프로세스로부터의 임의의 요청을 검사할 기회를 갖는 방법.

청구항 22.

제13항에 있어서, 상기 객체가 상기 국부 디렉토리에 캐싱되어 있지 않으면, 상기 원격 디렉토리로부터 상기 객체를 획득하고 그것을 상기 국부 디렉토리에 캐싱하는 단계를 더 포함하는 방법.

청구항 23.

제13항에 있어서, 상기 객체에 액세싱하기 위한 요청이 상기 객체를 생성하는 요청이면, 상기 요청의 방향을 상기 원격 디렉토리로 바꾸는 단계를 더 포함하는 방법.

청구항 24.

제13항에 있어서, 상기 객체에 액세싱하기 위한 요청이 상기 객체를 변경시키는 요청이면, 상기 요청의 방향을 상기 원격 디렉토리로 바꾸는 단계를 더 포함하는 방법.

청구항 25.

제13항에 있어서, 상기 객체가 디렉토리이면, 상기 요청의 방향을 상기 원격 디렉토리로 바꾸는 방법.

청구항 26.

제13항에 있어서,

상기 객체는 파일 시스템에 상기 객체를 식별시키는 이름을 가지며,

상기 리파스 포인트는 상기 파일 시스템에 디렉토리를 식별시키는 이름을 갖는 디렉토리에 관련되며,

상기 객체 이름의 접두어가 상기 디렉토리의 이름을 포함하면, 상기 객체는 리파스 포인트에 관련된 방법.

청구항 27.

캐싱을 위한 장치로서,

객체들을 저장하도록 구성된 국부 파일 시스템;

어느 객체들이 상기 국부 파일 시스템에 저장되어 있는지를 판정하도록 구성된 컴포넌트; 및

입출력을 모니터링하고 상기 컴포넌트에게 액세스가 요청된 객체의 이름을 보고하도록 구성된 필터

를 포함하는 장치.

청구항 28.

제27항에 있어서, 상기 객체는 디렉토리 및 파일을 포함하는 장치.

청구항 29.

제27항에 있어서, 상기 필터는 상기 국부 파일 시스템에의 또는 그로부터의 임의의 통신을 검사할 기회를 갖는 장치.

청구항 30.

제27항에 있어서, 상기 필터는 원격 파일 시스템에의 또는 그로부터의 임의의 통신을 검사할 기회를 갖는 장치.

청구항 31.

제30항에 있어서, 상기 필터는 생성 동작에 관련된 상기 원격 파일 시스템에의 또는 그로부터의 임의의 통신을 통지받도록 등록 배치된 장치.

청구항 32.

제31항에 있어서, 상기 생성 동작은 이미 존재하는 객체를 열기 위한 동작을 포함하는 장치.

청구항 33.

제31항에 있어서, 상기 생성 동작은 새로운 객체를 생성하기 위한 동작을 포함하는 장치.

청구항 34.

제27항에 있어서, 상기 필터는,

원격 서버에 저장되어 있는 객체에 액세스하기 위한 요청을 수신하는 것; 및

상기 요청을 상기 원격 서버에 전송할지 또는 상기 요청의 방향을 상기 국부 파일 시스템으로 바꿀지를 판정하는 것

를 포함하는 액션들을 수행하도록 구성된 장치.

청구항 35.

제34항에 있어서, 상기 필터는, 상기 요청이 새로운 객체를 생성하거나, 기존 객체를 수정하거나 또는 디렉토리를 열기 위한 요청을 포함하면, 상기 원격 서버에 상기 요청을 전송하도록 더 구성된 장치.

청구항 36.

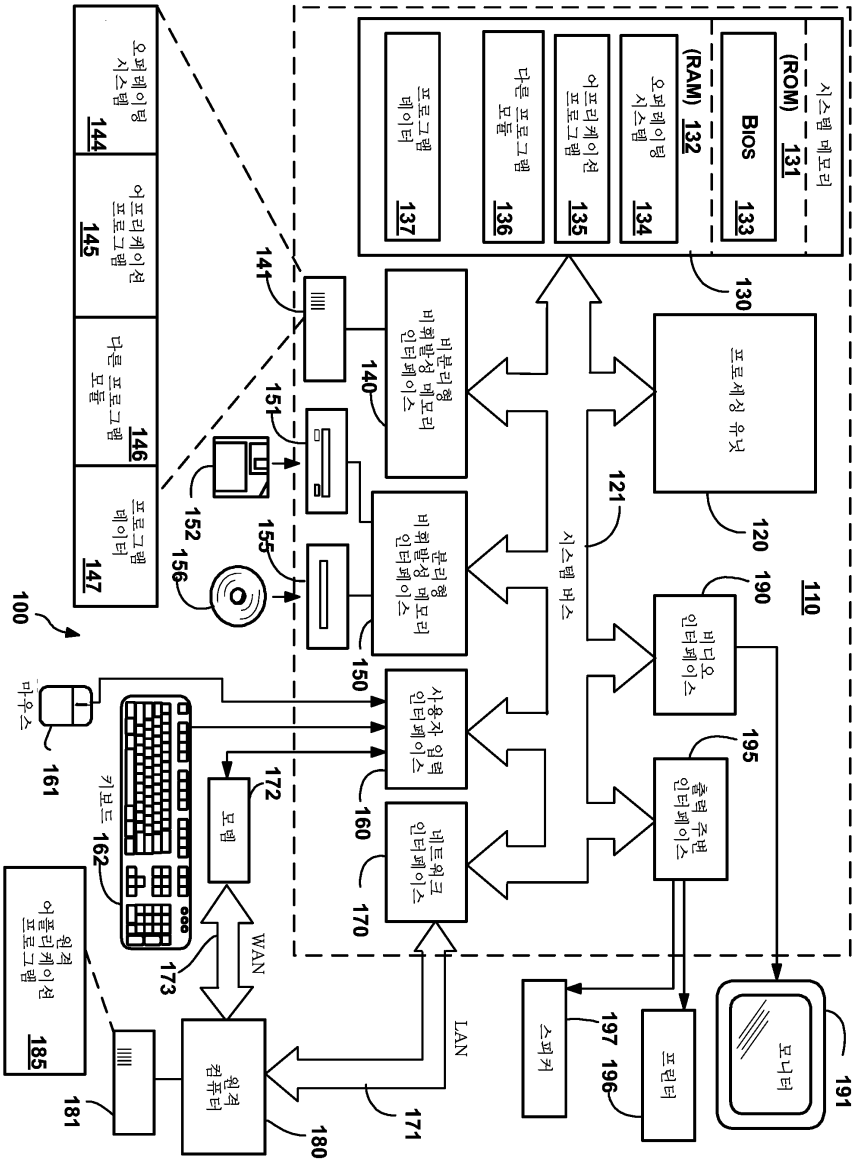
제34항에 있어서, 상기 필터는, 상기 객체가 상기 국부 파일 시스템에 캐싱되어 있으면, 상기 국부 시스템에 상기 요청을 전송하도록 더 구성된 장치.

청구항 37.

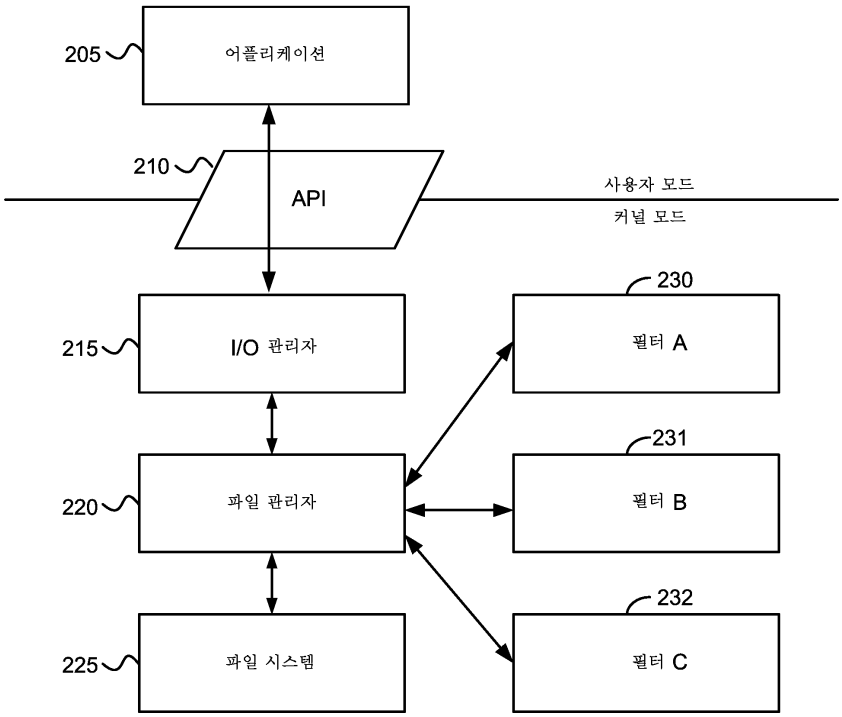
제27항에 있어서, 상기 컴포넌트는, 어떤 객체들이 상기 국부 파일 시스템에 저장되어 있는지를 판정하기 위해, 상기 필터가 보고한 상기 객체들에 정책을 적용시키는 장치.

도면

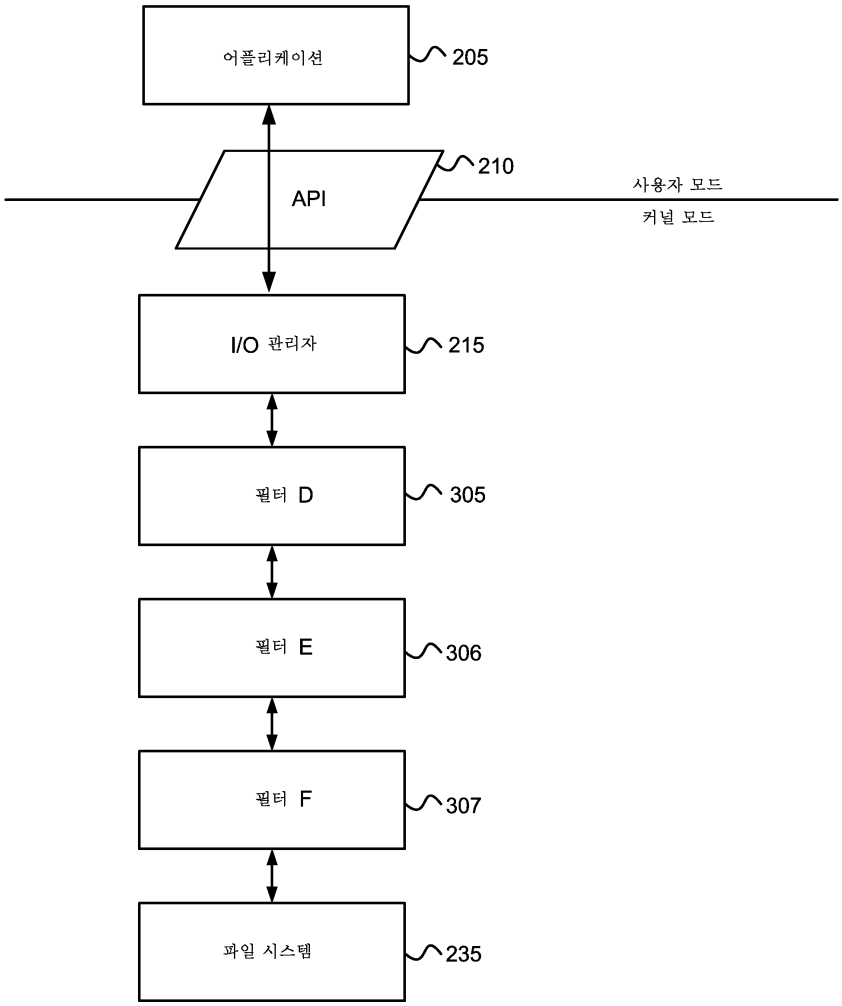
도면1


시스템 메모리
(ROM) 131
Bios 133
(RAM) 132
오퍼레이팅 시스템 134
어프리케이션 프로그램 135
다른 프로그램 모듈 136
프로그램 데이터 137
130
프로세싱 유닛
120
110
시스템 버스
121
비디오 인터페이스
190
출력주변 인터페이스
195
네트워크 인터페이스
170
사용자 입력 인터페이스
160
비휘발성 메모리 인터페이스
150
분리형 비휘발성 메모리 인터페이스
140
141
151
152
155
156
100
오퍼레이팅 시스템 144
어프리케이션 프로그램 145
다른 프로그램 모듈 146
프로그램 데이터 147
마우스
161
키보드
162
172
모뎀
173
WAN
LAN
원격 컴퓨터
180
171
원격 어플리케이션 프로그램 185
181
스피커
197
프린터
196
모니터
191

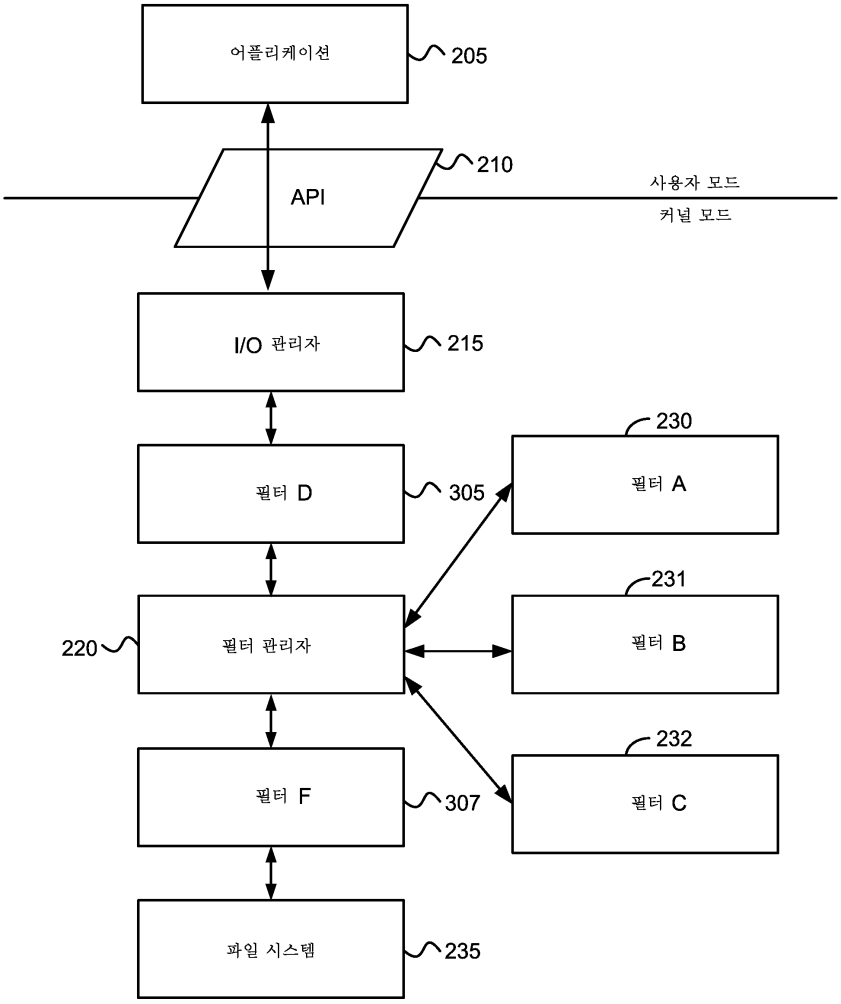
도면2


205
어플리케이션
210
API
사용자 모드
커널 모드
215
I/O 관리자
230
필터 A
231
필터 B
220
파일 관리자
232
필터 C
225
파일 시스템

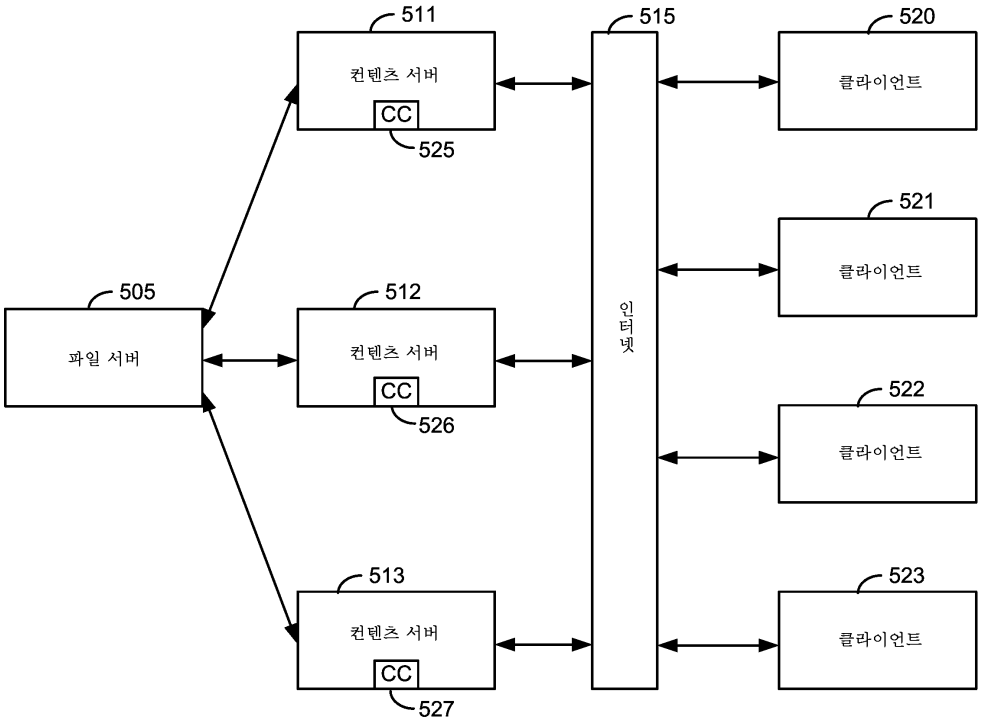
도면3


어플리케이션
205
210
API
사용자 모드
커널 모드
I/O 관리자
215
필터 D
305
필터 E
306
필터 F
307
파일 시스템
235

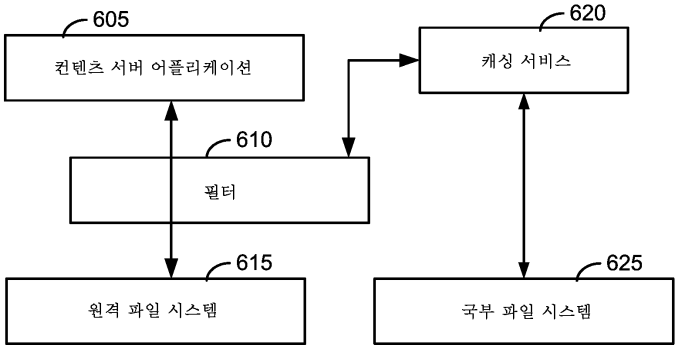
도면4


어플리케이션
205
API
210
사용자 모드
커널 모드
I/O 관리자
215
필터 D
305
필터 A
230
필터 관리자
220
필터 B
231
필터 F
307
필터 C
232
파일 시스템
235

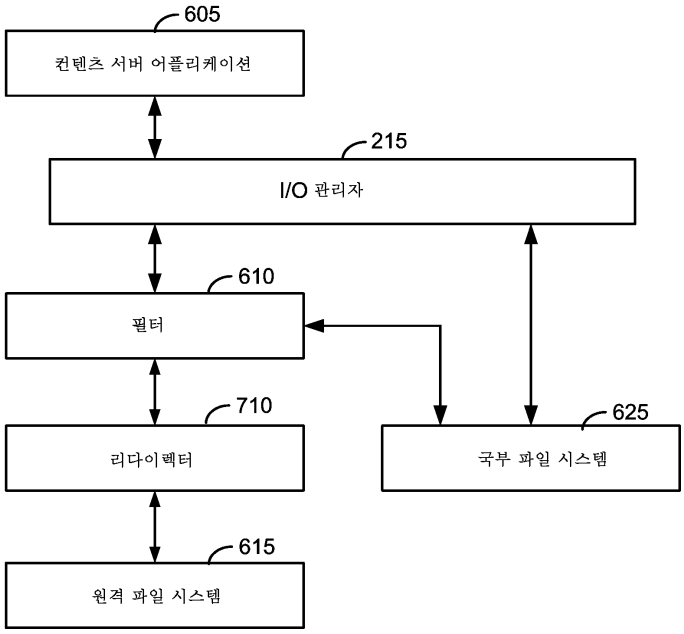
도면5


511
515
520
컨텐츠 서버
CC
525
클라이언트
521
505
512
인터넷
파일 서버
컨텐츠 서버
CC
526
클라이언트
522
513
523
컨텐츠 서버
CC
527
클라이언트

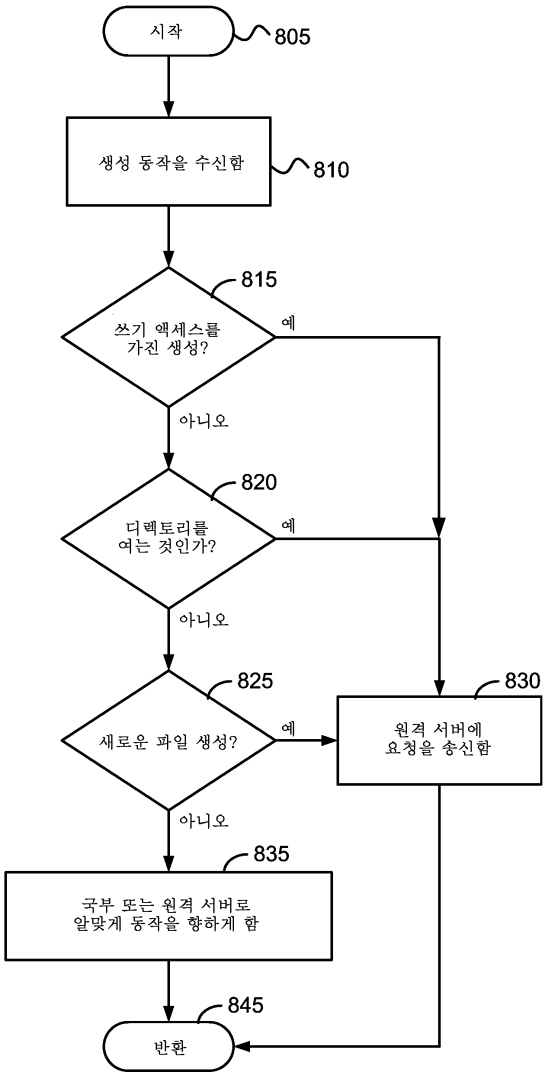
도면6


605
620
컨텐츠 서버 어플리케이션
캐싱 서비스
610
필터
615
625
원격 파일 시스템
국부 파일 시스템

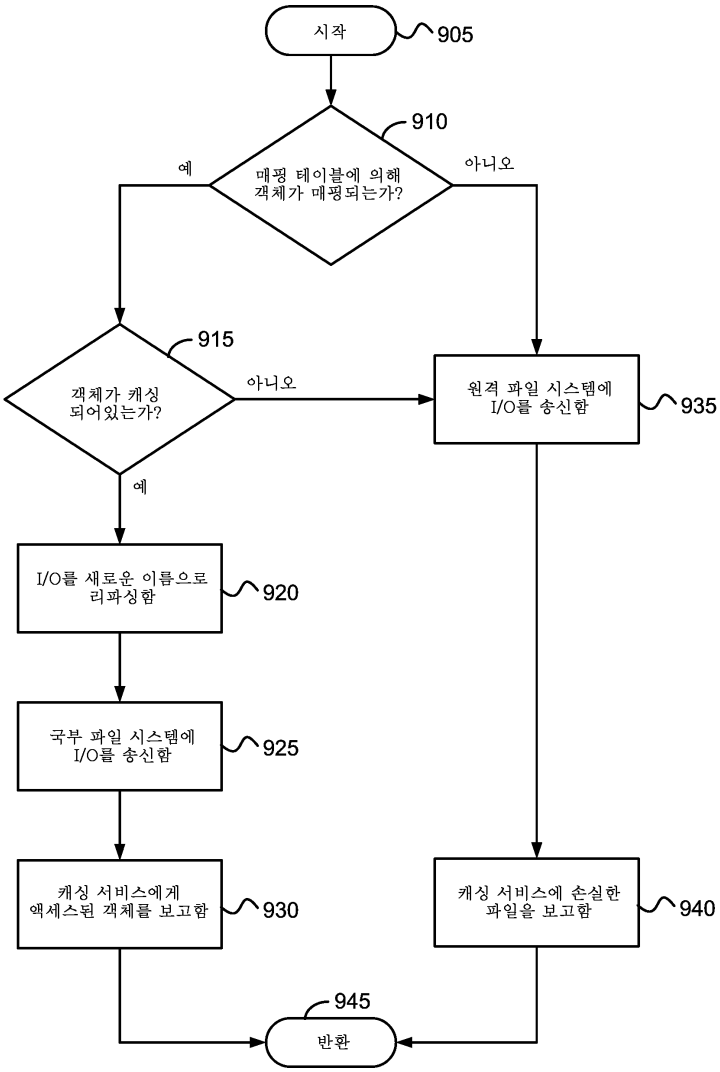
도면7


605
컨텐츠 서버 어플리케이션
215
I/O 관리자
610
필터
710
리다이렉터
625
국부 파일 시스템
615
원격 파일 시스템

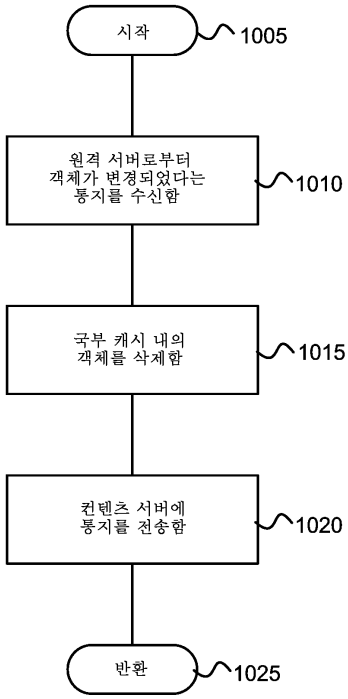
도면8


시작
805
생성 동작을 수신함
810
815
쓰기 액세스를 가진 생성?
예
아니오
820
디렉토리를 여는 것인가?
예
아니오
825
새로운 파일 생성?
예
830
원격 서버에 요청을 송신함
아니오
835
국부 또는 원격 서버로 알맞게 동작을 향하게 함
845
반환

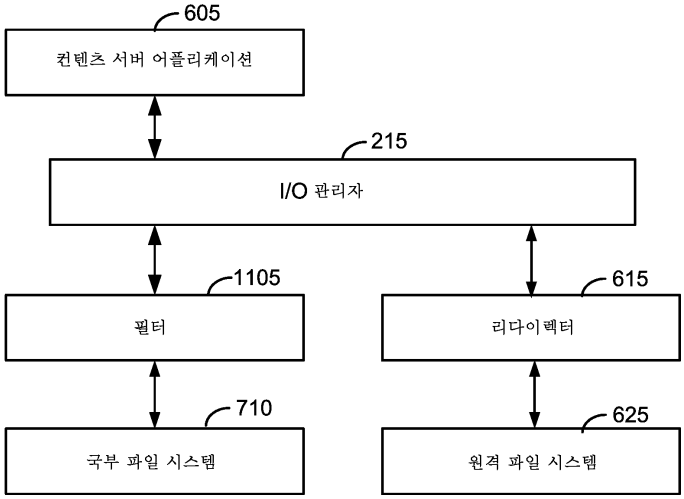
도면9


시작
905
910
매핑 테이블에 의해
객체가 매핑되는가?
예
아니오
915
객체가 캐싱
되어있는가?
아니오
예
원격 파일 시스템에
I/O를 송신함
935
I/O를 새로운 이름으로
리파싱함
920
국부 파일 시스템에
I/O를 송신함
925
캐싱 서비스에게
액세스된 객체를 보고함
930
캐싱 서비스에 손실한
파일을 보고함
940
945
반환

도면10


시작
1005
원격 서버로부터
객체가 변경되었다는
통지를 수신함
1010
국부 캐시 내의
객체를 삭제함
1015
컨텐츠 서버에
통지를 전송함
1020
반환
1025

도면11


605
컨텐츠 서버 어플리케이션
215
I/O 관리자
1105
필터
615
리다이렉터
710
국부 파일 시스템
625
원격 파일 시스템