(19) 대한민국특허청(KR)
(12) 등록특허공보(B1)

(51) 。Int. Cl. [7]
G06F 17/30

(45) 공고일자 2003년03월26일
(11) 등록번호 10-0377715
(24) 등록일자 2003년03월13일

(21) 출원번호 10-2000-7002247
(22) 출원일자 2000년03월02일
번역문 제출일자 2000년03월02일
(86) 국제출원번호 PCT/GB1998/02920
(86) 국제출원출원일자 1998년09월28일
(65) 공개번호 특2001-0023599
(43) 공개일자 2001년03월26일
(87) 국제공개번호 WO 1999/17227
(87) 국제공개일자 1999년04월08일

(81) 지정국 국내특허 : 캐나다, 중국, 체코, 헝가리, 이스라엘, 일본, 대한민국, 폴란드, 러시아,
EP 유럽특허: 오스트리아, 벨기에, 스위스, 독일, 덴마크, 스페인, 프랑스, 영국, 그리스, 아일랜드, 이탈리아, 룩셈부르크, 모나코, 네덜란드, 포르투칼, 스웨덴, 핀랜드, 사이프러스,

(30) 우선권주장 08/939,277 1997년09월29일 미국(US)

(73) 특허권자 인터내셔널 비지네스 머신즈 코포레이션
미국 10504 뉴욕주 아몬크

(72) 발명자 말킨피터켄네스
미국10502뉴욕주아드슬리브램블브룩로드64
유필립시-룽
미국10514뉴욕주챔파큐스토너웨이18

(74) 대리인 김창세
장성구
김원준

심사관 : 류동현

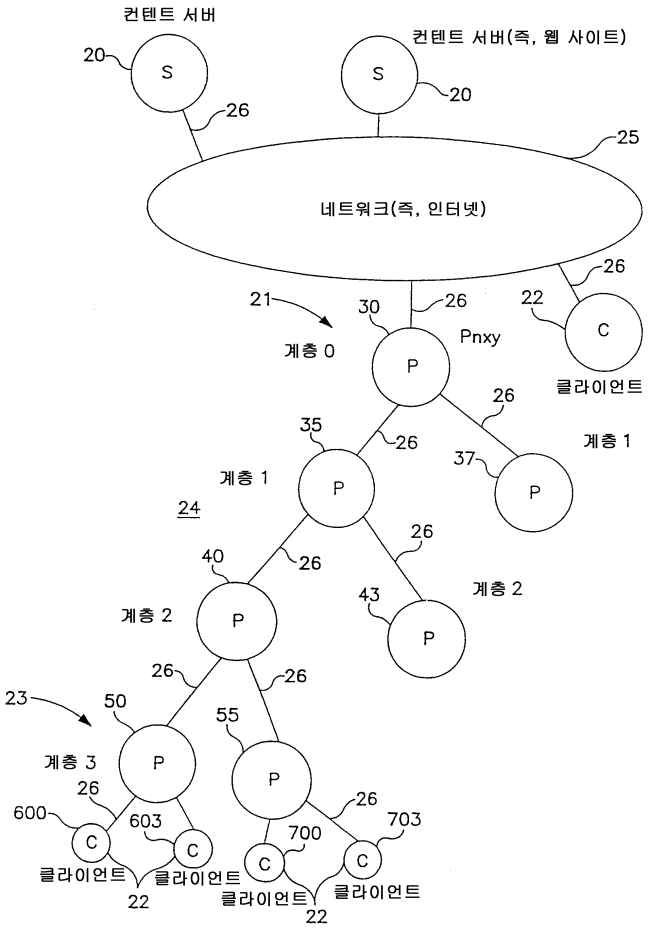
(54) 처리 시스템

요약

본 발명은 프록시 서버 계층과 연관된 클라이언트를 위한 데이터 사전 인출 및 클라이언트로의 개선된 컨텐트 서버 데이터 전송(an enhanced content server data transfer)을 위한 방법 및 시스템에 관한 것이다. 데이터를 사전 인출하는 방법은 데이터 액세스 패턴을 식별하고 상기 데이터 액세스 패턴의 동적 해석에 기초하여 선택 정보를 사전 인출한다. 컨텐트 서버나 프록시 서버는 컨텐트 서버나 프록시 서버와 연관된 클라이언트의 데이터 액세스 조회 패턴을 식별한다. 데이터 액세스 패턴은 통계 테이블 내에 저장되는 것이 바람직하다. 컨텐트 서버는 상기 통계 테이블로부터의 입

력에 기초하여 사전 인출 힌트 정보를 발생시킨다. 사전 인출 힌트 정보는 클라이언트에 의해 요청된 정보 또는 객체에 주석을 단다(annotate). 사전 인출 힌트 정보는 요청된 정보가 프록시 서버 계층을 통과할 때 갱신된다. 사전 인출 힌트 정보는 클라이언트에 의해 요청된 정보에 주석을 단다. 클라이언트에 대해 선택 정보를 사전 인출하는 궁극적인 판단은 사전 인출 힌트 정보와 사전 인출 힌트 값에 기초하여 행해진다. 사전 인출 힌트 값은 저장 제한 조건(storage constraints)이나 액세스 시간과 같은 지역 노드-고유 인자(local node-specific factors)에 의해 계산된다.

대표도



명세서

기술분야

본 발명은 클라이언트를 위한 정보 또는 객체의 동적 사전 인출(DYNAMIC PREFETCHING) 방법 및 장치에 관한 것으로서, 클라이언트는 월드 와이드 웹(World Wide Web) 상에서의 웹 객체의 사전 인출과 관련되는 것과 같이 프록시 서버의 계층과 연관되는 것이 바람직하다.

배경기술

용어 해설(GLOSSARY OF TERMS)

인터넷(Internet) : TCP/IP 짝(suite) 프로토콜 을 사용하는 네트워크나 게이트웨이(gateway)의 네트워크이다.

클라이언트(Client) : 커맨드와 관련된 작업을 수행하는 서버로 커맨드를 발행(issue)하는 컴퓨터이다.

서버(Server) : 다른 컴퓨터의 커맨드에 따라 작업을 수행하는 임의의 컴퓨터이다. 웹 서버는 전형적으로 하나 이상의 클라이언트를 지원한다.

월드 와이드 웹(World Wide Web : WWW 또는 Web) : 사용자가 (하이퍼링크(hyperlinks)로 알려진) 하이라이트(highlight)된 관심 단어나 구(phrases)를 선택("클릭 온(click on)")함으로써, 서버에서 서버로, 데이터베이스에서 데이터베이스로 인터넷 스위치 상에서 정보를 검색하도록 하는 인터넷 애플리케이션이다. 인터넷 WWW 서버는 클라이언트를 지원하고, 클라이언트에게 정보를 제공한다. 웹은 모든 자원이 어드레스(address)되는 인터넷으로 간주되며, HTML을 이용하여 URL에 대응하는 정보를 디스플레이하고, 다른 URL에 대해 포인트-및-클릭(point-and-click) 인터페이스를 제공한다.

URL(Universal Resource Locator) : 인터넷 상의 정보를 고유하게 식별하거나 어드레스하는 방법으로서, 전자 우편 어드레스의 웹 문서 버전(Web document version)으로 간주될 수 있다. URL은 하이퍼링크를 이용하여 액세스할 수 있다. URL 식별(identification)의 예로는 "http://www.hpilipyu.com:80/table.html"이 있다. URL은 4 가지 구성 요소를 갖는다. 이 예의 왼쪽부터 시작해서, 첫 번째 요소는 ":"에 의해 로케이터(locator)의 나머지 부분으로부터 구별되며 사용할 프로토콜(이 경우는 http)을 명시한다. 다음 요소는 목표 호스트의 호스트네임(hostname) 또는 IP 어드레스로서, 이 요소는 왼쪽은 이중 슬래쉬"//", 오른쪽은 단일 슬래쉬"/"(또는 선택적으로 ":")로 제한된다. 포트 번호는 선택 사양이며 호스트네임으로부터 왼쪽은 ":", 오른쪽은 단일 슬래쉬 "/"로 제한된다. 네 번째 요소는 실제 파일 이름이나 프로그램 이름(이 경우에는, table.html)으로, 단일 슬래쉬를 더 이용하여 통상적인 방식으로 디렉토리나 서브 디렉토리를 분리한다. 이 예에서, ".html"은 이것이 HTML 파일임을 나타내는 확장자이다.

HTML(HyperText Markup Language) : 웹 서버에 의해 사용되어 웹 클라이언트에 의해 관찰되는 문서를 생성하고 접속하는 언어이다. HTML은 하이퍼텍스트 문서를 발생시킨다.

HTTP(HyperText Transfer Protocol) : 클라이언트로부터 서버로의 모든 요청이 독립적으로 처리되는 무상태(stateless) 프로토콜이다. 서버는 이전 접속에 대한 기록이 없다. URL의 시작 부분의 "http"는 요청하는 클라이언트와 목표 서버가 지정된 자원에 관한 HTTP 프로토콜을 이용하여 통신해야 함을 나타낸다.

인터넷 브라우저 또는 웹 브라우저(Internet Browser or Web browser) : HTTP와 같은 인터넷 프로토콜을 실행하고 그 결과를 사용자의 스크린 상에 디스플레이하는 그래픽 인터페이스 도구(a graphical interface tool)이다. 브라우저는 사용자가 인터넷을 "서핑(surf)"할 때 그림같은 데스크탑(pictorial desktops), 디렉토리, 검색 도구와 함께 인터넷 여행 가이드(Internet tour guide) 역할을 할 수 있다. 이 애플리케이션에서 웹 브라우저는 월드 와이드 웹과 통신하는 클라이언트 서비스이다.

클라이언트 캐쉬(Client cache) : 전형적으로 클라이언트에 의해 액세스되는 객체의 주된 그룹(캐쉬)으로서 사용된다. WWW 환경에서, 클라이언트 캐쉬는 전형적으로 웹 브라우저에 의해 구현되고, 현재 호출(current invocation) 중에 액세스되는 캐쉬 객체, 즉 비영속(non persistent) 캐쉬일 수도 있고, 호출과 교차하는(across invocations) 객체를 캐싱할 수도 있다.

캐싱 프록시(Caching proxies) : 네트워크 내에서 클라이언트를 대신해서, 캐쉬 복사본(a cached copy)을 반환하여 객체를 배치하는 에이전트(agent) 역할을 하는 전문화된 서버이다. 캐싱 프록시는 클라이언트 캐쉬로부터의 캐쉬 실패(cache misses)의 결과로서 호출되므로 전형적으로 2 차 계층 이상의 캐쉬 역할을 한다.

META-태깅(tagging) : 주어진 객체와 정보의 집합이다. 예를 들어, HTTP에서, 정보는 HTTP 헤더의 필드 내에 있는 요청 및 응답 모두와 연합될 수 있다. 예를 들어, HTTP 클라이언트는 돌려보낸 페이지(a returned page)가 요청되는 URL을 명시할 수 있다.

월드 와이드 웹(WWW 또는 Web)의 인기가 증가함에 따라, 인터넷을 통한 트래픽도 상당히 증가해왔다. 웹은 이제 네트워크 성능의 주된 병목(bottlenecks) 중 하나가 되었다. 예컨대, 웹 사용자는 느린 네트워크 링크를 통해 서버에 접속한 사용자가 문서나 정보를 요청할 때 상당한 지연을 경험할 것이다. 웹이 네트워크를 통해 사용자에게 정보를 전송하는 것은 네트워크를 통한 트래픽의 계층을 더 증가시킨다. 증가된 트래픽은 다른 사용자의 요청에 이용할 수 있는 대역폭을 감소시켜, 추가적인 잠재 지연을 일으킨다.

클라이언트를 위한 액세스 지연을 감소시키기 위해, 종래 기술은 네트워크 노드에서 사용자에게 보다 가까운 곳에 인기있는 정보의 문서의 복사본을 저장하거나 캐싱(cache)해서, 액세스 지연을 보다 수용가능하게 한다. 캐싱은 네트워크상의 다양한 지점에서 구현될 수 있다. 예를 들어, 큰 대학이나 회사는 자신의 지역 캐쉬를 소유할 수 있어서, 이곳으로부터 그 네트워크에 가입한 모든 사용자들이 문서를 인출할 수 있다. 지역 캐쉬는 전문 서버 형태로 구현될 수 있다.

전문 서버는, 캐싱 프록시라 불리는데, 클라이언트를 대신해서 요청된 정보의 임의의 잠재적인 캐싱 복사본을 배치하는 에이전트 역할을 할 수 있다. 캐싱 프록시는 실패한 클라이언트 캐쉬로부터의 남겨진 실패(misses)와만 관련되므로 보통 2 차 계층 이상의 캐쉬 역할을 한다. 클라이언트 캐쉬는 다양한 웹 브라우저로 구축된다. 클라이언트 캐쉬는 클라이언트 호출 중에 액세스된 문서(모자이크(Mosaic)와 같은 비영속 캐쉬)만을 저장할 수도 있고, 호출과 교차하여 문서를 캐싱할 수도 있다.

웹 문서를 캐싱하는 이전의 작업은 온-디맨드 캐싱(on-demand caching) 영역에 주로 존재해왔다. 온 디맨드 캐싱은 객체가 요청된 이후에 캐싱이 결정되는 것을 의미한다. 이를테면, 에이브람즈(Abrams) 등의 "Caching Proxies " Limitations and Potentials", Proc. 4th International World Wide Web Conference(1996)는 웹을 위해 조사된 서로 다른 온 디맨드 캐싱 방침을 설명하고 있다.

인터넷 서비스 제공자(Internet Service Provider : ISP) 네트워크나 협력 네트워크에서, 프록시 계층 내의 서로 다른 지역 프록시나 부문 프록시(departmental proxies)는 서로 다른 조회 습성을 관찰하거나 서로 다른 네트워크 트래픽과 지연을 직면할 수 있다. 프록시 계층의 존재는 모든 중간 프록시, 클라이언트, 이들의 부분집합이 사전 인출(prefectch)을 수행할 지여 여부에 대한 사전 인출 결정을 더 복잡하게 한다. 따라서, 프록시 서버 계층과 관련 클라이언트에 대해 효과적으로 사전 인출 활동을 조정할 수 있는 사전 인출 방법 및 시스템이 필요하다.

따라서, 본 발명은 컨텐트 서버(a content server)로부터 선택 정보(select information)를 사전 인출(prefetch)하여 통신 네트워크와 하나 이상의 프록시 서버를 통해 컨텐트 서버와 연관된 하나 이상의 클라이언트에 대해 선택 정보가 용이하게 액세스될 수 있도록 하는 방법을 제공하는데, 이 방법은 프록시 서버 중 적어도 하나로부터 정보를 요청하는 단계와, 클라이언트 중 어느 하나에 의한 이전 인출 또는 요청된 정보로의 액세스에 기초하여 클라이언트를 위한 사전 인출 힌트 정보(prefetch hint information)를 설정하는 단계와, 사전 인출 힌트 정보를 이용하여 요청된 정보에 주석(annotate)을 다는 단계와, 사전 인출 힌트 정보 및 요청 정보에 기초하여 서버 중 어느 하나로부터 선택 정보를 사전 인출하는 단계를 포함한다.

전형적으로, 요청된 정보는 요청된 객체를 포함하고, 선택 정보는 인터넷 데이터와 같은 사전 인출 후보 객체를 포함한다. 이 경우, 요청된 객체는 인터넷 데이터와 연관된 HTTP 링크를 통해 선택 객체를 조회할 수 있는데, 프록시 서버는 요청된 객체의 객체 헤더 내에 저장된 요청된 객체 URL의 HTTP 프로토콜을 통해 요청된 객체에 대한 선택 객체의 관계를 식별한다.

바람직하게는, 사전 인출 힌트 정보 설정 단계는 클라이언트와 관련된 데이터 조회 패턴(data reference patterns)을 추적하는 단계와, 요청된 객체와 사전 인출 후보 객체 사이의 관계에 따라 데이터 조회 패턴을 객체 식별자 그룹으로 조직하는 단계와, 상기 그룹 중 어느 것이 사전 인출 후보 객체와 요청된 객체 사이에서 선택 정보로서 사전 인출 후보 객체의 사전 인출을 보장하기에 충분한 관계를 갖는지를 결정하는 단계를 포함한다. 따라서, 상기 클라이언트의 데이터 조회 패턴은 요청된 정보의 과거의 인출이나 이전 액세스에 기초하는 것이 유리한데, 데이터 조회 패턴은 상기 프록시 서버 중 하나와 연관된 상기 클라이언트 중 다른 하나에 의한 과거의 인출이나 액세스 내에 존재하는 임의의 공통적인 경향을 나타내도록 선택된다. 이는 사전 인출 힌트 정보를 정의하는 데 사용될 수 있다.

데이터 조회 패턴 추적 및 조직 단계는 조회 데이터 패턴을 프록시 서버에 저장하는 데이터베이스를 설정하는 단계와, 각각의 데이터베이스 내에 필드를 배열하되, 필드는 객체 식별자와 객체 카운트(object counts)를 포함하고, 객체 식별자는 요청된 객체를 서로 구별하며, 객체 카운트는 요청된 객체가 상기 서버 중 하나에 의해 요청된 회수를 나타내는 단계와, 데이터베이스와 액세스 리스트(access lists)를 결합하되, 각각의 액세스 리스트는 각각의 요청된 객체와 연관된 하나 이상의 사전 인출 후보 객체를 포함하는 단계를 포함한다.

이러한 데이터베이스는 요청된 객체와 사전 인출 후보 객체 사이의 관계를 규정하고 요청된 객체와 사전 인출 후보 객체에 관한 액세스 통계를 추적하는 데 사용되는데, 액세스 통계는 요청된 객체의 부모 객체(a parent-object) 액세스 빈도와 사전 인출 후보 객체의 자녀 객체(a child0object) 액세스 빈도를 포함하고, 부모 객체 액세스 빈도는 요청된 객체 중 특정한 하나가 상기 클라이언트 중 적어도 하나에 의해 요청된 회수를 나타내고, 자녀 객체 액세스 빈도는 특정한 하나의 액세스 바로 다음에 사전 인출 후보 객체 중 하나 이상이 실제로 요청된 회수를 나타낸다. 이로 인해, 자녀 객체 액세스 빈도와 연관된 상기 부모 객체 액세스 빈도에 대한 상기 자녀 객체 액세스 빈도의 비율에 기초하여 사전 인출 후보 객체의 사전 인출 리스트를 선택할 수 있는데, 사전 인출 리스트는 참조 확률을 초과하는 비율을 갖는 사전 인출 후보 객체만을 포함하게 된다.

일반적으로, 사전 인출 후보 객체의 서브셋만이 실제로 사전 인출되고 상기 프록시 서버 중 하나 또는 상기 클라이언트 중 하나에서 수신되는데, 전형적으로 이러한 사전 인출 후보 객체를 선택 정보로서 선택하는 것은 상기 프록시 서버 중 하나에서 이용가능한 저장 크리 및/또는 상기 프록시 서버 중 하나에서의 액세스 시간과 같은 인자에 기초한다. 따라서, 사전 인출 후보 객체는 소정의 대응하는 사전 인출 값과 연관된 우선 순위에 따라 사전 인출된다.

바람직한 실시예에서, 요청된 정보에 주석을 다는 단계는 인터넷 컨텐트 선택을 위한 플랫폼(a Platform for Internet Content Selection : PICS) 포맷, 또는 PICS 포맷과 호환가능한 라벨 프로토콜(a label protocol)을 갖는 사전 인출 라벨 내에 상기 사전 인출 힌트 정보를 배치하는 단계를 포함한다. 이러한 PICS 포맷의 구체적인 구현에 있어서, 객체 식별자 카테고리와 조회 식별자 카테고리(referrer identifier category)가 정의되고, 상기 방법은 상기 조회 식별자 카테고리를 상기 컨텐트 서버로 돌려보내, 실제로 조회된 임의의 사전 인출 후보 객체를 식별하는 단계를 더 포함한다. 인터넷 컨텐트 선택 프로토콜이나 PICS 호환가능 프로토콜을 위한 플랫폼에 따라 사전 인출 힌트 정보를 조직하는 것은 이미 많은 브라우저에 의해 지원되는 포맷을 이용한다.

전형적으로 상기 프록시 서버에는 계층이 존재한다. 송신 노드(a sending node)와 수신 노드(a receiving node)가 정의되는데, 사전 인출 힌트 정보는 초기 상태에서 최종 상태로 수정하는 것이 바람직하고, 초기 상태는 송신 노드에 의해, 최종 상태는 상기 수신 노드에 의해 결정된다. (사전 인출 선택 정보가 수신 노드에 의해 개시된다고 가정하면) 상기 최종 상태의 사전 인출 힌트 정보는 궁극적인 사전 인출 결정을 위한 기초로서 형성된다.

달리 말해서, 사전 인출 힌트 정보는 계층을 통해 프록시 서버로부터 클라이언트에게 전달될 때 해석되고 갱신될 수 있다. 바람직하게, 프록시 서버는 요청된 정보에 대한 지역 조회(local references)가 존재하고 상기 지역 계층 서버의 카운트가 최소 문턱 카운트를 초과하면(즉, 지역적으로 믿을 수 있는 통계를 이용할 수 있으면), 상기 사전 인출 힌트 정보를 갱신한다. 또한, 사전 인출 힌트 정보는 프록시 계층 내의 캐싱 상태(예를 들어, 계층 내의 어느 곳에서 특정한 사전 인출 후보 객체가 캐싱될 수 있는지를 나타냄)에 기초하여 갱신될 수 있다.

본 발명은 또한 네트워크(25)를 통해 컨텐트 서버(20)로부터 선택 정보를 사전 인출하는 데이터 처리 시스템을 제공하는데, 상기 시스템은 네트워크와 연관된 하나 이상의 프록시 서버(24), 컨텐트 서버와 하나 이상의 상기 프록시 서버와 상기 네트워크를 통해 통신하는 클라이언트를 포함하며, 상기 시스템은 프록시 서버 중 적어도 하나로부터 정보를 요청하는 수단과, 클라이언트 중 어느 하나에 의한 이전 인출 또는 요청된 정보로의 액세스에 기초하여 클라이언트를 위한 사전 인출 힌트 정보를 설정하는 수단과, 사전 인출 힌트 정보를 이용하여 요청된 정보에 주석을 다는 수단과, 사전 인출 힌트 정보 및 요청 정보에 기초하여 서버 중 어느 하나로부터 선택 정보를 사전 인출하는 수단을 포함한다.

바람직한 실시예에서, 요청된 정보는 요청된 객체를 포함하며, 선택 정보는 사전 인출 후보 객체를 포함하고, 상기 각각의 서버와 클라이언트 중 적어도 하나는 주된 캐쉬와 사전 인출 캐쉬 사이에 적어도 논리적 분리(logical separation)를 갖는 메모리를 구비하며, 상기 사전 인출 선택 정보는 상기 사전 인출 캐쉬에 저장된다.

사전 인출 수단은 객체를 사전 인출하고 사전 인출된 객체를 우선 순위 방안(a priority scheme)에 기초하여 사전 인출 캐쉬에 저장하되, 우선 순위 방안은 최고 사전 인출 값을 갖는 객체를 사전 인출 캐쉬의 이용가능한 메모리에 대해 먼저 저장하고, 사전 인출 값은 하나 이상의 클라이언트에 의한 사전 인출 객체에 대한 개별적이거나 통합된 조회 가능성(reference probability)에 기초하는 것이 바람직하다. 특히, 사전 인출 값은 사전 인출 힌트 정보 리스트 상의 사전 인출 후보 객체가 클라이언트 중 특정한 하나에 의해 액세스될 추정된 액세스 확률과 사전 인출 후보 객체 중 선택된 하나를 저장하는 데 필요한 저장 크기에 기초한 객체의 액세스 비용에 기초할 수 있다.

바람직한 실시예에서, 사전 인출 수단은 사전 인출 값을 계산하는 계산 수단과 계산 수단에 의해 계산된 사전 인출 값을 평가하는 평가 수단을 포함하는 사전 인출 처리기를 포함하고, 계산 수단은 사전 인출 힌트 정보 리스트 상의 각각의 사전 인출 객체 식별자에 대해 사전 인출 값을 계산하고, 평가 수단은 다수의 후보 사전 인출 객체의 사전 인출 값을 평가하고 사전 인출 힌트 정보 리스트 상의 최고 사전 인출 값과 연관된 후보 사전 인출 객체의 캐싱의 우선 순위를 매긴다. 사전 인출 수단은 프록시 서버 중 적어도 하나 안에 제 1 사전 인출 처리기를, 클라이언트 중 적어도 하나 안에 제 2 사전 인출 처리기를 포함한다(즉, 사전 인출은 계층 내의 서로 다른/다수의 장소에서 수행될 수 있다).

또한, 바람직한 실시예는 클라이언트 중 하나의 메모리에 저장된 클라이언트 객체 요청 처리기를 포함하되, 클라이언트 객체 요청 처리기는 프록시 서버나 네트워크로부터 요청된 객체를 요정하기 이전에 요청된 객체를 위한 주된 캐쉬(main cache)나 사전 인출 캐쉬를 검색하고, 클라이언트 객체 요청 처리기는 객체가 사전 인출 캐쉬 내에 있는 경우, 사전 인출 객체가 상기 클라이언트에 의해 실제로 액세스되었음을 나타내는 사전 인출 조회 메시지(a prefetch referenced message)를 서버로 발생시킴으로써 컨텐트 서버나 프록시 서버로 통지한다. 마찬가지로, 프록시 서버 중 적어도 하

나는 요청된 객체가 사전 인출 캐쉬 내에 위치하는 경우 캐쉬 관리자 루틴(a cache manager routine)을 호출한 후 사전 인출 조회 메시지를 전달하는 사전 인출 조회 메시지 처리기를 포함하는 것이 바람직하다. 이는 서버가 자녀 객체에 직접 접속되지 않더라도 클라이언트나 프록시 서버 내에 있는 사전 인출 캐쉬로부터 정확한 액세스 기록을 이용할 수 있기 때문에, 서버가 정확한 액세스 기록을 유지할 수 있도록 보장한다.

본 발명은 이러한 데이터 처리 시스템에서 사용되는 컨텐트 또는 프록시 서버를 제공하는데, 요청된 정보는 요청된 객체를 포함하고, 선택 정보는 사전 인출 후보 객체를 포함하며, 상기 서버는 부모 객체 식별자와 연관된 자녀 객체 식별자를 저장하는 통계 테이블과, 통계 테이블 내의 정보에 기초하여 사전 인출 힌트 정보 리스트를 발생시키는 발생 수단을 포함하되, 통계 테이블은 부모 객체의 제 1 카운트를 포함하고, 제 1 카운트는 부모 객체가 클라이언트에 의해 조회된 빈도를 나타내며, 통계 테이블은 자녀 객체의 제 2 카운트를 포함하고, 제 2 카운트는 대응 자녀 객체가 부모 객체 중 선택된 하나가 조회된 후 조회되는 빈도를 나타낸다.

일 실시예에서, 발생 수단은 통계 테이블로부터 제 1 카운트, 제 2 카운트, 자녀 객체 식별자, 부모 객체 식별자의 입력을 수용하고, 발생 수단은 부모 객체와 그들의 연관된 자녀 객체 사이의 액세스 관계를 평가하는 평가 수단을 포함하되, 평가 수단은 대응 부모 객체에 대한 클라이언트에 의한 조회 이전에, 자녀 객체가 바로 다음에 또는 후속적으로 적절한 빈도로 조회되는 경우, 사전 인출 힌트 정보 리스트 상에 자녀 객체 식별자를 배치한다. 적절한 빈도는 대응 부모 액세스에 대한 자녀 액세스의 비율에 기초할 수 있다. 전형적으로, 이는 상기 부모 객체 식별자 각각에 대해 제 1 카운트에 대한 제 2 카운트의 비율을 계산하고, 비율을 힌트 문턱값과 비교하며, 비율이 힌트 문턱값을 초과하는 경우 사전 인출 힌트 정보 상의 자녀 객체와 연관된 부모 객체를 그룹화하거나 연합하는 것과 관련된다.

바람직하게, 서버는 실제로 요청되는 자녀 객체에 기초하여 통계 테이블을 갱신하는 통계 갱신 수단을 더 포함하되, 통계 갱신 수단은 부모 객체 중 조회하는 하나(a referring one)의 신원을 먼저 검색하되, 조회하는 하나, 즉 조회자 객체(referrer object)는 자녀 객체가 실제로 요청되기 이전에 조회되며, 자녀 객체는 자신의 대응 제 2 카운트를 상기 통계 테이블 내에서 갱신시키면서 실제로 요청된다. 통계 테이블은 사전 인출 조회 메시지 처리기로부터의 사전 인출 조회 메시지 수신 시 갱신되어야 한다.

서버는 프록시 서버이고, 사전 인출 힌트 정보 리스트를 수신하고 갱신하는 사전 인출 갱신 수단을 더 포함하는 것이 바람직하다. 사전 인출 갱신 수단은 제 1 지역 문턱 카운트와 제 1 카운트를 비교하고, 사전 인출 갱신 수단은 제 1 카운트가 제 1 지역 문턱 카운트를 초과하는 자녀 객체 식별자 각각에 대해 발생 수단을 호출하며, 사전 인출 갱신 수단은 제 1 카운트가 제 1 지역 문턱 카운트를 초과하지 않는 경우, 원래 사전 인출 힌트 정보 리스트를 유지한다. 보다 정교한 접근 방법에서, 사전 인출 갱신 수단은 제 1 카운트를 제 1 및 제 2 지역 문턱 카운트와 비교하여, (1) 제 1 카운트가 제 1 지역 문턱 카운트를 초과하지 않으면 원래 사전 인출 힌트 정보 리스트를 유지하고, (2) 제 1 카운트가 제 1 지역 문턱 카운트를 초과하면서 제 2 지역 문턱 카운트를 초과하지 않으면 원래 사전 인출 힌트 정보 리스트를 수정하며, (3) 제 1 카운트가 제 1 및 제 2 지역 문턱 카운트를 초과하면 원래 사전 인출 힌트 정보 리스트를 교체한다.

바람직하게, 서버는 요청된 객체를 요청한 하위 계층의 프록시 서버나 클라이언트에게로 요청된 객체를 돌려보내는 프록시 객체 처리기를 더 포함하되, 프록시 객체 처리기는 요청된 객체가 사전 인출 객체가 아닌 경우 요청된 객체를 돌려보내고 따라서 통계 테이블을 갱신한다. 따라서, 프록시 객체 처리기는 사전 인출 캐쉬로부터 요청된 객체를 요청 클라이언트나 프록시 서버로 돌려보내고, 객체가 사전 인출 요청에 따라 요청된 경우 상기 사전 인출 캐쉬로부터 삭제가능한 것으로 객체를 표시한다. 프록시 객체 처리기는 요청된 객체가 주된 캐쉬나 사전 인출 캐쉬 내에 있지 않으면 객체

요청을 다른 서버로 보낸다. 프록시 객체 처리기는 요청된 객체가 사전 인출 요청과 연관되지 않고 요청된 객체가 사전 인출 캐쉬나 주된 캐쉬 내에 위치하면 통계 테이블을 갱신한다.

다른 측면에서 보면, 본 발명은 컨텐트 서버로부터 선택 객체를 사전 인출하여 통신 네트워크를 통해 하나 이상의 프록시 서버와 연관된 클라이언트를 위해 선택 객체를 용이하게 이용할 수 있도록 만드는 방법을 제공하되, 이 방법은 프록시 서버 중 적어도 하나로부터 요청 정보를 요청하는 단계와, 클라이언트에 의한 이전 인출 또는 요청된 정보로의 액세스에 기초하여 대응 클라이언트의 데이터 조회 패턴을 식별하는 단계와, 데이터 조회 패턴으로부터 사전 인출 힌트 정보 리스트로 사전 인출 힌트 정보를 발전시키는 단계와, 사전 인출 힌트 정보를 이용하여 요청된 객체에 주석을 다는 단계와, 사전 인출 힌트 정보와 현재 요청된 객체에 기초하여 상기 서버 중 적어도 하나로부터 선택 객체를 사전 인출하는 단계를 포함한다.

따라서, 사전 인출된 선택 정보는 사전 인출 후보 객체를 포함하고, 사전 인출 힌트 정보를 설정하는 단계는 상기 프록시 서버의 계층 내의 송신 노드에서 사전 인출 힌트 정보를 설정하는 단계를 더 포함하되, 사전 인출 후보 객체는 상기 프록시 서버의 계층을 통과하며, 선택 정보는 사전 인출 힌트 정보에 기초하여 계층 내의 수신 노드에서 사전 인출된다.

전형적으로 네트워크와 연관된 프록시 서버 계층이 존재하는데, 프록시 서버 계층은 다른 프록시 서버보다 클라이언트와 보다 긴밀하게 연관된 하위 계층을 포함한다. 바람직한 일 실시예에서, 사전 인출 힌트 정보는 요청된 정보에 대한 지역 조회가 존재하고 지역 참조 서버의 카운트가 최소 문턱값을 초과하면 프록시 서버 계층을 통해 갱신되고, 이와는 달리(또는 추가적으로) 사전 인출 힌트 정보는 상위 계층에 있는 캐싱 상태를 분석함으로써 갱신될 수도 있는데, 계층의 최상위 계층은 네트워크와 가장 긴밀히 연관되어 있다.

다른 측면에서 볼 때, 본 발명은 네트워크를 따라 서버로부터 클라이언트로의 데이터 흐름을 개선하는 처리 시스템을 제공하는데, 이 처리 시스템은 네트워크와 연관되는 프록시 서버와, 네트워크와 연관되는 컨텐트 서버와, 상기 프록시 서버 중 하나 이상과 통신하도록 구성된 클라이언트와, 상기 프록시 서버 중 적어도 하나에 저장되며, 부모 객체 식별자와 연관된 자녀 객체 식별자, 부모 객체의 제 1 카운트, 자녀 객체의 제 2 카운트를 저장하되, 제 1 카운트는 부모 객체가 클라이언트에 의해 조회되는 회수를 나타내고, 제 2 카운트는 대응 자녀 객체가 상기 부모 객체 중 하나가 조회된 후 조회되는 회수를 나타내는 통계 테이블과, 제 1 카운트와 대응 제2 카운트의 입력을 수용하여 사전 인출 힌트 정보 리스트를 발생시켜, 사전 인출 정보 리스트를 유도하는 발생 수단을 포함하되, 상기 발생 수단은 상기 부모 객체 식별자 각각에 대해 제 1 카운트에 대한 제 2 카운트의 비율을 계산하는 계산 수단과, 힌트 문턱값과 비율을 비교하고, 비율이 힌트 문턱값을 초과하면 사전 인출 힌트 정보 상의 연관된 부모 객체와 자녀 객체를 그룹화하거나 연합시키는 수단과, 사전 인출 힌트 정보와 클라이언트에 의해 실제로 요청된 부모 객체에 기초하여 자녀 객체를 사전 인출하는 사전 인출 수단을 포함한다.

바람직한 실시예에서, 상기 프록시 서버 각각은 제 1 프로세서, 제 1 메모리, 데이터를 저장하는 제 1 데이터 저장 수단, 제 1 주된 캐쉬와 제 1 사전 인출 캐쉬 사이에 적어도 논리적인 분리를 갖는 제 1 메모리, 데이터를 저장하는 제 2 저장 수단을 구비하고, 적어도 하나의 클라이언트는 제 3 프로세서, 제 3 메모리, 데이터를 저장하는 제 3 저장 수단을 구비하되, 제 3 메모리는 제 2 주된 캐쉬와 제 2 사전 인출 캐쉬 사이에 적어도 논리적인 분리를 갖고, 사전 인출 처리 수단은 제 1 사전 인출 캐쉬나 제 2 사전 인출 캐쉬 내에 자녀 객체를 저장한다.

바람직한 실시예에서, 사전 인출 수단은 사전 인출 값을 계산하는 계산 수단과, 계산 수단에 의해 계산된 사전 인출 값을 평가하는 평가 수단을 포함하고, 상기 계산 수단은 사전 인출 힌트 정보 리스트 상에 있는 각각의 자녀 객체 식별자에 대해 사전 인출 값을 계산하고 최고 사전 인출 값을 갖는 자녀 객체의 캐싱을 사전 인출 힌트 정보 리스트 상에 우선순위를 매기며, 사전 인출 처리 수단은 사전 인출 캐쉬 내의 이용가능한 공간을 관리한다.

바람직하게, 시스템은 사전 인출 힌트 정보 리스트를 갱신하는 사전 인출 갱신 수단을 더 포함하는데, 사전 인출 갱신 수단은 제 1 카운트를 제 1 지역 문턱 카운트 및 제 2 지역 문턱 카운트와 비교하고, 제 1 지역 문턱 카운트는 한 서버로부터의 통계에 기초하며, 제 2 지역 문턱 카운트는 다른 서버로부터의 통계에 기초하고, 상기 사전 인출 갱신 수단은 제 1 카운트가 제 1 지역 문턱 카운트와 제 2 지역 문턱 카운트를 초과하는 경우의 자녀 객체 식별자 각각에 대해 발생 수단을 호출하며, 사전 인출 갱신 수단은 제 1 카운트가 제 1 지역 문턱 카운트를 초과하지 않으면 원래 사전 인출 힌트 정보 리스트를 유지한다.

이러한 시스템은 통계 테이블을 갱신하는 통계 갱신 수단과, 상기 프록시 서버 중 하나에 저장되며, 사전 인출 힌트 정보 리스트 갱신 루틴을 호출하고, 하위 계층 프록시 서버가 요청된 객체를 요청하면 하위 계층 프록시 서버로 요청된 객체를 반환하며, 요청된 객체가 사전 인출 객체가 아니면 요청된 객체를 반환하고 통계 갱신 수단을 호출하는 프록시 객체 처리기와, 상기 프록시 서버 중 하나와 연관되는 사전 인출 캐쉬를 관리하되, 프록시 객체 처리기는 하위 계층 서버가 요청된 객체를 요청하지 않을 때 호출하는 사전 인출 캐쉬 관리자와, 상기 프록시 서버 중 하나에 저장되는 프록시 객체 요청 처리기를 포함하는 것이 바람직하되, 프록시 객체 요청 처리기는 객체가 사전 인출 캐쉬 내에 존재하고 객체가 사전 인출 요청에 따라 요청되면 객체를 상기 클라이언트 중 요청하는 하나로 반환하고 그 객체에 사전 인출 캐쉬로부터 삭제가능하다는 표시를 하며, 프록시 객체 요청 처리기는 요청된 객체가 주된 캐쉬 또는 사전 인출 캐쉬 내에 있지 않으면 객체 요청을 다른 서버로 전달하며, 프록시 객체 요청 처리기는 요청된 객체가 사전 인출 요청과 연관되지 않고 요청된 객체가 사전 인출 캐쉬나 주된 캐쉬 내에 위치하면 통계 갱신 수단과 통신한다.

또한, 이 시스템은 통계 테이블을 갱신하는 통계 갱신 수단과, 상기 프록시 서버 중 하나에 저장되는 사전 인출 조회 메시지 처리기를 포함하는 것이 바람직하되, 사전 인출 조회 메시지 처리기는 통계 갱신 수단을 호출하고, 요청된 객체가 사전 인출 캐쉬 내에 존재하면 캐쉬 관리자 루틴을 호출한 후 사전 인출 조회 메시지를 전달한다.

또한, 본 발명은 네트워크를 따라 서버로부터 클라이언트로의 데이터 흐름을 개선하는 처리 시스템을 제공하는데, 이 처리 시스템은 네트워크와 연관되는 프록시 서버와, 네트워크와 연관되는 컨텐트 서버와, 상기 프록시 서버 중 하나 이상과 통신하도록 구성된 클라이언트와, 상기 컨텐트 서버와 프록시 서버 중 적어도 하나에 저장되며, 부모 객체 식별자와 연관된 자녀 객체 식별자, 부모 객체의 제 1 카운트, 자녀 객체의 제 2 카운트를 저장하되, 제 1 카운트는 부모 객체가 클라이언트에 의해 조회되는 회수를 나타내고, 제 2 카운트는 대응 자녀 객체가 상기 부모 객체 중 선택된 하나가 조회된 후 바로 다음에 일시적으로 조회되는 회수를 나타내는 통계 테이블과, 통계 테이블로부터 제 1 카운트, 제 2 카운트, 자녀 객체 식별자, 부모 객체 식별자를 입력으로 수용하며, 부모 객체와 그들의 연관된 자녀 객에 사이의 관계를 평가하는 평가 수단을 포함하되, 평가 수단은 자녀 객체가 대응 부모 객체에 대한 클라이언트의 조회 이전에 바로 다음에 후속하여 적절한 빈도로 조회된 경우 자녀 객체 식별자를 사전 인출 힌트 정보 리스트 상에 배치하는 발생 수단과, 사전 인출 힌트 정보와 클라이언트에 의해 실제로 요청된 부모 객체에 기초하여 자녀 객체를 사전 인출하는 사전 인출 수단을 포함한다.

바람직한 실시예에서, 사전 인출 수단은 사전 인출 힌트 정보 리스트 상에서 식별된 자녀 객체에 저장 우선 순위를 할당하는 할당 수단과, 저장 우선 순위에 따라 자녀 객체 중 선택된 하나를 저장하는 선택적 저장 수단을 더 포함한다. 할당 수단은 사전 인출 힌트 정보 리스트 상의 자녀 객체가 클라이언트 중 특정한 하나에 의해 액세스될 추정 액세스 확률 및 자녀 객체 중 선택된 하나를 저장하는 데 필요한 저장 크기에 기초한 자녀 객체의 액세스 비용에 기초하여 저장 우선 순

위를 할당한다.

따라서, 정보는 프록시 계층 내에 사전 인출되어 클라이언트의 네트워크(즉, 인터넷)를 통한 액세스 시간을 감소시킨다. 선택 정보는 컨텐트 서버로부터 사전 인출되어 선택 정보를 네트워크 내의 컨텐트 서버와 연관된 클라이언트가 용이하게 액세스할 수 있도록 만든다. 클라이언트는 네트워크 내의 프록시 서버인 것이 바람직하다. 클라이언트는 프록시 서버 중 적어도 하나로부터 정보를 요청한다. 컨텐트 서버는 클라이언트에 의해 요청된 정보의 과거 데이터 액세스 패턴에 기초하여 클라이언트를 위한 사전 인출 힌트 정보를 발생시킨다. 컨텐트 서버는 사전 인출 힌트 정보를 이용하여 요청된 정보에 주석을 단다. 하나 이상의 프록시 서버는 동적 사용 통계에 기초하여 사전 인출 힌트 정보를 동적으로 갱신할 수 있다. 사전 인출 힌트 정보는 객체가 프록시 계층을 통과할 때 갱신되어 수행된 사전 인출 동작, 프록시 계층의 상위 계층에서의 캐싱 상태, 지역 조회 패턴과 같은 기타 지역 고려 사항을 반영한다. 선택 정보는 사전 인출 힌트 정보와 요청된 정보에 기초하여 컨텐트 서버나 프록시 서버 중 임의의 하나로부터 사전 인출될 수 있다. 사전 인출 힌트 정보는 조회 액세스 확률과 저장 고려 사항에 기초하여 사전 인출 값을 할당받을 수 있다. 사전 인출 힌트 정보 리스트 내에서 최고 사전 인출 값을 갖는 선택 정보는 사전 인출 캐쉬가 가득 찰 때까지 가장 먼저 그리고 잠재적으로 캐싱된다.

도면의 간단한 설명

도 1은 컨텐트 서버(content server)(즉, 웹 사이트(Web sites), 네트워크(즉, 인터넷), 프록시 서버 계층, 클라이언트를 나타내는 블럭도,

도 2는 도 1보다 상세한 컨텐트 서버, 클라이언트, 프록시 서버의 블럭도,

도 3a는 프록시 서버 계층을 통해 컨텐트 서버로부터 데이터를 동적으로 사전 인출하는 방법을 나타내는 흐름도,

도 3b는 통계 테이블의 예를 도시하는 도면,

도 4는 컨텐트 서버의 동작을 나타내는 흐름도,

도 5는 통계 갱신을 도시하는 흐름도,

도 6은 사전 인출 힌트 정보의 발생을 나타내는 흐름도,

도 7은 클라이언트 동작을 나타내는 흐름도,

도 8은 사전 인출을 나타내는 흐름도,

도 9는 클라이언트 객체 처리기의 동작을 나타내는 흐름도,

도 10은 프록시 서버의 동작을 나타내는 흐름도,

도 11은 프록시 객체 요청 처리기의 동작을 나타내는 흐름도,

도 12는 사전 인출 조회 메시지 처리기(a prefetch referenced message handler)의 동작을 나타내는 흐름도.

도 13은 프록시 객체 처리기의 동작을 나타내는 흐름도,

도 14는 사전 인출 힌트 정보 갱신 루틴의 동작을 나타내는 흐름도.

바람직한 실시예의 설명

이하 첨부한 도면을 참조하여 예로서 본 발명의 바람직한 실시예를 상세히 설명할 것이다.

도 1은 클라이언트(22), 프록시 서버 계층(24), 네트워크(25), 컨텐트 서버(20)를 포함하는 데이터 프로세싱 시스템을 도시하고 있다.

클라이언트는 부모 객체, 자녀 객체, 요청된 객체, 선택 객체, 페이지, 요청된 정보, 선택 정보의 형태로 정보를 요청할 수 있다. 부모 객체가 하나 이상의 연관된 자녀 객체에 대한 후속 요청을 계속하도록 하는 방식으로 부모 객체는 자녀 객체와 연관된다. 요청된 객체는 부모 객체와 유사하지만, 선택 객체는 자녀 객체와 유사하다. 요청된 객체는 하나 이상의 클라이언트에 의해 실제로 요청된 객체이다. 한편, 선택 객체는 선택 객체가 요청된 객체 이후에 바로 다음 일시적으로 요청될 수 있도록 요청된 객체와 그룹화되거나 연관되는 객체이다. 요청된 정보는 요청된 객체, 부모 객체, 기타 클라이언트에 의해 요청된 데이터 형태를 포함한다. 선택 정보는 자녀 객체, 선택 객체, 기타 사전 인출될 수 있는 데이터 형태를 포함한다.

클라이언트(22)는 컨텐트 서버(20) 또는 프록시 서버(24)로부터 정보를 요청할 수 있다. 그러나, 프록시 서버(24)는 보통 네트워크(25)와 컨텐트 서버(20)에 대해 클라이언트(22)를 위한 매개 역할을 한다. 유리하게도, 프록시 서버(24)는 프록시 서버 계층이 요청된 정보를 포함하거나, 요청된 작업을 완료할 능력이 있는 경우에는 컨텐트 서버(20)에 접속하지 않고 클라이언트 정보 요청을 완수할 수 있다. 도 1의 데이터 처리 시스템은 인터넷에 적용될 수 있다. 예를 들어, 컨텐트 서버(20)는 웹 사이트를 포함할 수 있고, 네트워크(25)는 인터넷을 포함할 수 있다.

도 1은 데이터 처리 시스템의 구성 요소 사이에 있는 물리적 접속과 대조되는 논리적 접속(26)을 도시하고 있다. 논리적 접속(26)은 데이터 처리 시스템 내의 객체와 요청의 정보 흐름을 나타낸다. 논리적 접속(26)은 전형적으로 프록시 서버(24) 사이의 물리적 접속을 나타내지는 않는다. 논리적 접속(26)은 부하(workload) 및/또는 노드 또는 통신 링크 장애와 같은 물리적 사건으로 인해 변경될 수 있다. 서로 다른 유형의 객체는 서로 다른 전송의 논리적 경로를 따라 클라이언트(22)에게 도달한다.

사실상, 임의 개수의 클라이언트(22)와 프록시 서버(24)를 실제로 사용할 수 있지만, 도 1의 실시예는 4 개의 클라이언트(22)와 4 계의 프록시 서버(24) 계층을 나타낸다. 실제로, 프록시 서버 계층 내에는 어떠한 수의 계층도 허용되며, 클라이언트(22)는 프록시 서버 계층의 어떠한 계층에도 논리적으로 접속될 수 있다. 예를 들어, 프록시 서버(24)의 계층은 단일 프록시 서버일 수도 있다. 본 발명은 단일 클라이언트(22)와 단일 프록시 서버(24)만을 이용하여 실시될 수 있다.

도시한 바와 같이, 최고 계층 프록시 서버는 계층-0 프록시 서버(30)로 지정된다. "계층-0"은 도 1의 실시예에서 다른 프록시 서버에 대한 한 프록시 서버의 상태를 정의하는 임의의 지정일 뿐이다. 최고 계층 프록시 서버는 네트워크(25)에 논리적으로 접속될 수 있다. 상위 계층 프록시 서버(21)는 하위 계층 프록시 서버(23)보다 논리적으로 긴밀하게 네트워크(25)와 연관된다. (따라서, 이 용어 중에서 계층-0 프록시 서버는 최상위 프록시 계층이고, 계층-0은 계층-1보다 "상위"이며, 계층-1은 계층-2보다 "상위"인 것을 나타냄을 이해할 것이다.)

제 1 계층-1 프록시 서버(35)와 제 2 계층-1 프록시 서버(37)는 계층-0 프록시 서버(30)에 논리적으로 접속되어 있다. 제 1 계층-1 프록시 서버(35)는 제 1 계층-2 프록시 서버(40)와 제 2 계층-2 프록시 서버(43)에 접속되어 있다. 제 1 계층-2 프록시 서버(40)는 제 1 계층-3 프록시 서버(50)와 제 2 계층-3 프록시 서버(55)에 논리적으로 접속되어 있다.

클라이언트(22)는 제 1 클라이언트(600), 제 2 클라이언트(603), 제 3 클라이언트(700), 제 4 클라이언트(703)를 포함한다. 제 1 클라이언트(600)와 제 2 클라이언트(603)는 제 1 계층-3 프록시 서버(50)에 논리적으로 접속되어 있다. 제 3 클라이언트(700)와 제 4 클라이언트(703)는 제 2 계층-3 프록시 서버(55)에 논리적으로 접속되어 있다.

이제 데이터 처리 시스템은 제 2 클라이언트(603)의 관점에서 설명된다. 제 2 클라이언트(603)는 자신의 바로 상위 계층 프록시에 접속되어 있다. 설명한 바와 같이, 바로 상위 계층 프록시는 제 1 레벨-3 프록시(50)이다. 제 2 클라이언트(603)는 제 1 계층-2 프록시 서버(40), 제 1 계층-1 프록시 서버(35), 계층-0 프록시 서버(30)에 각각 접속된다. 제 2 클라이언트(603)는 네트워크(25)를 통해 다양한 컨텐트 서버(20)를 액세스할 수 있다. 제 2 클라이언트(603)를 위한 전형적인 통신 경로는 바로 상위 계층 프록시(즉, 제 1 계층-3 프록시 서버(50))를 통해서이지만, 제 2 클라이언트(603)나 하위 계층 프록시가 다른 상위 계층 프록시 서버(24)나 네트워크(25)와 직접 통신할 수도 있다. 제 2 클라이언트(603)는 캐싱을 관리하기 위해 자신의 클라이언트를 가질 수도 있다.

제 2 클라이언트(603)의 관점에서, 소정의 프록시 서버는 자신의 프록시 계층의 일부가 아니다. 예를 들어, 제 2 계층-3 프록시 서버(55), 제 2 계층-2 프록시 서버(43), 제 2 계층-1 프록시 서버(37)는 제 2 클라이언트의 프록시 서버 계층의 일부가 아니다.

클라이언트(22)는 개인용 컴퓨터, 워크 스테이션, 셋 탑 박스(set top box) 등을 포함할 수 있지만, 여기에 국한되지는 않는다. 네트워크(25)는 인터넷, 월드 와이드 웹, 인트라넷, LAN(local area networks) 등을 포함할 수 있지만, 여기에 국한되지는 않는다.

프록시 서버(24)와 컨텐트 서버(20)는 상용화된 다양한 컴퓨터를 포함한다. 예를 들어, 프록시 서버(24)는 IBM 사에서 판매하는 인터넷 접속 서버(Internet Connection Server : ICS)로서 구현될 수도 있다. 마찬가지로, 컨텐트 서버(20)는 (역시 IBM 사로부터 입수가능한) 로터스 고 웹 서버(Lotus Go Web server), 로터스 도미노 서버(Lotus Domino server) 등으로서 구현될 수도 있다. 프록시 서버(24) 또는 컨텐트 서버(20)는 임의의 컴퓨팅 노드(computing node) 상에서 실행될 수 있는데, 이는 (역시 IBM 사로부터 입수가능한) S/390 SYSPLEX, SP2, RS6000 워크 스테이션과 같은 제품을 포함할 수 있으나, 여기에 국한되지는 않는다. 요컨대, 프록시 서버(24) 또는 컨텐트 서버(20)는 월드 와이드 웹 페이지 액세스, 원격 파일 전송, 전자 우편, 클라이언트(22)에 대한 트랜잭션 지원을 제공할 수 있는 임의의 범용 컴퓨터 및 관련 소프트웨어를 포함할 수 있다.

도 1의 데이터 처리 시스템은 통상적인 프록시 계층의 속성을 이용하여 데이터 전송 효율을 유지할 수 있다. 예컨대, 요청된 객체가 하위계층 프록시 서버 상에서 지역적으로 이용가능하지 않은 경우, 다음 상위 계층 프록시로 이용불가능한 객체에 대한 요청이 만들어진다. 상위 계층 프록시가 이전에 자신의 캐쉬 메모리 내에 그 객체를 캐싱했으면, 상위 계층 프록시는 하위 계층 프록시 서버를 통해 클라이언트에게 요청된 객체를 전달할 것이다. 그렇지 않으면, 상위 계층 프록시 서버는 자신의 다음 상위 계층 프록시로부터 그 객체를 얻으려고 시도할 것이다. 요청된 객체를 얻으면, 객체는 하위 계층 프록시나 그 객체를 요청한 클라이언트에게 전달된다.

도 2는 도 1에 도시한 데이터 처리 시스템의 보다 상세한 예를 도시하고 있다. 데이터 처리 시스템은 프록시 서버(24), 컨텐트 서버(20), 네트워크(25), 클라이언트(22)를 포함한다. 데이터 처리 시스템은 사전 인출 유닛(250), 통계 테이블(261), 통계 갱신 유닛(252), 주된 캐쉬(297), 사전 인출 캐쉬(298)를 포함하는 것이 바람직하다.

사전 인출 유닛(250)은 프록시 서버(24) 내에 제 1 사전 인출 처리기(248)를, 클라이언트(22) 내에 제 2 사전 인출 처리기를 포함하는 것이 바람직하다. 통계 테이블(261)은 프록시 서버(24)와 연관된 제 1 통계 테이블(293)과 컨텐트 서버와 연관된 제 2 통계 테이블(267)을 포함한다. 통계 갱신 유닛(252)은 제 1 통계 테이블(293)을 갱신하는 제 1 갱신 통계 유닛(264)과 제 2 통계 테이블(267)을 갱신하는 제 2 갱신 통계 유닛(269)을 포함한다. 주된 캐쉬(297)는 각 프록시 서버(24) 내에 제 1 주된 캐쉬(294)를, 각 클라이언트(22) 내에 제 2 주된 캐쉬(246)를 포함한다. 사전 인출 캐쉬(298)는 프록시 서버(24) 내에 제 1 사전 인출 캐쉬(296)를, 클라이언트(22) 내에 제 2 사전 인출 캐쉬(249)를 포함한다.

프록시 서버(24)는 네트워크(25)를 통해 요청을 서빙(serve)할 수 있는 컴퓨팅 노드이다. 프록시 서버(24)는 제 1 프로세서(280), 제 1 메모리(290), 제 1 저장 유닛(281)을 포함한다. 제 1 메모리(290)는, 예를 들어, RAM(random access memory)를 포함할 수 있다. 제 1 저장 유닛(281)은, 이를테면, 직접 액세스 저장 디바이스(direct access storage device : DASD)를 포함할 수 있다. 제 1 메모리(290)는 프록시 서버 로직(295)을 저장하는데, 이는 컴퓨터 실행가능 코드로서 구현된다. 프록시 서버 로직(295)은 제 1 프로세서(280)에 의한 실행을 위해 제 1 저장 유닛(281)으로부터 제 1 메모리(290)로 로딩된다. 프록시 서버 로직(295)의 상세한 동작은 도 10에서 도시하고 있다.

프록시 서버 로직(295)은 프록시 객체 요청 처리기(283), 프록시 객체 처리기(284), 제 1 사전 인출 처리기(248), 사전 인출 조회 메시지 처리기(285), 사전 인출 갱신 수단(291), 제 1 갱신 통계 루틴(264), 제 1 통계 테이블(293), 제 1 주된 캐쉬(294), 제 1 사전 인출 캐쉬(296)를 포함한다.

프록시 객체 요청 처리기(283)의 상세한 동작은 도 11에서 도시하고 있다. 프록시 객체 처리기(284)의 상세한 동작은 도 13에서 도시하고 있다. 사전 인출 조회 메시지 처리기(285)의 동작은 도 12에서 도시하고 있다. 통계 갱신 유닛(252)(즉, 제 1 갱신 통계 루틴(264)의 상세한 동작은 도 5에서 도시하고 있다.

사전 인출 유닛(250)(즉, 제 1 사전 인출 처리기(248))의 동작은 도 8에서 상세하게 도시하고 있다. 사전 인출 갱신 유닛(291) 또는 사전 인출 힌트 정보 갱신 루틴의 동작은 도 14에서 도시하고 있다. 프록시 서버 로직(295)은 제 1 주된 캐쉬(294)와 제 1 사전 인출 캐쉬(296)를 유지한다.

컨텐트 서버(20)는 네트워크(25)를 통해 요청을 서빙하는 컴퓨팅 노드이다. 컨텐트 서버(20)는 제 2 프로세서(260), 제 2 메모리(263), 제 2 저장 유닛(265)을 포함한다. 제 2 메모리(263)는, 예를 들어, RAM(random access memory)를 포함할 수 있다. 제 2 저장 유닛(265)은, 이를테면, 직접 액세스 저장 디바이스(direct access storage device : DASD)를 포함할 수 있다. 제 2 메모리(263)는 컨텐트 서버 로직(268) 또는 컨텐트 서버 소프트웨어를 저장한다.

컨텐트 서버(268)의 동작은 도 4를 조회하여 이후에 보다 상세히 설명할 것이다. 컨텐트 서버 로직(268)은 제 2 프로세서(260)에 의한 실행을 위해 제 2 저장 유닛(265)으로부터 제 2 메모리(263)로 로딩되는 컴퓨터 실행가능 코드로서 구현된다. 컨텐트 서버 로직(268)은 제 2 통계 테이블(267), 제 2 통계 테이블을 갱신하는 제 2 갱신 통계 유닛, 사전 인출 힌트 정보를 발생시키는 발생 유닛(266)을 포함한다. 제 2 통계 테이블(267)은 클라이언트에 의해 요청된 정보와 요청된 정보와 연관된 잠재적인 사전 인출 정보에 관한 데이터를 저장한다. 통계 테이블(261)은 클라이언트의 실제 조회 습성에 기초하는 사전 인출 힌트 정보의 발생에 기여한다. 통계 테이블(261)에 관한 세부 사항은 도 3B에 도시하고 있다.

제 2 갱신 통계 유닛(269)의 동작은 도 5를 조회하여 보다 상세하게 설명한다. 발생 유닛(266)은 제 2 통계 테이블(267) 내의 정보와 요청된 정보로부터 사전 인출 힌트 정보를 발생시킨다. 발생 유닛(266)은 도 6에 도시한 바와 같은 사전 인출 힌트 정보(prefetch hint information : PHI) 발생 루틴으로서 구현된다.

클라이언트(22)는 제 3 프로세서(240), 제 3 메모리(245), 제 3 저장 유닛(242)을 포함한다. 제 3 메모리(245)는, 예를 들어, RAM을 포함할 수 있다. 제 3 저장 유닛(242)은, 예를 들어, DASD를 포함할 수 있다. 제 3 메모리(245)는 클라이언트 로직(244)이나 클라이언트 소프트웨어를 저장한다. 클라이언트 로직(244)의 동작은 도 7을 참조하여 보다 상세히 설명할 것이다. 클라이언트 로직(244)은 제 3 프로세서(24)에 의한 실행을 위해 제 3 저장 유닛(242)으로부터 제 3 메모리(245)로 로딩되는 컴퓨터 실행가능 코드로서 구현된다.

클라이언트 로직(244)은 클라이언트 요청 처리기(247), 제 2 사전 인출 처리기(241), 제 2 주된 캐쉬(246), 제 2 사전 인출 캐쉬(249)를 포함한다. 클라이언트 객체 요청 처리기(247)는 제 2 주된 캐쉬(246), 제 2 사전 인출 캐쉬(249), 프록시 서버(24), 또는 컨텐트 서버(20)로부터 클라이언트를 위한 요청된 객체를 얻는다. 클라이언트 객체 요청 처리기(247)의 상세한 동작은 도 9에서 도시하고 있다.

클라이언트 로직(244)은 제 2 주된 캐쉬(246)와 제 2 사전 인출 캐쉬(249)를 유지한다. 제 2 주된 캐쉬(246)는 자주 조회되는 객체를 저장하는 데 사용되는 반면, 제 2 사전 인출 캐쉬(249)는 이들이 실제로 조회되기 이전에 객체를 저장하는 데 사용된다. 사전 인출 객체는 조회된 객체와 관련하여 최근에 또는 이전에 조회되지 않은 객체이다. 바람직한 실시예에서, 주된 캐쉬(297)는 통상적인 LRU(least recently used) 알고리즘에 의해 관리된다. 당업자는 임의의 다른 대체적인 방안도 주된 캐쉬(297)를 유지하는 데 적용될 수 있다는 사실을 이해할 것이다.

주된 캐쉬(297)와 사전 인출 캐쉬(298)는 논리적으로 분리될 수 있다. 사전 인출 객체를 주된 캐쉬(297) 내의 실제로 조회된 객체로부터 논리적으로 분리하면, 사전 인출 객체가 너무 많은 저장 공간을 차지하는 것을 방지하게 된다. 이러한 사전 인출 캐쉬와 주된 캐쉬의 논리적인 분리를 달성하는 데에는 다양한 대체적인 방법이 사용될 수 있다. 예를 들어, 단일 캐쉬는 사전 인출 객체의 개수에 부과된 상한(upper limit) 및/또는 사전 인출 객체의 의해 점유된 공간에 부과된 상한을 가질 수 있다. 주된 캐쉬(297)와 사전 인출 캐쉬(298)는 서로 다른 목표를 지원하려 하고 있다. 주된 캐쉬(297)는 반복적으로 조회될 가능성이 있는 인기있는 객체를 캐싱하려고 한다. 사전 인출 캐쉬(298)는 그 세션에서 다음에 조회될 가능성이 가장 큰 객체를 사전 인출함으로써 특정 브라우징 세션을 지원하는 데에만 사용된다. 바람직한 실시예에서, 계층 내의 하위 계층 노드가 상위 계층 노드의 사전 인출 캐쉬 내의 객체에 대한 사전 인출 요청을 발행하면, 그 객체는 하위 계층 노드로 전달되고 사전 인출 목적이 달성되었으므로 그 객체는 상위 계층 노드의 사전 인출 캐쉬(298)로부터 삭제가능하다고 표시될 것이다. 사전 인출 객체가 실제로 조회되면, 사전 인출 객체는 주된 캐쉬(297)의 관리 정책에 따라 주된 캐쉬(297)로 이동할 수 있다. LRU 대체 방안을 이용하는 바람직한 실시예에, 사전 인출 객체는 주된 캐쉬로 이동한다.

대체적인 표준은 사전 인출 캐쉬(298)로부터 주된 캐쉬(297)로 객체를 수용하도록 고안될 수 있다. 에이브람즈(Abrams) 등의 "Caching Proxies : Limitations and Potentials", Proc. 4th International World Wide Web Conference(1996)에서는, 결정 과정에서 객체 크기를 고려하지 않는 소정의 캐쉬 대체 방안이 논의된 바 있다. 기타 캐쉬 대체 방안은 당업자에게 주지되어 있다.

도 3a는 컨텐트 서버(20) 또는 프록시 서버 계층으로부터 클라이언트를 위해 데이터를 동적으로 사전 인출하는 방법을 나타내는 흐름도이다. 첫 번째로, 블럭(100)에서, 클라이언트는 프록시 서버나 컨텐트 서버(20)로부터 정보를 요청한다. 두 번째로, 블럭(102)에서 컨텐트 서버(20)와 프록시 서버에서의 역사적인 데이터 액세스 또는 데이터 조회 패턴이 식별된다. 실제로, 블럭(100)과 블럭(102)을 완료하는 순서는 바뀔 수도 있다.

세 번째로, 블럭(104)에서, 컨텐트 서버(20)는 블럭(102)에서 식별된 데이터 조회 패턴에 기초하여 요청된 정보로의 관련 액세스에 관한 사전 인출 힌트 정보(PHI)를 발생시킨다. 네 번째로, 블럭(105)에서, 사전 인출 힌트 정보는 요청된 정보에 주석을 달 수 있다. 예를 들어, 사전 인출 힌트 정보는 요청된 정보(즉, 객체)와 함께 운반(piggy-backed)

된 메타 데이터(meta data)로서 전달될 수 있다. 사전 인출 힌트 정보를 설정하거나 발생시키는 단계에 대해, 컨텐트 서버(20)는 다수의 동시 사용자 또는 클라이언트로부터의 사용/조회 패턴을 추적한다. 실제로, 컨텐트 서버(20)는 어느 객체 그룹이 요청된 정보 또는 요청된 객체와 관련있는지를 판단한다. 컨텐트 서버(20)는 요청된 정보와 함께 사전 인출 힌트 정보(PHI)를 발생시키고 제공한다. 사전 인출 단계 또는 바로 직전에, 프록시 서버는 지역적 고려 사항에 비추어 PHI를 해석하여 어느 선택 객체를 사전 인출할 지를 결정할 수 있다.

다섯 번째로, 블럭(106)에서, 사전 인출 힌트 정보는 요청된 정보가 프록시 서버 계층을 통과할 때 갱신된다. 블럭(106)에서의 갱신은 수행된 사전 인출 동작, 상위 계층에서의 캐싱 상태, 지역 조회 패턴과 같은 기타 지역 고려 사항을 반영한다. 프록시 서버는 객체를 다음(하위) 계층 프록시나 클라이언트 스테이션으로 전달하기 이전에 PHI를 수정하여 자신의 사전 인출 결정과 다양한 지역 고려 사항을 반영하도록 한다. 객체가 프록시 계층을 통과할 때, 송신 노드는 초기 PHI를 설정하고, 수신 노드는 초기 PHI로부터 최종 PHI를 유도할 것이다. 최종 PHI는 요청된 객체에 기초하여 어떤 선택 객체를 사전 인출할 지를 정의한다. 사전 인출 동작 및 결정은 수신기에서 개시되며 송신기에서 개시되지 않는다.

마지막으로, 블럭(108)에서는, 사전 인출 힌트 정보와 요청된 정보에 기초하여 선택 정보가 사전 인출된다. 프록시 서버나 클라이언트는 수신된 객체 내의 PHI 정보, (이용가능할 수 있다면) 지역적으로 유지된 조회 정보, 이용가능한 저장 크기, 액세스 시간과 같은 기타 지역적 고려 사항에 기초하여 PHI 내에 명시된 객체의 서브셋(subset)을 사전 인출할지 여부를 결정한다.

블럭(104)에서 사전 인출 힌트 정보를 발생시키는 것은 PICS 프로토콜 또는 PICS 양립 가능 포맷(아래 조회)을 이용하는 것이 바람직하다. 블럭(104)에서 발생한 PHI는 캐싱된 객체가 요청될 때 객체의 PHI 정보가 조사되어 적절한 사전 인출 동작을 개시할 수 있도록 자신의 메타 데이터의 일부로서 각 캐싱된 객체와 함께 저장된다. PHI 정보는 PICS 프로토콜을 통해 제공될 수 있다. PICS 프로토콜은 각 PHI 객체의 임계점(criticality)에 관한 정보와 상위 계층에 관한 크기, 캐싱 상태와 같은 자신의 특성을 제공한다.

사전 인출 힌트 정보를 갱신하는 것은 프록시 서버 계층을 통해 달성될 수 있다. 프록시 서버는 지역 정보를 수집하고, 사전 인출 결정을 프록시 계층으로 분배 또는 분산시킬 수 있다. 또한, 중간 프록시가 객체를 사전 인출한 정보는 저장 및 대역폭 사용을 최적화하기 위해 하위 계층 프록시와 클라이언트 스테이션의 사전 인출 결정에 암시를 줄 수 있다.

프록시 서버 계층 내의 프록시 서버는 수신기 지향 사전 인출 결정(a receiver-oriented prefetch decision)을 제공하기 위해 서로 협력한다. 사전 인출 결정은 각 프록시와 클라이언트에 의해 협력하여 수행될 수 있다. 메타 정보 형식의 사전 인출 힌트 정보는 요청된 정보에 주석을 단다(즉, 운반한다). 사전 인출 힌트 정보는 잠재적인 사전 인출 후보에 관한 정보를 제공한다. 중간 프록시 서버는 지역 조건에 기초하여 메타 정보나 사전 인출 힌트 정보를 수정할 수 있고, 자신이 캐싱될 때 객체와 함께 메타 정보를 저장하여 객체에 대한 이후의 조회 시 사전 인출이 트리거(trigger)될 수 있도록 한다.

도 3b는 통계 테이블(261) 내의 데이터 포맷을 도시하고 있다. 통계 테이블(261)은 일반적으로 제 1 통계 테이블(293)과 제 2 통계 테이블(267) 중 하나를 조회한다. 하나 이상의 프록시 서버(24)는 제 1 통계 테이블(293)을 유지한다. 컨텐트 서버(20)는 제 2 통계 테이블(267)을 유지할 것이다. 제 1 통계 테이블(293)과 제 2 통계 테이블(267)을 위한 포맷은 실질적으로 유사하거나 동일하여, 프록시 서버 계층 내의 서로 다른 프록시 서버에서의 통계 테이블(261)의 갱신을 촉진하는 것이 바람직하다. 통계 테이블(261)은 통계 테이블(261)과 연관된 프록시 서버나 컨텐트 서버(20)의 관점에서 실제 관찰자 조회 습성(viewer reference behavior)에 기초하여 사전 인출 힌트 정보(PHI)를 제공하

는 것을 돕는다. 통계 테이블(261)은 프록시 서버 계층 내에서 자신보다 하위인 통신 클라이언트 모두의 요청을 관찰할 수 있다.

예컨대, 도 1에서, 제 1 계층-3 프록시 서버(50) 내의 제 1 통계 테이블(293)은 제 1 클라이언트(600)와 제 2 클라이언트(603)의 통합 데이터 액세스 습성을 추적한다. 한편, 제 2 계층-3 프록시 서버(55) 내의 제 1 통계 테이블(293)은 제 3 클라이언트(700)와 제 4 클라이언트(703)의 통합 데이터 액세스 습성을 추적한다. 제 1 계층-2 프록시 서버(40) 내의 제 1 통계 테이블은 제 1 클라이언트(600), 제 2 클라이언트(603), 제 3 클라이언트(700), 제 4 클라이언트(703)의 통합 습성을 추적한다. 제 1 계층-3 프록시 서버(50)와 같은 하위 계층 프록시가 클라이언트(600)의 관심을 더 잘 추적할 수도 있지만, 하위 계층 프록시는 충분한 샘플 크기의 클라이언트로부터 충분한 조회 통계를 얻어서 의미있는 사전 인출 힌트 정보(PHI)를 제공할 수 없을 수도 있다.

바람직한 실시예에서, 객체 O가 컨텐트 서버(20)로부터 요청되면, 서버(20)는 자신의 제 2 통계 테이블(267) 내의 정보에 기초하여 초기 사전 인출 힌트 정보 PHI를 제공한다. 객체 O가 프록시 계층을 따라 아래로 전달되면서, 클라이언트(22)로 가는 경로 상에 있는 임의의 프록시 서버(24)는 객체 O에 대한 충분한 지역 조회가 있는 경우에는 객체 헤더 내의 사전 인출 힌트 정보(PHI)를 수정하여 제 1 통계 테이블(293) 내의 지역 상황에 PHI를 맞출 수 있다. 제 1 통계 테이블(293)은 하위 계층 프록시 서버에 의한 특정 클라이언트의 선호도 또는 잘 색인된 관심(well-indexed interests)과 같은 인자에 기초하여 제 2 통계 테이블(267)을 대체(supersede)할 수 있다.

바람직한 실시예에서, 컨텐트 서버(20)는 제 2 통계 테이블(267)을 유지하는데, 이는 요청된 정보에 대한 클라이언트(22)의 데이터 액세스 빈도의 데이터베이스이다. 데이터베이스는 요청하는 프록시 서버(24)의 데이터 액세스 빈도를 포함하도록 확장될 수 있다. 마찬가지로, 하나 이상의 프록시 서버(24)가 제 1 통계 테이블(293)을 유지한다. 이를테면, 통계 테이블(261)은 제 1 데이터 구조(304)와 제 2 데이터 구조(306)를 갖는 제 1 어레이(383)를 구비한다. 도시한 바와 같이, 제 1 데이터 구조(304)와 제 2 데이터 구조(306)는 기록된다. 통계 테이블(261) 내의 각 기록은 부모 객체 식별자(301)(즉, ObjectID), 제 1 카운트(302)(즉, TCount), 액세스 리스트 데이터 포인터(303)(즉, Alist)를 포함하는 수 개의 필드로 이루어진다.

부모 객체 식별자 필드(301)는 객체 식별자를 포함한다. 예를 들어, 부모 객체 식별자(301)는 인터넷이나 관련 애플리케이션의 URL을 포함한다. 제 1 카운트 필드(302)는 서버에 의해 조회된 부모 객체 식별자와 연관된 부모 객체의 회수를 포함한다. 즉, 제 1 카운트 필드(302)는 요청된 객체의 실제 요청 빈도를 포함한다. 제 1 카운트(302)는 부모 객체 액세스 빈도를 나타낸다. 액세스 리스트 데이터 포인터(303)는 액세스 리스트(370)에 대한 데이터 포인터를 포함한다.

통계 테이블(261)은 액세스 리스트(370)를 포함하는데, 이는, 예를 들어, 제 1 액세스 기록(340), 제 2 액세스 기록(350), 제 3 액세스 기록(360), 제 4 액세스 기록(380)을 포함한다. 도시한 바와 같이, 제 1 데이터 구조와 연관된 부모 객체 식별자는 제 1 액세스 기록(340), 제 2 액세스 기록(350), 제 3 액세스 기록(360)의 링크 리스트(linked list)로 이루어진 액세스 리스트(370)를 포함한다. 다음 기록 데이터 포인터 필드(313)는 특정 부모 객체 식별자(301)와 연관된 액세스 리스트(370) 상의 다음 액세스 기록에 대한 포인터를 포함한다. 제 2 데이터 구조(306)와 연관된 부모 객체 식별자(301)에 대한 액세스 리스트는 제 4 액세스 기록(308)만을 포함한다.

액세스 리스트 상의 각 기록 또는 데이터 구조는 자녀 객체 식별자(310)(즉, AObjectID), 제 2 카운트(312)(즉, RCount), 다음 기록 데이터 포인터(313)(즉, Nxt)를 포함하는 수 개의 필드로 이루어져 있다. 데이터 구조는 일반적인 데이터 구조, 기록, 어레이 형태일 수 있다. 도 4에 도시한 바와 같이, 데이터 구조는 액세스 리스트(370) 내에 링크 리스트를 형성하는 기록이다.

대체적인 실시예에서, 통계 테이블의 데이터 포맷은 트리 구조 내에 있는 다수의 교대 2차 어레이(alternate secondary arrays)와 관련된 1차 어레이(primary array)를 포함할 수 있다. 1차 어레이는 부모 객체와 관련된 정보를 포함할 것이고, 2차 어레이는 대응 부모 객체의 자녀 객체에 관한 정보를 포함할 것이다.

자녀 객체 식별자 필드(310)는 객체의 객체 식별자(URL)를 포함한다. 부모 객체 식별자(301)는 액세스 리스트 상의 하나 이상의 자녀 객체 식별자(310)와 관련된다. 자녀 객체 식별자(310)는 자녀의 부모 객체가 실재로 조회되는 경우 잠재적인 사전 인출 후보를 식별한다. 제 2 카운트(312)는 자신의 부모 객체를 통해 자녀 객체가 조회된 회수를 포함한다. 달리 말해서, 제 2 카운트는 자녀 객체가 연관된 부모 객체의 이전 조회 바로 다음에 후속적, 일시적으로 조회된 회수이다. 제 2 카운트(312)는 자녀 객체 액세스 빈도를 나타낸다.

액세스 리스트(370) 상의 자녀 객체 식별자(310)는 클라이언트의 실제 조회 습성을 관찰하여 동적으로 얻을 수 있다. 예를 들어, HTML에서, 웹 페이지는 다른 웹 페이지로의 HTTP 링크를 구비할 수 있다. HTTP 링크에 의해 표시된 이들 웹 페이지는 HTTP 링크를 포함하는 (부모) 웹 페이지에 대한 잠재적인 자녀 객체이다. 부모 객체가 관찰될 때, 부모 객체 내에 포함된 조회(HTTP) 링크가 클릭(click)되면, 대응 자녀 객체가 조회된다. 자녀 객체에 대한 조회 링크가 다수의 다른 부모 객체 내에 포함될 수 있으므로, 자녀 객체는 하나 이상의 부모를 가질 수 있다. HTTP 프로토콜 하에서, 자녀 객체에 대한 요청은 자신의 헤더 내의 조회자 정보(referrer information)로서 부모 URL을 포함한다. 따라서, 서버는 (자녀 헤더 내의 조회자 정보를 통해) 후속적인 자녀 조회를 자신의 부모에 동적으로 관련시킬 수 있다. 클라이언트가 자녀 객체를 자주 클릭하거나 선택하면, 자녀 객체는 사전 인출 후보나 선택 정보가 될 것이다.

관찰자 조회 습성을 동적으로 추적할 뿐 아니라, 웹 로그(Web log) 상의 정적 분석 또는 발굴(mining)도 주기적으로 수행되어 통계 테이블(261)을 위한 객체 그룹에 관한 정보를 유도할 수 있다. 발굴 알고리즘은 함께 조회된 객체 그룹을 식별하여 PHI를 위한 잠재적인 사전 인출 후보를 얻는다. 웹 로그를 발굴하여 조회 패턴을 유도하는 한 방법은 건룽우(Kun-lung Wu) 등에 의한 "SpeedTracer : A Web Usage Mining and Analysis Tool", IBM Research Report 20895, (1997년 5월)(IBM Systems Journal v37, nl, p89-105, 1998 조회)에 설명되어 있다. SpeedTracer 알고리즘은 사용자 세션에서 함께 조회된 상습 페이지 그룹과 상습 통과 경로에 관한 정보를 제공한다. SpeedTracer 알고리즘은 즉시 함께 조회되지 않는 관련 조회를 드러낼 수 있는 면밀한(in-depth) 유형의 발굴이다. 또한, 주어진 객체로의 통과 경로에 기초하여 조회된 객체의 다음 세트에 관해 보다 나은 예측을 할 수도 있다.

다른 유형의 데이터 발굴 알고리즘은 통계 테이블(261)에 대한 데이터 액세스 패턴을 식별하는 데 사용될 수 있다. 예컨대, 다른 데이터 발굴 알고리즘은 조회 패턴을 얻기 위한 웹 로그 분석에도 응용될 수 있는 분류 및 군집(classification and clustering)을 포함한다. 또한, PHI는 자신의 객체 관계에 대한 지식에 기초하여 컨텐트 제공자에 의해 공급될 수도 있다. 이를테면, PHI는 요청된 소프트웨어 다운로드, 요청된 JAVA 애플릿을 위한 자바 클래스, 요청된 객체 내에 포함된 (GIF와 같은) 객체를 인출하고 업그레이드(upgrade)할 수 있지만, 여기에 국한되지는 않는다.

바람직한 실시예에서, 통계 자료를 수집할 때, 클라이언트 스테이션에 의한 객체의 실제 조회는 객체에 대한 사전 인출 요청과는 구별된다. 통계 테이블(261) 내의 제 1 카운트(302)(즉, TCount)와 제 2 카운트(310)(즉, RCount)는 사전 인출이 아닌 실제 조회를 반영한다. 즉, 사전 인출 동작으로 인해 제 1 통계 테이블(293)이나 제 2 통계 테이블(267)을 갱신하게 되지 않으며, 실제 객체 조회만이 이들을 갱신할 것이다.

사전 인출 힌트 정보나 사전 인출 힌트 정보 리스트는 요청된 정보나 요청된 객체에 주석을 달기 위해 다양한 데이터 구조를 사용할 수 있다. 예를 들어, 사전 인출 힌트 정보는 운반된 메타 데이터를 이용하여 요청자와 서버 노드 사이에서 통신할 수 있다. HTTP 구현에서, 정보 교환은 기존의 웹 프로토콜을 이용하여 객체 헤더 내에 있는 메타 데이터로서 포함될 수 있다. PICS(Platform for Internet Content Selection)는 전자 컨텐트(electronic content)에 관한 메타 정보를 전송하는 방법을 명시한다. PICS는 (1) 요청된 정보에 주석 달기, (2) 선택 정보, 사전 인출 후보, 또는 자녀 객체의 사전 인출, (3) 정보를 사전 인출한 후 사전 인출 동작 승인에 적용할 수 있다.

PICS는 당업자에게는 웹 컨소시엄 프로토콜 권장(Web Consortium Protocol Recommendation)으로 알려져 있다. PICS는 "How much nudity is associated with this content"와 같은 가치 참조 평가 라벨(values-based rating labels)을 송신하기 위해 먼저 사용되었지만, 메타 정보의 포맷과 의미는 전적으로 일반적이다. PICS에서 전자 컨텐트에 관한 메타 정보는 "평가 서비스(rating services)"나 정보의 생산자-및-의도된-사용(producer-and-intended-usage)에 따라 그룹화되고, 이러한 하나의 그룹 내에서, 임의 개수의 카테고리나 치수의 정보가 전송될 수 있다. 각 카테고리는 허용된 값의 범위를 갖고, 특정 컨텐트에 대해 특정 카테고리는 단일한 값 또는 다수의 값을 가질 수 있다. 또한, ("PICS 라벨"로 알려진) 메타 정보 그룹은 만기 정보(expiration information)를 포함할 수 있다. 또한, PICS 라벨이 하나 이상의 전자 컨텐트에 적용되도록 하는 설비도 있다. 특정 전자 컨텐트를 위한 각각의 PICS 라벨은 독립적으로 컨텐트에 추가되거나 컨텐트로부터 제거될 수 있다.

예를 들어, 영상 파일은 자신의 "평가 서비스" 필드가 자신이 "SafeSurf" 평가 시스템에 따른 가치 참조 평가 라벨을 포함하고 있음을 나타내는 PICS 라벨과 함께 서버로부터 전달될 수 있다. 본 발명의 바람직한 실시예에 따르면, 파일이 기획 프록시(enterprise proxy)를 통과할 때, 영상 파일은 PICS 라벨을 위한 새롭게 갱신된 카테고리 값을 갖게 되어, "평가 서비스"의 지역적 고려 사항을 반역할 수 있다. 따라서, 클라이언트 컴퓨터는 PICS 라벨의 새로운 카테고리 값만을 보게 된다. HTTP프로토콜은 자신의 요청 헤더와 응답 헤더가 PICS를 지원하도록 증가(augment)되었다. NNTP와 같은 기타 일반적인 애플리케이션 프로토콜을 규정하는 기술적인 개체(technical bodies)도 이제 PICS 지원을 추가할 것을 고려하고 있다. 이들 프로토콜의 일부로서, PICS 라벨 유형 리스트가 요청과 함께 포함될 수 있다. 또한, PICS는 중앙 라벨 사무국 서버(central label bureau server)로부터 PICS 정보를 요청하는 문의 포맷을 명시하고 있다.

샘플 PICS 라벨은 다음과 같으며,

("http://the.content" exp "1997.07.09T08:15-0500" r (n 4 s 3 v 2)을 위한 PICS-1.1 "http://the.rating.service" 라벨)

여기서, "n", "s", "v"는 다양한 메타 정보 유형의 전송 이름이고, 이 컨텐트에 대한 응용가능 값은 (n에 대해) 4, (s에 대해) 3, (v에 대해) 2이다. ID "http://the.rating.service"를 인식하는 소프트웨어만이 이들 카테고리와 값을 해석하는 방법을 알 것이다.

바람직한 실시예에서는, 3 개의 다른 종류의 PICS가 도입된다. 첫 번째 종류의 PICS 라벨은 사전 인출 라벨(즉, P 라벨)이라 지칭하며, PHI를 제공하기 위해 서버 노드에 의해 사용된다. 두 번째 종류의 PICS 라벨은 요청 라벨(즉, R 라벨)이라 지칭하며, 사전 인출 후보 객체를 요청하는 데 사용된다. 세 번째 종류의 PICS 라벨은 조회 라벨(X 라벨)이라 지칭하며 어느 사전 인출 객체가 실제로 조회되었는가에 기초하여 사전 인출의 효율에 관한 피드백을 제공한다.

사전 인출 라벨은 식별자 카테고리, 확률 카테고리, 크기 카테고리, 캐싱 카테고리를 포함하는 다음 카테고리의 임의의 조합을 포함할 수 있으나, 여기에 국한되지는 않는다. 식별자 카테고리(즉, ID 카테고리)는 사전 인출 후보의 URL을 전달하는 값을 갖는다. 확률 카테고리(즉, PROB 카테고리)는 사전 인출의 유용성의 추정치를 전달하는 값을 갖는다.

바람직한 실시예에서는 조회 확률의 추정치가 사용된다. 조회 확률은 사전 인출 후보가 조회되는 확률을 나타낸다. 조회 확률은 사전 인출 후보가 조회될 확률을 나타낸다. 크기 카테고리는 사전 인출 후보의 크기 또는 메모리 조건을 전달하는 값을 갖는다. 캐싱 카테고리는 자신의 제 1 주된 캐쉬(294)나 자신의 제 1 사전 인출 캐쉬(296) 내에 사전 인출 후보 객체의 복사본을 갖는 임의의 상위 계층 프록시 서버에 관한 정보를 전달하는 값을 갖는다. 바람직한 실시예에서는 가장 가까운 상위 계층 프록시만이 식별된다.

요청 라벨은 식별자 카테고리(즉, ID 카테고리)를 포함하는 다음 카테고리를 포함할 수 있으나, 여기에 국한되지는 않는다. 식별자 카테고리 값은 사전 인출될 객체의 URL을 전달한다. 바람직한 실시예에서, (프록시 또는 클라이언트) 노드가 제시된 사전 인출 후보 객체를 사전 인출하려고 하면, HTTP 헤드 요청이 요청 라벨을 이용하여 프록시 계층을 통해 컨텐트 서버(20)로 돌려보내져서 사전 인출될 객체의 URL을 전달한다. 프록시 서버(24)가 하위 계층 노드로부터 사전 인출 요청을 수신할 때, 객체가 자신의 주된 캐쉬나 사전 인출 캐쉬 내에 있으면 그 요청을 서빙할 것이다. 그렇지 않으면, 사전 인출 요청을 다음 상위 계층 프록시로 전달한다.

조회 라벨은 객체 식별자 카테고리(Object-ID 카테고리)와 조회 식별자 카테고리(Referrer-ID 카테고리)를 포함하는 다음 카테고리를 포함할 수 있지만, 여기에 국한되지는 않는다. 객체 식별자 카테고리는 조회된 사전 인출 객체의 URL을 전달하는 값을 갖는다. 조회 식별자 카테고리는 객체 식별자 카테고리 값이 나타내는 자신의 PHI 객체 내에 포함된 객체의 URL을 전달하는 값을 갖는다.

바람직한 실시예에서, 사전 인출 객체가 실제로 조회되면, HTTP 헤드 요청이 조회 라벨을 이용하여 프록시 계층을 통해 컨텐트 서버(20)로 돌려보내져서, 조회된 사전 인출 객체와 자신의 조회자 객체(referrer object)의 URL을 전달한다. 이러한 유형의 HTTP 헤드 요청을 이후로부터는 사전 인출 조회 메시지로 지칭한다. 프록시 서버(24)가 하위 계층 노드로부터 사전 인출 조회 메시지를 수신하면, 자신의 제 1 통계 테이블(293)을 갱신하고 그 요청을 다음 상위 계층 프록시 서버로 전달한다.

도 4는 컨텐트 서버(20) 내의 제 2 통계 테이블(267)을 갱신하는 서버 로직(268)의 예를 도시하고 있다. 제 2 통계 테이블(267)의 갱신은 클라이언트(22)로부터의 객체 요청에 의해 트리거된다. 컨텐트 서버(20)는 요청된 객체를 클라이언트(22)에게 보내기 이전에 객체의 객체 헤더로의 주석 달기 또는 삽입을 위한 사전 인출 힌트 정보를 발생시키는 것이 바람직하다.

조회 블럭(405)에서 시작하여, 컨텐트 서버(20)는 입력을 기다린다. 판단 부호(410)에서 수신된 입력이 객체 O에 대한 객체 요청이면, 통계 갱신 유닛(252) 또는 제 2 갱신 통계 루틴(269)이 조회 블럭(415) 내에서 호출된다. 조회 블럭(420)에서는, 사전 인출 힌트 정보(266)를 발생시키는 발생 유닛(266) 또는 사전 인출 발생 루틴이 호출된다. 발생 유닛(266)이나 PHI 발생 루틴의 상세한 설명은 도 6을 조회하여 이후에 설명하겠다. 조회 블럭(425)에서, PHI는, 예를 들어, 요청된 객체의 객체 헤더로 삽입된다. 그렇지 않으면, 사전 인출 힌트 정보는 블럭(425)에서 요청된 객체 또는 요청된 정보에 주석을 단다. 바람직한 실시예에서는 PICS 프로토콜을 사용하여 요청된 객체에 주석을 단다. PHI는 P 라벨 내에서 포착(capture)되는데, 여기서 식별자 카테고리 값은 사전 인출 후보의 URL을 반영한다. 블럭(430)에서, 요청된 객체는 요청하는 클라이언트나 요청하는 프록시 서버로 돌아간다. 판단 부호(435)에서, 수신된 입력이 사전 인출 조회 메시지(즉, 사전 인출 객체 O가 실제로 조회되었음을 나타내는 X 라벨을 갖는 HTTP 헤드 요청)이면, 통계 갱신 수단(252) 또는 제 2 갱신 통계 루틴(269)이 단계(440)에서 호출된다. FTP(file transfer protocol) 요청과 같은 다른 형태의 입력에 대한 것은 본 발명의 초점이 아니며, 조회 블럭(445)에서 적절한 기타 처리기가 호출될 수 있다.

도 5는 통계 갱신 유닛(252)이 상기 통계 테이블(261) 중 하나를 갱신하는 방법을 도시하고 있다. 통계 갱신 유닛(252)은 제 1 통계 테이블(293)을 갱신하는 제 1 갱신 통계 루틴(264)이나 제 2 통계 테이블(267)을 갱신하는 제 2 갱신 통계 루틴(269), 또는 제 1 갱신 통계 루틴(264)과 제 2 갱신 통계 루틴(269) 모두를 조회한다. 통계 테이블(261)은 제 1 통계 테이블(293)이나 제 2 통계 테이블(267), 또는 제 1 통계 테이블(293)과 제 2 통계 테이블(267) 모두를 조회한다. 통계 갱신 유닛(252)은 먼저 조회자 객체 내의 정보를 찾는다. 이어서, 요청된 객체는 액세스 리스트나 조회자 객체의 사전 인출 리스트 상에 배치된다.

조회 블럭(505)에서 시작하여, 통계 테이블(261)은 제 1 카운트를 증가시킴으로써 객체 O가 한번 더 조회되었음을 반영하도록 갱신된다. 달리 말해서, 블럭(505)의 TCount(O)가 1만큼 증가한다. 객체 O가 통계 테이블(261) 내에 아직 존재하지 않으면, TCount(O)를 1로 초기화하여 부모 객체 식별자가 통계 테이블로 삽입된다. 통계 테이블(261) 내에 이용가능한 빈 슬롯이 없으면, 새로운 부모 객체 식별자는 가장 먼저 조회된 엔트리를 대체한다. 판단 부호(510)에서, 요청된 객체는 예를 들어, 요청된 객체의 객체 헤더를 조사함으로써 조회자 정보를 위해 조사된다. 이를테면, HTTP 프로토콜에서, 조회자 객체가 헤더 내에 제공되어, 어느 객체가 요청된 객체로의 HTTP 링크를 포함하는지를 나타낸다. 조회자 객체(R)가 헤더 내에서 발견되면, 단계(520)에서 객체(O)가 이미 객체(R)의 액세스 리스트 상에 존재하는지는 알기 위해 통계 테이블(261)이 조사된다. 객체(O)가 이미 존재하면, 블럭(530)에서 요청된 객체에 대한 제 2 카운트(즉, RCount(O))가 1만큼 증가한다. 그렇지 않으면, 제 2 카운트(즉, RCount(O))를 1로 초기화하여 블럭(525)에서 요청된 객체(즉, 객체 O)가 부모 객체(즉, 객체 R)의 액세스 리스트로 삽입된다.

도 6은 사전 인출 힌트 정보를 사전 인출 힌트 정보 리스트로 발생시키는 발생 유닛(266)의 예를 도시하고 있다. 사전 인출 힌트 리스트는 제 1 카운트에 대한 제 2 카운트의 비율이 힌트 문턱값을 초과하면 자녀 객체의 자녀 객체 식별자를 포함한다. 사전 인출 힌트 정보 리스트는 정제된, 또는 필터링된 액세스 리스트를 나타낸다. 즉, 발생 유닛(266)은 통계 테이블(261)로부터 입력을 수용하고 입력을 필터링하여 사전 인출 힌트 정보 리스트를 얻는다.

사전 인출 힌트 정보를 발생시키는 발생 유닛(266)은 조회 블럭(610)에서 시작되는 사전 인출 힌트 정보(PHI) 발생 루틴(266)을 나타낸다. 조회 블럭(610)에서, 부모 객체(즉, 객체 O)의 액세스 리스트가 확인되어 리스트가 비었는지(즉, Alist(O)가 0인지)를 알기 위해 확인된다. 액세스 리스트가 비어있지 않으면, 조회 블럭(620)에서 Cj를 부모 객체(즉, 객체 O)의 액세스 리스트 상에서 식별된 다음 자녀 객체(즉, 자녀 객체 식별자)로 설정한다.

판단 부호(630)에서, 자녀 객체 Cj의 조회 확률은 사전결정된 문턱값(예컨대, 0.75)인 힌트 문턱값(즉, TH)과 비교된다. 바람직한 실시예에서, Cj의 조회 확률은 제 1 카운트에 대한 제 2 카운트의 비율(즉, RCount(Cj)/TCount(O))로 정의된다. 이 비율은 부모 객체(객체 O)가 조회된 후 자녀 객체(객체 Cj)가 조회될 확률의 추정치이다. 판단 부호(630)에서, Cj의 조회 확률이 힌트 문턱값(즉, TH)보다 크면, 자녀 객체 식별자(Cj)는 조회 블럭(640)에서 PHI 리스트 내에 포함될 것이다. 달리 말해서, 비율이 힌트 문턱값을 초과하면, 자녀 객체 식별자와 연관된 정보가 사전 인출 힌트 정보 리스트 상에 배치된다.

발생 유닛(266)은 조회 블럭(630)에서의 비교를 위해 소프트웨어 서브루틴이나 소프트웨어 구성요소를 요청할 수 있다. 특히, 사전 인출 힌트 정보를 사전 인출 힌트 정보 리스트로 발생시키는 발생 유닛(266)은 조회 확률을 계산하는 계산 수단과 조회 확률을 힌트 문턱값과 비교하는 비교 수단을 포함할 수 있다.

바람직한 실시예에서, PICS 프로토콜을 사용하여 PHI 정보가 P 라벨 내에 저장되는데, 자녀 객체(Cj)의 URL은 식별자 카테고리(즉, ID 카테고리 값)로서 취해지고, 그 참조 확률과 크기는 확률 카테고리(즉, PROB)와 크기 카테고리 값으로서 각각 취해진다. 판단 부호(645)에서, 객체 O의 액세스 리스트가 완전히 스캐닝되지 않으면, 조회 블럭(620)이 재실행될 것이다.

사전 인출 후보를 통계 테이블(261)로부터 선택하기 위해 다양한 대체적인 참조가 사용될 수 있다. 예를 들어, 과거 시간 간격을 통한 조회 확률의 변화하는 평균이 참조으로서 사용될 수도 있다. 또한, 사전 인출 후보의 선택 과정을 반복적으로 만들 수도 있다. 자녀 객체(객체 Cj)가 부모 객체(객체 O)의 사전 인출 후보로 선택되면, 자녀 객체(객체 Cj)의 액세스 리스트 상에 있는 임의의 자녀 객체(객체 Cji)는 부모 객체(객체 O)의 사전 인출 리스트 내에 포함되지 않았다면 포함될 지 여부를 판단하기 위해 평가될 수 있다. 판단 부호(630)에서, 비교 유닛이나 평가 유닛은 이제 RCount(Cj)/TCount(O)와 RCount(Cji)/TCount(Cj)의 결과가 힌트 문턱값(TH)보다 큰지를 테스트한다.

도 7은 클라이언트 로직(244)의 동작의 예를 도시하고 있다. 도 7은 객체가 요청되면 클라이언트 객체 요청 처리기(247)가 호출되는 것을 도시한다. 그렇지 않으면, 사전 인출 처리기(241)가 요청되지 않은 객체를 처리하는데, 클라이언트(22)의 제 2 사전 인출 캐쉬(249) 내에 저장될 필요는 없다.

블럭(710)에서 시작하여, 클라이언트(22)는 입력을 대기한다. 판단 부호(720)에서, 수신된 입력이 객체이면, 제 2 사전 인출 처리기(241)가 조회 블럭(730)에서 호출된다. 블럭(740)에서는, 클라이언트 객체 처리기가 호출되어 객체를 클라이언트(22)에서 요청하는 애플리케이션으로 반환한다. 판단 부호(750)에서 수신된 입력이 객체 요청이면, 조회 블럭(760)에서 클라이언트 객체 요청 처리기(247)가 호출된다. (푸쉬(push) 요청과 같은) 다른 유형의 입력에 대해서는, 본 발명의 논점이 아니며, 적절한 기타 처리기가 조회 블럭(770)에서 호출될 수 있다.

도 8은 제 1 사전 인출 처리기(248)나 제 2 사전 인출 처리기(241), 또는 제 1 사전 인출기(248)와 제 2 사전 인출 처리기(241) 모두를 포함하는 사전 인출 유닛(250)의 동작의 예를 도시하고 있다. 간단히 말해서, 사전 인출 유닛(250)은 도 6의 발생 유닛(266)에 의해 발생한 사전 인출 리스트를, 사전 인출 힌트 정보 리스트 상에서 조회된 자녀 객체가 사전 인출 캐쉬(298) 중 하나로의 저장을 보장하는지를 판단하는 입력으로 사용한다. 사전 인출 유닛(250)은 사전 인출 힌트 정보 리스트 상의 각 사전 인출 객체 후보나 자녀 객체 식별자에 대한 사전 인출 값을 계산한다. 사전 인출 값은 자녀 객체의 액세스 확률과 자녀 객체를 저장하는 액세스 비용을 고려한다. 최고 문턱값을 갖는 자녀 객체는 사전 인출 캐쉬(298) 중 하나 내의 이용가능한 공간이 찰 때까지 먼저 캐싱된다.

조회 블럭(805)에서 시작하여, L을 사전 인출 후보 객체의 PHI 리스트로 설정한다. 사전 인출 후보 객체는 사전 인출 힌트 정보 리스트 상의 자녀 객체 식별자의 관점에서 표현될 수 있다. 지역 캐쉬(즉, 사전 인출 캐쉬 또는 주된 캐쉬) 내에 나타난 임의의 사전 인출 후보는 L로부터 제거된다. 블럭(810)에서는, L 내의 각 객체의 사전 인출 값(PV)이 계산된다. 바람직한 실시예에서, PV값은 조회 확률과 액세스 비용의 곱으로 정의된다. 액세스 비용은 PHI 리스트 상에서 식별된 사전 인출 후보 객체나 자녀 객체를 캐싱하는 가장 가까운 사이트(site)에 기초하여 추정된다. 실제적인 문제로서, 자녀 객체는 전체 자녀 객체로서보다는 자녀 객체 식별자에 의해 PHI 리스트 상에서 식별되어 저장 공간을 절약하는 것이 바람직하다. 조회 확률은 확률 카테고리(PROB 카테고리) 값으로부터 구하고, 가장 가까운 캐싱 사이트는 PICS 라벨의 캐싱 카테고리 값으로부터 구하는 것이 최적이다. 블럭(820)에서는, 최고 PV 값을 갖는 사전 인출 후보 객체가 선택된다. 블럭(830)에서는, 사전 인출 객체 후보(즉, Oj)를 수용하는 데 이용가능한 사전 인출 버퍼 공간의 양이 판단된다. 이 공간은 평가되고 있는 현재 사전 인출 객체 후보(즉, Oj)의 PV 값의 일부분(예컨대, 절반)인 보다 낮은 PV 값을 갖는 객체에 의해 현재 사용되거나 점유되고 있지 않다. 사전 인출 캐쉬(298) 중 하나에서 삭제가능한 것으로 표시된 객체에 의해 점유된 임의의 저장 공간은 현재 사용 중이 아닌 것으로 간주된다. 판단 부호(840)에서, 사전 인출 캐쉬(298) 중 하나를 조사하여 Oj를 위한 충분한 공간이 존재하는지를 확인한다. 충분한 공간이 존재하면, 조회 블럭(850)에서 객체 Oj가 사전 인출 캐쉬(298) 중 하나로 사전 인출된다. 블럭(860)에서는 Oj가 PHI 리스트(즉, L)로부터 제거된다. 조회 블럭(870)에서, L이 비어있지 않으면, 조회 블럭(820)과 연관된 단계가 재실행된다.

바람직한 실시예에서는, 각 객체가 한 번에 하나씩 사전 인출되지만, 하나의 요청에 따라 다수의 객체가 함께 사전 인출될 수도 있다. 조회 블럭(850)에서는, Oj가 사전 인출 리스트 상에 오를 수 있다. 블럭(870)에서, L이 비었으면, 사전 인출 요청이 전체 사전 인출 리스트와 함께 발생한다. 또한, 사전 인출 캐쉬 내에 충분한 저장 공간이 존재하지 않아도, 대기 사전 인출 리스트(a pending prefetch list)가 유지되어, 저장이 가능할 때 추가 사전 인출을 개시할 수도 있다.

HTTP 프로토콜을 이용함으로써, 사전 인출 중에 통신 세션을 개방한 채로 유지하여 통신 오버헤드를 줄일 수 있다. 각 사전 인출에 대해 세션을 재설정할 필요는 없다.

도 9는 클라이언트 객체 요청 처리기(247)의 동작의 예를 도시하고 있다. 요약하면, 클라이언트 객체 요청 처리기(247)는 먼저 제 2 주된 캐쉬(246)를 검색하고, 다음에 제 2 사전 인출 캐쉬(249)를 검색한다. 객체가 제 2 주된 캐쉬(246) 내에도 존재하지 않고, 제 2 사전 인출 캐쉬(249) 내에도 존재하지 않으면, 객체 요청이 서버로 전달된다. 객체가 제 2 사전 인출 캐쉬(249) 내에 존재하면, 사전 인출 조회 메시지가 서버 중 하나로 전달되어 사전 인출 객체가 실제로 조회되었음을 나타낸다. 그런 다음, 제 2 사전 인출 처리기(241)와 클라이언트 객체 처리기가 사용된다.

판단 부호(905)에서 시작하여, 요청된 객체 O가 제 2 주된 캐쉬(246) 내에서 발견되면, 제 2 사전 인출 처리기(241)가 조회 블럭(925)에서 호출되어 객체 O와 연관된 PHI에 기초하여 적절한 사전 인출 동작을 개시한다. 조회 블럭(930)에서는, 클라이언트 객체 처리기가 호출되어 클라이언트 측의 요청 애플리케이션으로 요청된 객체를 반환한다. 판단 부호(910)에서, 요청된 객체 O가 제 2 사전 인출 캐쉬(249)에서 발견되면, 자신의 제 2 통계 테이블(267)을 갱신하여 사전 인출 객체의 실제 조회를 포착할 수 있도록, 블럭(920)에서 사전 인출 조회 메시지가 컨텐트 서버(20)로 전달된다. 사실 상, 사전 인출 조회 메시지가 프록시 계층을 통해 컨텐트 서버(20)로 전달될 때, 각각의 중간 프록시 내의 제 2 통계 테이블(267)도 갱신될 것이다. 블럭(910)에서, 객체가 제 2 사전 인출 캐쉬(249) 내에 존재하지 않으면, 조회 블럭(915)에서 객체 요청이 프록시 계층을 통해 컨텐트 서버(20)로 전달된다.

사전 인출 조회 메시지는 일괄 처리(batch)될 수 있다. 즉, 사전 인출 조회 리스트가 유지될 수 있고, 주기적으로 합성 사전 인출 조회 메시지가 상위 계층 서버로 전달되어, 사전 인출 조회 리스트 상의 모든 객체의 조회를 승인한다.

도 10은 프록시 서버 로직(295)의 동작의 예를 도시하고 있다. 간단히 말해서, 입력이 객체 요청인지, 객체 전송인지, 사전 인출 조회 메시지인지에 따라 입력이 저장된다. 입력에 따라, 3 가지 서로 다른 처리기 또는 소프트웨어 루틴, 즉, 프록시 객체 요청 처리기(283), 프록시 객체 처리기(284), 사전 인출 조회 메시지 처리기(285)를 이용할 수 있다.

조회 블럭(1005)에서 시작하여, 프록시 서버(24)는 입력을 대기한다. 판단 부호(1010)에서, 수신된 입력이 객체 O에 대한 객체 요청이면, 프록시 서버(24)는 블럭(1015)에서, 그 객체 O에 관해 대기 중인 요청이 이미 존재하는지를 확인한다. 대기 중인 요청은 이 프록시 서버(24)에 의해 발생한 그 객체 O에 대한 사전 인출 요청 또는, 프록시 서버 계층에서 하위인 다른 프록시 서버 노드로부터 기인한 객체 O에 대한 이전 요청으로 인한 것일 수 있다. 조회 블럭(1015)에서 객체 O에 대해 대기 중인 요청이 존재하지 않으면, 프록시 객체 요청 처리기(283)가 블럭(1020)에서 호출된다. 판단 부호(1025)에서, 수신된 입력이 (상위 계층 프록시로부터의) 객체이면, 프록시 객체 처리기(284)가 조회 블럭(1030)에서 호출된다. 그렇지 않으면, 블럭(1040)에서, 수신된 블럭이 사전 인출 조회 메시지이면, 사전 인출 조회 메시지 처리기(285)가 블럭(1045)에서 호출된다. (FTP 요청과 같은) 다른 유형의 입력에 대해서는, 본 발명의 논점이 아니며, 블럭(1060)에서 적절한 기타 처리기가 호출될 수 있다.

도 11은 프록시 객체 요청 처리기(283)의 동작의 예를 도시하고 있다. 요약해서, 객체가 사전 인출 캐쉬 내에 있으면, 그 객체는 요청하는 클라이언트나 요청하는 프록시 서버로 반환되거나 전달된다. 그런 다음, 요청된 객체가 사전 인출 요청이면, 요청된 객체는 프록시 서버(24)의 제 1 사전 인출 캐쉬(296)로부터 삭제가능한 것으로 표시된다. 요청된 객

체에 대해 사전 인출 요청이 제공되지 않으면, 사전 인출 조회 메시지가 컨텐트 서버(20)로 전달된다. 그러나, 사전 인출 조회 메시지는 요청된 객체가 제 1 사전 인출 캐쉬(296)로부터 전달된 때에만 전달된다. 요청된 객체가 제 1 사전 인출 캐쉬(296) 내에도 존재하지 않고, 제 1 주된 캐쉬(294) 내에도 존재하지 않으면, 객체 요청이 다른 프록시 서버(24)나 컨텐트 서버(20)로 전달된다. 제 1 주된 캐쉬(294) 내에서 발견된 요청된 객체는 요청하는 클라이언트나 프록시 서버(24)로 반환되거나 전달되고, 요청이 사전 인출 요청이 아니었으면 제 1 통계 테이블(293)이 갱신된다.

판단 부호(1105)에서, 요청된 객체 O가 사전 인출 캐쉬 내에서 발견되면, 객체 O는 블럭(1130)에서 요청하는 노드로 반환된다. 판단 부호(1135)에서, 요청이 사전 인출 요청이면, 제 1 사전 인출 캐쉬(296) 내의 객체 O는 블럭(1150)에서 삭제가능한 것으로 표시된다. 제 1 사전 인출 캐쉬(296)로부터 객체를 삭제하면, 사전 인출 객체 O가 이제는 프록시 서버(24)의 하위 계층에 있는 소정 노드에 의해 사전 인출된 것이므로, 장래의 사전 인출을 위한 공간을 만들게 된다. 그렇지 않으면, 객체 O가 실제로 조회된 것이므로, 조회 블럭(1140)에서, 캐쉬 관리자가 호출되어 객체 O를 제 1 사전 인출 캐쉬(296)로부터 제 1 주된 캐쉬(294)로 이동시킨다. 블럭(1160)에서는, 사전 인출 조회 메시지가 프록시 계층 서버를 통해 컨텐트 서버(20)로 전달된다. 블럭(1170)에서는, 갱신 통계 루틴이 호출되어, 사전 인출 객체 O의 실제 조회를 포착하도록 지역 제 1 통계 테이블(293)을 갱신한다. 판단 부호(1110)에서, 객체 O가 제 1 주된 캐쉬(294) 내에서 발견되지 않으면, 조회 블럭(1125)에서 객체 요청이 프록시 서버 계층을 통해 컨텐트 서버(20)로 전달된다. 판단 부호(1115)에서, 객체 요청이 사전 인출 요청이 아니면, 블럭(1120)에서 제 1 갱신 통계 루틴이 호출되어, 객체의 실제 조회를 포착하도록 지역 제 1 통계 테이블(293)을 갱신한다. 블럭(1180)에서, 객체 O는 요청 노드로 반환된다.

도 12는 사전 인출 조회 메시지 처리기(285)의 동작의 예를 도시하고 있다. 사전 인출 조회 메시지 처리기(285)는 먼저 객체가 제 1 사전 인출 캐쉬(296) 내에 있는지 여부를 조사한다. 객체가 제 1 사전 인출 캐쉬(296) 내에 있으면, 사전 인출 조회 메시지를 전달하기 이전에 캐쉬 관리자 루틴이 호출된다. 객체가 제 1 사전 인출 캐쉬(296) 내에 존재하지 않으면, 사전 인출 조회 메시지가 전달된다.

단계(1205)에서 시작하여, 제 1 갱신 통계 루틴(264)이 호출되어, 사전 인출 객체(O)의 실제 조회를 포착하도록 지역 제 1 테이블(293)을 갱신한다. 판단 부호(1210)에서, 객체 O가 제 1 사전 인출 캐쉬(296) 내에 존재하면, 객체가 실제로 조회된 것이므로, 블럭(1215)에서 캐쉬 관리자가 호출되어 객체 O를 제 1 사전 인출 캐쉬(296)로부터 제 1 주된 캐쉬(294)로 이동시킨다. 블럭(1220)에서, 사전 인출 조회 메시지는 다음 상위 계층 프록시로 전달된다.

도 13은 프록시 객체 처리기(284)의 동작의 예를 도시하고 있다. 프록시 객체 처리기(284)는 예외 없이 항상 사전 인출 갱신 유닛(291)이나 사전 인출 힌트 정보 갱신 루틴을 먼저 호출한다. 사전 인출 객체 요청은 사전 인출 객체가 하위 레벨 서버에 의해 요청되었는지 여부에 따라 달리 처리된다. 사전 인출 객체가 하위 레벨 서버에 의해 요청되었으면, 그 객체는 하위 레벨 서버로 반환되고, 그렇지 않으면, 사전 인출 캐쉬 관리자가 호출된다. 사전 인출 객체가 요청되지 않으면, 그 객체는 요청하는 프록시 서버(24)나 클라이언트에게 반환되고, 제 1 갱신 통계 루틴(264)이 호출된다.

조회 블럭(1305)에서 시작하여, PHI 갱신 루틴이나 사전 인출 갱신 수단(291)이 호출되어 객체 O의 PHI를 갱신한다. 조회 블럭(1310)에서, 수신된 객체가 사전 인출 객체가 아니면, 제 1 사전 인출 처리기(248)가 단계(1315)에서 호출된다. 조회 블럭(1320)에서는, 객체가 (하위 계층에 있는) 요청하는 노드로 반환된다. PICS 프로토콜을 이용하는 바람직한 실시예 하에서, (식별자 카테고리 값으로 표시된) 대응 사전 인출 후보가 이 프록시 서버(24)에 의해 사전 인출되면, P 라벨 내의 캐싱 카테고리 값이 갱신된다. 블럭(1325)에서는, 캐쉬 관리자가 호출되어 주된 캐쉬 내에 객체를 저장한다. 조회 블럭(1355)에서는, 갱신 통계 루틴이 호출된다. 블럭(1330)에서, 수신된 사전 인출 객체가 계층 내의 하위 레벨 노드에 의해 요청되면, 객체는 조회 블럭(1350)에서 요청자에게 반환된다. 그렇지 않으면, 객체는 현재 노드에 의해 요청된 것이고, 사전 인출 캐쉬 관리자가 조회 블럭(1340)에서 호출되어, 객체를 위한 공간을 만들기 위해 필요한 교체(replacement)를 함으로써 객체를 사전 인출 캐쉬 내에 저장한다.

도 14는 사전 인출 갱신 수단(291) 또는 PHI 갱신 루틴의 예를 도시하고 있다. PHI 갱신 루틴은 요청된 객체의 지역 카운트를 문턱 카운트와 비교한다. 요청된 객체가 제 1 최소 문턱값을 초과하면, 제 2 최소 문턱값 테스트를 받게 된다. 문턱값은 실험 결과에 기초하여 설정될 수 있다. 지역 카운트가 제 1 최소 문턱값보다 작으면, 원래 사전 인출 힌트 정보 리스트가 유지된다. 지역 카운트가 제 1 문턱값보다 크면서 제 2 문턱값보다는 크지 않으면, 사전 인출 힌트 정보 리스트가 수정된다.

판단 부호(1405)에서 시작하여, 지역 제 1 통계 테이블(293) 내의 객체(O)의 지역 카운트(즉, TCount)가 제 1 최소 문턱값(즉, CTM)보다 크지 않으면, 객체 O의 헤더 내의 원래 PHI가 유지된다. 이는 프록시 서버(24)의 하위 계층 내의 객체가 임의의 갱신 동작을 보장할 만큼 충분히 조회하지 않았기 때문이다. 판단 부호(1415)에서, 지역 통계 테이블(261) 내의 객체(O)의 지역 카운트(즉, TCount)가 제 2 최소 문턱값(즉, TCH)보다 크면, 객체 헤더 내의 원래 PHI는 무시되고, PHI 발생 루틴이 블럭(1420)에서 호출되어, 지역 제 1 통계 테이블(293)에 기초하여 PHI를 발생시킨다. 이는 지역 습성에 기초하여 PHI를 발생시키기 위해 프록시 서버(24)의 하위 계층 내에 있는 객체에 대한 충분한 조회가 있기 때문이다. 그렇지 않으면, 조회 블럭(1425)에서 PHI 수정기가 호출되어, 각 객체에 대해 조회 확률의 지역 추정치와 PHI 리스트 내의(상위 계층에 의해 설정된) 헤더 내의 값의 평균을 취함으로써 확률 카테고리 값을 수정한다.

프록시 서버(24) 중 일부가 제 1 통계 테이블(293)을 유지하지 않고 지역 조회 패턴을 추적하지 않더라도, 사전 인출 방안을 제 역할을 한다. 비추적(non-tracking) 프록시 서버는 상위 레벨 프록시 서버나 컨텐트 서버(20)로부터 전달된 PHI 내의 확률(즉, PROB) 카테고리 값을 단순히 수용한다.

또한, 사전 인출 방안은 이종(heterogeneous) 프록시 서버 환경에서도 동작하는데, 여기서 프록시 서버의 일부는 사전 인출 프로토콜을 이해하지 못하고 협동 작업에 참여하지 않는 통상적인 프록시 서버이다. 이들 통상적인 프록시 서버는 객체 헤더를 해석하여 사전 인출 동작을 개시하지도 않을 것이며, 제 1 통계 테이블(293)을 유지하지도 않을 것이다. 통상적인 프록시 서버는 객체를 지역적으로 캐싱하기로 결정하면, 메타 데이터(예컨대, 헤더 내의 PICS 라벨)만을 객체와 함께 다음 계층 프록시 서버로 전달하거나, 메타 데이터를 객체와 함께 저장할 것이다.

프록시 계층이 없어도, 본 발명은 임의의 네트워크나 클라이언트 서버 환경에 적용할 수 있다. 객체 송신기 노드는 조회 패턴을 추적하고, PHI를 유도하며, 이를 객체 헤더 내에 포함시킬 수 있다. 또한, 객체 수신기 노드는 PHI를 자신의 캐쉬 내에 객체와 함께 저장할 수 있다. 객체 조회 시, 저장된 PHI, 캐쉬 크기와 전술한 액세스 비용과 같은 지역적 인자에 기초하여 사전 인출 동작이 개시될 수 있다.

PHI를 캐싱된 객체와 함께 저장하면, 사전 인출을 위한 추가적인 기회를 제공한다. 보다 많은 저장이 가능해지거나, 추가 조회가 발생하거나, 사전 인출 객체가 수신되는 것과 같은 이벤트는 캐싱된 객체의 PHI를 조사함으로써 추가적인 사전 인출을 트리거하는 데 사용될 수 있다. 사전 인출 동작은 송신기 개시 접근 방법에서처럼 객체가 서버로부터 요청되는 경우에만 제한되지 않는다.

바람직한 실시예에는, 웹과 프록시 서버를 위한 일반적인 사전 인출 방안이 설명되었지만, 당업자는 본 발명이 사전 인출될 객체가 유사한 특성을 갖는 임의의 상황에 적용가능하며, 인터넷이나 WWW 애플리케이션에 제한될 필요가 없다는 것을 이해할 것이다.

또한, 본 발명의 바람직한 실시예는 계층 내의 부모 및 자녀 프록시 사이의 협동 작업인 사전 인출을 위한 것이지만, 형제 프록시(sibling proxies) 사이의 협동 작업을 포함하도록 용이하게 변형될 수 있다. 예를 들어, 상위 계층 내의 프록시가 요청된 객체나 사전 인출 후보를 캐싱하지 않았으면, 프록시는 형제 프록시에게 문의할 수도 있다.

(57) 청구의 범위

청구항 1.

삭제

청구항 2.

삭제

청구항 3.

삭제

청구항 4.

삭제

청구항 5.

삭제

청구항 6.

삭제

청구항 7.

삭제

청구항 8.

삭제

청구항 9.

삭제

청구항 10.

삭제

청구항 11.

삭제

청구항 12.

삭제

청구항 13.

삭제

청구항 14.

삭제

청구항 15.

삭제

청구항 16.

삭제

청구항 17.

삭제

청구항 18.

네트워크를 통해 서버로부터 클라이언트로의 데이터 흐름을 개선시키는 처리 시스템에 있어서,

상기 네트워크와 연관된 프록시 서버들과,

상기 네트워크와 연관된 컨텐트 서버와,

상기 프록시 서버들 중 하나 이상과 통신하도록 구성된 클라이언트와,

상기 프록시 서버들 중 적어도 하나의 프록시 서버에 저장되어 있는 통계 테이블 - 상기 통계 테이블은 부모 객체 식별자 및 연관된 자녀 객체 식별자를 저장하며, 상기 통계 테이블은 부모 객체가 상기 클라이언트에 의해 조회되는 빈도를 나타내는 부모 객체의 제 1 카운트와, 상기 부모 객체 중 선택된 하나가 조회된 후에 대응 자녀 객체가 조회되는 빈도를 나타내는 자녀 객체의 제 2 카운트를 포함함 - 과,

사전 인출 힌트 정보 리스트를 유도하기 위하여, 상기 제 1 카운트 및 상기 대응하는 제 2 카운트의 입력을 수용하여 상기 사전 인출 힌트 정보 리스트를 발생시키는 발생 수단을 포함하고,

상기 발생 수단은

각각의 상기 부모 객체 식별자에 대해 상기 제 1 카운트에 대한 상기 제 2 카운트의 비율을 계산하는 계산 수단과,

힌트 문턱값(hint threshold)에 대한 비율을 비교하는 비교 수단 - 상기 비교 수단은 상기 비율이 상기 힌트 문턱값을 초과하면 상기 사전 인출 힌트 정보 리스트 상에서 자녀 객체 및 연관된 부모 객체를 그룹화하거나 연관시킴 - 과,

상기 사전 인출 힌트 정보 및 상기 클라이언트에 의해 실제로 요청된 부모 객체에 기초하여 자녀 객체를 사전 인출하는 사전 인출 수단 - 상기 사전 인출 수단은 상기 프록시 서버들 중 적어도 하나에 저장되어 있는 제 1 사전 인출 처리기를 포함하고, 상기 사전 인출 수단은 사전 인출 값을 계산하는 계산 수단과, 상기 계산 수단에 의해 계산된 상기 사전 인출 값을 평가하는 평가 수단을 포함하고, 상기 계산 수단은 상기 사전 인출 힌트 정보 리스트 상의 각각의 자녀 객체 식별

자에 대해 사전 인출 값을 계산하며, 상기 평가 수단은 다수의 자녀 객체의 상기 사전 인출 값을 평가하고 상기 사전 인출 힌트 정보 리스트 상의 최고 사전 인출 값과 연관된 자녀 객체의 캐싱의 우선 순위를 매기고, 상기 제 1 사전 인출 처리기는 상기 프록시 서버들 중 하나와 연관된 사전 인출 캐쉬 내의 가용 공간을 관리함 - 을 포함하는

처리 시스템.

청구항 19.

네트워크를 통해 서버로부터 클라이언트로의 데이터 흐름을 개선시키는 처리 시스템에 있어서,

상기 네트워크와 연관된 프록시 서버들과,

상기 네트워크와 연관된 컨텐트 서버와,

상기 프록시 서버들 중 하나 이상과 통신하도록 구성된 클라이언트와,

상기 프록시 서버들 중 적어도 하나의 프록시 서버에 저장되어 있는 통계 테이블 - 상기 통계 테이블은 부모 객체 식별자 및 연관된 자녀 객체 식별자를 저장하며, 상기 통계 테이블은 부모 객체가 상기 클라이언트에 의해 조회되는 빈도를 나타내는 부모 객체의 제 1 카운트와, 상기 부모 객체 중 선택된 하나가 조회된 후에 대응 자녀 객체가 조회되는 빈도를 나타내는 자녀 객체의 제 2 카운트를 포함함 - 과,

사전 인출 힌트 정보 리스트를 도출하기 위하여 상기 제 1 카운트 및 상기 대응하는 제 2 카운트의 입력을 수용하여, 상기 사전 인출 힌트 정보 리스트를 발생시키는 발생 수단을 포함하고,

상기 발생 수단은

각각의 상기 부모 객체 식별자에 대해 상기 제 1 카운트에 대한 상기 제 2 카운트의 비율을 계산하는 계산 수단과,

힌트 문턱값에 대한 비율을 비교하는 비교 수단 - 상기 비교 수단은 상기 비율이 상기 힌트 문턱값을 초과하면 상기 사전 인출 힌트 정보 리스트 상에서 자녀 객체 및 연관된 부모 객체를 그룹화하거나 연관시킴 - 과,

상기 사전 인출 힌트 정보 및 상기 클라이언트에 의해 실제로 요청된 부모 객체에 기초하여 자녀 객체를 사전 인출하는 사전 인출 수단 - 상기 사전 인출 수단은 상기 프록시 서버들 중 적어도 하나에 저장되어 있는 제 1 사전 인출 처리기와 상기 클라이언트 중 적어도 하나에 저장되어 있는 제 2 사전 인출 처리기를 각각 포함하고, 상기 사전 인출 수단은 사전 인출 값을 계산하는 계산 수단과, 상기 계산 수단에 의해 계산된 상기 사전 인출 값을 평가하는 평가 수단을 포함하고, 상기 계산 수단은 상기 사전 인출 힌트 정보 리스트 상의 각각의 자녀 객체 식별자에 대해 사전 인출 값을 계산하고, 상기 사전 인출 힌트 정보 리스트 상의 최고 사전 인출 값과 연관된 자녀 객체의 캐싱의 우선 순위를 매기며, 상기 사전 인출 처리 수단은 상기 사전 인출 캐쉬 내의 가용 공간을 관리함 - 을 포함하는

처리 시스템.

청구항 20.

제 18 항 또는 제 19 항에 있어서,

상기 프록시 서버들 중 하나는 사전 인출 캐쉬를 가지며, 상기 사전 인출 수단은 자녀 객체를 사전 인출하고 우선 순위 방안(a priority scheme)에 기초하여 상기 사전 인출 캐쉬에 상기 자녀 객체를 저장하고, 상기 우선 순위 방안은 상기 사전 인출 캐쉬가 채워질 때까지 최고 사전 인출 값을 갖는 객체를 먼저 저장하고, 상기 사전 인출 값은 하나 이상의 클라이언트에 의한 상기 자녀 객체에 대한 개별적이거나 통합된 조회 가능성(reference probability)에 기초하는

처리 시스템.

청구항 21.

제 18 항 또는 제 19 항에 있어서,

상기 프록시 서버는 각각 제 1 프로세서, 제 1 메모리, 데이터를 저장하는 제 1 저장 수단을 포함하고, 상기 제 1 메모리는 제 1 주된 캐쉬(main cache)와 제 1 사전 인출 캐쉬 사이에 적어도 논리적인 분리를 가지며,

상기 컨텐트 서버는 제 2 프로세서, 제 2 메모리, 데이터 저장을 위한 제 2 저장 수단을 가지며,

상기 클라이언트들 중 적어도 한 클라이언트는 제 3 프로세서, 제 3 메모리, 데이터를 저장하는 제 3 저장 수단을 구비하되, 상기 제 3 메모리는 제 2 주된 캐쉬와 제 2 사전 인출 캐쉬 사이에 적어도 논리적인 분리를 갖고, 상기 사전 인출 처리 수단은 상기 제 1 사전 인출 캐쉬나 상기 제 2 사전 인출 캐쉬 내에 자녀 객체를 저장하는

처리 시스템.

청구항 22.

제 18 항 또는 제 19 항에 있어서,

실제로 요청된 자녀 객체에 기초하여 상기 통계 테이블을 갱신하는 통계 갱신 수단을 더 포함하되, 상기 통계 갱신 수단은 상기 부모 객체 중 조회하는 하나(a referring one)의 신원을 먼저 검색하고, 상기 조회하는 하나, 즉 조회자 객체(referrer object)는 상기 자녀 객체가 실제로 요청되기 이전에 조회되며, 상기 자녀 객체는 상기 통계 테이블 내에 자신의 대응하는 제 2 카운트를 갖는

처리 시스템.

청구항 23.

네트워크를 통해 서버로부터 클라이언트로의 데이터 흐름을 개선시키는 처리 시스템에 있어서,

상기 네트워크와 연관된 프록시 서버들과,

상기 네트워크와 연관된 컨텐트 서버와,

상기 프록시 서버들 중 하나 이상과 통신하도록 구성된 클라이언트와,

상기 프록시 서버들 중 적어도 하나의 프록시 서버에 저장되어 있는 통계 테이블 - 상기 통계 테이블은 부모 객체 식별자 및 연관된 자녀 객체 식별자를 저장하며, 상기 통계 테이블은 부모 객체가 상기 클라이언트에 의해 조회되는 빈도를 나타내는 부모 객체의 제 1 카운트와, 상기 부모 객체 중 선택된 하나가 조회된 후에 대응하는 자녀 객체가 조회되는 빈도를 나타내는 자녀 객체의 제 2 카운트를 포함함 - 과,

사전 인출 힌트 정보 리스트를 도출하기 위하여 상기 제 1 카운트 및 상기 대응하는 제 2 카운트의 입력을 수용하여, 상기 사전 인출 힌트 정보 리스트를 발생시키는 발생 수단

- 상기 발생 수단은

각각의 상기 부모 객체 식별자에 대해 상기 제 1 카운트에 대한 상기 제 2 카운트의 비율을 계산하는 계산 수단과,

힌트 문턱값에 대한 비율을 비교하는 비교 수단 - 상기 비교 수단은 상기 비율이 상기 힌트 문턱값을 초과하면 상기 사전 인출 힌트 정보 리스트 상에서 자녀 객체 및 연관된 부모 객체를 그룹화하거나 연관시킴 - 과,

상기 사전 인출 힌트 정보 및 상기 클라이언트에 의해 실제로 요청된 부모 객체에 기초하여 자녀 객체를 사전 인출하는 사전 인출 수단

을 포함하는 발생 수단임 - 과,

상기 사전 인출 힌트 정보 리스트를 갱신하는 사전 인출 갱신 수단 - 상기 사전 인출 갱신 수단은 상기 제 1 카운트를 한 서버로부터의 통계에 기초하는 제 1 지역 문턱 카운트(first local threshold counts) 및 다른 서버로부터의 통계에 기초하는 제 2 지역 문턱 카운트(second local threshold counts)를 비교하고, 상기 사전 인출 갱신 수단은 상기 제 1 카운트가 상기 제 1 지역 문턱 카운트와 제 2 지역 문턱 카운트를 모두 초과하는 상기 자녀 객체 식별자 각각에 대해 상기 발생 수단을 호출하며, 상기 사전 인출 갱신 수단은 상기 제 1 카운트가 상기 제 1 지역 문턱 카운트를 초과하지 않는 경우, 상기 원래의 사전 인출 힌트 정보 리스트를 유지함 - 을 포함하는

처리 시스템.

청구항 24.

제 23 항에 있어서,

상기 통계 테이블을 갱신하는 통계 갱신 수단과,

상기 프록시 서버들 중 하나에 저장되어 있는 프록시 객체 처리기 - 상기 프록시 객체 처리기는 사전 인출 힌트 정보 리스트 갱신 루틴을 호출하고, 상기 프록시 객체 처리기는 하위 레벨 프록시 서버가 객체를 요청하면 상기 요청된 객체를 상기 하위 레벨 프록시 서버에게 반환하고, 상기 프록시 객체 처리기는 상기 요청된 객체가 사전 인출 객체가 아닌 경우에는 상기 요청된 객체를 반환하고 상기 통계 갱신을 호출함 - 와,

상기 프록시 서버들 중 하나와 연관된 사전 인출 캐쉬를 관리하는 사전 인출 캐쉬 관리자 - 상기 프록시 객체 처리기는 상기 하위 레벨 서버가 상기 요청된 객체를 요청하지 않는 경우에는 상기 사전 인출 캐쉬 관리자를 호출함 - 와,

상기 프록시 서버들 중 하나에 저장되어 있는 프록시 객체 요청 처리기 - 상기 프록시 객체 요청 처리기는 상기 클라이언트들 중 요청하는 한 클라이언트에게 객체를 반환하고, 객체가 상기 사전 인출 캐쉬 내에 있으며 상기 객체가 상기 사전 인출 요청에 따라 요청된 경우 상기 사전 인출 캐쉬로부터 삭제가능한 것으로 상기 객체를 표시하고, 상기 프록시 객체 요청 처리기는 상기 요청된 객체가 주된 캐쉬나 상기 사전 인출 캐쉬 내에 있지 않으면 객체 요청을 다른 서버로 보내고, 상기 프록시 객체 요청 처리기는 상기 요청된 객체가 상기 사전 인출 요청과 연관되지 않고 상기 요청된 객체가 상기 사전 인출 캐쉬나 상기 주된 캐쉬 내에 위치하면 상기 통계 갱신 수단과 통신함 - 를 더 포함하는

처리 시스템.

청구항 25.

제 23 항에 있어서,

상기 통계 테이블을 갱신하는 통계 갱신 수단과,

상기 프록시 서버들 중 하나에 저장되어 있는 사전 인출 조회 메시지 처리기(a prefetch referenced message handler) - 상기 사전 인출 조회 메시지 처리기는 상기 통계 갱신 수단을 호출하고, 상기 사전 인출 조회 메시지 처리기는 요청된 객체가 상기 사전 인출 캐쉬 내에 위치하고 있는 경우 캐쉬 관리자 루틴을 호출한 후 사전 인출 조회 메시지를 전달함 - 를 더 포함하는

처리 시스템.

청구항 26.

제 23 항에 있어서,

상기 프록시 서버들 중 하나의 프록시 서버는 사전 인출 캐쉬를 가지며, 상기 사전 인출 수단은 자녀 객체를 사전 인출하여 상기 자녀 객체를 우선 순위 방안에 기초하여 상기 사전 인출 캐쉬 내에 저장하고, 상기 우선 순위 방안은 상기 사전 인출 캐쉬의 이용가능한 메모리가 채워질 때까지 최고 사전 인출 값을 갖는 객체를 먼저 저장하고, 상기 사전 인출 값은 하나 이상의 클라이언트에 의한 상기 자녀 객체에 대한 개별적이거나 통합된 조회 가능성(reference probability)에 기초하는

처리 시스템.

청구항 27.

제 23 항에 있어서,

상기 프록시 서버는 각각 제 1 프로세서, 제 1 메모리, 데이터 저장을 위한 제 1 저장 수단을 포함하고, 상기 제 1 메모리는 제 1 주된 캐쉬와 제 1 사전 인출 캐쉬 사이에 적어도 논리적 분리를 가지며,

상기 컨텐트 서버는 제 2 프로세서, 제 2 메모리, 데이터 저장을 위한 제 2 저장 수단을 가지며,

상기 클라이언트 중 적어도 한 클라이언트는 제 3 프로세서, 제 3 메모리, 데이터 저장을 위한 제 3 저장 수단을 가지며, 상기 제 3 저장 수단은 제 2 주된 캐쉬와 제 2 사전 인출 캐쉬 사이에 적어도 논리적인 분리를 가지며, 상기 사전 인출 처리 수단은 상기 제 1 사전 인출 캐쉬 또는 상기 제 2 사전 인출 캐쉬 내에 자녀 객체를 저장하는

처리 시스템.

청구항 28.

제 23 항에 있어서,

실제로 요청된 자녀 객체에 기초하여 상기 통계 테이블을 갱신하는 통계 갱신 수단을 더 포함하되, 상기 통계 갱신 수단은 상기 부모 객체 중 조회하는 부모 객체의 신원을 먼저 검색하고, 상기 조회하는 부모 객체, 즉, 조회자 객체는 실제로 요청된 상기 자녀 객체에 앞서 조회되고, 실제로 조회된 상기 자녀 객체는 상기 통계 테이블에서 갱신된 자신의 대응하는 제 2 카운트를 갖는

처리 시스템.

청구항 29.

네트워크를 통해 서버로부터 클라이언트로의 데이터 흐름을 개선시키는 처리 시스템에 있어서,

상기 네트워크와 연관된 프록시 서버들과,

상기 네트워크와 연관된 컨텐트 서버와,

하나의 객체로부터 다른 하나의 객체를 요청하기 위해 상기 프록시 서버들 중 하나 이상과 통신하도록 구성된 클라이언트 - 상기 하나의 객체와 다른 하나의 객체는 상기 요청을 제공하기 전에 복수의 논리 링크들 중 하나의 논리 링크에 의해 서로 결합됨 - 와,

상기 컨텐트 서버 및 상기 프록시 서버들 중 적어도 하나의 프록시 서버에 저장되어 있는 통계 테이블 - 상기 통계 테이블은 부모 객체 식별자 및 연관된 자녀 객체 식별자를 저장하며, 상기 통계 테이블은 부모 객체가 상기 클라이언트에 의해 상기 논리 링크들 중 하나에 의해 요청되는 빈도를 나타내는 부모 객체의 제 1 카운트와, 상기 부모 객체 중 선택된 하나가 조회된 직후에 상기 대응 자녀 객체가 조회되는 빈도를 나타내는 자녀 객체의 제 2 카운트를 포함함 - 과,

상기 통계 테이블로부터 제 1 카운트, 제 2 카운트, 자녀 객체 식별자, 부모 객체 식별자의 입력을 수용하는 발생 수단 - 상기 발생 수단은 부모 객체와 그들의 연관된 자녀 객체 사이의 액세스 관계를 평가하는 평가 수단을 포함하되, 상기 평가 수단은 대응 부모 객체에 대한 클라이언트에 의한 조회 이전에, 자녀 객체가 바로 다음에 후속적으로 적절한 빈도로 조회되는 경우, 사전 인출 힌트 정보 리스트 상에 자녀 객체 식별자를 배치함 - 을 포함하고,

상기 발생 수단은

각각의 상기 부모 객체 식별자에 대해 상기 제 1 카운트에 대한 상기 제 2 카운트의 비율을 계산하는 계산 수단과,

힌트 문턱값(hint threshold)에 대한 비율을 개별적으로 비교하는 비교 수단 - 상기 비교 수단은 상기 힌트 문턱값을 초과하는 부모 객체에 대하여 상기 사전 인출 힌트 정보 리스트 상에서 자녀 객체 및 연관된 부모 객체를 그룹화하거나 연관시킴 - 과,

상기 사전 인출 힌트 정보 및 상기 클라이언트에 의해 실제로 요청된 부모 객체에 기초하여 상기 논리 링크들 중 하나의 논리 링크를 통해 자녀 객체를 사전 인출하는 사전 인출 수단을 포함하는

처리 시스템.

청구항 30.

제 29 항에 있어서,

상기 발생 수단은 대응 부모 액세스에 대한 자녀 액세스의 비율에 기초하여 상기 적절한 빈도를 계산하는 계산 수단을 더 포함하는

처리 시스템.

청구항 31.

제 29 항에 있어서,

상기 사전 인출 수단은 저장 우선 순위를 상기 사전 인출 힌트 정보 리스트 상에서 식별된 자녀 객체에 할당하는 할당 수단과, 상기 저장 우선 순위에 따라서 상기 자녀 객체들 중 선택된 하나를 저장하는 선택적 저장 수단을 더 포함하는

처리 시스템.

청구항 32.

제 29 항에 있어서,

상기 할당 수단은 상기 사전 인출 힌트 정보 리스트 상의 자녀 객체가 상기 클라이언트 중 특정한 하나에 의해 액세스될 추정 액세스 확률 및 상기 자녀 객체 중 선택된 하나를 저장하는 데 필요한 저장 크기에 기초한 상기 자녀 객체의 액세스 비용에 기초하여 저장 우선 순위를 할당하는

처리 시스템.

청구항 33.

삭제

청구항 34.

삭제

청구항 35.

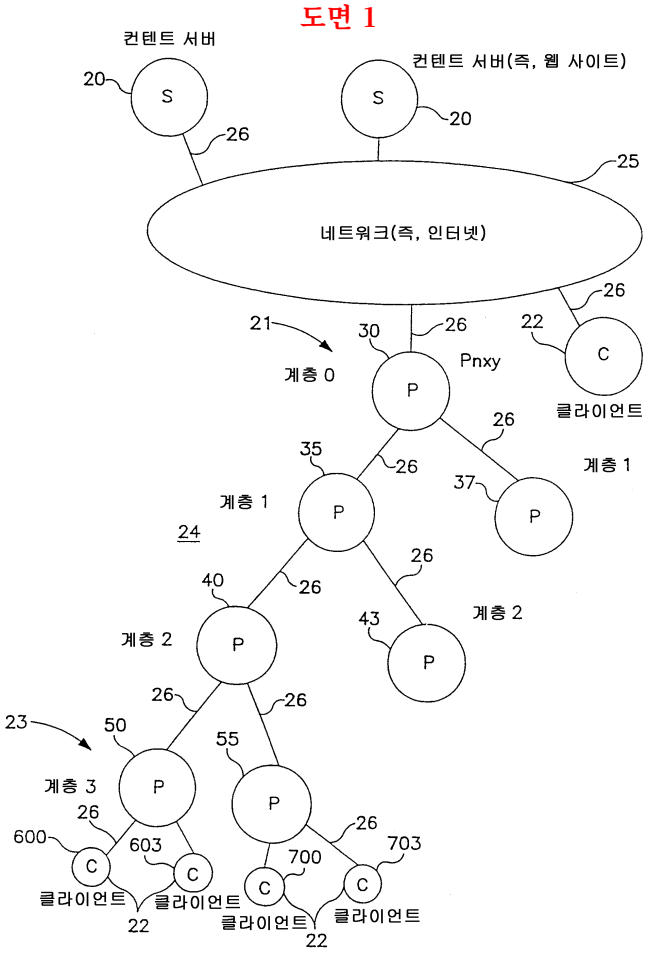
삭제

청구항 36.

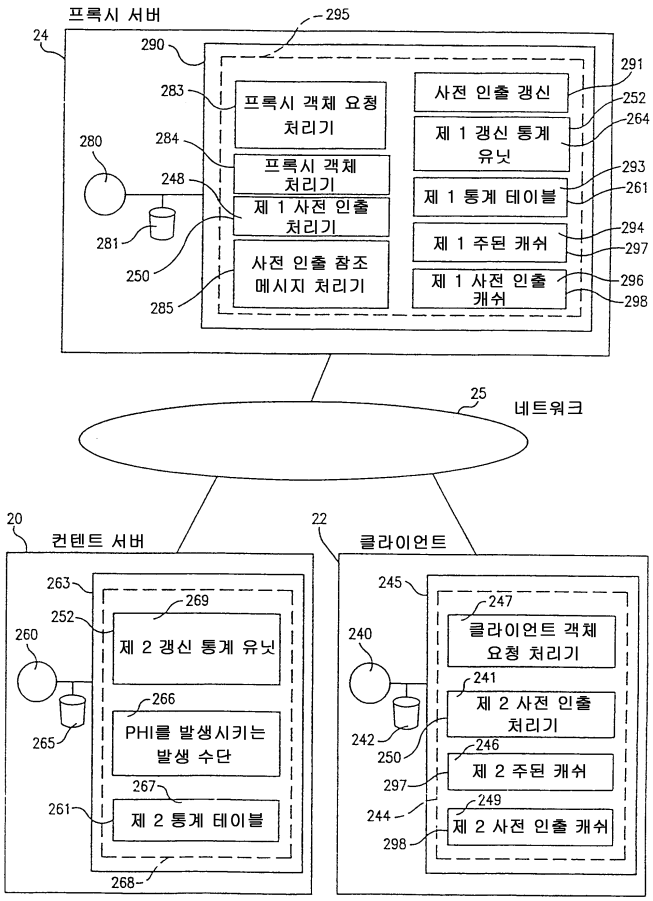
삭제

도면

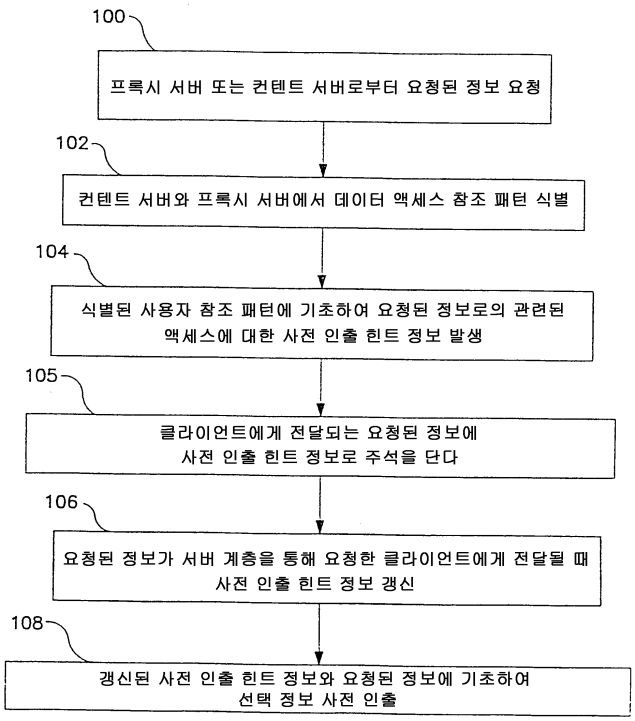
도면 1


컨텐트 서버
20
S
26
컨텐트 서버(즉, 웹 사이트)
S
20
25
네트워크(즉, 인터넷)
26
21
30
26
22
C
계층 0
P
Pnxy
클라이언트
26
35
26
계층 1
37
P
계층 1
P
24
26
40
26
43
계층 2
계층 2
P
P
26
26
23
50
55
P
P
계층 3
26
26
600
603
700
703
C
C
C
C
클라이언트
클라이언트
클라이언트
클라이언트
22
22

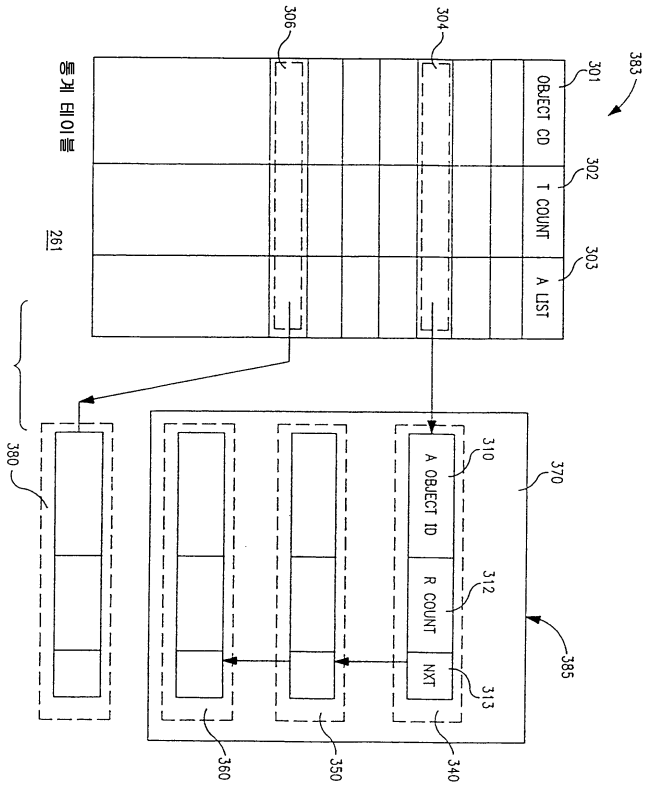
도면 2


프록시 서버
24
295
290
283
프록시 객체 요청 처리기
284
프록시 객체 처리기
248
제 1 사전 인출 처리기
사전 인출 참조 메시지 처리기
285
250
281
280
291
사전 인출 갱신
252
264
제 1 갱신 통계 유닛
293
261
제 1 통계 테이블
294
297
제 1 주된 캐쉬
296
298
제 1 사전 인출 캐쉬
25
네트워크
20
컨텐트 서버
263
269
252
제 2 갱신 통계 유닛
260
266
PHI를 발생시키는 발생 수단
265
267
261
제 2 통계 테이블
268
22
클라이언트
245
247
클라이언트 객체 요청 처리기
240
241
제 2 사전 인출 처리기
242
250
246
제 2 주된 캐쉬
297
244
249
제 2 사전 인출 캐쉬
298

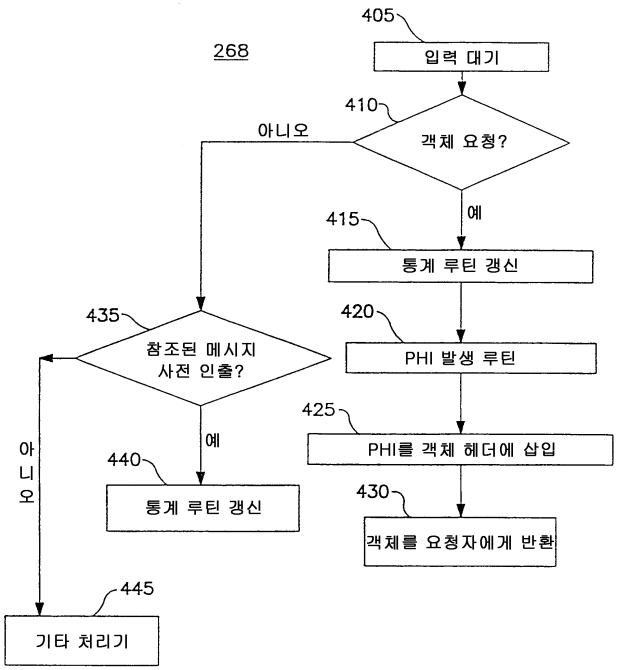
도면 3a


100
프록시 서버 또는 컨텐트 서버로부터 요청된 정보 요청
102
컨텐트 서버와 프록시 서버에서 데이터 액세스 참조 패턴 식별
104
식별된 사용자 참조 패턴에 기초하여 요청된 정보로의 관련된 액세스에 대한 사전 인출 힌트 정보 발생
105
클라이언트에게 전달되는 요청된 정보에 사전 인출 힌트 정보로 주석을 단다
106
요청된 정보가 서버 계층을 통해 요청한 클라이언트에게 전달될 때 사전 인출 힌트 정보 갱신
108
갱신된 사전 인출 힌트 정보와 요청된 정보에 기초하여 선택 정보 사전 인출

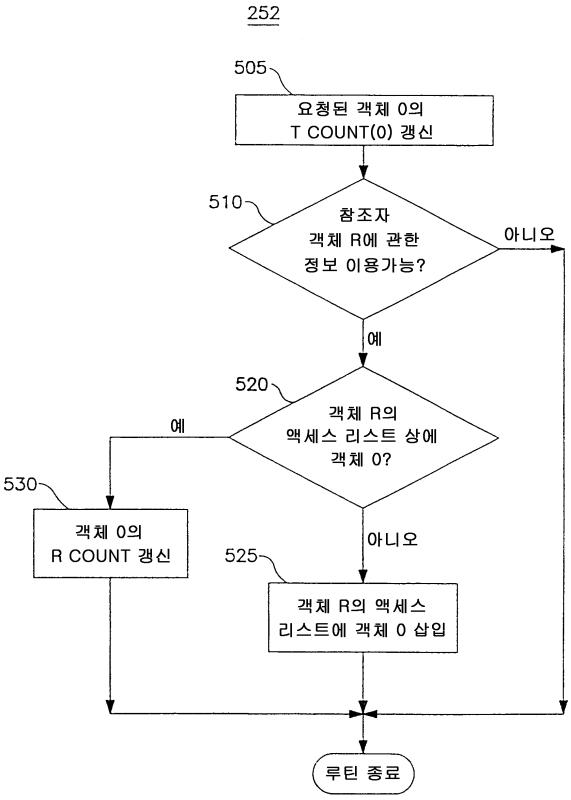
도면 3b


306
304
383
301
OBJECT CD
302
T COUNT
303
A LIST
통계 테이블
261
380
310
A OBJECT ID
370
312
R COUNT
385
313
NXT
360
350
340

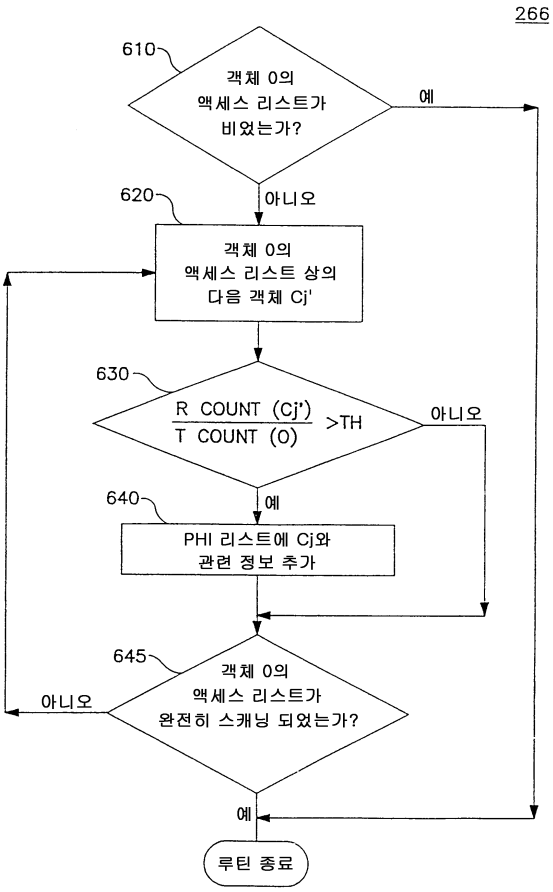
도면 4


268
405
입력 대기
410
객체 요청?
아니오
예
415
통계 루틴 갱신
420
PHI 발생 루틴
425
PHI를 객체 헤더에 삽입
430
객체를 요청자에게 반환
435
참조된 메시지 사전 인출?
예
440
통계 루틴 갱신
아니오
445
기타 처리기

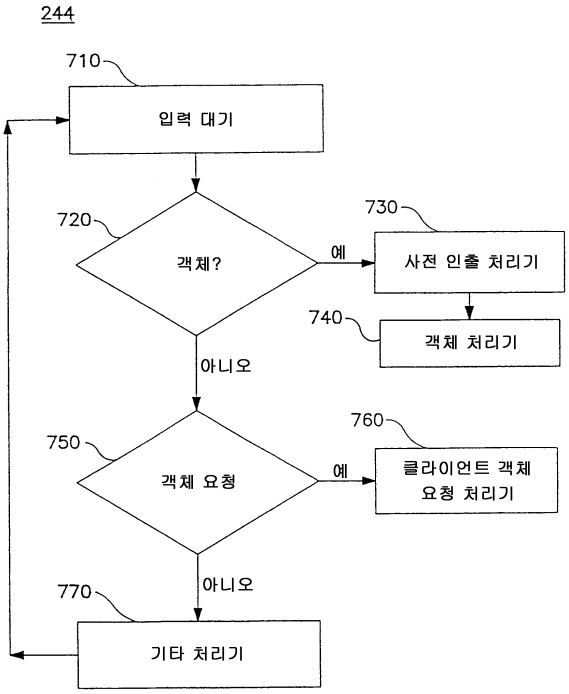
도면 5


252
505
요청된 객체 O의
T COUNT(O) 갱신
510
참조자
객체 R에 관한
정보 이용가능?
아니오
예
520
객체 R의
액세스 리스트 상에
객체 O?
예
530
객체 O의
R COUNT 갱신
아니오
525
객체 R의 액세스
리스트에 객체 O 삽입
루틴 종료

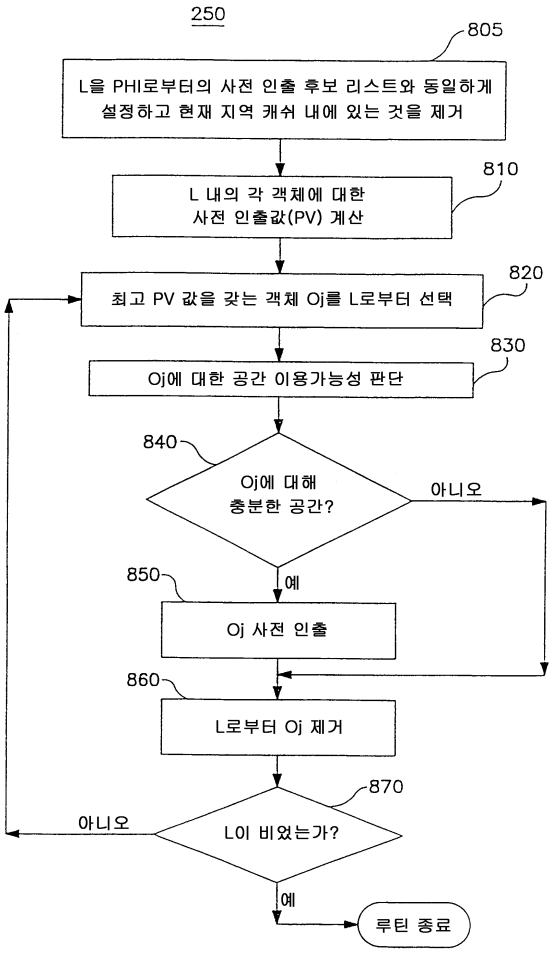
도면 6


266
610
객체 O의
액세스 리스트가
비었는가?
예
아니오
620
객체 O의
액세스 리스트 상의
다음 객체 Cj'
630
R COUNT (Cj')
T COUNT (O)
>TH
아니오
640
예
PHI 리스트에 Cj와
관련 정보 추가
645
객체 O의
액세스 리스트가
완전히 스캐닝 되었는가?
아니오
예
루틴 종료

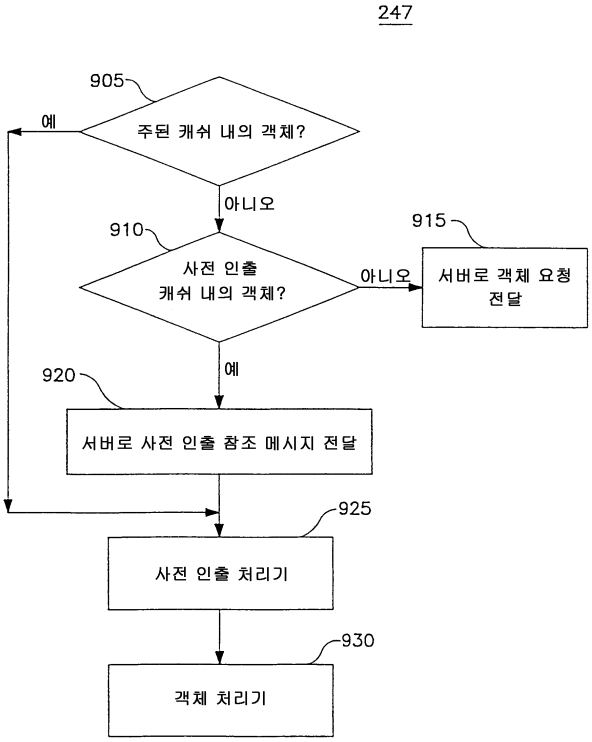
도면 7


244
710
입력 대기
720
객체?
예
730
사전 인출 처리기
740
객체 처리기
아니오
750
객체 요청
예
760
클라이언트 객체 요청 처리기
아니오
770
기타 처리기

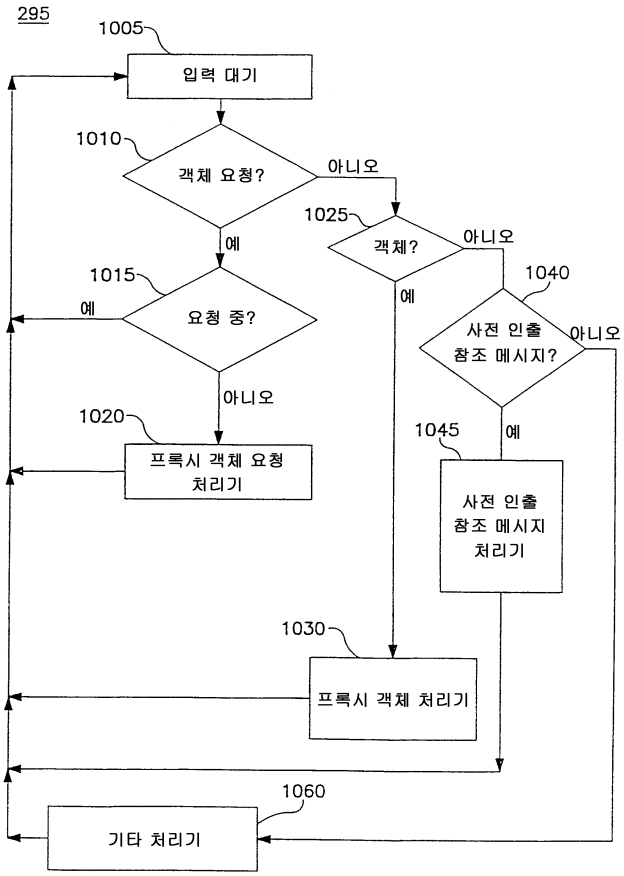
도면 8


250
805
L을 PHI로부터의 사전 인출 후보 리스트와 동일하게 설정하고 현재 지역 캐쉬 내에 있는 것을 제거
810
L 내의 각 객체에 대한 사전 인출값(PV) 계산
820
최고 PV 값을 갖는 객체 Oj를 L로부터 선택
830
Oj에 대한 공간 이용가능성 판단
840
Oj에 대해 충분한 공간?
아니오
예
850
Oj 사전 인출
860
L로부터 Oj 제거
870
L이 비었는가?
아니오
예
루틴 종료

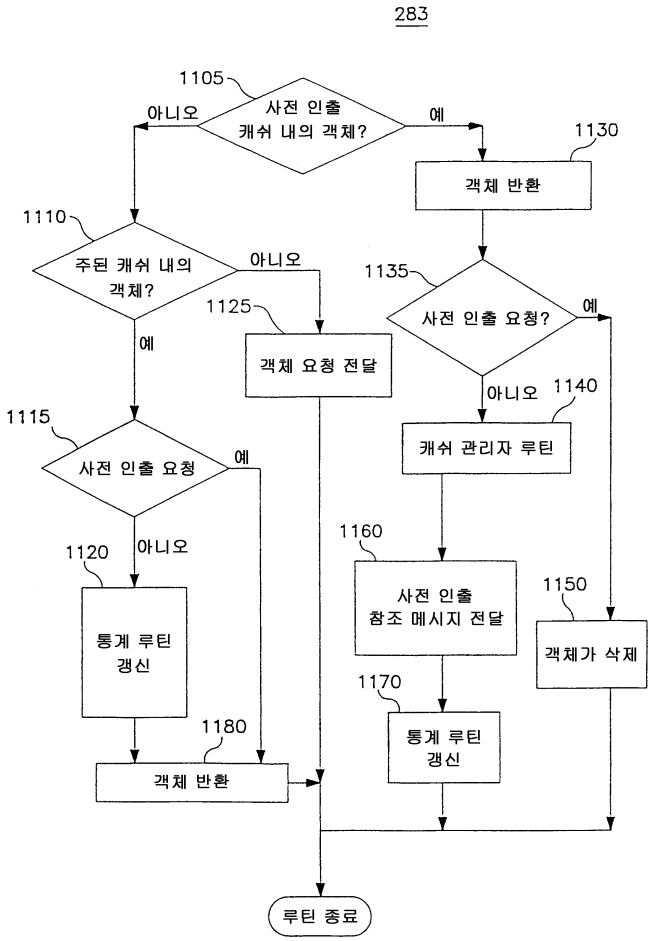
도면 9


247
905
주된 캐쉬 내의 객체?
예
아니오
910
사전 인출
캐쉬 내의 객체?
아니오
915
서버로 객체 요청
전달
예
920
서버로 사전 인출 참조 메시지 전달
925
사전 인출 처리기
930
객체 처리기

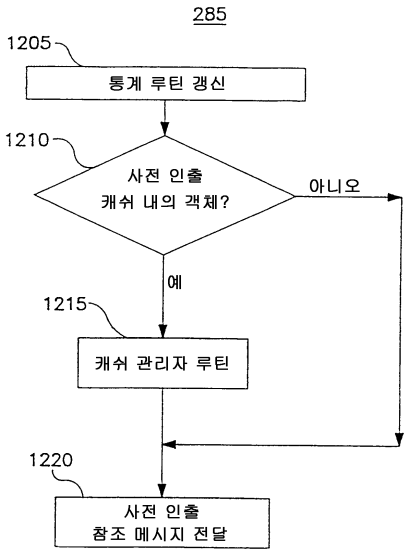
도면 10


295
1005
입력 대기
1010
객체 요청?
아니오
예
1025
객체?
아니오
예
1015
요청 중?
예
아니오
1040
사전 인출 참조 메시지?
아니오
예
1020
프록시 객체 요청 처리기
1045
사전 인출 참조 메시지 처리기
1030
프록시 객체 처리기
1060
기타 처리기

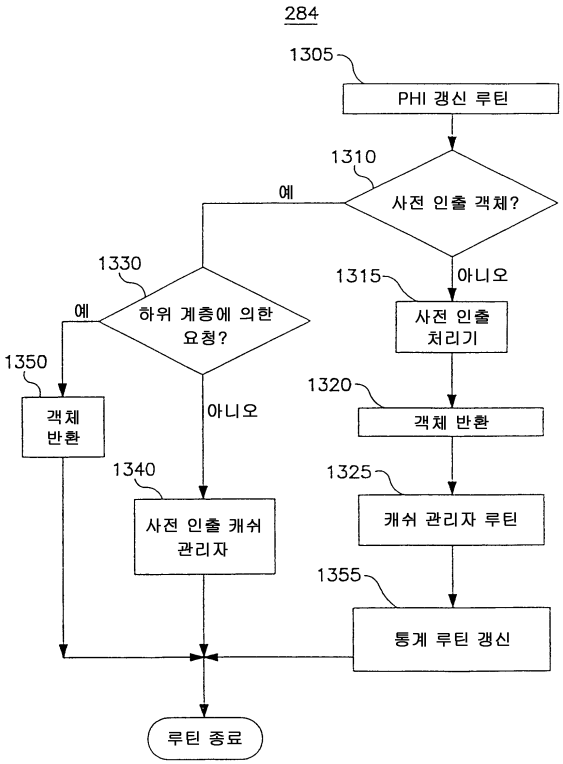
도면 11


283
1105
사전 인출 캐쉬 내의 객체?
아니오
예
1130
객체 반환
1110
주된 캐쉬 내의 객체?
아니오
예
1135
사전 인출 요청?
예
1125
객체 요청 전달
아니오
1140
캐쉬 관리자 루틴
1115
사전 인출 요청
예
아니오
1160
사전 인출 참조 메시지 전달
1150
객체가 삭제
1120
통계 루틴 갱신
1170
통계 루틴 갱신
1180
객체 반환
루틴 종료

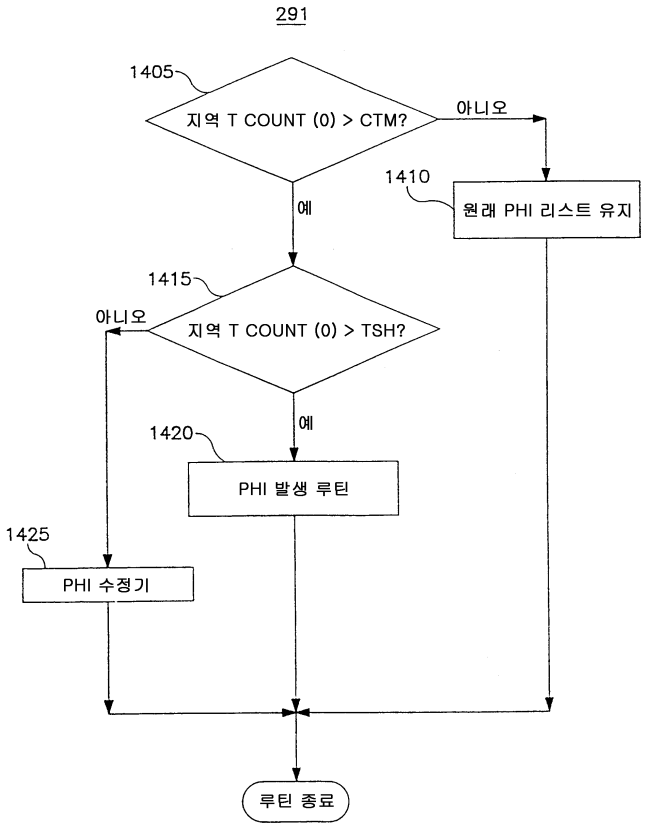
도면 12


285
1205
통계 루틴 갱신
1210
사전 인출
캐쉬 내의 객체?
아니오
예
1215
캐쉬 관리자 루틴
1220
사전 인출
참조 메시지 전달

도면 13


284
1305
PHI 갱신 루틴
1310
사전 인출 객체?
예
아니오
1330
하위 계층에 의한
요청?
예
아니오
1315
사전 인출
처리기
1350
객체
반환
1320
객체 반환
1340
사전 인출 캐쉬
관리자
1325
캐쉬 관리자 루틴
1355
통계 루틴 갱신
루틴 종료

도면 14


291
1405
지역 T COUNT (0) > CTM?
아니오
1410
원래 PHI 리스트 유지
예
1415
아니오
지역 T COUNT (0) > TSH?
예
1420
PHI 발생 루틴
1425
PHI 수정기
루틴 종료