

특허청구의 범위

청구항 1.

요청을 비동기적으로 처리하기 위한 방법에 있어서:

클라이언트로부터 수신된 유니폼 리소스 로케이터(URL)의 세션 개체를 취득하는 단계; 세션 식별자와 상기 URL에 기초하여 응답 식별자를 형성하는 단계; 상기 URL이 상기 응답 식별자에 기초하여 상기 클라이언트에 의해 이미 요청되었는지를 판정하는 단계; 및 상기 클라이언트가 상기 URL에 대한 후속의 요청을 자동으로 보내게 하기 위한 시간 값을 포함하는 응답 갱신 헤더를 형성하는 단계를 포함하는 방법.

청구항 2.

제1항에 있어서, 상기 요청에 대한 임시 응답 및 상기 응답 갱신 헤더를 상기 클라이언트에 보내는 단계를 더 포함하는 방법.

청구항 3.

제1항에 있어서, 상기 응답 식별자에 따라서 상기 요청에 대한 최종 응답을 캐시하기 위한 명령으로 상기 요청을 응답 캐싱 시스템에 보내는 단계를 더 포함하는 방법.

청구항 4.

청구항 4은(는) 설정등록료 납부시 포기되었습니다.

제1항 내지 제3항 중 어느 한 항에 있어서, 상기 요청은 웹 요청인 방법.

청구항 5.

청구항 5은(는) 설정등록료 납부시 포기되었습니다.

제1항 내지 제3항 중 어느 한 항에 있어서, 상기 응답 식별자는 상기 세션 식별자 및 상기 URL의 해쉬를 포함하는 방법.

청구항 6.

제1항 내지 제3항 중 어느 한 항에 있어서, 상기 형성 단계 이전에 상기 요청에 대한 최종 응답이 완료되었는지를 판정하는 단계; 상기 최종 응답이 완료된 경우 상기 클라이언트에 상기 최종 응답을 보내는 단계를 더 포함하고, 상기 갱신 헤더는 상기 최종 응답이 완료된 경우 형성되지 않아 상기 임시 응답으로 클라이언트에 보내지 않는 방법.

청구항 7.

제1항 내지 제3항 중 어느 한 항에 있어서, 상기 요청에 대한 최종 응답의 형성을 시작하는 단계; 및 상기 최종 응답이 완료될 때 상기 응답 식별자에 따라 상기 최종 응답을 캐시에 저장하는 단계를 포함하는 방법.

청구항 8.

청구항 8은(는) 설정등록료 납부시 포기되었습니다.

제7항에 있어서, 상기 응답 갱신 헤더의 상기 시간 값의 만료 이후에 상기 클라이언트로부터 상기 URL에 대한 후속 요청을 수신하는 단계; 상기 세션 개체를 취득하는 단계; 상기 응답 식별자를 생성하는 단계; 상기 URL이 상기 응답 식별자에 기초하여 이미 요청되었는지를 판정하는 단계; 상기 응답 식별자에 기초하여 요청에 대한 상기 최종 응답에 대해 상기 캐시를 체크하는 단계; 및 상기 최종 응답이 완료되면 상기 클라이언트에 상기 최종 응답을 보내는 단계를 더 포함하고, 상기 최종 응답이 완료되지 않은 경우 새로운 갱신 헤더가 형성되어 새로운 임시 응답으로 상기 클라이언트에 보내지는 방법.

청구항 9.

제1항 내지 제3항 중 어느 한 항에 있어서, 상기 판정 단계는 상기 응답 식별자에 대해 상기 세션 개체의 테이블을 체크하여 상기 URL이 상기 클라이언트에 의해 이미 요청되었는지를 판정하는 단계를 포함하는 방법.

청구항 10.

제1항 내지 제3항 중 어느 한 항에 있어서, 클라이언트로부터 유니폼 리소스 로케이터 (URL)에 대한 요청을 수신하는 단계를 더 포함하는 방법.

청구항 11.

청구항 11은(는) 설정등록료 납부시 포기되었습니다.

제2항 또는 제3항에 있어서, 상기 URL이 상기 클라이언트에 의해 이미 요청된 경우 상기 응답 식별자에 기초하여 상기 요청에 대한 상기 최종 응답에 대해 캐시를 체크하는 단계를 더 포함하는 방법.

청구항 12.

요청을 비동기적으로 처리하기 위한 시스템에 있어서: 클라이언트로부터 수신된 유니폼 리소스 로케이터 (URL)에 대한 요청에 대해 세션 개체를 취득하기 위한 개체 시스템; 세션 식별자와 상기 URL에 기초하여 응답 식별자를 형성하기 위한 응답 식별자 시스템; 상기 URL이 상기 클라이언트에 의해 이미 요청되었는지를 판정하기 위한 요청 체크 시스템; 및 상기 클라이언트가 URL에 대한 후속의 요청을 자동으로 보내도록 하기 위한 시간 값을 포함하는 응답 갱신 헤더를 형성하기 위한 헤더 형성 시스템을 포함하는 시스템.

청구항 13.

삭제

청구항 14.

제12항에 있어서, 상기 응답 식별자에 기초하여 상기 요청에 대한 최종 응답에 대해 캐시를 체크하기 위한 캐시 체크 시스템을 더 포함하고, 상기 최종 응답은 완료시 상기 클라이언트에 보내지는 시스템.

청구항 15.

삭제

청구항 16.

삭제

청구항 17.

삭제

청구항 18.

삭제

청구항 19.

청구항 19은(는) 설정등록료 납부시 포기되었습니다.

제14항에 있어서, 상기 응답 갱신 헤더의 상기 시간 값의 만료 이후에 상기 클라이언트로부터 상기 URL에 대한 후속 요청을 수신하기 위한 입력 시스템을 더 포함하는 시스템.

청구항 20.

삭제

청구항 21.

제1항 내지 제3항 중 어느 한 항의 모든 단계를 실행하는 데에 적합한 프로그램 코드 수단을 포함하고, 컴퓨터 상에서 실행되는 컴퓨터 판독가능 기록매체.

명세서

기술분야

본 발명은 요청을 비동기적으로 처리하기 위한 방법, 시스템 및 프로그램 제품에 관한 것이다. 본 발명은 특히 클라이언트와의 지속적인 연결을 유지하지 않고도 네트워크 계통의 요청 (예를 들어, 웹 요청)이 어플리케이션 서버에 의해 처리될 수 있도록 한다.

배경기술

인터넷의 이용이 더욱 보편화되면서, 웹 유저는 정보원으로 월드 와이드 웹에 점차 더 의존하게 된다. 전형적인 구현예로, 유저는 클라이언트에서 웹 브라우저를 운용하게 되며, 특정 웹 페이지의 "요청"을 서버에 제출한다. 서버 상의 하나 이상의 "서블렛" (등)이 요청을 처리하여 적당한 웹 페이지를 브라우저에 보내게 된다. 이렇게 보편적으로 이용되고 있는 하나의 특정 기술이 웹 포털 페이지의 개념이다. 일반적으로, 웹 포털 페이지는 유저가 목표로 하는 개인화된 콘텐츠를 수신하는 메커니즘을 제공한다. 통상, 포털 페이지는 각각 유저의 선호도에 따라 선택되어 포맷된 특정 포털 콘텐츠를 포함하는 섹션이나 실제 포틀렛을 포함한다. 예를 들어, 유저는 뉴스, 날씨 및 스포츠의 섹션을 갖는 그들만의 포털 페이지를 만들 수 있다. 포털 페이지가 요청되면, 서버 상의 포털 프로그램은 적당한 콘텐츠 프로바이더로부터 원하는 콘텐츠를 취득하게 된다. 일단 취득되면, 포털 콘텐츠가 모아지고, 다음에 웹 포털 페이지로서 적당한 섹션에 디스플레이된다. 이 기술은 개인 웹 유저가 개인화 "홈" 페이지의 전개에 이를 수 있게 한다.

불행히도, 이들 각각의 경우에, 요청과 응답의 취급은 동기적으로 행해지지 않는다. 즉, 요청을 서버에 보내고, 클라이언트와 서버 간의 연결은 응답이 돌아올 때 까지 유지되게 된다. 이런 식으로 연결을 유지하는 것은 다른 작업을 실행하려는 클라이언트의 능력을 제한하거나 방해할 뿐만 아니라, 서버가 다른 클라이언트와 연결하는 능력을 제한하거나 방해할 수 있다. 이것은 응답이 서블렛/포틀렛으로 처리되어야 하는 경우 특히 그렇다. 예를 들어, 포털 페이지의 경우, 요청에 대한 응답을 형성하는 것은 수많은 콘텐츠 소스와의 인터페이스를 필요로 한다. 이와 같이, 응답의 형성은 수초 걸릴 수 있다. 클라이언트와 서버 간의 연결을 이 전 시간에 걸쳐 유지하는 경우, 상기한 문제가 발생하게 된다.

이 기술에서는 알려진 바와 같이, 하이퍼텍스트 트랜스퍼 프로토콜 (HTTP)에서는, 서버가 일반적으로, 클라이언트와의 연결을 초기화할 수가 없다. 그보다, 클라이언트가 서버와의 연결을 초기화해야 한다. 따라서, 클라이언트와 서버 간의 연결이 요청이 처리되고 있는 동안 종료되게 되면, 클라이언트만이 새로운 연결을 초기화하여 응답을 수신할 수 있다. 현재, 기존의 기술로는 나중에 연결을 재성립하기 위해 클라이언트의 세심한 수동적 변경을 필요로 하지 않고는, 요청이 처리되고 있는 중에 클라이언트와 서버 간의 연결이 종료되는 것이 가능하지가 않다.

발명의 상세한 설명

본 발명은 클라이언트로부터의 요청이 처리되고 있는 동안 클라이언트와 서버 간의 연결이 종료되게 하고 클라이언트 측에서의 수정이나 신중한 동작을 필요로 하지 않고 나중에 클라이언트가 서버와의 새로운 연결을 자동으로 설정할 수 있게 한다.

일반적으로, 본 발명은 요청을 비동기식으로 처리하기 위한 방법, 시스템 및 프로그램 제품을 제공한다. 특히, 본 발명에서는, 유니폼 리소스 로케이터(URL)의 요청이 서버상에서 클라이언트로부터 수신된다. 수신시, 대응하는 세션 개체가 취득되고, 응답 식별자가 형성된다. 응답 식별자에 기초하여, URL이 클라이언트에 의해 이미 요청되었는지가 판정된다. 그렇지 않다면, 최종 응답의 형성이 시작된다. 응답이 형성되고 있기 때문에, 응답 갱신 헤더가 형성되어 임시 응답으로 클라이언트에 보내진다. 응답 갱신 헤더는 클라이언트가 URL에 대한 후속의 요청을 자동으로 보내도록 하기 위한 시간 값을 포함한다. 최종 응답의 형성이 완료된 후에, 이것은 응답 식별자에 따라 캐시에 저장된다. 다음에, 후속의 요청이 응답 갱신 헤더의 시간 값 만료 이후에 클라이언트로부터 수신될 때, 최종 응답이 응답 식별자에 기초하여 캐시로부터 검색되어, 클라이언트에 보내진다.

본 발명의 제1 형태는 클라이언트로부터 유니폼 리소스 로케이터(URL)에 대한 요청을 수신하는 단계; 요청에 대응하는 세션 개체를 취득하는 단계; 세션 식별자와 URL에 기초하여 응답 식별자를 형성하는 단계; URL이 응답 식별자에 기초하여 클라이언트에 의해 이미 요청되었는지를 판정하는 단계; 및 클라이언트가 URL에 대한 후속의 요청을 자동으로 보내게 하기 위한 시간 값을 포함하는 응답 갱신 헤더를 생성하는 단계를 포함하는, 요청을 비동기적으로 처리하기 위한 방법을 제공한다.

본 발명의 제2 형태는 클라이언트로부터 유니폼 리소스 로케이터(URL)에 대한 요청을 수신하는 단계; 요청에 대응하는 세션 개체를 취득하는 단계; 세션 식별자와 URL에 기초하여 응답 식별자를 형성하는 단계; URL이 응답 식별자에 기초하여 클라이언트에 의해 이미 요청되었는지를 판정하는 단계; URL이 이미 클라이언트에 의해 요청된 경우, 응답 식별자에 기초하여 요청에 대한 최종 응답에 대해 캐시를 체크하는 단계; 및 최종 응답이 완료되지 않은 경우, 클라이언트가 URL에 대한 후속의 요청을 자동으로 보내도록 하기 위한 시간 값을 포함하는 응답 갱신 헤더를 형성하는 단계를 포함하는, 요청을 비동기적으로 처리하기 위한 방법을 제공한다.

본 발명의 제3 형태는 클라이언트로부터 수신된 유니폼 리소스 로케이터 (URL)에 대한 요청에 대해 세션 개체를 취득하기 위한 개체 시스템; 세션 식별자와 URL에 기초하여 응답 식별자를 형성하기 위한 응답 식별자 시스템; URL이 클라이언트에 의해 이미 요청되었는지를 판정하기 위한 요청 체크 시스템; 및 클라이언트가 URL에 대한 후속의 요청을 자동으로 보내도록 하기 위한 시간 값을 포함하는 응답 갱신 헤더를 형성하기 위한 헤더 형성 시스템을 포함하는, 요청을 비동기적으로 처리하기 위한 시스템을 제공한다.

본 발명의 제4 형태는 요청을 비동기적으로 처리하기 위한 기록 가능 매체 상에 저장된 프로그램 제품을 제공하며, 이것은 실행시, 클라이언트로부터 수신된 유니폼 리소스 로케이터(URL)에 대한 요청에 대해 세션 개체를 취득하기 위한 프로그램 코드; 세션 식별자와 URL에 기초하여 응답 식별자를 형성하기 위한 프로그램 코드; URL이 클라이언트에 의해 이미 요청되었는지를 판정하기 위한 프로그램 코드; 및 클라이언트가 URL에 대한 후속의 요청을 자동으로 보내도록 하기 위한 시간 값을 포함하는 응답 갱신 헤더를 형성하기 위한 프로그램 코드를 포함한다.

실시예

상술한 바와 같이, 본 발명은 요청을 비동기적으로 처리하기 위한 방법, 시스템 및 프로그램 제품을 제공한다. 구체적으로, 본 발명에서는 유니폼 리소스 로케이터 (URL)에 대한 요청이 서버 상의 클라이언트로부터 수신되게 된다. 수신시, 대응하는 세션 개체가 취득되고, 응답 식별자가 형성된다. 응답 식별자에 기초하여, URL이 클라이언트에 의해 이미 요청되었는지가 판정된다. 그렇지 않다면, 최종 응답의 형성이 시작된다. 응답이 형성되고 있기 때문에, 응답 갱신 헤더가 형성되어 임시 응답으로 클라이언트에 보내진다. 응답 갱신 헤더는 클라이언트가 URL에 대한 후속의 요청을 자동으로 보내도록 하기 위한 시간 값을 포함한다. 최종 응답의 형성이 완료된 후에, 이것은 응답 식별자에 따라 캐시에 저장된다. 다음에, 후속의 요청이 응답 갱신 헤더의 시간 값 만료 이후에 클라이언트로부터 수신될 때, 최종 응답이 응답 식별자에 기초하여 캐시로부터 검색되어, 클라이언트에 보내진다. 따라서 여기에 개시된 것은 하이퍼텍스트 트랜스퍼 프로토콜 (HTTP)를 지원하는 모든 브라우저/시스템과 관련하여 구현될 수 있다.

먼저 여기에서 이용되는 용어 "요청"은 웹 요청과 같이 클라이언트로부터 서버에 발해진 네트워크 계통의 요청을 말한다는 것을 이해해야 한다. 통상적으로, 요청은 특정 유니폼 리소스 로케이터 (URL)에 대한 것이다. 이렇게 이하 후술되는 바와 같이, 요청은 하나 이상의 서블렛, 포틀렛 등에 의해 서버 상에서 처리될 수 있다.

이하 도 1을 참조하여 설명하면, 요청을 비동기식으로 처리하기 위한 시스템(10)이 나타나 있다. 본 발명에서, 클라이언트(12) 및 서버(14)는 임의 유형의 컴퓨터화 시스템을 나타낼 수 있다. 예를 들어, 클라이언트(12) 및/또는 서버(14)는 퍼스널 컴퓨터, 워크스테이션, 랩톱, 핸드헬드 장치 등일 수 있다. 일반적으로, 클라이언트(12)는 네트워크 상에서 서버(14)와 통신한다. 더구나, 이하 후술되는 바와 같이 클라이언트(12)와 서버(14) 간의 통신은 인터넷과 같은 임의 유형의 공중 네트워크 또는 근거리 통신망 (LAN), 원거리 통신망(WAN), 가상 사설망(VPN) 등의 임의의 사설망에서 발생할 수 있다. 일 실시예에서, 서버(14)는 포털 페이지를 클라이언트(12)에 전달하는 포털 서버 등의 어플리케이션 서버이다. 어느 경우이나 유저(16)는 웹 페이지를 서버(14)로부터 요청하기 위해 클라이언트(12)에서 웹 브라우저(18)를 운용하게 된다. 서버(14)는 여러 콘텐츠 소스(20)로부터 콘텐츠를 취득함으로써 웹 페이지 (예를 들어, 요청에 대한 최종 응답)를 형성하게 된다. 이것이 일단 형성되면, 웹 페이지는 요청한 클라이언트(12)로 다시 돌아간다.

나타낸 바와 같이, 서버(14)는 일반적으로 중앙 연산 유닛(CPU; 22), 메모리(24), 버스(26), 입/출력(I/O) 인터페이스(28), 외부 장치/리소스(30) 및 저장 유닛(32)을 포함한다. CPU(22)는 하나의 처리 유닛을 포함하거나, 예를 들어, 클라이언트와 컴퓨터 시스템 상에서, 하나 이상의 위치에서 하나 이상의 처리 유닛에 걸쳐 분산될 수 있다. 메모리(24)는 자기 매체, 광학 매체, 랜덤 액세스 메모리(RAM), 판독 전용 메모리(ROM), 데이터 캐시(40), 데이터 개체 등을 포함하는 임의의 공지된 유형의 데이터 저장소 및/또는 전송 매체를 포함할 수 있다. 더구나, CPU(22)와 유사하게, 메모리(24)는 하나 이상의 유형의 데이터 저장소를 포함하는 하나의 물리적 위치에 놓일 수 있거나, 여러 유형으로 된 복수의 물리적 시스템에 걸쳐 분산될 수 있다.

I/O 인터페이스(28)는 외부 소스에/로부터 정보를 교환하기 위한 시스템을 포함할 수 있다. 외부 장치/리소스(30)는 스피커, CRT, LCD 스크린, 핸드 헬드 장치, 키보드, 마우스, 음성 인식 시스템, 음성 출력 시스템, 프린터, 모니터/디스플레이, 팩시밀리, 페이저 등을 포함하는 공지된 유형의 외부 장치를 포함할 수 있다. 버스(26)는 서버(14)의 각 구성 요소 간의 통신 링크를 제공하며 유사하게 전기, 광학, 무선 등을 포함하는 공지된 유형의 전송 링크를 포함할 수 있다.

저장 유닛(32)은 본 발명에서 정보 저장을 제공할 수 있는 시스템(예를 들어, 데이터베이스)일 수 있다. 이런 정보는 무엇보다도, 세션 개체, 응답 식별자, 임시 응답 등을 포함할 수 있다. 이와 같이, 저장 유닛(32)은 자기 디스크 드라이브 또는 광 디스크 드라이브 등의 하나 이상의 저장 장치를 포함한다. 다른 실시예에서, 저장 유닛(32)은 예를 들어, 근거리 통신망 (LAN), 원거리 통신망(WAN), 또는 저장 장치 영역망(SAN)(도시 생략)에 걸쳐 분산된 데이터를 포함한다.

상기된 바와 같이, 본 발명의 개시는 인터넷, 원거리 통신망 (WAN), 근거리 통신망(LAN), 가상 사설망(VPN), 등에 의한 네트워크 환경에서 보통 구현된다. 이렇게 클라이언트(12)와 서버(14) 간의 통신은 직접적 유선 연결 (예를 들어, 직렬 포트)를 통해, 또는 유선 및/또는 무선 전송법의 조합을 이용할 수 있는 어드레스 가능한 연결을 통해 발생할 수 있다. 서버(14)와 클라이언트(12)는 토큰 링(Token Ring), 이더넷(Ethernet), WiFi 또는 그 외 종래의 통신 표준과 같은 종래의 네트워크 연결을 이용할 수 있다. 더구나, 종래의 TCP/IP 소켓 계통의 프로토콜에 의해서도 연결이 제공될 수 있다. 이 경우, 클라이언트(12)는 인터넷 서비스 프로바이더를 이용하여 서버(14)에의 연결을 성립할 수 있다.

메모리(24)에는 요청 처리 시스템(36), 서블렛/포틀렛(38) 및 캐시(40)가 나타나 있다. 일반적으로, 요청 처리 시스템(36)은 클라이언트(12) 또는 브라우저(18) 측에서의 수정이나 동작을 요하지 않고, 클라이언트(12)로부터의 요청이 비동기적으로 처리될 수 있도록 한다. 또한 요청 처리 시스템(36)은 현재의 포털 또는 어플리케이션 서버 프로그램의 구성 요소 중 일부나 모두를 포함할 수 있다. 예를 들어, 요청 처리 시스템(36)은 웹스피어(WebSphere) 어플리케이션 서버 및/또는 웹스피어 포털 서버의 구성 요소와 결합될 수 있다 (웹스피어는 IBM사의 등록 상표임).

도 1 및 도 2를 함께 참조하여, 이하 본 발명을 더욱 상세히 설명한다. 도 2에 나타난 요청 처리 시스템(36)의 실시예는 오직 설명을 위한 것임을 미리 이해해 두어야 한다. 이와 같이, 이의 서브시스템은 더 많거나 적은 서브시스템으로 구현될 수 있다. 예를 들어, 입력 시스템(50) 및 출력 시스템(68)은 하나의 "통신 시스템"에 결합될 수 있다. 어느 경우에서나, 본 발명에 의하면 요청이 클라이언트(12)로부터 보내지면, 입력 시스템(50)에 의해 수신되게 된다. 수신시, 개체 시스템(52)은 클라이언트(12)와 서버 시스템(14) 간의 세션에 대응하는 세션 개체를 취득하게 된다. 세션 개체는 무엇보다도, 세션에 대응하는 세션 식별자를 나타낸다. 세션이 새로운 세션이면, 개체 시스템(52)은 세션 개체를 형성하여 이를 저장 유닛(32)에 저장한다. 반대로, 세션이 이미 형성되어 있는 경우에는, 개체 시스템(52)은 저장 유닛(32)으로부터 세션 개체를 검색할 수

있다. 그래도, 일단 세션 개체가 취득되면, 응답 식별자 시스템(54)은 응답 식별자를 형성한다. 통상, 응답 식별자는 세션 개체로부터의 세션 식별자 및 요청된 유니폼 소스 로케이터(URL)의 해쉬(hash)를 포함한다. 이로써 요청된 특정 세션과 URL 둘다를 참조할 수가 있다.

응답 식별자가 형성되면, 요청 체크 시스템(56)은 URL이 클라이언트(12)에 의해 이미 요청된 것인지를 판정하게 된다. 구체적으로, 요청 체크 시스템(56)은 세션 개체의 "요청" 테이블을 액세스하게 된다. 응답 식별자가 여기에 목록화되어 있는 경우, 이것은 URL이 클라이언트(12)에 의해 이미 요청된 것을 의미한다. 이 경우, 캐시 체크 시스템(58)은 캐시(40)를 체크하여 요청에 대한 최종 응답이 완료되었는지를 판정한다. 일반적으로, 요청에 대한 최종 응답은 응답 식별자에 따라 응답 캐시 시스템(66)에 의해 캐시된다. 따라서, 캐시 체크 시스템은 일치하는 응답 식별자에 대해 캐시(40)를 체크할 수 있다. 최종 응답이 완료되면, 이것은 출력 시스템(68)을 거쳐 클라이언트(12)로 돌아간다.

그러나, 요청 체크 시스템(56)이 응답 식별자가 세션 개체의 요청 테이블에 목록화되어 있지 않다고 (즉, URL이 클라이언트(12)에 의해 이미 요청되지 않았다고) 판정하게 되면, 이 요청을 명령 캐싱 시스템(66)에 대한 명령으로 통신하게 된다. 이 명령은 요청에 대한 최종 응답이 응답 식별자에 따라 캐시(40)에 저장되는 것을 나타낸다. 동시에, 응답 형성 시스템(64)은 최종 응답의 형성을 시작하게 된다. 통상의 실시예에서, 응답 형성 시스템(64)은 콘텐츠 소스(20)로부터 대응하는 콘텐츠를 취득하게 되는 하나 이상의 서블렛/포틀렛(38) (도 1)을 호출하게 된다. 이렇게, 서블렛/포틀렛(38)은 표준 또는 원격 포틀렛 (예를 들어, 원격 포탈용 웹 서비스)를 포함하는 반면, 콘텐츠 소스(20)는 독자적 기구, 데이터 구조, 저장 유닛 등을 포함할 수 있다. 어느 경우에도, 최종 응답이 형성되고 있으면, 헤더 형성 시스템(60)은 클라이언트(12)가 동일한 URL에 대한 후속의 요청을 자동으로 보낼 수 있도록 하기 위한 시간 값을 포함하는 응답 갱신 헤더를 형성하게 된다. 이 시간 값은 프로그래머(44) (도 1)에 의해 성립될 수 있으며 최종 응답을 형성하는 데에 걸리게 되는 시간양이 된다. 예를 들어, 이 요청에 대한 최종 응답이 일반적으로 생성하여 돌아가는데 10초 정도 걸리면, 응답 갱신 헤더의 시간 값은 11초 가 될 수 있다. 이로 인해 클라이언트(12)와 서버(14) 간의 연결이 최종 응답이 형성되고 있는 동안에 종료될 수가 있다.

응답 갱신 헤더는 임시 응답 시스템(62)에 의해 형성되는 임시 응답과 함께 출력 시스템(68)에 의해 클라이언트에 돌아가게 된다. 시간 값과 유사하게, 임시 응답은 프로그래머(44)에 의해 정의 가능하다. 예를 들어, 임시 응답은 "요청이 처리되고 있음"이라고 기재된 페이지일 수 있다. 최종 응답이 완료되면, 응답 캐싱 시스템(66)은 이를 응답 식별자에 따라 캐시(40)에 저장하므로, 쉽게 상호 참조될 수가 있다.

클라이언트(12)는 브라우저(18)에서 응답 갱신 헤더 및 임시 응답을 수신하게 된다. 응답 갱신 헤더의 시간 값의 만료 이후에, 브라우저(18)는 URL에 대한 후속의 요청을 자동으로 서버(14)에 보내게 된다. 구체적으로, HTTP에서는, 브라우저(18)가 본 발명의 응답 갱신 헤더와 같은 헤더를 취급 및 처리하도록 구성되어 유저(16) 측의 세심한 수동적인 동작 없이도 요청을 자동으로 형성하여 보낼 수가 있다. 따라서, 응답 갱신 헤더는 클라이언트(12) 또는 브라우저(18)의 변형 없이도 요청 프로세스가 비동기적이 되게 할 수 있다.

후속의 응답은 입력 시스템(50)에 의해 수신되게 된다. 이전의 요청과 유사하게, 개체 시스템(52)은 대응하는 세션 개체를 취득하게 된다. 이것이 후속의 응답이기 때문에, 세션 개체는 이미 존재한다. 따라서, 개체 시스템(52)은 이를 저장 유닛(32)으로부터 검색할 수 있다. 세션 개체를 취득한 후에, 응답 식별자는 응답 식별자 시스템(54)에 의해 재형성된다. 상기한 바와 같이, 응답 식별자는 세션 식별자와 요청된 URL의 해쉬를 포함한다. 응답 식별자를 이용하여, 응답 체크 시스템(56)은 URL이 이미 요청되었는지를 판정한다. 구체적으로, 요청 체크 시스템(56)은 세션 개체의 요청 테이블을 체크하게 된다. URL이 클라이언트(12)에 의해 이미 요청되었기 때문에, 응답 식별자는 요청 테이블에 목록화되어 있다. 따라서, 캐시 체크 시스템(58)은 캐시(40)를 체크하여 최종 응답이 완료되었는지를 판정한다. 구체적으로, 캐시 체크 시스템(58)은 응답 식별자를 이용하여 최종 응답을 체크한다. 최종 응답이 완료되면, 캐시 체크 시스템(58)에 의해 캐시(40)로부터 검색되어 브라우저(18)에서의 디스플레이를 위해 출력 시스템(68)을 거쳐 클라이언트(12)에 보내진다. 그러나, 최종 응답이 아직 완료되지 않았다면, 헤더 형성 시스템(60)은 시간 값을 갖는 새로운 응답 갱신 헤더를 형성하게 된다. 이 시간 값은 이전의 응답 갱신 헤더와 동일할 수도 있고, 새로운 다른 값일 수도 있다. 어쨌든, 새로운 응답 갱신 헤더가 새로운 임시 응답으로 출력 시스템(68)을 거쳐 클라이언트(12)에 보내진다. 새로운 응답 갱신 헤더에서의 시간 값의 만료 이후에, 클라이언트(12)는 URL에 대한 다른 요청을 제출하고, 이는 유사한 방식으로 처리되게 된다.

이하 도 3을 참조하여, 본 발명의 흐름도(100)를 나타낸다. 도시된 바와 같이, URL의 요청은 단계 S1에서 수신된다. 단계 S2에서, 세션 개체가 취득되고, 단계 S3에서 응답 식별자가 형성된다. 단계 S4에서, URL이 응답 식별자에 기초하여 클라이언트에 의해 이미 요청되었는지가 판정된다. URL이 이미 요청되지 않았다면, 최종 응답의 형성은 단계 S5에서 시작된다. 단계 S6에서, 명령은 응답 식별자에 따라서 최종 응답을 완료시 캐시에 저장하기 위해 응답 캐싱 시스템에 요청으로 전달된다. 최종 응답이 형성되고 있으면, 응답 갱신 헤더가 단계 S7에서 형성된다. 응답 갱신 헤더는 단계 S8에서 임시 응답과 함께 클라이언트에 보내진다.

응답 갱신 헤더의 시간 값의 만료 이후, 단계 S9에서 클라이언트는 URL에 대한 후속의 요청을 보낸다. 단계 S1에서의 수신시, 단계 S2-S4가 반복된다. 구체적으로, 세션 개체가 취득되고, 응답 식별자가 형성되고, URL이 클라이언트에 의해 이미 요청되었는지가 판정되게 된다. URL이 클라이언트에 의해 이미 요청되었기 때문에, 단계 S10에서 요청에 대한 최종 응답이 완료되었는지를 판정하기 위해 캐시가 체크된다. 완료되면, 단계 S11에서 최종 응답이 캐시로부터 검색되어 클라이언트에 보내진다. 최종 응답이 아직 완료되지 않은 경우, 단계 S7에서 새로운 응답 갱신 헤더가 형성되게 되어 단계 S8에서 새로운 임시 응답으로 클라이언트에 보내진다. 프로세스는 최종 응답이 클라이언트에 보내질 때까지 계속 반복될 수 있다.

본 발명이 하드웨어, 소프트웨어, 또는 하드웨어와 소프트웨어의 조합으로 실현될 수 있다는 것이 이해되어야 한다. 어느 종류의 컴퓨터/컴퓨터 시스템 - 또는 그 외 상술된 방법을 실현하는 데에 적합한 장치라도 적합하다. 통상의 하드웨어와 소프트웨어의 조합이 로딩되어 실행될 때 여기 개시된 각 방법을 실행하는 컴퓨터 프로그램을 갖는 범용의 컴퓨터 시스템일 수 있다. 다르게, 본 발명의 기능적 작업들 중 하나 이상을 실행하기 위한 특수 하드웨어를 포함하는 특정한 용도의 컴퓨터를 이용할 수 있다. 본 발명은 또한 컴퓨터 프로그램 제품으로 구체화될 수 있으며, 이는 여기 개시된 방법의 구현을 가능하게 하는 모든 각각의 특성을 포함하며, 또한 컴퓨터 시스템에 로딩될 때 이들 방법을 실행할 수 있다. 컴퓨터 프로그램, 소프트웨어 프로그램, 프로그램, 또는 소프트웨어는 본 컨텍스트에서 정보 처리 능력을 갖는 시스템이 특정 기능을 직접적으로 또는 (a) 다른 언어, 코드 또는 주해로의 변환; 및/또는 (b) 여러 재료 유형으로의 재형성 중 하나나 둘 다 이후에 실행할 수 있도록 하는, 언어, 코드 또는 주해로 된 명령의 세트의 표현을 의미한다.

본 발명의 바람직한 실시예의 다음 설명은 설명의 목적으로 제시된 것이다. 본 발명을 정확히 개시된 형태로 제한하고자 하는 것은 아니고 여러가지 수정 및 변형이 가능하다. 당업자에게는 명백하게 되는 이런 수정 및 변형은 첨부한 청구범위에 의해 정의되는 바와 같이 본 발명의 영역 내에 포함되는 것이다.

도면의 간단한 설명

본 발명은 첨부한 도면에서 나타난 바와 같이, 바람직한 실시예를 참조하여 오직 예시적으로만 이하 설명될 것이다.

도 1은 본 발명에 따라서 요청을 비동기식으로 처리하기 위한 시스템을 도시한다.

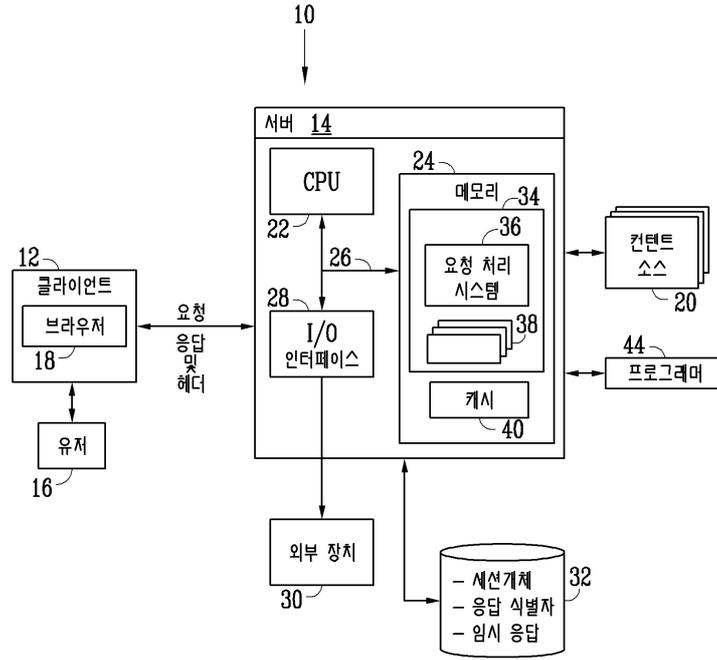
도 2는 도 1의 요청 처리 시스템을 더욱 상세히 나타낸다.

도 3은 본 발명에 따른 방법의 흐름도를 나타낸다.

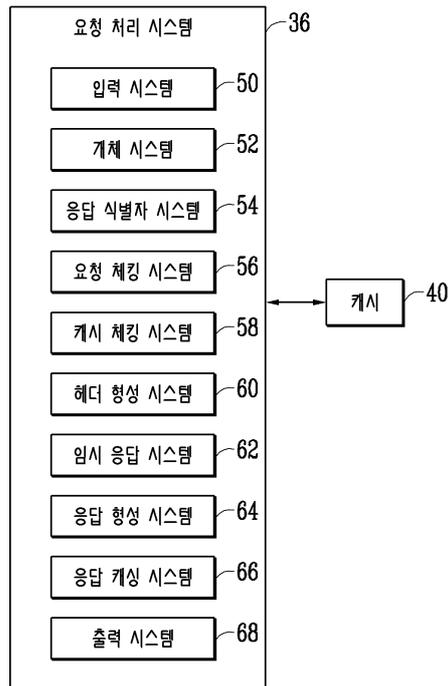
도면은 본 발명의 특정 파라미터를 표현하고자 하는 것이 아니고, 단순히 개략적으로 나타내고 있는 것이다. 또한 이 도면은 본 발명의 전형적인 실시예만을 나타내기 위한 것이지, 본 발명의 영역을 한정하고자 하는 것이 아니다. 도면에서 유사한 부호는 유사한 소자를 나타낸다.

도면

도면1



도면2



도면3

