

(19) 대한민국특허청(KR)
(12) 공개특허공보(A)

(51) Int. Cl.⁷
H04L 12/02

(11) 공개번호 특2000-0057718
(43) 공개일자 2000년09월25일

| | |
|------------|--|
| (21) 출원번호 | 10-2000-0000284 |
| (22) 출원일자 | 2000년01월05일 |
| (30) 우선권주장 | 9/231,081 1999년01월13일 미국(US) |
| (71) 출원인 | 인터내셔널 비지네스 머신즈 코포레이션 포만 제프리 엘 |
| (72) 발명자 | 미국 10504 뉴욕주 아몬크 맬킨피터케네스 미국뉴욕10502아드슬레이브램블브룩로드64 유틸립쉬-롱 |
| (74) 대리인 | 미국뉴욕10514참파과스토노웨이브18 원석희, 박해천 |

심사청구 : 있음

(54) 자각-개시 푸시를 제공하기 위한 방법 및 장치

요약

본 발명은 특정 클라이언트가 특정 상태로 들어가는 것에 근거하여 소스 서버나 또는 중간 프록시 서버로부터 자주 분리되는 클라이언트 컴퓨터로의 데이터 전송을 개시시키기 위한 상태 통지를 가능하게 한다. 중간 프록시 서버는 소스 서버로부터 타겟 클라이언트로의 푸시 요구를 큐잉한다. 상태 통지는 상이한 기준에 근거한 상태 정보를 포함한다. 상태 통지에는 소정의 데이터를 전송할 최소 비용 장치를 판단하기 위해 이용될 수 있는 푸시 정보가 명시될 수 있다. 이러한 상태 통지는 또한, 각각의 폴(poll)을 개시함으로써 네트워크 폴링 부하를 감소시킨다.

대표도

도1

색인어

데이터 푸시, 자각 서버, 중간 프록시, 상태 정보, 네트워크 폴링

명세서

도면의 간단한 설명

도1은 독립형 푸시 프록시를 포함하는 본 발명의 특징을 가진 네트워크를 도시한 도면.

도2는 자각 서버의 구성요소를 도시한 도면.

도3은 자각 요구 핸들러를 포함하는 자각 서버 로직의 흐름도.

도4는 자각 통지 핸들러의 흐름도.

도5는 푸시 프록시의 구성요소를 도시한 도면.

도6은 프록시형 푸시 요구 핸들러 및 프록시형 푸시 핸들러를 포함하는 푸시 프록시 로직의 흐름도.

도7은 서버측 푸시 프록시를 포함하는 본 발명의 특징을 가진 네트워크를 도시한 도면.

* 도면의 주요 부분에 대한 부호의 설명

1000:네트워크

1010:자각 서버

1020, 1030:PSS

1040:푸시 프록시

1050, 1060: 클라이언트

발명의 상세한 설명**발명의 목적****발명이 속하는 기술분야 및 그 분야의 종래기술**

본 발명은 컴퓨터 네트워크에서의 데이터 전송에 관한 것으로서, 특히, 의도된 수신자(intended recipients)의 특정 상태에 따라 데이터가 수신자에게 전송되는 교유의 데이터 푸시(push) 기술에 관한 것이다.

표준 클라이언트-서버 정보 푸시 모델은 대부분의 시간 동안 네트워크로부터 분리되어 있는 보급형 컴퓨팅(pervasive computing:PvC) 엔티티에는 효율적이지 못하다. 서버가 분리된 클라이언트로 데이터를 푸시하기 위해 헛되게 시도함으로써, CPU, RAM 및 네트워크 대역폭과 같은 자원을 낭비한다.

푸시(push)

HTTP 푸시와 같은 표준 푸시 모델에서, 소정의 푸시 소스 서버(PSS)는 새로운 정보를 모든 가입된 클라이언트로 전송한다. 이것은 클라이언트가 대부분의 시간 동안 네트워크에 능동적으로 접속되어 있고, 네트워크가 충분한 전송 대역폭을 갖고 있는 경우에 네트워크에서 양호하게 작용한다. 이러한 클라이언트는 그 전송된 것을 수신할 수 있다. 그러나, PvC 장치는 대부분 네트워크로부터 분리되어 있기 때문에, 대부분의 전송이 실패하게 되며, 따라서, 이 모델은 도움이 되지 않는다.

푸시 소스 서버는 각각의 전송 이전에 목표 클라이언트의 가용성(availability)을 검사하여 분리된 클라이언트로의 전송을 피하기 위해 시도할 수 있다. 이러한 방법은 여전히 매우 많은 상태 검사 프로브(status checking probe)로 인해 네트워크에 심한 부하를 초래한다.

풀(pull)

소정의 클라이언트 머신이 예를 들어, HTTP GET 요구를 이용하여 그것으로 전송되는 모든 정보를 요구하도록 강요된 경우에, 모든 데이터 전송은 이용가능한 클라이언트로 보내지도록 보장된다. 이러한 풀(pull) 방식에 있어서의 문제점은 소정의 소스 서버가 새로운 정보를 갖고 있는지 여부 또는 그 때를 소정의 클라이언트가 알도록 할 방법이 없기 때문에, 클라이언트는 모든 새로운 정보가 검색되도록 보장하기 위해서는 불필요한 많은 풀 요구를 해야한다. 이것도 역시 매우 많은 불필요한 풀 요구로 인해 네트워크에 심한 부하를 초래하게 된다.

풀(poll)

풀 기술과 유사한 또다른 방법으로는, 개인화된 뉴스 및 정보를 직접 클라이언트의 컴퓨터 스크린으로 방송하는 인터넷 뉴스 서비스인, 포인트캐스트 네트워크(PointCast Network)(<http://www.pointcast.com>)에 의해 이용되는 것과 같은 풀링 방법이 있다. 푸시 방법과 마찬가지로 풀링 방법에 있어서도, 모든 새로운 데이터가 검색되도록 보장하기 위해, 실제로 이용가능한 새로운 정보가 없는 경우조차도 지속적으로 소스 서버로부터 정보를 요구해야 하는 클라이언트에 의해 추가적인 네트워크 부하가 생성된다.

PvC 엔티티는 이미 대역폭 한계에 다다랐기 때문에, 정보, 특히, 버그 픽스(bug fixes) 또는 바이러스 체크 커 갱신과 같이, 중요하고 드물게 제공되는 것에 대한 정보의 자동적인 배달을 가능하게 하는 방법이 요구된다.

발명이 이루고자 하는 기술적 과제

전술한 요구에 따라, 본 발명의 목적은 네트워크 서버가 예를 들어, 네트워크로의 접속과 같은 클라이언트 엔티티의 특정 상태에 근거하여, 소스 서버로부터 특히, PvC 장치와 같이 자주 분리되는 클라이언트 엔티티로의 데이터의 전송을 개시할 수 있도록 하기 위한 방법 및 장치를 제공하는 것이다. 네트워크 서버(이후에 '자각 서버(awareness server)'로서 불리게 됨)는 사용자, 장치, 애플리케이션 등과 같은 네트워크 클라이언트 엔티티의 일시적 상태 정보(transient status information)에 대한 액세스를 제공한다. 상태 정보는 소정의 장치가 네트워크에 접속되어 있는지, 소정의 이용자가 로그인되어 있는지, 그리고 소정의 이용자가 소정의 애플리케이션에 능동적으로 참여하고 있는지를 명시한다. 자각 서버가 데이터 전송을 개시하는 프로세스는 다음의 단계를 포함한다. 즉,

소스 서버가 특정 클라이언트 엔티티가 특정 상태에 있게 되는 때를 통지하도록 자각 서버에게 요구하는 단계;

상기 클라이언트 엔티티가 특정 상태에 있거나 또는 특정 상태로 들어가는 단계;

자각 서버가 특정 클라이언트 엔티티의 상태를 학습한 이후에 상기 요구한 서버로 통지를 전송하는 단계; 및

상기 소스 서버가 통지를 수신하고 특정 클라이언트 엔티티로 데이터를 전송하는 단계를 포함한다.

또한, 각각 서버는 소스 서버로부터 타겟 클라이언트 엔티티로의 푸시 요구가 큐잉되어 있는 중간 프록시 서버로부터 데이터 전송을 개시할 수 있다. 상이한 기준에 근거하여 각각 통지가 발생될 수 있도록 하는 다른 방법도 제공된다. 상이한 기준은 다음의 기준을 포함한다.

- * 소정의 클라이언트 엔티티가 접속되었는지 여부 및 접속 방식. 예를 들어, 타겟 클라이언트 엔티티는 14.4 Kbps 또는 그 이상의 속도로 링크된 TCP/IP가 된다.
- * 소정의 클라이언트 엔티티가 예를 들어, 웹 브라우저와 같은 특정 애플리케이션을 실행하고 있는지 여부 및 그 실행 방식.
- * 예를 들어, 타겟 클라이언트 엔티티가 SSL을 이용하여 링크될 때, 소정의 클라이언트 엔티티에 대한 접속이 안전한지 여부 및 접속 방식.
- * 소정의 이용자가 현재 네트워크에 접속되어 있는지 여부. 통지에서 이용자의 현재 장치의 네트워크 어드레스를 전송함.
- * 예를 들어, 타겟 클라이언트 엔티티가 특정 이용자에 의해 이용되고 있을 때, 누가 소정의 클라이언트 엔티티를 이용하고 있는지에 관한 정보.

또한, 다음의 단계를 수행함으로써 각각 서버가 상태 정보를 얻기 위한 방법이 제공된다.

요구한 상태 및 상태 변화를 능동적으로 모니터링하는 단계;

타겟 클라이언트 엔티티 자체로부터 상태 또는 상태 변화를 학습하는 단계; 및

예를 들어, 소정의 클라이언트가 네트워크에 접속될 때 인터넷 액세스 공급자로부터 학습하는 것과 같이, 다른 서버로부터 상태 또는 상태 변화를 학습하는 단계.

또한, 다음의 단계를 수행하는 다른 방법이 제공된다.

예를 들어, 소정의 푸시를 전송할 애플리케이션의 포트 넘버와 같은 푸시에 필요한 정보를 각각 통지에 명시하는 단계;

각각 서버를 이용하여 소정의 데이터를 전송할 최소 비용 클라이언트 장치를 판단하는 단계; 및

각각 통지에 의해 각각의 폴(poll)을 개시함으로써, 네트워크 SNMP-기반 관리 서버에 의해 수행되는 바와 같은 네트워크 상의 폴링의 부하를 감소시키는 단계.

발명의 구성 및 작용

본 발명의 기술적 목적 및 장점은 첨부도면과 함께 하는 바람직한 실시예에 관한 다음의 상세한 설명을 참조함으로써 이 기술분야에 통상의 지식을 가진 자에게 보다 쉽게 이해될 것이다. 도면에서, 유사한 구성요소들은 동일한 참조번호로 표시되어 있다.

바람직한 실시예에서, 본 발명은 HTTP 푸시 전송을 제공하기 위한 방법 및 장치와 관련된 것이다. 이 기술분야에 통상의 지식을 가진 자는 e-메일 및 멀티미디어 방송을 포함하여 다른 형태의 전송도 역시 유사하게 처리될 수 있다는 것을 이해할 것이다.

도1은 전체적인 논리적 네트워크 토폴로지를 도시하고 있다. PSS(Push Source Server)(1020,1030), 각각 서버(1010) 및 푸시 프록시 서버(1040)는 네트워크(1000)에 직접 접속된다. 이러한 네트워크(1000)의 예로는 인터넷, 월드 와이드 웹, 인트라넷 및 근거리 통신망(LAN) 등이 있으며, 이들에 제한되는 것은 아니다.

클라이언트 장치(1050,1060)는 푸시 프록시 서버(1040)에 지역적으로 접속된다. 클라이언트(1050,1060)와 네트워크(1000) 사이의 모든 통신은 푸시 프록시 서버(1040)를 통해 전달된다. 비록 2개의 PSS(1020,1030)만이 도시되어 있지만, 본 발명은 어떠한 수의 서버에도 적용될 수 있다는 것을 이해할 것이다. 마찬가지로, 본 발명은 어떠한 수의 클라이언트에도 적용될 수 있다.

PSS(1020,1030)는 IBM사의 인터넷 접속 서버(ICS)를 포함하여, 기본적으로 표준 HTTP 서버이다. PSS(1020,1030)의 유일한 특징은 이들이 특정 푸시 프록시 서버(1040)를 통해 모든 푸시를 라우팅한다는 것이다. 이러한 특징은 ICS 서버의 동작을 개별화(customize)하는 모듈을 개발하고 포함시키는 것을 가능하게 하는 ICS의 API(ICSAPI)를 이용한 ICS-기반 서버에 부가될 수 있다. ICS 서버는 요구를 처리하는 동안에 특정 포인트에 있는 모듈에 대한 제어를 핸드-오프(hand-off)하도록 구성될 수 있다. 모듈은 그 태스크를 수행하고, 다음에 전송을 완료하기 위해 제어를 서버로 복귀시킬 수 있으며, 또는 그 자체적으로 전송을 완료하고 그것이 수행되면 서버로 통지할 수도 있다.

PSS(1020,1030)의 경우에, 부가된 모듈은 데이터를 클라이언트(1050,1060)로 전송하는 것이 아니라 데이터가 가입된 클라이언트(1050,1060)로 최종적으로 푸시되기 전에 제어를 행하고, 푸시 프록시(1040)에 접속하고, 데이터가 전송되어야 하는 장소를 명시한다. 이러한 프록시 라우팅은 HTTPD RFC에 명시된 HTTP

표준의 일부이다. 자각 서버(1010)와 PSS(1020, 1030) 및 푸시 프록시 서버(1040)는 IBM S/390 SYSPLEX, SP2, 또는 RS6000 워크스테이션과 같은 제품을 포함하여 어떠한 컴퓨팅 노드에서도 구현될 수 있으며, 이들에 제한되는 것은 아니다.

클라이언트(1050, 1060)는 IBM WorkPad, PC, 다양한 워크스테이션, 셋탑 박스 등을 포함할 수 있다. 클라이언트(1050, 1060) 상에서 실행되는 소프트웨어는 넷스케이프 네비게이터 웹 브라우저를 포함할 수 있으며, 이것에 제한되는 것은 아니다.

자각 서버(1010)는 특정 클라이언트 또는 타겟이 예를 들어, 네트워크에 접속된 상태와 같은 원하는 상태에 도달했을 때의 통지를 제공한다. 도2에 도시된 바와 같이, 자각 서버(1010)(도1)는 CPU(3000), RAM과 같은 메모리(3020), 및 디스크 또는 다른 직접 액세스 저장 장치(DASD)와 같은 저장 장치(3010)를 포함한다.

바람직하게는 실행가능한 컴퓨터 코드로 구현되는 자각 서버(1010)(도1) 로직은 CPU(3000)에 의한 실행을 위해, 네트워크를 통해 원격으로, 또는 CD-ROM이나, 디스크와 같은 자기 저장장치 또는 DASD(3010)로부터 국부적으로 메모리(3020)로 로딩된다. 메모리(3020)로 로딩된 실행가능한 컴퓨터 코드는 바람직하게는, 클라이언트의 요구를 수신하고 그 요구에 서비스하기 위한 자각 요구 핸들러(3030), 이들 요구의 상태를 클라이언트로 통지하기 위한 자각 통지 핸들러(3040) 및 현재의 요구를 저장하기 위한 요구 버퍼(3050)를 포함한다.

도3에 도시된 바와 같이, 자각 서버(1010)(도1)는 단계(4000)에서, 입력, 즉, 네트워크로부터 수신되는 상태 통지를 위한 요구를 포함하여 가능한 입력을 대기한다. 입력이 수신된 이후에, 단계(4010)에서는 그 입력이 자각 요구(awareness request)인지, 즉, 특정 클라이언트가 특정 상태에 도달했다는 것을 통지하기 위한 요구인지 판단한다. 만일 그 요구가 상태 통지를 위한 요구가 아니면, 단계(4030)에서 각종 핸들러가 호출된다. 단계(4030)에서의 각종 핸들러의 실행 이후에, 프로그램 제어는 또다른 입력을 대기하기 위해 단계(4000)로 복귀한다.

그러나, 만일 입력이 자각 통지 요구라면, 자각 요구 핸들러(4020)가 호출된다. 단계(5010)에서, 요구 핸들러는 그 요구로부터 요구자(requestor) 및 타겟 클라이언트의 ID, 예를 들어, 통지받을 것을 요구한 서버의 호스트명(hostname) 및 그 상태가 알려져 있는 클라이언트 장치의 호스트명을 판독한다. 단계(5020)에서, 자각 요구 핸들러(4020)는 요구자 및 타겟 클라이언트의 추출된 ID들을 나타내는 새로운 엔트리를 요구 버퍼(3050)(도2)에 추가한다. 자각 요구 핸들러(4020)의 실행 이후에, 프로그램 제어는 또다른 입력을 대기하기 위해 단계(4000)로 복귀한다.

도4는 자각 서버(1010)의 CPU(3000)에서 백그라운드 프로세스(background process)로서 지속적으로 실행되고 있는 자각 통지 핸들러(3040)의 처리를 도시하고 있다.

단계(6000)에서, 요구 버퍼(3050)내의 제1 엔트리로 로컬 포인터가 이동된다. 단계(6010)에서는, 요구자 및 클라이언트의 ID들이 지시된 엔트리로부터 판독된다. 다음에, 단계(6020)에서 타겟 클라이언트의 상태가 검사된다.

상태는 접속 상태가 될 수 있으며, PING과 같은 프로그램에 의해 검사되거나 또는 소정의 클라이언트 장치가 접속할 때마다 타겟 클라이언트 장치로부터 수신되는 통지로부터 검사될 수 있다. 특정 애플리케이션에 참여하거나 또는 활성적으로 접속된 소정의 타겟 클라이언트의 상태를 확인하는 것은 애플리케이션 서버로부터 타겟 클라이언트의 상태를 요구함으로써 달성될 수 있다. 자각 서버(1010)는 그 자신이 외부 서버에 의해 통지받도록 요구할 수 있다. 예를 들어, 자각 서버는 소정의 클라이언트가 접속할 때 타겟 클라이언트의 인터넷 액세스 공급자에게 통지할 것을 요구함으로써 소정의 타겟 클라이언트가 접속하는 때를 학습할 수도 있다. 만일 통지가 수신되면, 타겟 클라이언트는 접속된 것이고, 만일 그렇지 않으면 클라이언트는 여전히 분리되어 있는 것이다.

단계(6030)에서는, 타겟 클라이언트가 원하는 상태에 도달했는지 여부에 관한 판단이 이루어진다. 만일 타겟 클라이언트가 원하는 상태에 도달했다면, 단계(6040)에서 TCP, UCP 또는 e-메일과 같은 다른 네트워크 트랜스포트 프로토콜 또는 그 상위 레벨의 애플리케이션을 이용하여 요구자에게 통지가 전송된다. 다음에는, 단계(6050)에서 요구 버퍼(3050)로부터 현재 엔트리가 삭제된다. 그러나, 만일 타겟 클라이언트가 필요한 상태에 도달하지 못했다면, 이들 단계(6040, 6050)는 스킵된다.

단계(6060)에서, 만일 요구 버퍼(3050)에 더 이상의 요구가 없는 것으로 판단이 이루어지면, 처리는 단계(6000)에서 계속된다. 만일 요구 버퍼(3050)에 다른 요구가 있다면, 단계(6070)에서 로컬 포인터는 다음 엔트리로 나아가고, 처리는 단계(6010)에서 계속된다.

자각 서버(1010)(도1)는 여러 가지 이유로 통지를 전송할 수 있다. 이들 이유중 일부는 다음 사항을 포함할 수 있다.

- * 소정의 클라이언트 엔티티가 접속되었는지 여부 또는 접속 방식. 예를 들어, 타겟 클라이언트 엔티티는 14.4 Kbps 또는 그 이상의 속도로 링크된 TCP/IP가 된다.
- * 예를 들어, 타겟 클라이언트 엔티티가 웹 브라우저를 실행하고 있는 경우에 소정의 클라이언트 엔티티가 특정 애플리케이션을 실행하고 있는지 여부 또는 그 애플리케이션의 실행 방식.
- * 소정의 클라이언트 엔티티에 대한 접속이 안전한지 여부 및 접속 방식. 예를 들어, 클라이언트 엔티티는 SSL을 이용하여 링크됨.
- * 누가 소정의 클라이언트 엔티티를 이용하고 있는가, 예를 들어, 특정 이용자가 클라이언트임.

또한, 자각 서버(1010)는 통지에서 웹 브라우저와 같은 타겟 클라이언트 엔티티의 푸시 수신자(recipient) 애플리케이션이 실행되고 있는 포트 번호와 같은 임계(critical) 정보를 전송할 수도 있다. 이러한 임계 정보는 통지가 수신될 때 푸시 프록시 서버(1040)(도1)에 의해 판독되어 이용될 수 있다.

도5에 도시된 푸시 프록시 서버(1040)는 푸시 요구를 큐잉하고 있다가, 소정의 클라이언트 장치가 전송

신호를 수신할 수 있다는 것을 자각 서버가 통지한 이후에 타겟 클라이언트 장치로 최종 전송을 실행한다. 푸시 프록시 서버(1040)(도1)는 CPU(10000), RAM과 같은 메모리(10020) 및 디스크나 DASD와 같은 저장 장치(10010)를 포함하고 있다.

본 발명에 따르면, 실행가능한 컴퓨터 코드로 구현된 푸시 프록시 로직은 CPU(10000)에 의한 실행을 위해, 네트워크를 통해 원격으로, 또는 광학 CD-ROM이나, 디스크와 같은 자기 저장장치 또는 DASD(10010)로부터 국부적으로 메모리(10020)로 로딩된다. 메모리(10020)로 로딩된 컴퓨터 코드는 프록시형(proxyed) 푸시 요구 핸들러(10030), 프록시형 푸시 핸들러(10040) 및 IBM사의 sDB2 데이터베이스 제품과 같은 프록시형 푸시 데이터베이스(10050)를 포함한다.

도6은 푸시 프록시 프로그램 흐름을 도시하고 있다. 도6에 도시된 바와 같이, 푸시 프록시는 단계(1100)에서 입력을 대기한다. 입력이 수신되면, 단계(1101)에서 그 입력이 푸시 요구인지 여부에 관한 판단이 이루어진다. 만일 푸시요구이면, 프록시형 푸시 요구 핸들러(11020)가 호출된다. 만일 그렇지 않으면, 처리는 단계(11030)에서 계속되어, 그 입력이 자각 통지인지 여부에 관한 판단이 이루어진다. 만일 입력이 자각 통지이면, 프록시형 푸시 핸들러(11040)가 호출된다. 만일 그렇지 않고 입력이 자각 통지가 아니면, 적절한 각종 핸들러가 호출되며, 그 이후에 처리는 단계(11000)에서 계속된다.

프록시형 푸시 요구 핸들러(11020)는 PSS(1020, 1030)로부터의 푸시 요구를 큐잉한다. 단계(12000)에서는 데이터 및 클라이언트의 호스트명 또는 IP 어드레스와 같은 타겟 클라이언트의 ID가 요구로부터 판독된다. 만일 단계(12010)에서 소정의 ID가 프록시형 푸시 데이터베이스(10050)(도5)에 이미 엔트리를 갖고 있지 않은 것으로 판단이 이루어지면, 단계(12020)에서 엔트리가 생성된다.

단계(12030)에서는, 판독된 ID를 위해 큐잉된 데이터에 새로운 데이터가 추가된다. 단계(12040)에서는, 자각 서버(1010)(도1)로 그 자체 및 타겟 클라이언트에 대한 ID를 명시하는 통지 요구가 전송되고, 그 이후에 처리는 단계(11000)에서 계속된다.

전송한 바와 같이, 특정 클라이언트 장치를 위한 자각 서버(1010)(도1)로부터 통지를 수신한 이후에, 프록시형 푸시 핸들러(11040)는 큐잉된 데이터를 그 특정 클라이언트 장치로 푸시한다.

특히, 단계(13000)에서 프록시형 푸시 핸들러는 이 통지로부터 클라이언트의 ID를 판독하고, 단계(13010)에서 프록시형 푸시 데이터베이스(10050)(도5)로부터 소정의 ID에 대해 큐잉된 데이터를 검색한다. 다음에, 단계(13020)에서 이 데이터는 타겟 클라이언트로 푸시된다. 단계(13030)에서는, 소정의 ID에 대한 프록시형 푸시 데이터베이스 엔트리가 검출되고, 처리는 단계(11000)에서 계속된다.

타겟 클라이언트가 네트워크에 접속되어 청취하고 있다는 것을 통지받은 이후에만 큐잉된 푸시를 전송함으로써, 이러한 지식이 없는 경우에는 실행될 수도 있었던 많은 불필요한 전송 시도를 없앨 수 있다. 본 발명의 이러한 관점은 또한, 예를 들어, SNMP-기반 네트워크 관리 애플리케이션에 의해 수행되는 상태 폴링(status polling)과 같은 폴링 애플리케이션을 최적화시키기 위해 이용될 수 있다. 만일 자각 서버로부터의 통지에 의해 나타나는 바와 같이 타겟 네트워크 장치가 접속된 이후에만 상태 요구가 전송되는 경우에는, 분리된 장치에 대한 자원-대기 요구를 없앨 수 있다.

도7에 도시된 바와 같이, 푸시 프록시 서버(1040)(도1)는 독립형 머신이 될 필요가 없다. 원격으로 실행되는 푸시 프록시를 구비하는 대신에, 각각의 PSS(2020, 2030)는 PSS 시스템과 병렬로 동시에 실행되는 그 고유의 서버측 프록시를 구비할 수도 있다. 이러한 구성에 있어서, 소정의 PSS는 그 고유의 호스트명 또는 IP 어드레스를 푸시 프록시의 호스트명 또는 IP 어드레스로서 이용하게 된다.

처리 및 네트워크 부하를 분산시키기 위해 복수의 푸시 프록시가 사용될 수도 있다. 복수의 푸시 프록시를 이용하는 경우에, 각각의 PSS는 어느 것을 이용할 것인지 선택하게 된다. 처리 및 네트워크 부하를 분산시키는 다른 방식은 복수의 자각 서버를 사용함으로써, 푸시 프록시가 하나 이상의 자각 서버로부터 통지를 요구할 수 있도록 하는 것이다.

자각 서버(2010)로 전송된 타겟 클라이언트의 ID는 192.62.1.1 등의 네트워크 어드레스와 같은 장치 ID가 아니라 'LuandChen' 등의 이름과 같은 사용자 ID가 될 수 있다. 사용자 ID가 주어지면, 자각 서버는 그 이용자의 현재 장치의 어드레스를 통지에 명시함으로써 특정 이용자가 접속될 때 푸시 프록시에 통지할 수 있다.

또한, 자각 서버 통지는 가능한 한 비용이 절감되는 방식으로 전송을 라우팅하기 위해 이용될 수 있는데, 예를 들어, 특정 이용자에 대한 자각 통지 요구에 응답하여, 자각 서버(2010)는 현재 이용가능한 모든 이용자 장치를 판단하고, 데이터를 전송하기 위한 최소 비용의 장치를 판단하고, 마지막으로 그 장치의 어드레스를 명시하는 통지를 함으로써 통지 요구에 응답할 수 있다.

비록 본 발명이 그 바람직한 실시예에 관해 특정하여 도시 및 설명되었지만, 이 기술분야에 통상의 지식을 가진 자는 첨부된 청구범위에 의해서만 제한되어야 하는 본 발명의 사상 및 범위를 벗어나지 않고 그 형태 및 세부사항에서 다른 변경이 이루어질 수도 있다는 것을 이해할 것이다.

발명의 효과

전송한 바와 같은 본 발명의 데이터 푸시 기술에 따르면, 서버가 자주 분리되는 클라이언트로 데이터를 푸시하기 위해 불필요하게 시도하는 것을 방지함으로써, CPU, RAM 및 네트워크 대역폭과 같은 자원의 낭비를 줄일 수 있는 효과가 있다.

(57) 청구의 범위

청구항 1

네트워크내의 적어도 하나의 소스 엔티티(source entity)로부터 상기 네트워크내의 적어도 하나의 클라이언트 엔티티 - 상기 클라이언트 엔티티는 특정 동작 상태를 가짐 - 로 데이터를 전송하기 위한 방법에 있어서,

상기 클라이언트 엔티티가 상기 특정 동작 상태에 있게 될 때까지 상기 클라이언트 엔티티의 상태를 반복적으로 판단하는 단계; 및

통지를 전송함으로써 상기 상태를 상기 소스 엔티티로 통신하는 단계 - 여기서, 상기 소스 엔티티는 상기 통지에 응답하여 상기 클라이언트 엔티티로 데이터를 전송함 -

를 포함하는 데이터 전송 방법.

청구항 2

제 1 항에 있어서,

상기 소스 엔티티는 소스 서버이고, 상기 소스 서버는 상기 통지를 요구하는

데이터 전송 방법.

청구항 3

제 2 항에 있어서,

상기 소스 엔티티는 중간 프록시(intermediary proxy)를 포함하고, 상기 중간 프록시는 소스 서버로부터 푸시 요구를 수신하고 상기 통지를 요구하는

데이터 전송 방법.

청구항 4

제 3 항에 있어서,

상기 통지에 명시된 정보를 이용하여 데이터가 전송되는

데이터 전송 방법.

청구항 5

제 4 항에 있어서,

상기 특정 동작 상태는 상기 클라이언트 엔티티가 상기 네트워크에 접속되어 있는 상태를 포함하는

데이터 전송 방법.

청구항 6

제 5 항에 있어서,

상기 특정 동작 상태는 상기 클라이언트 엔티티가 상기 네트워크에 접속된 접속 형태를 포함하는

데이터 전송 방법.

청구항 7

제 4 항에 있어서,

상기 특정 동작 상태는 특정 클라이언트 엔티티가 상기 네트워크에 접속되어 있는 상태를 포함하는

데이터 전송 방법.

청구항 8

제 7 항에 있어서,

상기 특정 동작 상태는 상기 특정 클라이언트 엔티티가 상기 네트워크에 접속된 접속 형태를 포함하는

데이터 전송 방법.

청구항 9

제 4 항에 있어서,

상기 특정 동작 상태는 특정 이용자가 상기 네트워크에 접속되어 있는 상태를 포함하고, 상기 통지에서 상기 특정 이용자의 현재 장치의 어드레스가 전송되는

데이터 전송 방법.

청구항 10

제 9 항에 있어서,

상기 특정 동작 상태는 상기 특정 이용자가 특정 클라이언트 엔티티를 이용하고 있는 상태를 포함하는 데이터 전송 방법.

청구항 11

제 4 항에 있어서,

상기 특정 동작 상태는 상기 클라이언트 엔티티가 특정 애플리케이션을 실행하고 있는 상태를 포함하는 데이터 전송 방법.

청구항 12

제 11 항에 있어서,

상기 특정 동작 상태는 상기 클라이언트가 상기 특정 애플리케이션을 실행하고 있는 방식을 포함하는 데이터 전송 방법.

청구항 13

제 4 항에 있어서,

상기 특정 동작 상태는 상기 클라이언트 엔티티를 능동적으로 모니터링함으로써 판단되는 데이터 전송 방법.

청구항 14

제 13 항에 있어서,

상기 클라이언트 엔티티의 능동적 모니터링은 상기 네트워크 상에서의 상기 클라이언트 엔티티의 폴링 (polling)에 의해 수행되는

데이터 전송 방법.

청구항 15

제 4 항에 있어서,

상기 특정 동작 상태는 상기 클라이언트 엔티티로부터의 통신에 의해 판단되는 데이터 전송 방법.

청구항 16

제 4 항에 있어서,

상기 특정 동작 상태는 상기 네트워크내의 하나 또는 그 이상의 엔티티로부터의 통신에 의해 판단되는 데이터 전송 방법.

청구항 17

제 4 항에 있어서,

상기 특정 동작 상태에 근거하여 상기 데이터를 상기 클라이언트 엔티티로 전송하기 위한 최소 비용 방법을 판단하는 단계

를 더 포함하는 데이터 전송 방법.

청구항 18

네트워크내의 적어도 하나의 소스 엔티티(source entity)로부터 상기 네트워크내의 적어도 하나의 클라이언트 엔티티 - 상기 클라이언트 엔티티는 특정 동작 상태를 가짐 - 로 데이터를 전송하기 위한 장치에 있어서,

상기 클라이언트 엔티티가 상기 특정 동작 상태에 있게 될 때까지 상기 클라이언트 엔티티의 상태를 반복적으로 판단하기 위한 수단; 및

통지를 전송함으로써 상기 상태를 상기 소스 엔티티로 통신하기 위한 수단 - 여기서, 상기 소스 엔티티는 상기 통지에 응답하여 상기 클라이언트 엔티티로 데이터를 전송함 -

을 포함하는 데이터 전송 장치.

청구항 19

제 18 항에 있어서,

상기 소스 엔티티는 소스 서버이고, 상기 소스 서버는 상기 통지를 요구하는

데이터 전송 장치.

청구항 20

제 19 항에 있어서,

상기 소스 엔티티는 중간 프록시(intermediary proxy)를 포함하고, 상기 중간 프록시는 소스 서버로부터 푸시 요구를 수신하고 상기 통지를 요구하는

데이터 전송 장치.

청구항 21

네트워크내의 적어도 하나의 소스 엔티티(source entity)로부터 상기 네트워크내의 적어도 하나의 클라이언트 엔티티 - 상기 클라이언트 엔티티는 특정 동작 상태를 가짐 - 로 데이터를 전송하기 위한 방법의 단계들을 수행하기 위해 머신에 의해 실행가능한 명령어 프로그램을 실제로 구현하는 머신 판독가능 컴퓨터 프로그램 장치에 있어서,

상기 방법은,

상기 클라이언트 엔티티가 상기 특정 동작 상태에 있게 될 때까지 상기 클라이언트 엔티티의 상태를 반복적으로 판단하는 단계; 및

통지를 전송함으로써 상기 상태를 상기 소스 엔티티로 통신하는 단계 - 여기서, 상기 소스 엔티티는 상기 통지에 응답하여 상기 클라이언트 엔티티로 데이터를 전송함 -

를 포함하는 컴퓨터 프로그램 장치.

청구항 22

제 21 항에 있어서,

상기 소스 엔티티는 소스 서버이고, 상기 소스 서버는 상기 통지를 요구하는

컴퓨터 프로그램 장치.

청구항 23

제 22 항에 있어서,

상기 소스 엔티티는 중간 프록시(intermediary proxy)를 포함하고, 상기 중간 프록시는 소스 서버로부터 푸시 요구를 수신하고 상기 통지를 요구하는

컴퓨터 프로그램 장치.

청구항 24

제 23 항에 있어서,

상기 통지에 명시된 정보를 이용하여 데이터가 전송되는
컴퓨터 프로그램 장치.

청구항 25

제 24 항에 있어서,
상기 특정 동작 상태는 상기 클라이언트 엔티티가 상기 네트워크에 접속되어 있는 상태를 포함하는
컴퓨터 프로그램 장치.

청구항 26

제 25 항에 있어서,
상기 특정 동작 상태는 상기 클라이언트 엔티티가 상기 네트워크에 접속된 접속 형태를 포함하는
컴퓨터 프로그램 장치.

청구항 27

제 24 항에 있어서,
상기 특정 동작 상태는 특정 클라이언트 엔티티가 상기 네트워크에 접속되어 있는 상태를 포함하는
컴퓨터 프로그램 장치.

청구항 28

제 27 항에 있어서,
상기 특정 동작 상태는 상기 특정 클라이언트 엔티티가 상기 네트워크에 접속된 접속 형태를 포함하는
컴퓨터 프로그램 장치.

청구항 29

제 24 항에 있어서,
상기 특정 동작 상태는 특정 이용자가 상기 네트워크에 접속되어 있는 상태를 포함하고, 상기 통지에서
상기 특정 이용자의 현재 장치의 어드레스가 전송되는
컴퓨터 프로그램 장치.

청구항 30

제 29 항에 있어서,
상기 특정 동작 상태는 상기 특정 이용자가 특정 클라이언트 엔티티를 이용하고 있는 상태를 포함하는
컴퓨터 프로그램 장치.

청구항 31

제 24 항에 있어서,
상기 특정 동작 상태는 상기 클라이언트 엔티티가 특정 애플리케이션을 실행하고 있는 상태를 포함하는
컴퓨터 프로그램 장치.

청구항 32

제 31 항에 있어서,
상기 특정 동작 상태는 상기 클라이언트가 상기 특정 애플리케이션을 실행하고 있는 방식을 포함하는
컴퓨터 프로그램 장치.

청구항 33

제 24 항에 있어서,
 상기 특정 동작 상태는 상기 클라이언트 엔티티를 능동적으로 모니터링함으로써 판단되는
 컴퓨터 프로그램 장치.

청구항 34

제 33 항에 있어서,
 상기 클라이언트 엔티티의 능동적 모니터링은 상기 네트워크 상에서의 상기 클라이언트 엔티티의 폴링
 (polling)에 의해 수행되는
 컴퓨터 프로그램 장치.

청구항 35

제 24 항에 있어서,
 상기 특정 동작 상태는 상기 클라이언트 엔티티로부터의 통신에 의해 판단되는
 컴퓨터 프로그램 장치.

청구항 36

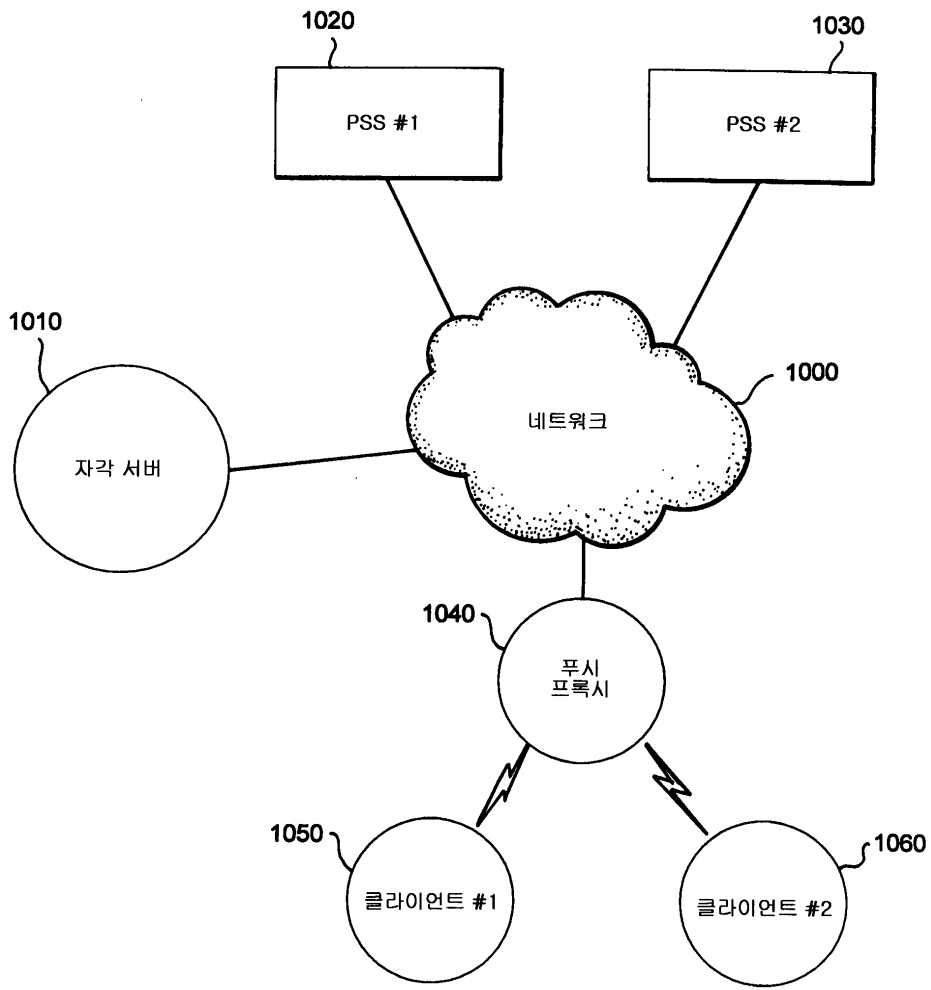
제 24 항에 있어서,
 상기 특정 동작 상태는 상기 네트워크내의 하나 또는 그 이상의 엔티티로부터의 통신에 의해 판단되는
 컴퓨터 프로그램 장치.

청구항 37

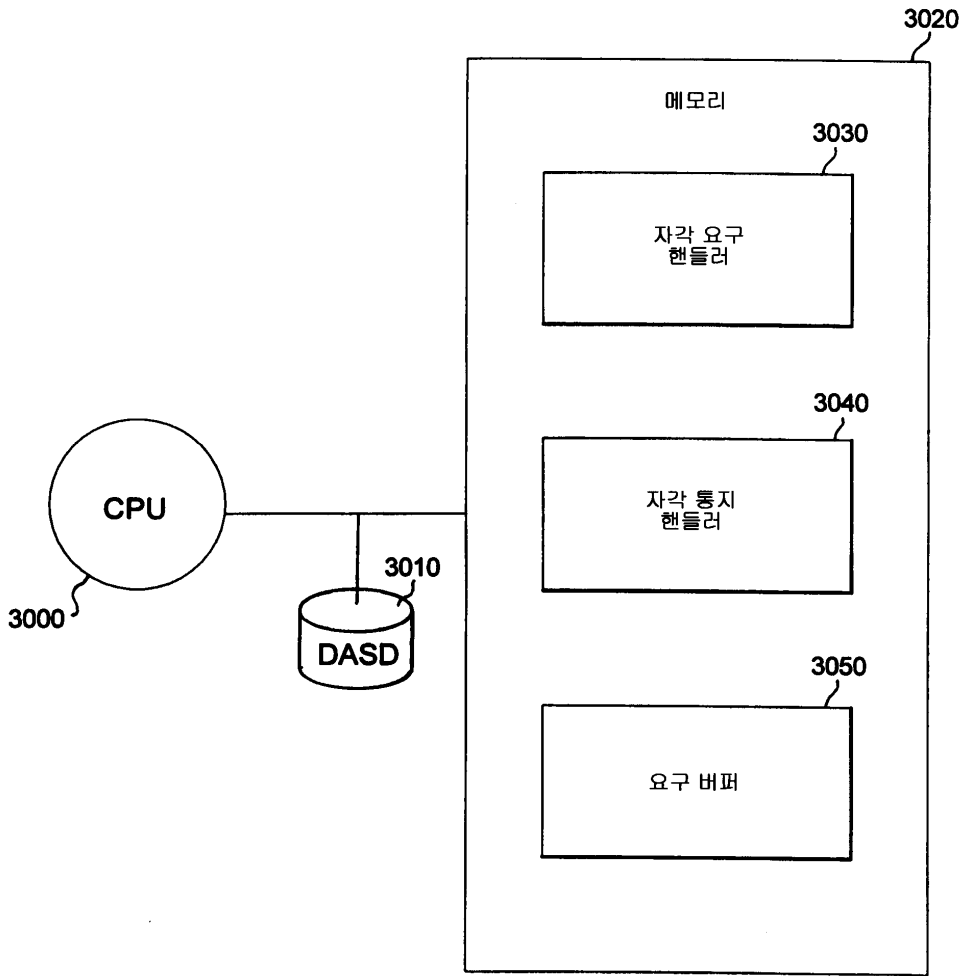
제 24 항에 있어서,
 상기 특정 동작 상태에 근거하여 상기 데이터를 상기 클라이언트 엔티티로 전송하기 위한 최소 비용 방
 법을 판단하는 단계
 를 더 포함하는 컴퓨터 프로그램 장치.

도면

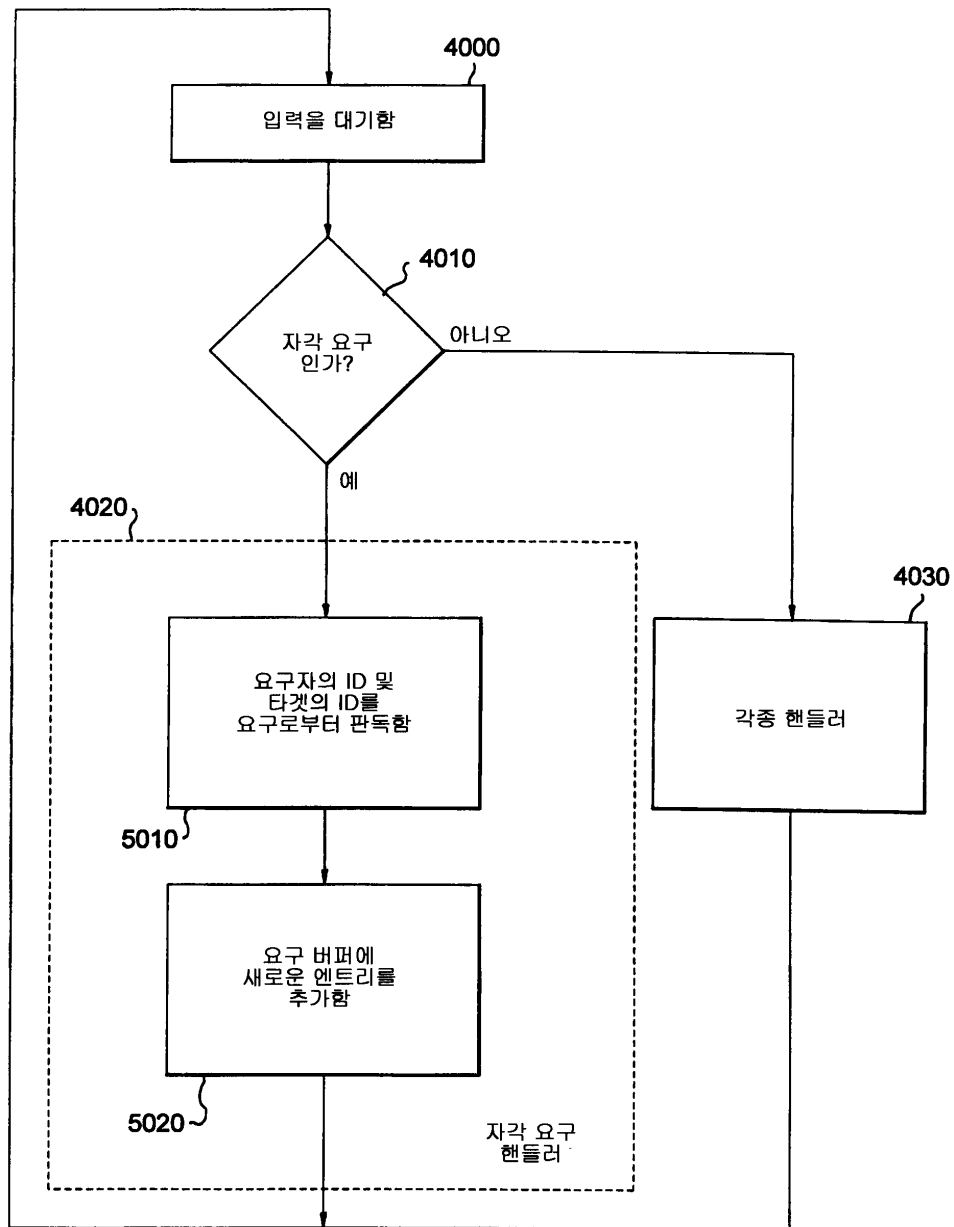
도면1



도면2

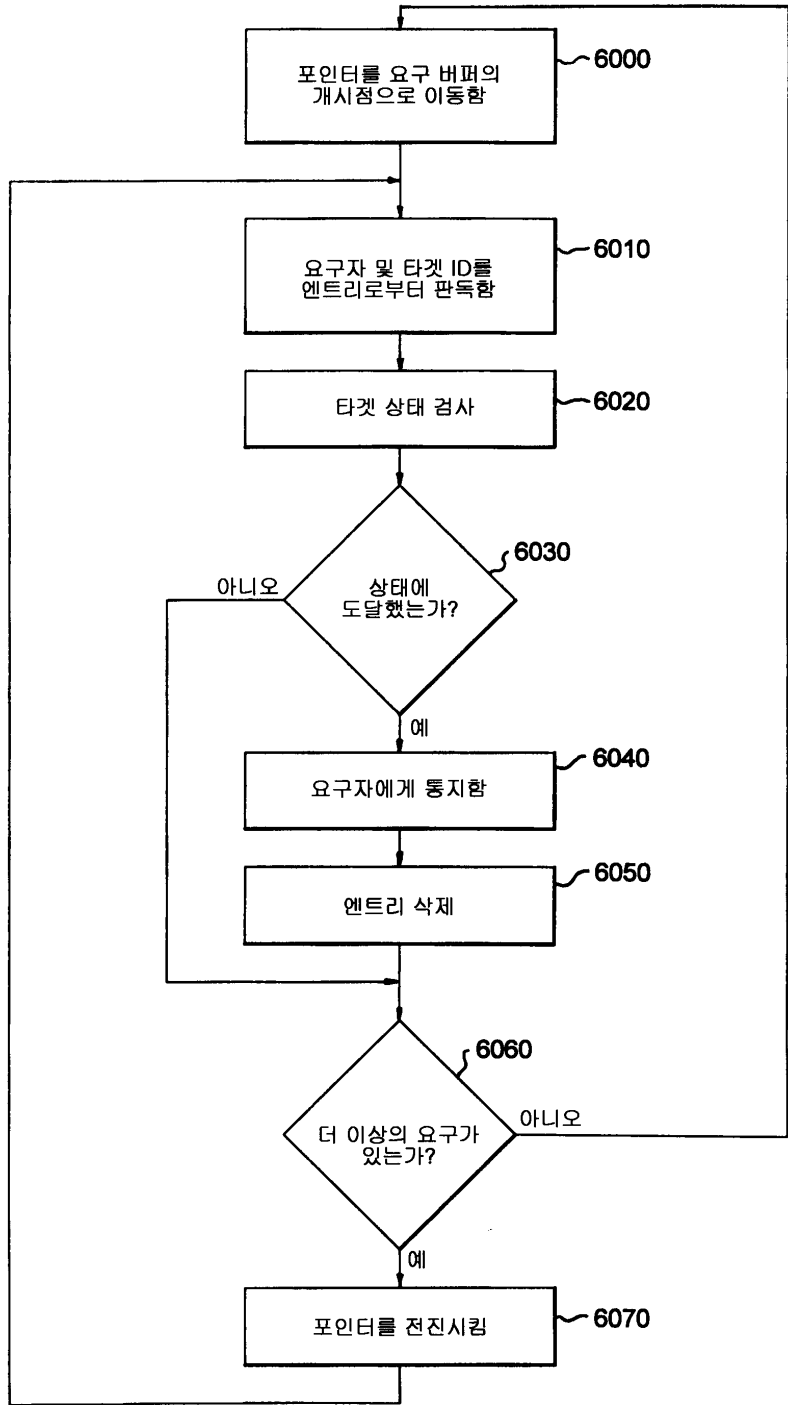


도면3

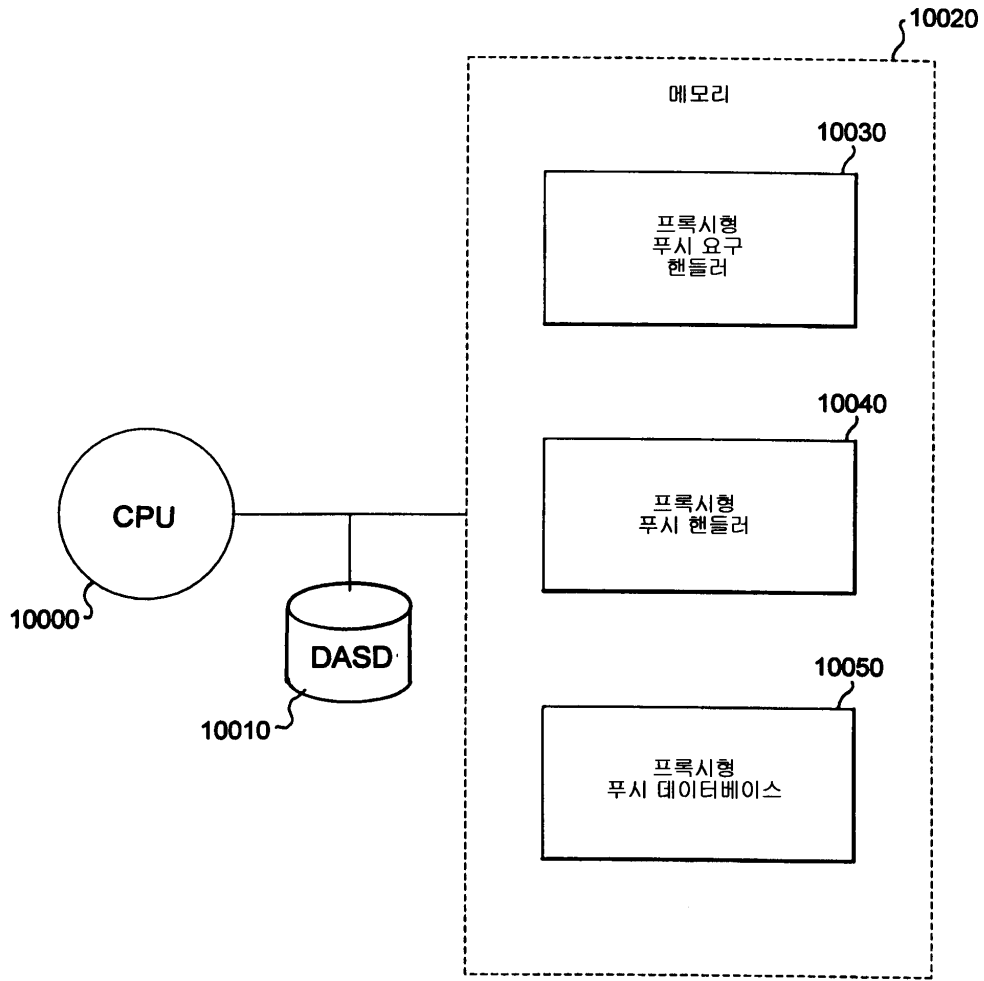


* 자각 통지 핸들러는 병렬로 실행됨(백그라운드)

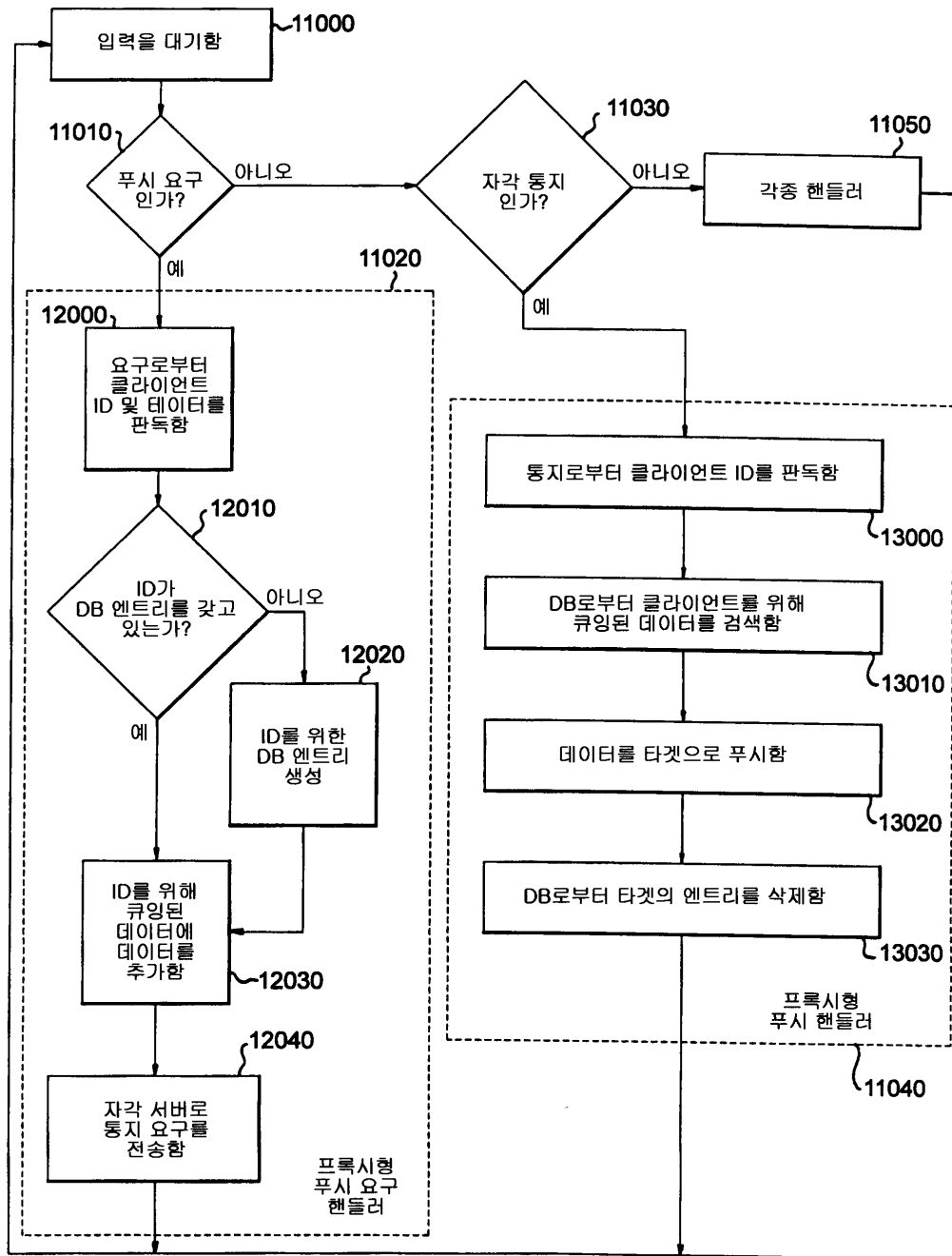
도면4



도면5



도면6



도면7

