



(19)대한민국특허청(KR)
(12) 등록특허공보(B1)

(51) 。 Int. Cl.

H04L 29/06 (2006.01)

H04L 29/02 (2006.01)

H04L 12/28 (2006.01)

(45) 공고일자

2007년08월10일

(11) 등록번호

10-0748259

(24) 등록일자

2007년08월03일

(21) 출원번호

10-2005-0132283

(65) 공개번호

10-2007-0069800

(22) 출원일자

2005년12월28일

(43) 공개일자

2007년07월03일

심사청구일자

2005년12월28일

(73) 특허권자

성균관대학교산학협력단

경기 수원시 장안구 천천동 300 성균관대학교내

(72) 발명자

임병인

서울 중랑구 중화2동 302-12

김형준

서울 마포구 공덕2동 삼성아파트 105동 803호

이승현

경기 수원시 장안구 율전동 388-7번지

이규민

경기 성남시 중원구 중동 331번지 5층

최기현

경기 수원시 장안구 율전동 304-18번지 202호

신호진

강원 원주시 우산동 289-6

장경수

경기 수원시 장안구 율전동 468 삼성아파트 103동 1604호

신동렬

경기 군포시 오금동 소월아파트 361-1603

(74) 대리인

특허법인정직과특허

(56) 선행기술조사문헌

KR1020050107293 A

WO02091682 A1

KR1020060024268 A

WO2004001519 A2

심사관 : 임대식

전체 청구항 수 : 총 20 항

(54) 애드-혹 상에서 우선순위 기반 적응적 디스커버리 시스템 및 그 방법

(57) 요약

제한된 네트워크 트래픽을 사용한 디스커버리 알고리즘 및 동적인 네트워크 환경에 적응하는 애드-혹 상에서 서비스 디스커버리 시스템 및 그 방법에 관한 것으로, 유비쿼터스 서비스 제공하는 애플리케이션 계층, 캐시 관리자(Cache Manager) 및 정책 관리자(Policy Manager)를 구비한 서비스 관리 계층 및 시스템간의 통신 및 서비스 전달을 실행하는 네트워크 통신 계층을 포함하며, 상기 캐시 관리자는 주변 노드들이 제공하는 서비스 정보를 저장하고 관리하기 위한 모듈이고, 정책 관리자는 유저의 선호도, 디바이스의 컴퓨팅 능력, 네트워크의 상태에 따라 서비스 광고에 대한 주기 결정, 서비스 캐시 갱신 정책, 네트워크를 통해 들어온 패킷에 대한 처리를 실행하는 모듈인 구성을 마련한다.

상기와 같은 애드-혹 상에서 우선순위 기반 적응적 디스커버리 시스템 및 그 방법을 이용하는 것에 의해, 애드-혹 네트워크의 높은 패킷 손실에 대항하여 빠르고 신뢰성 있는 서비스를 제공할 수 있다.

대표도

도 1

특허청구의 범위

청구항 1.

네트워크로 연결된 다수의 모바일 디바이스의 각각에 탑재되어 서비스 디스커버리 및 전달을 가능하게 하는 애드-혹 상의 우선순위 기반 적응적 디스커버리 시스템으로써,

상기 네트워크를 통해 유비쿼터스 서비스 제공하는 애플리케이션 계층(100),

상기 시스템 동작을 조율하고, 주변 노드들이 제공하는 서비스 정보를 저장하고 관리하는 캐시 관리자(Cache Manager) 및 유저의 선호도(Preference), 상기 디바이스의 컴퓨팅 능력, 상기 네트워크의 상태에 따라 서비스 광고에 대한 주기 결정, 서비스 캐시 갱신 정책, 네트워크를 통해 들어온 패킷에 대한 처리를 실행하는 정책 관리자(Policy Manager)를 구비한 서비스 관리 계층(200) 및

상기 시스템간의 통신 및 서비스 전달을 실행하는 네트워크 통신 계층(300)을 포함하는 것을 특징으로 하는 우선순위 기반 적응적 디스커버리 시스템.

청구항 2.

제 1항에 있어서,

상기 서비스 디스커버리의 수행은 델타 메시지와 더미 메시지를 사용하여 중복된 메시지를 최소화하는 것을 특징으로 하는 우선순위 기반 적응적 디스커버리 시스템.

청구항 3.

제 1항에 있어서,

상기 캐시 관리자는 우선순위를 가지는 트리 구조로 서비스 캐시를 관리하는 것을 특징으로 하는 우선순위 기반 적응적 디스커버리 시스템.

청구항 4.

제 1항에 있어서,

상기 네트워크 통신 계층은 UDP기반의 멀티캐스트(Multicast)와 TCP기반의 유니캐스트(Unicast), HTTP(hypertext transport protocol), 분산 환경에서 구조화된 정보를 교환하기 위한 경량 프로토콜인 SOAP(Simple Object Access Protocol)를 포함하는 것을 특징으로 하는 우선순위 기반 적응적 디스커버리 시스템.

청구항 5.

제 1항에 있어서,

상기 정책 관리자는 서비스 캐시를 이용한 디스커버리 과정과 우선순위에 기반한 시스템 운영을 통하여 상기 네트워크 및 시스템의 오버헤드를 줄이며 동적인 환경에 적응시키는 것을 특징으로 하는 우선순위 기반 적응적 디스커버리 시스템.

청구항 6.

제 5항에 있어서,

상기 정책 관리자는 상기 네트워크 및 시스템의 오버헤드를 줄이기 위해 상기 네트워크를 통해 들어오는 주변 서비스에 대한 정보를 우선순위 큐를 이용하여 관리하는 것을 특징으로 하는 우선순위 기반 적응적 디스커버리 시스템.

청구항 7.

제 2항에 있어서,

상기 서비스 디스커버리의 수행은 XML(eXtensible Markup Language) 프로세싱을 사용하여 실행되는 것을 특징으로 하는 우선순위 기반 적응적 디스커버리 시스템.

청구항 8.

제 7항에 있어서,

상기 XML 프로세싱은 패킷 필드를 형성하여 데이터를 주고받으며 디스커버리를 수행하고,

상기 패킷은 서비스 광고 메시지의 패킷, 서비스 요청 메시지의 패킷 또는 서비스 응답 메시지의 패킷 중의 하나인 것을 특징으로 하는 우선순위 기반 적응적 디스커버리 시스템.

청구항 9.

제 8항에 있어서,

상기 패킷 필드는

멀티캐스트 주소로 메시지가 들어올 때 구분하는 헤더,

메시지 수신시 처리 루틴을 결정하는 Cmd필드,

서비스에 대한 고유한 식별자인 UUID필드,

서비스에 대한 키워드로써 서비스 매칭 모듈에서 사용되는 ServiceType 필드,

사용자에게 가식성을 높여주기 위한 FriendlyName 필드,

서비스 제공자의 주소를 나타내는 SourceAddress,

메시지의 응답을 위한 수신자의 주소를 나타내는 ReceiveAddress필드 및 서비스 신뢰성을 높이기 위해 서비스 유효기간을 표시하는 TTL필드를 포함하는 것을 특징으로 하는 우선순위 기반 적응적 디스커버리 시스템.

청구항 10.

서버인 서비스 공급자와 네트워크로 연결된 모바일 디바이스인 서비스 소비자가 네트워크로 연결되고, 상기 서비스 소비자의 각각에 탑재되어 서비스 디스커버리 및 전달을 가능하게 하는 애드-혹 상의 우선순위 기반 적응적 디스커버리 방법으로써,

(a) 상기 서비스 소비자가 로컬 캐시를 체크하는 단계,

(b) 상기 체크 단계에서 로컬 캐시가 존재하지 않으면, 상기 서비스 소비자가 상기 서비스 공급자에게 서비스를 요청하는 단계,

(c) 상기 서비스 소비자의 요청에 따라 상기 서비스 공급자가 서비스 응답을 통지하는 단계,

(d) 상기 (c) 단계에서의 서비스 응답에 따라 상기 서비스 소비자가 상기 서비스 공급자에게 서비스 기술(Description)을 요청하는 단계,

(e) 상기 (d) 단계에서의 요청에 따라, 상기 서비스 공급자가 서비스 기술 응답을 서비스 소비자에게 보내는 단계,

(f) 상기 (e) 단계에서 서비스 공급자로부터 받은 서비스 기술에 따라 서비스 소비자가 연속하는 XML 및 XSD 유효를 체크하는 단계,

(g) 상기 (f)단계에 이어 상기 서비스 소비자가 서비스 공급자에게 SOAP RPC(Remote Procedure Call)를 요청하는 단계,

(h) 상기 (g)단계에 이어 상기 서비스 공급자가 비연속 XML 및 XSD 유효를 체크하고, 요청된 서비스를 호출하는 단계,

(i) 상기 (h)단계에서 호출된 서비스에 따라 상기 서비스 공급자가 서비스 소비자에게 SOAP RPC응답을 보내는 단계를 포함하는 것을 특징으로 하는 우선순위 기반 적응적 디스커버리 방법.

청구항 11.

제 10항에 있어서,

상기 (a)단계에서 서비스가 존재하면, 상기 (b) 단계 및 상기 (c)단계를 통하지 않고 상기 (d) 단계로 진행하는 것을 특징으로 하는 우선순위 기반 적응적 디스커버리 방법.

청구항 12.

제 11항에 있어서,

상기 서비스 기술(Description)의 요청 및 응답은 HTTP를 거쳐 실행되는 것을 특징으로 하는 우선순위 기반 적응적 디스커버리 방법.

청구항 13.

제 12항에 있어서,

상기 요청 및 응답의 단계에서 상기 모바일의 노드가 주기적으로 캐시 안에 존재하는 서비스에 대해 유효성을 체크하는 폴링(Polling)과 상기 서비스 공급자가 자신의 정보가 변경하게 되면, 상기 모바일의 주변 노드에게 알림으로써 서비스 변경에 대해 즉각적인 광고하는 통지(Notification)를 지원하는 것을 특징으로 하는 우선순위 기반 적응적 디스커버리 방법.

청구항 14.

제 11항에 있어서,

상기 서비스 디스커버리, 상기 캐시의 관리, 동적인 시스템 운영은 서비스 공급자의 정책 관리자에 의해 실행되고,

상기 정책 관리자는 유저의 선호도(Preference), 상기 디바이스의 컴퓨팅 능력, 상기 네트워크의 상태에 따라 서비스 광고에 대한 주기 결정, 서비스 캐시 갱신 정책, 네트워크를 통해 들어온 패킷에 대한 처리를 실행하는 것을 특징으로 하는 우선순위 기반 적응적 디스커버리 방법.

청구항 15.

제 14항에 있어서,

상기 서비스 디스커버리의 수행은 델타 메시지와 더미 메시지를 사용하여 중복된 메시지를 최소화하는 것을 특징으로 하는 우선순위 기반 적응적 디스커버리 방법.

청구항 16.

제 15항에 있어서,

상기 캐시 관리는 우선순위를 가지는 트리 구조로 관리되는 것을 특징으로 하는 우선순위 기반 적응적 디스커버리 방법.

청구항 17.

제 14항에 있어서,

상기 정책 관리자는 상기 네트워크 및 시스템의 오버헤드를 줄이기 위해 상기 네트워크를 통해 들어오는 주변 서비스에 대한 정보를 우선순위 큐를 이용하여 관리하는 것을 특징으로 하는 우선순위 기반 적응적 디스커버리 방법.

청구항 18.

제 15항에 있어서,

상기 서비스 디스커버리의 수행은 XML(eXtensible Markup Language) 프로세싱을 사용하여 실행되는 것을 특징으로 하는 우선순위 기반 적응적 디스커버리 방법.

청구항 19.

제 16항에 있어서,

상기 XML 프로세싱은 패킷 필드를 형성하여 데이터를 주고받으며 디스커버리를 수행하고,

상기 패킷은 서비스 광고 메시지의 패킷, 서비스 요청 메시지의 패킷 또는 서비스 응답 메시지의 패킷 중의 하나인 것을 특징으로 하는 우선순위 기반 적응적 디스커버리 방법.

청구항 20.

제 19항에 있어서,

상기 패킷 필드는

멀티캐스트 주소로 메시지가 들어올 때 구분하는 헤더,

메시지 수신시 처리 루틴을 결정하는 Cmd필드,

서비스에 대한 고유한 식별자인 UUID필드,

서비스에 대한 키워드로써 서비스 매칭 모듈에서 사용되는 ServiceType 필드,

사용자에게 가식성을 높여주기 위한 FriendlyName 필드,

서비스 제공자의 주소를 나타내는 SourceAddress,

메시지의 응답을 위한 수신자의 주소를 나타내는 ReceiveAddress필드 및 서비스 신뢰성을 높이기 위해 서비스 유효기간을 표시하는 TTL필드를 포함하는 것을 특징으로 하는 우선순위 기반 적응적 디스커버리 방법.

명세서

발명의 상세한 설명

발명의 목적

발명이 속하는 기술 및 그 분야의 종래기술

본 발명은 제한된 네트워크 트래픽을 사용한 디스커버리 알고리즘 및 동적인 네트워크 환경에 적응하는 애드-혹 상에서 서비스 디스커버리 시스템 및 그 방법에 관한 것으로, 특히 홈 네트워크, 회의장, 공항, 전쟁 필드 등 네트워크 인프라가 갖추어져 있지 않은 애드-혹 환경에서 각각의 디바이스가 제공하는 서비스 간의 디스커버리 및 전달을 위한 우선순위 기반 시스템 운용 방법에 관한 것이다.

최근에 신문이나 잡지를 보면 '유비쿼터스'란 단어를 자주 접하게 된다. 라틴어에서 유래한 유비쿼터스는 '언제 어디에나 존재하는'이란 뜻이며, 유비쿼터스 컴퓨팅의 줄임말이다. 이는 사용자가 장소에 구애받지 않고 네트워크상에 존재하면서 서비스를 받는 정보기술(IT) 패러다임을 뜻한다.

이러한 유비쿼터스 컴퓨팅 환경을 효과적으로 구축하기 위해서는 기간 망과 함께 애드-혹(ad-hoc) 네트워크 구축이 필수적이다. 따라서, 최근 들어 애드-혹 네트워크 기술이 주목을 받고 있으며, 그 안에 존재하는 디바이스 및 서비스간의 효율적인 디스커버리 기술에 관한 많은 연구발명이 이루어지고 있다.

애드-혹 타입의 네트워크에서는 각 사용자 단말(이하, '이동 노드'라고 부른다)이 나머지 이동 노드에 대한 기지국 또는 라우터로서 작용할 수 있고, 그 결과, 고정 인프라 스트럭처인 기지국이 필요 없다. 따라서, 발신측 노드에서 착신측 노드로 전송되는 데이터 패킷은 착신측 노드에 도달하기 이전에 다수의 중간 이동 노드를 거쳐서 라우팅 되는 것이 통상적이다. 이러한 애드-혹 네트워크의 세부 내용은 미국 특허 제5,943,322호에 개시되어 있다.

또한, 더욱 세련된 애드-혹 네트워크도 개발되고 있는데, 이 애드-혹 네트워크에서는 종래의 애드-혹 네트워크에서와 같이 이동 노드들이 서로 간에 통신할 수 있게 하는 것 이외에도, 이 이동 노드들이 고정 네트워크에 더 액세스할 수 있게 하고, 그 결과, 다른 타입의 사용자 단말, 예를 들어 공중 교환 통신망(PSTN) 및 인터넷과 같은 다른 통신망 상의 사용자 단말과도 통신할 수 있게 한다.

즉, 자체-구성 와이어리스 무선 통신 시스템인 애드-혹 시스템은 셀룰러 무선 통신 시스템들과 관련해서도 점차적으로 더욱 중요해지고 있다. 예를 들어, GSM 및 UMTS와 같은 셀룰러 무선 통신 시스템들과 대조적으로, 애드-혹 시스템들에서의 통신은 이동 단말기들 간에 적어도 부분적으로는 직접 이루어진다.

애드-혹 시스템들에서, 무선국들은 중앙 호-처리 장치가 없이도 그들 간의 무선 접속을 형성할 수 있다. 두 무선국들 간의 접속은 직접적으로 형성되거나 또는 더 큰 거리의 경우에는 접속을 위한 중계국을 형성하는 추가적인 무선국들을 통해서 형성된다. 따라서, 유용한 정보는 무선국으로부터 무선국의 무선 범위에 상응하는 거리보다 멀리 있는 무선국에 전송된다. 애드-혹 시스템들 네트워크의 무선국들은 이동 무선국들(예를 들어, 개인용 이동 무선 장치들 또는 운송 차량의 그러한 장치들) 또는 주로 고정된 무선국들(예를 들어, 컴퓨터들, 프린터들, 가전제품들)일 수 있다. 애드-혹 네트워크를 구성하는 부분들이 되기 위해 무선국은 적어도 하나의 인접하는 무선국의 무선-커버리지 범위 내에 위치해야 한다. HiperLAN 및 IEEE 802.11과 같은 WLAN(Wireless Local Area Networks)은 애드-혹 네트워크들의 일예이다.

또, 애드-혹 환경은 디바이스들이 공간상에 독립적으로 존재하면서 자신이 인식할 수 있는 반경 안의 모든 기기들을 연결하는 네트워크 방식으로, 유선 네트워크상에서는 이미 UPnP, Jini, SLP등 많은 디스커버리 엔진 및 기술들이 개발되어 왔다.

이와 같은 애드-혹 무선 통신 시스템들의 특별한 장점들은 큰 이동성 및 융통성이다. 그러나, 이러한 요인들은 또한 라우팅 방법들에 큰 문제점을 제기한다. 다수의 무선국들로 구성되는 무선 통신 시스템에서는 송신기(데이터 패킷을 전송하는 수 개의 무선국들에 적용 가능함)로부터 수신기로의 데이터 패킷 경로가 발견되어야 한다. 경로를 선택하는 것을 라우팅이라 한다. 만약 무선국들이 이동 무선국들이라면, 네트워크의 토폴로지(topology)는 주로 시간에 따라 변할 것이다. 적합한 라우팅 방법은 이러한 연속적인 변화들을 고려해야 한다.

또한 이러한 애드-혹에 관한 일 예가 국제공개공보 WO 2004/079995(2004년09월16일 공개 : 애드-혹 접속에 의해 확장되는 셀룰러 무선 통신시스템에서 정보를 전송하기 위한 방법 및 기지국), 국제공개공보 WO 2003/079611(2003년09월25일 공개 : 애드-혹 네트워크에서 데이터 전송률과 전송 파워에 대한 적응적 제어를 제공하기 위한 시스템과 방법), 대한민국 공개특허공보 2005-0087404(2005년08월31일 공개 : 애드-혹 네트워크에서 애드-혹 노드 간 안전한 통신방법)등에 개시되어 있다.

즉, 상기 공보 WO 2004/079995에 있어서는 도 8에 도시된 바와 같이, 자체-구성 무선 네트워크 성분(AD-HOC)의 노드로서 사용자 스테이션(N1)으로부터 진행하는 기지국으로의 가능한 몇 개의 경로들을 나타내는 구조에 대해 개시되어 있다. 이러한 경로들은 다음과 같은 루트들이다.

i) N1→N2→N3→N6인 홑을 가짐.

ii) N1→N2→N4→N5→N6인 홑을 가짐.

iii) N1→N8→N9→N10→N11인 홑을 가짐.

상기 경로 i) 및 ii)은 각각 경로 iii)와 분리된다.

도 8에 도시된 것에 따른 라루팅 알고리즘의 사용의 일 예를 설명하면 다음과 같다.

사용자 스테이션은 UMTS 인터페이스를 통해 기지국에 접속하여 시도하며, 차단되거나 또는 범위 밖에 위치한다. 관련된 에러 메시지를 수신한 이후에는, 사용자 스테이션은 애드-혹 네트워크에 ARCE 알고리즘을 적용한다. 이 스테이션은 제일 먼저 경로가 앞서 전송된 패킷들로부터 기지국에 통보되었는지 여부를 검사한다. 만약 그렇다면, 소스는 데이터 경로로서 그 경로를 즉시 사용할 것이다. 그렇지 않다면, 소스는 접속이 형성되기 이전에 목적지를 탐색하기 위해서 네트워크를 통한 요청이 범람할 것이다. 이러한 요청은 대개 초기에는 단지 일반 데이터, 즉 예컨대 TTL(time-to-life) 필드, 소스 주소 및 목적지 주소와 요청-식별 정보(요청 ID)를 포함한다. 소스 주소가 또한 제 1 링크의 제 1 주소로서 입력된다.

다음에 이러한 요청은 방송을 통해 바로 인접한 모든 것들에 전송된다.

또 상기 공개특허공보 2005-0087404에는 애드-혹 네트워크로서 중앙 집중적인 제어를 필요로 하는 현재의 이동통신 네트워크 즉 인프라 구조(infrastructure) 네트워크와는 달리 두 애드-혹 노드(Ad Hoc Node: AHN))간에 무선통신이 이루어지는 네트워크에 대해 기술되어 있다.

즉, 애드-혹 네트워크에서 각 AHN는 송수신을 수행하는 노드의 역할과 함께 다른 노드가 전송한 전송 트래픽을 라우팅해주는 라우터의 역할까지를 겸하고, 애드-혹 네트워크는 기반 망과 완전히 독립적인 형태로 존재하거나, 게이트웨이에 의해 기반 망에 인터워킹 되는 형태로 존재하며, 다수의 애드-혹 노드(AHN)들을 구비한 애드-혹 네트워크에서, 애드-혹 게이트웨이(AHG)가 각 애드-혹 노드(AHN)의 인증서 등록과 분배를 관리함으로써 두 AHN간의 통신을 보다 안전하게 수행할 수 있는 통신방법에 대해 기술되어 있다.

발명이 이루고자 하는 기술적 과제

그러나, 상술한 공보 등에 개시된 기술에 있어서, 애드-혹 네트워크는 다이나믹한 토폴로지의 변화, 디바이스들의 간헐적인 연결, 낮은 대역폭 및 디바이스의 컴퓨팅 능력 문제 등으로 기존의 유선망에서 사용되던 디스커버리 기술을 적용하기에는 부적합하다는 문제점이 있었다.

따라서 본 발명자들은 애드-혹 네트워크에 적합한 디스커버리 엔진에 대한 연구를 수행하였다. 기존에 연구 개발되었던 애드-혹 네트워크에서의 디스커버리 기술들은 크게 2가지 문제점을 가지고 있다.

첫째, 주기적인 멀티캐스팅과 브로드캐스팅을 통해 디스커버리를 수행함으로써 많은 네트워크 트래픽을 발생하게 된다.

둘째, 애드-혹은 네트워크 토폴로지가 자주 바뀌고 네트워크 데이터의 손실이 많기 때문에 네트워크를 통해 받은 서비스 정보를 캐시(Cache)에 저장하고 있지만 캐시 오버 플로우시 이에 대한 대비가 없으며, 또한 서비스에 대한 선호도를 반영하여 캐시 데이터를 유지하는 기법에 대한 연구가 이루어 지지 않았다.

본 발명의 목적은 상술한 바와 같은 문제점을 해결하기 위해 이루어진 것으로서, 고정된 네트워크 인프라가 없는 환경에서도 애드-혹 네트워크를 통하여 서비스를 주고 받을 수 있는 유비쿼터스 컴퓨팅 패러다임을 실현할 수 있는 우선순위 기반 적응적 디스커버리 시스템 및 그 방법을 제공하는 것이다.

본 발명의 다른 목적은 최소한의 네트워크 자원으로 효율적으로 서비스를 찾고 전달하는 우선순위 기반 적응적 디스커버리 시스템 및 그 방법을 제공하는 것이다.

발명의 구성

상기 목적을 달성하기 위해 본 발명에 따른 우선순위 기반 적응적 디스커버리 시스템은 네트워크로 연결된 다수의 모바일 디바이스의 각각에 탑재되어 서비스 디스커버리 및 전달을 가능하게 하는 애드-혹 상의 시스템으로써, 유비쿼터스 서비스 제공하는 애플리케이션 계층, 캐시 관리자(Cache Manager) 및 정책 관리자(Policy Manager)를 구비한 서비스 관리 계층 및 시스템간의 통신 및 서비스 전달을 실행하는 네트워크 통신 계층을 포함하며, 상기 캐시 관리자는 주변 노드들이 제공

하는 서비스 정보를 저장하고 관리하기 위한 모듈이고, 상기 정책 관리자는 유저의 선호도(Preference), 상기 디바이스의 컴퓨팅 능력, 상기 네트워크의 상태에 따라 서비스 광고에 대한 주기 결정, 서비스 캐시 갱신 정책, 네트워크를 통해 들어온 패킷에 대한 처리를 실행하는 모듈인 것을 특징으로 한다.

본 발명에 따른 우선순위 기반 적응적 디스커버리 시스템에 있어서, 상기 서비스 디스커버리의 수행은 델타 메시지와 더미 메시지를 사용하여 중복된 메시지를 최소화하는 것을 특징으로 한다.

본 발명에 따른 우선순위 기반 적응적 디스커버리 시스템에 있어서, 상기 캐시 관리자는 우선순위를 가지는 트리 구조로 서비스 캐시를 관리하는 것을 특징으로 한다.

본 발명에 따른 우선순위 기반 적응적 디스커버리 시스템에 있어서, 상기 네트워크 통신 계층은 UDP기반의 멀티캐스트(Multicast)와 TCP기반의 유니캐스트(Unicast), HTTP(hypertext transport protocol), 분산 환경에서 구조화된 정보를 교환하기 위한 경량 프로토콜인 SOAP(Simple Object Access Protocol)를 포함하는 것을 특징으로 한다.

본 발명에 따른 우선순위 기반 적응적 디스커버리 시스템에 있어서, 상기 정책 관리자는 서비스 캐시를 이용한 디스커버리 과정과 우선순위에 기반한 시스템 운영을 통하여 상기 네트워크 및 시스템의 오버헤드를 줄이며 동적인 환경에 적응시키는 것을 특징으로 한다.

본 발명에 따른 우선순위 기반 적응적 디스커버리 시스템에 있어서, 상기 정책 관리자는 상기 네트워크 및 시스템의 오버헤드를 줄이기 위해 상기 네트워크를 통해 들어오는 주변 서비스에 대한 정보를 우선순위 큐를 이용하여 관리하는 것을 특징으로 한다.

본 발명에 따른 우선순위 기반 적응적 디스커버리 시스템에 있어서, 상기 서비스 디스커버리의 수행은 XML(eXtensible Markup Language) 프로세싱을 사용하여 실행되는 것을 특징으로 한다.

본 발명에 따른 우선순위 기반 적응적 디스커버리 시스템에 있어서, 상기 XML 프로세싱은 패킷 필드를 형성하여 데이터를 주고받으며 디스커버리를 수행하고, 상기 패킷은 서비스 광고 메시지의 패킷, 서비스 요청 메시지의 패킷 또는 서비스 응답 메시지의 패킷 중의 하나인 것을 특징으로 한다.

본 발명에 따른 우선순위 기반 적응적 디스커버리 시스템에 있어서, 상기 패킷 필드는 멀티캐스트 주소로 메시지가 들어올 때 구분하는 헤더, 메시지 수신시 처리 루틴을 결정하는 Cmd필드, 서비스에 대한 고유한 식별자인 UUID필드, 서비스에 대한 키워드로써 서비스 매칭 모듈에서 사용되는 ServiceType필드, 사용자에게 가식성을 높여주기 위한 FriendlyName 필드, 서비스 제공자의 주소를 나타내는 SourceAddress, 메시지의 응답을 위한 수신자의 주소를 나타내는 ReceiveAddress필드 및 서비스 신뢰성을 높이기 위해 서비스 유효기간을 표시하는 TTL필드를 포함하는 것을 특징으로 한다.

또한 상기 목적을 달성하기 위해 본 발명에 따른 우선순위 기반 적응적 디스커버리 방법은 서버인 서비스 공급자와 네트워크로 연결된 모바일 디바이스인 서비스 소비자가 네트워크로 연결되고, 상기 서비스 소비자의 각각에 탑재되어 서비스 디스커버리 및 전달을 가능하게 하는 애드-혹 상의 우선순위 기반 적응적 디스커버리 방법으로써, (a) 상기 서비스 소비자가 로컬 캐시를 체크하는 단계, (b) 상기 체크 단계에서 로컬 캐시가 존재하지 않으면, 상기 서비스 소비자가 상기 서비스 공급자에게 서비스를 요청하는 단계, (c) 상기 서비스 소비자의 요청에 따라 상기 서비스 공급자가 서비스 응답을 통지하는 단계, (d) 상기 (c) 단계에서의 서비스 응답에 따라 상기 서비스 소비자가 상기 서비스 공급자에게 서비스 기술(Description)을 요청하는 단계, (e) 상기 (d) 단계에서의 요청에 따라, 상기 서비스 공급자가 서비스 기술 응답을 서비스 소비자에게 보내는 단계, (f) 상기 (e) 단계에서 서비스 공급자로부터 받은 서비스 기술에 따라 서비스 소비자가 연속하는 XML 및 XSD 유효를 체크하는 단계, (g) 상기 (f)단계에 이어 상기 서비스 소비자가 서비스 공급자에게 SOAP RPC(Remote Procedure Call)를 요청하는 단계, (h) 상기 (g)단계에 이어 상기 서비스 공급자가 비연속 XML 및 XSD 유효를 체크하고, 요청된 서비스를 호출하는 단계, (i) 상기 (h)단계에서 호출된 서비스에 따라 상기 서비스 공급자가 서비스 소비자에게 SOAP RPC응답을 보내는 단계를 포함하는 것을 특징으로 한다.

또 본 발명에 따른 우선순위 기반 적응적 디스커버리 방법에 있어서, 상기 (a)단계에서 서비스가 존재하면, 상기 (b) 단계 및 상기 (c)단계를 통하지 않고 상기 (d) 단계로 진행하는 것을 특징으로 한다.

또 본 발명에 따른 우선순위 기반 적응적 디스커버리 방법에 있어서, 상기 서비스 기술(Description)의 요청 및 응답은 HTTP를 거쳐 실행되는 것을 특징으로 한다.

또 본 발명에 따른 우선순위 기반 적응적 디스커버리 방법에 있어서, 상기 요청 및 응답의 단계에서 상기 모바일의 노드가 주기적으로 캐시 안에 존재하는 서비스에 대해 유효성을 체크하는 폴링(Polling)과 상기 서비스 공급자가 자신의 정보가 변경하게 되면, 상기 모바일의 주변 노드에게 알림으로써 서비스 변경에 대해 즉각적인 광고하는 통지(Notification)를 지원하는 것을 특징으로 한다.

본 발명의 상기 및 그 밖의 목적과 새로운 특징은 본 명세서의 기술 및 첨부 도면에 의해 더욱 명확하게 될 것이다.

먼저 본 발명의 개념에 대해 설명한다.

본 발명에 있어서는 다음과 같은 기술적 과제를 해결하고자 하는 것이다.

첫째, 중복된 메시지의 송. 수신을 제거하여 네트워크 트래픽을 최소화하는 디스커버리 알고리즘 및 프로토콜의 제공.

둘째, 서비스에 대한 캐시를 유지관리 하는데 있어서 서비스에 대한 선호도를 반영하여 저장, 제거, 교체하는 관리기법 제공.

셋째, 동적인 네트워크상에서 적응적인 시스템을 위한 정책기반의 시스템 운용 기법의 제공.

이하, 본 발명의 구성을 도면에 따라서 설명한다.

또한, 본 발명의 설명에 있어서는 동일 부분은 동일 부호를 붙이고, 그 반복 설명은 생략한다.

본 발명자들은 애드-혹과 같은 동적인 네트워크 환경에서 효율적인 서비스 디스커버리를 위한 LSD(Lightweight Service Discovery) 엔진에 대한 연구를 수행하였었다.

도 1은 LSD 엔진의 전체 시스템 구성도를 나타낸다.

도 1에 있어서, (100)은 애플리케이션 계층(Application Layer)이고, (200)은 서비스 관리 계층(Service Management Layer)이며, (300)은 네트워크 통신 계층(Network Communication Layer)으로서 총 3개의 계층으로 구성되어 있다.

애플리케이션 계층(100)에서는 유비쿼터스 서비스 제공을 위한 SDK 및 애플리케이션이 제공된다.

서비스 관리 계층(200)에서 LSD는 캐시 관리자(Cache Manager) 및 정책 관리자(Policy Manager)로 구성된 2개의 주요 모듈을 구비한다. 캐시 관리자는 주변 노드들이 제공하는 서비스 정보를 저장하고 관리하기 위한 모듈이고, 정책 관리자는 유저의 선호도(Preference), 디바이스의 컴퓨팅 능력, 네트워크의 상태에 따라 서비스 광고에 대한 주기 결정, 서비스 캐시 갱신 정책, 네트워크를 통해 들어온 패킷에 대한 처리 등 LSD엔진의 전체적인 시스템 동작을 조율한다.

마지막으로 네트워크 통신 계층(300)은 UDP기반의 멀티캐스트(Multicast)와 TCP기반의 유니캐스트(Unicast), HTTP(hypertext transport protocol), 분산 환경에서 구조화된 정보를 교환하기 위한 경량 프로토콜인 SOAP(Simple Object Access Protocol) 등으로 구성되어 시스템간의 통신 및 서비스 전달을 담당한다.

본 발명에서는 서비스 캐시를 이용한 디스커버리 과정과 우선순위에 기반한 시스템 운영을 통하여 네트워크, 시스템 오버헤드를 줄이며 동적인 환경에 적응하기 위한 기법을 제공한다.

서비스 디스커버리를 위해서는 서비스에 대해 사용자가 이해하기 쉽고 유연한 표현이 필요하므로, 본 발명에서는 1996년 W3C(World Wide Web Consortium)에서 제안되어 구조화된 표현을 가능하게 하는 XML(eXtensible Markup Language)을 사용한다.

도 2는 네트워크에 존재하는 디바이스의 정보를 XML로 표현한 도면이다. 상단부분에는 해당 노드에 대한 위치(location field), 종류(device Type field), 유효시간(TTL field) 등이 기술되어있다. 하단부분에는 해당 노드에서 제공하고 있는 서비스에 대한 식별자(serviceID), 종류(serviceType), 더불어 서비스 활성화를 위한 SOAP 메시지 호출을 위한 기능명(functionName), 파라미터 종류(parameterType) 등이 기술되어있다.

LSD는 도 2의 XML 정보를 주고받으면서 서비스 디스커버리를 수행하게 된다. 하지만 XML 자체를 전송하게 되면 네트워크 및 시스템 오버헤드가 증가하게 된다.

따라서, XML 프로세싱을 통하여 도 3과 같은 구조체 형태의 패킷 필드를 형성하게 되고 그 데이터를 주고받으며 디스커버리를 수행한다.

도 3a는 서비스 광고 메시지의 패킷을 나타내고, 도 3b는 서비스 요청 메시지의 패킷을 나타내며, 도 3c는 서비스 응답 메시지의 패킷을 나타낸다.

도 3의 헤더(Header)는 LSD와 동일한 멀티캐스트 주소로 메시지가 들어올 때 구분하기 위한 필드이다. 메시지 수신시 처리 루틴을 결정하기 위한 Cmd필드, 서비스에 대한 고유한 식별자인 UUID필드, 서비스에 대한 키워드로써 서비스 매칭 모듈에서 사용되는 ServiceType필드, 사용자에게 가식성을 높여주기 위한 FriendlyName 필드 등이 있다.

또한, 서비스 제공자의 주소를 나타내는 SourceAddress, 메시지의 응답을 위한 수신자의 주소를 나타내는 ReceiveAddress필드 및 서비스 신뢰성을 높이기 위해 서비스 유효기간을 표시하는 TTL필드 등이 있다.

LSD는 내부에 고유한 멀티캐스트 메시지를 듣기 위한 데몬이 동작 중이며 수신된 메시지는 필요에 따라 캐시에 저장된다.

도 4는 LSD의 서비스 디스커버리 과정을 나타내는 도면이다.

대부분의 디스커버리 시스템이 그러하듯, LSD도 캐시기반의 디스커버리를 수행한다. 도메인 안에 존재하는 가용한 서비스 정보를 캐시에 저장하고 이를 기반으로 동작한다.

도 4는 네트워크상에 노드 3개(노드 1, 노드 2, 노드 3)가 존재할 때 서비스 디스커버리를 위한 광고, 질의, 응답 과정을 보여준다.

도 4의 케이스 1은 서비스 광고의 과정을 나타낸다.

노드 1이 자신의 서비스 정보를 주변에 광고하기 위해 멀티캐스팅을 하면, 메시지를 받은 노드는 자신의 캐시에 해당 서비스 정보가 없다면 저장을 하게 된다. 그리고 나서, 자신의 가지고 있는 추가적인 정보를 사용하여 Delta message(ex. Local cache{a, b, c}, Advertisement message{a} = Delta message{b, c})를 만들고 랜덤 시간 동안 대기하게 된다. 랜덤시간 대기하는 이유는 노드들의 동시응답으로서 네트워크 기기의 고장이나 브리지에서의 루프 발생 등의 원인으로 네트워크상에동보 통신용 패킷이 발생하는 브로드캐스팅 소동(Broadcast storm)을 방지하기 위함이다.

노드 2가 활성화되면 자신의 델타 메시지(Delta message)를 보내게 되고, 델타 메시지를 받은 노드는 자신의 캐시에 해당 서비스 정보가 없다면 저장을 하고, 대기 중인 자신의 델타 메시지를 삭제하게 된다.

노드 3이 활성화되면 동일한 과정을 거치게 된다. 이런 과정을 통하여 모든 노드들은 중복되는 메시지를 최소화하며, 현재 네트워크에 존재하는 서비스에 대한 정보를 공유할 수 있게 된다.

도 4의 케이스 2는 임의의 상태에서의 서비스 발견을 나타내고, 케이스 3은 수렴(convergent)상태에서 서비스 발견을 나타낸다.

새로운 노드가 네트워크에 참여할 때, 노드는 더미 메시지를 보내게 된다. 더미 메시지를 받은 노드는 앞의 케이스 1과 같은 과정을 반복하여 네트워크 안의 정보를 알려주게 된다. 케이스 2 및 케이스 3에서 알 수 있는 바와 같이, 노드가 수렴 상태에 가깝게 있을수록 적은 네트워크 트래픽과 프로세싱만으로 정보의 공유가 가능해진다.

차후, 서비스 디스커버리 시 대부분의 정보는 로컬 캐시 안에 존재하게 되고, 로컬 캐시 안에 존재하지 않을 때, 하나의 노드에서 온 하나의 패킷을 라우터에 접속되어 있는 다른 모든 노드로 전달하는 플러딩(flooding)을 통해 민감한(Reactive)한 디스커버리를 수행하게 된다.

애드-혹 환경에서의 서비스 디스커버리는 네트워크의 동적인 특성을 고려해야만 한다. 이를 해결하기 위해 본 발명에 따른 LSD에서는 주변 노드의 정보를 캐시에 저장하여 활용하는 캐시기반 디스커버리를 수행한다.

LSD상에는 캐시 관리자라는 주요 컴포넌트가 위치하게 되며 캐시의 운용과 관련된 모든 업무를 담당한다. 서비스의 광고를 받으면 내부 정책(Policy)에 따라 서비스를 캐시에 저장하게 된다. 만약 서비스에 대한 디스커버리를 원하면 캐시 관리자는 가장 먼저 자신의 로컬 캐시를 탐색하게 되고, 만약 원하는 데이터가 존재한다면 유니캐스트를 통하여 자세한 정보를 요청하게 된다.

단지 로컬 캐시에 해당 정보가 존재하지 않을 경우만 멀티캐스트를 함으로써 불필요한 네트워크 트래픽을 줄일 뿐만 아니라, 동적인 네트워크에 적응력을 향상시킬 수 있다. 하지만, 캐시의 제한된 용량으로 인하여 추가, 교체, 삭제 등의 문제를 고려해야 한다.

본 발명에 따른 LSD에서는 디바이스가 네트워크에 참여시 디바이스의 저장능력을 고려하여 우선적으로 캐시 크기를 동적으로 결정하게 된다. 그리고 나서, 네트워크를 통해 들어오는 주변 서비스에 대한 정보는 도 5와 같은 우선순위 큐를 이용하여 관리하게 된다. 우선순위 큐는 FIFO(First In First Out) 데이터 구조로써 데이터에 가중치를 부여하여 관리를 편하게 하여 효율성을 높일 수 있는 자료구조이다.

서비스에 대한 광고나 요청(또는 질의)에 따라 큐 안의 데이터 우선순위를 변경함으로써 선호도(Popularity)가 높은 서비스에 대해 더욱 많은 서비스 광고를 하거나 장시간 캐싱(Caching)을 하는 등의 효과를 볼 수 있다. 또한, 캐시 오버플로우시 서비스에 대한 재정렬, 데이터 삭제시, 희생노드(victim node) 선정 시에도 유용하게 사용될 수 있다.

도 6은 LSD의 서비스 캐싱을 위한 동작을 나타내는 도면이다.

도 6에 있어서, 하나의 모바일 노드 캐시에 주변 서비스 정보 S3, S2, S4가 존재하고 그 우선순위가 각각 2, 5, 5라 할 때, 이 노드가 S4에 관한 광고 메시지를 수신하게 되면 그 우선순위를 1 감소하고 트리를 재배열한다(도 6의 2번째 단계).

만약 새로운 서비스 S7이 네트워크에 조인하게 되고, 그 광고메시지를 수신하게 되면 해당 서비스를 캐시에 추가하게 되며, 그 우선순위를 1로 설정하게 된다. 그리고 나서 마찬가지로 트리의 재배열을 수행하게 된다(도 6의 3번째 단계).

LSD상에 동작하는 타이머에 의해 오랜 기간 동안(예를 들어, 1분) 모든 노드의 서비스 접근이 발생하지 않으면, 도 6의 4번째 단계와 같이 그 우선순위가 모두 1씩 증가하게 된다. 마찬가지로 특정 노드의 접근이 없을 시에는 해당 노드의 우선순위만 증가하게 된다(도 6의 4번째 단계). 따라서 오래된 데이터는 캐시의 하단에 위치하게 되며 서비스에 대한 가중치가 특정 값(예를 들어, 10 priority) 이상이 되면 캐시에서 삭제하게 된다.

만약 큐의 오버플로우가 발생하게 되면 도 6의 5번째 단계와 같이, 우선순위가 가장 높은 최하단의 노드를 삭제하게 된다. LSD상에서는 네트워크상에서 특정 서비스에 대한 광고 빈도나 접근 요청을 반영하여 캐시에 우선순위로 반영하여 선호도(Popularity)가 높은 서비스는 트리 상의 상단에 위치시켜 캐시의 검색 속도를 증가시키게 된다. 또한 접근빈도가 낮은 서비스 데이터를 우선적으로 캐시에서 삭제하여 서비스 적중률(Service hitting rate)을 증가시키는 효과를 볼 수 있다.

본 발명에 따른 LSD는 우선순위를 반영한 서비스 캐시관리 기법과 더불어 서비스의 광고주기를 동적으로 변화시킨다. 모든 서비스는 디폴트로 2분의 간격으로 자신의 존재, 변화를 알리는 광고 메시지를 보내게 된다.

만약 서비스에 대한 요청(Request) 메시지를 주변 노드로부터 받게 되면, 해당 서비스에 대한 선호도가 증가하는 것으로 판단하여 현재 광고주기의 20%를 줄이게 된다. 캐시운영과 동일한 방법으로 특정시간 접근이 없으면, 현재 광고주기 20%를 증가하는 방법을 사용한다. 여기서 만약 서비스에 접근이 아무리 많을지라도 너무 자주 광고메시지를 보내게 된다면 네트워크 트래픽이 많이 발생하게 되므로, 본 발명에서는 최대 2분에서 최소 30초의 사이에서 광고주기를 동적으로 변화하게 한다.

서비스를 위한 캐시의 관리시 중요한 또 하나의 이슈는 캐시 데이터 간의 일관성을 유지하는 것이다. 부패한 캐시는 오히려 네트워크 트래픽 및 시스템 오버헤드를 증가시킨다. 잘못된 네트워크 메시지를 주고받으며 서비스의 유효성을 체크하면 신뢰도를 증가시킬 수 있다. 하지만 그만큼 네트워크 트래픽은 증가하게 된다. 그러므로 최소한의 메시지를 주고 받으며 캐시 데이터 간 신뢰성을 증가시키는 연구가 필요하다.

따라서 본 발명에 따른 LSD에서는 캐시 데이터 간 일치성을 보장하기 위해, 여러 개의 단말 장치에 대하여 차례로 송신 요구의 유무를 문의하고 요구가 있을 경우에는 그 단말 장치에 송신을 시작하는 폴링(Polling)과 통지(Notification)를 지원한다.

폴링은 각각의 모바일 노드가 주기적으로 캐시 안에 존재하는 서비스에 대해 유효성을 체크하게 되고, 서비스의 변경된 정보를 갱신하게 된다. 통지는 각각의 서비스 공급자가 자신의 정보가 변경하게 되면, 주변 노드에게 알림으로써 서비스 변경에 대해 즉각적인 광고를 하게 된다. 이때 폴링과 통지에서도 캐시에 저장된 우선순위를 반영하여 빈도수를 조정함으로써, 적은 네트워크 트래픽으로 캐시 동기화 문제를 일부 해결할 수 있다.

상술한 바와 같이, 본 발명에 따른 LSD의 디스커버리 과정, 캐시관리 기법, 동적인 시스템 운영방법은 LSD상에 존재하는 정책 관리자에 의해 수행되게 된다. 최종적으로 찾아진 서비스는 도 7과 같은 과정에 의하여 SOAP를 이용하여 활성화 및 전달받을 수 있다.

즉, 도 7에 있어서, 서비스 소비자는 로컬 캐시를 체크한다(스텝 1). 이때 로컬 캐시가 존재하면 스텝 4로 진행한다.

스텝 1에서 로컬 캐시가 존재하지 않으면, 서비스 소비자는 서비스 공급자에게 서비스를 요청한다(스텝 2).

서비스 소비자의 요청에 따라 서비스 공급자는 서비스 응답을 통지한다(스텝 3).

이 서비스 응답에 따라 서비스 소비자는 HTTP를 거쳐 서비스 공급자에게 서비스 기술(Description)을 요청한다(스텝 4). 이에 따라, 서비스 공급자는 HTTP를 거쳐 서비스 기술 응답을 서비스 소비자에게 보낸다(스텝 5).

스텝 5에서 서비스 공급자로부터 받은 서비스 기술에 따라 서비스 소비자는 연속하는 XML 및 XSD 유효를 체크한다(스텝 6). 다음에 서비스 소비자는 서비스 공급자에게 SOAP RPC(Remote Procedure Call)를 요청한다(스텝 7).

본 발명에서 분산 환경에서 구조화된 정보를 교환하기 위한 경량 프로토콜인 SOAP(Simple Object Access Protocol)를 사용하는 이유는 HTTP의 요청과 응답의 메시지에 SOAP 메시지가 포함될 수 있기 때문이다.

다음에 서비스 공급자는 비연속 XML 및 XSD 유효를 체크하고(스텝 8), 요청된 서비스를 호출한다(스텝 9). 다음에 스텝 9에서 호출된 서비스에 따라 서비스 공급자는 서비스 소비자에게 SOAP RPC응답을 보낸다.

상술한 설명에 있어서, 서비스 공급자는 서버로서 기능하고, 서비스 소비자는 클라이언트로서 기능한다.

이상 본 발명자에 의해서 이루어진 발명을 상기 실시 예에 따라 구체적으로 설명하였지만, 본 발명은 상기 실시 예에 한정되는 것은 아니고 그 요지를 이탈하지 않는 범위에서 여러 가지로 변경 가능한 것은 물론이다.

발명의 효과

상술한 바와 같이, 본 발명에 따른 우선순위 기반 적응적 디스커버리 시스템 및 그 방법에 의하면, 네트워크상의 서비스 선호도(Popularity)에 따라 우선순위를 부여하여 서비스 캐시에 관리하게 하므로, 캐시에 저장된 서비스의 우선순위에 기반하여 캐싱(Caching) 정책, 서비스 광고주기, 네트워크 패킷처리 등을 동적으로 조율함으로써 애드-혹과 같은 환경에 적응적인 시스템 운용이 가능하게 할 수 있다는 효과가 얻어진다.

또, 본 발명에 따른 우선순위 기반 적응적 디스커버리 시스템 및 그 방법에 의하면, LSD(Lightweight Service Discovery)가 네트워크 안에서 서비스의 접근빈도와 같은 선호도에 따라 우선순위를 부여하게 하고, 그에 기반하여 서비스의 저장, 갱신, 그리고 삭제 수행하므로, 애드-혹 네트워크의 높은 패킷 손실에 대항하여 빠르고 신뢰성 있는 서비스를 제공할 수 있다는 효과가 얻어진다.

도면의 간단한 설명

도 1은 본 발명에 따른 LSD 시스템의 구성도,

도 2는 본 발명에서 사용되는 디바이스 및 서비스를 기술하고 있는 XML 디스크립션 구조도,

도 3은 본 발명에 따라 디스커버리를 수행하기 위해 사용되는 패킷구조도,

도 4는 본 발명에 따라 디스커버리를 수행하기 위한 과정을 보여주는 흐름도,

도 5는 본 발명에 따라 서비스를 캐싱하기 위한 큐의 구조도,

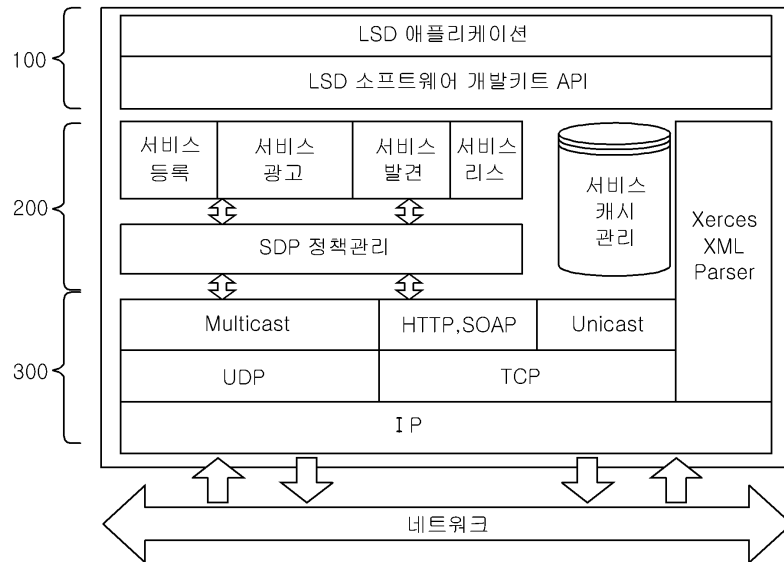
도 6은 본 발명에 따라 네트워크상의 서비스 접근빈도와 같은 선호도에 따라 우선순위를 부여하고, 그에 기반하여 캐시의 저장, 갱신, 그리고 삭제를 수행하는 과정을 보여주는 흐름도,

도 7은 본 발명에서 서비스 디스커버리 및 전달까지의 과정을 보여주는 순차도,

도 8은 종래의 기술로서 AD-HOC의 노드로서 사용자 스테이션으로부터 진행되는 경로들을 나타내는 구조도.

도면

도면1



도면2

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE lsdk [
  <!-- device's property -->
  <deviceID>UUID:/34b10c4c-95a1-4178-8af9-ce771e5fa3c3</deviceID>
  <location>um4sd.skku.ac.kr</location>
  <deviceType>printer</deviceType>
  <friendlyName>hp-printer</friendlyName>
  <manufacturer>hp</manufacturer>
  <model>
    <modelName>hp-s1024</modelName>
    <modelNumber>4520ctg</modelNumber>
  </model>
  <controlURL>http://nrlab.skku.ac.kr</controlURL>
  <TTL>
    <start>2004-10-15T10:50:40</start>
    <end>2006-10-16T10:50:30</end>
  </TTL>
  <!-- available services and function -->
  <services>
    <serviceID>UUID:/ed8a2c53-93c9-47dd-a489-7df019aa9007</serviceID>
    <serviceType>printing service</serviceType>
    <serviceDescription>set parameter for printing number</serviceDescription>
    <functionName>invokeprint</functionName>
    <parameterType>boolean</parameterType>
  </services>
</device>
</lsdkroot>
```

도면3a

Header	Cmd	UUID	ServiceType	FriendlyName	SourceAddress	TTL
--------	-----	------	-------------	--------------	---------------	-----

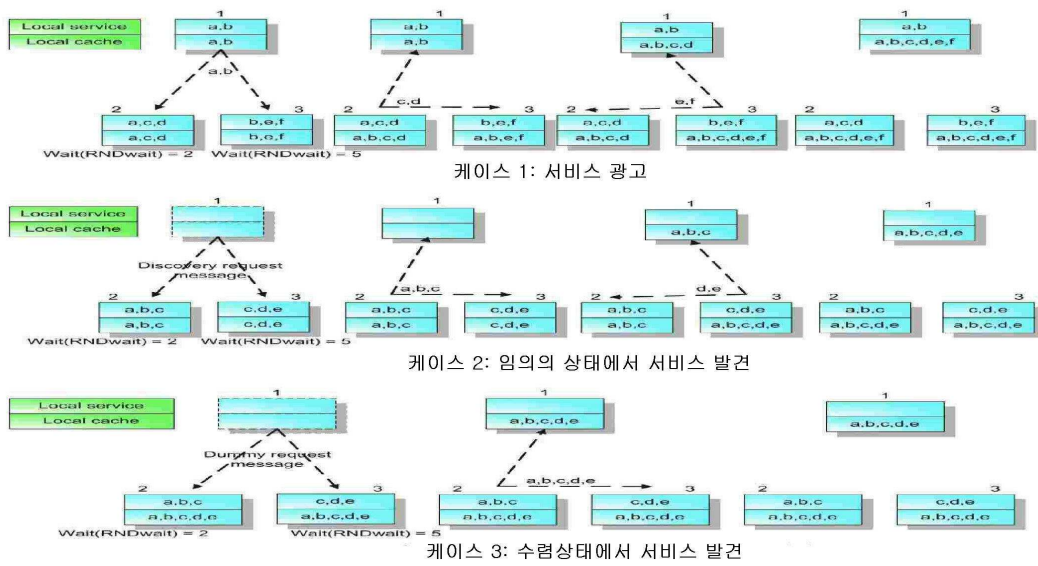
도면3b

Header	Cmd	ServiceType	FriendlyName	ReceiveAddress	TTL
--------	-----	-------------	--------------	----------------	-----

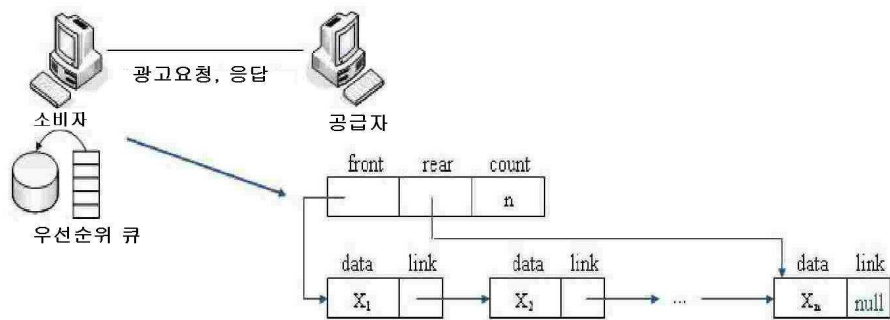
도면3c

Header	Cmd	UUID	ServiceType	FriendlyName	SourceAddress	TTL
--------	-----	------	-------------	--------------	---------------	-----

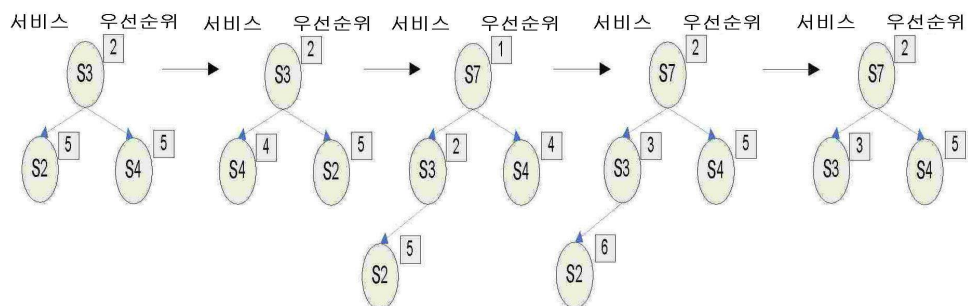
도면4



도면5



도면6



도면7



도면8

