

액티브 네트워크 메커니즘을 이용한 이동 콘텐츠 적응형 네트워크

(Mobile Contents Adaptation Network using Active Network Mechanisms)

김 기 조 [†] 이 준 호 ^{**} 임 경 식 ^{***}
(Kijo Kim) (Junho Lee) (Kyungshik Lim)

오 승 희 ^{****} 남 택 용 ^{****} 손 승 원 ^{****}
(Seung-Hee Oh) (Taekyong Nam) (Sungwon Sohn)

요 약 이동 단말의 짧은 개발 주기에 따라 장치 및 적용 표준이 다양화됨으로써, 무선 콘텐츠 제공자가 효과적인 콘텐츠 서비스를 제공하는데 많은 장애 요소가 되고 있다. 이러한 문제점들을 효과적으로 개선하기 위하여 공용의 원본 콘텐츠를 이동 단말에 전달되는 과정에서 동적으로 단말에 최적화 시킬 수 있는 서비스 플랫폼이 이상적인 대안이 될 수 있다. 그러므로, 본 논문에서는 동적으로 네트워크 프로토콜을 설치하기 위한 기술인 액티브 네트워크 메커니즘을 이용하여, 콘텐츠 서비스를 효과적으로 지원하기 위한 무선 콘텐츠 적응형 네트워크(Mobile Contents Adaptation Network : MobiCAN)를 제안한다. MobiCAN은 액티브 네트워크 메커니즘을 적용한 오버레이 네트워크 기술이며, MobiCAN 노드와 이를 관리하기 위한 오버레이 관리 프로토콜로 구성된다. MobiCAN 노드는 효과적인 서비스 설치, 실행 및 유지 보수 기능을 제공하고, 서비스 개발이 용이하도록 서비스 계층화 및 커스텀마이즈 기능을 제공한다. 특히, 마이크로 서비스는 노드의 기본적인 동작 단위이며 서비스 계층화 기능과 규격화된 인터페이스 함수를 제공한다. 그리고, 이러한 오버레이 네트워크를 제어 및 관리하기 위하여 OMPs를 제안한다. 이 프로토콜은 노드 정보 동기화를 위한 NISP, 오버레이 정보 절의를 위한 NIQP 및 노드 설정 및 오버레이 관리를 위한 NCMP로 구성된다. 그리고 마지막으로, MobiCAN의 실용성을 검증하기 위하여 구현된 동적 콘텐츠 최적화 프록시(Dynamic Contents Customization Proxy : DCCP) 서비스에 대하여 소개한다.

키워드 : 액티브 네트워크, 무선 인터넷, 콘텐츠 기반 라우팅, CC/PP, 오버레이 네트워크

Abstract Mobile contents service providers have some difficulties to timely and proper service deployment due to rapid development cycle of diverse portable devices with different capabilities. A way to resolve the problem is to introduce a mobile contents service platform that can adapt original mobile contents to diverse devices dynamically and automatically. In this paper, we propose a mobile contents service platform based on active network mechanisms, called Mobile Content Adaptation Network(MobiCAN). The MobiCAN node provides effective service deployment, execution, and maintenance features and accommodates service layering and service customization capabilities for easy deployment. The basic functional units of the MobiCAN node are micro services with well-defined service interfaces and service layering features. For reliable services among the MobiCAN nodes, we design new distributed and robust Overlay Management Protocols(OMP). As an

* 본 연구는 한국과학재단 목적기초연구(R01-2003-000-10562-0) 지원으로 수행되었음

[†] 비 회 원 : SK텔레텍연구소 연구원
kjkim@skteletel.com

^{**} 비 회 원 : 경북대학교 컴퓨터학과
jhlee@ccmc.knu.ac.kr

^{***} 종신회원 : 경북대학교 컴퓨터학과 교수

kslim@ccmc.knu.ac.kr

^{****} 비 회 원 : 한국전자통신연구원 정보보호연구단 연구원
seunghee5@etri.re.kr

tynam@etri.re.kr

swsohn@etri.re.kr

논문접수 : 2003년 7월 30일

심사완료 : 2004년 3월 23일

example of practical MobiCAN applications, we finally describe Dynamic Contents Customization Proxy(DCCP) service.

Key words : Active Network, Wireless Internet, Contents Based Routing, CC/PP, Overlay Network

1. 서 론

효과적인 무선 인터넷 서비스를 위한 기존에 많은 연구에서 제안된 다양한 프로토콜, 서비스 아키텍처 및 플랫폼들은 이동 단말의 열악한 환경과 무선 네트워크의 다양성에 따른 문제점들을 상당부분 개선하였다. 그러나, 현재 무선 콘텐츠 서비스에 대한 요구가 지속적으로 증가하고 있고, 이동 단말의 짧은 개발 주기와 각 단말 간의 성능 차이가 커지면서 단말의 환경에 따라 독립적인 콘텐츠 서버를 구축하는 기존의 고비용 서비스 플랫폼은 시의 적절한 콘텐츠 서비스를 제공하는 데는 많은 한계점을 노출하고 있다. 그러므로, 공용의 콘텐츠 서버로부터 단말에 콘텐츠가 전달되는 경우 네트워크 상에서 단말의 환경에 최적화된 상태로 콘텐츠를 동적으로 변환시키는 메커니즘이 이러한 고비용 구조를 완화시킬 수 있는 이상적인 접근 방법이 될 수 있다. 이러한 메커니즘을 적용한 기존 연구로는 IETF의 Open Pluggable Edge Services(OPES)[1]가 있다. OPES는 콘텐츠 서버와 클라이언트를 연결하는 경우 네트워크에서 콘텐츠를 단말에 적합하게 수정할 수 있는 서비스를 제공하기 위한 프레임워크이다. 그러나, 이 모델은 이러한 서비스를 효과적으로 지원하기 위하여 다양한 응용 프로그램을 OPES 엔터티에 설치, 갱신 및 제어하기 위한 기능이 제공되지 않는 문제점이 있다.

이처럼, 콘텐츠를 단말의 환경에 적합하도록 변환하기 위해서는 단말과 콘텐츠 서버 사이에 무선 인터넷 콘텐츠 서비스를 위한 새로운 형태의 전용 플랫폼이 필요하다. 이 플랫폼은 콘텐츠를 효과적으로 처리할 수 있는 소프트웨어를 쉽고 빠르게 설치 및 실행할 수 있는 환경을 제공해야 하며, 이를 구현하기 위하여 동적으로 네트워크 프로토콜을 설치하기 위한 기술인 액티브 네트워크 메커니즘을 활용하는 것이 유력한 대안이 될 수 있다. 그러므로, 무선 콘텐츠 서비스를 지원하기 위한 플랫폼에서 서비스의 개발 및 실행을 담당하기 위한 노드는 기존의 두 가지 대표적인 액티브 노드 모델인 DARPA[2,3] 모델과 P1520[4] 모델의 장점을 결합한 구현 모델을 활용하여 설계하는 것이 적절하다. DARPA 모델은 노드운영체제, 실행환경 및 액티브 응용 환경으로 구성되며, 서비스의 실행과 동적 배포를 위한 구체적인 메커니즘을 제공하고 있다. 그리고, P1520 모델은 네트워크 소프트웨어 모듈성, 재사용성 및 확장성을 높이기 위하여 잘 정의된 인터페이스를 이용함으

로써 서비스 계층화 및 서비스 커스텀이즈 기능을 제공한다[5]. 그리고, 이러한 액티브 노드들을 유기적으로 연동하여 사용하기 위해서는 '오버레이 네트워크 관리 메커니즘이 필요하며, 이에 대한 대표적인 연구로는 X-Bone과 그리드를 이용한 다양한 방법이 있다. 그러나, X-Bone을 이용한 방법에서는 오버레이 상의 리소스 정보를 수집하는데 멀티캐스트를 이용하므로 오버헤드가 크며, Grid를 이용한 방법에서도 마찬가지로 Grid Index Information Service(GIIS)가 수직적인 구조를 가지고 분산 처리를 제공하지 않으므로 GIIS 서버가 정지하거나 또는 비정상적인 질의가 GIIS에 집중될 경우에는 전체 서비스가 정지되는 문제가 발생한다. 그러므로, 무선 콘텐츠 서비스를 위한 플랫폼을 효과적으로 운용하기 위해서는 이러한 두 가지 문제점과 함께 서비스 수행 중에 액티브 노드에서 동적으로 발생할 수 있는 예외적인 상황을 효과적으로 처리함으로써 오류에 강한 관리 메커니즘이 필요하다.

본 논문에서는 무선 콘텐츠 서비스를 효과적으로 제공하기 위하여 다음과 같은 요구 사항들을 만족하는 액티브 네트워크 기반 서비스 플랫폼인 무선 콘텐츠 적응형 네트워크(Mobile Contents Adaptation Network : MobiCAN)를 제안한다. 먼저, MobiCAN에서 수행되는 서비스의 기본 실행 단위는 마이크로 서비스이며, 각각의 마이크로 서비스는 독립적으로 수행된다. 특히, MobiCAN을 위한 전용 액티브 노드인 MobiCAN 노드는 DARPA 모델을 참조하여 효과적인 마이크로 서비스의 배포, 실행 및 유지 보수 기능을 제공하고, P1520 모델에서 서비스 개발 용이성을 높이기 위하여 사용되는 서비스 계층화 및 커스텀이즈 기능을 수용한다. 그리고, 전용 노드는 기존의 무선 인터넷 서비스들과 연동하기 위하여 TCP/IP 모든 기능을 지원하며, MobiCAN에서 서비스를 효율적이고 안정적으로 제공하기 위해서는 오류 발생에 효과적으로 대처할 수 있는 분산형 관리 프로토콜을 지원한다. 이 프로토콜은 노드 정보를 오버레이 상에서 동기화시키고, 노드 정보 및 오버레이 정보를 질의하고, 동적인 서비스 설치와 실행 그리고, 오버레이 상에서 서비스가 수행될 때 발생하는 예외적인 상황을 처리할 수 있는 기능을 제공해야 한다. 마지막으로, 동적으로 배포되는 서비스와 노드에 전달되는 액티브 패킷을 안전하게 처리하기 위한 보안 메커니즘이 필요하다.

제2장에서는 이러한 요구 사항을 기반으로 설계된 MobiCAN 구조에 대하여 개략적으로 설명하고, MobiCAN을 구성하는 MobiCAN 노드의 구현 구조와 오버레이 네트워크 관리를 위한 Overlay Management Protocols(OMPs)에 대하여 설명한다. 제3장에서는 MobiCAN의 실용적인 적용 사례로써 동적 콘텐츠 최적화 프록시(Dynamic Content Customization Proxy : DCCP) 서비스를 소개하며, 마지막으로 결론 및 향후 연구 방향을 기술한다.

2. MobiCAN 개요 및 구조

2.1 개요

MobiCAN 플랫폼은 콘텐츠 변환, 캐싱 및 콘텐츠 기반 라우팅 등의 다양한 마이크로 서비스들을 동적으로 설치하고 실행할 수 있는 기능을 제공함으로써, 공용의 콘텐츠 서버로부터 전달된 콘텐츠를 이동 단말에 효과적으로 최적화시킬 수 있는 환경을 제공한다. 그리고, MobiCAN 노드가 TCP/IP 전 기능을 지원하므로 콘텐츠 서버와 단말 간의 연결을 가로챌 수 있는 마이크로 서비스를 개발함으로써, 기존의 무선 인터넷 콘텐츠 서비스 기술을 수정 없이 활용할 수 있다. 이러한 MobiCAN 플랫폼에서의 콘텐츠 최적화 서비스는 단말에서 전달된 장치 환경 정보와 콘텐츠로부터 추출된 콘텐츠 정보를 기반으로 콘텐츠 최적화를 위한 다양한 마이크로 서비스들이 유기적으로 동작함으로써 제공된다. 제안된 플랫폼에서는 이처럼 다양한 마이크로 서비스들 간의 연동 기능을 제공하기 위하여, MobiCAN 노드에서 서비스 계층화 및 커스텀마이즈 기능을 제공하도록 설계 한다. 이 외에도, 인터넷상에서 분산되어 설치된 노드들로 구성된 오버레이 네트워크를 효과적으로 관리하기 위하여 OMPs 프로토콜이 사용된다. 이 프로토콜은

오버레이 네트워크 상에서 노드 정보의 동기화, 질의처리, 노드 설정, 서비스 실행 및 예외 상황을 처리할 수 있는 메커니즘을 제공함으로써, 안정적이고 효율적인 무선 콘텐츠 전용 서비스 플랫폼이 구축 될 수 있도록 한다.

2.2 MobiCAN 노드 구현 구조

2.2.1 노드운영체제와 실행환경

그림 2는 DARPA 모델을 기반으로 실행환경과 노드 운영체제로 구성된 MobiCAN 노드 구조를 간략하게 나타낸 것이다[2,3]. 노드운영체제는 실행환경과 다양한 하드웨어 장치를 중재하는 역할을 수행하며, 하드웨어 및 커널 독립성, 서비스 동적 배포와 실행을 위한 코드 이동성 그리고 서비스 커스텀마이즈를 위한 소프트웨어 재사용성을 제공하기에 적합한 자바 플랫폼을 이용하여 설계하였다. 특히, 노드운영체제가 제공하는 기능은 인터페이스 함수를 통하여 공개되고, 실행환경의 다양한 컴포넌트들과 마이크로 서비스들은 이 함수를 이용하여 구현된다. 그리고, 실행환경은 DARPA 모델의 대표적인 실행환경 구현 모델 중에서 분리 방식을 활용한다.

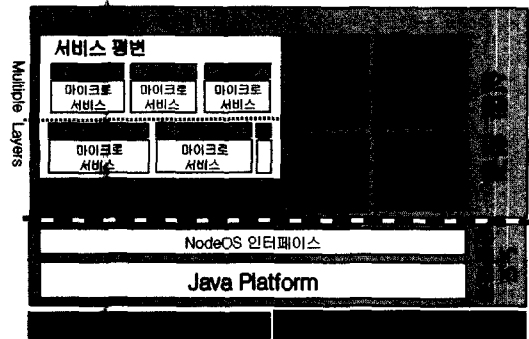


그림 2 MobiCAN 노드 구조

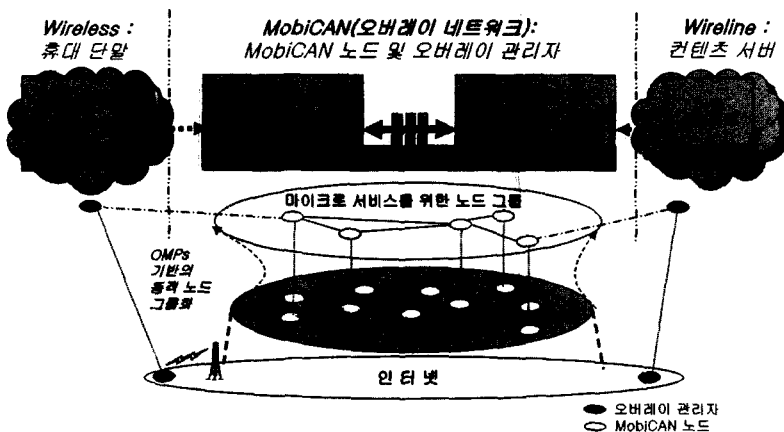


그림 1 MobiCAN 플랫폼 구조 개요

DARPA 모델의 대표적인 실행환경 구현 모델에는 통합 방식과 분리방식이 있다. 통합방식은 각 패킷에 프로그램이 포함되어 전달되기 때문에 불필요한 트래픽을 유발하고, 일반적으로 프로그램의 크기가 제한되어 있기 때문에 크고 복잡한 서비스를 개발하는데 적합하지 않다[2]. 그러나, 분리방식은 액티브 패킷 내부에 서비스에 대한 식별자만을 포함하여 전달하고 실제 프로그램은 노드의 로컬 영역에 저장하여 사용함으로써, 효율적으로 대역폭을 사용하고 프로그램 크기에도 제한이 적기 때문에 다양한 서비스 개발에 효과적이다[2,3]. 그러므로, 콘텐츠 변환, 라우팅 서비스와 같은 비교적 크고 복잡한 서비스를 제공하는 MobiCAN에서는 분리방식이 적합하다. 뿐만 아니라, 실행환경은 소프트웨어 재사용성 및 개발의 용이성을 높이기 위하여 서비스 계층화 기능을 제공한다. 이러한 계층화 기능은 P1520 모델에서 제안하는 계층간 인터페이스 함수 정의를 통한 계층간 통신 모델을 활용하여 제공된다.

실행환경은 액티브 서비스 실행을 담당하는 서비스평면과 서비스평면을 제어하는 제어평면으로 구분된다. 서비스평면은 다양한 마이크로 서비스가 실행되는 가상적 공간으로 서비스 계층화 및 커스트마이즈를 지원한다. 서비스 계층화 기능은 복잡한 서비스를 여러 개의 마이크로 서비스로 분리함으로써 모듈화 및 간결성을 높이며, 커스트마이즈 기능은 마이크로 서비스의 응용 프로그래밍 인터페이스(Application Programming Interface : API)를 정의함으로써 재사용성 및 확장성을 높여준다. 이러한 기능들은 개발자가 기존의 다양한 마이크로 서비스들을 이용하여 새로운 서비스를 쉽게 개발하고, 기존의 인터넷 서비스들과 쉽게 인터워킹 할 수 있도록 한다. 그리고, 제어평면은 이러한 서비스평면을 관리하는 기능을 수행하며 다음과 같은 컴포넌트들로 구성된다. 먼저 서비스 아카이브 저장소는 오버레이 관리자에 의하여 설치된 서비스를 저장 및 접근할 수 있는 기능을 제공하며, 실행기는 관리자에 의하여 활성화된 마이크로 서비스를 서비스 저장소로부터 로딩하여 실행하는 역할을 수행한다. 그리고, OMPs 처리기는 노드 정보 동기화, 질의 처리, 오버레이 네트워크의 설정 및 예외 상황 처리에 대한 역할을 수행하며, 노드 상태 모니터링은 MobiCAN 노드의 컴퓨팅 능력, 부하량, 대역폭 등의 상태 정보를 수집하고 검사하는 역할을 수행한다. 마지막으로 보안 관리자는 마이크로 서비스 배포를 안전하게 수행하고 노드로 전달되는 패킷을 안전하게 처리하기 위한 보안 기능을 제공한다.

2.2.2 서비스 계층화 및 커스트마이즈

MobiCAN 노드는 서비스 계층간 인터페이스를 정의함으로써 인터페이스 호출을 통한 서비스 계층화 및 커

스트마이즈 기능을 제공할 수 있는 P1520 모델을 수용하고 있다. 그러나, P1520 규격에서는 구현을 위한 구체적인 모델을 제시하고 있지 않으므로, 대표적인 프로토콜 구현 모델인 서버 모델과 액티브 스레드 모델을 참조하여 설계한다[9,10]. 서버 모델은 계층화된 서비스들을 독립적으로 실행하기에 적합한 구조이며, 액티브 스레드 모델은 기존의 서비스를 쉽게 사용할 수 있는 함수 호출 메커니즘을 제공한다. 그러므로, 본 제안 모델에서는 계층화된 마이크로 서비스들이 독립적으로 실행될 수 있는 서버 모델을 기본 설계 모델로 활용한다. 그러나, 이 모델은 계층간 연동을 위하여 메시지 통신 방법을 이용하므로 서비스 커스트마이즈 기능을 제공하기에는 적합하지 않고 개발자가 사용하기에도 용이하지 않다. 이러한 문제점을 보완하기 위한 방법으로 액티브 스레드 모델을 활용한다. 이 모델은 계층간 통신에 함수 호출 방식을 이용하므로 P1520 모델과 유사하고, 서비스를 커스트마이즈하고 개발자가 사용하는데 있어서 서버 모델의 메시지 전달 방식에 비하여 상대적으로 용이하다는 특징이 있다.

결과적으로 본 절에서는 이러한 두 가지 프로토콜 개발 모델을 참조하여 그림 3과 같은 서비스 계층화 및 커스트마이즈 모델을 제안하며, 본 논문에서는 이 모델을 서버 스레드 모델이라 정의한다. 서버 스레드 모델의 주요 구성요소는 다음과 같다. 먼저, APIs는 서비스 제공자의 기능을 사용할 수 있는 인터페이스이며, 서비스 제공자와 서비스 사용자간의 함수 호출이나 타이머 만료를 처리하기 위하여 각각에 대한 정보를 재구성한 자료구조인 Message가 사용된다. 그리고, Event는 Message와 일대일 대응 관계에 있으며, 타이머 만료, 입출력 메시지 큐로의 메시지 입력 등과 같은 상태 변화 정

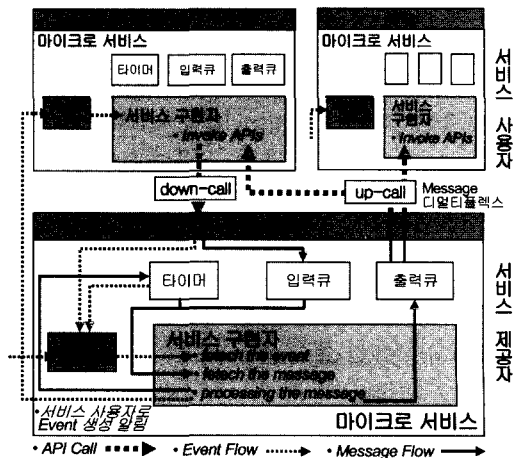


그림 3 서버 스레드 모델

보를 표현하는데 사용된다. 입·출력 및 이벤트 큐는 *Event* 또는 *Message*가 처리되기 전까지 일시적으로 저장하는 역할을 수행하며, 타이머는 시간과 관련된 작업 일정을 관리하고 그 작업들의 만료를 알리는 역할을 수행한다. 마지막으로 서비스 구현자는 마이크로 서비스의 기능을 실제로 구현하는 부분으로써, 이벤트 큐로부터 *Event*를 출력하여 이와 대응관계에 있는 *Message*를 이벤트 트리본 방식으로 처리한다. 이벤트 트리본 방식은 이벤트가 없을 경우에는 수행이 정지되고, 이벤트가 발생하면 이를 비동기적으로 처리하는 방식을 의미한다. 부가적으로, *APIs*는 *down-call*과 *up-call* 두 가지 형식으로 구분할 수 있으며, *down-call*은 메시지를 서비스 사용자에서 서비스 제공자로 방향으로 전달하고 *up-call* 형식은 이와 반대 방향으로 전달한다. 특히, *up-call*의 경우에 서비스 사용자는 서비스 제공자의 출력 큐에 저장된 메시지 정보를 처리할 수 있는 *API*를 구분할 수 없기 때문에, 이를 식별할 수 있는 부가적인 인터페이스 함수를 제공한다.

2.3 OMPs

MobiCAN에서 오버레이 관리자들은 도메인 관리자가 지정한 이웃한 오버레이 관리자들과 양방향 유니캐스트 터널을 생성함으로써 관리용 메시 토폴로지를 구성하고, OMPs는 이렇게 구성된 오버레이 관리자들이 MobiCAN 노드들로 구성된 오버레이를 관리하는데 사용된다. 이 프로토콜은 오버레이 상에서 노드 정보를 동기화하기 위한 Node Information Synchronization Protocol(NISP), 질의를 처리하기 위한 Node Information Query Protocol(NIQP), 그리고 오버레이 상의 각 노드들을 설정하고 마이크로 서비스가 수행되는 동안에 발생하는 예외적인 상황을 처리하기 위한 Node Configuration and Management Protocol(NCMP)로 구성된다. 본 단락에서는 각 프로토콜의 목적과 개략적인 동작만을 기술하며, 이때 사용되는 몇 가지 용어를 아래와 같이 정의한다. 먼저, Autonomous System(AS)는 단일 관리자의 관리 하에 있는 오버레이 네트워크를 의미하며, 오버레이 관리자(Overlay Manager : OM)는 OMPs를 수행하는데 중심적인 역할을 수행하는 시스템을 의미한다. 그리고, 최상위 오버레이 관리자(root OM)는 자율 시스템 내에서의 대표적인 노드 관리자이며, 자율 시스템 간의 접근점 역할을 수행하는 시스템을 의미한다.

2.3.1 NISP 프로토콜

NISP는 오버레이 네트워크 상에서 MobiCAN 노드 정보를 수집하고 동기화하며 다음과 같은 네 가지 기능으로 구분된다. 등록 및 해지 단계에서는 노드가 MobicAN 오버레이 네트워크에 참가 및 탈퇴하기 위한

기능을 제공한다. 동기화 단계는 노드 관리자들이 메시 상에서 노드 정보를 거리 벡터 라우팅 알고리즘[11]에 따라 포워딩함으로써, 오버레이 상에서 노드 정보를 동기화하는 역할을 수행한다. 그리고, 적응 단계는 오버레이 관리자의 일시적인 서비스 정지와 같은 예외적인 상황에서 관리용 메시 토폴로지를 재구성하고 정지된 오버레이 관리자에 등록되었던 고아 노드들을 오버레이 네트워크에 재가입 시킨다. 그리고, 이 단계에서는 오버레이 관리자가 재가동되면 기존의 토폴로지로 복귀하는 기능을 제공함으로써 예외적인 상황에서도 정상적인 기능을 수행할 수 있는 적응력을 높여준다.

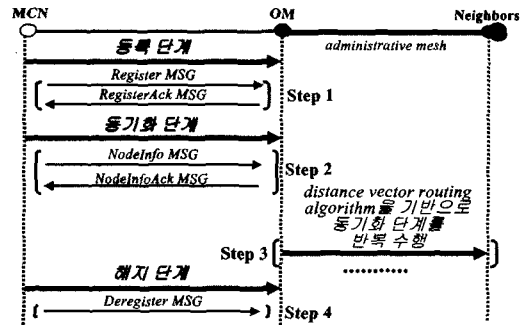


그림 4 NISP 주요 기능

그림 4에서 MCN은 MobiCAN 노드이며 OM은 오버레이 관리자 그리고, 메시 상에서 OM의 이웃한 오버레이 관리자들은 Neighbors로 표기하였다. 위 그림은 NISP에서 주요한 기능인 노드의 등록, 해지 및 동기화에 대하여 간략하게 나타낸 것이며 각 기능별 세부 절차를 간략히 설명하면 아래와 같다.

Step 1. 등록 단계: MCN은 오버레이에 참가하기 위하여 노드의 식별자와 인증자를 포함한 Register 메시지를 OM에 전달하고, 이 오버레이 관리자는 RegisterAck 메시지로 응답함으로써 등록 과정이 완료된다. 이러한 등록 단계를 거친 이후에 OM은 MCN의 *parent*가 되며 MCN은 OM의 *child*가 된다.

Step 2. 동기화 단계: 등록이 완료되면 MCN은 저장된 자신의 노드 정보를 주기적으로 갱신하기 위하여 OM에 NodeInfo 메시지를 전달하며 오버레이 관리자는 NodeInfoAck 메시지로 응답한다. NodeInfo 메시지는 노드 식별값, 최근 갱신 시간, 포워딩 홉 수, 포워딩 경로, 설치된 서비스 패키지, 활성화된 서비스 패키지, 인증 정보 등을 포함한다. 이때, 포워딩 홉 수는 초기 값을 16으로 하고 노드 관리자를 지날 때 마다 1 씩 감소하여 0이 되면 버려짐으로써 무한 루프를 방지하며, 포워딩 경로 정보는 적응 단계에서 메시 토폴로지를 재구

성하고 고아 노드들을 오버레이에 재등록 시키는데 사용된다. 또한, NodeInfo 메시지는 동기화 단계의 성능 향상을 위하여 갱신된 정보만을 전달할 수 있으며, 오버레이 관리자는 현재 자신의 정보를 갱신하기 위하여 강제적으로 노드에 동기화를 요청할 수도 있다.

Step 3. 동기화 단계: 오버레이 관리자들은 *child* 노드들에 대한 정보를 수집하고, 거리 벡터 라우팅 알고리즘을 기반으로 인접한 관리자들에게 주기적으로 Node-Info 메시지를 포워딩 함으로써 갱신된 최신 노드 정보를 오버레이 네트워크 상에 알린다.

Step 4. 해지 단계: MCN은 노드의 식별자와 인증자 정보를 포함한 Deregister 메시지를 전달함으로써, 오버레이 네트워크와의 등록을 해지한다.

2.3.2 NIQP 프로토콜

NIQP 프로토콜은 노드 관리자에 저장된 MobiCAN 노드 정보를 요청하기 위한 질의 기능을 제공하며, 데이터 베이스 정보에 접근하기 위한 언어인 SQL 언어와 유사한 조건 검색 형태이다. 그리고, 각 노드는 질의 정보를 로컬에 캐싱하여 불필요한 네트워크 트래픽과 노드 관리자의 부하를 줄여주며, 질의 처리 기능을 통하여 노드의 프로세싱 부하, 대역폭 등의 노드 상태 정보를 알려주는 기능도 부가적으로 제공한다. 또한, 이 프로토콜은 질의가 처리되는 방식에 따라서 직접질의와 간접질의 두 가지로 구분되며, 직접질의는 자율 시스템 내부에서 오버레이 네트워크 정보를 획득하는데 사용되고 간접질의는 자율 시스템간의 질의를 처리하기 위하여 사용된다.

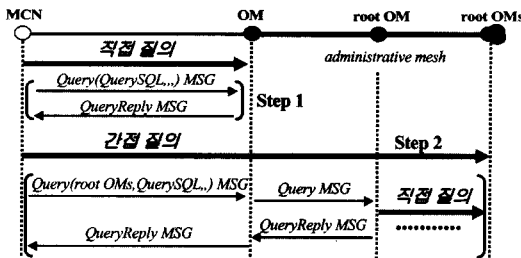


그림 5 NIQP 프로토콜 주요 기능

그림 5에서는 직접질의 및 간접질의 동작을 간략하게 표현하였으며, 위 그림에 표현된 질의 처리 과정을 설명하면 아래와 같다.

Step 1. 직접질의: MCN은 노드에 대한 정보를 획득하기 위하여 질의 명령문인 QuerySQL를 포함한 Query 메시지를 OM에 전달하며, 오버레이 관리자는 질의 결과를 포함한 QueryReply 메시지를 이용하여 이에 응답한다. 그리고, 이러한 직접질의는 노드의 상태

정보를 획득하기 위하여 목적 노드에 직접 요청되기도 한다.

Step 2. 간접질의: MCN은 외부 자율 시스템의 노드 정보를 획득하기 위하여, root OM에 대한 리스트와 QuerySQL 문을 포함한 Query 메시지를 OM에 전달한다. 이렇게 전달된 메시지는 최상위 노드 관리자로 포워딩되며, 최상위 노드 관리자는 이 정보를 이용하여 각각의 최상위 노드 관리자에 직접질의를 수행한다. 이렇게 수집된 질의 결과는 QueryReply 메시지로 재구성되고, 이 메시지는 목적 노드를 담당하는 노드 관리자를 경유하여 MCN으로 전달된다.

2.3.3 NCMP 프로토콜

NCMP 프로토콜은 자율 시스템 내에서 마이크로 서비스를 설치, 제어 및 서비스 수행 중에 발생하는 예외 상황을 처리하기 위한 기능을 제공하며, 오버레이 관리 (Overlay Administration Protocol : OAP) 프로토콜과 가상 네트워크 제어 (Overlay Control Protocol : OCP) 프로토콜로 구성된다. OAP는 오버레이 네트워크 상에서 마이크로 서비스의 설치, 삭제, 활성화 및 비활성화를 위한 기능을 제공한다. 설치 과정에서는 마이크로 서비스 아카이브를 각 MobiCAN 노드에 다운로드하고 이를 서비스 레지스트리에 등록하는 것을 의미하며, 활성화 과정은 설치된 마이크로 서비스를 로딩하여 실행시키는 것을 의미한다. 그 외, 마이크로 서비스의 삭제나 비활성화는 각각 설치와 활성화에 반대되는 의미를 가진다. 특히, OAP 메시지는 오버레이 관리 메시 상에서 목적 노드의 포워딩 정보를 이용하여 포워딩되며, 노드 관리자들은 이 메시지를 최종 목적 노드에 전달하여 메시지에 포함된 관리 명령이 수행 될 수 있도록 한다.

그리고, MobiCAN에서는 마이크로 서비스가 수행되는 과정에서 여러 가지 예외적인 상황이 발생 할 수 있다. 예를 들면, MobiCAN 노드가 서비스를 중지하거나 또는 마이크로 서비스가 삭제된 경우에 이러한 정보가 모든 오버레이 네트워크 상에서 동기화되는 데는 상당한 시간이 소요된다. 그러므로, 일부 노드들은 일시적으로 잘못된 정보 때문에 처리될 수 없는 메시지를 수신할 수 있다. 이러한 예외적인 상황을 처리하기 위하여 OCP가 동작한다.

그림 6은 MobiCAN에서의 마이크로 서비스 설치, 삭제, 활성화 및 비활성화를 위한 OAP 프로토콜의 동작 중에서 설치 과정만을 간략히 나타낸 것이다. 그 외 삭제, 활성화 및 비활성화 과정은 설치 과정의 Step 1과 동일한 과정으로 수행된다.

Step 1. 설치 명령 전달: 네트워크 관리자는 서비스 설치 명령 식별자, 서비스를 설치할 노드 주소, 마이크로 서비스 아카이브를 획득할 수 있는 코드서버의 URL

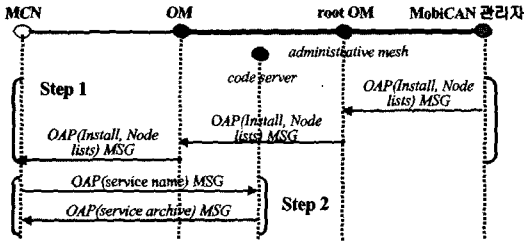


그림 6 OAP 주요 동작

주소 및 인증 정보를 포함한 OAP 메시지를 root OM에 전달한다. 최상위 노드 관리자는 이 메시지를 포워딩 정보를 이용하여 전달하며, 이 메시지는 목적 노드를 담당하는 노드 관리자를 경유하여 최종 목적지 노드에 전달된다.

Step 2. 코드 다운로드: 목적 노드는 OAP 메시지 내에 포함된 URL 정보를 이용하여 코드 서버 정보로부터 필요한 서비스 패키지들을 다운로드 받아 인증 과정 후에 시스템 내에 등록한다.

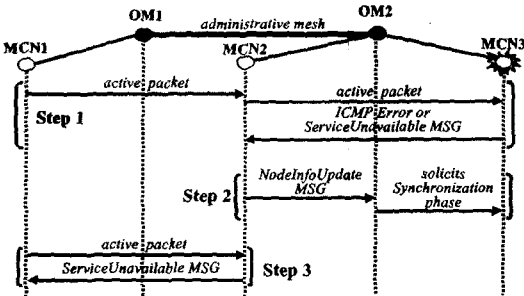


그림 7 OCP 프로토콜 동작

그림 7에서는 잘못된 노드 정보를 지닌 MCN1이 MCN2를 경유하여 목적 노드인 MCN3로 액티브 패킷을 전달한다고 가정 할 때, OCP의 동작 과정을 간략하게 설명하면 아래와 같다.

Step 1. 예외 상황 인지: MCN3가 포워딩된 액티브 패킷을 처리할 수 없는 경우에, MCN2는 Service-Unavailable 메시지나 또는 ICMP 에러 메시지를 수신하게 됨으로써 현재 오버레이 상에서 MCN3에 대한 노드 정보가 유효하지 않음을 알 수 있다.

Step 2. 동기화 요청: 이러한 예외적 상황을 감지한 MCN2 노드는 NodeInfoUpdate 메시지를 MCN3의 parent인 OM2로 전달하여 노드 정보 동기화를 요청함으로써 갱신된 노드 정보가 전체 오버레이 네트워크 상에서 신속히 갱신 될 수 있도록 한다.

Step 3. 예외 상황 처리: 만약 MCN2가 MCN3를 목적지로 하는 액티브 패킷을 재차 수신하게 되면, 이

노드는 액티브 패킷을 포워딩 한 이전 노드로 Service-Unavailable 메시지를 전달하고 Step 2 과정이 반복되도록 한다. 결과적으로, 처리될 수 없는 액티브 패킷을 생성하는 MCN1은 ServiceUnavailable 메시지나 오버레이 관리자의 정보 갱신 과정에 의하여 이러한 예외 상황을 인지하고 더 이상 처리될 수 없는 액티브 패킷을 생성하지 않게 된다.

3. MobicAN 응용 사례

MobicAN은 동적 서비스 설치, 서비스 계층화 및 커스텀이즈 기능을 제공함으로써, 무선 인터넷 콘텐츠 서비스를 제공하는데 많은 제약점이었던 높은 개발 및 설치 비용을 효과적으로 줄여준다. 본 절에서는 이러한 서비스 플랫폼을 검증하기 위하여 구현된 Composite Capability/Preference Profile(CC/PP)[12], Contents Based Routing(CBR), Content Proxy(CP) 및 Wireless Proxy(WP) 마이크로 서비스와 이러한 서비스들을 커스텀이즈하여 단말에 적합한 콘텐츠를 동적으로 생성하는 DCCP 서비스에 대하여 기술한다. 먼저, CC/PP는 W3C에서 클라이언트의 다양한 하드웨어 및 소프트웨어 환경을 기술하기 위하여 제안한 XML 기반의 표준으로, CC/PP 마이크로 서비스는 이동 단말의 최신 환경 정보를 획득하기 위한 서비스를 제공한다. CBR 마이크로 서비스는 XML 기반의 라우팅 디스크립션을 기반으로 액티브 패킷을 적절하게 처리해 줄 수 있는 노드로 패킷을 포워딩하는 기능을 제공한다. CP 마이크로 서비스는 콘텐츠 변환 및 캐싱 기능을 이용하여 단말의 환경에 적합한 콘텐츠를 동적으로 생성하는데 사용되며, WP 마이크로 서비스는 단말과 웹 서버 사이에서 발생하는 웹 트랜잭션을 가로채어 MobicAN에서 처리될 수 있도록 한다. 특히, WP 서비스는 HTTP 표준에 따라 동작함으로써, 단말과 웹 서버는 기존의 동작 메커니즘을 수정 없이 사용하는 투명성을 제공한다.

그림 8은 다양한 마이크로 서비스를 조합하여 DCCP 서비스를 간략하게 표현한 것이다. 위 그림에서는 MobicAN 노드를 EN_i로 표기하고, EN_i에서의 WP와 CP 마이크로 서비스는 각각 WP_i와 CP_i로 표기한다. 특히, 위 그림에서 CP_i는 캐싱 기능을 제공함을 가정한다. DCCP 서비스는 요구 단계와 응답 단계로 구분하여 설명할 수 있으며, 개략적인 동작 과정은 아래와 같다.

먼저, 요구 단계에서는 이동 단말과 웹 서버 사이에서 발생하는 웹 트랜잭션을 WP₁이 가로채며, 트랜잭션을 요구한 단말의 환경 정보와 웹 세션 정보를 수집하여 관리한다. CP₁은 WP₁으로부터 전달받은 웹 요구 메시지를 액티브 패킷으로 변환하고 이를 포워딩하기 위한 라우팅 디스크립션을 생성한 이후에, CBR 마이크로 서

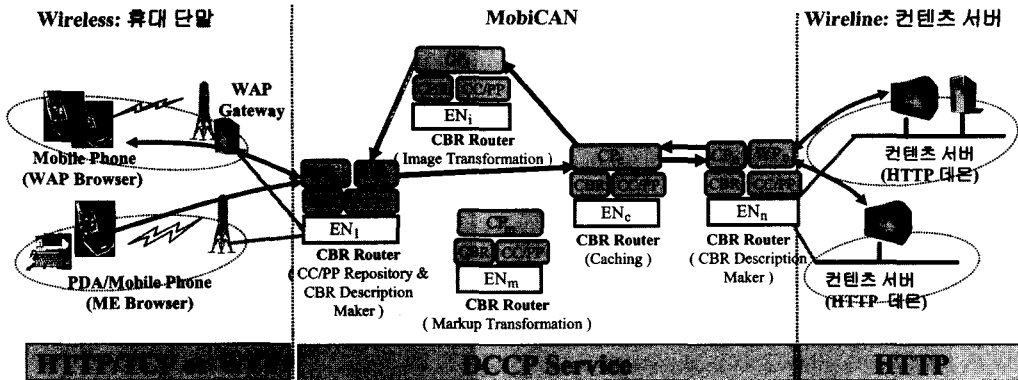


그림 8 MobiCAN 응용 사례 : DCCP 서비스

비스와 오버레이 관리자에 저장된 노드 정보를 이용하여 포워딩을 수행한다. 액티브 패킷은 라우팅 디스크립션에 따라 여러 가지 서비스가 적용된 이후에 WP_n 으로 전달되고, 이 서비스는 원본 웹 트랜잭션 메시지를 웹 서버로 전달한다. 만약, 액티브 캡슐이 CP_c 를 경유하는 과정에서 목적 콘텐츠가 캐싱된 경우에는 즉시 응답 단계로 전이하게 된다. 응답 단계에서 WP_n 은 웹 서버로부터 전달된 HTTP 메시지를 CP_n 으로 전달하고, CP_n 은 이 메시지를 이용하여 액티브 패킷 및 라우팅 디스크립션을 생성한다. 이렇게 생성된 액티브 패킷은 CBR 마이크로 서비스에 따라 CP_n 에서 CP_1 으로 전달되면서, 액티브 패킷에 포함된 콘텐츠가 목적 단말에 적절한 형태로 변환된다. 결과적으로, WP_1 은 목적 단말에 최적화된 콘텐츠를 수신하게 되고 이를 단말에 전달한다.

본 절에서는 MobiCAN을 활용한 구현 사례로서 DCCP 서비스에 대하여 간략하게 설명하였다. 이 전용 네트워크는 이외에도 응용 계층 멀티캐스트, 콘텐츠 분배 네트워크 서비스, 웹 방송, 메시징 서비스, 무선 TCP, 네트워크 계층 멀티캐스트 서비스 등 콘텐츠 적합화 서비스를 위한 다양한 마이크로 서비스 개발에도 활용할 수 있다.

4. 결론 및 향후 연구

무선 인터넷 환경에서 콘텐츠 서비스를 제공하는 방법은 크게 독립적 콘텐츠 서버 구성, 프록시 서버 활용 및 MobiCAN을 이용하는 방법으로 구분할 수 있다. 먼저, 단말의 환경에 따라 독립적 콘텐츠 서버를 이용하는 방법은 동일한 서비스를 제공하기 위하여 다수의 독자적인 콘텐츠 서버를 구축해야 하므로 서비스 개발 비용이 높고 서비스 관리가 힘들다. 그리고, 프록시 서버를 이용하는 방법에서는 프록시에서 콘텐츠를 단말에 적절하게 변환함으로써 복수의 콘텐츠 서버를 구축할 필요

는 없으나, 프록시 서버에서의 서비스 관리나 부하 집중 문제가 있다. 그러나, MobiCAN에서는 분산 환경 하에서 서비스를 동적으로 설치하고 실행함으로써, 동일한 콘텐츠를 서버를 이용하여 다양한 단말에 서비스를 제공하고 적은 비용으로 서비스를 관리할 수 있는 장점을 제공할 수 있었다.

그리고, 본 논문에서 콘텐츠 적합화 서비스를 위한 효과적인 개발 및 실행하기 위하여 제안한 MobiCAN은 액티브 네트워크 메커니즘을 이용하였으며, 마이크로 서비스의 실행을 담당하는 MobiCAN 노드와 각 노드들로 구성된 오버레이 네트워크를 관리하기 위한 OMPs 프로토콜로 구성된다. 특히, MobiCAN 노드는 DARPA 모델과 P1520 모델의 장점을 참조하여 설계하였으며, 서비스의 실행과 함께 서비스 계층화 기능 및 커스텀마이즈 기능을 제공함으로써 서비스 개발이 용이한 특징이 있다. 그리고, 인터넷상에서 분산되어 설치된 노드들로 구성된 오버레이 네트워크를 효과적으로 관리하기 위하여 OMPs 프로토콜을 제안하였으며, 이 프로토콜은 오버레이 네트워크 상에서 노드 정보를 동기화하기 위한 NISP, 질의처리를 위한 NIQP 그리고 서비스 설치, 활성화 및 예외적인 상황 처리를 위한 NCMP 프로토콜로 구성된다. 마지막으로, 본 논문에서는 이러한 무선 콘텐츠 서비스를 위한 플랫폼인 MobiCAN의 실용성을 검증하기 위한 구현 사례로서 단말과 콘텐츠 서버 사이에서 단말에 적합한 형태로 변환된 콘텐츠를 동적으로 생성해 주는 DCCP 서비스에 대하여 간략하게 소개하였다. 향후에는 콘텐츠 기반 라우팅 기술과 NISP, NIQP 및 NCMP 프로토콜을 세부적으로 설계하고 성능을 검증하고자 하며, MobiCAN 보안 메커니즘 및 QoS 제공 방안에 대한 연구를 수행하고자 한다.

참 고 문 헌

- [1] A. Barbir, R. Chen, M. Hofmann, H. Orman, R. Penno, An Architecture for Open Pluggable Edge Service, 2003, <http://www.ietf.org>
- [2] Stephen F. Bush, Amit B. Kulkarni, Active Networks and Active Network Management, Kluwer Academic/Plenum Publishers, ISBN 0-306-46560-4, 2001.
- [3] Konstantinos psounis, "Active Networks: Applications, Security, Safety, and Architecture," IEEE Communications Surveys, 1999.
- [4] Jit Biswas, Aurel A. Lazar, et al., "The IEEE P1520 Standards Initiative for Programmable Network Interfaces," IEEE Communications Magazine, pp.64-70, October 1998.
- [5] Yu-Shun Wang, Joe Touch, "Application Deployment in Virtual Networks Using X-Bone," Proceedings of the DARPA Active Networks Conference and Exposition(DANCE'02), pp.484-491, 2002.
- [6] Joe Touch, "Dynamic Internet Overlay Deployment and Management Using XBone," Preprint of version in Computer Networks, pp.117-135, July 2001.
- [7] Craig Lee, Eric Coe, J.Matt Clark. Brooks Davis, "Managing Advanced Communication Services Using Active Network Overlays in Grid Environments," Proceedings of the Fourth Annual International Workshop on Active Middleware Services(AMS'02), 2002.
- [8] The Globus Metacomputing Project, <http://www.globus.org>, 1998.
- [9] Liba Svobodova, "Implementing OSI System," IEEE J. Selected Areas Communications, vol. 7, no. 7, pp.1115-1130, September 1989.
- [10] Douglas C. Schidt, "Transport System Architecture Services for High-Performance Communications Systems," IEEE Journal on Selected Areas in Communications, vol. 11, no. 4, pp.489-560, May 1993.
- [11] Robert E. Larson, Corwin S. Low, Paulden Rodriguez, CCNP Routing, pp.107-131, ISBN 1-57610-778-7, The Coriolis Group, 2000.
- [12] Graham Klyne, Franklin Reynolds, et al., Composite Capability/Preference Profile : Structure and Vocabularies, March 2001, <http://www.w3c.org/>



김 기 조

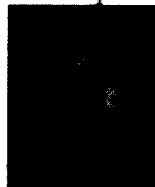
1999년 경북대학교 컴퓨터과학과(이학사). 2002년 경북대학교 컴퓨터과학과(이학석사). 2004년 경북대학교 컴퓨터과학과(이학박사 수료). 2004년~현재 SK텔레텍연구소 연구원. 관심분야는 컴퓨터 통신, 이동 컴퓨팅, 네트워크 보안, 액티

브 네트워크



이 준 호

2002년 경북대학교 컴퓨터과학과(이학사). 2002년~현재 경북대학교 컴퓨터과학과 석사과정. 관심분야는 컴퓨터 통신, 이동 컴퓨팅, 네트워크 보안, 액티브 네트워크



임 경 식

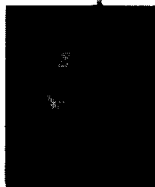
1982년 경북대학교 전자공학과(공학사) 1985년 한국과학기술원 전산학과(공학석사). 1994년 University of Florida 전산학과(공학박사). 1985년~1998년 한국전자통신연구원 책임연구원, 실장. 1998

년~현재 경북대학교 컴퓨터과학과 부교수. 관심분야는 이동 컴퓨팅, 무선 인터넷, 홈 네트워킹, 컴퓨터통신



오 승 희

1999년 전북대학교 컴퓨터과학과 이학사 2001년 이화여자대학교 컴퓨터학과 공학석사. 2001년~현재 한국전자통신연구원 정보보호연구단 네트워크보안구조연구팀 연구원. 관심분야는 정보보호, 차세대 네트워크보안, 액티브 네트워크 등



남 태 용

1987년 충남대학교 계산통계학과 이학사 1990년 충남대학교 대학원 계산통계학과 이학석사. 1987년~현재 한국전자통신연구원 정보보호연구단 개인정보보호연구팀 팀장. 관심분야는 네트워크보안, 액티브 네트워크, 차세대네트워크구조 등



손 승 원

1984년 경북대학교 전자공학과 공학사. 1994년 연세대학교 전자공학과 공학석사. 1999년 충북대학교 컴퓨터공학과 공학박사. 1983년~1986년 삼성전자 연구원. 1986년~1991년 LG 전자(주) 중앙연구소 HI8mm 캠코더 팀장. 1991년~현재

한국전자통신연구원 정보보호연구단 단장. 관심분야는 네트워크보안, 차세대인터넷, 액티브 인터넷 등