

유비쿼터스 환경에서 콘텐츠 적응화를 위한 CC/PP기반의 유비쿼터스 프로파일 및 운영 아키텍처 설계

김 경 식^{*} · 이 재 동^{**}

요 약

본 논문에서는 유비쿼터스 환경에서 콘텐츠 적응화를 위한 CC/PP기반의 유비쿼터스 프로파일 및 운영 아키텍처를 설계한다. 유비쿼터스 프로파일의 컴포넌트와 속성들은 유비쿼터스 환경에서 콘텐츠 적응화 서비스 시나리오를 기반으로 정의하였다. 유비쿼터스 프로파일 설계 시 유비쿼터스 환경에서 효율적으로 사용하기 위해서 RDF, RDF Schema, CC/PP Structure, 표준 어휘 등의 기법들을 적용하였다. 또한, 유비쿼터스 프로파일의 효율적인 운영을 위해 게이트웨이 중심의 운영 방법, 저장소를 이용한 메타데이터 구성 방법, 동적 구성 방법 등의 아이디어들을 기반으로 운영 아키텍처를 제안하고 설계하였다. 제안한 운영 아키텍처에 대한 성능 평가 결과 기존의 프로파일 운영 방법 보다 더 효율적이었다. 제안된 유비쿼터스 프로파일 및 운영 아키텍처는 유비쿼터스 환경에서 사용자에게 적응화된 콘텐츠를 서비스를 위한 연구의 기반을 마련해 주었다.

키워드 : 프로파일, 콘텐츠 적응화, CC/PP, 유비쿼터스

A Design of Management Architecture and Ubiquitous Profile Based on CC/PP for Content Adaptation in Ubiquitous Environment

Lee, Jae Dong[†] · Kim, Kyung Sik^{**}

ABSTRACT

In this paper, the management architecture and the ubiquitous profile based on CC/PP for contents adaptation in the ubiquitous environment is designed. The components and the attributes of the ubiquitous profile are defined based on the services scenario of contents adaptation in the ubiquitous environment. The ubiquitous profile was applied to techniques such as standard vocabulary, CC/PP structure, RDF schema and RDF to use effectively in the ubiquitous environment during the design. Also we design and propose the another management architecture based on ideas such as the dynamic constitution method, the metadata constitution method using a repository and the management method of gateway on center for the effective management of the ubiquitous profile. The result of performance evaluation for the proposed management architecture is more effective compared with the previous management method of the profile. The proposed the ubiquitous profile and the management architecture has provided the foundation of the research for a user to adaptation contents services in the ubiquitous environment.

Key Words : Profile, Content Adaptation, Composite Capability/Preference Profile, Ubiquitous

1. 서 론

정보 통신 기술의 급속한 발달로 인하여 다양한 디바이스(Desktop, Notebook, PDA, Cellular Phone, Smart Phone 등)들이 인터넷에 접속하여 다양한 콘텐츠를 이용할 수 있게 되었다[1]. 그러나 디바이스들은 인터넷에 존재하는 콘텐

츠 중에서 자신의 디바이스 환경에 맞게 또는 비슷하게 제작되어진 콘텐츠만을 이용할 수 있어, 콘텐츠 이용에 한계를 가지고 있다. 콘텐츠 제공 업체들은 해결책으로써 디바이스 명세에 따라 콘텐츠를 별도로 제작하는 방법을 사용하고 있다. 그러나 이 방법은 콘텐츠의 수가 많아지면 제작비용과 유지보수 비용이 기하급수적으로 증가한다는 문제점을 가지고 있다. 또한, 디바이스 장치들이 더욱 다양화되기 때문에 디바이스 명세에 따라 별도로 콘텐츠를 제작하는 방법은 어렵게 될 것이다. 더욱이 디바이스 보급이 증가하면서 사용자들은 하나의 콘텐츠를 다양한 디바이스를 통해 서비스 받기를 원하고 있다. 이러한 문제점을 해결하기 위

※ 이 연구는 한국문화콘텐츠진흥원 문화콘텐츠기술(CT) 연구소 육성 사업 지원(유통 및 서비스 분야)으로 연구되었음.

※ 이 연구는 2005학년도 단국대학교 대학 연구비의 지원으로 연구되었음.

†준 회 원 : 단국대학교 정보컴퓨터학과 박사과정

**정 회 원 : 단국대학교 정보컴퓨터학과 부교수

논문접수 : 2006년 3월 22일, 심사완료 : 2006년 7월 15일

해서 프로파일을 이용한 콘텐츠 적응화 기술이 대두되었다. 프로파일은 디바이스 정보, 네트워크 정보, 사용자 정보 등의 사용자 관련정보를 이용하여 구성되며, 콘텐츠 적응화시 구성된 프로파일을 적용함으로써 사용자들은 디바이스에 상관없이 콘텐츠 서비스 받을 수 있다. 향후에는 프로파일을 이용한 콘텐츠 적응화 기술과 유비쿼터스 기술이 융합되면서 사용자의 감성/상황 정보에 따라 적응화된 콘텐츠를 제공하게 될 것이다. 프로파일을 이용한 콘텐츠 적응화 기술은 프로파일 구성에 따라 콘텐츠가 생성되게 된다. 그러므로 프로파일 구성에 대한 연구는 콘텐츠 적응화를 위한 기반이 되며, 따라서 콘텐츠 적응화를 위한 다른 연구들보다 선행되어야 한다.

그러나 현재 대부분의 콘텐츠 적응화를 위한 프로파일 연구는 디바이스 능력, 네트워크 정보 등 디바이스 중심으로 진행되고 있고, 이를 이용하여 콘텐츠를 적응화하기 때문에 사용자의 감성/상황을 반영한 콘텐츠를 제공할 수 없다. 예를 들어 유비쿼터스 시대에서는 사용자가 존재하는 공간의 감성/상황 정보가 중요한 요소지만, 현재 프로파일 연구에서는 반영되고 있지 않다. 또한 프로파일 운영에 대한 표준의 부재로 인하여 벤더마다 각기 다르게 운영하기 때문에 상호 운용에 있어서의 문제점을 가지고 있으며 향후 통합에 어려움을 가지게 될 것이다.

본 논문에서는 유비쿼터스 환경에서 콘텐츠 적응화를 위해 CC/PP(Composite Capability/Preference Profiles)[14] 기반의 유비쿼터스 프로파일 및 운영 아키텍처를 설계한다. 프로파일 구성 시 필요한 요구사항을 도출하기 위해 유비쿼터스 환경에서 콘텐츠 적응화 서비스에 대해 살펴보고, 콘텐츠 적응화를 위해 갖추어야 할 특징들을 정의한다. 이를 적용한 프로파일을 CC/PP기반으로 디바이스, 상황, 사용자, 콘텐츠, 서비스등 5개의 카테고리로 분류하고 11개의 컴포넌트를 이용하여 구성하였다. 프로파일은 CC/PP의 Structure와 표준 어휘들을 준수하며, 컴포넌트의 속성에 대한 제약사항을 기술하기 위해 RDF Schema[13]를 사용하였고, 메타데이터 기술을 위해 제안된 RDF (Resource Description Framework)[12]를 적용하여 설계하였다. 또한 설계한 프로파일의 효율적인 운영 방안을 위한 게이트웨이 중심의 프로파일 운영 방법, 저장소를 통한 경량 프로파일 구성 방법, 프로파일의 동적 구성 방법 등을 제안하고, 이를 기반으로 프로파일 운영 아키텍처를 설계하고 성능을 평가한다.

본 논문은 다음과 같이 구성된다. 2장에서는 CC/PP 기반의 프로파일 연구들에 대해 알아본다. 3장에서는 유비쿼터스 환경에서 프로파일을 이용한 콘텐츠 적응화에 대해서 알아보고, 4장에서는 유비쿼터스 프로파일을 구성하고 설계한다. 5장에서는 프로파일의 효율적인 운영을 위한 아이디어들을 제시하고, 이를 기반으로 아키텍처를 설계 한다. 6장에서는 제안한 유비쿼터스 프로파일에 대한 분석 및 운영 아키텍처에 대한 성능 평가를 수행한다. 7장에서는 결론 및 향후 연구 방향에 대해 기술한다.

2. 관련 연구

2.1 콘텐츠 적응화를 위한 프로파일

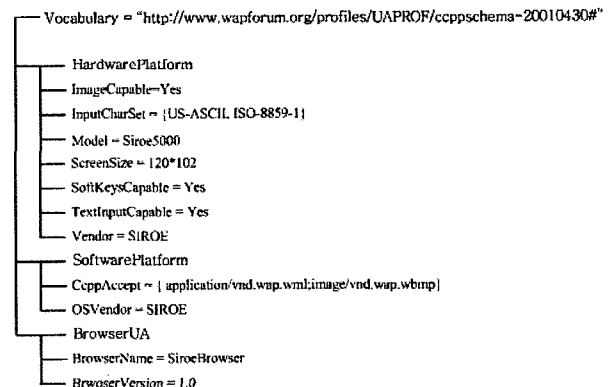
콘텐츠 적응화를 위해 제안된 프로파일로는 W3C[18] Independent Group의 CC/PP가 있다. CC/PP는 사용자 정보 및 선호도 정보를 표현하기 위해 제안 되었으며 RDF를 이용하여 표현한다. CC/PP는 Component, Attributes, Values의 구조로 되어있으며, 각 컴포넌트에 대한 기본 값을 가지고 있는 Default 값들을 가질 수 있다. 이와 같은 CC/PP의 구조는 상호교환, 분해, 균일, 확장에 좋다. 그러나 이러한 단순한 구조는 복잡한 프로파일 구성에는 부적합하며 프로파일을 구성하는 각 속성에 대한 제약사항을 기술할 수 없다.

(그림 1)은 CC/PP의 간단한 예제로서 HardwarePlatform, SoftwarePlatform, Browser로 구성된 프로파일을 나타낸다.

UAProf(User Agent Profile)[15]은 Open Mobile Alliance에서 WAP(wireless application protocol)용 모바일 디바이스를 위해 제안한 프로파일이다. UAProf는 CC/PP를 기반으로 하고 있으며 RDF를 이용하여 표현하였다. UAProf는 WAP(Wireless Application Protocol)용 모바일 단말기에 최적화하기 위해 Hardware Platform, Software Platform, Browser UA, Network Characteristic, WAP Characteristic, Push Characteristic 등으로 컴포넌트를 구성하였다. 각 컴포넌트의 속성에 대한 제약사항을 기술하기 위해 RDF Schema를 이용하고 있다.

그러나 UAProf는 모바일 단말기 기능에 초점을 맞춰 프로파일을 설계하여 다른 디바이스에서 사용하기에는 부적합하며, 이 프로파일을 사용하기 위해서는 단말기 및 Proxy 서버에 맞는 프로파일 처리 모듈이 필요하다.

UPS(Universal Profiling Schema)[5~8]는 INRIA의 Opera 프로젝트의 일부로서 콘텐츠 교섭 및 멀티미디어 콘텐츠 적응화를 위해 제안되었다. UPS는 CC/PP를 기반으로 하고 있으며 RDF를 사용하여 표현하였다. UPS는 Client Profile Schema, Document Instance Profile Schema, Resource Profile Schema, Adaption Method Profile, Client Resource Schema, Network Profile Schema로 컴포넌트를



(그림 1) CC/PP 예제

구성하였다. UPS는 프로파일 구성 시 사용자에 대한 감성/상황 정보, 사용자 선호도 등을 프로파일 구성에 포함하지 않고 있어, 사용자에게 최적화된 콘텐츠를 제공하기 위한 프로파일로는 적합하지 않다.

3. 유비쿼터스 환경에서의 콘텐츠 적응화

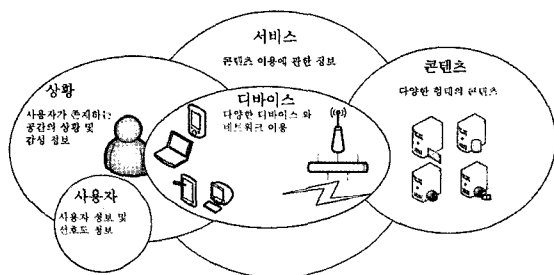
유비쿼터스(ubiquitous)는 “언제 어디서나”, “동시에 존재한다”는 의미로서 최초의 유비쿼터스 컴퓨팅은 제록스 팰러앨토 연구소(PARC)의 마크 와이저(Mark Weiser)에 의해서 제안되었다[2]. 유비쿼터스 환경에서 적응화된 콘텐츠를 사용자에게 제공하는데 영향을 주는 요소를 도출하기 위하여 사용자가 유비쿼터스 환경에서 콘텐츠 서비스를 받고 있다고 가정하고, 이를 이용하여 다음과 같은 사항들을 정의하였다.

- 사용자는 네트워크와 디바이스 종류 등 하드웨어에 독립적으로 콘텐츠를 이용할 것이다.
- 사용자는 자신이 존재하는 공간의 감성/상황정보를 콘텐츠 적응화에 이용할 것이다.
- 사용자는 인터넷에 존재하는 다양한 형태의 콘텐츠를 자유롭게 이용할 것이다.
- 사용자는 자신의 정보와 선호도 정보 등이 반영된 콘텐츠를 이용할 것이다.
- 사용자는 자신의 특성에 따라 개인화된 서비스를 요구할 것이다.

(그림 2)는 유비쿼터스 환경에서 콘텐츠 적응화를 위해 필요한 요소들을 반영한 환경을 나타내고 있다.

유비쿼터스 환경에서 위와 같은 사항들을 고려하여 적응화된 콘텐츠 서비스를 제공하기 위해 필요한 특징들을 다음과 같이 정의하였다.

- 명시성 : 프로파일을 구성하는 컴포넌트, 속성, 값 등은 명시적인 형태로 정의되어야 한다. 프로파일을 이용하는 시스템들에서 컴포넌트에 해당하는 컴포넌트, 속성, 값을 인식할 수 있어야 한다.
- 확장성 : 프로파일은 콘텐츠 적응화를 위한 새로운 컴포넌트, 속성 등 새로운 요소의 추가는 시스템 변경이 없이 이루어져야 한다.



(그림 2) 유비쿼터스 환경에서 콘텐츠 적응화를 위한 환경

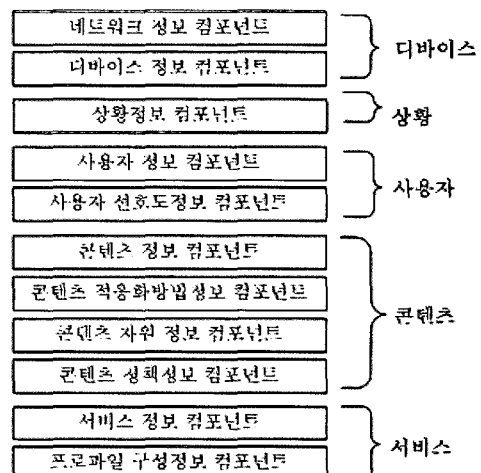
- 기밀성 : 프로파일은 사용자 정보, 사용자 선호 정보, 상황 정보 등 개인의 프라이버시에 관한 내용들이 다양한 네트워크를 통해 전송되기 때문에 전송되는 도중에 외부의 부정적인 접근으로부터 안전하게 보호되어야 한다.
- 상호 호환성 : 프로파일을 구성하는 컴포넌트의 전체 또는 일부는 다른 시스템에서 결합되어 사용될 수 있기 때문에 프로파일 구성 정보를 제공해야 하며, 컴포넌트, 속성의 이름은 표준 어휘를 준수해야 한다.
- 유연성 : 프로파일의 컴포넌트들은 필요에 따라 동적으로 구성될 수 있어야 한다. 필요한 컴포넌트에 따라 프로파일을 구성함으로써 프로파일 처리 및 네트워크 처리에 효율적이어야 한다.

4. 유비쿼터스 프로파일 설계

유비쿼터스 프로파일은 유비쿼터스 환경에서 사용자가 소유하고 있는 디바이스에 독립적으로 적응화된 콘텐츠를 제공하기 위해 본 논문에서 제안한 프로파일이다. 유비쿼터스 프로파일은 다음과 같이 정의될 수 있다.

“유비쿼터스 환경에서 사용자 디바이스에 독립적으로 다양한 형태의 콘텐츠를 사용자에게 제공하기 위한 사용자 관련 정보의 집합으로서 디바이스 정보, 상황 정보, 사용자 정보, 콘텐츠 정보, 서비스 정보 등으로 구성된다.”

유비쿼터스 환경에서 콘텐츠 적응화를 위한 프로파일은 사용자의 다양한 정보들이 유기적으로 융합되어 구성되어야 한다. 본 논문에서는 유비쿼터스 환경에서 콘텐츠 적응화에 대한 필요한 요소들을 3장의 콘텐츠 적응화를 위한 환경을 기반으로 디바이스, 상황, 사용자, 콘텐츠, 서비스 등으로 분류하고 네트워크, 디바이스, 상황 정보, 사용자 정보, 사용자 선호도, 콘텐츠 정보, 콘텐츠 적응화 방법, 콘텐츠 자원, 콘텐츠 정책, 서비스, 구성 정보 등 11개의 컴포넌트로 프로파일



(그림 3) 유비쿼터스 프로파일

일을 구성하였다. 상황정보 컴포넌트는 이미 구축되어 있는 상황정보를 이용할 수 있다고 가정하고, 상황 정보 중에서 콘텐츠 적용화에 필요한 요소만을 이용하여 구성하였다. (그림 3)은 유비쿼터스 프로파일을 나타내고 있다.

4.1 유비쿼터스 프로파일 설계

(1) 디바이스

네트워크 정보 컴포넌트는 사용자의 디바이스와 게이트웨이 간 사용하는 네트워크 정보를 표현한다. 네트워크 정보 컴포넌트에는 프로토콜, 전송 속도, QoS(Quality of Services)에 대한 사항 등이 포함된다. 디바이스 정보 컴포넌트는 사용자 디바이스 정보를 표현한다. 사용자가 사용하는 디바이스의 종류는 매우 다양하며, 향후에는 더욱 다양화 될 것이다. 따라서 디바이스들에 대한 각각의 특성에 따라 컴포넌트를 구성하는 방법은 부적합하다. 그렇기 때문에 본 논문에서는 콘텐츠 사용 시에 영향을 주는 디바이스 요소들을 추출하고, 이를 일반화하여 디바이스 컴포넌트를 구성하였다.

(2) 상황

상황정보 컴포넌트는 사용자가 존재하는 환경에 대한 정보를 표현한다. 상황정보 컴포넌트는 콘텐츠 적용화에 직접적으로 영향을 줄 수 있는 감성, 밝기, 소리, 이동성 등으로 구성된다.

(3) 사용자

사용자 정보 컴포넌트는 사용자를 식별할 수 있도록 하는 정보를 표현한다. 본 논문에서는 사용자를 식별할 수 있는 대표적인 방법인 아이디와 패스워드를 이용한다.

사용자 선호도정보 컴포넌트는 사용자가 이용하는 콘텐츠와 서비스들을 데이터베이스에 저장한 후 기본 서비스 요청 시 사용자에게 자동으로 맞춤화하여 서비스할 수 있도록 콘텐츠 종류, 콘텐츠 적용화 방법, 서비스 품질 등으로 구성된다.

(4) 콘텐츠

콘텐츠 정보 컴포넌트는 사용자가 콘텐츠를 요청 시 콘텐츠 서비스를 위해 필요한 정보를 표현한다. 콘텐츠 정보 컴포넌트에는 콘텐츠의 주소, 콘텐츠 제공 업체 정보 등으로 구성된다. 콘텐츠 적용화정보 컴포넌트는 콘텐츠 적용화를 위해 필요한 사항을 표현한다. 콘텐츠 정책정보 컴포넌트는 사용자의 콘텐츠에 대한 접근 권한, 접근 허용, 접근 금지 등 콘텐츠에 대한 사용권한을 표현한다. 콘텐츠 자원정보 컴포넌트는 콘텐츠를 구성하는 콘텐츠의 종류, 크기, 제작년도 등 자원에 대한 사항 등으로 구성된다.

(5) 서비스

서비스 정보 컴포넌트는 서비스되고 있는 정보들을 표현한다. 콘텐츠 이용 시간, 콘텐츠 이용 우선순위, 콘텐츠 품

질, 과금 정보 등으로 구성된다. 프로파일 구성정보 컴포넌트는 프로파일을 구성하는 컴포넌트들의 정보들을 이용하여 구성된다. 이 정보는 현재 구성된 컴포넌트들에 대한 정보를 알 수 있다.

4.2 유비쿼터스 프로파일 설계

본 논문에서는 프로파일 설계 시 3장에서 정의한 프로파일들이 가져야 할 특징들을 적용하기 위해 다음과 같은 기법들을 도입하였다.

- CC/PP Structure : 다른 시스템들과 상호 호환을 위하여 CC/PP Structure를 준수하였다.
- RDF : RDF(Resource Description Framework)는 XML 기반으로 메타데이터 기술을 위해 W3C에서 제안한 방법이다. 프로파일 설계 시 RDF를 사용함으로써 XML의 장점 등을 수용할 수 있다.
- RDF Schema: 컴포넌트 속성에 대한 제약사항을 기술하기 위해서 사용된다. 컴포넌트에 대한 속성 이름과 데이터타입을 지정한다.
- 표준 어휘 : 프로파일 설계 시 Component, Attribute들에 대한 이름들은[3, 8~11, 14, 15]에서 정의된 이름을 사용하였다.

본 논문에서는 4.1절의 프로파일 구성정보 컴포넌트, CC/PP Structure, RDF, RDF Schema, 표준어휘 등을 적용하여 유비쿼터스 프로파일이 가져야 할 특징들을 만족시켰다. <표 1>은 유비쿼터스 프로파일이 가져야 할 특징을 적용하기 위해 본 논문에서 적용한 기법들을 나타내고 있다.

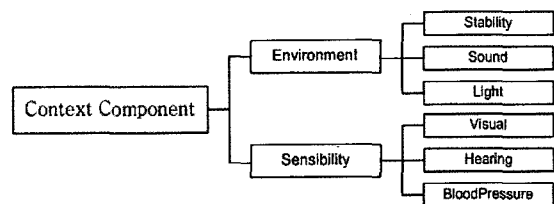
여기에서는 제안한 기법들을 유비쿼터스 프로파일 설계에 적용한다.

(그림 4)는 상황정보 컴포넌트의 구성을 나타내고 있다. 이 구성 정보를 기반으로 <표 1>의 기법들을 적용하여 <표 2>와 같은 유비쿼터스 프로파일을 구성하였다.

라인 3~11까지는 동적 구성 정보 컴포넌트로서 프로파일을 구성하고 있는 컴포넌트들에 대한 정보를 나타내고 있

<표 1> 특징과 제안한 기법

특징	제안한 기법
명시성	RDF Schema
확장성	프로파일 구성정보 컴포넌트, RDF
기밀성	RDF(XML Encryption)
상호 호환성	CC/PP Structure, 표준 어휘
유연성	프로파일 구성정보 컴포넌트



(그림 4) 상황정보 컴포넌트의 구성요소

〈표 2〉 상황정보를 포함한 유비쿼터스 프로파일의 일부분

```

1. <?xml version="1.0" encoding="UTF-8"?>
2. <rdf:rdff xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#" xmlns:ccpp="http://www.w3.org/2000/07/04-ccpp#" xmlns:upc="http://www.itpioneer.org/schema/ubiquitouschema/context#" xmlns:upd="http://www.itpioneer.org/schema/ubiquitouschema/context#" >
3. <rdf:description rdf:id="dynamiccomposite">
4. <ccpp:component>
5. <rdf:description rdf:id="elements">
6. <rdf:type rdf:resource="http://www.itpioneer.org/schema/ubiquitouschema/dynamiccomposite"/>
7. <rdf:bag>
8. <rdf:li>context</rdf:li>
9. </rdf:bag>
10. </ccpp:component>
11. </rdf:description>
12. <rdf:description rdf:id="Context">
13. <ccpp:component>
14. <rdf:description rdf:id="environment">
15. <rdf:type rdf:resource="http://www.itpioneer.org/schema/ubiquitouschema/context#environment"/>
16. <upc:stability>stable</upc:stability>
17. <upc:sound>silent</upc:sound>
18. <upc:light>norma</upc:light>
19. </rdf:description>
20. <rdf:description rdf:id="sensitivity">
21. <rdf:type rdf:resource="http://www.itpioneer.org/schema/ubiquitouschema/context#sensitivity"/>
22. <upc:visual>stable</upc:visual>
23. <upc:hearing>silent</upc:hearing>
24. <upc:bloodpress>norma</upc:bloodpress>
25. </rdf:description>
26. </ccpp:component>
27. </rdf:description>
28. </rdf:rdff>
    
```

다. 라인 12~28까지는 (그림 4)에 해당하는 상황 정보를 나타내고 있다.

5. 효율적인 운영을 위한 아키텍처 설계

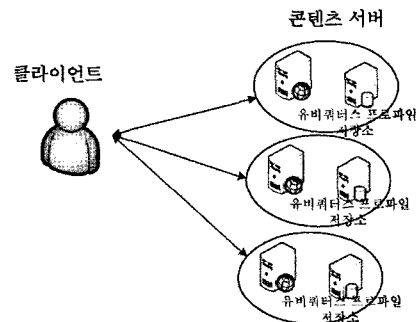
본 장에서는 유비쿼터스 프로파일의 효율적인 운영을 위한 아이디어를 제시한 후 이를 기반으로 하여 프로파일 운영 아키텍처를 설계한다.

5.1 효율적인 운영을 위한 아이디어 제시

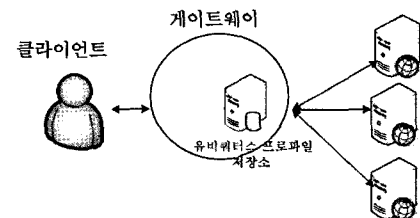
유비쿼터스 프로파일의 효율적인 운용을 위하여본 논문에서는 세 가지 아이디어를 제시한다.

첫째, 게이트웨이 중심의 프로파일 운영 방법이다. 사용자에게 대한 프로파일들을 게이트웨이 안의 유비쿼터스 프로파일 저장소에서 관리하며, 프로파일을 요청 시 프로파일을 제공하는 방법이다. 기존에 콘텐츠 서버에서 별도로 유지되던 프로파일들을 게이트웨이에서 통합 운용함으로써 운영의 효율성 및 처리의 효율성을 가져올 수 있다. 또한 프로파일을 관리하는 게이트웨이의 저장소는 콘텐츠 서버에 대해 독립적인 운영이 가능하다. (그림 5)는 기존 프로파일 연구들에서 사용하는 방법으로 콘텐츠 서버에서 프로파일을 운영하는 방법을 나타내고 있으며, (그림 6)은 본 논문에서 제안하는 게이트웨이 중심의 프로파일 운영 방법을 나타내고 있다.

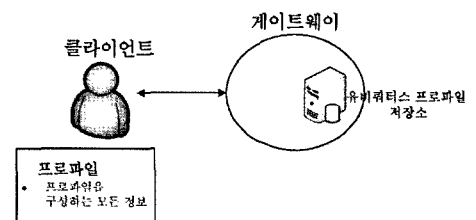
둘째, 저장소들을 이용한 메타데이터 구성 방법이다. 프로파일 구성 시 정적인 정보들은 저장소에서 유지 관리하며, 해당 정보의 메타데이터를 이용하여 프로파일을 구성하는 방법이다. 따라서 사용자는 프로파일 요청 시 동적인 정보와 정적 정보의 메타데이터를 이용하여 프로파일을 구성한다. 이 방법을 사용하면 프로파일의 크기가 작아지므로 네트워크 전송 및 처리의 효율성을 가져올 수 있다. 또한 프로파일 교환 시 사용자는 정적 정보에 대한 메타데이터를 이용하기 때문에 정적 정보의 변경에 독립적일 수 있다. (그림 7)은 게이트웨이에서 프로파일 저장소 없이 통합 운영하는 방법을 나타내고 있으며, (그림 8)은 정적 정보들을 프로파일 저장소에 운영하며 프로파일 교환은 정적 정보에 해당하는 메타데이터와 동적 정보를 이용하여 프로파일을 구성한 후 교환하는 방법을 나타내고 있다.



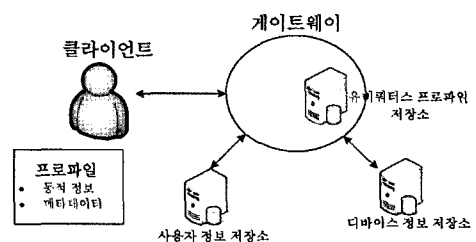
(그림 5) 콘텐츠 서버 중심의 프로파일 운영방법



(그림 6) 게이트웨이 중심의 프로파일 운영 방법



(그림 7) 전체 프로파일 구성 방법

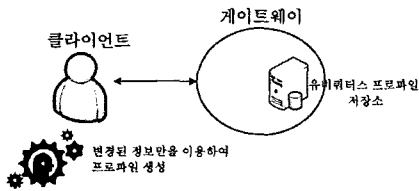


(그림 8) 저장소를 이용한 메타데이터 구성 방법

<표 3> 제안한 운영 아키텍처에서의 프로파일 운영 정보

O: 지원 X: 지원하지 않음

구분	컴포넌트 명	컴포넌트 생성 장소	데이터 구성 형태	주기적/비주기적 교환 형태	메타데이터를 이용한 컴포넌트 구성
디바이스	디바이스 정보 컴포넌트	클라이언트	정적	비주기적	O
		디바이스 정보 저장소	정적	비주기적	X
상황	상황 정보 컴포넌트	클라이언트	정적	비주기적	X
		클라이언트	동적	주기적	O
사용자	사용자 정보 컴포넌트	클라이언트	동적	비주기적	O
		사용자 정보 저장소	동적	비주기적	X
	사용자 선호도정보 컴포넌트	클라이언트	동적	주기적	X
콘텐츠	콘텐츠 정보 컴포넌트	클라이언트	동적	비주기적	O
	콘텐츠 적응화방법정보컴포넌트	콘텐츠 서버	정적	비주기적	X
	콘텐츠 자원 컴포넌트	콘텐츠 서버	정적	비주기적	X
	콘텐츠 정책컴포넌트	콘텐츠 서버	정적	비주기적	X
서비스	서비스 정보 컴포넌트	클라이언트	동적	주기적	X
	프로파일 구성정보 컴포넌트	클라이언트	정적	주기적·비주기적	X
		콘텐츠 서버	정적	주기적·비주기적	X
		사용자 정보 저장소	정적	주기적·비주기적	X
		상황정보 저장소	정적	주기적/비주기적	X
	디바이스 정보 저장소	정적	주기적/비주기적	X	



(그림 9) 프로파일의 동적 구성 방법

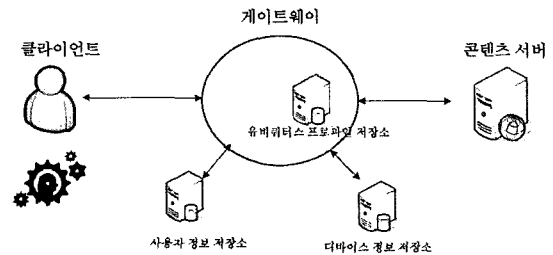
셋째, 프로파일의 동적 구성 방법이다. 클라이언트와 프로파일 저장소 간의 프로파일 교환은 변경된 정보만을 이용하여 교환하는 방법이다. 프로파일을 구성하는 컴포넌트 중에는 주기적·비주기적으로 정보 변경이 일어나는 컴포넌트와 한번 생성된 후 변경이 일어나지 않는 컴포넌트로 나뉜다. 따라서 전체 프로파일을 전송하는 것보다 변경이 이루어진 컴포넌트만을 전송하여 프로파일을 구성하면, 프로파일의 크기를 줄일 수 있어 네트워크 전송에 있어서 효율적이고, 적은 컴포넌트 요소들만을 포함하고 있기 효율적으로 처리할 수 있다. (그림 9)는 프로파일 전송 시 변경된 정보만을 이용하여 프로파일을 구성하고 교환하는 방법을 나타내고 있다.

5.2 프로파일 운영을 위한 아키텍처 설계

본 절에서는 5.1절의 세 가지 아이디어들을 기반으로 프로파일 운영을 위한 (그림 10)과 같은 유비쿼터스 프로파일 운영 아키텍처를 설계한다.

유비쿼터스 프로파일 운영을 위한 아키텍처는 클라이언트, 게이트웨이, 콘텐츠 서버로 구성하였다. 게이트웨이는 클라이언트와 콘텐츠 서버 사이의 중간 매개체 역할을 담당하면서 다음과 같은 기능을 가진다.

첫째, 유비쿼터스 프로파일을 관리한다. 프로파일들은 게이트웨이 안에 있는 유비쿼터스 프로파일 저장소에서 생성, 삭제, 추가, 업데이트 기능을 수행한다. 둘째, 프로파일 구성



(그림 10) 유비쿼터스 프로파일 운영을 위한 아키텍처

시 필요한 정보를 가지고 있는 서버들과 교환을 통해 필요한 정보를 수집한다. 운영에 필요한 서버들은 사용자에게 대한 정보, 선호도 정보를 가지고 있는 사용자 정보 저장소, 디바이스 명세에 대한 정보를 가지고 있는 디바이스 정보 저장소로 구성된다. 콘텐츠 서버는 사용자에게 제공되는 콘텐츠와 콘텐츠에 대한 메타데이터를 가지고 있다. 메타데이터에는 콘텐츠 적응화 방법, 자원 정보, 콘텐츠 정책 정보 등을 가지고 있다.

<표 3>은 제안한 운영 아키텍처에서 프로파일 운영 정보를 나타내고 있다. 컴포넌트 생성 장소는 유비쿼터스 프로파일을 구성하는 컴포넌트들이 생성되는 장소로서 컴포넌트들은 클라이언트, 사용자 정보 저장소, 디바이스 정보 저장소 콘텐츠 서버에서 생성되며, 생성된 컴포넌트들은 게이트웨이의 유비쿼터스 프로파일 저장소에서 융합되어 유비쿼터스 프로파일을 생성하게 된다. 데이터 구성 형태는 컴포넌트를 구성하는 정보의 형태로서 사용자에게 따라 각각 다른 동적 데이터와 사용자에게 상관없이 동일한 정보를 제공하는 정적 데이터로 나뉜다. 주기적·비주기적 교환형태는 컴포넌트들을 이용하여 유비쿼터스 프로파일을 생성한 후 지속적인 콘텐츠 서비스를 제공하기 위하여 유비쿼터스 프로파일을 지속적으로 업데이트하기 위해 클라이언트, 유비쿼터스 프로파일 저장소, 콘텐츠 서버 등과 프로파일을 교환하는

<표 4> 유비쿼터스 프로파일의 기능적 특성 비교

O: 지원 X: 지원하지 않음

구분	UAProf[15]	UPS[8]	유비쿼터스 프로파일
프로파일 구성	네트워크 정보	O	O
	디바이스 정보	O	O
	상황 정보	X	X
	사용자 정보	X	X
	사용자 선호정보	X	O
	콘텐츠 정보	X	O
	콘텐츠 적응화 정보	X	O
	콘텐츠 자원 정보	X	O
	콘텐츠 정책 정보	X	X
	서비스 정보	X	X
구성 정보	X	X	
CC/PP 준수	O	O	O
프로파일 표현 언어	RDF	RDF	RDF
계약 사항 기술	RDF Schema	RDF Schema	RDF Schema
표준 어휘 사용	[14] 참고하였으며, 필요한 어휘는 새롭게 정의하여 사용.	[14][15] 참고하였으며, 필요한 어휘는 새롭게 정의하여 사용.	[3][8][9][10][11][14][15]참고하였으며, 필요한 어휘는 새롭게 정의하여 사용.
프로파일 운영 아키텍처	프로파일 운영에 필요한 아키텍처를 제한하여 사용함.	WAP를 사용하는 통신업체에서 운영할 수 있도록 함.	게이트웨이 중심의 프로파일 운영 방식, 저장소를 통한 경량 프로파일 구성 방식, 프로파일의 동적 구성 등을 적용함으로써 다양한 환경에서 적용 가능함.

주기를 나타낸다. 메타데이터를 이용한 컴포넌트 구성은 컴포넌트 생성 시 정적 데이터의 키 값을 이용하여 컴포넌트를 구성하는 방법이다. 유비쿼터스 프로파일 저장소에서는 컴포넌트에서 메타데이터의 값을 분리한 후 이것을 이용하여 컴포넌트 구성에 필요한 정보를 저장소들에서 수집하게 된다.

6. 제안한 유비쿼터스 프로파일 분석 및 운영 아키텍처 성능 분석

본 장에서는 <표 4>의 항목들을 기준으로 기존의 프로파일 연구들과 본 논문에서 제안한 유비쿼터스 프로파일의 기능적 특성을 비교 분석하고, 제안한 운영 아키텍처에 대한 성능을 평가한다.

6.1 프로파일 분석

(1) 프로파일 구성

제안한 유비쿼터스 프로파일은 <표 4>와 같이 기존 프로파일 연구에서 반영하지 않고 있는 사용자 정보, 상황 정보, 서비스 정보, 구성 정보 등 다양한 요소들을 반영하고 있기 때문에 콘텐츠 적응화에 적합하다. 상황정보를 통해 개인의 감성까지 반영된 서비스가 가능하므로 개인 맞춤형 서비스가 가능하다. 또한 프로파일의 구성에 대한 정보를 가지고 있어 경량의 프로파일 구성이 가능하므로 프로파일 처리 및 네트워크 전송에 있어서 효율적이다.

(2) 표준 어휘

제안된 유비쿼터스 프로파일은 CC/PP를 기반으로 [3, 8~11, 14, 15]의 어휘들을 사용하여 작성하였기 때문에 기존 연구들보다 프로파일 교환을 위한 상호 호환성면에서 우수

하다.

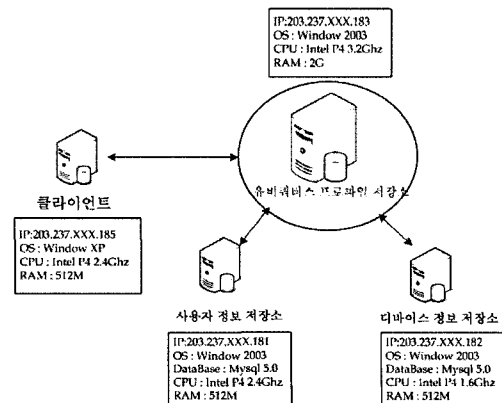
6.2 운영 아키텍처에 대한 성능 분석

성능 분석은 기존 프로파일 연구의 정적 구성 방법과 본 논문에서 제안한 운영 아키텍처의 동적 구성 방법, 프로파일 저장소 및 동적 구성 방법을 결합한 프로파일 저장소를 이용한 동적 구성 방법에 대하여 성능을 평가한다.

(1) 실험 환경 구축 및 실험 패키지 구현

운영 아키텍처에 대한 성능평가를 위해 (그림 11)과 같이 클라이언트, 유비쿼터스 프로파일 저장소, 사용자 정보 저장소, 디바이스 정보 저장소를 이용하여 실험 환경을 구축하였다.

실험 환경 구축 시 사용자 정보 저장소, 디바이스 정보 저장소의 데이터베이스는 Mysql[17]를 사용하였으며, 클라이언트와 유비쿼터스 프로파일 저장소에서는 프로파일 생성 및 처리를 위해 RDF 개발 툴킷인 JENA[18]를 사용하였다.



(그림 11) 운영 아키텍처에 대한 성능 평가를 위한 실험 환경

<표 5> 실험을 위해 구현한 패키지 목록

패키지 명	처리 기능
clientprofile	프로파일을 구성하는 컴포넌트의 정보를 관리하는 패키지
vocabulary	프로파일을 구성하는 컴포넌트들의 속성에 대한 Vocabulary 정보를 가지고 있는 패키지
profileelement	컴포넌트 속성 정보들을 관리하기 위한 패키지
profiledata	사용자의 프로파일 정보를 소유하고 있는 패키지
clientprofilecreator	프로파일을 RDF 형태로 생성, 변경, 삭제 등의 기능을 가지고 있는 패키지
rdfprofile	RDF 형태의 프로파일을 객체형태로 변환하는 패키지
sessionidgenerator	생성된 프로파일을 식별하기 위해 고유한 키를 생성하는 패키지
client	유비쿼터스 프로파일 저장소와 프로파일 교환을 위한 패키지 (클라이언트)
server	클라이언트와 프로파일 교환을 위한 패키지(서버)
resourcemanager	프로파일 저장소와 사용자 프로파일 저장소, 디바이스 저장소 간의 프로파일 교환을 위한 패키지

클라이언트에서 필요한 프로파일 생성 패키지들과 프로파일 저장소에서 필요한 프로파일 처리 및 관리 패키지들은 Java를 이용하여 구현하였다. <표 5>는 실험을 위해 구현한 패키지들의 목록을 나타내고 있다. 실험에서 클라이언트와 프로파일 저장소간의 프로파일 교환은 클라이언트/서버 구조로 소켓을 이용하여 구현하였으며, 유비쿼터스 프로파일 저장소와 사용자 정보 저장소, 디바이스 정보 저장소간의 연결은 JDBC를 이용하여 구현하였다.

(2) 실험 항목 및 평가 방법

실험 항목은 <표 3>의 운영 아키텍처의 프로파일 운영 정보를 기반으로 콘텐츠 적응화를 위해 필요한 프로파일로 설정하였다. <표 6>은 실험에 사용한 프로파일의 특성을 나타내고 있다.

성능 측정은 기존 연구의 프로파일 정적 구성 방식, 프로파일 동적 구성 방식, 프로파일을 이용한 동적 구성 방법에 따라 유비쿼터스 프로파일 저장소에 프로파일 처리가 완료된 후 결과를 클라이언트가 전송 받는 시점까지를 기준으로 하여 프로파일 생성 시간, 프로파일 전송 시간, 프로파일 처리시간의 총합을 프로파일의 속성 수와 클라이언트 연결 수를 증가시키면서 성능을 평가하였다. 컴포넌트의 속성 수는 동적 정보 컴포넌트를 제외한 각 컴포넌트들의 속성을 2개씩 증가하면서 실험을 수행하였다. 클라이언트 연결은 스프레드를 이용하였으며, 연결 수를 5개씩 증가하면서 실험을 수행하였다.

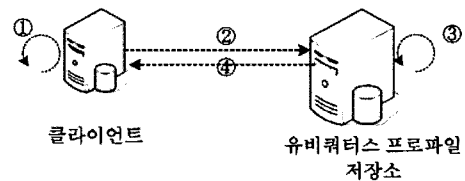
3) 실험 시나리오

실험은 사용자가 콘텐츠를 이용하는 동안 프로파일변경이 상황 정보 변경, 콘텐츠 변경, 사용자 선호도 변경에 따라 4번 발생했다고 가정하였다. 프로파일이 처음 생성된 후 프

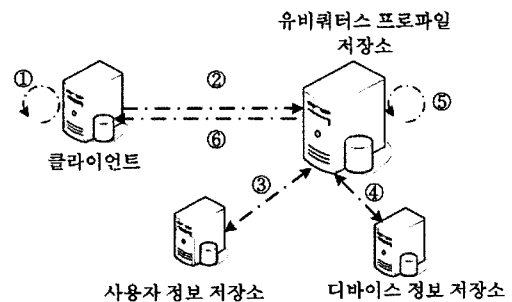
<표 6> 실험에서 사용한 프로파일의 특성

O : 지원 X: 지원하지 않음

컴포넌트 이름	컴포넌트의 속성 개수	동적 구성	메타데이터 구성
네트워크 정보 컴포넌트	5	X	X
디바이스 정보 컴포넌트	35	X	O
사용자 정보 컴포넌트	12	X	O
사용자 선호 정보 컴포넌트	6	O	O
상황 정보 컴포넌트	8	O	X
콘텐츠 정보 컴포넌트	5	O	X
서비스 정보 컴포넌트	8	O	X
프로파일 구성 정보 컴포넌트	7	X	X



(그림 12) 정적 구성 방식과 동적 구성 방식의 실험 단계



(그림 13) 저장소를 이용한 동적 구성 방식의 실험 단계

로파일 변경 순서는 상황 정보 변경→콘텐츠 정보 변경→상황 정보 변경→사용도 선호도 변경 순으로 작성하였다.

(그림 12)는 정적 구성 방법과 동적 구성 방법의 시간 측정을 위한 실험 단계를 나타내고 있다.

- ① 클라이언트에서 사용자의 상황에 따라 프로파일을 생성한다.
- ② 클라이언트는 생성된 프로파일을 유비쿼터스 프로파일 저장소로 전송한다.
- ③ 유비쿼터스 프로파일 저장소에서는 전송받은 프로파일을 처리한 후 프로파일을 저장소에 저장한다.
- ④ 유비쿼터스 프로파일 저장소는 처리 결과를 클라이언트로 전송한다.

(그림 13)은 저장소를 이용한 동적 구성 방법에 대한 시간 측정을 위한 실험 단계를 나타내고 있다.

- ① 클라이언트에서 사용자의 상황에 따라 메타데이터를 이용하여 프로파일을 생성한다.
- ② 클라이언트는 생성된 프로파일을 유비쿼터스 프로파일 저장소로 전송한다.
- ③ 유비쿼터스 프로파일 저장소에서는 프로파일을 처리하

여 프로파일을 구성하는 메타 데이터 정보를 추출한다. 추출한 정보를 따라 ④,⑤ 단계를 수행한다.

- ④ 추출한 정보를 이용하여 사용자 정보 저장소에 사용자 정보 또는 사용자 선호 정보를 요청한다.
- ⑤ 추출한 정보를 이용하여 디바이스 정보 저장소에 디바이스 정보를 요청한다.
- ⑥ 클라이언트에서 전송된 프로파일과 ④,⑤에서 가져온 컴포넌트 정보를 통합한 후 프로파일을 저장한다.
- ⑦ 유비쿼터스 프로파일 저장소는 처리 결과를 클라이언트로 전송한다.

(4) 실험 결과 및 고찰

(그림 14)는 <표 6>의 프로파일을 기반으로 클라이언트 연결 개수를 5로 설정한 후 프로파일 속성 수를 14개씩 증가하면서 실험한 결과이다. 프로파일을 구성하는 컴포넌트 속성의 수가 증가할수록 프로파일 저장소를 이용한 동적 구성 방법이 가장 좋은 성능을 나타냈다. 프로파일 저장소를 이용한 동적 구성 방법은 동적 구성 방법과 비교하여 평균 6% 성능 향상을 나타냈으며, 정적 구성 방법과 비교하여 26%의 성능 향상을 나타냈다.

(그림 15)는 <표 6>의 프로파일을 기반으로 클라이언트 연결 수를 5개씩 증가시키면서 실험한 결과를 나타내고 있다. 클라이언트 연결 수가 증가 할수록 프로파일 저장소를

이용한 동적 구성 방법이 좋은 성능을 나타냈다. 프로파일 저장소를 이용한 동적 구성 방법은 동적 구성 방법과 비교하여 평균 5% 성능 향상을 나타냈으며, 정적 구성 방법과 비교하여 39%의 성능 향상을 나타냈다.

유비쿼터스 시대에서는 사용자에게 최적화된 콘텐츠를 제공하기 위해 사용자의 많은 정보들을 이용하여 프로파일을 구성하게 될 것이다. 또한 프로파일을 이용하는 서비스가 증가하고 이에 따라 사용자 수도 증가할 것이다. 그러므로 본 논문에서 제안한 프로파일 운영 아키텍처는 프로파일의 속성 수 증가 및 사용자 증가에 따른 프로파일 운영 문제의 해결책이 될 수 있을 것이다.

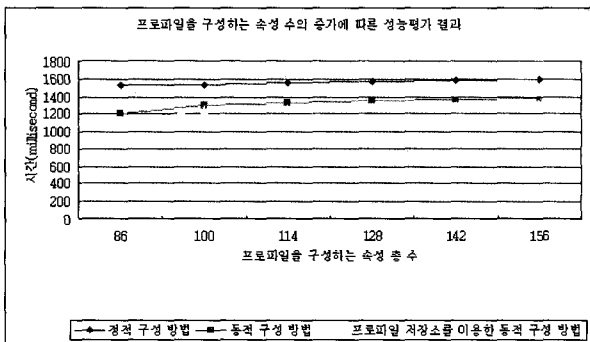
7. 결론 및 향후 연구

본 논문에서는 유비쿼터스 환경에서 콘텐츠 적응화를 위한 CC/PP기반의 유비쿼터스 프로파일 및 운영 아키텍처를 설계하였다. 유비쿼터스 프로파일은 유비쿼터스 환경에서 콘텐츠 서비스 시나리오를 기반으로 디바이스, 상황, 사용자, 콘텐츠, 서비스로 분류하고 11개의 컴포넌트로 구성하였다. 유비쿼터스 프로파일은 기존 프로파일에 비해 다양한 정보들로 구성되어, 사용자에게 보다 최적화된 콘텐츠를 제공할 수 있다. 또한, 유비쿼터스 프로파일 설계 시 CC/PP의 Structure, 표준 어휘, RDF, RDF Schema을 적용함으로써 프로파일이 가져야할 특징인 명시성, 확장성, 기밀성, 상호 호환성, 유연성 등을 만족시켰다. 프로파일의 효율적인 운영을 위해 게이트웨이 중심의 운영 방식, 저장소를 이용한 메타데이터 구성 방식, 동적 구성방식에 대한 아이디어들을 제시하였고, 이를 기반으로 운영 아키텍처를 설계하고 성능 평가를 수행하였다. 제안한 아키텍처에서 게이트웨이 중심 운영 방식은 기존 콘텐츠 서버마다 별도로 운영하던 방식에 비해 통합 운영함으로써 프로파일 관리에 효율적이었으며, 프로파일 제공 기능을 통하여 콘텐츠 서버뿐만 아니라 다양한 서비스 분야에 프로파일을 사용할 수 있다. 또한 프로파일 운영에 대한 실험 결과 프로파일을 이용한 동적 구성 방법은 프로파일 속성의 수 증가할수록 동적 방법에 비교하여 6%, 정적 방법 비교하여 26%의 성능 향상을 나타냈으며, 클라이언트 수가 증가할수록 동적 방법에 비교하여 5%, 정적 방법 비교하여 39%의 성능 향상을 나타냈다.

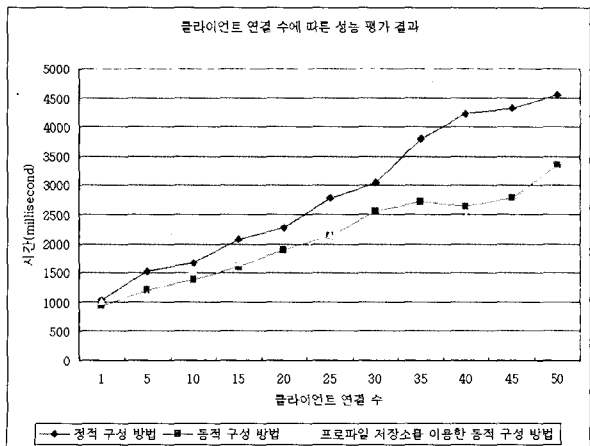
향후에는 제안한 유비쿼터스 프로파일을 다양한 서비스 분야에서 사용할 수 있도록 프로파일의 어휘 및 기능을 확장한 프로파일 설계한다. 또한 본 논문에서 제안한 운영 아키텍처를 기반으로 프로파일을 서비스 할 수 있도록 하는 시스템을 설계 및 구현한다.

참고 문헌

[1] A. Coles, E. Deliot, T. Melamed, K. Lansard, "A framework for coordinated multi-modal browsing with multiple clients," International World Wide Web Conference, pp. 718-726, May



(그림 14) 컴포넌트의 속성 수 증가에 따른 성능 평가 결과



(그림 15) 클라이언트 연결 수의 증가에 따른 성능 평가결과

2003

[2] M. Weiser, "Some computer science issues in ubiquitous computing," *ACM SIGMOBILE Mobile Computing and Communications Review*, vol 36. issue 7, pp.75-84, July, 1993.

[3] P. Korpipaa, J. Mantyjarvi, J. Kela, H. Keranen, J. E. Malm, "Managing context information in mobile devices," *Pervasive Computing. IEEE*, Vol.2, issue 3, pp.42-51, July-Sept., 2003.

[4] S. buchholz, T. Hanmann, Gerald Hubsch, "Comprehensive Structured Context Profile(CSCP) : Design and Experience," *Pervasive Computing and Communications Workshops, 2004. Proceedings of the Second IEEE Annual Conference on*, pp.43-47, Mar., 2004.

[5] T. Lemlouma, N. Layaïda, "Encoding Multimedia Presentation for User Preferences and Limited Environments," *IEEE International Conference on Multimedia & Expo(ICME)*, pp.165-168, July, 2003.

[6] T.Lemlouma, N. Layaïda, "Content Adaptation and Generation Principles for Heterogeneous Clients," *W3C Workshop on Device Independent Authoring Techniques*, Sept., 2002.

[7] T. emlouma, N. Layaïda, "The Negotiation of Multimedia Content Services in Heterogeneous Environments," In the *MMM 2001: the 8th International Conference on Multimedia Modeling*, pp.187-206, Nov., 2001.

[8] T. emlouma, N. Layaïda, "Universal Profiling Schema for Content Negotiation," *INRIA* Jan., 2002.

[9] G. Klyne, "RFC 2533: A Syntax for Describing Media Feature Sets," *IETF Request for Comments: ftp://ftp.isi.edu/in-notes/rfc2533.txt*, Mar., 1999.

[10] K. Holtman, A. Mutz, T. Hardie, "RFC 2506: Media Feature Tag Registration Procedure," *IETF Request for Comments: ftp://ftp.isi.edu/in-notes/rfc2506.txt*, Mar., 1999.

[11] L. Masinter, D. Wing, A. Mutz, K. Holtman, "RFC 2534: Media Features for Display, Print, and Fax," *IETF Request for Comments: ftp://ftp.isi.edu/in-notes/rfc2534.txt*, Mar., 1999.

[12] B. Bridge, Hewlett-Packard Laboratories, "RDF Primer," *W3C*, Feb., 2004.

[13] B. Bridge, "RDF Vocabulary Description Language 1.0 :RDF Schema," Feb., 2004.

[14] G. klyne, et al, "Composite Capability/Preference Profiles

(CC/PP) : Structure and Vocabularies 1.0," *W3C*, Jan., 2004.

[15] Open Mobile Alliance, "UAProf(User Agent Profile)," *Open Mobile Alliance*, May, 2003.

[16] Jena Sematic Web Framework, <http://jena.sourceforge.net/>

[17] Mysql, <http://www.mysql.com/>

[18] The World Wide Web Consortium(W3C), "<http://www.w3c.org>"



김 경 식

e-mail : letsdoit@dku.edu

2002년 단국대학교 전자계산학과(학사)

2004년 단국대학교 컴퓨터학과(석사)

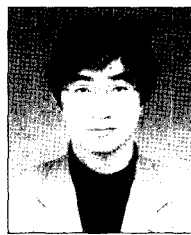
2006년 단국대학교 컴퓨터학과(수료)

2006년3월~현재 용인송담대학교 겸임교수

2006년4월~현재 단국대학교 문화콘텐츠

기술 연구소 연구원

관심분야: Ubiquitous Computing, Web Services, Contents Adaptation, Profile



이 재 동

e-mail : letsdoit@dku.edu

1985년 인하대학교 전자계산학 학사

1991년 Cleveland State University 석사

1996년 Kent State University 박사

1997년3월~현재 단국대학교 자연과학대학

정보컴퓨터학부 컴퓨터과학전공

부교수

2006년 4월~현재 단국대학교 문화콘텐츠 기술 연구소 소장

2004년 7월~2006년6월 단국대학교 정보통신원 원장(C.I.O)

2002년11월~현재 농협중앙회 전산고문

2005년 1월~현재 전국대학정보화 협의회 이사

2005년 8월~현재 문화관광부 KOCCA CT포럼/전략기획 운영
위원/분과위원장

2004년 1월~현재 (사)이러닝 산업협회 이사

2005년 1월~현재 교통안전공단 자문위원

관심분야: Ubiquitous Computing, Contents Technologies(Mobile),
Internet Technologies/Applications, GIS Technologies
and Applications, Many aspects of parallel/
distributed processing