

유비쿼터스 환경에서 정확한 상황인지를 지원하는 컨텍스트 지향 미들웨어

이권진, 송성근, 윤희용
성균관대학교 정보통신공학부
kwonjin@ece.skku.ac.kr, kskk103@hotmail.com, youn@ece.skku.ac.kr

A New Context-Oriented Middleware for supporting Exact Context-Awareness in Ubiquitous Environment

Kwon-jin Lee, Sung Keun Song, Hee Young Youn
School of Information and Communications Engineering, SungKyunKwan University

요 약

오늘날 유비쿼터스 환경에서 이종 객체간의 상호운용성 지원 및 애플리케이션 개발을 지원하는 미들웨어의 중요성 부각되고 있다. 현재 유비쿼터스 환경에 적합한 미들웨어가 연구 개발되고 있으며 그 중 메시지 기반의 미들웨어가 널리 사용 되고 있다. 그러나 기존 미들웨어 및 연구 개발되고 있는 유비쿼터스 미들웨어는 네트워크 상황을 고려하지 않은 채 단순히 컨텍스트 전송을 지원한다. 이로 인해 시스템들은 잘못된 상황 판단할 수 있다. 본 논문에서는 이러한 문제점을 해결하는 컨텍스트 지향 미들웨어를 제안한다. 본 논문에서 제안하는 미들웨어는 컨텍스트의 효율적이고 신뢰성 있는 전송을 지원하며, 이로 인해 시스템 또는 에이전트가 신속하고 정확한 상황인지 및 판단을 할 수 있도록 지원한다.

1. 서론

최근 몇 년 동안 이슈가 되어온 유비쿼터스 컴퓨팅 기술의 연구 동향 및 발전 추세는 다양한 에이전트들이 상황 인지 및 추론, 협업을 통해 사용자들에게 상황에 맞는 최적의 서비스를 심리스(Seamless)하게 제공하는 방향으로 나아가고 있다. 이러한 추세 속에 유비쿼터스 환경에서 이종 객체간의 상호운용성 지원 및 애플리케이션 개발을 지원하는 미들웨어의 중요성 부각되었다.

유비쿼터스 환경에서 발생하는 수많은 컨텍스트(Context) 및 데이터를 보다 효율적으로 전송하고 처리하기 위해 많은 미들웨어가 연구 개발되고 있으며 그 중 메시지 기반의 미들웨어(MOM: Message-oriented Middleware)[1]는 대용량 처리와 비동기통신, 고 가용성으로 인하여 널리 사용 되고 있다. 이러한 메시지 기반 미들웨어는 대표적으로 JMS[2], CORBA Event Service[3], IBM의 WebSphere[4] 그리고 Microsoft의 MSMQ[5] 등이 있으며, Hermes[6] 등 메시지 기반의 유비쿼터스 미들웨어[7-10]가 연구 개발되고 있다.

유비쿼터스 환경에서 사용자에게 상황에 맞는 최적인 서비스를 심리스하게 제공하기 위해서는 미들웨어가 시스템이 추론을 통해 사용자의 상황을 신속하고 정확하게 판단할 수 있도록 각 센서로부터 오는 컨텍스트들을 발생 순서에 따라 신속하게 전송해야 하고 이를 통해 상황에 맞는 서비스를 제공할 수 있도록 이종 에이전트간의 협업을 지원해야 한다. 그러나 기존 미들웨어 및 연구 개발되고 있는 유비쿼터스 미들웨어는 이종 에이전트간의 협업은 지

원하지만 컨텍스트의 발생 순서를 고려하지 않은 채 단순한 전송만을 지원한다. 즉, 센서들의 성능 및 네트워크 상황에 따라 컨텍스트의 발생 순서와 전송 순서가 달라질 수 있다는 경우를 고려하지 않은 채 단순히 컨텍스트 전송을 지원한다. 이로 인해 시스템들은 잘못된 상황판단으로 잘못된 서비스를 사용자에게 제공할 수 있다.

본 논문에서는 이러한 문제점을 해결하는 컨텍스트 지향 미들웨어(Context-oriented Middleware)를 제안한다. 본 논문에서 제안하는 미들웨어는 CORBA Event Service 기반으로서 센서들을 그룹별로 CORBA 기반의 Naming Service[11]로 관리하고 그룹내의 센서들은 컨텍스트의 발생 순서와 전송순서가 일치하도록 이벤트 서버에서 2 계층 버퍼링(2-tier buffering)을 활용하여 어떠한 네트워크 상황에서도 순서를 맞추어 주어 이 문제를 해결한다. 컨텍스트 기반 미들웨어는 컨텍스트의 효율적이고 신뢰성 있는 전송을 지원하며, 이로 인해 시스템 또는 에이전트가 신속하고 정확한 상황인지 및 판단을 할 수 있도록 지원한다.

본 논문의 구조는 다음과 같다. 서론에 이어 2장에서는 대표적인 MOM 중 하나이며, 우리 미들웨어의 기반인 CORBA Event Service에 대해서 살펴본다. 3장에서는 우리가 제안하는 컨텍스트 지향 미들웨어를 소개 한다. 마지막 4장에서는 본 논문의 결론과 향후 연구 과제에 대해 논하도록 한다.

2. CORBA Event Service

제공자들이 만든 이벤트 메시지는 이벤트 채널을 통해 이벤트 서버로 보내지게 되고, 소비자는 제공자와 같은 이벤트 채널을 통해 이벤트 메시지를 받는다. 그리고 이러한 제공자와 소비자는 하나 이상의 이벤트 채널을 통해 서로 통신을 하게 된다. 이벤트 채널은 서비스 제공자로부터 발생하는 이벤트를 소비자에게 전송하게 되고, 소비자는 제공자에 대한 정보를 가지고 있지 않아도 이벤트 메시지를 전송 받아 사용할 수 있다. Event Service에서 이벤트 채널은 이벤트 서버에서 제공자와 소비자간의 통신을 가능하게 하는 중재자 역할을 수행하게 된다. 중재자 역할을 하는 이벤트 채널은 제공자와 소비자의 등록, 제거, timely, 그리고 제공자에서 발생하는 이벤트를 소비자들에게 이벤트 손실 없이 전송될 수 있도록 신뢰성을 보장하며, 그리고 여러 제어에 대한 기능을 제공한다. 이러한 서비스를 제공하기 위해 이벤트 서버는 푸시(Push)와 풀(Pull)모형을 제공한다. 푸시(Push)모형은 제공자가 이벤트를 프락시소비자(Proxy_Consumer)이벤트 채널에 보내게 되고, 그리고 이벤트 채널의 ProxyConsumer는 제공자로부터 들어온 이벤트를 소비자에게 다시 전송한다. 그림 1(a)은 푸시(Push)모형의 이벤트 전달을 보여준다.

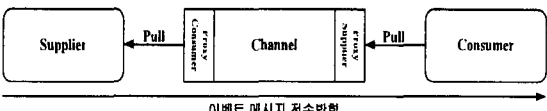
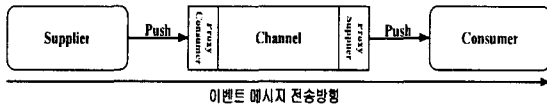


그림 1 Push/Pull Model

풀 모델은 푸시(Push) 모델과는 반대로 소비자가 이벤트 채널의 ProxyConsumer를 통해 이벤트 메시지를 Pull하게 되고, ProxyConsumer는 제공자에서 이벤트 메시지를 pull하게 된다. 그림 1(b)는 풀 (Pull) 모델의 이벤트 전달을 보여준다.

또한, 이벤트 채널은 다수의 제공자와 소비자를 하나로 묶여질 수 있는 방안을 제공해 준다. 그리고, push와 pull 모델의 조합도 가능하다. 이벤트 채널 서버는 push/push, push/pull, pull/push, 그리고 pull/pull과 같은 4개의 다른 Push모델과 Pull 모델의 조합을 제공한다.

3. 컨텍스트 지향 미들웨어

본 논문에서 제안하고자 하는 미들웨어는 기존 CALM(Component-based Autonomic Layered Middleware)[12] 연구 프로젝트내에서 진행되고 있는 컨텍스트 지향 미들웨어이다. CALM의 구조는 그림 2와 같으며 컨텍스트 지향 미들웨어는 그림 2에서 커뮤니케이션 플랫폼(Communication platform)에 해당한다.

3.1 컨텍스트 지향 미들웨어 서비스의 개념

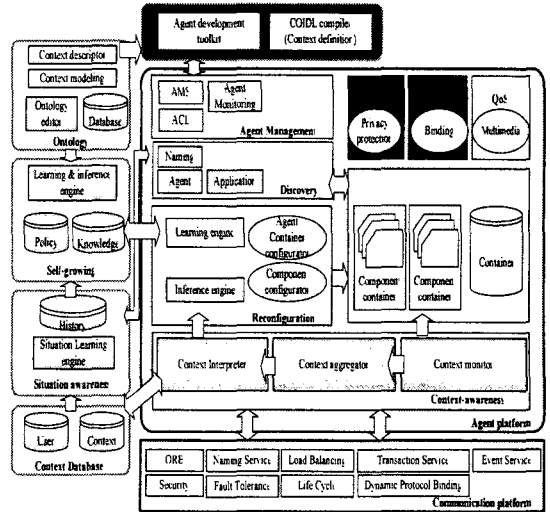


그림 2 CALM 구조

컨텍스트 지향 미들웨어는 CORBA Naming Service를 이용하여 그림 3처럼 각 센서들을 지역적으로 또는 센서 특성 별로 그룹을 지어 관리한다. 그룹내의 각 센서들은 네이밍 서버로부터 해당 채널 정보를 얻어와 해당 채널에 접속하고 컨텍스트들을 전송한다. 그러면 채널 서버에서 2계층 버퍼링을 이용하여 그 그룹 내에서 발생한 컨텍스트들이 네트워크 상황 및 센서들의 성능에 상관없이 그림 3에서의 경우1이 아닌 경우2처럼 컨텍스트 발생 순서와 컨텍스트 소비자에게 전송되는 순서가 일치하도록 지원함으로써 컨텍스트 소비자 즉, 시스템 또는 에이전트들이 일정 시점에서 신속하게 정확하게 상황인지 및 판단을 할 수 있도록 지원한다.

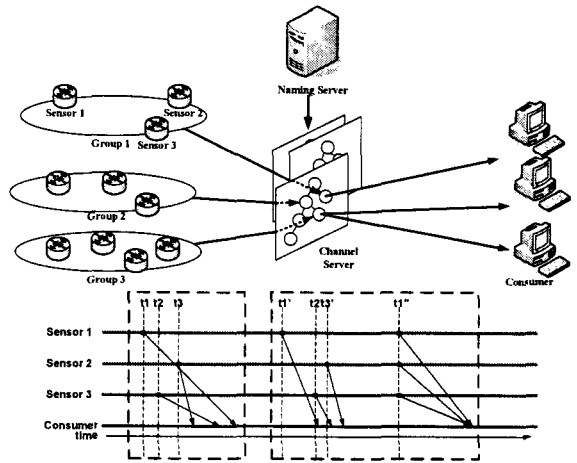


그림 3 컨텍스트 지향 미들웨어 서비스의 개념도

3.2 데이터 포맷

컨텍스트 지향 미들웨어에서, 데이터 포맷은 컨텍스트 발생 순서 및 전송 되는 순서를 일치시키는데 필요한 두 가지 필드, 타임스탬프(Timestamp)와 시퀀스넘버(Sequence Number)을 포함하여 여러 가지 필드를 가지게 된다. 타임스탬프와 시퀀스넘버는 컨텍스트 지향 미들웨어의 어댑터에서 할당되는데, 타임스탬프는 어댑터에서 자체적으로 생성해서 필드를 채우고, 시퀀스넘버는 어댑터가 네이밍 서버에 요청하여 발생한 컨텍스트에 해당하는 순서를 받아 채우게 된다. 여기서 네이밍 서버는 센서들을 그룹별로 관리를 하면서 그룹 내에서 발생한 컨텍스트의 순서 즉 시퀀스넘버를 그룹별로 관리한다. 이 두 필드는 전송되는 컨텍스트 데이터에 포함되어 이벤트 서버에서 컨텍스트 발생 순서와 전송 순서를 일치시키는데 사용된다.

3.3 이벤트 서버

이벤트 서버는 센서 그룹별로 해당 채널을 만들어 센서들을 관리하고 센서 그룹에서 전송되는 컨텍스트 데이터를 받아 컨텍스트 소비자 즉 시스템 및 에이전트들에게 전송한다. 여기서 이벤트 서버에는 각 그룹에 해당하는 채널들이 여러 개가 존재하며 각 채널에는 컨텍스트 모니터(Context Monitor)가 존재하면서 데몬 형태로 이벤트 채널의 큐를 모니터링하여 전송되는 컨텍스트의 발생 순서 및 전송 순서를 타임스탬프와 시퀀스 넘버로 일치 여부를 확인하여 이상이 있을 시 2 계층 버퍼링을 이용하여 순서를 맞추어 주게 된다. 그림 4는 이벤트 서버에 있는 채널의 구조를 나타내고 있다. 프락시 소비자는 센서가 채널에 접속하면 생성되어 각 센서 별로 매칭되고 센서로부터 오는 컨텍스트 데이터를 받아 마스터 큐(Master Queue)에 신뢰성(Reliability)을 위해 저장한다. 저장된 컨텍스트 데이터는 컨텍스트 모니터가 검사하여 이상이 없을 시에는 프락시 공급자에게 전달되고 이상이 있을 시에는 슬레이브 큐(Slave Queue)에 옮겨져서 순서를 맞춘 다음 프락시 공급자에서 전달된다. 그 후 프락시 공급자가 시스템 및 에이전트에게 전달한다.

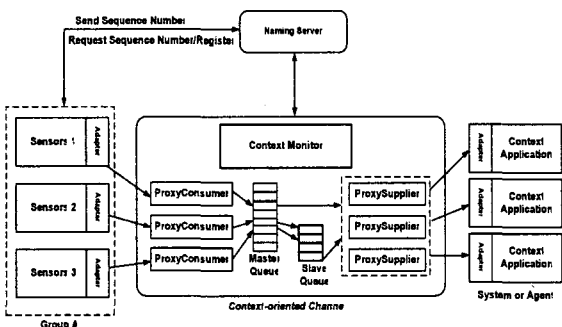


그림 4 채널 구조

4. 결론

본 논문에서는 네트워크 상황을 고려하지 않은 채 단순

이 컨텍스트 전송을 지원하는 미들웨어 시스템으로 인해 시스템들은 잘못된 상황 판단할 수 있는 문제점을 해결하는 컨텍스트 지향 미들웨어를 제안하였다. 이와 같이 제안된 2 계층 버퍼링 방식은 컨텍스트의 발생 순서와 전송 순서가 일치하도록 설계 되었다. 또한, 이를 활용하여 어떠한 네트워크 상황에서도 순서를 맞추어 줌으로써 컨텍스트의 효율적이고 신뢰성 있는 전송을 지원하게 될 것이다.

향 후, 본 논문에서 제안된 2계층 버퍼링 방식을 CALM에 적용함으로써 컨텍스트의 효율적이고 신뢰성 있는 전송을 지원하게 될 것이다.

참고문헌

- [1] Mirco Musolesi., et al.:_EMMA: Epidemic Messaging Middleware for Ad hoc networks, Personal and Ubiquitous Computing, ACM, Volume 10 Issue 1, Pages: 28 ~ 36, December 2005
- [2] Sun Microsystems, Inc., Java Message Service (JMS), 2006, <http://java.sun.com/products/jms/>
- [3] Object Management Group: CORBA Event Service Specification, v1.2, 2004. <http://www.omg.org/cgi-bin/doc?formal/2004-10-02>
- [4] IBM Corporation, 2006, <http://www-306.ibm.com/software/websphere/>
- [5] Microsoft Corporation MSMQ, 2006, <http://www.microsoft.com/msmq/>
- [6] Pietzuch, P.R.; Bacon, J.M.:Hermes: a distributed event-based middleware architecture, IEEE, 2-5, Page(s):611 ~ 618, July 2002
- [7] Yau, S.S.; Karim, F.; Yu Wang; Gupta, S.K.S.: Reconfigurable context-sensitive middleware for pervasive computing, IEEE, Volume 1, Issue 3, Page(s):33 ~ 40, 2002
- [8] Urs H. and Peter S.: Implementing access control to people location information, Proceedings of the ninth ACM symposium on Access control models and technologies, SESSION: Next generation access control models, Pages: 11 ~ 20 ,2004
- [9] Manuel Román, Christopher K., et al.: Gaia: A Middleware Infrastructure to Enable Active Spaces, IEEE Pervasive Computing, pp. 74-83, 2002
- [10] Anand Ranganathan, et al.:_MiddleWhere: a middleware for location awareness in ubiquitous computing applications,_Proceedings of the 5th ACM/IFIP/USENIX international conference on Middleware, SESSION: Middleware for mobility, Pages: 397 ~ 416 ,2004
- [11] Object Management Group: CORBA Naming Service Specification, v1.3, 2004. <http://www.omg.org/cgi-bin/doc?formal/04-10-03>
- [12] Youn, H.Y. et Al. CALM: An Intelligent Agent-based Middleware Architecture for Community Computing, SEUS 2006