

# 상황인지 미들웨어를 위한 에이전트 기반 자율 센싱 프레임워크

\*한만형, \*\*이승룡

\*경희대학교, \*\*경희대학교

\*[smiley@oslab.khu.ac.kr](mailto:smiley@oslab.khu.ac.kr), \*\*[sylee@oslab.khu.ac.kr](mailto:sylee@oslab.khu.ac.kr)

## Agent-based Autonomic Sensing Framework for Context-Aware Middleware

\*Manhyung Han, \*\*Sungyoung Lee

\*Kyunghee Univ., \*\*Kyunghee Univ.

### 요약

본 논문에서는 유비쿼터스 서비스를 제공하기 위한 상황인지 미들웨어로부터 상황 정보에 대한 수집 요청을 받아 사용자의 환경 정보를 수집하고 미들웨어로 전달하는 센싱 프레임워크의 기능을 개선하기 위해, 자율 컴퓨팅(Autonomic Computing)의 개념을 도입한 센싱 프레임워크인 에이전트 기반의 자율 센싱 프레임워크를 제안한다. 기존의 데이터 수집 방식인 센서에서 미들웨어로의 일방적인 데이터 전달 관계에서 벗어나, 센서들 간의 통신 및 센서 내에서의 전처리 과정을 두어 데이터 수집 및 전송에 대한 에너지 효율과 성능 향상을 기대할 수 있다. 미들웨어 측면에서는 수많은 센서들로부터 받은 컨텍스트를 처리해야 하는 중앙집중식 처리 구조에서 센서 내의 전처리 과정을 통해 한 단계 필터링 된 컨텍스트를 받아 처리함으로써 데이터 처리에 대한 부하를 분산시킬 수 있는 분산형 처리 구조로, 센서 측면에서 보면 수집된 정보를 무조건적으로 미들웨어로 전달하는게 아니라 센서내의 전처리 과정을 통해 보낼 데이터와 버릴 데이터를 추출함으로써 데이터 전송에 소요되는 에너지를 줄일 수 있다. 따라서 에이전트 기반의 센싱 프레임워크는 미들웨어 부하의 분산과 에너지 효율을 증가 그리고 고급 컨텍스트의 제공으로, 상황인지 서비스 제공을 위한 미들웨어의 성능 개선 효과를 기대할 수 있다.

### 1. 서론

지속적으로 발전하고 있는 컴퓨팅 패러다임은 이제 사용자가 컴퓨터나 네트워크를 인식하지 않는 상황에서, 시간과 장소에 구애를 받지 않고 다양한 서비스를 제공받을 수 있는 환경을 가능하게 하는 유비쿼터스 컴퓨팅의 시대에 와있다. [1] 사용자의 명시적인 요구나 명령 없이 사용자의 의도를 파악하고 이에 알맞은 서비스를 제공하기 위해서는, 사용자 환경정보 수집의 정확성과 수집된 정보들을 기반으로 적절한 서비스의 선택의 정확성이 요구된다. 환경정보 수집의 정확성은 다양한 센서들이 현재 사용자 주위 환경에 대한 정보(온도, 습도, 위치 등)와 같은 1차원적인 정보와, 미들웨어에서 이를 바탕으로 추론해낸 고차원의 정보(사용자의 현재 상태, 과거 정보를 기반으로 미래 상태의 예측 등)를 얻어낼 수 있다. 이렇게 얻어진 상태 정보, 즉 컨텍스트를 기반으로 하여 사용자에게 제공해야할 적절한 서비스를 선택하는 것이 현재 상황인지 미들웨어의 궁극적인 목적이며 역할이다.

상황인지 미들웨어의 최종 목적인 적절한 서비스의 제공에 기본 정보가 되는 1차원적인 정보, 즉 원시정보는 센서로부터 얻어져 미들웨어로 전송되어진다. 현재의 상황인지 컴퓨팅의 구조는 센서가 상황정보를

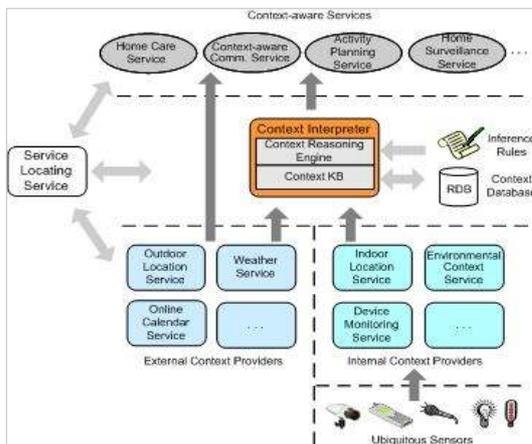
습득 후 이를 미들웨어로 바로 전달하고 이 데이터에 대한 모든 처리와 가공은 미들웨어에 집중되어 있다. 때문에 센서는 상황정보를 습득할 때 마다 무조건 전송하게 되며 미들웨어는 이 정보들을 전부 전송받아 모두 처리해야 한다. 이 과정에서 두 가지 문제점이 존재한다. 첫째로 센서의 무조건적인 수집 데이터의 전송에 따른 에너지 소비의 문제이다. 에너지의 효율적인 사용이 가장 큰 이슈가 되고 있는 센서 네트워크에서 환경 정보가 수집될 때마다 미들웨어로의 전송은, 센서의 에너지 소비 비율 중 가장 높은 부분을 차지하고 있는 무선 통신의 비율을 높이는 결과[2], [3]를 가져오므로 에너지 효율을 낮춰 센서의 수명을 줄이게 된다. 둘째는, 상황정보 처리의 중앙 집중식 구조로 인한 미들웨어의 성능 저하이다. 수많은 센서들로부터 많은 데이터들이 전송되는 미들웨어는 이 데이터들을 전부 처리해야 하며, 특히 센서들이 보내는 데이터가 중복 데이터이거나 원시레벨의 데이터인 경우 데이터 필터링과 여러 컨텍스트들을 조합하여 고급 컨텍스트로 만들어내는 과정들이 모두 미들웨어에서 이루어진다. 따라서 미들웨어로 모든 처리가 집중되는 현재의 구조는 미들웨어의 부하를 가중시켜 전체적인 성능 저하를 가져온다.

본 논문에서는 위와 같은 문제점을 해결하기 위해 에이전트 기반의 자율컴퓨팅의 요소를 적용한 센싱 프레임워크를 제안한다. 에이전트 개념의 도입으로 센서 내에서의 컨텍스트의 필터링과 같은 전처리 기술을 적용하고, 자율 컴퓨팅 요소의 개념 도입으로 센서 간 통신 기술을 적용하여 센서의 데이터 통신에 따른 에너지 소비를 줄이고 전 처리된 고급 컨텍스트의 전송으로 미들웨어의 부하를 줄일 수 있다. 전체적인 시스템에서 볼 때, 미들웨어 및 센서의 성능과 신뢰성의 증가는 사용자에게 적절한 서비스를 빨리 제공해야하는 유비쿼터스 컴퓨팅의 궁극적인 목적을 달성할 수 있도록 한다.

본 논문의 구성은 다음과 같다. 먼저 2장에서는 상황인지 미들웨어 및 센싱 프레임워크의 기술과 관련된 유사 연구를 살펴보고, 3장에서는 자율 컴퓨팅과 에이전트의 요소 기술 소개와 본 논문에서 제안하는 센싱 프레임워크의 어떠한 부분에 적용시킬 것인지 살펴본다. 그리고 4장에서는 에이전트 기반의 자율 센싱 프레임워크의 전체적인 구조와 세부 모듈에 대한 설명을, 마지막 5장에서는 제안한 프레임워크의 한계와 문제점 그리고 이를 극복하기 위한 향후 연구과제 및 방향을 제시한다.

## 2. 관련연구

유비쿼터스 서비스를 제공하기 위한 상황인지 미들웨어는 연구는 상당히 활발히 진행되고 있으며 실제 기초적인 단계나 상용화된 기술도 존재한다. 하지만 현재까지 제안된 대부분의 유비쿼터스 상황인지 시스템들은 주로 상황인지 자체의 메커니즘과 성능 개선에만 관심을 두고 있으며, 정작 정보를 제공하는 센서나 네트워크에서의 컨텍스트 처리나 전달과 관련된 기술에 대한 연구는 미진한 실정이다.



[그림 1] 상황인지 미들웨어의 전형적인 시스템 구조[6]

상황인지 미들웨어의 대표적인 연구 중 하나인 Context Toolkit 프로젝트[4]는 widget 기반의 구조로써 센서로부터의 습득한 데이터를 widget 형태의 일관된 형태로 저장하여 상위 레이어로부터 데이터의 은닉화를 피하고 있지만[5], 데이터의 습득과 전달 과정에서 센서는 기존의 단순한 데이터의 수집과 전달의 역할을 할 뿐이다. 서비스 지향형의 상황인지 미들웨어 아키텍처인 SOCAM 프로젝트[6] 또한 온톨로지 기반

의 컨텍스트 처리와 정형화된 컨텍스트의 표현에 대한 연구이며, 수집된 데이터의 처리만을 다룰 뿐 수집 과정에 대한 연구는 부족하다.[7] 그 외에도 CoBra[8], Gaia[9] 등과 같은 많은 유비쿼터스 컴퓨팅 및 상황인지 미들웨어에 대한 연구에서도 수집된 센싱 데이터의 표현 및 처리에 대한 연구만 있어, 상황 정보의 습득과 전달에 대한 연구가 요구된다.

## 3. 자율 컴퓨팅의 요소 및 에이전트의 정의

자율 컴퓨팅(Autonomic Computing)의 가장 중요한 요소는 스스로 관리하는 기능이다. 시스템이 점차 복잡해지고 대형화 되어 감에 따라 빠르게 변화하는 다양한 구성 요소들을 관리하고 안정성 및 신뢰성을 유지하는 것이 힘들게 되었다. 이에 대한 해법으로 Self-configuring,



[그림 2] 자율 컴퓨팅의 네 가지 핵심 요소

Self-healing, Self-protecting 그리고 Self-optimizing의 특성을 가지는 자율 컴퓨팅의 개념[10]이 대두되었다. 시스템이 대형화 되어가면서 수많은 구성요소들의 관계들이 복잡해지고 시스템이 복잡해질수록 더 이상 시스템 관리자가 일일이 관리할 수 없게 된다. 이에 대한 해법으로 스스로 구성요소를 조직하고(Self-configuring) 시스템의 안정성을 위해 스스로 복구하며(Self-healing), 일괄적인 관리가 힘들어지면서 외부의 침입이나 오류를 스스로 방어하고(Self-protecting) 수많은 구성요소들의 유기적인 결합을 통해 스스로 최적화하여 성능을 유지하는(Self-optimizing) 자율 컴퓨팅의 기술이 생겨난 것이다.

에이전트는 자율성, 추론, 학습 그리고 지식의 교류와 같이 인간의 특징들을 가지고 있는 소프트웨어 개체로써[11], 대형 시스템이나 자가 관리 및 구성이 요구되는 시스템에 적합하다. 이러한 장점들 때문에 분산 환경에서의 컴퓨팅 자원의 효율적인 관리를 위해 널리 쓰이고 있다. 본 논문에서 제안하는 센싱 프레임워크는 이 같은 장점을 지닌 에이전트 기반의 자율 센싱 프레임워크로, 궁극적으로 추구하는 목표인 센서 내 전처리를 통한 미들웨어 부하 분산과 자가 추론 및 필터링을 거친 센싱 데이터의 선택적 전송으로 인한 에너지 효율성 향상을 이룰 수 있다.

## 4. 에이전트 기반 자율 센싱 프레임워크

상황인지 컴퓨팅의 구성요소인 센싱 프레임워크에서 센서는 사용자의 환경 자료를 수집하고 전송하는 역할을 할 뿐 데이터의 처리는 모두 미들웨어로 넘겨졌다. 이와 같이 모든 처리를 Back-end 시스템으로 넘기는 인프라 기반의 구조는 데이터를 전송하는 센서의 수가 적을 경우 문제가 없지만, 센서의 수가 많아지고 미들웨어와 센서의 구성이 복잡하며 유동적일 경우 여러 가지 문제가 있다. 우선 처리해야할 데이터가 한곳으로 집중되어 부하가 집중되어 전체적인 성능의 저하를 가져온다.

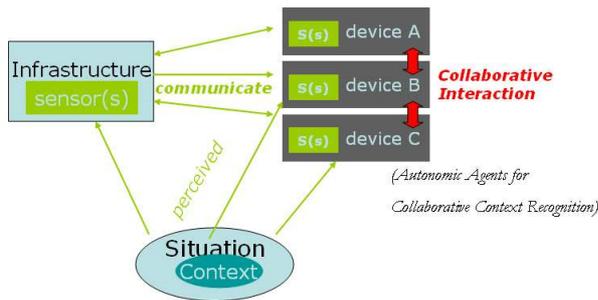
또한 센서와 미들웨어가 직접적으로 연결되어있어 센서의 다양한 망의 구성(Ad-hoc)이 불가능하며, 미들웨어에서 처리되기 전까지는 무의미한 형태의 데이터로 어플리케이션에서의 직접적인 사용이 불가능하다는 단점이 있다.



[그림 3] 미들웨어 기반의 센싱 프레임워크

반면 컨텍스트의 처리를 Back-end 시스템으로 넘기지 않고 센서 디바이스 스스로 상황정보를 수집하고 처리하는 미들웨어에 의존적이지 않은 구조인 경우, 센서의 에너지 소모가 크고 센서의 작은 연산 처리 능력으로 인해 고급 상황정보의 추론이 불가능하다.

본 논문에서 제안하는 에이전트 기반의 자율 센싱 프레임워크에서 센서는 자료를 수집하고 전달하는 기존의 전형적인 센서의 개념을 좀 더 확장하여, 자가 관리 및 처리가 가능한 에이전트 즉, 작은 미들웨어가 센서 디바이스에 올라간 것으로 위에서 제시한 단점들을 극복할 수 있다. 수집된 데이터의 전처리 및 주위 센서들과의 통신을 통해 한 단계 필터링 된 데이터를 미들웨어로 전송하므로 미들웨어의 부하를 줄이고 통신 비용을 낮춰 에너지 효율을 높일 수 있다. 또한 센서 내 데이터 처리를 통해 미들웨어를 거치지 않고 직접 어플리케이션과 통신하여 직접 서비스를 제공할 수 있는 장점이 있다. 그리고 센서의 작은 크기인 컴퓨팅 파워의 제약을 미들웨어와 연동하여 해결함으로써 간단한 처리는 센서 내에서, 고급 컨텍스트의 생성을 위한 복잡한 추론이 필요한 경우에는 미들웨어에서 처리하여 주어진 컴퓨팅 파워를 효율적으로 사용할 수 있다. 나아가 센서 간 통신이 가능해지므로 다양한 센서 망 구성이 가능해져 좀 더 사용자 중심의 서비스 제공이 가능해진다.

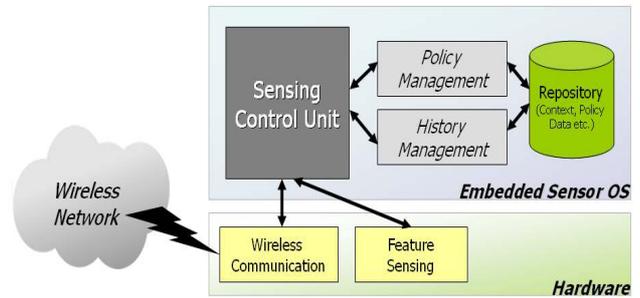


[그림 4] 에이전트 기반의 자율 센싱 프레임워크

상황인지 미들웨어에 에이전트 기반 자율 센싱 프레임워크의 도입을 위해서는, 실제 데이터를 수집하고 전송하는 물리적인 센서에 올라가는 내장형 운영체제 시스템에 적용이 되어 기존의 센서 디바이스의 기능들을 효율적으로 이용하여야 한다.

기존의 센서 역할은 환경 데이터를 수집 한 후에 이를 미들웨어로 전달하는 싱크노드로서의 전송을 하는 것으로 역할이 한정되어 있었다. 온

도, 습도, 조도 등과 같은 정보를 수집하여 특정 값으로 변환한 뒤, 이 값을 ZigBee, RFID와 같은 통신프로토콜을 사용하여 전송하는데 그쳤지만, 에이전트의 개념이 도입된 센서는 현재 수집되어 디지털화 된 값을 한 단계 필터링 하는 과정이 있다. 이 과정에서 센서는 중복 데이터 및 불량 데이터의 제거, 간단한 처리가 요구되는 컨텍스트의 가공 등과 같은 과정을 거쳐 전송할 데이터와 버릴 데이터를 결정한다. 예를 들어, 현재 수집한 값을 저장되어 있는 이전 값과 비교하여 일정한 수준의 변화가 없으면 버리거나 임계값을 두어 이 임계치에 다다르지 못한 값을 전송하지 않는다. 또한 원시 레벨의 컨텍스트를 센서간 통신을 통해 보다 고차원의 컨텍스트를 생성하여 미들웨어에 전송하는 방법들이 있다. 이를 통해 전송 빈도를 낮춰 에너지 효율 높이고, 미들웨어의 컨텍스트 처리 횟수를 낮춰 부하 분산의 효과를 가진다.



[그림 5] 에이전트 기반의 센서 아키텍처

그림 5는 본 논문에서 제안하는 에이전트 기반의 센서에 대한 개념적인 내부 구조도로써, 센서 내 작은 운영체제인 Embedded Sensor OS 부분과 실제 통신을 하고 데이터를 수집하는 Hardware 부분으로 나뉘어져 있다. 하드웨어 계층은 ZigBee, RFID와 같은 통신 모듈과 온도, 조도, 습도 등과 같은 환경정보 센서들로 이루어져 있으며 이 하드웨어들은 Embedded Sensor OS의 제어에 따라 작동된다. Embedded Sensor OS 계층의 구성요소는 다음과 같다:

- Sensing Control Unit (SCU): 센서의 중심부로서 데이터수집, 전처리, 통신, 관리 등 모든 기능을 총괄
- Policy Management: 데이터수집 및 처리에 대한 정책(임계값, 데이터 삭제 기준 등)관리
- History Management: 수집된 이전 데이터의 관리
- Repository: 수집된 데이터, 정책 데이터 등을 저장

센서 시스템을 모두 총괄하는 SCU는 하드웨어 계층의 무선 통신 모듈을 조작하여 센서와 미들웨어와의 통신은 물론, 필요시 다른 센서에 데이터를 요구하거나 전송할 수 있다. 그리고 SCU는 수집된 데이터의 처리가 이루어지는 곳으로 정책 관리자로부터 정책을 받거나 갱신하며 이전 데이터들을 받거나 갱신하여 센서 내 전처리 및 선별적인 통신 그리고 에너지 관리 등이 이루어진다. 즉 자율성, 센서 내 데이터 처리, 센서 간 통신 등이 가능한 에이전트 기반 센서의 역할을 할 수 있게 되는 것이다.

## 5. 결론

에이전트 기반의 자율 센싱 프레임워크는 본 논문에서 제안하는 두 개의 장점인 미들웨어의 부하 분산과 에너지의 효율적인 이용을 만족시키는 개념으로, 결국 유비쿼터스 컴퓨팅의 최종 목적인 사용자에게의 신속하고 정확한 서비스 제공을 만족시킨다. 하지만 에너지 효율을 높이기 위한 센서의 각종 연산과 이에 사용되는 에너지의 비율이 아직 정량적으로 조사되어 있지 않다. 비록 센서에서의 가장 큰 에너지 소비는 통신에 의한 것이지만 이를 줄이기 위한 각종 연산들도 에너지를 소비하는 행위이므로 이에 대한 정성적, 정량적이 조사를 통한 적절한 사용이 요구된다. 또한 부하 분산을 위해 제안한 본 프레임워크는 센서의 수가 상대적으로 적은 스마트 홈, 스마트 오피스와 작은 공간에서는 효율적이지 못하다. 많은 사용자와 센서, 잦은 네트워크의 변화와 같은 역동적인 공간에서의 상황인지 미들웨어에 적합한 기술로써, 향후 각광 받고 있는 홈 네트워크와 같은 곳에서의 효율적인 적용을 위한 연구가 요구된다.

본 논문에서 제안한 에이전트 기반 자율 센싱 프레임워크는 현재의 범용적인 상황인지 미들웨어의 부하 분산과 센서의 에너지 효율 향상을 통해 전체적인 성능 향상을 제공하고 있다. 제안한 프레임워크가 현재의 센싱 프레임워크에 바로 적용하기에는 미들웨어의 적합성 여부와 에너지의 효율성 부분들이 장애가 있을 수 있다. 하지만 모든 사물에 센서의 개념을 도입한 스마트 오브젝트의 패러다임이 다가오고 있고 에너지 지원 자체에 대한 많은 연구와 효율적인 사용에 대한 많은 연구들이 활발히 진행되고 있어 향후 유비쿼터스 컴퓨팅의 발전을 위한 핵심적인 기술로써 기대된다.

## 참 고 문 헌

- [1] Mark Weiser "The Computer for the 21st Century", Scientific American, vol.253, no.3, pp.94-104, Sep. 1991
- [2] Schwieger, K., Kumar, A., Fettweis, G. "On the impact of the physical layer on energy consumption in sensor networks", Wireless Sensor Networks, 2005. Proceedings of the Second European Workshop on 31 Jan.-2 Feb. 2005, pp.13-24
- [3] Miller, M.J., Vaidya, N.H. "Minimizing energy consumption in sensor networks using a wakeup radio", Wireless Communications and Networking Conference 2004, 21-25 March 2004, IEEE Volume 4 pp.2335-2340
- [4] Context Toolkit Project, <http://www.cs.cmu.edu/~anind/context.html>
- [5] Daniel Salber, Anind K. Dey and Gregory D. Abowd. "The Context Toolkit: Aiding the Development of Context-Enabled Applications", In the Proceedings of the 1999 Conference on Human Factors in Computing Systems (CHI '99), Pittsburgh, PA, May 15-20, 1999, pp. 434-441
- [6] SOCAM Project, [http://www.comp.nus.edu.sg/~gutao/gutao\\_NUS/SOCAM.htm](http://www.comp.nus.edu.sg/~gutao/gutao_NUS/SOCAM.htm)
- [7] T. Gu, H. K. Pung, D. Q. Zhang. "A Service-Oriented Middleware for Building Context-Aware Services", Elsevier Journal of Network and

Computer Applications (JNCA), Vol. 28, Issue 1, pp. 1-18, Jan. 2005

- [8] CoBra Project, <http://cobra.umbc.edu>
- [9] Gaia Project, <http://gaia.cs.uiuc.edu>
- [10] Sterritt, R., Bustard, D. "Autonomic Computing - a means of achieving dependability?", Engineering of Computer-Based Systems, 2003. Proceedings. 10th IEEE International Conference and Workshop on the 7-10 April 2003, pp. 247-251
- [11] K. Sycara, K. Decker et al., "Distributed Intelligent Agents", IEEE Expert, pp. 36-45, December 1996