

# 무선 센서 네트워크에서 전송 품질 개선을 위한 메타 데이터 설계 제안

전혜경\*, 박양재\*\*

인하대학교 컴퓨터공학과\*, 가천대학교 컴퓨터공학과\*\*

## Metadata Design Proposal for Improving the Transmission Quality in Wireless Sensor Network

Hye-Kyoung Jeon\*, Yang-Jae Park\*\*

Dept. of Computer Science & Information Technology, Inha University\*

Dept. of Computer Engineering, Gachon University\*\*

**요 약** 무선 센서 네트워크 어플리케이션의 기능 요구를 달성하기 위한 주요한 수단은 실세계에서 얻는 데이터이므로 어플리케이션 개발 과정에 있어서 데이터에 관한 품질 개선은 중요한 과제이다. 어플리케이션 개발 시 저비용의 프로토타입을 개발하고 빠른 기간 내에 품질을 개선하기 위한 것과 상세한 품질개선에 대한 요구 사항을 만족하는 것이 가능해야 한다. 그러나 기존의 개발방법은 추상도를 일관되게 사용하기 위한 기술능력과 기술의 용이성간의 이율배반적인 관계로 인하여 동시에 두 가지를 만족할 수 없다. 본 논문에서는 추상도가 다른 복수의 모델링 언어를 병용해 사용함으로써 데이터 품질 개선을 지원하는 메타 데이터 설계를 제안한다. 본 논문에서는 설계대상마다 적절한 추상도를 사용하도록 모델링 언어를 정의하고 이들의 사용방법을 제안한다. 또한 상위모델 설계를 하위 모델에 반영함으로써 실제와 근접한 자동모델교환규칙을 정의할 수 있도록 한다.

**주제어** : WSN, 프로토타입, 모델링, 메타데이터

**Abstract** In the application development process is an important task to improve the quality of the data is the principal means for achieving the functional requirements of the wireless sensor network applications because it is the data obtained in the real world. It should be possible to meet the requirements for the quality improvement to those detailed for improving the quality in a short period of time for developing a prototype of an application and low cost. However, development of the existing methods can not be satisfied simultaneously in order to use a consistent abstraction because antinomical relation between ability of description and ease of description. In this paper, we propose a meta-data designed to support the improvement of data quality through abstract modeling language is also used in combination to a multiple of the other.

**Key Words** : WSN, ProtoType, Meta Data, Modeling

Received 19 June 2014, Revised 31 August 2014  
Accepted 20 September 2014  
Corresponding Author: Yang-Jae Park(Gachon University)  
Email: parkyj@gachon.ac.kr

© The Society of Digital Policy & Management. All rights reserved. This is an open-access article distributed under the terms of the Creative Commons Attribution Non-Commercial License (<http://creativecommons.org/licenses/by-nc/3.0>), which permits unrestricted non-commercial use, distribution, and reproduction in any medium, provided the original work is properly cited.

ISSN: 1738-1916

## 1. 서론

무선 센서 네트워크는 센서의 측정 데이터를 무선통신에 의해 전파시키고, 유선과 비교해서 설치시의 제약 사항이 적기 때문에[1], 실세계와 밀접하게 연결해야 하는 시스템의 관측기반으로써 중요한 역할을 갖는다. 센서 노드는 프로그램풀(program pool)이 있고 쉽게 확장할 수 있는 어플리케이션을 설치할 수 있으므로 다목적으로 이용될 수 있는 가능성이 있다. WSN 어플리케이션은 주로 센서 데이터의 계측과 그 데이터의 처리 로직으로서 제공된다. 최근 전형적인 WSN 어플리케이션은 정기적인 감시를 수행하는 [2,3]이 있다. 또한 건축물이나 야외환경 환경감시 등에서 이용되어지고 있다[4].

WSN 어플리케이션 개발자는 기능요구와 함께, 비기능 요구를 만족하도록 설계를 할 필요가 있다. WSN 어플리케이션에서는 취득한 데이터에 관한 품질이 중요한 비기능적 요구가 되는 것으로 알려져 있다[5]. 데이터품질은 WSN의 배치방법과 장애물이 있는 실제 환경에 강하게 영향을 받는 것으로 알려져 있다[6]. 예를 들어, 건축물 등 장애물이 많은 환경에서는 무선통신실패에 의해 데이터 손실률이 증가하기 때문에, 통신방식의 설계를 주의 깊게 할 필요가 있다. 그러나 데이터 품질개선을 위한 설계과제는 실행환경에 따라 변하기 때문에, 설계 시에 특정하는 것이 어렵다. 그래서 데이터품질에 대한 요구를 만족하기 위해서, 프로토타입을 설계하고, 프로토타입의 실행결과를 기초로 하는 데이터 품질 개선이 필요로 하게 된다.

프로토타입 개발은 어플리케이션 개발 초기 단계에서 프로토타입을 개발하여 실행 환경 하에서 개발자는 대상 환경에 있어서의 데이터품질을 측정하고 어떻게 하면 데이터 품질을 개선할 수 있을지를 확실하게 할 수 있다. 데이터 품질개선은, 데이터 흐름, 통신방식, 태스크할당이 있는 WSN 어플리케이션에 필요한 설계상의 결정을 변경하는 것으로 데이터 품질 개선을 생각해 볼 수 있다. WSN 어플리케이션은 요구된 로직에 대응하는 데이터 흐름을 분산 네트워크상에 실현한다. 즉, 개발자는 설계 단계에 있어서 1) 데이터 흐름 설계, 2) 데이터 흐름의 다중 태스크의 분산과 태스크 간의 통신방식결정으로 만들어지는 태스크 구조 설계 3) 분산된 태스크의 노드에 할당된 상세설계 등 세 가지를 수행할 필요가 있다.

데이터 품질개선을 쉽게하는 WSN 어플리케이션 개발에서는 아래 두 가지 사항이 요구된다.

- 요건 1. 프로토타입을 저비용 개발 가능해야 할 것
- 요건 2. 통신방식, 태스크 할당의 설계변경을 포함한 구체적인 품질 개선이 가능해할 것

현재는 위 두 가지 요건을 동시에 만족하는 WSN 어플리케이션을 개발하는 것은 어렵다. 현재 일반적인 WSN 어플리케이션에서는 WSN의 전문가가 각각의 노드의 동작을 기술하는 도메인 특화 언어(DSL: Domain-Specific Language)을 이용하고 있다[7]. 이 방법에서는 로직, 통신방식, 태스크 할당 설계 모두를 노드가 동작하도록 설치해야하고, 기술이 너무 장황하고 복잡해서 요건1을 만족할 수 없다. 이러한 부담을 감소시키기 위해서 통신 방식, 태스크 할당으로서의 요소를 추상화하는 DSL이 제안되어지고 있다[3]. 그러나 이 DSL에서는 통신방식이나 태스크 할당의 요소를 추상화하기 위해 데이터품질 개선이 제한되어 요건 2를 만족하지 않는다. 이처럼 단일 DSL을 쉽게 개발하는 것은 DSL에서 기술 가능한 요소의 폭넓음과 기술의 용이성 간에 이율배반적인 관계가 존재하기 때문에, 앞서 기술한 요건 1,2를 동시에 만족할 수가 없다.

본 논문에서는 데이터 품질 개선을 지원하기위해 WSN 어플리케이션 개발에 관해서 요건 1, 2를 동시에 만족하는 모델 구동 개발(MDD : Model-Driven Development) 프로세스를 제안한다. MDD는 어플리케이션을 추상적인 모델로서 기술하고 보다 상세한 모델로의 교환, 수정을 수행하는 개발방법이다. 제안한 개발 프로세스에서는 데이터 흐름설계, 태스크 구조 설계, 상세설계 각각에 맞는 추상도 도메인 특화 모델링 언어(DSML : Domain-Specific Modeling Language)을 다중정의 하고 병용한다. 이에 따라 프로토타입 개발 시에는 데이터 흐름설계만을, 품질개선 시에는 3개의 설계를 알맞은 추상도에 각각 적용함으로써, 저비용의 프로토타입 개발과 상세한 품질 개선을 양립할 수 있게 한다.

본 연구에서는 이 설계 교환을, 추상도간을 연결하는 자동 모델교환규칙을 정의하였다.

## 2. 관련연구

### 2.1 기존연구

본 연구와 동일한 형태로 WSN 어플리케이션 개발에서 MDD를 적용해서 개발효율을 향상시키는 연구는 계속해서 존재해 왔다. [8]에서는, 특정 미들웨어나 OS에 의존적이지 않는 WSN용 모델링 언어를 정의하고 있다. 이 모델링 언어는 같은 종류의 태스크를 담당하는 노드집합의 동작을 표현하고, Group-level에 속하는 모델링 언어라고 말할 수 있다. 기술된 모델은 단계적으로 NesC의 코드로 자동 교환된다. 또 WSN에 대한 개발 프레임워크 [8]에서는 각 노드의 동작을 표현하는 모델링 언어를 정의하고, 그것은 Node-level에 속하는 모델링 언어라고 말할 수 있다. 작성된 모델은 [8]의 연구와 같은 방식으로 노드로 자동 교환된다. 이러한 연구는 모델로부터의 코드 생성에 의한 생산성 향상을 실현할 수 있지만, 단일 추상도 모델링 언어밖에 제공하지 않고, 단일의 DSL을 이용한 경우의 개발과 같은 형태의 문제를 가지고 있다. 이것에 대해, 본 논문에서는 세가지 추상도의 DSML을 적용한 단계를 이용한 개발 프로세스를 제안하고 그 문제를 해결한다.

본 개발 프로세스와 같이, 재설계를 반복하는 것으로 품질 개선을 수행하는 개발방식을 지원하는 연구도 존재한다. [9]은 특징 모델을 이용한 개발트리이다. 개발자는 [9] 상에서 어플리케이션의 종류나 네트워크 구성의 특징을 선택하는 것으로 모델을 작성한다. 또한 [9]에서는 작성한 특징 모델로부터 성능추출을 수행하는 기능을 제공하고 있다. 또, [10]에서는 프로토콜 선택의 어려움에서 착안해, MAC 라우팅 프로토콜 방법을 기술할 수 있는 트리를 제공하고 있다. [10]는 선택된 프로토콜의 성능추출을 수행하는 기능도 제공하고 있다. 이들 연구에서는 모델을 이용한 설계와 그 설계의 성능의 추정기능을 조합한 트리를 제공하는 것으로 WSN 어플리케이션의 재설계의 반복에 의한 품질 개선을 지원하고 있다. 그러나 이 트리에 있어서 모델은 네트워크 구성이나 프로토콜의 설계에 초점을 맞추고, 로직을 기술할 수는 없다. 본 논문에서는 세가지 추상도의 DSML을 이용한 것으로, 로직을 데이터 흐름으로 기술한 것과 동시에 통신방식, 태스크 할당의 설계도 기술가능하다.

### 2.2 무선 센서 네트워크에서의 DSL

기존의 많은 WSN 어플리케이션은 WSN에 DSL을 이용해 개발되어지고 있다. 또한 WSN에서의 DSL은 개발자에게 주어진 추상화의 시점으로부터 3가지로 분류가 가능하다[3]. 본 논문에서는 이 세 가지 추상화를 데이터 흐름레벨, 그룹레벨, 노드레벨이라고 부른다.

데이터 흐름레벨에 속한 DSL은 로직만을 기술하는 DSL이다. [11]에서는 WSN 전체를 데이터베이스로 보고 SQL을 기본으로 한 형식으로 어플리케이션에게 필요한 데이터와 그 데이터에의 집약처리를 기술한다.

그룹레벨에 속한 DSL은, 노드그룹의 구성과 매크로적인 동작을 기술하는 DSL이다. [12]에서는 지리적으로 가까운, 같은 종류의 태스크를 실행하는 노드집합을 region라고 정의하고, region의 구성기준(예를 들면 거리)와 region의 동작을 기술한다.

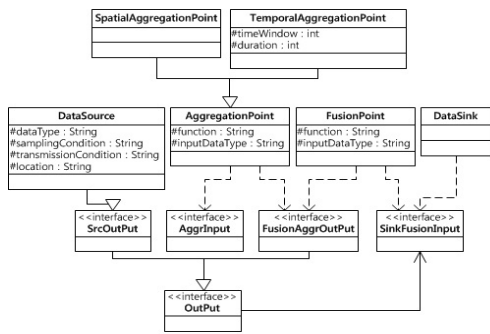
노드레벨에 속한 DSL은, 각각의 노드의 동작을 기술하는 DSL이다. [13]에서는 각각의 노드에 대해 계측, 통신 등의 처리를 기술한다.

## 3. 데이터 품질개선을 위한 메타 데이터 설계

### 3.1 DSML의 설계

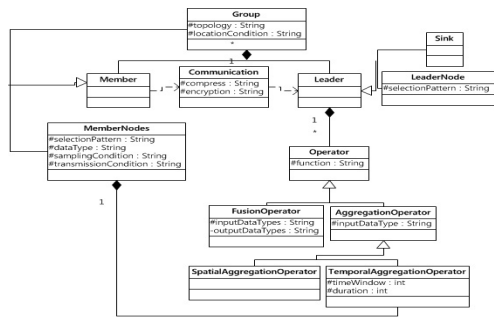
본 개발 프로세스에서는 세가지 DSML을 공용한다. 세 개의 DSML은 각 설계단계에 적절한 추상도를 대응하고 가장 전형적이고 정기적 계측, 통신을 수행하는 WSN 어플리케이션을 대상으로 설계한다. 또 대상이되는 WSN은 [2,3]에 의하여 가장 전형적이고, 노드가 이동성을 갖지 않고 데이터 집계지역(Sink)가 단일한 것으로 한다.

DataflowML은 WSN 어플리케이션의 로직으로써 네트워크에 비존재적인 데이터 흐름 설계를 기술하기 위한 DSML이다. DataflowML의 메타데이터를 [Fig. 1]에 표시한다. DataflowML에서 데이터의 출력을 수행하는 DataSource, 취득한 데이터에 대해서 집약적인 융합처리를 수행하는 AggregationPoint와 FusionPoint, 최종적으로 데이터를 수집하는 DataSink와 이 요소들을 접속하고 데이터 흐름을 표현하는 링크를 구성요소로 한다. DataflowML을 사용하는 것으로, WSN 어플리케이션으로 실현하고 싶은 로직을 데이터 흐름으로 간소하게 기술가능하다.



[Fig. 1] Meta Model of DataflowML

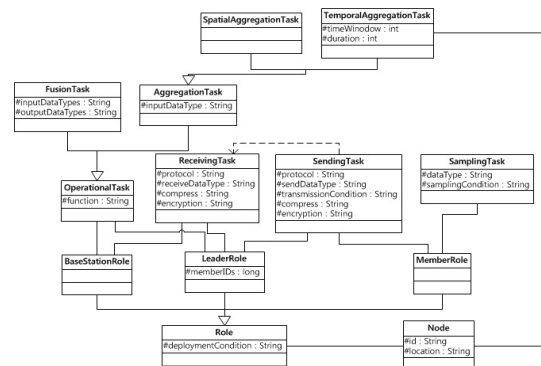
GroupML은 태스크 구조 설계를 노드그룹의 구성으로서 기술되어지기 위한 DSML이다. GroupML의 메타 모델을 [Fig. 2]에서 보인다. 노드그룹은 같은 태스크를 담당하는 노드집합이고 각 그룹은 리더와 멤버형의 구성을 상정하고 있다. GroupML에서는 Group을 가장 기본적인 요소로 하고, 각 그룹은, 리더로서 WSN 전체의 데이터를 모으는 Sink가 그룹 내의 데이터를 모으는 LeaderNode을, 멤버로서 자식 요소의 Group이 계층처리를 담당한 MemberNodes을 각각 담당한다. GroupML에서 모델 파라미터와 그래프 구성을 수정하는 것으로 통신방식이나 그룹구성 설계가 표현 가능하다. 예를 들면 데이터 압축을 이용하지 않는 경우는 파라미터 compress의 수치를 NONE으로 하고, 이용하는 경우에는 압축 알고리즘에 대응하는 어휘로 설정하는 것이 표현 가능하다. 또, 그룹 클러스터링을 이용하는 경우는 모델의 그래프 구성을 계층적으로 하고, 이용하지 않는 경우 모든 요소를 하나의 그룹 내에 포함하도록 하는 것을 표현할 수 있다.



[Fig. 2] Meta Model of GroupML

GroupML을 이용하는 노드그룹의 구성을 기술하는 것으로, 그룹단위에서의 태스크 분할과 그룹 간/그룹 내에서의 통신방식을 기술할 수 있다.

NodeML은 상세설계를 각각의 노드의 동작으로 기술하기 위한 DSML이다. NodeML의 메타 모델을 [Fig. 3]에 나타내었다. NodeML에서는 노드에의 태스크 할당을 표현하기 위해 물리적인 노드를 나타내는 개념(Node)와 각 노드에 할당된 태스크를 나타내는 개념(Task)을 구성요소로 한다. 또, 각 노드는 복수의 태스크를 갖는 것으로 하나의 역할을 갖기 때문에, 이것을 Role로 해서 NodeML의 구성요소로 한다. NodeML을 이용하는 태스크와 역할을 각각의 노드에 할당하는 것으로, 개별 노드의 동작을 태스크 할당으로서 기술 가능하다.



[Fig. 3] Meta Model of NodeML

### 3.2 교환규칙의 설계

#### 3.2.1 데이터 흐름레벨에서 그룹레벨로의 교환

우선, 데이터 흐름레벨부터 그룹레벨로의 자동교환에서는 데이터 흐름을 네트워크상에서 노드 집합의 동작으로 교환한다. 이 교환에서는 입력 모델의 DataSink부터 처리를 개시하는 것을 상정하고, [Fig. 4]에 표현된 것처럼 링크를 더듬어 요소를 탐색, 발견된 요소에 대응해서 그룹레벨 모델을 생성해 간다. 그룹레벨 모델 생성 시에 결정해야 할 필요가 있는 설계는 교환규칙에서 초기설정을 부여하고 있다. 예를 들면, 토폴로지는 초기 값으로 트리형으로 설정하며 부여된 초기설정은 <Table 1>과 같이 나타내었다.

<Table 1> Default Value of Group-Level

Design of Group Level	Initial Setup Value
Network Topology	Tree Type
Data Compression	No
Encryption	No
Condition of Selecting Leader	Random
Condition of Selecting Member	All
Processing Location	Inside Network
Clustering	Yes

이 초기설정들은 그룹레벨에서 취한 설계상의 선택지 중에 하나가 되는데, 예를 들어 이상한 토폴로지를 이용한 경우에는 교환 후에 수동으로 초기설정을 변경할 필요가 있다. 교환규칙에 있어 요소 간 대응 관계와 미결정사항의 초기 설정을 부여 받음으로서, 데이터 흐름레벨부터 그룹레벨로의 자동교환을 실현한다. [Fig. 4]은 입력모델의 탐색순서와 속서의 대응, 그리고 초기설정 방식에 대해 나타내고 있다.

```

1: function GENERATEGROUPMODEL(dataflowModel)
2:   groupModel ← makeGroup(dataflowModel)
3:   return groupModel
4: end function
5: function MAKEGROUP(inputModel)
6:   head ← inputModel.root
7:   group ← vertex with Group
8:   if head.equals(DataSink) then
9:     group.append(LeaderNode)
10:    group.append(MemberNodes)
11:  else
12:    if head.equals(DataSink) then
13:      group.append(Sink)
14:    else if head.equals(FusionPoint) then
15:      group.append(LeaderNode with
16:        FusionOperator)
17:    else if head.equals(AggregationPoint) then
18:      group.append(LeaderNode with
19:        AggregationOperator)
20:    end if
21:    for each child in head.children do
22:      group.append(makeGroup(child))
23:    end for
24:  end if
25:  return group
26: end function
    
```

[Fig. 4] Algorithm1 of Data Flow Level to Group Level

교환에서는 입력이 되는 데이터흐름레벨 모델(을 DataSink부터 탐색한다.(알고리즘 1행 1-2). 우선 현재 탐색 중인 입력 모델 내에서 루트 요소를 얻고(6행), 교환 전 정보를 보유하기 위한 Group을 작성한다(7행). 이 Group의 작성 시에는, 수동교환의 경우는 사람의 손으로 토폴로지의 속성 topology의 값을 결정할 필요가 있지만,

본 자동교환에서는 이 속성에, <Table 1>에 나타나는 초기 설정 값을 부여한다. 이 후 처리는 탐색요소에 대응해서 나뉘게 된다.

우선 DataSink을 다루기 위해, 이것에 대응하는 Sink를 7행에서 작성한 Group에 추가한다(12-13행). 이어서 입력 모델의 하위로의 탐색을 진행하기 위해, 하위의 요소군의 취득과 이것을 이용한 Group의 작성을 수행한다(19-21행). 이 예에서는 DataSink에 접속되는 SpatialAggregationPointdglk의 모델 요소를 취득하고, 재귀적으로 makeGroup을 호출한다.(20행)

이후에는 앞에서 기술한 DataSink의 경우와 같이, 모델 내의 루트 요소를 취득(6행), Group을 만들고(7행), SpatialAggregationPoint에 대응하는 LeaderNode와 SpatialAggregationOperator를 Group에 추가한다(16-18행). 이것은 자동교환규칙에서는 데이터처리를 네트워크 내에서 실행하는 초기 설정을 부여하기 위해서이다. Leader 작성 후에는 DataSource을 입력으로 해서 makeGroup을 호출한다.

루트요소가 DataSource인 경우에는 LeaderNode와 DataSource에 대응하는 속성을 갖는 MemberNodes를 Group에 추가하고(8-10행), 이 그룹을 리턴 값으로 하여 종료한다(23행). 이것은 데이터흐름레벨 모델에서 있어서는 DataSource보다 하위에 접속된 요소가 존재하지 않기 때문이다. 이 때, LeaderNode, MemberNodes의 할당 조건인 속성 selectionPattern과 통신에 관한 요소 Communication의 속성에는 <Table 1>에 표시한 초기설정을 준다. makeGroup의 리턴 값은 호출한 원 makeGroup 내에서 작성된 Group에 자식 그룹으로 추가한다(20행). 이처럼 makeGroup의 리턴 값은 Group에 추가하고, 재귀적으로 계층적인 그룹을 생성한다. 이 재귀적 실행결과, 그룹레벨 모델을 자동교환에 의해 얻을 수 있다(3행)

### 3.2.2 그룹레벨에서 노드레벨로의 교환

그룹레벨에서 노드레벨로의 자동교환에서는, 노드 집합의 동작을 각각의 노드의 동작으로 교환하고 있다. 이 교환에 있어서는, 우선 그룹레벨 모델의 구성요소를 어플리케이션을 실행하기 위하여 필요한 역할과 태스크로 교환한다. 입력 모델의 최상위 Group부터 처리를 시작하고, [Fig. 5]에 나타난 것처럼 관련된 요소를 탐색, 발견한

요소를 대응하는 노드레벨의 Role을 생성한다. 생성된 역할과 태스크를 WSN을 구성하는 노드에 할당하는 것으로 노드레벨 모델을 생성한다. 역할과 태스크를 노드에 할당하기 위해서는 실제의 WSN을 구성하는 노드에 관한 정보가 필요하기 때문에, 교환 시에는 네트워크 정보를 부여한다. 네트워크 정보로는 대상 WSN에 존재하는 노드의 ID와 장소에 관한 정보를 나타낸다. 이 네트워크 정보와 추출한 역할, 태스크를 배치조건에 근거하여 노드에 할당하는 것으로 노드 레벨 모델의 자동교환을 실현한다.

```

1: function GENERATEROLESET(groupModel)
2:   roleSet ← makeRole(groupModel, roleSet)
3:   return roleSet
4: end function
5: function MAKEROLE(inputModel, roleSet)
6:   if inputModel.root.equals(MemberNodes) then
7:     roleSet.add(MemberRole with
8:       SamplingTask & SendingTask)
9:   else
10:    group ← inputModel.root
11:    if group.leader.equals(Sink) then
12:      roleSet.add(BaseStationRole with
13:        ReceivingTask for each
14:        member in group.members)
15:    else if group.leader.equals(LeaderNode) then
16:      leaderRole ← vertex with LeaderRole
17:      if group.leader.contains(Operator) then
18:        leaderRole.append(OperationalTask)
19:      end if
20:      roleSet.add(leaderRole with
21:        ReceivingTask & SendingTask
22:        for each member in group.members)
23:    end if
24:    for each member in group.members do
25:      roleSet ← makeRole(member)
26:    end for
27:  end if
28:  return roleSet
29: end function

```

[Fig. 5] Algorithm2 of Group Level to Node Level

교환에서는 입력인 그룹레벨 모델을 최상위의 Group에서 makeRole을 호출하는 탐색을 시작한다(1-2행). 우선 makeRole의 입력모델을 조사하고, MemberNodes인지 아닌지에 의해 처리를 분기한다(6, 8행). 최초의 탐색 대상은 Group이기 때문에, 입력 모델의 최상위 요소인 Group을 얻는다(9행). 해당 그룹의 리더는 Sink이기 때문에 조건분기에 의해 대응하는 BaseStationRole을 수신하는 태스크(ReceiveTask)와 함께 작성한다(10-11행).

이 후, 해당 그룹의 각 멤버에 대해서 makeRole을 호출하는 (19-21행) 것으로, 하위 그룹에 대해서도 Role의 작성을 수행한다. 예에서는 중간층에 해당하는 Group이

하를 입력해서 makeRole을 호출한다.

이 경우 조건 분기의 결과, LeaderNode에 대응하는 LeaderRole이 생성된다.(12-18행). 다시 멤버 요소를 이 용해서 makeRole의 재귀호출을 반복 탐색하고, 교환을 진행한다. makeRole에 있어서, 입력이 MemberNodes인 경우에는 대응하는 MemberRole을 작성한다.(6-7행). 이것은 그룹 레벨 모델에 있어서 MemberNodes보다 하위에 접속된 요소가 존재하지 않기 때문이다. 재귀호출에 의한 자동교환의 결과 Role을 얻을 수 있다. Role의 생성 후는, 각 Role의 deploymentCondition에 기초해서, 조건을 만족하는 Node에 할당을 수행한다. BaseStationRole은 Node의 속성 ID의 값이 0인 노드에, LeaderRole은 어느 것이든 하나의 Node로, MemberRole은 어떤 Role도 할당되지 않는 모든 Node에 각각 할당되어 지고 있다.

#### 4. 결론

본 논문에서는 WSN 어플리케이션 개발시 데이터 품질개선을 지원하기 위한 메타데이터 설계를 제안하였다.

제안된 방법은 어플리케이션 개발 시 저비용의 프로토타입을 개발하고 빠른 기간 내에 품질을 개선하고 상세한 품질개선에 대한 요구 사항을 만족할 수 있다. 기존의 개발방법은 추상도를 일관되게 사용하기 위한 기술 능력과 기술의 용이성 간의 모순성으로 인하여 동시에 두 가지를 만족할 수 없다. 본 논문에서는 추상도가 다른 복수의 모델링 언어를 병용해 사용함으로써 데이터 품질 개선을 지원하는 메타 데이터 설계를 제안한다.

프로세스 구현시 정의된 추상도가 다른 세 개의 DSML와 기술된 모델간의 자동교환규칙에 관해서도 상세히 설명하였다. 제안된 메타 데이터 설계를 이용하면 일정하게 수행되는 WSN 어플리케이션에서 실행환경마다 품질을 개선하는데 지원 가능할 것으로 보인다.

향후 데이터 품질 개선을 위한 메타 데이터 설계를 기반으로 저비용으로 운용되는 WSN 프레임 워크 설계에 대해 연구할 예정이다.

#### REFERENCES

[1] I.F. Akyildiz, W. Su, Y. Sankarasubramaniam, and

E. Cayirci, "A survey on sensor networks," *IEEE Commun. Mag.*, vol.40, No.8, pp. 102-114, 2002.

[2] L.S. Bai, R.P. Dick, and P.A. Dinda, "Archetype-based design: Sensor network programming experts," *Proc. Intl. Conf. on Information Processing in Sensor Networks*, pp. 85-96, 2009.

[3] L. Mottola and G.P. Picco, "Programming wireless sensor networks: Fundamental concepts and state of the art," *ACM Comput. Surv.*, vol.43, pp.19:1-19:51, 2011.

[4] M. Ceriotti, L. Mottola, G.P. Picco, A.L. Murphy, S. Guna, M. Corra, M. Pozzi, D. Zonta, and P. Zanon, "Monitoring heritage buildings with wireless sensor networks: The torre aquila deployment," *Proc. Intl. Conf. on Information Processing in Sensor Networks*, pp 277-288, 2009.

[5] W. Seah and Y.K. Tan, *Sustainable wireless sensor networks*, chap 13, InTech, 2010.

[6] J. Zhao and R. Govindan, "Understanding packet delivery performance in dense wireless sensor networks," *Proc. Intl. Conf. on Embedded networked sensor systems*, pp. 1-13, 2003.

[7] G.P. Picco, "Software engineering and wireless sensor networks: Happy marriage or consensual divorce?," *Proc. FSE/SDP Workshop on Future of Software Engineering Research*, pp. 283-286, 2010.

[8] F. Losilla, C. Vicente-Chicote, B. Alvarez, A. Iborra and P. Sanchez, "Wireless sensor network application development: An architecture-centric mde approach," *Software Architecture*, ed. by F. Oquendo, vol.4758, pp. 179-194, 2007

[9] P. Boonma and J. Suzuki, "Moppet: A Model-Driven performance engineering framework for wireless sensor networks," *Comput. J.*, vol 53, No. 10, pp. 1674-1690, 2010.

[10] J.B. Lim, B. Jang, S. Yoon, M.L. Sichitiu, and A. G. Dean, "Raptex: Rapid prototype tool for embedded communication systems," *ACM Trans. Sen. Netw.*, vol. 7, pp. 7:1-7:40, 2010.

[11] S.R. Madden, M.J. Franklin, J.M. Hellerstein, and

W. Hong, "TinyDB: an acquisitional query processing system for sensor networks," *ACM Trans. Database Syst.*, vol.30, pp. 122-173, 2005.

[12] M. Welsh and G. Mainland, "Programming sensor networks using abstract regions," *Proc. Conf. on Symposium on Networked Systems Design and Implementation Vol. 1*, pp. 3-3, 2004.

[13] D.Gay, P. Levis, R. vonBehren, M. Welsh, E. Brewer, and D. Culler, "The nesc language: A holistic approach to networked embedded systems," *Proc. ACM SIGPLAN 2003 Conf. on Programming language design and implementation*, pp. 1-11, 2003.

**전혜경(Jeon, Hye Kyoung)**



- 1995년 2월 : 인하대학교 일문과 (문학사)
- 1999년 8월 : 인하대학교 정보공학과(공학석사)
- 2002년 9월 ~ 2005년 8월 : 인하대학교 컴퓨터정보학과 박사수료
- 2009년 4월 ~ 2013년 12월 : 이스트립 선임연구원

- 관심분야 : 상황인식, 센서 네트워크, 유비쿼터스, 사물인터넷
- E-Mail : jhk7010@nate.com

**박양재(Park, Yang Jae)**



- 1990년 8월 : 인하대학교 정보공학과 (공학석사)
- 2003년 8월 : 인하대학교 전자계산공학과 (공학박사)
- 2001년 2월 ~ 2002년 12월 : ㈜이메디피아 원격 의료 연구소 연구소장
- 1993년 2월 ~ 현재 : 가천대학교 IT 대학 컴퓨터공학과 교수

- 관심분야 : HCI, 영상처리, 모바일 네트워크, 비전시스템, 조명시스템
- E-Mail : parkyj@gachon.ac.kr