

---

# 적응형 패킷 전송에 기반한 DDS 미들웨어 간의 효율적인 데이터 교환 방법

안성우\*

Efficient Method for Exchanging Data between DDS Middlewares  
based on Adaptive Packet Transmission

Sung-woo Ahn\*

## 요 약

본 논문은 데이터 중심 통신 미들웨어인 DDS에서 데이터 교환 시 미리 정의된 프로토콜에 의해 고정된 패킷 전송 방식을 사용함으로써 발생하는 문제점을 분석한다. 고정된 패킷 전송 방식은 노드 및 네트워크에 가해지는 부하가 동적으로 변경되는 환경에서 필요한 노드 리소스 사용량 증가에 적절히 대처하지 못하는 문제가 있다. 만약 노드와 네트워크에 가해지는 부하가 한계점을 넘으면 패킷 전송 성능이 급격히 떨어지게 되며 DDS 미들웨어의 가장 중요한 사용 목적인 실시간성을 보장하지 못하게 된다. 이를 해결하기 위하여 본 논문에서는 패킷 전송 방식을 실시간으로 조정하기 위한 적응형 패킷 전송 기법을 제안한다.

## ABSTRACT

In this paper, we analyze the problems that the DDS middleware, which is a standard data-centric communication interface, uses the fixed packet transmission method by the pre-defined protocol for exchanging data packets. The packet transmission method selected in a fixed manner cannot handle appropriately the increasing of resource overhead in an environment where the load of the DDS network changes dynamically. If the load on the node and network exceeds the threshold, the performance of the packet transmission may be degraded rapidly. This results in a failure of ensuring the real-time characteristic of DDS middleware. To solve this problem, we propose the scheme of the adaptive packet transmission for adjusting the transmission method in real-time based on the overhead on the DDS network.

## 키워드

DDS, 데이터 분배 서비스, 적응형 패킷 전송, 유니캐스트, 멀티캐스트

## Key word

DDS, Data Distribution Service, Adaptive Packet Transmission, Unicast, Multicast

---

\* 정회원 : 경남정보대학교 (교신저자, ahnsw@kit.ac.kr)

접수일자 : 2012. 06. 01

심사완료일자 : 2012. 06. 01

## I. 서 론

최근 통신 기술의 비약적인 발전으로 유무선 네트워크를 통해 대용량의 데이터를 빠른 속도로 전달하는 것이 가능해졌다. 이에 따라 네트워크에 연결된 시스템은 이전보다 많은 양의 정보를 처리해야 할 뿐만 아니라 교환된 정보를 실시간으로 사용하고자 하는 사용자의 요구에 응답할 수 있어야 한다. 이러한 요구를 처리하기 위해 분산 환경에서 실시간으로 데이터 교환을 지원하는 다양한 통신 미들웨어 기술들이 소개되고 있으며 최근에는 데이터 중심 통신 미들웨어인 데이터 분배 서비스(Data Distribution Service: DDS)에 대한 관심이 증대되고 있다 [1,2].

DDS 미들웨어는 OMG(Object Management Group) [3]에서 표준으로 정의한 통신 미들웨어로서 발간/구독(publish/subscribe) 방식을 기반으로 미들웨어 간에 데이터를 교환한다. DDS는 불특정 다수의 발간자(publisher)와 구독자(subscriber)가 네트워크에 참여하며 발간자가 발간한 정보를 구독자가 구독함으로써 데이터의 교환이 이루어진다. 교환되는 메시지는 별도의 서버를 거치지 않고 발간자와 구독자 사이에서 직접적으로 전달된다. 따라서 발간/구독 모델은 많은 수의 단말에서 동시다발적으로 데이터 통신이 발생하고 실시간성을 요구하는 데이터 교환이 많은 시스템에 적합하다 [2].

DDS 미들웨어를 통해서 교환되는 데이터는 토픽(Topic)으로 정의되며, 네트워크에 참여하는 개체인 참여자(Participant)와 데이터 발간/구독의 주체가 되는 단말개체(Endpoint)는 자신이 관심을 가지는 토픽을 디스커버리 과정을 거쳐 네트워크를 통해서 등록한다. 이후 발간개체(Writer)는 관심 토픽이 동일한 구독개체(Reader)에게 자신의 토픽을 전달한다 [4].

단말개체 간에 토픽 정보를 교환하기 위해서는 네트워크를 통하여 패킷을 송수신하며 이때 두 가지 방식을 사용할 수 있다 [5]. 첫 번째 방법은 참여자가 포함되어 있는 노드의 IP 주소를 참조하여 해당 노드로 직접 패킷을 전송하는 유니캐스트 전송방법이다. 두 번째 방법은 개별 구독개체에게 직접 패킷을 전달하지 않고 참여자가 포함되어 있는 네트워크의 모든 노드에 패킷을 전파하는 멀티캐스트 전송방법이다. 각 방법은 전송해야 하는 패킷의 크기, 전송 주기, 수신해야 하는 노드의 개수 등에 따라 노드 및 네트워크에 가해지는 부하가 달라진

다. 따라서 노드의 리소스 사용 비용을 최소화하고 네트워크 활용을 최대화할 수 있도록 적절하게 패킷 전송방법을 선택할 수 있어야 한다.

그러나 표준 DDS 미들웨어는 부하에 따라 자동으로 패킷 전송 방식을 결정할 수 있는 방법을 제공하지 않는다. 이에 따라 DDS 미들웨어를 사용하는 응용은 토픽을 전달하기 위해 발간개체의 패킷 전송 방식을 직접 지정해야 하기 때문에 노드와 네트워크 리소스를 효율적으로 활용하기 어려운 문제가 있다. 이 논문에서는 이러한 문제를 해결하기 위하여 미들웨어에 추가되는 부하에 따라 자동으로 패킷 전송 방식을 결정하기 위한 방법을 제시하고자 한다.

이 논문의 구성은 다음과 같다. 2장에서는 DDS 미들웨어의 구성요소 및 통신 방식에 대해서 살펴본다. 3장에서는 고정된 패킷 전송 방식을 사용함으로써 발생하는 문제점에 대해서 분석한다. 4장에서는 3장에서 분석한 문제를 해결하기 위한 적응형 DDS 데이터 패킷 전송 방법을 제안한다. 마지막으로 5장에서 결론을 기술한다.

## II. DDS 미들웨어의 통신 방식

DDS 미들웨어는 분산 환경에서 응용 간에 데이터를 교환하기 위한 발간/구독 방식의 통신 미들웨어 표준이며 DDS 네트워크를 구성하는 요소는 그림 1과 같다. DDS 네트워크는 데이터의 공유 영역인 도메인(Domain)으로 구분되며 도메인 내에는 하나 이상의 참여자를 포함하고 있다. 각 참여자는 하나 이상의 발간자 또는 구독자를 가질 수 있으며 발간자는 자신의 정보를 전달하기 위해서 하나 이상의 발간개체를, 구독자는 상대편 발간자의 정보를 수신하기 위해 하나 이상의 구독개체를 포함할 수 있다 [4,6].

응용에서는 특정 데이터의 주제를 나타내는 토픽을 생성하고 이를 발간개체와 구독개체에 관심 토픽으로 등록함으로써 이들 간에 정보를 교환하게 된다. 관심 토픽이 다른 발간자와 구독자는 토픽 정보를 교환하지 않기 때문에 이들 간에는 간섭이 발생하지 않는다. 관심 토픽에 대응되는 정보는 발간개체를 통하여 데이터 분배 서비스 네트워크를 통해서 “DDS 데이터” 패킷으로 전송되며 구독개체는 수신된 “DDS 데이터” 패킷에 포함된 토픽 정보를 추출하여 응용에게 전달한다.

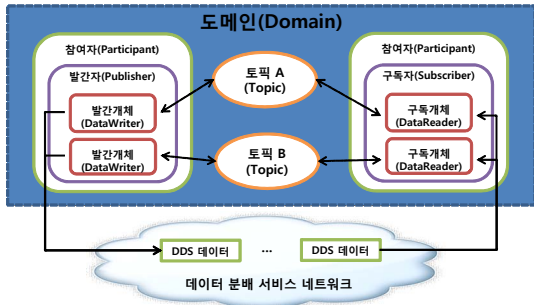


그림 1. DDS 네트워크 구성요소  
Fig. 1 Components of DDS network

발간개체와 구독개체를 통한 DDS 미들웨어 간의 데이터 교환은 두 단계로 나누어진다 [5].

첫 번째 단계에서는 디스커버리 과정을 통하여 단말개체 정보 교환 및 단말개체 간 연결 설정을 한다. DDS 미들웨어의 단말개체가 정보를 교환하기 위해서는 서로 관심을 가지는 토픽과 토픽 전달방법을 알고 있어야 한다. 이를 위해서 표준 명세에서는 DDS 미들웨어에서 제공해야 할 공통 디스커버리 프로토콜을 정의하고 있다. DDS 미들웨어 디스커버리는 먼저 SPDP (Simple Participant Discovery Protocol)를 통하여 네트워크에 존재하는 참여자의 정보를 교환한다. 그 후 SEDP (Simple Endpoint Discovery Protocol) 과정을 통하여 참여자에 속한 단말개체의 정보를 교환한다. SPDP와 SEDP 이후 발간개체와 구독개체는 관심 토픽이 동일한 단말개체의 연결정보를 저장한다.

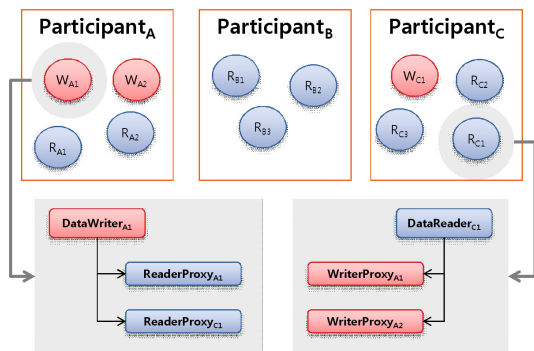


그림 2. 토픽을 이용한 단말개체 간 연결 설정  
Fig. 2 Connection Setting between Endpoints using a Topic

예를 들어, 그림 2와 같이 구성된 네트워크 환경에서 “토픽<sub>A</sub>”에 관심을 가지는 단말개체가  $W_{A1}, W_{A2}, R_{A1}, R_{C1}$  이라고 가정하자. 이 경우 발간개체인  $W_{A1}$ 은 자신이 발간하는 토픽에 관심을 가지는 구독개체에 대한 연결정보를 ReaderProxy라는 구조로 저장해놓는다. 따라서  $W_{A1}$ 은  $ReaderProxy_{A1}, ReaderProxy_{C1}$ 을 생성해서 저장한다. 마찬가지로 방법으로 구독개체는 관심 토픽이 동일한 발간개체의 연결정보를 WriterProxy로 저장한다. 그러므로  $R_{C1}$ 은  $WriterProxy_{A1}, WriterProxy_{A2}$ 를 생성하여 저장해놓는다.

데이터 교환의 두 번째 단계에서는 연결된 구독개체에 발간개체가 “DDS 데이터” 패킷을 전달한다. 발간개체는 자신이 발간하는 토픽에 관심을 가지는 구독개체의 연결정보를 ReaderProxy를 통하여 알고 있으므로 “DDS 데이터” 패킷을 생성해서 네트워크를 통해서 전달한다. 연결정보는 해당 구독개체를 포함하는 노드의 IP이며 이는 유니캐스트 IP와 멀티캐스트 IP를 포함하고 있다. 발간개체는 두 가지의 IP 중에서 하나를 선택하여 “DDS 데이터” 패킷을 전달한다.

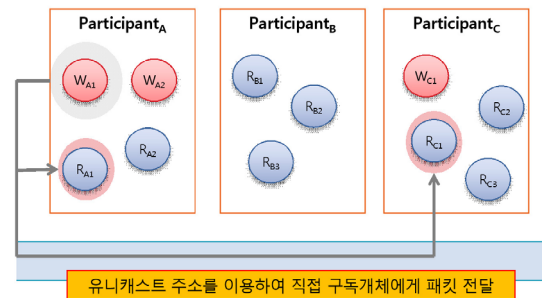


그림 3. 유니캐스트 IP를 이용한 패킷 전송  
Fig. 3 Packet Transmission using Unicast IP Address

유니캐스트 IP를 이용한 패킷 전송은 그림 3과 같이 구독개체가 포함된 노드에 패킷을 직접 전달하는 방식이다. 만약 그림 2에서와 동일하게 “토픽<sub>A</sub>”에 관심을 가지는 단말개체가 설정되었다면 발간개체  $W_{A1}$ 이 토픽을 발간할 때에는 연결된 구독개체가 있는 Participant<sub>A</sub>, Participant<sub>C</sub>를 포함한 노드로만 패킷이 전달될 것이다.

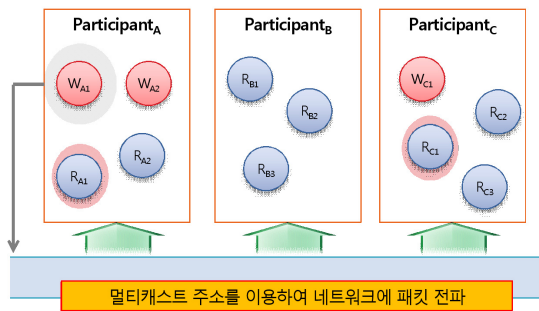


그림 4. 멀티캐스트 IP를 이용한 패킷 전송  
Fig. 4 Packet Transmission using Multicast IP Address

멀티캐스트 IP를 이용한 패킷 전송은 그림 4와 같이 관심 토픽을 가지는 구독개체를 포함하는 참여자의 위치에 상관없이 네트워크에 멀티캐스트 IP 주소로 “DDS 데이터” 패킷을 전파하며 네트워크에 참여하고 있는 모든 노드로 패킷이 전송된다. 따라서 “토픽A”에 관심을 가지는 구독개체를 포함하는 Participant<sub>A</sub>, Participant<sub>C</sub> 뿐만 아니라 네트워크에 참여하고 있는 Participant<sub>B</sub>에게도 “토픽A”에 대한 “DDS 데이터” 패킷이 전송된다.

### III. 문제 정의

2장에서 살펴보았듯이 발간개체는 네트워크를 통하여 서로 연결된 구독개체에게 “DDS 데이터” 패킷을 보낼 때 ReaderProxy에 저장된 연결정보를 참조하여 주소의 형태에 따라 멀티캐스트 또는 유니캐스트 IP로 전송한다. 패킷을 전송하는 방식에 따라 발간개체와 구독개체가 속한 노드의 부하와 네트워크 대역폭 사용률이 달라진다. 즉, 발간개체가 구독개체에게 “DDS 데이터” 패킷을 전달할 때 어떠한 패킷 전송 방식을 사용하느냐에 따라서 DDS 미들웨어 시스템의 성능에 차이가 발생할 수 있다.

예를 들어, 그림 2의 DDS 미들웨어 구성에서 각 참여자는 모두 별도로 구분된 노드에서 동작한다고 가정(참여자<sub>A</sub>-노드<sub>A</sub>, 참여자<sub>B</sub>-노드<sub>B</sub>, 참여자<sub>C</sub>-노드<sub>C</sub>)하자. 만약 발간개체 W<sub>A1</sub>이 관심 토픽 정보를 “DDS 데이터” 패킷으로 네트워크에 전달할 때 유니캐스트 전송방식을 사용하면 패킷은 노드<sub>A</sub>에서 노드<sub>C</sub>로만 전달된다. 만약 멀티캐스트 전송방식을 사용하면 패킷은 네트워크 스위

치를 거쳐 네트워크에 참여하고 있는 모든 노드로 전달된다. 그러나 참여자B에는 발간개체 W<sub>A1</sub>에 발간하는 토픽에 관심을 가지는 구독개체가 없으므로 노드B로 전달된 패킷은 참여자B의 구독개체들에게 전달되지 않고 폐기된다. 만약 W<sub>A1</sub>의 토픽에 관심을 가지지 않는 구독개체를 포함하는 노드가 많아질수록 전달되지 않고 폐기되는 패킷은 노드 수에 비례해서 증가할 것이다.

유니캐스트 전송방식의 장점은 네트워크에 있는 단말개체의 수에 상관없이 정보에 관심을 가지는 단말개체가 포함된 노드로만 패킷이 발송되기 때문에 네트워크에 불필요한 패킷의 전송을 방지할 수 있다. 그러나 특정 발간개체가 전송하는 정보를 수신하고자 하는 구독개체의 수가 증가하게 되면 발간개체에서 전송해야 하는 패킷의 수도 비례해서 증가하게 된다. 이는 발간개체가 위치한 노드의 CPU 부하를 증가시켜 성능 저하를 가져온다.

멀티캐스트 전송방식의 장점은 발간개체가 전송하는 정보를 수신하고자 하는 구독개체의 수가 증가하더라도 발간개체가 위치한 노드는 한 번의 패킷만 네트워크를 통해 전파하면 되므로 패킷 전송 횟수 증가에 따른 노드 성능 저하를 방지할 수 있다. 그러나 해당 정보를 필요로 하지 않는 구독개체를 포함한 노드에도 패킷을 전달하기 때문에 네트워크의 트래픽이 증가하며 이로 인한 네트워크 효율 저하가 발생된다.

만약 DDS 미들웨어에서 단말개체 간에 정보 교환 시 CPU 사용 비용을 최소화하고 네트워크 활용을 최대화할 수 있도록 패킷 전송 방식을 실시간으로 조정하는 기능을 제공한다면 DDS를 사용하는 시스템의 효율을 높일 수 있을 것이다. 그러나 기존의 DDS 미들웨어는 패킷 전송 방식을 입력되는 부하에 따라 자동으로 조정하는 기능을 제공하지 않으며 응용이 패킷 전송 방식을 직접 지정하도록 하고 있다.

### IV. 적응형 DDS 데이터 패킷 전송

이 장에서는 DDS 미들웨어에서 발간개체와 구독개체 간에 패킷을 교환하는 방식을 입력되는 부하 정보를 기반으로 자동으로 결정하기 위한 적응형 DDS 데이터 패킷 전송 방법을 제안하고 구현을 위한 모듈 설계를 한다.

패킷 전송 방식의 선택은 노드로 입력되는 부하정보를 기반으로 한다. 부하정보는 전송하고자 하는 패킷의 크기, 전송 속도, 전송하고자 하는 구독개체의 개수 등을 포함한다. 일반적으로 전송하고자 하는 정보를 작은 크기의 패킷으로 나누어 빠른 속도로 전송되는 경우 CPU의 부하는 증가하고 네트워크 대역폭 사용은 감소한다. 반대로 패킷의 크기를 크게 하여 전송 속도를 감소시키면 CPU 부하는 감소하고 네트워크 대역폭 사용은 증가한다. 이와 같이 패킷 크기와 전송 속도는 서로 간에 trade-off 관계에 있다. 그러나 발간개체가 패킷을 전달할 구독개체의 개수가 증가하면 CPU 부하와 네트워크 대역폭 사용은 비례해서 동시에 증가한다.

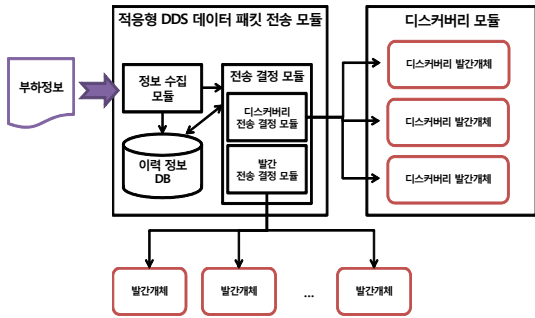


그림 5. 적응형 DDS 데이터 패킷 전송 모듈 구성  
Fig. 5 Module Configuration of Adaptive DDS Data Packet Transmission

입력되는 실시간 부하정보를 이용하여 CPU 부하와 네트워크 대역폭 사용량을 측정함으로써 시스템 전체 부하의 임계점(threshold)을 넘기지 않도록 패킷의 전송 방식을 실시간으로 결정할 수 있다. 그림 5는 전송 방식 결정을 위한 적응형 DDS 데이터 패킷 전송 모듈의 구성을 나타내고 있다. 디스커버리 과정 및 발간개체가 패킷을 전송할 때마다 패킷 크기, 전송 속도, 발간개체에 연결된 구독개체의 개수가 부하정보로써 “정보 수집 모듈”에 수집되며 이 정보는 “이력 정보 DB”에 저장됨과 동시에 “전송 결정 모듈”로 전달된다. “전송 결정 모듈”은 서브모듈인 “디스커버리 전송 결정 모듈”과 “발간 전송 결정 모듈”로 구분되며 각각 해당 발간개체에서 수집된 부하정보를 이용하여 패킷 전송 방식에 따른 CPU 부하 및 네트워크 대역폭 사용량에 대응되는 시스템 부하를 계산한다.

“전송 결정 모듈”에서는 발간개체 별로 시스템 부하가 최소가 되는 패킷 전송 방식을 선택한다. 선택된 전송 방식은 각 발간개체에게 전달되는데 “디스커버리 전송 결정 모듈”을 통하여 디스커버리를 담당하는 “디스커버리 발간개체”로 전달되며, “발간 전송 결정 모듈”을 통하여 일반 토픽 정보를 전달하는 “발간개체”에게 전달된다. 마지막으로 계산 이력을 “이력 정보 DB”에 저장함으로써 다음 계산 시에 활용할 수 있도록 한다. 그림 6은 적응형 DDS 데이터 패킷 전송 모듈의 동작 순서도로 보여주고 있다.

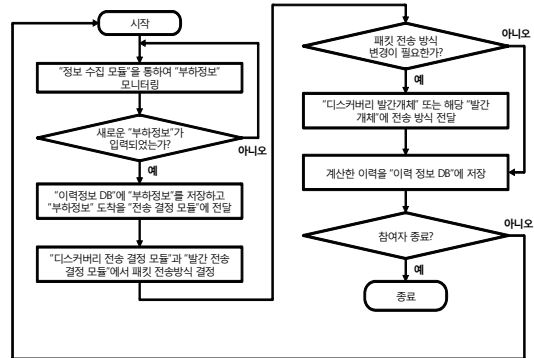


그림 6. 적응형 DDS 데이터 패킷 전송 모듈 동작 순서도  
Fig. 6 Flowchart of Module of Adaptive DDS Data Packet Transmission

패킷 전송 방식 결정을 위해 시스템 부하를 계산할 때에는 실시간으로 입력되는 부하정보와 함께 “이력 정보 DB”에 저장되어 있던 이전 부하정보를 기반으로 한 계산 이력을 추출하여 함께 사용한다. 이를 통하여 현재 입력된 정보만을 이용할 때 발생할 수 있는 오차를 이력 정보를 활용하여 보정할 수 있다. 그림 7은 이력 필터를 이용하여 이전에 저장된 이력 정보를 보정하는 단계를 보여주고 있다. 시스템 부하에 대한 이력은 시간이 지남에 따라 계속해서 누적되며 “이력 정보 필터”를 통해서 현재 입력된 부하정보와 함께 보정된다. 예를 들어,  $t_{n-1}$  시점에 계산된 이력 정보는 “이력 정보 DB”에 저장되어 있다가  $t_n$  시점에 시스템 부하를 계산할 때  $t_n$ 에 입력된 부하정보와  $t_{n-1}$ 에 “이력 정보 DB”에 저장된 이력이 “이력 정보 필터”에 함께 입력이 되어 보정된다. 필터를 이용한 보정 방법은 칼만 필터(Kalman filter) [7]와 같은 시간

의 흐름에 따른 변화를 예측하기 위한 재귀 계산법을 활용할 수 있다.

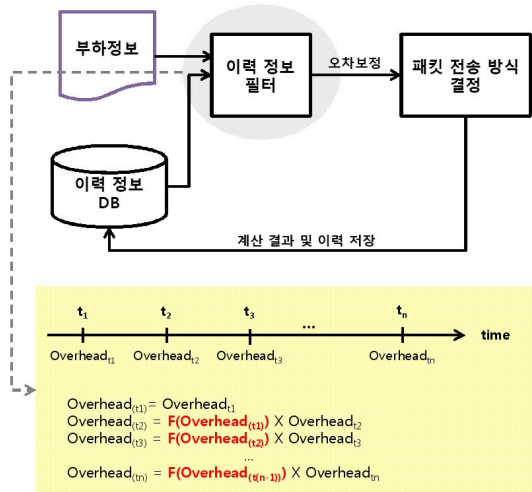


그림 7. 이력 필터를 이용한 결과 보정  
Fig. 7 Result Correction using History Filter Method

### V. 결 론

기존의 DDS 미들웨어는 패킷을 전송할 때 시스템의 성능 변화에 상관없이 미리 선택된 방식으로 전달함으로써 선택된 패킷 전송 방식에 따라 노드 및 네트워크의 부하가 증가하여 DDS 미들웨어 전체 시스템 성능 저하를 가져오는 문제가 있다. 본 논문에서는 노드의 부하를 최소화하고 네트워크 활용을 최대화할 수 있도록 패킷 전송 방식을 실시간으로 조정하기 위한 방법과 이를 처리하기 위한 구조를 제시하였다. 논문에서 제시한 구조는 DDS 미들웨어에 포함되어 구현된다면 응용에서 패킷 전송 방식을 직접 지정하여 발생하는 불편함을 제거할 수 있을 것이다. 또한 입력되는 정보를 기반으로 한 적응형 패킷 전송 방식을 제공함으로써 고정된 패킷 전송 방식을 사용할 때 발생할 수 있는 시스템 과부하로 인한 비효율성을 제거할 수 있을 것이다.

### 참고문헌

- [1] 전형국, 이수형, 김원태, 김경태, 박승민, “DDS 미들웨어 표준 기술 동향,” 정보통신산업진흥원 주간기술동향 1456호, pp.1-13, 2010.
- [2] D.C. Schmidt, A. Corsaro, and H. Hag, “Addressing the Challenges of Tactical Information Management in Net-Centric Systems with DDS,” The Journal of Defense Software Engineering, pp.24-29, 2008.
- [3] OMG DDS Portal, <http://www.omgwiki.org/dds>
- [4] G. Pardo-Castellote, “OMG Data-Distribution Service Architectural Overview,” International Conference on Distributed Computing Systems Workshops, pp.200-206, 2003.
- [5] OMG, “The Real-time Publish-Subscribe Wire Protocol DDS Interoperability Wire Protocol Specification,” Version 2.1, OMG, 2008.
- [6] OMG, “Data Distribution Service for Real-time Systems,” Version 1.2, OMG, 2007.
- [7] G. Welch and G. Bishop, “An Introduction to the Kalman Filter,” TR 95-041, University of North Carolina, 2004.

### 저자소개



안성우(Sung-Woo Ahn)

1999년 부산대학교  
컴퓨터공학과(공학사)  
2001년 부산대학교  
컴퓨터공학과(공학석사)

2009년 부산대학교 컴퓨터공학과(공학박사)  
2010년 ~ 2012년 2월 삼성탈레스 해양시스템연구소  
전문연구원  
2012년 3월 ~ 현재 경남정보대학교 모바일콘텐츠계열  
교수

※관심분야: 통신 미들웨어, LBS, 모바일 GIS