임베디드 시스템을 위한 경량화된 DDS 미들웨어의 설계*

류상현, 김인혁, 엄영익
성균관대학교 정보통신공학부
e-mail:{shryu, kkojiband, yieom}@ece.skku.ac.kr

A Design of Light-weight DDS for Embedded System

Sanghyun Ryu, Inhyuk Kim, Young Ik Eom
School of Information and Communication Engineering
Sungkyunkwan University.

요 약

최근 분산된 노드들 간의 자료를 송수신하는 임베디드 어플리케이션들이 증가함에 따라 이를 지원하기 위한 분산 미들웨어도 함께 발전하고 있다. 분산미들웨어는 원격 객체 관리를 위한 클라이언트-서버 모델, 큐를 이용한 메시지 전송을 목적으로 하는 메시지 패싱 모델, 분산 컴퓨팅 환경에서 자료의 전송을 목적으로 하는 출판-구독 모델이라는 3가지 모델로 분류 된다. 본 논문에서는 3가지 분산 미들웨어 모델들에 대하여 살펴보고 출판-구독 모델의 대표적 분산미들웨어인 Data Distribution Service를 소개한다. 또한 출판-구독 모델이 단말 간 P2P를 지원함에 따라 생기는 문제점인 과다한 트래픽을 해결하기 위해 자료들을 그룹화 시켜 전송하거나 특정 토픽에 관련된 자료들을 미리 예약된 채널을 통해 전송하는 기법들을 제안하고 일반 DDS와 비교를 통해 그 효과를 예측해본다.

1. 서론

최근 임베디드 소프트웨어 어플리케이션들은 네트워크 상에서 여러 노드간의 자료를 송수신을 목표로 한다. 이와 함께 컴퓨터 환경의 급격한 발전에 따라 이들을 지원하기 위한 분산 미들웨어 기술 역시 함께 발전하고 있다.

분산 미들웨어의 대표적인 3가지 모델로는 클라이언트-서버 모델, 메시지 패싱 모델, 출판-구독 모델이 존재한다. 클라이언트-서버 모델은 원격에 존재하는 서버들에 존재하는 자료 혹은 기능들을 이용하기 위한 다대일 모델이고, 메시지 패싱 모델은 큐를 이용한 다대일 메시지 전송을 목적으로 한다. 출판-구독 모델의 경우 출판자와 구독자를 구분하여 자료 전송과 수신에 적합한 모델이다.

위의 미들웨어 모델 중 출판-구독 모델은 각 구독자와 출판자가 직접 자료를 주고받는 구조로 되어 있어 자료 전송에 적합한 특성을 가진다. 본 논문에서는 위에서 언급한 3가지 분산 미들웨어 모델들에 대하여 살펴보고, 이중 대표적인 출판-구독 모델인 Data Distribution Service소개한다.

DDS와 같은 출판-구독 분산 미들웨어의 경우, 자료를 전송하는 출판단말 측과 자료를 수신하는 구독단말 측이 P2P로 전송하기 때문에 과도한 트래픽이 문제 될 수 있다. 본 논문에서는 이를 해결하기 위해 자료들을 그룹화하여 전송하는 기법과 신속한 자료의 전송을 요하는 자료들에 대하여 특정 채널을 예약하여 전송하는 기법을 제안하고 DDS의 전송방법과 비교하여 그 효과를 예측해본다.

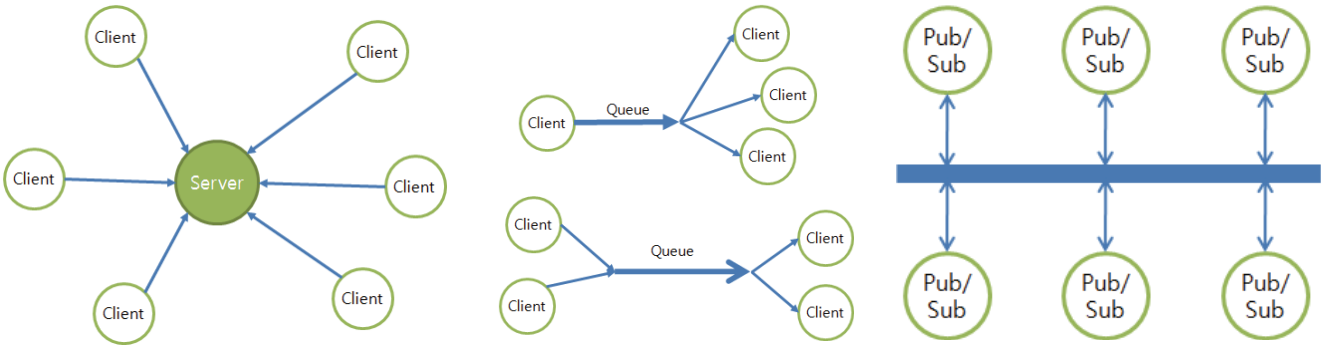
본 논문은 다음과 같은 구조로 구성된다. 2장 관련연구에서는 분산 미들웨어의 3가지 종류에 대하여 좀 더 자세히 알아보고 그 중 출판-구독 모델의 대표적 모델인 DDS의 구조에 대하여 살펴본다. 3장에서는 DDS의 트래픽 량을 감소시키고 보다 경량화하기 위한 제안 기법들을 제안하고 정성적으로 비교해본다. 4장에서는 본 논문에서 살펴본 분산 미들웨어와 제안 기법에 대하여 결론을 짓고 차후 연구할 점들에 대하여 생각해본다.

2. 관련연구

분산 미들웨어 모델은 다음과 같이 3가지로 분류된다.[1]

첫째로, 원격 프로시져 호출을 위한 클라이언트-서버 모델이 있다. 클라이언트-서버 모델은 그림1의 (a)와 같이 자료를 저장하고 있는 서버측과 자료를 요청하거나 기능을 호출 하는 클라이언트로 구성되어 있는 모델로서, 대표적인 미들웨어로서는 CORBA가 존재한다. 대부분의 클라이언트-서버 모델의 클라이언트는 서버 측의 자료 또는 기능을 이용하기 위해 API를 이용한다. 즉, 원격 클라이언

* 본 연구는 지식경제부 및 정보통신산업진흥원의 대학 IT연구센터 지원사업의 연구결과로 수행되었음 (NIPA-2010-(C1090-1021-0008))



(a) 클라이언트-서버 모델 (b) 메시지 패싱 모델 (c) 출판-구독 모델

그림 1 대표적 분산 모델

트에서 서버에 존재하는 원격 객체에 접근하여 호출함으로서 마치 클라이언트에 존재하는 객체를 이용하는 것처럼 사용하는 것이다. 클라이언트-서버 모델은 기본적으로 여러 클라이언트와 하나의 서버로 이루어지는 다대일 구조로 되어있다. 이런 하나의 서버를 중심으로 하는 중앙집중형 모델은 중앙 파일 서버나 데이터베이스 같은 시스템에 적합하지만 만약 여러 클라이언트에서 정보를 생성해내 서버로 전송한다면 서버 측에 병목현상이 발생하는 문제점을 안고 있다. 또한 클라이언트-서버 구조는 기본적으로 TCP 통신을 하기 때문에 신뢰성 있는 통신이 가능하지만 통신 속도가 다른 모델에 비해 느리다는 단점을 가진다.

두 번째로, 메시지의 전송을 목표로 하는 메시지 패싱 모델이 존재한다. 메시지 패싱 모델은 메시지 큐를 이용한 전송을 기본 구조로 한다. 그림 1의 (b)처럼 메시지 패싱 모델 내의 객체들은 큐를 생성하거나 메시지를 송신하거나 수신한다. 메시지 패싱 모델은 대다일의 클라이언트-서버 구조를 확장한 모델이라고도 볼 수 있다. 이런 메시지 패싱 모델의 대표적인 구현체로서는 IBM사의 MQSeries가 존재한다. 메시지 패싱 모델은 클라이언트 간 P2P 통신을 가능하게 하지만 클라이언트 간의 특정 채널이나 ID로 이루어지는 경로를 탐색해야 하므로 성능 저하를 유발 할 수 있다. 즉, 메시지 패싱 모델의 클라이언트들은 메시지 전송 시마다 메시지가 어디서 출발하고 어디로 도착할 지를 탐색해야 한다.

마지막으로 자료의 신속한 전달을 위해 제안된 출판-구독 모델이 있다. 그림 1의 (c)와 같이 출판-구독 모델은 출판자와 구독자로 구성되어 출판자는 자료를 생성하여 단순히 출판하고, 구독자들은 자신이 원하는 출판자로부터 자료를 구독하는 구조를 가진다. 논리적으로 출판-구독 모델에서 메시지들은 각 출판자들과 구독자들 노드사이를 직접적으로 통신한다. 이런 직접통신을 이용한 메시지 전송은 전송 시간을 중요시하는 시스템에서 유용하게 사용할 수 있다. 중간에 거쳐야 하는 노드들을 감소시킴으로서 여러 출판자들이 여러 구독자들에게 전송할 때도 더욱 효과적으로 전송 가능하게 해준다. 또한 자료 전송에 대한 적절한 서비스 품질을 설정할 수 있게 해주어 올바른 자료가 올바른 목적지에 도달할 수 있게 한다. 출판자들과 구독자들은 다대다의 형태를 띠고 있으며 모델 내의 개체들이 서로 약한 관계를 지니기 때문에 새로운 출판자나 구독자의 등장하거나 기존에 참여하던 출판자나 구독자가 모델에서 제외 될 때 복잡한 절차를 필요로 하지 않는다.

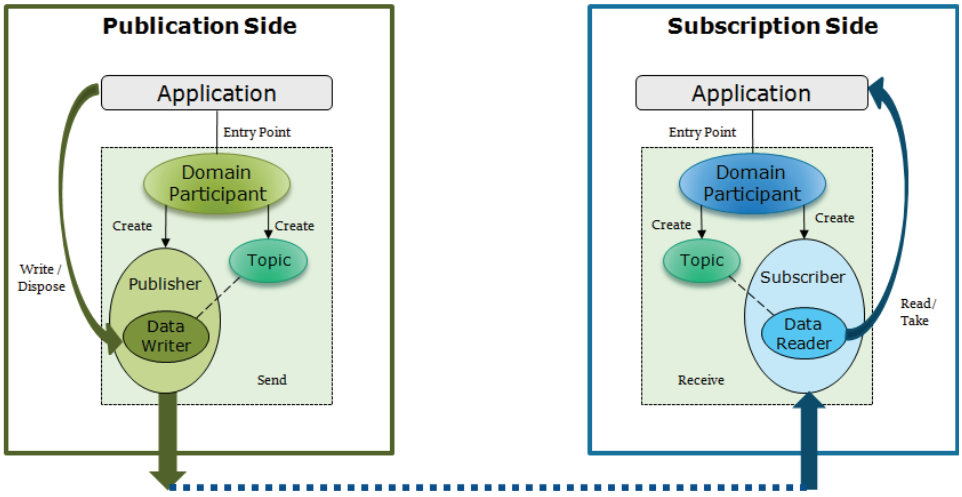
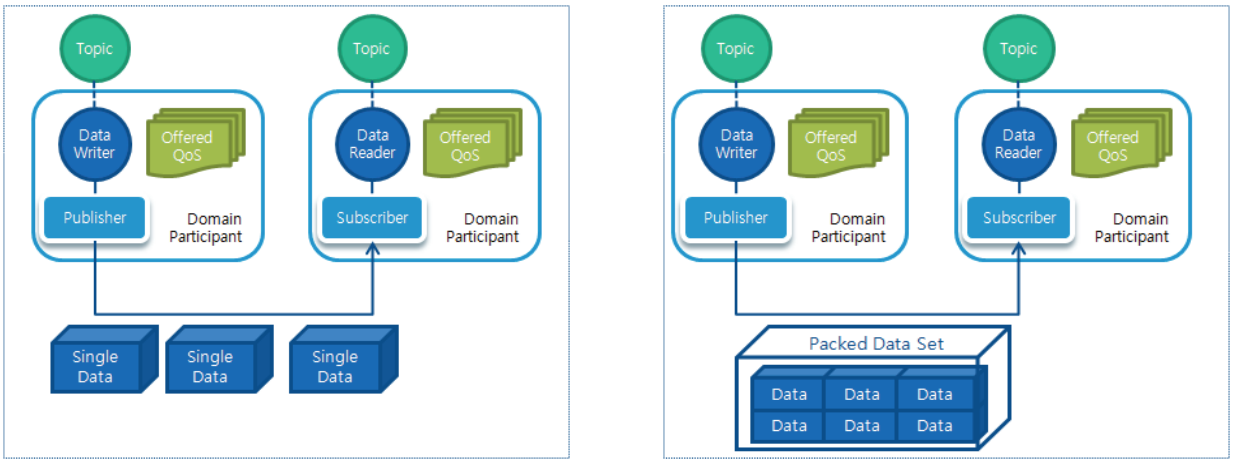


그림 2 Data Distribution Service의 구조



(a) 기존 DDS의 자료 전송 (b) 트래픽 감소를 위한 그룹 화된 자료 전송

그림 3 트래픽 감소를 위한 DDS의 자료 전송 방법

대표적인 구현체로는 Data Distribution Service가 존재한다.

Data Distribution Service[2][3]는 CORBA와 UML의 표준을 제정한 Object Management Group(OMG)[4]에서 제안한 자료 중심형 출판-구독 모델이다. DDS는 앞에서 설명한 출판-구독 모델처럼 자료를전송하는 출판자들과 자료를 구독하는 구독자들을 이어준다. DDS는 그림 2에서 보이는 DDS의 구조는 다음과 같이 구성된다. Global Data Space라 불리우는 하나의 도메인 영역과 그 안에 참가하고 있는 여러 호스트들이 각각 하나의 Participant로서 존재한다. 각각의 Participant 내에는 하나 혹은 여러 개의 Publisher와 Subscriber가 존재\하고 각 Publisher 내에는 실제 사용자가 자료를 출판하기 위한 Data Writer가 존재한다. 이와 유사하게 각 Subscriber 내에는 출판된 자료를 구독하기 위한 Data Reader가 존재한다. 각각의 Data Writer와 Data Reader는 출판 혹은 구독하려는 객체의 정보를 Topic이라는 이름으로 정의하고 있다. Topic은 OMG에서 제정한 Interface Definition Language(IDL)로 정의 되어 있으며 이를 통해 특별한 정의 없이 자료를 주고받게 된다.

또 다른 DDS의 큰 특징중 하나는 다양하고 폭넓은 서비스 품질 지원이다. DDS는 서비스 품질 지원을 위하여 리소스 관리, 자료 스케쥴, 자료의 사용 가능성, 자료의 전송 관리라는 큰 4가지 항목에 대한 서비스 품질을 지원한다. 각각의 항목은 해당 항목에 대한 세부 항목으로 나뉘어 DDS에서 자료 전송 시 출판은 원하는 출판자가 원하는 자료를 올바른 때 전송할 수 있게 해주고 구독을 원하는 구독자가 올바른 자료를 올바른 상황에 가질 수 있게 돕는다.

3. 제안 기법

3장에서는 DDS의 트래픽 감소를 위하여 전송 자료들을 그룹화 하여 보내는 기법과 중요한 특정 토픽에 대한 자료 전송 시 예약된 채널을 사용하는 기법을 제안한다.

전송 자료의 그룹화는 그림 3과 같이 자료 전송 시 일정 기간 동안의 자료들을 그룹화 함으로써 기본적인 전송에 필요한 메시지 크기를 줄이고 준비 작업들을 생략하여 불필요한 트래픽들을 감소시키는 기법이다. 이를 수행하기 위하여 출판자는 해당 자료의 그룹화 가능성을 결정해야 한다. 자료를 그룹화 하여 전송할 경우 일정 기간 동안의 자료를 묶어 전송하게 되므로 빠르게 처리해야 하는 자료들은 그룹화가 불가능 하다. 출판자 측에서 그룹화를 선택하면 Data Writer는 설정된 값에 따라 자료들을 그룹화 한다. 너무 많은 자료들을 그룹화 할 경우, 자료가 가지는 시간적 의미가 사라질 수도 있으므로 자료의 중요도에 따라 그 값을 조절한다. 각 자료를 구분하기 위해 각각의 자료는 목적지를 위한 헤더정보 대신 타임스탬프 정보를 가진다. 자료의 그룹화가 완료되면 출판자들은 구독자들에게 전송한다. 이때 해당 자료의 그룹화 크기만큼의 헤더파일들이 생략되므로 트래픽 량을 감소시키는 효과가 발생한다. 구독자들은 수신된 그룹 자료를 각각의 자료들로 분리하여 필요에 따라서 사용한다. 구독자들 역시 각각의 자료를 받을 때 보다 자료를 받는데 걸리는 작업들이 줄어들어 더 빠르고 손쉬운 자료의 처리가 가능해진다.

특정 토픽을 위해 예약된 채널을 사용하는 기법은 그림 4와 같이 정확한 목적지와 시간에 보내져야 하는 중요한 토픽에 관련된 자료들을 위하여 사용한다. 특정 채널을 예약해 둠으로서 해당 토픽에 관한 자료들을 올바른 시간에 올바른 구독자들에게 전달하게 된다. 이를 위해 채널 예약을 원하는 토픽의 경우, 토픽 생성 시에 특정 채널을 예약해야한다. 만일 채널이 부족하여 예약이 불가능할 경우 여분의 채널이 사용 가능할 때 까지 기다린다. 이렇게 특정 채널을 사용하는 토픽이 생성 되면, 해당 토픽의 자료를 출판하려는 출판자의 Data Writer나 구독하려는 구독자의

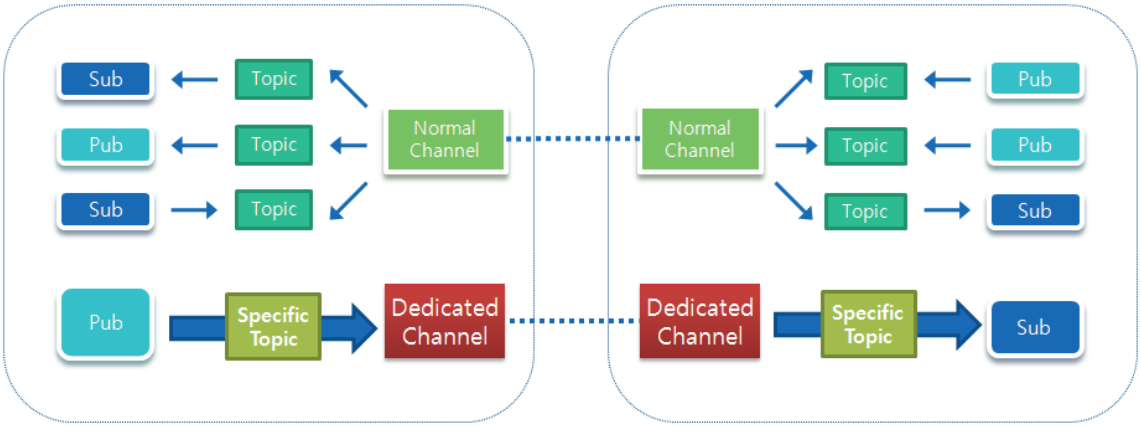


그림 4 특정 토픽을 위한 특정 채널 사용

Data Reader 모두 채널에 접근 가능함을 확인해야 한다. 이렇게 특정 채널을 예약해 사용할 경우 일반적인 DDS의 자료 전송과 비교하여 다음과 같은 장점을 지닌다. 첫째로, 채널을 예약해 사용할 경우 목적지를 탐색하기 위한 발견 모듈을 사용할 이유가 없어지므로 전송 트래픽을 보다 경량화 시킬 수 있다. 둘째로 특정 토픽에 대한 내용이므로 구독자가 어떤 토픽에 대한 내용인지 특별히 분리해낼 필요가 없다.

위의 제안 기법들에 대하여 전송 속도와 전송 메시지의 크기를 비교하면 표 1과 같다. 표1에서 볼 수 있듯이 n개의 자료를 그룹화 하여 전송 시, 기존 DDS가 n에 대한 각각의 헤더정보와 자료 정보를 담아야 하는 반면, 그룹화 시에는 헤더정보가 하나로 줄어듦을 알 수 있다. 또한 특정 채널을 이용 시, 기존 DDS의 전송시간과 비교하여 특정 채널 사용 시 채널 검색시간이 감소됨을 알 수 있다.

표 1 기존 DDS 전송방식과 제안 기법의 비교

n개 자료의 전송 시간	기존 DDS	$(T_{Header} + T_{Send} + T_{FindChannel} + T_{ParsingHeader}) * n$
	그룹화된 자료전송	$T_{Header} + T_{Send}' + T_{FindChannel} + T_{Group} + T_{ParsingHeader}$
	특정 채널 이용	$(T_{Header} + T_{Send} + T_{ParsingHeader}) * n$
전송 자료 크기	기존 DDS	$(S_{Header} + S_{Data}) * n$
	그룹화된 자료전송	$S_{Header} + (S_{Data} + S_{TimeStamp}) * n$

4. 결론

본 논문에서는 3가지 대표적인 분산 미들웨어 모델에 대하여 알아보고 그중 분산 자료처리에 적합한 출판-구독 모델에 대하여 보다 자세히 알아보았다. 특히 그중 OMG에서 제안한 출판-구독 모델인 DDS에 대하여 자세히 살펴보았다. 또한 트래픽 감소를 위하여 단일 자료들을 하나의 그룹으로 묶어 전송하는 그룹화 자료 전송 기법과 중요한 자료를 전달하는 특정 토픽을 위해 예약된 채널에 실어 보내는 기법을 제안하였다. 이와 같은 기법들을 통해 출판-구독 모델의 과다한 트래픽 양을 감소시킬 수 있음을 확인하였다.

향후 연구로서는 DDS의 도메인 내에서 사용자들이 신뢰성 있는 자료를 출판-구독 할 수 있도록 접근제어 기법을 이용하여 신뢰성을 향상시키는 기법에 대하여 연구할 것이다.

참고문헌

[1] Stan Schneider, Bert Farabaugh, "Is DDS for You?," 2005

[2] Wu, CH and Ip, WH and Chan, CY, "Real-time distributed vision-based network system for logistics applications," International Journal of Intelligent Systems Technologies and Applications, Vol.6, Num.3, PP309-322, 2009

[3] http://www.omgwiki.org/dds/

[4] http://www.omg.org/