
확장성과 효율성을 고려한 DDS 참여자 디스커버리 기법

권기정* · 유용덕** · 최훈*

A Scalable and Effective DDS Participant Discovery Mechanism

Ki Jung Kwon* · Yong Duck You** · Hoon Choi***

저자 최훈은 본 연구를 방위사업청과 국방과학연구소의 지원으로 수행하였으며,
이에 대해 깊이 감사를 드립니다.(계약번호 UD070030AD)

요 약

DDS(Data Distribution Service)는 데이터 중심의 통신 기술로서 DDS 디스커버리(discovery) 기법을 이용하여 각 데이터(토픽: Topic)에 대한 참여자의 위치정보 자동설정을 통해 참여자의 동적 플러그 앤 플레이를 지원하는 효율적인 통신 서비스를 제공한다.

본 논문에서는 기존 DDS 디스커버리 기법들의 성능 및 문제점들을 비교 및 분석하여 대규모 분산시스템에 적합한 계층적 구조 기반의 DDS 디스커버리 기법(SPDP-TBF)을 제안한다. 제안하는 SPDP-TBF 기법은 등록/검색을 담당하는 별도의 계층별 관리자들을 두어 연관된 참여자들만의 주기적인 디스커버리를 수행함으로써 참여자 정보를 관계된 참여자에게만 전송하여 메시지 전송의 효율성을 높이며, 나아가 계층별 관리자 노드를 통해 계층적 디스커버리를 수행함으로써 대규모 분산시스템에도 적용할 수 있는 높은 확장성을 제공할 수 있다.

ABSTRACT

The DDS (Data Distribution Service) is a data-centric communication technology that provides an efficient communication service that supports a dynamic plug & play through an automatic setting of participants' location information for each data (Topic) by using DDS discovery technique.

This paper proposes the hierarchical-structured DDS discovery technique (SPDP-TBF) suitable for the large-scale distributed systems by comparing and analyzing the existing DDS discovery techniques in terms of performance and problem areas. The proposed SPDP-TBF performs the periodic discovery of the involved participants only by having separate hierarchical managers which take charge of the registration and search (of participants) so that a participant sends its information to the related participants only, and it enhances the effectiveness of the message transfer. Moreover, the proposed SPDP-TBF provides the improved scalability by performing the hierarchical discovery through hierarchical manager nodes so that it can be applied to the large-scale distributed system.

키워드

발간/구독 통신, Data Distribution Service, 디스커버리 기법, SDP-Bloom, SPDP-TBF

* 충남대학교

** 관계아 솔루션

접수일자 2009. 04. 22

심사완료일자 2009. 05. 26

I. 서 론

최근 공항 관계 시스템, 금융 거래 처리 시스템 및 산업 자동화 시스템과 같은 대규모 어플리케이션(application) 분야에서는, 네트워크(network)를 통한 데이터의 수집/배포를 실시간으로 처리하고 시간의 경과에 따라 데이터 전송에 참여하는 네트워크 구성 개체들의 동적인 변경을 수용할 수 있는 데이터 중심의 소프트웨어 기술을 요구하고 있다[1, 2].

데이터 중심(data-centric)의 통신 기술 중 하나인 발간/구독(publish/subscribe) 통신은 데이터 발간자가 네트워크에 데이터를 발간하고 데이터 구독자는 실시간으로 갱신되는 데이터를 구독하는 단순한 통신 방식을 제공함으로써, 최근 실시간 내장형 분산 시스템의 새로운 패러다임(paradigm)으로서 가능성을 보여주고 있다[3, 4, 5].

OMG(Object Management Group)에서는 실시간 내장형 시스템을 위한 발간/구독 통신 방식의 미들웨어 API(Application Programming Interface)로서 DDS를 표준화하였다[6, 7, 8]. 실시간 데이터 배포에 뛰어난 성능을 보이고 있는 DDS는 응용프로그램이 물리적 주소와 같은 상대방의 정보가 없어도 익명(anonymous) 통신이 가능하도록 각 토픽에 대한 발간자와 구독자 정보를 지속적으로 추적하는 디스커버리 기능을 제공한다. 이러한 DDS 디스커버리 기능은 여러 응용프로그램들이 동적으로 도메인을 형성하고 참여/탈퇴가 빈번한 이동 통신 시스템 또는 유니쿼터스 시스템(ubiquitous system), 네트워크상의 장애나 각 참여자의 고장 상황에 능동적으로 대처하기 위한 고장 허용 시스템(fault-tolerant system) 및 규모 확장이 용이한 분산시스템 개발에서 매우 유용하다[9]. 그러나 기존 DDS 디스커버리 기법은 통신을 개시한 다음에는 각 참여자들이 다른 모든 참여자들에게 자신의 생존을 알리기 위해 주기적/지속적으로 메시지를 전송해야 하기 때문에 대규모 시스템에 적용하기에는 한계가 있다. 따라서 DDS를 대규모 분산시스템에 적용하기 위하여 기존의 DDS 디스커버리 기법을 보완하는 확장성 있고, 효율적인 디스커버리 기법이 필요하다[10, 11].

이를 위하여 본 논문에서는 기존의 DDS 표준 디스커버리 기법을 분석하고, 최근에 발표되고 있는 DDS 디스커버리 기법을 조사하여, 대규모 분산시스템에 적합한

계층적 구조 기반의 DDS 디스커버리 기법을 새롭게 제안하고자 한다. 제안하는 기법에서는 등록/검색을 담당하는 별도의 계층별 관리자들을 활용하여 상호 연관된 참여자들 간의 디스커버리를 수행함으로써 관계없는 참여자와 불필요한 메시지의 전송을 감소시킬 수 있고, 관리자 노드를 통해 계층적 디스커버리를 수행함으로써 높은 확장성을 제공할 수 있다.

본 논문의 구성은 다음과 같다. 2장에서는 관련 연구로서 OMG의 분산시스템 미들웨어인 DDS와 실시간으로 멤버십(membership) 검색을 수행하는 블룸 필터(Bloom Filter)에 대하여 기술하고, 3장에서는 기존의 표준 DDS 디스커버리 기법과 최근에 제안된 SDP(Simple Discovery Protocol)-Bloom 기법에 대해 설명한다. 4장에서는 본 논문에서 제안하는 디스커버리 기법인 SPDP-BF(Simple Participant Discovery Protocol-Topic Bloom Filter)의 구조 및 동작 과정에 대해 기술하고, 5장에서는 제안하는 SPDP-TBF 기법과 함께 기존 방식들에 대한 성능을 비교 및 분석한다. 마지막으로 6장에서는 결론 및 향후 연구 방향에 대해 제시한다.

II. 관련 연구

2.1 DDS 정의 및 구조

DDS는 분산 응용프로그램들 간에 데이터 중심 발간/구독 방식의 통신을 위한 표준 소프트웨어 API로서 OMG에서 표준화되었다. DDS는 크게 데이터 공유 영역인 도메인(Domain)과 도메인 참여자(Domain Participant)로 구성되며, 도메인 참여자는 각각의 데이터에 대하여 발간자(Publisher)와 구독자(Subscriber)로 나뉘게 된다. 발간자는 데이터를 도메인에 발간하기 위하여 데이터 송신자(DataWriter)를 이용하며, 구독자는 도메인으로부터 데이터를 수신하기 위하여 데이터 수신자(DataReader)를 이용한다. 데이터의 송신자와 수신자는 데이터에 의해 논리적 링크가 성립되기 때문에 송신자와 수신자는 이름이 같은 데이터만을 송수신 가능하며, 이름이 다른 데이터는 처리할 수 없다. DDS 데이터 전송 방식에는 데이터의 정확한 전송을 보장하는 신뢰형(reliable) 전송 방식과 수신자측의 수신 여부를 확인하지 않는 최선형(best-effort) 전송 방식이 있다(그림 1).

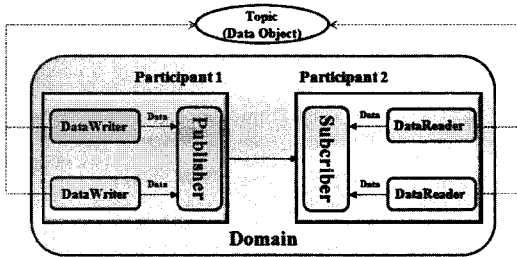


그림 1. DDS 구조
Fig. 1 DDS architecture

2.2 블룸 필터

블룸 필터는 m 개의 비트로 구성된 비트 벡터(bit vector)(BF)로서 n 개의 데이터를 가진 유한 집합 $A = \{x_1, x_2, \dots, x_n\}$ 에서 $O(1)$ 의 시간 복잡도로 멤버십 검색을 가능하게 한다[12,13,14].

[그림 2]에서 데이터 x_i 를 k 개의 서로 다른 독립 해쉬(hash) 함수 $h_j(x_i)$ 를 이용하여 매핑(mapping)시키고, 계산된 값 y_j 를 블룸 필터 BF 내에서의 비트 위치로 사용하여, 임의의 x 에 대하여 $BF[y_j]$ 가 모두 1인지를 검사함으로써, x 가 집합 A 에 포함되는가를 검색할 수 있다. 만약 하나 이상 위치에서 BF 비트 값이 0인 경우는 x 는 집합 A 에 속하지 않는 것이다.

$$h_j(x_i) = y_j, BF[y_j] = 1, 1 \leq j \leq k, 1 \leq i \leq n, 1 \leq y_j \leq m$$

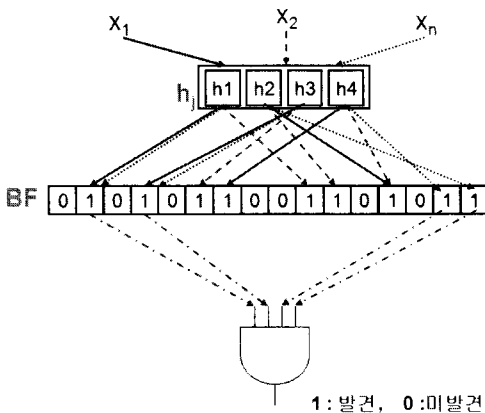


그림 2. 블룸 필터 사용 예
Fig. 2 Example of bloom filter

블룸 필터는 많은 양의 데이터를 줄여서 공간 효율적으로 빠르게 검색할 수 있는 장점을 제공한다.

III. 디스커버리 기법

3.1 기존 DDS 디스커버리 기법

DDS는 네이밍 서버(naming server)를 사용하지 않고 도메인 내 모든 참여자(participant)와 단말(endpoint)들 간의 존재와 속성에 대한 정보를 획득할 수 있도록 디스커버리 서비스를 규정하고 있다. DDS 디스커버리 기법은 참여자들 간에 참여자 정보(participant data)를 송수신하며, 참여자의 도메인 참여, 생존 통지 및 타 참여자 검색을 위한 SPDP와 참여자의 단말 정보(데이터 송신자, 데이터 수신자 및 토픽)를 서로 교환하여 세부 속성을 찾아내는 SEDP(Simple Endpoint Discovery Protocol)로 구성된다. 두 개 이상의 참여자가 SPDP를 통해 상호 발견하면, 이후 SEDP를 이용하여 참여자가 가지고 있는 단말 정보를 서로 교환한다. 이와 같은 DDS 디스커버리 기법을 이용하여 참여자는 시스템에 동적으로 플러그 앤 플레이(plug & play)를 수행할 수 있다[6, 7, 8].

가. SPDP

SPDP는 동일 도메인 내의 참여자를 발견하기 위해 참여자 간의 ‘DiscoveredParticipantData’를 주기적으로 전송하는 기법이다. 참여자들은 최선형 방식으로 SPDP를 위해 정의된 개체(built-in entity)인 ‘SPDPbuiltinParticipant Writer’와 ‘SPDPbuiltinParticipantReader’를 이용하여 참여자 정보(참여자 위치, BuiltInEndpoints 및 leaseDuration)를 송수신한다.

기존 참여자가 새로운 참여자로부터 참여자 정보를 수신하면 내부 데이터베이스에 저장한 후, 해당 참여자를 관리하는 ParticipantProxy 개체를 생성하고 자신의 참여자 정보를 주기적으로 송신한다. 각 참여자는 주기적으로 내부 데이터베이스에 저장된 각 참여자 정보를 조사하여 일정한 기간(lease duration)동안 갱신되지 않으면 해당 참여자가 탈퇴한 것으로 간주하고 관련 ParticipantProxy와 참여자 정보를 삭제한다.

나. SEDP

SEDP는 참여자의 단말에 대한 정보(데이터 송신자,

데이터 수신자 및 토픽)를 얻기 위한 기법으로서 참여자 간에 미리 정의된 데이터(Discovered PublicationData, Discovered SubscriptionData 및 Discovered TopicData)를 송신하고, 수신된 데이터들은 내부 데이터베이스에 저장한다. PublicationData, SubscriptionData, 및 TopicData는 송신자와 수신자의 위치 정보, QoS 및 데이터 등의 정보를 포함하고 있으며, 참여자의 송신자/수신자/토픽이 생성, 삭제 및 수정 시에 관련 참여자에게 신뢰형 통신 방식으로 전송된다. 이후 참여자는 수신한 단말 정보를 이용하여 상대 참여자와 메시지 통신을 담당하는 ReaderProxy, WriterProxy를 생성 또는 수정한다. 데이터 송신을 위한 ReaderProxy는 상대 참여자의 Stateful Reader와 메시지 통신을 담당하며, 데이터 수신을 위한 WriterProxy는 상대 참여자의 StatefulWriter와의 메시지 통신을 담당한다.

이와 같이, DDS 표준 디스커버리 개념은 도메인 내의 모든 참여자들이 서로를 발견한 후, SEDP를 수행하여 발견/구독하는 토픽이 일치하는 경우에 단말 개체(entity)간의 연결을 설정하고 통신을 수행하는 것이다. 통신을 개시한 다음에는 참여자들이 생존(liveness)을 알리기 위해 주기적으로 $(N \cdot (N - 1))$ 개의 메시지를 전송하는 SPDP를 지속적으로 수행하며, 이는 시스템 전체 성능에 직접적인 영향을 미친다. 또한, SEDP는 SPDP에서 발견된 모든 참여자의 단말 정보를 수신하여 저장한다. 이는 네트워크 트래픽과 노드의 메모리 요구량 관점에서 자신과 통신해야 하는 참여자의 단말 정보만을 관리하는 것보다 비효율적이다.

3.2 SDP-Bloom 디스커버리

SDP-Bloom 디스커버리 기법은 각 참여자의 단말(데이터 송신자 및 데이터 수신자) 집합을 대표하는 EBF(Endpoints Bloom Filter)를 이용하여 참여자 간의 단말 연관성을 찾아내어 활용하는 기법이다[10].

[그림 3]은 SDP-Bloom 방식이 적용된 참여자간의 디스커버리 기법을 보여주고 있다. SPDP는 새로운 참여자가 추가될 때 수행되며(1), 이후 SDP-Bloom이 주기적으로 수행되어 단말 연관성을 검색한다(2). SEDP는 연관된 참여자 단말이 변경(추가/삭제/수정)되는 경우에만 수행된다(3). 한 참여자가 SPDP를 통하여 새로운 참여자를 발견하면 주기적으로 EBF를 서로 교환하며, 참여

자는 다른 참여자로부터 수신한 EBF와 자신의 단말 연관성을 조사하여 관련이 있고, 변동사항이 있는 경우에만 해당 참여자와 SEDP를 수행한다.

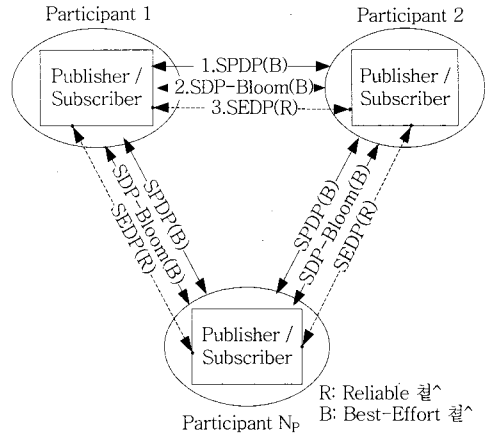


그림 3. SDP-Bloom 디스커버리 프로토콜
Fig. 3 SDP-Bloom discovery protocol

SDP-Bloom 방식은 모든 참여자가 SPDP를 최소 1회 수행하고, 이후에는 주기적으로 EBF를 전송하며, 관계가 있는 참여자들 사이에만 SEDP를 수행하기 때문에 표준 DDS 디스커버리 기법보다 메시지 트래픽을 줄일 수 있어 효율적이다([그림 4]).

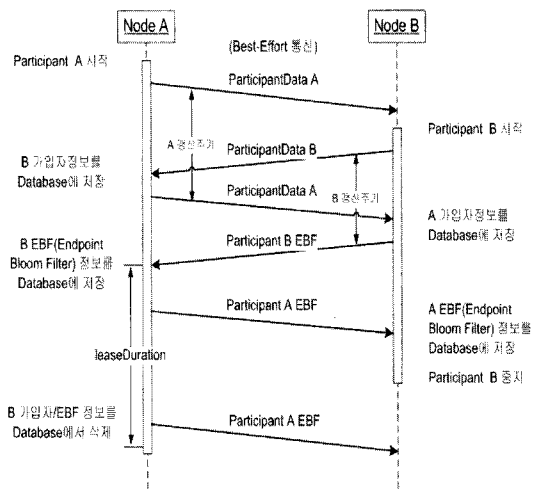


그림 4. SDP-Bloom 기법 실행 과정
Fig. 4 Procedure of SDP-Bloom

그러나 SPDP에서 주기적으로 발생하는 전체 메시지 전송량을 줄이는 문제는 여전히 존재한다.

IV. 제안하는 디스커버리 기법 (SPDP-TBF)

본 논문에서는 참여자간의 연관성을 검색하기 위해 토픽 블룸 필터(TBF: Topic Bloom Filter)를 활용하고, 관리자 노드를 두어 계층별로 참여자 디스커버리를 관리하도록 하는 SPDP-TBF 디스커버리 메커니즘을 제안한다(그림 5). 기존 DDS 표준 디스커버리가 플랫(flat) 방식이라면, 제안하는 SPDP-TBF는 계층적(hierarchical) 방식이라 할 수 있으며, SDP-Bloom이 데이터 수신자, 데이터 송신자 등의 단말에 대해 블룸 필터를 유지하는데 비해, 본 방식은 데이터, 즉 토픽에 적용한다.

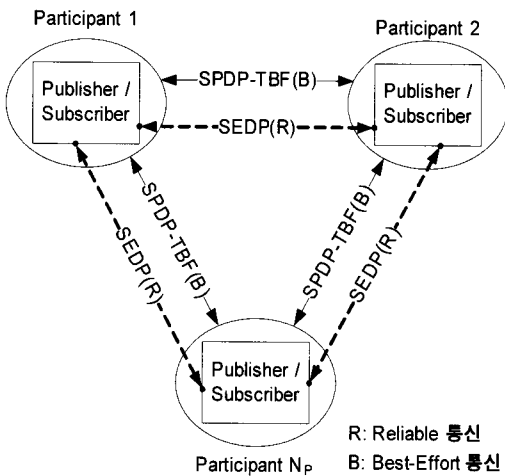


그림 5. SPDP-TBF 디스커버리 프로토콜
Fig. 5 SPDP-TBF discovery protocol

SPDP-TBF 디스커버리의 개념은 참여자 디스커버리 단계에서 TBF를 이용하여 관계가 있는 참여자만을 검색하여 디스커버리를 수행함으로써 관계없는 참여자와의 불필요한 메시지 전송을 감소시켜 효율을 높이고, 관리자 노드를 통해 계층적 디스커버리를 수행함으로써 대규모 시스템에도 적용이 용이하도록 확장성을 제공

하는 효과가 있다.

4.1 SPDP-TBF 디스커버리 구조

SPDP-TBF 방식은 참여자간의 연관성을 효율적으로 검색하기 위해 도메인 내의 노드들을 일정한 개수로 그룹화하여 클러스터로 나누었으며, 도메인 관리자(DM: Domain Manager), 클러스터 관리자(CM: Cluster Manager) 및 노드 관리자(NM: Node Manager)를 두어 계층적으로 토픽의 TBF 등록과 검색이 수행되도록 하였다(그림 6).

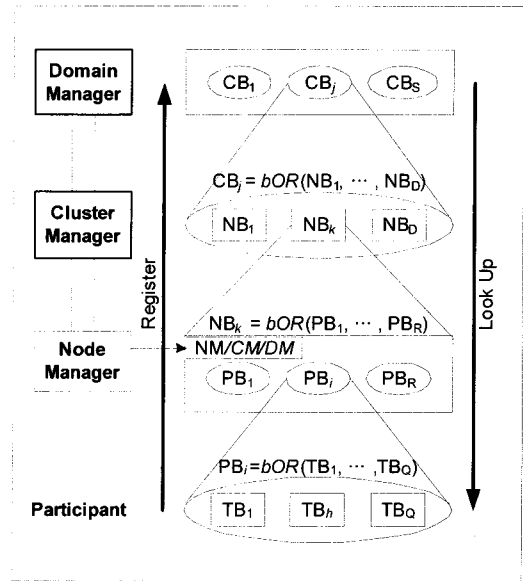


그림 6. SPDP-TBF 디스커버리 구조
Fig. 6 Architecture of SPDP-TBF discovery

- 노드 관리자: 단위 프로세서 노드마다 실행되는 디스커버리 데몬(daemon)으로서, 노드 내에서 실행되는 R개 참여자 PB(ParticipantTBF)를 생성/관리하고, 다른 관련 참여자에게 새로운 참여자/데이터 정보(TopicBF)를 전송한다. 참여자/데이터 정보를 수신한 기존 참여자는 참여자/데이터 정보를 새로운 참여자에게 전송함으로써 상호 연결이 설정된다. 노드 관리자는 NB(NodeTBF)를 계산하여 상위 클러스터 관리자에게 전달한다.
- 클러스터 관리자: 클러스터는 DDS의 파티션(partition)을 중심으로 구성되며, 특정 개수를 초과하

면 분할하여 새로운 클러스터를 생성한다. 만약 파티션으로 구분되지 못하는 경우에는 서브넷(subnet)을 중심으로 특정 개수(D)의 노드로 구성된다. 각 클러스터는 클러스터 관리자(CM)에 의해 관리된다. 클러스터 관리자는 구성 노드의 NB(NodeTBF)를 관리하면서 새로운 데이터가 등록되면 연관된 노드를 검색하여 해당 노드 관리자에게 참여자/데이터 정보를 전송하고, CB(ClusterTBF)를 계산하여 상위 도메인 관리자에게 전달한다. 클러스터 관리자는 클러스터를 구성하는 노드(NM) 중에서 다음의 조건을 만족하는 노드가 선출된다.

- ▷ 노드 내에서 실행되는 참여자 수가 적은 노드
- ▷ 참여자 수가 동일한 경우는 노드 UID(Unique ID)가 가장 작은 노드

- 도메인 관리자: 시스템 설계자에 의해 별도로 지정되거나, 클러스터 관리자의 선출 방식과 유사한 방법을 적용하여 클러스터 관리자 중에서 우선순위가 높은 것이 선출된다. 도메인 관리자는 도메인 내의 S개 클러스터 CB(ClusterTBF)을 관리하면서 새로운 데이터가 등록되면 연관 클러스터를 찾아 해당 클러스터 관리자에게 참여자/데이터 정보를 전달하는 기능을 담당한다.

4.2 SPDP-TBF 동작 방식

SPDP 과정에서 참여자간에 송수신하는 참여자 정보에는 위치 정보, Built-In-Endpoint 및 lease duration 속성이 있다.

제안하는 SPDP-TBF에서는 참여자 정보에 추가로 참여자가 발간/구독하고자 하는 모든 토픽을 이용하여 계산된 TBF가 추가된다. SPDP 초기 시점에 토픽이 정의되지 않은 경우 TBF는 0이며, 데이터 수신자, 데이터 송신자가 생성되면서 토픽을 정의하면 TBF를 구할 수 있다. TBF는 구독하고자 하는 SubscriptionTBF, 발간하고자 하는 PublicationTBF와 추가/삭제할 데이터의 TopicBF가 있다([그림 7]).

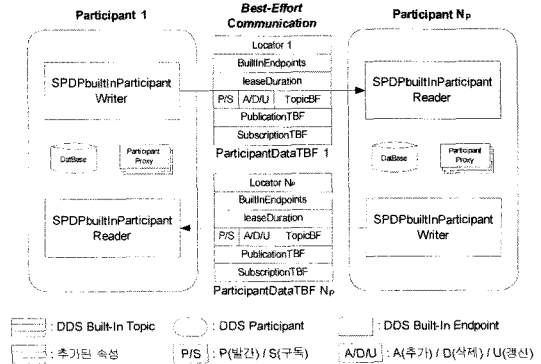


그림 7. SPDP-TBF 개체
Fig. 7 SPDP-TBF entities

참여자는 새로운 참여자로부터 참여자 정보를 수신한 후 TopicBF와 자신이 발간/구독하는 데이터와의 매칭 여부를 확인하여 연관이 있는 경우에만 참여자 정보를 내부 데이터베이스에 저장하고 해당 참여자를 관리하는 ParticipantProxy를 생성하며, 연관이 없는 경우에 참여자 정보는 처리하지 않는다. 생성된 ParticipantProxy는 새 참여자에게 자신의 참여자 정보를 송신하고 SEDP를 일대일로 수행한다. SEDP에 의해 상대 단말 정보를 획득하면, 관련 단말을 이용하여 토픽에 대한 발간/구독을 시작한다.

참여자는 새로운 참여자로부터 참여자 정보를 수신한 후 TopicBF와 자신이 발간/구독하는 데이터와의 매칭 여부를 확인하여 연관이 있는 경우에만 참여자 정보를 내부 데이터베이스에 저장하고 해당 참여자를 관리하는 ParticipantProxy를 생성하며, 연관이 없는 경우에 참여자 정보는 처리하지 않는다. 생성된 ParticipantProxy는 새 참여자에게 자신의 참여자 정보를 송신하고 SEDP를 일대일로 수행한다. SEDP에 의해 상대 단말 정보를 획득하면, 관련 단말을 이용하여 토픽에 대한 발간/구독을 시작한다.

가. 새로운 토픽(TPh) 등록 과정

[그림 6]에서 노드 관리자 NM_k 는 노드 k 내부에서 실행되는 모든 참여자(P_1, \dots, P_R)의 TBF(PB_1, \dots, PB_R)를 생성하고 관리한다.

참여자 P_i 가 발간/구독할 데이터 TP_n 에 대한 발간(P)/구독(S) 및 추가(A)/삭제(D)를 자신이 속한 노드 관

리자 NM_k 에게 등록을 요청하면, NM_k 는 해당 참여자 P_i 의 ParticipantTBF(PB_i), $TB_h(TP_h$ 의 TopicBF), SubscriptionTBF, PublicationTBF를 계산한 후 [PublicationTBF, SubscriptionTBF, TopicBF(TB_h)]를 리턴한다. PB_i 는 해당 참여자와 관련된 모든 데이터의 $TB(TBF)$ 가 Bit-OR된 $PB_i = bOR(TB_1, \dots, TB_Q)$ 값이며, $bOR(PublicationTBF, SubscriptionTBF)$ 값과 같다.

참여자 P_i 의 PB_i 가 변경되면 NM_k 는 NB_k 를 재계산한다. NB_k 는 노드 k 내부의 모든 참여자 PB 가 Bit-OR된 값($NB_k = bOR(PB_1, \dots, PB_n)$)으로서 NM_k 는 갱신된 NB_k 를 자신이 속한 클러스터 관리자 CM_j 에게 전달한다. 클러스터 관리자 CM_j 은 NM_k 로부터 NB_k 를 수신하여 기존 NB_k 와 다를 경우 CB_j 를 재계산한다.

CB_j 는 클러스터 j 에 속한 모든 노드 NB 가 Bit-OR된 값($CB_j = bOR(NB_1, \dots, NB_D)$)으로서 CM_j 는 갱신된 CB_j 를 자신이 속한 도메인 관리자 DM 에게 전달한다. 도메인 관리자 DM 은 도메인 내부의 모든 클러스터로부터 $bOR(CB_1, \dots, CB_s)$ 를 수신하여 관리한다.

나. 데이터(TPh) 관련 참여자 검색 과정

참여자 P_i 가 데이터 TP_h 에 대해 노드 관리자 NM_k 에게 등록을 요청하면 TP_h 와 연관된 관련 모든 참여자를 검색하여 데이터 TP_h 의 발간(P)/구독(S) 및 추가(A)/삭제(D)를 알리기 위한 검색 과정이 병행하여 수행된다.

검색은 영역에 따라 노드 내의 다른 관련 참여자를 찾는 노드 영역 검색, 클러스터 내의 다른 관련 노드를 찾는 클러스터 영역 검색 및 도메인 내의 다른 관련 클러스터를 찾는 도메인 영역 검색으로 구분된다.

검색 영역별로 수행하는 동작은 다음과 같다.

- 노드 영역 검색: 참여자 P_i 는 자신이 속한 노드 관리자 NM_k 에게 등록 요청 시 [ParticipantData(PD_i), 데이터(TP_h)]를 전달한다. NM_k 는 TP_h 등록 요청을 받으면 PB_i 와 TB_h 를 계산한 후, 다른 $PB_{p \neq i}$ 와 TB_h 매칭을 조사하여, 매칭되는 해당 참여자 $P_{p \neq i}$ 들에게 $[PD_i, TB_h]$ 를 송신한다. 참여자 $P_{p \neq i}$ 들은 자신의 $[PD_{p \neq i}, TB_h]$ 를 참여자 P_i 에게 전송함으로써 상호 검색이 이루어진다. 이후, NM_k 는 수정된

PB_i 를 이용하여 NB_k 를 갱신한 후 클러스터 관리자 CM_j 에게 $[PD_i, TB_h, NB_k]$ 를 송신한다.

- 클러스터 영역 검색: CM_j 는 클러스터 내부의 다른 $NM_{n \neq k}$ 와 TB_h 포함 여부를 조사하여, 매칭되는 노드 관리자 $NM_{n \neq k}$ 들에게 $[PD_i, TB_h]$ 를 전달한다. 데이터 TB_h 와 연관 있는 각 $NM_{n \neq k}$ 들은 노드 영역 검색을 수행하며, CM_j 는 수신된 NB_k 를 이용하여 CB_j 를 수정한 후 DM 에게 $[PD_i, TB_h, CB_j]$ 를 송신한다.
- 도메인 영역 검색: DM 은 도메인 내부의 다른 $CB_{c \neq j}$ 와 TB_h 매칭을 조사하여, 매칭되는 클러스터 관리자 $CM_{c \neq j}$ 들에게 $[PD_i, TB_h]$ 를 전달한다. 데이터 TB_h 와 연관 있는 각 $CM_{c \neq j}$ 들은 클러스터 영역 검색을 처리한다.

검색 과정은 [그림 8]과 같이 단계별로 진행되며, 도메인 내의 모든 참여자 디스커버리가 최대 4 단계 내에서 완료된다.

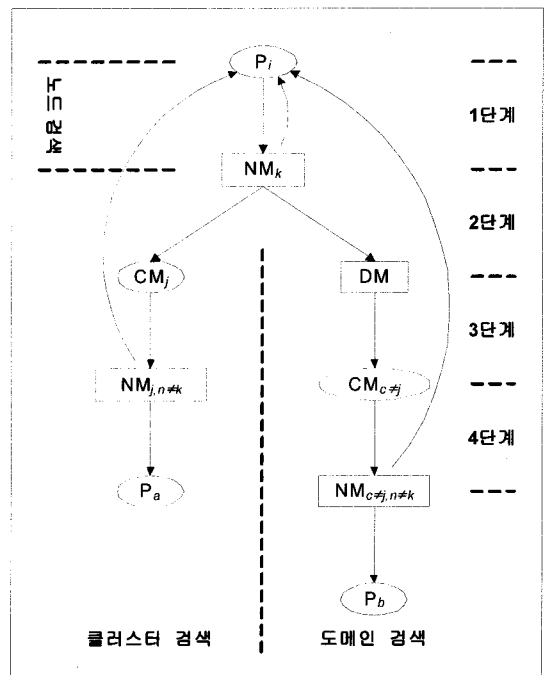


그림 8. SPDP-TBF 검색과정
Fig.8 Search procedure of SPDP-TBF

- 1 단계: 참여자 P_i 가 속한 노드 k 의 관리자 NM_k 에 의해 노드 영역 검색을 수행한다.
- 2 단계: 노드 k 가 속한 클러스터 j 의 관리자 CM_j 에 의해 클러스터 영역 검색을 수행한다. 또한 NM_k 가 DM 에게 직접 $[PD_i, TB_h]$ 를 전달함으로써 도메인 관리자 DM 에 의한 도메인 영역 검색이 병행하여 수행된다.
- 3 단계: CM_j 에 의해 검색된 노드 관리자 $NM_{n \neq k}$ 에 의해 노드 영역 검색을 수행한다. 또한 도메인 영역 검색에서 매칭되는 클러스터 관리자 $CM_{c \neq j}$ 들에게 $[PD_i, TB_h]$ 를 전달하여 클러스터 영역 검색을 수행한다.
- 4 단계: 클러스터 관리자 $CM_{c \neq j}$ 들에 의해 검색된 노드 관리자 NM 에 의해 노드 영역 검색을 수행한다.

V. 성능 분석

표준 DDS, SDP-Bloom 및 제안하는 SPDP-TBF 디스커버리 방식의 성능을 비교하기 위해 동일 규모의 시스템에서 일정 시간 (T) 동안의 네트워크 트래픽 오버헤드(overhead), 즉 메시지 발생량(전송횟수)을 분석하였다.

[11]은 메시지 발생량과 메모리 소모량 분석을 통해 표준 DDS 디스커버리 방식과 SDP-Bloom 디스커버리 방식의 성능을 비교하였는데, 메시지 발생량의 변화에 따라 표준 DDS 디스커버리 방식에서는 메모리 소모량이 비례함을 보였으며, 블룸 필터를 사용하는 SDP-Bloom 디스커버리 방식의 경우 두 파라미터 모두 상대적으로 매우 적음을 보였다.

DDS 디스커버리 과정에서의 시간 지연은 크게 연관된 참여자/Endpoint를 검색하는데 걸리는 지연과 검색된 참여자/단말과 SPDP/SEDP 메시지를 주고받는데 소요되는 시간 지연으로 구성된다. 연관된 참여자/단말을 검색하는데 소요되는 시간 지연은 블룸 필터를 사용함으로써 개선된다. 블룸 필터는 메모리 소모량 감소 및 $O(1)$ 의 시간 복잡도(time complexity)로 빠른 검색을 가능하게 한다는 사실은 이미 여러 연구를 통해 입증된 바

있다[10,11,12,13].

분산시스템에서 노드 간에 송수신되는 메시지 발생량은 시스템에서 메시지 처리를 위한 프로세서, 메모리 및 네트워크 등의 리소스 소모량에 직접적으로 영향을 주기 때문에, 본 논문에서는 다음과 같은 가정에서 3가지 디스커버리 방식의 메시지 발생량을 모델링하고 각각의 결과를 분석함으로써, 디스커버리 메시지 교환으로 인한 성능을 분석한다.

[가정] 시스템을 구성하는 각 노드의 컴퓨팅 리소스(프로세서, 메모리, 네트워크 등)는 동일하다.

[가정] 최대 메시지 전송(worst case)이 발생하도록 각 노드는 하나의 참여자로 구성되며, 참여자는 하나의 단말을 가진다. 따라서 도메인에서 각 노드, 참여자, Endpoint의 전체 개수는 서로 동일하다.

$$N_N = N_P = N_E$$

[가정] 노드 내부의 참여자 간에는 내부 프로세스(process)간 통신을 이용하고, 노드 간의 참여자 통신에서는 네트워크를 경유한 메시지 전송을 한다.

[가정] SPDP는 멀티캐스트(multicast)도 가능하지만 라우터 기능 제한 등의 이유로 멀티캐스트가 지원되지 않을 수 있으므로 본 논문에서는 유니캐스트(unicast) 통신만을 고려한다.

[가정] 최선형(best-effort: BE) 통신 방식의 경우 데이터를 전송하는데 필요한 메시지 수는 1개, 신뢰형(reliable: RL) 통신 방식의 경우는 ACK 응답을 고려하여 전송 메시지 수는 2개로 정의한다.

가. 기존 DDS 디스커버리 기법

기존 DDS 디스커버리 기법은 SPDP와 SEDP를 통하여 도메인 내의 참여자, 단말 및 토픽 정보를 상호 교환한다[8,9].

SPDP는 최선형 방식으로 통신을 수행하며 참여자 수가 N_P 인 시스템에서 일정 T 시간 동안 T_P ($T_P \ll \text{leaseDuration}$) 주기로 통신을 수행하면서 발생하는 메시지 전송 회수 N_{SPDP} 는 식(1)과 같다.

$$N_{SPDP} = N_p(N_p - 1)(T/T_p) \quad (1)$$

N_p : 도메인 전체 참여자 수
 T : 시스템 가동 시간
 T_p : SPDP 갱신 주기 (leaseDuration)

참여자는 시스템에 참여하면서 전체 기존 참여자에 대해 최소 1 회의 SEDP를 수행하며, 단말이 변경(추가/삭제/수정)될 때 마다 추가로 수행한다. SEDP는 신뢰형 방식의 비주기 통신을 이용한다. 성능 분석을 위해 단말 변경은 arrival rate λ_E 의 Poisson 분포를 따른다고 가정한다. 시간 T 동안 도메인에서 발생하는 평균 Endpoint 변경 횟수는 $\lambda_E T$ 이며, SEDP 메시지 전송 회수 N_{SEDP} 는 식(2)와 같다.

$$N_{SEDP} = 2(N_p(N_p - 1))\lambda_E T \quad (2)$$

따라서 기존 DDS 디스커버리 기법의 총 메시지 전송 회수 N_{SDDS} 는 식(3)과 같다.

$$\begin{aligned} N_{SDDS} &= N_{SPDP} + N_{SEDP} \\ &= N_p(N_p - 1)(1/T_p + 2\lambda_E) T \end{aligned} \quad (3)$$

나. SDP-Bloom 디스커버리 모델

새로운 참여자는 기존 각 참여자에 대해 SPDP를 수행한 후, SDP-Bloom이 주기 T_p 로 반복적으로 수행한다고 가정하면, 참여자의 디스커버리 메시지 전송 회수 N_{BPDP} 는 식(4)와 같다[12].

$$N_{BPDP} = N_p(N_p - 1)(T/T_p) \quad (4)$$

그러나 SEDP는 연관성이 있는 참여자들 사이에 수행되며, 참여자들 간의 Endpoint 상관율이 β_E 라 할 때, SEDP 메시지 전송 회수 N_{BEDP} 는 식(5)와 같다.

$$\begin{aligned} N_{BEDP} &= 2((N_p(N_p - 1))\beta_E)(\lambda_E T) \\ 0 &\leq \beta_E \leq 1 \end{aligned} \quad (5)$$

이에 따라 SDP-Bloom 디스커버리 기법의 총 메시지 전송 회수 $N_{SDPBLOOM}$ 은 식(6)과 같다.

$$\begin{aligned} N_{SDPBLOOM} &= N_{BPDP} + N_{BEDP} \\ &= N_p(N_p - 1)(1/T_p + 2\beta_E \lambda_E) T \end{aligned} \quad (6)$$

다. SPDP-TBF 디스커버리 기법

제안하는 SPDP-TBF 디스커버리 기법은 참여자의 TBF 등록과 검색을 담당하는 노드 관리자(NM), 클러스터 관리자(CM) 및 도메인 관리자(DM)를 통하여 새로운 참여자는 연관된 참여자만을 디스커버리한다. DDS는 동일한 데이터에 의해 참여자의 Endpoint들이 서로 논리적으로 연결되고, 참여자 간의 메시지 전송이 이루어진다. 따라서 참여자 연관성(β_p)이란 단말 연관성(β_E)을 의미하고, 단말 연관성은 데이터 연관성(β_T)을 의미한다.

SPDP-TBF 기법에서 관리자에서 관리자 또는 관리자에서 참여자에게 메시지 전송은 신뢰형 통신이며, S개의 클러스터로 구성된 도메인 내의 참여자 디스커버리를 위해서는 기본적으로 관리자 간에 신뢰형 메시지 전송 [$NM_{j,k} \Rightarrow CM_j$ (1회), $CM_j \Rightarrow DM$ (1회), $DM \Rightarrow CM_{c \neq j}$ (S-1회)]이 $2(2+(S-1))$ 회, 즉 $(2S+2)$ 회 발생한다. S개의 클러스터로 구성된 도메인에서 참여자 상관율 β_p 이고, 주기 T_p 로 SPDP-TBF를 수행한다고 가정하면, T 시간 동안 참여자 디스커버리 메시지 전송 회수 N_{TPDP} 는 식(7)과 같다.

$$\begin{aligned} N_{TPDP} &= N_p((\beta_p N_p) + (2S+2))(T/T_p) \\ &= (\beta_p N_p^2 + 2SN_p + 2N_p)(T/T_p) \end{aligned} \quad (7)$$

$$0 \leq \beta_p \leq 1, \beta_p = \beta_E = \beta_T$$

β_p : 참여자간 평균 상관율
 β_T : 참여자간 토폴 평균 상관율

참여자 단말 상관율이 β_E 일 때, 연관성이 있는 참여자들 사이에 신뢰형 방식의 SEDP 메시지 전송 회수 N_{TEDP} 는 식(8)과 같으며, 연관성 있는 참여자 사이에

SEDP를 수행하는 점에서 대규모 네트워크에서는 SPDP-TBF/SEDP와 SDP-Bloom/SEDP의 메시지 전송 회수는 큰 차이가 없다. 그러나 SPDP-Bloom/SEDP가 도메인 전체 네트워크 내에서 메시지 전송이 이루어지는데 비해 SPDP-TBF/SEDP는 주로 클러스터 내에서 전송이 발생하므로 메시지 전달 경로를 고려한 네트워크 부하가 상대적으로 감소한다.

$$N_{TEDP} = N_p((2\beta_p N_p) + (2S + 2))(\lambda_E T)$$

$$= (2\beta_p N_p^2 + 2SN_p + 2N_p)(\lambda_E T) \quad (8)$$

$$0 \leq \beta_E \leq 1, \beta_p = \beta_E = \beta_T$$

SPDP-TBF 디스커버리 기법의 총 메시지 전송 회수 $N_{SPDPTBF}$ 은 식(9)와 같다.

$$N_{SPDPTBF} = N_{TPDP} + N_{TEDP} \quad (9)$$

위에서 분석한 표준 DDS 디스커버리 기법, SDP-Bloom 기법 및 SPDP-TBF 기법에서 발생하는 메시지 전송 회수(트래픽)를 비교하면 [그림 9]와 같다.

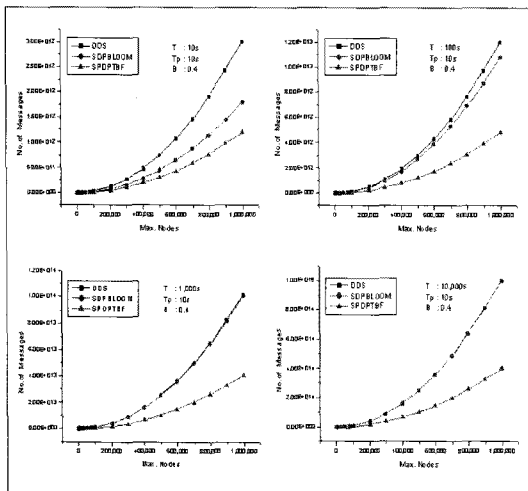


그림 9. 메시지 전송 회수 비교
Fig. 9 Comparison of message transmission frequency

[그림 9]에서 기존 DDS 디스커버리 기법은 도메인 내의 모든 참여자 간의 참여자 정보(SPDP)와 단말 정보(SEDP)를 주고받기 때문에 노드 수 증가에 따라 전송되는 메시지 전송 횟수가 기하급수적으로 증가함을 볼 수 있다.

SDP-Bloom 기법은 모든 참여자간 참여자 정보(SPDP)를 전송하지만, 참여자간의 EBF를 전달하고, 서로 관련 있는 참여자들만이 단말 정보(SEDP)를 주고받기 때문에, 도메인 전체 노드 수 대비 참여자 단말 평균 상관율(β_E)이 0.4(40%)이고, 참여자 갱신 횟수가 1 (T/T_p)회 일 때 전송 메시지 전송 회수가 DDS 표준 모델에 비해 40%가 감소된다. 그러나 동작 시간 T ($T_p \ll T$)가 커지면 DDS 표준 모델과 대등 수준까지 메시지 발생량이 증가한다.

제안하는 SPDP-TBF 기법의 메시지 발생은 참여자 상관율(β_p)이 0.4(40%)이고, 참여자 갱신 횟수가 1 (T/T_p)회 일 때, DDS 표준 모델에 비해 1/2.49 수준으로 감소되며, SDP-Bloom 모델 대비 1/1.5 수준까지 낮출 수 있다. 또한 동작 시간 T 가 증가하더라도 기존 DDS 디스커버리 기법과 비교할 때 상관율 수준의 메시지만 발생한다.

[그림 10]은 참여자 상관율(β_p) 변화에 따른 SPDP-TBF 대비 DDS 표준 모델과 SDP-Bloom 메시지 발생 배수를 보여주고 있다.

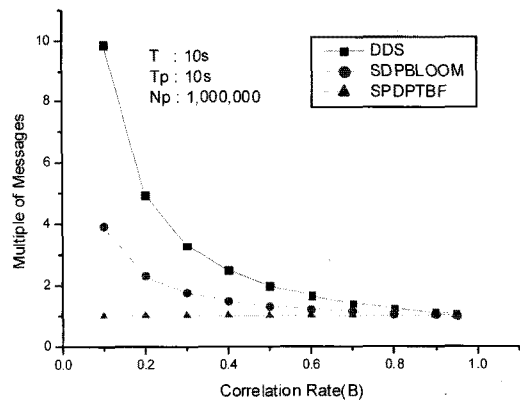


그림 10. 상관율에 따른 메시지 발생 비율
Fig. 10 Message generation rate according to correlation rate

상관율(β_p)이 0.1 인 경우 SPDP-TBF 모델의 메시지 발생 배수가 DDS 표준 모델 대비 1/9.8, SDP-Bloom 대비 1/3.9로서 최소 4배 이상의 효율을 보이나, 0.5에서는 DDS의 1/1.99, SDP-Bloom의 1/1.33로서 메시지 발생차 감소하며, 상관율이 1.0에 근접하면 각 기법 간에 메시지 발생 차이는 거의 없어진다. 따라서, 도메인 내의 참여자간 상관율이 최소 0.5 이하인 경우에는 제안한 SPDP-TBF 모델을 적용함으로써 최소 2배 이상의 메시지 발생 감소 효과를 기대할 수 있다.

VI. 결론 및 향후 연구

기존의 DDS 디스커버리 기법은 효율적인 통신 서비스를 제공하기 위하여 모든 참여자에 대한 정보를 유지해야 하기 때문에 시스템의 규모가 증가될 경우 정보 유지를 위한 저장 공간이 크게 요구되어 대규모 시스템에 적용하기에는 한계가 있다. 따라서, 본 논문에서는 기존 DDS 디스커버리 기법들의 성능 및 문제점들을 비교 및 분석하고, 대규모 분산시스템에 적합한 계층적 구조 기반의 새로운 DDS 디스커버리 기법을 제안하였다.

제안하는 SPDP-TBF 기법은 단말간 평균 상관율이 40%인 경우 기존 DDS 디스커버리 기법 및 SDP-Bloom 기법과 비교하여 노드 수 증가에 따른 전송되는 메시지 회수에 있어 각각 1/2.49과 1/1.5 수준까지 낮출 수 있어 높은 확장성을 제공하며, 계층적으로 TBF를 등록/검색을 담당하는 별도의 계층별 관리자들을 활용하여 연관된 참여자들의 디스커버리를 수행함으로써 불필요한 메시지 전송을 제거하는 높은 효율성을 제공한다. 그러나 제안하는 SPDP-TBF 기법은 관리자 노드에 부하가 집중되고, 참여자간의 주기적인 참여자 정보 전송은 대규모 시스템에서 상당한 네트워크 부하로 작용될 수 있다.

향후 연구로는 관리자 노드의 부하를 줄이면서 주기적 메시지 전송을 참여자간이 아닌 클래스를 대표하는 관리자 노드간만의 전송으로 대체함으로써 네트워크의 부하를 줄여 보다 높은 확장성을 제공하고자 한다.

감사의 글

저자 최훈은 본 연구를 방위사업청과 국방과학연구소의 지원으로 수행하였으며, 이에 대해 깊이 감사를 드립니다.(계약번호 UD070030AD)

참고문헌

- [1] D.S.Havaladar, G.Katti, N.DasGupta and A.DasGupta, "Subthreshold Current Model of FinFETs Based on Analytical Solution of 3-D Poisson's Equation," IEEE Trans. Electron Devices, vol. 53, no.4, pp. 1-10, 2006.
- [2] Rajive Joshi, "Architecting High-performance Data-centric Systems Using Data Distribution Service and Switched Fabrics," http://rfdesign.com/military_defense_electronics/datacentric-systems-defense/.
- [3] Roel Wuyts, Stephane Ducasse, and Oscar Nierstrasz, "A Data-centric Approach to Composing Embedded, Real Time Software Components," Journal of Systems and Software, Vol.74, Issue 1, pp.25-34, 2005.
- [4] Gianpaolo Cugola and Gian Pietro Picco, "REDS: A Reconfigurable Dispatching System," Foundations of Software Engineering, Proceedings of the 6th International Workshop on Software Engineering and Middleware, pp.9-16, 2006.
- [5] Paolo Costa, Geoff Coulson, Cecilia Mascolo, Gian Pietro Picco, and Stefanos Zachariadis, "The RUNES Middleware: A Reconfigurable Component-based Approach to Networked Embedded Systems," IEEE 16th International Symposium on Personal, Indoor and Mobile Radio Communications, Vol. 2, pp.11-14, Sep. 2005.
- [6] B. Zieba, M. Glandrup, M. van Sinderen, and M. Wegdam, "Reconfiguration Service for Publish/

Subscribe Middleware Systems," Technical Report TR-CTIT-06-60, Centre for Telematics and Information Technology, University of Twente, Enschede. ISSN 1381-3625, <http://www.ctit.utwente.nl/library/techreports/techreps2006.shtml>, 2006.

[7] OMG(Object Management Group) Specification: Data Distribution Service for Real-Time Systems, Version 1.2, formal/2007-01-01, <http://www.omg.org/spec/DDS/1.2/PDF/>, 2007.

[8] DDS Technical White Paper: "An Introduction to DDS and Data Centric Communications," www.omg.org/news/whitepapers/Intro_To_DDS.pdf.

[9] OMG(Object Management Group) Specification: The Real-time Publish-Subscribe Wire Protocol DDS Interoperability Wire Protocol Specification, Version 2.1, formal/2009-01-05, <http://www.omg.org/spec/DDS/2.1/PDF/>, 2009.

[10] Ki-Jeong Kwon, Choong-Bum Park, and Hoon Choi, "DDSS: A Communication Middleware based on the DDS for Mobile and Pervasive Systems," The 10th International Conference on Advanced Communication Technology, Vol.2, pp.1364-1369, Feb. 2008.

[11] J. Sanchez-Monedero, J. Povedano-Molina, and J. Lopez-Soler, "Scalable DDS Discovery Protocols based on Bloom Filters," Real-time & Embedded Systems Workshop, OMG, Arlington, VA, USA, 11th Jul. 2007.

[12] 차다함, 강민균, 김용연, 권기정, 이균정, 최훈, "멀티 서브네트워크 환경을 위한 DDS Discovery 메커니즘", 한국정보과학회 컴퓨터종합학술대회(KCC2008) 춘계학술발표회 논문집, 제35권 제1호, pp. 251-252, 2008.06.

[13] M. C. Little, S. K. Shrivastava, and N. A. Speirs, "Using Bloom Filters to Speed-up Name Lookup in Distributed Systems," The Computer Journal, Vol.45, No.6, pp.645-652, 2002.

[14] 강부중, 노인규, 임을규, "공간 효율적인 블룸 필터링 방법의 소개", 한국컴퓨터종합학술대회 논문집, Vol. 35, No. 1(D), pp.1-4, 2008.

[15] 권남, 박재현, "Attenuated Bloom Filter의 전달을 이용한 APS의 성능 개선", 한국컴퓨터종합학술대회 논문집, Vol. 33, No. 1(D), pp.67-69, 2006. 홍길동, 허준, "고속 태그 식별을 위한 동적 알고리즘," 한국해양정보통신학회, 제10권, 제5호, pp.1-10, 2006.

저자소개

권기정(Ki Jung Kwon)



1988년 전북대학교 컴퓨터공학과
학사
1991년 전북대학교 전자공학과
석사

2005년 - 현재 충남대학교 박사과정 재학 중
1991년 - 현재 국방과학연구소 선임연구원
※관심분야: 분산 시스템 미들웨어

유용덕(Yong Duck You)



1999년 충남대학교 컴퓨터공학과
학사
2002년 충남대학교 컴퓨터공학
석사

2007년 충남대학교 컴퓨터공학 박사
2007년 - 2008년 충남대학교
소프트웨어 연구소 전임연구원
2008년 - 현재 관계아 솔루션
※관심분야: 모바일 컴퓨팅/분산 시스템 미들웨어



최 훈(Hoon Choi)

1983년 서울대학교 컴퓨터공학과
학사

1990년 Duke University 전산학과
석사

1993년 Duke University 전산학과 박사

1983년 - 1996년 한국전자통신연구원
광대역통신망연구부 근무

1996년 - 현재 충남대학교 컴퓨터공학과 교수

2000년 미국 NIST(National Institute of Standards and
Technologies) 객원연구원

※관심분야: 모바일 컴퓨팅/분산 시스템 미들웨어,
운영체제