해군 전투체계 환경에서의 인터페이스 설계도구에 관한 연구

김건형*, 임국현*, 윤관섭*
*LIG넥스원 전투체계연구센터
e-mail:gunhyung.kim@lignex1.com

A Study on Interface Design Tools for Naval Combat System

Keon-Hyung Kim*, Kook-Hyun Yim*, Kwan-Seob Yoon*
*Combat System R&D Lab, LIG Nex1

요 약

본 논문에서는 해군 전투체계 시스템과 같이 다중 노드 네트워크 환경에서의 인터페이스 설계도구에 대한 연구를 진행하였다. 이를 위해 해군 전투체계 시스템에서 사용되는 Real-time Middleware protocol 인 DDS(Data Distribution System)에 대해 설명하였고 이를 이용하여 데이터 통신을 할 경우, 메시지를 정의하고 송수신할 수 있는 노드를 정의하는 설계도구의 요구사항을 정리하였다. 이러한 요구사항을 바탕으로 개발한 인터페이스 설계 도구인 ATHENA(Advanced Topic management Host for ENterprise Application)를 소개하고 마지막으로는 향후 발전 방향에 대하여 논의하였다.

1. 서론

최근 해군 전투체계 시스템은 데이터 통신을 위해 DDS(Data Distribution System)를 이용하고 있다. DDS는 분산 시스템을 위한 데이터 중심의 Publish/Subscribe 구조의 실시간 미들웨어 서비스이다. DDS는 여러 노드들이 동시다발적으로 데이터를 송수신하는 시스템에 알맞도록 설계되어 있기 때문에 해군 전투체계와 같이 여러 노드들이 정보를 생성하고 처리하는 복잡한 네트워크 환경에서의 데이터 처리 방법을 단순화 해주는 네트워크 미들웨어라고 볼 수 있다.

하지만, 이미 널리 DDS가 해군 전투체계에 활용되고 있음에도 불구하고, Topic 및 Publish/Subscribe 설계에 관한 도구 지원은 미비하다.

그리하여, 본 논문에서는 보다 효율적인 DDS 인터페이스 설계 도구인 ATHENA(Advanced Topic management Host for ENterprise Application)에 대하여 소개하고 기술적 특징을 설명하여 개발자에게 도움을 주고자 한다.

2. DDS 소개

DDS는 OMG(Object Management Group)표준의 실시간 데이터 분산 서비스를 제공하는 미들웨어이다. 아래의 그림은 DDS의 기본 구조이다. 송수신하고자 하는 데이터 단위를 Topic이라고 부르며, 이 Topic들을 송수신하는 노드를 DomainParticipant라고 부른다. DomainParticipant는 자신이 가입한 Domain에 속한 DomainParticipant들 간에만 Topic의 송수신이 가능하다. DomainParticipant에 속하여 Topic을 직접 송신하는 부분은 Publisher, 수신하는 부분은 Subscriber라고 부른다. 각각의 DomainParticipant들은 자신이 송신하고자 하는 Topic을 Publisher에 수신하고자 하는 Topic은 Subscriber에 등록해 놓으면 해당 Topic의 송수신을 할 수 있게 된다.

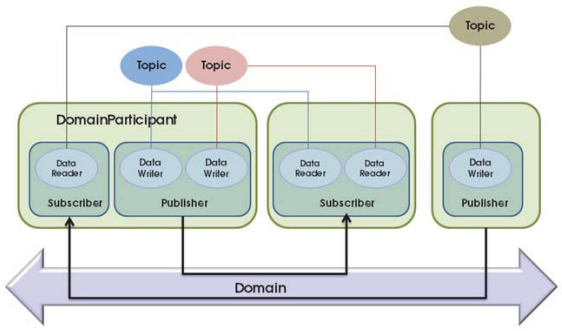

그림 1 DDS 기본 구조

DDS는 비연결지향 방식으로, 자신이 가입해있는 Domain에 속한 DomainParticipant들에 대한 식별을 자동으로 수행하고 그 리스트를 미들웨어 단에서 처리하기 때문에 어플리케이션에서는 연결에 대한 관리를 수행할 필요가 없으며 유지보수 비용이 절감되는 효과를 얻을 수 있다.

3. 인터페이스 개발도구의 요구사항

개발자에게 보다 편리한 인터페이스 설계도구를 제공하기 위해서는 개발자의 작업을 대행해 줄 수 있는 설계도구가 제공되어야 한다. 또한, 설계도구의 사용보급을 위해서 모든 개발자가 쉽게 인터페이스를 설계할 수 있는 도구가 요구된다. 이러한 기능을 효율적으로 제공하기 위해 인터페이스 설계도구가 갖추어야 할 요구사항을 다음과 같이 세 가지 측면에서 제안한다.

(1) 확장성

확장성의 문제는 개발도구의 확장으로 볼 수 있다. 개발도구는 사용자의 편의와 효율을 위하여 개발도구의 구조와 패러다임이 지속적으로 변할 수 있다. 예를 들면, 개발도구 구조 설계시 고려하지 않은 특정 서비스를 지원하기 위해 새로운 모듈의 추가, 접속방식의 변경으로 인한 모듈 변경 등이다. ATHENA는 RUBY를 이용한 MVC(Model, View, Controller)구조로 개발하여 향후 추가하는 모듈을 쉽게 통합할 수 있도록 하였다.

(2) 무결성

인터페이스 설계 시 가장 중요한 점은 외부인터페이스의 무결성이다. 각각의 모듈에서 서로 다른 인터페이스를 사용하여 개발하게 된다면 통합시험 단계에서 많은 수정이 뒤따르게 된다. ATHENA는 웹 환경에서의 인터페이스 설계 환경을 제공하여 서로 다른 모듈간 동일한 Topic을 정의하여 Publish/Subscribe를 등록할 수 있도록 지원하고 Topic의 변경사항을 기록하여 무결성을 지원한다.

(3) 자동화

인터페이스가 복잡해짐에 따라 설계한 인터페이스를 코드에 바로 적용시킬 수 있는 기능이 요구된다. ATHENA는 목적프로그램에서 별도의 작업 없이 DDS통신을 할 수 있도록 하는 Wrapper Class를 생성하여 개발소요시간을 단축시킨다.

4. ATHENA의 구조 및 동작

(1) 구조

ATHENA의 구조는 그림2에서 나타낸 것과 같은 구조를 가진다.

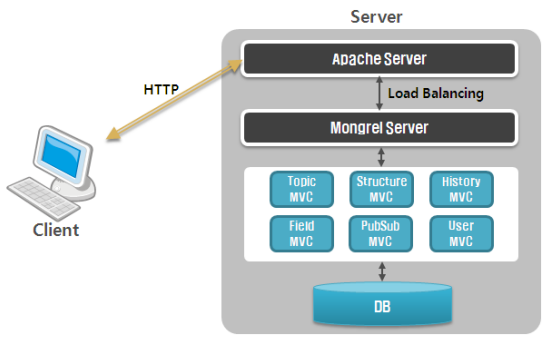

그림 2 ATHENA의 구조

ATHENA의 구성은 Apache Server, Mongrel Server, 여섯 개의 MVC구조를 가진 모듈과 DB로 이루어져있다. Apache Server와 Mongrel Server는 HTTP를 통한 Client와의 입출력을 처리하고, 여섯 개의 MVC구조를 가진 모듈은 사용자의 입력을 받아 처리하여 그 결과를 DB에 반영하고 화면에 보여주는 역할을 담당한다. DB는 여러 사용자간에 동일한 데이터를 처리하고 저장할 수 있도록 하기 위해 사용되었다.

(2) 동작

ATHENA를 이용하여 인터페이스를 설계하기 위해서는 Server관리자가 프로젝트 정보를 등록해주어야 한다. 프로젝트 생성에 필요한 정보는 프로젝트명, CSCI-CSC(Computer Software Configuration Item - Computer Software Componet) 구조, 사용자 정보이다. 그림3은 프로젝트를 등록하여 ATHENA 서버가 구동된 결과이다.

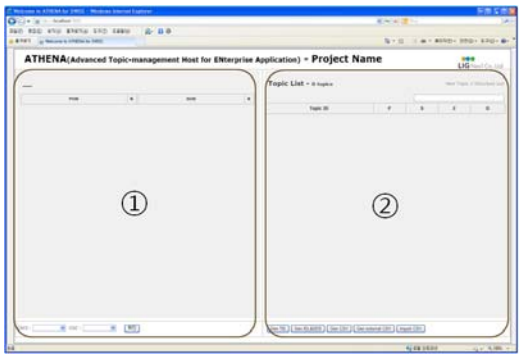

그림 3 ATHENA 초기 화면

1번 영역은 CSCI-CSC를 선택하여 Publish/Subscribe하는 Topic을 전시하는 영역이며, 2번 영역은 전체 Topic의 리스트 및 필드정보를 전시하는 영역이다.

다음으로는 Topic을 생성하여 Publish/Subscribe 목록에 등록을 해주어야 한다. 그림4는 Topic을 생성하여 선택한 CSCI-CSC의 Publish 목록에 등록한 결과이다.

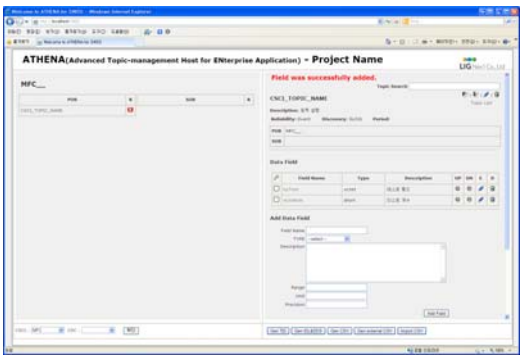

그림 4 Topic생성 및 등록

여기서 등록한 Topic은 다른 CSCI-CSC 담당자도 동일하게 조회가 가능하며 Publish/Subscribe 목록에 등록 또한 가능하다.

마지막으로 설계한 인터페이스를 코드화 하기 위하여 Wrapper Class 생성을 요구하면 ATHENA는

Publish/Subscribe 목록에 등록된 Topic을 바탕으로 Wrapper Class 파일을 생성한다. 이렇게 생성된 파일은 목적프로그램에서 include하면 DDS통신을 할 수 있는 기능을 제공한다.

5. 결론

본 연구에서는 해군 전투체계와 같이 다중 노드의 네트워크 환경을 가진 시스템의 인터페이스를 설계할 때의 문제점들을 해결하기 위해 인터페이스 설계도구가 가져야 할 요구사항을 정리하고, 이러한 요구사항을 바탕으로 설계도구를 개발하였다. 앞으로 본연구에서 개발한 인터페이스 설계도구인 ATHENA를 이용하면 기능의 확장이 용이하며, 인터페이스의 무결성을 보장 받을 수 있으며 코드생성 기능을 통해 DDS 비전문가이더라도 손쉽게 DDS를 이용한 어플리케이션 개발할 수 있을 것이다.

앞으로는 설계한 인터페이스를 자동으로 문서화하여 인터페이스 설계 기술서로 활용할 수 있도록 하는 기능, 인터페이스 시뮬레이터 연동 기능, 사용자 권한 관리 기능의 지원을 위한 ATHENA 기능 확장을 진행할 것이다.

참고문헌

[1] RTI Data Distribution Service Core Libraries and Utilities Users's Manual, Version 4.5
[2] DDS standard, OMG, www.omgwiki.org/dds
[3] Real Time Innovations (RTI), www.rti.com