DDS 응용 개발 시 통합개발지원 환경의 효율적 활용을 위한 기법 개발 연구

**1차년도 보고서**

통합개발지원 환경을 활용한 DDS 응용 개발을 위한

설계 방법론 수립

2014.06.25

고려대학교 기술경영전문대학원 강석중 교수

목차

[서론 4](#_Toc389746183)

[DDS 응용프로그램 개발을 위한 설계방법 5](#_Toc389746184)

[DIDE 컴포넌트 소프트웨어 5](#_Toc389746185)

[DIDE 개발 환경 5](#_Toc389746186)

[개발 절차 7](#_Toc389746187)

[결론 9](#_Toc389746188)

# 서론

본 보고서에서는 DIDE를 이용하여 DDS 응용프로그램을 개발하고자 하는 경우에 DIDE를 구성하는 각 컴포넌트를 이용하여 개발 및 시험하고자 하는 절차에 대하여 설명하고, 각 절차에 따라 지켜야 할 주의 사항에 대하여 언급하고자 한다.

*DIDE에 대하여 간략하게 소개한다.*

# DDS 응용프로그램 개발을 위한 설계방법

## DIDE 컴포넌트 소프트웨어

표 . DIDE 컴포넌트 기능 요약

|  |  |  |
| --- | --- | --- |
| **컴포넌트 명칭** | **기능** | **사용 단계** |
| 응용 모델러, | * Eclipse plug-in 형태 * 토픽 IDL 파일 * 소스 코드 생성 | 설계 및 구현 |
| 토픽 레포지토리 | * 토픽 정의, 생성, 등록, 알림, 버전관리 | 설계 및 구현 |
| 토픽 분석기 | * 실시간 DDS 엔티티 속성, QoS 등의 정보 수집 및 비교 | 모니터링 및 분석 |
| 실시간 모니터 | * CPU/Memory usage * DDS 엔티티의 QoS 체킹 * DDS 토픽 통신 통계정보 수집 | 모니터링 및 분석 |
| 로그 관리자 | * 로그 수집 및 저장 * 로그 데이터 분석 | 모니터링 및 분석 |
| 토픽 기록기 | * 토픽 및 샘플 정보 수집 및 저장 * 토픽 재생 | 시험 및 개선 |
| 토픽 시뮬레이터 | * 시나리오 작성 및 관리 * 시나리오 실행 | 시험 및 개선 |
| 토픽 튜너 | * 로컬 및 리모트 DDS 엔티티 속성 조회 * 로컬 및 리모트 DDS 엔티티 QoS 설정 | 시험 및 개선 |
| DDS-DB 연동 | * DB 연동 기능 제공 | 지원도구 |
| 고가용성 지원 | * 프로세스 모니터링을 통하여 고가용성 기능 제공 | 지원도구 |
| 형상관리 도구 | * 산출물 버전 관리 기능 제공 | 지원도구 |
| DDS 엔진 연동 | * 다양한 DDS 엔진에 대하여 단일 인터페이스 제공 기능 | 지원 도구 |

## DIDE 개발 환경

DIDE 전체 도구의 구조도는 아래 그림과 같다. 이 도구들은 설계, 구현, 시험 등의 일반적인 SW개발 과정과 크게 다르지 않으며, 각 개발 단계에서 효율적으로 사용될 수 있도록 개발되어 있다.

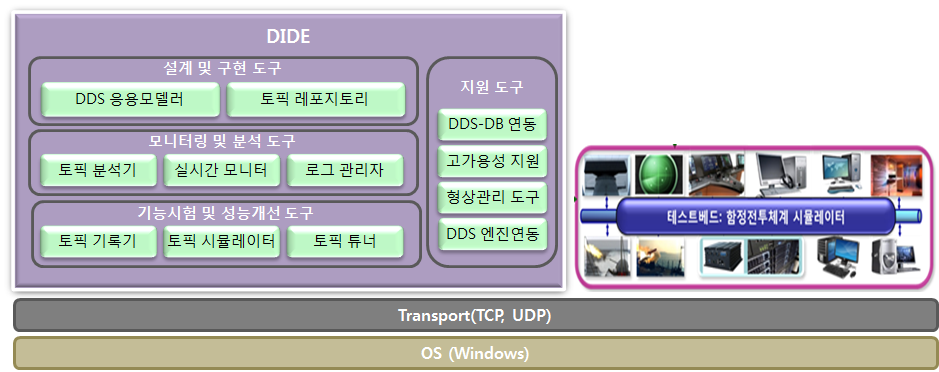


그림 . DIDE 구조도

위의 구조도를 다음 그림과 같이 연관 관계를 살펴볼 수 있다.

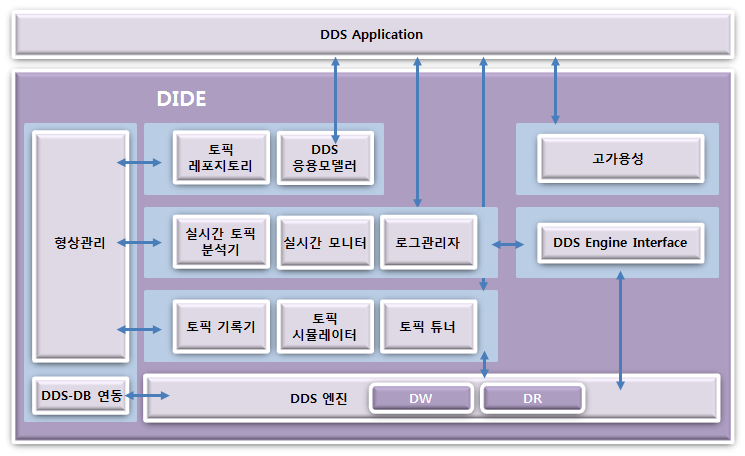


그림 . DIDE 컴포넌트 소프트웨어 구조도

그림 . 컴포넌트간 연관 관계

## 개발 절차

본 절에서는 개략적으로 DIDE 컴포넌트를 이용하여 DDS 응용프로그램을 설계, 개발하고 시험하는 과정에 대하여 간략하게 소개하고자 한다.

각 컴포넌트를 이용하여 DDS 응용프로그램을 개발하고 시험하는 일반적인 절차는 아래 그림과 같다.

사용자는 이클립스 플러그-인 형태로 개발된 응용 모델러를 이용하여 토픽 설계서에 정의된 토픽에 대하여



그림 . DDS 응용프로그램 개발 및 시험 절차

Step 1. 응용 모델러를 이용하여 DDS 응용프로그램 개발.

* DDS 응용프로그램 타입의 프로젝트를 생성한다.
* 생성된 프로젝트에 DDS 엔티티를 갖기 위한 Application 모듈을 생성한다. 이 Application 모듈은 향후 실행 파일로 변환된다.
* 생성된 Application 모듈 내에 GUI Editor를 이용하여 Domain participant를 생성한다.
* Domain participant가 생성된 후 Publisher 또는 Subscriber를 Domain participant내에 생성한다.
* 토픽을 주고 받을 DataReader 또는 DataWriter를 Publisher 또는 Subscriber내에 생성한다.
* Partition을 생성한 후 GUI Editor를 이용하여 이 partition을 통하여 통신할 DataReader와 DataWriter를 연결한다.

# 결론