

## 격납건물 누설시험용 웹 모니터링 시스템 개발

### Web Monitoring System Development for ILRT

김 성 철\*, 조 형 돈\*\*, 이 동 길\*\*\*  
(Sung Chul Kim and Hyoung Don Cho)

\*가온소프트(주)(전화:(055)231-7707, 팩스:(055)232-6667, E-mail : ksc@gaonsoft.com)

\*\*가온소프트(주)(전화:(055)231-7707, 팩스:(055)232-6667, E-mail : chohd@gaonsoft.com)

\*\*\*가온소프트(주)(전화:(055)231-7707, 팩스:(055)232-6667, E-mail : leedongkil@gaonsoft.com)

**Abstract** : This paper describes the architecture and development of real-time monitoring system for ILRT (Integrated Leakage Rate Test) . Object Oriented approach is used during the analysis and design. The architecture can be spited into two modules. First one is the Interface module, which, takes care of the communication between the server and the clients, and the second module is the Graphical User Interface (GUI) module, which takes care of the GUI in the web browser of the client. The technologies used includes Java , Java Bean and Java Applet. Etc.

**Keywords** : Web Monitoring, ILRT(Integrated Leakage Rate Test), Java Applet, Java Bean

#### 1. 서론

원자력 발전소에서의 격납건물은 원자로를 비롯한 핵 중기 공급계통을 수용하고 있는 구조물로서, 외부로부터의 비산물에 대한 방호뿐만 아니라 원자로 냉각재 상실사고와 같은 가상사고(DBA) 및 중대사고시 핵분열 생성물에 의하여 외부로 누출되지 않도록 하는 중요한 차폐물로서의 역할을 한다. 따라서 격납건물의 기밀성 보장은 환경으로의 방사능 유출을 방지한다는 의미에서 발전소 가동 전에 그 성능을 확인하기 위한 시험은 대단히 중요한 의미를 가지는 것이다. 그리고, 격납건물내에서 격납건물 예상 최대 압력(Pa)으로 가압한 상태에서 격납건물 전체에 대한 누설률을 측정하는 시험을 종합 누설률 시험(ILRT; Integrated Leakage Rate Test)이라고 한다.

종합 누설률 시험의 일반적인 시험방법은 격납건물 전 관통부 및 격리밸브를 폐쇄시킨 후 격납건물 내부를 공기압축기 가압관을 통해 시험압력(51.5, -0.0~+2.0 psig)까지 가압한 후 시험 착수 전 4시간 이상의 격납건물 대기 안정화 시간을 가져야 하며, 시험기간 동안 이상기체상태로 가정한 격납건물 내부의 공기건구온도, 압력, 습증기온도(수증기압)을 최소한 24시간 이상 동일시간 간격(15분)으로 30개조 이상의 데이터를 수집하여 누설률을 측정한다[1].

ILRT 시험을 위한 웹 모니터링 시스템은 ILRT 전 시험 과정을 통하여 수집되는 계측데이터 및 시험상황을 웹을 통해 실시간으로 모니터링하기 위한 시스템이다[2]. 본 연구과제에서의 개발 내용은 다음과 같이 크게 두 부분으로 되어있다. 첫째, 실시간 계측데이터에

대한 처리 및 전송을 위한 웹 모니터링 인터페이스(Web Monitoring Interface) 및 기능 모듈 개발 둘째, 클라이언트의 웹 브라우저에서 웹 모니터링 화면을 디스플레이하기 위해 사용하는 그래픽 모듈 개발.

실시간 웹 모니터링 시스템이 갖추어야 할 요구사항 중 시스템의 동작이나 행위와 관련된 전반적인 제약사항을 크게 다음과 같이 네 가지로 정의하였다.

#### 가. 성능

ILRT 전체 시험기간은 약 5일 정도이며 시험기간 동안 이상기체상태로 가정한 격납건물 내부의 공기건구온도, 압력, 습증기온도(수증기압)을 최소한 24시간 이상 동일시간 간격(1~2분)으로 30개조 이상의 데이터를 수집하여 누설률을 측정한다. 따라서, 각 계측장비로부터 실시간 계측되는 온도·압력 데이터 들을 웹을 통해 그래프 형태로 사용자에게 실시간 제공해 주어야 한다. 또한, 클라이언트 사용자의 웹 브라우저 화면에서의 동적인 그래프 화면 디스플레이를 위한 기능이 요구된다.

#### 나. 신뢰도

ILRT 웹 모니터링을 위한 시스템은 시험기간(5일 정도)이 매우 짧고 동시 사용자가 많지 않기 때문에 시스템 부하에 대한 문제는 덜 중요하다. 반면에, 웹 서버와의 데이터 통신 및 웹 모니터링을 위한 그래픽 모듈들을 다운로드 받아 클라이언트의 웹 브라우저에서 실행해야 하므로 실행 코드의 안전성에 대한 검증 문제가 대두된다. 이 문제의 해결을 위해, 자바 애플릿을 사용하였으며, 자바코드 자체에서 실행 전에 먼저, 실행코드에 대한 검증과정을 거치기 때문에 안전성을

보장할 수 있다.

다. 사용자 편의성

실제로 웹 모니터링 시스템의 개발 목적이 사용자 편의성을 도모하기 위한 것인 만큼 사용자 인터페이스 및 화면을 사용자 관점에서 일관되며 효과적으로 정의할 수 있어야 한다. 이를 위해, 클라이언트의 모니터링을 위한 그래픽 모듈 개발 시 필요한 클래스들을 계층 구조를 사용하여 모듈 별로 개발함으로써 모듈의 변경 및 확장이 유연하게 하며, 각 오브젝트(object) 별로 사용자가 쉽게 해당 속성항목들을 변경할 수 있도록 해야 한다. 또한, 디스플레이와 관련된 클래스들을 하나의 압축파일 형태(JAR 파일)로 만들으로써 웹 서버로부터의 다운로드 속도 및 실행시간을 단축할 수 있도록 한다.

라. 유지보수성

구축된 시스템은 운용되는 과정에서 다른 플랫폼으로의 이식, 기능의 추가 및 변경 등의 이유에 의해서 수정될 있다. 따라서, 이러한 유지보수를 쉬우면서 효율적으로 할 수 있어야 한다. 이를 위하여 시스템 설계 및 개발 시 유스케이스 모델링과 같은 객체지향적인 개발 방법론을 적용하여 개발하였으며 모듈개발시 MVC(Model-View-Control) 모델을 적용하여 시스템의 확장 및 변경이 쉽도록 되어있다[3].

II. 웹 모니터링 시스템 설계

1. 시스템 구성

실시간 웹 모니터링 시스템 구성을 위한 H/W 및 S/W 구성은 다음과 같다..

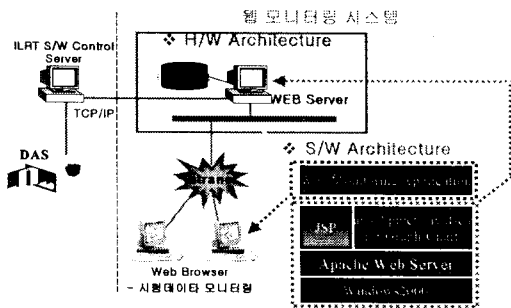


그림 1. 전체시스템 아키텍처

1) H/W Architecture

웹 모니터링 시스템 구축을 위한 H/W 구성요소들은 다음과 같다.

- 웹 서버 (Pentium III 이상)

실시간 웹 모니터링을 위한 웹 서버로써의 역할을 수행하며, Control 서버로부터 실시간 계측 데이터를 받아 모든 클라이언트 사용자에게 전송하거나 계측 데이터를 샘플링하여 오라클 DB에 저장한 후 시험 종

료 후 리포팅을 위한 서비스를 제공한다

- 오라클 DB 서버 (Oracle 9i)

시험 데이터에 대한 리포팅을 위하여 실시간으로 받은 계측 데이터들 중 일부를 저장해 두기 위한 데이터베이스 서버로 사용된다.

2) S/W Architecture

기본적으로 웹 페이지(html), 자바 애플릿과 자바 빈 클래스들은 Window2000 환경의 Apache 웹 서버에서 서비스 될 것이며, 클라이언트에서는 웹 서버로부터 자바 애플릿, 자바 빈 그리고 다수의 그래프 관련 클래스들을 담고 있는 하나의 JAR 파일을 다운로드 받아 클라이언트의 웹 브라우저 상에서 구동함으로써 파일 다운로드 및 실행속도의 향상을 도모했다.

웹 모니터링 기능구현을 위한 S/W 구성요소들에 대한 설명은 다음과 같다.

- 자바 애플릿(Java Applet)과 자바 빈(Java Bean): 클라이언트 사용자의 웹 브라우저 화면에서의 그래프 출력력을 위한 모듈
- Web Monitoring Application: Control 서버 및 웹 서버, 웹 서버와 클라이언트 프로그램 사이에서의 실시간 데이터 전송 및 처리를 위한 모듈

2. S/W Interface 및 기능 모듈

그림 2은 웹 모니터링을 위한 각 시스템들 간의 인터페이스 및 실시간 계측데이터에 대한 처리 및 전송을 위한 기능 모듈들 그리고 모듈들 간의 구성관계를 나타낸 것이다. 여기서, 각 계측장비로부터 온도, 압력과 같은 계측데이터를 실시간으로 수집하는 DAS 시스템과 수집한 계측데이터에 대한 분석 및 처리를 담당하는 Control Server는 웹 모니터링 시스템에 포함되지 않지만 전체적인 시스템에서의 연관관계를 설명하기 위하여 포함시켰다.

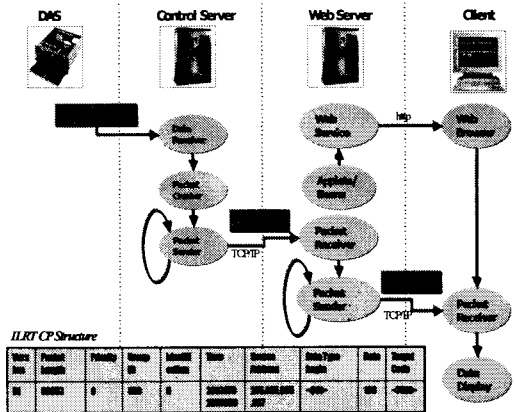


그림2. S/W Interface 및 기능 모듈

위의 그림을 토대로 본 시스템의 전체적인 흐름을 설명하면 다음과 같다. 먼저, 클라이언트 사용자가 웹 서버의 웹 페이지에 접속하면, 웹 페이지에 포함되어 있는 자바 애플릿과 자바 빈 클래스를 다운로드 받게 되며, 웹 서버와의 연결을 만들게 된다. 그 다음에 Control 서버가 DAS로부터 수집된 계속 데이터 정보를 받게 되면, 받은 데이터를 본 시스템에서 정한 데이터 포맷에 맞추어 하나의 패킷 형태로 만들고, 그것을 웹 서버로 전송한다. 그러면, 웹 서버에서는 현재 웹 서버에 연결되어 있는 모든 클라이언트들에게 웹 서버로부터 받은 계속 데이터 자료를 넘겨주게 된다. 마지막으로 각각의 클라이언트에서 새로운 계속데이터가 수신될 때마다 클라이언트의 웹 브라우저의 화면을 갱신함으로써 실제적인 웹 모니터링을 할 수 있게 된다.

각 모듈에 대한 보다 자세한 설명은 다음과 같다.

- Data Receiver: DAS 시스템으로부터 계속 데이터를 실시간으로 받는 모듈
- Packet Creator: 수신한 계속데이터를 CP(Communication Protocol)에 맞추어 하나의 Packet 을 만드는 모듈
- Packet Sender: Packet을 전송하는 모듈
- Web Service: 클라이언트에게 제공되어질 웹 서비스 내용
- Applets/Beans: 자바 애플릿 및 자바 빈 클래스들로써 그래프 관련 모듈들
- Web Browser: 클라이언트에서 웹 모니터링을 위해 사용할 웹 브라우저 프로그램이며, Internet Explorer 6.0 이상을 기준으로 함
- Packet Receiver: Packet을 수신함
- Data Display: 수신한 Packet 정보를 파싱하여 클라이언트 화면에 데이터를 디스플레이 함

### III. 웹 모니터링 시스템 개발

#### 1. 웹 모니터링을 위한 그래픽 모듈 개발

웹 모니터링을 위한 그래픽 모듈은 클라이언트 사용자의 웹 브라우저 화면상에서 그래프 등과 같은 디스플레이를 위한 모듈이며, 크게 자바 애플릿(Java Applet), 자바 빈(Java Bean) 그리고 그래프를 위한 여러 클래스들로 구성되어 있으며, Rational Rose라는 분석툴과 NetBean이라는 자바 개발 툴을 사용하여 개발하였다[4].

#### 가. Applet & GraphBean Classes

웹 모니터링 화면은 기본적으로 웹 페이지(html)안에 포함된 자바 애플릿(Java Applet)를 통해 디스플레이 되며, 실제적으로 그래프를 그리는 등의 일을 수행하는 것은 자바 애플릿 안에 있는 자바 빈(예: LineGraphBean) 클래스가 이 일을 맡아서 처리한다. 여기서, DataCollector는 쓰레드로 구동되며, 소켓을 통

하여 데이터를 수집하기 위한 클래스이다.

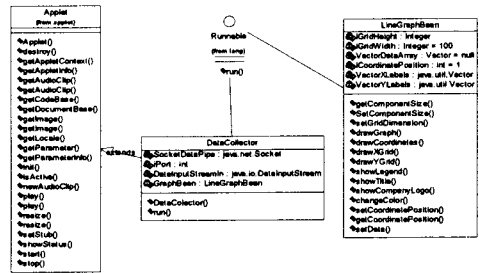


그림 3. Applet & GraphBean Classes

#### 나. Graph Display Classes

클라이언트의 웹 브라우저 화면에서 그래프를 그리는 데 필요한 그래픽 관련 주요 클래스는 GLine, GAxis, GLineStyle, GLabel 등이 있으며 클래스에 대한 정의는 다음과 같다.

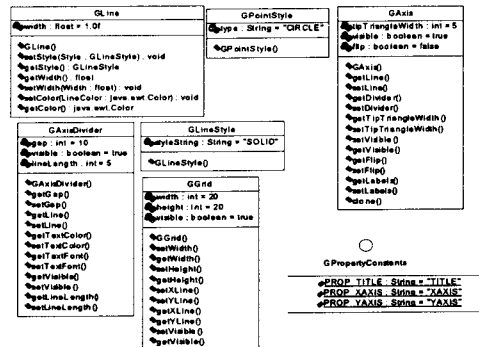


그림 4. Graph Display Classes

#### 2. 실시간 웹 모니터링을 위한 기능 모듈 개발

실시간 웹 모니터링 기능모듈은 서버와 클라이언트 사이에서 계속데이터에 대한 전송 및 수신과 관련된 기능을 주로 수행하는 모듈이며, 웹 서버 및 클라이언트 각각의 기능 모듈은 다음과 같다.

#### 가. Server Side Classes

웹 모니터링을 위한 서버 측 클래스들은 주로 클라이언트와의 연결관리 및 실시간 데이터 수신 및 클라이언트로의 데이터 전송 등의 일을 한다.

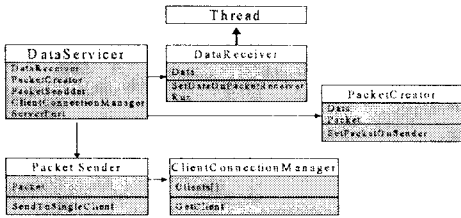


그림 5. Server Side Classes

각 클래스에 대한 설명은 다음과 같다.

- DataServicer: 서버측 메인 구동 프로그램으로 클라이언트와의 연결관리 및 클라이언트로 데이터 전송 등의 일을 하는 다른 모듈들을 실행함
- DataReceiver: 쓰레드로 구동되며, 각종 취득 데이터를 수신하는 모듈
- PacketCreator: 데이터 송수신을 위한 데이터 포맷을 만들
- PacketSender: 현재 웹 모니터링 서버에 연결되어 있는 모든 클라이언트들에게 위에서 만든 데이터 패킷을 전송
- ClientConnectionManager: 클라이언트와의 연결관리를 담당

#### 나. Client Side Classes

웹 모니터링을 위한 클라이언트 측 클래스들은 서버로부터 받은 취득데이터를 서버로부터 수신하여 분석한 후 웹 화면에 디스플레이 하는 일을 담당하며, 다음과 같은 클래스들로 구성되어 있다.

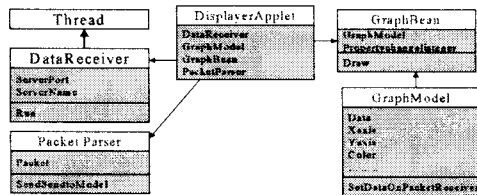


그림 6. Client Side Classes

각 클래스에 대한 설명은 다음과 같다.

- DisplayApplet: 웹 모니터링에 필요한 클라이언트 측 클래스들을 사용하는 자바 애플릿 클래스
- GraphBean: 그래프를 그리기 위한 자바 빈 클래스
- DataReceiver: 쓰레드로 구동되며, 각종 취득 데이터를 수신하는 모듈
- PacketParser: 서버로부터 받은 패킷 데이터를 원하는 데이터 형태로 파싱하는 클래스
- GraphModel: GraphBean에서 사용되는 클래스로 그래프에 대한 데이터 모델 임

#### IV. 결론

본 논문은 ILRT 시험에서와 같이 다양한 계측 시험

데이터에 대한 실시간 웹 모니터링 기능이 필요한 경우 이러한 시스템의 구현을 위한 하나의 방법론을 제시하고자 한다. 아래 그림은 웹 모니터링 시스템의 Interface 및 기본 그래픽 모듈에 대한 테스트 예이다. 이 프로그램은 크게 실시간 데이터 전송을 위한 인터페이스 모듈과 클라이언트의 웹 화면에서 취득 데이터에 대한 실시간 디스플레이 모듈로 나뉘며, 1초 간격으로 샘플 데이터를 서버에서 클라이언트로 전송하여 테스트한 결과 양호한 응답 성능을 확인할 수 있었다.

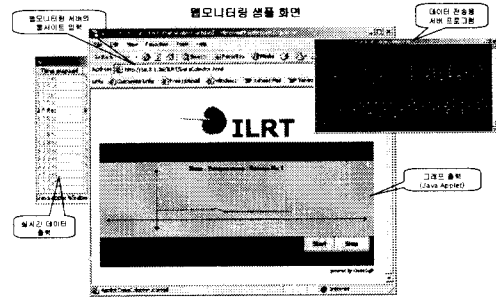


그림 7. 웹 모니터링 샘플 화면

각 모듈에 대한 보다 자세한 설명은 다음과 같다.

- 데이터 전송을 위한 서버프로그램  
계측 데이터를 클라이언트에 실시간으로 전송하기 위한 프로그램이며, Control 서버에서의 데이터 전송 모듈이다.
- 웹모니터링을 위한 클라이언트 화면  
웹서버로부터 웹모니터링을 위한 자바 애플릿을 다운로드 실행함으로써 웹 서버와의 TCP/IP 연결을 통해 실시간 계측데이터를 받으며, 애플릿의 그래프 빈 클래스를 통해 그래프 형태로 데이터가 출력되게 된다.

#### 참고문헌

[1] ANSI/ANS 56.8, 1994.  
 [2] 김윤기, 강문설, 김병기, “제조실행시스템의 기능 보안을 위한 웹 기반 공장 모니터링시스템의 설계 및 구현”, 정보처리학회 논문지, 제9권, 제4호, pp. 667-676, 2002.  
 [3] Budd, T., Understanding object-oriented programming with JAVA, Addison-Wesley, 1998  
 [4] 채홍석, 객체지향 CBD 개발 Bible, 도서출판 한빛 미디어, 서울, 2003.