

## Chapter 1. Overview

### 1.1. LENA는 무엇입니까?

LENA는 Enterprise 환경에서 JAVA J2EE Web Application을 서비스하는데 필요한 모든 구성 요소가 포함된

웹 미들웨어 솔루션이다.

LENA는 실질적인 웹 서비스를 제공하는 Server 제품들과 이를 통합적으로 관리하기 위한 Web UI 기반의

Manager Console로 이루어져있다. 사용자는 Manager Console을 통해 Server 설치부터 Parameter 설정 및

각 제품간 연동 등을 손쉽게 수행할 수 있으며, LENA만의 사용자 친화적인 UX/UI 설계로 웹 미들웨어 솔루션

사용에 익숙하지 않은 사용자라도 빠르게 사용법을 습득하여 웹 미들웨어 관련 지식을 갖출 수 있다.

LENA는 데이터 센터/클라우드 환경에서 웹 미들웨어 운영자들로부터 다년간의 운영 노하우를 집약하여

다양한 편의 기능을 제공한다. 이와 함께 대규모 트랜잭션을 안정적으로 처리하고 서비스 장애를 최소화 할

수 있도록 고가용성, 내결함성을 보완한 Clustering, 장애 진단/대응 기능을 제공한다.

다음 절은 LENA의 특징들에 대해 더 상세히 설명한다.

### 1.2. LENA 특징

Enterprise 요건 제공

LENA Web Application Server EE (J2EE Edition)는 Enterprise Web Application을 실행시키는데

필요한 EJB, JTA/XA, JMS, JAX-WS 등 J2EE Spec을 지원한다. 또한 타사 WAS에 비해 기동성능,

Application Deploy성능의 향상을 이루었고, CPU/Memory 등 자원 사용 효율성 역시 향상되

었다.

고가용성을 위해 필수인 Session Clustering을 위해 Session Server를 Enterprise Edition에서 기본으로 제공한다.

Open Source 호환성 보장

LENA Web Server, LENA WAS는 Open Source Base로 구현되어 완벽한 Open Source 호환성을

보장한다. Open Source 기반으로 작성된 Web Application이 별도의 수정 없이 LENA에 적용되므로

전환 Effort를 대폭 절감할 수 있다. 또한 Library 및 설정 관련하여 표준 기술을 사용하므로 Vendor

종속성을 해결하여 사용자의 IT Ownership을 강화할 수 있다.

지능형 장애 진단/대응 및 서비스 추적

Thread Pool Full / Hang / Out of memory 등 대표적인 장애 상황들에 대해 사전 정의된 Rule

기반으로 진단을 수행하여 적절한 조치를 수행할 수 있다. 실시간으로 수집되는 Monitoring 데이터를

기반으로 분석하여 Server가 실제 장애로 이어지기 전에 Fake Page 우회 혹은 강제 재기동 등을 통해

장애 상황을 방지하거나 장애로 이어지더라도 장애 시간을 최소화할 수 있다. 이와 동시에 관련된

Dump와 Report를 제공하여 원인 분석을 수행할 수 있게 한다.

Multi-Server 관리 및 Centralized Operation

여러 LENA Web Server와 LENA WAS를 하나의 Cluster로 묶을 수 있어 Single Operation으로 여러

Server를 동시에 제어할 수 있다. 일관적인 설정 동기화와 함께 Graceful Restart혹은 Rolling

Restart를 통해 안정적인 반영이 가능하다. 또한 Cluster로 묶이지 않은 Server들에 대해서도 Multi

Control을 지원한다.

운영 관점의 차별화 기능 제공

웹 미들웨어 솔루션을 운영하기에 편리한 다양한 편의 기능을 제공한다. Template 기반의 간단하고

빠른 Server설치와 Server 복제 기능을 이용하여 원하는 구성 Set을 짧은 시간 안에 구축이 가능하다.

Topology View를 통해 Server 모듈 간 구성 및 연동 정보를 한눈에 확인할 수 있어 가시성을 향상할 수

있다. Dashboard를 통해 운영 중인 시스템의 Event와 성능 현황 등을 확인할 수 있다. 그 밖에 Multi

Account 관리를 통해 메뉴/자원의 접근 권한 설정이 가능하며, 운영자 Action Tracing, 설정 변경 정보

History 추적 및 Restore 기능을 제공한다.

User Guide Chapter 1. Overview

Copyright© LG CNS. All rights reserved. 1

### 1.3. LENA 구성 요소 및 주요 개념

LENA는 Binary Package를 통해 제공이 되며 그 안에 필요한 모든 구성 요소를 포함하고 있다. 구성 요소는

크게 두 범주로 나누어진다.

- LENA를 운영 관리하기 위한 Management Module로 LENA Manager Console과 LENA Node Agent가

포함된다.

- 실질적인 Web Service를 담당하는 Server Module로 LENA Web Server, LENA WAS(Web

Application

Server), LENA Session Server를 포함된다.

다음은 각 구성 요소 별 상세 설명과 함께 이에 따른 주요 개념을 다룬다.

### 1.3.1. Management Module

LENA Manager

LENA Manager는 Web UI를 통해 LENA의 모든 자원/기능을 설정 및 제어할 수 있도록 설계된 Web

Application이다. LENA Package에 준비된 스크립트를 통해 설치 및 기동할 수 있다. LENA

Manager를 통해 Server 설치/관리를 수행하려면 Node Agent 및 Advertiser와의 연동을 구성해야

한다.

다음은 LENA Manager에서 제공하는 대표적인 기능과 개념에 대해 설명한다. 아래에 설명되지 않은

상세 사항들에 대해서는 “운영자가이드”를 참조한다.

- Dashboard

LENA Node, Server의 자원 현황과 Event 확인

- Server

LENA Node의 등록, Web Server, WAS, Session Server 를 설치, 설정 관리 및 기동/중지 제어

- System

LENA Node, Server가 관리되는 최소 단위. 하나의 System하위에는 여러 Node를 등록할 수 있으나, 하나의 Node는 여러 System에 중복 등록이 불가하다.

- Node

Node Agent와 1:1로 대응되는 개념. Manager에서 원격지에 있는 Server에 명령을 수행하기

위해선 Node Agent를 통해야 한다.

- Cluster

Multi-Server 관리. 하나의 System 내의 Server들에 대해서만 Cluster 구성이 가능하며, 하나의 Server는 여러 Cluster에 속할 수 없다.

- Server Cluster

여러 LENA Web Server, WAS를 하나의 묶음으로 관리할 수 있다. 설정 동기화, 통합 재 기동 등의 다양한 편의 기능 제공.

- Resource

LENA에서 제공하는 Module이 아니지만 LENA Server와 밀접하게 연동되는 자원에 대해 명세를

정의함으로써 Resource로 사용한다. Resource는 LENA WAS별로 Local 설정할 수 있지만,

Resource 메뉴를 통해 Global 설정하여 WAS에서 공통 Import하는 방식을 통해 중복된 작업을

피할 수 있다.

- Database

DBMS의 IP, Port, Driver 등 물리적 명세를 정의. 하나의 DBMS와 1:1로 대응된다.

- Datasource

DB Connection Pool을 LENA WAS에서 구성하기 위해 JNDI Name, Url, User

ID/Password등을 지정한다. 하나의Database하위에 여러 Datasource를 구성할 수 있다.

- Application

LENA WAS를 통해 실행할 Application의 위치와 Context Path를 지정한다.

User Guide Chapter 1. Overview

Copyright© LG CNS. All rights reserved. 2

- Topology

LENA Manager상에 설치되고 연동 구성되어 있는 LENA Web Server, LENA WAS, LENA Session Server 등의 구성 현황을 Topology Diagram 형식으로 표현한다. 이 기능을 통해 Server의

간단한 설치와 기동/중지 제어도 가능하다.

- Diagnostics

LENA Node와 Server에 대한 자원 모니터링과 그에 연계된 다양한 기능 탑재.

LENA Node Agent

LENA Node Agent는 LENA Manager에 Node를 등록할 때 Node와 1:1로 대응되며 LENA Package에 기본적으로 설치되어 있으며 준비된 스크립트를 통해 기동할 수 있다. 주요 역할은 LENA

Manager를 통해 Node 하위의 Server에 명령을 수행할 때 Node Agent가 이를 담당하며 모니터링 및

상태 데이터를 LENA Manager로 전송하는 기능도 수행한다. 물리 서버 당 1개의 Node Agent를

설치하는 것이 기본이지만, 필요에 따라 여러 개의 Node Agent를 구성할 수 있다. LENA Web Server,

LENA WAS, LENA Session Server는 LENA Node 하위에 구성되며, LENA Node는 하나의 System

하위에 구성된다.

### 1.3.2. Server Module

LENA Web Server

LENA Web Server는 정적 콘텐츠를 전송할 수 있으며 LENA WAS와 AJP 혹은 Reverse Proxy 형태로

연동하여 LENA WAS가 제공하는 Web Application 서비스의 front-end 역할을 수행한다. 그

외

선택적으로 다양한 추가 기능을 이용할 수 있는데, Domain/URI 기반 분기 및 Load Balancing 기능과

보안 레이어(SSL)를 제공하는 것이 대표적이다.

LENA WAS (Web Application Server)

LENA WAS는 Java Web Application을 실행하여 Web Application 서비스를 제공한다. DB

Connection Pool을 이용하기 위한 Datasource 연동 기능과 Session Clustering을 이용하기 위한

Session Server 연동 기능을 포함하고 있다. 다음은 LENA WAS의 두 가지 Type을 설명한다.

- LENA WAS SE (Standard Edition)

Java Class 파일을 처리하기 위한 Servlet Engine과 JSP 파일을 처리하기 위한 JSP Engine으로만

구성되어 있으며 WAR Type의 Web Application만 실행할 수 있다. LENA WAS EE에 비해 상대적으로 가볍고 빠르다는 장점이 있다.

- LENA WAS EE (Enterprise Edition)

LENA WAS SE가 제공하는 Engine 외에도 EJB Engine, JMS Engine 등 J2EE Spec을 지원하며,

WAR/EJB/EAR Type의 Application을 실행할 수 있다. 또한 2 Phase Commit을 위한 XA

Datasource를 지원한다.

## C

LENA WAS는 내부적으로 Advertiser Module을 탑재하고 있으며 이는 LENA WAS의

JVM 내부 모니터링 결과를 JMX를 통해 수집하여 LENA Manager로 전달한다.

LENA Session Server

LENA Session Server는 여러 LENA WAS 간의 Session을 공유하여 WAS 다중화를 통한 고 가용성을

보장하기 위해 제공된다. LENA Enterprise Edition을 사용할 경우 기본으로 제공되며 2개의 LENA

Session Server를 Active-Active Clustering (양방향 Active-Standby)을 통해 Session Clustering을

구성할 수 있다. LENA Session Server를 사용하기 위해 Application에서 별도의 설정을 할 필요

없이 LENA Manager에서 간단한 설정만으로 구현이 가능하다.

☞

단, Session-clustering을 하기 위해선 HTTP Session에 저장되는 모든 객체는

직렬화(Serializable)되어야한다.