

OSGi 안내서

JEUS 9.1

TMAXSOFT

저작권 공지

Copyright 2026. TmaxSoft Co., Ltd. All Rights Reserved.

회사 정보

(주)티맥스소프트

주소 : 경기도 성남시 분당구 정자일로 45, 티맥스소프트타워

기술 서비스 센터: 1544-8629

홈페이지: <https://www.tmaxsoft.com>

제한된 권리

이 소프트웨어(JEUS®) 사용설명서와 프로그램은 저작권법과 국제 조약에 의해 보호됩니다. 사용설명서와 프로그램은 TmaxSoft Co., Ltd.와의 사용권 계약 하에서만 사용할 수 있으며, 사용설명서는 사용권 계약의 범위 내에서만 배포 또는 복제할 수 있습니다. 이 사용설명서의 전부 또는 일부를 TmaxSoft의 사전 서면 동의 없이 전자, 기계, 녹음 등의 수단으로 전송, 복제, 배포하거나 2차적 저작물을 작성할 수 없습니다.

이 소프트웨어 사용설명서와 프로그램의 사용권 계약은 어떠한 경우에도 사용설명서 및 프로그램과 관련된 지적 재산권(등록 여부를 불문)을 양도하는 것으로 해석되지 않으며, 브랜드나 로고, 상표 등을 사용할 권한을 부여하지 않습니다. 사용설명서는 오로지 정보 제공만을 목적으로 하며, 이로 인한 계약상의 직접적 또는 간접적 책임을 지지 않습니다. 또한 사용설명서 상의 내용이 법적 또는 상업적인 특정 조건을 만족시킬 것을 보장하지 않습니다. 사용설명서는 제품의 업그레이드나 수정에 따라 예고 없이 변경될 수 있으며, 내용상의 오류가 없음을 보장하지 않습니다.

상표 공지

JEUS®는 TmaxSoft Co., Ltd.의 등록 상표입니다.

Java, Solaris는 Oracle Corporation 및 그 자회사, 관계회사의 등록 상표입니다.

Microsoft, Windows, Windows NT는 Microsoft Corporation의 등록 상표 또는 상표입니다.

HP-UX는 Hewlett Packard Enterprise Company의 등록 상표입니다.

AIX는 International Business Machines Corporation의 등록 상표입니다.

UNIX는 X/Open Company, Ltd.의 등록 상표입니다.

Linux는 Linus Torvalds의 등록 상표입니다.

Noto는 Google Inc.의 상표입니다. Noto 글꼴은 오픈 소스입니다. 모든 Noto 글꼴은 SIL Open Font License, 버전 1.1에 따라 게시됩니다. (<https://www.google.com/get/noto/>)

기타 모든 제품들과 회사 이름은 각각 해당 소유주의 상호, 상표, 또는 등록 상표입니다.

본 사용설명서에 기재된 회사, 시스템, 제품 이름 등에 반드시 상표 표시(™, ®)를 하지는 않습니다.

오픈 소스 소프트웨어 공지

본 제품의 일부 파일 또는 모듈은 다음의 라이선스를 준수합니다.: APACHE2.0, CDDL1.0, EDL1.0, OPEN SYMPHONY SOFTWARE1.1, TRILEAD-SSH2, Bouncy Castle, BSD, MIT, SIL OPEN FONT1.1

관련 상세 정보는 제품의 다음 디렉터리에 기재된 사항을 참고하시기 바랍니다. :

`${INSTALL_PATH}/license/oss_licenses`

유지 보수

구분	지원항목	서비스 내용
제품지원	패치 & 업그레이드	무상 패치 서비스 제공 메이저 버전 업그레이드 시 할인 혜택 웹 지원을 통한 패치 내역 제공
기술 지원 - 기본 서비스	장애 지원	장애 발생 시 원인 분석 및 조치 Service Desk팀 → 기술팀 → R&D의 3단계 장애 분석 및 조치
	일상 지원(온라인 지원)	E-mail, 전화, 원격, 웹 사이트 등 온라인 자원을 통한 질의 응답 서비스
	고객 맞춤 지원(방문 지원)	고객의 요청으로 수행하는 방문 지원 서비스
기술 지원 - 옵션 서비스	예방 지원	정기 점검을 통한 시스템 운영현황 보고 및 장애 예방 ◦ 관리자 또는 운영자의 요구사항 수렴 ◦ 운영 현황(시스템, 엔진 운영) 보고서 제공 ◦ 필요 시 시스템 개선 권장 사항 보고
유지 보수 비용 및 기간	계약 시 별도 협의	계약 시 EOL/EOS 문서 제공

안내서 이력

제품 버전	안내서 버전	발행일	비고
JEUS 9.1.1	3.4.1	2026-06-08	Fix 릴리스
JEUS 9.1	3.3.1	2025-09-30	GA 버전
JEUS 9 Fix#1	3.2.2	2025-07-10	JDK 21 지원 내용 추가
JEUS 9 Fix#1	3.2.1	2025-06-30	RC 버전
JEUS 9	3.1.2	2025-01-03	-

제품 버전	안내서 버전	발행일	비고
JEUS 9	3.1.1	2024-09-27	-

목차

1. 소개	1
1.1. 개요	1
1.2. OSGi Framework	1
1.3. JEUS OSGi 개요	1
2. OSGi Framework 설정 및 제어	3
2.1. 개요	3
2.2. OSGi Framework 설정 항목	3
2.3. OSGi Framework 설정 추가	5
2.4. OSGi Framework 설정 수정	6
2.5. OSGi Framework 설정 삭제	7
2.6. JNDI를 통해 OSGi Framework에 접근하는 방법	8
3. 번들 제어 및 모니터링	9
3.1. 번들 모니터링	9
3.2. 번들 설치	9
3.3. 번들 시작	10
3.4. 번들 정지	10
3.5. 번들 갱신	10
3.6. 번들 삭제	10
4. 웹 애플리케이션 배포	11

1. 소개

본 장에서는 OSGi 기술에 대해 간략하게 소개한다.

1.1. 개요

OSGi란 OSGi Alliance에서 정의한 기술로, 자바 환경에서 사용할 수 있는 동적 컴포넌트 시스템을 구성하기 위한 일련의 스펙들로 이루어져 있다. OSGi 스펙에 맞게 작성한 애플리케이션은 번들(Bundle)이라 불리는 작고 재사용 가능하며, 동적으로 구성 가능한 컴포넌트들로 이루어진다. 각 컴포넌트들은 OSGi 서비스라 불리는, OSGi 서비스 레지스트리에 의해 관리되는 자바 객체들을 통해 서로 상호 작용을 수행하게 된다.



OSGi 기술에 대한 자세한 내용은 [OSGi 공식 사이트](#)를 참고한다. OSGi 기술을 구성하고 있는 여러 specification에 대한 문서나 API에 대한 Javadoc 등 다양한 정보를 제공하고 있다.

1.2. OSGi Framework

OSGi 프레임워크는 OSGi 스펙에서 핵심이 되는 구성 요소이다. OSGi 프레임워크는 번들(Bundle)이라 불리는 확장성이 있고 다운로드 가능한 애플리케이션 컴포넌트들에 대한 관리 및 배포 등을 안전하게 수행할 수 있도록 하는 기능을 제공한다.

OSGi 프레임워크에서 제공하는 주요 기능들은 다음과 같다.

- 번들(Bundle)의 생명 주기를 관리한다. 번들이란 OSGi 스펙에서 정의한 속성값들을 META-INF/Manifest.MF 파일에 기록한 JAR 파일이며, OSGi 프레임워크가 관리하는 기본 단위이다. OSGi 프레임워크 자체도 번들 형태를 취하고 있으며, OSGi 프레임워크 번들은 시스템 번들(System Bundle)이라고도 한다.
- 각 번들들 사이에 의존관계(dependency)를 관리하고, 필요한 번들들을 연결(wiring)해준다. 각 번들은 필요한 패키지나 자신이 제공할 수 있는 패키지 및 버전 등에 대한 정보를 Manifest.MF에 OSGi 스펙이 규정한 방식대로 기술하고, OSGi 프레임워크는 이 정보들을 이용하여 각 번들 간에 의존 관계를 파악한 후 연결(wiring)한다.
- 번들이 제공하는 서비스를 관리한다. OSGi에서 서비스란 특정 기능을 제공하는, 일반적인 자바 객체를 의미한다. 각 번들들은 OSGi 스펙에서 규정한 방법을 사용하여 OSGi 프레임워크가 관리하는 서비스 레지스트리(Service Registry)에 서비스를 등록하거나, 등록되어 있는 서비스를 가져와 사용할 수 있다.

1.3. JEUS OSGi 개요

JEUS OSGi 모듈은 Jakarta EE 기반 웹 애플리케이션 서버인 JEUS에서 OSGi 프레임워크 인스턴스를 생성하고 사용할 수 있는 환경을 제공한다. JEUS OSGi 모듈이 제공하는 기능을 사용하여 할 수 있는 일들은 다음과 같다.

- 서버 시작 시 설정에 따라 OSGi 프레임워크 인스턴스를 생성하고 기동할 수 있다.
- JNDI를 통해 애플리케이션이 OSGi 프레임워크 번들에 접근할 수 있다.
- OSGi 프레임워크에 대한 상태를 확인하거나, OSGi 번들을 설치, 시작, 정지, 갱신, 제거할 수 있다.

- 웹 애플리케이션에서 OSGi 프레임워크가 관리하는 서비스에 접근할 수 있는 방법을 제공한다.

2. OSGi Framework 설정 및 제어

본 절에서는 JEUS OSGi 모듈을 사용하여 OSGi 프레임워크를 설정하고 사용하는 방법에 대해 설명한다.

2.1. 개요

OSGi 기술이 제공하는 스펙에 따라 작성한, 번들로 이루어진 애플리케이션을 실행하기 위해서는 OSGi 프레임워크가 필요하다. OSGi 프레임워크는 OSGi 기술에서 핵심이 되는 구성 요소로서, 번들들에 대한 배포 및 관리 기능을 수행하고, 번들 사이에 의존 관계를 관리하며, OSGi 서비스 레지스트리(Service Registry)를 통해 서비스를 등록하고 관리할 수 있도록 하는 기능 등을 제공한다.

JEUS OSGi 모듈에서는 사용자가 OSGi 환경을 쉽게 준비할 수 있도록, 서버 설정을 통해 JEUS 서버가 기동할 때 자동으로 OSGi 프레임워크 인스턴스를 생성하여 초기화시키고, 사전에 설정한 번들들을 설치해 주는 기능을 제공한다. 이와 관련한 작업들은 JEUS에서 제공하는 관리 도구를 통해 수행할 수 있다.



JEUS에서는 OSGi 프레임워크 구현체로 Apache Felix를 사용한다. Apache Felix에 대해서는 [Apache Felix 공식 문서](#)를 참고한다.

2.2. OSGi Framework 설정 항목

OSGi Framework에 대한 설정 항목들에 대한 설명은 다음과 같다.



OSGi와 관련하여 JEUS에서 제공하는 설정 항목들은 동적 설정 반영을 지원하지 않는다. 운영 중인 서버에 대해 설정을 변경한 경우에는 서버를 재기동해야 적용된다.

- Name

OSGi 프레임워크에 대한 고유한 이름을 지정한다.

- JNDI Export Name

OSGi 프레임워크 객체를 JNDI에 등록할 때 사용할 JNDI 이름을 지정한다. 지정하지 않는 경우 Name 항목에 지정한 프레임워크 이름을 JNDI 이름으로 사용한다. 사용자 애플리케이션이 OSGi 프레임워크에 직접 접근해야 할 경우, JNDI에 등록되어 있는 프레임워크 객체를 lookup하여 사용할 수 있다. JNDI에 등록한 OSGi Framework 객체는 해당 서버에 대해서만 유효하며, 원격지에서 lookup하여 사용할 수는 없다.

- Target Servers

OSGi 프레임워크를 사용할 서버를 지정한다.

- Framework Configuration Properties

OSGi 프레임워크를 생성하고 초기화할 때 넘겨줄 설정값들을 지정한다. 각 설정값은 키와 값의 쌍으로 구성된 프로퍼티 형태이다. OSGi 스펙에서 사전 정의해 둔 프로퍼티들과, Apache Felix Framework에서 사전 정의해

값들을 사용할 수 있으며, 애플리케이션이 사용할 설정에 대한 프로퍼티를 지정할 수도 있다. OSGi 스펙에서 정의한 프로퍼티들에 대해서는 [OSGi Core Specification](#)을, JEUS에서 사용하는 Apache Felix에서 정의한 프로퍼티들에 대해서는 [Apache Felix Framework Configuration Properties](#)을 참고한다.

사용자가 직접 설정하지 않은 경우, JEUS OSGi 모듈이 기본값으로 설정하는 프로퍼티들이 존재한다. 해당 프로퍼티들에 대한 목록은 다음과 같다.

프로퍼티	값
felix.startlevel.bundle	30
org.osgi.framework.bundle.parent	framework
org.osgi.framework.startlevel.beginning	100
org.osgi.framework.storage	SERVER_HOME/.workspace/osgi/Framework_NAME

• Initial Bundles Directory 관련 설정 항목들

JEUS OSGi 모듈에서는 Initial Bundles Directory를 지정하고, 이에 대한 동작을 제어할 수 있는 설정 항목들을 제공한다. Initial Bundles Directory를 지정하고 사용하도록 설정한 경우, JEUS OSGi 모듈은 프레임워크 초기화를 완료한 후 해당 위치를 검사하여 디렉터리 안에 있는 JAR 파일들을 번들로 간주, 프레임워크에 설치하는 작업을 수행한다.



JEUS에서는 지정한 디렉터리가 포함하고 있는 번들을 프레임워크 기동 시 설치해 주는 기능만 제공하며, 해당 디렉터리에 대한 관리나 동기화 등에 대한 기능은 제공하지 않는다.

초기 번들 디렉터리와 관련하여 설정 가능한 항목들은 다음과 같다.

항목	설명
Install Initial Bundles	OSGi 프레임워크 기동 시 Initial Bundles Directory에 담긴 번들들을 설치할지 여부를 지정한다.
Initial Bundles Directory Location	Initial Bundles Directory에 대한 위치(파일 경로)를 지정한다. 지정하지 않은 경우 기본 경로로 JEUS_HOME/lib/osgi/Framework_NAME을 사용한다.
Default Start Level	Initial Bundles Directory에 위치한 번들들에 대한 start level 기본값을 지정한다.
Fail On Error	번들 설치가 제대로 이루어지지 않았을 경우 이를 오류로 간주할지 여부를 설정한다.

• Bundle Installation Descriptors

Bundle Installation Descriptor XML 파일들에 대한 위치를 설정한다. 상대 경로로 지정할 경우 Initial Bundles Directory 경로를 기준으로 파일 위치를 결정한다.

Bundle Installation Descriptor는 JEUS OSGi 모듈에서 사용하는, 설치할 번들에 대한 정보를 담은 XML 파일이다. 사용자가 설치할 번들들에 대한 정보를 담아 Bundle Installation Descriptor를 작성한 후, OSGi 프레임워크 설정에 XML 파일 위치를 명시하면, JEUS OSGi 모듈은 OSGi 프레임워크 초기화 작업 및 초기 번들 디렉터리에 있는 번들들을 설치하는 작업을 완료한 후, XML 파일을 읽어들이어 해당 파일에 적힌 정보를 바탕으로 번들을 설치하는 작업을 수행한다.

다음은 Bundle Installation Descriptor에 대한 예제이다.

Bundle Installation Descriptor 예제

```
<?xml version="1.0" encoding="UTF-8"?>
<bundles xmlns="http://www.tmaxsoft.com/xml/ns/jeus/osgi/bundles">
  <!-- XML 파일 내에 기재되어 있는 번들들을 설치할 때 적용할 start level 기본값 -->
  <default-start-level>15</default-start-level>
  <!-- 번들 설치에 실패했을 경우 이를 오류로 간주할지 여부 -->
  <fail-on-error>true</fail-on-error>
  <bundle>
    <!-- 번들에 대한 start level 값. default-start-level에 지정한 값에 우선한다. -->
    <start-level>5</start-level>
    <!-- 번들을 설치할지 여부 -->
    <install>true</install>
    <!-- 번들을 설치한 후 시작시킬지 여부 -->
    <start>true</start>
    <!-- 설치할 번들의 URL -->
    <location>http://central.maven.org/maven2/org/apache/logging/log4j/log4j-api/2.8.2/log4j-api-2.8.2.jar</location>
  </bundle>
</bundles>
```

2.3. OSGi Framework 설정 추가

본 장에서는 OSGi 프레임워크에 대한 설정을 추가하는 방법을 설명한다.

콘솔 툴 사용

다음은 콘솔 툴을 사용하여 OSGi 프레임워크에 대한 설정을 생성하는 예제이다.

```
[MASTER]domain1.adminServer>list-osgi-frameworks
No OSGi Frameworks exists.
[MASTER]domain1.adminServer>add-osgi-framework -name framework-0 -servers server1
Successfully performed the ADD operation for OSGi Framework framework-0, but all changes were non-
dynamic. They will be applied after restarting.
Check the results using "list-osgi-frameworks".
[MASTER]domain1.adminServer>list-osgi-frameworks
List of OSGi Frameworks
=====
+-----+-----+-----+
| Framework Name | JNDI Export Name | Target Servers |
+-----+-----+-----+
| framework-0    |                   | server1        |
+-----+-----+-----+
=====
[MASTER]domain1.adminServer>
```



예제에서 사용한 명령어들에 대한 자세한 설명은 JEUS Reference 안내서의 "OSGi 관련 명령어"를 참고한다.

설정이 정상적으로 이루어진 경우, 서버를 재기동하면 다음과 같이 OSGi 프레임워크가 성공적으로 기동되었다는 메시지가 출력된다.

```
...
[2017.10.25 16:06:24][1] [server1-1] [OSGi-0003] OSGi Framework [framework-0] is started
successfully.
...
```

2.4. OSGi Framework 설정 수정

본 장에서는 OSGi 프레임워크에 대한 설정을 수정하는 방법을 설명한다.

콘솔 툴 사용

다음은 콘솔 툴을 사용하여 OSGi 프레임워크에 대한 설정을 수정하는 예제이다.

```
[MASTER]domain1.adminServer>list-osgi-frameworks -framework framework-0
OSGi Framework [framework-0]
=====
+-----+-----+
| Attribute Name | Value |
+-----+-----+
| Framework Name | framework-0 |
| JNDI Export Name | Not Specified |
| Target Servers | server1 |
| Install Initial Bundles | true |
| Initial Bundles Directory Location | Not Specified |
| Default Start Level | 30 |
| Fail On Error | true |
+-----+-----+
=====

Framework Configuration Properties
=====
+-----+-----+
| Key | Value |
+-----+-----+
(No data available)
=====

Bundle Installation Descriptors
=====
+-----+-----+
| Location |
+-----+-----+
(No data available)
=====

[MASTER]domain1.adminServer>modify-osgi-framework -name framework-0 -dsl 45
Successfully performed the MODIFY operation for OSGi Framework framework-0, but some changes were
non-dynamic. They will be applied after restarting.
Check the results using "list-osgi-frameworks".
[MASTER]domain1.adminServer>list-osgi-frameworks -framework framework-0
OSGi Framework [framework-0]
```

```
=====
```

Attribute Name	Value
Framework Name	framework-0
JNDI Export Name	Not Specified
Target Servers	server1
Install Initial Bundles	true
Initial Bundles Directory Location	Not Specified
Default Start Level	45
Fail On Error	true

```
=====
```

Framework Configuration Properties

```
=====
```

Key	Value
-----	-------

(No data available)

```
=====
```

Bundle Installation Descriptors

```
=====
```

Location

(No data available)

```
=====
```

[MASTER]domain1.adminServer>



1. 예제에서 사용한 명령어들에 대한 자세한 설명은 JEUS Reference 안내서의 "OSGi 관련 명령어"를 참고한다.
2. OSGi와 관련하여 JEUS에서 제공하는 설정 항목들은 동적 설정 반영을 지원하지 않는다. 운영 중인 서버에 대해 설정을 변경한 경우에는 서버를 재기동해야 적용된다.

2.5. OSGi Framework 설정 삭제

본 장에서는 OSGi 프레임워크에 대한 설정을 삭제하는 방법을 설명한다.

콘솔 툴 사용

다음은 콘솔 툴을 사용하여 OSGi 프레임워크에 대한 설정을 삭제하는 예제이다.

```
[MASTER]domain1.adminServer>list-osgi-frameworks
```

List of OSGi Frameworks

```
=====
```

Framework Name	JNDI Export Name	Target Servers
----------------	------------------	----------------

```
=====
```

framework-0	server1
<pre>[MASTER]domain1.adminServer>remove-osgi-framework -name framework-0 Successfully performed the REMOVE operation for OSGi Framework framework-0. Check the results using "list-osgi-frameworks". [MASTER]domain1.adminServer>list-osgi-frameworks No OSGi Frameworks exists. [MASTER]domain1.adminServer></pre>	



1. 예제에서 사용한 명령어들에 대한 자세한 설명은 JEUS Reference 안내서의 "OSGi 관련 명령어"를
2. OSGi와 관련하여 JEUS에서 제공하는 설정 항목들은 동적 설정 반영을 지원하지 않는다. 운영 중인 서버에 대해 설정을 변경한 경우에는 서버를 재기동해야 적용된다.

2.6. JNDI를 통해 OSGi Framework에 접근하는 방법

OSGi 프레임워크를 생성하고 나면, JEUS OSGi 모듈은 시스템 번들 컨텍스트(OSGi 프레임워크 번들의 bundle context)를 JEUS JNDI에 등록한다. 등록하는 객체의 타입은 `org.osgi.framework.launch.Framework`이며, JNDI 이름은 설정에 지정한 이름을 사용한다. 별도로 설정하지 않은 경우 OSGi 프레임워크 이름을 JNDI 이름으로 사용한다. 애플리케이션은 필요한 경우 JNDI를 통해 OSGi 프레임워크 객체를 가져와 사용할 수 있다.

아래는 JNDI에 'framework-0'라는 이름을 사용하여 등록되어 있는 OSGi 프레임워크 객체를 가져온 후, 서비스 객체를 얻는 작업을 수행하는 코드 예제이다.

JNDI에 등록되어 있는 OSGi 프레임워크를 통해 서비스 객체에 접근하는 예제

```
...
    InitialContext ctx = new InitialContext();
    BundleContext bundleContext = ((Framework) initialContext.lookup("framework-
0")).getBundleContext();

    ServiceReference sref = bctx.getServiceReference(ServiceA.class);
...
    ServiceA service = (ServiceA)bctx.getService(sref);
...
```

3. 번들 제어 및 모니터링

본 장에서는 콘솔 툴을 사용해서 JEUS OSGi가 관리하는 OSGi 프레임워크에 설치되어 있는 번들들을 제어하고 모니터링하는 방법에 대해 설명한다.



예제에서 사용한 명령어들에 대한 자세한 설명은 JEUS Reference 안내서의 "OSGi 관련 명령어"를 참고한다.

3.1. 번들 모니터링

다음은 콘솔 툴을 사용하여 OSGi 프레임워크에 설치되어 있는 번들 목록을 확인하는 예제이다.

```
[MASTER]domain1.adminServer>show-osgi-framework-info -server server1 -framework framework-0 -lb
OSGi Framework [framework-0] Information in the server[framework-0]
Framework [framework-0] Overview
=====
+-----+-----+-----+-----+-----+-----+
| System Bundle Name | Version |
+-----+-----+-----+-----+-----+
| org.apache.felix.framework | 5.4.0 |
+-----+-----+-----+-----+-----+
=====

Installed Bundles in the framework [framework-0].
=====
+---+---+---+---+---+---+
| Id | Symbolic Name | Version | Location | State | Start-level |
+---+---+---+---+---+---+
| 0 | org.apache.felix.framework | 5.4.0 | System Bundle | ACTIVE | 0 |
+---+---+---+---+---+---+
| 2 | org.ops4j.pax.url.mvn | 1.3.7 | https://repo.maven.apache.org/maven2/org/ops4j/pax/url/pax-url-mvn-1.3.7/pax-url-mvn-1.3.7.jar | ACTIVE | 30 |
+---+---+---+---+---+---+
=====

[MASTER]domain1.adminServer>
```

3.2. 번들 설치

다음은 콘솔 툴을 사용하여 OSGi 프레임워크에 번들을 설치하는 예제이다.

```
[MASTER]domain1.adminServer>install-bundle -location
https://repo.maven.apache.org/maven2/org/ops4j/pax/url/pax-url-mvn-1.3.7/pax-url-mvn-1.3.7.jar
-framework framework-0 -server server1
Installation completed successfully.
```

```
[MASTER]domain1.adminServer>
```

3.3. 번들 시작

다음은 콘솔 툴을 사용하여 OSGi 프레임워크에 설치된 번들을 시작하는 예제이다.

```
[MASTER]domain1.adminServer>start-bundle -name org.ops4j.pax.url.mvn -framework framework-0 -server
server1
The bundle has been started successfully.
[MASTER]domain1.adminServer>
```

3.4. 번들 정지

다음은 콘솔 툴을 사용하여 OSGi 프레임워크에 설치된 번들을 정지하는 예제이다.

```
[MASTER]domain1.adminServer>stop-bundle -name org.ops4j.pax.url.mvn -framework framework-0 -server
server1
The bundle has been stopped successfully.
[MASTER]domain1.adminServer>
```

3.5. 번들 갱신

다음은 콘솔 툴을 사용하여 OSGi 프레임워크에 설치된 번들을 갱신하는 예제이다.

```
[MASTER]domain1.adminServer>update-bundle -name org.ops4j.pax.url.mvn -framework framework-0 -server
server1
The bundle has been updated successfully.
[MASTER]domain1.adminServer>
```

3.6. 번들 삭제

다음은 콘솔 툴을 사용하여 OSGi 프레임워크에 설치된 번들을 삭제하는 예제이다.

```
[MASTER]domain1.adminServer>uninstall-bundle -name org.ops4j.pax.url.mvn -framework framework-0
-server server1
The bundle has been uninstalled successfully.
[MASTER]domain1.adminServer>
```

4. 웹 애플리케이션 배포

본 장에서는 웹 애플리케이션에 OSGi 번들을 포함시켜 배포하는 방법에 대해 설명한다.

JEUS에 배포하는 일반적인 웹 애플리케이션에서 라이브러리로 OSGi 번들을 사용할 경우 해당 애플리케이션에 번들을 같이 패키징하여 deploy할 때 지정한 OSGi 프레임워크에 번들이 설치되게 할 수 있다. 이를 위해서는 애플리케이션 패키징 시 아래와 같은 추가 작업을 해 주어야 한다.

1. 사용할 번들을 애플리케이션 내부에 둔다.
2. jeus-web-dd.xml에 OSGi 관련한 설정을 명시한다.

jeus-web-dd.xml 설정 예제

```
<?xml version="1.0"?>
<jeus-web-dd xmlns="http://www.tmaxsoft.com/xml/ns/jeus">
    ....
    <osgi>
        <!-- Framework Name -->
        <framework-name>fwk-0</framework-name>
        <!-- Initial Bundles Directory 관련 설정 항목들 -->
        <install-initial-bundles>true</install-initial-bundles>
        <initial-bundles-directory-location>WEB-INF/osgi</initial-bundles-directory-location>
        <default-start-level>11</default-start-level>
        <fail-on-error>true</fail-on-error>
        <!-- Class Loading Delegation 관련 설정 항목들 -->
        <class-loading-delegation>
            <application-bundle>
                <bundle-symbolic-name>jeus.osgi.hello-api</bundle-symbolic-name>
                <version>1.0.0.Final</version>
            </application-bundle>
        </class-loading-delegation>
    </osgi>
    ....
</jeus-web-dd>
```

각 설정 항목에 대한 설명은 다음과 같다.

- Framework Name

웹 애플리케이션에서 사용할 OSGi 프레임워크의 이름을 지정한다.

- Initial Bundles Directory 관련 설정 항목들

웹 애플리케이션 내부에 번들을 위치시킨 디렉터리를 JEUS에서는 Initial Bundles Directory라는 용어로 지칭한다. jeus-web-dd.xml 설정 항목들을 통해 해당 디렉터리의 위치와 이에 대한 동작을 제어할 수 있다. Initial Bundles Directory를 지정하고 사용하도록 설정한 경우 웹 애플리케이션을 deploy할 때 지정한 위치를 검사하여 포함되어 있는 번들을 지정한 OSGi 프레임워크에 설정한다. **웹 애플리케이션에 포함되어 있는 번들은 웹 애플리케이션을 deploy 시 지정한 OSGi 프레임워크에 설치되고, undeploy 시 삭제된다.**

초기 번들 디렉터리와 관련하여 설정 가능한 항목들은 다음과 같다.

항목	설명
Install Initial Bundles	웹 애플리케이션 배포 시 Initial Bundles Directory에 담긴 번들들을 설치할지 여부를 지정한다.
Initial Bundles Directory Location	Initial Bundles Directory에 대한 위치를 지정한다.
Default Start Level	Initial Bundles Directory에 위치한 번들들에 대한 start level 기본값을 지정한다.
Fail On Error	번들 설치가 제대로 이루어지지 않았을 경우 이를 오류로 간주할지 여부를 설정한다.

◦ Class Loading Delegation 관련 설정 항목들

웹 애플리케이션에서 OSGi 프레임워크에 설치한 번들에 있는 클래스를 참조하고자 하는 경우 설정한다. 클래스를 참조할 번들에 대한 bundle symbolic name과 version을 지정하면 해당 번들로부터 클래스를 가져와 사용할 수 있다. 참조할 번들은 여러 개를 지정할 수 있으며, 여러 개를 지정한 경우 지정한 순서대로 클래스를 찾으려고 시도한다.



1. JEUS 클래스 로더 계층 구조에서 클래스나 리소스를 찾지 못한 경우에만 OSGi 번들로부터 클래스나 리소스를 찾으려고 시도한다.
2. 웹 애플리케이션에서 참조하고 있는 클래스를 로딩한 번들이 갱신된 경우, 웹 애플리케이션에서 클래스 관련 오류가 발생할 수 있다. 이런 경우 웹 애플리케이션을 redeploy하거나 reload 해 주어야 한다. 이러한 상황을 최소화하기 위해 웹 애플리케이션에서는 변경이 상대적으로 적은 API 번들들을 참조하여 사용하는 것을 권장한다.

OSGi 번들을 포함한 애플리케이션을 배포한 경우 관련 설정 등은 콘솔 명령어를 통해 확인할 수 있다.

콘솔 툴 사용

콘솔 명령어를 사용하여 웹 애플리케이션에 대한 상세 정보를 조회할 경우 OSGi 관련 설정이 되어 있다면 해당 정보가 같이 출력된다.

```
[MASTER]domain1.adminServer>application-info -server server1 -id hello-osgi.war -type WAR
There are no EJBs in this module.
General information about the web module [hello-osgi].
=====
+-----+-----+-----+-----+
| Module Name | Unique Module Name | Context Path | Target Session Cluster |
+-----+-----+-----+-----+
| hello-osgi | hello-osgi          | /hello-osgi  |                          |
+-----+-----+-----+-----+
=====

Servlets
=====
+-----+-----+-----+-----+-----+-----+-----+
| Name      | Class      | State | Count | Attribute | RegType | URLPatterns |
+-----+-----+-----+-----+-----+-----+-----+
```

appServlet	org.springframework.web.servlet.DispatcherServlet	READY	0	SYNC	WEB_XML	/
------------	---	-------	---	------	---------	---

...

...

OSGi Application Information

Attribute Name	Value
Framework Name	framework-0
Install Initial Bundles	true
Initial Bundles Directory Location	/home/user/workspace/jeus/jeus/target/jeus/domain1/osgi/servers/server1/.workspace/deployed/hello-osgi.war/hello-osgi_war___/WEB-INF/osgi
Default Start Level	11
Fail On Error	true

Delegate Bundles

Bundle Symbolic Name	Version
jeus.osgi.hello-api	1.0.0.Final

EJBs

Bean Name	Type	Local Export Name	Remote Export Name
(No data available)			

[MASTER]domain1.adminServer>