

설치 안내서

OpenFrame OSI 7.2

TMAXSOFT

저작권 공지

Copyright 2025. 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(Tmax OpenFrame®) 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

Tmax®와 Tmax OpenFrame®은 TmaxSoft Co., Ltd.의 등록 상표입니다. 본 사용설명서에 기재된 모든 제품과 회사 이름은 각각 해당 소유주의 상표로서 참조용으로만 사용되며 반드시 상표 표시 (™, ®)를 하지는 않습니다.

오픈소스 소프트웨어 공지

본 제품은 "OpenSSL", "RSA Data Security, Inc.", "Apache Foundation" 및 "Jean-loup Gailly와 Mark Adler"에 의해 개발 또는 라이선스된 오픈 소스 소프트웨어를 포함합니다. 관련 상세 정보는 제품의 다음 디렉터리에 기재된 사항을 참고하시기 바랍니다. : \${INSTALL_PATH}/license/oss_licenses

안내서 이력

제품 버전	안내서 버전	발행일	비고
OpenFrame OSI 7.2	3.2.1	2025-08-14	
OpenFrame OSI 7.2	3.1.2	2023-12-29	
OpenFrame OSI 7.2	3.1.1	2023-08-30	

목차

1. 설치 전 준비사항	1
1.1. 개요	1
1.2. 시스템 요구사항	1
1.3. 디스크 용량 확인	1
1.4. OpenFrame 제품군 설치 확인	2
1.5. 설치용 속성 파일	2
1.5.1. OpenFrame 디렉터리 설정	3
1.5.2. 저장 장치 환경설정	3
1.5.3. VTAM 환경설정	4
1.5.4. Default Region 환경설정	4
1.5.5. 기타 환경설정	5
2. 설치 및 제거	7
2.1. 설치	7
2.1.1. 인스톨러를 이용한 자동 설치	7
2.1.2. 스크립트를 이용한 수동 설치	8
2.2. 설치 확인	8
2.2.1. 디렉터리 확인	9
2.2.2. 기동 확인	12
2.3. 제거	15
2.4. 제거 확인	16
3. 문제해결	17
3.1. 개요	17
3.2. 에러 유형 및 해결방법	17
3.2.1. 라이선스 파일	17
3.2.2. 기동서버 상태	18
3.2.3. 보안 모듈	18
Appendix A: 설치용 속성파일 예제	20
Appendix B: 설치 검증	21
B.1. 소개	21
B.2. OSI IVP	21
B.2.1. 환경설정	21
B.2.2. 애플리케이션 등록	22
B.2.3. 검증	25

1. 설치 전 준비사항

본 장에서는 OpenFrame OSI를 설치하기 전에 필요한 준비사항에 대해 기술한다.

1.1. 개요

OpenFrame OSI(이하 OSI)를 설치하기에 앞서 필요한 준비사항은 다음과 같다.

- 시스템 요구사항 확인
- 디스크 용량 확인
- OpenFrame 제품군 설치 확인
- 설치용 속성파일 작성

1.2. 시스템 요구사항

OSI는 UNIX 또는 Linux 시스템 환경에서만 설치할 수 있다. OSI 설치를 위한 시스템 요구사항은 다음과 같다.

	요구사항
Platform	Solaris 11(SunOS 5.11) 이상 (32bit, 64bit)
	Linux x86 2.6 이상 (32bit, 64bit)
	Linux ia64 2.6 이상 (32bit, 64bit)
Hardware	최소 5GB 공간
	최소 8GB 이상 메모리 공간
Database	Tibero 6 FS07
Compiler	MF-COBOL 컴파일러, NET-COBOL 컴파일러, OpenFrame COBOL 컴파일러
	OpenFrame PL/I 컴파일러
	OpenFrame ASM 컴파일러
OpenFrame 제품군	OpenFrame Base 7.1, OpenFrame Batch 7.1, OpenFrame HiDB 7.2

1.3. 디스크 용량 확인

OSI 설치에 앞서 UNIX 시스템에 남아있는 디스크의 여유공간을 확인한다. OSI를 설치하기 위해서는 최소 200MB의 여유공간을 필요로 한다. 하드웨어 요구사항에 대한 자세한 내용은 [시스템 요구사항](#)을 참고한다.

UNIX 시스템에서 하드웨어의 여유공간을 확인하려면 UNIX 프롬프트에서 **df** 명령어를 실행한다.

다음은 HP-UX 운영체제를 사용하는 HP 머신에서 df 명령어를 실행하여 디스크 용량이 얼마나 남아있는지 확인하는 예제이다. 출력 결과를 KB 단위로 표시하기 위해 -k 옵션을 사용했다.

```
$ df -k /home
```

다음은 위의 명령을 실행한 결과이다.

```
/home                (/dev/vg01/lv00) :    30425199 total allocated KB
                    4422401 free allocated KB
                    26002798 used allocated KB
                    85 % allocation used
```

/home 디렉터리는 /dev/vg01/lv00 디스크에 속해 있고, 전체 용량 30,425,199KB 중 사용한 용량은 26,002,798KB, 남은 용량은 4,422,401KB로 전체의 85%가 사용 중이라는 것을 의미한다.

1.4. OpenFrame 제품군 설치 확인

OSI 제품의 경우에는 다른 독립 제품의 영향을 받는다. OSI를 설치 및 사용하기 위해서는 먼저 OpenFrame Base, OpenFrame Batch, OpenFrame HiDB 제품이 순서대로 설치되어 있어야 한다.

다음은 **ofversion** 툴을 사용하여 설치된 OpenFrame 제품군을 확인하는 예제이다.

```
$ ofversion
ofversion version 7.1.0(0) obuild@tplinux64:ofsrc71/base(#1) 2021-03-17 23:07:27
Print OpenFrame Version Information

OpenFrame Solution is Composed of
- OpenFrame Base 7.1 64bit
- OpenFrame Batch MVS 7.1 64bit
- OpenFrame TACF 7.1 64bit
- OpenFrame HiDB 7.2 64bit Tibero
```



OpenFrame 제품군의 설치 및 확인에 대해서는 각 제품의 설치 안내서를 참고한다.

1.5. 설치용 속성 파일

OSI의 설치를 위한 각종 속성 내용을 하나의 파일에 모두 작성한 후 인스톨러를 실행하면 인스톨러는 속성 파일에 작성된 설정 내용을 읽어 들여 설정된 항목에 따라 설치를 진행한다.

본 절에서는 OSI 설치용 속성파일에 작성되어야 하는 항목에 대해 기술한다. 설치용 속성파일의 예제는 본 안내서의 [설치용 속성파일 예제](#)에서 확인할 수 있다.



설치용 속성파일의 이름은 UNIX 시스템에서 허용하는 파일명 내에서 제약없이 작성할 수 있다. 단, 파일 형식은 일반 텍스트 파일 형식이어야 하며, 인스톨러와 함께 제공되는 예제 파일을 참고하여 작성할 것을 권장한다.

1.5.1. OpenFrame 디렉터리 설정

OSI를 설치할 디렉터를 설정하는 항목으로, OpenFrame Base 제품이 설치되어 있는 절대경로를 입력한다.

다음은 OpenFrame의 홈 디렉터를 설정하는 항목이다.

```
OPENFRAME_HOME=<path>
```

항목	설명
OPENFRAME_HOME	UNIX의 디렉터리 또는 파일 경로를 지정한다.

다음은 OpenFrame 홈 디렉터를 /home/oframe/OpenFrame으로 설정하는 예이다.

```
OPENFRAME_HOME=/home/oframe/OpenFrame
```

1.5.2. 저장 장치 환경설정

OpenFrame에서 사용되는 저장 장치, 데이터 셋에 관련된 환경값을 설정한다.

다음은 저장 장치의 환경설정을 위한 항목이다.

```
SYSTEM_TABLESPACE<tablespace>  
DEFAULT_VOLSER=<volume>  
DEFAULT_VOLPATH=<path>  
COMPILER_TYPE<complier>
```

항목	설명
SYSTEM_TABLESPACE	시스템 테이블들을 관리할 테이블스페이스명을 지정한다. 해당 테이블스페이스는 설치전 미리 생성되어 있어야 한다.
DEFAULT_VOLSER	OpenFrame에서 기본 볼륨으로 사용되는 볼륨 이름을 설정한다. 기본 볼륨 이름은 OpenFrame Base를 설치할 때 생성한 볼륨 이름을 사용해야 한다.
DEFAULT_VOLPATH	DEFAULT_VOLSER로 지정한 볼륨의 실제 경로를 지정한다.
COMPILER_TYPE	OSI 설치시점에 컴파일 수행할 OIVP 샘플 소스를 선택한다.

다음은 저장 장치 관련 환경설정의 예이다.

```
SYSTEM_TABLESPACE=OFM_REPOSITORY  
DEFAULT_VOLSER=DEFVOL  
DEFAULT_VOLPATH=$OPENFRAME_HOME/volume_DEFVOL
```

1.5.3. VTAM 환경설정

OSI 시스템에서 연동할 VTAM 포트를 설정한다.

다음은 VTAM 환경설정을 위한 항목이다.

```
VTAM_PORT=<number>
```

항목	설명
VTAM_PORT	VTAM에서 사용할 포트 번호를 10진수로 설정한다.



- 공유 메모리 설정은 16진수로 기술하고 다른 공유 메모리 키 값과 중복되지 않도록 주의한다.
- VTAM_PORT 항목은 OpenFrame GW 환경의 lu.port 값과 동일해야 한다. OpenFrame GW 환경설정 파일($\{OFGW_HOME\}/ofgwconf/ofgw.properties$)을 참고한다.

다음은 VTAM 환경설정의 예이다.

```
VTAM_PORT=8401
```

1.5.4. Default Region 환경설정

인스톨러를 실행하는 경우 Default Region(IMSID : IMSA)을 생성하는데 해당 Region의 환경을 설정한다.

다음은 DefaultRegion 환경설정을 위한 항목이다.

```
IMSA_JCLLIB=<pds_name>
IMSA_MODBLKS=<dataset_name>
IMSA_IMSACB=<dataset_name>
IMSA_DFSRELB=<dataset_name>
IMSA_FORMAT=<dataset_name>
IMSA_STEPLIB=<dataset_name>
IMSA_MQLIB=<datase_namet>
```

항목	설명
IMSA_JCLLIB	OSI에서 사용되는 각종 JCL을 저장하는 PDS 이름을 지정한다.

항목	설명
IMSA_MODBLKS	OSI에서 지원하는 각종 리소스의 설정이 저장된 Staging Library 데이터 셋의 이름을 설정한다.
IMSA_IMSACB	ACB가 저장된 Staging Library 데이터 셋의 이름을 설정한다. IMSA_IMSACB 항목은 OpenFrame 환경설정에 ims 서브젝트, IMS_DEFAULT 섹션에 ACBLIB_NAME 키의 VALUE 항목에 지정되어 있는 값으로 지정한다.
IMSA_DFSRELB	데이터 셋의 동적 사용에 필요한 MDA 정보를 저장하는 라이브러리로 해당 데이터 셋의 이름을 지정한다. IMSA_DFSRELB 항목은 OpenFrame 환경설정에 ims 서브젝트, IMS_DEFAULT 섹션에 RESLIB_NAME 키의 VALUE 항목에 지정되어 있는 값으로 지정한다.
IMSA_FORMAT	MFS가 저장된 Staging Library 데이터 셋의 이름을 설정한다.
IMSA_STEPLIB	애플리케이션 개발자가 작성한 프로그램의 Shared Object 형태 실행 모듈을 보관하는 데이터 셋 이름을 지정한다.
IMSA_MQLIB	OSI에서 사용하는 MQ(Message Queue)의 데이터 셋의 이름을 지정한다.



OpenFrame 환경설정에 대한 자세한 내용은 OpenFrame OSI "환경설정 안내서"를 참고한다.

다음은 Default Region 환경설정의 예이다.

```
IMSA_JCLLIB=SYS1.JCLLIB
IMSA_MODBLKS=OSI.IMSA.DEFLIB
IMSA_IMSACB=IMS.ACBLIB
IMSA_DFSRELB=IMS.RESLIB
IMSA_FORMAT=OSI.IMSA.MFSLIB
IMSA_STEPLIB=OSI.IMSA.STEPLIB
IMSA_MQLIB=OSI.IMSA.MQLIB
```

1.5.5. 기타 환경설정

OSI의 라이선스 경로를 설정한다.

다음은 환경설정을 위한 항목이다.

```
OSI_LICENSE_PATH=<path>
```

항목	설명
OSI_LICENSE_PATH	OSI에서 사용하는 라이선스 파일의 사용자가 위치시킨 경로를 설정한다. OSI 설치과정에서 <code>\${OPENFRAME_HOME}/license</code> 디렉터리로 복사된다.

다음은 환경설정의 예이다.

```
OSI_LICENSE_PATH=/home/oframe1/license
```

2. 설치 및 제거

본 장에서는 OpenFrame OSI 제품의 설치 및 설치 검증, 설치 제거 방법에 대하여 기술한다.

2.1. 설치

본 절에서는 OSI 인스톨러를 이용해서 자동 설치하는 방법과 스크립트를 이용한 수동 설치하는 방법에 대해서 설명한다.

2.1.1. 인스톨러를 이용한 자동 설치

[설치용 속성 파일](#)에서 설명한 설치용 속성 파일을 작성했다면, 인스톨러를 이용하여 간단하게 OSI 제품을 설치할 수 있다.

다음의 설치방법은 모든 플랫폼에서 동일하게 적용된다.

1. OSI를 설치할 시스템의 원하는 위치에 인스톨러 바이너리 파일을 복사한다.
2. 인스톨러 바이너리에 실행권한이 주어지지 않은 경우에는 다음과 같이 **chmod** 명령어를 사용하여 실행권한을 부여한다.

다음은 Linux x86 64 Bit용 인스톨러 바이너리에 실행권한을 부여하는 예이다.

```
$ chmod a+x OpenFrame_OSI7_2_Linux_x86_64.bin
```

3. 콘솔 프롬프트에서 해당 인스톨러 바이너리가 있는 디렉터리에서 다음과 같이 수행한다.

```
<installer_file_name> -f <configuration_file_name>
```

다음은 설치용 속성 파일을 **osi.properties**로 작성해서 실행하는 방법이다. 다음 2가지 방법 중 하나를 선택하고 <Enter> 키를 누른다.

- 방법 1

```
$ ./OpenFrame_OSI7_2_Linux_x86_64.bin -f osi.properties
```

- 방법 2

```
$ sh OpenFrame_OSI7_2_Linux_x86_64.bin -f osi.properties
```

4. 다음과 같은 화면이 나타나고 자동으로 OSI의 설치를 진행한다.

```
Preparing to install...
Extracting the JRE from the installer archive...
Unpacking the JRE...
Extracting the installation resources from the installer archive...
Configuring the installer for this system's environment...
```

2.1.2. 스크립트를 이용한 수동 설치

설치 스크립트 파일들은 `${OPENFRAME_HOME}/scripts` 디렉터리에 위치하며, 현재 인스톨러에 의해 다음 순서대로 스크립트가 자동 실행된다. 설치 과정에서 스크립트 동작이 외부 환경요인으로 인해 일부 비정상 동작했을 경우 OpenFrame 설치 완료 후 수동으로 스크립트를 실행해야 한다. **반드시 다음 순서에 맞춰 스크립트를 실행해야 한다.**

1. link_lib_osi.sh

OSI에서 제공하는 라이브러리들의 symbolic link 작업을 수행한다.

2. osi_ofconfig.sh

openframe_osi.conf, openframe_osi_IMSA.conf 환경 파일을 import 한다.

3. osi_init.sh

OSI에서 사용하는 DB 테이블을 생성한다.

4. osi_mv_copybook.sh

IMSA Region에서 사용하는 데이터 셋을 생성할 때 사용하는 copybook을 OpenFrame 환경설정에 **ds** 서브젝트, **TSAM_RDB** 섹션의 **COPYBOOK_PATH** 키에 설정되어 있는 위치로 이동시킨다. OpenFrame 환경설정에 대한 자세한 내용은 OpenFrame OSI "환경설정 안내서"를 참고한다.

5. osi_define.sh

OSI 시스템에 필요한 맵과 데이터셋들을 생성하고 리소스를 등록한다.

6. osi_ivp.sh

OIVP 테스트에 필요한 맵과 COBOL 소스를 컴파일하고 리소스를 등록한다.



인스톨러에 의해 실행되는 스크립트들의 수행결과는 `${OPENFRAME_HOME}/UninstallerData/log/install_OSI.log`에 저장된다.

2.2. 설치 확인

OSI를 설치한 후에는 다음의 과정을 실행하여 제품이 정상적으로 설치되었는지 확인할 것을 권장한다.

- [디렉터리 확인](#)

- 기동 확인

2.2.1. 디렉터리 확인

OSI 제품을 설치한 후 다음 디렉터리를 확인한다. OSI의 경우 OpenFrame Base, OpenFrame Batch, OpenFrame HiDB 가 설치된 이후에만 설치가 가능하므로 해당 제품이 설치된 후 생성된 디렉터리를 기준으로 설명한다.

```

${OPENFRAME_HOME}
+---- UninstallerData
+---- bin
+---- config
+---- core
+---- cpm
+---- data
+---- include
+---- lib
+---- license
|---- log
    +---- cmd
    +---- data
    +---- sys
+---- ofdb_scripts
|---- osi
    +---- oivp
    +---- resource
+---- outputq
+---- profile
+---- sample
+---- schema
+---- scripts
+---- shared
+---- spbackup
+---- spool
+---- spunpack
+---- temp
+---- tsam
+---- util
+---- volume_DEFVOL
+---- webde
```

UninstallerData

시스템에 설치되어 있는 OpenFrame 독립 제품을 각각 삭제하기 위한 파일들이 위치하는 디렉터리이다.

bin

OSI에서 사용하는 각종 실행 파일들이 위치하는 디렉터리이다.

config

OSI에서 사용하는 환경설정 파일들이 위치하는 디렉터리이다. 환경설정에 대한 자세한 설명은 OpenFrame OSI "환경설정 안내서"를 참고한다.

core

OpenFrame에서 사용하는 Tmax 관련 파일들이 위치한다. OpenFrame을 구성하는 Tmax 서버 프로그램, 서비스 프로그램 및 Tmax 설정 파일을 포함한다.

cpm

문자코드 변환 테이블 파일들이 위치한다.

data

에러코드 테이블 및 Lock 서버에서 사용하는 데이터 파일들이 위치한다.

include

업무 애플리케이션을 컴파일할 때 사용될 헤더 파일들이 위치한다.

lib

OSI에서 사용하는 라이브러리 파일들이 위치하는 디렉터리이다.

license

OpenFrame의 독립 제품이 사용하는 라이선스 파일들이 위치한다. 다음은 OpenFrame의 라이선스 파일의 종류이다.

파일명	설명
licbase.dat	Base 라이선스 파일
lictjes.dat	Batch(TJES) 라이선스 파일
lictacf.dat	TACF 라이선스 파일
lichidb.dat	HiDB 라이선스 파일
licosc.dat (MVS)	OSC 라이선스 파일
licosi.dat (MVS)	OSI 라이선스 파일
licaim.dat (AIM)	AIM 라이선스 파일

log

OpenFrame의 독립 제품이 생성하는 로그 파일들이 위치한다. log의 하위 디렉터리로 로그 파일의 종류에 따라 cmd, sys, data가 있다. 자세한 내용은 OpenFrame Base "Base 안내서"의 "로그 관리"를 참고한다.

디렉터리명	설명
cmd	OpenFrame의 조작 로그가 위치한다.
sys	OpenFrame의 시스템 로그가 위치한다.
data	OpenFrame의 데이터 로그가 위치한다.

ofdb_scripts

TSAM / NDB / HiDB 사용에 필요한 템플릿과 스크립트가 위치한다.

osi

다음은 하위 디렉터리에 대한 설명이다.

하위 디렉터리	설명
oivp	OSI의 IVP 샘플 애플리케이션 파일들이 위치하는 디렉터리이다.
resource	OSI의 시스템 설정 파일, XA 관련 설정 및 JCL 샘플 등이 위치하는 디렉터리이다.

outputq

실제 프린트되기 위한 출력용 데이터가 저장되는 디렉터리이다. 멀티노드로 구성하는 경우에는 shared 디렉터리의 하위 디렉터리로 구성하는 것을 권장한다.

profile

OpenFrame의 사용자별 프로파일이 위치한다.

sample

OpenFrame 제품의 예제 파일들이 위치한다.

schema

cobgensch, pligensch 틀로 생성한 데이터셋 스키마 파일이 저장되는 디렉터리이다.

scripts

OSI를 설치, 삭제 할 때 필요한 스크립트 파일들이 위치하는 디렉터리이다.

shared

OpenFrame 독립 제품이 모두 공유하는 공통 디렉터리이다.

spbakup

Spool 데이터 셋 백업용 디렉터리이다.

spool

Spool 데이터 셋이 저장되는 디렉터리이다. 멀티노드로 구성하는 경우에는 shared 디렉터리의 하위 디렉터리로 구성하는 것을 권장한다.

spunpack

백업된 Spool 데이터를 restore하는 디렉터리이다.

temp

OpenFrame에서 임시작업 파일이 위치한다.

tsam

OpenFrame 에서 사용하는 TSAM 관련 정보를 관리하는 디렉터리이다.

util

OSI 시스템에서 사용하는 유틸리티가 위치한다.

volume_DEFVOL

OpenFrame의 여러 제품에서 기본 볼륨으로 사용하는 디렉터리이다.

webde

OpenFrame에서 사용하는 WEBDE 관련 정보를 관리하는 디렉터리이다.

2.2.2. 기동 확인

다음은 OSI의 기동을 확인하는 순서이다.

1. 설치용 속성 파일에서 스크립트 자동생성 옵션을 모두 YES로 선택했다면

`{OPENFRAME_HOME}/UninstallerData/log/install_OSI.log` 파일을 텍스트 데이터로 직접 열어 오류 로그가 없는지 확인한다.

```
Add oframe_osi.m
Run link_lib_osi.sh
#### osi_ofconfig.sh START ####
boot and tacflogin
Logged-in as [ROOT]

TMBOOT for node(NODE1) is starting:
(E) BOOT3005 Tmax is already running [BOOT0008]
Dataset lock flag disable (Only during OSI installation)
COMPLETED SUCCESSFULLY.
COMPLETED SUCCESSFULLY.
create error code & config
COMPLETED SUCCESSFULLY!
COMPLETED SUCCESSFULLY.
COMPLETED SUCCESSFULLY.
#### osi_ofconfig.sh END ####

#### osi_init.sh START ####
osiinit version 7.2.0(4) obuild@tplinux32:ofsrc7/osi(#1) 2008-02-21 20:10:57
Initialize OpenFrame OSI System Tables

Creating OFM_OSI_MQ_MASTER...
> "OFM_OSI_MQ_MASTER" created...
Creating OFM_OSI_MQ...
> "OFM_OSI_MQ" created...
Creating OFM_OSI_MQ_SEQ...
> "OFM_OSI_MQ_SEQ" created...
Creating OFM_OSI_MQ_BMP...
> "OFM_OSI_MQ_BMP" created...
Creating OFM_OSI_MQ_BMP_SEQ...
> "OFM_OSI_MQ_BMP_SEQ" created...
Creating OFM_OSI_MQ_TRIGGER...
> "OFM_OSI_MQ_TRIGGER" created...
Creating OFM_OSI_REGION...
> "OFM_OSI_REGION" created...
Creating OFM_OSI_SD_DATABASE...
> "OFM_OSI_SD_DATABASE" created...
Creating OFM_OSI_SD_APPLCTN...
> "OFM_OSI_SD_APPLCTN" created...
Creating OFM_OSI_SD_TRANSACT...
> "OFM_OSI_SD_TRANSACT" created...
Creating OFM_OSI_SD_TRANSACT_IDX1...
> "OFM_OSI_SD_TRANSACT_IDX1" created...
Creating OFM_OSI_SD_TERMINAL...
> "OFM_OSI_SD_TERMINAL" created...
Creating OFM_OSI_SD_LTERM...
> "OFM_OSI_SD_LTERM" created...
Creating OFM_OSI_SD_LTERM_IDX1...
> "OFM_OSI_SD_LTERM_IDX1" created...
```

```

Creating OFM_OSI_RTSD_DATABASE...
> "OFM_OSI_RTSD_DATABASE" created...
Creating OFM_OSI_RTSD_APPLCTN...
> "OFM_OSI_RTSD_APPLCTN" created...
Creating OFM_OSI_RTSD_TRANSACT...
> "OFM_OSI_RTSD_TRANSACT" created...
Creating OFM_OSI_RTSD_TRANSACT_IDX1...
> "OFM_OSI_RTSD_TRANSACT_IDX1" created...
Creating OFM_OSI_RTSD_TERMINAL...
> "OFM_OSI_RTSD_TERMINAL" created...
Creating OFM_OSI_RTSD_LTERM...
> "OFM_OSI_RTSD_LTERM" created...
Creating OFM_OSI_RTSD_LTERM_IDX1...
> "OFM_OSI_RTSD_LTERM_IDX1" created...
Creating OFM_OSI_MODSTAT...
> "OFM_OSI_MODSTAT" created...
Creating OFM_OSI_CI...
> "OFM_OSI_CI" created...

#### osi_init.sh END ####

...

acbgen version 7.2.0(4) obuild@tplinux32:ofsrc7/ims(#2) 2007-11-10 10:43:12
Application Control Block Generation Program

ACBGEN COMMAND=BUILD,OPERAND=(PSB=ALL),ACBLIB=IMS.ACBLIB
-----
*** ACBGEN DELETE PSB=ALL
-----
*** ACBGEN DELETE PSB=ALL SUCCESS
-----
-----
*** ACBGEN BUILD PSB=ALL
-----
*** ACBGEN BUILD PSB=ALL SUCCESS
-----
PROGRAM COMPLETED SUCCESSFULLY.
acbgen version 7.2.0(4) obuild@tplinux32:ofsrc7/ims(#2) 2007-11-10 10:43:12
Application Control Block Generation Program

ACBGEN COMMAND=COPY,OPERAND=(),ACBLIB=IMS.ACBLIB
ACBGEN COPY SUCCESS
-----
*** ACBGEN DELETE PSB=ALL
-----
*** ACBGEN DELETE PSB=ALL SUCCESS
-----
-----
*** ACBGEN COPY PSB=ALL
-----
-----
PROGRAM COMPLETED SUCCESSFULLY.
Dataset lock flag enable
COMPLETED SUCCESSFULLY.
COMPLETED SUCCESSFULLY.

```

2. `${OPENFRAME_HOME}/license`에 OSI 라이선스가 들어있는지 확인한다.

3. osiboot를 실행하여 OpenFrame을 기동한다.

```
$ osiboot
```

4. 다음과 같이 기동 화면이 오류없이 출력되는 것을 확인한다.

```
TMBOOT for node(NODE1) is starting:
  TMBOOT: TMM is starting: Thu Apr  8 18:16:12 2021
  TMBOOT: CLL is starting: Thu Apr  8 18:16:12 2021
  TMBOOT: CLH is starting: Thu Apr  8 18:16:12 2021
  TMBOOT: TLM(tlm) is starting: Thu Apr  8 18:16:12 2021
[2021-04-08T18:16:12.437356] [osiboot(9402)          ] [M] [OSI7061M] System server(TPFMAGENT)
booting ok
[2021-04-08T18:16:12.440478] [osiboot(9402)          ] [M] [OSI7061M] System server(ofrsasvr)
booting ok
[2021-04-08T18:16:12.443474] [osiboot(9402)          ] [M] [OSI7061M] System server(ofrlhsrv)
booting ok
[2021-04-08T18:16:12.446754] [osiboot(9402)          ] [M] [OSI7061M] System server(ofrdmsvr)
booting ok
[2021-04-08T18:16:12.451046] [osiboot(9402)          ] [M] [OSI7061M] System server(ofrdsedt)
booting ok
[2021-04-08T18:16:12.455364] [osiboot(9402)          ] [M] [OSI7061M] System server(ofrcmsvr)
booting ok
[2021-04-08T18:16:12.459852] [osiboot(9402)          ] [M] [OSI7061M] System server(ofruisvr)
booting ok
[2021-04-08T18:16:12.464157] [osiboot(9402)          ] [M] [OSI7061M] System server(ofrsmlog)
booting ok
[2021-04-08T18:16:12.471293] [osiboot(9402)          ] [M] [OSI7061M] System server(obmjmsvr)
booting ok
[2021-04-08T18:16:12.475912] [osiboot(9402)          ] [M] [OSI7061M] System server(obmjstd)
booting ok
[2021-04-08T18:16:12.480158] [osiboot(9402)          ] [M] [OSI7061M] System server(obmjinit)
booting ok
[2021-04-08T18:16:12.490924] [osiboot(9402)          ] [M] [OSI7061M] System server(obmjhist)
booting ok
[2021-04-08T18:16:12.500370] [osiboot(9402)          ] [M] [OSI7061M] System server(obmjspb)
booting ok
[2021-04-08T18:16:12.510117] [osiboot(9402)          ] [M] [OSI7061M] System server(ofrpsvr)
booting ok
[2021-04-08T18:16:12.517094] [osiboot(9402)          ] [M] [OSI7061M] System server(obmtmgr)
booting ok
[2021-04-08T18:16:12.521643] [osiboot(9402)          ] [M] [OSI7061M] System server(obmjtimr)
booting ok
[2021-04-08T18:16:12.527060] [osiboot(9402)          ] [M] [OSI7061M] System server(tmsvr)
booting ok
[2021-04-08T18:16:12.533321] [osiboot(9402)          ] [M] [OSI7061M] System server(osiomsvr)
booting ok
[2021-04-08T18:16:12.533343] [osiboot(9402)          ] [M] [OSI7051M] Booting process complete.
```

5. **tmadmin**을 실행한 후 **si** 명령어로 서버목록을 출력하여 모든 서버 프로세스가 정상적으로 기동되었는지 확인한다.

아래 예와 같이 **osi.ofsys.seq**에 지정된 서버들과 **osiomsvr** 서버까지 **RDY** 상태로 표시되면 제품이 정상적으로 설치되어 기동되고 있다는 것을 의미한다. 그 이외의 서버들은 Control Region을 기동하거나, BMP 작업이

수행 중일 때 자동으로 기동된다.

```
$ tadmin
--- Welcome to Tmax Admin (Type "quit" to leave) ---

$$1 NODE1 (tmadm): si
-----
  clh  svrname  (svri)  status  count  qcount  qpcount  emcount
-----
  0    TPFMAGENT ( 4)    RDY     0       0       0       0
  0    ofrsasvr  ( 5)    RDY     0       0       0       0
  0    ofrlhsvr  ( 6)    RDY     0       0       0       0
  0    ofrdmsvr  ( 7)    RDY     0       0       0       0
  0    ofrdsedt  ( 8)    RDY     0       0       0       0
  0    ofrcmsvr  ( 9)    RDY     0       0       0       0
  0    ofruisvr  (10)    RDY     0       0       0       0
  0    ofrsmlog  (11)    RDY     0       0       0       0
  0    obmjmsvr  (12)    RDY     0       0       0       0
  0    obmjjschd (13)    RDY     1       0       0       0
  0    obmjinit  (14)    RDY     2       0       0       0
  0    obmjhist  (15)    RDY     0       0       0       0
  0    obmjspbk  (16)    RDY     0       0       0       0
  0    ofrpmsvr  (17)    RDY     0       0       0       0
  0    obmtsmgr  (18)    RDY     0       0       0       0
  0    obmjtimr  (19)    RDY     0       0       0       0
  0    tmsvr     (20)    RDY     0       0       0       0
  0    osiomsvr  (21)    RDY     0       0       0       0
  0    osissehd  (22)    NRDY    0       0       0       0
  0    osicmdsv  (23)    NRDY    0       0       0       0
  0    OSIMPPSVR (24)    NRDY    0       0       0       0
  0    osibmpsv  (25)    NRDY    0       0       0       0
  0    IMSASCHD  (26)    NRDY    0       0       0       0
  0    IMSACMMD  (27)    NRDY    0       0       0       0
  0    IMSAMPP_TCL1 ( 28)  NRDY    0       0       0       0
  0    IMSAMPP_TCL2 ( 29)  NRDY    0       0       0       0
  0    IMSAMPP_TCL3 ( 30)  NRDY    0       0       0       0
  0    IMSAMPP_TCL4 ( 31)  NRDY    0       0       0       0
```

2.3. 제거

OSI는 인스톨러에서 제공하는 제거 유틸리티를 실행해서 제거할 수 있다.

다음은 OSI를 제거하는 과정이다.

1. 설치된 제품을 제거하기 전에 먼저 Region 서버(Control Region, 사용자 서버)가 기동되어 있는 경우 **imscmd**를 이용하여 Region을 종료한다.

```
$ imscmd IMSA /CHE FREEZE
```

2. Region 서버를 종료하고, **osidown** 명령을 실행하여 OpenFrame을 종료한다.

```
$ osidown
```

- 다음과 같이 오류없이 종료 메시지가 출력되는지 확인한다.

```
20180908:182629 I OSI7141I [20: OSIDOWN: 0: 13817: 67] System server(OSIGW001) shutdown ok
20180908:182629 I OSI7141I [20: OSIDOWN: 0: 13817: 67] System server(tmsvr) shutdown ok
20180908:182629 I OSI7141I [20: OSIDOWN: 0: 13817: 67] System server(obmjtimr) shutdown ok
20180908:182629 I OSI7141I [20: OSIDOWN: 0: 13817: 67] System server(obmtsmgr) shutdown ok
20180908:182629 I OSI7141I [20: OSIDOWN: 0: 13817: 67] System server(obmjspb) shutdown ok
20180908:182629 I OSI7141I [20: OSIDOWN: 0: 13817: 67] System server(obmjhist) shutdown ok
20180908:182629 I OSI7141I [20: OSIDOWN: 0: 13817: 67] System server(obmjshd) shutdown ok
20180908:182629 I OSI7141I [20: OSIDOWN: 0: 13817: 67] System server(obmjinit) shutdown ok
20180908:182629 I OSI7141I [20: OSIDOWN: 0: 13817: 67] System server(obmjmsvr) shutdown ok
20180908:182629 I OSI7141I [20: OSIDOWN: 0: 13817: 67] System server(ofrpsvr) shutdown ok
20180908:182629 I OSI7141I [20: OSIDOWN: 0: 13817: 67] System server(ofrsmlog) shutdown ok
20180908:182629 I OSI7141I [20: OSIDOWN: 0: 13817: 67] System server(ofruisvr) shutdown ok
20180908:182629 I OSI7141I [20: OSIDOWN: 0: 13817: 67] System server(ofrcmsvr) shutdown ok
20180908:182629 I OSI7141I [20: OSIDOWN: 0: 13817: 67] System server(ofrdsedt) shutdown ok
20180908:182629 I OSI7141I [20: OSIDOWN: 0: 13817: 67] System server(ofrdmsvr) shutdown ok
20180908:182629 I OSI7141I [20: OSIDOWN: 0: 13817: 67] System server(ofrlhsvr) shutdown ok
20180908:182629 I OSI7141I [20: OSIDOWN: 0: 13817: 67] System server(ofrsasvr) shutdown ok
Do you really want to down whole Tmax? (y : n):
TMDOWN for node(NODE1) is starting:
TMDOWN: CLL downed: Sat Sep 8 18:26:29 2018
TMDOWN: TLM downed: Sat Sep 8 18:26:29 2018
TMDOWN: CLH downed: Sat Sep 8 18:26:29 2018
TMDOWN: TMM downed: Sat Sep 8 18:26:29 2018
TMDOWN: TMAX is down
```

- OpenFrame이 종료된 것을 확인한 후 UNIX 콘솔에서 `#{OPENFRAME_HOME}/UninstallerData` 디렉터리에 있는 `Uninstall_Osi` 파일을 실행한다.

```
$ ./Uninstall_Osi
```

- 설치 제거 프로그램에 의해 OSI 설치 시 생성된 파일들이 모두 삭제되고, 해당 환경의 Tmax 설정에서 OSI가 모두 제거한다.

2.4. 제거 확인

설치 제거 프로그램이 OSI를 설치할 때 생성한 파일에 대해서만 삭제하므로 불필요하게 남아 있는 디렉터리 및 파일들은 수동으로 삭제해야 한다.

3. 문제해결

본 장에서는 OpenFrame OSI를 설치할 때 발생할 수 있는 에러 중 자주 발생하는 유형을 소개하고, 유형별 해결방법을 기술한다.

3.1. 개요

OpenFrame OSI 설치 과정에서 발생하는 에러는 다음의 3가지 경로를 통해 발견할 수 있다.

- `$OPENFRAME_HOME/UninstallerData/log/install_OSI.log` 파일을 텍스트 에디터로 직접 열어 확인한다.
- OpenFrame을 설치한 후 수동으로 스크립트를 실행할 때 기록되는 에러를 확인한다.
- OpenFrame 기동을 확인할 때 나타나는 서버 상태 로그 정보를 확인한다.

3.2. 에러 유형 및 해결방법

다음은 OpenFrame OSI 설치 중 발생하는 에러 및 유형별 해결방법이다.

3.2.1. 라이선스 파일

라이선스 파일이 존재하지 않거나 잘못된 경우 또는 라이선스 기간이 만료된 경우 다음과 같은 에러가 발생한다.

- 유형

OpenFrame OSI 라이선스가 없거나 잘못된 경우 리전을 기동할 때 MPP 서버 로그에 다음과 같은 메시지가 발생한다.

```
[2021-04-08T18:36:08.996310] [IMSAMPP_TCL2(12094) ] [U] [CHK0001]
/home/oframe6/OpenFrame/license/licosi.dat file open fail: 2
[2021-04-08T18:36:08.996327] [IMSAMPP_TCL2(12094) ] [E] [OSI2051E] server initialization
failed - tlic_check_license error(-28010)
[2021-04-08T18:36:08.996334] [IMSAMPP_TCL2(12094) ] [E] [OSI0161E] osi_res_ds_svrinit:
osi_res_svrinit error(-28010)
[2021-04-08T18:36:08.996340] [IMSAMPP_TCL2(12094) ] [E] [OSI0161E] osi_res_mpp_svrinit:
osi_res_ds_svrinit error(-28010)
12094.183608:(I) SVR3016 SVR IMSAMPP_TCL2 (41/140) tpsvrinit fail. [SVR0031]
```

- 해결방법

라이선스 관련 문의는 TmaxSoft의 기술 지원에 문의한다.

3.2.2. 기동서버 상태

osiomsvr가 정상기동 상태가 아닐 때 사용자 리전 서버 기동에 실패하는 경우의 설명이다.

- 유형

osiomsvr가 NRDY 상태일 때 컨트롤 리전을 기동할 때 해당 SPOOL 로그에 아래와 같은 메시지가 발생한다.

```
[2021-04-08T18:56:18.129700] [DFSMVRC0(14672) ] [U] [MVRC0901M] DFSMVRC0 PROGRAM START
[2021-04-08T18:56:18.129726] [DFSMVRC0(14672) ] [U] [MVRC0903M] INITIALIZE CTL REGION
[2021-04-08T18:56:18.130087] [DFSMVRC0(14672) ] [U] [MVRC0402E] The tpcall(OSIOMSVRBOOT)
TMAX call failed - error=no service or server ready(TPENOREADY)
[2021-04-08T18:56:18.130100] [DFSMVRC0(14672) ] [U] [MVRC0301E] execute_ctlmain() failed
- IMSID=IMSA,RC=-1
```

- 해결방법

tadmin으로 osiomsvr 기동 상태를 확인한다. RDY 상태가 아니라면 아래 명령어 중 하나를 이용하여 서버를 기동시킨다.

```
> tmbot -s osiomsvr
> osiboot -o
```

3.2.3. 보안 모듈

다음은 보안 모듈에서 라이브러리 접근을 차단하는 경우의 설명이다.

- 유형

Linux의 보안 모듈인 SELinux를 사용하는 일부 Linux 시스템의 경우, SELinux의 보안 정책상 일부 라이브러리의 접근을 차단하여 다음과 같은 에러가 발생할 수 있다.

```
“cannot restore segment prot after reloc: Permission denied”
```

- 해결방법

다음 2가지 방법을 사용하여 문제를 해결할 수 있다.

- Permission denied가 발생하는 라이브러리를 대상으로 다음과 같이 chcon을 실행한다. 라이브러리에 따라서는 슈퍼 유저 권한이 필요할 수도 있다.

```
chcon -t texrel_shlib_t [원하는 so 라이브러리]
```

- 슈퍼 유저로 접속한 다음 아래와 같이 /etc/sysconfig/selinux 파일을 수정하여 SELinux를 비활성화한다. 단, 이 방법은 보안 정책을 약화시킬 수 있으므로 권장하지는 않는다.

SELINUX=disabled

Appendix A: 설치용 속성파일 예제

다음은 [설치용 속성 파일](#)에서 설명한 설치 스크립트의 한 예를 보여준다. OpenFrame/OSI 인스톨러를 실행하면 인스톨러는 속성 파일에 따라 설치를 한다.

```
# OPENFRAME GENERAL SECTION
OPENFRAME_HOME=/home/oframe6/OpenFrame

# STORAGE SECTION
SYSTEM_TABLESPACE=OFM_REPOSITORY
DEFAULT_VOLSER=DEFVOL
DEFAULT_VOLPATH=$OPENFRAME_HOME/volume_DEFVOL
COMPILER_TYPE=OFCOBOL

# VTAM SECTION
VTAM_PORT=8401

# DEFAULT REGION SECTION
IMSA_JCLLIB=SYS1.JCLLIB
IMSA_MODBLKS=OSI.IMSA.DEFLIB
IMSA_IMSACB=IMS.ACBLIB
IMSA_DFSRELB=IMS.RESLIB
IMSA_FORMAT=OSI.IMSA.MFSLIB
IMSA_STEPLIB=OSI.IMSA.STEPLIB
IMSA_MQLIB=OSI.IMSA.MQLIB

# ETC SECTION
OSI_LICENSE_PATH=/mnt/nfs/License/OPENFRAME/ofLinux64
```

Appendix B: 설치 검증

본 장에서는 OpenFrame/OSI의 정상 설치를 확인하기 위한 OSI IVP를 설명한다.

B.1. 소개

IVP(Installation Verification Procedure)는 제품 설치 과정에서 시스템이 정상적으로 구축되어 기본적인 기능이 동작하는지 검증하기 위하여 개발되었다. 기존에는 시스템에 문제가 발생했을 경우 운영자가 직접 설정을 모두 확인해야 했지만 현재는 IVP를 사용하여 보다 직관적이고 편리하게 문제점을 확인할 수 있다.

B.2. OSI IVP

OSI IVP는 인스톨러를 통해 기본 설치된다. 관련 샘플 파일은 OpenFrame/osi/oivp에서 확인할 수 있다.

다음은 OSI IVP의 환경설정, 애플리케이션 등록, 검증에 대한 설명이다.

B.2.1. 환경설정

OIVP를 운영하기 위한 Region(IMSA) 설정, 사용자 서버, OSI 데이터 셋의 리소스가 자동으로 생성된다. 모든 등록 절차에 필요한 기본 설치 파일은 인스톨러를 통해서 제공을 하고 있으며, OSI IVP 정상 동작을 위해서는 설치 파일을 수정하지 않고 그대로 사용하는 것을 권장한다.

다음은 시스템과 사용자 서버 설정, 관련 리소스들을 등록하는 순서이다.

1. 시스템 서버 설정

시스템 서버와 해당 서버에서 사용하는 서비스가 Tmax 환경 파일에 등록된다.

2. 사용자 서버 설정

사용자 서버와 해당 서버에서 사용하는 서비스가 Tmax 환경 파일에 등록된다.

3. 리소스 정의 등록

인스톨러에 포함된 osi_define.sh 스크립트는 데이터 셋을 등록한다.

설치용 속성파일에서 자동으로 설치된 항목을 확인하고 필요한 경우 수동으로 스크립트를 실행 및 점검할 수 있다.

다음의 \$OPENFRAME_HOME/scripts/osi_define.sh 명령은 제품 설치 이후 IMSA Region 설정을 수행한다. idcams툴을 통하여 관련 데이터 셋이 생성된다.

```
$ OPENFRAME_HOME/scripts/osi_define.sh
```

다음은 해당 셸의 주요 세부 수행내용이다.

```
idcams define -t CL -n OSI.IMSA.DEFLIB -o KS -k 10,0 -l 100,32760 -s 1024,128,128 -v DEFVOL  
${OPENFRAME_HOME}/bin/osisdgen osi_sdlib_COBOL.dat IMSA
```



시스템 서버 설정, 사용자 서버 설정, 데이터 셋 생성에 대한 자세한 내용은 OpenFrame OSI "운영자 안내서"를 참고한다.

B.2.2. 애플리케이션 등록

환경설정이 완료되면 Map과 COBOL 소스코드를 컴파일한 후 등록된다.

Map과 COBOL 소스코드를 컴파일하고 등록하는 방법은 다음과 같은 순서대로 진행된다. 현재 인스톨러를 통해 자동으로 설치된다.

1. Map 컴파일

OSI IVP에서 사용하는 Map 파일을 복사하고 컴파일한다.

다음의 `OPENFRAME_HOME/scripts/osi_ivp.sh` 스크립트는 내에 OSI IVP Map을 생성하는 명령이 포함되어 있다.

```
$ OPENFRAME_HOME/scripts/osi_ivp.sh
```

다음은 해당 셸의 Map 컴파일 수행내용이다.

```
#mapgen  
cd ${OPENFRAME_HOME}/osi/oivp/mfs/COBOL  
${OPENFRAME_HOME}/bin/osimfsgen -m OSI.IMSA.MFSLIB OIVP001.TXT  
${OPENFRAME_HOME}/bin/osimfsgen -m OSI.IMSA.MFSLIB OIVP002.TXT  
${OPENFRAME_HOME}/bin/osimfsgen -m OSI.IMSA.MFSLIB OIVP003.TXT  
${OPENFRAME_HOME}/bin/osimfsgen -m OSI.IMSA.MFSLIB OIVP004.TXT  
${OPENFRAME_HOME}/bin/osimfsgen -m OSI.IMSA.MFSLIB OIVP005.TXT  
${OPENFRAME_HOME}/bin/osimfsgen -m OSI.IMSA.MFSLIB OIVP006.TXT  
${OPENFRAME_HOME}/bin/osimfsgen -m OSI.IMSA.MFSLIB OIVP014.TXT  
${OPENFRAME_HOME}/bin/osimfsgen -m OSI.IMSA.MFSLIB OIVP015.TXT  
${OPENFRAME_HOME}/bin/osimfsgen -m OSI.IMSA.MFSLIB OIVP024.TXT
```

2. COBOL 소스코드 컴파일

OSI IVP에서 사용하는 COBOL 소스파일을 컴파일한다.

컴파일은 `OPENFRAME_HOME/osi/oivp/src/compile.sh` 스크립트를 사용한다.

```
$ OPENFRAME_HOME/osi/oivp/src/compile.sh
```

다음은 해당 셸의 주요 세부 수행내용이다. (COBOL 샘플소스 기준)

```
ofcob --enable-cbltdli --force-trace -o ${base}.so ${base}.cob
```

3. 리소스 정보 등록 및 데이터 구조 Mapping

IVP 리소스 정보(osi_sdlib.dat) 등을 등록하고 데이터 구조를 매핑하는 과정이 수행된다.

```
$ OPENFRAME_HOME/scripts/osi_define.sh
```

다음은 해당 셸의 리소스 정보 등록에 대한 세부 수행내용이다.

```
#REGISTER SYSTEM DEFINITION
cd ${OPENFRAME_HOME}/osi/resource/data
${OPENFRAME_HOME}/bin/vtamgen vtam.dat
${OPENFRAME_HOME}/bin/osisdgen osi_sdlib_COBOL.dat IMSA
```

4. OSI 시스템 부팅

다음은 OSI 시스템을 부팅하는 명령어이다.

```
TMBOOT for node(NODE1) is starting:
  TMBOOT: TMM is starting: Thu Apr  8 19:12:08 2021
  TMBOOT: CLL is starting: Thu Apr  8 19:12:08 2021
  TMBOOT: CLH is starting: Thu Apr  8 19:12:08 2021
  TMBOOT: TLM(tlm) is starting: Thu Apr  8 19:12:08 2021
[2021-04-08T19:12:08.892931] [osiboot(16341)          ] [M] [OSI7061M] System server(TPFMAGENT)
booting ok
[2021-04-08T19:12:08.896760] [osiboot(16341)          ] [M] [OSI7061M] System server(ofrsasvr)
booting ok
[2021-04-08T19:12:08.899698] [osiboot(16341)          ] [M] [OSI7061M] System server(ofrlhsrv)
booting ok
[2021-04-08T19:12:08.903173] [osiboot(16341)          ] [M] [OSI7061M] System server(ofrdmsvr)
booting ok
[2021-04-08T19:12:08.906781] [osiboot(16341)          ] [M] [OSI7061M] System server(ofrdsedt)
booting ok
[2021-04-08T19:12:08.910483] [osiboot(16341)          ] [M] [OSI7061M] System server(ofrcmsvr)
booting ok
[2021-04-08T19:12:08.914132] [osiboot(16341)          ] [M] [OSI7061M] System server(ofruisvr)
booting ok
[2021-04-08T19:12:08.918033] [osiboot(16341)          ] [M] [OSI7061M] System server(ofrsmlog)
booting ok
[2021-04-08T19:12:08.923688] [osiboot(16341)          ] [M] [OSI7061M] System server(obmjmsvr)
booting ok
[2021-04-08T19:12:08.930825] [osiboot(16341)          ] [M] [OSI7061M] System server(obmjstd)
booting ok
[2021-04-08T19:12:08.935196] [osiboot(16341)          ] [M] [OSI7061M] System server(obmjinit)
booting ok
[2021-04-08T19:12:08.941099] [osiboot(16341)          ] [M] [OSI7061M] System server(obmjhist)
booting ok
[2021-04-08T19:12:08.947371] [osiboot(16341)          ] [M] [OSI7061M] System server(obmjspbk)
```

```

booting ok
[2021-04-08T19:12:08.958199] [osiboot(16341) ] [M] [OSI7061M] System server(ofrpmsvr)
booting ok
[2021-04-08T19:12:08.969419] [osiboot(16341) ] [M] [OSI7061M] System server(obmtsmgr)
booting ok
[2021-04-08T19:12:08.979715] [osiboot(16341) ] [M] [OSI7061M] System server(obmjtimr)
booting ok
[2021-04-08T19:12:08.990644] [osiboot(16341) ] [M] [OSI7061M] System server(tmsvr)
booting ok
[2021-04-08T19:12:08.994759] [osiboot(16341) ] [M] [OSI7061M] System server(osiomsvr)
booting ok
[2021-04-08T19:12:08.994781] [osiboot(16341) ] [M] [OSI7051M] Booting process complete.

```

다음은 Default Region(IMSА)을 부팅하는 명령어이다.

```

$ tjesmgr r IMSACTL
Input USERNAME : ROOT
>
Command : [r IMSACTL]
Node name : A N Y
(JOB00541) /home/oframe6/OpenFrame/volume_DEFVOL/SYS1.JCLLIB/IMSACTL is submitted as
IMSACTL(JOB00541).

$ imscmd IMSA /NRE
IMS control region : [IMSА]
Requested command : [NRE]
-----
NRESTART COMMAND IN PROGRESS
*21098/191312*
-----
Command '/NRE' execution done

$ imscmd IMSA /START DC
IMS control region : [IMSА]
Requested command : [START DC]
-----
START COMMAND COMPLETED
*21098/191315*
-----
Command '/START DC' execution done

$ imscmd IMSA /START REGION IMSAMSG
IMS control region : [IMSА]
Requested command : [START REGION IMSAMSG]
-----
START COMMAND COMPLETED
*21098/191318*
-----
Command '/START REGION IMSAMSG' execution done

```



Map 컴파일, COBOL 소스코드 컴파일, 엔진 기동 관련된 자세한 내용은 OpenFrame OSI "운영자 안내서"를 참고한다.

B.2.3. 검증

OSI IVP을 통한 검증은 MPP(Message Processing Program) 검증, BMP(Batch Message Processing program) 검증으로 나뉜다.

- MPP 검증

IVP로 Online 환경에서 메시지를 처리하는 프로그램을 검증하고, 각 절차에 따라 작업 내역을 확인할 수 있다.

- BMP 검증

실제 데이터 관리를 사용하고, OSI에서 사용자 데이터 셋에 액세스하여 확인할 수 있다. 사용자가 메시지 큐에 적재되어 있는 데이터를 실시간으로 처리하지 않고 JCL을 기동하여 JOB을 실행한 후 결과를 확인한다.

B.2.3.1. MPP 검증

OSI IVP에서는 개인 정보를 관리하는 애플리케이션을 통해 OSI에서 지원하는 리소스 및 명령어를 검증한다. OSI IVP를 시작하려면 터미널을 통해서 OSI 시스템에 접속을 한 후, /FOR 명령으로 IVP 프로그램을 호출한다.

터미널을 통해서 OSI 시스템에 다음과 같은 방법으로 접속한다.

```
IMSA <ENTER>
```

OSI 시스템 접속 후, IVP 프로그램을 다음과 같은 방법으로 호출한다. (COBOL 샘플소스 기준)

```
<CLEAR>  
/FOR 0IVP0010 <ENTER>
```

다음은 IVP의 각 화면에서 사용되는 키와 키 기능 설명이다.

키	기능	적용 화면
CLEAR	시작 화면	전체
ENTER	다음 단계 진행	전체

다음은 OSI IVP 시작 화면이다.

03/05/08

** WELCOME TO OPENFRAME ONLINE **

** OSI INSTALLATION VERIFICATION PROCEDURE **

ACCOUNT 111-11-111 (XXX-XX-XXX)
NAME TMAX
DATE 2012.03.05
AMOUNT 10000
INTEREST 3.450
TAX TYPE FREE

PRINTER XXXXXXXXXX

OIVPI002 -DBSEG(111-11-111), STATUS CODE() ; INQUIRY SUCCESS
ENTER APPROPRIATE VALUES AND PRINTER TO INQUIRY.

[SYS INFO] - MPP/IO-PCB,DB-PCB,ALT-PCB TESTING NOW

11:05:21

COPYRIGHT(C) 2012, TMAXSOFT, ALL RIGHTS RESERVED.

OSI IVP 시작 화면

OSI IVP의 화면 구성은 크게 3부분으로 나뉜다.

- 상단

OpenFrame 시스템에서 제공하는 날짜 및 시간, 터미널 정보를 출력한다.

- 중간

실제 IVP 애플리케이션이 실행된다.

- 하단

저작권 정보를 표시한다.

OSI IVP는 4개의 화면(INQR, INSR, UPDT, DELT)으로 구성되며 CODE를 입력하여 해당 화면의 사용이 가능하다.



INQR, UPDT, DELT의 경우 FILE에 레코드가 존재해야 실행 가능하므로 INSR를 사용하여 레코드를 먼저 입력해야 한다.

- **INQR 화면**

INQR 화면은 파일에 저장된 특정 레코드를 보여준다. 시작 화면에서 CODE 값을 입력하여 액세스한다. 잘못된 CODE를 입력하면 시작 화면에서 에러가 발생한다.

03/05/08

** WELCOME TO OPENFRAME ONLINE **

** OSI INSTALLATION VERIFICATION PROCEDURE **

```
ACCOUNT  111-11-111   (XXX-XX-XXX)
NAME     TMAX
DATE     2012.03.05
AMOUNT   10000
INTEREST 3.450
TAX TYPE FREE

PRINTER  ██████████
```

OIVPI002 -DBSEG(111-11-111), STATUS CODE() ; INQUIRY SUCCESS
ENTER APPROPRIATE VALUES AND PRINTER TO INQUIRY.

[SYS INFO] - MPP/IO-PCB,DB-PCB,ALT-PCB TESTING NOW

11:05:21

COPYRIGHT(C) 2012, TMAXSOFT, ALL RIGHTS RESERVED.

OSI IVP INQR 화면

• INSR 화면

INSR 화면에서는 새 레코드를 입력한다. 레코드의 각 필드는 ACCOUNT, NAME, DATE, AMOUNT, INTEREST, TAX TYPE로 구성된다. 순차적으로 해당 항목을 기입하고 등록한다.

03/05/08

** WELCOME TO OPENFRAME ONLINE **

** OSI INSTALLATION VERIFICATION PROCEDURE **

```
ACCOUNT  111-11-111   (XXX-XX-XXX)
NAME     TMAXSOFT
DATE     2012.03.05   (YYYY.MM.DD)
AMOUNT   10000
INTEREST 3.450       (0.000)
TAX TYPE FREE       (FREE, NORM)
```

OIVPI003 - DBSEG(111-11-111), STATUS CODE() INSERT SUCCESS
ENTER APPROPRIATE VALUES TO INSERT.

[SYS INFO] - MPP/IO-PCB,DB-PCB TESTING NOW

11:04:32

COPYRIGHT(C) 2012, TMAXSOFT, ALL RIGHTS RESERVED.

OSI IVP INSR 화면

• UPDT 화면

UPDT 화면에서는 기존 레코드를 수정한다. 시작 화면에서 CODE 값을 입력하여 액세스한다. 잘못된 CODE를 입력하면 시작 화면에서 에러가 발생한다.

```

03/05/08
** WELCOME TO OPENFRAME ONLINE **
-----
** OSI INSTALLATION VERIFICATION PROCEDURE **

ACCOUNT 111-11-111 111-11-111
NAME    TMAX          TMAXSOFT
DATE    2012.03.05 2012.03.05
AMOUNT  10000       22222
INTEREST 3.450     3.450
TAX TYPE FREE       NORM

-----
OIVPI004 - DBSEG(111-11-111), STATUS CODE( ) ; UPDATE SUCCESS
DATA UPDATED.
-----
[SYS INFO] - BMP/IO-PCB TESTING NOW                               11:06:11
-----
COPYRIGHT(C) 2012, TMAXSOFT, ALL RIGHTS RESERVED.

```

OSI IVP UPDT 화면

• DELT 화면

DELT 화면에서는 입력되어 있는 기존 레코드를 삭제한다. 시작 화면에서 CODE 값을 입력하여 액세스한다. 잘못된 CODE를 입력하면 시작 화면에서 에러가 발생한다.

```

03/05/08
** WELCOME TO OPENFRAME ONLINE **
-----
** OSI INSTALLATION VERIFICATION PROCEDURE **

ACCOUNT 111-11-111 (XXX-XX-XXX)
NAME    TMAXSOFT
DATE    2012.03.05
AMOUNT  22222
INTEREST 3.450
TAX TYPE NORM

-----
OIVPI005 - DBSEG(111-11-111), STATUS CODE( ) ; INQUIRY SUCCESS
ENTER APPROPRIATE VALUES TO DELETE.
-----
[SYS INFO] - IFP/IO-PCB,DB-PCB TESTING NOW                               11:06:30
-----
COPYRIGHT(C) 2012, TMAXSOFT, ALL RIGHTS RESERVED.

```

OSI IVP DELT 화면

B.2.3.2. BMP 검증

BMP 검증은 데이터 셋에 누적된 메시지를 확인하는 것이다. tjesmgr 톨로 JOB을 수행하고 해당 로그 정보를 확인할 수 있다.

다음은 tjesmgr 명령으로 JCL을 수행하는 BMP를 활용하는 과정이다.

1. tjesmgr 를 실행한다.

```
$ tjesmgr
```

2. JCL 수행을 위하여 boot 명령을 수행하고, IMSABMP2를 동작(RUN)시킨다.

```
Tmax JES Manager 7.1
+-----+
|
| JOB ID   : JOB00543      NODE NAME  : NODE1
| JOB NAME : OSIBMPT
| JOB CLASS : A , JOB STATUS : Done(R00000) , JOB PRY  : 5 , JCLRUNNER INDEX : 5
| JOB USER  : ROOT
| JCL PATH  : /home/oframe6/OpenFrame/volume_DEFVOL/SYS1.JCLLIB/IMSABMP2
| TIME STAMP : READY : 20210408/19:14:29, START : 20210408/19:14:30
|                                     END   : 20210408/19:14:31
| RES USAGE : PROCESS - 1s CPU - (0s,0%) MEM - 0Byte
|
| STEP LIST :
|-----+
| [ TSTEP1] START : 20210408/19:14:30, RC=R0000, CPU - 0s (0s)
|-----+
|
| SPOOL LIST :
|-----+
| NO  STEP          DDNAME      SIZE  DSNAME
|-----+
| 0  -----          INPJCL        214  INPJCL
| 1  -----          SYSMMSG         1K  SYSMMSG
| 2  -----          CATPROC          0  CATPROC
| 3  -----          CONJVJCL        211  CONJVJCL
| 4  -----          JESMSG         730  JESMSG
| 5  -----          JESJCL         824  JESJCL
| 6  TSTEP1          SYSOUT          2K  oframe6.OSIBMPT.JOB00543.D000001
|-----+
|
| OUTPUT PROCESSING STATUS : all outputs were processed
```

3. tjesmgr의 PODD @j di=6 명령을 이용하여 INQR된 정보를 확인하고 업데이트 상황을 조회할 수 있다.

```
|Command : [podd @j di=5]
|2021-04-08T19:14:30.609042] [DFSRRRC00(16835)          ] [M] [RRC0901M] DFSRRRC00 PROGRAM START
|2021-04-08T19:14:30.609077] [DFSRRRC00(16835)          ] [M] [RRC0903M] INITIALIZE BMP REGION
|2021-04-08T19:14:30.609226] [DFSRRRC00(16835)          ] [M] [RRC0906M] SHARED OBJECT TYPE
|2021-04-08T19:14:30.609626] [DFSRRRC00(16835)          ] [M] [RRC0921M] CHILD PROCESSID -
pid=16841
```

```

[2021-04-08T19:14:30.609827] [DFSRR00(16841) ] [D] [DEBUGP1D]
path=/home/oframe6/OpenFrame/core//appbin/osibmpsv
[2021-04-08T19:14:30.649096] [osibmpsv(16841) ] [M] [OSI0291M] IMSA server boots -
resource manager initialization starts
[2021-04-08T19:14:30.650632] [osibmpsv(16841) ] [M] [OSI0101M] osimfs version: 7.2.0(4)
obuild@tplinux64:ofsrc71/osi(#1) 2021-03-26 14:11:10
[2021-04-08T19:14:30.652909] [osibmpsv(16841) ] [M] [OSI0291M] IMSA server boots -
resource manager initialization completed
[2021-04-08T19:14:30.671149] [osibmpsv(16841) ] [M] [OSI5251M] osibmpsv initialization ok
[2021-04-08T19:14:30.671185] [osibmpsv(16841) ] [M] [OSI5263M] DFSRR00 PARM :
MBR=OIVPIL02
[2021-04-08T19:14:30.671192] [osibmpsv(16841) ] [M] [OSI5263M] DFSRR00 PARM :
PSB=OIVPIL02
[2021-04-08T19:14:30.671197] [osibmpsv(16841) ] [M] [OSI5263M] DFSRR00 PARM :
IN=OIVBMP2
[2021-04-08T19:14:30.671203] [osibmpsv(16841) ] [M] [OSI5263M] DFSRR00 PARM : OUT=
[2021-04-08T19:14:30.671208] [osibmpsv(16841) ] [M] [OSI5263M] DFSRR00 PARM : IMSID=IMSA
[2021-04-08T19:14:30.671216] [osibmpsv(16841) ] [M] [OSI5263M] DFSRR00 PARM : AGN=
[2021-04-08T19:14:30.684645] [osibmpsv(16841) ] [M] [OSI2010M] pgmpath :
/home/oframe6/OpenFrame/volume_DEFVOL/OSI.IMSA.STEPLIB/OIVPIL02.so
MSG-COUNT : [00001]
BEFORE GU
MESSAGE EXIST
-----[ INQUIRY ]-----
ACCOUNT : [ ]
-----
MSG-COUNT : [00002]
BEFORE GU
MESSAGE EXIST
MSG-COUNT : [00001]
BEFORE GU
MESSAGE NOT EXIST
[2021-04-08T19:14:30.749841] [osibmpsv(16841) ] [M] [OSI0202M] IMSA server downs -
resource manager finalization starts
[2021-04-08T19:14:30.749893] [osibmpsv(16841) ] [M] [OSI0202M] IMSA server downs -
resource manager finalization completed
[2021-04-08T19:14:30.752047] [DFSRR00(16835) ] [M] [RRC0922M] CHILD TERMINATE - RC=R0000
[2021-04-08T19:14:30.756974] [DFSRR00(16835) ] [M] [RRC0902M] DFSRR00 PROGRAM FINISH

```



TJES 유틸리티에 대한 자세한 내용은 OpenFrame Batch "TJES 안내서"를 참고한다.