

Tech Info(vol.11)

Informix Storage Manager 설치 가이드

소개

ISM이란

ISM(Informix Storage Manager)는 Informix 데이터베이스 서버에 데이터 저장 관리 서비스를 제공합니다. ISM은 ON-Bar로부터 백업 및 복원 요청을 수신하며, 저장 장치에 설치된 저장 볼륨과 데이터를 직접 주고 받습니다. ISM은 데이터베이스 관리자 또는 시스템 관리자가 결정하는 데이터 수명 주기 동안 백업된 데이터를 찾으며 자동으로 저장 장치 및 저장 볼륨을 관리할 수 있습니다.

ISM 설치 방법

준비 사항

- 모든 OS 지원
- Informix IDS 7.24 이상의 모든 버전

UNIX 설치 방법

root 권한에서 아래와 같이 실행합니다.

1. Informix 환경을 확인합니다.

InformixDIR, InformixSERVER, PATH 등 Informix 환경이 설정되어 있는지 확인합니다.

env로 확인

```
InformixDIR=/u1/IDS731UC6 <Informix가 설치된 directory>
```

```
InformixSERVER=ifx731 <DB server name>
```

```
PATH=$PATH:$InformixDIR/bin
```

Informix가 설치된 directory

이밖에 필요에 따라

```
DB_LOCALE=ko_kr.ksc
```

```
SERVER_LOCALE=ko_kr.ksc
```

```
CLIENT_LOCALE=ko_kr.ksc
```

```
LD_LIBRARY_PATH=$InformixDIR/lib:$InformixDIR/lib/esql :
```

```
$LD_LIBRARY_PATH
```

(solaris, sequent일 경우 LD_LIBRARY_PATH, HP_UX 경우는 SHLIB_PATH, AIX 경우는 LIB_PATH)

2. \$InformixDIR/etc/\$ONCONFIG parameter를 setting합니다.

```
ISM_DATA_POOL ISMDiskData
```

```
ISM_LOG_POOL ISMDiskLogs
```

3. 2번에서 수정한 parameter를 적용하기 위해 DB engine을 다시 시작합니다.

```
$) onmode -ky
```

```
$) oninit
```

4. root user로 login하여 ism_startup -init 명령을 실행하는데, 이것은 ISM을 초기화한 후 실행시킵니다.

```
# ism_startup -init
```

```
ism_startup: Configure devices manually using ism_add.
```

5. ism daemon이 올라왔는지 확인해봅니다.

```
# ps -ef | grep nsr
root 19051 18968 2 16:47:22 ttyAH/AUHp 0:00 grep nsr
root 18993 1 0 16:46:41 ? 0:00
/u1/IDS731UC6/bin/nsrexecd
root 18994 18993 0 16:46:41 ? 0:00
/u1/IDS731UC6/bin/nsrexecd
root 18996 1 0 16:46:44 ? 0:04
/u1/IDS731UC6/bin/nsrd
root 19003 18996 10 16:46:47 ? 0:00
/u1/IDS731UC6/bin/nsrmmdbd
root 19004 18996 10 16:46:50 ? 0:00
/u1/IDS731UC6/bin/nsrindexd
root 19005 18996 0 16:46:52 ? 0:00
/u1/IDS731UC6/bin/nsrmmmd -n 1
```

6. Informix를 ISM 사용자로 만들어 Informix가 일상적으로 ISM작업을 수행하도록 하기 위해 다음 명령을 실행합니다.

```
# ism_add -admin Informix@server_name
```

*) 여기서 server_name은 hostname입니다.

7. backup시 data를 저장할 directory를 생성합니다.

```
# cd /u1/lullu/ismspace
# mkdir ISMDATA
```

```
# mkdir ISMLOGS  
  
# chown Informix:infomrix ISMDATA ISMLOGS
```

8. ISM에서 사용할 device를 정의합니다.

```
# ism_add -device /u1/lullu/ismspace/ISMDATA -type file  
# ism_add -device /u1/lullu/ismspace/ISMLOGS -type file  
# ism_watch  
getting size: Invalid argument  
getting size: Invalid argument  
  
Server: kor-sqnt Fri Mar 23 17:34:15 2001  
  
Up Version: Informix_Storage_Manager ISM.2.20.UC1.114-08/13 Server Edition  
Saves: 0 session(s) Recovers: 0 session(s)
```

Device	type	volume
/dev/nrst8	8mm 5GB	(none)
mspace/ISMDATA	file	(none)
mspace/ISMLOGS	file	(none)

```
Sessions:  
  
Messages:  
  
Fri 17:28:57 server notice: started  
Fri 17:29:07 index notice: completed checking 1 client(s)  
  
Pending:
```

9. 각 device에 대한 레이블을 지정합니다.

```
# ism_op -label /u1/lullu/ismspace/ISMDATA -volume 1 -pool
```

```
ISMDiskData
```

```
# ism_op -label /u1/lullu/ismspace/ISMLOGS -volume 2 -pool
```

```
Tue 09:17:35 /u1/lullu/ismospace/ISMLOGS label without mount operation in prog Tue 09:17:38
```

```
/u1/lullu/ismospace/ISMLOGS labeled file disk 2
```

```
Pending:
```

10. 각 device를 mount합니다.

```
# ism_op -mount /u1/lullu/ismospace/ISMDATA
```

```
file disk 1 mounted on /u1/lullu/ismospace/ISMDATA, write enabled
```

```
# ism_op -mount /u1/lullu/ismospace/ISMLOGS
```

```
file disk 2 mounted on /u1/lullu/ismospace/ISMLOGS, write enabled
```

```
# ism_watch
```

```

/u1/lullu/ismspace/ISMLOGS labeled file disk 2
Tue 09:19:11 /u1/lullu/ismspace/ISMDATA mount operation in progress
Tue 09:19:13 /u1/lullu/ismspace/ISMDATA mounted file disk 1
Tue 09:20:45 /u1/lullu/ismspace/ISMLOGS mount operation in progress
Tue 09:20:47 /u1/lullu/ismspace/ISMLOGS mounted file disk 2
Pending:

```

11. On-Bar Backup과 관련된 ONCONFIG parameter들은 다음과 같습니다. 이 parameter들을 변경할 경우, 변경된 값을 적용하기 위해서는 DB엔진을 재시동해야 합니다.

LTAPEDEV	/u1/lullu/ltape.dev
(onbar로 백업을 받을 때, LTAPEDEV가 /dev/null이면 안됨)	
BAR_BSA_LIB_PATH	\$InformixDIR/lib/libbsa.so ; X/BSA 라이브러리 위치
BAR_DEBUG_LEVEL	0 ; 디버그 메시지 레벨 (0~9 레벨, 초기값 0)
BAR_DEBUG_LOG	/tmp/bar_dbug.log ; 디버그 로그 파일 위치
BAR_ACT_LOG	/tmp/bar_act.log
BAR_MAX_BACKUP	0
BAR_RETRY	1
BAR_NB_XPORT_COUNT	10
BAR_XFER_BUF_SIZE	15
	(default 15 for PAGESIZE=4k, default 31 for PAGESIZE=2k)
LOG_BACKUP_MODE	{CONT MANUAL} ; no default

12. ism_startup -init을 했을 때 생성되는 sm_versions파일을 사용하기 위해 sysutil database를 rebuild합니다.

```
# $InformixDIR/etc/bldutil.sh
```

13. ISM이 정상적으로 작동하는지 확인 하기 위해 OnBar를 사용하여 백업과 복구를 테스트해봅니다.

```
# onbar -b -w ; 전체 백업 시작  
  
# onbar -r -w -p ; logical log 없이 전체 restore
```

설치 제거 방법

1. Informix 사용자로 login 을 합니다.
2. ISM 실행 파일 directory 로 이동합니다.

```
# ism_shutdown -q
```

5. daemon이 종료되었는지 확인해 봅니다.

```
# ps -ef | grep nsr
```

6. ISM 실행 파일을 임시 디렉토리로 이동시킵니다.

```
# mv scanner* ism.tmp  
  
# mv uasm* ism.tmp  
  
# mv recover* ism.tmp
```

7. ISM 카탈로그를 임시 디렉토리로 이동합니다.

```
# cd $InformixDIR# mv ism ism.bak
```

8. ISM 카탈로그로 연결된 link를 제거합니다.

```
# rm /nsr
```

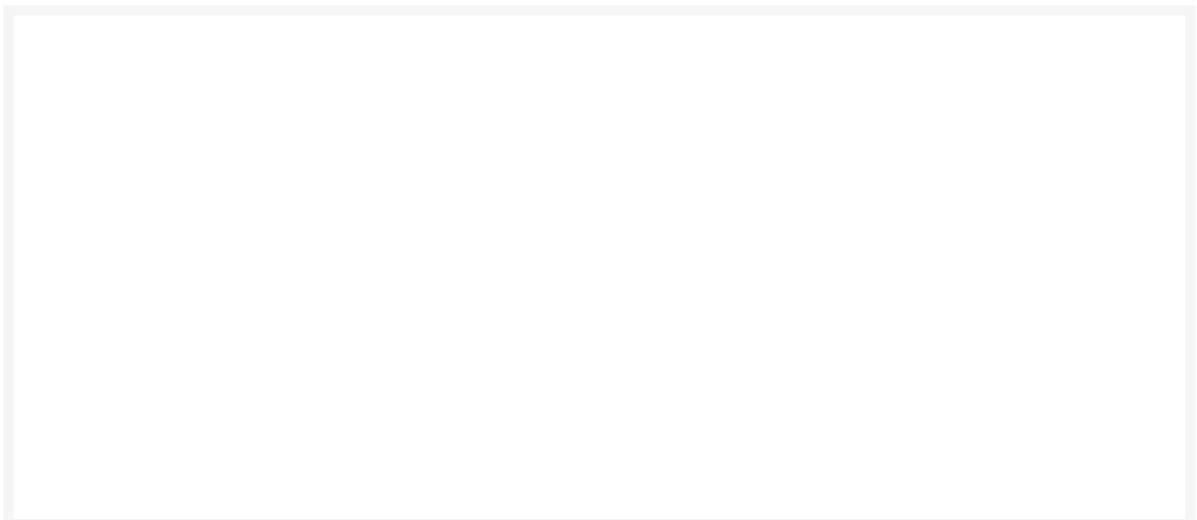
9. \$InformixDIR/bin/onbar 스크립트에서의 ism_catalog 호출을 제거합니다.

Informix C-ISAM DataBlade Module 1.0

Informix C-ISAM 및 C-ISAM DataBlade 소개

Informix의 C-ISAM는 ISAM 파일을 효율적으로 관리하는 C 함수 라이브러리입니다. Informix 제품 가운데 가장 오래되고 가장 널리 사용되어 온 C-ISAM은 8년 이상 미국 및 유럽에서 업계 표준 UNIX용 ISAM 제품으로서 군림했습니다. ISAM(Indexed Sequential Access Method) 파일 관리 기술은 오늘날의 RDBMS의 모체입니다. ISAM은 RDBMS의 오버헤드 없이 데이터에 신속하고 비용 효과적으로 액세스할 수 있도록 지원하기 때문에 앞으로도 계속해서 많은 기업들을 위한 경쟁력 있는 환경이 될 것입니다. 그러나 ISAM 환경은 SQL 액세스, 저장 및 복구, 고가용성, 중복 등 RDBMS 환경에서 제공되는 주요 기능을 지원하지 못하고 있습니다. 그래서, C-ISAM을 이용해 애플리케이션을 개발해 온 많은 고객들 및 소프트웨어 공급 업체들이

.....



- 사용자가 커스텀 프로그래밍의 양을 대폭 줄여주는 표준 SQL 클라이언트 툴을 통해 C-ISAM 데이터에 액세스할 수 있도록 함으로써 **개발 비용을 최소화**하게 됩니다.
- C-ISAM 애플리케이션을 위한 신속하고 효율적인 마이그레이션 경로를 제공함으로써 RDBMS로 **신속하게 마이그레이션** 할 수 있도록 합니다.
- 데이터가 실제로 저장된 위치와 관계없이 C-ISAM 과 SQL 애플리케이션 모두를 통해 데이터에 동시 액세스 할 수 있도록 하는 **동시 액세스 기능**을 제공합니다.

C-ISAM DataBlade Module

Informix C-ISAM DataBlade 모듈은 2개의 컴포넌트로 구성되어 있습니다.

- SQL Access
- Server Storage

SQL Access 컴포넌트는 C-ISAM 데이터에 대한 SQL 인터페이스를 제공하고, Server Storage 컴포넌트는 사용자가 Informix Dynamic Server 데이터베이스 내에 C-ISAM 데이터를 직접 저장할 수 있도록 하는 한편 C-ISAM 애플리케이션이 데이터에 지속적으로 액세스 하도록 지원합니다.

SQL Access를 사용해 C-ISAM 데이터에 대한 SQL 인터페이스 제공

C-ISAM DataBlade 모듈의 SQL Access 컴포넌트는 기존 C-ISAM 데이터 파일의 상단에 SQL 인터페이스를 위치시킵니다. 이는 사용자가 C-ISAM에 사용자 데이터를 지속적으로 유지할 수 있도록 할 뿐만 아니라 SQL을 통한 해당 데이터에 대한 동시 액세스를 제공합니다. 사용자는 SELECT, INSERT, UPDATE, DELETE 등 SQL Access로 모든 표준 SQL 문을 실행할 수 있습니다.

Server Storage를 이용한 C-ISAM 애플리케이션에서 데이터베이스 데이터 액세스하기

많은 고객 및 소프트웨어 공급업체들은 C-ISAM 데이터를 마이그레이션시켜 직접 Informix 데이터베이스에 저장하기를 원하고 있습니다. 사용자는 C-ISAM DataBlade 모듈의 Server Storage 컴포넌트를 이용해 애플리케이션 코드를 변경하지 않고도 C-ISAM 애플리케이션에서 이 데이터에 계속 액세스할 수 있는데, 이를 통해 C-ISAM 사용자들은 데이터를 마이그레이션시킬 때 C-ISAM 애플리케이션을 마이그레이션하지 않고도 RDBMS가 가진 모든 이점을 활용할 수 있게 되는 것입니다. 즉, 데이터는 Informix 데이터베이스에 있는 물리적 테이블에서 유지되며 C-ISAM과 SQL 애플리케이션은 동시에 이 데이터에 액세스할 수 있는 것입니다.

C-ISAM DataBlade 설치 시 시스템 요구 사항

지원 운영 체제 (2000 년 현재)

- Solaris 2.5.1 이상에서 운영되는 Sun SPARC
- HP-UX 10.20 이상에서 운영되는 HP
- AIX 4.2.1 이상에서 운영되는 IBM
- Sequent 4.4.2 이상

지원 데이터베이스 서버

- Informix Dynamic Server™ 9.14 이상(Universal Data Option 포함)

최소 메모리

- 사용자 당 RAM 32K

최소 디스크 공간

- 50MB 디스크 공간

C-ISAM DataBlade 설치 및 사용 방법

준비 사항

- Informix Dynamic Server 9.14 이상
(여기서는 Informix Dynamic Server 9.21 UC3를 사용하겠습니다.)

설치 방법 (SunOS 5.6 , C-ISAM DataBlade 1.10.UC1 기준)

1. C-ISAM DataBlade를 install 합니다.

a. Informix 환경을 확인합니다.

```
InformixDIR=/CS1/IDS921UC3 <Informix가 설치된 directory>
```

```
PATH=$PATH:$InformixDIR/bin
```

```
ONCONFIG=onconfig.mig_test
```

이밖에 필요에 따라

```
DB_LOCALE=ko_kr.ksc
```

```
SERVER_LOCALE=ko_kr.ksc
```

```
CLIENT_LOCALE=ko_kr.ksc
```

```
InformixSQLHOSTS=/home2/lullu/sqlhosts
```

```
LD_LIBRARY_PATH=$InformixDIR/lib:$INFOMRIXDIR/lib/esql
```

```
:$LD_LIBRARY_PATH
```

(solaris, sequent일 경우 LD_LIBRARY_PATH, HP_UX 경우는 SHLIB_PATH, AIX 경우는 LIB_PATH)

b. C-ISAM DataBlade 설치할 디렉토리에 제품을 tar나 cpio로 풉니다.

```
$) pwd
```

```
/home2/lullu
```

```
$) tar -xvf cisamdb.1.10.UC1_prod.tar
```

c. 제품을 풀면 CISAMDB라는 디렉토리가 생기고 CISAMDB라는 디렉토리 아래에 제품이 설치됩니다.

d. CISAMDB_DIR이라는 환경변수를 setting하고, library path에 CISAMDB_DIR/lib를 추가해줍니다.

```
CISAMDB_DIR=/home2/lullu/CISAMDB
```

```
LIB_PATH )
```

e. CISAMDB라는 디렉토리에 들어가서 Informix 환경의 super user로 installcisamdb를 실행합니다.

```
$) cd $CISAMDB_DIR
```

```
$) installcisamdb
```

Installation and Configuration Script

This installation procedure must be run by a privileged user

(Super User)

It will change the owner, group, mode, (and other file attributes on Secure systems) of all files of this package in this directory.

There must be a user "Informix" and a group "Informix" known to the system.

Press RETURN to continue, or the interrupt key (usually CTRL-C or DEL) to abort.

CISAMDB_DIR is being set to [/home2/lullu/CISAMDB].

Enter your serial number (for example, INF#X999999) >

(serial number를 입력)

Enter your serial number KEY (uppercase letters only) >

(serial number key를 입력)

WARNING!

This software, and its authorized use and number of users, are subject to the applicable license agreement with Informix Software, Inc.

If the number of users exceeds the licensed number, the excess users may be prevented from using the software. UNAUTHORIZED USE OR COPYING MAY SUBJECT YOU AND YOUR COMPANY TO SEVERE CIVIL AND CRIMINAL LIABILITIES.

Press RETURN to continue,
or the interrupt key (usually CTRL-C or DEL) to abort.

Installing directory .

Installing directory etc/

Installing directory bin/

Installing directory release/

Installing directory demo/

Installing directory demo/cisamdb/

Installing directory demo/udt/

Installing directory demo/mfcobol/

Installing directory lib/

Installing directory incl/

Installing directory extend/

Installing directory extend/cisamdb.1.10.UC1/

Installing directory msg/

Installing directory msg/en_us/

Installing directory msg/en_us/0333/

Installing directory gls/

Installing directory gls/cm3/

Installing directory gls/lc11/

Installing directory gls/lc11/en_us/

Installing directory /CS1/IDS921UC3/extend/

Installing directory /CS1/IDS921UC3/extend/cisamdb.1.10.UC1/

```
Installing directory /CS1/IDS921UC3/extend/cisamdb.1.10.UC1/
```

```
Installation of Informix C-ISAM DataBlade complete.
```

f. \$InformixDIR/extend/cisamdb.version이라는 디렉토리가 생성되어 있는지 확인해 봅니다.

```
$) cd $InformixDIR/extend
```

```
$) ls
```

```
cisamdb.1.10.UC1
```

2. Informix 사용자로 cisam database에 cisamblade를 등록합니다.

a. dbaccess에서 cisam database를 생성하고 CISAMDB_DATABASE 환경변수를 setting합니다.

```
database sysmaster:
```

```
create database cisamdb
```

```
$) export CISAMDB_DATABASE=cisamdb
```

b. cisamdb 에 cisam datablade를 등록합니다.

i. blademgr을 실행

ii. list cisamdb

iii. show modules

iv. register cisamdb.version cisamdb

```
$) blademgr
```

```
cisamserver>list cisamdb
```

```

There are no modules registered in database cisamdb.

cisamserver>show modules

1 DataBlade modules installed on server cisamserver:

cisamdb.1.10.UC1

cisamserver>register cisamdb.1.10.UC1

cisamdbRegister module cisamdb.1.10.UC1 into database cisamdb?

[Y/n]y

Registering DataBlade module... (may take a while).

DataBlade cisamdb.1.10.UC1 was successfully registered in database cisamdb.

cisamserver>list cisamdb

DataBlade modules registered in database cisamdb:

cisamdb.1.10.UC1

cisamserver>quit

Disconnecting...

$)

```

3. \$ONCONFIG 파일을 다음 파라미터를 추가한 후, 이 값들을 적용하기 위해 DB server를 재시동 합니다.

a. CISAMDB_MAXUSERS

b. CISAMDB_TRANSFERSIZE

(byte단위, record size 보다 크거나 같은 값)

c. CISAMDB_WAITTIME

```
$) vi $InformixDIR/etc/$ONCONFIG
```

```
..
```

```
CISAMDB_MAXUSERS 5
```

```
..
```

```
$) onmode -ky
```

```
$) oninit
```

4. 다음의 환경 변수들이 제대로 setting되었는지 확인합니다.

a. InformixDIR

b. InformixSERVER

c. InformixSQLHOSTS

d. ONCONFIG

e. CISAMDB_DIR

f. CISAMDB_DATABASE (단 Cisam Appls에서 사용하기 위함)

g. PATH

h. Shared Library Path : LIB_PATH (Aix), LD_LIBRARY_PATH (Sun,Sequent), SHLIB_PATH (HP)

5. root user로 cisamdatablade 데몬을 실행시킵니다.

a. PATH에 \$CISAMDB_DIR/bin을 추가합니다.

b. \$CISAMDB_DIR/bin/cisamdbd

c. cisamdbd가 실행되었는지 확인합니다.

```
$) onstat -m
```

29108

d. cisamdbd를 중단하려면 다음 명령을 사용하시면 됩니다.

```
$) cisamdbd -kill  
  
CISAMDB Daemon process 29108 killed by 'cisamdbd -kill'  
  
$) cisamdbd -r  
  
No cisamdbd process is currently running.
```

e. cisamdb에 할당된 memory와 semaphore를 제거하려면 다음 명령을 사용합니다.

```
$) cisamdbd -d  
  
Shared Memory removed. Id = 139.  
  
Semaphore set no. 1 removed. Id = 74.  
  
Semaphore set no. 2 removed. Id = 75
```

6. cisam database가 사용할 external space를 생성합니다.

```
onspaces -c -x extspace_name -l full_directory
```

```
$) pwd  
  
/home2/lullu/CISAMDB  
  
$) mkdir DBS$) onspaces -c -x cisamdb -l /home2/lullu/CISAMDB/DBS  
  
External space successfully created.
```

7. cisam에서 사용하는 data와 index 파일을 SQL문으로 접근하기 위해서 data파일의 정보를 가지고 table을 생성합니다.

```
create table ( coll ... ) in cisam_extspace USING cisamdb ( directory=" . . . " , filename=" . . . " );
```

```
database cisamdb;
```

```
create table employee
( emp_number integer,
emp_last_name char(20),
emp_first_name char(20),
emp_address char(20),
emp_city char(20)
) in cisamdb USING CISAMDB( directory = '/home2/lullu/CISAMDB/test', filename='employee');
```

이때, directory는 cisam 파일 (.dat 와 .idx 파일) 이 있는 위치를 나타내고, filename은 cisam 파일의 이름을 의미합니다.

a. c-isam 데이터 파일을 테이블로 생성하기 위하여 column의 data type은 다음과 같이 변경하면 됩니다.

Cisam - database (size)

i. char - char

ii. short - smallint (2)

iii. long - integer (4)

iv. float - smallfloat (4)

v. double - float (8)

vi. decimal - decimal

vii. variable length type 은 지원되지 않습니다.

8. Query를 실행시켜봅니다.

database에서 사용하는 select, insert, update, delete 등 모든 표준 SQL문을 사용할 수 있습니다.

```
database cisamdb ;
```

(퀴리문 수행);