



UNIT 04

인스턴스



DB2 엔진의 기능을 사용할 수 있는 논리적인 환경을 DB2 인스턴스라고 합니다. 한 서버에서 독립적인 환경을 가지는 한 개 이상의 인스턴스를 운영할 수 있습니다. Linux 및 Unix서버에서 인스턴스는 특정한 사용자 계정과 연관되며, db2icrt 명령어를 이용하여 root 사용자가 생성합니다.

DB2 9.7 운영자 가이드

Administrator Edition

- 인스턴스 개요
- 인스턴스 생성
- 인스턴스 갱신
- 인스턴스 제거
- 인스턴스 기동
- 인스턴스 중지
- 인스턴스 강제 중지
- 인스턴스 구성 파일
- 인스턴스 지정 방법
- DB2 레지스터리 변수
- DB2 관리 서버
- 원격 클라이언트 지원
- DPF 환경 설정



Point



DB2 엔진의 기능을 사용할 수 있는 논리적인 환경을 DB2 인스턴스라고 합니다. 한 서버에 여러 개의 인스턴스를 생성하여 독립적인 DB2 엔진으로 운영할 수 있습니다. LINUX/UNIX에서 인스턴스는 OS 사용자 계정과 연관됩니다.

Tip

데이터베이스 관리자를 간단히 DBM 이라고 합니다.

Tip

인스턴스 수준에서 설정하는 환경 변수는 db2 registry (db2set) 정보와 dbm cfg 입니다.

1

인스턴스는 DB2 엔진의 기능을 사용할 수 있는 논리적인 환경으로서 DB2 데이터베이스 관리자라고 합니다.

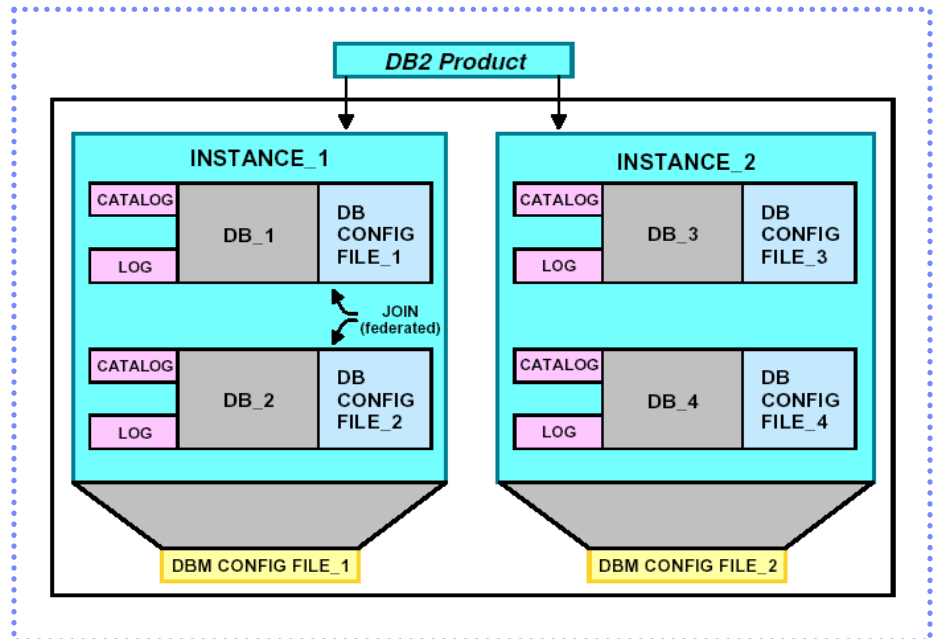


Figure 0401A DB2 인스턴스

2

DB2 엔진의 기능을 사용하고, 데이터베이스를 구축하려면 DB2 서버 제품을 설치한 후에 DB2 인스턴스를 생성해야 합니다. UNIX에서 인스턴스는 OS 사용자 계정과 연관됩니다. Windows 에서는 인스턴스와 OS의 사용자 계정이 연관되지 않습니다.

3

한 서버에 한 개 이상의 인스턴스를 생성할 수 있으며, 각 인스턴스는 독립적인 DB2 엔진으로 운영되므로 개발용 인스턴스와 운영용 인스턴스를 따로 생성할 수 있습니다.

4

각 인스턴스는 고유의 환경을 구성하는 인스턴스 구성 파일을 가지게 됩니다. 인스턴스 구성 파일에는 다양한 인스턴스 구성 변수들이 있습니다.

5

동일한 서버에 존재하는 DB2 인스턴스를 지역 인스턴스라고 하고, 원격 서버에 존재하는 DB2 인스턴스를 원격 인스턴스라고 합니다.

6

클라이언트와 통신을 위한 프로토콜 및 서비스 포트의 지정은 인스턴스 수준에서 정의됩니다.

Tip

한 서버에 여러 개의 인스턴스를 생성하는 경우에도 DB2 제품은 한번만 설치 합니다.

Point



root 사용자로 로그인하여 db2icrt 명령어를 이용하여 한 개 이상의 인스턴스를 생성합니다. 인스턴스 사용자로 사용될 OS 사용자 계정과 그룹이 필요합니다.

Tip

인스턴스 사용자 계정의 일자 그룹은 고유한 그룹명으로 지정하는 것을 권장합니다.

Tip

<fenced 사용자명>은 <인스턴스명> 과 동일하게 설정 가능하나, 안전 저장 프로시저의 사용을 위해 별도의 Fenced 사용자를 정의하여 사용하기를 권장합니다.

Tip

DB2 인스턴스와 관련된 TCP/IP 포트의 번호는 변경할 수 있습니다.

Tip

Windows 에서는 설치 시에 DB2 라는 이름의 인스턴스가 자동으로 생성됩니다. 인스턴스를 추가로 생성하는 것이 가능하지만, 기본 인스턴스인 DB2를 사용하는 것을 권장합니다.

1 Linux 및 UNIX에서 인스턴스를 생성하려면, OS 그룹과 사용자 계정이 필요합니다.

```
$ login root
$ mkgroup <일자 그룹명>
$ mkuser pgrp=<일자 그룹명> <인스턴스 사용자명>
$ passwd <인스턴스 사용자명>
```

2 DB2 제품이 설치된 디렉토리에 있는 db2icrt 명령어를 이용하여 생성합니다.

```
$ cd /opt/ibm/db2/V9.7/instance
$ ./db2icrt -u <fenced 사용자명> <인스턴스명>
```

3 인스턴스 사용자의 홈디렉토리에는 sqllib 라는 서브 디렉토리가 생성되어 DB2 9.7 엔진에서 제공하는 실행 모듈들이 링크됩니다.

```
$ ls <인스턴스 사용자의 홈디렉토리>
```

4 /etc/services 파일에 인스턴스와 관련된 4개의 포트가 추가됩니다.

```
$ grep <인스턴스명> /etc/services
DB2_<인스턴스명> 60000/tcp
DB2_<인스턴스명>_1 60001/tcp
DB2_<인스턴스명>_2 60002/tcp
DB2_<인스턴스명>_END 60003/tcp
```

```
IM FTSS
File Edit View Options Transfer Script Tools Help
[root@server1 instance]# ./db2icrt -u db2inst1 db2inst1
DB11070I  Program db2icrt completed successfully.

[root@server1 instance]# ./db2ilist
db2inst1
[root@server1 instance]# grep db2inst1 /etc/services
DB2_db2inst1c 50000/tcp
DB2_db2inst1 60000/tcp
DB2_db2inst1_1 60001/tcp
DB2_db2inst1_2 60002/tcp
DB2_db2inst1_END 60003/tcp
[root@server1 instance]#
```

Figure 0402A... db2icrt 명령어

Point



root 사용자가 db2iupt 명령어를 이용하여 기존 인스턴스의 환경을 갱신합니다. FixPack을 적용한 후에 인스턴스의 라이브러리 링크를 갱신하고자 할 때 사용합니다.

- 1 인스턴스의 환경을 갱신하기 위해서는 먼저 인스턴스를 중지해야 합니다.

```
$ login <인스턴스 사용자명>
$ db2stop force
```

- 2 root 사용자가 db2iupdt 명령어를 이용하여 인스턴스의 환경을 갱신합니다.

```
$ login root
$ cd /opt/ibm/db2/V9.7/instance
$ ./db2iupdt <인스턴스명>
$ ./db2ilist
```

- 3 복수개의 코드 레벨(DB2 엔진)이 존재 할 경우, 한 경로의 상위 코드 레벨에서 다른 경로에 설치된 낮은 코드 레벨로 인스턴스를 이동합니다

```
$ login root
$ cd /opt/ibm/db2/V9.7/instance
$ ./db2iupdt -D db2inst1
```

- 4 인스턴스 사용자가 db2start 명령어로 인스턴스를 재 기동하고, DB2 제품 정보를 확인합니다.

```
$ login <인스턴스 사용자명>
$ db2start
$ db2level
```

```
IM FTSS
File Edit View Options Transfer Script Tools Help
[db2inst1@server1 ~]$ db2stop force
2009-08-19 20:49:54      0      0      SQL1064N  DB2STOP 처리가 완료되었습니다.
SQL1064N  DB2STOP 처리가 완료되었습니다.
[db2inst1@server1 ~]$ su -
암호:
[root@server1 ~]# cd /opt/ibm/db2/V9.7/instance
[root@server1 instance]# ./db2iupdt db2inst1
DBI1070I  Program db2iupdt completed successfully.

[root@server1 instance]# su - db2inst1
[db2inst1@server1 ~]$ db2start
08/19/2009 20:51:26      0      0      SQL1063N  DB2START 처리가 완료되었습니다.
SQL1063N  DB2START 처리가 완료되었습니다.
[db2inst1@server1 ~]$ db2level
DB21085I  인스턴스 "db2inst1"은(는) "32"비트와 레벨 ID "08010107"의 DB2 코드
릴리스 "SQL09070"을(를) 사용합니다.
정보용 토큰은 "DB2 v9.7.0.0", "s090521", "LINUX1A3297" 및 FixPack "0"입니다.
제품이 "/opt/ibm/db2/V9.7"에 설치되었습니다.

[db2inst1@server1 ~]$
```

Figure 0403A... db2iupdt 명령어

Tip

기존 인스턴스에서 사용자가 구성한 환경은 유지됩니다.

Tip

기존 인스턴스에서 생성한 데이터베이스도 유효합니다.

Point



root 사용자가 db2idrop 명령어를 이용하여 불필요한 인스턴스를 제거합니다. 인스턴스를 제거하면, DB2 엔진 기능을 사용할 수 없으며, 인스턴스 사용자는 DB2와는 무관한 일반 사용자가 됩니다.

1

인스턴스를 제거하기 위해서는 먼저 인스턴스를 중지해야 합니다.

```
$ login <인스턴스 사용자명>
$ db2stop force
```

2

root 사용자가 db2idrop 명령어를 이용하여 인스턴스를 제거합니다.

```
$ login root
$ cd /opt/ibm/db2/V9.7/instance
$ ./db2idrop <인스턴스명>
$ ./db2ilist | grep <인스턴스명>
```

3

인스턴스 사용자의 홈 디렉토리에 존재하던 sqllib 라는 서브 디렉토리가 제거된 것을 확인할 수 있습니다.

```
$ ls <인스턴스 사용자의 홈디렉토리>/sqllib
```

4

/etc/services 파일에 추가된 포트는 자동으로 제거되지 않습니다.

```
$ grep <인스턴스명> /etc/services
DB2_<인스턴스명>60000/tcp
DB2_<인스턴스명>_1          60001/tcp
DB2_<인스턴스명>_2          60002/tcp
DB2_<인스턴스명>_END       60003/tcp
```

Tip

- 인스턴스를 제거해도 데이터베이스 설치 파일들은 제거되지 않으며, 사용했던 데이터베이스도 그대로 유지됩니다. 따라서 인스턴스를 다시 생성하여 데이터베이스를 재 카탈로그 하면 기존의 데이터베이스를 다시 사용할 수 있습니다.

Tip

- /etc/services 파일에 추가된 인스턴스와 관련된 TCP/IP 포트 정보는 설치 후에도 제거되지 않습니다. 인스턴스를 제거한 후에 사용자가 직접 제거하십시오.

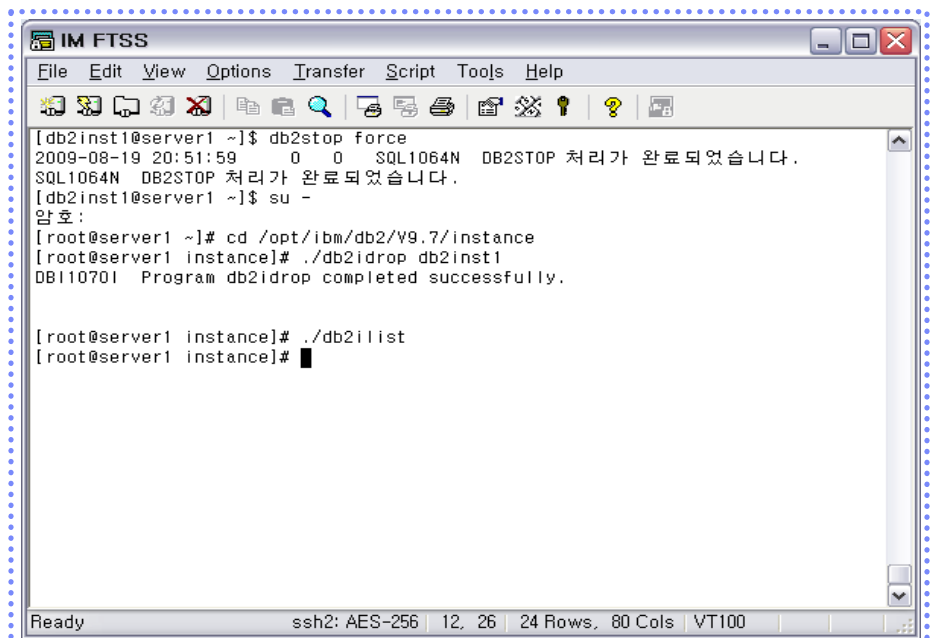


Figure 0404A db2idrop 명령어

Point



인스턴스 사용자는 db2start 명령어를 이용하여 인스턴스를 기동합니다. 인스턴스가 성공적으로 기동되면, db2sysc를 비롯한 여러 개의 프로세스 및 스레드가 생성되고, 인스턴스 수준의 공유 메모리가 할당됩니다.

Tip

단순히 db2 또는 db2bp라는 이름을 가진 프로세스는 DB2 세션을 위한 프로세스입니다. db2 라는 글자로 시작하지만, DB2 엔진과는 무관합니다.

Tip

DB2 9.5부터는 프로세스 방식이 아닌 스레드 방식을 이용합니다. db2pd -edu 로 DB2 엔진과 관련된 스레드를 확인할 수 있습니다.

1 인스턴스 사용자로 로그인합니다.

```
$ login <인스턴스 사용자명>
```

2 인스턴스 기능을 시작하려면 db2start 명령어를 이용합니다.

```
$ db2start
```

3 db2sysc 를 비롯한 여러 개의 프로세스가 생성됩니다. DB2 엔진과 관련된 프로세스의 이름은 대부분 db2 라는 글자로 시작됩니다. 프로세스 이름의 마지막 부분에 나오는 숫자는 데이터베이스 파티션 번호입니다. DPF인 경우에는 동일한 엔진 프로세스가 데이터베이스 파티션별로 생성됩니다.

```
$ ps -ef | grep <인스턴스 사용자명>
$ db2pd -edu
```

4 인스턴스 공유 메모리가 할당된 것을 확인할 수 있습니다.

```
$ db2mtrk -i
인스턴스용 메모리
monh      other
16.0K     1.3M
```

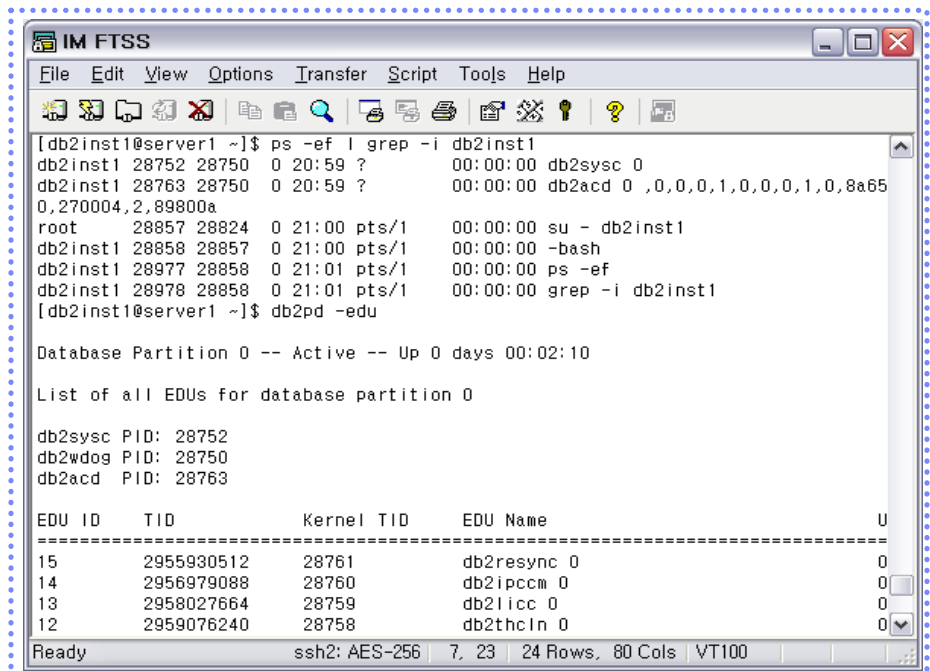


Figure 0405A... db2start 명령어

Point



인스턴스 사용자는 db2stop 명령어를 이용하여 인스턴스를 중지시킬 수 있습니다. 먼저 데이터베이스에 접속된 모든 응용프로그램들은 중지되어야 합니다. 인스턴스와 관련된 모든 프로세스가 제거되고, 인스턴스 수준의 공유 메모리도 해제됩니다.

Tip

데이터베이스에 접속된 응용프로그램이 존재하면 SQL1025N 오류가 반환되고, 인스턴스는 중지되지 않습니다.

Tip

db2stop 명령어가 성공적으로 완료된 이후에도 인스턴스 사용자와 관련된 DB2 프로세스가 비정상적으로 남아있다면, 인스턴스 강제 중지를 실행합니다. (db2_kill 명령어)

- 1 인스턴스 중지는 인스턴스 사용자가 담당합니다.

```
$ login <인스턴스 사용자명>
```

- 2 인스턴스를 중지하기 전에 데이터베이스에 접속된 응용프로그램을 모두 중지시켜야 합니다. db2stop 명령어에서 force 옵션을 이용하면, 접속된 응용프로그램을 모두 강제로 중지하고 인스턴스를 중지하게 합니다.

```
$ db2stop force
```

- 3 인스턴스가 중지되면, 인스턴스 사용자의 DB2 프로세스는 모두 제거됩니다.

```
$ ps -ef | grep <인스턴스 사용자명>
$ db2pd -edu
```

- 4 db2bp 라는 이름의 프로세스는 db2stop 명령어를 실행해도 제거되지 않습니다. db2bp는 DB2 세션용 프로그램을 위한 별도의 프로세스이므로 kill 명령어로 제거해도 무방합니다.

```
$ kill -9 <db2bp 프로세스의 pid>
```

- 5 인스턴스 공유 메모리가 해제된 것을 확인할 수 있습니다.

```
$ db2mtrk -i
인스턴스가 시작되지 않음.
```

```

[db2inst1@server1 ~]$ db2 force applications all
DB20000I  FORCE APPLICATION 명령이 완료되었습니다.
DB21024I  이 명령은 비동기식으로 실행되므로 즉시 적용되지 않을 수도 있습니다.

[db2inst1@server1 ~]$ db2 terminate
DB20000I  TERMINATE 명령이 완료되었습니다.
[db2inst1@server1 ~]$ db2stop force
2009-08-19 21:02:41      0      0      SQL1064N  DB2STOP 처리가 완료되었습니다.
SQL1064N  DB2STOP 처리가 완료되었습니다.
[db2inst1@server1 ~]$ ps -ef | grep -i db2inst1
root      28857 28824      0 21:00 pts/1      00:00:00 su - db2inst1
db2inst1 28858 28857      0 21:00 pts/1      00:00:00 -bash
db2inst1 29027 28858      0 21:02 pts/1      00:00:00 ps -ef
db2inst1 29028 28858      0 21:02 pts/1      00:00:00 grep -i db2inst1
[db2inst1@server1 ~]$ db2pd -edu
Unable to attach to database manager on partition 0.
Please ensure the following are true:
- db2start has been run for the partition.
- db2pd is being run on the same physical machine as the partition.
- DB2NODE environment variable setting is correct for the partition
  or db2pd -dbp setting is correct for the partition.
Another possibility of this failure is the Virtual Address Space Randomization i
s currently enabled on this system.
[db2inst1@server1 ~]$
  
```

Figure 0406A • db2stop 명령어

Point



db2stop 명령어를 이용하여 인스턴스를 정상적으로 중지할 수 없으면, db2_kill 명령어를 이용하여 강제로 중지할 수 있습니다. Ipclean 명령어와 ipcs 명령어를 이용하여 DB2가 사용한 리소스를 명시적으로 정리하도록 합니다.

Tip

db2_kill 은 비상시에 사용하는 명령어로 정상적인 경우에 사용하는 것은 권장하지 않습니다. 정상적으로 인스턴스를 중지할 수 있다면 db2stop 명령어를 사용합니다.

Tip

Ipclean 명령어를 이용한 후에도 DB2가 사용한 리소스가 남아있는 경우에는 ipcrm 명령어를 이용하여 제거합니다.

Tip

db2_kill 명령어를 실행한 후에도 DB2와 관련된 프로세스가 남아있는 경우에는 kill 명령어로 제거합니다.

1 인스턴스 중지는 인스턴스 사용자가 담당합니다.

```
$ login <인스턴스 사용자명>
```

2 인스턴스를 중지하기 전에 데이터베이스에 접속된 응용프로그램을 모두 중지시켜야 합니다.

```
$ db2 force applications all
```

3 db2_kill 명령을 사용하여 인스턴스를 강제로 중지시킵니다. 인스턴스를 비정상적으로 중지한 경우에는 ipclean 명령어를 이용하여 DB2가 사용하던 리소스를 정리하도록 합니다.

```
$ db2_kill
$ ipclean -a
$ ipcs -a | grep <인스턴스 사용자명>
```

4 인스턴스 사용자의 DB2 프로세스가 모두 제거된 것을 확인합니다.

```
$ ps -ef | grep <인스턴스 사용자명>
$ db2pd -edu
```

5 db2start 명령어를 이용하여 인스턴스를 재 시작합니다.

```
$ db2start
```

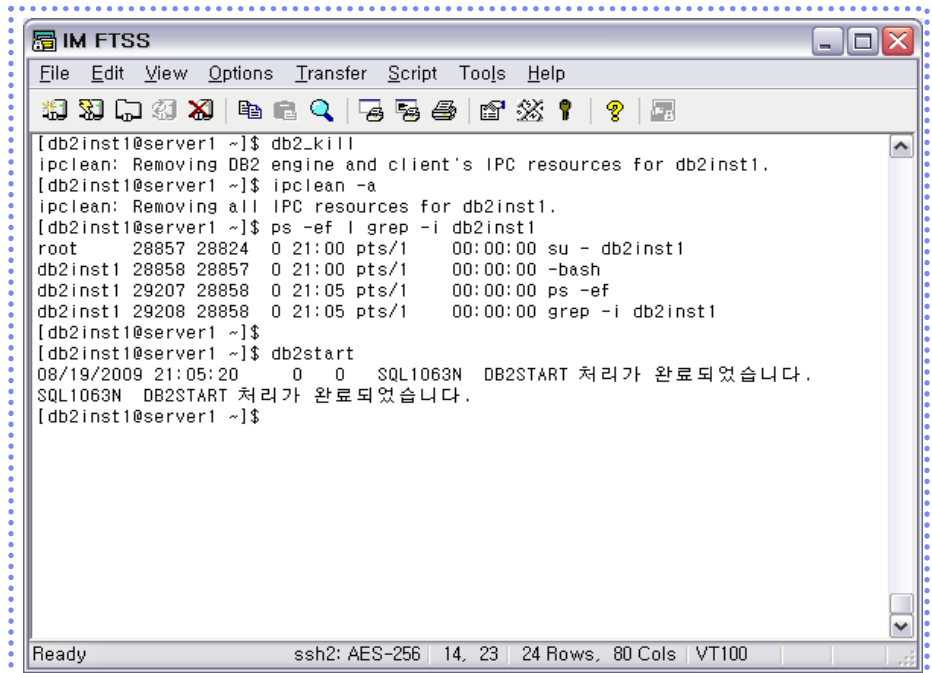


Figure 0407A- db2_kill 명령어

Point



인스턴스 구성 파일에 저장되는 인스턴스 구성 변수를 이용하여 인스턴스 별로 고유한 환경을 구성합니다. 인스턴스 구성 파일은 get dbm cfg, update dbm cfg, reset dbm cfg 명령어를 이용하여 인스턴스 사용자가 관리합니다.

Tip

인스턴스 구성 파일을 DBM 구성 파일이라고 하고, 인스턴스 구성 변수는 DBM 구성 변수라고도 합니다.

Tip

show detail 옵션은 attach 명령어로 인스턴스에 접속해야 사용할 수 있습니다.

Tip

값을 변경하는 즉시 반영되는 인스턴스 구성 변수를 '온라인 구성 가능한 인스턴스 구성 변수'라고 합니다.

1

인스턴스 구성 파일은 인스턴스 사용자가 관리합니다.

```
$ login <인스턴스 사용자명>
```

2

get dbm cfg 명령어를 이용하여 인스턴스 구성 파일을 확인합니다. show detail 옵션을 이용하면 설정값과 현재값을 확인할 수 있습니다.

```
$ db2 attach to <인스턴스명>
$ db2 get dbm cfg show detail | grep <구성변수명>
$ db2 detach
```

3

인스턴스를 재 기동하지 않고 즉시 반영되는 구성 변수의 값을 변경할 때는 attach 명령어로 인스턴스에 접속하여 update dbm cfg 명령어를 사용합니다.

```
$ db2 attach to <인스턴스명>
$ db2 update dbm cfg using <구성변수명> <값>
$ db2 detach
```

4

reset dbm cfg 명령어는 모든 인스턴스 구성 변수를 초기값으로 변경합니다. 일부 인스턴스 구성 변수는 인스턴스를 재 기동해야 변경한 값이 반영됩니다.

```
$ db2 reset dbm cfg
$ db2stop force
$ db2start
```

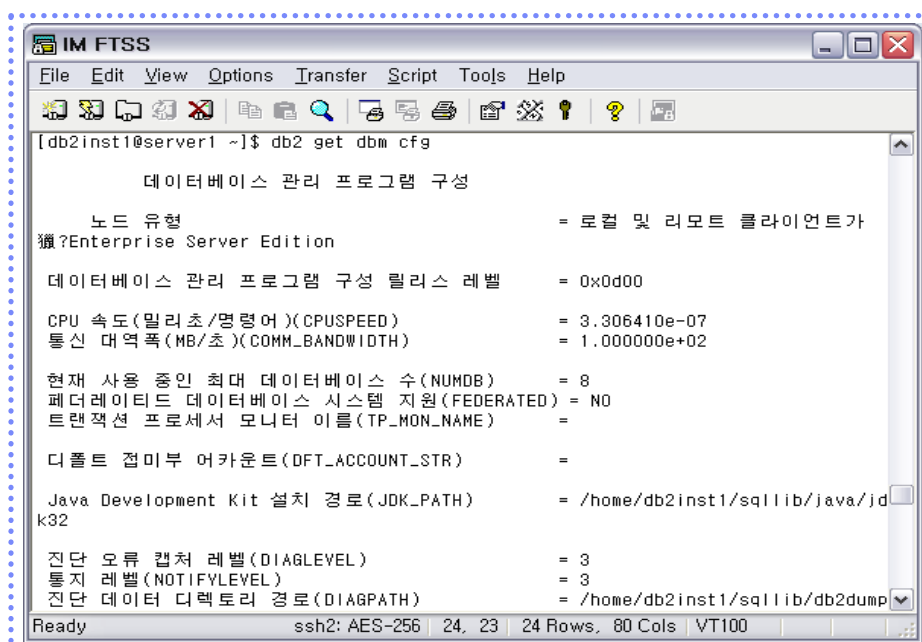


Figure 0408A 인스턴스 구성 변수

Point



DB2 사용자는 DB2INSTANCE라는 환경 변수를 이용하여 원하는 인스턴스를 선택할 수 있습니다. DB2 명령을 실행하여 인스턴스의 서비스를 받으려면 PATH 등의 기본적인 환경 변수의 설정이 필요합니다.

Tip

Windows 에서 인스턴스를 지정 할 때에는 제어판 또는 DB2 명령 창을 이용하여 DB2INSTANCE 환경 변수를 설정합니다.

1 DB2 사용자로 로그인합니다.

```
$ login <DB2 사용자명>
```

2 DB2INSTANCE 라는 OS 환경 변수 명을 이용하여 인스턴스 명을 지정합니다.

```
$ export DB2INSTANCE=<인스턴스명>
$ echo $DB2INSTANCE
```

3 DB2 명령을 실행하려면 PATH 등의 기본적인 환경 변수 설정이 필요합니다.

```
$ export PATH=$PATH:<인스턴스 사용자 홈디렉토리>/sqllib/bin
```

4 DB2INSTANCE, PATH 등의 환경 변수를 간단하게 설정하려면, 인스턴스 사용자의 홈 디렉토리에 생성된 sqllib 라는 서브 디렉토리에 있는 db2profile 을 실행하면 됩니다.

```
$ . <인스턴스 사용자 홈디렉토리>/sqllib/db2profile
$ echo $DB2INSTANCE
```

5 다른 인스턴스를 지정하려면, 원하는 인스턴스 사용자의 db2profile을 선택하여 실행합니다.

```
$ . <원하는 인스턴스 사용자 홈디렉토리>/sqllib/db2profile
$ echo $DB2INSTANCE
```

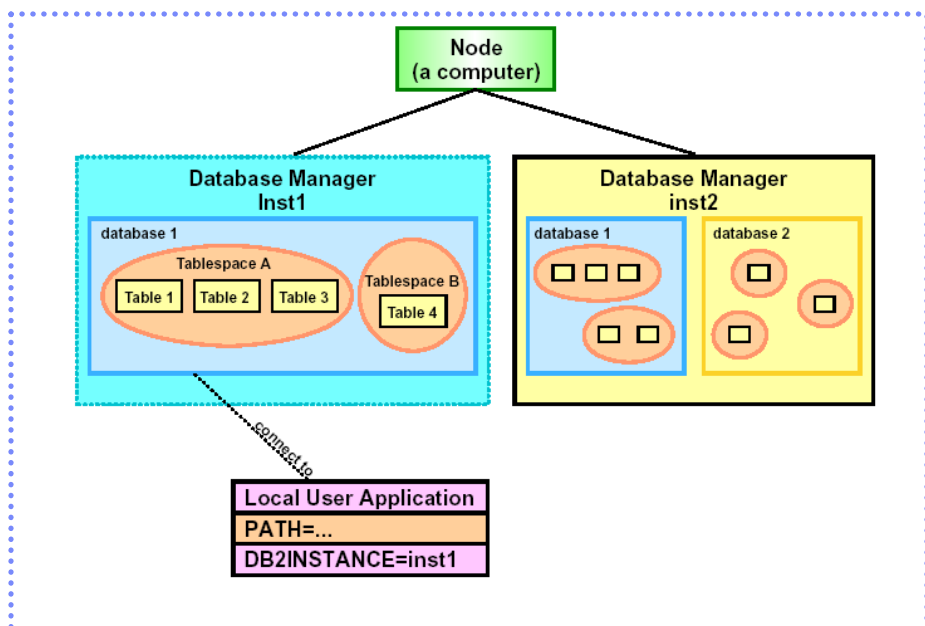


Figure 0409A DB2INSTANCE 환경 변수

Point



DB2 인스턴스에만 적용되는 시스템 환경 변수를 DB2 레지스터리 변수라고 합니다. DB2 레지스터리 변수는 인스턴스 사용자가 db2set 명령어를 이용하여 관리하며, OS 환경 변수가 아니므로 시스템의 재기동이 필요하지 않습니다.

Tip

별도의 옵션이 없으면, 현재 세션이 지정한 인스턴스에만 적용됩니다.

Tip

레지스터리 변수의 값을 설정할 때는 = (등호) 전후에 공백 문자가 없도록 주의합니다.

Tip

지원되는 모든 레지스터리 변수는 db2set -lr 명령어로 확인 가능하며 그에 대한 자세한 설명은 정보 센터를 참조하십시오.

1 DB2 레지스터리 변수는 인스턴스 사용자가 관리합니다.

```
$ login <인스턴스 사용자>
```

2 db2set 명령어의 기본 형식은 db2set <레지스터리 변수명>=<값> 입니다.

```
$ db2set <레지스터리변수>=<값>
```

3 레지스터리 변수의 설정을 취소하려면 <값> 항목을 입력하지 않으면 됩니다.

```
$ db2set <레지스터리변수>=
```

4 - i 옵션을 사용하면 특정한 인스턴스에만 변수가 적용되게 합니다. - g 옵션은 현재의 서버에 존재하는 모든 인스턴스에 대해 변수를 적용합니다.

```
$ db2set -i <인스턴스명> <레지스터리변수>=<값>
```

```
$ db2set -g <레지스터리변수>=<값>
```

```
$ db2set -all
```

5 대부분의 레지스터리 변수는 설정 후에 인스턴스의 재기동이 필요합니다.

```
$ db2stop force
```

```
$ db2start
```

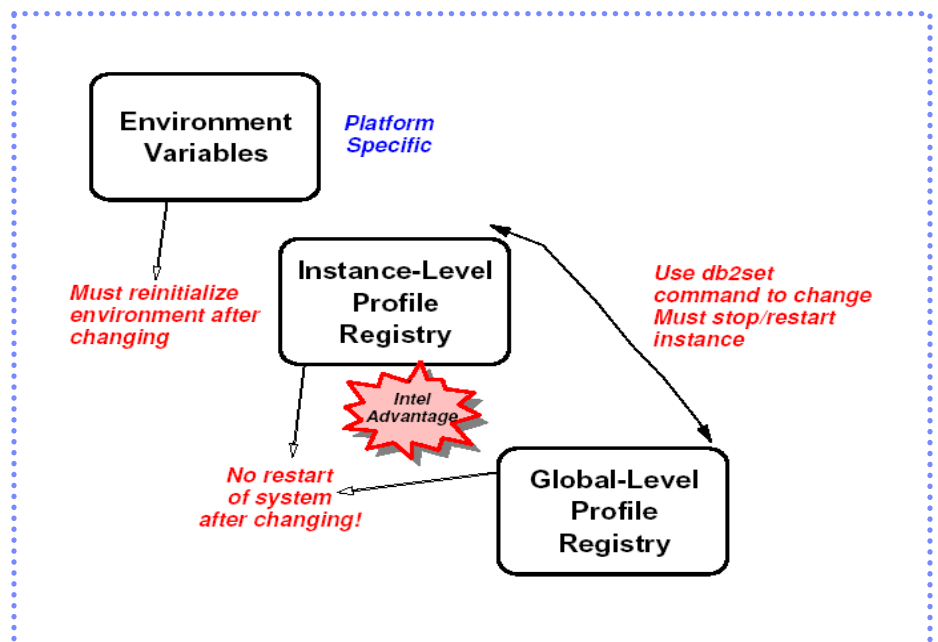


Figure 0410A DB2 레지스터리 변수

Point



제어 센터 등의 GUI 도구를 이용하여 DB2 를 운영할 때만 필요한 특수 인스턴스로서 한 서버에 한 개만 존재할 수 있습니다. dasCRT, daslist, dasupdt, dasdrop 등의 명령어로 관리하며, db2admin 명령어로 시작되고 중지됩니다.

Tip

DB2 9.7부터 DAS가 Deprecate되었으며, Optim Database Administrator로 기능 이전되었습니다.

1

UNIX에서 DAS 인스턴스를 생성하려면 OS 그룹과 사용자 계정이 필요합니다.

```
$ login root
$ mkgroup <일자 그룹명>
$ mkuser pgrp=<일자 그룹명> <DAS용 인스턴스 사용자명>
$ passwd <DAS용 인스턴스 사용자명>
```

2

DB2 제품이 설치된 디렉토리에 있는 dasCRT 명령어를 이용하여 생성합니다.

```
$ cd /opt/ibm/db2/V9.7/instance
$ ./dasCRT <DAS용 인스턴스명>
$ ./daslist
```

3

DAS용 인스턴스 사용자는 db2admin 명령어로 DAS를 시작하거나 중지합니다.

```
$ login <DAS 인스턴스명>
$ db2admin start
$ db2admin stop
```

4

DAS를 제거하려면 dasdrop 명령어를 이용합니다.

```
$ ./dasdrop dasusr1
```

Tip

DAS용 인스턴스 사용자 계정의 일자 그룹은 고유한 그룹명으로 지정하는 것이 권장됩니다.

Tip

DAS는 생성시에 자동으로 시작됩니다.

Tip

FixPack 적용 후에는 dasupdt 명령어를 이용하여 DAS를 갱신합니다.

Tip

Windows에서는 설치 시에 DB2DAS00 라는 이름의 DAS용 인스턴스가 자동으로 생성됩니다.

```
root@dmtest1:/usr/opt/db2_08_01/instance> dasCRT dasusr1
SQL4406W The DB2 Administration Server was started successfully.
DB11070I Program dasCRT completed successfully.

root@dmtest1:/usr/opt/db2_08_01/instance> ./daslist
dasusr1
root@dmtest1:/usr/opt/db2_08_01/instance> su - dasusr1
$ db2admin stop
SQL4407W The DB2 Administration Server was stopped successfully.
$ db2admin start
SQL4406W The DB2 Administration Server was started successfully.
$ ps -ef | grep dasusr1
dasusr1 5578930      1   0 23:17:59      -   0:00 /home/dasusr1/das/adm/db2da
srrm
dasusr1 5890114      1   0 23:17:38      -   0:00 /home/dasusr1/das/bin/db2fm
d -i dasusr1 -m /home/dasusr1/das/lib/libdb2dasgcf.a
dasusr1 6332434 7569428   1 23:18:11 pts/2    0:00 grep dasusr1
dasusr1 6660196 7569428   5 23:18:11 pts/2    0:00 ps -ef
dasusr1 7569428 6521046   0 23:17:43 pts/2    0:00 -ksh
$
```

Figure 0411A... DAS 생성과 기동

Point



DB2 서버 제품은 2개의 연속적인 TCP/IP 서비스 포트를 이용하여 UNIX, Windows 등의 원격 클라이언트 제품으로부터의 접속을 지원합니다.

Tip

- 포트 번호는 50000번 이상이 권장되며, 서비스 명은 임의로 정의합니다.

1

원격 클라이언트와 통신하려면 /etc/services 파일에 연속된 번호를 가진 2개의 TCP/IP 포트를 추가합니다.

```
$ login root
$ grep <인스턴스명> /etc/services
<인스턴스명>c          50000/tcp
<인스턴스명>i          50001/tcp
```

2

DB2 레지스터리 변수인 DB2COMM에 TCPIP를 지정합니다.

```
$ login <인스턴스 사용자>
$ db2set DB2COMM=TCPIP
```

3

/etc/services 파일에서 지정한 첫 번째 포트의 서비스 명을 DBM 구성 변수인 SVCENAME 에 지정하고, 인스턴스를 재 기동합니다.

```
$ db2 update dbm cfg using SVCENAME <서비스 명>
$ db2stop force
$ db2start
```

4

TCP/IP용 통신 리스너 프로세스인 db2tcpcm이 생성된 것을 확인합니다.

```
$ db2pd -edu | grep -i db2tcpcm
```

Tip

- Personal Edition은 클라이언트를 지원하지 않습니다.

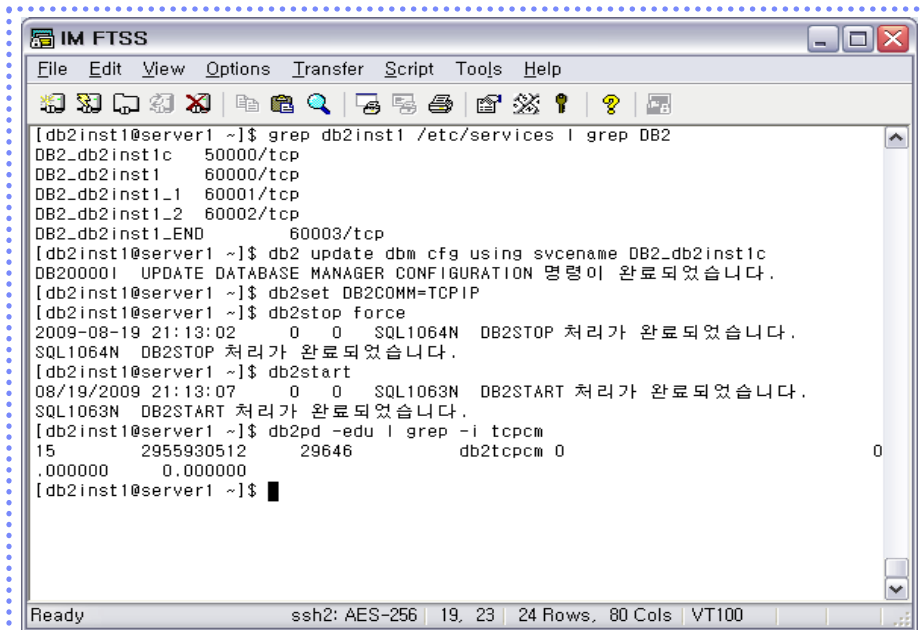


Figure 0412A... 원격 클라이언트를 위한 TCP/IP 포트 설정

Point



DB2 인스턴스는 기본적으로 한 개의 데이터베이스 파티션으로 구성되어 있습니다. DPF 옵션을 이용한 다중 데이터베이스 파티션을 구성하면, 병렬 데이터베이스를 구성할 수 있습니다.

- 1 서버의 호스트 명을 확인합니다. 호스트 명은 hostname 명령어의 실행 결과로 확인하며, /etc/hosts 파일의 정의 내용과 반드시 동일해야 합니다.

```
$ login <인스턴스 사용자명>
$ hostname
$ cat /etc/hosts
```

- 2 DPF는 rsh를 이용하므로 .rhosts 파일이 적절하게 생성되어야 합니다.

```
$ vi rhosts
```

- 3 인스턴스를 중지한 후에 db2nodes.cfg 파일을 수정합니다.

```
$ db2stop force
$ cd <인스턴스 사용자의 홈디렉토리>/sqllib
$ vi db2nodes.cfg
```

- 4 인스턴스를 재 기동하면, 각 파티션별로 인스턴스용 프로세스가 생성됩니다. 프로세스 이름 뒤에 표시된 번호는 데이터베이스 파티션 번호입니다.

```
$ db2start
$ ps -ef | grep <인스턴스명> | grep db2sysc
```

Tip

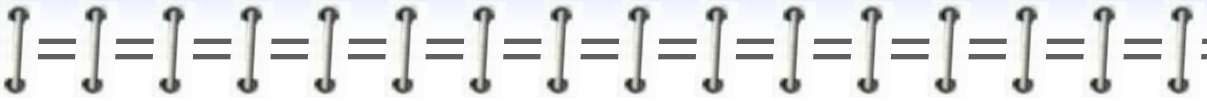
db2nodes.cfg 파일은 인스턴스가 중지된 상태에서만 수정이 가능합니다.

Tip

기본적으로 한 개의 물리적인 서버에 4개의 논리적인 데이터베이스 파티션을 지원합니다. 특정 서버의 논리적 파티션의 최대 개수가 4개를 초과하면 /etc/services 파일에 정의된 TCP/IP 포트의 설정을 변경해야 합니다.

```
IM FTSS
File Edit View Options Transfer Script Tools Help
[db2inst1@server1 ~]$ hostname
server1
[db2inst1@server1 ~]$ cat .rhosts
server1 db2inst1
[db2inst1@server1 ~]$ cat sqlib/db2nodes.cfg
0 server1 0
1 server1 1
2 server1 2
3 server1 3
[db2inst1@server1 ~]$ db2start
08/19/2009 22:50:46 1 0 SQL1063N DB2START 처리가 완료되었습니다.
08/19/2009 22:50:47 0 0 SQL1063N DB2START 처리가 완료되었습니다.
08/19/2009 22:50:57 2 0 SQL1063N DB2START 처리가 완료되었습니다.
08/19/2009 22:51:15 3 0 SQL1063N DB2START 처리가 완료되었습니다.
SQL1063N DB2START 처리가 완료되었습니다.
[db2inst1@server1 ~]$ ps -ef | grep -i db2sysc
db2inst1 9467 9430 1 22:50 ? 00:00:01 db2sysc 0
db2inst1 9469 9431 1 22:50 ? 00:00:01 db2sysc 1
db2inst1 9694 9509 2 22:50 ? 00:00:01 db2sysc 2
db2inst1 9767 9761 3 22:51 ? 00:00:01 db2sysc 3
db2inst1 9796 3474 0 22:51 pts/1 00:00:00 grep -i db2sysc
[db2inst1@server1 ~]$
```

Figure 0413A 다중 데이터베이스 파티션 구성



Memo ▶

