

Q & A (vol.6)

dbspace와 chunk를 추가하는 방법

Informix를 사용한 지 2년 정도 된 사용자입니다. 사용하다 보니 데이터 저장공간이 모자라는 것 같아 하드 디스크를 새로 구입하였습니다. Informix 데이터를 위한 공간을 추가하는 방법을 Q1 알고 싶습니다.

A Informix 는 데이터를 dbspace 라 불리는 논리 영역에 저장합니다. 그리고 이 dbspace 는 chunk 라는 물리적 디스크로 구성됩니다. 즉, 어떤 Informix 시스템에도 적어도 하나의 dbspace 가 반드시 있어야 하며, 초기 dbspace(root dbspace)는 사용자에게 의해 생성되는 데이터 뿐만 아니라 Informix 가 필요한 정보들을 기록하는 장소이기도 합니다. 시스템에 아무리 많은 하드 디스크가 연결되어 있다 하더라도 Informix 에서 관리하는 dbspace 로 할당된 디스크 외에는 사용할 수 없습니다. 따라서 먼저 dbspace 의 가용 영역을 알아본 후에 실제로 그 영역이 많이 사용되었다면 dbspace 를 추가하거나 이미 사용되고 있는 dbspace 에 chunk 를 추가할 수 있습니다.

먼저 할당된 dbspace 의 크기 정보를 알아보기 위한 명령은 onstat -d 입니다.

```
kor-k400:/bigfs> onstat -d

Informix OnLine Version 7.24.UC1 -- On-Line -- Up 6 days 07:59:33 -- 17408 Kbytes

Dbspaces

address  number  flags  fchunk  nchunks  flags  owner  name
c4a3c108 1      1      1        1        N      Informix rootdbs

1 active, 2047 maximum

Chunks

address  chk/dbs  offset  size    free    bpages  flags  pathname
c4a3c178 1    1    0    150000 113498      PO-    /post02/CASE_DB/case_root

1 active, 2047 maximum
```

현재 위 시스템에는 rootdbs 라고 이름 붙여진 dbspace 하나가 할당되어 있고, 그 dbspace 에 실제 하드 디스크는 /post02/CASE_DB/case_root 로 할당되어 있으며, 크기는 150,000 페이지(페이지 크기가 2KB 인 시스템에서는 300,000KB, 이하 페이지 크기가 2KB 라고 가정)이며 현재 남아있는 공간이 113,498 페이지입니다. 이 시스템에 또 하나의 데이터베이스를 만들기 위해 dbspace 를 추가하기를 원한다면 onspaces 명령이나 onmonitor 의 Dbpace

옵션의 Create 메뉴를 사용할 수 있습니다.

```
kor-k400:/bigfs> onspaces -c -d add_dbs -p /bigfs/  
chunk1 -o 0 -s 10000
```

```
Verifying physical disk space, please wait ...
```

```
DBspace/BLOBspace successfully added.
```

```
kor-k400:/bigfs>
```

위 명령은 /bigfs/chunk1 이라는 file 로 offset 은 0 이며 크기는 10,000 KB 인 chunk 를 포함하는 add_dbs 라는 dbspace 를 생성하는 명령입니다. 성공적으로 추가되었으므로 다시 onstat -d 로 확인해보면 다음과 같습니다.

```
kor-k400:/bigfs> onstat -d
```

```
Informix OnLine Version 7.24.UC1 -- On-Line -- Up 6 days 08:30:30 -- 17408 Kbytes
```

```
Dbspaces
```

address	number	flags	fchunk	nchunks	flags	owner	name
c4a3c108	1	1	1	1	N	Informix	rootdbs
c4a6b6c0	2	1	2	1	N	Informix	add_dbs

```
2 active, 2047 maximum
```

```
Chunks
```

address	chk/dbs	offset	size	free	bpages	flags	pathname
c4a3c178	1 1	0	150000	113498		PO-	/post02/CASE_DB/case_root
c4a6b730	2 2	0	5000	4947		PO-	/bigfs/chunk1

```
2 active, 2047 maximum
```

위에서 확인하는 바와 같이 add_dbs 는 2 번 dbspace 가 되며 chunk 도 2 번 dbs(dbspace)에 포함되는 2 번 chunk 로 추가되었습니다. 여기서 add_dbs 에

또 하나의 chunk 를 추가하고자 한다면 역시 onspaces 명령으로 추가할 수 있습니다.

```
kor-k400:/bigfs> onspaces -a add_dbs -p /bigfs/chunk2 -o 0  
-s 5000
```

```
Verifying physical disk space, please wait ...
```

```
Chunk successfully added.
```

```
kor-k400:/bigfs>
```

위의 onspaces 명령은 add_dbs 에 /bigfs/chunk2 라는 file 을 offset 은 0, 크기는 5,000KB 의 chunk 를 추가시키는 명령입니다. 역시 chunk 가 제대로 추가되었는지의 여부는 onstat -d 로 확인할 수 있습니다.

```
kor-k400:/bigfs> onstat -d
```

```
Informix OnLine Version 7.24.UC1 -- On-Line -- Up 6 days 08:46:43 -- 17408 Kbytes
```

```
Dbspaces
```

address	number	flags	fchunk	nchunks	flags	owner	name
c4a3c108	1	1	1	1	N	Informix	rootdbs
c4a6b6c0	2	1	2	2	N	Informix	add_dbs

```
2 active, 2047 maximum
```

```
Chunks
```

address	chk/dbs	offset	size	free	bpages	flags	pathname
c4a3c178	1 1	0	150000	113498		PO-	/post02/CASE_DB/case_root
c4a6b730	2 2	0	5000	4947		PO-	/bigfs/chunk1
c4a61548	3 2	0	2500	2497		PO-	/bigfs/chunk2

```
3 active, 2047 maximum
```

2번 db에 3번 chunk를 추가되었으면 화이한 수 있습니다

만일 기존에 사용하고 있던 rootdbs에 또 다른 chunk를 추가하고자 한다면, 위에서 사용한 방법으로 가능합니다.

```
kor-k400:/bigfs> onspaces -a rootdbs -p /bigfs/rootchunk -o 0
```

```
c4a61548 3 2 0 2500 2497 PO- /bigfs/chunk2
c4a61620 4 1 0 2500 2497 PO- /bigfs/rootchunk
4 active, 2047 maximum
```

이제 위에서 추가한 chunk 가 1 번 dbs 에 4 번 chunk 로 추가되었음을 확인할 수 있을 것입니다.

Raw Device를 dbspace의 chunk로 사용할 때의 장점과 주의 사항

dbspace의 chunk를 raw device로 사용하려고 합니다. 일반적으로 Informix에서는 raw device를 사용하도록 권유하는 것으로 알고 있는데, raw device를 사용할 때의 장점과 주의할 사항은 무엇입니까?

Q2

```
kor-k400:/bigfs> chgrp Informix /dev/vg02/rlvol1
```

```
kor-k400:/bigfs> ls -l /dev/vg02
```

```
crw-rw---- 1 Informix Informix 64 0x020001 Dec 10 21:23 rlvol1
```

이렇게 준비해 놓은 Raw Device 는 onspaces 명령으로 새로운 dbospace 로 추가 되거나 기존의 dbospace 의 또 다른 chunk 로 추가될 수 있습니다. 이런 Raw Device 를 onspaces 명령을 이용하여 chunk 로 사용할 때에는 해당 Disk Partition 의 실제 크기에 주의하여야 하며, Unix File 을 사용하는 Cooked Device 와는 달리 약간의 Offset 을 두는 것이 좋습니다.

onospace 명령은 앞의 "Q1"의 내용을 참고하십시오.

많은 양의 데이터 로드 시 주의해야 할 사항

데이터를 로드하기 전 로깅 모드를 No Logging 으로 변경한 후 데이터 로드가 끝난 후 다시 Buffered Logging 등 기존의 로깅 모드로 변경시키면 됩니다.

로깅 모드를 변경하는 명령은 다음과 같습니다.

```
kor-k400:/bigfs> ontape -s -N stores7
```

```
Archive to tape device '/dev/null' is complete.
```

```
TAPESIZE 300000      # Maximum amount of data to put  
  
                    # on tape(Kbytes)  
  
... ..
```

이렇게 데이터베이스의 로깅 모드를 바꾼 후에 작업을 하게 되면, 로드와 관련한 내역을 더 이상 논리로그 파일에 기록하지 않을 것이며, 따라서 논리 로그가 Full 이 된다면 Long Transaction 에 걸려서 작업이 멈추는 일은 없을 것입니다.

로드가 끝난 후에는 반드시 원래 로깅 모드로 돌려야 함을 잊어서는 안됩니다. No Logging 모드의 데이터베이스에서는 트랜잭션(BEGIN WORK, COMMIT WORK, ROLLBACK WORK 등의 문장)을 사용할 수 없으므로 기존에 사용 중인 모드로 되돌려 놓지 않는다면 응용프로그램에서 오류가 발생할 수도 있습니다.

```
kor-k400:/bigfs> ontape -s -B stores7
```

```
Archive to tape device '/dev/null' is complete.
```

```
Program over.
```

```
kor-k400:/bigfs>
```

stores7 이라는 데이터베이스를 다시 Buffered Logging 모드로 변경하는 명령입니다.

이렇게 작업을 완료하였으면 \$InformixDIR/etc/\$ONCONFIG 파일의 TAPEDEV 파라미터 값도 원래대로 되돌려 놓는 것에도 주의를 하셔야 합니다.

mirror chunk로부터 primary chunk를 복구하는 방법

디스크 장애로 인하여 특정 dbspace 의 chunk 가 down 되었습니다. 다행히 이 dbspace 에는 mirroring 이 되어 있어서 system 사용에는 문제가 없습니다. 장애가 생긴 디스크를 다시 교체한 후에 mirror chunk 로부터 primary chunk 를 복구하려 합니다.

Q4

```

Dbspaces
address  number  flags  fchunk  nchunks  flags  owner  name
ce84e108 1      2      1      1      M      Informix rootdbs

  1 active, 2047 maximum

Chunks
address  chk/dbs  offset  size    free    bpages  flags  pathname
ce84e178 1    1    0      5000   2221    PD-    /bigfs/chunk
ce84e250 1    1    0      5000   0      MO-    /post01/byrhee/mchunk

  1 active, 2047 maximum

```

primary chunk 의 상태가 "PD"임을 알 수 있고, 이는 "Primary Down"을 의미합니다.

이 경우, 위에서도 말한 바와 같이 mirror chunk 를 가지고 복구할 수 있습니다. 우선 장애가 발생한 chunk (여기서는 /bigfs/chunk)를 사용 가능하도록 한 후에 onspaces 명령을 사용하여 복구할 수 있습니다.

```

kor-k400:/bigfs> onspaces -s rootdbs -p
/bigfs/chunk -o 0 -O

WARNING: bringing a down chunk into recovery state.

Do you really want to continue? (y/n)y

Verifying physical disk space, please wait ...

Chunk status successfully changed.

```

onspaces 명령에서 -s 옵션을 주어 rootdbs 의 chunk 중 /bigfs/chunk 의 상태(status)를 OnLine (-O 옵션)으로 변경하라는 내용입니다. 이렇게 작업을 명령한 후 success 명령을 받으면 다시 onstat -d 로 확인해 볼 수 있습니다.

```

kor-k400:/bigfs> onstat -d

Informix OnLine Version 7.24.UC1 -- On-Line -- Up 00:37:49 -- 9096 Kbytes

Dbspaces

address  number  flags  fchunk  nchunks  flags  owner  name
ce84e108 1       2      1       1        M      Informix rootdbs

    1 active, 2047 maximum

Chunks

address  chk/dbs  offset  size    free    bpages  flags  pathname
ce84e178 1   1   0    5000  2221    PO-   /bigfs/chunk
ce84e250 1   1   0    5000   0      MO-   /post01/byrhee/mchunk

    1 active, 2047 maximum

```

따라서 미러링을 한 dbspace 라면 디스크 장애가 생겼을 때에도 mirror chunk 까지 모두 장애가 발생하지 않는 한 서비스가 중단되지 않으며, 장애가 발생한 디스크를 복구할 수 있습니다. 여기서 주의해야 할 점은 한 쪽 디스크에 발생했을 때 mirror chunk 까지 모두 장애가 발생하지 않도록 primary chunk 로 사용하는 디스크와 mirror chunk 로 사용하는 디스크를 서로 분리시켜 놓아야 한다는 점입니다. 물리적으로 같은 디스크에 primary 와 mirror chunk 를 같이 둔다면 한 쪽 디스크 장애만으로도 두 chunk 를 동시에 액세스 못할 수 있으므로, 이런 경우라면 미러링의 효과를 전혀 볼 수 없을 것입니다.

이렇게 장애가 난 디스크를 복구하는 방법은 onspaces 명령 외에 onmonitor 의 Dbspaces 메뉴의 Status 옵션을 선택하여서도 가능합니다.

MS ASP를 이용하여 프로그램할 때

MS ASP를 이용하여 web service를 하려고 합니다. Informix Database에 연결하기 위한 Connection string을 어떻게 써야 하나요?

Q5

A 우선 Informix 데이터베이스에 접근하기 위해서 ClientSDK 나 Iconnect 라는 두 가지 중 한 가지를 설치하여야 합니다. ClientSDK 는 개발을 위한 것이고, IConnect 는 데이터베이스와 연결하기 위한 프로그램들이 들어 있습니다. 이 두 가지 중 한 가지를 설치하고 나면 두 가지의 ODBC driver 가 인스톨 되었음을 ODBC manager 를 통해서 볼 수 있습니다. 하나는 CLI 2.8 이고 또 다른 하나는 Intersolv 3.10 입니다. Informix 데이터베이스가 Informix 7.x(관계형 데이터베이스)인 경우는 CLI 2.8 을 이용하고, Informix 9.x(객체형 관계데이터베이스)인 경우는 Intersolv 3.10 을 이용합니다. Windows NT 의 ODBC manager 를 이용하여 DataSource 를 만들어 이용하는 경우는 다음과 같이 ConnectionString 을 처리합니다.

```
szConnstring = "DSN=ODBC310;DB=stores7;HOST=kor-k400;
```

```
    SERV=7011;SRVR=kdyim_tcp;PRO=olsoctcp;
```

```
    UID=Informix;PWD=in4mix"
```

```
ActiveConn.Open szConnstring

set rs = ActiveConn.Execute("select fname, lname from customer

    where customer_num=101")

%>

<br>fname : <%=rs("fname")%><br>

lname : <%=rs("lname")%><br>

<%'end if

    set ActiveConn = Nothing

    set rs = nothing

%>

<br>
```

예제에 대한 결과 산출

```
<br>fname : Ludwig    <br>

lname : Pauli      <br>

<br>
```