

Q & A (vol.3)

특정DB의 크기를 구하는 방법

우리 회사 시스템에는 여러 개의 데이터베이스가 있습니다. 이 중에서 어느 특정 데이터베이스가 차지하고 있는 실제 크기를 구하고 싶습니다

Q₁

A

Informix는 데이터베이스 내의 각 테이블의 데이터를 저장할 때 *extent*라는 물리적 디스크 저장 단위로 할당하기 때문에 각 테이블이 차지하고 있는 extent의 갯수와 그 크기를 구하는 것이 중요합니다. 이는 Informix OnLine 시스템 내의 모든 chunk가 얼마 만큼 어느 테이블에 할당되고 있는지를 알아볼 수 있는 *oncheck -pe* (각4.x, 5.x에서는 tbcheck -pe) 명령으로 가능합니다.

다음의 두 shell script는 각 Informix OnLine Dynamic Server를 사용하는 사용자와 Informix OnLine, 즉 4.x, 5.x 이전 버전을 사용하는 사용자를 위한 것입니다. 또한 각 script는 page size가 2KByte인 HP-UX 10.10에서 테스트된 내용입니다

1. 6.x 이후 버전을 사용하는 사용자를 위한 shell script

```
#*****  
  
# Title : dbsize  
  
# Usage : dbsize DB-name  
  
#           (Using DSA)  
  
#*****  
  
#!/bin/sh
```

```
PAGESIZE=2048
```

```
if test $# -eq 1
```

```
then
```

```
echo ''
```

```
echo "Executing DB Size..."
```

```
else
```

```
echo "USAGE : dbsize database_name"
```

```
exit 0
```

```
fi
```

```
oncheck -pe | grep $1 > tempcheck1
```

```
awk '{print $3}' tempcheck1 > tempcheck2
```

```
a=0
```

```
for i in `cat tempcheck2`
```

```
do
```

```
a=`expr $a + $i`
```

```
done
```

```
echo ''
```

```
echo '>> Size of '$1
```

```
echo ''
```

```
echo '##### '$a ' Pages'
```

```
a=`expr $a W* $PAGESIZE`
```

```
echo '##### '$a ' Bytes'
```

```
a=`expr $a W/ 1024`
```

```
echo '##### '$a ' KBytes'
```

```
Wrm tempcheck?
```

```
cho ''
```

```
echo 'Executing completed !'
```

```
echo ''
```

2. Informix OnLine 4.x와 5.x 사용자를 위한 shell script

```
#*****
```

```
# Title : dbsite5
```

```
# Usage : dbsite5 DB_name
```

```
# (Using earlier 6.0 version)
```

```
#*****
```

```
#!/bin/sh
```

```
PAGESIZE=2048
```

```
if test $# -eq 1
```

```
then
```

```
echo "
```

```
echo "Executing DB Size..."
```

```
else
```

```
echo "USAGE : dbsite database_name"
```

```
exit 0
```

```
fi
```

```
tbcheck -pe | grep $1 > tempcheck1
```

```
awk '{print $3}' tempcheck1 > tempcheck2
```

```
a=0
```

```
for i in `cat tempcheck2`
```

```
do
```

```
a=`expr $a + $i`
```

```
done
```

```
echo ""
```

```
echo '>> Size of ' $1
```

```
echo ""
```

```
echo '##### ' $a ' Pages'
```

```
a=`expr $a W * $PAGESIZE`
```

```
echo '      ##### ' $a ' Bytes'
```

```
a=`expr $a W/ 1024`
```

```
echo '      ##### ' $a ' KBytes'
```

```
Wrm tempcheck?
```

```
echo ''
```

```
echo 'Executing completed !'
```

```
echo ''
```

사용 예는 다음과 같습니다.

```
$) dbsize stores7
```

```
Executing DB Size...
```

```
>> Size of stores7
```

```
##### 344 Pages
```

```
##### 704512 Bytes
```

688 KBytes

Executing completed !

Database Level의 Lock판정하기

어떤 데이터베이스를 백업하기 위하여 dbexport 명령을 사용했습니다. 그런데 -425 오류와 -107

오류가 발생했습니다. 그러나 백업 받고자 하는 데이터베이스를 사용하고 있는 사용자가 없는 듯

Q2 합니다. 실제로 그 데이터베이스가 사용되고 있는지, 그래서 lock이 걸려 있는지 확인할 수 있습니까?

```
K400:/post/byrhee/sh> dbexport stores7
-425 - Database is currently opened by another user.
-107 - ISAM error: record is locked.
K400:/post/byrhee/sh> onstat -k

INFORMIX-OnLine Version 7.13.UC1 -- On-Line -- Up 10 days 06:44:41 -- 21744 Kbytes

Locks
address wtlst owner lklist type tblsnum rowid key#/bsiz
c8588b5c 0 c918e948 0 HDR+S 100002 205 0
c8588b88 0 c918e9e0 0 HDR+S 100002 204 0
c8588bb4 0 c918e948 c8588be0 HDR+X 1000c7 300 0
c8588be0 0 c918e948 c8588b5c HDR+IX 1000c7 0 0
c8588c0c 0 c918e948 c8588c38 HDR+X 1000cc 200 0
c8588c38 0 c918e948 c8588bb4 HDR+IX 1000cc 0 0
c8588c64 0 c918e948 c8588c0c HDR+X 1000cc 100 1
c8588c90 0 c918e514 0 HDR+S 100002 201 0
0 active, 10000 total, 16384 hash buckets

K400:/post/byrhee/sh>
```

그림 1: onstat -k의 결과 위 그림에서는 Database Level Lock이 3개이며, 각각의 rowid는 205, 204, 201입니다.

dbaccess를 실행하여 주 메뉴에서 Query Language로 들어갑니다. 데이터베이스는 *sysmaster*를 선택하십시오. 그런 다음, 다음의 질의를 수행하십시오.

```
SELECT hex(rowid), name FROM sysdatabases;
```

이 질의에서 우리는 각 데이터베이스 이름과 그 데이터베이스의 rowid를 구할 수 있습니다. onstat -k로 보이는 rowid가 16진수이기 때문에 이 질의에서도 hex() 함수를 사용하여 값을 구한 것입니다.



```
K400
File Edit Options Send Receive Window Help
SQL: New Run Modify Use-editor Output Choose Save Info Drop Exit
Run the current SQL statements.
----- sysmaster@maserv_tcp --- Press CTRL-W for Help -----
(expression) name
0x00000201 sysmaster
0x00000202 syspgm4g1
0x00000204 bluesky
0x00000205 stores7
4 row(s) retrieved.
VT100 TCP/IP 16:42
```

그림 2: 위의 질의를 실행한 결과 이 질의의 결과를 통해서 rowid가 201, 204, 205인 데이터베이스는 각각 sysmaster, bluesky, stores7임을 알 수 있습니다.

이제 *onstat -uk*를 실행해 보십시오. 예를 들면, *stores7* 데이터베이스를 백업받기 위해 dbexport를 실행하고자 했다면 lock 리스트에서 rowid가 205번인 것을 찾으시면 될 것입니다. 그리고 이 lock의 *owner*가 바로 stores7 데이터베이스를 사용하고 있음을 알 수 있습니다. 다음 그림에서 볼 수 있듯이 205번 lock은 owner가 *c918e948*이며, 윗 부분의 *UserThreads* 정보에서 확인하면 user id가 *Informix*인 사용자가 *ttyp3* 터미널에서 사용하고 있다는 것을 알 수 있습니다. 이 사용자 세션을 삭제하고자 한다면 다음의 명령으로 해제할 수 있습니다. 이 명령어는 해당 세션을 삭제하기 때문에 그 세션이 잡고 있던 lock도 자연스럽게 해제할 것입니다.

```
$) onmode -z 45
```

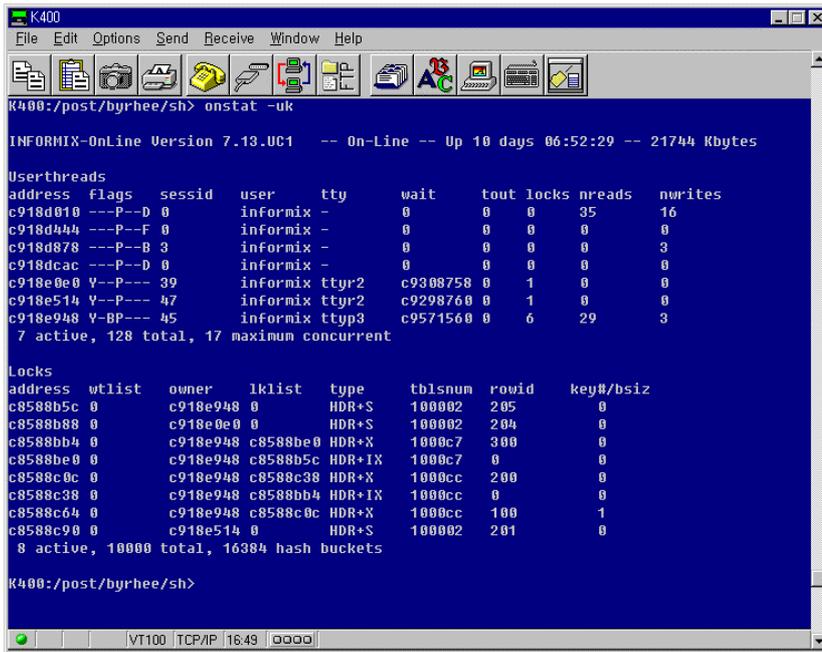


그림 3: onstat -uk로 현재 진행중인 lock과 user session을 한 번에 확인할 수 있으며, 따라서 각 lock의 owner를 쉽게 찾아볼 수 있습니다.

다음은 Database Level의 lock에서 하나의 rowid를 사용하여 그 rowid가 가리키는 데이터베이스의 이름을 찾을 수 있도록 간단하게 작성한 shell script입니다.

먼저 onstat -k를 수행하여 리스트된 rowid 중 하나를 선택하여 사용합니다.

```
#*****
```

```
# Title : lock_db
```

```
# Usage : lock_db ROWID
```

```
# (DATABASE Level Lock Checking)
```

```
# (Using upper case for all ROWID)
```

```
#*****
```

```
#!/usr/bin/sh
```

```
if test $# -eq 1
```

```
then
```

```
    DB=$1
```

```
else
```

```
    echo "USAGE : lock_db rowid "
```

```
    exit 0
```

```
fi
```

```
dbaccess << !
```

```
database sysmaster;
```

```
set isolation to dirty read;
```

```
select name from sysdatabases where hex(rowid) matches "*$1";
```

```
!
```

위 shell script를 사용한 예는 다음과 같습니다.

```
$) lock_db 204
```

```
Database selected.
```

```
Isolation level set.
```

```
name
```

```
bluesky
```

```
1 row(s) retrieved.
```

```
Database closed.
```

OnLine-DSA에서 logbuff를 증가시키면 더 많은 메모리를 사용합니다.

PHYSBUFF와 LOGBUFF 매개변수를 64k에서 256k로 증가시킬 때, OnLine 인스턴스가 갑자기

12 Mb의 메모리를 추가로 요구하는데, 왜 그런지 잘 모르겠습니다.

Q₃

따라서 매우 많은 오버헤드가 필요하게 됩니다. PHYSBUFF와 LOGBUFF가 256k에서 64k로 변경되면 online 로그 파일에는 다음과 같이 기록됩니다.

Thu Jul 20 16:55:15 1995

16:55:15 DR: DRAUTO is 0 (Off)

16:55:16 Informix OnLine Initialized -- Shared Memory Initialized.

16:55:16 Physical Recovery Started.

16:55:16 Physical Recovery Complete: 0 Pages Restored.

16:55:16 Logical Recovery Started.

16:55:17 dynamically allocated new shared memory segment

(size 4194304)

16:55:19 Logical Recovery Complete.

0 Committed, 0 Rolled Back, 0 Open, 0 Bad Locks

16:55:20 Onconfig parameter PHYSBUFF modified from 256 to 64.

16:55:20 Onconfig parameter LOGBUFF modified from 256 to 64.

16:55:20 Dataskip is now OFF for all dbspaces

16:55:20 On-Line Mode

16:55:20 Checkpoint Completed: duration was 1 seconds.

그러나 PHYSBUFF와 LOGBUFF가 64k에서 256k로 변경되는 그 내용은 아래와 같이 새로운 메모리 세그먼트를 할당받고 있습니다.

Thu Jul 20 16:56:28 1995

16:56:28 DR: DRAUTO is 0 (Off)

16:56:29 Informix OnLine Initialized -- Shared Memory Initialized.

16:56:29 Physical Recovery Started.

16:56:29 Physical Recovery Complete: 0 Pages Restored.

16:56:29 Logical Recovery Started.

16:56:29 dynamically allocated new shared memory segment

(size 4194304)

16:56:29 dynamically allocated new shared memory segment

(size 4194304)

16:56:29 dynamically allocated new shared memory segment

(size 4194304)

16:56:29 dynamically allocated new shared memory segment

(size 4194304)

16:56:32 Logical Recovery Complete.

0 Committed, 0 Rolled Back, 0 Open, 0 Bad Locks

16:56:32 Onconfig parameter PHYSBUFF modified from 64 to 256.

16:56:32 Onconfig parameter LOGBUFF modified from 64 to 256.

16:56:33 Dataskip is now OFF for all dbspaces

16:56:33 On-Line Mode

16:56:33 Checkpoint Completed: duration was 0 seconds.

null 값이 들어 있는 숫자 칼럼 계산

정수나 decimal 등 숫자 칼럼이 있습니다. 이 칼럼의 값을 더하여 합계나 평균 또는 최고치, 최

구해집니다. 이 null 값이 0으로 계산되게 할 수 있는 방법은 무엇입니까?

A

우선 해당 칼럼의 null 값을 0으로 갱신한다면 원하는 결과를 가져올 수 있을 것입니다. 그리고 테이블을 생성할 때 정수 칼럼에는 null이 입력되는 경우 0으로 저장되게끔 DEFAULT 값을 줄 수 있습니다.

그러나 이미 데이터가 저장되어 있는 상태에서 UPDATE 구문을 실행하기에 어려운 경우라면 다음의 프로시저를 사용하십시오.

```
CREATE PROCEDURE null_val(val decimal,  
  
                           newval decimal default 0)  
  
    RETURNING decimal;  
  
    IF val IS NOT null THEN  
  
        RETURN val;  
  
    ELSE  
  
        RETURN newval;  
  
    ENDIF;  
  
END PROCEDURE;
```

atest 라는 테이블로 예 를 들어보겠습니다.

```
CREATE TABLE atest
```

```
( num smallint,
```

```
price decimal(10,2));
```

```
INSERT INTO atest (num) values (1);
```

```
INSERT INTO atest (num) values (2);
```

```
INSERT INTO atest (num) values (3);
```

```
INSERT INTO atest (num) values (4);
```

```
INSERT INTO atest (num) values (5);
```

```
INSERT INTO atest (num) values (6);
```

```
INSERT INTO atest (num) values (7);
```

```
INSERT INTO atest (num) values (8);
```

```
INSERT INTO atest (num) values (9);
```

```
INSERT INTO atest (num) values (10);
```

위 예에서 볼 수 있는 것처럼 atest 테이블에는 모두 10개의 행이 입력되었습니다. 각 행은 num 칼럼에만 값이 들어가 있고 price 칼럼에는 모두 null로 채워졌습니다. 따라서 다음의 SELECT문을 실행하면 그 결과는 모두 null이 됩니다.

```
SELECT SUM(price), AVG(price), MAX(price), MIN(price) from atest;
```

그러나 사용자는 아마도 0이 리턴되기를 원했을 것입니다. 다시 다음과 같이 price 칼럼에 null이 아닌 값이 입력되었다고 가정하겠습니다.

```
INSERT INTO atest values (11,10);
```

이 경우, 위의 SELECT문의 결과는 다음과 같습니다.

(sum)	(avg)	(max)	(min)
10.00	10.000000000000000	10.00	10.00

그러나 역시 사용자는 위와 같은 결과를 기대하지는 않았을 것입니다.

다음과 같이 위의 내장 프로시저를 이용하여 SELECT문을 다시 구현할 수 있습니다.

```
Select SUM(null_val(price)),  
  
AVG(null_val(price)),  
  
MAX(null_val(price)),
```

MIN(null_val(price)) from atest;

(sum)	(avg)	(max)	(min)
10.0000000	0.9090909	10.0000000	0.0000000

위의 null_val 프로시저를 이용하여 null인 값을 모두 0으로 바꾸어서 작업한 결과입니다.

25572오류

Informix Engine을 Shut Down 하였다가 Bring Up할 때 종종 -25572 오류를 만나게 됩니다.

Q5 항상 이 오류가 나타나는 것은 아닌데, 왜 이런 현상이 일어나는지, 해결 방법은 무엇인지 알고 싶습니다.

이것은 Informix 가 할당받아 사용하고 있던 Service Port 를 Informix Engine 을 Shut Down 한 후에도 시스템에서 잡고 있다가 늦게 Free 시켜 주기 때문에 발생하는 오류입니다. 따라서 일정 시간이 지나면 정상적으로 Bring Up 될 것입니다. 이 때 *netstat* 명령으로 해당 Port 를 확인할 수 있습니다.

Sun Solaris 2.x 에서는 *nnd*라는 명령으로 Port-Free time 을 임의로 지정할 수 있습니다.

사용 방법은 다음과 같습니다.

```
$ /usr/sbin/nnd -set /dev/tcp_close_wait_interval 1000
```