

IBM GLOBAL SERVICES



# G13

## Exploring the Informix IDS Sysmaster Database

Lester Knutsen

IBM **DB2 Information Management**  
Technical Conference

Sept. 20-24, 2004

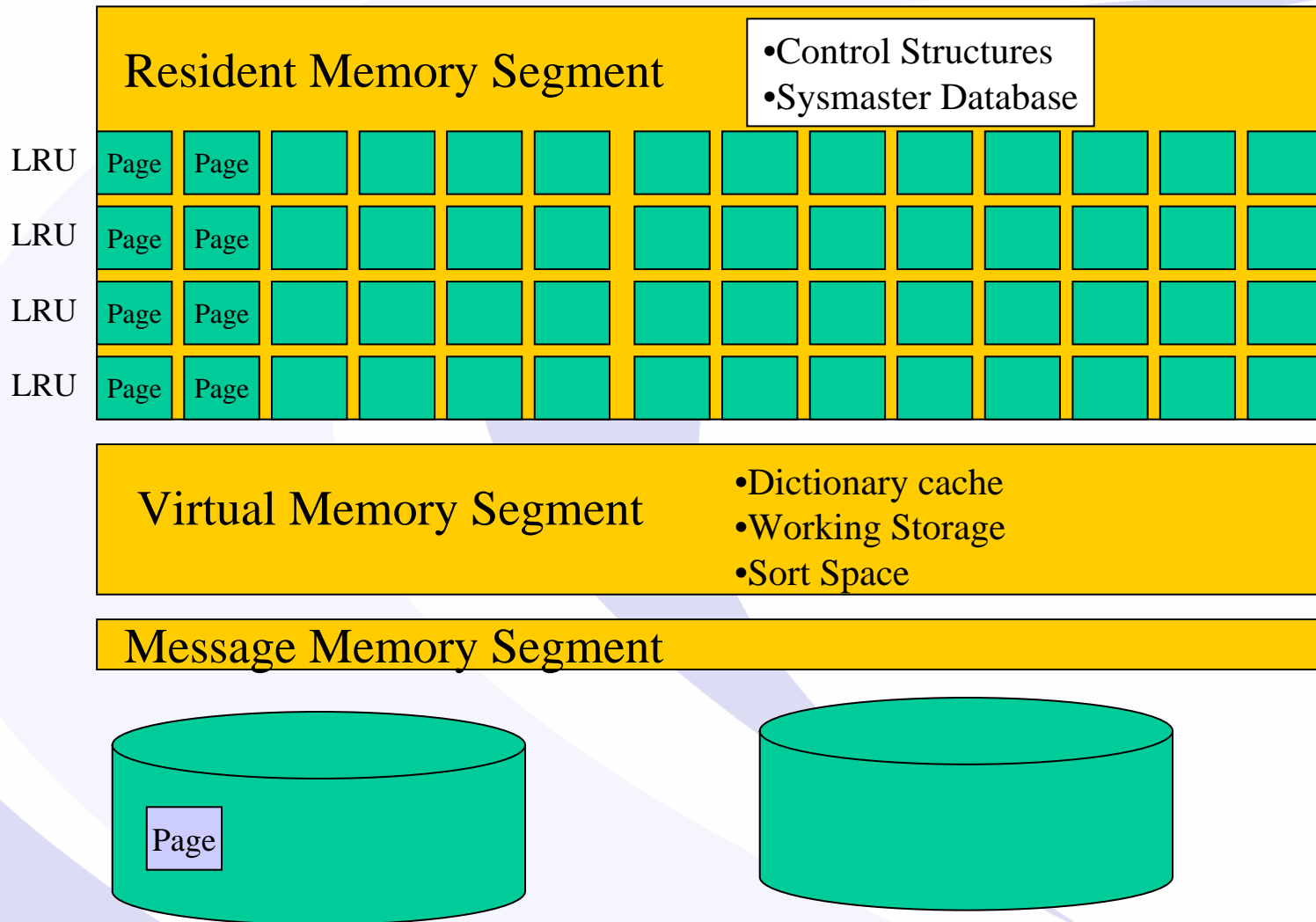
Las Vegas, NV



# What is the sysmaster database?

A database that peeks into  
the shared memory structures  
of an INFORMIX-Dynamic Server

# IDS Control Structures in Memory are the Sysmaster Database



# Sysmaster database contains:

- Server information
- Dbspace & chunk information
- Database & table information
- User session information



# How to manage and tune your Informix IDS Server using the sysmaster database

Objectives of this presentation:

- Performance tuning tips
- Scripts to monitor the health of your IDS server
- Explore the sysmaster database

# Monitoring and Tuning needs:

- Display information about your server configuration.
- Display key statistics – (buffer usage, waits, writes)
- Display free space available (like the unix df -k command).
- Display status and characteristics of each chunk device.
- Display blocks of free space to plan where to put large tables.
- Display I/O statistics by chunk devices.
- Display a layout of dbspace, databases, tables, and extents.
- Display table usage statistics sorted by reads, writes or locks.
- Display user session statistics sorted by reads, writes or locks.

# Performance of queries on sysmaster database

The data is in shared memory but:

- Views used by tables require disk access and may be slow
- Complex views used to hid complex data
- Some tables are large (250,000 locks)
- Unbuffered logging of temp tables



# Differences from other databases

- Do not update sysmaster tables as this may corrupt the server
- Cannot use dbschema on pseudo tables
- Cannot drop pseudo tables or the sysmaster database

# Isolation level is Dirty Read

- Data is dynamic and can change as you retrieve it (Dirty Read)
- Dynamic nature may return inconsistent results
- However, it uses Unbuffered logging and temp tables are logged

# Using Triggers and Stored Procedures

- Can create triggers and Stored Procedures
- Triggers will never be execute because tables do not change using normal SQL updates
- Use “polling” to check for changes
- Stored Procedures can be executed in the sysmaster database

# Sysmaster database may change

- Some undocumented tables and columns may change in future versions
- Scripts in this presentation using undocumented features may not work on all versions of Informix IDS
- Scripts have been run on versions 7.2 to 9.4

# Creating the sysmaster database

When OnLine is first initialized the sysmaster database is created using the script in \$INFORMIXDIR/etc/sysmaster.sql

- Create real tables with the structures of the pseudo tables
- Copy the structure of the real tables to temp tables
- Drop the real tables
- Update the systables.partnum to point to pseudo tables in shared memory
- Create the flags\_text table which has the interpretations for flags used in the tables
- Create stored procedures used in the views, two of which are interesting:
  - bitval() is a stored procedure for getting the boolean flag values
  - l2date() is a stored procedure for converting unix time() long values to dates
- Create the sysmaster views
- This process requires 2000KB of Logical Logs

# Interesting table flags\_text

table flags\_text

tablename char(128), -- sysmaster table

flags int, -- flag

txt char(50) – description of flag value

# Goals of Managing and Performance Tuning

- Reduce CPU Cycles
- Reduce Disk I/O
- Reduce Memory Requirements
- Reduce Network Bandwidth

# Server configuration and statistics tables:

- sysconfig - ONCONFIG File
- sysdri - Data Replication
- syslogs - Logical Logs
- sysprofile - Server Statistics
- sysvpprof - Virtual Processors



# Sysconfig (onstat -c)

**View sysconfig:** Configuration information from the IDS server.

This information was read from the ONCONFIG file when the server was started.

cf_id	integer,	unique numeric identifier
cf_name	char(18),	config parameter name
cf_flags	integer,	flags, 0 = in view sysconfig
cf_original	char(256),	value in ONCONFIG at boottime
cf_effective	char(256),	value effectively in use
cf_default	char(256)	value by default

# What is the current server configuration?

```
select
  cf_name      parameter,
  cf_effective effective_value
from          sysconfig
```

# SQL output

parameter	effective_value
ROOTNAME	rootdbs
ROOTPATH	/u3/dev/rootdbs1
DBSERVERNAME	train1
MIRRORPATH	/u3/dev/rootdbsm1
PHYSDBS	rootdbs
MSGPATH	/u3/informix7/online1.log
CONSOLE	/u3/informix7/console1.log
TAPEDEV	/dev/null
LTAPEDEV	/dev/null
ROOTOFFSET	0
ROOTSIZE	400000

# Sysdri (onstat -g dri)

**View sysdri:** Data replication state and configuration information.

type	char(50),	DR server type
state	char(50),	DR state
name	char(20),	DR server name
intvl	integer,	DR buffer flush interval
timeout	integer,	DR network timeout
lostfound	char(128)	DR lost+found pathname

# Syslogs (onstat -l)

**View syslogs:** Logical logs status.

number	smallint, logfile number
uniqid	integer, logfile uniqid
size	integer, pages in logfile
used	integer, pages used in logfile
is_used	integer, 1 for used, 0 for free
is_current	integer, 1 for current
is_backed_up	integer, 1 for backedup
is_new	integer, 1 for new
is_archived	integer, 1 for archived
is_temp	integer, 1 for temp
flags	smallint logfile flags

# What is the status of the logical logs?

```
-- List Logical Logs
select
    uniqid,
    used size_used,
    is_used,
    is_current,
    is_backed_up,
    is_archived
from syslogs
order by uniqid
```

# SQL output

uniqid	size_used	is_used	is_current	is_backed_up	is_archived
32	1000	1	0	1	1
33	1000	1	0	1	1
34	1000	1	0	1	1
35	1000	1	0	1	0
36	1000	1	0	1	0
37	1000	1	0	1	0
38	1000	1	0	1	0
39	1000	1	0	1	0
40	1000	1	0	1	0
41	1000	1	0	1	0
42	1000	1	0	1	0

# Sysprofile (onstat -p)

**View sysprofile:** Current statistics and performance information of the server.

name	char(16),	profile element name
value	integer	current value

The values are re-set to 0 when IDS is shutdown and started and when the command “onstat -z” is used.



# Sysprofile – Profile Names

dskreads  
isamtot  
iswrites  
isrollbacks  
latchwts  
ckptwts  
plgpagewrites  
llgwrites  
compress  
btradata  
seqscans  
maxsortspace

bufreads  
isopens  
isrewrites  
ovlock  
buffwts  
deadlks  
plgwrites  
pagreads  
fgwrites  
btraidx  
totalsorts

dskwrites  
isstarts  
isdeletes  
ovuser  
lockreqs  
lktouts  
llgreccs  
pagwrites  
lruwrites  
dpra  
memsorts

bufwrites  
isreads  
iscommits  
ovtrans  
lockwts  
numckpts  
llgpagewrites  
flushes  
chunkwrites  
rapgs\_used  
disksorts

# Sysprofile - onstat -p

```
Informix Dynamic Server Version 9.30.TC2-- On-Line -- Up 00:10:24 -  
58496 Kbytes
```

Profile

dskreads	pagreads	bufreads	%cached	dskwrits	pagwrits	bufwrits	%cached
350	372	2108	83.40	10	11	1	0.00
isamtot	open	start	read	write	rewrite	delete	commit
rollbk							
2839	94	122	457	0	0	0	0
0							
gp_read	gp_write	gp_rewrt	gp_del	gp_alloc	gp_free	gp_curs	
2	0	0	0	0	0	2	
ovlock	ovuserthread	ovbuff	usercpu	syscpu	numckpts	flushes	
0	0	0	3.93	0.87	2	6	
bufwaits	lokwaits	lockreqs	deadlks	dltouts	ckpwaits	compress	seqscans
67	0	966	0	0	0	0	4
ixda-RA	idx-RA	da-RA	RA-pgsused	lchwaits			
5	0	247	252	0			

# What are some of the key server statistics?

-- Select key Profile values

```
select name, value from sysprofile
```

where name in

```
( "ovlock", "ovuser", "ovtrans", "latchwts", "buffwts",  
  "lockwts", "ckptwts", "deadlks", "lktouts", "fgwrites",  
  "lruwrites", "chunkwrites" )
```

# SQL output

name	value
ovlock	0
ovuser	0
ovtrans	0
latchwts	41
buffwts	1617
lockwts	0
ckptwts	12
deadlks	0
lktouts	0
fgwrites	1190
lruwrites	21430
chunkwrites	4648

# What percent of I/O is from buffers?

```
-- Get % read cached
select dr.value dskreads, br.value bufreads,
       round ((( 1 - ( dr.value / br.value )) *100 ), 2) cached
from sysprofile dr, sysprofile br
where dr.name = "dskreads"
and   br.name = "bufreads";
-- Get % write cached
select dw.value dskwrites, bw.value bufwrites,
       round ((( 1 - ( dw.value / bw.value )) *100 ), 2) cached
from sysprofile dw, sysprofile bw
where dw.name = "dskwrites"
and   bw.name = "bufwrites"
```

# SQL output

dskreads	bufreads	cached
29209	1489235	98.04
dskwrites	bufwrites	cached
56228	414748	86.44

# Sysvpprof (onstat -g)

**View sysvpprof:** Current statistics on IDS Virtual Processors

vpid	integer,	VP id
txt	char(50)	VP class name
usecs_user	float,	number of unix secs of user time
usecs_sys	float	number of unix secs of system time

# What is the status of the virtual processors?

```
-- Select VP Statistics
select
  vpid,
  pid,
  txt[1,5] class,
  round( usecs_user, 2) usercpu,
  round( usecs_sys, 2) syscpu
from   sysvplst a, flags_text b
where  a.class = b.flags
and    b.tabname = "sysvplst"
```



# SQL output

vpid	pid class	usercpu	syscpu
1	295 cpu	309.26	23.58
2	296 adm	0.14	0.36
3	297 lio	0.27	5.57
4	298 pio	0.15	1.49
5	299 aio	5.00	46.16
6	300 msc	0.04	0.24
7	301 aio	4.65	43.75
8	302 tli	0.14	0.30
9	305 pio	0.22	1.56

# Interesting undocumented table – Sysshmvals\*

sh_mode	int, turbo mode number	sh_optstgbsnum	int, Subsystem Staging Blobspace
sh_boottime	int, boot time of day	sh_cpflag	int, TRUE => doing checkpoint
sh_pfcrltime	int, time profilers were last clr	sh_rapages	int, Number of pages to read ahead
sh_curtime	int, current mt_time	sh_rathreshold	int, When to start next read ahead
sh_bootstamp	int, boot time stamp	sh_lastlogfreed	int, last log (id) written to tape
sh_stamp	int, current time stamp	sh_rmdlktout	int, max timeout when distributed
sh_mainlooptcb	int, address of main daemon thread	sh_narchivers	int, number of active archives
sh_sysflags	int, system operating flags	sh_maxpdqpriority	int, max pdqpriority
sh_maxchunks	int, size of chunk table	sh_fuzcpflag	int, fuzzy checkpoint flag
sh_maxdbspaces	int, size of dbspace table	sh_needcpsync	int, hard checkpoint
sh_maxuserthreads	int, max # of user structures	sh_nfuzzy	int, # buffers marked fuzzy
sh_maxtrans	int, max # of trans structures	sh_nfuzzyprev	int, # buffers marked fuzzy in last ckpt
sh_maxlocks	int, # of locks total	sh_oldestlsnuq	int, lsn of oldest update not
sh_maxlogs	int, size of log table	sh_oldestlsnpos	int, flushed to disk
sh_nbufs	int, # of buffers total	sh_builddpt	int, builing DPT necessary
sh_pagesize	int, buffer size in bytes	sh_ndptentries	int, #entries in DPT
sh_nlrus	int, # of lru queues	sh_dptsize	int, size of DPT
sh_maxdirty	float,LRU can have this % dirty pages	sh_curmaxcons	int, max #connections in this run
sh_mindirty	float,LRU has % dirty pages after clean	sh_ovlmaxcons	int max #connections since server init
sh_ncleaners	int, # of cleaning/flushing procs		
sh_longtx	int, the long transaction flag		

## DbSPACE & chunk tables:

- sysdbspaces - DB Spaces
- syschunks - Chunks
- syschkio - I/O by Chunk
- syschfree\* - Free Space by Chunk

# Sysdbspaces (onstat -d)

**View sysdbspaces:** List all dbspaces on the server

dbnum	smallint,	dbspace number,
name	char(18),	dbspace name,
owner	char(8)	dbspace owner,
fchunk	smallint,	first chunk in dbspace,
nchunks	smallint,	number of chunks in dbspace,
is_mirrored	bitval,	dbspace mirrored, 1=Yes, 0=No
is_blobspace	bitval,	dbspace a blob space, 1=Yes, 0=No
is_temp	bitval,	dbspace temp, 1=Yes, 0=No
flags	smallint	dbspace flags

# Syschunks (onstat -d)

**View syschunks:** Lists all chunks on the server

chknum	smallint,	chunk number
dbnum	smallint,	dbspace number
nxchknum	smallint,	number of next chunk in dbspace
chksize	integer,	pages in chunk
offset	integer,	pages offset into device
nfree	integer,	free pages in chunk
is_offline	bitval,	chunk offline, 1=Yes, 0=No
is_recovering	bitval,	chunk recovering, 1=Yes, 0=No
is_blobchunk	bitval,	chunk blobchunk, 1=Yes, 0=No

# Syschunks (continued)

is_inconsistent	bitval,	chunk inconsistent, 1=Yes, 0=No
flags	smallint,	chunk flags converted by bitval
fname	char(128),	device pathname
mfname	char(128),	mirror device pathname
moffset	integer,	pages offset into mirror device
mis_offline	bitval,	mirror chunk offline, 1=Yes, 0=No
mis_recovering	bitval,	mirror chunk recovering, 1=Yes, 0=No
mflags	smallint	mirror chunk flags

# Syschkio (onstat -D)

**View syschkio:** Lists I/O statistics by chunk

chunknum	smallint,	chunk number
reads	integer,	number of read ops
pagesread	integer,	number of pages read
writes	integer,	number of write ops
pageswritten	integer,	number of pages written
mreads	integer,	number of mirror read ops
mpagesread	integer,	number of mirror pages read
mwrites	integer,	number of mirror write ops
mpageswritten	integer	number of mirror pages written

# Syschfree\*

**Table syschfree:** Lists free space on a chunk

chknum	integer,	chunk number
extnum	integer,	extent number in chunk
start	integer,	physical addr of start
leng	integer	length of extent



# How much dbspace is free?

```
-- dbsfree.sql
select    d.dbsnum,
          name dbspace,
          sum(chksize)                Pages_size, -- sum of all chunks size pages
          sum(chksize) - sum(nfree)   Pages_used,
          sum(nfree)                 Pages_free, -- sum of all chunks free pages
          round ((sum(nfree)) / (sum(chksize)) * 100, 2)    Percent_free
from      sysdbspaces d, syschunks c
where     d.dbsnum = c.dbsnum
and       d.is_blobspace = 0
group by 1, 2
order by 1;
```

# SQL output

dbspace	pages_size	pages_used	pages_free	percent_free
rootdbs	20000	5653	14347	71.74
logsdbs	12500	12053	447	3.58
datadbs	25000	6722	18278	73.11
tmpdbs	12500	53	12447	99.58

# How much blob space is free?

```
-- blobfree.sql
select
    name dbspace,
    sum(chksize)      Size_in_Pages,      -- sum of all chunks size pages
    sum(nfree)       Num_free_blob_page  -- sum of all chunks free
                                                pages
from    sysdbspaces d, syschunks c
where   d.dbsnum = c.dbsnum
and     d.is_blobspace = 1
group by 1
order by 1
```

# SQL output

dbspace	size_in_pages	num_free_blob_page
blobdbs	10000	2497

# Where are blocks of free dbspace?

```
-- chkflist.sql
select
    name dbspace,           -- dbspace name
    f.chknum,              -- chunk number
    f.extnum,              -- extent number of free space
    f.start,               -- starting address of free space
    f.leng free_pages      -- length of free space
from    sysdbspaces d, syschunks c, syschfree f
where   d.dbsnum = c.dbsnum
and     c.chknum = f.chknum
order by dbspace, free_pages desc
```

# SQL output

dbspace	chknun	extnum	start	free_pages
datadbs	4	0	3	12497
datadbs	3	31	9107	3393
datadbs	3	15	1921	976
datadbs	3	13	1705	160
datadbs	3	30	6069	160
datadbs	3	25	5429	128
datadbs	3	19	4853	96
datadbs	3	29	5909	96
datadbs	3	24	5333	64
datadbs	3	26	5621	64

# What chunks have the most I/O?

```
-- chkio.sql
select    name dspace, -- truncated to fit 80 char screen line
          chknum, "Primary" chktype,
          reads,  writes,
          pagesread,          pageswritten
from      syschktab c, sysdbstab d      where   c.dbsnum = d.dbsnum
union all
select    name dspace,
          chknum, "Mirror"  chktype,
          reads,  writes,
          pagesread,          pageswritten
from      sysmchktab c, sysdbstab d      where   c.dbsnum = d.dbsnum
order by 1,2,3;
```

# SQL output

dbspace	chknun	chktype	reads	writes	pagesread	pageswritten
blobdbs	6	Primary	21	3	31	10
datadbs	3	Primary	2082	31	9087	31
datadbs	4	Primary	5	0	7	0
logsdbs	2	Primary	176	996	1347	11704
rootdbs	1	Mirror	11616	26196	22499	30102
rootdbs	1	Primary	13340	26111	22271	30102
tmpdbs	5	Primary	13	2	13	3



# What is the status of chunks?

```
select name dbspace,      -- dbspace name
       d.dbsnum,         -- dbspace num
       is_mirrored,      -- dbspace is mirrored 1=Yes 0=No
       is_blobspace,    -- dbspace is blobspace 1=Yes 0=No
       is_temp,         -- dbspace is temp 1=Yes 0=No
       chunknum,        -- chunk number
       fname device,    -- dev path
       offset dev_offset, -- dev offset
       is_offline,      -- Offline 1=Yes 0=No
       is_recovering,   -- Recovering 1=Yes 0=No
       is_blobchunk,    -- Blobspace 1=Yes 0=No
       is_inconsistent, -- Inconsistent 1=Yes 0=No
       chksize Pages_size, - chunk size in pages
       nfree Pages_free, -- chunk free pages
       mfname mirror_device, -- mirror dev path
       mis_recovering_offse -- mirror recovering 1=Yes 0=No
from   sysdbspaces d, syschunks c
where  d.dbsnum = c.dbsnum
order by dbsnum, dbspace, chunknum
```

# SQL output

dbspace	rootdbs
dbnum	1
is_mirrored	1
is_blobspace	0
is_temp	0
chunknum	1
device	/u3/dev/rootdbs1
dev_offset	0
is_offline	0
is_recovering	0
is_blobchunk	0
is_inconsistent	0
pages_size	20000
pages_free	14355
mirror_device	/u3/dev/rootdbsm1
mirror_offset	0

# Database & table information tables:

- sysdatabases - Databases
- systabnames - Tables
- sysextents - Tables extents
- sysptprof - Tables I/O
- systabinfo\* - Tables information

# Sysdatabases

**View sysdatabases:** List of databases on the server.

name	char(18),	database name
partnum	integer,	table id for systables
owner	char(8),	user name of creator
created	integer,	date created
is_logging	bitval,	unbuffered logging, 1=Yes, 0=No
is_buff_log	bitval,	buffered logging, 1=Yes, 0=No
is_ansi	bitval,	ANSI mode database, 1=Yes, 0=No
is_nls	bitval,	NLS support, 1=Yes, 0=No
flags	smallint	logging flags

# Systabnames

**Table systabnames:** All tables on the server.

partnum	integer,	table id for table
dbname	char(18),	database name
owner	char(8),	table owner
tablename	char(18),	table name
collate	char(32)	collation associated with NLS DB

# Sysextsents (oncheck -pe)

**View sysextsents:** Tables and each extent on the server.

dbname	char(18),	database name
tabname	char(18),	table name
start	integer,	physical address for this extent
size	integer	size of this extent

# Sysptprof

**View sysptprof:** Tables IO profile.

dbsname	char(18),	database name
tabname	char(18),	table name
partnum	integer,	partnum for this table
lockreqs	integer,	lock requests
lockwts	integer,	lock waits
deadlks	integer,	deadlocks
lktouts	integer,	lock timeouts
isreads	integer,	reads
iswrites	integer,	writes
isrewrites	integer,	rewrites

# Sysptprof (continued)

isdeletes  
bufreads  
bufwrites  
seqscans  
pagreads  
pagwrites

integer,  
integer,  
integer,  
integer  
integer,  
integer

deletes  
buffer reads  
buffer writes  
sequential scans  
disk reads  
disk writes



# Systabinfo\*

**View systabinfo:** Table information

ti_partnum	integer,	table's partnum
ti_flags	smallint,	partition flags
ti_rowsize	smallint,	rowsize (max for variable)
ti_ncols	smallint,	number of varchar or blob columns
ti_nkeys	smallint,	number of indexes
ti_nextns	smallint,	number of extents
ti_created	integer,	date created
ti_serialv	integer,	current serial value
ti_fextsiz	integer,	first extent size ( in pages )
ti_nextsiz	integer,	next extent size ( in pages )
ti_nptotal	integer,	number of pages allocated

## Systabinfo\* (continued)

ti_npused	integer,	number of pages used
ti_npdata	integer,	number of data pages
ti_octptnm	integer,	OCT partnum (optical blobs only)
ti_nrows	integer	number of data rows

# What databases are on the server?

```
-- dblist.sql
select -- use dbinfo function to convert partnum to
       dbspace
       dbinfo("DBSPACE",partnum) dbspace,
       name database,
       owner,
       is_logging,
       is_buff_log
from   sysdatabases
order by dbspace, name;
```

# SQL output

dbspace	database	owner	is_logging	is_buff_log
datadbs	extentdb2	usr2	0	0
datadbs	zip1	usr1	0	0
datadbs	zip_lk	lester	0	0
rootdbs	extentdb	lester	0	0
rootdbs	extentdb1	usr1	0	0
rootdbs	onpload	lester	1	0
rootdbs	stores1	usr1	0	0
rootdbs	stores2	usr2	0	0
rootdbs	stores7	informix	0	0
rootdbs	sysmaster	informix	1	0

# What is the size of my databases?

```
select dbname,  
       sum( pe_size ) total_pages  
from systabnames, sysptnext  
where partnum = pe_partnum  
group by 1  
order by 2 desc
```

# What tables have extents?

```
-- tabextents.sql
select  dbsname,
        tabname,
        count(*)      num_of_extents,
        sum( pe_size ) total_size
from    systabnames, sysptnext
where   partnum = pe_partnum
group by 1, 2
order by 3 desc, 4 desc;
```

# SQL output

dbsname	tabname	num_of_extents	total_size
zip7	zip	50	1168
zip_lk	zip	27	1544
rootdbs	TBLSpace	8	400
sysmaster	syscolumns	6	56
datadbs	TBLSpace	4	200
sysmaster	sysviews	3	24
sysmaster	sysprocbody	3	24
sysmaster	systables	3	24
extentdb1	extent_sizes	2	24
sysutils	sysprocbody	2	16
sysmaster	sysconstraints	2	16
stores2	sysprocbody	2	16

# How calculate new extent sizes?

```
-- tabextprop.sql
select      dbname,
           tabname,
           count(*) num_of_extents,
           sum (pe_size ) current_pages_used,
           round (sum (pe_size )
                * 2 { Your systems page size in KB }
                * 1.2 { Add 20% Growth factor })
           Proposed_ext_size, { First Extent Size in KB }
           round (sum (pe_size )
                * 2 { Your systems page size in KB }
                * .2 { Estimated 20% Yearly Growth })
           Proposed_next_size { Next Extent Size in KB }
from        systabnames, sysptnext
where      partnum = pe_partnum
group by 1, 2
order by 3 desc, 4 desc;
```



# SQL output

dblname	zip7
tabname	zip
num_of_extents	50
current_pages_used	1168
proposed_ext_size	2803
proposed_next_size	467

dblname	zip_lk
tabname	zip
num_of_extents	27
current_pages_used	1544
proposed_ext_size	3706
proposed_next_size	618

# What tables have the most I/O?

```
-- tabprofile.sql
select
  dbname,
  tablename,
  -- uncomment the following lines as needed to show specific values
  isreads,      -- bufreads,      -- pagreads
  iswrites,     -- bufwrites,     -- pagwrites
  lockreqs     -- lockwts,       -- deadlks
from    sysptprof
order by isreads desc; -- change this sort to whatever you need to monitor.
```

# SQL output

dbsname	tabname	isreads	iswrites	lockreqs
zip	zip	41898	41898	830
sysmaster	systables	11402	0	67187
sysmaster	sysusers	10276	315	51373
sysmaster	sysviews	2653	0	15919
sysmaster	sysprocauth	2212	0	13272
zip_lk	zip	1399	0	1
sysmaster	sysprocedures	1108	0	6649
sysmaster	syscolumns	872	0	5182
sysmaster	sysdatabases	538	3	1469
sysmaster	flags_text	450	0	2546
zip	systables	101	34	239
sysmaster	systabauth	86	0	536

## What tables have sequence scans?

```
select      dbsname,  
            tabname,  
            sum(seqscans) total_scans  
from sysptprof  
where      seqscans > 0  
group      by 1, 2  
order      by 3 desc
```

## User session information tables:

- sysessions - Session data
- sysesprof - User statistics
- syslocks - Locks
- syseswts - Wait times

# Sysessions

## (onstat -g ses)

**View sysessions:** User session and connection information.

sid	integer,	Session id number
username	char(8),	User name
uid	smallint,	User unix id
pid	integer,	User process id
hostname	char(16),	Hostname
tty	char(16),	TTY port
connected	integer,	Time user connected
feprogram	char(16),	Program name
pooladdr	integer,	Pointer to private session pool

## Sysessions (continued)

is_wlatch	integer, Flag 1=Yes, 0=No, wait on latch
is_wlock	integer, Flag 1=Yes, 0=No, wait on lock
is_wbuff	integer, Flag 1=Yes, 0=No, wait on buffer
is_wckpt	integer, Flag 1=Yes, 0=No, wait on checkpoint
is_wlogbuf	integer, Flag 1=Yes, 0=No, wait on log buffer
is_wtrans	integer, Flag 1=Yes, 0=No, wait on a transaction
is_monitor	integer, Flag 1=Yes, 0=No, a monitoring process
is_incrit	integer, Flag 1=Yes, 0=No, in critical section
state	integer Flags

# Sysesprof (onstat -g ses)

**View sysesprof:** User session performance statistics.

sid	integer,	Session Id
lockreqs	decimal(16,0),	Locks requested
locksheld	decimal(16,0),	Locks held
lockwts	decimal(16,0),	Locks waits
deadlks	decimal(16,0)	Deadlocks detected
lktouts	decimal(16,0),	Deadlock timeouts
logrecs	decimal(16,0),	Logical Log records written
isreads	decimal(16,0),	Reads
iswrites	decimal(16,0),	Writes
isrewrites	decimal(16,0),	Rewrites
isdeletes	decimal(16,0),	Deletes



# Sysesprof (continued)

iscommits	decimal(16,0),	Commits
isrollbacks	decimal(16,0),	Rollbacks
longtxs	decimal(16,0),	Long transactions
bufreads	decimal(16,0),	Buffer reads
bufwrites	decimal(16,0),	Buffer writes
seqscans	decimal(16,0),	Sequential scans
pagreads	decimal(16,0),	Page reads
pagwrites	decimal(16,0),	Page writes
total_sorts	decimal(16,0),	Total sorts
dsksorts	decimal(16,0),	Sorts to disk
max_sortdiskspace	decimal(16,0),	Max space used by a sort
logspused	decimal(16,0),	Current log bytes used
maxlogsp	decimal(16,0)	Max bytes of logical logs used

# Syslocks (onstat -k)

**View syslocks:** Active locks on server.

dbname	char(18),	Database name
tablename	char(18),	Table name
rowidlk	integer,	Rowid for index key lock
keynum	smallint,	Key number of index key lock
owner	integer,	Session ID of lock owner
waiter	integer	Session ID of first waiter
type	char(4),	Type of Lock

# Syslocks – Type of Locks

- B - byte lock
- IS - intent shared lock
- S - shared lock
- XS - repeatable read shared key
- U - update lock
- IX - intent exclusive lock
- SIX - shared intent exclusive
- X - exclusive lock
- XR - repeatable read exclusive

# Syseswts

**View syseswts:** Wait status and times on objects.

sid	integer,	Session ID
reason	char(50),	Description of reason for wait
numwaits	integer,	Number of waits for this reason
cumtime	float,	Cumulative wait time for this reason
maxtime	integer	Max wait time for this reason

# Example SQL: dbwho.sql

```
select  sysdatabases.name database,      -- Database Name
        syssessions.username,          -- User Name
        syssessions.hostname,         -- Workstation
        syslocks.owner sid             -- Informix Session ID
from    syslocks, sysdatabases , outer syssessions
where   syslocks.tabname = "sysdatabases" -- Locks on sysdatabases
and     syslocks.rowidlk = sysdatabases.rowid -- Join to database
and     syslocks.owner = syssessions.sid    -- Use session ID
order by 1;
```

# Dbwho shell script

```
#!/bin/sh
# Program: dbwho   Description: List database, user and workstation of all db users
echo "Generating list of users by database ..."
dbaccess sysmaster - <<EOF
select      sysdatabases.name database,
            sysessions.username,
            sysessions.hostname,
            syslocks.owner sid
from        syslocks, sysdatabases , outer sysessions
where       syslocks.rowidlk = sysdatabases.rowid
and         syslocks.tabname = "sysdatabases"
and         syslocks.owner = sysessions.sid;
order by 1;
EOF
```

# List all Active Sessions

```
-- sessions.sql
select sid,
       username,
       pid,
       hostname,
       l2date(connected) startdate -- convert unix time to date
from   syssessions
```

## Sample Output

sid	username	pid	hostname	startdate
47	lester	11564	merlin	07/14/1997

# List Users Waiting on Resources

```
-- seswait.sql
select  username,
        is_wlatch, -- blocked waiting on a latch
        is_wlock, -- blocked waiting on a locked record or table
        is_wbuff, -- blocked waiting on a buffer
        is_wckpt, -- blocked waiting on a checkpoint
        is_incrit -- session is in a critical section of transaction (e.g writing to disk)
from    sysessions
order by username;
```

## Sample Output

username	is_wlatch	is_wlock	is_wbuff	is_wckpt	is_incrit
lester	0	1	0	0	0
lester	0	0	0	0	0
lester	0	0	0	0	0



# Monitor Resource Usage by User

```
-- sesprof.sql
select  username,
        syssesprof.sid,
        lockreqs,
        bufreads,
        bufwrites
from    syssesprof, syssessions
where   syssesprof.sid = syssessions.sid
order  by bufreads desc
```

# Some Undocumented Extras...

## Some Key systrans fields

tx_id	integer	pointer to transaction table
tx_logbeg	integer	transaction starting logical log
tx_loguniq	integer	transaction current logical log number

# Display Transactions and Logs

```
-- txlogpos.sql
select    t.username,
          t.sid,
          tx_logbeg,
          tx_loguniq,
          tx_logpos
from      sysstrans x, sysrstcb t
where     tx_owner = t.address
```

## SQL Output

username	sid	tx_logbeg	tx_loguniq	tx_logpos
informix	1	0	16	892952
lester	53	0	0	0
informix	12	0	0	0

# Display Transactions and Logs (1 of 3)

```
-- logstat.sql
-- select transaction data into a temp table
select tx_logbeg, tx_loguniq
from   systrans
into temp b;
-- count how many transactions begin in each log
select tx_logbeg, count(*) cnt
from B
where tx_logbeg > 0
group by tx_logbeg
into temp C;
```

# Display Transactions and Logs (2 of 3)

```
-- count how many transactions currently are in each log
select tx_loguniq, count(*) cnt
from B
where tx_loguniq > 0
group by tx_loguniq
into temp D;
```

# Display Transactions and Logs (3 of 3)

```
-- join data from counts with syslogs
select  uniqid,
        size,
        is_backed_up,    -- 0 = no, 1 = yes log is backed up
        is_archived,     -- 0 = no, 1 = yes log is on last archive
        c.cnt tx_beg_cnt,
        d.cnt tx_curr_cnt
from    syslogs, outer c, outer D
where   uniqid = c.tx_logbeg
and     uniqid = d.tx_loguniq
order  by uniqid
```

# Display Transactions and Logs

## SQL Output

<u>SQL Output</u>						
uniqid	size	is_backed_up	is_archived	tx_beg_cnt		tx_curr_cnt
10	500	1	1			
11	500	1	1			
12	500	1	1			
13	500	1	1			
14	500	1	1			
15	500	1	1			
16	500	0	1	1		2
16	500	0	1	1		2

# Display Current SQL

```
-- syssql.sql

select username,
       sqx_sessionid,
       sqx_conbno,
       sqx_sqlstatement
from syssqexplain, sysscb1st
where sqx_sessionid = sid
```



# Current SQL Output

```
username      lester
sqx_sessionid 55
sqx_conbno    2
sqx_sqlstatement select username,sqx_sessionid, sqx_conbno, sqx_sqlstatement
                from syssexplain, syscblst
                where sqx_sessionid = sid
```

```
username      lester
sqx_sessionid 51
sqx_conbno    0
sqx_sqlstatement update items set total_price = 300 where item_num = 1
```

# Build Your Own Monitoring System

- Provide a baseline of performance information to compare to future problems
- Collect data from:
  - sysmaster
  - sar
- Load into a database for review and analysis
- Save historical data for future comparisons

# Data Collection

- Create a cron job to run data collection scripts
  - Hourly/daily
  - Weekly
  - Monthly
- Build a database and load the data

# Hourly Data Collection

- From sysmaster:

lk\_sesprof.sql – syssession – User statistics

- From onstat:

onstat –g mgm – PDQ statistics

- From Unix:

ps –ef – collect user statistics

mpstat 5 5 – collect CPU statistics

# Daily Data Collection

- From sysmaster:
  - lk\_profile.sql – sysprofile – System statistics
  - lk\_chkio.sql – syschktab – Chunk I/O
  - lk\_dbsfree.sql – sysdbspaces, syschunks – Free space
  - lk\_vpprof.sql – sysvplst – VP statistics
  - lk\_tabprof.sql – sysptprof – Table I/O statistics
- From sar:
  - sar –u – CPU statistics
  - sar –b – Buffer statistics
  - sar –c – System calls
  - sar –d – Disk I/O statistics
  - sar –q – Run Queue statistics

# Weekly/Monthly Data Collection

- From sysmaster:
  - lk\_tablayout.sql - sysptnext, outer systabnames
  - lk\_tabextent.sql - systabnames, sysptnext
  - lk\_chkstatus.sql - sysdbspaces, syschunks
  - lk\_idsconfig.sql - sysconfig
- Save configuration for future reference

# Coming Soon – Scripts to Collect Data

- Check:

<http://www.advancedatools.com/TechInfo/InformixInfo.html>

# Thank You

Lester Knutsen

Email: [Lester@advanceddatatools.com](mailto:Lester@advanceddatatools.com)

Web: <http://www.advanceddatatools.com>

Phone: 703-256-0236 x102

***Advanced DataTools***