

Willkommen zum „IBM Informix Newsletter“

Inhaltsverzeichnis

Aktuelles.....	1
TechTipp: ONSTAT (onstat -g osi).....	2
TechTipp: ONSTAT (onstat -g cpu).....	3
TechTipp: SAVEPOINTS.....	4
TechTipp: Implizite Casts.....	5
TechTipp: SQLTRACE - DATABASE / USER / SESSION.....	7
TechTipp: ONSTAT (onstat -g his) (Auswertung SQLTRACE).....	8
TechTipp: SMI - Sysmaster Interface - syssqltrace.....	9
TechTipp: OpenAdminTool 2 zur Auswertung der Sqltraces.....	10
TechTipp: SQLTRACE - Memory Usage.....	11
TechTipp: IDS auf virtuellen Plattformen.....	12
Hinweis: Vergleich INFORMIX Express / Workgroup / Enterprise Edition.....	12
Hinweis: Chat with the Lab - Archiv.....	12
Termin: Information On Demand EMEA Konferenz.....	13
Termin: Informix Infobahn 21.-23. April in München, Frankfurt, Hamburg.....	14
Anmeldung / Abmeldung / Anmerkung.....	14
Die Autoren dieser Ausgabe.....	14

Aktuelles

Liebe Leserinnen und Leser,

das Jahresende ist immer die Zeit der Jahresrückblicke. Für INFORMIX war es mit der Freigabe der Version IDS 11.50 (Cheetah 2) am 06.Mai 2008 und zahlreichen neuen Installationen eine weitere Fortsetzung der Erfolgsgeschichte. Dies haben wir natürlich Ihnen als treue INFORMIX User zu verdanken. Vielen Dank dafür !

Wir hoffen, das Sie nicht dem Weihnachtsstress unterliegen, sondern die Festtage mit Ihren Familien geniessen können. In diesem Sinne wünschen wir Ihnen und Ihren Lieben Frohe Weihnachten und einen glücklichen und gesunden Start ins Neue Jahr !



Wie immer haben wir für Sie eine Reihe an Tipps und Tricks zusammengestellt.

Viel Spaß mit den Tipps der aktuellen Ausgabe.
Ihr TechTeam

TechTipp: ONSTAT (onstat -g osi)

"onstat -g osi" liefert eine Übersicht über das Betriebssystem, auf dem IDS läuft. Für die Ermittlung der Anzahl der CPUs, des zur Verfügung stehenden Speichers und der Einstellungen der Kernelparameter sind auf den unterschiedlichen Betriebssystemen meist eine Vielzahl von Tools notwendig. Oft können diese zudem nur als Benutzer ROOT aufgerufen werden. Die Ausgabe von onstat -g osi hingegen liefert auf allen unterstützten Plattformen mit einem Befehl alle notwendigen Informationen.

Beispiel:

```
IBM Informix Dynamic Server Version 11.50.UC3      -- On-Line -- Up 00:42:13 --
144144 Kbytes
```

Machine Configuration....

```
OS Name                Linux
OS Release             2.6.18-92.1.13.el5
OS Node Name           kalu.lindau.ibm.com
OS Version              #1 SMP Thu Sep 4 03:51:01 EDT 2008
OS Machine             i686
Number of processors    1
Number of online processors 1
System memory page size 4096 bytes
System memory          2018 MB
System free memory     133 MB
Number of open files per process 1024
shmmax                 536870912
shmmin                 1
shmids                 4096
shmNumSegs             50331648
semmap                 << Unsupported >>
semids                 1024
semnum                 32000
semundo                << Unsupported >>
semNumPerID            250
semops                 100
semUndoPerProc        << Unsupported >>
semUndoSize            20
semMaxValue            32767
```

Die Werte können auch mit SQL aus der Tabelle sysmaster:sysmachineinfo ausgelesen werden:

```
{ machine and operating information }
  create table sysmachineinfo
  (
    os_name           char(128),
    os_release        char(128),
    os_nodename       char(128),
    os_version         char(128),
    os_machine        char(128),
    os_num_procs      smallint,
    os_num_olprocs    smallint,
    os_pagesize       int8,
    os_mem_total      int8,
    os_mem_free       int8,
    ...
```

Zum Vergleich:

uname -a:

```
Linux kalu.lindau.ibm.com 2.6.18-92.1.13.el5 #1 SMP Thu Sep 4 03:51:01 EDT 2008
i686 i686 i386 GNU/Linux
```

cat /proc/cpuinfo:

```
model name      : Genuine Intel(R) CPU          T1300   @ 1.66GHz
```

cat /proc/meminfo

```
MemTotal:      2066432 kB
MemFree:       76228 kB
```

/sbin/sysctl -a:

```
kernel.shmmni = 4096
kernel.shmall = 50331648
kernel.shmmax = 536870912
kernel.sem = 250          32000   100      128
```

TechTipp: ONSTAT (onstat -g cpu)

Der Befehl "onstat -g cpu" liefert eine Übersicht der Statistiken aller laufenden Threads. Interessant für eine Performanceanalyse ist hierbei der Verbrauch an CPU-Zeit, wann der Thread zuletzt aktiv war, sowie die Information wie oft der betreffende Thread aktiv in der Verarbeitung war. Threads, die mit "*" gekennzeichnet sind, können nicht auf einen anderen virtuellen Prozessor (VP) wechseln.

Beispiel:

```
IBM Informix Dynamic Server Version 11.50.UC3      -- On-Line -- Up 00:59:34 --
144144 Kbytes
```

Thread CPU Info:

tid	name	vp	Last Run	CPU Time	#schedules	status
2	lio vp 0	4lio*	11/18 17:50:26	0.4700	153	IO Idle
3	pio vp 0	5pio*	11/18 17:00:26	0.1069	4	IO Idle
4	aio vp 0	6aio*	11/18 17:50:44	24.6252	91200	IO Idle
5	msc vp 0	7msc*	11/18 17:50:34	0.2871	24	IO Idle
6	aio vp 1	8aio*	11/18 17:50:44	0.9240	11316	IO Idle
7	main_loop()	1cpu	11/18 17:51:03	0.0431	3588	sleeping secs: 1
8	soctcpoll	9soc*	11/18 17:51:03	3573.9189	44391	running
9	soctcplst	1cpu*	11/18 17:50:34	0.2691	25	sleeping forever
10	flush_sub(0)	1cpu	11/18 17:51:03	0.0442	3567	sleeping secs: 1
...						
17	flush_sub(7)	1cpu	11/18 17:51:03	0.0097	3564	sleeping secs: 1
18	aio vp 2	11aio*	11/18 17:50:44	0.5538	4386	IO Idle
21	aslogflush	1cpu	11/18 17:51:03	0.5028	3564	sleeping secs: 1
22	btscanner_0	1cpu	11/18 17:50:32	0.0231	144	sleeping secs: 15
38	onmode_mon	1cpu*	11/18 17:51:03	0.0364	3563	sleeping secs: 1
39	periodic	1cpu	11/18 17:51:03	0.0337	3565	sleeping secs: 1
43	CDRSchedMgr	3cpu	11/18 17:50:45	0.0014	119	sleeping secs: 12
45	CDRDTCleaner	1cpu	11/18 17:46:47	0.0008	12	sleeping secs: 44
46	CDRCparse	3cpu	11/18 16:51:41	0.0067	8	cond wait

CDRCparse

49	dbScheduler	3cpu*	11/18	17:50:26	0.1292	71	sleeping secs: 263
50	dbWorker1	3cpu	11/18	17:41:14	0.2203	200	sleeping forever
51	dbWorker2	3cpu	11/18	17:50:26	0.1215	203	sleeping forever
....							
68	sqlexec	1cpu	11/18	17:49:56	16.3456	206070	cond wait netnorm
70	sqlexec	1cpu	11/18	17:50:22	0.0070	56	cond wait netnorm
72	sqlexec	1cpu	11/18	17:51:03	0.8319	3993	cond wait netnorm

Zur Analyse sind meist die Threads vom Typ "sqlexec" interessant. Eine grössere CPU-Time verrät Abfragen, die mehr CPU-Last auf dem System verursachen.

TechTipp: SAVEPOINTS

Mit IDS Version 11.50.xC3 wurde die Funktionalität der **SAVEPOINTS** eingeführt. SAVEPOINTS ermöglichen es, innerhalb einer Transaktion Zustände zu markieren, auf die im Verlauf der Verarbeitung ggf. wieder aufgesetzt werden soll.

Einem SAVEPOINT wird ein Identifier zugeordnet. Wird ein weiterer SAVEPOINT mit dem selben Namen gesetzt, so geht der vorherige SAVEPOINT verloren. Um unbeabsichtigtes Überschreiben eines SAVEPOINTS zu verhindern, kann ein SAVEPOINT mit dem Zusatz "UNIQUE" geschützt werden. Eine neue Vergabe des selben Namens führt in diesem Fall zu einem Fehler. Mit "**ROLLBACK TO SAVEPOINT**" kann auf den Zustand des neuesten SAVEPOINTS zurückgesetzt werden.

Mit "**RELEASE SAVEPOINT <xx>**" können SAVEPOINTS gelöscht werden. Hierdurch werden alle nachfolgenden SAVEPOINTS mit gelöscht.

SAVEPOINTS können nur in Datenbanken mit Transaktionslog genutzt werden und bestehen immer nur innerhalb einer Transaktion. Ein COMMIT bzw. ROLLBACK verwirft somit alle gesetzten SAVEPOINTS.

Beispiel für SAVEPOINTS:

```
BEGIN WORK;
UPDATE customer SET fname = "Marion" WHERE customer_num = 111;
SAVEPOINT mx42;
UPDATE customer SET fname = "Carmen" WHERE customer_num = 111;
SAVEPOINT mx43;
UPDATE customer SET fname = "Carina" WHERE customer_num = 111;
SAVEPOINT mx44;

RELEASE SAVEPOINT mx43;
ROLLBACK TO SAVEPOINT;

COMMIT WORK;

SELECT customer_num, fname FROM customer WHERE customer_num = 111;
```

Ergebnis:

```
customer_num fname
          111 Marion
```

Durch das Release des SAVEPOINTS mx43 wurden dieser und alle folgenden SAVEPOINTS (hier mx44) aufgehoben. Damit ist beim ROLLBACK TO SAVEPOINT der letzte gesetzte SAVEPOINT mx42. Alle Statements vor mx42 bleiben somit erhalten, die Statements danach werden zurückgerollt.

Es besteht zudem die Möglichkeit auf einen bestimmten SAVEPOINT zurückzugehen. Hierzu muss beim ROLLBACK der Name des SAVEPOINTS mit angegeben werden.

Beispiel:

```
BEGIN WORK;

UPDATE customer SET fname = "Marion" WHERE customer_num = 111;
SAVEPOINT mx42;
UPDATE customer SET fname = "Carmen" WHERE customer_num = 111;
SAVEPOINT mx43;
UPDATE customer SET fname = "Carina" WHERE customer_num = 111;
SAVEPOINT mx44;

ROLLBACK TO SAVEPOINT mx43;
COMMIT WORK;

SELECT customer_num, fname FROM customer WHERE customer_num = 111;
```

Ergebnis:

```
customer_num fname
          111 Carmen
```

TechTipp: Implizite Casts

IDS beinhaltet zu vielen Datentypen bereits implizite Casts, so dass die Umwandlung von "kompatiblen" Datentypen meist ohne Interaktion im Datenbankserver erfolgt. Werte vom Datentyp INTEGER können immer in den Datentyp CHAR umgewandelt werden. Dies geschieht implizit im folgenden Beispiel mit der tabid (int), die als Characterstring zum Tabellennamen hinzugefügt wird:

```
select tabid||substr(tabname,1,18)
from systables
where tabid > 99
```

Die Umwandlung von Characterstrings in numerische Werte erfolgt ebenfalls implizit, wenn dies möglich ist. Folgendes Beispiel würde wohl auf 95% der verfügbaren Datenbanksysteme zu einem Fehler der Art "Datatype conversion mismatch" führen. Bei INFORMIX erhält man, so lange die Werte konvertierbar sind, das Ergebnis:

```
create table cast_demo ( f1 char(12) );
insert into cast_demo values ("1");
insert into cast_demo values ("2");
insert into cast_demo values ("3");
update cast_demo set f1 = f1*4 where 1=1;
select * from cast_demo ;
```

Ergebnis:

```
f1
4.00000000000
8.00000000000
12.00000000000
```

Erst wenn ein nicht numerischer Wert im Characterstring steht, führt dies zu einem Fehler:
1213: Character to numeric conversion error

Um bei **Benutzerdefinierten Datentypen (UDTs)** auch implizite Casts zur Verfügung zu haben, müssen diese zuerst erstellt werden.

Das folgende Beispiel beschäftigt einen Teil der Redaktion immer wieder im Alltag:
Der Preisvergleich von Produkten in unterschiedlichen Währungen.

```
create distinct type chf as decimal(7,2);      -- Schweizer Franken
create distinct type eur as decimal(7,2);      -- Euro

create table pizza_de(
  id    int,
  desc  char(18),
  preis eur
);
load from pizza.de insert into pizza_de;      -- Laden der Preisliste in EUR

create table pizza_ch(
  id    int,
  desc  char(18),
  preis chf
);
load from pizza.ch insert into pizza_ch;      -- Laden der Preisliste in CHF

create procedure eur2chf(eur_in eur)          -- Prozedur zur Umrechnung
  returning chf
define chf_out chf;
let chf_out = (eur_in::dec(7,2) / 0.6464)::chf;
return chf_out;
end procedure;

create procedure chf2eur(chf_in chf)          -- Prozedur zur Umrechnung
  returning eur
define eur_out eur;
let eur_out = (chf_in::dec(7,2) * 0.6464)::eur;
return eur_out;
end procedure;

create implicit cast(eur as chf with eur2chf); -- Erstellen des Cast EUR->CHF
create implicit cast(chf as eur with chf2eur); -- Erstellen des Cast CHF->EUR
```

Der implizite Cast wird automatisch vom Server verwendet, um die Daten miteinander verknüpfen zu können. Der Datentyp des ersten Arguments bestimmt dabei die Ausgabeinheit:

```

select pizza_de.desc,                -- Beschreibung
       pizza_de.preis de_eur,        -- Preis in Euro (aus DE-Karte)
       pizza_ch.preis ch_chf,        -- Preis in CHF (aus CH-Karte)
       pizza_ch.preis::eur ch_eur,    -- Expliziter Cast von CHF->EUR
       pizza_de.preis - pizza_ch.preis as diff_in_eur    -- Differenz in EUR
from pizza_de, pizza_ch
where  pizza_de.id = pizza_ch.id

```

Ohne den impliziten Cast wäre dieser Select auf den Fehler

9634: No cast from chf to eur.

gelaufen, da IDS ohne Hinweis keine "Äpfel" mit "Birnen" vergleichen kann.

Ergebnis:

desc	de_eur	ch_chf	ch_eur	diff_in_eur
Tonno	8.90	18.50	11.96	-3.06
Funghi	8.10	15.50	10.02	-1.92
Quattro Formaggi	8.90	18.50	11.96	-3.06
Diavolo	8.60	18.50	11.96	-3.36
Capriciosa	8.90	19.50	12.60	-3.70

TechTipp: SQLTRACE - DATABASE / USER / SESSION

Ab IDS Version 11.50.xC3 wurde das SQL-Tracing um die Funktion USER erweitert. Zuvor war es nur möglich, alle Aktivitäten zu einer Datenbank oder einer Session-Id zu protokollieren. Da die Session-Id erst nach dem Connect zur Verfügung steht, war es damit sehr schwer, die Aktivitäten eines speziellen Benutzers zu erwischen.

Um SQLTRACE zu aktivieren muss entweder in der ONCONFIG ein Eintrag der Form "SQLTRACE level=high,ntraces=10000,size=64,mode=user" vorhanden sein, oder das Tracing mittels SQL-Task nach dem Muster

```
"execute function task ("SET SQL TRACING ON", "<num_traces>", "<trace_size>", "HIGH")"
```

aktiviert werden.

Die eingestellten Werte können abgefragt werden mittels:

```
"execute function task ("set sql tracing user clear");"
```

Das Tracing kann mit "execute function task ("SET SQL TRACING OFF")" ausgeschaltet werden. Auch "SUSPEND" und "RESUME" sind möglich.

Anschliessend können die Benutzernamen angegeben werden, deren SQL-Aktivitäten protokolliert werden sollen. Die Benutzernamen müssen einzeln angegeben werden. Eine Kombination wie "kalu carmen" oder "carina,marion" wird zwar ohne Fehler angenommen, aber es würden nur Benutzer protokolliert, die diese "Doppelnamen" haben".

Das Hinzufügen von Benutzern erfolgt mittels:

```
"execute function task ("set sql tracing user add", "carmen");"
```

```
"execute function task ("set sql tracing user add", "marion");"
```

Abfrage der protokollierten Benutzer:

```
"execute function task ("set sql tracing user list");"
```

Entfernen von einzelnen Benutzern aus der Protokollierung:

```
"execute function task ("set sql tracing user remove", "marion");"
```

Entfernen aller Benutzer aus der Protokollierung:

```
"execute function task ("set sql tracing user clear");"
```

TechTipp: ONSTAT (onstat -g his) (Auswertung SQLTRACE)

Zur Auswertung der SQL-Traces gibt es die Möglichkeiten über "onstat -g his":

Statement history:

```
Trace Level           High
Trace Mode            User
Number of traces     1000
Current Stmt ID      132
Trace Buffer size     65512
Duration of buffer   697 Seconds           (Verbleib in FiFo-Queue)
Trace Flags          0x00007F21
Control Block        0x4b476018
```

Statement # 444: @ 0x4d075018

Database: stores

Statement text:

```
select c.customer_num, c.lname, c.fname, o.order_num
from customer c, orders o
where c.customer_num = o.customer_num
order by lname desc, order_num desc
SELECT using tables [ customer orders ]
```

Iterator/Explain

=====

ID	Left	Right	Est Cost	Est Rows	Num Rows	Partnum	Type
3	0	0	2	23	23	3146074	Seq Scan
4	0	0	1	28	1	3146072	Index Scan
2	3	4	14	23	23	0	Nested Join
1	2	0	1	23	23	0	Sort

Statement information:

Sess_id	User_id	Stmt Type	Finish Time	Run Time	TX Stamp	PDQ
39	500	SELECT	11:18:46	0.0006	1d4292e	0

Statement Statistics:

Page	Buffer	Read	Buffer	Page	Buffer	Write
Read	Read	% Cache	IDX Read	Write	Write	% Cache
0	2	100.00	0	0	2	100.00

Lock	Lock	LK Wait	Log	Num	Disk	Memory
Requests	Waits	Time (S)	Space	Sorts	Sorts	Sorts
2	0	0.0000	192 B	2	0	2

Total	Total	Avg	Max	Avg	I/O Wait	Avg Rows
Executions	Time (S)	Time (S)	Time (S)	IO Wait	Time (S)	Per Sec
24	0.0141	0.0006	0.0009	0.000000	0.000000	35903.3038

Estimated	Estimated	Actual	SQL	ISAM	Isolation	SQL
Cost	Rows	Rows	Error	Error	Level	Memory
14	23	23	0	0	CR	20984



TechTipp: SMI - Sysmaster Interface - syssqltrace

Auch die direkte Abfrage in der Tabelle **sysmaster:syssqltrace** ist möglich:

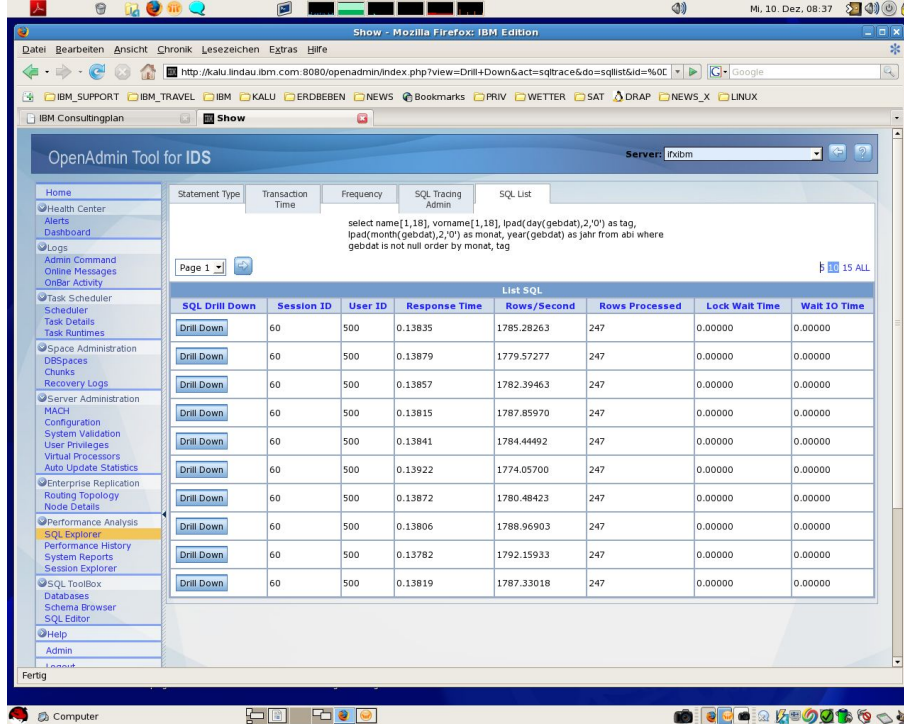
```

sql_id          444
sql_address     1292324888
sql_sid        39
sql_uid       500
sql_stmttype    2
sql_stmtname    SELECT
sql_finishtime  1227176326
sql_begintxtime 30681390
sql_runtime   0.000640807075
sql_pgreads     0
sql_bfreads     2
sql_rdcache   100.0000000000
sql_bfidxreads  0
sql_pgwrites    0
sql_bfwrites    2
sql_wrcache     100.0000000000
sql_lockreq     2
sql_lockwaits   0
sql_lockwttime  0.00
sql_logspace    192
sql_sorttotal   2
sql_sortdisk    0
sql_sortmem     2
sql_executions 24
sql_totalltime 0.014072235743
sql_avgtime   0.000586343156
sql_maxtime   0.000925881608
sql_numioawaits 0
sql_avgioawaits 0.00
sql_totalioawaits 0.00
sql_rowspersec  35892.23793385
sql_estcost     14
sql_estrows     23
sql_actualrows  23
sql_sqlerror    0
sql_isamerror   0
sql_isollevel   2
sql_sqlmemory   20984
sql_numiterators 4
sql_database    stores
sql_numtables   2
sql_tablelist   customer orders
sql_statement   select c.customer_num, c.lname, c.fname, o.order_num
                 from customer c, orders o
                 where c.customer_num = o.customer_num

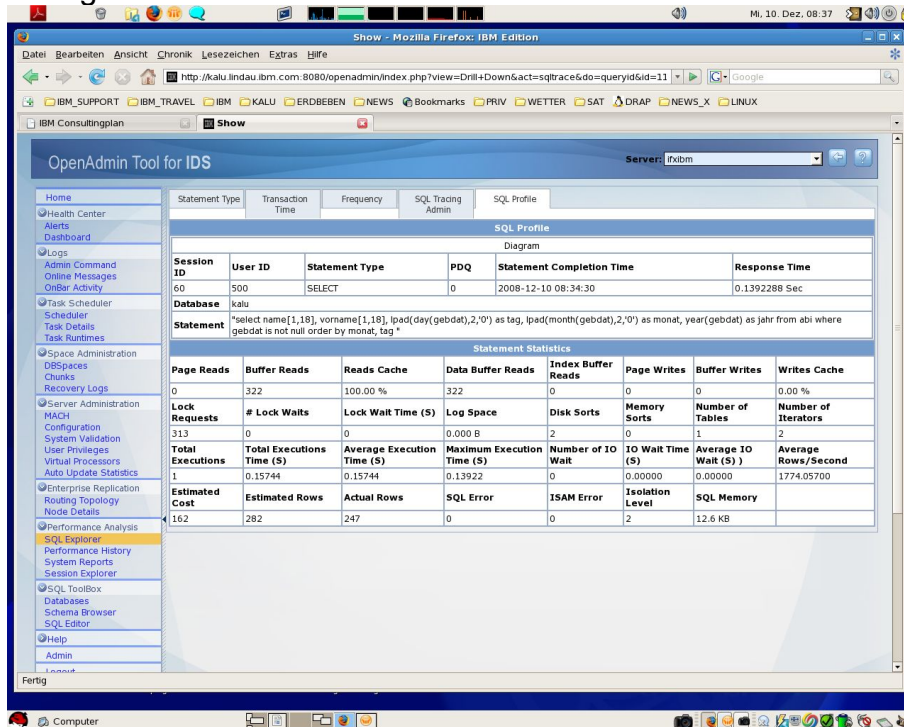
```

TechTipp: OpenAdminTool 2 zur Auswertung der Sqltraces

Im OpenAdminTool werden diese Werte in einer übersichtlichen Tabelle dargestellt.



Der Drilldown liefert Details, wie hier z.B. dass die Cacherate bei einem wiederholten Aufruf bei 100% lag:



TechTipp: SQLTRACE - Memory Usage

Die Daten, die das Tracing der SQL-Anweisungen sammelt, werden im Virtuellen SharedMemory in einem Puffer gehalten, der zyklisch Überschrieben wird. Seine Grösse richtet sich nach der Angabe der zu haltenden Statements (dem Wert von ntraces). So bewirkt allein die Aktivierung des SQLTRACE, dass die Instanz mehr Memory benötigt:

```
IBM Informix ... Version 11.50.UC3 -- On-Line -- Up 01:02:42 -- 144144 Kbytes
```

```
SQLTRACE level=high,ntraces=20000,size=64,mode=user
```

```
IBM Informix ... Version 11.50.UC3 -- On-Line -- Up 00:00:42 -- 1450384 Kbytes
```

Da der zusätzlich benötigte Speicher (im Beispiel 1,25 GB) meist nicht in den durch SHMVIRTSIZE vorgegebenen Bereich passt, allokiert die Instanz anhand der Vorgabe von SHMADD weitere Speichersegmente vom Betriebssystem:

```
IBM Informix Dynamic Server Version 11.50.UC3 -- On-Line -- Up 00:09:55 -- 1450384 Kbytes
```

Segment Summary:

id	key	addr	size	ovhd	class	blkused	blkfree
1933313	52c94801	44000000	114163712	892044	R	27868	4
1966082	52c94802	4ace0000	33439744	197024	V	8164	0
1998898	52c94803	4ccc4000	33439744	197024	V	8164	0
2031667	52c94804	4eca8000	33439744	197024	V	8164	0
....							
3145813	52c94826	928f0000	33439744	197024	V	8164	0
3178582	52c94827	948d4000	33439744	197024	V	8164	0
3211351	52c94828	968b8000	33439744	197024	V	8164	0
3244120	52c94829	9889c000	33439744	197024	V	8164	0
3276889	52c9482a	9a880000	33439744	197024	V	3276	4888
Total:	-	-	1485193216	-	-	357704	4892

Der Thread, der den Speicher hält ist im "onstat -g mem" zu sehen:

```
sqlhistory V 4b475028 1310806016 4280 20002 2
```

Ist SHMTOTAL so gesetzt, dass die Summe der benötigten Segmente größer ist als SHMTOTAL, fährt die Instanz nicht mehr hoch und bricht bei der Initialisierung ab mit:

```
Initializing rhead structure...succeeded
Initialization of Encryption...succeeded
Initializing ASF...succeeded
Initializing Dictionary Cache and SPL Routine Cache...Failed
```

TechTipp: IDS auf virtuellen Plattformen

Viele Nachfragen erhalten wir zum Thema "INFORMIX Dynamic Server" auf virtuellen Plattformen. Die dabei meist gestellte Frage, ob IBM den Support für seine Softwareprodukte auf virtuellen Systemen leistet, kann klar mit JA beantwortet werden. Voraussetzung ist ein gültiger Wartungsvertrag.

Zu diesem Thema gibt es zwei sehr interessante Artikel, die sowohl die Unterstützung von virtuellen Systemen, als auch Tests und Ergebnisse zur Performance beinhalten:

<http://www.ibm.com/developerworks/db2/library/techarticle/dm-0807fuerderer>

und

http://www-01.ibm.com/support/docview.wss?rs=630&context=SSGU8G&context=SSZ2HS&context=SSP6X2&context=SSVHPS&context=SSHPYE&q1=vmware&uid=wws1e333ce0912f7b152852571f60074d175&loc=en_US&cs=utf-8&lang=en

Hinweis: Vergleich INFORMIX Express / Workgroup / Enterprise Edition

Eine kompakte Übersicht in Tabellenform über die unterschiedlichen Versionen der IDS mit:

- Informix Dynamic Server Express Edition
Max: 2 CPUs / 4 GB RAM - verfügbar unter Linux® and Windows®
- Informix Dynamic Server Workgroup Edition
Max: Für V10 & V11 : 4 CPU, 8GB Memory
Für V7.31 & V9.4: 2 CPU, 2GB Memory
Linux, UNIX® and Windows
- Informix Dynamic Server Enterprise Edition
Max: Unlimited
Linux, UNIX® and Windows

ist unter: <http://www-01.ibm.com/software/data/informix/ids/ids-ed-choice> zu finden.

Hinweis: Chat with the Lab - Archiv

Folgende Chats können Sie unter <http://www-01.ibm.com/software/data/informix/labchats> nachlesen:

- IBM solidDB - Complementing Informix to Provide up to 10X Data Acceleration
- Informix Data - Ready, Set, Mash!
- IDS 11.5 Continuous Availability Feature [MACH 11]
- What's New with the OAT (OpenAdmin Tool), a graphical administration for IDS
- Get the Scoop on IBM/Informix Cheetah 2 - from the Product Architects

- Optim - Managing for Data Growth
- Optimizing IDS Performance
- Informix Roadmap and Futures
- The IDS High-Performance Loader
- IBM Data Studio and pureQuery
- HDR and BAR within IDS 11
- Location Based Services and RFID using Web Feature Services
- Using XML with IDS 11
- Tuning the IDS Server
- Continuous Availability in IDS 11
- Label Based Access Control (LBAC) in IDS Cheetah
- Using Sentinel to Monitor and Manage your IDS Server Performance
- Integrated Data Solutions for Information On Demand using IBM IDS 11 'Cheetah'
- Cheetah - Expanding the Administration Free Zone

Termin: Information On Demand EMEA Konferenz

02. - 05.06., Berlin

Die IBM IOD EMEA Konferenz ist das alljährliche Forum für IBM Kunden, Business Partner, Presse, Analysten und IBM Mitarbeiter in Europa, Mittlerem Osten und Afrika. Die Teilnehmer gewinnen auf der IBM Information On Demand Konferenz exklusive Einblicke in strategische Informationsmanagement-Konzepte. Und Sie erfahren, wie Sie Informationen nutzen können, um nachhaltig Wettbewerbsvorteile zu erzielen und Innovation möglich zu machen. Erleben Sie hochkarätige Referenten, Vordenker sowie erfahrene IBM Experten, die Ihnen neueste Erkenntnisse zu einem der wichtigsten strategischen Themen unserer Zeit geben: dem Management von Informationen. Und hören Sie von anderen Unternehmen, wie diese Informationsthemen angehen.

Konferenz Highlights

- Über 2300 Teilnehmer
- 300 Sessions: technisch und vertrieblich ausgerichtet
- Große Fachausstellung
- Executive Meeting Center mit Möglichkeit zu Einzelmeetings mit IBM Executives
- Meet the Experts Sessions
- u.v.m

Weitere Informationen finden Sie unter: ibm.com/software/uk/data/conf

Termin: Informix Infobahn 21.-23. April in München, Frankfurt, Hamburg

Diese kostenlose Veranstaltungsreihe ist bereits seit über 5 Jahren DIE Informationsplattform für alle Informix Kunden, Business Partner und alle Informix Interessierten. Hier werden Spezialisten aus den USA und Deutschland den Teilnehmern Rede und Antwort zu den neuesten Informix Dynamic Server Entwicklungen stehen. Gleichzeitig bieten wir noch den Ausblick auf die nächste Generation des Information Managements. Lassen Sie sich überraschen! Anmeldung und Information in Kürze unter: ibm.com/de/events/infobahn

Anmeldung / Abmeldung / Anmerkung

Der Newsletter wird ausschließlich an angemeldete Adressen verschickt. Die Anmeldung erfolgt, indem Sie eine Email mit dem Betreff „ANMELDUNG“ an ifmxnews@de.ibm.com senden. Im Falle einer Abmeldung senden Sie dies ebenfalls an diese Adresse.

Das Archiv der bisherigen Ausgaben finden Sie zum Beispiel unter:

http://www.iug.de/Informix/techn_news.htm
<http://www.informix-zone.com/informix-german-newsletter>
http://www.nsi.de/index.php?option=com_content&task=view&id=36&Itemid=87
http://www.bytec.de/de/software/ibm_software/newsletter/
<http://www.cursor-distribution.de/index.php/aktuelles/informix-newsletter>
<http://www.drap.de/link/informix>

Die hier veröffentlichten Tipps&Tricks erheben keinen Anspruch auf Vollständigkeit. Da uns weder Tippfehler noch Irrtümer fremd sind, bitten wir hier um Nachsicht falls sich bei der Recherche einmal etwas eingeschlichen hat, was nicht wie beschrieben funktioniert.

Die Autoren dieser Ausgabe

Dr. Elisabeth Bach IT Specialist, Informix Advanced Support
elisabeth.bach@de.ibm.com

Gerd Kaluzinski IT-Specialist Informix Dynamic Server und DB2 UDB
gerd.kaluzinski@de.ibm.com +49-175-228-1983

Martin Fürderer IBM Informix Entwicklung, München
martinfu@de.ibm.com

Annetta Fourkiotis Data Specialist
Annetta.Fourkiotis@de.ibm.com

Ein besonderer Dank geht diesmal an das Team "LaPerla", von dem die Anregung zum Preisvergleich der Pizzen stammt.