

## Willkommen zum „IBM Informix Newsletter“

### Inhaltsverzeichnis

Aktuelles.....	1
TechTipp: Laden von Daten - dbload.....	2
TechTipp: Laden von Daten - onpladm (Teil 1).....	4
TechTipp: SYSALARMPROGRAM.....	6
TechTipp: Flexible Backup Strategie mit On-Bar.....	8
TechTipp: IDS11 - PHYSDBS/PHYSFILE ändern.....	10
Termine: .....	11
Hinweis: Versionsverfügbarkeit.....	13
Nachlese Informix Infobahn 2008.....	13
Anmeldung / Abmeldung / Anmerkung.....	13
Wir über uns: Die Redaktion stellt sich vor – Alexander Körner.....	14
Die Autoren dieser Ausgabe.....	14

### Aktuelles

Liebe Leserinnen und Leser,

im April gingen die Aktionen rund um Cheetah2 weiter.

Die Infobahn Roadshow zur IDS 11.50 erfreute sich sehr hoher Beliebtheit, so dass die Veranstaltungen komplett ausgebucht waren. Bei dieser Gelegenheit konnte man mit hochkarätigen Rednern aus den USA diskutieren und somit direkten Kontakt zur Entwicklung bekommen. Die praktischen Vorführungen zu Cheetah2 und dem DataStudio 2.0 brachten einen Einblick, was uns mit Cheetah2 alles erwartet.

Nachdem wir im Newsletter März dazu aufgerufen hatten, uns Ihre Beiträge aus der Praxis zuzusenden, freuen wir uns, Ihnen schon in dieser Ausgabe einen sehr interessanten Beitrag von Herrn Wolfgang Voss der BARMENIA Versicherungen Wuppertal präsentieren zu dürfen. Sein Thema "Flexible Backup Strategie mit On-Bar" greift eines der wichtigsten Themen bei der Datenbankadministration auf. Herr Voss ist seit 1999 als Administrator für relationale Datenbank Management Systeme bei der Barmenia beschäftigt und setzt seit zwei Jahren die hier vorgestellte Backup Lösung für die IBM Informix Datenbanken ein.

Diese Beiträge aus der Praxis für die Praxis sind äußerst wertvoll. Lassen Sie sich hierdurch dazu inspirieren, auch eigene Lösungen auf der Plattform Newsletter vorzustellen.

Viel Spaß mit den Tipps der aktuellen Ausgabe.

Ihr TechTeam

## TechTipp: Laden von Daten - dbload

Für das Laden von Daten gibt es eine Reihe von Alternativen. Die einfachste Variante ist sicher das SQL-Statement "load from 'daten.unl' insert into Zieltabelle". Hierbei müssen alle Felder des Unload-File geladen werden. Fehler verursachen einen Abbruch und es ist nur ein "Delimited Format" möglich. Der Ladevorgang ist eine umfassende Transaktion, die bei Datenbanken mit Logging schnell zu einer "Long Transaction" führen können. Der dbload kann nur aus einer Unload-Datei lesen und damit nicht parallelisieren.

**dbload** bietet eine Reihe von weiteren Optionen beim Laden an:

- Aufteilen des Ladevorgangs in kleine Transaktionen
- Tolerierung von einer bestimmten Anzahl fehlerhafter Zeilen im Unload-File
- Protokollierung von Fehlern im Unload-File
- Skip einer bestimmten Anzahl an Zeilen am Start des Unload-Files
- Laden mit oder ohne Tabellensperre
- Laden eines "Fixed Format" oder "Delimited Format"
- Mapping von Spalten des Unload-Files auf Spalten der Tabelle

dbload benötigt eine Konfigurationsdatei in der die Informationen zum Unload-File und zur Tabelle stehen.

Usage:

```
dbload [-d dbname] [-c cfilename] [-l logfile] [-e errnum] [-n nnum]
        [-i inum] [-s] [-p] [-r | -k] [-X]
```

```
-d      database name
-c      command file name
-l      bad row(s) log file
-e      bad row(s) # before abort
-s      syntax error check only
-n      # of row(s) before commit
-p      prompt to commit or not on abort
-i      # or row(s) to ignore before starting
-r      loading without locking table
-X      recognize HEX escapes in character fields
-k      loading with exclusive lock on table(s)
```

Die Umgebungsvariable FET\_BUF\_SIZE bestimmt dabei die Größe der Puffer des INSERT CURSORS.

Beispiel zu Delimited Format:

Laden eines Unload-Files (Delimited Format) mit Command-File customer.dbload, bei dem die ersten 5 Zeilen übersprungen werden, nach jedem 3. Datensatz ein Commit erfolgt und bis zu 5 fehlerhafte Datensätze toleriert. Die fehlerhaften Datensätze werden nach xx.log geschrieben. Die Tabelle soll während des Ladevorgangs nicht gesperrt werden.

Das Command-File (hier: customer.dbload):

```
FILE 'customer.unl' DELIMITER '|' 10;
insert into customer_neu;
```

### Ausführung von dbload mit dem Command-File:

```
dbload -d stores -c customer.dbload -l xx.log -i 5 -n 3 -r -e 5
```

### Ausgabe des ausgeführten Kommandos:

```
DBLOAD Load Utility          INFORMIX-SQL Version 11.10.UC2
In INSERT statement number 1 of raw data file customer.unl.
Row number 6 is bad.
106|George|Watson|Watson & Son|1143 Carver Place||
Reached unexpected end of line while reading field

3 Row(s) loaded so far to table customer_neu.
6 Row(s) loaded so far to table customer_neu.
9 Row(s) loaded so far to table customer_neu.
12 Row(s) loaded so far to table customer_neu.
15 Row(s) loaded so far to table customer_neu.
18 Row(s) loaded so far to table customer_neu.
21 Row(s) loaded so far to table customer_neu.
Table customer_neu had 22 row(s) loaded into it.
```

### Beispiel zu Fixed Format:

Laden eines Unloadfiles (Fixed Format) mit Command-File customer.fix, bei dem die ersten 5 Zeilen übersprungen werden, nach jedem 3. Datensatz ein Commit erfolgt und bis zu 5 fehlerhafte Datensätze toleriert werden. Die fehlerhaften Datensätze werden nach xx.log geschrieben. Die Tabelle soll während des Ladevorgangs nicht gesperrt werden.

### Das Command-File (hier: customer.fix):

```
FILE 'customer_fix.unl' (customer_num 1-8, fname 9-23, lname 24-38, company
39-58,
    address1 59-78, address2 79-98, city 99-113, state 114-115,
    zipcode 116-120, phone 121-144 ) ;
insert into customer_neu ( customer_num, fname , lname , company ,
    address1 , address2 , city , state ,
    zipcode , phone ) ;
```

### Der Aufruf:

```
dbload -d stores -c customer.fix -l xx.log -i 5 -n 3 -r -e 5
```

Die Ausgaben sind analog zu denen des Delimited Formats.

## TechTipp: Laden von Daten - onpladm (Teil 1)

Der HighPerformanceLoader (HPL) kann sowohl graphisch mittels ipload als auch aus der Kommandozeile mit onpladm/onpload genutzt werden.

Der HPL kann massiv parallel arbeiten und z.B. parallel in viele Unloadfiles schreiben, bzw. daraus lesen. Optimiert wird der Durchsatz, wenn zudem die Tabelle fragmentiert ist und damit parallel aus unterschiedlichen Dbspaces/Partitions gelesen/geschrieben werden kann. Die Jobs werden in einer Datenbank "onpload" eingetragen die beim ersten Aufruf erstellt wird.

Um eine Tabelle laden/entladen zu können, muss zuerst ein Job angelegt werden:

```
onpladm create job mx42 -D stores -d /tmp/mx42.unl1 -t mx42
Successfully created Job mx42
```

In diesem Beispiel wird ein Job für die Tabelle customer in der Datenbank stores erstellt und als Unloadfile wird /tmp/customer.unl1 benutzt.

Nun kann der Job gestartet werden mittels:

```
onpladm run job mx42 -fl
```

Der HPL holt dann zuerst die Informationen aus der Datenbank onpload und startet dann den Load:

```
onpladm run job mx42 -fl
Connecting to onpload, Please wait...
Successful connection to onpload established
Sat Apr 5 15:35:44 2008

SHMBASE          0x53000000
CLIENTNUM       0x49010000
Session ID 1

Load Database    -> stores
Load Table       -> mx42
Device Array     -> mx42
Record Mapping   -> mx42
Convert Reject   -> /tmp/mx42.rej
Filter Reject    -> /tmp/mx42.flt
15:35:47 Records Processed -> 10000
15:35:47 Records Processed -> 20000
15:35:48 Records Processed -> 30000
15:35:48 Records Processed -> 40000
15:35:49 Records Processed -> 50000
15:35:49 Records Processed -> 60000
15:35:50 Records Processed -> 70000
15:35:50 Records Processed -> 80000
Table 'mx42' will be read-only until level 0 archive

Database Load Completed -- Processed 82533 Records
Records Inserted-> 82533
Detected Errors--> 0
Engine Rejected--> 0
Sat Apr 5 15:35:51 2008
Job Completed Successfully ... connection closed
```

Das Entladen der Daten erfolgt mit dem selben Befehl, einzig statt "-f" wird "-fu". Mehr Devices können entweder bei dem Anlegen des Jobs eingerichtet werden, indem ein DeviceArray angegeben wird (der unterstützte Weg), oder in der Datenbank onpload werden die Einträge in der Tabelle "device" mit fortlaufender Sequenznummer und unterschiedlichen Dateinamen vervielfacht.

Der HPL (im Express Mode - siehe unten) versetzt die Tabelle beim Load in den Mode "raw", so dass der Ladevorgang ohne Logging läuft. Fehlerhafte Datensätze (z.B. Falsche Anzahl an Spalten, Verletzung von Constraints, ...) werden in einer Violations Tabelle gesammelt (mehr Infos zu Violation Tables in einem der nächsten Newsletter).

Nach dem Laden von Tabellen mit dem HPL muss zwingend ein Archiv Level 0 erstellt werden, um ändernd auf die Daten zugreifen zu können.

### HPL-Job für alle Tabellen einer Datenbank

Um den Unload für alle Tabellen einer Datenbank zu erstellen wird ein Projekt erstellt:

```
onpladm create project mx42 -D stores -d /tmp
```

Es wird hierbei keine Tabelle angegeben und statt einem konkreten Unloadfile ein Verzeichnis. Je Tabelle wird dann ein Job erstellt, dessen Unloaddatei den Namen "<datenbank>\_<tabelle>.unl" hat.

Der Aufruf für ein gesamtes Projekt lautet:

```
onpladm run project mx42 -fu          (bzw. -fl für das Laden)
```

Anmerkung:

Jeder Job schreibt eine Logdatei in der die Ausgaben (siehe Beispiel oben) protokolliert werden. Diese Datei hat per Default den Namen <tabelle>.log. Die verworfenen Datensätze sammeln sich per Default in einer Datei <tabelle>.rej. Die Namen dieser beiden Dateien sind konfigurierbar.

Der HPL kann im Express- oder Deluxe-Mode laufen.

Express-Mode:

- Das Laden erfolgt ohne Logging.
- Constraints, Checks, Triggers und Indexe werden unterdrückt (disabled).
- Tabellen mit BLOBs können nicht im Express-Mode geladen werden.
- Die Tabelle wird exklusiv gesperrt.
- Die Rowsize der zu ladenden Datensätze darf die Pagesize nicht übersteigen.

Deluxe-Mode:

- Das Laden erfolgt mit Logging, es kann ein Commit-Interval angegeben werden
- Zugriff auf die Tabelle ist während des Ladens möglich
- Constraints, Indexe, Checks werden auf "FILTERING WITHOUT ERROR" gesetzt

Der onpladm liefert als RETURNCODE entweder 0 für SUCCESS oder -1 für ERROR. Fehlermeldungen werden sowohl an die Console, als auch ins Errorfile gemeldet.

**Achtung: Tabellen mit ETX-Indizes dürfen nicht parallel geladen werden !**

Mehr Tipps zum HPL finden Sie im Teil 2 (PLCONFIG, NonConversionJobs) im nächsten Newsletter.

**TechTipp: SYSALARMPROGRAM**

Verwendung von SYSALARMPROGRAM 'evidence.sh' .

Im Informix Dynamic Server gibt es eine Reihe von Funktionen, die im Fehlerfall Informationen zur Ursache zu erzeugen zur Analyse für den DBA und den technischen Support.

Zentraler Bestandteil ist ein Shell Skript , welches mit dem ONCONFIG Parameter SYSALARMPROGRAM spezifiziert wird. Normalerweise ist es das von IBM Informix ausgelieferte Skript 'evidence.sh', welches in \$INFORMIXDIR/etc liegt.

Das Skript wird bei einem schwerwiegenden Fehler (Assert Failure) vom oninit Prozess direkt aufgerufen. Es führt eine Reihe von 'onstat'-Befehlen aus und schreibt die Ausgabe in eine Datei, welche dem Support anschliessend zur Fehleranalyse zur Verfügung steht.

Manchmal kann es sinnvoll sein, das Skript zu erweitern um zusätzliche Kommandos, z.B. können nach der Zeile 855

```
ev_pretty uname -a
```

folgende Zeilen eingefügt werden

```
ev_pretty ps -eaf  
ev_pretty ipcs -ma
```

um zusätzlich Informationen über alle aktuell ausgeführten Prozesse und die IPC Ressourcen zu bekommen.

Sichern Sie eine Original-Kopie des Skriptes bevor Sie Änderungen vornehmen.

Erweitern Sie das Skript 'evidence.sh' ggf. nach Ihren Bedürfnissen und/oder in Absprache mit dem Informix Support, ersetzen Sie es jedoch nicht durch ein eigenes Shell-Script, da sonst evtl. im Fehlerfall keine brauchbaren Diagnosedaten erzeugt werden !

**! Testen Sie die geänderte Version mit der unten aufgeführten Methode !**

Manchmal ist es wünschenswert eine Reihe von Informationen über den Zustand des IDS zu erhalten, das SYSALARMPROGRAM wird aber nicht automatisch von IDS aufgerufen, da kein 'Assert Failure' ausgelöst wurde, z.B. der IDS Server hängt ohne ersichtlichen Grund.

Sie können in diesem Fall das SYSALARMPROGRAM 'evidence.sh' manuell aufrufen.

Das Skript 'evidence.sh' verlangt eine Reihe von Aufrufparametern. Da es nicht einfach ist, sich diese alle zu merken, können Sie auch das unten angehängte Skript

'gather\_evidence.sh' verwenden. Dieses Skript ruft das Shell-Script \$INFORMIXDIR/etc/evidence.sh mit den erforderlichen Parametern auf und schreibt die Ausgabe in eine Datei im aktuellen Verzeichnis.

Die hier verwendeten Parameterwerte sollten bereits passend sein.

```
-----
#!/bin/sh
# gather_evidence.sh
# a skript to gather diagnostics from a running IDS instance .
# Calls the skript evidence.sh with necessary parameters.
# I tested this script against IDS 11.50 and 10
# Author : Tilman Model-Bosch
# AS-IS !
#include <standard_disclaimer.h>

function usage ()
{
    echo "usage: $0 [diag_file]";
    echo "";
    echo "The script will dump diagnostic data from an IDS instance to
<diag_file>"
    echo "If the file does not exist it will create it in the current directory"
    echo "otherwise it will append it."
    echo "";
    echo "If no filename is given af.getdiag is used as default file name"
    echo "";
    exit 1 ;
}

if [ $# -gt 0 ]
then
if [ $# -gt 1 ] || [ $1 == "--help" ]
then
    usage ;
else
    DIAGFILE=$1
fi
fi

if [ -z $INFORMIXDIR ] || [ -z $ONCONFIG ] || [ -z $INFORMIXSERVER ]
then
echo "INFORMIXDIR, INFORMIXSERVER or ONCONFIG is not set, please ensure your
environment is set correctly"
exit 1
fi

ONSTAT=$INFORMIXDIR/bin/onstat
if [ ! -x $ONSTAT ]
then
echo "Can't execute $ONSTAT, exiting..."
exit 1
fi

EVIDENCE_SCRIPT=$INFORMIXDIR/etc/evidence.sh
if [ ! -x $EVIDENCE_SCRIPT ]
then
echo "$EVIDENCE_SCRIPT not found or not executable"
exit 1
fi

[ -z $DIAGFILE ] && DIAGFILE=af.getdiag
```

```
#Now call evidence skript
echo "Will write diagnostic output to $DIAGFILE"
echo ""
$EVIDENCE_SCRIPT 3 XX $DIAGFILE 0 xx 0 xx 1153
exit 0
```

---

## TechTip: Flexible Backup Strategie mit On-Bar

Wie häufig soll man ein RDBMS sichern?

Im Sinne einer möglichst schnellen Restaurierung von Datenbeständen so oft wie nur irgend möglich - also quasi ständig!

Die Ressourcen eines Storage Managers sind aber andererseits begrenzt.

Es gilt also, einen sinnvollen Kompromiss zu finden zwischen Sicherheit und Ressourceneinsatz.

Die meisten DBAs finden diesen Kompromiss, indem sie ein Zeitintervall abschätzen, nachdem die nächste Sicherung als "nötig" angesehen wird.

Dieses Intervall wird in einem Scheduler eingestellt und fertig ist das Sicherungskonzept.

Z.B. steht dann in der UNIX crontab:

```
# Jeden Samstag um 22 Uhr Vollsicherung:
0 22 * * * 6 onbar -b -w -L 0
# Jeden Mittwoch um 22 Uhr Inkrementsicherung:
0 22 * * * 3 onbar -b -w -L 1
```

Das Konzept enthält aber Unschärfen in beide Richtungen:

Einerseits ist es möglich, dass am Samstag gesichert wird, obwohl sich in der Datenbank kaum etwas geändert hat und die letzte Sicherung vollauf genügen würde.

Andererseits kann es so viel Traffic gegeben haben, dass man statt der Inkrementsicherung am Mittwoch besser eine Vollsicherung gemacht hätte oder (schlimmer), dass ein Point-In-Time-Restore auf Dienstag oder auf Samstag mehr Transaktionen nachfahren muss, als die Produktion zeitlich verkraften kann.

Mit Hilfe der sysutils Datenbank lässt sich der Traffic seit der letzten Sicherung feststellen.

Dazu definiert man am besten eine Sicht auf die erfolgreichen Sicherungen und das aktuelle log Log:

```
CREATE VIEW bar_suc_bkps
AS SELECT act_end, ins_level, ins_chpt_log, ins_verify, uniqid
FROM bar_server, bar_object, bar_action, bar_instance, sysmaster:syslogs
WHERE srv_name = '$INFORMIXSERVER'
AND obj_srv_name = srv_name
AND act_oid = obj_oid
AND ins_aid = act_aid
AND ins_oid = act_oid
AND act_type = 5 -- whole system backup
AND act_status = 0 -- successfull
AND is_current = 1; -- current logical log
```

Dann ist die Anzahl der Logs, die seit der letzten erfolgreichen Sicherung geschrieben wurden:

-- Logs since last backup:

```
SELECT CAST(uniqid - ins_chpt_log AS INT)
  FROM bar_suc_bkps
  WHERE act_end = (SELECT MAX(act_end)
                  FROM bar_suc_bkps
                  WHERE ins_level IN (0, 1));
```

und die Anzahl der Logs seit der letzten Vollsicherung:

-- Logs since last full backup:

```
SELECT CAST(uniqid - ins_chpt_log AS INT)
  FROM bar_suc_bkps
  WHERE act_end = (SELECT MAX(act_end)
                  FROM bar_suc_bkps
                  WHERE ins_level = 0);
```

Nun kann der DBA entscheiden, wie viele log Logs bei einem Restore im schlimmsten Fall gerade noch akzeptabel wären und mit dieser Vorgabe "rechtzeitig" - aber eben auch nicht unnötig früh - ein Backup auslösen.

Häufig behält der Storage Manager Sicherungen nicht endlos, sondern verwirft sie abhängig von einem eingestellten Retention Level nach einer bestimmten Zeit. Wenn dies der Fall ist, ist es natürlich extrem wichtig, dass diese Zeit auf keinen Fall überschritten wird.

Auch hier hilft die oben definierte View:

-- Days since last full backup:

```
SELECT CAST(CAST(TODAY - MAX(act_end) AS CHAR(9)) AS INT)
  FROM bar_suc_bkps
  WHERE ins_level = 0;
```

-- Days since last backup:

```
SELECT CAST(CAST(TODAY - MAX(act_end) AS CHAR(9)) AS INT)
  FROM bar_suc_bkps
  WHERE ins_level IN (0, 1);
```

Ein Backup Skript muss diese Informationen mit definierten Schwellwerten vergleichen. Es kann dann (z.B. stündlich) entscheiden, ob eine Sicherung auszulösen ist oder nicht. So schont man System Ressourcen und ist trotzdem sicher, ein aktuelles Backup zur Verfügung zu haben mit einer maximalen Anzahl an logischen Logs.

Man sollte die Validität seiner INFORMIX Sicherungen auch gelegentlich überprüfen:

**"onbar -v -w"**

Eine sinnvolle Richtlinie könnte etwa lauten:

"Überprüfe nach 90 Tagen oder nach einem Transaktionsaufkommen, das 500 Logs entspricht - je nach dem, was früher eintritt."

Mit dem definierten View ist:

-- Days since last verified backup:

```
SELECT TODAY - MAX(act_end) FROM bar_suc_bkps
WHERE ins_level IN (0, 1) AND ins_verify = 1;
```

-- Logs since last verified backup:

```
SELECT unqid - ins_chpt_log FROM bar_suc_bkps
WHERE act_end = (SELECT MAX(act_end) FROM bar_suc_bkps WHERE ins_level IN (0, 1)
AND ins_verify = 1);
```

Dieser Beitrag wurde uns freundlicherweise von Herrn Wolfgang Voss der BARMENIA Versicherungen Wuppertal zugesandt. Mit seiner Erfahrung seit 1999 als Administrator für relationale Datenbank Management Systeme bei der BARMENIA können wir sicher noch auf eine Reihe weiterer, wertvoller Beiträge aus der Praxis hoffen.

Vielen Dank !

## TechTipp: IDS11 - PHYSDBS/PHYSFILE ändern

Die Parameter PHYSDBS und PHYSFILE zur Bestimmung von Location und Größe des Physical Log können nach dem Initialisieren der Instanz ab der Version 11.10 nur noch mit dem Befehl "onparams" geändert werden. Wird versucht die Parameter in der ONCONFIG zu ändern und die Instanz durchzustarten, dann steht zwar im online.log die Meldung der Änderung (im Beispiel PHYSDBS von physdbs auf datadbs und PHYSFILE von 180000 auf 160000):

```
14:27:59 IBM Informix Dynamic Server Version 11.10.UC2 Software Serial
Number AAA#B000000
14:28:00 IBM Informix Dynamic Server Initialized -- Shared Memory Initialized.
14:28:00 Physical Recovery Started at Page (6:620).
14:28:00 Physical Recovery Complete: 0 Pages Examined, 0 Pages Restored.
14:28:00 Logical Recovery Started.
14:28:00 10 recovery worker threads will be started.
14:28:03 Logical Recovery has reached the transaction cleanup phase.
14:28:03 Logical Recovery Complete.
0 Committed, 0 Rolled Back, 0 Open, 0 Bad Locks

14:28:04 Onconfig parameter PHYSDBS modified from physdbs to datadbs.
14:28:04 Onconfig parameter PHYSFILE modified from 180000 to 160000.
14:28:04 Dataskip is now OFF for all dbspaces
14:28:04 On-Line Mode
```

Es wurde aber weder der Dbspace noch die Größe des Physical Log geändert. Der "onstat -l" zeigt weiterhin die alten Werte:

```
Physical Logging
Buffer bufused bufsize numpages numwrits pages/io
P-1 64 128 63 0 0.00
phybegin physize phypos phyused %used
6:53 90000 567 64 0.08
```



PHYSFILE (physize) zeigt weiterhin 90000 Pages (=180000k) an und die Adresse des PHYSDBS (physbegin) ist weiterhin im Dbspace 6 statt in den Datadbs (im Beispiel Dbspace 2) zu wechseln.

Lassen Sie sich also nicht davon täuschen, dass die Meldung im online.log erscheint. Auf die wahren Werte kommt es an, die mittels "onstat -l" ausgegeben werden können.

## Termine:

### Webcast

#### **IDS Cheetah2 Update 13.05.2008**

**15:00 - 16:00 Uhr**

Dieser technisch orientierte Edu-Call informiert über die neuesten Features und Funktionen in Cheetah 2

Referent: Michael Koester, Informix Channel Technical Pre Sales

Anmeldung an: michael.koester@de.ibm.com

Ansprechpartnerin: Christine Mayer

### **Informix Dynamic Server IDS 11.5 (Cheetah2) Deep Dives**

Diese kostenlose Veranstaltungreihe für technisch Interessierte bietet einen umfassenden Einblick in die Weiterentwicklung des Informix Dynamic Servers V11.5 mit Codenamen Cheetah 2.

Termine:

- **Hamburg**                      16. Juni / 17. Juni
- **München**                      15. Juli / 16. Juli
- **Düsseldorf**                    02. September / 03. September

Themen sind:

- IDS Cheetah 2 Überblick: Eine Erfolgsstory geht weiter...
- Live Demo von Cheetah 2: Der neue Maßstab für hochverfügbare Datenbanken
- Mach 11 Phase 2 im Detail (Update Unterstützung, Load Balancer, Flexible und automatische Fail- over-Konfiguration)
- Weitere, coole Features in Cheetah 2
  - Open Admin Tool Verbesserungen
  - Common API Erweiterungen
  - Neue Features in der Stored Procedure Language (SPL)
- Informix 4GL goes SOA (Neue Features in der klassischen Informix 4GL)

Für die Anmeldung bitte eine kurze E-Mail mit Stichwort: IDS Cheetah 2 Workshop an: **software@de.ibm.com**

Ansprechpartner: Christine Mayer, Michael Köster

In Österreich sind ebenfalls Workshops zu **Informix Dynamic Server IDS 11.5 (Cheetah2) Deep Dives** geplant, die genauen Termine standen allerdings bei

Redaktionsschluss noch nicht fest. Wir werden die Termine so bald als möglich hier veröffentlichen.

### **IBM Breakfast Briefing 2008 mit Software Happy Hour**

**06. - 08.05., Bielefeld**

**27. - 29.05., Köln**

**10. - 12.06., Hamburg**

**24. - 26.06., Berlin**

**08. - 10.07., Karlsruhe**

Nach dem Erfolg und Ihrer Unterstützung der Breakfast Tour im letzten Jahr starten wir jetzt wieder durch. Im letzten Jahr haben 5.000 Teilnehmer (800 mehr als im Jahr davor!) die Breakfasts besucht und sich mit Produktinfos und einem üppigen Frühstück versorgt. Laden Sie Ihre Kunden und Partner auch für 2008 wieder ein - und begleiten Sie sie zum Frühstück.

Wie üblich bieten wir Breakfasts rund um **System i**, **System p**, **System x** und **System z**. Hier präsentieren wir alle Neuigkeiten zur jeweiligen Systemfamilie - und was Sie und Ihre Kunden dabei auch zu Software und Services interessierten könnte.

Wenn Sie noch mehr zu Software wissen möchten:

IBM Software wird 2008 im Rahmen einer **Software Happy Hour** an jedem Ort in Deutschland dabei sein.

Hier gilt: Nach einem entspannten Überblick über die IBM Software werden jeweils Themen wie zum Beispiel SOA, Security, Information on Demand, Web 2.0 oder Collaboration vertieft. Als weiteren Leckerbissen gibt es an jeweils einem Nachmittag die **Storage Happy Hour**. Anhand von Beispielen aus der Praxis erfahren Sie das Neueste zu unseren innovativen Lösungen.

### **Information On Demand EMEA Conference 03.- 06.06., Den Haag, Niederlande**

Die IBM Information On Demand EMEA Conference 2008 ist der größte und wichtigste IBM Event für Kunden, Interessierte und Business Partner im Bereich Information Management in Europa!

Diese englischsprachige Konferenz richtet sich speziell an Führungskräfte und Techniker im Bereich Enterprise Content Management, Data-Servers und Information Plattform in allen Industrien. Insbesondere profitieren von der Konferenz: CEO/ Geschäftsführer, CIO/ CTO, IT Executive, LOB Executive IT Manager, LOB Manager sowie IT-Architekten, Consultants, Datenbankadministratoren, Entwickler, System Administratoren/ Systemprogrammierer und IBM Business Partner.

Anmeldung und weitere Informationen unter:

<http://www.ibm.com/software/uk/data/conf>

## Hinweis: Versionsverfügbarkeit

Für den 15.05.2008 ist die Freigabe der Version **IDS 9.40.xC10** geplant. Die Verfügbarkeit über Passport Advantage und andere Download-Wege kann wie immer einige Tage später sein, da die Zeit für die Bereitstellung noch eingerechnet werden muss.

## Nachlese Informix Infobahn 2008

Auch dieses Jahr konnten sich wieder über 160 Kunden in München, Frankfurt und Hamburg umfassend zu den Neuerungen bei der Entwicklung der Informix Datenbank informieren. Insgesamt haben wir ein sehr positives Feedback von allen Teilnehmern bekommen und die Inhalte waren optimal auf die Anforderungen unserer Kunden und Business Partner abgestimmt. Gleichzeitig nehmen unsere IBM Informix Direktoren aus USA Jerry Keesee und Manuel Corniel die Anregungen und die Diskussionspunkte nun mit auf und werden diese in künftige Entwicklungsschritte mit einbeziehen.

## Anmeldung / Abmeldung / Anmerkung

Der Newsletter wird ausschließlich an angemeldete Adressen verschickt. Die Anmeldung erfolgt, indem Sie eine E-Mail mit dem Betreff „ANMELDUNG“ an [ifmxnews@de.ibm.com](mailto: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.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.nsi.de/index.php?option=com_content&task=view&id=36&Itemid=87)

[http://www.bytec.de/de/software/ibm\\_software/newsletter/](http://www.bytec.de/de/software/ibm_software/newsletter/)

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.

**Wir über uns: Die Redaktion stellt sich vor – Alexander Körner****Alexander Körner**

IBM Software Group, Information Management  
IBM Sales & Distribution, Software Sales

Consulting IT-Specialist, Channel Technical Sales Germany - Informix,  
CTS Brand Focal Point Information Management

Email: [Alexander.Koerner@de.ibm.com](mailto:Alexander.Koerner@de.ibm.com)

**Alexander Körner** ist Diplom-Informatiker (TU-Berlin). Er begann seine berufliche Laufbahn 1989 bei der Informix Software GmbH, die zum 1.7.2001 in die IBM integriert wurde. Bis zur Integration von Informix in die IBM war Alexander Manager des deutschen Ablegers der Informix CTO Gruppe und ist seit Anfang 2002 IT-Specialist im Technical Sales bzw. Channel Technical Sales Information Management der IBM Deutschland GmbH. Alexander's aktuelle Fokusthemen umfassen IDS 11 / Mach 11, virtuelle Linux Appliances basierend auf IDS, IDS und SOA Integration, IDS und SolidDB Integration, Geodatenbanken basierend auf IDS und IDS 11 auf Apple's Mach OS X. Er ist Mitglied in der Gesellschaft für Informatik, Co-Autor mehrerer IBM Redbooks und Whitepapers zum Thema IDS, sowie aktiver Sprecher auf internationalen Konferenzen zum Thema Informix. Zusammen mit dem Niederlassungsleiter der IBM in München, Hans-Hermann Junge betreut er die Beziehung der IBM zur IHK München.

**Die Autoren dieser Ausgabe**

Gerd Kaluzinski IT-Specialist Informix Dynamic Server und DB2 UDB  
[gerd.kaluzinski@de.ibm.com](mailto:gerd.kaluzinski@de.ibm.com) +49-175-228-1983

Thomas Simoner OEM Sales Manager, Informix und DB2  
[thomas.simoner@de.ibm.com](mailto:thomas.simoner@de.ibm.com) +49-0160-907-59034

Tilman Model-Bosch IBM Software Group, Information Management  
IT Specialist , X-Brand Premium Support  
[tilman.model-bosch@de.ibm.com](mailto:tilman.model-bosch@de.ibm.com)

Martin Fürderer IBM Informix Entwicklung, München  
[martinfu@de.ibm.com](mailto:martinfu@de.ibm.com)

**Gastautor:****Wolfgang Voss**

**BARMENIA Versicherungen**  
UNIX-Betrieb  
Datenbankadministration