Willkommen zum Informix Newsletter

Liebe Leserinnen und Leser,

die Zeit der Weihnachtsmärkte ist gekommen. Glühwein, gebrannte Mandeln, Dinnete(*) und viele weitere Leckereien locken auch bei Kälte raus in den Schnee.
Wir wünschen allen Leserinnen und Lesern eine ruhige und besinnliche Zeit, viel Erholung und Freude während der Feiertage mit den Lieben.
Seit Ende November ist die Version 15.0.0.0 verfügbar. Die Tests in der Redaktion laufen. Da diese Ausgabe des Informix Newsletters jedoch zum Zeitpunkt des Erscheinens komplett gefüllt war, dürfen Sie sich auf die Ausgabe Q1/2025 freuen, in der wir den Schwerpunkt auf die Version 15 legen.

In dieser Ausgabe setzen wir den Artikel rund um das Auditing fort. Zudem stellen wir einige nützliche SQL Befehle vor, die man im Alltag selten nutzt, die jedoch die Arbeit erleichtern können.

Frohe Festtage und eine erholsame Zeit !

Ihr TechTeam

Inhaltsverzeichnis

TechTipp: INFORMIX Audit - Audit Masken

Im Newsletter 2024-Q3 haben wir das Informix Audit konfiguriert.
Allein mit der Konfiguration und dem Aktivieren des Audits werden jedoch noch keinerlei Aktivitäten protokolliert. Hierfür müssen Audit Masken erstellt werden.
Es gibt drei reservierte Namen für Audit Masken:

```
_default    # Default für alle User, die keine eigene UserMask besitzen
_require    # Pflichteinträge für alle User protokollieren, unabhängig
            # davon, ob sie eine Usermask besitzen, oder nicht.
_exclude    # Einträge, die für keinen User protokolliert werden.
```

Neben diesen Standardmasken, die nicht per Default im System vorhanden sind, sondern explizit erstellt werden müssen, kann je Loginname eine "Usermask" erstellt werden.
Es besteht die Option eine Maske als Template zu erstellen, die dann als Basis für weiteren "Usermasks" dient.

Die Syntax hierzu ist:

```
onaudit -a -u <mask> -e +<XXXX>,<YYYY>    # Erstellen einer Maske
onaudit -m -u <mask> -e +<XXXX>,<YYYY>    # Ändern einer Maske (Add Event)
onaudit -m -u <mask> -e -<XXXX>,<YYYY>    # Ändern einer Maske (Delete)
onaudit -d -u <mask>                      # Löschen einer Maske
onaudit -a -u <mask> -r <base_mask> -e +<XXXX>  # Create mit Template

onaudit -o -u <mask>                      # Anzeigen einer Maske
onaudit -o -y                             # Anzeigen aller Masken
```

Beispiel für die Erstellung der "Standardmasken":

```
onaudit -a -u _default -e +CRTB,DRTB      # Create Table, Drop Table
onaudit -a -u _require -e +FACTB          # Failed Access Table
onaudit -a -u _exclude -e +SACTB          # Successful Access Table
```

Eine Liste der möglichen Events mit Erklärungen ist in der Dokumentation zu finden.
Jedes Event kann mit vorangestelltem "F" für die fehlerhaften Zugriffe, oder mit vorangestelltem "S" für die erfolgreichen Zugriffe näher spezifiziert werden.

Die Auswahl der Events bestimmt den Umfang der Audit Protokolle.
Zu detailliertes Auditing, das Millionen Events je Sekunde erzeugt, kann zu erheblicher Beeinträchtigung der Performance führen.

Für die nachfolgenden Beispiele erstellen wir zwei Masken:

```
onaudit -a -u kalu -e +CRTB,DRTB,CRIX,DRIX
      # Create/drop Table, create/drop Index
onaudit -a -u informix -e +CRAM,DRAM,UPAM,ONAU,ONMO,....
      # Create/drop/update Auditmasks, Call onaudit
```

Mittels "onaudit -o" lassen sich die Audit Masken anzeigen:

```
_default      -    CRTB,DRTB
_exclude      -    ACTB
informix      -    ADCK,CRAM,CRTB,DRAM,ONAU,ONCH,ONLG,ONMO,ONST,ONTP,UPAM
kalu          -    CRIX,CRTB,DRIX,DRTB,ONCH,RDRW
```

TechTipp: Audit Auswertung - onshowaudit

Der Befehl "onshowaudit" bietet die Möglichkeit die Audit Daten auszugeben.
Die Optionen sind:

- -f <file> # Lesen einer ausgewählten Audit Datei
- -l # Ausgabe der Audit Informationen als "Delimited Records", um diese anschliessend in eine Datenbank zu laden.
- -n <num> # Lesen der Audit Konfiguration aus der Datei adtcfg.<num>
- -q # Ausgabe der Informationen ohne Header
- -s <name> # Audit Informationen nur zu einem Server $INFORMIXSERVER
- -u <user> # Audit Informationen nur zu einem User anzeigen

Die Ausgabe des "onshowaudit -l -q -n 0" zeigt für das Beispiel folgende Ausgaben:

```
ONLN|2024-11-11 15:07:22.279|kalu42|17537|ifx42|informix|0|ONMO|||||||||-ky |
ONLN|2024-11-11 15:07:44.333|localhost|17635|ifx42|informix|0|ONAU|||||||||-c|
ONLN|2024-11-11 15:13:37.893|localhost|19771|ifx42|informix|0|ONAU|||||||||-n|
ONLN|2024-11-11 15:26:46.789|localhost|24455|ifx42|informix|0|UPAM|||||||kalu|||
ONLN|2024-11-11 15:26:52.944|localhost|23659|ifx42|kalu|0|CRTB|test|217|test_tab||||kalu|0|-|
ONLN|2024-11-11 15:26:52.957|localhost|23659|ifx42|kalu|0|CRIX|test|217|i_test_tab||||kalu|1|datadbs|
ONLN|2024-11-11 15:26:52.966|localhost|23659|ifx42|kalu|0|DRTB|test|217|test_tab||||kalu|0|5245316|
```

ONMO	OnMode
ONAU	OnAudit
UPAM	Update Audit Mask
CRTB	Create Table
CRIX	Create Index
DRTB	Drop Table

User, der die Aktion ausgeführt hat.
Datenbank, auf der die Aktion durchgeführt wurde.

TechTipp: Row Level Audit

Das Row Level Audit ermöglicht das Auditing auf Row Ebene.
Hierfür wird der Parameter ADTROWS in der Konfigurationsdatei zum Audit genutzt.
Mögliche Werte sind:

- 0 Für Alle Tabellen wird das Row Level Audit aktiviert
- 1 Für Tabellen, die mit Audit versehen sind, werden die Aktivitäten protokolliert (create table … **with audit** bzw. alter table … **add audit**)
- 2 Für Tabellen, die mit Audit versehen sind. Zusätzlich wird der Primary Key ausgegeben, falls dieser vom Typ Integer ist, ansonsten die Rowid.

Beispiel SQL:

```
create table kunden (
kdnr  char(6),
kdname      char(40),
kdlnd char(3),
kdplz char(6),
kdort char(40)
) with audit;

insert into kunden values (1,"Kalu","D","88131","Lindau");
insert into kunden values (2,"Schneider","A","6922","Wolfurt");
insert into kunden values (3,"Sander","D","80921","Muenchen");

select * from kunden where kdlnd="A";
update kunden set kdort = "Ebersberg" where kdplz="80921";
delete from kunden where kdplz="80921";
```

Beispiel der Ausgabe:

```
ONLN|2024-11-12 12:09:04.087|localhost|80498|ifx42|kalu|0|CRTB|test|220|kunden||||kalu|0|-|
ONLN|2024-11-12 12:09:04.112|localhost|80498|ifx42|kalu|0|INRW|test|220|||5245324|257||||
ONLN|2024-11-12 12:09:04.120|localhost|80498|ifx42|kalu|0|INRW|test|220|||5245324|258||||
ONLN|2024-11-12 12:09:04.129|localhost|80498|ifx42|kalu|0|INRW|test|220|||5245324|259||||
ONLN|2024-11-12 12:09:04.137|localhost|80498|ifx42|kalu|0|RDRW|test|220|||5245324|258||||
ONLN|2024-11-12 12:09:04.138|localhost|80498|ifx42|kalu|0|RDRW|test|220|||5245324|259||||
ONLN|2024-11-12 12:09:04.138|localhost|80498|ifx42|kalu|0|UPRW|test|220||5245324|5245324|259||259||
ONLN|2024-11-12 12:09:04.146|localhost|80498|ifx42|kalu|0|RDRW|test|220|||5245324|259||||
ONLN|2024-11-12 12:09:04.146|localhost|80498|ifx42|kalu|0|DLRW|test|220|||5245324|259||||
```

CRTB Create Table
INRW Insert Row
RDRW Read Row
UPRW Update Row (hier wird die bisherige und die neue Rowid angezeigt)
DLRW Delete Row

Anmerkung der Redaktion:
Im Test wurde im Gegensatz zur Dokumentation immer die Rowid ausgegeben, nie der Primary Key, auch wenn dieser als "Integer" definiert war.

TechTipp: Unterschied adtcfg und adtcfg.<servernum>

Im Verzeichnis $INFORMIXDIR/aaodir gibt es eine Datei "adtcfg" in der die Default Audit Konfiguration abgelegt ist.
Ist das Auditing einer Instanz zur Informix Installation aktiv, **so muss diese Datei existieren**, ansonsten startet die Instanz nicht mit der Fehlermeldung:

```
WARNING: Cannot access configuration file /opt/informix/aaodir/adtcfg.
```

Die eigentliche Audit Konfiguration einer Instanz ist in der Datei adtcfg.<servernum> abgelegt.
Diese muss beim Start der Instanz nicht vorhanden sein, ist jedoch notwendig, damit das Auditing aktiv werden kann.
Durch diese Konfiguration ist es möglich, dass alle Instanzen, die zu zur selben Software im $INFORMIXDIR erstellt wurden, individuelle Audit-Konfigurationen besitzen können.

Achtung:
Wird beim "onshowaudit" nicht die Option "-n <servernum>" mitgegeben, so wird der AuditPath aus der Datei adtcfg genutzt, statt aus dem Pfad, der in der Datei zum Server festgelegt ist.

TechTipp: Audit Records im JSON Format

Das JSON Format wird in der Audit Konfiguration durch den Parameter ADT_CADF_ENABLED gesetzt, bzw. mittels "onaudit -C 1" aktiviert.
Ab diesem Zeitpunkt werden neben dem bisherigen Format zusätzlich Audit Logs im JSON Format geschrieben, das sich am den Standards der Cloud Auditing Data Federation (CADF) orientiert.
Wird Informix im "Cloud Pak for Data" genutzt, so werden die Records automatisch an den Audit Logging Service des CP4D geschickt.

Beispiel:


```
{ "outcome": "success", "typeURI": "http://schemas.dmtf.org/cloud/audit/1.0/event", "eventType":
"activity", "eventTime": "0024-11-12T13:15:18.041166+0000", "action": "Informix.database.RDRW ",
"severity": "normal", "initiator": { "id": "unknown", "name": "kalu", "typeURI": "service/secu-
rity/account/user", "credential": { "type": "user" } }, "target": { "id": "", "name": "ibm-in-
formix", "typeURI": "service/informix", "host": { "address": "" } }, "observer": { "name": "Common-
AuditService",   "id": "userActivity" }, "reason": { "reasonCode": 200 }, "attachments": [ { "con-
tentType": "service/content/cloudpak", "name": "Informix", "content": { "message": "Informix.data-
base.RDRW  0:RDRW:test:225:5245324:259::", "sourceCrn": "crn:v1::private:Informix::::", "kuber-
netes": { "container_id": "", "container_name": "informix-server", "pod": "", "namespace": "" } } }
] }
{ "outcome": "success", "typeURI": "http://schemas.dmtf.org/cloud/audit/1.0/event", "eventType":
"activity", "eventTime": "0024-11-12T13:15:18.041544+0000", "action": "Informix.database.DLRW ",
"severity": "normal", "initiator": { "id": "unknown", "name": "kalu", "typeURI": "service/secu-
rity/account/user", "credential": { "type": "user" } }, "target": { "id": "", "name": "ibm-in-
formix", "typeURI": "service/informix", "host": { "address": "" } }, "observer": { "name": "Common-
AuditService",   "id": "userActivity" }, "reason": { "reasonCode": 200 }, "attachments": [ { "con-
tentType": "service/content/cloudpak", "name": "Informix", "content": { "message": "Informix.data-
base.DLRW  0:DLRW:test:225:5245324:259::", "sourceCrn": "crn:v1::private:Informix::::", "kuber-
netes": { "container_id": "", "container_name": "informix-server", "pod": "", "namespace": "" } } }
] }
```

TechTipp: Audit ins SYSLOG

Mit Version 14.10.FC6 wurde die Funktionalität des Audit nach "syslog" ermöglicht.
Diese Option wurde von einigen grösseren Kunden nachgefragt, die diese eine Vielzahl an Servern zentral über die Meldungen des syslog überwachen.

Das Aktivieren des Audits nach Syslog erfolgt mittels des Parameters ADT_SYSLOG_ENABLED in der Audit Konfiguration, bzw. mittels "onaudit -E 1".
Zudem kann ein Identifyer gesetzt werden, um die Einträge im syslog besser unterscheiden und extrahieren zu können.

```
onaudit -E 1
onaudit -I XXXADTXXX
```

Zudem kann die Priorität gewäht werden (Default ist LOG_USER):

```
onaudit -P LOG_DEBUG
```

Im syslog sind die Audit Einträge dann in folgender Form zu finden:

```
grep XXXADTXXX /var/log/syslog
Nov 12 14:26:20 kalu42 XXXADTXXX: ONLN|2024-11-12 14:26:20.798|localhost|94804|kalu|infor-
mix|0:ONAU:-I XXXADTXXX
Nov 12 14:26:34 kalu42 XXXADTXXX: ONLN|2024-11-12 14:26:34.226|localhost|95002|kalu|infor-
mix|0:ONAU:-c
Nov 12 14:26:55 kalu42 XXXADTXXX: ONLN|2024-11-12 14:26:55.984|local-
host|95074|kalu|kalu|0:DRTB:test:225:kunden:kalu:0:5245324
Nov 12 14:26:56 kalu42 XXXADTXXX: ONLN|2024-11-12 14:26:56.054|local-
host|95074|kalu|kalu|0:CRTB:test:226:kunden:kalu:0:-
Nov 12 14:26:56 kalu42 XXXADTXXX: ONLN|2024-11-12 14:26:56.062|local-
host|95074|kalu|kalu|0:INRW:test:226:5245324:257::
Nov 12 14:26:56 kalu42 XXXADTXXX: ONLN|2024-11-12 14:26:56.070|local-
host|95074|kalu|kalu|0:INRW:test:226:5245324:258::
Nov 12 14:26:56 kalu42 XXXADTXXX: ONLN|2024-11-12 14:26:56.079|local-
host|95074|kalu|kalu|0:INRW:test:226:5245324:259::
Nov 12 14:26:56 kalu42 XXXADTXXX: ONLN|2024-11-12 14:26:56.087|local-
host|95074|kalu|kalu|0:RDRW:test:226:5245324:258::
Nov 12 14:26:56 kalu42 XXXADTXXX: ONLN|2024-11-12 14:26:56.088|local-
host|95074|kalu|kalu|0:RDRW:test:226:5245324:259::
Nov 12 14:26:56 kalu42 XXXADTXXX: ONLN|2024-11-12 14:26:56.088|local-
host|95074|kalu|kalu|0:UPRW:test:226:5245324:259:5245324:259::
Nov 12 14:26:56 kalu42 XXXADTXXX: ONLN|2024-11-12 14:26:56.095|local-
host|95074|kalu|kalu|0:RDRW:test:226:5245324:259::
Nov 12 14:26:56 kalu42 XXXADTXXX: ONLN|2024-11-12 14:26:56.096|local-
host|95074|kalu|kalu|0:DLRW:test:226:5245324:259::
```

TechTipp: Datenbank für Audit Protokolle

Die Audit Daten können im "Unload Format" ausgelesen werden.
Damit ist es möglich die Daten in einer Audit Datenbank zu speichern und ggf.
bei Bedarf mittels SQL die Aktivitäten zu einem User bzw. auf einer bestimmten Tabelle zu ermitteln.

Das folgende DDL kann die Daten der Version 14.10 aufnehmen:

```
CREATE TABLE audit_logs (
      adttag CHAR(4) NOT NULL,
      date_time DATETIME YEAR TO FRACTION(3) NOT NULL,
      hostname VARCHAR(128) NOT NULL,
      pid INTEGER NOT NULL,
      server VARCHAR(128) NOT NULL,
      username VARCHAR(32) NOT NULL,
      errno INTEGER NOT NULL,
      code CHAR(4) NOT NULL,
      dbname VARCHAR(128),
      tabid INTEGER,
      objname VARCHAR(128),
      extra_1 INTEGER,
      partno INTEGER,
      row_num INTEGER,
      login VARCHAR(32),
      flags INTEGER,
      extra_2 VARCHAR(160)
);
```

Beim Einlesen empfehlen wir einen Merge, so dass nur die neue Daten ergänzt werden.

TechTipp: DBSPACETEMP am RSS Server

Am RSS Server ist es möglich einen konsistenten Export der Datenbank zu erstellen, indem mittels der Parameter STOP_APPLY=1 gesetzt wird (mittels "onmode -wf"), und damit alle Schreibvorgänge angehalten werden, bis der Export der daten konsistent beendet wurde.

Wichtig ist, dass auf dem RSS Server ein Temporärer DBSpace existiert, und dieser in der $ONCONFIG als DBSPACETEMP eingetragen ist.
Sollte dies nicht der Fall sein, so kommt es zu folgender Fehlermeldung:

```
"The operation cannot be performed on this RS secondary server because no
 temporary dbspace exists. Set the DBSPACETEMP configuration parameter to a
 valid temporary dbspace, then rerun this command"
```

Ein Temporärer DBSpace kann nicht auf dem RSS Server erstellt werden. Dies erfolgt am Primary Server, indem der Plattenbereich für den Chunk vorbereitet wird, und der TempDBSpace am Primary server mittels "onspaces" mit der Option "-t" erstellt wird.
Das anschliessende Eintragen in die $ONCONFIG kann mittels "onmode -wf" erfolgen.

TechTipp: SQL - MINUS Operator

Im Newsletter Juni 2013 hatten wir bereits den Operator EXCEPT vorgestellt. Obwohl dieser Operator sehr hilfreich sein kann, wird er nach unserer Erfahrung eher selten genutzt. MINUS ist ein Synonym für Except und kann für den Ausschluss von Werten eingesetzt werden.

Beispiel (in der Datenbank stores):

```
select count(*) from customer;        # 28 Einträge
select count(*) from orders;          # 23 Einträge
select count(*) from no_commercials;  #  4 Einträge
```

Ermitteln der Kunden, die noch nie bestellt haben und nicht Werbung ablehnen:

```
-- Auswahl Aller Kunden
select customer_num
from customer
MINUS                -- ohne die Kunden, die schon bestellt haben

select customer_num
from orders
MINUS                -- ohne die Kunden, die Werbung untersagt haben

select customer_num
from no_commercials
order by 1
```

Ergebnis:

```
 customer_num
    102
    103
    108
    113
    114
    118
    125
```

Meist findet sich die folgende Schreibweise, die zum identischen Ergebnis führt:

```
select customer_num
from customer
where customer_num not in (
   select customer_num
   from orders
)
and customer_num not in (
   select customer_num
   from no_commercials
)
```

Die Schreibweise mit "MINUS" bietet den Vorteil, dass alle Teilstatements einzeln getestet und analysiert werden können. Gerade bei komplexen Berechnungen zahlt sich eine übersichtliche Programmierung aus und hilft Fehler zu vermeiden.

TechTipp: SQL - Zeilennummern generieren

Die Verwendung der Funktion rownumber() bei OLAP Funktionen haben wir bereits im Newsletter Juni 2013 vorgestellt. Diese steht allerdings für herkömmliche SQL Abfragen nicht zur Verfügung.
Ein einfacher Weg, eine Zeilennummer vor jede Zeile der Ausgabe zu erzeugen, ist die Verwendung einer Sequence.
Hierbei wird vor der Abfrage eine Sequence erstellt, die nach der Abfrage wieder verworfen wird.

Beispiel: Kunden aus der Tabelle "customer" mit vorangestellter Zeilennummer ausgeben:

```
create sequence row_num_gen increment by 1 start with 1;
select row_num_gen.nextval as nr, customer_num, lname, fname
from customer
order by customer_num;
drop sequence row_num_gen;


 nr customer_num lname           fname

 1          101 Pauli           Ludwig
 2          102 Sadler          Carole
 3          103 Currie          Philip
 4          104 Higgins         Anthony
 5          105 Vector          Raymond
 6          106 Watson          George
 7          107 Ream            Charles
 8          108 Quinn           Donald
```

Optional können die Werte der Seqence auch angepasst werden, um z.B. die Nummerierung konsistent fünfstellig auszugeben:

```
create sequence row_num_gen increment by 1 start with 100001;
select substr(row_num_gen.nextval,-5)::char(5) as nr, customer_num,
     lname, fname
from customer
order by customer_num;
drop sequence row_num_gen;

nr    customer_num lname           fname

00001          101 Pauli           Ludwig
00002          102 Sadler          Carole
00003          103 Currie          Philip
00004          104 Higgins         Anthony
00005          105 Vector          Raymond
00006          106 Watson          George
00007          107 Ream            Charles
00008          108 Quinn           Donald
```

TechTipp: SQL - SEQUENCE - ORDER/NOORDER

Sequences werden bei INFORMIX immer sortiert erstellt, so dass es keinen Unterschied macht, ob ORDER oder NOORDER bei der Anlage der Sequenz mitgegeben wird. Dieser Parameter dient lediglich der Kompatibilität zu anderen SQL Dialekten und wird ignoriert.

Hinweis: Installation V14.10.xC11 benötigt Java

(Ein Hinweis von Fedor Schönhals - IT Beratung fedor@it-schoenhals.de)

Ab Informix Version 14.10.xC11 wird keine Java Umgebung mehr mitgeliefert. In den Release Notes ist zu finden, dass Java 1.8 oder neuer Voraussetzung für die Installation ist.
Unter AIX ist zudem zu beachten, dass auch das Java Service Release mindestens Version 8 besitzt.

Hier der Hinweis vom IBM Support:

> 14.10.FC11 does not come bundled with jre/java any more.
> You need to make sure one of the below java versions are used and in your $PATH
>
> Java 1.8 Latest SR which either SR8 or SR9 (14.10.FC10W2 comes bundled with 1.8.0 SR8)
> Java 11 with it's latest SR for their OS version
> Java 17 with it's latest SR for their OS version
> When using either:
> Java 11 or later than 11.0.19
> Or
> Java 17 or later than 17.0.7
> Or
> any later version of JAVA,
> you must set the additional java option to command line before the -jar option:
> **-Djdk.util.zip.disableZip64ExtraFieldValidation=true**

https://www.ibm.com/support/pages/how-resolve-error-fatal-error-couldnt-determine-supported-encryption-library-while-installing-1410fc11

TechTipp: INFORMIX Version 15.0.0.0 verfügbar

Seit dem 19. November ist die Version 15.0.0.0 allgemein verfügbar.
Mit dieser Version wurde eine neue Lifecycle Strategie eingeführt "Support-Cycle-5":
5 years of support with critical fix extension of 1 year and usage and existing fixes for 3 (5+1+3).
https://www.ibm.com/support/pages/node/7173249

Erste Tests der Migration waren erfolgreich:

```
11/20/2024 15:52:40   A restore point was created and data is being stored in directory
/opt/informix_15.0.0.0/tmp/163.
11/20/2024 15:52:40   Conversion control block initialization
11/20/2024 15:52:40   Succeeded
11/20/2024 15:52:40   Initialization of Conversion Session
...
11/20/2024 15:52:41   Conv/rev: Finished conversion for component RSAM to component version 50
11/20/2024 15:52:41   Update reserved page status of conversion
11/20/2024 15:52:41   Succeeded
...
11/20/2024 15:52:42   Quiescent Mode
11/20/2024 15:52:42   (Compatibility mode)
...
11/20/2024 15:52:42   Converting database sysmaster ...
11/20/2024 15:52:42   The database sysmaster has been converted successfully.
...
11/20/2024 15:52:42   Converting database sysutils ...
11/20/2024 15:52:43   The database sysutils has been converted successfully.
...
11/20/2024 15:52:43   Converting database sysuser ...
11/20/2024 15:52:43   The database sysuser has been converted successfully.
...
11/20/2024 15:53:35
             *   * *   * *   * *   * *   * *   *
              * *   * *   * *   * *   * *   * *
               *     *     *     *     *     *
              Conversion Completed Successfully
               *     *     *     *     *     *
              * *   * *   * *   * *   * *   * *
             *   * *   * *   * *   * *   * *   *

11/20/2024 15:53:35   On-Line Mode
11/20/2024 15:53:35   (Compatibility mode)
```

Danach ist die Instanz nutzbar:

```
IBM Informix Dynamic Server Version 15.0.0.0AEE -- On-Line -- Up 00:02:23 -- 271660 Kbytes
2024-11-20 15:54:56 -- Infrastructure Version: 0 (Compatibility mode)
```

Ist man sicher, dass man nicht mehr mittels Reversion auf die Version 14.10 zurück will, dann kann die Infrastructure Version auf 1 gesetzt werden:

```
onbar -b -F -IV 1
```

In online.log wird dies mit einer Zeile notiert:

```
11/20/2024 15:55:24   Version 1 infrastructure has been enabled
```

Jeder Aufruf von "onstat" zeigt die Infrastructure Version:

```
IBM Informix Dynamic Server Version 15.0.0.0AEE -- On-Line -- Up 00:05:20 -- 271660 Kbytes
2024-11-20 15:57:52 -- Infrastructure Version: 1
```

TechTipp: INFORMIX Version 15.0 Edition Installer

Bei der Installation der Version 15 auf Linux kam es vereinzelt zu Problemen, dass der License Installer nicht von der Developer Edition auf die im JAR-File angegebene Edition wechseln konnte. Der Edition Installer endet ohne Fehlermeldung, aber ein "onstat" zeigt, dass weiterhin die Developer Edition (15.0.0.0DE) installiert ist.

Abhilfe bringt ein Link im $INFORMIXDIR/lib:

```
ln -s libisi.so.3 -> ./libisi_gsk.so.3.0
```

Danach den Edition Installer erneut aufrufen z.B. für die Enterprise Edition:

```
$INFORMIXDIR/jvm/jre/bin/java -jar ee_edition.jar
```

und die Instanz durchstarten.

Nutzung des INFORMIX Newsletters

Die hier veröffentlichten Tipps&Tricks erheben keinen Anspruch auf Vollständigkeit.
Die IUG hat sich dankenswerterweise dazu bereit erklärt, den INFORMIX Newsletter auf ihren Web Seiten zu veröffentlichen.
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.
Rückmeldungen hierzu sind herzlich Willkommen !
Die gefundenen Tippfehler dürfen zudem behalten und nach Belieben weiterverwendet werden.
Eine Weiterverbreitung in eigenem Namen (mit Nennung der Quelle) oder eine Bereitstellung auf der eigenen HomePage ist ausdrücklich erlaubt. Alle hier veröffentlichen Scripts stehen uneingeschränkt zur weiteren Verwendung zur Verfügung.

Die Autoren dieser Ausgabe

Dr. Andreas Legner **INFORMIX Advanced Support**
HCL Software

Dr. Elisabeth Bach **INFORMIX Advanced Support**
HCL Software

Dr. Martin Fuerderer **Database Development**
HCL Software

Gerd Kaluzinski **IBM Expert Labs**
Data & AI
gerd.kaluzinski@de.ibm.com +49-175-228-1983

Der Hinweis zur Installation bezüglich Java stammt von:
Fedor Schönhals **IT Beratung** **fedor@it-schoenhals.de**

Liebevollen Dank an die vielen Helfer im Hintergrund.

Nicht zu vergessen der Dank an die Informix User Group, ohne die es den INFORMIX Newsletters heute nicht mehr geben würde, und die dankenswerterweise die Verteilung übernimmt.

Foto Nachweis:
Weihnachtsmakrt Bregenz 2024 (Gerd Kaluzinski)

(*) Dinnete, eine Art Flammkuchen, aber auch wieder nicht. Wer mehr darüber erfahren will, dem empfehle ich einen Besuch auf dem Weihnachtsmarkt in Bregenz. Gleich am Stand gegenüber gibt es Glühwein (Private Führung nach Voranmeldung ist möglich).
Weitere Details:
https://de.wikipedia.org/wiki/Dinnete