

## Willkommen zum Informix Newsletter der IUG Deutschland

### Inhaltsverzeichnis

|  |    |
|--|----|
| Aktuelles.....   | 2  |
| TechTipp: Connection Manager Options.....                            | 3  |
| TechTipp: ONCONFIG 14.10.xC3 – Unterschiede zur Version 12.10.....   | 8  |
| TechTipp: ONCONFIG – DIRECT_IO für TempSpaces.....                   | 9  |
| TechTipp: .NET Core Compatibility.....                               | 10 |
| TechTipp: Wire Listener Enhancements ab 14.10.xC4.....               | 10 |
| TechTipp: Spatio-Temporal Enhancements ab 14.10.xC4.....             | 10 |
| TechTipp: Huge Pages – Linux Kernel Parameter.....                   | 10 |
| TechTipp: INFORMIX im Docker installieren.....                       | 11 |
| TechTipp: Anzeige von Spatialdaten aus INFORMIX mittels R.....       | 12 |
| TechTipp: Memory Cleaning bei Linux.....                             | 14 |
| TechTipp: MSG_DATE.....  | 15 |
| TechTipp: onstat -g top.....   | 16 |
| TechTipp: UUID() - Function.....                                     | 18 |
| TechTipp: INFORMIX Drivers - Webseite.....                           | 18 |
| TechTipp: Connection Manager – FOC vom Primary.....                  | 20 |
| TechTipp: SSL Verbindung – setdefault beim Zertifikat.....           | 20 |
| TechTipp: SSL Verbindung mit Connection Manager im Proxy Mode.....   | 20 |
| Versionsinfo: Support für Version 11.70 endet im September 2020..... | 21 |
| Versionsinfo: 12.10.xC13 ist verfügbar.....                          | 21 |
| Versionsinfo: 14.10.xC1 ist verfügbar.....                           | 21 |
| Versionsinfo: 14.10.xC2 ist verfügbar.....                           | 21 |
| Versionsinfo: 14.10.xC3 ist verfügbar.....                           | 21 |
| Versionsinfo: 14.10.xC4 ist verfügbar.....                           | 21 |
| Tipp: Urlaub daheim – Entdecken Sie die Umgebung.....                | 22 |
| Nutzung des INFORMIX Newsletters.....                                | 22 |
| Archiv zum INFORMIX Newsletter.....                                  | 23 |
| Die Autoren dieser Ausgabe.....                                      | 23 |

## Aktuelles

Liebe Leserinnen und Leser des INFORMIX Newsletters,

der INFORMIX Newsletter meldet sich nach langer Zeit wieder zurück. Dies liegt nicht daran, dass es bei INFORMIX nichts Neues gegeben hätte, sondern an vielen einzelnen Ereignissen wie Krankheit, Abteilungswechsel ... und nicht zuletzt GDPR, das verhindert hat, dass die alten Listen zur Verteilung genutzt werden konnten.

Ein weiteres Schreckgespenst namens "Corona" brachte nun den Newsletter zurück, da als Nebeneffekt des notwendigen „HomeOffice“ nun viele Stunden Reisezeit für andere Dinge frei wurden ... nicht zuletzt für die Erstellung dieses Newsletters.

Die INFORMIX USER GROUP als neue Plattform bot sich an, da diese die ideale Heimat für alle Neuigkeiten rund um INFORMIX bieten kann.

Viel Spass mit dem neuen INFORMIX Newsletter !

P.S.: Die erste, selbst gebastelte Version der Maske (siehe Foto) wurde inzwischen (nicht zuletzt auf vielfachen Wunsch meiner Partnerin) durch professionelle Masken ersetzt.



## TechTipp: Connection Manager Options

Der INFORMIX Connection Manager (CM), der mit dem C-SDK kostenlos ausgeliefert wird, bietet die Möglichkeit, Verbindungen zum Datenbank Server auf unterschiedliche Weise zu konfigurieren.

Die Hauptfunktionalitäten hierbei sind:

1. Redirect auf die aktive Instanz einer Replikation
2. Redirect auf die passive Instanz einer Replikation
3. Lastverteilung anhand der Auslastung oder anderer Kriterien
4. Failover Koordination beim Ausfall eines Knotens einer Replikation
5. Proxy Dienst für Clients zum Netzwerk der Datenbank Server

Der Connection Manager erhält seine Informationen über eine Konfigurationsdatei. Hierfür liegt im \$INFORMIXDIR/etc eine Beispieldatei cmsm.cfg.sample in mehreren Varianten für unterschiedliche Clustertypen.

Der Start des Connection Managers erfolgt mittels:

**oncmsm -c <Konfigurationsdatei>**

Eine Rekonfiguration ist möglich mittels:

**oncmsm -r -c <Konfigurationsdatei>**

Gestoppt wird der Connection Manager mittels:

**oncmsm -k <cm\_name>** oder  
**oncmsm -r -k <Konfigurationsdatei>**

Alternativ kann die Umgebungsvariable CMCONFIG gesetzt werden. Ist dies der Fall, so können die Befehle ohne Angabe der Konfigurationsdatei ausgeführt werden.

Der grundlegende Aufbau der Konfigurationsdatei des Connection Managers umfasst einen allgemeinen Teil und dann eine oder mehrere Cluster Definitionen.

Der allgemeine Teil kann folgende Parameter enthalten:

|                 |   |   |
|-----------------|---|---|
| <b>NAME</b>     | Name des Connection Managers                              |   |
| <b>LOGFILE</b>  | Datei in der die Meldungen des Connection Managers landen |   |
| <b>LOG</b>      | Switch für die Protokollierung der Verbindungen           |   |
|                 | Werte:  |   |
|                 | 0 (default)   | # kein Logging  |
|                 | 1   | # Log Proxy und Redirect Information                              |
|                 | 2   | # Log Data Send und Receive im Proxy Mode und<br># CM Information |
|                 | 3   | # Log Data Content im Proxy Mode und CM Information               |
| <b>LOCAL_IP</b> | Liste der lokalen IP-Adressen die der CM überwacht        |   |
| <b>DEBUG</b>    | Debug Informationen                                       |   |
|                 | Werte:  |   |
|                 | 0 (Default)   | # Keine Messages  |
|                 | 1   | # SQL und ESQL Messages werden protokolliert                      |

**CM\_TIMEOUT** Timeout (in Sekunden) für das Warten auf ein weiteres Failover Event

**EVENT\_TIMEOUT** Timeout (in Sekunden) das Warten auf ein Event vom PRIMARY bevor der Failover gestartet wird.

**SECONDARY\_EVENT\_TIMEOUT** Timeout (in Sekunden) für das Warten auf ein Event vom SECONDARY bevor dieser disconnected wird.

**SSL\_LABEL** Label des CM-eigenen Zertifikats bei SSL Verschlüsselung

**SQLHOSTS** Welche SQLHOSTS Datei wird vom CM durchsucht  
Werte:

LOCAL # Die SQLHOSTS des Connection Managers  
REMOTE # Die SQLHOSTS des Datenbank Servers  
LOCAL+REMOTE # Beide

Eine Cluster Konfiguration beginnt jeweils mit dem Wort CLUSTER und schliesst die Definition in geschweifte Klammern ein (Mehrere Einträge sind möglich)

Eine Cluster Konfiguration hat folgenden Aufbau:

```

CLUSTER <Cluster Name>
{
  INFORMIXSERVER <Liste der Informix Server oder Gruppen für diesen Cluster>
  # Service Level Agreements (SLAs)
  SLA <sla_name> DBSERVER=<DBSERVER-Liste oder Funktion> <Optionen>
  SLA ... (beliebig viele SLAs möglich)

  # Fail Over Configuration
  FOC ORDER=ENABLED/DISABLED PRIORITY=<Priorität> TIMEOUT=<time in s>
}

```

Die DBSERVERLISTE innerhalb der SLAs kann entweder die DBSERVERNAMEN, DBSERVERALIASES oder Gruppen aus der SQLHOSTS enthalten. Ebenso sind die Funktionen PRIMARY, SECONDARY, HDR, RSS, SDS möglich, denen alle Informixserver im jeweiligen Status entsprechen.

Als Optionen im SLA sind möglich:

DBSERVERS # ANY, Groupname, Servername, PRIMARY,  
# SECONDARY, RSS, SDS, HDR

MODE # REDIRECT oder PROXY (default REDIRECT)

USEALIASES # ON oder OFF (default ON)

POLICY # ROUNDROBIN, WORKLOAD,  
# SECAPPLYBACKLOG:<pages>

WORKERS # Anzahl Threads (default 4)

Werden bei einem SLA mehrere DBSERVER mit Komma getrennt angegeben, so werden diese in der angegebenen Reihenfolge nach Verfügbarkeit genutzt, so lange die Policy nichts Anderes angibt.

Werden bei einem SLA die DBSERVER in Klammern gruppiert, dann sind alle in der Klammer befindliche Server gleichwertig und der CM versucht diese je nach Last mit Anfragen zu versorgen.

Bei der Fail-Over-Configuration FOC gibt es die Option den automatischen Failover zu aktivieren (ENABLED) oder deaktivieren (DISABLED). Voraussetzung hierfür ist, dass **alle** INFORMIX Server mit der Option DRAUTO 3 konfiguriert wurden.

Die Reihenfolge in welcher der Failover über die Server hinweg ausgeführt wird, hängt allein vom Parameter HA\_FOC\_ORDER in der \$ONCONFIG des PRIMARY Servers ab. Hier können die Werte SDS, HDR und RSS in der gewünschten Reihenfolge eingetragen werden. Zudem ist es derzeit noch möglich hier direkt INFORMIXSERVER anzugeben, die übernehmen sollen. Diese Option wird in einer der kommenden Versionen nicht mehr unterstützt. Ebenso wurde die Option hier in der Konfiguration des CM unter ORDER die Reihenfolge anzugeben bereits deaktiviert.

Der Parameter PRIORITY gibt die Priorität des Connection Managers an bei der Ausführung des Failover. Kleinere Zahlen bedeuten eine höhere Priorität. Werden mehrere Connection Manager genutzt, so müssen sich diese im Namen und der Priorität unterscheiden.

Als weitere Option kann der TIMEOUT angegeben werden, der die zusätzliche Zeit in Sekunden bestimmt, bevor ein Failover gestartet wird, nachdem die Kriterien dafür erfüllt sind.

Im Fall eines Failovers kann ein Shellscript ausgeführt werden. Als Default ist das \$INFORMIXDIR/etc/cmalarmprogram.sh gesetzt, in dem Mails an die Administratoren verschickt werden können. Weitere Aktionen nach einem Failover sind hier vorstellbar wie z.B. der Neustart eines Application Servers.

### **Mehrere Connection Manager:**

Um auch für den Connection Manager eine Ausfallsicherheit zu gewährleisten, können mehrere Connection Manager für die selbe Aufgabe gestartet werden. Diese müssen sich zumindest im Namen und in der Priorität unterscheiden, damit klar ist, welcher Connection Manager den Failover steuern soll. Die Connection Manager sehen sich indirekt über die Einträge am PRIMARY Server in der Datenbank syssha.

Mehrere unterschiedliche Cluster von Servern können entweder in einer Konfiguration zusammengefasst werden und in einem Connection Manager gestartet werden, oder über getrennte Konfigurationen in getrennten Connection Managern laufen. Der Vorteil von getrennten Konfigurationen besteht darin, dass einige Werte wie z.B. LOG nur für alle Cluster in einem Connection Manager gesetzt werden können und nicht je Cluster Eintrag.

## Beispiel einer Konfiguration des Configuration Managers und der Darstellung im „onstat“ und im OpenAdminTool:

```
#####
#
# IBM Corporation
# Licensed Material - Property of IBM Corporation
# Copyright IBM Corporation 2012, 2013
# (c) Copyright HCL Technologies Ltd. 2017. All Rights Reserved.
#
# Example Connection Manager configuration file setting for a
# High-availability cluster.
#
# See also the sqlhosts.sqlhosts.cmsm.sample file.
#####

#####
# Global Configuration Parameters
#####

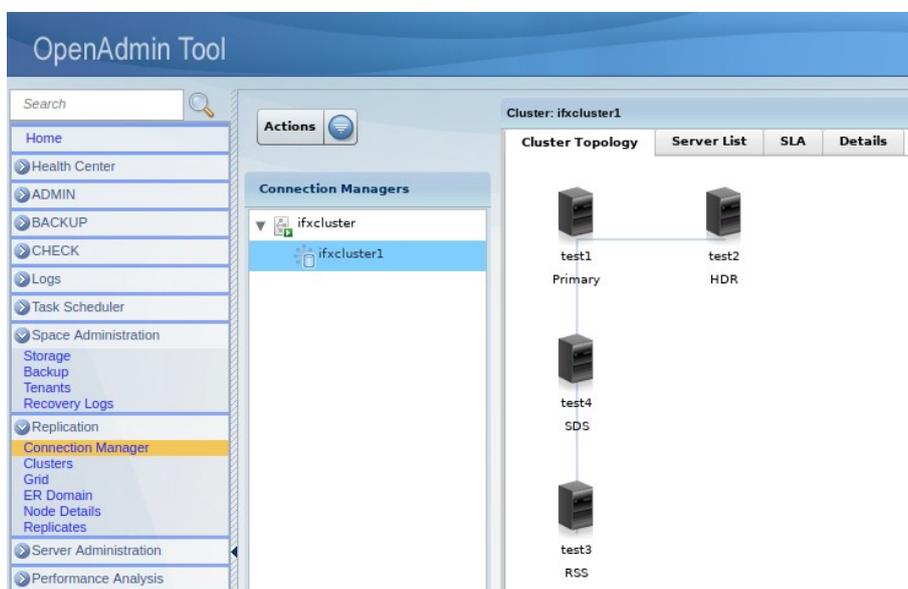
NAME          ifxcluster
LOGFILE       ${INFORMIXDIR}/tmp/cm1.log
LOG           2

CLUSTER      ifxcluster1
{
    INFORMIXSERVER    test1,test2,test3,test4,test5

    SLA x_write DBSERVERS=PRIMARY WORKERS=4 MODE=PROXY USEALIASES=ON
    SLA x_read  DBSERVERS=(HDR,RSS) WORKERS=4 MODE=PROXY USEALIASES=ON
    SLA x_any   DBSERVERS=(PRIMARY,SDS,HDR,RSS) POLICY=ROUNDROBIN WORKERS=4 \
                MODE=PROXY USEALIASES=ON

    FOC ORDER=SDS,HDR,RSS PRIORITY=1 TIMEOUT=5
    CMALARMPROGRAM ${INFORMIXDIR}/etc/cmalarmprogram.sh
}

```



Cluster: ifxcluster1

| Server | Server Type | Server Status | Updates Enabled |
|--------|-------------|---------------|-----------------|
| test1  | PRIMARY     | Active        | ✓               |
| test2  | HDR         | on            | ✓               |
| test3  | RSS         | Active        | ✓               |
| test4  | SDS         | Active        | ✓               |

Cluster: ifxcluster1

Connection Manager: ifxcluster

| SLA     | Servers               | Service port r | Mode  |
|---------|-----------------------|----------------|-------|
| x_write | PRIMARY               |                | PROXY |
| x_read  | (HDR,RSS)             |                | PROXY |
| x_any   | (PRIMARY,SDS,HDR,RSS) |                | PROXY |

Cluster: ifxcluster1

Connection Manager: ifxcluster

Type: CLUSTER

Failover configuration: ORDER=SDS,HDR,RSS PRIORITY=1 TIMEOUT=5

Nach einem Failover steht die Primary Instanz zur Verfügung und alle Verbindungen werden entsprechend ihrer Vorgaben in den SLAs zum entsprechenden Server verbunden.

Bisherige RSS Server verbinden sich, falls sie nicht zum Primary wurden, zum neuen Primary Server, ebenso ggf. SDS Server, falls ein anderer SDS Server die Rolle des Primary übernommen hat.

## TechTipp: ONCONFIG 14.10.xC3 – Unterschiede zur Version 12.10

In der Konfigurationsdatei onconfig.std sind nur relativ wenige Änderungen gegenüber der Version 12.10 zu finden. Einige dieser Änderungen werden in den kommenden Newslettern (bzw. die fett gedruckt in diesem Newsletter) näher beschrieben. Hier vorab schon einmal die Liste der Änderungen mit dem Kommentar, wie er in der onconfig.std zu finden ist:

```
# DIRECT_IO      - Set of flags that may be combined to control "special" I/O
#                 0x0 = Neither direct I/O nor concurrent I/O is used
#                 0x1 = Direct I/O is used if available
#                 0x2 = Concurrent I/O is enabled (on AIX operating systems)
#                 0x4 = Special I/O is enabled for TEMP DBspaces

# BAR_ENCRYPTION - Specifies a cipher algorithm and either a
#                 keyfile or credential store
#                 use to retrieve encryption key to encrypt
#                 both dbspace and log backups
# BAR_DECRYPTION - Specifies a keyfile or credential store
#                 to decrypt backups that were encrypted using
#                 BAR_ENCRYPTION, this parameter is optional.

# STMT_CACHE_QUERY_PLAN - Stores query plan for each SQL statement
#                         Acceptable values are:
#                         0 Disabled
#                         1 Enabled

# SHARD_EDGE_NODE Used to designate local server as shard edge server
#                 where local server can participate in shard query
#                 as a edge server without having to define
#                 Enterprise Replication for local server.
#                 Edge server cannot be used as a coordinator for for shard
#                 query.
#                 0 - Default. Disabled.
#                 1 - Shard edge server.

# RSS_NONBLOCKING_CKPT- Enable non-blocking checkpoint at RS secondary server.
#                 1 - Enable non-blocking checkpoints, continue log replay
#                 while flushing modified buffers to disk during
#                 checkpoint operation.
#                 0 - (Default) Disable non-blocking checkpoints
# SEC_APPLY_POLLTIME - In micro seconds, how long apply thread should
#                 poll for new work before yielding.
#                 Recommended value for smaller systems
#                 (between 1 to 8 CPUVPS): 0
#                 Recommended value for medium systems
#                 (between 8 and 16 CPUVPS): 10
#                 Recommended value for larger systems
#                 (> 16 CPUVPS): 1000
#                 Note: with poll time > 0, recommended to configure poll
#                 threads to run on NET VP. See NETTYPE config parameter.
# SEC_LOGREC_MAXBUFFS - Configure number of 16K buffers to be used
#                 for replaying log records.
#                 If set to zero, default value:
#                 5 * OFF_RECVRY_THREADS.
```

```

#                               Recommended value : 1000 buffers
# SEC_DR_BUFS                   - Configure number of DR buffers to be used
#                               for replicating log records to secondary server.
#                               Buffer size is same as LOGBUFF config value.
#                               Default value: 12

# DUMPCNT                       - An integer representing the maximum number of shared memory
#                               dumps or core files for a single thread, or, for managed
#                               shared memory dumps, the extended format:
#                               thrdlimit=<#>,instlimit=<#>,insttime=<#>,interval=<#>
#                               where all parameters are optional and have these meanings:
#                               - thrdlimit = max number of shared memory dumps per thread
#                               - instlimit = max number of shared memory dumps for the
#                               instance in the time period defined by
#                               insttime. The default value value of 0
#                               implies no limit.
#                               - insttime = time period in seconds for the limit of shared
#                               memory dumps defined by instlimit. The default
#                               is 0 which means no time limit and instlimit
#                               then becomes a maximum count until the instance
#                               is restarted.
#                               - interval = minimum interval in seconds between the
#                               completion of one shared memory dump before
#                               another is permitted. The default is 300 seconds.
#

```

## TechTipp: ONCONFIG – DIRECT\_IO für TempSpaces

Bisher war es nur möglich den Direct-IO für „normale“ DBSpaces zu aktivieren.  
Hierfür standen die Optionen:

- 0 = disabled
- 1 = enabled
- 2 = Sonderfall auf AIX Filesystem

zur Verfügung.

Mit Version 14.10 wurde die Option eingeführt, dass auch Temporäre DBSpaces vom Direct-IO profitieren können. Dies kann mit dem Flag 0x4 erfolgen. Die bisherigen Parameter bleiben unverändert erhalten und die Features sind als additive Hexwerte (sogenannte „bit Flags“) zu interpretieren.

```

# DIRECT_IO   - Set of flags that may be combined to control "special" I/O
#             0x0 = Neither direct I/O nor concurrent I/O is used
#             0x1 = Direct I/O is used if available
#             0x2 = Concurrent I/O is enabled (on AIX operating systems)
#             0x4 = Special I/O is enabled for TEMP Dbspaces

```

So bewirkt z.B. der Wert „**DIRECT\_IO 0x5**“, dass sowohl temporäre, als auch nicht temporäre DBSpaces von diesem Feature der libaio profitieren können.

## TechTipp: .NET Core Compatibility

Mit CSDK 4.50.xC4 ist im C-SDK zusätzlich zum „Informix .NET Provider“ auch der syntaktisch kompatible Treiber „INFORMIX .NET Core“ enthalten, der viele neue Möglichkeiten bietet und nicht nur auf Windows, sondern auch auf Linux zur Verfügung steht. Der Umstieg von .NET Provider auf .NET Core bedingt allerdings eine Compilation unter Einbindung der neuen DDL, die im \$INFORMIXDIR/bin zu finden ist.

## TechTipp: Wire Listener Enhancements ab 14.10.xC4

Der Wire Listener unterstützt nun auch die Mongo API der Versionen 3.2, 3.4, 3.6, 4.0 und 4.2. Es muss der Wert für [mongo.api.version](#) gesetzt werden, um mit neueren Versionen der Mongo API kompatibel zu sein.

Die Unterstützung der Authentication im REST und MQTT Listener wurde verbessert. Nun kann die Authentifizierung über REST zur Informix Datenbank mittels User und Password erfolgen.

## TechTipp: Spatio-Temporal Enhancements ab 14.10.xC4

Informix Spatial Datentypen folgen nun der OpenGIS Simple Features Specification für SQL Revision 1.1 und der ISO/IEC 13249-3 SQL/MM Part 3: Spatial.

Die Informix Spatial Lösung basiert auf ESRI SDE 10.8 Shape und Projection Engine (PE) libraries.

## TechTipp: Huge Pages – Linux Kernel Parameter

Der Informix Server nutzt Speicher, der über die Pages Table auf physischen Speicher gemappt wird. Die Pagesize ist hierbei im Default 4k. Somit ist die Verwaltung grösserer Speicherblöcke sehr kostspielig und erfordert erhebliche Ressourcen.

Mit Linux Kernel 2.6 wurden die sogenannten „Huge Pages“ eingeführt, mit denen der Speicher in Einheiten von 2MB statt 4k verwaltet wird.

Finden sich im „online.log“ Einträge der Art:

**Insufficient free huge pages in /proc/meminfo for shared memory segment.**  
so ist Linux nicht für die Nutzung der Huge Pages konfiguriert.

Dies kann ganz einfach mit folgendem Befehl überprüft werden:

```
$ grep Huge /proc/meminfo
AnonHugePages:          0 kB
ShmemHugePages:        0 kB
HugePages_Total:       0
HugePages_Free:        0
HugePages_Rsvd:        0
HugePages_Surp:        0
Hugepagesize:         2048 kB
```

Es sind im Beispiel also keine Huge Pages verfügbar.  
 Dies kann (als User root) geändert werden mit einem Eintrag in der Datei sysctl.conf:

```
vm.nr_hugepages=306
```

und anschliessendem Aktivieren mittels

```
$ sysctl -p
```

Die Ausgabe sollte danach folgendermassen aussehen:

```
$ grep Huge /proc/meminfo

AnonHugePages:          0 kB
ShmemHugePages:        0 kB
HugePages_Total:       8192
HugePages_Free:        8050
HugePages_Rsvd:        2917
HugePages_Surp:         0
Hugepagesize:          2048 kB
```

Beim Neustart der INFORMIX Instanz sollte dann je Bufferpool im "online.log" folgende Meldung erscheinen:

```
04/24/20 00:13:23 Shared memory segment will use huge pages.
```

**Achtung: Die Huge Pages gelten nur für den Start der Instanz, nicht für das nachträglich hinzufügen von Memory Segmenten.**

## TechTipp: INFORMIX im Docker installieren

Das aktuelle Docker Image der INFORMIX Developer Edition kann aus dem Docker Repository geladen werden:

```
docker pull ibmcom/informix-developer-database:latest
```

Das Image ist nach dem Laden zu sehen mittels:

```
docker images -a
REPOSITORY                                TAG      IMAGE ID      CREATED      SIZE ...
ibmcom/informix-developer-database        latest   596546576504 1 months ago 881MB ...
```

Aus diesem Image kann nun ein Container mit Vorgabewerten erstellt werden:

```
docker run -it --name ifx --privileged --cpus="2" --memory="2000m" -p
9088:9088 -p 9089:9089 -p 27017:27017 -p 27018:27018 -p 27883:27883 -v
/Users/kalu/IFXDATA:/opt/ibm/data -e TYPE=oltp -e LICENSE=accept
ibmcom/informix-developer-database:latest
```

Die Ports, die nicht nur im Docker, sondern auch für Aufrufe vom Server aus verfügbar sein sollen, müssen mittels der Option -p bekannt gemacht werden.

Der Container sieht dann folgendermassen aus:

```
docker container ls

638d0b2963c7  ibmcom/informix-developer-database:latest
"/opt/ibm/dinit /opt..." 1 months ago      Up 30 minutes (healthy)
0.0.0.0:9088-9089->9088-9089/tcp, 0.0.0.0:27017-27018->27017-27018/tcp,
8080/tcp, 0.0.0.0:27883->27883/tcp  ifx
```

Um mit dem Docker Container arbeiten zu können, muss dieser gestartet werden:

```
docker start ifx
docker attach ifx
```

Der Zugriff auf die Informix Instanz mittels BASH wird durchgeführt über:

```
docker exec -i -t ifx /bin/bash
```

## TechTipp: Anzeige von Spatialdaten aus INFORMIX mittels R

Die Programmiersprache R, die für Auswertungen und Statistiken immer beliebter wird, bietet eine komfortable Schnittstelle zu Datenbanken mittels JDBC.

Dazu muss der entsprechende INFORMIX JDBC Treiber bereits installiert sein.

In R muss zudem dass das Package RJDBC installiert werden:

```
install.packages("RJDBC",dep=TRUE)
versuche URL
'https://ftp.fau.de/cran/bin/macosx/el-capitan/contrib/3.6/RJDBC_0.2-8.tgz'
Content type 'application/x-gzip' length 107445 bytes (104 KB)
=====
downloaded 104 KBDie heruntergeladenen Binärpakete sind in
/var/folders/ck/0bmmkm5s3bv2hs_v8xgv84th0000gn/T//Rtmpn4XI2j/downloaded_packages
```

In R ist zusätzlich die Library DBI notwendig, um auf die Datenbank zugreifen zu können:

```
> library(DBI)
>
> library(RJDBC)
Lade nötiges Paket: rJava
```

Nun kann der Treiber für Informix geladen und die Verbindung zur Datenbank definiert werden:

```
> drv <-
JDBC("com.informix.jdbc.IfxDriver", "/Users/kalu/JDBC/Informix_JDBC_Driver/lib/
ifxjdbc.jar")
> conn <- dbConnect(drv,
"jdbc:informix-sqli:127.0.0.1:9088/<database>:INFORMIXSERVER=<ifx_servername>",
"<user>", "<password>")
```

Ab diesem Zeitpunkt sind Abfragen im SQL möglich:

```
> dbGetQuery(conn, "select count(*) from systables")
(count(*))
1 102
```

R-Studio kann auch Spatialdaten aufbereiten. Dazu wird die Funktion leaflet benötigt. Die Library „leaflet“ muss explizit installiert und aktiviert werden. Die im Beispiel verwendeten Daten wurden von den HomePages von IBM D und A in die Datenbank geladen und mit einem Internetdienst um Latitude und Longitude erweitert in einer Tabelle abgelegt.

### Beispiel: IBM Locations anzeigen

```
> install.packages("leaflet")
> library("leaflet")
```

### JDBC Treiber laden:

```
> drv <-
JDBC("com.informix.jdbc.IfxDriver", "/Users/kalu/JDBC/Informix_JDBC_Driver/lib/
ifxjdbc.jar")
```

### Verbindung zur Datenbank herstellen:

```
> conn <- dbConnect(drv,
"jdbc:informix-sqli:127.0.0.1:9088/<database>:INFORMIXSERVER=<ifxserver>",
"<login>", "<passwd>")
```

### Daten aus der Datenbank in die Variable ibm\_locations einlesen:

```
> ibm_locations=dbGetQuery(conn, "
select land||' '||plz||' '||ort||' '||str as wo, latitude, longitude, case when
land='D' then 0 else 1 end flag
from ibm_locations ;
")
```

```
# table ibm_locations (
# land    char(3),
# plz     char(5),
# ort     char(18),
# str     char(28),
# latitude dec(12,6),
# longitude dec(12,6)
# )
```

### Zur Unterscheidung der Länder spezifische Farben für D und Rest im Marker definieren:

```
> getColor <- function(ibm_locations) {
  sapply(ibm_locations$fläg, function(flag) {
    if(flag == 0) {
      'green'
    } else {
      'red'
    } })
}
```

### Marker mit Farbvorgaben erstellen:

```
> icons <- awesomeIcons(
  icon = 'ios-close',
  iconColor = 'black',
  library = 'ion',
  markerColor = getColor(ibm_locations)
```

)

Viele weitere Optionen sind hier möglich wie z.B. Marker zu nutzen, die gleich auf die HomePage der Location verweisen oder einen Anfahrtsplan anbieten.

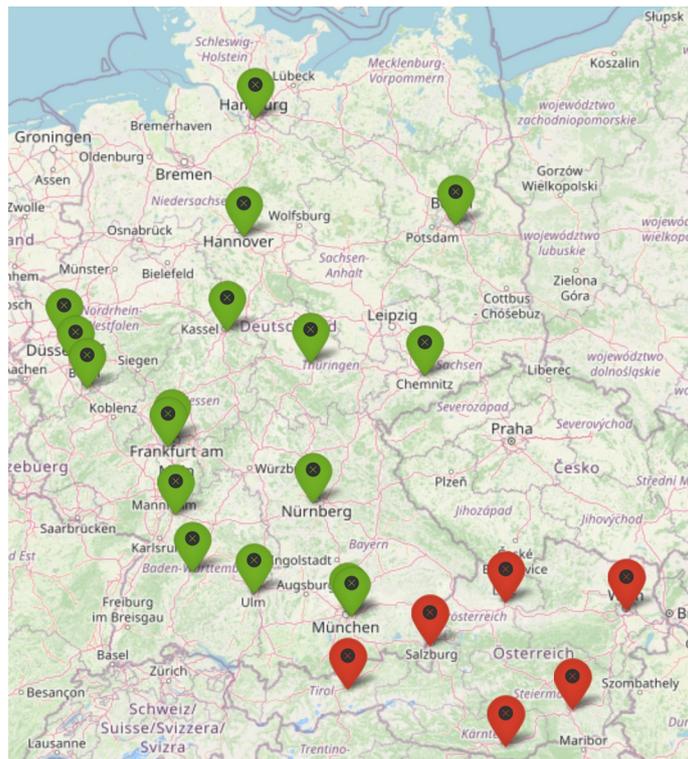
**Karte erstellen:**

```
> map <- leaflet() %>% addTiles() %>%  
addAwesomeMarkers(lng=ibm_locations$longitude, lat=ibm_locations$latitude,  
popup=ibm_locations$wo, icon=icons)
```

**Karte anzeigen:**

```
> map
```

**Ergebnis:**



Insider Anmerkung: Die Niederlassung BREGENZ in Vorarlberg ist leider seit diesem Jahr Geschichte. Eine der schönsten Locations, direkt am Ufer des Bodensees mit Blick auf den Hafen.

## TechTipp: Memory Cleaning bei Linux

Linux versucht sich gegen lang laufende Programme zu schützen, die sehr viel Memory nutzen.

INFORMIX lebt von Memory und dem Umstand, dass es nahezu nie neu gestartet werden muss. Daher sollte folgender Parameter bei Linux angepasst werden:

**To avoid database crashes, set RemoveIPC=no in /etc/systemd/logind.conf and run „systemctl reboot“ to reboot the system.**

## TechTipp: MSG\_DATE

Mit Version 14.10.xC4 wurde die Möglichkeit das Datum im „online.log“ zu setzen erweitert. Im Default steht vor jeder Zeile lediglich die Uhrzeit. Mit Version 9.40 kam (damals noch nicht dokumentiert) die Option 1 hinzu, bei der zusätzlich das Datum im Format MM/DD/YYYY vor die Uhrzeit gesetzt werden konnte. Nun kamen neu die Optionen 2 und 3 hinzu. Beim Wert 2 wird zusätzlich noch die Anzahl Millisekunden seit dem 01.01.1970 vor das Datum und die Uhrzeit gesetzt.

Der Wert 3 setzt das Datum als DATETIME vor die Meldung, wobei das Format von DATETIME leider nicht änderbar ist. Für die meisten Auswertungen ist jedoch das Format YYYY-MM-DD HH:MM:SS.FFF durchaus eine sehr gute Wahl.

### Beispiel:

```
onstat -g cfg full MSG_DATE
```

#### Configuration Parameter Info

| id  | name     | type | maxlen | units | rsvd | tunable |
|-----|----------|------|--------|-------|------|---------|
| 150 | MSG_DATE | INT4 | 12     |       |      | *       |

```
min/max : 0,3
```

```
default : 0
```

```
onconfig: 3
```

```
current : 3
```

#### Description:

Use the MSG\_DATE configuration parameter to specify the format of the timestamp preceding messages in the log. The choices are as follows:

| Value | Format   |
|-------|--|
| 0     | HH:MM:SS   |
| 1     | MM/DD/YYYY HH:MM:SS<br>(Locale-dependent ordering of MM and DD)                            |
| 2     | Milliseconds since epoch + MM/DD/YYYY HH:MM:SS<br>(Locale-dependent ordering of MM and DD) |
| 3     | YYYY-MM-DD HH:MM:SS.FFF  |

```
09:17:15 Value for MSG_DATE (0) was saved in config file.
```

```
09:17:15 Value of MSG_DATE has been changed to 0.
```

```
07/20/2020 09:17:19 Value for MSG_DATE (1) was saved in config
```

```
07/20/2020 09:17:19 Value of MSG_DATE has been changed to 1.
```

```
1595229444878 07/20/2020 09:17:24 Value for MSG_DATE (2) was saved
```

```
1595229444878 07/20/2020 09:17:24 Value of MSG_DATE has been  
changed to 2.
```

```
2020-07-20 09:17:27.873 Value for MSG_DATE (3) was saved in ...
```

```
2020-07-20 09:17:27.873 Value of MSG_DATE has been changed to 3.
```

## TechTipp: onstat -g top

Mit Version 14.10.xC4 wurde der „onstat -g“ um die Option „top“ erweitert, die helfen soll, Engpässe im aktuell laufenden System bezüglich CPU Time, I/O oder Speichernutzung zu identifizieren.

Es kann die Anzahl zu überwachender Elemente (z.B. 10 Sessions mit dem höchsten Wert an Reads), das Interval in Sekunden und die Anzahl der Wiederholungen angegeben werden.

Syntax:

```
>>onstat -g top [<keyword pair>] [<max consumers> [<interval>
[<repeats>]]]
```

Onstat -g top can take an optional pair of keyword arguments:

- **Top Threads**

- onstat -g top thread cpu: Threads currently using the most CPU time.
- onstat -g top thread drd: Threads currently performing the most disk reads.
- onstat -g top thread bfr: Threads currently performing the most buffer reads.
- onstat -g top thread bfw: Threads currently performing the most buffer writes.
- onstat -g top thread plg: Threads currently performing the most physical logging.
- onstat -g top thread llg: Threads currently performing the most logical logging.

- **Top Sessions**

- onstat -g top session cpu: Sessions currently using the most CPU time.
- onstat -g top session drd: Sessions currently performing the most disk reads.
- onstat -g top session bfr: Sessions currently performing the most buffer reads.
- onstat -g top session bfw: : Sessions currently performing the most buffer writes.
- onstat -g top session plg: Sessions currently performing the most physical logging.
- onstat -g top session llg: Sessions currently performing the most logical logging.

- **Top Chunks**
  - `onstat -g top chunk ios`: Chunks currently performing the most reads and writes.
  - `onstat -g top chunk art`: Chunks with the highest average read time.
  - `onstat -g top chunk awt`: Chunks with the highest average write time.
- **Top Spaces**
  - `onstat -g top space ios`: Spaces currently performing the most reads and writes.
  - `onstat -g top space art`: Spaces with the highest average read time.
  - `onstat -g top space awt`: Spaces with the highest average write time.
- **Top Virtual Memory Pools**
  - `onstat -g top mempool gro`: Memory pools currently growing the fastest.
- **Top Session Memory**
  - `onstat -g top sessmem gro`: Sessions currently allocating the most memory.
- **Top Partitions**
  - `onstat -g top partition drd`: Partitions currently performing the most disk reads.
- **Top Tables**
  - `onstat -g top table drd`: Tables currently performing the most disk reads.

Example

Figure 1. `onstat -g top` command examples

**Top 3 spaces for average write times**

```
onstat -g top space awt 3
```

**top 10 tables for disk reads over a 1 minute interval**

```
onstat -g top partition drd 10 60
```

**Top 15 threads for CPU time, sampling every 20 seconds, and repeating 5 times**

```
onstat -g top thread cpu 15 20 5
```

**Memory pools that are growing, updating every 30 seconds, repeating indefinitely**

```
onstat -g top mempool gro 0 30 0
```

**The least active session in terms of buffer reads, repeating 10 times, with 5 seconds between each update**

```
onstat -g top session BFR 1 5 10
```

## TechTipp: UUID() - Function

Immer wieder kam die Frage, ob INFORMIX eine UUID erstellen kann statt einer Serial oder BigSerial Nummer. Dies konnte bisher nur über eine „external function“ z.B. in C bewerkstelligt werden.

Mit Version 14.10.xC2 stellt Informix die Funktion UUID() zur Verfügung, die allerdings erst als Java Funktion in der Datenbank verankert werden muss.

Dies kann z.B. über den folgenden Aufruf erfolgen:

```
create function uuid() returning char(36) external name
'com.informix.judrs.IfXStrings.getUUID()' language java;
grant execute on UUID to public;
```

Danach kann die Funktion in allen SQL Calls genutzt werden.  
Beispiel:

```
select first 5 tabid, uuid() as uuid
from systables
```

```
tabid uuid
```

```
1 1eab479c-129d-4c96-8b82-56b54dbdaf65
2 2ffe6406-86db-409a-a17d-727164e2dde1
3 255a9cda-3756-4a4b-a2f8-b95ab0c609c5
4 fb6a675e-6d1f-42cc-9b2d-4f49fb53724a
5 75a56a78-3dd6-4ea6-abe6-4dbbcf53ff45
```

## TechTipp: INFORMIX Drivers - Webseite

Wer einen Treiber für INFORMIX sucht, der findet diesen mit sehr hoher Wahrscheinlichkeit auf der Webseite: <https://github.com/OpenInformix>

Hier sind aktuell zu finden:

```
node-informixdb
```

```
# Informix native node.js driver for scalable enterprise and IoT solutions.
# JavaScript
```

```
demo-ASPCore-Ecommerce
```

```
# ECommerce web application in ASP .Net Core (MVC) without EF core.
# JavaScript
```

```
# IfxPy
```

```
# Informix native Python driver for scalable enterprise and IoT solutions.
# Python
```

demo-ASPCore-webapp  
# C#

demo-dotnetcore-wpf  
# WPF Demo Application for Dotnet Core  
# C#

IfxNode2  
# ODBC bindings for node  
# C++

SmartCheckIn  
# Smart check-in demo with Informix Machine Learning  
# JavaScript

IfxNode  
# Informix native node.js driver for scalable enterprise and IoT solutions.  
# JavaScript

InformixAlchemy  
# Informix SQLAlchemy adapter  
# Python

informix-grafana  
# Informix TS plug-in for Grafana 5.x  
# JavaScript

ODBCExamples  
# Informix ODBC Application Examples  
# C

erlang-informixdb  
# Erlang steps to connect with the Informix Database

IfxDotNet  
# Informix .NET Core Seed project  
# C#

NodeFullStack  
# A Simple Nodejs REST API starter kit  
# JavaScript

nodesample  
# Informix Node.js Samples  
# JavaScript

PyPredictStoreSales  
# Python predict store sales  
# Jupyter Notebook

```
IfxGo
```

```
# odbc driver written in go
```

```
# Go
```

```
testconn
```

```
# Basic Connectivity Test with Informix Database
```

```
# JavaScript
```

```
IfxR
```

```
# Informix native R driver for Data Science and Machine Learning
```

```
# C
```

```
ODBC_BSON_JSON
```

```
# Informix ODBC sample application by using BSON and JSON data type
```

```
# C
```

```
ODBC-Specification
```

```
# Microsoft ODBC Specification
```

## **TechTipp: Connection Manager – FOC vom Primary**

Die Definition der Reihenfolge des Failover wurde in der Vergangenheit leider mehrfach geändert. Aktuell ist die FOC\_ORDER in der \$ONCONFIG des PRIMARY Servers entscheidend. Aus diesem Grund muss der CONNECTION MANAGER eine Verbindung zum PRIMARY Server aufbauen können, damit dieser dort die FOC\_ORDER aus der Konfiguration auslesen kann. Besteht beim Start keine Verbindung zum PRIMARY Server kann bricht der Start des Connection Managers ab.

## **TechTipp: SSL Verbindung – setdefault beim Zertifikat**

Wird mit dem GSKit die Basis für eine SSL Verbindung definiert, so muss das Zertifikat als Default definiert werden. Die Option „Trusted“ reicht beim „Personal Certificate“ nicht aus. Der Aufruf hierzu muss somit den Parameter „-setdefault“ enthalten:

```
gskit... -cert -setdefault -db ...
```

## **TechTipp: SSL Verbindung mit Connection Manager im Proxy Mode**

Werden SLAs am Connection Manager im Proxy Mode betrieben und sollen diese über SSL kommunizieren, so muss im \$INFORMIXDIR/ssl des Connection Managers eine KeystoreDB existieren, die den Namen des ConnectionManagers trägt.

Dies ist ab Version 12.10.xC13 dringend notwendig, davor konnte eine SSL Verbindung über den Connection Manager auch ohne diesen Eintrag aufgebaut werden.

**Versionsinfo: Support für Version 11.70 endet im September 2020**

Neue Versionen beinhalten nicht nur neue Features, sondern auch Verbesserungen gegenüber den bisherigen Releases. Aus diesem Grund ist es ratsam, die in der Produktion eingesetzten Versionen auf einem aktuellen Stand zu halten. Bei INFORMIX werden ältere Versionen immer noch einige Zeit unterstützt, und das Ende des Supports wird frühzeitig angekündigt.

Die Version 11.70, deren erstes Release im April 2011 als Growth Edition bzw. Ultimate Edition zur Verfügung stand, wird zum 30. September 2020 aus dem Support genommen.

Unsere klare Empfehlung ist, auf die aktuelle Version 14.10 zu migrieren. Die Migration ist in 99% der Fälle InPlace möglich.

**Versionsinfo: 12.10.xC13 ist verfügbar**

Seit Juni 2019 ist die Version 12.10.xC13 für alle unterstützten Plattformen und Editionen verfügbar. Hierin sind einige Patches enthalten, die Fehler beheben und neue Features verbessern.

**Versionsinfo: 14.10.xC1 ist verfügbar**

Seit März 2019 ist die Version 14.10.xC1 für alle unterstützten Plattformen und Editionen verfügbar. Hierin sind viele neue Features und einige Patches enthalten, die Fehler beheben.

**Versionsinfo: 14.10.xC2 ist verfügbar**

Seit September 2019 ist die Version 14.10.xC2 für alle unterstützten Plattformen und Editionen verfügbar. Hierin sind neue Features und Patches enthalten, die Fehler beheben.

**Versionsinfo: 14.10.xC3 ist verfügbar**

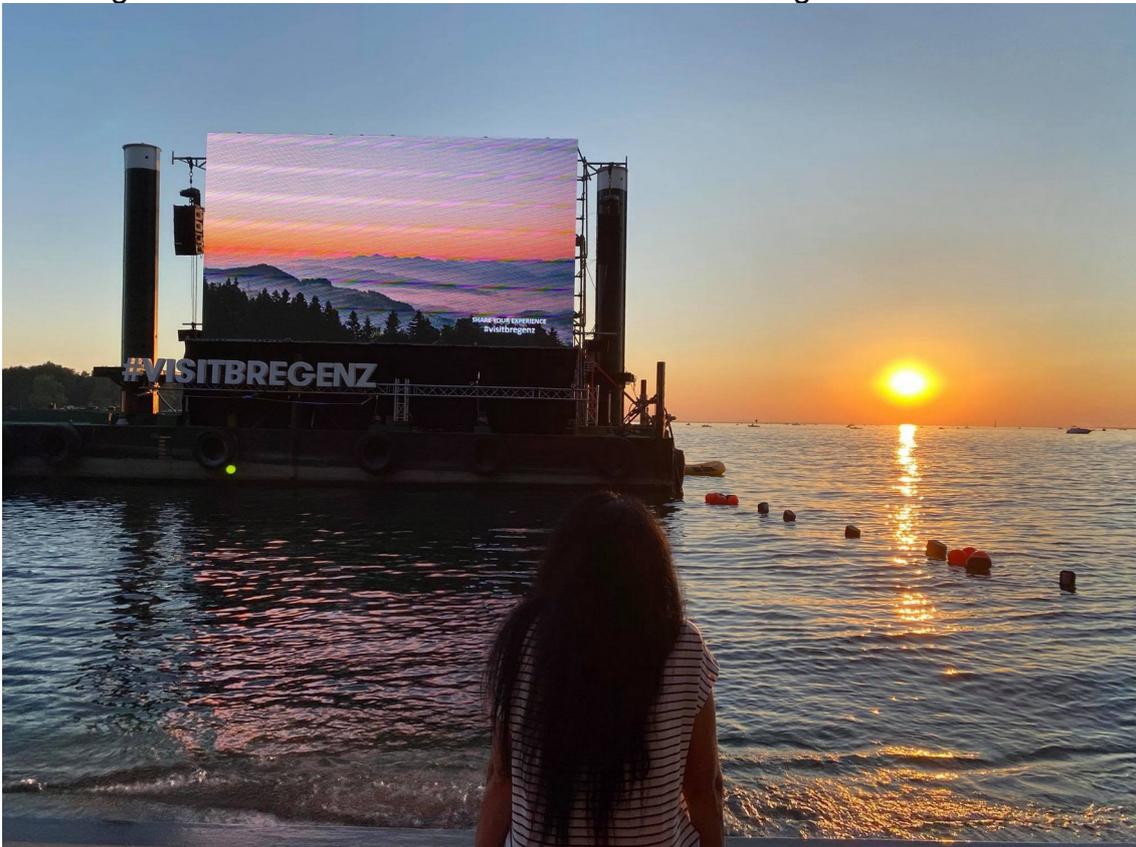
Seit Dezember 2019 ist die Version 14.10.xC3 für alle unterstützten Plattformen und Editionen verfügbar. Hierin sind einige Patches enthalten, die Fehler beheben.

**Versionsinfo: 14.10.xC4 ist verfügbar**

Seit April 2020 ist die Version 14.10.xC4 für alle unterstützten Plattformen und Editionen verfügbar. Hierin sind einige Patches enthalten, die Fehler beheben.

## Tipp: Urlaub daheim – Entdecken Sie die Umgebung

Bei Urlaub denken Viele sofort an Ferienflüge, Strand und Meer. Dabei gibt es in der unmittelbaren Umgebung so viel Schönes zu entdecken. Stöbern Sie einfach mal auf den Tourismus Seiten der Städte in der Umgebung und Sie werden sicher viele Überraschungen finden. Wir wurden keine 5km entfernt fündig ...



## Nutzung des INFORMIX Newsletters

Die hier veröffentlichten Tipps&Tricks erheben weder Anspruch auf Vollständigkeit noch auf Fehlerfreiheit.

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 oder eine Bereitstellung auf der eigenen HomePage ist ausdrücklich erlaubt, so lange die Quelle angegeben wird. Alle hier veröffentlichten Scripts stehen uneingeschränkt zur weiteren Verwendung zur Verfügung.

## Archiv zum INFORMIX Newsletter

Der INFORMIX Newsletter steht allen Lesern ab sofort zum Download unter der alt bekannten Adresse bei der IUG Deutschland bereit:

<https://iug.de/informix-newsletter/>

Dort ist zudem eine Inhaltsangabe über alle Themen der bisherigen Ausgaben zu finden, sowie die entsprechenden Ausgaben des INFORMIX Newsletters selbst.

## Die Autoren dieser Ausgabe

Gerd Kaluzinski                      IT-Specialist Informix, DB2, InfoSphere CDC, DataStage  
   IBM Analytics  
   [gerd.kaluzinski@de.ibm.com](mailto:gerd.kaluzinski@de.ibm.com)                      +49-175-228-1983

Martin Fürderer                      Informix Development  
   HCL Technologies Germany GmbH

Andreas Legner                      HCL | IBM Informix Support  
   HCL Technologies, Products and Platforms

Vielen Dank für die Unterstützung der Informix User Group, ohne die es so schnell keine neuen Ausgaben des Informix Newsletters gegeben hätte.

Sowie unterstützende Teams im Hintergrund.

Fotonachweis: Gerd Kaluzinski

- Die erste selbst gebastelte Maske, um im Geschäft eine Maske kaufen zu können.
- #VISITBREGENZ – Opernkino Bregenz