

Willkommen zum „IBM Informix Newsletter“

Inhaltsverzeichnis

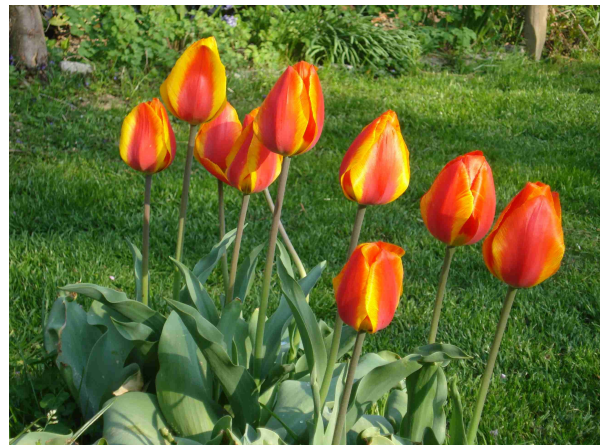
Aktuelles.....	1
TechTipp: Warehouse Accelerator – Check der Voraussetzungen.....	2
TechTIPP: Neu in der ONCONFIG der Version 12.10.xC3.....	3
TechTipp: Zeitreisen leicht gemacht – die LAG-Funktion in TimeSeries.....	4
TechTipp: TimeSeries Abfragen mit unterschiedlicher Granularität.....	5
TechTipp: TimeSeries und Enterprise Replication (CDR).....	7
Versionsinfo: 12.10.xC3 ist verfügbar.....	8
Anmeldung / Abmeldung / Anmerkung.....	9
Die Autoren dieser Ausgabe.....	9

Aktuelles

Liebe Leserinnen und Leser,

mit welcher anderen Datenbank können sie einfach die Zeit nach Bedarf verschieben und behalten trotzdem den Boden unter den Füßen ?

Zeitreisen mittels „LAG-Funktion“ und andere spannende Dinge gehören zum Alltag bei BigData, denn immer mehr Daten aus allen möglichen Zeitzonen und Datenquellen wollen miteinander kombiniert werden. Dass es statt einem Winter ein Frühjahr gab, und dass der Sommer ab März folgt, dafür kann Informix nun nichts, aber wir genießen trotzdem die Sonne nach einem arbeitsreichen Tag.



Wie immer haben wir für Sie eine Reihe an Tipps und Tricks zusammengestellt. Viel Spaß mit den Tipps der aktuellen Ausgabe.

Ihr TechTeam

TechTipp: Warehouse Accelerator – Check der Voraussetzungen

Das neue Hilfsprogramm **ondwachk** wird ab Version 12.10.xC1 mit dem Informix Server installiert. Beim Aufruf in der Umgebung des Informix Server prüft es einfach, ob die Konfiguration des Informix Server die Voraussetzungen für die Nutzung des Informix Warehouse Accelerator (IWA) erfüllt.

Die Ausgabe von "ondwachk" berichtet, welche Prüfungen erfolgreich waren, welche aufgrund der aktuellen Konfiguration übersprungen wurden, und welche Prüfungen nicht erfolgreich waren. Desweiteren kann eine Warnung enthalten sein, dass eine bestimmte Prüfung nicht ausgeführt werden konnte und wie sie manuell erfolgen kann.

Beispielausgabe von "ondwachk":

```
WARNING: The check for compatible versions is not done.
The Informix server is version 12.10.FC2.
The accelerator server version is unknown.
Check the version for the acclereator server by running the command "ondwa -V"
in its respective environment.
PASSED: An sbspace is configured for the Informix server.
PASSED: The SBSPACENAME parameter is set to the name of the default sbspace in
the ONCONFIG file.
PASSED: A network connection type of sockets with TCP/IP protocol is defined in
the sqlhosts file.
SKIPPED: This is a standard server and the Scheduler is running. The check for
a dwavp virtual processor is skipped.
SKIPPED: This is a standard server. The checks for a secondary server
configuration are skipped.
PASSED: The user informix has write access to the sqlhosts file and the
directory that the file is in.
PASSED: An entry for an accelerator is found in the sqlhosts file.
```

TechTIPP: Neu in der ONCONFIG der Version 12.10.xC3

Die aktuell verfügbare Version 12.10.xC3 von Informix beinhaltet neben einer Vielzahl an Verbesserungen auch kleine Änderungen in die Konfigurationsdatei onconfig.std.

Alle vier neuen Parameter können auch dynamisch mittels „onmode -wf/-wm“ gesetzt werden.

Die folgenden Parameter kamen neu hinzu:

CDR_AUTO_DISCOVER

- 0 disabled
- 1 enabled

Steht der Parameter CDR_AUTO_DISCOVER auf 1, so kann mittels des Befehls „cdr autoconfig server ...“ die Connectivity für die Einrichtung der Replikation automatisch erstellt werden. Details dazu und ein praktisches Beispiel finden sie im Informix Newsletter April 2014. Der Parameter ist zudem Voraussetzung für die Nutzung der Option „--autoconf“ der Funktion ifxclone.

AUTOLOCATE

- 0 disabled (default)
- 1 – 32 enabled

Ist der Parameter gesetzt, so erfolgt eine automatische Speicherung von Datenbankobjekten in den „optimalen“ DBSpaces. Zusätzlich werden Tabellen zunächst ohne initiales Extent angelegt (es sei denn EXTENT SIZE wurde spezifiziert) und mittels „Round-Robin“ fragmentiert, die Anzahl solcher Fragmente wird durch den Parameter festgelegt. Der Parameter ist auch als Session Environment Variable über das SET ENVIRONMENT Kommando verfügbar. Beispiele hierzu finden sie in der Ausgabe April 2014 des Informix Newsletters.

INFORMIXCONTIME <sec> (Default 60)

Gibt an wie viele Sekunden ein Verbindungsaufbau zu einem anderen Server wartet, bevor eine Fehlermeldung zurückgegeben wird.

INFORMIXCONRETRY <Anzahl> (Default 1)

Anzahl zusätzlicher Verbindungsversuche, die innerhalb des Timeout (INFORMIXCONTIME) gemacht werden sollen.

Beide Parameter entsprechen den seit langer Zeit bekannten Client Umgebungsvariablen gleichen Namens. Hiermit wird diese Konfiguration nun auch für (I-Star) Server-Server-Verbindungen verfügbar.

TechTipp: Zeitreisen leicht gemacht – die LAG-Funktion in TimeSeries

Eine der Herausforderungen bei der Erfassung von Messwerten unterschiedlicher Länder sind die Zeitzonen. Sinnvollerweise werden intern die Daten einheitlich als UTC abgelegt. UTC kennt weder doppelte Zeiten, noch Zeitsprünge bei Zeitumstellungen.

Würde man zeitbasierte Daten unterschiedlicher Regionen direkt miteinander vergleichen, so würden z.B. die Ladenöffnungszeiten in Deutschland beim Vergleich mit Daten aus Kalifornien dort in die sehr frühen Morgenstunden fallen (wobei ich aus Erfahrung sagen kann, dass dort auch dann einige Geschäfte offen sind).

Der Vergleich macht jedoch mehr Sinn, wenn man jeweils die lokalen Zeiten miteinander vergleicht. Ein Ansatz wäre, die unterschiedlichen Zeiträume umgerechnet auf UTC abzufragen, und diese dann miteinander zu vergleichen. Allerdings stimmen dann z.B. die Tagesgrenzen nicht mehr mit der Realität überein, und eine Summe über einen Tag (group by) würde falsche Ergebnisse liefern.

Um weiterhin die internen Funktionen der TimeSeries wie SUM(), AVG(), ... auch auf Tagesbasis (Woche/Monat,...) verwenden zu können, macht es daher viel mehr Sinn die Funktion LAG() zu verwenden, die die Basis für eine Zeitreihe verschiebt, so dass die Daten bei der Berechnung wieder als „local time“ zur Verfügung stehen.

Beispielausgabe der Stundenwerte in UTC-Zeit und mit Verschiebung um -9 Stunden:

```
SELECT
  t.timestamp::DATETIME YEAR TO MINUTE AS tstamp,
  t.value1 AS ts1val,
  t.value2 AS ts2val
FROM TABLE (
  TRANSPOSE ((
    SELECT UNION(
      '2014-01-01 00:00:00.00000'::DATETIME YEAR TO FRACTION(5),
      '2014-01-01 20:00:00.00000'::DATETIME YEAR TO FRACTION(5),
      AggregateBy('SUM($value)', 'ts_1hour',
        t1.values, 0)::TimeSeries(kalu_data),
      AggregateBy('SUM($value)', 'ts_1hour',
        LAG(t2.values, -9), 0)::TimeSeries(kalu_data)
      )::TimeSeries(kalu_data2)
    FROM kalu_orig_ts t1, kalu_orig_ts t2
    WHERE t1.id = t2.id
  )) AS RESULT (t);
```

Das Ergebnis:

tstamp	ts1val	ts2val
2014-01-01 03:00	1.30	
2014-01-01 04:00	1.40	
2014-01-01 05:00	1.50	
2014-01-01 06:00	31.60	
2014-01-01 12:00		1.30
2014-01-01 13:00		1.40
2014-01-01 14:00		1.50
2014-01-01 15:00		31.60

TechTipp: TimeSeries Abfragen mit unterschiedlicher Granularität

Eine der besonders nützlichen Eigenschaft von TimeSeries ist, dass mittels einfachem Tausch des Kalenders die „Granularität“ der Summierung geändert werden kann.

Die ausserordentliche Performance durch die optimierte Speicherung und die Nutzung interner Funktionen der TimeSeries soll das folgende Beispiel zeigen:

Versuchsaufbau: Daten, gespeichert als Datentyp TimeSeries

- Werte von Messstellen mit Granularität 15 Minuten
- 35040 Messwerte je Messstelle und Jahr (365 * 24 * 4)
- TimeSeries Tabelle „meters_ts“, TimeSeries Spalte „values“ mit Wert in „value“
- Plattform im Test: Linux auf ThinkPad mit Informix 12.10.FC3

Die Summe über den Jahresverbrauch kann abgefragt werden mittels:

```
SELECT t.timestamp, t.value
FROM TABLE (TRANPOSE ((
    SELECT AggregateBy(
        'SUM($value)',
        'ts_1year',
        values, 0,
        '2013-01-01 00:00'::datetime year to minute,
        '2013-12-31 23:45'::datetime year to minute)::TimeSeries(meterdata)
    FROM meters_ts
    WHERE meter_id = '42'))
)
AS TAB (t);
```

Das Ergebnis für die Jahressumme im Aufruf mit Zeitmessung (time dbaccess ...) hat die Form:

```
timestamp                value
2013-01-01 00:00:00.00000 78142.13

1 row(s) retrieved.
real 0m0.052s
user 0m0.008s
sys 0m0.004s
```

Tauscht man im Aufruf lediglich den Kalender 'ts_1year' durch den Kalender 'ts_1month' aus, so erhält man die Monatswerte:

```
...
12 row(s) retrieved.
real 0m0.052s
user 0m0.008s
sys 0m0.000s
```

Der Aufruf mit dem Tageskalender '**ts_1day**' liefert die Tagessummen:

```
...  
365 row(s) retrieved.  
real 0m0.060s  
user 0m0.000s  
sys 0m0.008s
```

Sollen schliesslich die Summen je Stunde ausgegeben werden, so wird der Kalender durch '**ts_1hour**' ersetzt:

```
...  
8760 row(s) retrieved.  
real 0m0.195s  
user 0m0.028s  
sys 0m0.024s
```

An den Antwortzeiten ist zu sehen, dass diese zum grossen Teil durch die Ausgabezeiten (im Testfall nach /dev/null) bedingt sind.

Einige weitere Kalender stehen als Default zur Verfügung.

Sie können jedoch auch eigene Kalender selbst definieren wie z.B. Ladenöffnungszeiten, Quartal, Kernarbeitszeit, Urlaubskalender,....

Die Erstellung von Kalendern und weitere nützliche Tipps und Tricks zur Verwendung der Kalender erfahren sie in der Ausgabe April 2014 des Informix Newsletters.

TechTipp: TimeSeries und Enterprise Replication (CDR)

Tabellen, die den Datentyp **TimeSeries** enthalten, können ab Version 12.10 an der Enterprise Replikation (CDR) teilnehmen. Vor dem Einrichten der Replikate müssen jedoch alle Server, die an der Replikation von Spalten vom Typ **TimeSeries** teilnehmen, zusätzlich vorbereitet werden.

Voraussetzung ist, dass alle Informix Instanzen, die an der Replikation teilnehmen, bereits auf Version 12.10.xC1 oder höher betrieben werden.

Folgende Schritte sind dabei vorzunehmen:

1. In der ONCONFIG der Informix Server muss der Parameter **CDR_TSINSTANCEID** für alle an der Replikation teilnehmenden Instanzen auf unterschiedliche IDs gesetzt sein (der Parameter kann mittels „onmode -wf“ dynamisch gesetzt werden). Es können nur TimeSeries repliziert werden, die **nach** Setzen dieses Parameters erstellt wurden (TSCreate/TSCreateIrr).
2. Die Container der TimeSeries müssen in allen Instanzen die selben Namen besitzen. TimeSeries, die in automatisch erstellten Containern (z.B. über Rolling Window Funktion) gespeichert werden, können nicht repliziert werden. Nach dem Aufsetzen der Replikation müssen weitere Container auf allen Servern mit den selben Namen erstellt werden (dies ist mittels der Grid Funktionalität möglich), bevor darauf Replikate erstellt werden können.
3. Auf allen an der Replikation beteiligten Instanzen müssen die selben Kalender vorhanden bzw. erstellt sein.
4. Die Tabellen, in denen die TimeSeries gespeichert sind, müssen auf allen Servern vor dem Erstellen des Replikats erstellt werden. Die Funktion „autocreate“ oder die Verwendung von „Templates“ ist nicht möglich. Eine TimeSeries, die an der Replikation beteiligt ist, darf keine weiteren Daten vom Typ TimeSeries enthalten.
5. Bei der Erstellung der TimeSeries muss der Container bereits angegeben werden.

Für die Replikation von Tabellen mit Spalten vom Typ **TimeSeries** müssen folgende Voraussetzungen erfüllt sein:

- Das Replikat muss als „Master Replicate“ erstellt werden (Default in V12.10).
- Die „SELECT“-Definition der zu replizierenden Spalten kann nicht eingeschränkt werden. Es müssen alle Spalten repliziert werden.
- Die „WHERE“-Bedingung bei der Erstellung des Replikats darf keinen Datentyp **TimeSeries** enthalten.
- Es können keine „Read-Only“ Teilnehmer der Replikation definiert werden.
- Als Konfliktbehandlung muss „Always-Apply“ gewählt werden.
- Der Replikationskey kann keinen „opaque“ Datentyp enthalten.
- Die Konvertierung von oder nach „UTF-8 (Unicode)“ kann nicht genutzt werden, wenn Replikate zwischen Datenbanken mit unterschiedlichen Definitionen der DB_LOCALE definiert werden.

- Das „Autocreate“ für Tabellen kann bei der Definition der Replikate nicht bei Tabellen genutzt werden, die Spalten vom Typ **TimeSeries** enthalten. Die Tabellen müssen vor der Definition des Replikates auf allen Instanzen erstellt werden.
- Fehler, bzw. Warnungsausgaben in ATS oder RIS Dateien können nicht im XML Format erstellt werden. ATS und RIS Dateien werden im Textformat erstellt.
- Tabellen mit Spalten vom Type **TimeSeries** unterstützen derzeit keine „Shared Query“ über mehrere Server. Die Abfrage von „Grid Queries“ auf den „Virtuellen Tabellen zu den Tabellen der TimeSeries“ ist unterstützt.

Die folgenden Befehle sind auf Replikaten mit Spalten vom Typ **TimeSeries** nicht unterstützt:

- cdr alter
- cdr remaster
- cdr start sec2er
- cdr swap shadow

Der Aufruf „cdr repair“ bei **TimeSeries** unterstützt derzeit nicht die folgenden Optionen:

- --deletewins (bei Konflikten in Löschvorgängen)
- --extratargetrows=merge
- --since
- --timestamp
- --where

Versionsinfo: 12.10.xC3 ist verfügbar

Seit einigen Tagen ist die Version 12.10.xC3 für alle unterstützten Plattformen und Editionen verfügbar. Da es in jeder Version eine Reihe an Verbesserungen gibt, sollte immer eine der neueren Versionen eingesetzt werden.

Anmeldung / Abmeldung / Anmerkung

Der Newsletter wird ausschließlich an angemeldete Adressen verschickt. Die Anmeldung erfolgt, indem Sie eine Email mit dem Betreff „**ANMELDUNG**“ an ifmxnews@de.ibm.com senden.

Im Falle einer Abmeldung senden Sie „**ABMELDUNG**“ an diese Adresse.

Das Archiv der bisherigen Ausgaben finden Sie zum Beispiel unter:

<http://www.iiug.org/intl/deu>

http://www.iug.de/index.php?option=com_content&task=view&id=95&Itemid=149

<http://www.informix-zone.com/informix-german-newsletter>

<http://www.drap.de/link/informix>

<http://www.nsi.de/informix/newsletter>

<http://www.cursor-distribution.de/index.php/aktuelles/informix-newsletter>

<http://www.listec.de/Newsletter/IBM-Informix-Newsletter/View-category.html>

<http://www.bereos.eu/software/informix/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.

Die Autoren dieser Ausgabe

Gerd Kaluzinski IT-Specialist Informix Dynamic Server und DB2 UDB
IBM Software Group, Information Management
gerd.kaluzinski@de.ibm.com +49-175-228-1983

Martin Fuerderer IBM Informix Entwicklung, München
IBM Software Group, Information Management
martinfu@de.ibm.com

Alexander Körner Channel Technical Sales, IMT Germany -
Informix/SOA Integration and Informix IDS
Certified Senior IT-Specialist
IBM Certified Solutions Expert IDS, DB2, Rational & XML
akoerner@de.ibm.com

Die Versionsinfo stammt aus dem Versions-Newsletter der CURSOR Software AG
<http://www.cursor-distribution.de/download/informix-vinfo>

Sowie unterstützende Teams im Hintergrund

Fotonachweis: Gerd Kaluzinski

(Redaktionsgarten)