

---

# Tenant-Datenbanken

## Ein neues Feature in der Praxis

**Sprecher: Dr. Peter Ijewski**  
**[peter@ijewski.de](mailto:peter@ijewski.de)**



## Agenda

---

- Vorstellung
- Tenant-Datenbank – Was ist das?
- Tenant-Datenbank – Wie geht das?
- Tenant-Datenbanken in der Praxis
  - Motivation
  - Migration
  - Neueinrichtung
  - Monitoring
  - Ab 12.10 xC6: Tenant-Restore
- Zusammenfassung
- Fragen



## Vorstellung

---

- *i&P* Dr. Peter Ijewski & Partner ≙ EDV-Systeme & IT-Beratung
- INFORMIX seit 1989
- ERP-System für Bauhaupt- und -nebengewerbe
- Unternehmensberatung
- IT-Consulting (u. a. für den VDEK)
- INFORMIX, DB2, ORACLE, MS-SQL, MySql, PostgreSQL, u. a.
- Diverse Lehraufträge an der DHBW



## Vorstellung

---

- Der Verband der Ersatzkassen e. V. (vdek) mit Sitz in Berlin ist Interessenvertretung und Dienstleister aller Ersatzkassen.
- Verarbeitung von Sozialdaten
- Datenbanksysteme INFORMIX, ORACLE ✦, DB2 ▲, Netezza
- 40 Entwickler



# Tenant-Datenbank – Was ist das?

---

On-Premises

Cloud Computing

Einzelne Kunden

Viele Kunden

Gemeinsame Nutzung

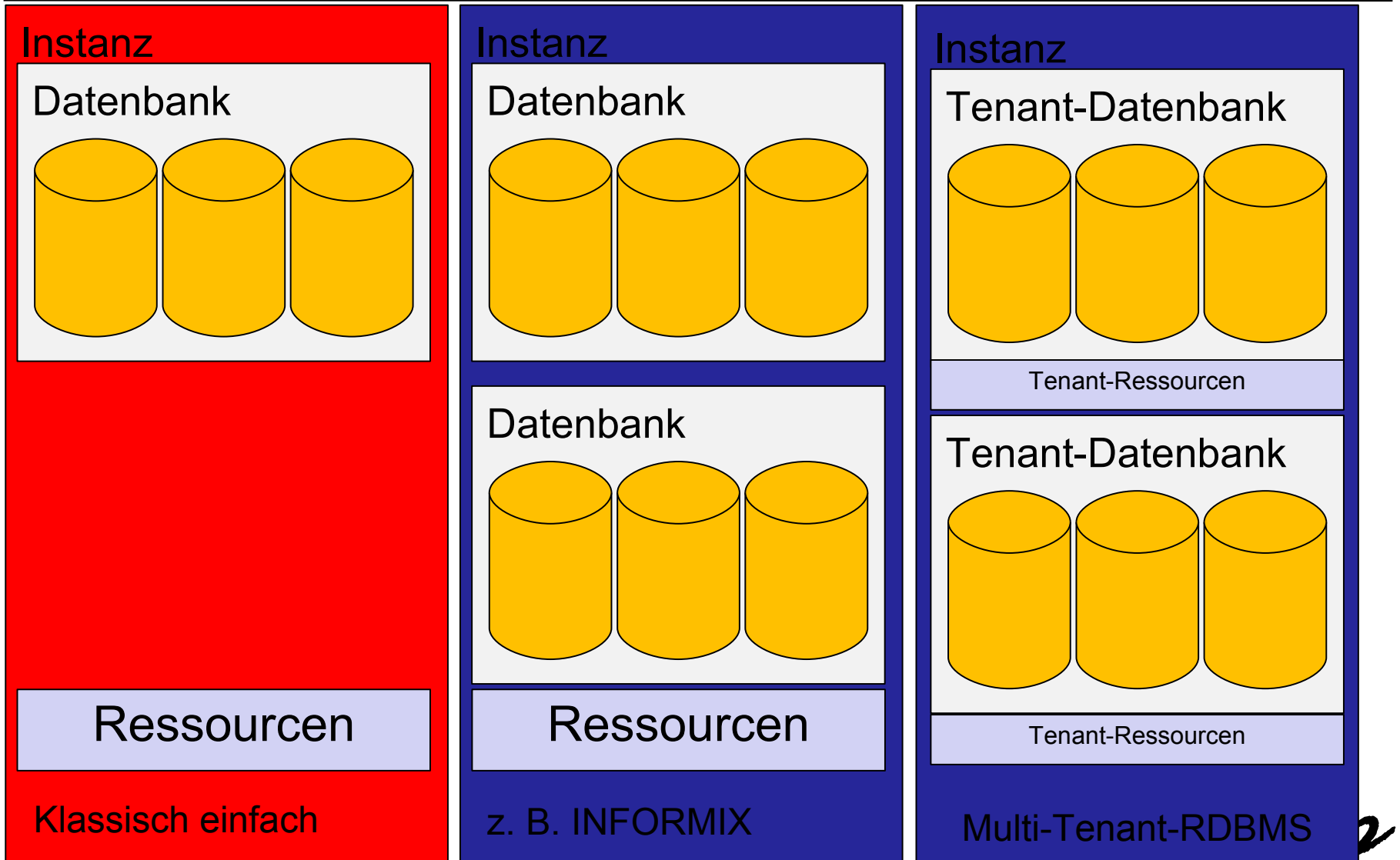
Isolierte Nutzung



Multitenant-Data-Architecture



# Tenant-Datenbank – Was ist das?

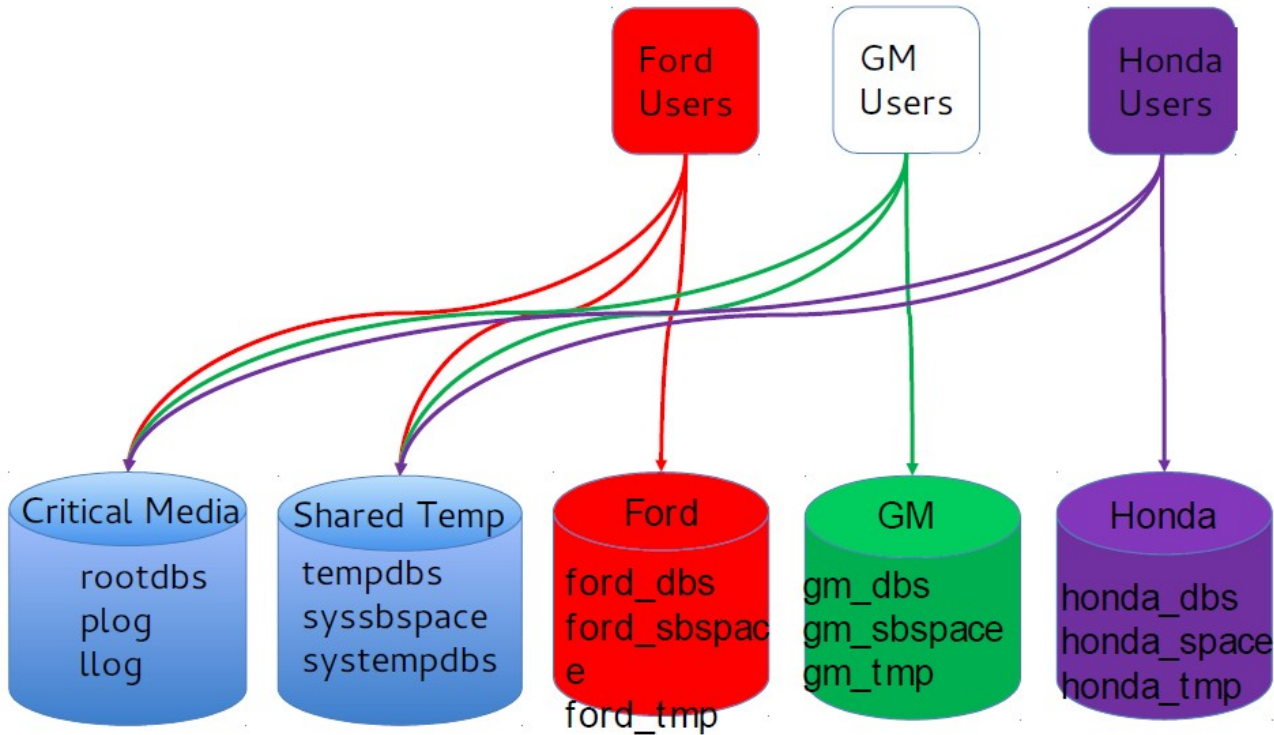


## Tenant-Datenbank – Was ist das?

---

- Isolierte Datenbank in einer Instanz
- Isolierte Tablespaces
- Isolierte oder geteilte Ressourcen (CPU, Netzwerk)
- Kontingentierung von Shared Memory
- Kontingentierung von Locks, Transaktionen
- d. h.: Definierte Zuweisung der Ressourcen an einen „Untermieter“.

# Tenant-Datenbank – Was ist das?





# Tenant-Datenbank – Wie geht das?

## Klassische Syntax zum Anlegen einer Datenbank

```
>>-CREATE DATABASE--+-----+--database--+-----+-->
               '-IF NOT EXISTS-'               '-IN--dbspace-'

                               .-NLSCASE SENSITIVE-----.
>+-----+-----+-----+-----+-----+-----+-----+-----+-----+
'-WITH--+--+-----+--LOG--+-'   '-NLSCASE INSENSITIVE-'
   | '-BUFFERED-' |
   '-LOG MODE ANSI-----'
```

## „Neue“ Syntax zum Anlegen einer Tenant-Datenbank

```
EXECUTE FUNCTION task('tenant create', 'database_name',
                      '{dbspace: "datendbs, indexdbs",
                        sbspace: "blobdbs",
                        logmode: "unbuffered"
                        case: "sensitive",
                        locale: "en_US.819",

                        session_limit_locks: "10000",
                        session_limit_logspace: "5 GB",
                        ...
                      }',
                      );
```

## Tenant-Datenbank – Wie geht das?

---

### Syntax zum Ändern einer Tenant-Datenbank

```
EXECUTE FUNCTION task('tenant update', 'database_name',  
                        '{session_limit_txn_time:"300",  
                        ... }'  
);
```

### Syntax zum Entfernen einer Tenant-Datenbank

```
EXECUTE FUNCTION task('tenant drop', 'database_name');
```

## Tenant-Datenbanken in der Praxis - Motivation

---

- Multitenancy – ein Feature für Cloud Computing?
  
- Multitenancy – ein Feature inspiriert durch Cloud Computing!
  
- Multitenancy – sinnvoll im on-premises-Einsatz!
  
- Multitenancy – Kontingentierung von Ressourcen!
  
- Aufgabenstellung
  - 40 Entwickler
  - 50+ Entwicklerdatenbanken
  - limitiere Ressourcen, (z. B. 8 GB in der Developer-Lizenz)

## Tenant-Datenbanken in der Praxis - Motivation

---

- Weitere Aufgabenstellungen:

### **Enge Vorgaben nach SGB zur Datentrennung**

Bisher : Isolation durch unterschiedliche Server/Instanzen  
und ggf. DB-Vendor

Zukünftig: Isolation durch unterschiedliche Tenants

### **Proaktiver Schutz der prod. Instanzen gegen Fehlbedienungen, etc.**

Bisher : ggf. Probleme durch lange Transaktionen u. a.

Zukünftig: Bessere Einflussnahme durch Kontingentierung

### **Definierter Restore einer einzelnen Datenbank**

Bisher : aufwändig, ggf. "archecker"

Zukünftig: Komfortable, simple Lösung



## Tenant-Datenbanken in der Praxis - Migration

---

- Typischer Vorgang:

```
$dbexport database_name -nw -ss -o outdir
```

Ggf. Detail-Anpassungen in *outdir/database\_name.sql* (Tablespaces, etc.)

```
$dbimport tenant_database -d datendbs -i indir
```



## Tenant-Datenbanken in der Praxis - Migration

---

- Undokumentierte Option:

```
$dbimport -exists tenant_database -d datendbs -l -i indir
```



## Tenant-Datenbanken in der Praxis - Migration

---

- Ablauf:
  - Analyse der neuen Möglichkeiten
  - Script schreiben und testen
  - Datenerhebung
  - Durchführung
  
- Ergebnis:
  - Einfache schnelle Durchführung
  - "geräuschlos", elegant

## Tenant-Datenbanken in der Praxis - Neueinrichtung

---

### ■ Script (Prinzipdarstellung, Auszug)

```
cd /dbSPACE
touch $1_datendbs.000
touch $1_indexdbs.000
# Rechtevergabe
chmod 660 $1_datendbs.000
chmod 660 $1_indexdbs.000
chown informix:informix $1_datendbs.000
chown informix:informix $1_indexdbs.000
echo "$(date) Anlegen von $1_datendbs mit einer Größe von $2 MB"
sizeDBS=`expr $2 \\* 1024`
onspaces -c -d $1_datendbs -p /dbSPACE/$1_datendbs.000 -o 0 -s $sizeDBS
echo "$(date) Anlegen von $1_indexdbs mit einer Größe von $3 MB"
sizeIDX=`expr $3 \\* 1024`
onspaces -c -d $1_indexdbs -p /dbSPACE/$1_indexdbs.000 -o 0 -s $sizeIDX
...
#
$INFORMIXDIR/bin/dbaccess sysadmin <<EOSQL
EXECUTE FUNCTION task('tenant create', '$1',
    '{dbSPACE:" $1_datendbs, $1_indexdbs",
    logmode:"unbuffered"}');
EOSQL
```

\$1 Tenant, \$2 Größe des datendbs, \$3 Größe des indexdbs





# Tenant-Datenbanken in der Praxis - Neueinrichtung

---

## ■ Durchführung

```
$create_tenant test_ije 100 50
.
.
.
$INFORMIXDIR/bin/dbaccess test_ije <<EOSQL

grant ... ;

EOSQL
```

## Tenant-Datenbanken in der Praxis - Monitoring

The screenshot shows the OpenAdmin Tool interface for monitoring Informix databases. The server is identified as 'informix:dev\_soc@vdek-aixdevdbi'. The interface is divided into several sections:

- Informationen:** Shows details for the 'test\_mau' database, including 1 Active Session, 0.000 B Session Memory, and 0 Session Limits Locks.
- Tabellen und Indizes:** A table showing 136 tables and 271 indexes for the 'test\_mau' database, with 0 compressed indexes.
- Tenant VP:** A table for monitoring tenant virtual processors, currently empty.
- Speicherbereiche:** A table showing the 'test\_mau\_dbs' storage area with a size of 40 MB and a status of 'belegt' (indicated by a green dot).
- Chunks:** A table showing a single chunk for 'test\_mau\_dbs' with a size of 40 MB and a path of '...ace/test\_mau\_dbs.000'.

A left-hand navigation menu lists various system management tasks such as Alerts, Protokolle, Task-Scheduler, and Bereichsverwaltung. The 'Tenants' section is currently selected.

## Tenant-Datenbanken in der Praxis - Monitoring

```
$INFORMIXDIR/bin/dbaccess sysadmin <<EOSQL
  select tenant_dbsname,
         tenant_create_time,
         tenant_last_updated,
         tenant_resources::json as info
  from tenant
  order by tenant_dbsname;
EOSQL
```

```
tenant_dbsname      test_ije
tenant_create_time  2016-06-06 12:58:47
tenant_last_updat+  2016-06-06 12:58:47
info                {"dbspace":" test_ije_datendbs, test_ije_indexdbs","session
                  _limit_locks":"1000","logmode":"unbuffered"}
```

# Tenant-Datenbanken in der Praxis - Monitoring

```
$onstat -g cac all
.
.
.
      ID 9      Ordval 70      TxID 0      Flags Active Hits 7      ucount 0
Last access 06/06/2016.13:11
database      : test_ije 4600002
dbspace      : test_ije_datendbs,test_ije_indexdbs,
session_limit_locks : 1000
memory_limit : 0.00 MB
used_memory  : 0.00 MB
connection_limit : 0
used_connections : 0
.
.
.
```

## Tenant-Datenbanken in der Praxis - Monitoring

```
$onstat -g ses 9587
```

```
IBM Informix Dynamic Server Version 12.10.FC6DE -- On-Line -- Up 6 days 03:02:27 -- 1048576 Kbytes
```

session id	user	effective user	tty	pid	hostname	#RSAM threads	total memory	used memory	dynamic explain
9587	ije	-	1	10223858	vdek-aix	1	131072	121576	off

```
Program :
```

```
/informix/server/bin/dbaccess
```

```
.  
. .  
. .  
. .  
. .  
. .
```

```
Session Limits
```

	Limit	Current
Locks	1000	1

## Tenant-Datenbanken in der Praxis - Restore

---

```
$onbar -r -T test_ije -t "2016-06-06 14:30:00" -o  
  
13:28:06 Level 0 Archive started on test_ije_dbs  
13:28:07 Archive on test_ije_dbs Completed.  
13:28:08 Logical Log 4698 Complete, timestamp: 0x1d0e90d2.  
13:28:08 Tenant PIT restore process finished (test_ije: rc=0, tenant_state=ok).  
13:28:08 Update of tenant_state for tenant database (test_ije) succeeded.
```

Siehe auch Informix-Newsletter Januar 2016

Gerd Kaluzinski: „Restore einer einzelnen Datenbank aus einem binären Backup“

## Zusammenfassung

---

- Migration bestehender Datenbanken
- Neueinrichtung weiterer Tenants
- Isolation
- Restore einzelner (Tenant-) Datenbanken
- Akzeptanz der Benutzer



### Fazit:

- Tenant-Datenbanken – ein modernes und nützliches Feature
- Nicht nur für die "Cloud"
- Typisch INFORMIX: simple, elegant und ohne Zusatzkosten!

# Fragen?

Sprecher: Dr. Peter Ijewski  
[peter@ijewski.de](mailto:peter@ijewski.de)



*i&tp*





**Danke für Ihre  
Aufmerksamkeit!**



*i&tp*