

Von ENTW über TEST nach PROD - Drei SAS-Umgebungen bei der Fraport AG

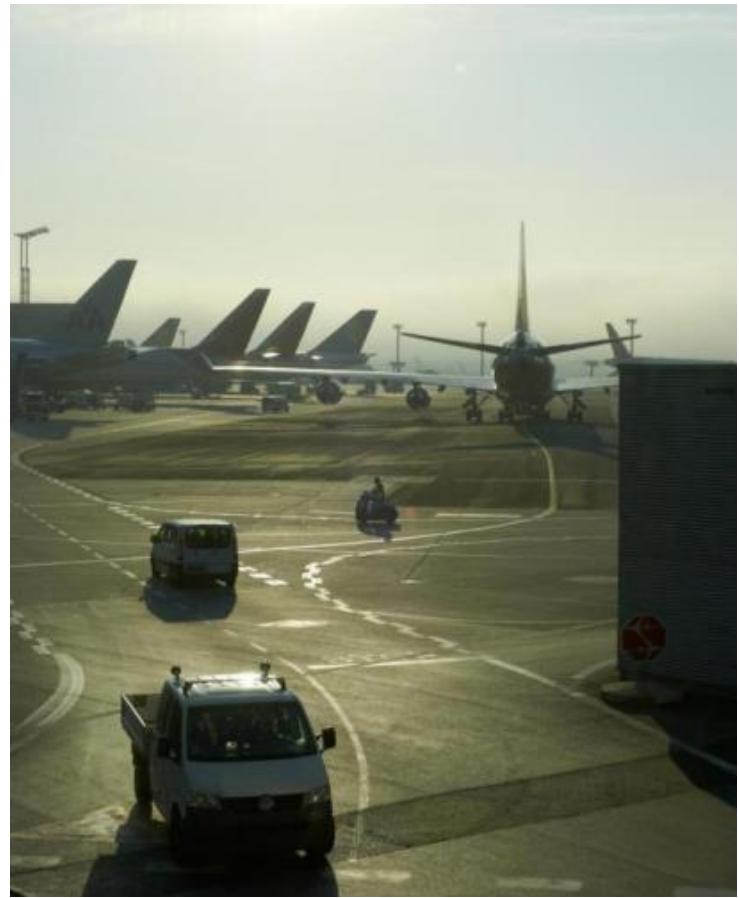
Frankfurt, 26. September 2014

Stefan Sabatzki



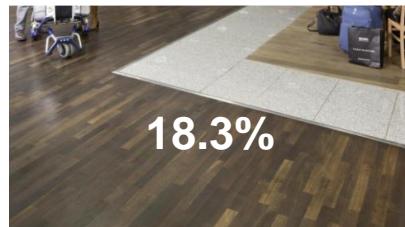
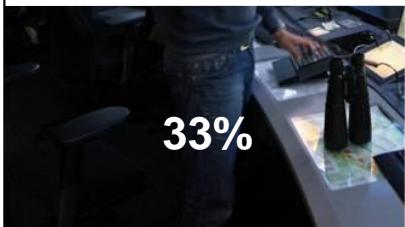
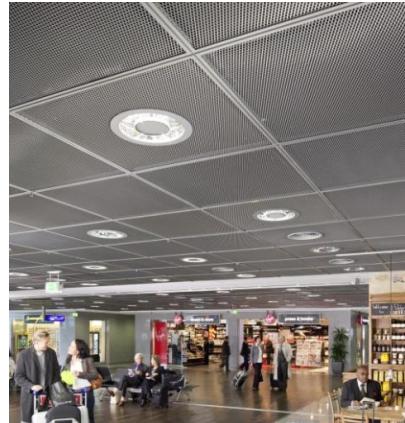
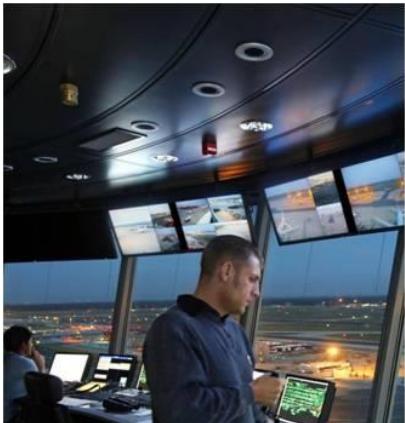
Agenda

- Unternehmenszahlen auf einen Blick
- Hintergrund und Motivation
- Synchronisation
- Fazit



Unternehmenszahlen auf einen Blick

Kerngeschäftsfelder



Aviation

845 mil. Euro

Ground Handling

656 mil. Euro

**Retail &
Real Estate**

469 mil. Euro

**External Activities
& Services**

591 mil. Euro

Frankfurt Airport Verkehr 2013

Veränderung 13/12 (%)



Passagiere 58.042.554

0,9



Luftfracht (t) 2.048.729

1,4



Luftpost (t) 79.165

-1,5



Bewegungen 472.692

-2,0

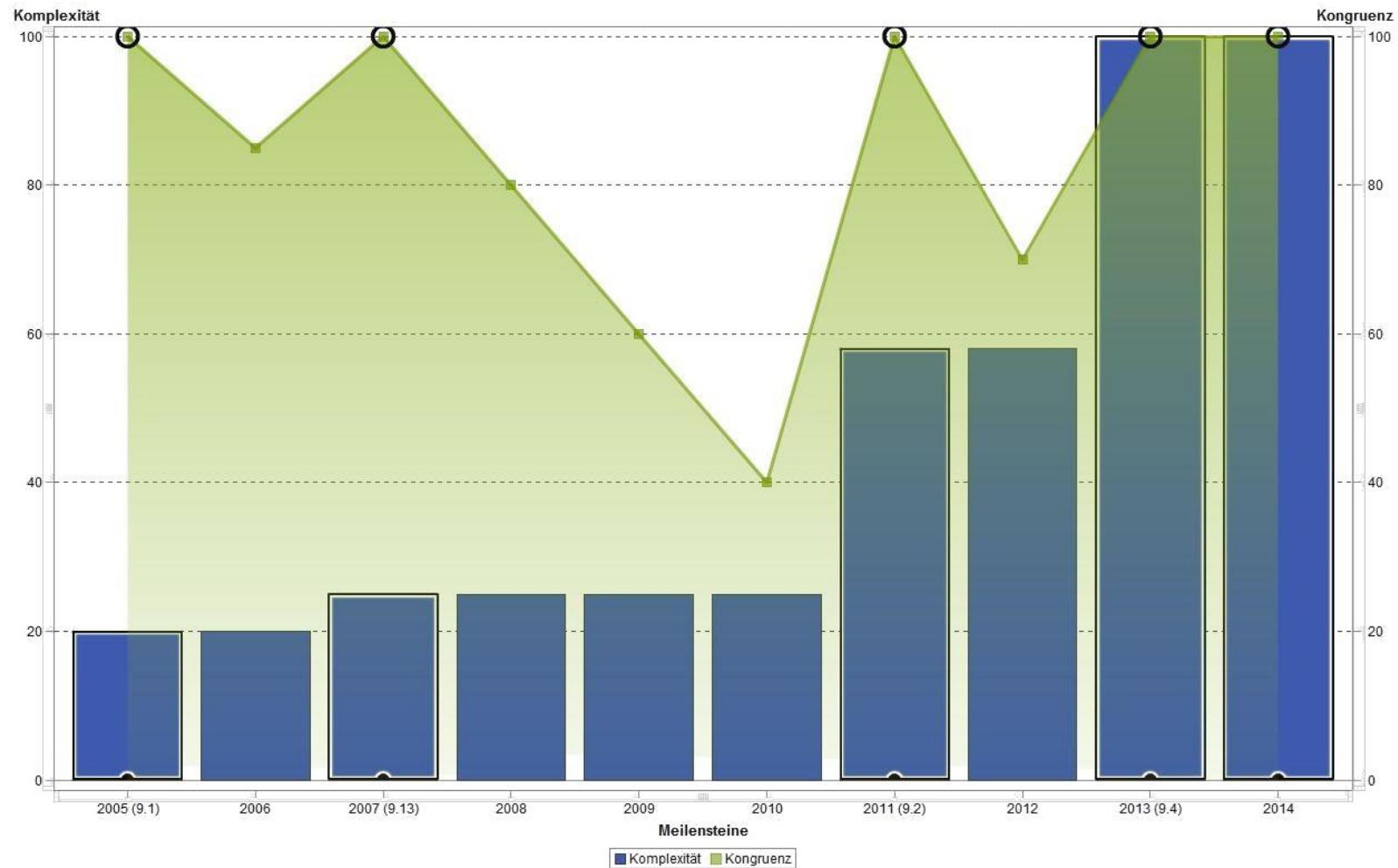


MTOW (t) 28.415.678

-1,7

Hintergrund und Motivation

Hintergrund



Motivation

- Welche Auswirkungen hat das Einspielen von einem Hotfix?
- Laufen meine SAS Prozesse noch, wenn ich an der Konfiguration etwas ändere?
- Wie komme ich an produktive Daten, um meinen Bericht zu testen?
- Ist diese Version des ETL-Jobs in ENTW/TEST aktuell?
- Wie kann ich ein infrastrukturelles Problem analysieren, ohne die produktive Umgebung zu beeinflussen?
- Aktuelle, synchrone und produktionsnahe Entwicklungs- und Testumgebungen erleichtern die Beantwortung dieser Fragen enorm.

Synchronisation

Synchronisation: Was denn eigentlich?

- Nutzdaten & Datenmodell
- Konfigurationsdateien
- Sourcecode (DI-Jobs, STPs, Macros, Includes)
- WRS Reports

- Konfiguration in den Metadaten (z.B. Serverkontexte)
- User / Berechtigungen
- Infomaps
- VA-Reports und Explorationen

- Grundsätzlich gilt: Die Produktionsumgebung ist der Master

Synchronisation: Allgemeines

Tue Jun 3 10:41:36 CEST 2014

Greenplum-DIFF Report

Tue Jun 3 10:41:36 CEST 2014

979 differences found for database biafhist! For more details please take a look at /BIAF/TEST/bin/compare/diffgp/diff_qp_biafhist.txt
208 differences found for database biafops! For more details please take a look at /BIAF/TEST/bin/compare/diffgp/diff_qp_biafops.txt

Conf-DIFF Report

Tue Jun 3 10:42:45 CEST 2014

List of unequal files

{env}/configcompute/Lev1/sas.servers.mid
{env}/configcompute/Lev1/sas.servers.mid2
{env}/configcompute/Lev1/SASREP/WorkspaceServer/logconfig.xml
{env}/configcompute/Lev1/SASREP/PooledWorkspaceServer/logconfig.xml
{env}/configcompute/Lev1/SASREP/StoredProcessServer/logconfig.xml
/BIAF/bincompute/gpODBC/odbc.sh
/BIAF/bincompute/gpODBC/odbc.ini
/BIAF/bincompute/SASFoundation/9.4/bin/sasenv_local

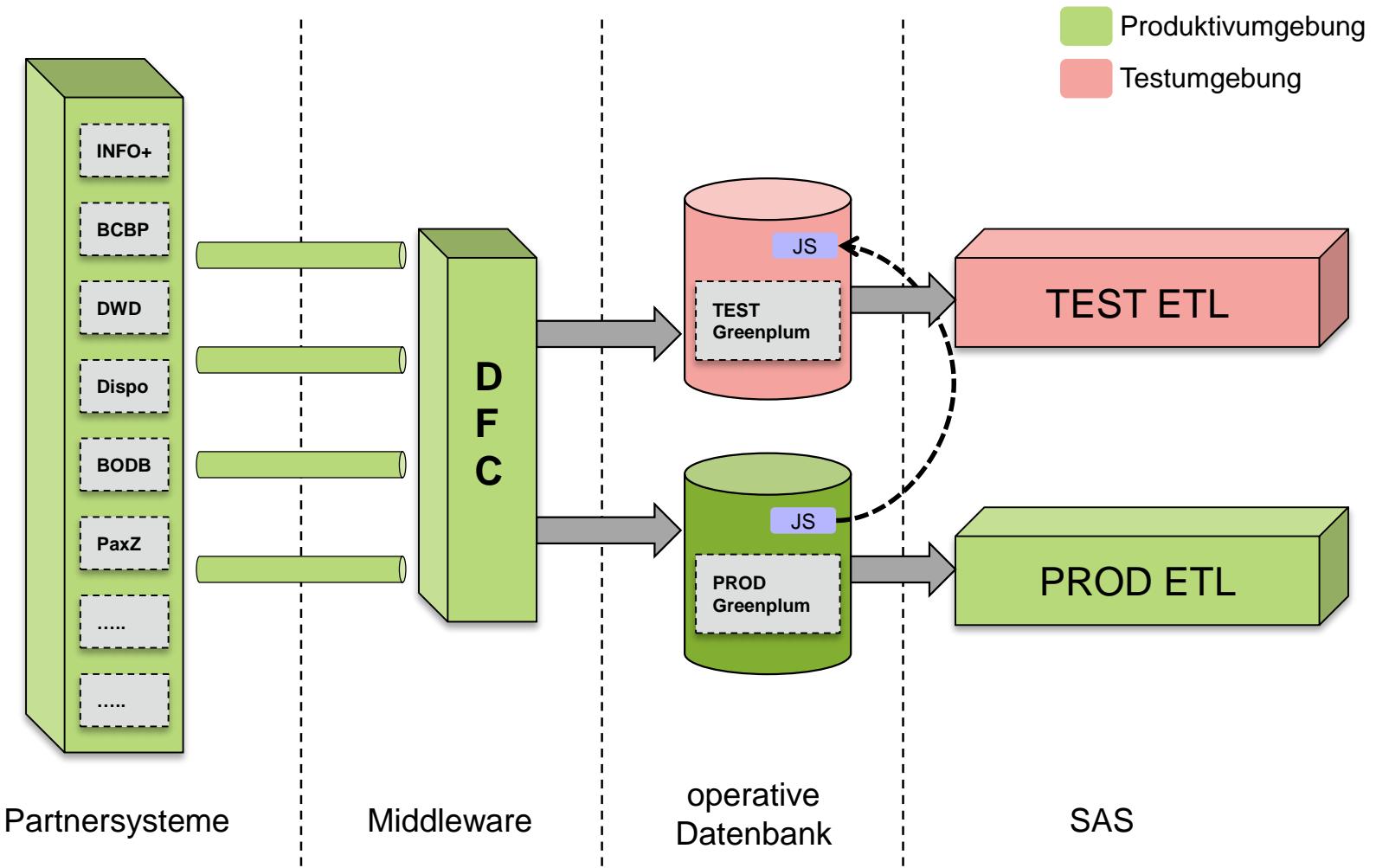
Synchronisation: Zu beachten

- Ausnahmen müssen definiert werden (sed und RegEx helfen hier)
 - Hostnames
 - Benutzernamen
 - Metadaten-IDs
 - Libnames
 - Generierte Kommentare
- Konfiguration der Clients standardisieren
 - Spracheinstellungen
 - DI Studio: Jobstatus Generierung, metadata macro variables
- Jeder Abgleich sollte vollumfänglich sein
- Verantwortlichkeiten sollten geregelt sein

Synchronisation: Nutzdaten

- Data Feed Connector (DFC)
 - Middleware-Komponente, eigenentwickelt in Java
 - Zentrale Anbindung von liefernden Systemen (Partnersysteme)
 - Messaging Infrastruktur auf Basis von Queues
 - Andere Übertragungswege werden unterstützt (SFTP, SCP, HTTP)
 - Bietet Transaktionssicherheit
 - Persistiert Daten in der Greenplum-Datenbank
 - Kann Daten „splitten“
- Delta-Load
 - ETL-Strecken verarbeiten i.d.R. nur neue Daten
 - Delta wird anhand des letzten Jobstatus ermittelt

Synchronisation: Nutzdaten II



Synchronisation: Datenmodell

- DDL wird mit Hilfe eines Greenplum-Skriptes „gedumpt“
- Verglichen werden
 - Tabellenstruktur
 - Grants
 - Spezielle Parameter (z.B. DISTRIBUTED BY)
- Ausnahme: Sequenzen

Synchronisation: Konfigurationsdateien

- Sehr umfangreich: Stand heute 207 Konfigurationsdateien
- SAS Plattform
 - sasv9.cfg
 - sasenv_local
- SAS Serverkontexte
 - Autoexecs
 - Start-/Stoppskripte
- ObjectSpawner
- LSF-System
 - Clusterkonfigurationen
 - Grid-Queue Definitionen
- Interfaces: ODBC.ini

Synchronisation: Sourcecode

- SAS Code ist vollumfänglich in den Metadaten
- Erste Idee: PROC METADATA
 - Pro: integrierter Ansatz, Abgleich an der Quelle, hohes Potential
 - Contra: intensive Einarbeitung, Objekte oft etwas unhandlich in den Metadaten
- Zweite Idee: Wiederverwendung des diff-Ansatzes
 - Pro: schnell umsetzbar, Erfahrung vorhanden
 - Contra: keine 100% Lösung
- Dritte Idee: Hybrid aus Batchdeployment und diff-Ansatz
 - Pro: Löst die Unzulänglichkeiten von Idee 2
 - Contra: Batchdeployment geht nicht mit 9.4 M0

Synchronisation: WRS Reports

- Format: XML
- Ablageort: WebDAV, Download via SAS-Programm
- Abgleich der Ordnerstruktur ebenfalls möglich
- Herausforderungen
 - Zeilenumbrüche

```
<?xml version="1.0" encoding="UTF-8"?><SASReport name="Rollzeiten-Mittelwerte über">
```

```
    <View style:scheme="_ReportScheme_">
```

```
<?xml version="1.0" encoding="UTF-8"?>
```

```
<SASReport name="Rollzeiten-Mittelwerte über Betriebsrichtung" version="1.0">
```

```
    <View style:scheme="_ReportScheme_">
```

- Schließende Tags

```
<Cell style:textAlign="right"></Cell>
```

```
<Cell style:textAlign="right"/>
```

Fazit

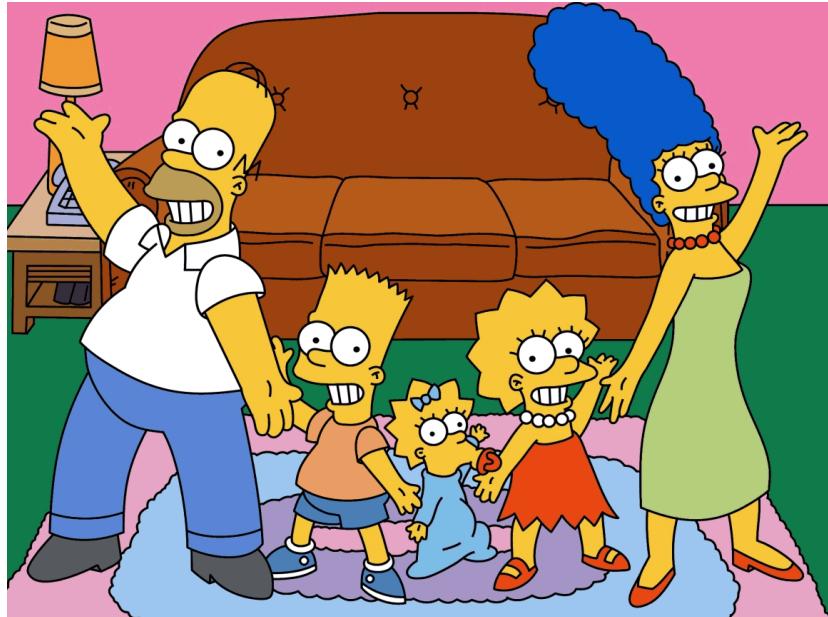
Die Erfahrung der letzten Jahre zeigt

- Synchronisation ist mit zunehmendem Komplexitätsgrad sehr wichtig
- Synchronisation ist eine enorme Arbeitserleichterung
- Synchronisation erhöht den Qualitätsgrad einer SAS Umgebung
- Fraport ist noch nicht am Ziel, aber ein Fundament ist vorhanden

Feature requests

- Integrierte Lösung, zum Beispiel in der Management Console
- Native Lösung, zum Beispiel mit PROC METADATA
- Deploy- und planbar, zum Beispiel als SAS-Job
- Umfangreiche Produktunterstützung neben den o.g. Beispielen
 - Enterprise Guide Projekte
 - Berichtsergebnisse (ActiveX, Images)

Viele Grüße aus Springfield



Vielen Dank für Ihre Aufmerksamkeit!



BACKUP – sed / RegEx in DI-Jobs

```
# gewollte/nicht vermeidbare Unterschiede vor dem Abgleich behandeln
sed -i 's/biafp/{METAENV}/g' $rootdir/prodsources/StoredProcess/$P_STPNAME
sed -i 's/fravm017090\|fravm017091\|fravm017106/{METAHOST}/g' $rootdir/prodsources/StoredProcess/$P_STPNAME
sed -i 's/p_biafhist/{HISTDBENV}/g' $rootdir/prodsources/StoredProcess/$P_STPNAME
sed -i 's/p_biafops/{OPSDBENV}/g' $rootdir/prodsources/StoredProcess/$P_STPNAME
sed -i 's/([SAS002]375CB633106666DE0FC00EB94C1B4BB5111B5B46\|{SAS002}670D353057ED23F52AEE6EE213F1A816\|{SAS002}670D35304D8E323126175D834AF1B329/{PASSWORD}/g' $root
sed -i 's/A5KLOBQK\.[A-Z0-9]\{8\}/{METADATAID}/g' $rootdir/prodsources/StoredProcess/$P_STPNAME
sed -i 's/[e0-9]\{5,6\}@fraport/{EMPLOYEEID}/g' $rootdir/prodsources/StoredProcess/$P_STPNAME
sed -i 's/* Generated on:.**//* Generated on: {GENERATEDDATE} */g' $rootdir/prodsources/StoredProcess/$P_STPNAME
sed -i 's/delete W[A-Z0-9]\{5,7\};/delete {WORKTABLE};/g' $rootdir/prodsources/StoredProcess/$P_STPNAME
sed -i 's/work\.\W[A-Z0-9]\{5,7\}\.work.{WORKTABLE}/g' $rootdir/prodsources/StoredProcess/$P_STPNAME
sed -i 's/index create XRef[A-Z0-9]\{6,8\}/index create {IDXNAME}/g' $rootdir/prodsources/StoredProcess/$P_STPNAME
sed -i 's/etls_loop[A-Z0-9]\{6,8\};/etls_loop{LOOPNAME};/g' $rootdir/prodsources/StoredProcess/$P_STPNAME
sed -i 's/etls_processToLoop[A-Z0-9]\{6,8\}/etls_processToLoop{LOOPNAME}/g' $rootdir/prodsources/StoredProcess/$P_STPNAME
sed -i 's/etls_job[A-Z0-9]\{6,8\};/etls_job{LOOPNAME};/g' $rootdir/prodsources/StoredProcess/$P_STPNAME
sed -i 's/\$let etls_controlName = L[0-9]\{2\}_;/\$let etls_controlName = {LOOPCTRLNR};/g' $rootdir/prodsources/StoredProcess/$P_STPNAME

sed -i 's/biaft/{METAENV}/g' $rootdir/testsources/StoredProcess/$T_STPNAME_TMP
sed -i 's/fravm017100\|fravm017101\|fravm017107/{METAHOST}/g' $rootdir/testsources/StoredProcess/$T_STPNAME_TMP
sed -i 's/t_biafhist/{HISTDBENV}/g' $rootdir/testsources/StoredProcess/$T_STPNAME_TMP
sed -i 's/t_biafops/{OPSDBENV}/g' $rootdir/testsources/StoredProcess/$T_STPNAME_TMP
sed -i 's/([SAS002]670D35304D8E323126175D834AF1B329\|{SAS002}670D353057ED23F52AEE6EE213F1A816/{PASSWORD}/g' $rootdir/testsources/StoredProcess/$T_STPNAME_TMP
sed -i 's/A5FKUPWV\.[A-Z0-9]\{8\}/{METADATAID}/g' $rootdir/testsources/StoredProcess/$T_STPNAME_TMP
sed -i 's/[e0-9]\{5,6\}@fraport/{EMPLOYEEID}/g' $rootdir/testsources/StoredProcess/$T_STPNAME_TMP
sed -i 's/* Generated on:.**//* Generated on: {GENERATEDDATE} */g' $rootdir/testsources/StoredProcess/$T_STPNAME_TMP
sed -i 's/delete W[A-Z0-9]\{5,7\};/delete {WORKTABLE};/g' $rootdir/testsources/StoredProcess/$T_STPNAME_TMP
sed -i 's/work\.\W[A-Z0-9]\{5,7\}\.work.{WORKTABLE}/g' $rootdir/testsources/StoredProcess/$T_STPNAME_TMP
sed -i 's/index create XRef[A-Z0-9]\{6,8\}/index create {IDXNAME}/g' $rootdir/testsources/StoredProcess/$T_STPNAME_TMP
sed -i 's/etls_loop[A-Z0-9]\{6,8\};/etls_loop{LOOPNAME};/g' $rootdir/testsources/StoredProcess/$T_STPNAME_TMP
sed -i 's/etls_processToLoop[A-Z0-9]\{6,8\}/etls_processToLoop{LOOPNAME}/g' $rootdir/testsources/StoredProcess/$T_STPNAME_TMP
sed -i 's/etls_job[A-Z0-9]\{6,8\};/etls_job{LOOPNAME};/g' $rootdir/testsources/StoredProcess/$T_STPNAME_TMP
sed -i 's/\$let etls_controlName = L[0-9]\{2\}_;/\$let etls_controlName = {LOOPCTRLNR};/g' $rootdir/testsources/StoredProcess/$T_STPNAME_TMP
```

BACKUP – Ergebnis mit Unterschieden

```
* p_av_01_fkt_flugbewegungen_V4.2          |      * p_av_01_fkt_flugbewegungen_V4.1
* V4.2 - 28.05.2014 - Sabatzki             <
* EM_BEWEGUNGEN_IST berücksichtigt keine Rückkehrer vom Vor <
*
((CASE      WHEN IP_FLIGHTTYPE = 'X' THEN 0      |      ((CASE IP_FLIGHTTYPE WHEN 'X' THEN 0 WHEN 'S' THEN 0
WHEN IP_FLIGHTTYPE = 'S' THEN 0              <
WHEN IP_FLIGHTTYPE = 'Z' THEN 0              <
ELSE 1                                     <
END)) as EM_BEWEGUNG_IST length = 8          <
```

```
* sp_av_aogh_performance_input_V2.0          |      * sp_av_aogh_performance_input_V1.0
* XX.XX.XXXX - Stefan Sabatzki - V2.0       <
* Migration nach 9.4                         <
*
```