

# Automatisiertes System Cloning für SAP

Franz Diegruber

Libelle AG

Stuttgart

## Schlüsselworte

Oracle & SAP. System Cloning. System Rename.

## Einleitung

Vor umfassenden und komplexen Veränderungen in der SAP-Landschaft, wie sie beispielsweise EHP-Upgrades, Modulaktualisierungen oder die Einführung umfangreicher Entwicklungsprojekte etc., werden diese klassischerweise in nicht-produktiven Umgebungen auf Herz und Nieren getestet. Nicht immer ist es hierbei gewünscht mit diesen Sonderprojekten die bestehenden QA-Systeme zu belasten, so dass zusätzliche, neue Projekt-/Sandboxsysteme mit aktuellen Daten benötigt werden, die quasi „auf der grünen Wiese“ aufgebaut werden.

Während für System- und Mandantenkopien zum Datenrefresh vorhandener Systeme reife Werkzeuge im Markt existieren, mit denen man diese üblicherweise mit hohem manuellem Aufwand verbundenen Aufgabenstellungen vollständig automatisieren kann, gab es für das sogenannten System Cloning bislang noch keine durchgängige Automatisierungsmethode.

Dieser Vortrag zeigt, mit welchen Mitteln und auf welcher Basis inzwischen auch System Clones in einen flexiblen Automatismus übernommen werden können, der den Aufwand des Neuerstellens von Projekt-/Sandboxsystemen auf ein absolutes Minimum reduziert.

Es werden integrierte Vorgehensweisen vorgestellt, die auf Basis alternativer unterschiedlicher Ausgangspositionen (z.B. Oberkante Betriebssystem, Oberkante Oracle DBMS, Oberkante SAPinstall) zu einem Projektsystem oder auch zu vollständigen Projekt-Landschaften führen.

In diesem Vortrag wollen wir das homogene System Cloning durchgehen. Hierfür ist Voraussetzung dass das OS und DBMS auf Quell- und Zielsystem gleich ist und auch mit dem identischen Release. Ggfs. können auf dem Zielsystem dann Updates/upgrades nach dem SystemClone gefahren werden. Was ja u.a. ein Grund für den Aufbau des Zielsystems ist.

## Anforderungen

Zuerst müssen die Anforderungen an die neue Umgebung geklärt werden:

- Identischer SAP-Systemnamen wie Quellsystem und auch identische DB-SID oder geändertere Namen und SID
- Leeres System oder produktive Daten. Wenn produktive Daten genutzt werden, dann sollte auch bedacht werden ob die Daten maskiert werden müssen.
- Isoliertes System oder mit Verbindungen zu anderen Systemen.

## Scenario

Hier ist zu klären was ist auf dem Zielsystem schon vorhanden:

- Oberkante Betriebssystem
- Oberkante Oracle DBMS
- Oberkante SAPinstall

## Zu berücksichtigen im SAP System

Wird das bestehende System kopiert, muss auf dem Zielsystem darauf geachtet werden dass gewisse Dinge angepasst bzw. gelöscht werden. Ansonsten kann dies zur Beeinflussung/Korruption der produktiven Landschaft führen.

- Jobs

- User
- RFC connections
- BDLS
- SGEN ausführen

### Arten das Zielsystem zu erzeugen

- System auf storage Ebene erzeugen
- System auf virtuelle Ebene erzeugen
- System mit Custom Schritten erzeugen

### Umsetzung

Wir wollen uns darauf konzentrieren die Umsetzung mit Custom Schritten durchzuführen.

### Auf Skriptbasis

Dies ist die flexibelste Möglichkeit, da hier jeder einzelne Schritt individuell definierbar ist und hier wieder aufgesetzt werden kann.

Wir gehen von folgender Ziel-Umgebung aus:

- Oberkante Betriebssystem
- Kein DBMS installiert
- Kein SAP System installiert

### Schematischer Überblick:

- Ermitteln Oracle und SAP User, IDs und Gruppen auf Quellsystem
- Anlegen Oracle und SAP User, IDs und Gruppen auf Zielsystem
- Kopieren von Oracle und SAP Verzeichnissen
- Anpassen SAP profile
- Nacharbeiten Oracle DBMS Kopie
- Erzeugen neue Secure Store Properties Files für J2EE
- Kopieren der DB
- DB auf dem Zielsystem öffnen mit geänderter SID
- Erzeugen OPS\$ user. Neu SSFS
- Reset ABAP und JAVA schema user password mit BRCONNECT
- Start Oracle Listener auf Zielsystem
- Start SAP auf Zielsystem

Hier gibt es auch von Libelle ein Framework (**Libelle SystemClone®**) welches den Ablauf bereits standardisiert hat und auch während der Durchführung auf saubere Ausführung prüft.

The screenshot shows the 'CLONE\_ORA\_UNIX' window with a sidebar on the left containing menu items: General, Systems, Categories, Copy Tool, Data Masking, Tasks (highlighted), Alarm, Global Parameters, Snippets, and Execution. The main area contains a task table with the following data:

Phase	UID	Name	Type	Wait
Pre 1	LDBGETUSRID	DB: Get ID of DB User	cmd	No
Pre 2	LDBGETGRPID	DB: Get ID of DB Group DBA	cmd	No
Pre 3	LDBGETGRPID2	DB: Get ID of DB Group OPER	cmd	No
Pre 4	LDBGRPCRT	Create DB Groups on Target System	cmd	Yes
Pre 5	LSAPGETUSRID	DB: Get ID of SAP User	cmd	No
Pre 6	LSAPGETGRPID	DB: Get ID of SAP Group SAPSYS	cmd	No
Pre 7	LOSGRPCRT	Create OS Group on Target System	cmd	No

Abb. 1: Libelle System Clone GUI

## **SAPinst / Software provisioning manager (SPM)**

Der SAP Landscape Virtualization Manager (LVM) bietet auch verschiedene Möglichkeiten Systeme zu clonen, soll hier aber nicht betrachtet werden, da hiermit ja gewisse Kosten verbunden sind.

Hier wird das SAP Tool SAPinst/SPM genutzt.

Es sollte darauf geachtet werden dass jeweils die letzte verfügbare Version des SPM heruntergeladen wurde und dass auch entsprechend (muss zum SPM passen) die Packages zur Installation verfügbar sind.

Die Durchführung des SPM kann interaktiv über einen Java GUI erfolgen oder automatisiert. Für die automatisierte Variante müssen allerdings einige Dinge vorbereitet werden.

### **Interaktiv**

Hier müssen in der dialog-phase verschiedene Parameter in Textfeldern mitgegeben werden.

Nach dieser dialog-phase wird eine Art batch-phase gestartet in der dann die angegebenen Parameter für die Installation genutzt werden.

### **Automatisiert**

Um eine Automatisierung zu ermöglichen, müssen die Parameter für die Installation in flat-files mitgegeben werden.

- inifile.xml : Parameter file. Dies wird normalerweise in der dialog-phase befüllt.
- start\_dir.cd : Liste der mount points zu den Installations-Medien.

Beide Files müssen im Installationsverzeichnis (SAPINST\_CWD) zur Verfügung gestellt werden und werden dann automatisch als input files erkannt.

Um den SAPinst dann im unattended mode zu starten müssen ein paar Optionen mitgegeben werden:

```
/<PATH_TO_IM>/sapinst  
  
SAPINST_PARAMETER_CONTAINER_URL=inifile.xml  
  
SAPINST_EXECUTE_PRODUCT_ID=<PRODUCT ID>  
  
SAPINST_SKIP_DIALOGS=true  
  
-nogui  
  
-noguiserver
```

### **Weitere wichtige Files des SAPinst**

- control.xml: Enthält die komplette code basis von jedem Installations Szenario. Der code ist in JavaScript geschrieben.
- keydb.xml: Stellt das status file dar. Jeder Schritt in der Installation ist bei einem XML-element präsentiert. Bei Erfolg wird das Element dann auf „OK“ gesetzt ansonsten auf „ERROR“. Wenn der Prozess aufgrund eines Fehler beendet wurde, dann wird auf dem ersten fehlerhaften Schritt wiederaufgesetzt.
- sapinst.log/sapinst\_dev.log: Diese beiden Files stellen logfiles dar für die gesamte Installation. Der hauptsächliche Unterschied ist dass im sapinst\_dev.log die Informationen mehr detailliert enthalten sind. Diese Files können genutzt werden um Fehler zu ermitteln, während der SAPinst ausgeführt wird. Sie enthalten auch Hinweise wo der Fehler gefunden werden kann.

### Voraussetzung für einen sauberen Lauf des SAPinst

- Sauberes Zielsystem da der SAPinst hier sehr empfindlich reagiert, falls hier schon etwas vorhanden ist:
  - keine SAPSID user vorhanden
  - keine ports in /etc/services welche zu SAPSID und/oder DBSID gehören
  - keine „alte“ Installation in /usr/sap/SAPSID
  - keine „alte“ Installation im DBMS Verzeichnis

### Schematischer Überblick:

- Installation Oracle RDBMS plus patches auf Zielsystem
- SAPinst ausführen und Fortschritt überwachen
- SAPinst ggfs. nochmals starten (bei verteilten Systemen)

**Achtung:** Für die Zeit des exports der DB aus dem Quellsystem ist eine downtime erforderlich. Ausserdem ist eine storage für diesen export erforderlich, welcher auch vom Zielsystem aus zugegriffen werden kann.

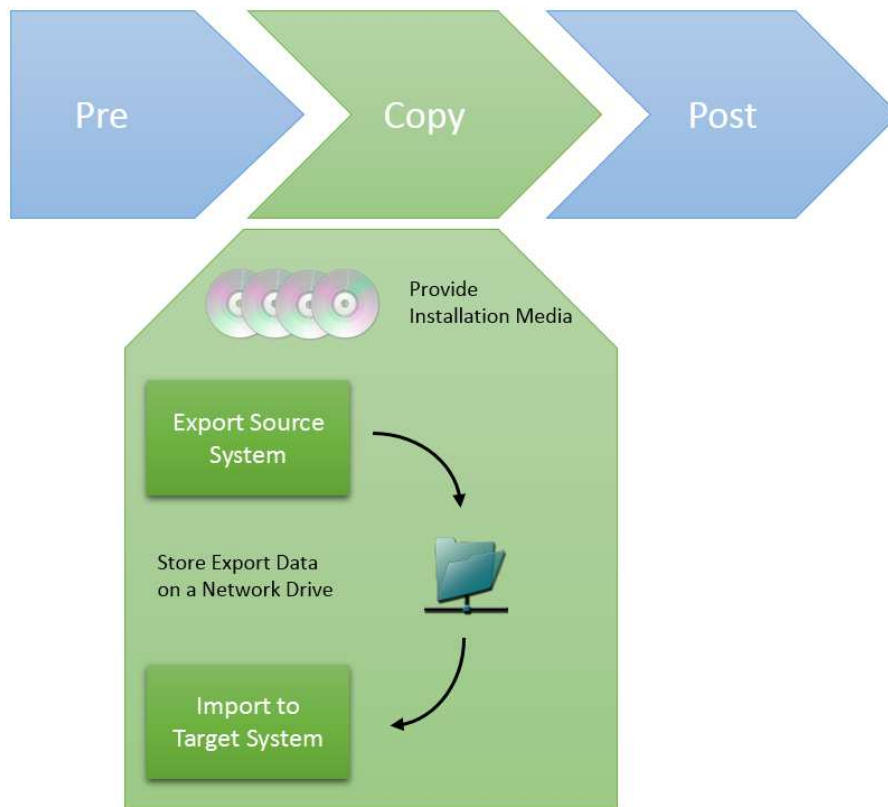


Abb. 2: Ablauf mit dem SAPinst

### Zusammenfassung

Es gibt verschiedene Möglichkeiten das Erstellen eines Clones einer SAP Umgebung zu automatisieren. Eine vollständige Automatisierung ist im Moment noch nicht immer gegeben. Es sind auch einige Vorarbeiten zu leisten. Welche Möglichkeit für die Erstellung gewählt wird ist sehr stark davon abhängig was die Intention ist den Clone zu erstellen.

Der Aufbau mittels Skripten bzw. Kopieren der kompletten Maschine (auf Storage Ebene oder als virtuelle Maschine) ist erforderlich damit bei einem Upgrade-Check von den wirklich gleichen Voraussetzungen ausgegangen wird wie bei der Quell-Maschine. Würde man hierfür den SAPinst nutzen, hätte man ein „cleanes“ System erzeugt welches z.B. keinen Schmutz von vorhergehenden/alten Installation wie ggfs. auf dem alten Quell-System hat und der Upgrade-Test wäre nicht aussagekräftig da hier von unterschiedlichen Voraussetzungen bezüglich Umgebung ausgegangen wird.

Will man allerdings ein komplett neues System z.B. für Schulungszwecke aufbauen, will man genau diese Unsauberkeiten vermeiden und somit ist der Aufbau über SAPinst vorzuziehen.

**Kontaktadresse**

Franz Diegruber  
Libelle AG  
Gewerbestr. 42  
D-70565 Stuttgart

Telefon: +49 (0) 711-78335 312  
Fax: +49 (0) 711-78335 148  
E-Mail: [fdiegruber@libelle.de](mailto:fdiegruber@libelle.de)  
Internet: [www.libelle.de](http://www.libelle.de)