



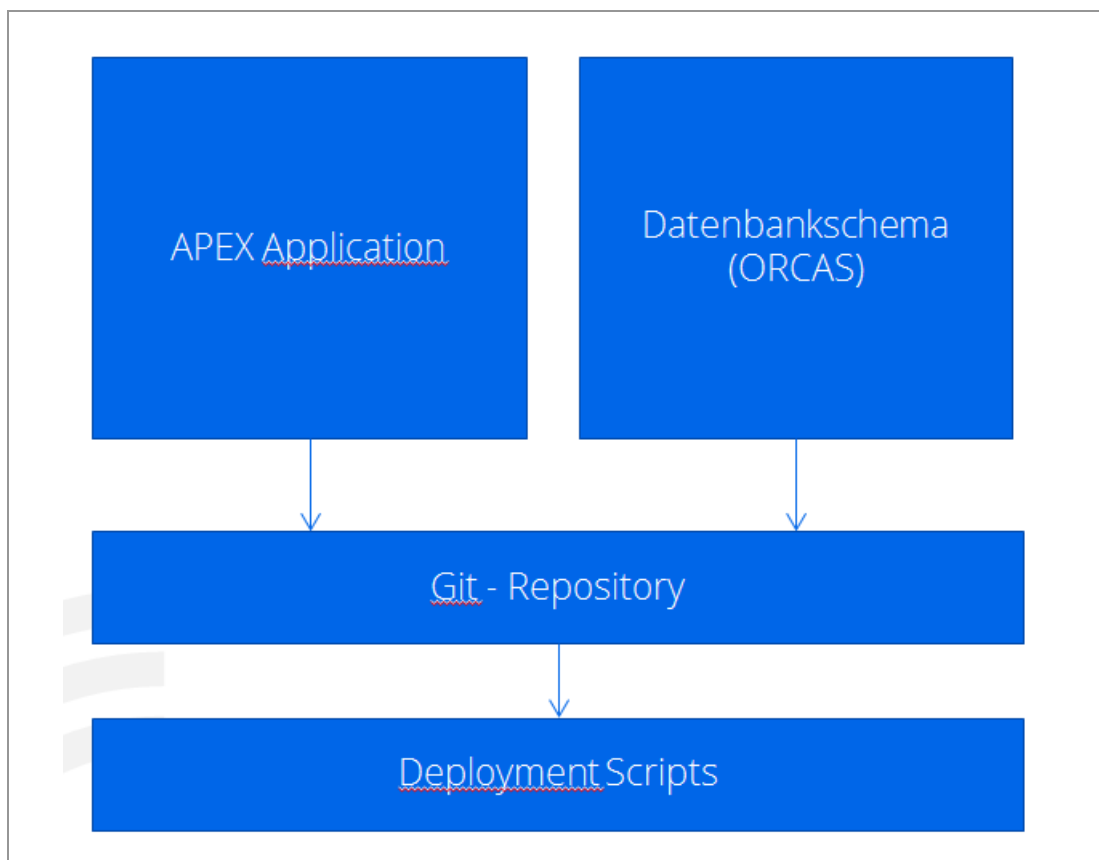
# CI von APEX Projekten

## 1 Einleitung

In jedem Application Development Projekt wird sich zu Beginn über ein Vorgehen für Delivery und Versionierung des Produkts unterhalten. Gerade in Bezug auf agile Softwareentwicklungsprojekte ist dies ein ausschlaggebender Faktor.

Im Bereich der .NET oder Java Entwicklung stoßen wir häufig auf CI-Lösungen wie Jenkins o.ä. In APEX Projekten, gehen wir selten den Weg über ein CI Tool. Warum nicht?

Der entscheidende "Nachteil", wir haben ein Datenbankschema, dass eng mit unserer Anwendung verbunden ist und bei einem Deployment genauso berücksichtigt werden muss wie jede andere Quelle auch (siehe Abb.1).



**Abb. 1.** Konzeptbild



Das Konzept des entwickelten Deployment-Prozesses sieht daher vor, den Source Code der APEX Applikation und aller zugehörigen Datenbankschemata zu extrahieren, in ein VCS einzuspielen und im Idealfall, das Deployment durch ein Continuous Integration Tool abzubilden.

In diesem Dokument wird beschrieben, wie der Source Code von APEX Anwendungen und Datenbankschemata extrahiert werden kann.

## 2 Installation

Wie oben bereits beschrieben, müssen zum Deployment von APEX Anwendungen mehrere Aspekte betrachtet werden:

- Die APEX Anwendung
- Das Datenbankschema

Die APEX Anwendung kann mittels dem von ORACLE bereitgestellten Programm APEXExport extrahiert werden. Dieses erstellt, genau wie die Export-Funktion innerhalb von APEX, ein einzelnes großes SQL File. In dem entwickelten Ansatz wird diese Datei deshalb mit Hilfe des APEXSplitters noch in einzelne Dateien zerteilt. Dies ermöglicht eine bessere Nachverfolgung der vorgenommenen Änderungen beim Commit in ein VCS.

Das Extrahieren des Datenbankschemas wird über ORCAS gelöst. ORCAS ist ein Open Source Projekt, dass es ermöglicht, ein Datenbankschema in einen in Textdateien, beschriebenen Zielzustand zu überführen. Dabei ist es ebenso möglich, ein Datenbankschema in einem für ORCAS lesbaren Format zu extrahieren, auf dessen Grundlage das Zielschema angepasst werden kann.

Um mittels ORCAS und APEXExport APEX Projekte zu exportieren, müssen die im Folgenden beschriebenen Schritte durchgeführt werden.

### 2.1 Installation ORCAS

Um ORCAS zu nutzen zu können brauchen Sie die folgende Software:

Software	Link	Version
Ant	<a href="http://ant.apache.org/bindownload.cgi">http://ant.apache.org/bindownload.cgi</a>	1.10.0
Ant Contrib	<a href="https://sourceforge.net/projects/ant-contrib/files/ant-contrib">https://sourceforge.net/projects/ant-contrib/files/ant-contrib</a>	latest
Gradle	<a href="https://services.gradle.org/distributions">https://services.gradle.org/distributions</a>	3.2.5
Java	<a href="http://www.oracle.com/technetwork/java/javase/downloads/jdk8-downloads-2133151.html">http://www.oracle.com/technetwork/java/javase/downloads/jdk8-downloads-2133151.html</a>	JDK8
ORCAS	<a href="https://github.com/opitzconsulting/orcas">https://github.com/opitzconsulting/orcas</a>	latest

**Tab. 1.** Benötigte Software



Nachdem Sie die benötigte Software heruntergeladen haben, bitte der folgenden Anleitung folgen (für Oracle Enterprise Linux):

1. Einen neuen Benutzer orcas anlegen (useradd orcas)
2. Gradle installieren
  - a. Gradle in ein Verzeichnis ihrer Wahl extrahieren
  - b. Eine Umgebungsvariable GRADLE\_HOME für Benutzer orcas anlegen (Ziel: <Installationspfad>/bin)
3. Ant installieren
  - a. Ant in ein Verzeichnis ihrer Wahl extrahieren
  - b. Eine Umgebungsvariable ANT\_HOME für Benutzer orcas anlegen (Ziel: <Installationspfad>/bin)
4. Ant Contrib entpacken
  - a. Die Datei ant-contrib-1.0b3.jar nach \$ANT\_HOME/lib kopieren
  - b. Der Dateiname kann ggf. je nach Version abweichen
5. Java8 JDK installieren (Anleitung für RHEL liegt bei)
6. Die PATH Variable um \$GRADLE\_HOME und \$ANT\_HOME ergänzen
7. ORCAS in ein Verzeichnis ihrer Wahl extrahieren

## 2.2 Konfiguration der Ex- und Importskripte

Wenn Sie die mitgelieferte zip-Datei entpacken, finden Sie darin die folgenden Ordner und Skripte:

Datei/Ordner	Zweck
apex_sync	Konfigurationsdateien ORCAS
app_install.sql	Template-Datei zur Installation der APEX-Anwendung
copyFilesTo.sh	Kopiert alle zu einer Applikation gehörenden Dateien an einen beliebigen Ort
create_export_script.sh	Setup
deploy.sh	Template Datei zum Starten des Deployment Prozesses
export_app.sh	Template-Datei zum Export der APEX-Anwendung
export_fxx.sh	Template-Datei zum Starten des Exports
export_splitter.sh	Template-Datei zum aufspalten des Export-Files von APEXExport
README	Weitere Informationen zum Konfigurieren von ORCAS und den generierten Dateien

**Tab. 2.** APPLE-Files

Um die für den Export und das Deployment wichtigen Dateien zu generieren, starten Sie einfach das Skript create\_export\_script.sh. Es werden eine Reihe von Informationen vom Skript abgefragt (siehe Tab. 3.). Sobald diese Informationen angegeben wurden, haben Sie einen Großteil der Arbeit schon erledigt. Es müssen nun noch die Dateien build.xml und location.properties die in einem Unterordner von apex\_sync zu finden sind, und die Datei app\_install.sql angepasst werden.



Variable	
DBHOST	Host der Datenbank, Standard: 127.0.0.1
DBPORT	Port der Datenbank, Standard: 1521
SID	SID der Datenbank, Standard: xe
Application ID	ID der Applikation für die die Skripte generiert werden sollen
ORCAS_HOME	Pfad der build.xml nach der Konfiguration
ORACLE_HOME	Der Installationspfad der Oracle Instanz
JAVA_HOME	Der Installationspfad von Java
APEX_HOME	Installationsdateien von APEX

**Tab. 3.** Setupinformationen

## 2.2.1 Anpassungen an der build.xml

Bevor mit der Konfiguration der build.xml begonnen wird, kopieren Sie die Dateien mittels des copyFilesTo-Skripts, in das Zielverzeichnis. Danach rufen Sie den Unterordner ".../apex\_sync/schema\_sync/distribution/" auf, indem sich die build.xml befindet.

In der build.xml müssen in den folgenden Zeilen die Werte in <GROßBUCHSTABEN> bearbeitet werden:

Zeile	Inhalt
17	<property name="orcas_dir" value="<PATHTOORCAS>"/>
50	<orcas_extract outputfolder="<EXPORTPATH>/db/tables" user="{username_schemaowner}" password="{password_schemaowner}" tnsname="{database}" jdbcurl="{jdbc_url}" />
51	<orcas_extract_replaceables outputfolder="<EXPORTPATH>/db/replaceables" user="{username_schemaowner}" password="{password_schemaowner}" tnsname="{database}" jdbcurl="{jdbc_url}" />
56	<orcas_execute_statics createmissingfkindexes="false" scriptfolder="<EXPORTPATH>/db/tables" spoolfolder="{binrundir}/log" logname="tables" dropmode="false" user="{username_schemaowner}" password="{password_schemaowner}" tnsname="{database}" jdbcurl="{jdbc_url}" />
57	<orcas_update_replaceables scriptfolder="<EXPORTPATH>/db/replaceables" spoolfolder="{binrundir}/log" logname="sync_replaceables" user="{username_schemaowner}" password="{password_schemaowner}" tnsname="{database}" jdbcurl="{jdbc_url}" />

**Tab. 4.** build.xml



## 2.2.2 Anlegen einer Location

Im Ordner ".../apex\_sync/schema\_sync/distribution/location/" befindet sich die Datei location.properties. In dieser Datei sind die folgenden Werte einzufügen:

Variable	Wert
database	TNS-Name der Datenbank
jdbc_host	Hostname der Datenbank
jdbc_sid	SID der Datenbank
jdbc_port	Port der Datenbank (Standard: 1521)
username_schemaowner	Benutzername des Besitzers der Datenbankobjekte (Applikationsschema)
password_schemaowner	Passwort des Datenbankusers

**Tab. 5.** location.properties

Sobald dies erledigt ist, sollte die Anwendung sauber exportiert werden können.

## 2.2.3 Anpassungen in der app\_install.sql

In der Datei app\_install.sql gibt es einige Platzhalter die Sie anpassen müssen, bevor Sie eine Anwendung auf ein Zielschema ausliefern wollen.

Die folgenden Werte sind dabei anzupassen:

Variable	Wert
#WORKSPACE_NAME#	Name des Zielworkspace
#SCHEMA#	Benutzername des Besitzers der Datenbankobjekte (Applikationsschema)
#APPLICATION_ALIAS#	Name des Zielworkspace

**Tab. 6.** app\_install.sql

Die Datei muss letztendlich im Exportordner (f<Applikationsnummer>/) der Anwendung liegen. Das bedeutet die Anwendung muss zunächst einmal exportiert werden.

An diesem Punkt sind Sie fertig mit der Installation.



## 3 Nutzung

Im Folgenden wird erläutert, wie APEX Applikationen exportiert und importiert werden können.

### 3.1 Exportieren einer APEX Applikation

Um APEX Anwendungen zu exportieren benutzen Sie einfach das Skript `export_f<Applikationsnummer>`, das als Übergabeparameter das Passwort des Benutzers "system" erwartet.

Folgende Schritte werden ausgeführt:

Komponente	Zweck
APEXexport	Exportieren der APEX Applikation als "große" sql-Datei
APEXsplitter	Umbauen der Export sql-Datei in eine feingranulare Ordner-/Dateistruktur
Extract	Exportieren aller benötigten Datenbankobjekte

**Tab. 7.** APEX Export

An diesem Punkt nicht vergessen die `app_install.sql` in den Exportordner zu kopieren. Sie werden nach jedem Exportvorgang auch explizit nochmal darauf hingewiesen.

HINWEIS: Im Normalfall sollte die `app_install.sql` nur einmal kopiert/verschoben werden müssen.

### 3.2 Importieren einer APEX Applikation

Um eine APEX Anwendung zu importieren, benutzen Sie einfach das Skript `deploy_f<Applikationsnummer>.sh`. Dieses Skript erwartet als Übergabeparameter den Benutzernamen und Passwort des Zielschemas.

Folgende Schritte werden ausgeführt:

Komponente	Zweck
Deploy	Synchronisation der DB-Objekte
<code>app_install.sql</code>	Installation der APEX Anwendung

**Tab. 8.** APEX Import