

Tipps und Tricks aus Gerds Fundgrube

Heute: PLL vs. PLX

Gerd Volberg, OPITZ CONSULTING Deutschland GmbH

Libraries, die in Oracle Forms eingebunden werden, dürfen zur Laufzeit als PLL oder PLX hinterlegt werden. Beide Formate sind unterstützt. Aber welche Variante ist besser?

Schauen wir uns vor der Beantwortung dieser Frage zwei Fälle an:

Beispiel 1: Bei der Entwicklung werden sowohl PLL als auch PLX genutzt. Der Forms Builder benutzt das Arbeitsverzeichnis „C:\Forms“ und im „FORMS_PATH“ steht der Verweis auf „C:\Forms\Lib“. Die Sourcecodes heißen „lib.pll“, „menu.mmb“ und „emp.fmb“, die generierte Library „lib.plx“. Die Libraries werden in folgender Reihenfolge gesucht (*siehe Abbildung 1*):

1. Wenn die „lib.plx“ im Arbeitsverzeichnis liegt, wird sie genutzt
2. Wird sie nicht gefunden, suche im „FORMS_PATH“ nach der PLX
3. Wird sie nicht gefunden, suche im Arbeitsverzeichnis nach der PLL
4. Wird sie nicht gefunden, suche im „FORMS_PATH“ nach der PLL
5. Wird sie nicht gefunden, erzeugt Forms eine Fehlermeldung

Dies gilt genauso für Forms-Masken und Menüs, jedoch mit der Ausnahme, dass dort nur die kompilierten Versionen FMX und MMX relevant sind.

Wenn das PLX im lokalen Verzeichnis nun eine ältere Version ist als die entsprechende PLL am gleichen Ort, dann führt dies zu großen Problemen während der

Laufzeit. Den Fehler zu finden, ist nahezu unmöglich, solange der Entwickler die Zeitstempel der Libraries nicht überprüft.

Beispiel 2: Bei der Entwicklung wird nur PLL eingesetzt. Während der Entwicklung und zur Laufzeit hat man keine Probleme. Wenn eine Library nicht im Arbeitsverzeichnis liegt, wird sie im „FORMS_PATH“ gefunden.

Fazit

Der Autor nutzt keine PLX mehr, da er in zu vielen Projekten mit den beschrie-

benen Problemen konfrontiert war. Die Fehlersuche bei größeren Forms-Pfaden ist enorm aufwändig, wenn dort irgendwo eine veraltete PLX liegt.

Viele Entwickler sind der Meinung, dass PLX schneller ist. Dies konnte er bisher nicht nachvollziehen und hatte auch noch nie Performance-Probleme. Andere Entwickler sagen, PLX sei kompilierter Code, somit kann niemand den Sourcecode einsehen oder stehlen – in diesem Fall empfiehlt der Autor, einmal Folgendes auszuprobieren: Eine Library mit folgendem Package erzeugen und daraus eine PLX generieren (*siehe Listing 1*).

Sucht man in der PLX mit einem Editor nach „HiddenPW“, wird man sowohl den Namen der Konstanten als auch deren Wert finden. Eine PLX schützt also diesbezüglich nicht. Aus Sicht des Autors überwiegen die Vorteile, wenn man ausschließlich mit PLLs arbeitet.

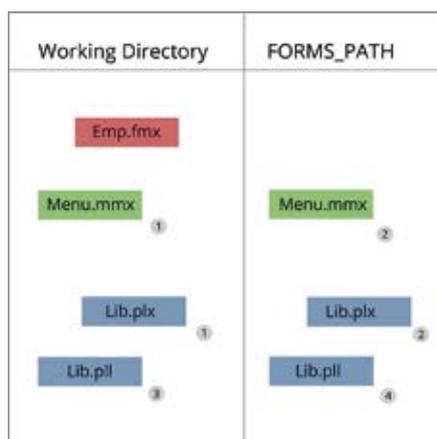


Abbildung 1: Die Such-Reihenfolge

```
PACKAGE Const IS
  HiddenPW      CONSTANT VARCHAR2 (100) := 'HiddenPW';
END Const;
```

Listing 1



Gerd Volberg

gerd.volberg@opitz-consulting.com
talk2gerd.blogspot.com