

# Boost your Forms Development with GIT and Forms API Master

**Adam Lukaszewski**  
**Veolia Umweltservice GmbH**  
**Hamburg**

## **Schlüsselworte**

Oracle Forms, Agiles Software Development, Git, Versionsverwaltung, Forms API Master, Don't Scare the Merge-Conflicts

## **Einleitung**

Mit dem aktuellen 12c Forms Release spricht sich Oracle klar für ein Fortbestehen dieser Technologie aus. Allgemein als stabiles und reifes Entwicklungswerkzeug bekannt, wird es weltweit genutzt, um Business Needs und neuerdings sogar Integrationsaufgaben systemübergreifender Prozesse abzubilden.

Aller zukunftsweisender Trends zum Trotz beklagt das Veolia Entwicklerteam, dass eine effektive, gemeinsame Zusammenarbeit auf Grundlage des Forms-Binärformats (\*.fmb) nur unzureichend möglich ist. Zu restriktiv sind die Möglichkeiten, parallel an einzelnen Masken zu arbeiten. Zu aufwändig und veraltet sind die damit verbundenen, notwendigen - teilweise organisatorischen - Maßnahmen, um hochwertige Software bereitzustellen. Ein Mehr an Agilität im Entwicklungsprozess ist das Ziel von Veolia!

Unter dem Motto „Welcome to the 21st century, dear Forms Developer!“ wird anhand von konkreten Praxisbeispielen erläutert, wie durch Rückgriff auf Git und den Forms API Master der Entwicklungsprozess wesentlich effektiver ausgerichtet werden kann. Hierbei werden nicht nur unterschiedliche Entwicklungsstände im Formskontext verglichen (git diff), sondern auch im Rahmen von release-vorbereitenden Schritten zu einer Zielversion zusammengeführt (git merge).

## **Der Use Case**

Ein agiler Entwicklungsprozess kann nur so stark sein, wie die Fähigkeiten des Teams und die vorhandenen Werkzeuge dies zulassen. Mit der Einführung von Git bei der Entwicklung eines zum ERP-System vorgelagerten Auftragsverwaltungssystems ist dem Veolia Team ein großer Schritt gelungen.

Die stark auf den Oracle Forms, PL/SQL und APEX Stack ausgerichteten Entwicklungstätigkeiten konnten mit dem Wechsel zu Git wesentlich verbessert werden. Branching, Nicht-lineare Entwicklung, dezentrale Datendateiverwaltung (Distributed Version Control Systems), Patches und viele andere Funktionen von Git schaffen die notwendigen Freiheitsgrade die der Entwicklung von Software inhärent sind!

Längst gehören die starr ausgerichteten Prozessdefinitionen, die primär durch ein inflexibles, pessimistisch-sperrendes Versionierungswerkzeug motiviert waren, der Vergangenheit an. Die Gleichung an aktiven Sprints gemeinsam innerhalb eines Branches zu entwickeln scheint mit Git auch ohne Integrationsschmerzen wieder aufzugehen.

Trotzdem bleibt ein wesentlicher Störfaktor aus dieser Gleichung herauszulösen. Die Arbeit mit Git legt ein wesentliches Umdenken bei den Entwicklern zugrunde: „Keine Angst von Merge-Konflikten!“

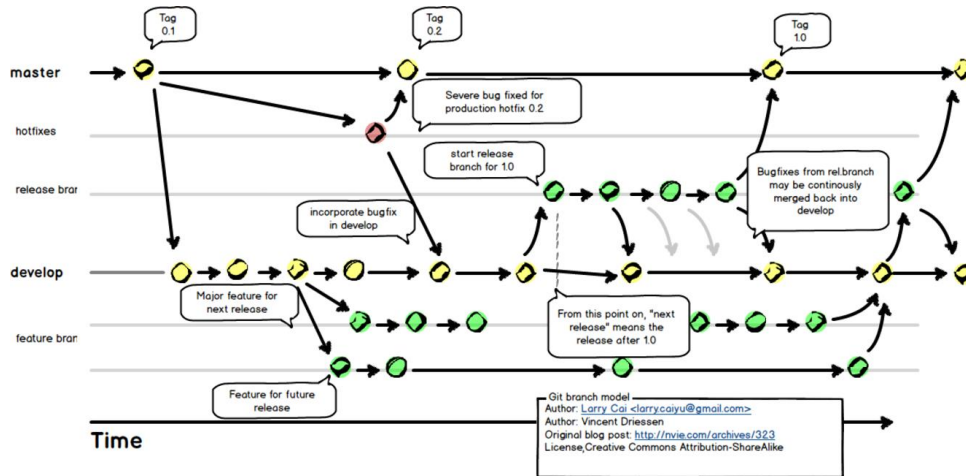


Abb. 1: The Power of Branching

Durch die dezentrale Verwaltung von Ressourcen innerhalb von Git nimmt auch die Wahrscheinlichkeit zu Konflikte beseitigen zu müssen. Was bei Plain-Text-Dateien Git mit seinen Merge-Strategien nahezu vollständig dem Entwickler abnimmt, bleibt für die Oracle Forms FMBs und Report RDFs (Binärformat) der manuelle Merge als Herausforderung weiterhin bestehen.

### Oracle Forms, Forms API Master, Git ein starkes Trio!

Diese Lücke füllt der Forms API Master mit seinen Merge und Diff Funktionalitäten perfekt aus, ohne auf die bereits angesprochenen Vorteile von Git in der Forms-Entwicklung verzichten zu müssen. Der direkte Aufruf des Merge bzw. Diff-Modus des Forms API Master aus der Kommandozeile heraus, liefert die notwendigen Befehle, um in Git nahtlos integriert werden zu können.

```
FAPIMaster.exe /COMPARE /MODULE1="c:\testme\test.fmb"  
/MODULE2="c:\testme\test2.fmb"
```

In der %USERPROFILE%/.gitconfig schafft Git einen Rahmen, der es dem Entwickler erlaubt seine favorisiertes Merge- bzw. Diff-Tools einzukonfigurieren:

```
[difftool "fapi"]  
  cmd = fapi_diff.sh $LOCAL $REMOTE  
  
[mergetool "fapi"]  
  cmd = fapi_merge.sh $BASE $LOCAL $REMOTE
```

Das Script fapi\_diff.sh und fapi\_merge.sh delegieren die beim Diff oder Merge abgeleiteten Dateiversionen an die Kommandozeile und rufen den Forms Api Masters im jeweiligen Modus auf:

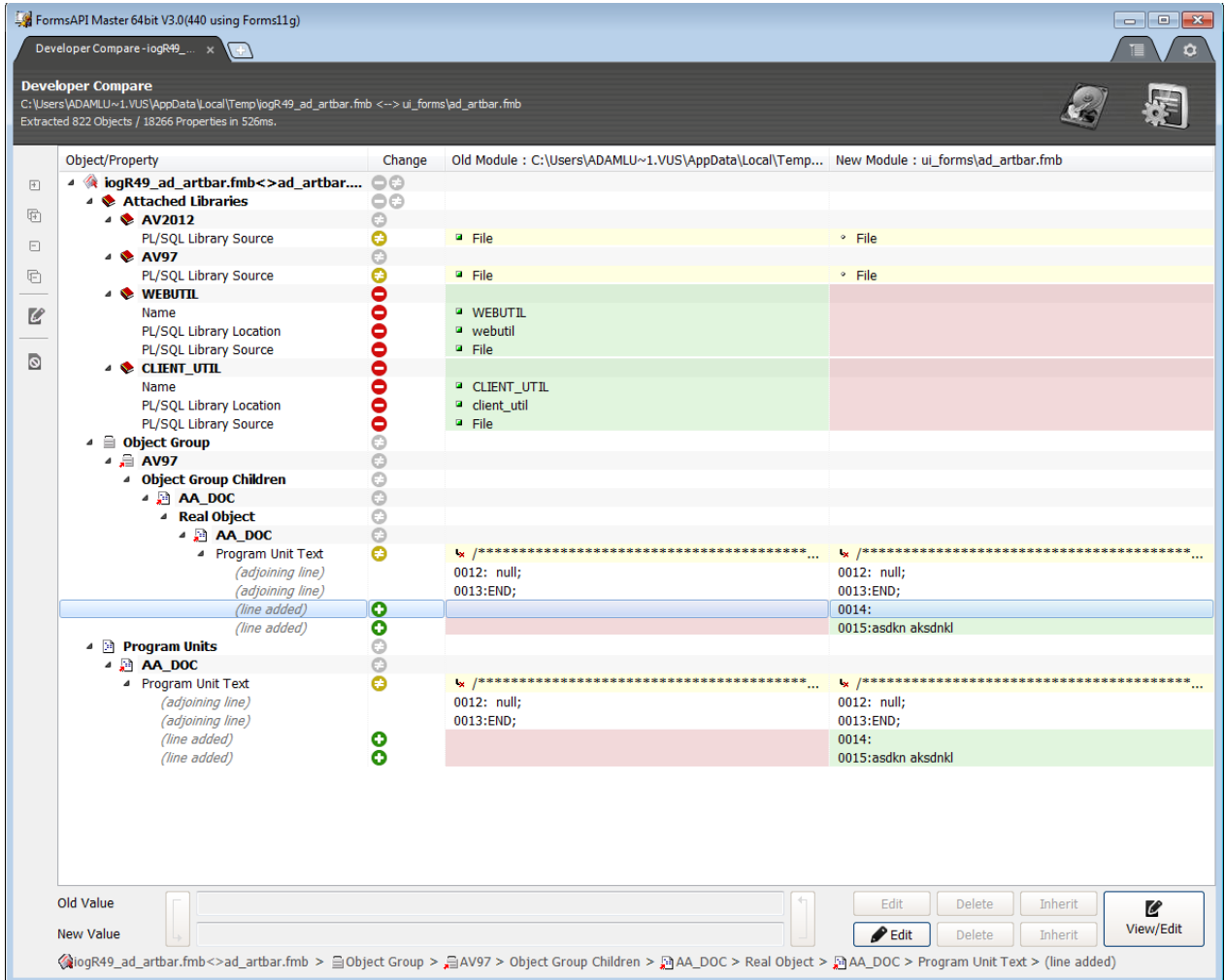
```
#!/bin/sh  
LOCAL=$(echo $1 | sed 's|/|\\|g')  
REMOTE=$(echo $2 | sed 's|/|\\|g')  
cmd.exe /C "FapiMaster.exe /COMPARE /MODULE1=$LOCAL /MODULE2=$REMOTE"  
  
#!/bin/sh  
MODULEBASE=$(echo $1 | sed 's|/|\\|g')  
MODULE1=$(echo $2 | sed 's|/|\\|g')
```

```

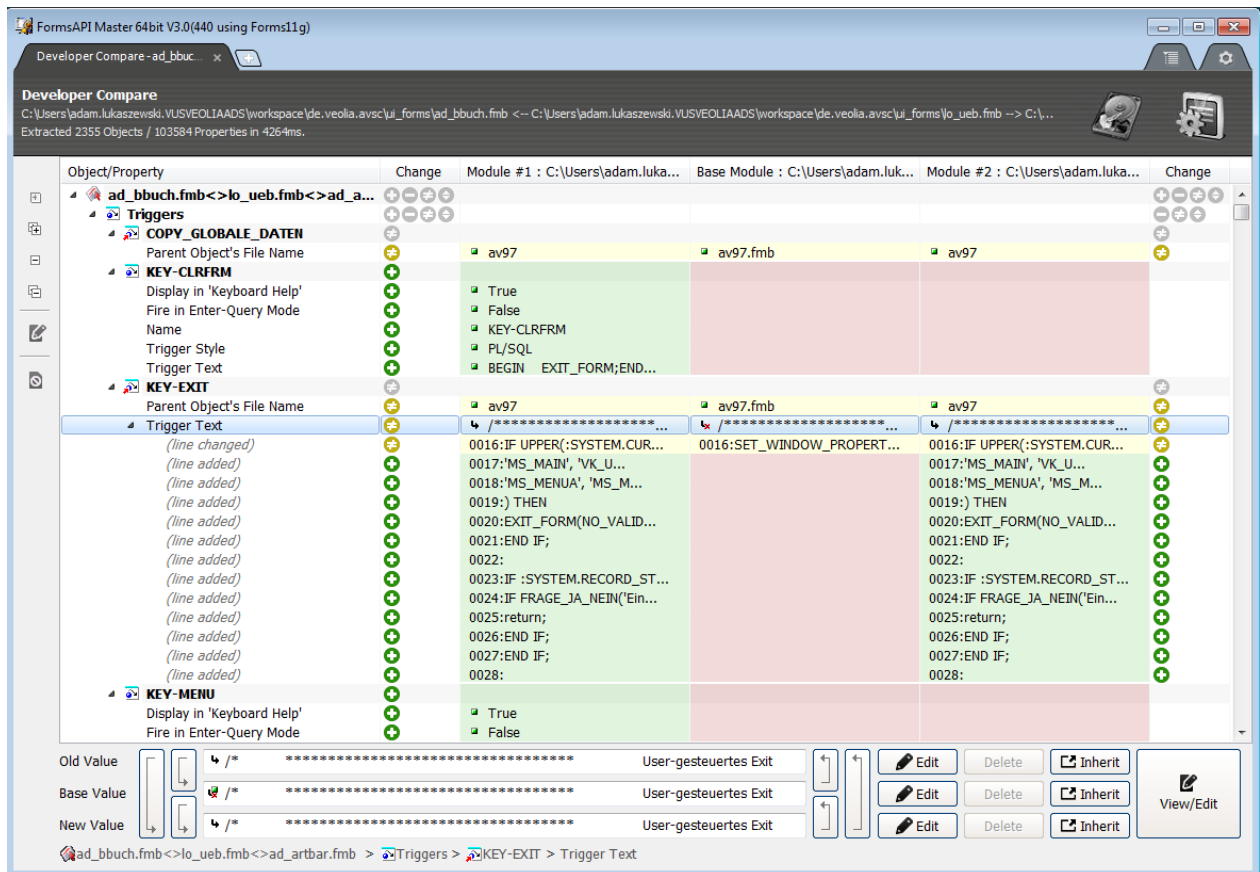
MODULE2=$(echo $3 | sed 's|/|\\|g')
cmd.exe /C "FapiMaster.exe /COMPARE /MODULEBASE=$MODULEBASE
/MODULE1=$MODULE1 /MODULE2=$MODULE2"

```

Das Ergebnis ist ein direkter Sprung aus der Kommandozeile in den Forms API Master, welcher die geänderte und originale Version in einer sauberen Gegenüberstellung präsentiert. Das nachträgliche Ändern der aktuellen Version kann direkt im Tool vorgenommen werden und ggf. innerhalb der Versionsverwaltung wieder abgelegt werden.



Wurde die gleiche Binärdatei von zwei unterschiedlichen Entwicklern angepasst, ist ein Merge erforderlich. Diesen Umstand trägt Git mit der Auswahl des eingestellten Forms API Masters als Merge-Tool Rechnung:



**Kontaktadresse:**

Name Adam Lukaszewski  
 Firma Veolia Umweltservice GmbH  
 Hammerbrookstr., 69  
 D-20097 Hamburg

Telefon: +49 (0) 40-78101 728  
 Fax: +49 (0) 40-78101 9714  
 E-Mail: adam.lukaszewski@veolia.com  
 Internet: www.creative-doing.de