## Koncepcja

Użycie liquibase w automatyczny, promowany przez twórców sposób uniemożliwia uzyskanie pełnej kontroli nad wdrażaniem zmian oraz utrzymywanie w repozytorium plików źródłowych w postaci poprawnych SQL. Tryb manualny jest zaś dość uciążliwy dla programistów. Dlatego powstało narzędzie LqPackageBuilder (auto Krzysztof Własiuk [wlasiukk@gmail.com](mailto:wlasiukk@gmail.com)) służące do automatyzacji pracy z kodem SQL/PLSQL oraz APEX.

Narzędzie jest pomyślane w taki sposób, aby aktywnie używać gałęzi w git do grupowania zmian – idealnie sprawdza się to jeśli mamy zintegrowaną Jirę z git. Kazda większa zmiana biznesowa powinna zaczynać się od utworzenia gałęzi

Wszystkie komendy uruchamiane są jaku superuser dbarchitect, domyślnym schematem jest SARS.

Ustawienia połączenia do bazy znajdują się w pliku tools\liquibase.properties

### Układ katalogów:

..\sourceDirectory\

.git\

src\

someSourceFile.sql

outputDirectory\

versionChangeLogName.xml

branchName\

branchName.xml

installDirectory\

someSourceFile.sql

rollback\

someSourceFile\_rollback.sql

Przykład dla brancz’a JIRA\_0000001:

..\repo\_lcore\_dbsrc

.git\

db\

sars\

table\

load\_log-alter-1.sql

changes\

master\_changelog.xml

v1.0.0\

version100.xml

JIRA\_0000001\

JIRA\_0000001.xml

install\

lcore\

table\

load\_log-alter-1.sql

rollback\

lcore\

table\

load\_log-alter-1\_rollback.sql

## Instalacja

1. Instalujemy git – polecam paczkę gitextentions ale może być cokolwiek innego
2. Musimy mieć zainstalowaną java w wersji 1.8 lub wyższej
3. pobieramy liquibase 3.9.0 :

<https://github.com/liquibase/liquibase/releases/download/v3.9.0/liquibase-3.9.0.zip>

Rozpakowujemy, np. do katalogu d:\tools\liquibase\liquibase-3.9.0

1. Kopiujemy plik drivera jdbc Oracle ojdbc8.jar do podkatalogu lib, np. d:\tools\liquibase\liquibase-3.9.0\lib
2. Pobieramy sqlcl

<https://www.oracle.com/pl/database/technologies/appdev/sqlcl.html>

Rozpakowujemy do katalogu, np. d:\tools\oracle\sqlcl

1. Tworzymy katalog na kod źródłowy, klonujemy tam repozytorium z BB

<https://USERNAME@bb.csioz.gov.pl/scm/ewp/ewp2.git>

Na potrzeby dokumentacji będziemy nazywać ten katalog c:\work\project1

1. Kopiujemy plik „c:\work\project1\tools\setenv\_sample.bat” do pliku “setenv.bat” w tym samym katalogu. Modyfikujemy zawartość tak, aby był dostosowany do nasze środowiska – zaznaczone na czerwono części pliku.

|  |
| --- |
| @echo off  REM adding liquibase to path  set np=d:\tools\liquibase\liquibase-3.9.0  echo %PATH%|find /i "%np%">nul || set PATH=%PATH%;%np%  REM adding oracle sqlcl to path  set sp=d:\tools\oracle\sqlcl\bin  echo %PATH%|find /i "%sp%">nul || set PATH=%PATH%;%sp%  REM location of LqPackageBuilder.jar  set lqbuilder=LqPackageBuilder.jar  REM directory where out project is cloned from git repository  set sourceDirectory= c:\work\project1\ewp-repo  REM directory inside sourceDirectory where changesets will be saved  set outputDirectory=changes |

1. Testujemy działanie liquibase uruchamiając skrypt status.bat z katalogu tools – połączenie zdalne trochę trwa …

|  |
| --- |
| d:\priv\csioz\git\EWP-repo\ewp-repo\tools>status.bat  Liquibase Community 3.9.0 by Datical  DBARCHITECT@jdbc:oracle:thin:@//172.16.30.11:1521/SARSDEV is up to date  Liquibase command 'status' was executed successfully.  Press any key to continue . . . |

## Skrypty

W katalogu tools znajdziemy kilka przydatnych skryptów ułatwiających pracę:

|  |  |
| --- | --- |
| liquibase.properties | Parametry do połączenia się z bazą, nazwą changeLogFile, classpath |
| lq.bat | Skrót do liquibase – używa parametrów z setenv.bat |
| lqb.bat | Skrót do LqPackageBuilder – bez parametrów pokaże help |
| lqb\_all.bat | Automatyczne dodanie wszystkich zmian z git – używać jeśli mamy wyczyszczoną listę zmian – szczególnie z plików których nie chcemy uruchamiać (można je zakommitowac wcześniej) |
| lqb\_added.bat | Automatyczne dodanie wszystkich zmian z git ze statusem „A” (nowe dodane pliki) – używać jeśli mamy wyczyszczoną listę zmian – szczególnie z plików których nie chcemy uruchamiać (można je zakommitowac wcześniej) |
| lqb\_changes\_to\_be\_committed.bat | Automatyczne dodanie wszystkich zmian z git ze statusem „new file” – używać jeśli mamy inne zmiany niedokończone – ale nowe pliki gotowe są w „staged” |
| lqb\_onefile.bat | Dodanie zmiany pojedynczego pliku – ścieżka musi być relatywna do katalogu użytego w zmiennej sourceDirectory  Np.:  Katalog główny (sourceDirectory)  c:\work\project1\ewp-repo  Pełna ścieżka do pliku  c:\work\project1\ewp-repo\db\sars\tables\nowa\_tablica.tab  Wywołanie dodania pliku  lqb\_onefile.bat db\sars\tables\nowa\_tablica.tab |
| setenv.bat | Ustawienia lokalne programisty – ten plik nie powinien być po kontrolą git (znajduje się w .gitignore) |
| setenv\_sample.bat | Ustawienia lokalne programisty – plik przykładowy |
| status.bat | Liquibase status – oddaje zmiany nie wgrane jeszcze na bazę |
| update.bat | Liquibase update – wgrywa zmiany nie wgrane jeszcze na bazę |

## Użycie LqPackageBuilder

### Dodanie nowej zmiany w trybie automatycznym (lqb\_all.bat)

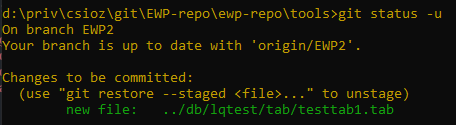
Wszystkie zmienione pliki, nawet te jeszcze nie dodane do git, będą dodane do zmian.

### Dodanie nowej zmiany w trybie manualnym (lqb\_onefile.bat)

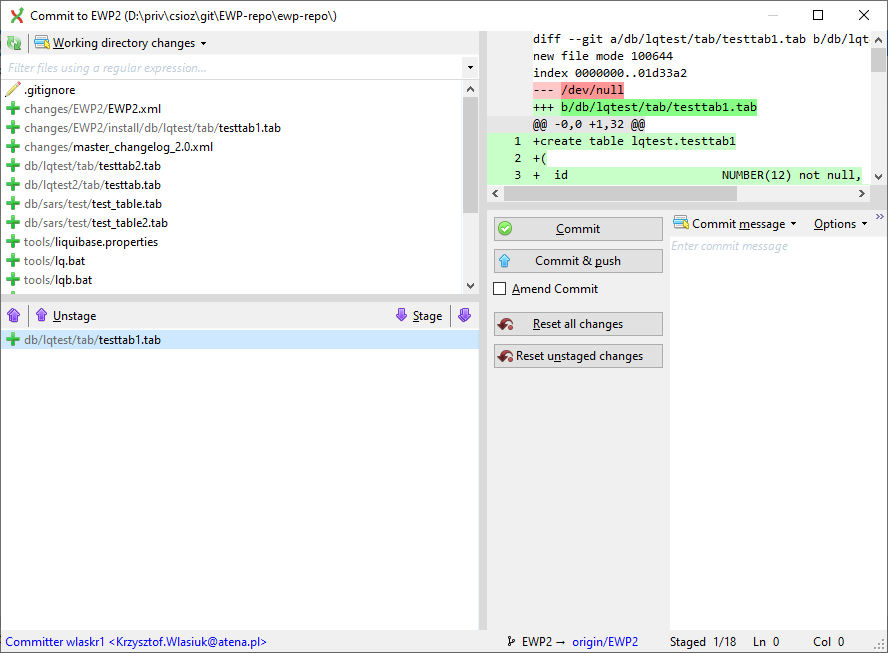
Opcja przydatna jeśli chcemy mieć pełną kontrolę nad kolejnością dodawania plików

### Dodanie nowej zmiany w trybie automatycznym – tylko pliki „new file”

Dodane zostaną tylko nowe pliki dodane komendą „git add …”



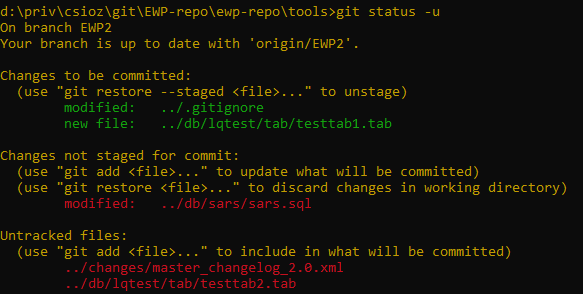
W GitExtentions taki plik będzie z zielonym plusem w sekcji Stage:

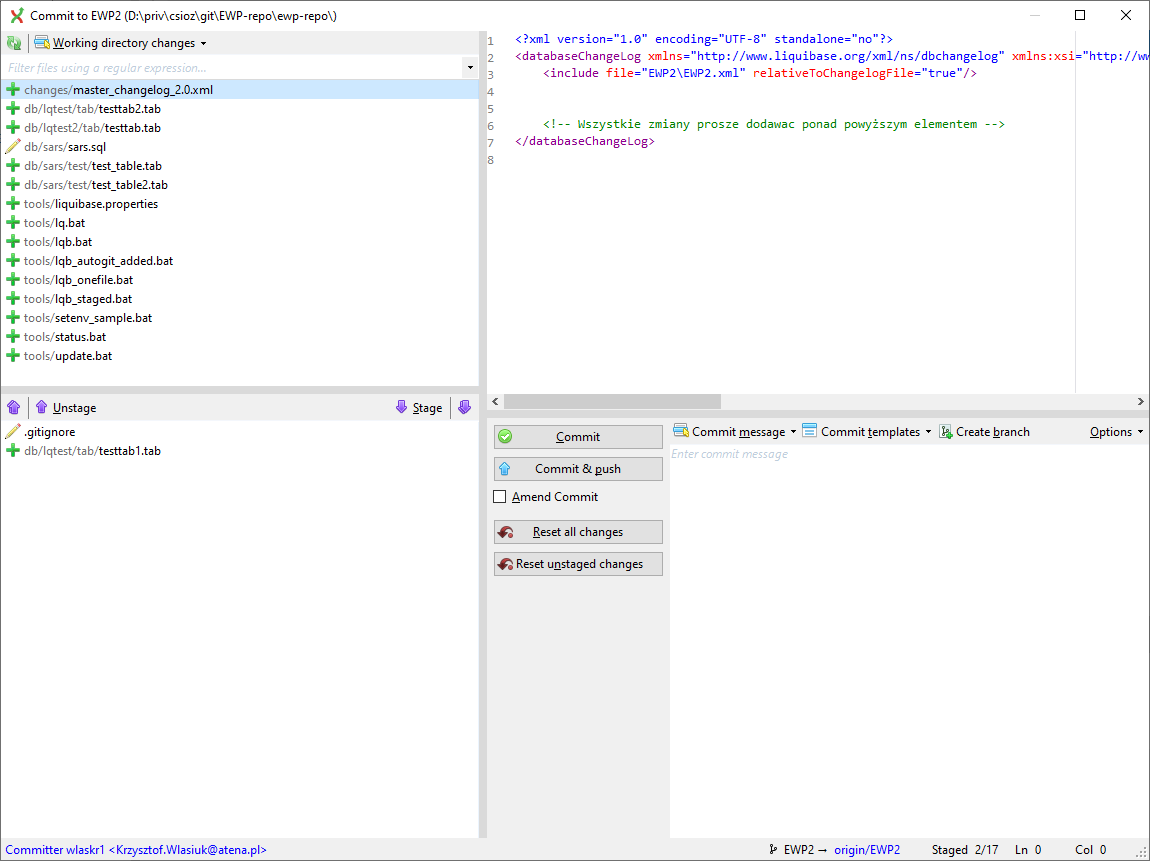


Uruchomienie skryptu lqb\_autogit\_added.bat spowoduje dodanie wszystkich plików ze statusem „new file”

|  |
| --- |
| d:\priv\csioz\git\EWP-repo\ewp-repo\tools>lqb\_autogit\_added.bat  2020-05-30 23:14:17 INFO pl.wlasiukk.lqpackagebuilder.BuilderContext <init> package name from branch name will be : EWP2  2020-05-30 23:14:18 INFO pl.wlasiukk.lqpackagebuilder.LqPackageBuilder buildPackage processing db/lqtest/tab/testtab1.tab  Press any key to continue . . . |

### Dodanie wszystkich zmian w stage (lqb\_onefile.bat)





|  |
| --- |
| d:\priv\csioz\git\EWP-repo\ewp-repo\tools>lqb\_staged  2020-05-30 23:28:05 INFO pl.wlasiukk.lqpackagebuilder.BuilderContext <init> package name from branch name will be : EWP2  2020-05-30 23:28:06 INFO pl.wlasiukk.lqpackagebuilder.LqPackageBuilder buildPackage processing .gitignore  2020-05-30 23:28:06 INFO pl.wlasiukk.lqpackagebuilder.LqPackageBuilder buildPackage processing db/lqtest/tab/testtab1.tab  Press any key to continue . . . |

### Modyfikacja już istniejącego pliku

Przy ponownym uruchomieniu, plik zostanie nadpisany w swoim pierwszym changeset w obecnej gałęzi. Dlatego modyfikacja już istniejącej tablicy powinna być osobnym plikiem z „alter table”.

Należy pamiętać, że LqPackageBuilder promuje pracę na gałęziach git. Jeśli do każdej grupy zmian będziemy stosować nowy gałąź, to np. modyfikacja istniejącego pakietu odłoży nowy plik w katalogu changes. Jeśli po jakimś czasie musimy dodać nowszą wersję pakietu w tej samej gałęzi należy użyć komendy „lqb.bat -f \_plik\_ -n” – wtedy zostanie dodana nowa zmiana z obecną wersją pakietu.

Jeśli modyfikujemy już istniejący pakiet w nowej gałęzi to zostanie odłożona nowa zmiana, więc nie musimy używać opcji „-n”.

### Renumeracja zmian

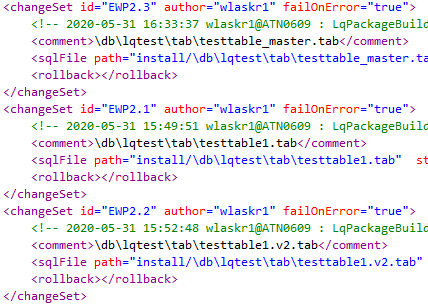
Jeśli dodaliśmy wiele plików (np. poprzez uruchomienie lqb\_all.bat) i chcemy zmienić ich kolejność uruchamiania (np. indeksy po tablicach) to możemy zrobić to ręcznie w pliku zmian naszej gałęzi. Niestety jest to uciążliwy proces ze względu na to że każda zmiana ma własny id składający się z nazwy gałęzi oraz kolejnego numeru – np. JIRA123.1 , JIRA123.2 itp.

Rozwiązaniem jest opcja renumeracji „-r” która poprawi w identyfikatorach zmian ich numery.

Oryginał :



Po ręcznej zmianie kolejności (przesuwamy całą zmianę EWP2.3 na pierwsze miejsce) mamy zmianę "EWP2.3" przed "EWP2.1":



Po uruchomieniu renumeracji (lqb.bat -r) testtable\_master ma id z numerem 1 :



### Usunięcie pliku

Jeśli zmiana została już wgrana na bazę inną niż DEV polecamy dodanie zmiany wycofującej jej.

Jeśli zmiana jeszcze nie została uruchomiona to możemy usunąć całą zmianę – „<changeSet> …</changeSet>” z pliku zmian xml.

Jeśli zmiana została już wgrana tylko na naszym środowisku DEV i możemy ją ręcznie wycofać (np. usunąć tablicę) , to należy usunąć całą zmianę z pliku zmian xml oraz usunąć 1 wiersz z tablicy dbarchitect.databasechangelog.

### Własny plik rollback

Domyślnie LqPackageBuilder tworzy plik rollback na podstawie wcześniejszej wersji pliku w repozytorium. Jeśli plik jest nowy (tutaj testtab1.tab) rollback będzie pusty.



Takie podejście świetnie sprawdza się w przypadku obiektów udostępniających składnię „OR REPLACE” - pakietów, procedur, funkcji, widoków itp.

W przypadku tablic i indeksów należy pisać własny skrypt rollback – np. drop table, drop index itp.

Można także przyjąć strategię "fix forward", nie pisać ręcznie plików rollback i przy potrzebie wycofania zmian wygenerować nowe zmiany usuwające poprzednie zmiany. Administrator wdrażający zmiany musi wiedzieć o przyjęci takiej strategii. Skrypty CI/CD także muszą nie używać funkcjonalności rollback wbudowanej w liquibase.

### Dodanie pliku w schemacie innym niż głównym (tutaj SARS)

Niestety liquibase nie wspiera wielu połączeń do baz lub specyfikacji schematu w tagu sqlFile.

Jeśli chcemy wdrażać zmiany z innych schematów to trzeba używać w nich pełnej ścieżki do obiektu zawierającej schemat, np.:

Create table inny\_schemat.nowa\_tablica …

Inne zmiany nie są wymagane.

## Aplikacje APEX

Jeśli w jakimkolwiek katalogu nadrzędnym skryptu znaleziony zostanie plik apex\_source LqPackageBuilder będzie traktował wszystkie podkatalogi jako źródła aplikacji APEX.

Skutkuje to użyciem dla skryptów specjalnej transformaty umożliwiającej podmianę pojedynczych elementów aplikacji APEX jak strona, element nawigacyjny itp.

#### Eksport aplikacji

Do eksportowania zmian w aplikacji należy się posłużyć skryptem export\_apex\_app\_107.bat

Żeby działał poprawnie wymaga ORACLE sqlcl dostępnego w ścieżce – patrz tworzenie pliku setenv.bat w rozdziale instalacja