

GitのBundleファイル運用

Gitの使い方としてローカルのリポジトリとリモートの共有リポジトリという構成が基本だろう。だが、オンラインでのリポジトリアクセスができない場合がある。こういう時にBundleファイルを用いてリポジトリ間でデータ同期が可能である。BundleファイルはGitコマンドで作成したり読み込みができるのだが、何度も行う作業であればスクリプト化してしまうべきであろう。これらのバッチ/スクリプトをリポジトリ直下に置いて起動すれば、バンドルファイルの作成や読み込みがメニュー選択するだけで実行可能になる。



これらのスクリプトでバンドルファイルを読み込むとimportブランチに反映される。確認して問題が無ければmasterブランチにマージする。

- 2025/06/01 Bundleファイルの対象とするコミットを選択する機能を追加

バッチ版

Git-Bundle.bat

```
@echo off
set BUNDLE=Git-RepoExport.bundle
set ARCHIVE_ZIP=Git-Archive.zip
set ARCHIVE_TGZ=Git-Archive.tgz
set REFNAME=refs/heads/master
set COMMIT=HEAD
:MENU
cd %~dp0
echo.
echo *** MENU ***
echo 1: Verify Git bundle file
echo 2: Fetch Git bundle file
echo 3: Create Git bundle file
echo L: List of commits
echo S: Select a commit
echo T: Create Git tgz-archive file
echo Z: Create Git zip-archive file
echo q: Quit
set /P INPUT="%COMMIT%>>> "
echo.
set SELECT=%INPUT:~0,1%
if "%SELECT%"=="1" goto SELECT_1
if "%SELECT%"=="2" goto SELECT_2
if "%SELECT%"=="3" goto SELECT_3
if "%SELECT%"=="L" goto SELECT_L
if "%SELECT%"=="S" goto SELECT_S
if "%SELECT%"=="T" goto SELECT_T
if "%SELECT%"=="Z" goto SELECT_Z
if "%SELECT%"=="q" goto eof
goto MENU
```

```
:SELECT_1
echo [ Verify Git bundle file ]
git bundle verify %BUNDLE%
goto MENU
:SELECT_2
echo [ Fetch Git bundle file ]
git fetch %BUNDLE% master:import
goto MENU
:SELECT_3
echo [ Create Git bundle file ]
if "%COMMIT%"=="HEAD" (
    git bundle create %BUNDLE% HEAD %REFNAME%
) else (
    git bundle create %BUNDLE% %REFNAME% ^^%COMMIT%
)
goto MENU
:SELECT_L
git log --oneline
goto MENU
:SELECT_S
echo [ Select a commit ]
set /P COMMIT="%COMMIT% -> "
goto MENU
:SELECT_T
echo [ Create Git tgz-archive file ]
git archive HEAD --format=tgz -o %ARCHIVE_TGZ%
goto MENU
:SELECT_Z
echo [ Create Git zip-archive file ]
git archive HEAD --format=zip -o %ARCHIVE_ZIP%
goto MENU
:eof
```

shell版

[Git-Bundle.sh](#)

```
#!/bin/sh
#
BUNDLE=Git-RepoExport.bundle
ARCHIVE_ZIP=Git-Archive.zip
ARCHIVE_TGZ=Git-Archive.tgz
REFNAME=refs/heads/master
COMMIT=HEAD
#
# Function
menu_list() {
```

```
n=1
cat << END | while read text
*** Menu ***
1) Verify Git bundle file
2) Fetch Git bundle file
3) Create Git bundle file
L) List of commits
S) Select a commit
T) Create Git tgz-archive file
Z) Create Git zip-archive file
q) Quit
END
do
    echo "$text"
    n=`expr $n + 1`
done
return
}
menu_list
echo -n "${COMMIT}>>> "
while read input
do
    case $input in
        1)
            echo "[ Verify Git bundle file ]"
            git bundle verify $BUNDLE
            ;;
        2)
            echo "[ Fetch Git bundle file ]"
            git fetch $BUNDLE ${REFNAME}:import
            ;;
        3)
            echo "[ Create Git bundle file ]"
            if [ "${COMMIT}" = "HEAD" ]
            then
                git bundle create $BUNDLE HEAD ${REFNAME}
            else
                git bundle create $BUNDLE ${REFNAME} ^${COMMIT}
            fi
            ls -ld $BUNDLE
            ;;
        S)
            echo "[ Select a commit ]"
            echo -n "${COMMIT} --> "
            read COMMIT
            test -z "${COMMIT}" && COMMIT=HEAD
            ;;
        L)
            echo "[ List of commits ]"
            git log --oneline
            ;;
    esac
done
```

```
T)
  echo "[ Create Git tgz-archive file ]"
  git archive HEAD --format=tgz -o $ARCHIVE_TGZ
  ls -ld $ARCHIVE_TGZ
  ;;
Z)
  echo "[ Create Git zip-archive file ]"
  git archive HEAD --format=zip -o $ARCHIVE_ZIP
  ls -ld $ARCHIVE_ZIP
  ;;
q)
  echo "Bye!"
  break
  ;;
*)
  echo "?"
  ;;
esac
echo
menu_list
echo -n "${COMMIT}>>> "
done
```

From: <https://kijima.mydns.jp/dokuwiki/> - Kijima's private wiki

Permanent link: https://kijima.mydns.jp/dokuwiki/doku.php?id=bundle_file%E9%81%8B%E7%94%A8&rev=1748739363

Last update: 2025/06/01 00:56

