

バージョン管理システム

利用手順書

2015 年 10 月 19 日



改 訂 履 歴

日 付	改訂者	改訂	改 訂 内 容
2014/04/14	野村勉	—	初版
2015/10/16	イエシ	1	Git サーバの使用に応じて、内容を追加
2015/10/19	イエシ	2	チームワークフローのブランチ戦略を追加

承 認	確 認	作 成
		

バージョン管理システム利用手順書

目次

I.	環境インストール.....	4
1.	Git のインストール	4
2.	TortoiseGit のインストール.....	9
3.	GoogleDrive のインストール.....	11
4.	他のインストール	12
II.	初期設定	13
1.	TortoiseGit の設定	13
2.	GoogleDrive の設定	15
III.	基本の操作.....	17
1.	操作とファイル状態の概要	17
2.	リモートレポジトリが既に存在している場合に開始する	18
3.	リモートレポジトリがない場合に開始する	19
4.	リモートレポジトリが不要な場合に開始する	20
5.	ファイルを管理対象にする(Add).....	21
6.	ファイル変更を記録する(Commit).....	23
7.	ログの表示.....	24
8.	リモートのレポジトリに変更を送る(Push).....	25
9.	リモートのレポジトリの変更を更新(Pull)	27
10.	ブランチを作成.....	28
11.	タグを作成	30
IV.	共有について	31
1.	すぐに LAN の PC に Git のデータベースを共有する(Daemon).....	31
2.	Skype でローカルの Git データベースを共有する	33
V.	その他の操作.....	34
1.	リモートから他のブランチをローカルに同期する(Fetch).....	34
2.	スタッシュへ変更を隠すのはどの場合に適用する？(Stash save).....	35
3.	GoogleDrive とリモートサーバ(VPN 経由)を一緒に使う	38
VI.	ブランチ戦略について.....	41
VII.	注意項目	43
1.	ファイル変更の無視(Ignore)について	43
2.	GoogleDrive をリモートレポジトリとして使う場合の注意	45
VIII.	その他の操作（参考用 URL）	45

I. 環境インストール

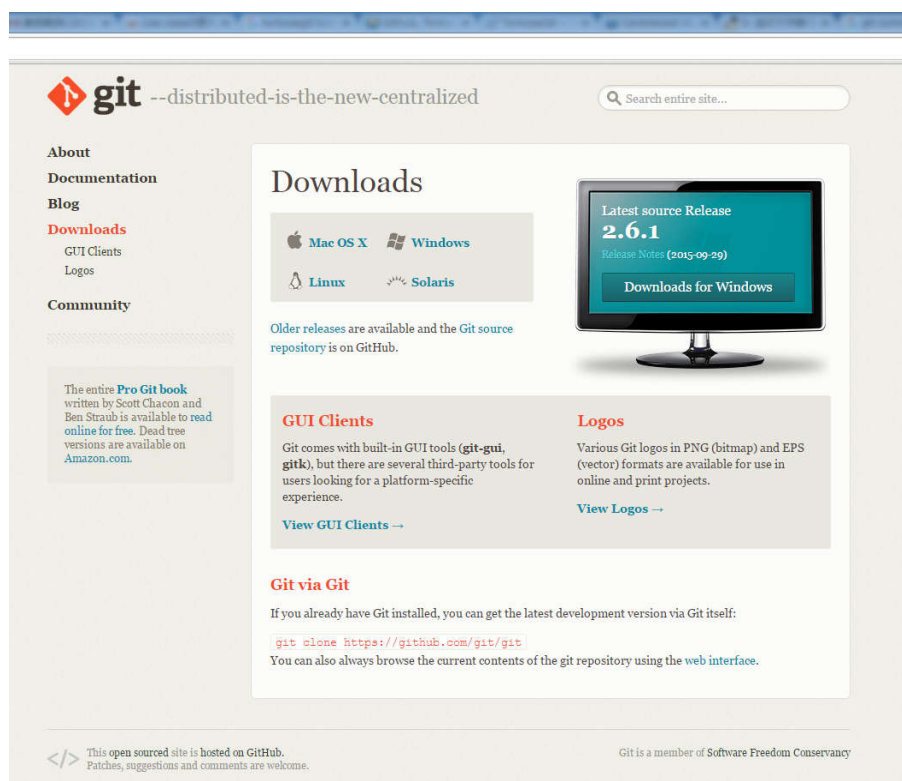
1. Git のインストール

※ インストールする PC がインターネットに接続できる環境にしておくこと

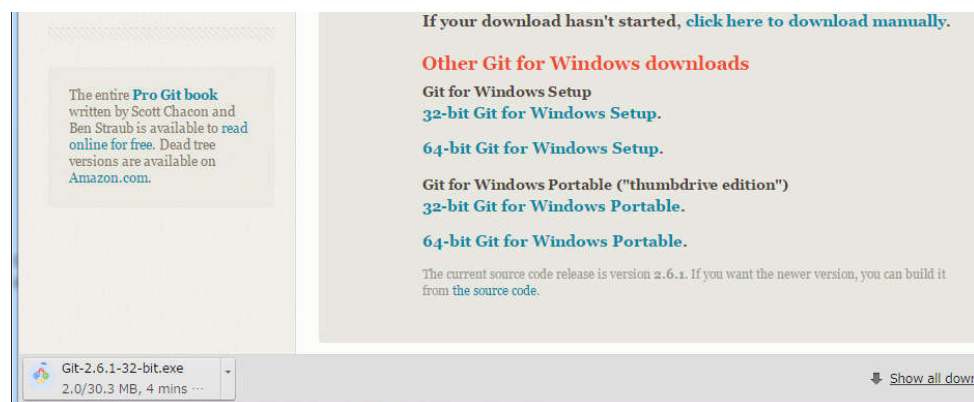
① ダウンロードサイトにアクセスする

<https://git-scm.com/downloads>

② 下記画面の「Download for Windows」をクリック



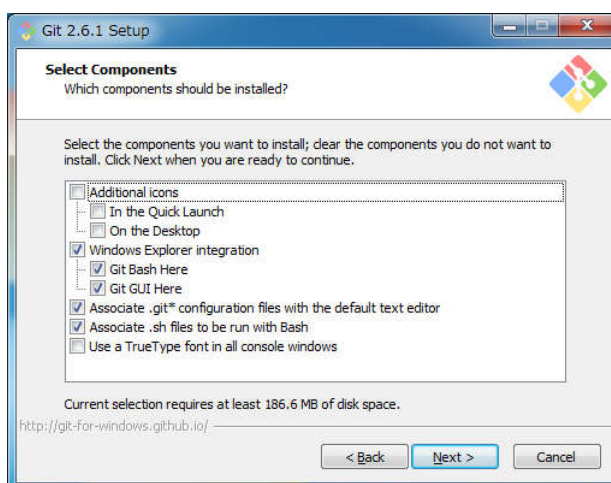
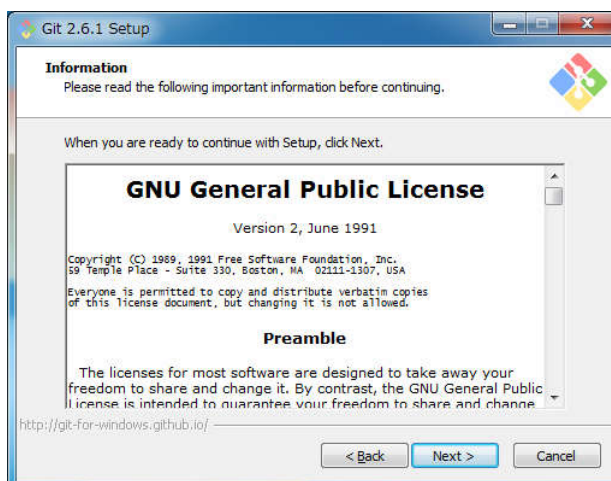
OS の bit 仕様にあった方を自動的にダウンロードします。

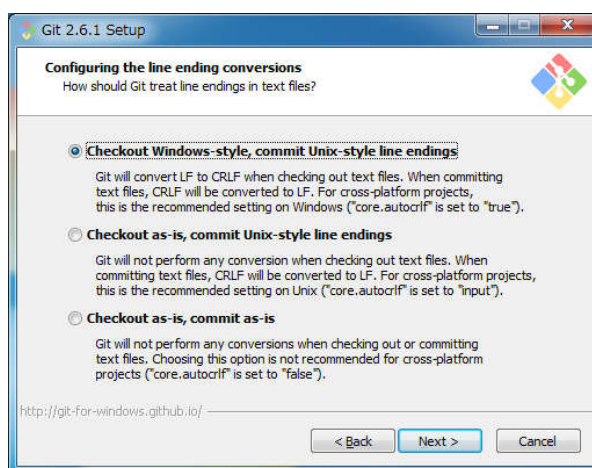
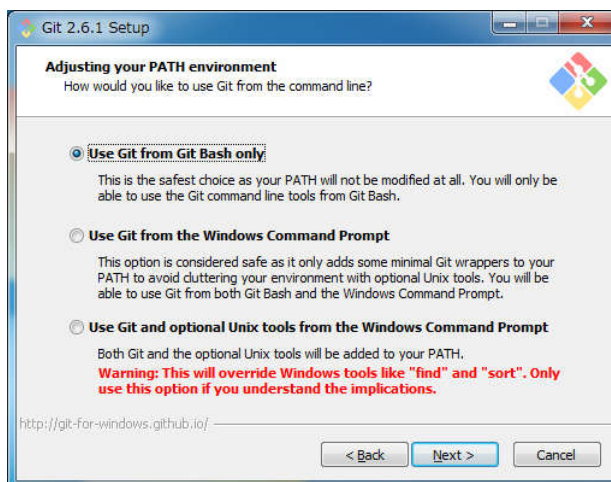


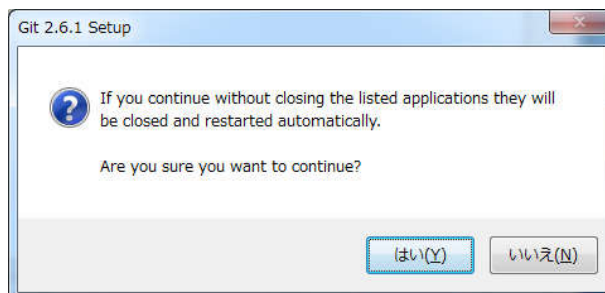
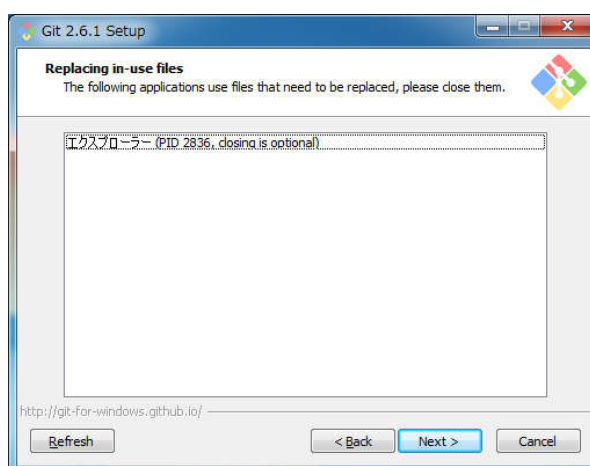
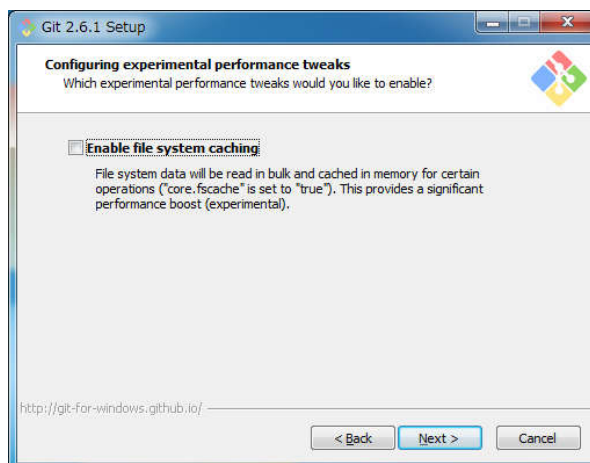
③ ダウンロードしたプログラムインストーラを実行する

2015 年 10 月 12 日の時点では 2.6.1 バージョンになります。

④ 以下の画面どおりにインストールを行ってください。

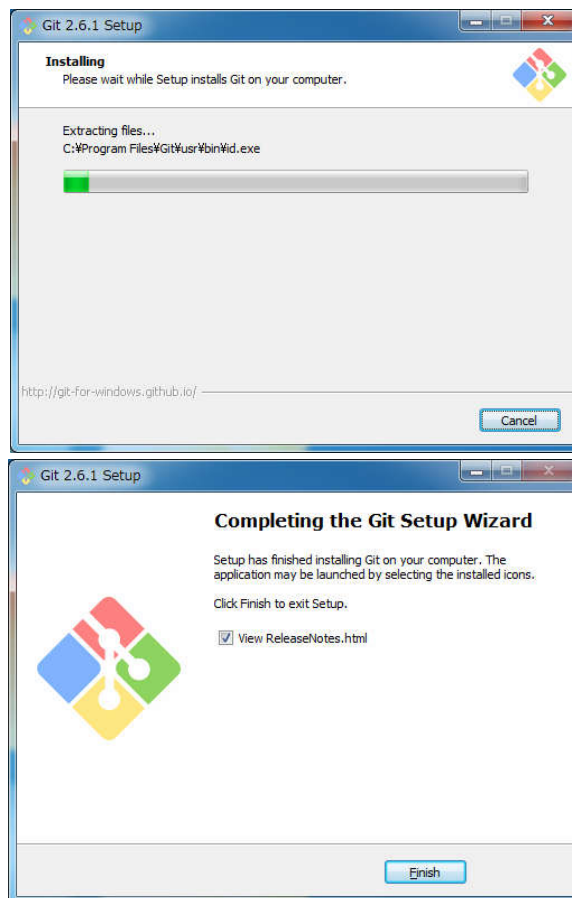






「はい」を選択してください。

⑤ インストール完了を待つ



2. TortoiseGit のインストール

① ダウンロードサイトにアクセスする

<https://tortoisegit.org/download/>

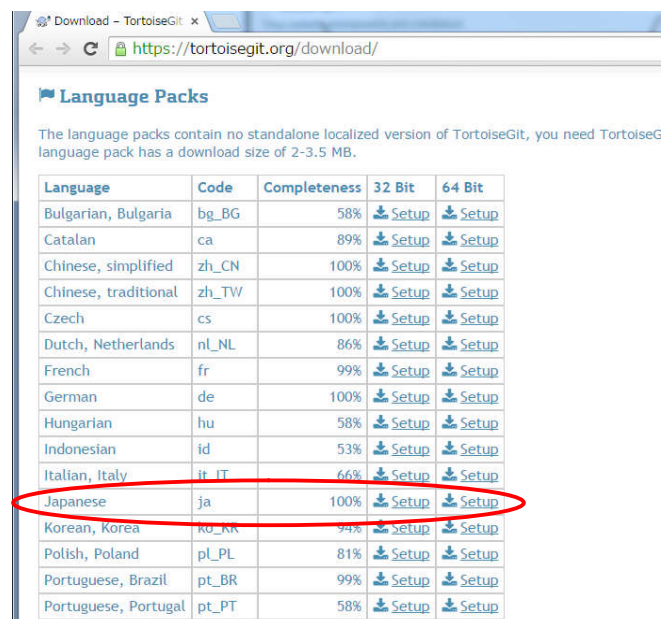
② OS の bit 仕様に合った方をダウンロードする



③ Language Pack をダウンロードする

前述の画面の下部にあります。

OS の bit 仕様に合った日本語用 Language Pack をダウンロードします。

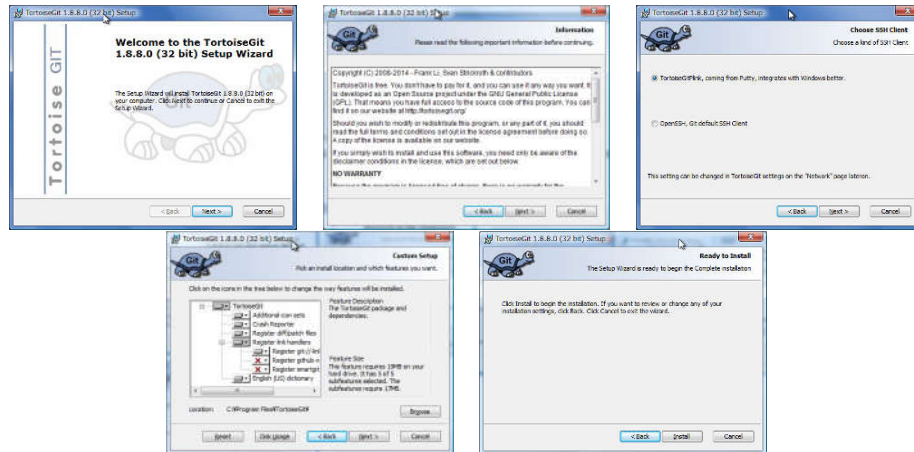


④ ダウンロードしたプログラムインストーラを実行する

2015 年 10 月時点では、「TortoiseGit-1.8.15.0-32bit.msi」というファイル名です

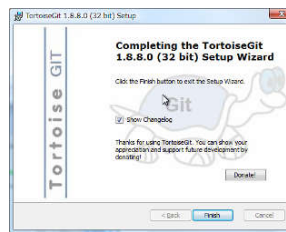
⑤ 下記画面において、「Next」をクリック。

最後の画面では、「Install」をクリック。



⑥ インストール完了を待つ

下記画面を表示したら終了です。「Finish」をクリックします。

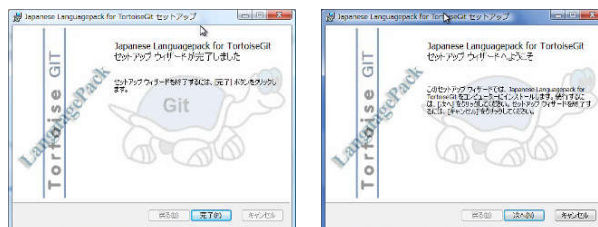


⑦ ダウンロードした日本語パックインストーラを実行する

2015 年 10 月時点では、バージョン 1.8.15.0 となっています。

⑧ 下記画面において、「次へ」をクリック。

最後の画面では、「完了」をクリック。



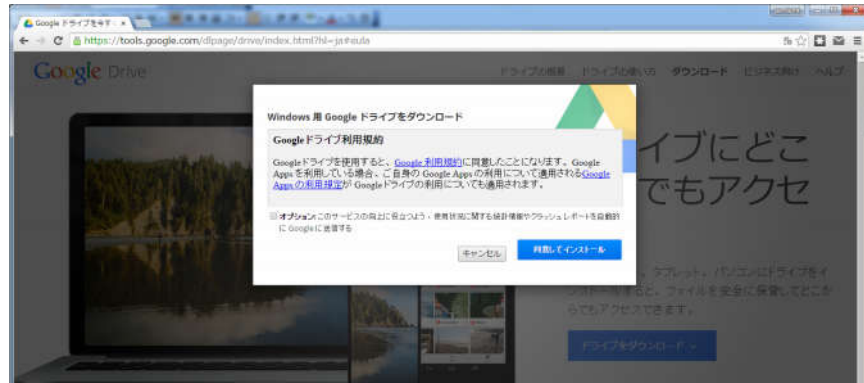
3. GoogleDrive のインストール

※ Google のアカウントは予め作成のこと

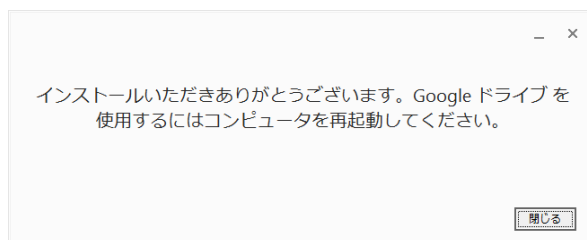
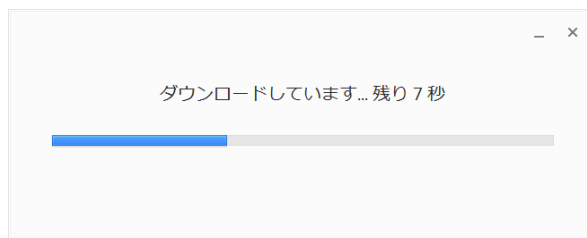
① ダウンロードサイトにアクセスする

<https://www.google.com/intl/ja/drive/download>

② パソコン版をダウンロード



③ インストーラを実行します



④ パソコンを再起動します。

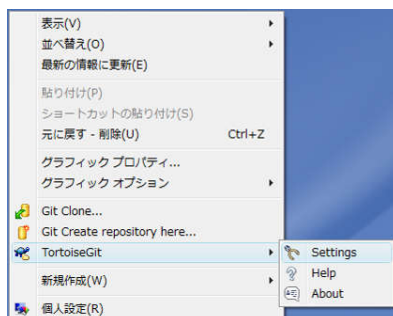
4. 他のインストール

- ① Windows 用比較・マージツールの WinMerge のインストールはおすすめです。
(<http://winmerge.org/downloads/?lang=ja>)

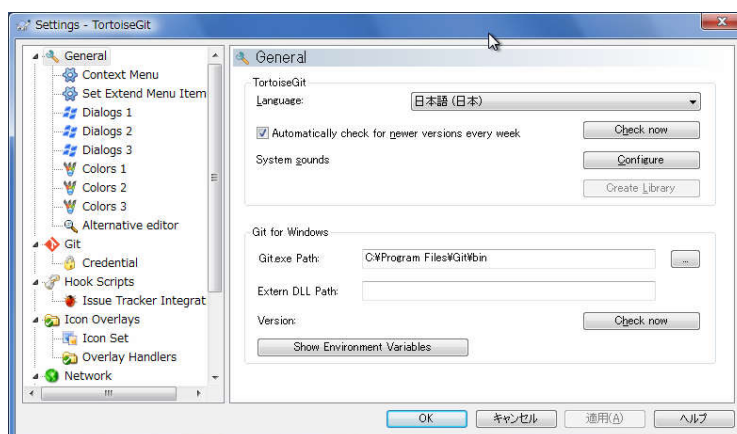
II. 初期設定

1. TortoiseGit の設定

① デスクトップで右クリック

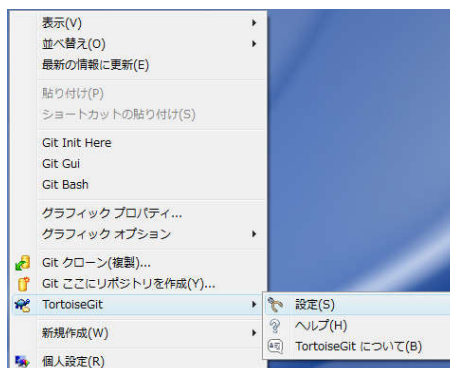


② 言語設定を日本語に変え、「OK」をクリックする

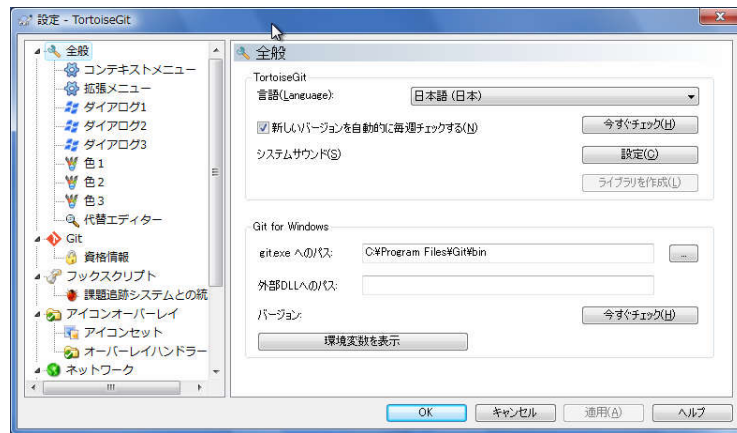


③ 再びデスクトップで右クリック

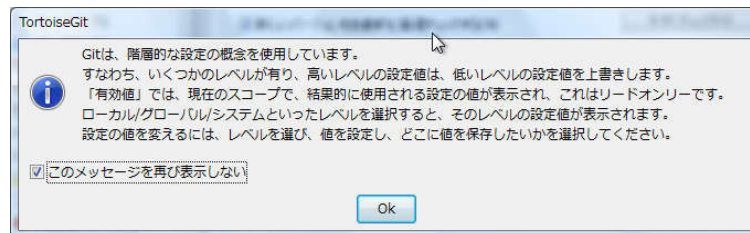
「TortoiseGit」の中の「設定」をクリックする



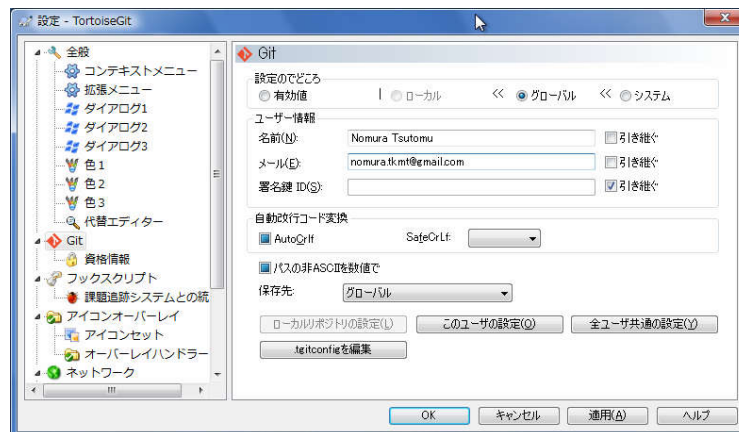
④ 「Git」をクリックする



⑤ 下記ウィンドウが表示されたら、チェックボックスにチェックを入れて「OK」をクリックする

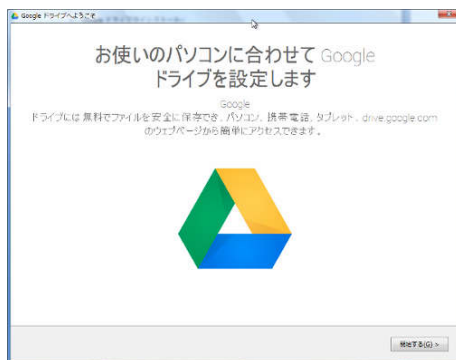


⑥ 名前とメールアドレス（G メールアカウント）を設定し、OK をクリックする



2. GoogleDrive の設定

- ① GoogleDrive を起動します。
- ② 下記画面を表示したら、右下の「開始する」をクリック



- ③ Google のメールアドレスとパスワードを入力して、自分のアカウントにログインする



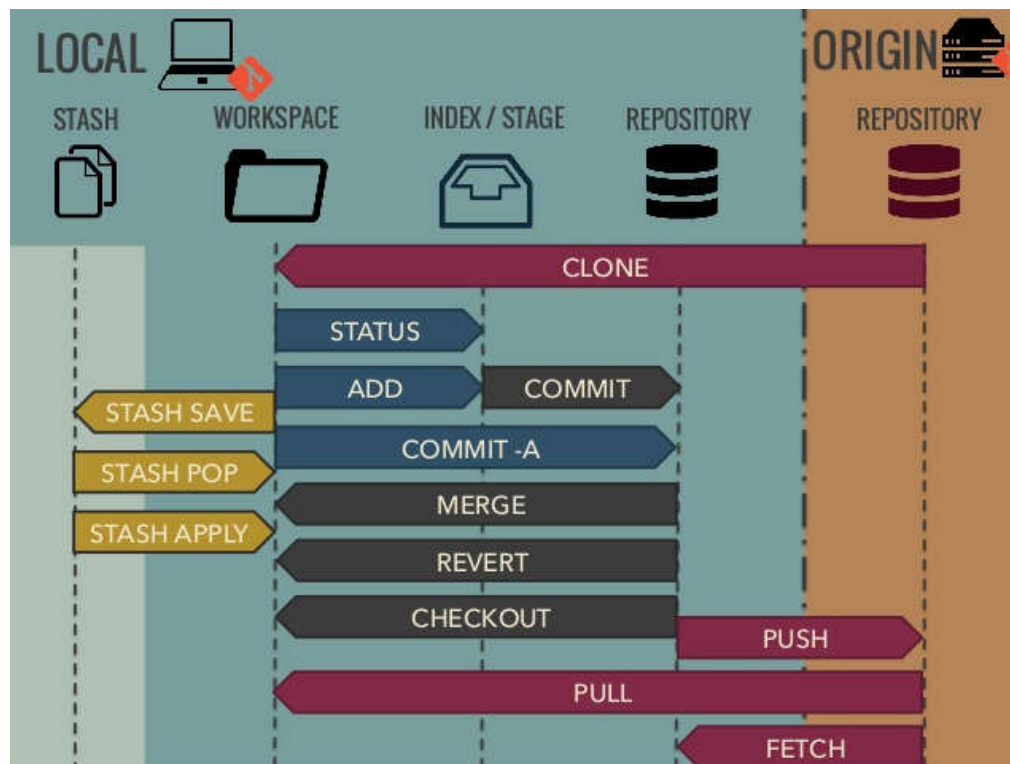
- ④ 下記画面を表示したら、画面右下の「次へ」をクリックする



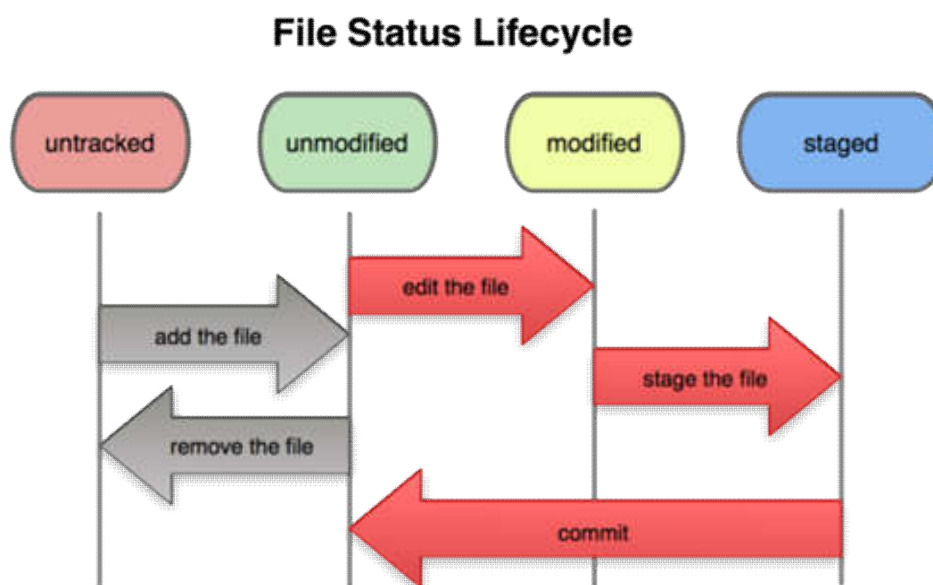
III. 基本の操作

1. 操作とファイル状態の概要

① Git の操作

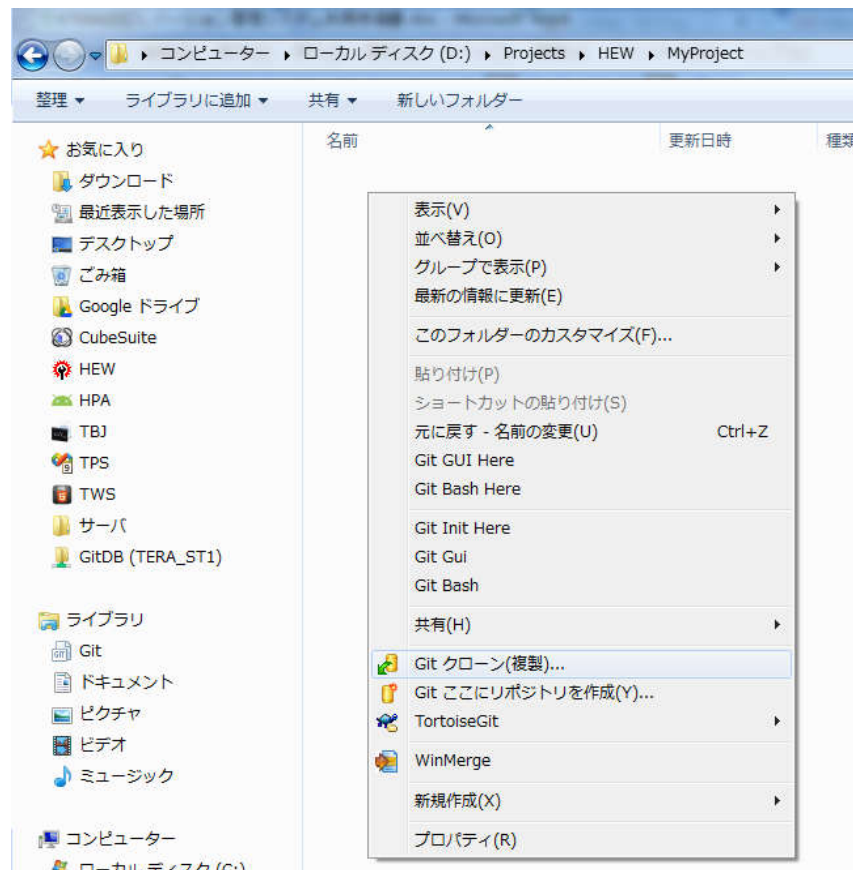


② ファイル状態

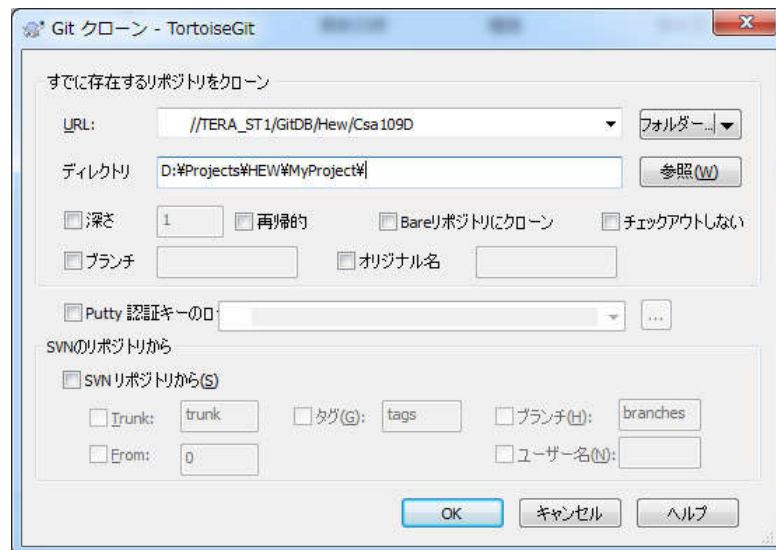


2. リモートレポジトリが既に存在している場合に開始する

① プロジェクトの場所で右クリックする



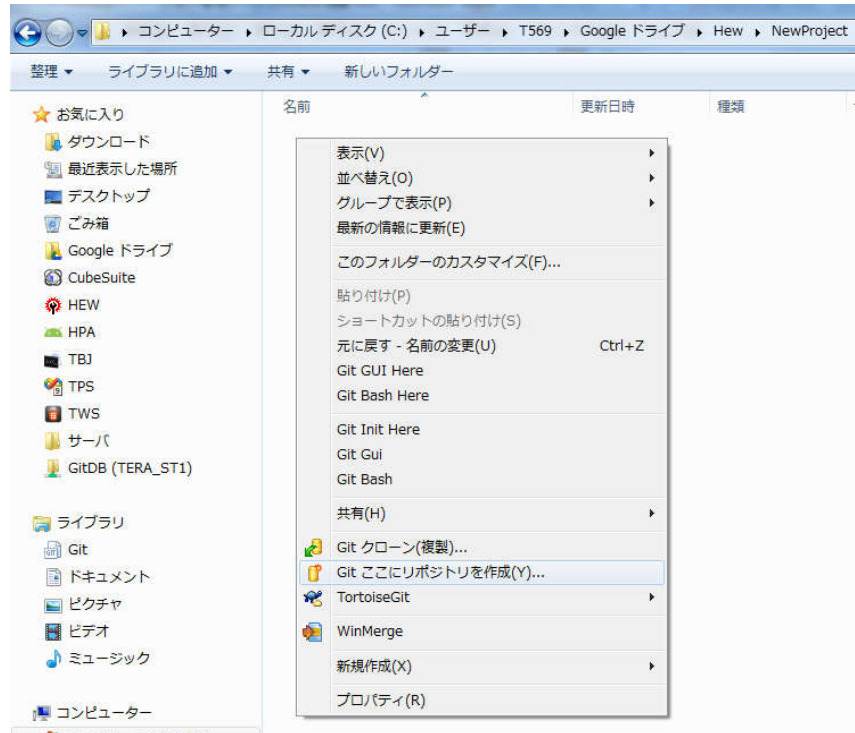
② リモートレポジトリの URL を入力して、「OK」をクリックする



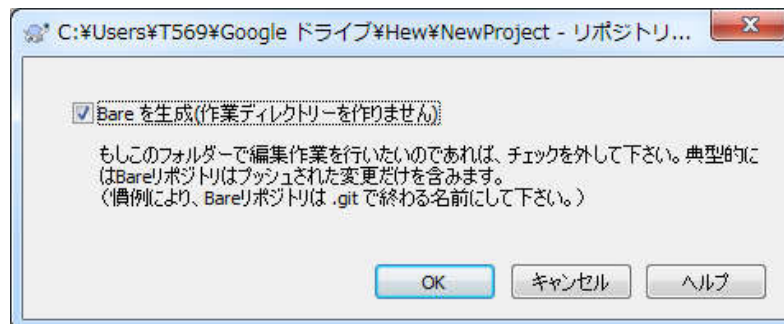
3. リモートレポジトリがない場合に開始する

① リモートレポジトリを作成する

- ・リモートレポジトリに移動して、選択フォルダで右クリックする
- ・「Git ここにリポジトリを作成(Y)...」を選択する



- ・下記の画面のように選択してください。



- ② 「III-2 リモートレポジトリが既に存在している場合に開始する」の順番に続けてください。

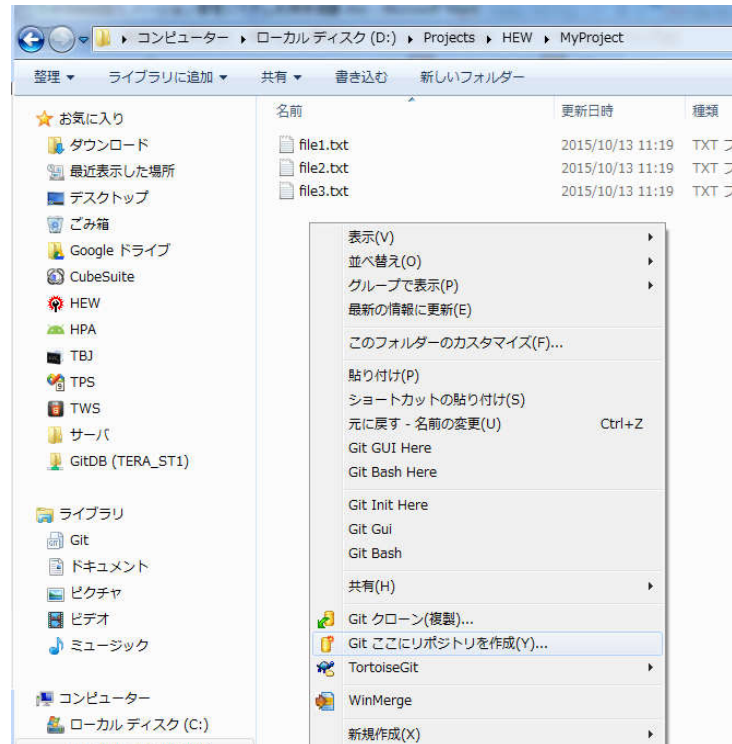
4. リモートレポジトリが不要な場合に開始する

他のところに同期しなくて、開発進捗を自分のパソコンで管理したい場合があります。

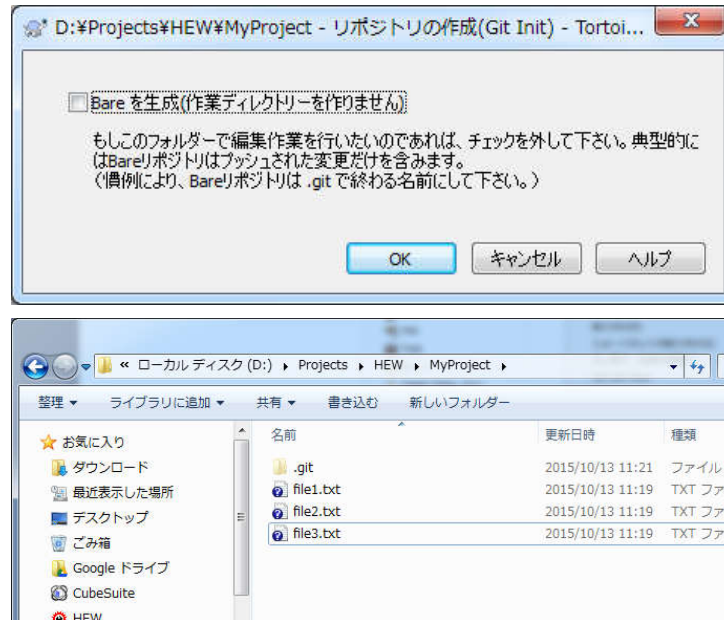
その時にリモートレポジトリが不要になります。

下記のステップどおりに行ってください。

- ① プロジェクトフォルダに右クリックして、「Git ここにリポジトリを作成...」を選択する

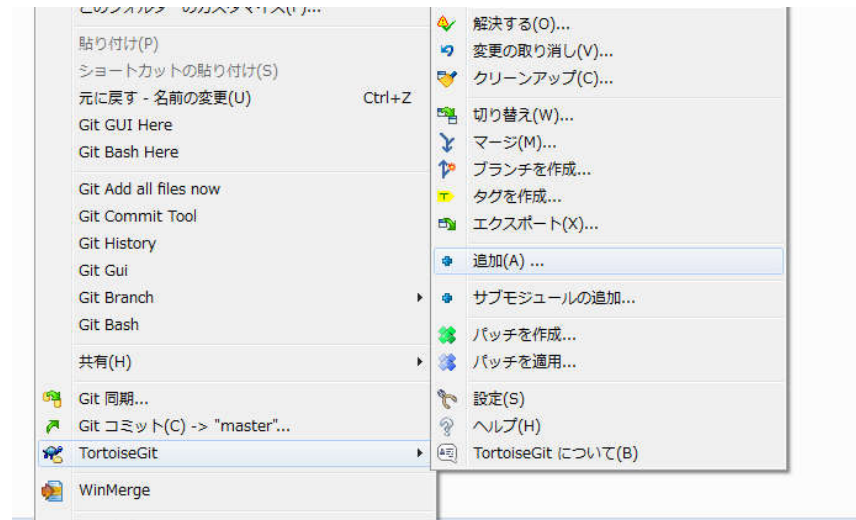


- ② チェックボックスを入れずに「OK」をクリックする

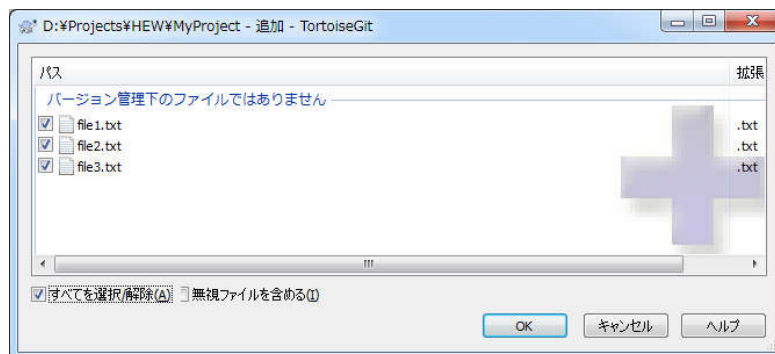


5. ファイルを管理対象にする(Add)

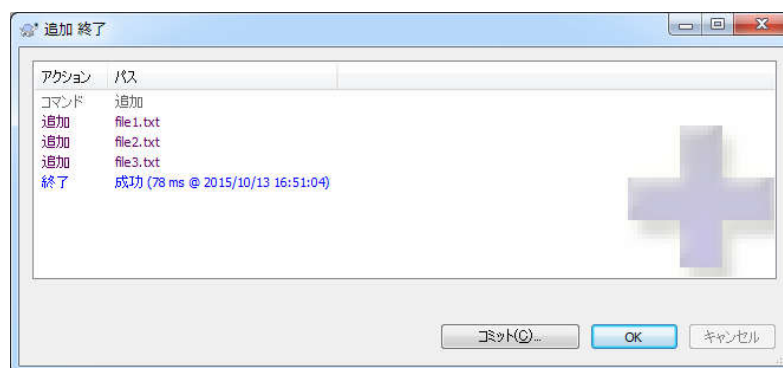
- ① プロジェクトのルートフォルダで右クリックする
- ② Git のメニューから「追加(A)...」を選択する



- ③ 確認画面が表示されたら、登録したいファイルにチェックが入っていることを確認し、「OK」をクリックする。



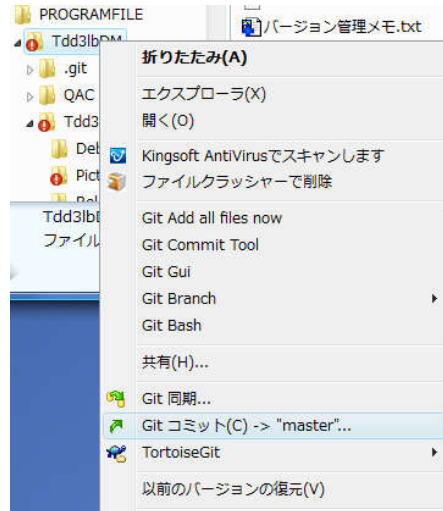
- ④ 追加操作後、再び確認画面が表示されるので、「OK」をクリックする



6. ファイル変更を記録する(Commit)

- ① プロジェクトのルートフォルダを右クリックする
- ② 「Git コミット (C) →...」をクリックする

例：「C:\Hew3\PROGRAMFILE\Tdd3lbDM」の master ブランチの場合

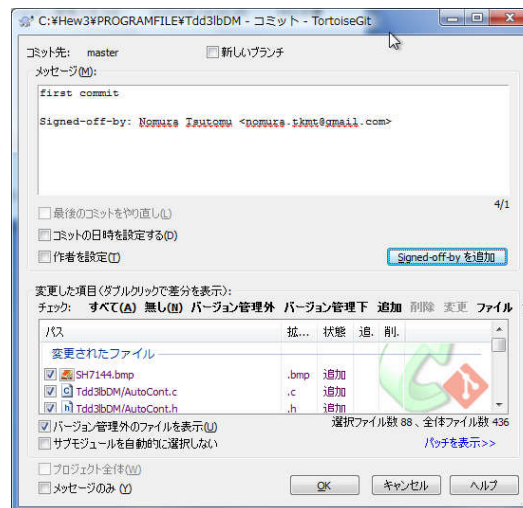


- ③ メッセージ欄にコメントを入力し、「OK」をクリックする

初回登録時は、「first commit」や「初回登録」等が望ましい。

以降の登録時は、変更点などを簡潔にまとめる。

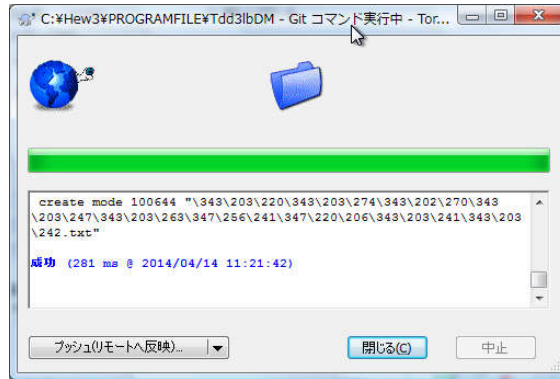
「signed-off-by を追加」をクリックすると、コメント末尾に署名が入る。



④ 成功を確認したら、「閉じる」をクリックする

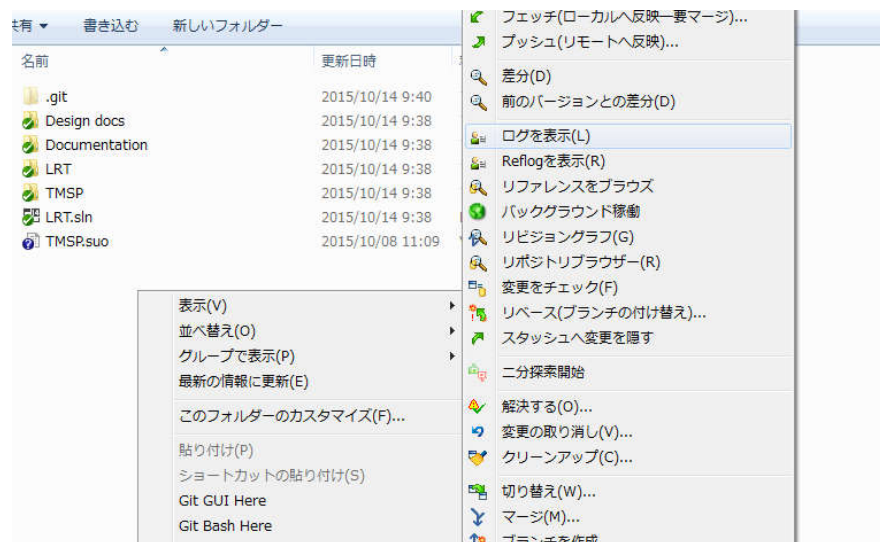
失敗した場合は、他の人の変更を先に反映させる必要がある。

(後に説明する、pull の操作を行う必要がある)

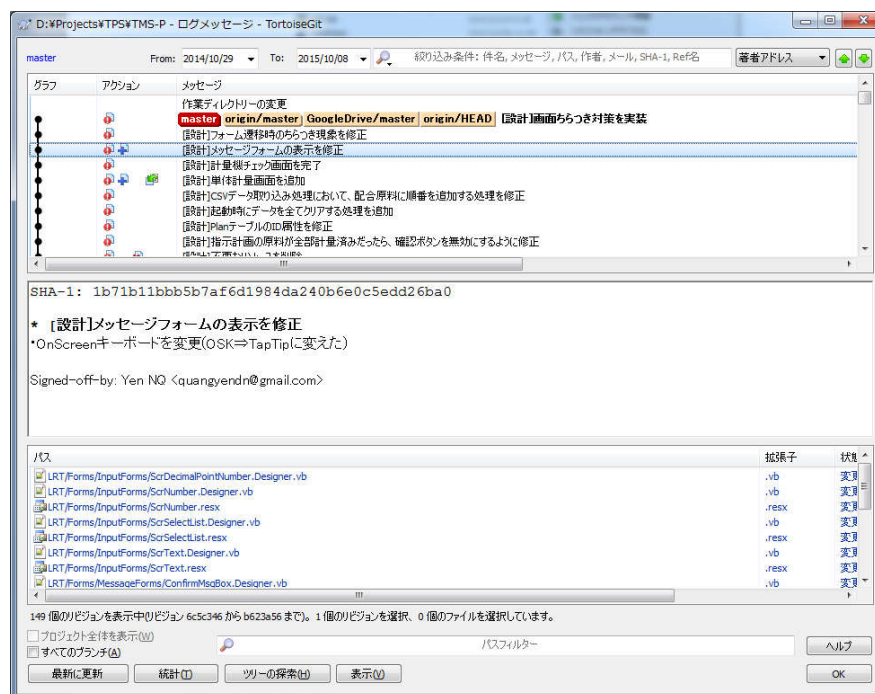


7. ログの表示

① プロジェクトのルートフォルダを右クリックする



② Git のメニューの「ログを表示(L)」をクリックする



8. リモートのレポジトリに変更を送る(Push)

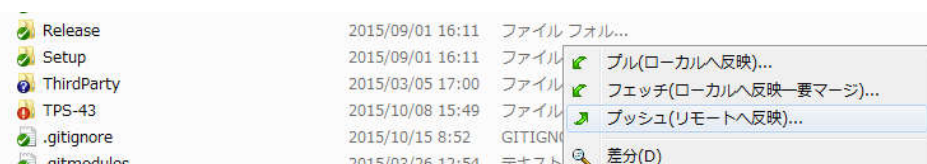
ローカルブランチのログをリモートブランチに送ります。

ローカルブランチとリモートブランチは同じブランチのこと。

リモートブランチがローカルブランチと異なる場合にはプッシュができますが、異常な結果が出る可能性があります。

下記の手順で行ってください。

- ① プロジェクトのルートフォルダで右クリックする
- ② Git メニューの「プッシュ(リモートへ反映...)」をクリックする

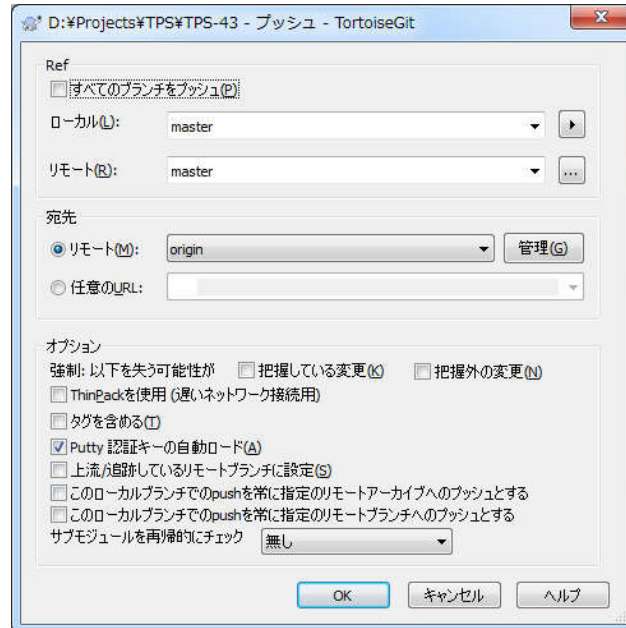


③ プッシュ画面が表示されたら、ローカルブランチとリモートブランチを選択する

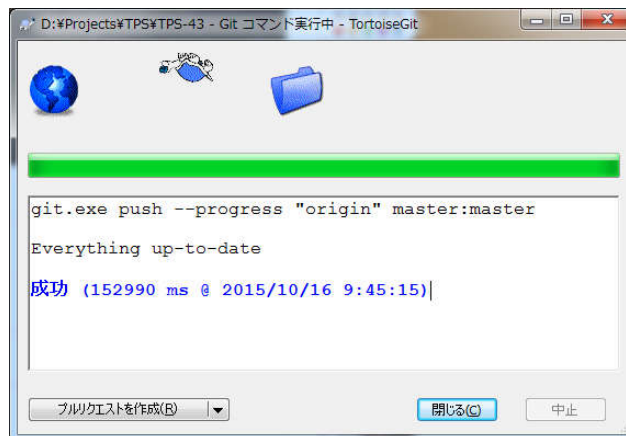
＊ ローカルブランチとリモートブランチは同じのこと

④ リモートレポジトリを選択する

⑤ 作成したタグをプッシュしたいならば、「タグを含める」に入れる

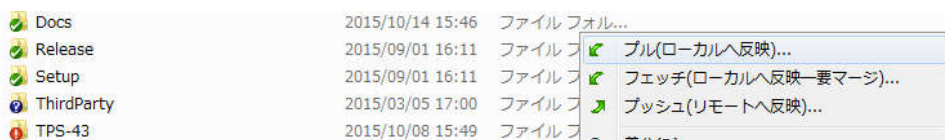


⑥ 「OK」をクリックし、完了を待つ

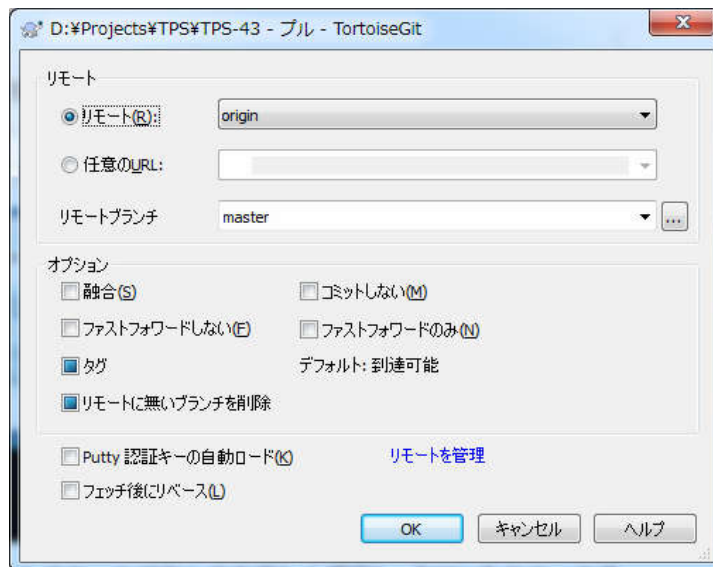


9. リモートのレポジトリの変更を更新(Pull)

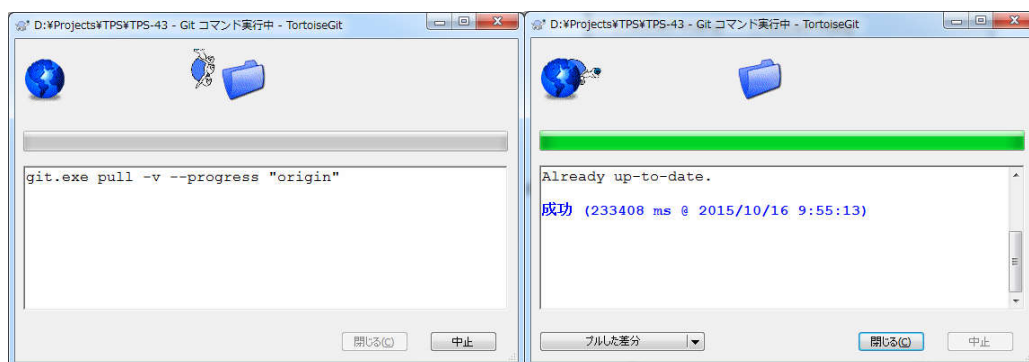
- ① プロジェクトのルートフォルダを右クリックする
- ② Git のメニューの「プル(ローカルへ反映...)」をクリックする



- ③ リモートレポジトリとリモートブランチを正しく選択し、「OK」をクリックする



- ④ 成功したことを確認し、「閉じる」をクリックする

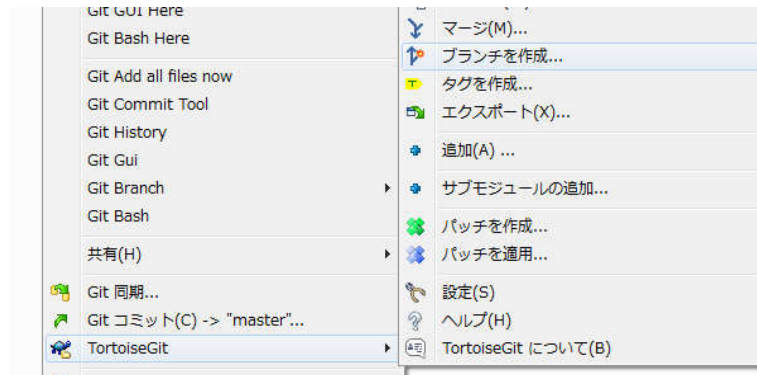


10. ブランチを作成

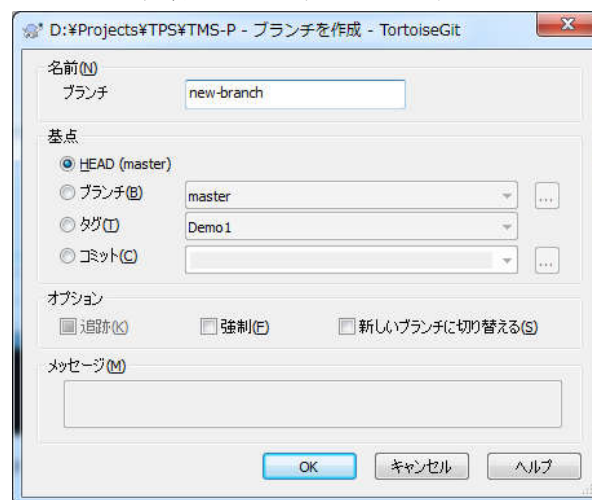
ブランチの作成は 2 つの方法があります。

① 方法 1：ブランチを作成してからソースコードを修正する

- ・プロジェクトのルートフォルダを右クリックする
- ・Git メニューの「ブランチを作成...」をクリックする



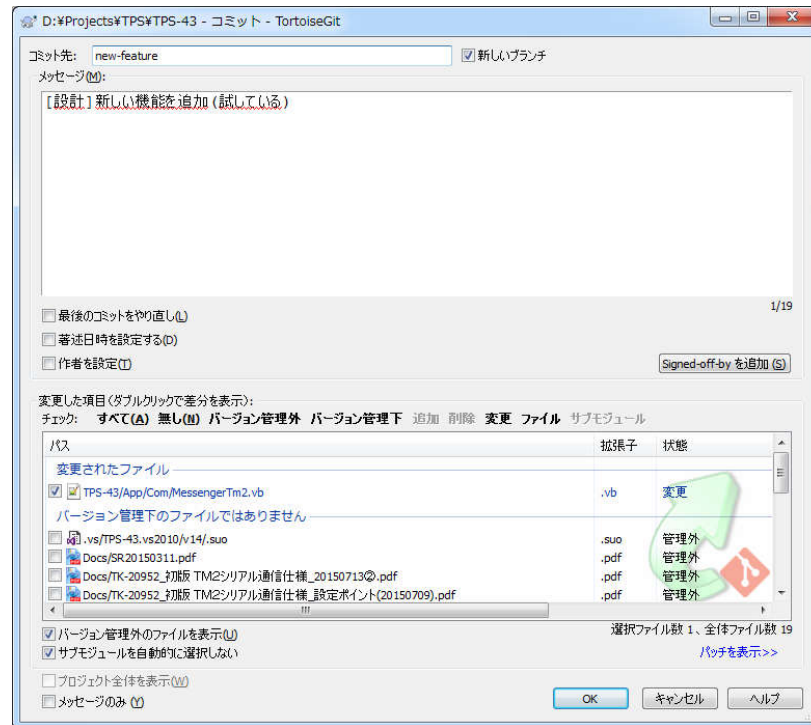
- ・ブランチの名前を入力し、ブランチの基点を選択する
- ・すぐに新しいブランチに切替えるかどうかを選択する



- ・「OK」をクリックし、完了まで待つ
- ・新しいブランチに切替えたら、ソースコードの修正を始めます。

② 方法 2 : ソースコードを修正してからブランチを作成する

- ・ソースコードの修正を行う
- ・コミットを行う
- ・コミット画面で「新しいブランチ」を入れ、ブランチ名を入力する

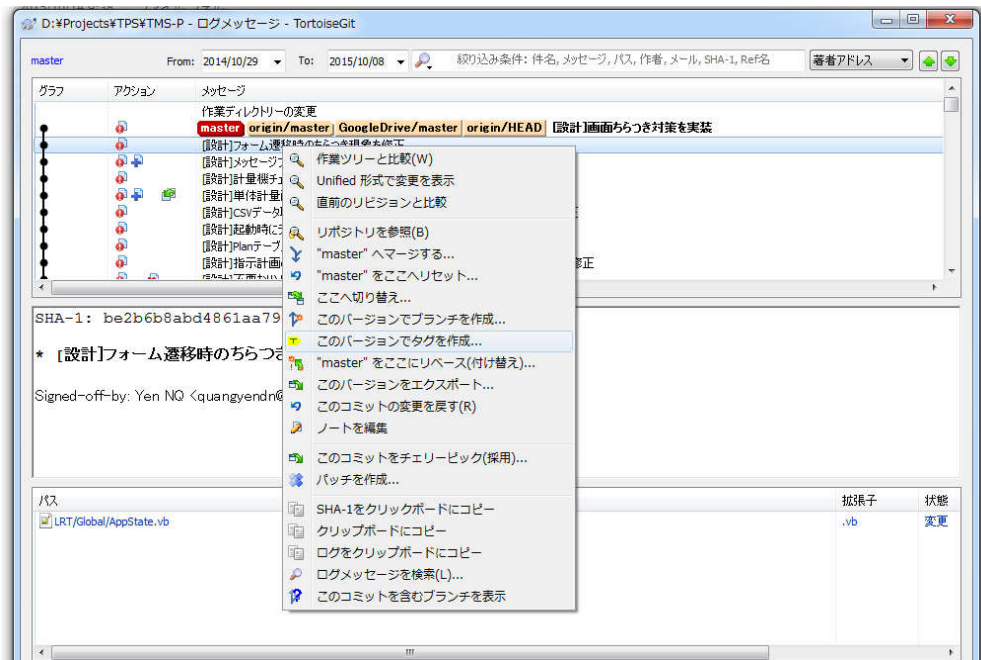


- ・「OK」をクリックしたら、新しいブランチに自動的に切り替えます

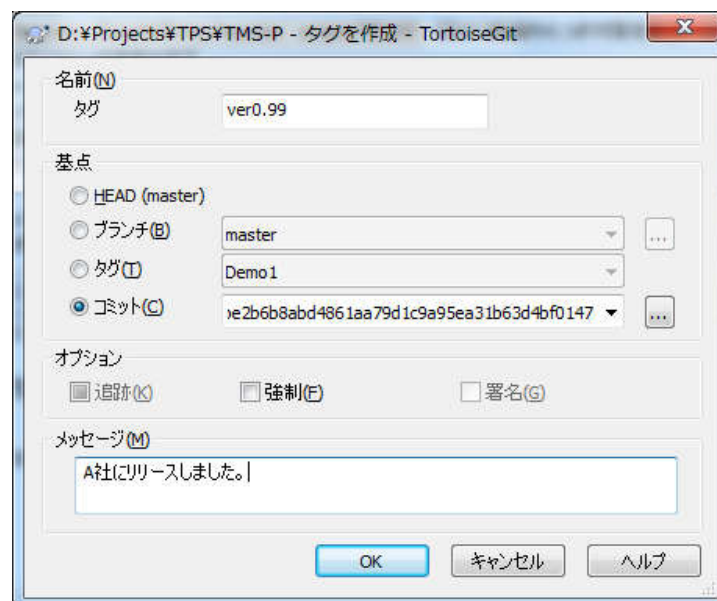
11. タグを作成

バージョンアップ、ソフトリリースバージョンなどをマークする為に、タグを付けます。

- ① ログを表示する
- ② ソースコードのバージョンを選択し、右クリックする
- ③ 「このバージョンでタグを作成...」をクリックする



- ④ タグ名とメッセージを入力し、「OK」をクリックする



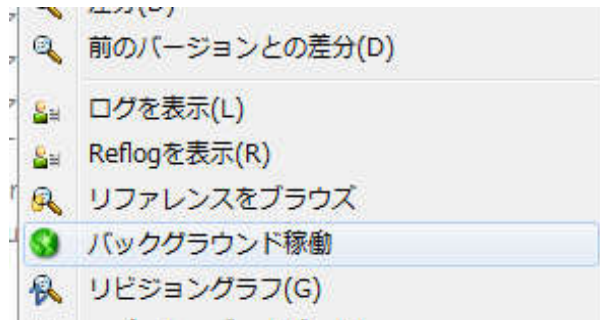
IV. 共有について

1. すぐに LAN の PC に Git のデータベースを共有する(Daemon)

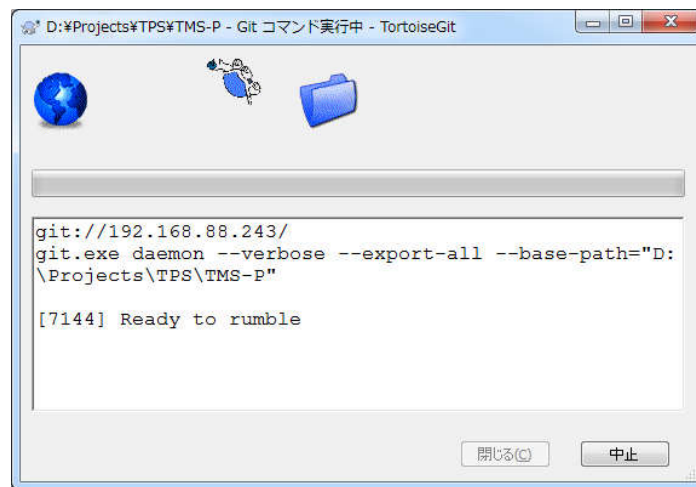
Daemon は LAN ネットワークで開発者 1 のレポジトリを開発者 2 に共有したい場合に適用します。

以下のような手順で行ってください。

- ① 開発者 1 のパソコンでプロジェクトのルートフォルダを右クリックする
- ② Git のメニューの「バックグラウンド稼働...」をクリックする

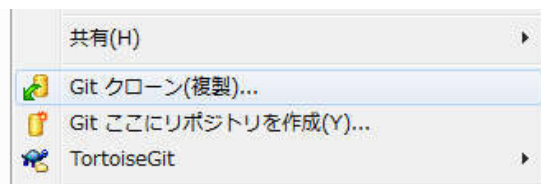


- ③ Daemon 画面が表示したら、共有の URL を確認する

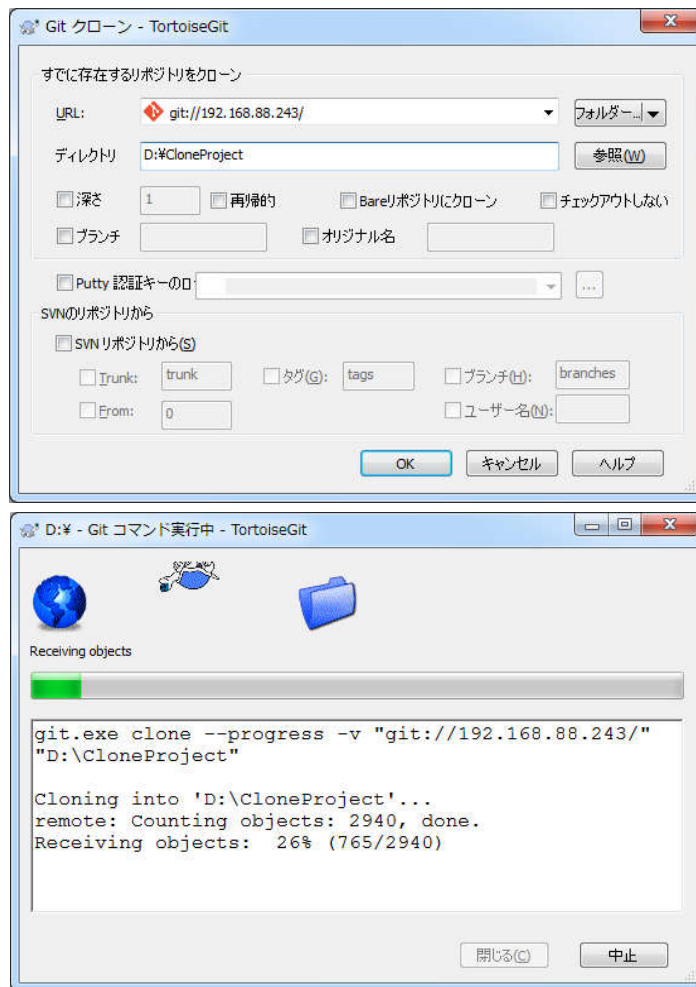


この場合では共有 URL は <git://192.168.88.243/> になっています。

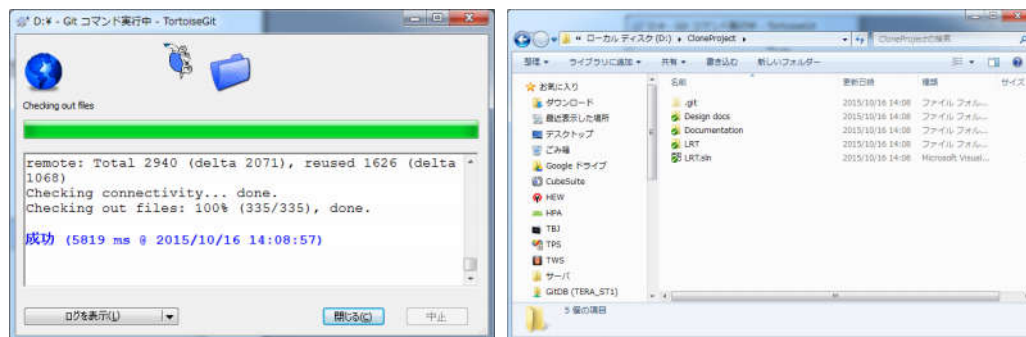
- ④ 開発者 2 のパソコンでプロジェクトをクローンしたい場所に移動する
- ⑤ 右クリックし、「Git クローン(複製)...」をクリックする



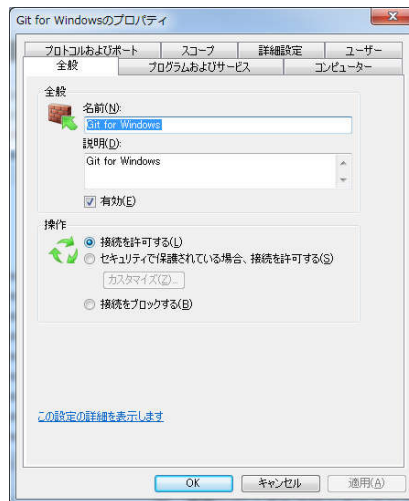
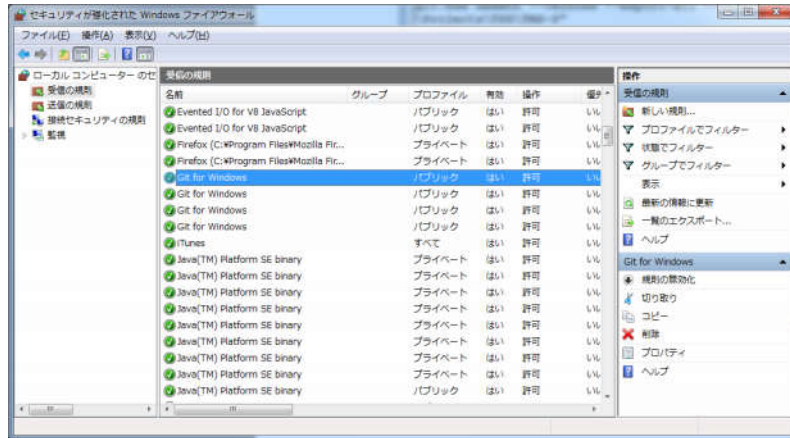
⑥ 先ほどの URL を入力し、「OK」をクリックする



⑦ クローンが完了したら、Deamon 画面を閉じて大丈夫です



注意 : Windows のファイルウォールで「Git for Windows」の接続を許可すること



2. Skype でローカルの Git データベースを共有する

Daemon が無理で、すぐに他の開発者にログをシェアしたい場合に適用します。

- ① 開発者 1 はローカルレポジトリ(プロジェクトの「.git」フォルダ)を圧縮する
- ② 圧縮したファイルを Skype などでする
- ③ 開発者 2 の側は受信した「.git」フォルダを展開する
- ④ 展開したフォルダはリモートレポジトリとして、クローンする
- ⑤ この段階ではログが確認できるようになりました

V. その他の操作

1. リモートから他のブランチをローカルに同期する(Fetch)

以下の場合にフェッチを適用する

- ・開発中ソースコードに影響せずにリモートレポジトリの最新ログを確認したい
- ・他の開発者が変更したデータをローカルレポジトリに反映させたい

以下の手順でフェッチを行ってください。

① フェッチしたいブランチに切り替えする(Switch)

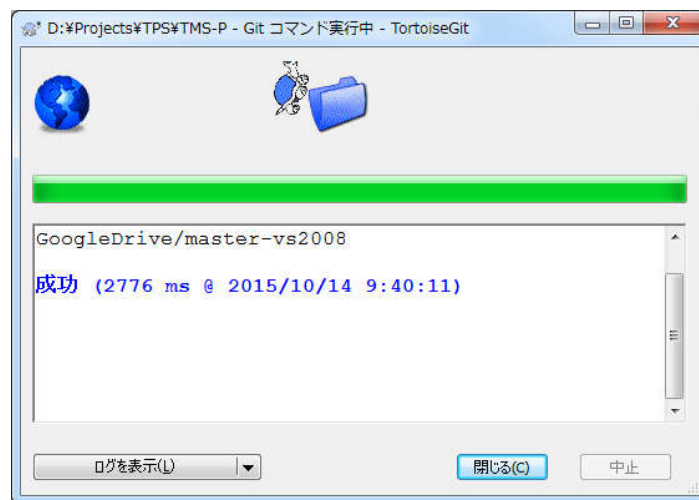
*現在のブランチはフェッチしたいブランチなら切り替えしなくても大丈夫です。

② プロジェクトのルートフォルダで右クリックする

③ Git メニューの「フェッチ」をクリックする

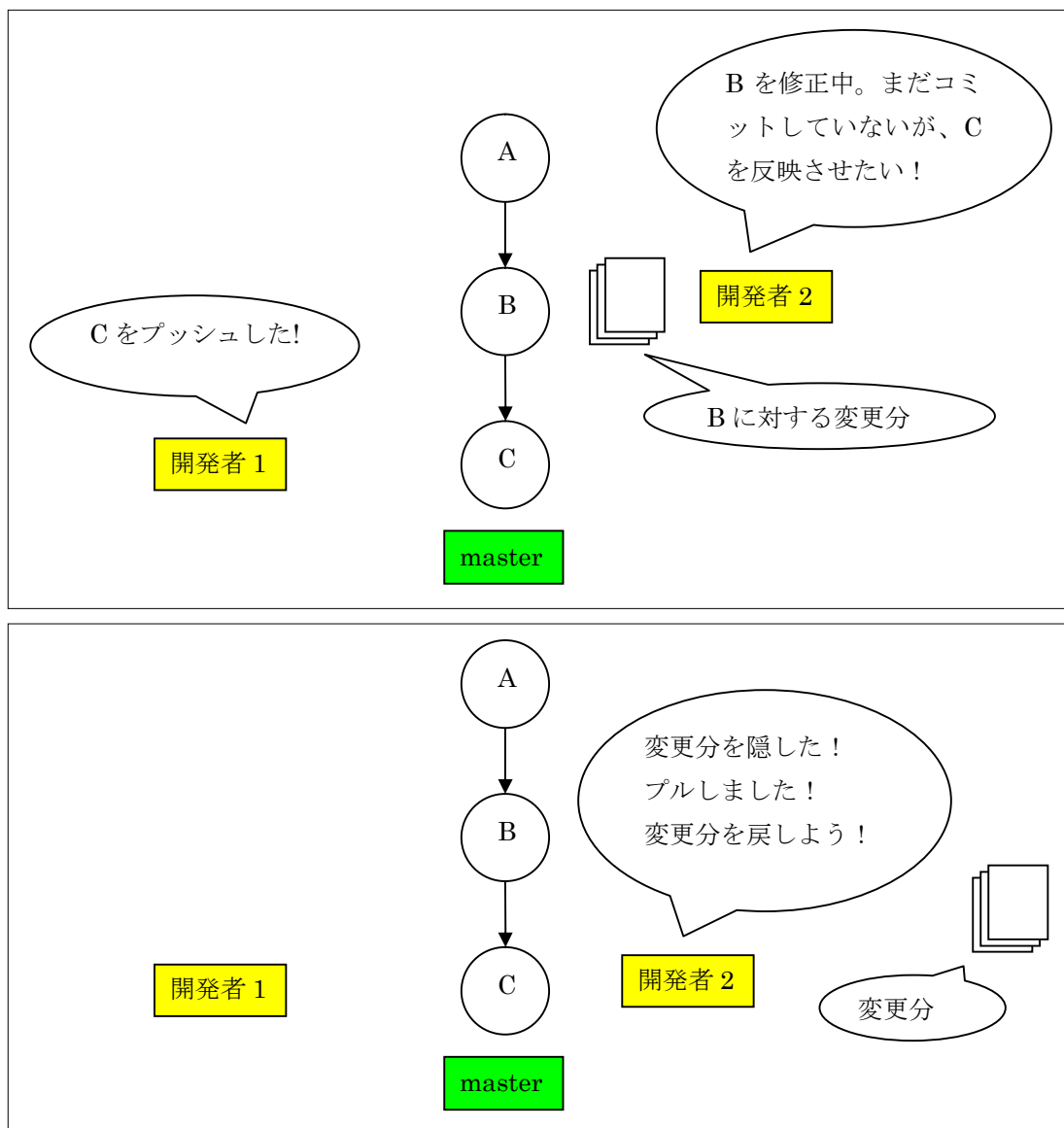
④ フェッチ画面を表示したら、リモートレポジトリを選択する

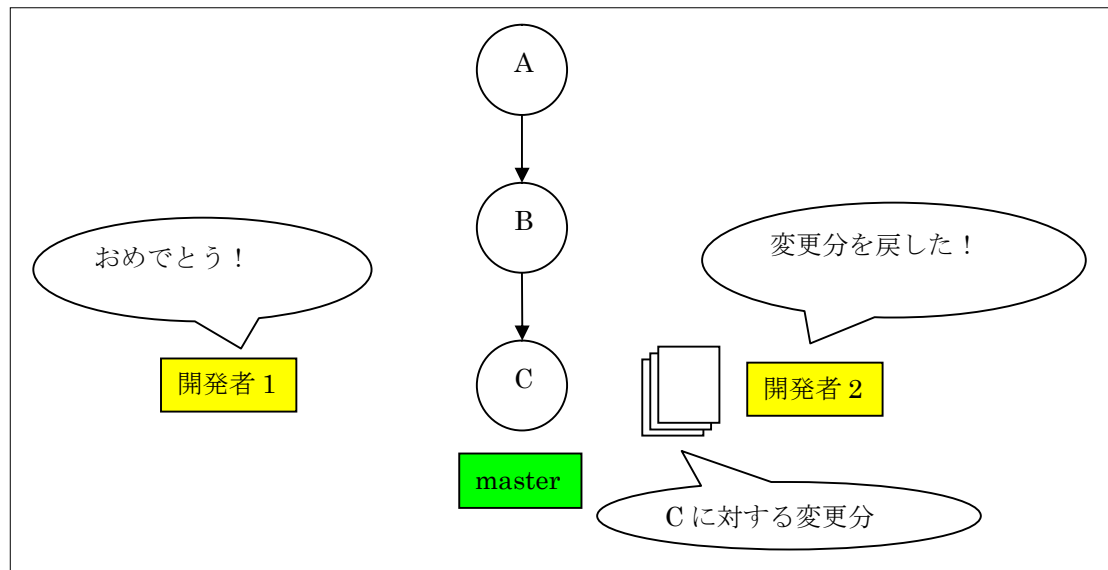
⑤ 「OK」をクリックし、完了を待つ



2. スタッシュへ変更を隠すのはどの場合に適用する？(Stash save)

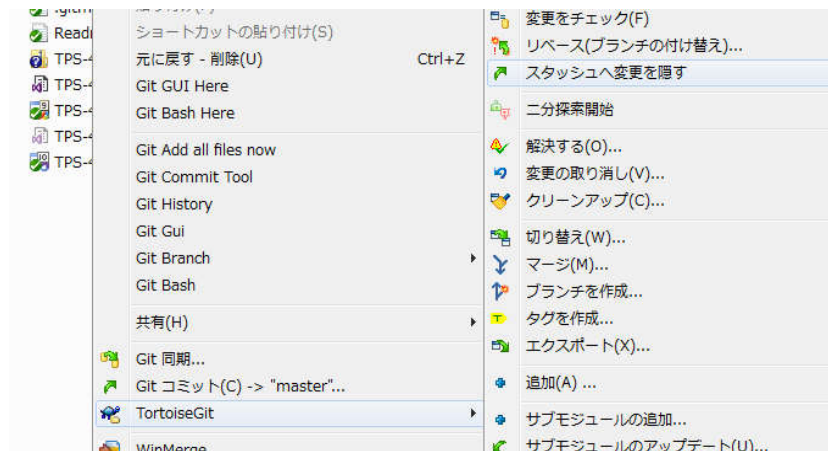
開発中ソースコードにはまだコミットしていないファイルがありますが、他の開発者からの修正を自分のソースコードに反映させたい場合にスタッシュが適合です。



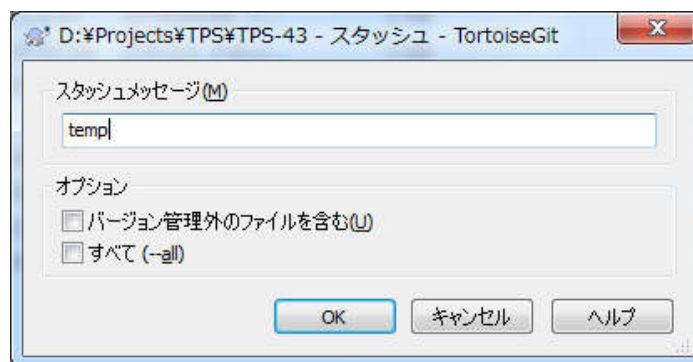


上記の画像からスタッシュへの変更を隠すのを知りましたか？

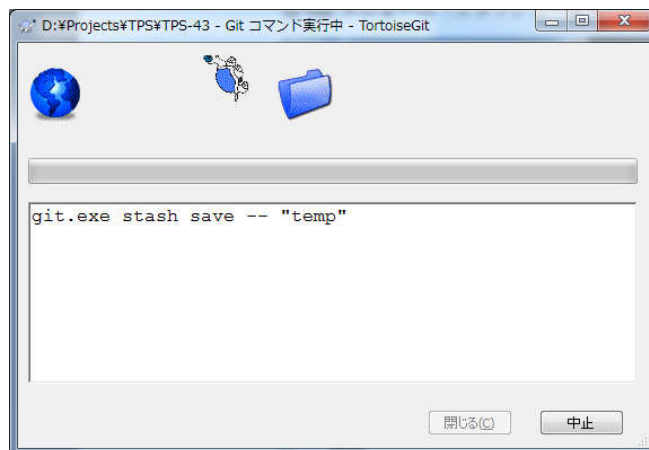
- ① プロジェクトのルートフォルダを右クリックする
- ② Git メニューの「スタッシュへ変更を隠す」をクリックする



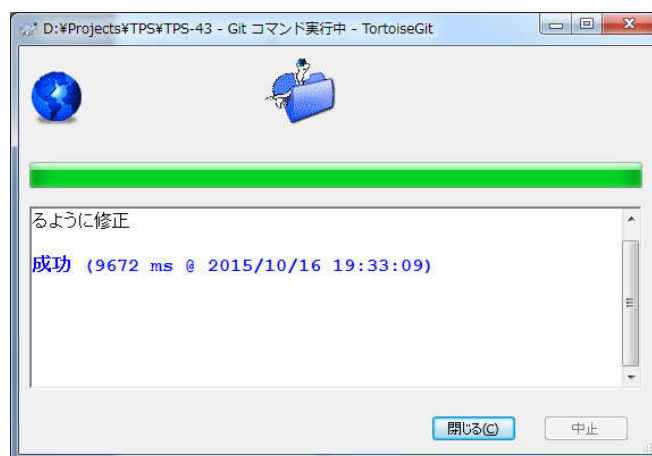
- ③ スタッシュメッセージを入力し、「OK」をクリックする



④ 完了を待つ

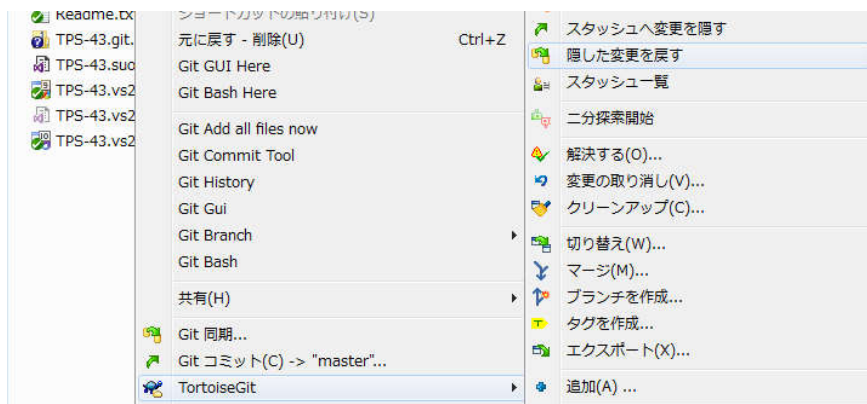


⑤ 閉じる



この段階でソースコードがプルできます。

プルした後に、プロジェクトのルートフォルダを右クリックし、「隠した変更を戻す」を選択してください。



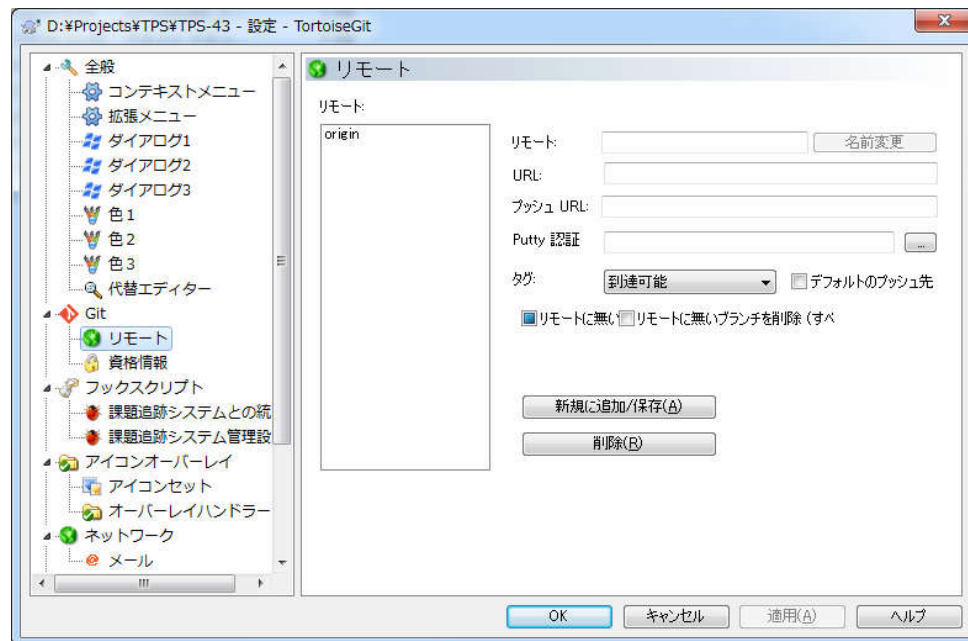
3. GoogleDrive とリモートサーバ(VPN 経由)と一緒に使う

VPN 接続はいつも正常に使えるわけではありません。そのため、リモートレポジトリは GoogleDrive と Git サーバを併用したほうが安心して使えます。

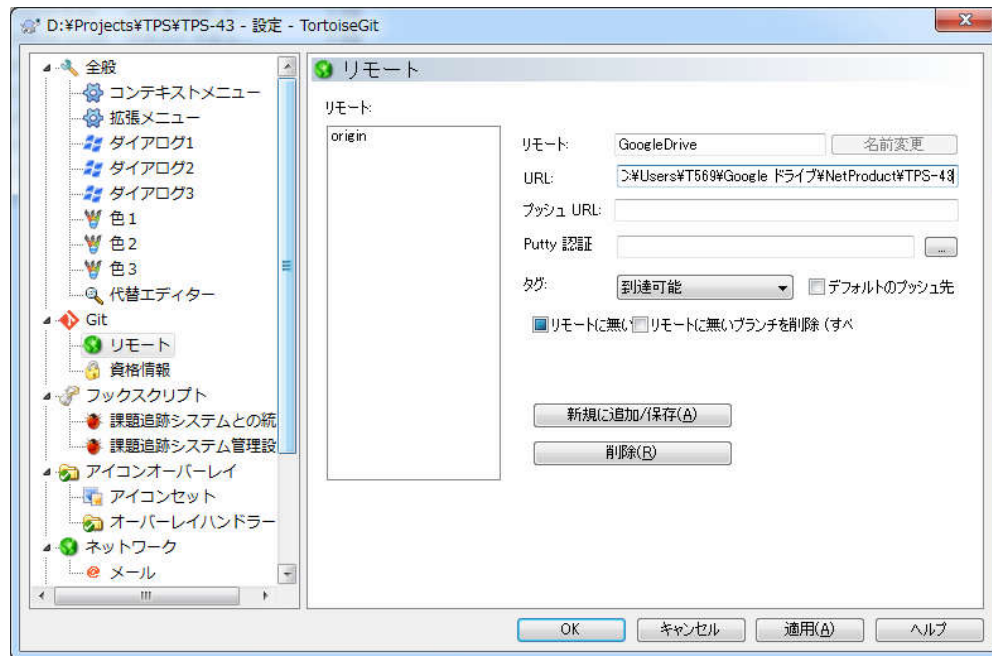
普通の使用と違うのはプッシュとプルの時だけです。

① 設定方法

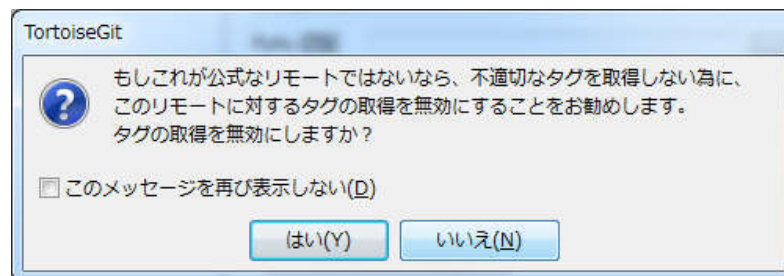
- ・プロジェクトの Git 設定を開く
- ・Git⇒リモートに移動する



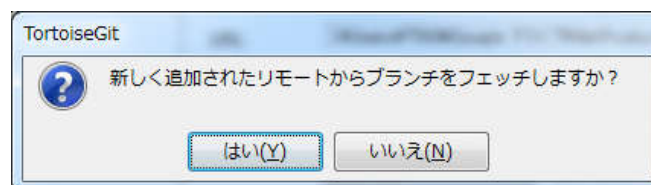
- ・「リモート名」に GoogleDrive を入力する
- ・「URL」に GoogleDrive に於けるレポジトリの URL を入力する



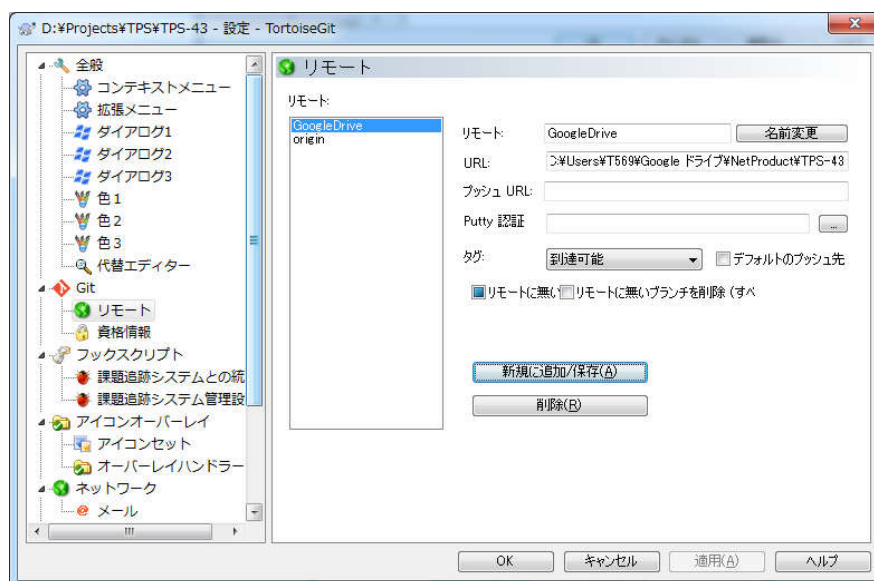
- ・「新規に追加／保存」をクリックする
- ・GoogleDrive リモートに対して、タグの取得を無効するダイアログが表示したら、「いいえ」をクリックする（同一のレポジトリなので、タグも同じはず）



- ・GoogleDrive リモートからフェッチするかどうかの確認画面が表示されたら、「いいえ」をクリックする（すぐにフェッチしないという意味）



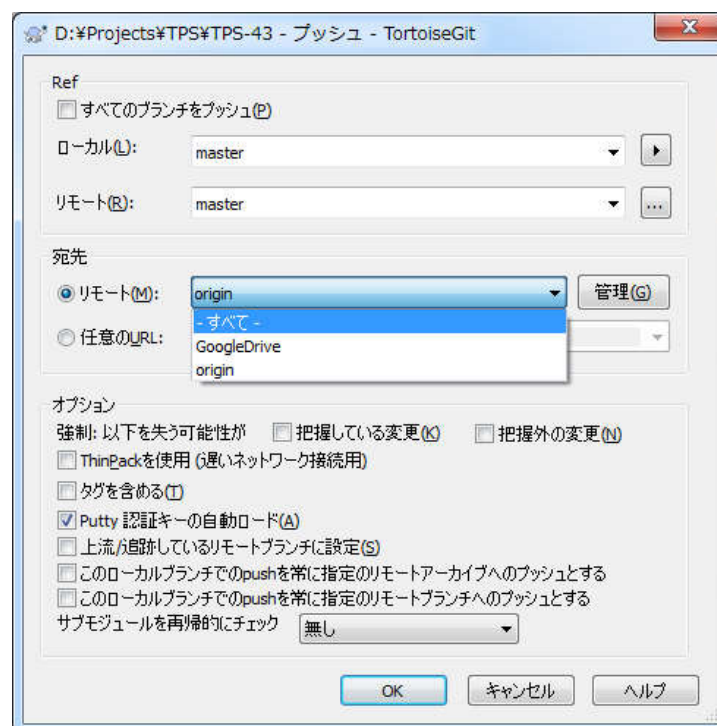
- ・「OK」をクリックして完了します。



② プッシュの時

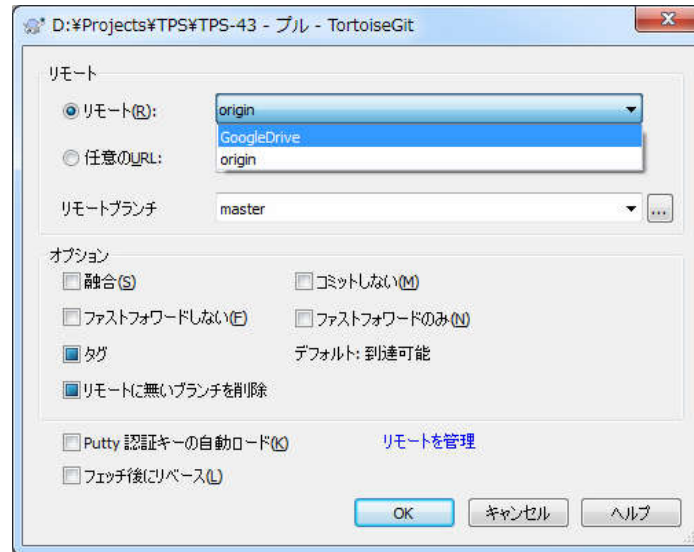
プッシュの時には基本的に両方にプッシュします。

プッシュ画面で宛先のリモートは「-すべて-」を選択してください。



③ プルの時

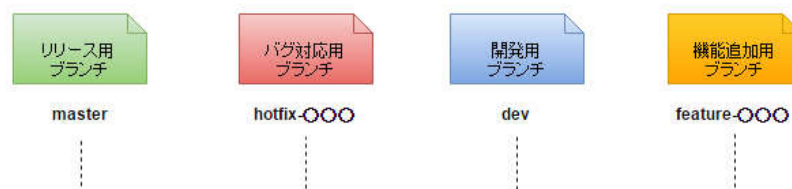
普段は `origin` だけからプルしたら良いですが、VPN 接続が良くないか、`origin` のサーバに接続できません場合には `GoogleDrive` からプルします。



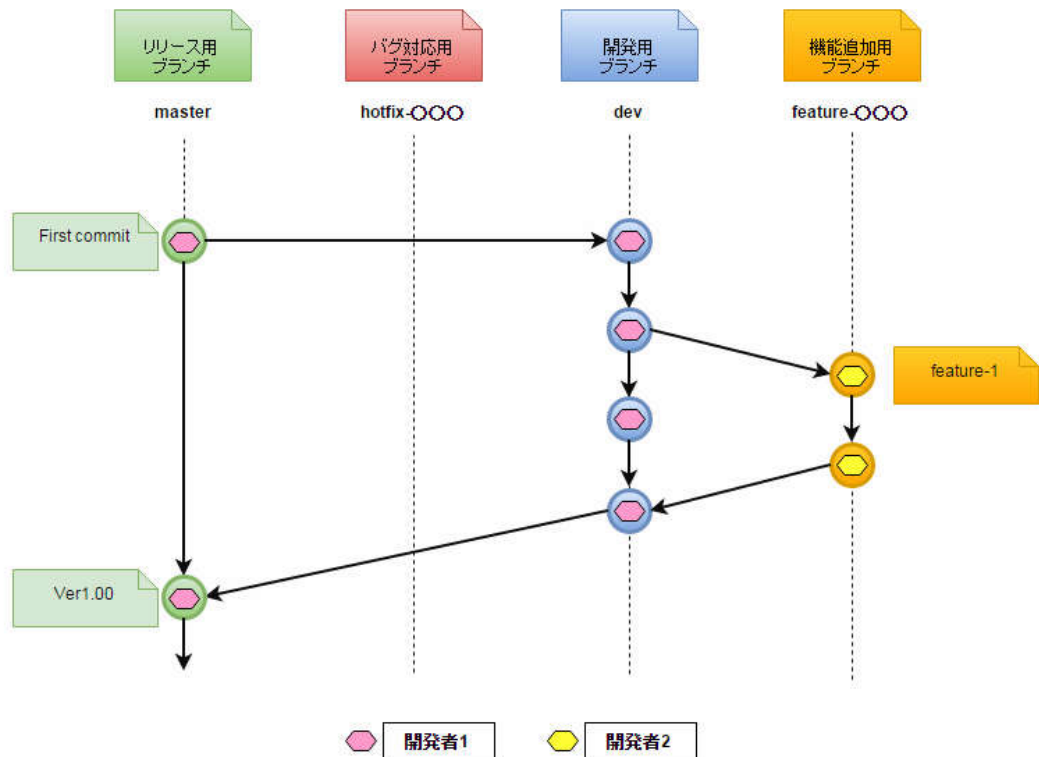
VI. ブランチ戦略について

何が起こったのと何をしたのをつかりやすくなる為に、下記のブランチ戦略がお勧めします。

- 基本のブランチ : *master*、*dev*
- 機能追加用ブランチ : *feature-〇〇〇*
- リリース後の緊急修正用ブランチ : *hotfix-〇〇〇*



下記のリビジョングラフを確認しましょう。



やっぱり、開発者 1 はこのプロジェクトの主役ですね。

「First commit」を **master** にプッシュした後、**dev** で開発を行うようにしました。
途中で、開発者 2 に機能追加を依頼しました。

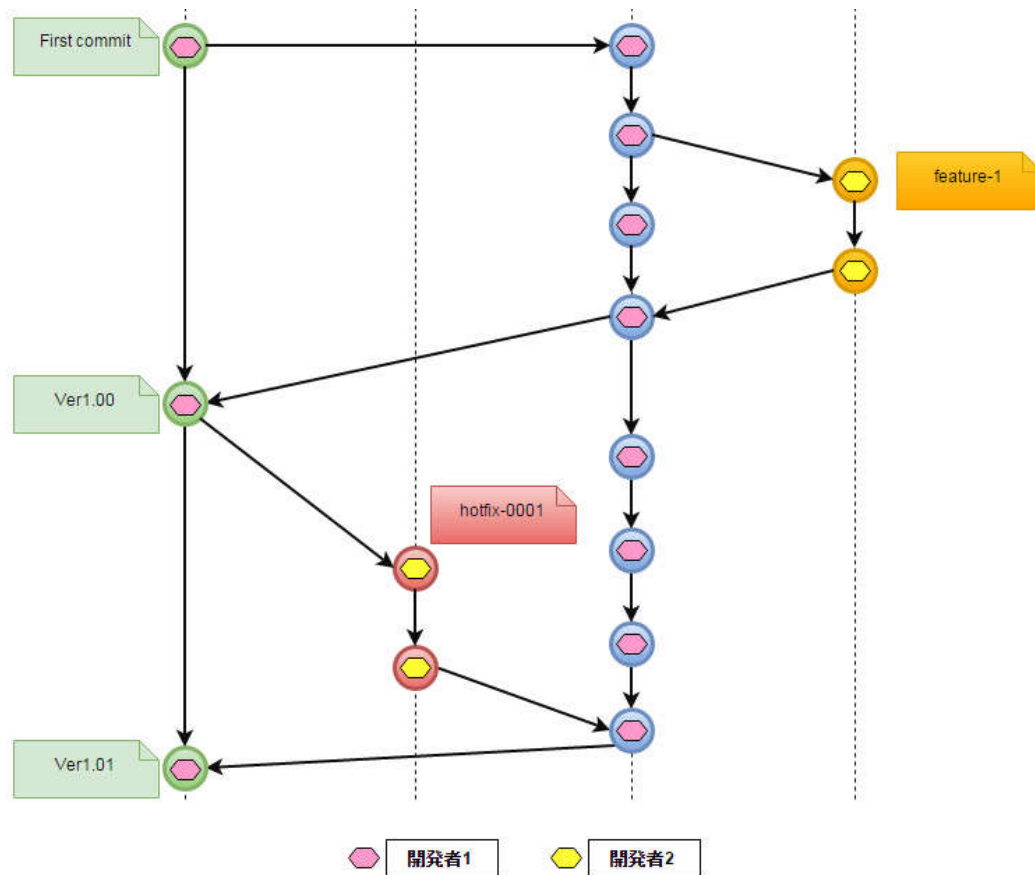
⇒ 開発者 2 は **dev** から **feature-1** ブランチを作成し、機能を追加しました。

⇒ 追加後に開発者 1 は **feature-1** を **dev** にマージしました。

ソフトの動作確認を行った後、**dev** ブランチを **master** にマージしました。

この段階では **master** は最初のバージョンになりました。

このプロジェクトは次のように続きました。



開発者 2 は機能 1 を追加した時に、よく確認しなかったので、Ver1.00 にバグが発生しました。

master の Ver1.00 時点のリビジョンから *hotfix-0001* を作成し、Ver1.00 に対して不具合を修正します。(0001 は不具合の番号)

開発者 2 は *hotfix-0001* を修正して確認した後で、開発者 1 に連絡し、開発者は *hotfix-0001* を *dev* ブランチにマージします。

そして、*dev* を *master* にマージし、Ver1.01 をリリースする

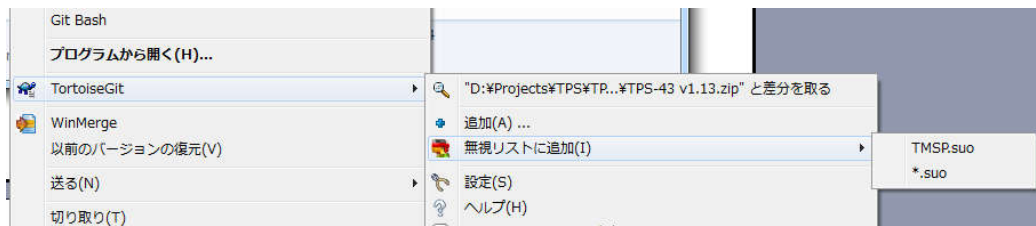
VII. 注意項目

1. ファイル変更の無視(Ignore)について

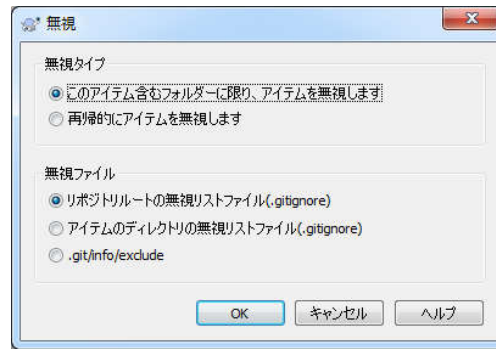
プロジェクトタイプにより、バージョン管理に登録する必要のないファイルもあります。

例：Visual Studio プロジェクトには **.suo*、**.vbproj.user*、*bin* フォルダ、*obj* フォルダはバージョン管理に登録する必要はないので、無視するようにしましょう。

- ① *TMSP.suo* を右クリックし、無視メニューの「**.suo*」をクリックする(全ての *.suo* ファイルを無視する)

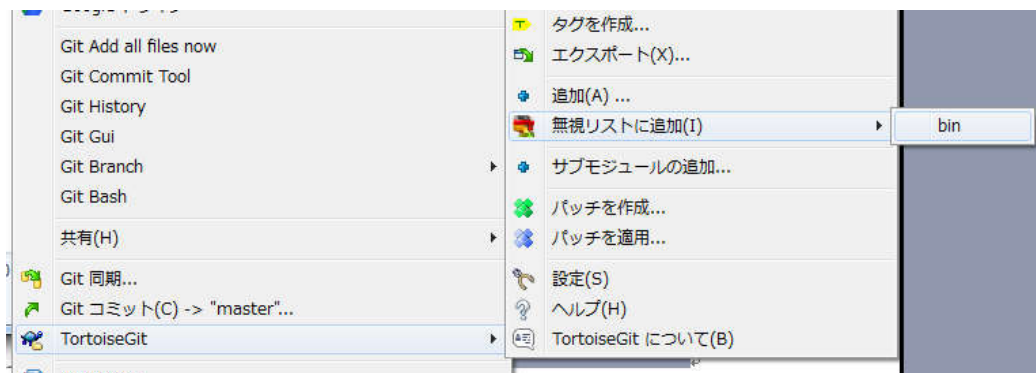


無視確認の画面が表示されたら、「OK」をクリックする

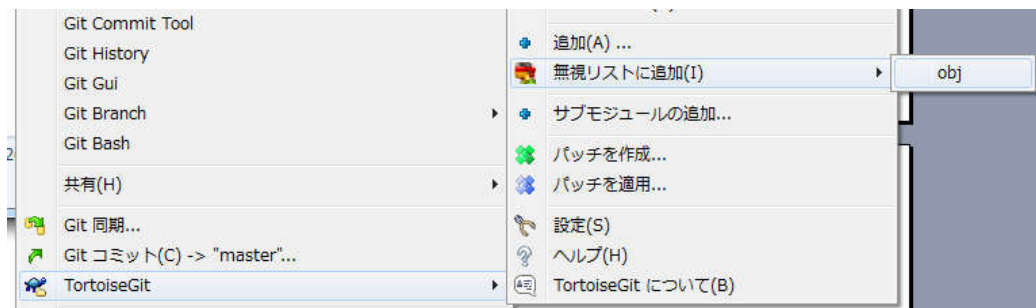


無視情報はプロジェクトのルートフォルダの *.gitignore* ファイルに自動的に追加されます。

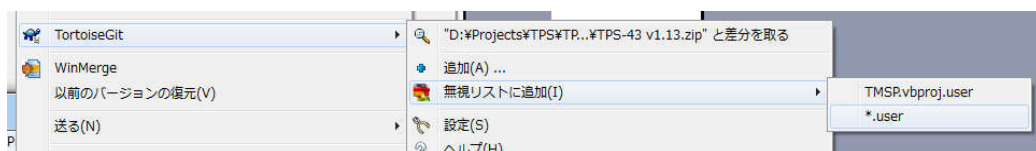
② 同じく *bin* フォルダを無視するように設定する



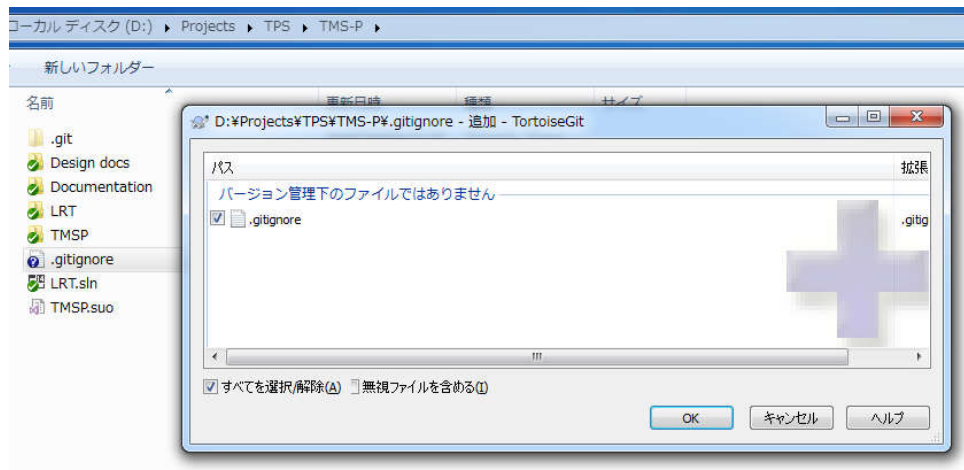
③ *obj* フォルダも無視するように設定する



④ 次に *TMSP.vbproj.user* を右クリックし、「*.user」をクリックする



- ⑤ 最後に *.gitignore* ファイルもバージョン管理に登録し、コミットする(他の開発者がこのソースをチェックアウトしたら、上記の項目が自動的に無視されるようになる)



2. GoogleDrive をリモートレポジトリとして使う場合の注意

GoogleDrive はデータの同期をすると時間がかかります。

そのため、GoogleDrive にプッシュした後はシステムトレイの GoogleDrive アイコンをよく確認するのをおすすめします。



GoogleDrive のアイコン

ブランチ名ファイル(/refs/heads/の中にある)が同期されたまでには待たないといけません。

例えば、**master** ブランチを GoogleDrive にプッシュしたらシステムトレイの GoogleDrive のアイコンを右クリックして同期の状態を確認します。

/refs/heads/master ファイルが同期したら、レポジトリも完全に同期されました。

ずっと待っても **master** ファイルが同期されない場合では GoogleDrive を再起動してください。

VIII. その他の操作 (参考用 URL)

下記サイトをご参照ください。

サルでもわかる Git 入門

<http://www.backlog.jp/git-guide/>