

# RStudio と Git でバージョン管理

*Shouhei TAKEUCHI*

*May 5, 2016*

## Contents

<b>このドキュメントの目的</b>	<b>2</b>
最終更新時の情報 . . . . .	2
<b>Git とは？</b>	<b>2</b>
<b>Git のインストール</b>	<b>2</b>
<b>Git のセットアップ</b>	<b>2</b>
Git の設定 . . . . .	2
RStudio で Git を有効にする . . . . .	3
プロジェクトで Git を有効にする . . . . .	3

## このドキュメントの目的

Windows 環境で、RStudio と Git を使って、R のコードのバージョン管理（変更履歴を残す）および共有ができるようになることを目的とする。<sup>1</sup> また、RStudio と Git の連携において、テキストファイル以外を使うためには、工夫が必要なので、そちらについても軽く触れておく。<sup>2</sup>

## 最終更新時の情報

- このファイルの最終更新日時：2016-05-05 17:45:28
- R のバージョン：3.2.4 Revised
- RStudio のバージョン：00.99.893
- Git のバージョン：2.7.1

## Git とは？

コードを書いたファイル（テキストファイル）をバージョン管理する際に、変更のたびにファイル名を変えて履歴を管理すると、ファイル数がふくれあがっていく。Git は、そういう問題を解決するために、ファイルの「変更履歴を記録」しておくための仕組みとなる。

サルでもわかる Git 入門<sup>3</sup>の入門編までを読んでみると使い方の簡単な説明があり、すごく理解の助けになる。最終的には、発展編の branch まで理解しておきたい。

## Git のインストール

Git 本家<sup>4</sup>から最新版（Git-2.8.2-64-bit.exe：64bit システムの場合）をダウンロードし、インストールする。オプションはよく読んで設定する。

## Git のセットアップ

残念ながら、Git はインストールしてすぐ簡単に使えるものではない。最低限必要な設定などを以下にまとめておく。

## Git の設定

最低限やることは以下の通り。

---

<sup>1</sup>Mac は手元にないので、試せない。

<sup>2</sup>この資料では、R と RStudio のインストールが終わっていることを前提とする。

<sup>3</sup><http://www.backlog.jp/git-guide/>

<sup>4</sup><https://git-scm.com/>

1. すべてのプログラム->Git->Git Bash より、以下のコマンドを用いて Git に作業者の情報を登録する。当然、メールアドレスと、名前は自分のものに変更する。名前は、半角英数のみに限定しておいた方が、余計なトラブルを起こさない。<sup>5</sup>

```
git config --global user.email "Git や GitHub、Bitbucket で使うメールアドレス"  
git config --global user.name "Git で使う名前"
```

ここの設定を怠ると、誰がコードに変更を加えたのかがわからなくなるので、十分に注意して設定する。特に GitHub や Bitbucket などリモート環境での共有・管理を考えている場合は、登録したメールアドレスを使っておくと楽である。(さらにそもそも GitHub や Bitbucket に登録するユーザ名・メールアドレスは統一しておくとう便利。) 下記のコマンドで設定の確認ができるので、試しておくとう良い。

```
git config --list
```

## RStudio で Git を有効にする

RStudio 側でも Git を有効にする必要がある。<sup>6</sup>

1. RStudio で、「Tools」->「Global Options…」->「Git/SVN」を選択
2. Enable version control interface for RStudio projects にチェックを入れる。
3. Git executable: に「C:/Program Files/Git/bin/git.exe」を追加 (Browse…から git.exe を探せば良い)

## プロジェクトで Git を有効にする

RStudio で Git を有効にしても、勝手に全てのプロジェクトで Git が有効になるわけではない。プロジェクトごとに Git を有効にする必要がある。<sup>7</sup>

### Git が有効な新規プロジェクトの作成

RStudio と Git の連携では、基本的にリモートのレポジトリを使うことが想定されている。なので、事前に GitHub か Bitbucket にレポジトリを作成しておくとうよい。

1. 「New project」->「Version Control」->「Git」を選択
2. 「Repository URL:」にリモートリポジトリの URL を貼り付ける。<sup>8</sup> 問題がなければ、自動で、「Project directory name:」も入力される。
3. 「Create project as subdirectory of:」に、プロジェクトのフォルダを作成するフォルダを指定する。<sup>9</sup>

<sup>5</sup>個人的には、トラブルは可能な限り避けたいので、ニックネームを使って名前にもスペースを入れないようにしている。

<sup>6</sup>勝手に選んでくれている時もあるが、一応確認しておく。

<sup>7</sup>もちろん、有効にしなくても普通に RStudio は使える。

<sup>8</sup>例: <https://github.com/takeshou/rstudiogit.git>

<sup>9</sup>Mac だと自動では入らない上、うまく行かない場合もある。コレについては、手元に Mac がないので、調べられないが、どうにかしたいので、知っている人はコメントなどで教えていただけると助かります。

### 既存のプロジェクトで Git を有効にする

1. 「Tools」 -> 「Project Options…」 -> 「Git/SVN」を選択
2. 「Version control system:」で「Git」を選択

この場合、リモートの指定は、Git タブの More から、Shell を開いて、以下のように行う。

```
git remote add origin "リモートリポジトリの URL"
```

この作業は、最初は複雑に思えるので、できる限り、新規にプロジェクトを作るように考えるとよい。