

To: Jeff Hardy

From: Bruce Marron

Subject: Data Management Architecture, Part 2

Date: July 31, 2018

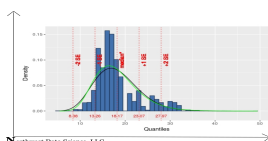
Hi Jeff! This memo documents the proposed front-end of the data management architecture: the ability to create a collection of datasets for subsequent analysis from the total universe of proprietary datasets. The process is depicted in Figure 1 below.

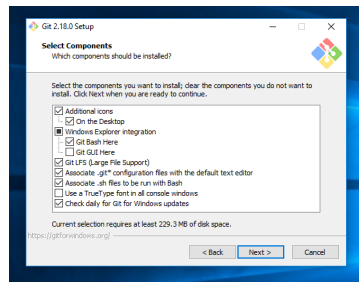
Rather than try to set up a complicated "Library of Congress"-style tracking system for the proprietary datasets, we can use the power that is provided by Linux-based utilities to search both for and within ASCII .csv files. This means you can name an original dataset whatever you like as long as the moniker is unique. We might consider adding a simple date to the .csv file name (in yyyy/mm/dd format). As we know, the total universe of proprietary datasets is expected to grow over time so we want a very flexible system. The proposed system is an analog of the hard copy storage system at the new, ultra-modern James B. Hunt, Jr library at North Carolina State University. Printed materials are literally thrown into giant bins without regard to content. Everything is completely mixed up! It's the sorting algorithm that is used to find and pull a book. Check it out (the actual printed storage facility is at 40 to 51 seconds in the video): library

To make this a possibility, we'll install the "Git for Windows" package in our Windows boxes and supplement the installation with a whole suite of Linux-based utilities obtained from the CygWin collection (<https://cygwin.com/>). Installing "Git for Windows" provides the Git version control system (software) for tracking changes in computer files. The suite of Linux-based utilities from CygWin will make searching, sorting, and collecting .csv files possible with the utmost precision.

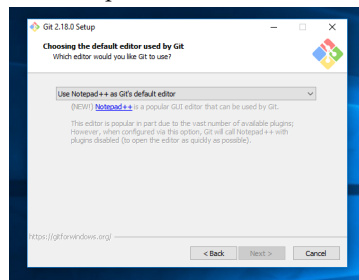
A. Install Git 2.18.0

1. Go to the "Git for Windows" website (left-click here)
 - a. Click on the "Download" button to download, `Git-2.18.0-64-bit.exe`
 - b. Run the `Git-2.18.0-64-bit.exe` installer accepting defaults EXCEPT
 - i. Check daily for Git Windows updates

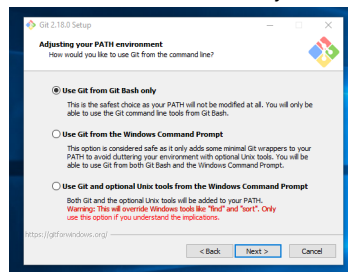




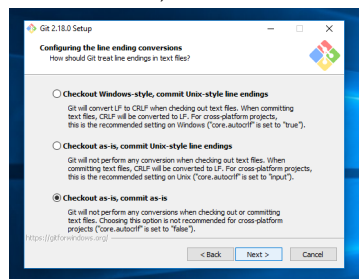
ii. Use Notepad++ as Git's default editor (We'll install Notepad++ in a moment.)



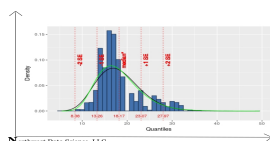
iii. Use Git from Git Bash only

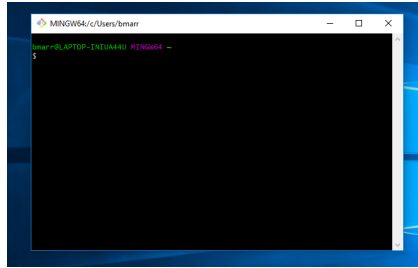


iv. Checkout as-is, commit as-is



- c. Upon completion of the installation a GitBash icon should appear on your Desktop. If convenient you can leave it there or pin it to your Taskbar.
- d. Double-click on the icon to open Git Bash
 - i. A new, command line window will appear





- e. We will use this interface to run the Git version control system and to rapidly find proprietary datasets.

B. Supplement Git Bash with Linux utilities

1. Unpack the `LinuxUtilities_Windows.zip` file
2. Use File Explorer in Windows to locate the `\bin` directory of the Git installation (`C:\Program Files\Git\bin`)
3. Copy and paste all of the individual files from `LinuxUtilities_Windows.zip` into `C:\Program Files\Git\bin`. Note that these are mostly `.exe` files.

C. Install Notepad++ 7.5.8

1. Go to the Notepad++ website (left-click here)
2. Scroll down until you find the link, "Notepad++ Installer 64-bit x64"
3. Click on the link to download the Notepad++ installer, `npp.7.5.8.Installer.x64.exe`
4. Install accepting all defaults

