Source Code Management using Git and GitHub
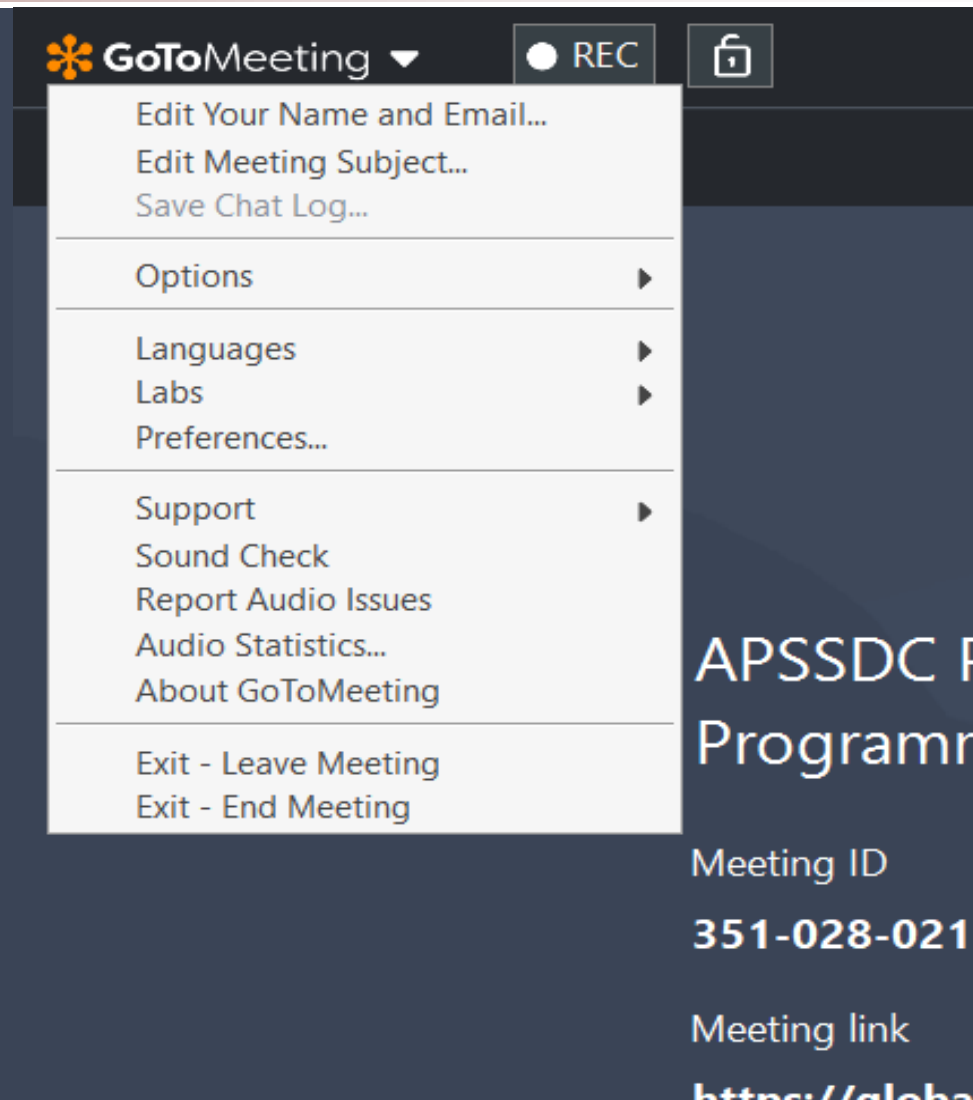
A distributed version control system

For Attendance and Verification Purpose

RollNo-Name-College-Code/
CollegeName
And Registered Email
ID

GoToMeeting ▼   ● REC

Edit Your Name and Email...
Edit Meeting Subject...
Save Chat Log...

Options   ▶

Languages   ▶
Labs   ▶
Preferences...

Support   ▶
Sound Check
Report Audio Issues
Audio Statistics...
About GoToMeeting

Exit - Leave Meeting
Exit - End Meeting

APSSDC
Programr

Meeting ID
351-028-021

Meeting link

https://globa
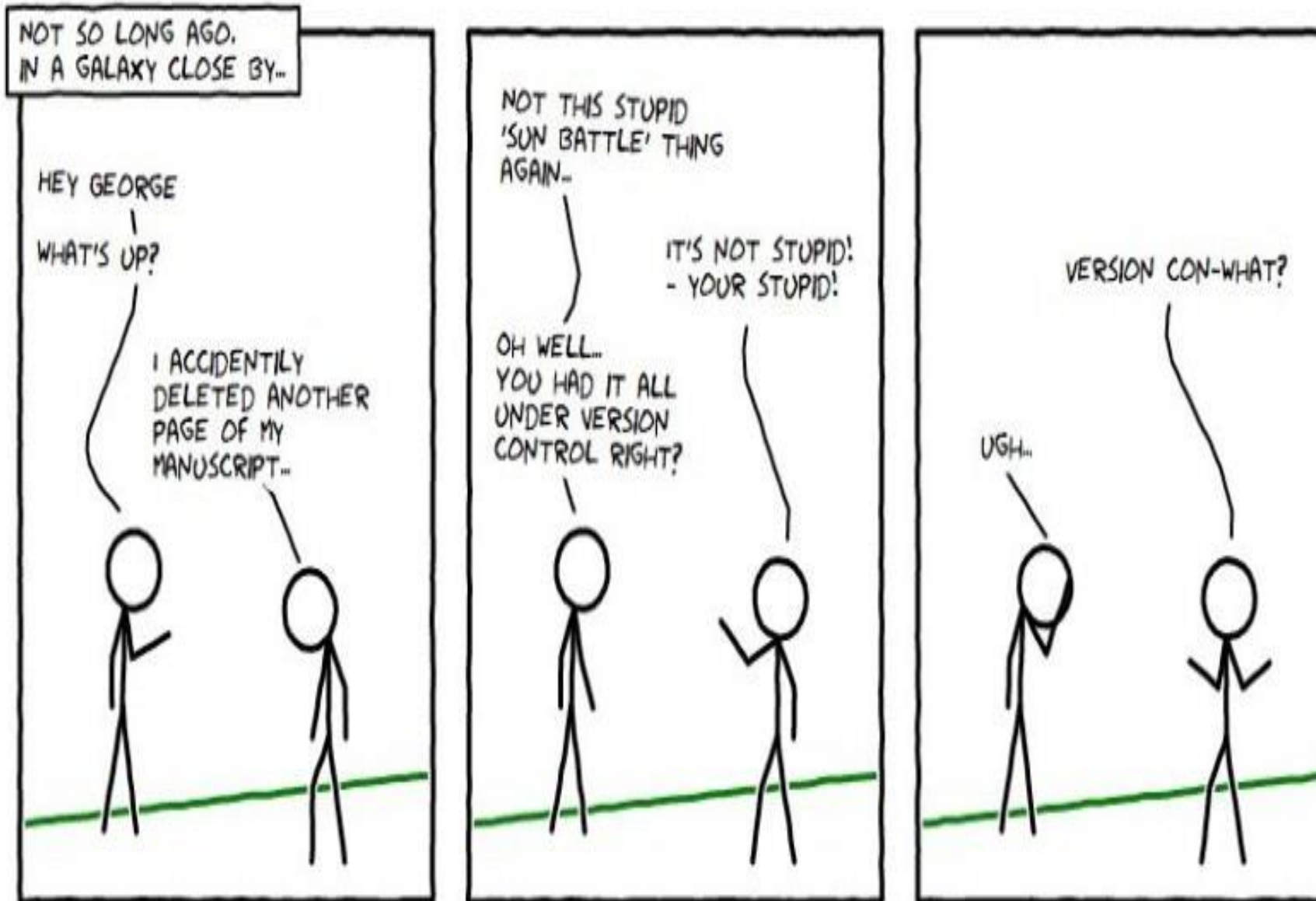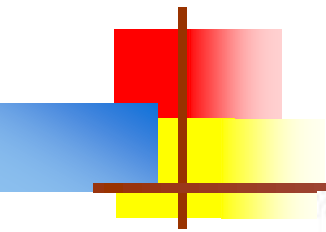
https://bit.ly/scm-git-ebl

Why you have joined in this training program

- I want to learn about git and github in detail

- to know about github and some techniques

- to learn smtg new

- I want to know about github and will be useful for my jobs

- free of cost and also git and github is important

- recently i am working on project our faculty told to post in github but i dont know about actually github

- SIR I AM FAMILIAR WITH GITHUB, I HAVE KNOWN ABOUT GITHUB FROM MY LECTURER...SO TO KNOW ABOUT THI I AM HERE

- Recently working on academic project we discuss among them by using git

- GitHub is a code hosting platform for version control and collaboration. It lets you and others work together on projects from anywhere

- One App build by many developer on many machines finally to one appl
- 1 Year -> 365 days
  - Some errors
  - I got x - 5 till yesterday need x but today I got y
    - 1000 half perfect today 100 lines project collapsed
      - removing yesterday's code
      - somewhat complicated
    - Made a duplicate of yesterdays file and modified the duplicated -> 365
      - 1GB -> 2GB -> 3Gb -> 365
      - 10GB -> 3650
  - Unexpectedly deleted the app

Distributed Revision Control and Source Code Management (SCM)

Created by Linus Torvalds, creator of Linux, in 2005

– Came out of Linux development community

– Designed to do version control on Linux kernel

Goals of Git:

- Speed

- Support for non-linear development (thousands of parallel branches)

- Fully distributed

- Able to handle large projects efficiently

Version control systems

- Version control (or revision control, or source control) is all about managing multiple versions of documents, programs, web sites, etc.
    - Almost all "real" projects use some kind of version control
    - Essential for team projects, but also very useful for individual projects
- Some well-known version control systems are CVS, Subversion, Mercurial, and Git
    - CVS and Subversion use a "central" repository; users "check out" files, work on them, and "check them in"
    - Mercurial and Git treat all repositories as equal
- Distributed systems like Mercurial and Git are newer and are gradually replacing centralized systems like CVS and Subversion

Why version control?

- For working by yourself:
  - Gives you a "time machine" for going back to earlier versions
  - Gives you great support for different versions (standalone, web app, etc.) of the same basic project
- For working with others:
  - Greatly simplifies concurrent work, merging changes
- For getting an internship or job:
  - Any company with a clue uses some kind of version control
  - Companies without a clue are bad places to work

Why Git?

- Git has many advantages over earlier systems such as CVS and Subversion

    - More efficient, better workflow, etc.

    - See the literature for an extensive list of reasons

    - Of course, there are always those who disagree
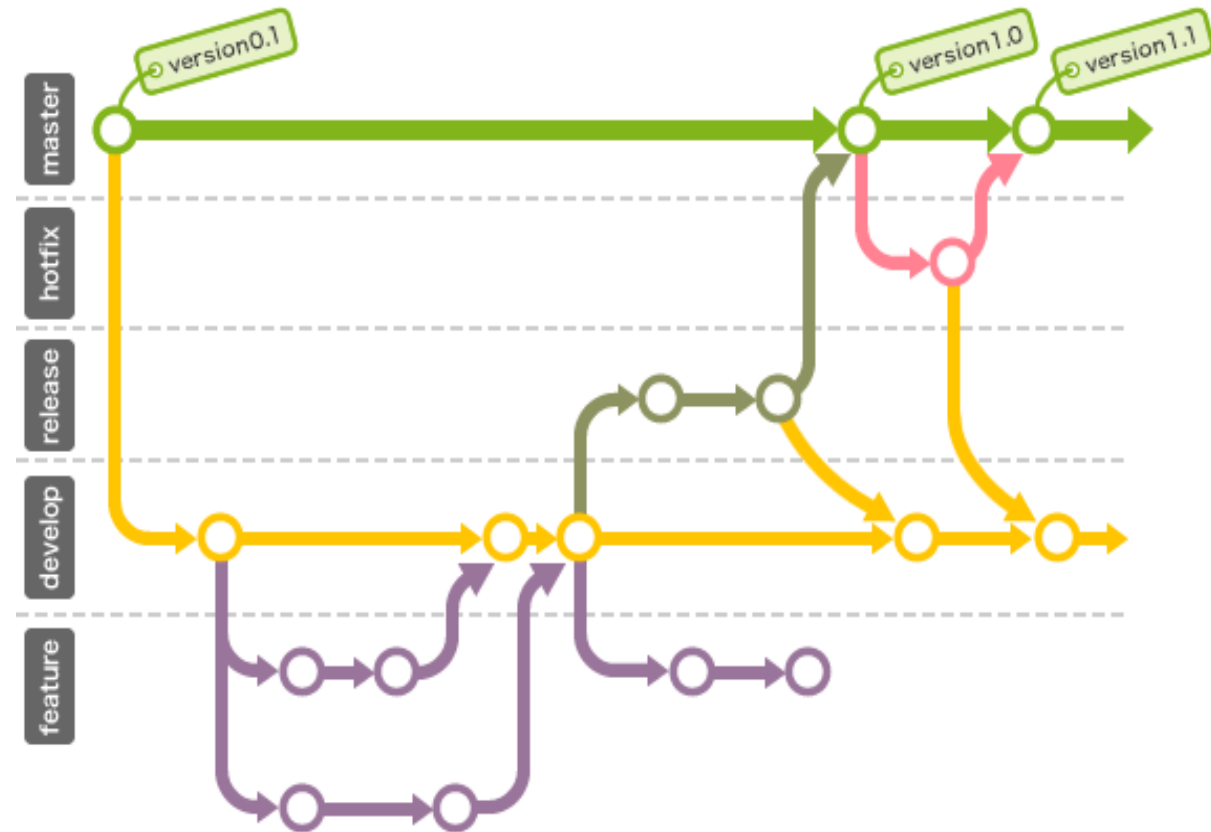
Git as Version Control

Git Terminology

Repository - Your project directory

- Local
- Remote

Branch - A version of your repository

Commit - A checkpoint on a branch

Master Branch - The main branch of your software
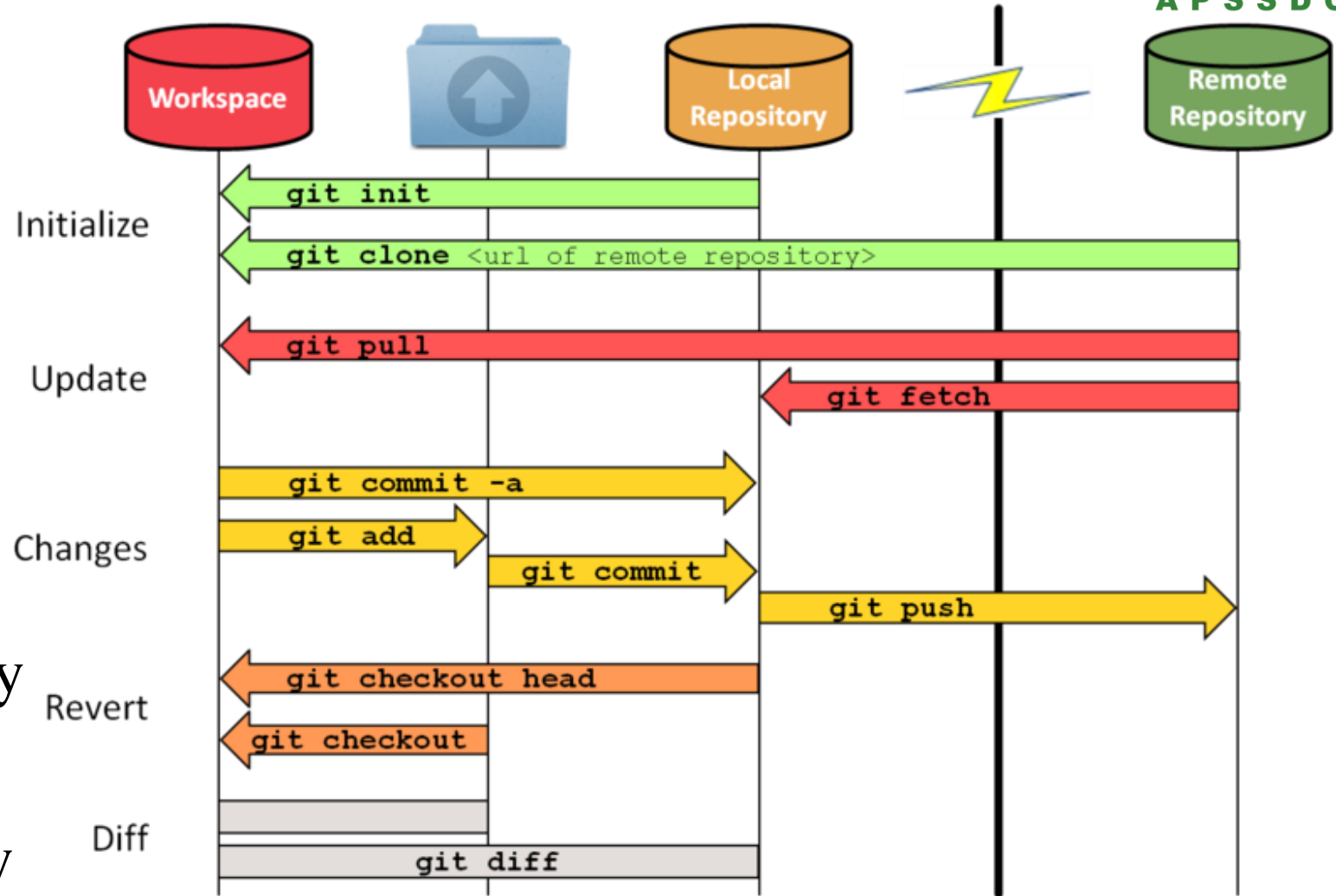
Where is the Code?

The Git process

1. Initialize a Repo

2. Write code

3. Commit to local repository

4. Push to Remote repository

GitHub

Tom Preston Werner (co-founder)

Werner is also the creator of Gravatar - the traveling profile image site
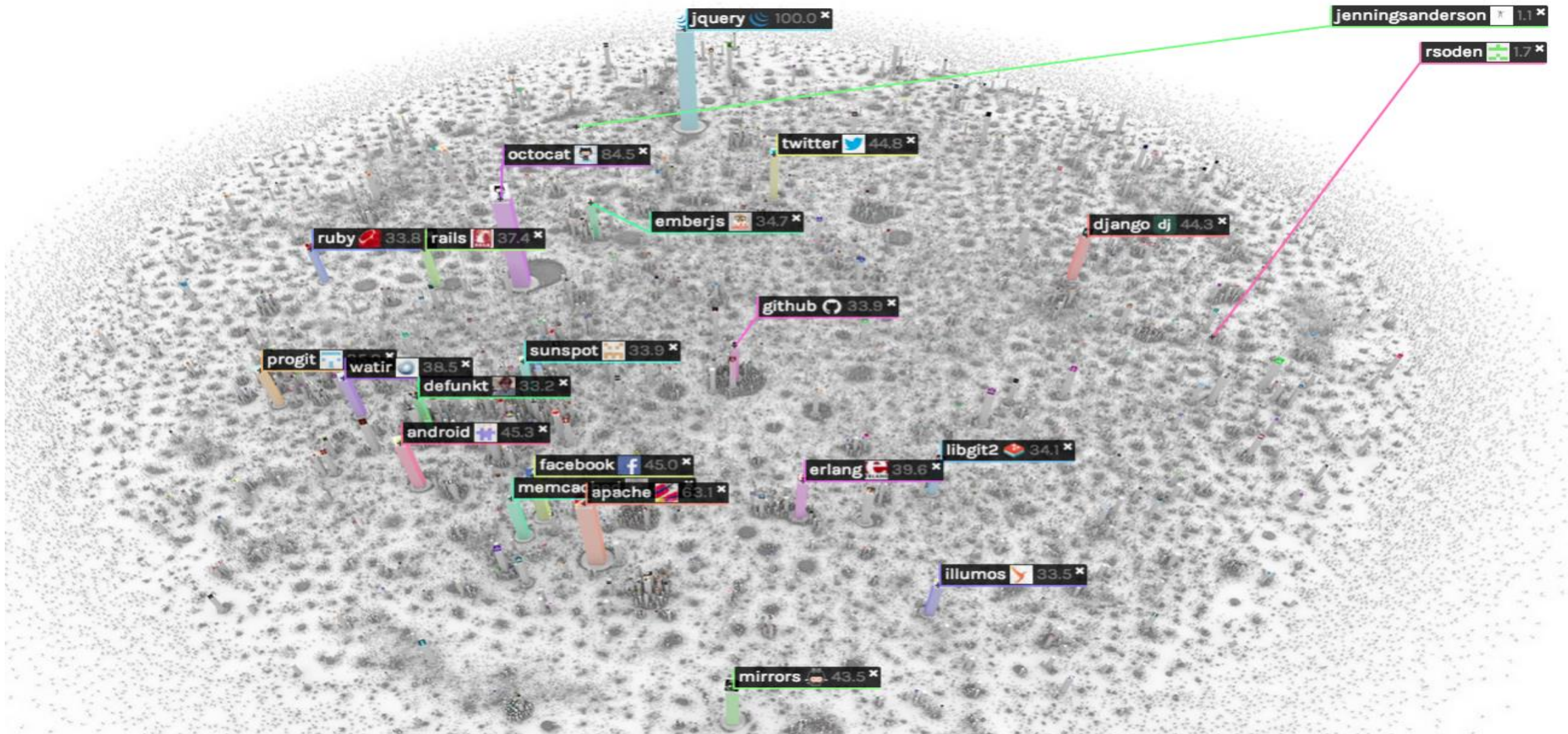
GitHub is now BIG - $750 million big

Git & GitHub

Are

- Free & Open

- Becoming Industry Standards

- Collaborative Tools

Are Not

- Code Editing Software (though Github provides limited functionality)

GitHub – Global Community

Projects

- ## Open Source:
  - Source code is open to all
  - Free to use no license required
    - Signal Application – blue color -> pink, brown
  - Secure

- ## Non – Open Source
  - Source code is not open to all
  - Few Free/paid to use no license required
  - Secure/ non secure

GitHub is a hub of remote git repositories

Forking – Making your own copy of the repository

Pull Request – Asking the repository owner to merge your changes

Download and install Git

- There are online materials that are better than any that I could provide

- Here's the standard one:
  http://git-scm.com/downloads

- Here's one from StackExchange:
  http://stackoverflow.com/questions/315911/git-for-beginners-the-definitive-practical-guide#323764

- Note: Git is primarily a command-line tool

- Prefer GUIs over command-line tools, but…

- The GIT GUIs are more trouble than they are worth

By Anil Kumar APSSDC