Salvatore Rinzivillo

VISUAL ANALYTICS

DEVELOPMENT FRAMEWORK

Node.js and NPM

What is Node.js

- "An asynchronous event driven Javascript runtime"
  - Non-blocking, event-driven I/O operations
  - Lightweight and efficient for data-intensive applications
  - Distributed computation and load balancing
- Available for download at https://nodejs.org/

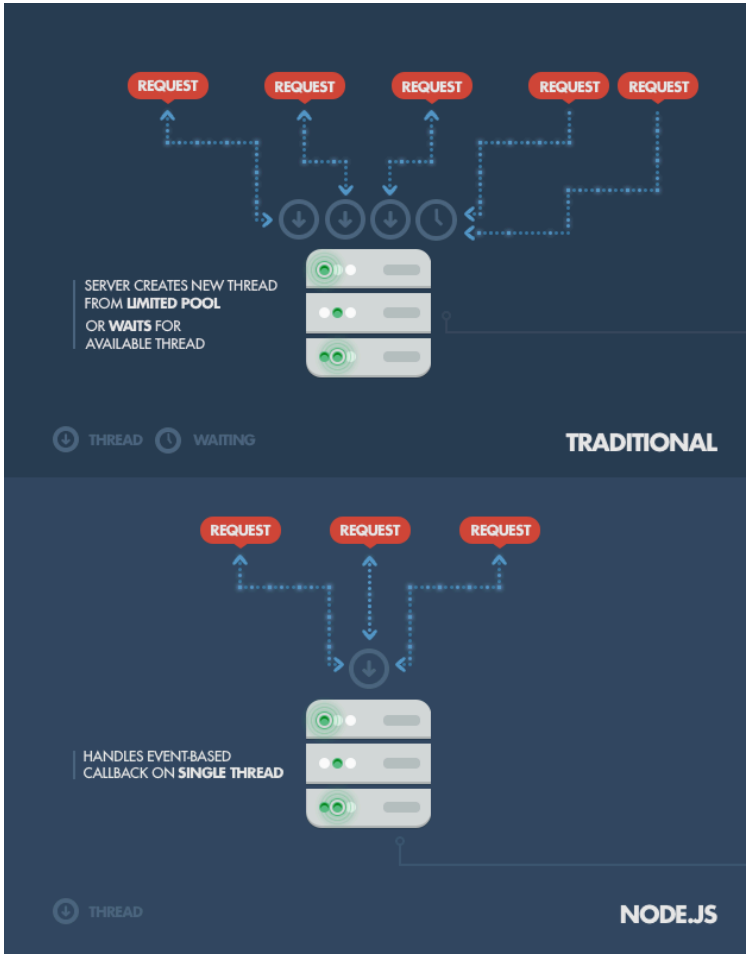


Image source: https://www.toptal.com/nodejs/why-the-hell-would-i-use-node-js

NPM – Node Package Manager

- Node.js has a large library of public available, reusable components
- Components are available through a repository
- Manage libraries for global use and local projects
- Handle all dependencies

NPM - Commands

- `npm init`
  - Initialize a project, creating a file `package.json`
- `npm install <module>`
  - Download and install module within the directory `node_modules`
  - With the flag `--save`, add the module to the `package.json` list of dependencies
  - With the flag `--global` (or `-g`) the module is installed globally on the system

Most used packages

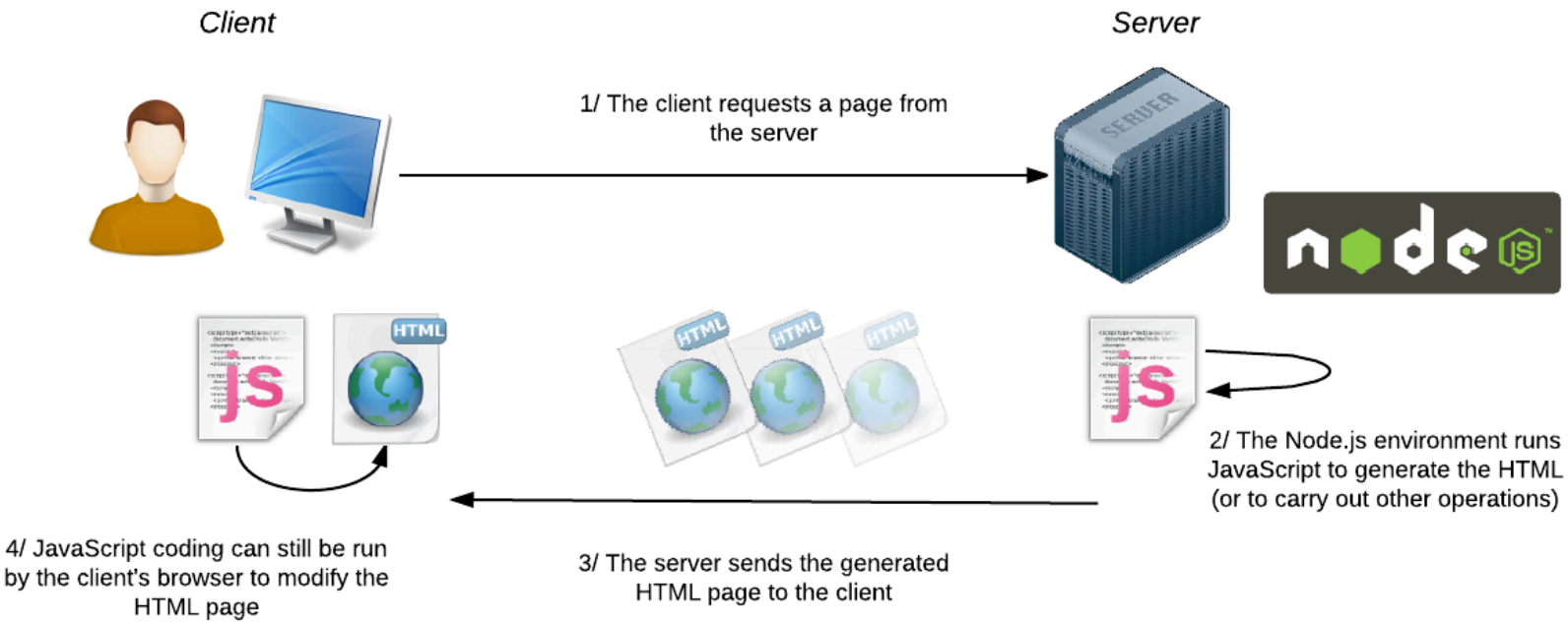
- Express: a web application development framework for node.js
- Lodash: general utilities for handling data structures in javascript
- http-server
- Specifically for the course:
  - D3
  - Nvd3
  - Bootstrap
  - Jquery
  - …

Exercise – Create a project with Node.js

- Demo

Web server

Web Server for Node.js



Web Server in Node.js

- There are several modules available for running a web server
- A very simple http server:
  - `npm install -g http-server`
- A more sophisticated application server:
  - `npm install -g express`

Exercise – Use http-server to access our project

- Demo

Version Control with GIT

What is Version Control?

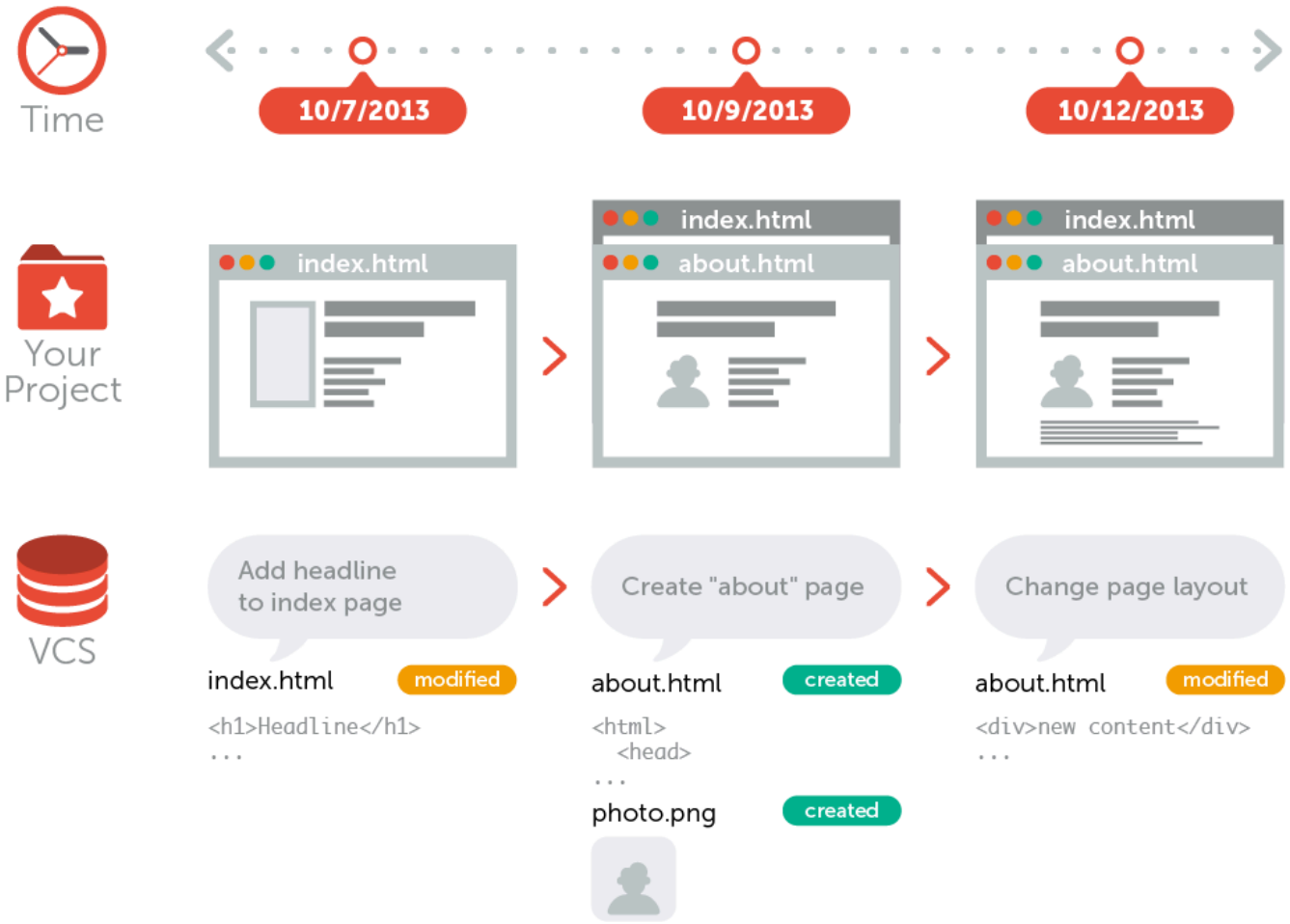


Image Source: https://www.git-tower.com/learn/git/ebook/en/command-line/basics/what-is-version-control

Why Use a Version Control System?

- Collaboration
  - Any member of a team can work on any file at any time
  - Merge of contribution is handle by the VCS
- Storing versions
  - Tracking of changes through periodic saves of snapshots
  - Only one version of a project at any time
    - Other versions are packed within the VCS
- Restore previous versions
- Follow the development of the project
- Backup, when using external repositories

Which VCS? Introducing GIT

- Download a client from public repositories
  - For example: GITHub, BitBucket
- Use clients specific for your OS
  - For example: `brew install git` (for MacOsX)
- Initial configuration
  - `git config --global user.name “rinziv”`
  - `git config --global user.email “rinzivillo@isti.cnr.it”`
  - `git config --global`

GIT – Creating a Repository

- GIT handles two kinds of repositories
  - Local repository
    - Contained within a folder `.git` in the root of the project folder
    - Only on person access this repo
  - Remote repository
    - Located into a remote server
    - Locally stored within the `.git` folder
    - Team members work concurrently on remote repository

GIT – Create a local repository

- Move within the project root directory
- Use `git init` to start versioning tracking
- The root of the project is called working copy
  - There is only one working copy at any moment
  - It is possible to update the current working copy with previous versions from the repository
- Some files (usually related to the OS) can be ignored for the versioning
  - Create a file called `.gitignore` in the root of the project folder
  - List the files to ignore within the file

GIT – Clone a remote repository

- A remote repository have a URL of the form:
  - ssh://user@server/git-repo.git
  - user@server:git-repo.git
  - http://example.com/git-repo.git
  - https://example.com/git-repo.git
  - git://example.com/git-repo.git

GIT - Commit

- Commit operation save the snapshot of the working copy on the repository
  - `git add –A`
  - `git commit –m “Initial commit”`

GIT – Status of the project

- Each file within the project have one of the following state
  - Untracked: the file is not under version control. GIT do not track any change on this file
  - Tracked: GIT reports changes on these files
- `git status` reports the list of files within the project that have changed and those that are not tracked

GIT – Staging Area

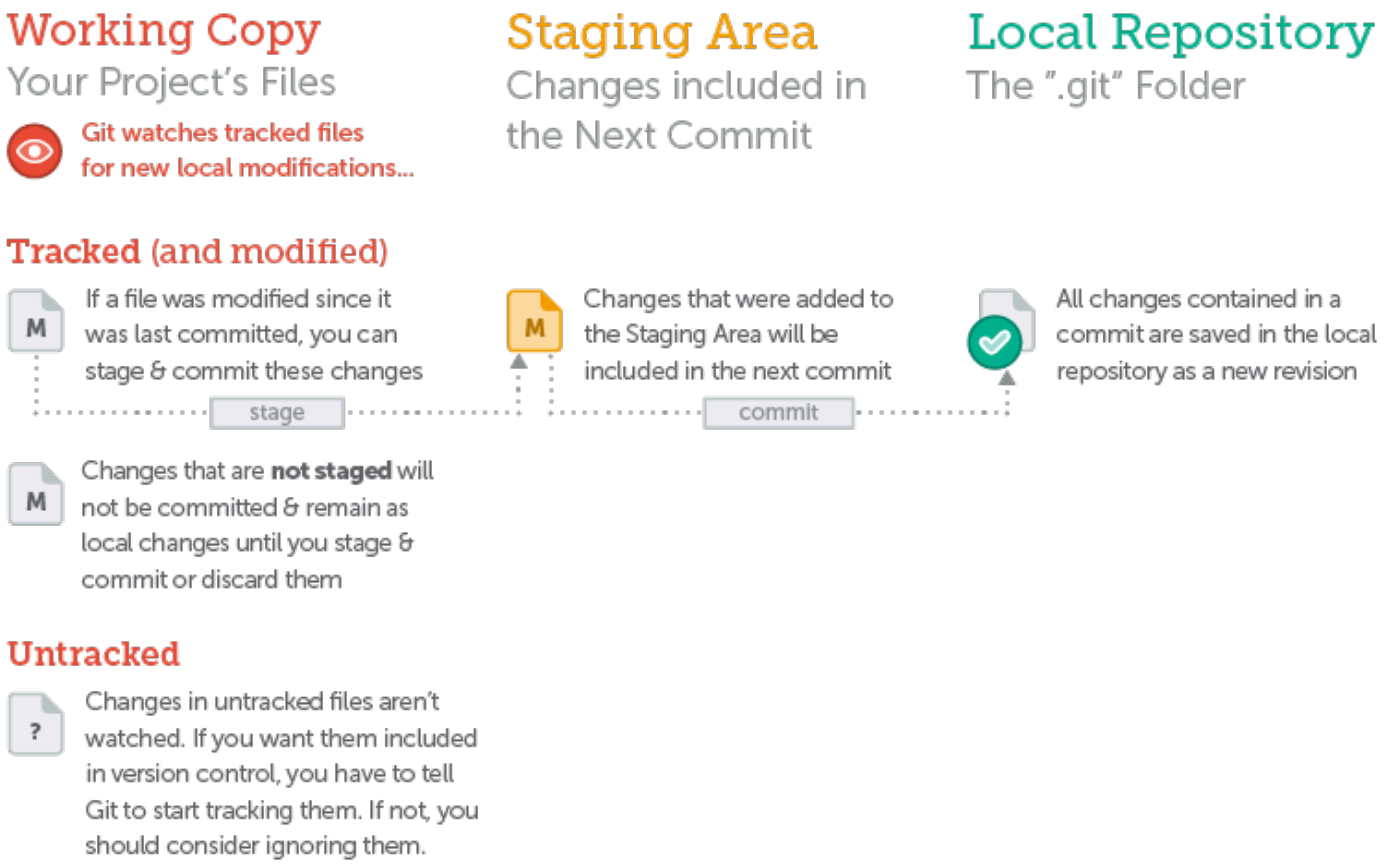


Image Source: https://www.git-tower.com/learn/git/ebook/en/command-line/basics/working-on-your-project#start

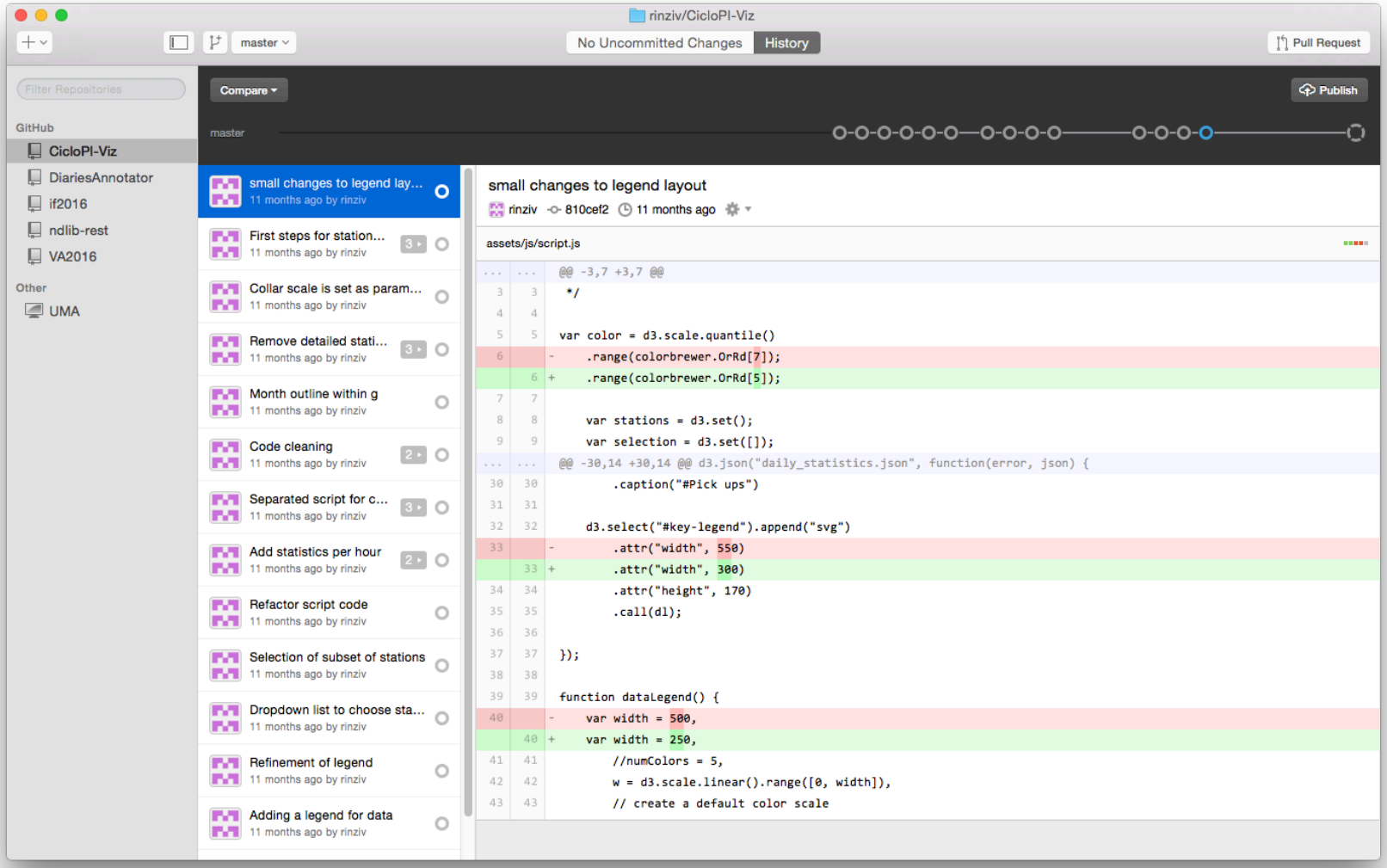
GIT – Preparing the staging area and commit

- Add the files to include in the staging area
  - git add new-page.html index.html css/*
  - git rm error.html
  - Check the status with: git status
- Commit changes
  - git commit -m "Implement the new login box”

When to commit?

- Each commit should contains changes related to a single topic
- Save only completed work (for temporary saving use Stash)
- Test the project before commiting
- Add descriptive message
- Commit often

Desktop GUI Version



Exercise – Create a repository for our project

- Demo