

# Version Control System GIT

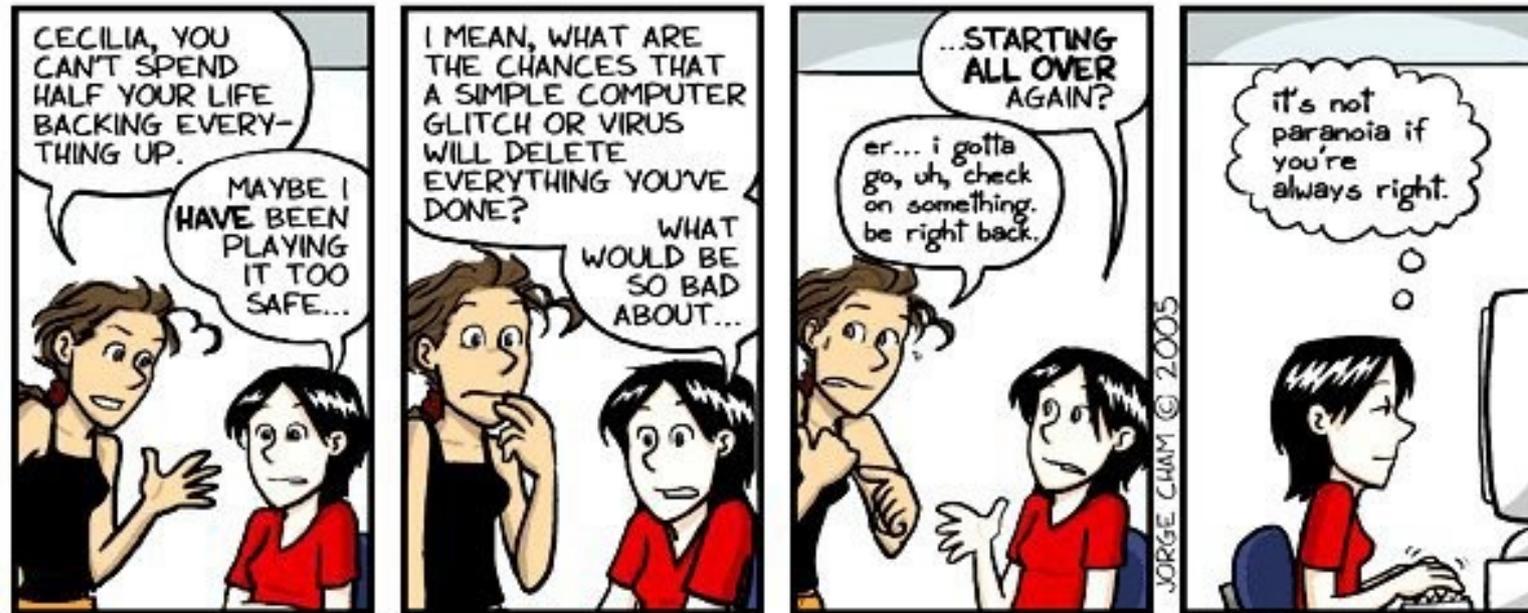


# Version Control System

---

- Version (revision) control systems are software that help you track changes you make in your code over time. As you edit to your code, you tell the version control system to take a snapshot of your files. The version control system saves that snapshot permanently so you can recall it later if you need it.
- Automates the storing, retrieval, logging, identification, and merging of revisions (versions).

# Version Control System



www.phdcomics.com

# Version Control System

"FINAL".doc



FINAL.doc!



FINAL\_rev.2.doc



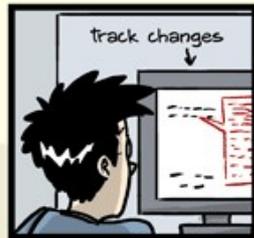
FINAL\_rev.6.COMMENTS.doc



FINAL\_rev.8.comments5.  
CORRECTIONS.doc



JORGE CHAN © 2012



FINAL\_rev.18.comments7.  
corrections9.MORE.30.doc



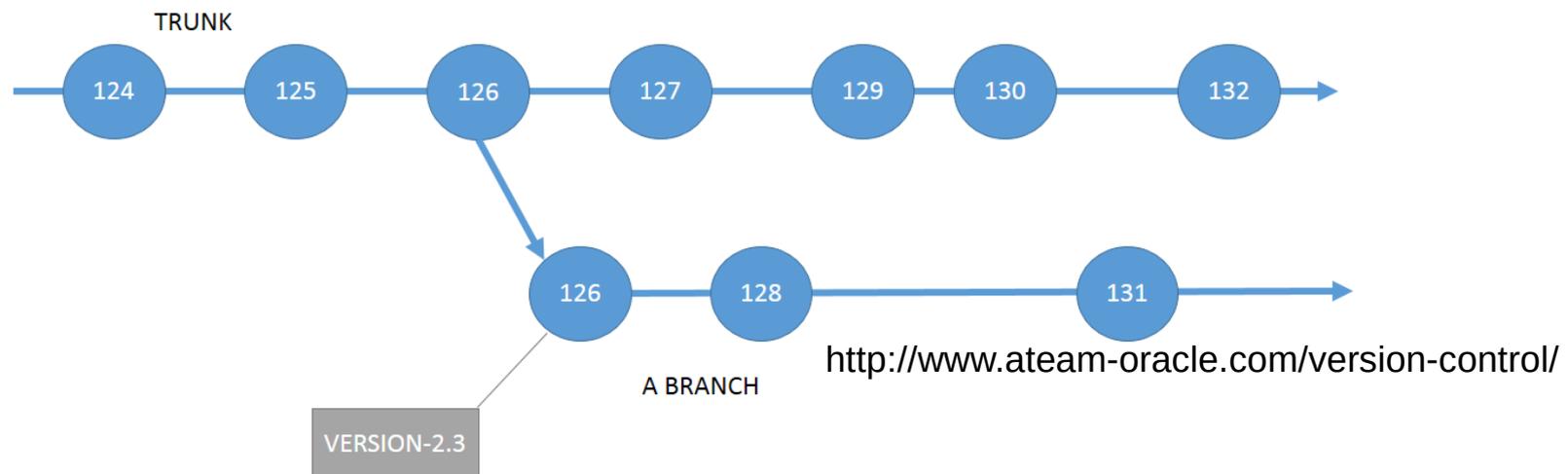
FINAL\_rev.22.comments49.  
corrections.10.#@\$%WHYDID  
ICOMETOGRADSCHOOL?????.doc



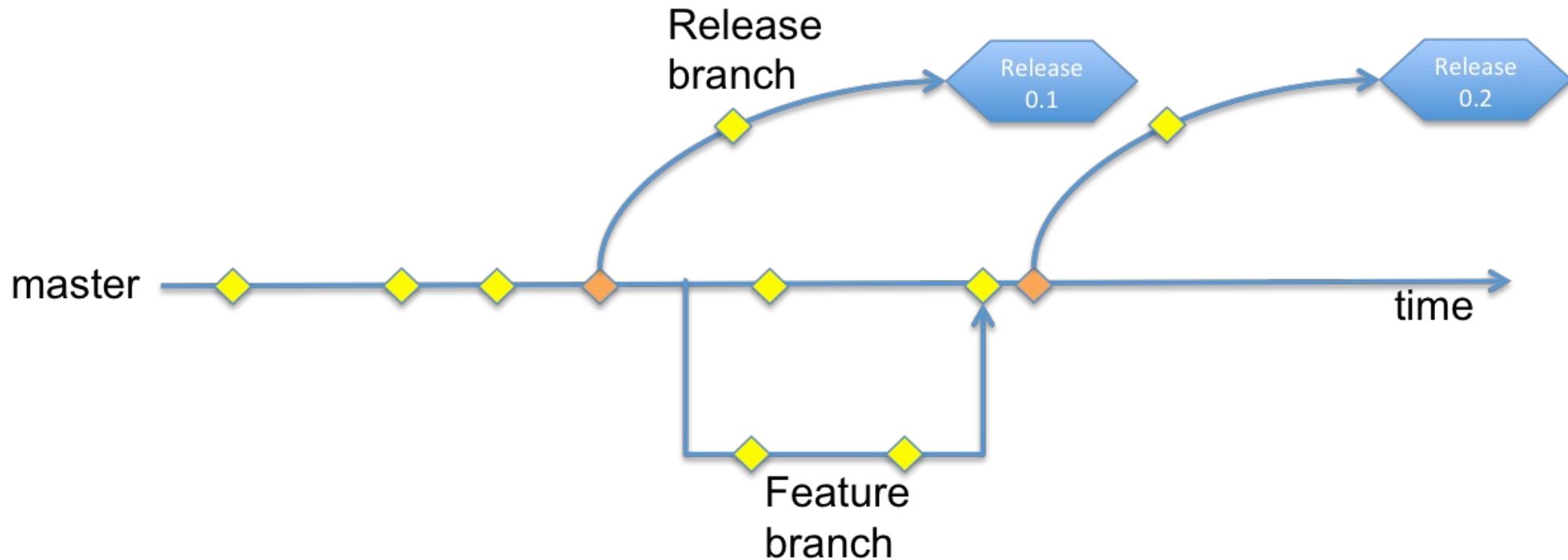
WWW.PHDCOMICS.COM

# Version Control System

- Version Control helps with:
  - Backups
  - Reproducibility
  - Comparing arbitrary historical versions
  - Maintaining multiple live versions
  - Team coding
  - Workflows (process enforcement and permissions)



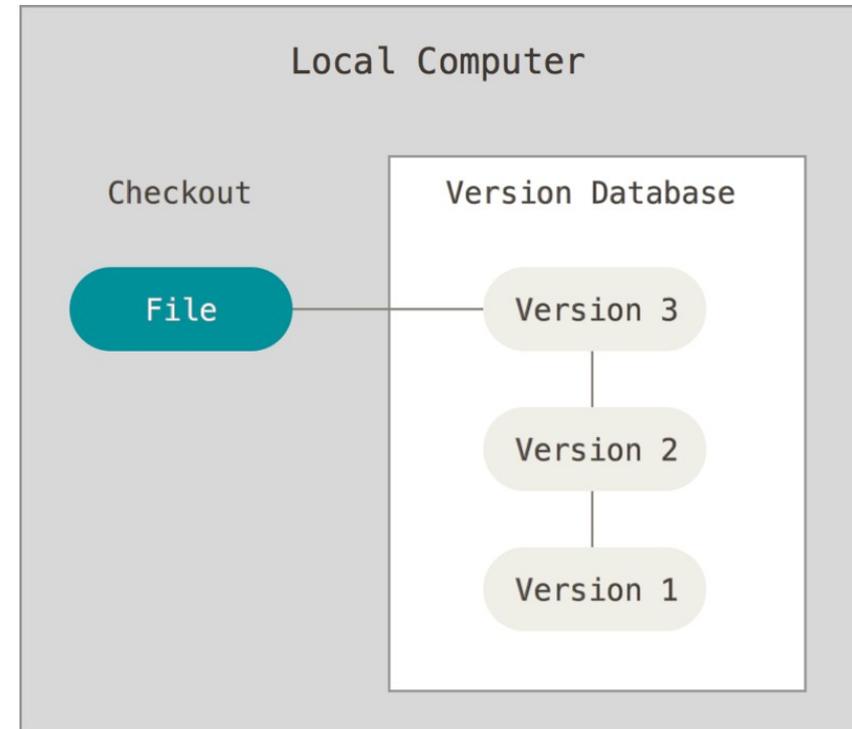
# Version Control System



<http://cswsolutions.com/blog/git-non-developers-part-ii/>

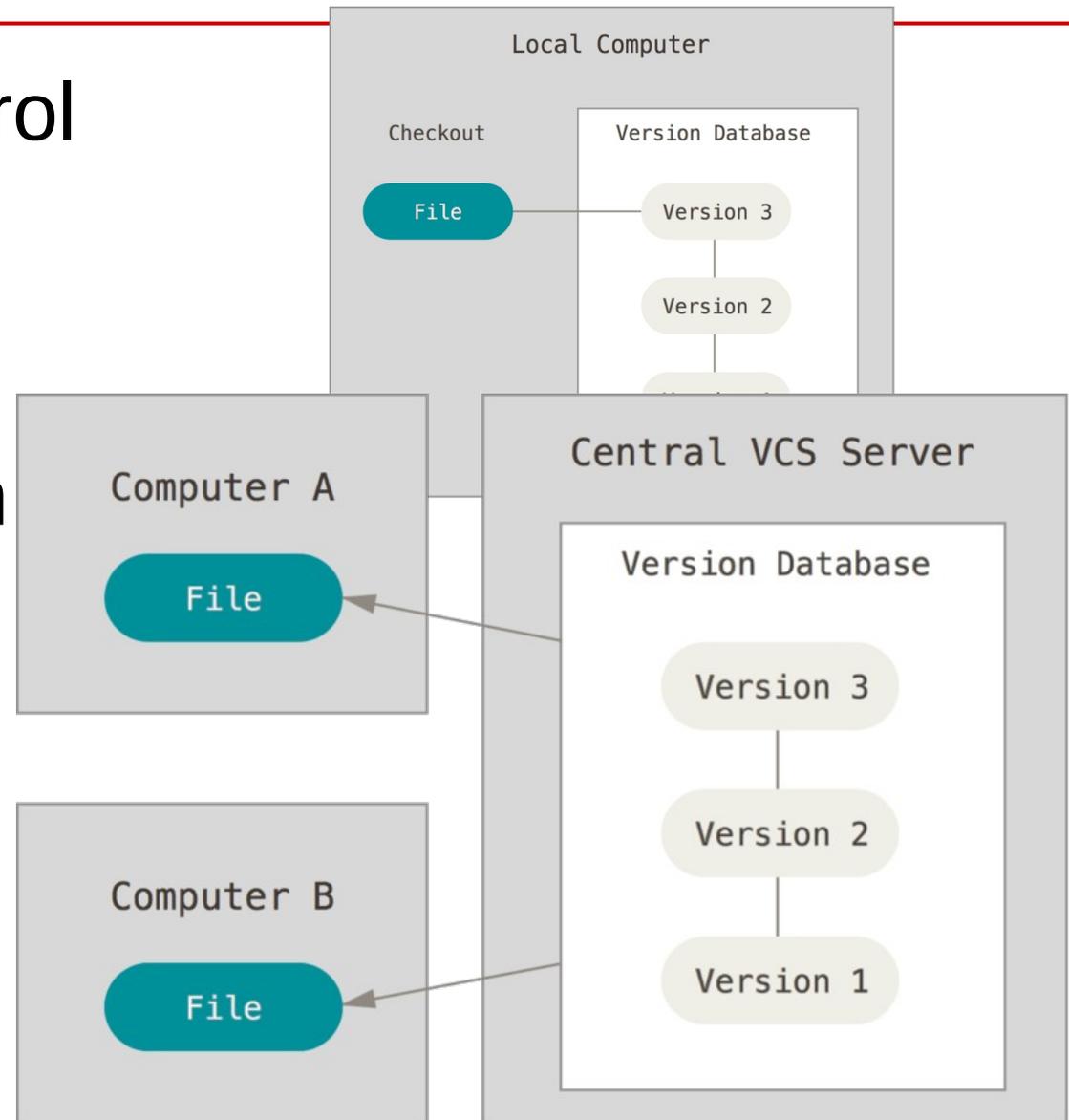
# Version Control

- Local Version Control



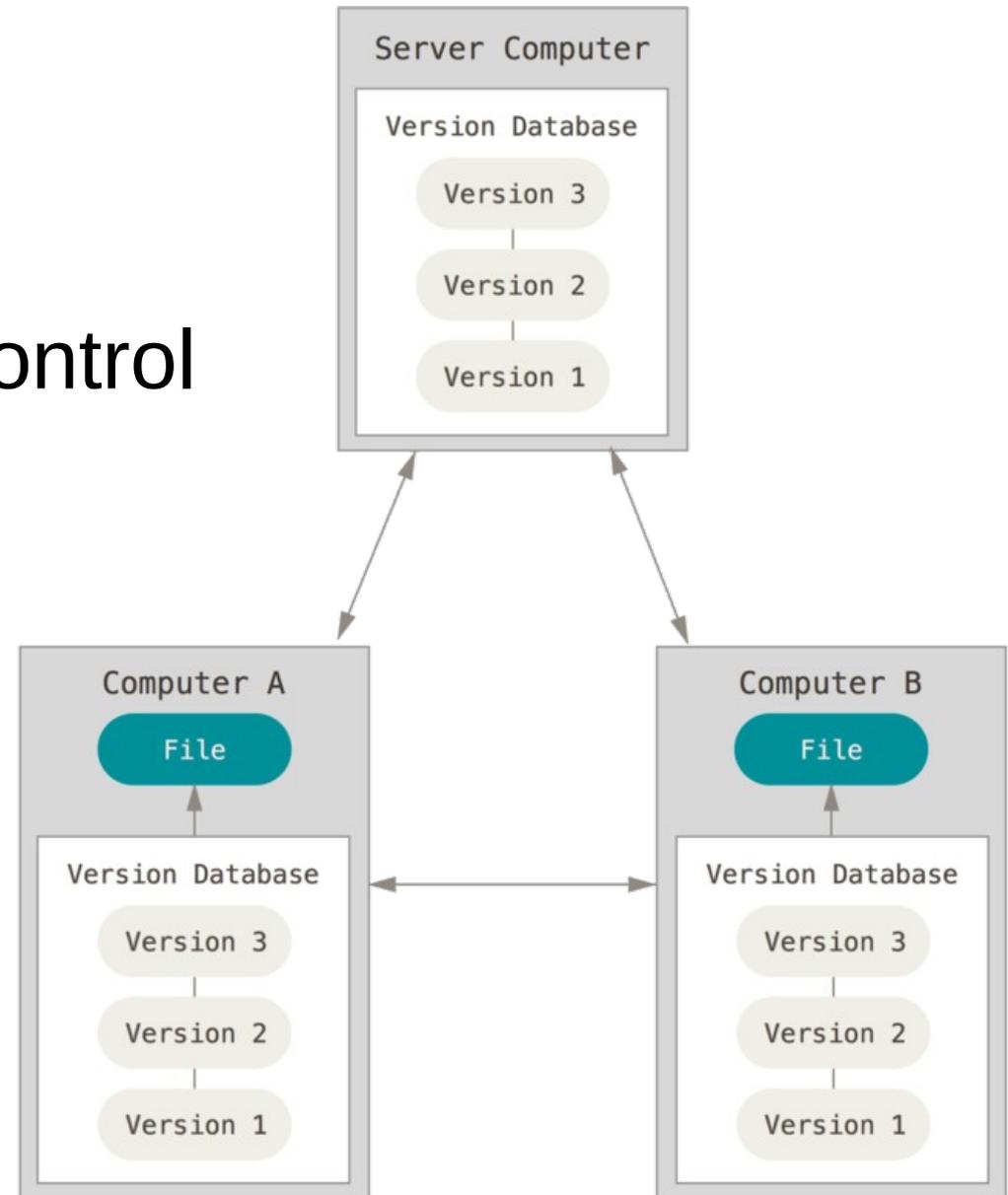
# Version Control

- Local Version Control
- Centralized Version Control
  - Subversion (SVN)



# Version Control

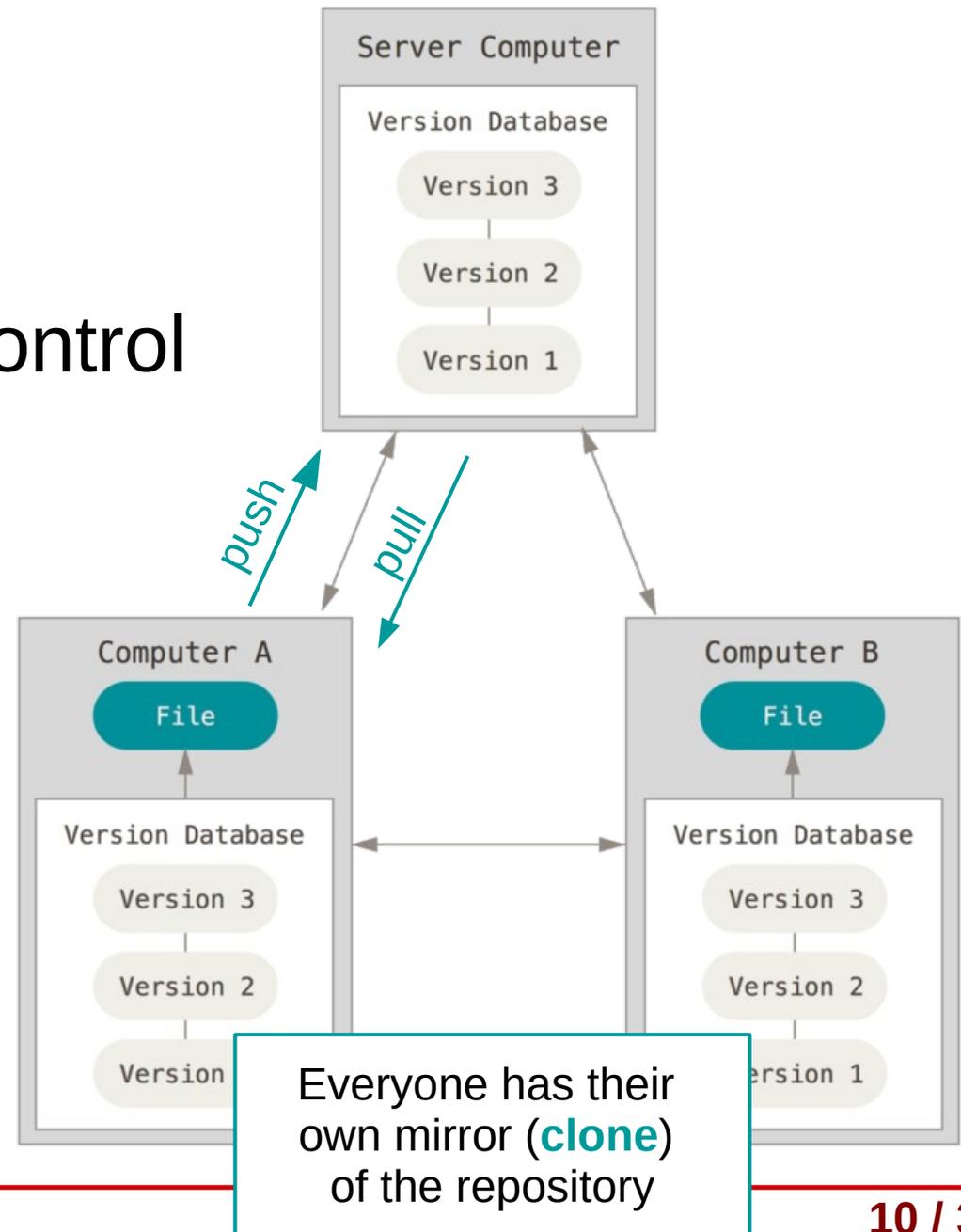
- Local Version Control
- Centralized Version Control
  - Subversion (SVN)
- Distributed Version Control
  - GIT



<https://git-scm.com/book/en/v2/Getting-Started-About-Version-Control>

# Version Control

- Local Version Control
- Centralized Version Control
  - Subversion (SVN)
- Distributed Version Control
  - GIT



---

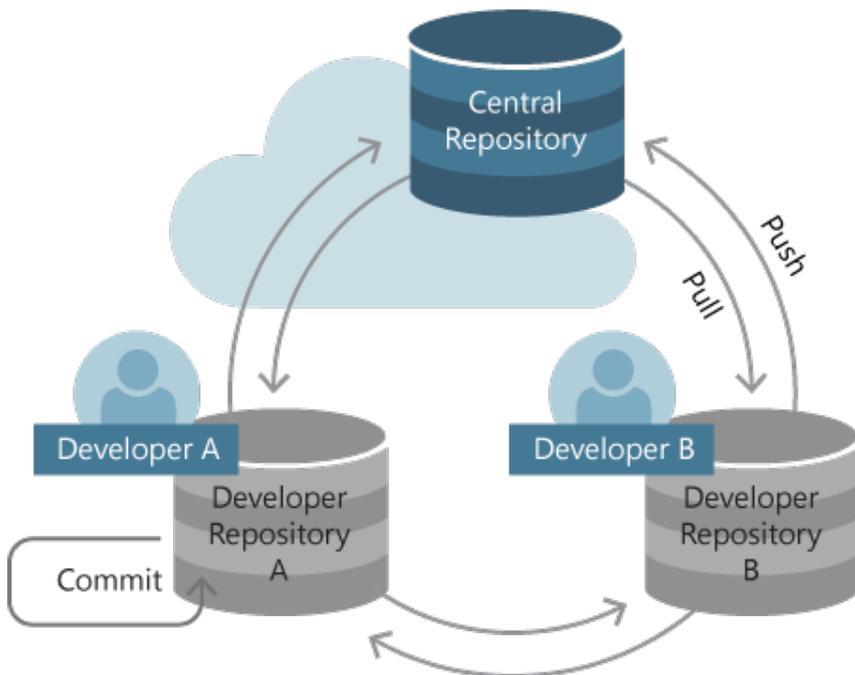
# GIT in Visual Studio



<https://www.visualstudio.com/learn-git/>

# Creating GIT repository

- A Git repository, or repo, is a folder that you've told Git to help you track file changes in.
- A Git repo contains every version of every file saved in the repo. Git saves these files very efficiently, so having a large number of versions doesn't mean that it uses a lot of disk space.



<https://www.visualstudio.com/en-us/docs/git/tutorial/creatingrepo>  
<https://www.visualstudio.com/learn/set-up-a-git-repository/>

# Creating GIT repository

## 1) From a new solution



Name: TypeScriptApp  
Location: C:\work\frank\FabrikamTypeScriptApp  
Solution: Create new solution  
Solution name: FabrikamTypeScriptApp

Create directory for solution  
 Create new Git repository

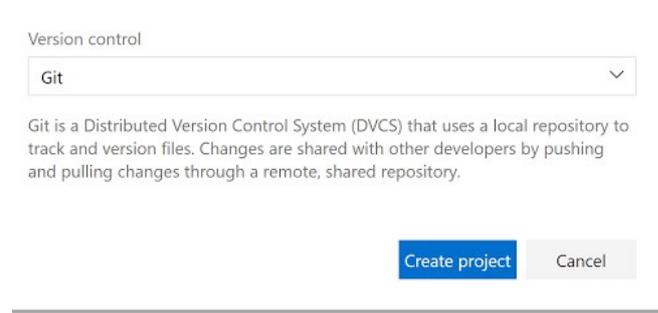
While creating new solution check appropriate checkbox

## 2) From an existing solution

To create a repo from an existing solution, select the **Add to Source Control** button in the bottom-right of the lower status bar Visual Studio publish button. This creates a new Git repo in the same directory as your solution and opens up the Publish view in Team Explorer so you can push your code to Visual Studio Team Services or another remote Git repository.

## 3) Create repo in Team Services

Create New project.  
See next slides.



Version control

Git

Git is a Distributed Version Control System (DVCS) that uses a local repository to track and version files. Changes are shared with other developers by pushing and pulling changes through a remote, shared repository.

Create project Cancel

## 4) In an empty folder

## 5) Connect local repo to a remote

<https://www.visualstudio.com/en-us/docs/git/tutorial/creatingrepo>  
<https://www.visualstudio.com/learn/set-up-a-git-repository/>

# Team Services

Microsoft Azure

SALES 1-800-867-1389 MY ACCOUNT PORTAL Search

Free Account

Documentation Pricing Training Marketplace

Visual Studio Team Services

Share code, track work, and ship software for any language from your IDE.

- ✓ Unlimited free private code repositories
- ✓ Track bugs, work items, feedback, and more
- ✓ Agile planning tools
- ✓ Continuous integration builds

Get started >

Quick Launch (Ctrl+Q)

Sign in

Solution Explorer

Search Solution Explorer (C)

Solution 'ConsoleApplication1' (1 project)

ConsoleApplication1

Visual Studio can sync your settings, help you plan projects, collaborate with your team and manage your code online from anywhere.

Callouts: "Create account through webpage" (pointing to the main heading), "or through Visual Studio" (pointing to the Sign in button), and "yourlogin.visualstudio.com" (pointing to the URL in the second screenshot).

yourlogin.visualstudio.com

Welcome back, Małgorzata Janik

Projects My favorites My work items My pull requests

Projects Filter projects and teams New Project

Recent

- Csharp2017\_Lab1\_ConsoleApp
- Csharp2017\_Lab1\_WindowsForms

# Create repo in Team Services

- Log into Team Services and create New Project

The screenshot shows the Microsoft Team Services web interface. At the top, a blue navigation bar contains the user's name 'majanik', a search bar for work items, and navigation links for 'Projects', 'My favorites', 'My work items', and 'My pull requests'. Below the navigation bar, the main content area displays 'Welcome back, Małgorzata Janik'. A 'Projects' section includes a filter box and a 'New Project' button, which is highlighted with a red box. Below this, a 'Recent' section lists two projects: 'Csharp2017\_Lab1\_ConsoleApp' and 'Csharp2017\_Lab1\_WindowsForms'. The 'Create new project' form is shown on the right, with fields for 'Project name \*', 'Description', 'Version control' (set to 'Git'), and 'Work item process' (set to 'Agile'). The 'Version control' dropdown is highlighted with a red box. At the bottom right, 'Create' and 'Cancel' buttons are visible, with 'Create' highlighted by a red box.

# GIT Task

---

- Using Team Services web account create new GIT repository for today's project
  - You can also create a solution (**Console Application**)
- Create local copy of the repository (Clone) using Team Explorer in Visual Studio (see next slides)
- Apply some changes (ask user to press key to continue), commit them, and push to the repository (see next slides)

# Git repository hosting services

Personal Open source Business Explore Pricing Blog Support Search GitHub Sign in Sign up

## How people build software

Millions of developers use GitHub to build personal projects, support their businesses, and work together on open source technologies.

Pick a username

Use at least one letter, one numeral, and seven characters.

Sign up for GitHub

By clicking "Sign up for GitHub", you agree to our [terms of service](#) and [privacy policy](#). We'll occasionally send you account related emails.

<https://github.com/>

Atlassian  
**Bitbucket**

Features Integrations Server Data Center

## Code, Manage, Collaborate

Bitbucket is *the* Git solution for professional teams

Get started for free

<https://bitbucket.org/>

Host it yourself with Bitbucket Server

GitLab Download Features Products Community Explore Docs Blog Contact Sign In Sign Up

Get paid to write for GitLab! Join the Community Writers Program!

Tools for modern developers  
GitLab unifies issues, code review, CI and CD into a single UI

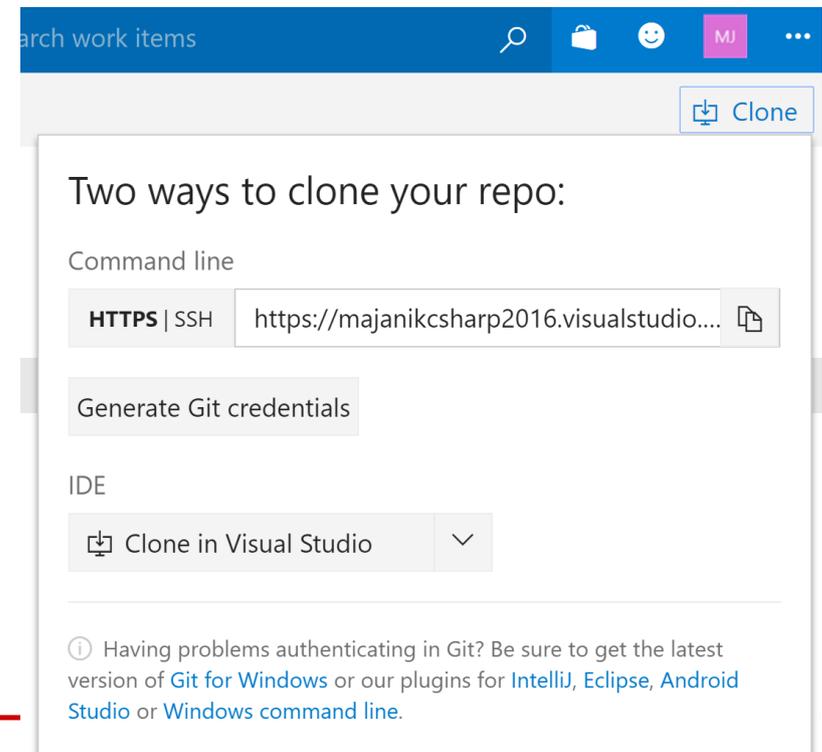
IDEA → ISSUE → PLAN → CODE → COMMIT → TEST → REVIEW → STAGING → PRODUCTION → FEEDBACK

View Features Explore Products

<https://gitlab.com/>

# Creating local repository

- Create a complete local copy of an existing Git repo by cloning it.
- 1<sup>st</sup> step: obtain URL of the source of your repo
  - In Team Services it can be found via web interface
  - GitHub / other git providers: similar, there will be „Clone” or „Download” button



# Clone URL



This repository Search

Pull requests Issues Gist



CALMmodel / CALM

Watch 1

Star 0

Fork 0

Code

Issues 16

Pull requests 0

Projects 0

Wiki

Pulse

Graphs

Settings

ConservAtion Laws Model

Edit

26 commits

3 branches

0 releases

1 contributor

Branch: master

New pull request

Create new file

Upload files

Find file

Clone or download

zaborowska total mass not needed for total energy distribution

cmake/Modules make -> cmake

Bitbucket

atlassianlabs

ACTIONS

Clone

Compare

Fork

NAVIGATION

Overview

Source

Commits

Branches

Pull requests

Issues

Downloads

Pull requests

MERGED CST-1003 → master

Added setting to run events based automation rule in synchronous threads

Overview Commits Activity

Author

Reviewers

Comments (2)

Very nice.

I know we've talked about difficulty writing a functional/unit test. I believe in this case we can test `DefaultAutomationEventHandler.java` and provide a mock `ExecutorService`.

Agreed that we can test if the executor was properly called / skipped based on the setting. You could also create a test Rule that provides a simple call count + reset.

Clone with HTTPS

Use SSH

Use Git or checkout with SVN using the web URL.

https://github.com/CALMmodel/CALM.git

Download ZIP

a year ago

a year ago

# 2<sup>nd</sup> step: clone the repo

- In Team Explorer, open the Connect view.
- Select Clone under Local Git Repositories and enter the URL.
- Select a folder where you want your cloned repo to be kept.
- Select Clone to clone the repo

Local Git Repositories (1)  
New ▾ | Add ▾ | Clone ▾ | View Options ▾

Enter the URL of a Git repo to clone <Required>

C:\Users\majanik\Source\Repos\NewRepo ...

Recursively Clone Submodules

Clone Cancel

Team Explorer - Connect

Connect | Offline

Manage Connections ▾

Hosted Service Providers

Visual Studio Team Services  
Microsoft Corporation

Services to help you ship high quality software. On time, every time. Focus on your code. We'll simplify the rest.

Connect... Get started for free →

Local Git Repositories (2)  
New ▾ | Add ▾ | Clone ▾ | View Options ▾

https://github.com/jamalfab/FabrikamFiber.git

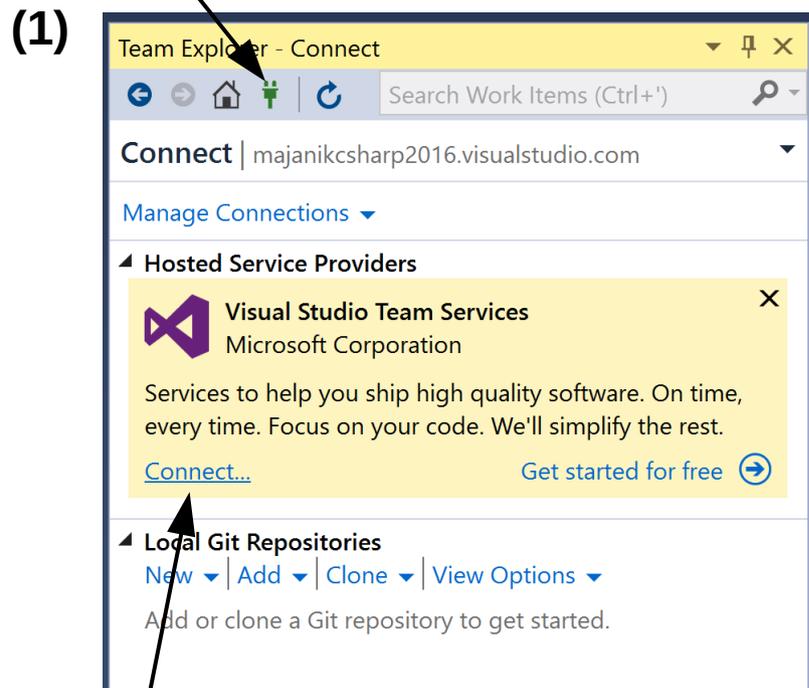
C:\work\jama\FabrikamFiber ...

Recursively Clone Submodules

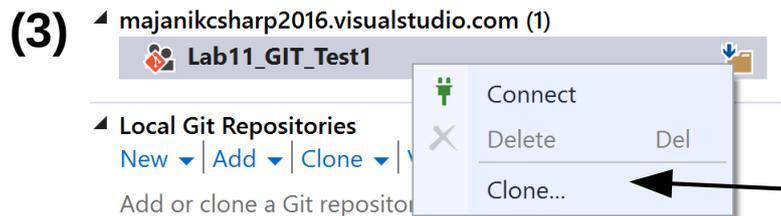
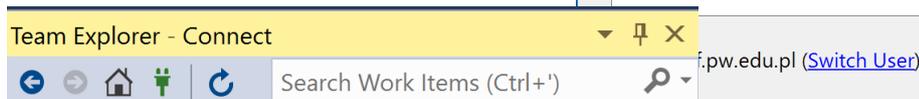
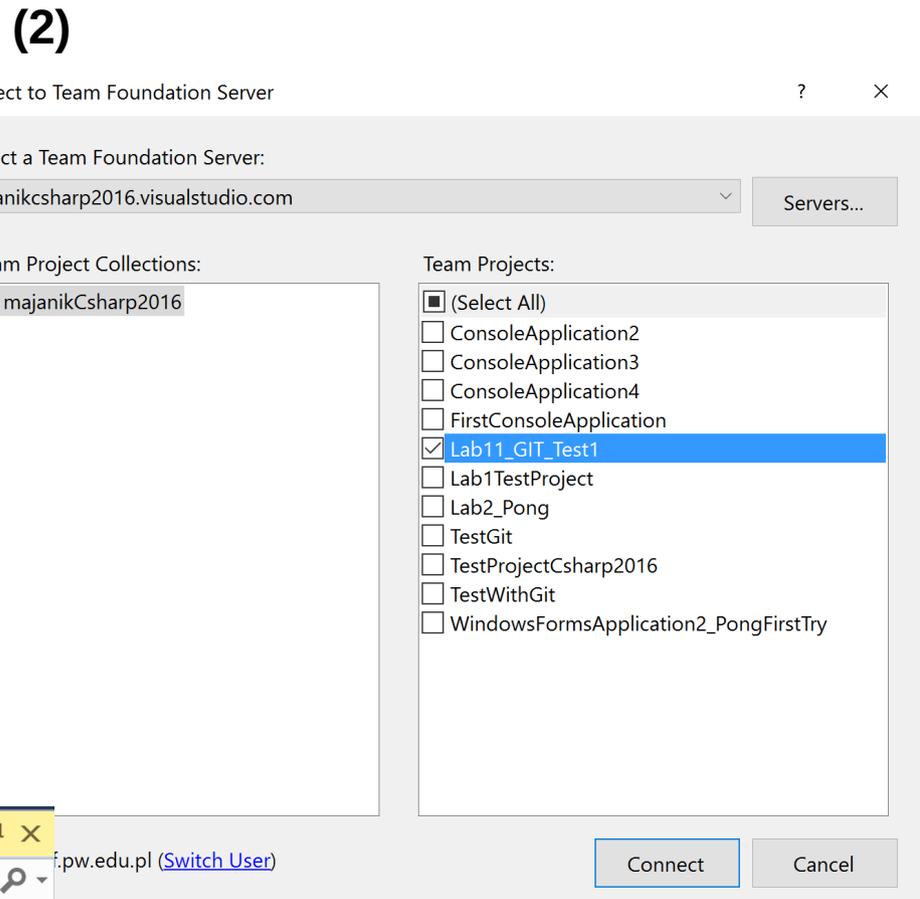
Clone Cancel

# If you use VS Team Services...

## Manage Connections



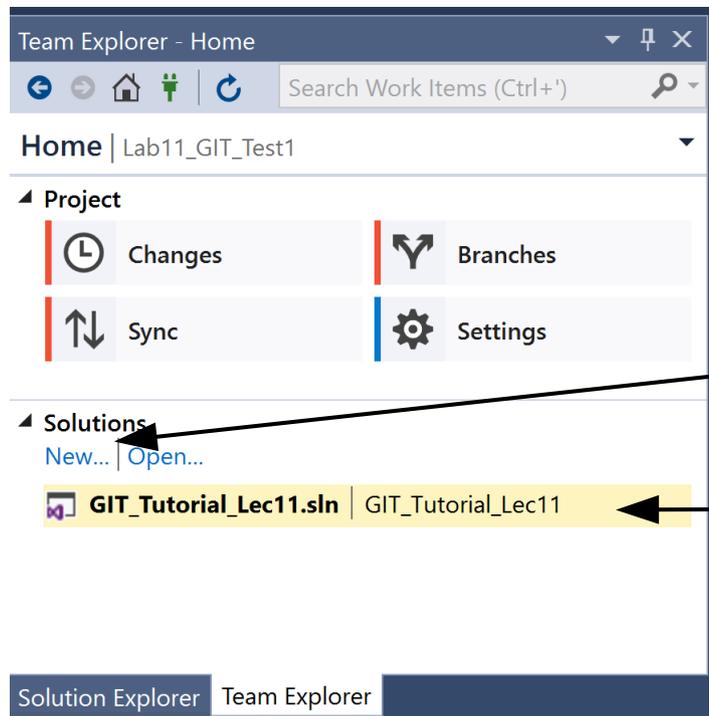
Connect



Clone

# Open Cloned solution

- When the solution is cloned (you have a local copy) you can just double-click it from the Team Explorer



Create new solution if there is nothing on the list

Open solution from git

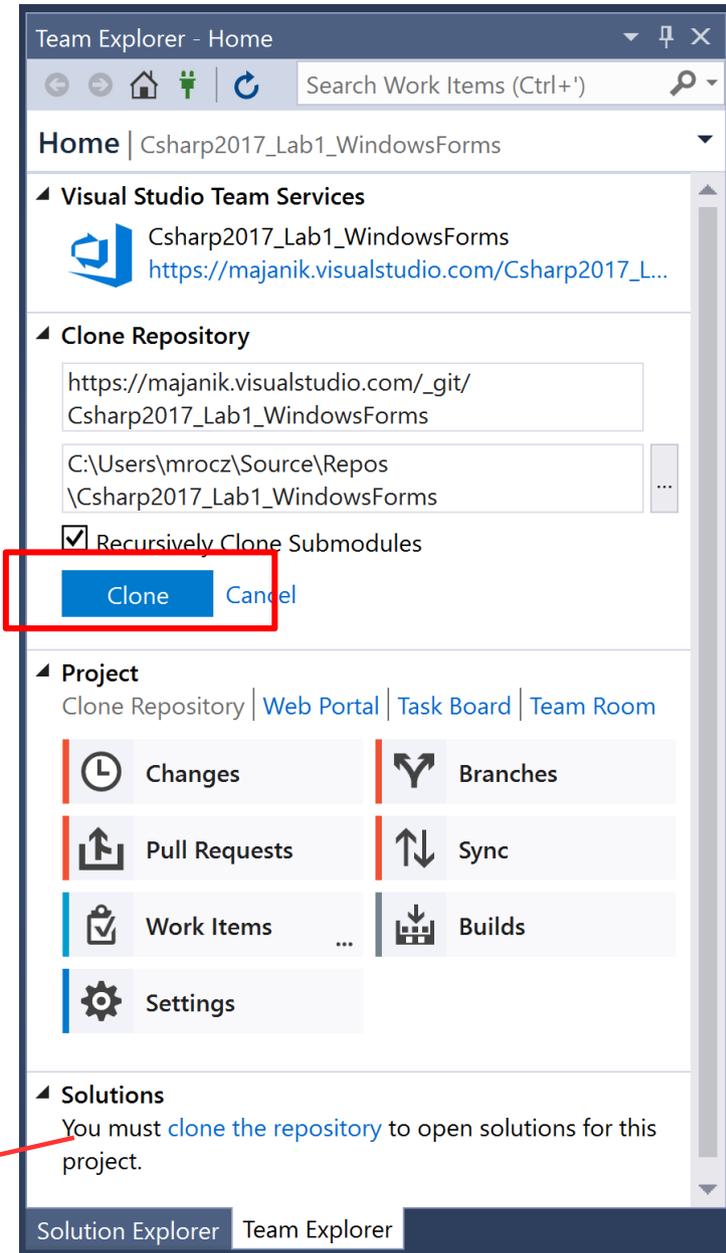
# Alternatively....

- create project in TS
- synchronize it with local repository
  - as on Lab 1

Visual Studio

 [Open in Visual Studio](#)  
Requires Visual Studio 2013+

 [Get Visual Studio](#)  
See Visual Studio downloads



Team Explorer - Home

Search Work Items (Ctrl+') 🔍

Home | Csharp2017\_Lab1\_WindowsForms

Visual Studio Team Services

Csharp2017\_Lab1\_WindowsForms  
[https://majanik.visualstudio.com/Csharp2017\\_L...](https://majanik.visualstudio.com/Csharp2017_L...)

Clone Repository

Recursively Clone Submodules

[Clone](#) [Cancel](#)

Project

Clone Repository | [Web Portal](#) | [Task Board](#) | [Team Room](#)

 Changes |  Branches

 Pull Requests |  Sync

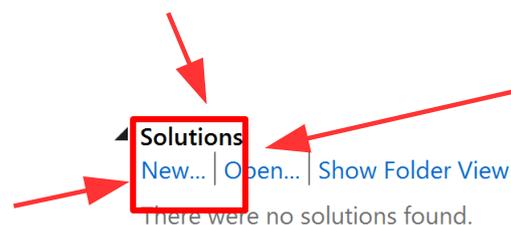
 Work Items ... |  Builds

 Settings

Solutions

You must [clone the repository](#) to open solutions for this project.

Solution Explorer | Team Explorer



[Solutions](#)

[New...](#) | [Open...](#) | [Show Folder View](#)

There were no solutions found.

# Updating the repository

---

## – *sharing your code*

- **Stage changes** (show git which files you would like to update)
- Create a **commit** (save staged files to your repo)
- **Push** code to the repository (update the remote repository = share your code with others)

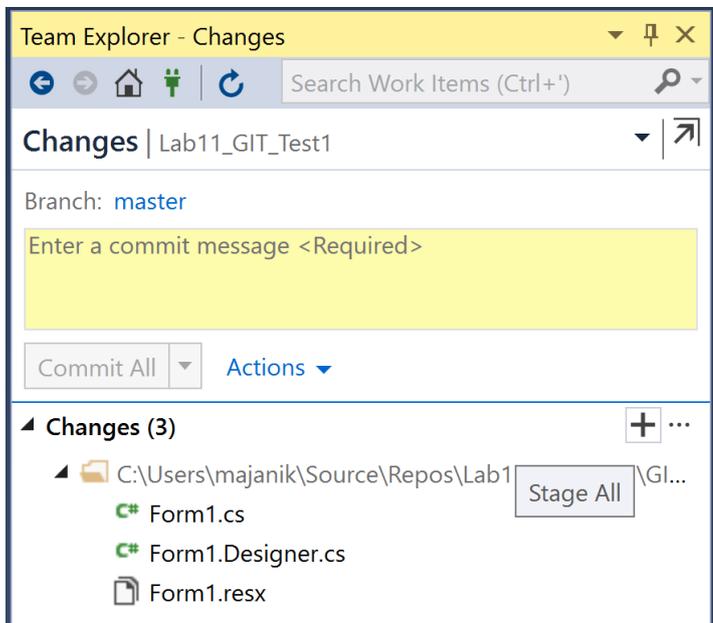
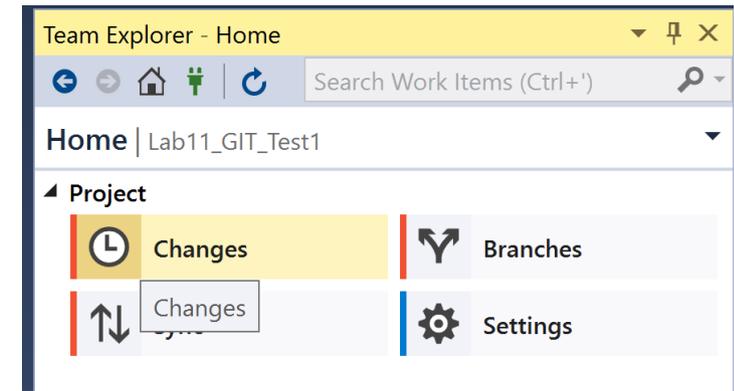
# Commits

---

- What is a commit?
  - Commit saves changes to your local git repo
- Commit consists of:
  - The file(s) changed in the commit (**staged files**)
  - Message describing commit (important!)
  - Reference to previous commit

# Commits via Team Explorer

- Open „Changes” view in Team Explorer
- Stage changes  
(possibility to select specific files)



- Write a commit message and  
***Commit All / Stage Commit***

- If needed, it is possible to update your last commit to correct small errors without making a new commit.  
*Action → Amend Previous Commit.*

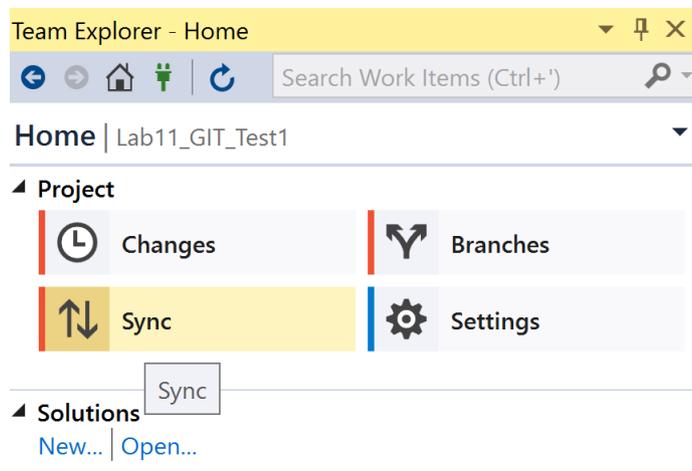
# Share code with push

---

- You can share the changes in commits using **push** command
- This command uploads changes to (**synchronizes**) the central repository
- Git makes sure that pushed changes are consistent with the remote branch so others can pull your commits and merge them into their own local copy

# Synchronization

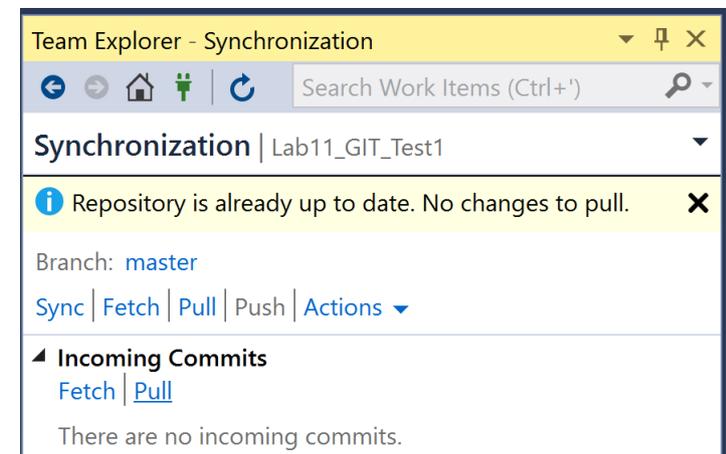
- Open Sync view in Team Explorer



- **Push** will push the commits to the remote branch.
- Resolve merge conflicts before pushing
  - If there are conflicts between your local commits on the commits on the remote branch, you must first resolve these conflicts before you can push your changes. You should pull the changes from others first, resolve the conflicts and commit the changes, then re-attempt the push.

# Update local repository

- Update the code in your local repo with the changes from other members of your team using the following commands:
  - **fetch** , which downloads the changes from your remote repo but does not apply them to your code.
  - **merge** , which applies changes taken from fetch to a branch on your local repo.
  - **pull** , which is a combined command that does a fetch and then a merge.
    - In Visual Studio:  
use Sync option in Team Explorer



<https://www.visualstudio.com/en-us/docs/git/tutorial/pulling>

# GIT Task

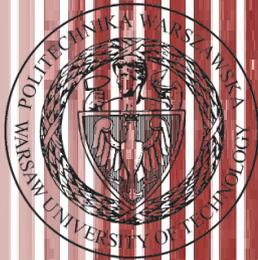
---

- **Make sure you have tested all the steps:**
- Using Team Services web account create new GIT repository for today's project
  - You can also create a solution (**Console Application**)
- Create local copy of the repository (Clone) using Team Explorer in Visual Studio
- Apply some changes (ask user to press key to continue), commit them, and push to the repository

# References

---

- GIT
  - <https://www.visualstudio.com/en-us/docs/git/tutorial/gitworkflow>
  - <https://www.visualstudio.com/en-us/docs/git/gitquickstart>



# THE END

dr inż. Małgorzata Janik  
[malgorzata.janik@pw.edu.pl](mailto:malgorzata.janik@pw.edu.pl)