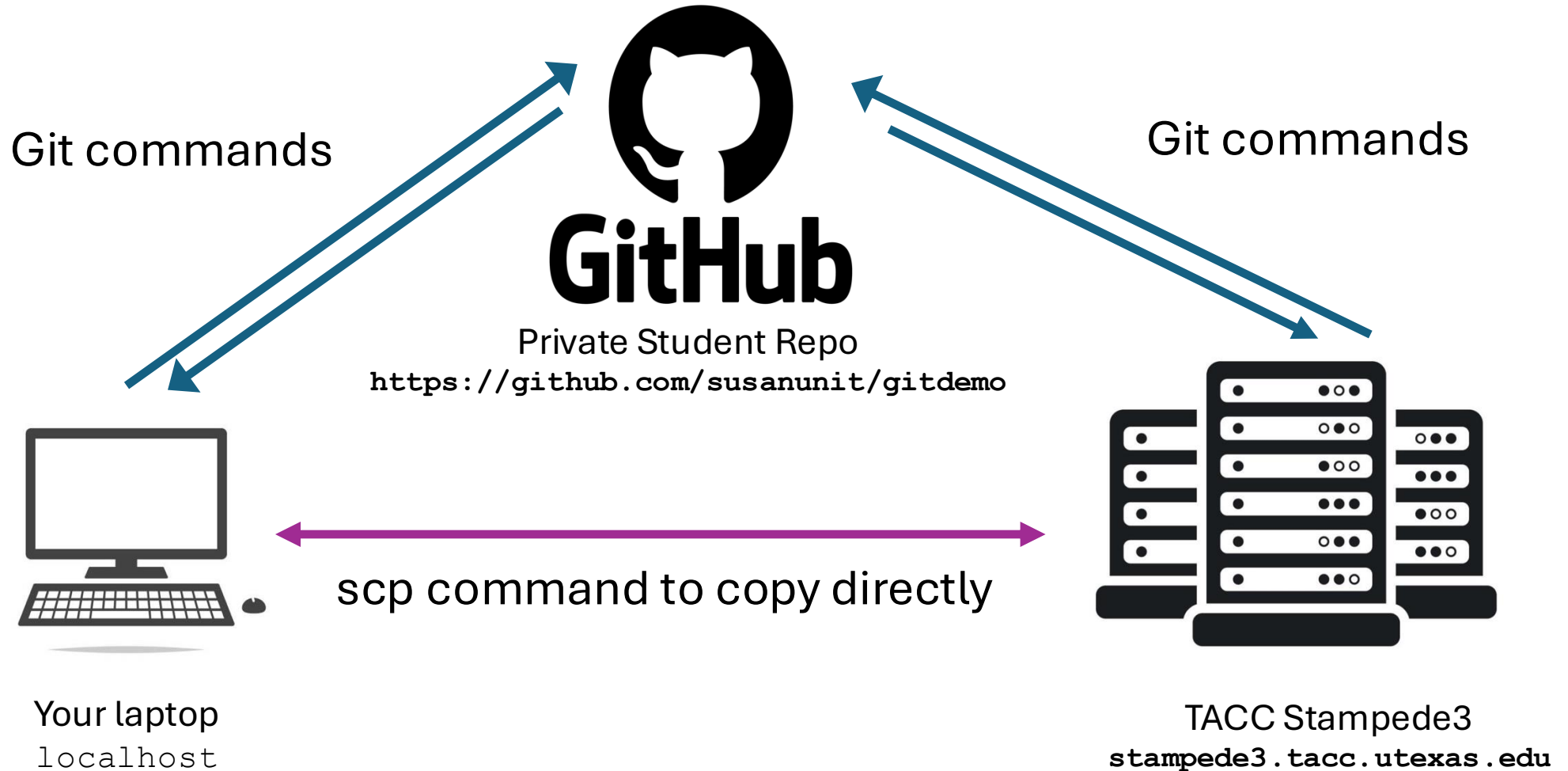


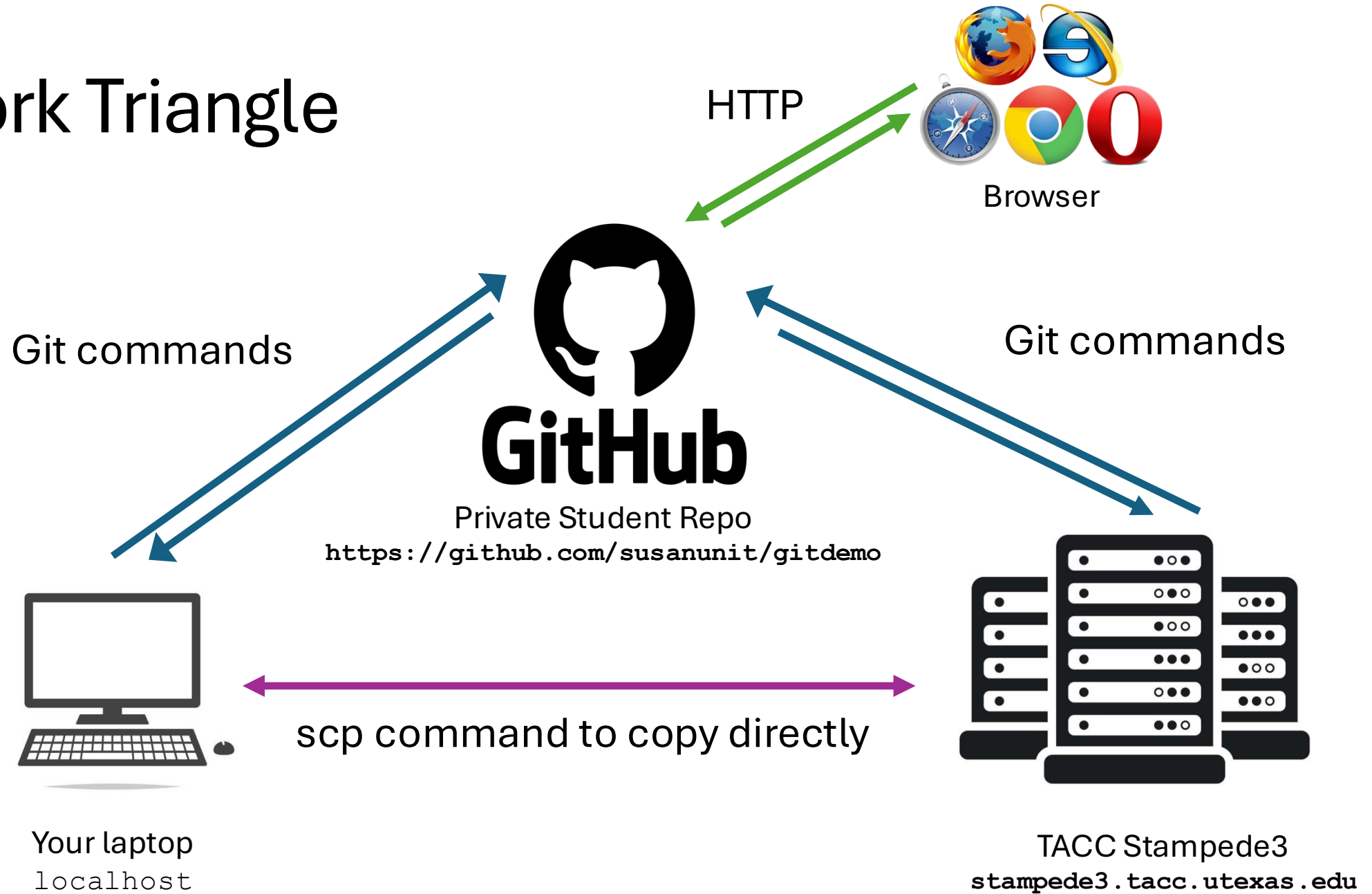
COE 322: GitHub Workflow

Victor Eijkhout, Susan Lindsey

Work Triangle



Work Triangle



Cloning and Configuring

On your laptop:

```
git clone git@github.com:username/hw1.git  
git config user.name "Your Name"  
git config user.email "your_email@utexas.edu"
```

On Stampede3:

```
ssh username@stampede3.tacc.utexas.edu  
git clone git@github.com:username/hw1.git
```

Very Useful `git` commands

<code>git status</code>	Display repository files' status
<code>git diff, git diff <file></code>	Show line-level edits since last commit
<code>git log --oneline</code>	Display short commit history
<code>git add <file></code>	Stage changes
<code>git commit -m "msg"</code>	Save snapshot
<code>git push</code>	Send changes to GitHub
<code>git pull</code>	Get updates from GitHub

Workflow 1 – Stampede3, GitHub

You can choose to develop only on Stampede3:

- SSH to Stampede3
- Edit and compile on Stampede3
- Commit and push to GitHub

WARNING: VSCODE users, don't do this. The latest updates to VSCODE do not work well with TACC machines, and VSCODE users are unable to log in or are frequently disconnected.

Workflow 1 – Stampede3, GitHub

Your git commands and workflow will look something like this:

```
$ emacs myprogram.cpp
```

```
$ git add !
```

```
$ git commit -m "this is what I did"
```

```
$ git pull # in case there are changes from  
another clone
```

```
$ git push
```

Workflow1 – laptop, Stampede3, gitHub

- localhost\$ Edit and compile on laptop
- localhost\$ Commit and push to GitHub
- localhost\$ SSH to Stampede3
- stampede3\$ Go to repo and pull the latest changes
- stampede3\$ Confirm your code compiles on Stampede3 (use `icpx` compiler).
- stampede3\$ Edit code if necessary, then commit and push to GitHub

Merge Conflicts

Typical error when out of sync:

```
! [rejected] main -> main (fetch first)
error: failed to push some refs
hint: Updates were rejected because the remote contains work that you do not have locally.
```

Conflict markers are located inside source files.

```
<<<<<< HEAD
printf("Hello from local machine\n");
=====
printf("Hello from Stampede3\n");
>>>>>> origin/main
```

Fix steps:

1. Edit the file to resolve the conflict.
2. `git add <file>`
3. `git commit -m "resolve conflict"`
4. `git push`

A common misstep

```
slindsey@wireless-10-145-49-81 ~ % git status
fatal: not a git repository (or any of the parent directories): .git
slindsey@wireless-10-145-49-81 ~ %
```

Ooops you are not in the repository directory or subdirectory

Git Best Practices

- When accessing a repository, first always update to the latest with a `git pull` command
- Do not edit/upload directly on GitHub.com - leads to conflicts
- Commit changes to your project early and often
 - Commit with clear and informative messages
- Commit only source files (`*.cpp`, `*.h`).
 - Your source files are valuable, revisions of a project
 - Do not commit binary files (`a.out`). No point – these are easily generated and not worth saving

Git – What not to do

- Corollary – don't do this:
 - `git add *`
 - `git add .`
- Be specific about the files you add to your repository. Do not clog it with junk.
- Branches – **Do not use branches in this class.** A branch is a fork in the code base for diverging development. No need for branches in this class.