BASIC OF GIT

What are Git and GitHub?

JOHN BRYCE
Leading in IT Education
a matrix company

To prosper within the team collaboration, storing, tracking and integration during the code evolving, ample of the developers use **version control systems (VCS)**.

Git is a VCS created by Linus Torvalds in 2005 for development of the Linux kernel, having multiple advantages over the other systems available, it stores file changes more efficiently and ensures file integrity better.

GitHub is a web-based hosting service for version control using Git. It offers all the distributed VCS functionality of Git as well as adding its own features: access control, bug tracking, feature requests, task management and wikis for every project.

Why the GitHub is so popular?

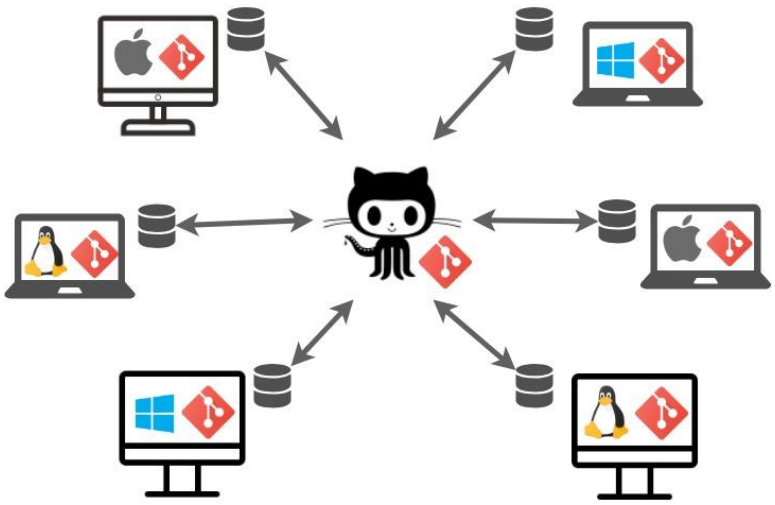
JOHN BRYCE
Leading in IT Education
a matrix company

In 2017, the GitHub community reached **24 million developers** from **1.5 million organizations** working across **67 million repositories:**

- Distributed Version Control
- Open source and free software
- Compatible with Unix Systems (Linux, Mac OS , Solaris) and Microsoft Windows
- Faster than other Source Code Version systems

Why the GitHub is so popular?

JOHN BRYCE
Leading in IT Education
a matrix company



Why the GitHub is so popular?

JOHN BRYCE
Leading in IT Education
a matrix company

- Performing a diff
- Viewing the file history
- Committing Changes
- Merging branches
- Obtaining other revision of file
- Switching branches

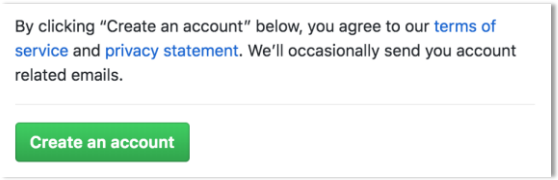
Open GitHub account

JOHN BRYCE
Leading in IT Education
a matrix company

Since the GitHub is open for everyone we could create the personal **account for free**. In that case we would get the Free plan account with unlimited public (opened to everyone) repositories options. For the additional fee it is possible to purchase the private repository functionality as well.

To create an account please proceed with the GitHub official portal:

- https://github.com/join
- Fill in the Username, Email and Password fields
- Press the ”Create an account” button



Configuring Git

JOHN BRYCE
Leading in IT Education
a matrix company

To start working on your PC, there is a need to configure the Git with the account credentials first:

- git config --global user.name “<yourname>”
- git config --global user.email “<yourname@domain.com>”
- git config --list

Working with GitHub: Getting started

JOHN BRYCE
Leading in IT Education
a matrix company

- git init - creates a new local repository on current directory

  john@john-vm:/tmp/new_component$ ***git init***
  Initialized empty Git repository in /tmp/new_component/.git/
  john@john-vm:/tmp/new_component$

- git clone - copies existed remote repository to local PC

  john@john-vm:/tmp$ ***git clone*** *git@bitbucket.org:automatitdevops/countryinfo.git*
  Cloning into 'countryinfo'...
  remote: Counting objects: 269, done.
  remote: Compressing objects: 100% (237/237), done.
  remote: Total 269 (delta 49), reused 189 (delta 17)
  Receiving objects: 100% (269/269), 1.87 MiB | 962.00 KiB/s, done.
  Resolving deltas: 100% (49/49), done.
  john@john-vm:/tmp$

Working with GitHub: Getting started

JOHN BRYCE
Leading in IT Education
a matrix company

- **git commit** - set of changes under source control

```
john@john-vm:/tmp/countryinfo$ git status
On branch master
Your branch is up-to-date with 'origin/master'.
Changes not staged for commit:
 (use "git add <file>..." to update what will be committed)
 (use "git checkout -- <file>..." to discard changes in working directory)
modified:   currency.php
no changes added to commit (use "git add" and/or "git commit -a")
john@john-vm:/tmp/countryinfo$ git add currency.php
john@john-vm:/tmp/countryinfo$ git commit -m 'Remove new line'
[master f7c452e] Remove new line
1 file changed, 1 insertion(+), 2 deletions(-)
```


Working with GitHub: Getting started

JOHN BRYCE
Leading in IT Education
a matrix company

- **git push** - upload local commits to remote repository

```
john@john-vm:/tmp/countryinfo$ git push
Counting objects: 3, done.
Delta compression using up to 2 threads.
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 306 bytes | 306.00 KiB/s, done.
Total 3 (delta 2), reused 0 (delta 0)
To bitbucket.org:automatitdevops/countryinfo.git
  5ae168a..f7c452e  master -> master
john@john-vm:/tmp/countryinfo$
```

Working with GitHub: Getting started

JOHN BRYCE
Leading in IT Education
a matrix company

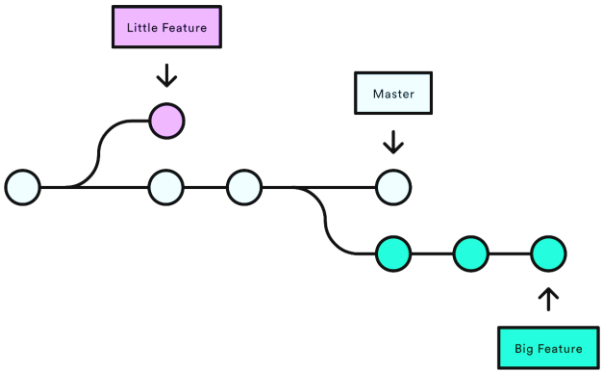
- **git pull** - download commits from centralized repository

```
john@john-vm:~/repos/countryinfo$ git pull
remote: Counting objects: 9, done.
remote: Compressing objects: 100% (7/7), done.
remote: Total 9 (delta 4), reused 0 (delta 0)
Unpacking objects: 100% (9/9), done.
From bitbucket.org:automatitdevops/countryinfo
  547f0ca..f7c452e  master -> origin/master
Updating 547f0ca..f7c452e
Fast-forward
codestyles/Default.xml | 1 +
currency.php           | 3 +--
2 files changed, 2 insertions(+), 2 deletions(-)
create mode 100644 codestyles/Default.xml
```


Branching and Merging

JOHN BRYCE
Leading in IT Education
a matrix company

Branch - ordered set of commits. Git branches are effectively a pointer to a snapshot of your changes. When you want to add a new feature or fix a bug—no matter how big or how small—you spawn a new branch to encapsulate your changes.



Branching and Merging

JOHN BRYCE
Leading in IT Education
a matrix company

Create new branch:

```
john@john-vm:~/repos/countryinfo$ git checkout -b new_web_handler
Switched to a new branch 'new_web_handler'

List of all branches:
john@john-vm:~/repos/countryinfo$ git branch -a
 master
 new_cool_feature
* new_web_handler
 remotes/origin/HEAD -> origin/master
 remotes/origin/master
 remotes/origin/new_cool_feature
john@john-vm:~/repos/countryinfo$
```


Branching and Merging

JOHN BRYCE
Leading in IT Education
a matrix company

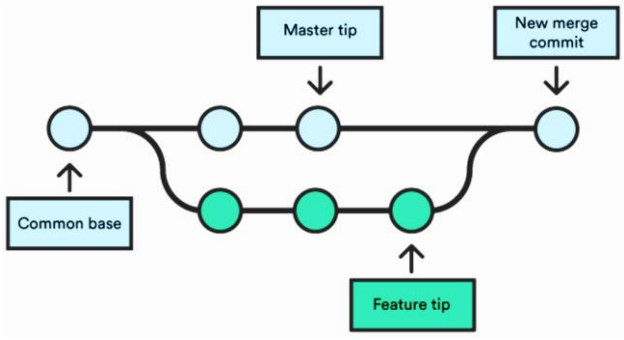
Pushing new branch to upstream repository:

```
john@john-vm:~/repos/countryinfo$ git push --set-upstream origin test_feature2
Counting objects: 3, done.
Delta compression using up to 2 threads.
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 303 bytes | 303.00 KiB/s, done.
Total 3 (delta 2), reused 0 (delta 0)
remote:
remote: Create pull request for test_feature2:
remote:   https://bitbucket.org/automatitdevops/countryinfo/pull-requests/new?source=test_feature2&t=1
remote:
To bitbucket.org:automatitdevops/countryinfo.git
* [new branch]      test_feature2 -> test_feature2
Branch test_feature2 set up to track remote branch test_feature2 from origin.
```

Branching and Merging

JOHN BRYCE
Leading in IT Education
a matrix company

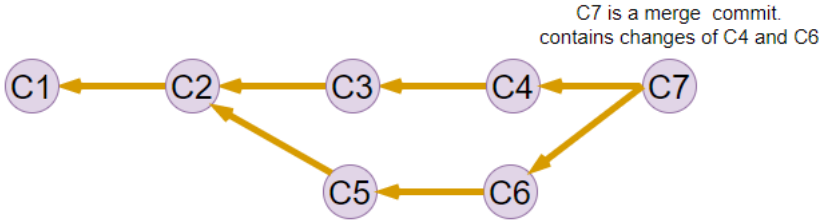
Merge - join two or more branches together. The git merge command lets you take the independent lines of development created by git branch and integrate them into a single branch.



Branching and Merging

JOHN BRYCE
Leading in IT Education
a matrix company

Checkout target branch:



john@john-vm:~/repos/countryinfo$ ***git checkout master***
Switched to branch 'master'
Your branch is up-to-date with 'origin/master'.

Branching and Merging

JOHN BRYCE
Leading in IT Education
a matrix company

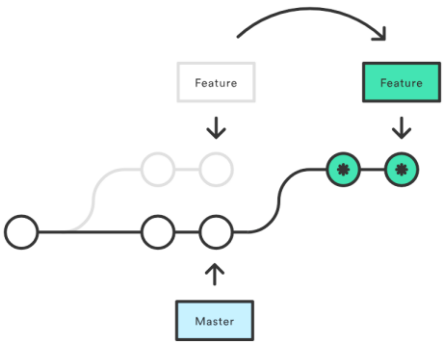
Merge new_web_handler into master branch

```
john@john-vm:~/repos/countryinfo$ git merge new_web_handler
Updating f7c452e..403279d
Fast-forward
rest_api.php | 4 +---
1 file changed, 1 insertion(+), 3 deletions(-)
john@john-vm:~/repos/countryinfo$
```


Rebasing Changes

JOHN BRYCE
Leading in IT Education
a matrix company

Rebase - apply commit (changes) to another commit. It is the process of moving or combining a sequence of commits to a new base commit. Rebasing is most useful and easily visualized in the context of a feature branching workflow.



Rebasing Changes

JOHN BRYCE
Leading in IT Education
a matrix company

Rebase the master branch

```
john@john-vm:~/repos/countryinfo$ git rebase master
First, rewinding head to replay your work on top of it...
Applying: delete unnecessary space from editor.xml
john@john-vm:~/repos/countryinfo$
```


Workflows

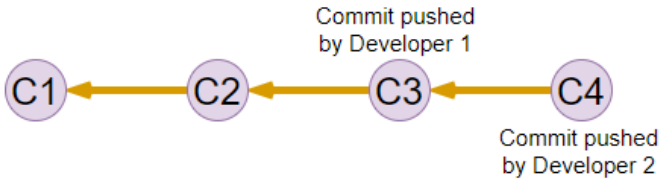
JOHN BRYCE
Leading in IT Education
a matrix company

- Centralized
- Feature branch
- Gitflow
- Forking

Workflows: Centralized

JOHN BRYCE
Leading in IT Education
a matrix company

The main idea of the **Centralized** workflow that all the developers push changes to into one master branch.



Pros:

- Minimum operations to deliver changes to master branch

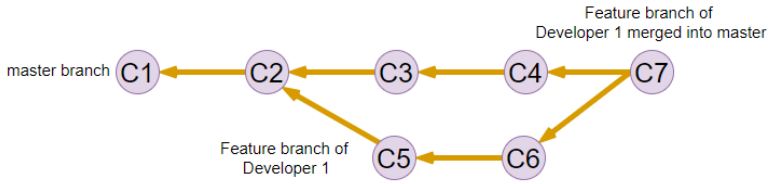
Cons:

- A lot of merges (and merge conflicts).
- Master branch can have broken code.

Workflows: Feature branch

JOHN BRYCE
Leading in IT Education
a matrix company

The main idea of the **Feature branch** workflow that all feature development should take place in a dedicated branch instead of the master branch.



Pros:

- Only one merge per feature.
- Potentially master branch can have stable and tested code.

Cons:

- Additional operations to deliver changes to master branch.

Workflows: Gitflow

JOHN BRYCE
Leading in IT Education
a matrix company

Gitflow workflow doesn't add any new concepts or commands beyond what's required for the Feature Branch Workflow. Instead, it assigns very specific roles to different branches and defines how and when they should interact.

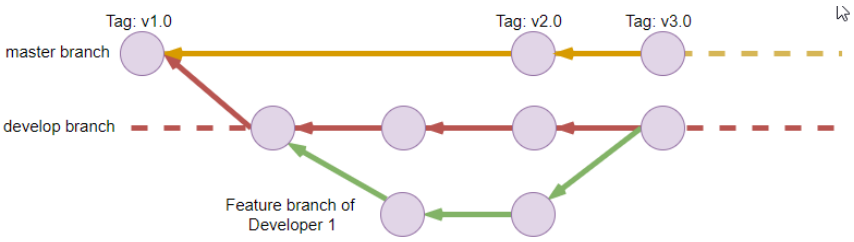
Mandatory branches:

- **Master** - official release history, all commits are tagged.
- **Develop** - main branch to integrate features.
- **Release** - temporary branch to prepare release.
- **Feature** - branches for a features
- **Hotfix** - bugfixes for release

Workflows: Gitflow

JOHN BRYCE
Leading in IT Education
a matrix company

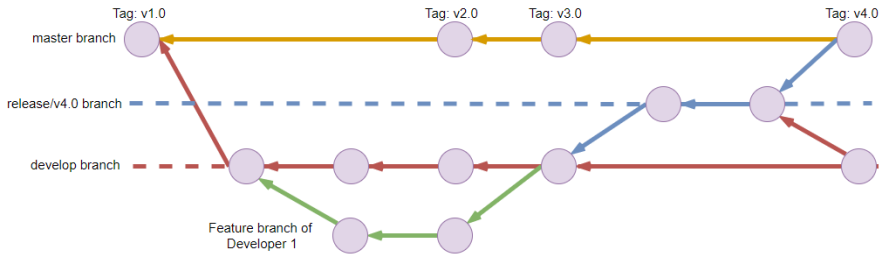
Develop branch- main branch to integrate features:



Workflows: Gitflow

JOHN BRYCE
Leading in IT Education
a matrix company

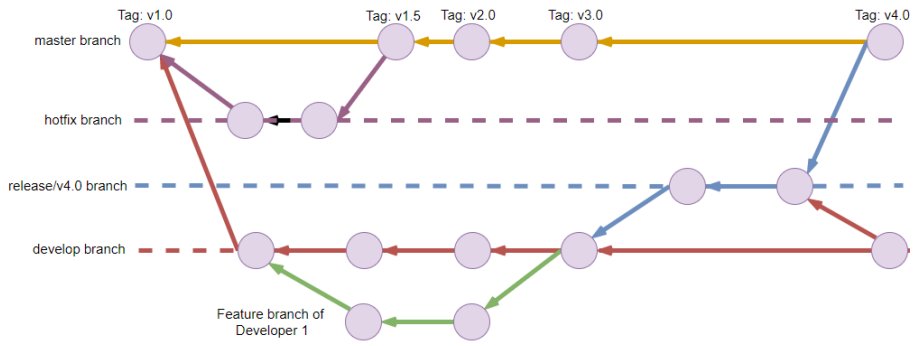
Release branch- temporary branch to prepare release:



Workflows: Gitflow

JOHN BRYCE
Leading in IT Education
a matrix company

Hotfix branch- bugfixes for release:

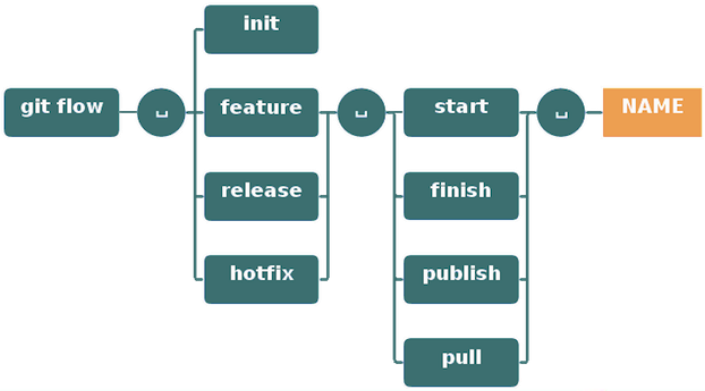


Workflows: Gitflow

JOHN BRYCE
Leading in IT Education
a matrix company

Gitflow extension

The Gitflow are a set of git extensions to provide high-level repository operations for Gitflow branching model:



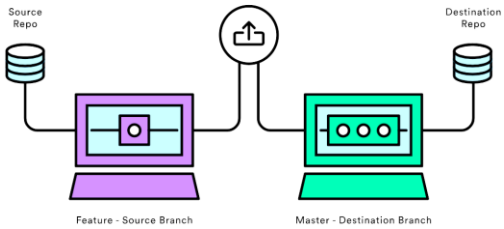
Workflows: Forking

JOHN BRYCE
Leading in IT Education
a matrix company

The **Forking Workflow** is fundamentally different than the other workflows discussed here. Instead of using a single server-side repository to act as the "central" codebase, **it gives every developer a server-side repository**. This means that each contributor has not one, but two Git repositories: a private local one and a public server-side one.

Pull Requests

When you file a **pull request**, all you're doing is *requesting* that another developer (e.g., the project maintainer) *pulls* a branch from your repository into their repository.



This means that you need to provide 4 pieces of information to file a pull request: the source repository, the source branch, the destination repository, and the destination branch.

Pull Requests

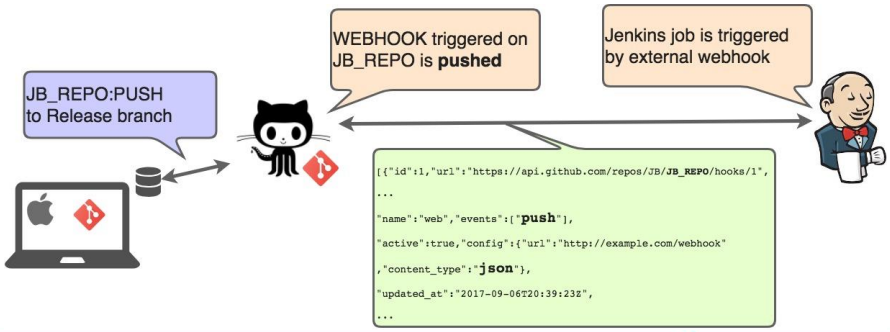
Pull requests are a feature that makes it easier for developers to collaborate. Due to PR all merges can be reviewed and approved or declined.

- Pull requests can be used in conjunction with the Feature Branch Workflow, the Gitflow Workflow, or the Forking Workflow.
- Pull request requires either two distinct branches or two distinct repositories, so they will not work with the Centralized Workflow.

Webhook

JOHN BRYCE
Leading in IT Education
a matrix company

Webhooks provide a way for notifications to be delivered to the integrated external system whenever certain actions occur on a repository or organization. Once installed, the webhook will be triggered each time one or more subscribed events occurs.



Webhook: Events

JOHN BRYCE
Leading in IT Education
a matrix company

Events

- When configuring a webhook, you can choose which events you would like to receive payloads for. You can even opt-in to all current and future events.
- Only **subscribing** to the specific events you plan on handling is useful for limiting the number of HTTP requests to your server.
- You can change the list of subscribed events through the API or UI anytime. By **default**, webhooks are only subscribed to the **PUSH** event.
- The GitHub has about **37 available events**

Thanks!!!

JOHN BRYCE
Leading in IT Education
a matrix company