



# Continuous Integration with Github Actions

Gabriele Graffieti  
Biolab research meeting  
30/04/2021



# Software development in the real world

1. Write (or copy from stack overflow) some code into my project.
2. Do some tests to check if the new code works.
3. I'm a super developer!

But you are sure that:

- Your code is actually correct in any test case?
- Your code works in a different PC (different OS, different libraries, different software installed...)
- What about documentation? What about producing artifacts?
  - Tedious and easy-to-forget work!
- All the dependency of your software are easily downloadable by anyone? And are updated?
- ...

# What are the problems here?

- We are human, we often forget to do things.
  - Test the software before publication
  - Write tests for new pieces of software
  - Correctly document the software
- We are human, we have a limited time
  - Always test the software in different machines, OS, ...
  - Manage an up-to-date site containing all the documentation
  - Producing the artifacts (conda/pip package, docker images, manuals, executable files, ...)
- We are human, we cannot evaluate software without using it
  - Other people who make contributions to our software
  - Code quality metrics (PEP8, Flake...)

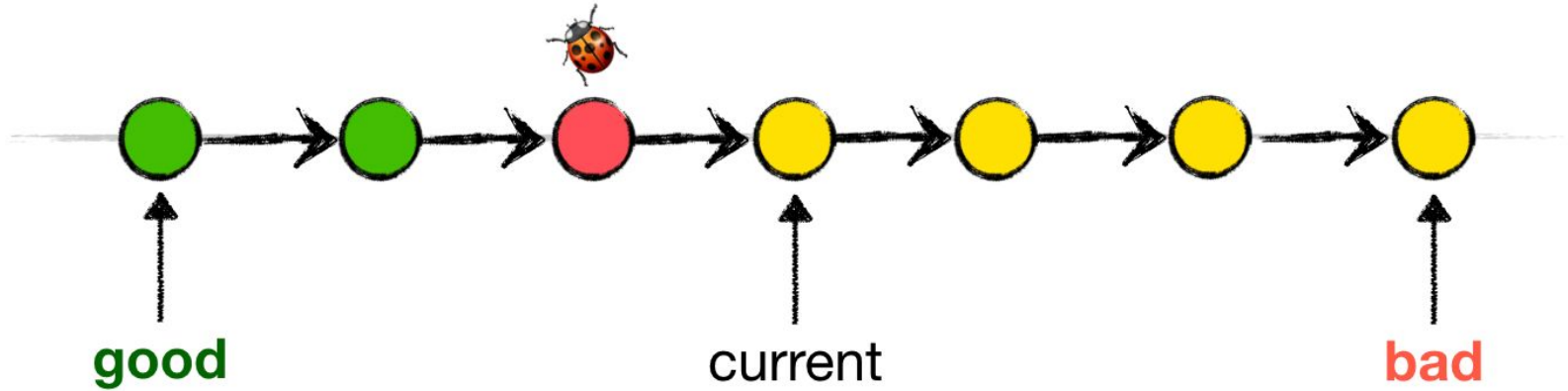
# Continuous Integration

Continuous Integration (CI) is a development practice where developers integrate code into a shared repository frequently, preferably several times a day. Each integration can then be verified by an automated build and automated tests.

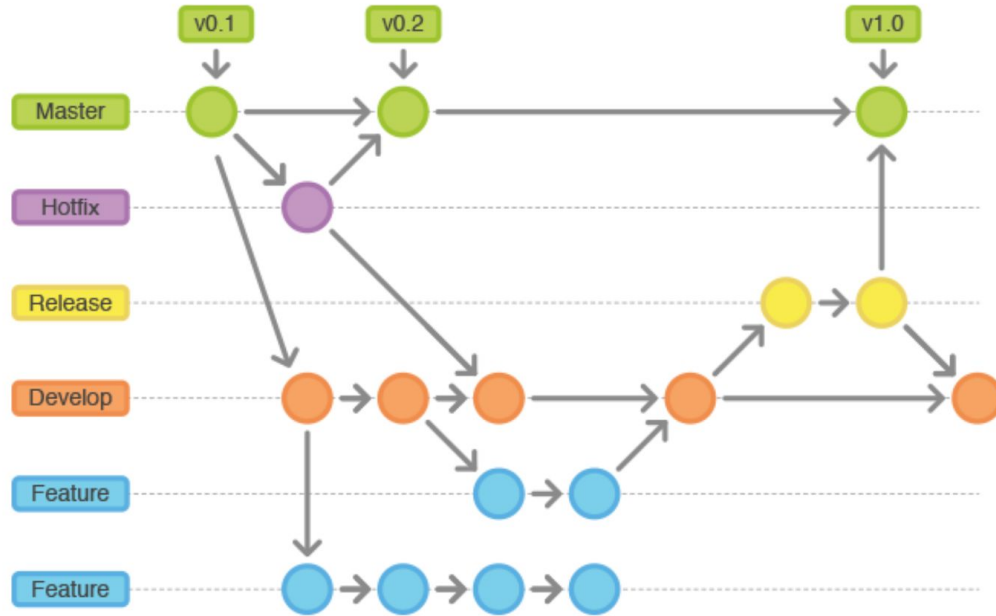
- Automate the majority of the side work.
- Just focus on ideas and write software.
- The continuous integration pipeline will do anything else for you.
- Detect bugs and problems quickly and locate them easily.

Continuous integration + Continuous deployment + Continuous delivery = happy developers and better software

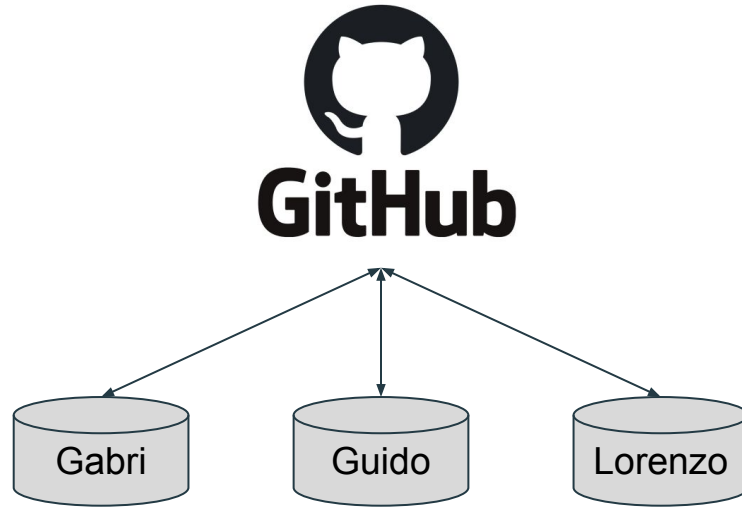
# Basic: git



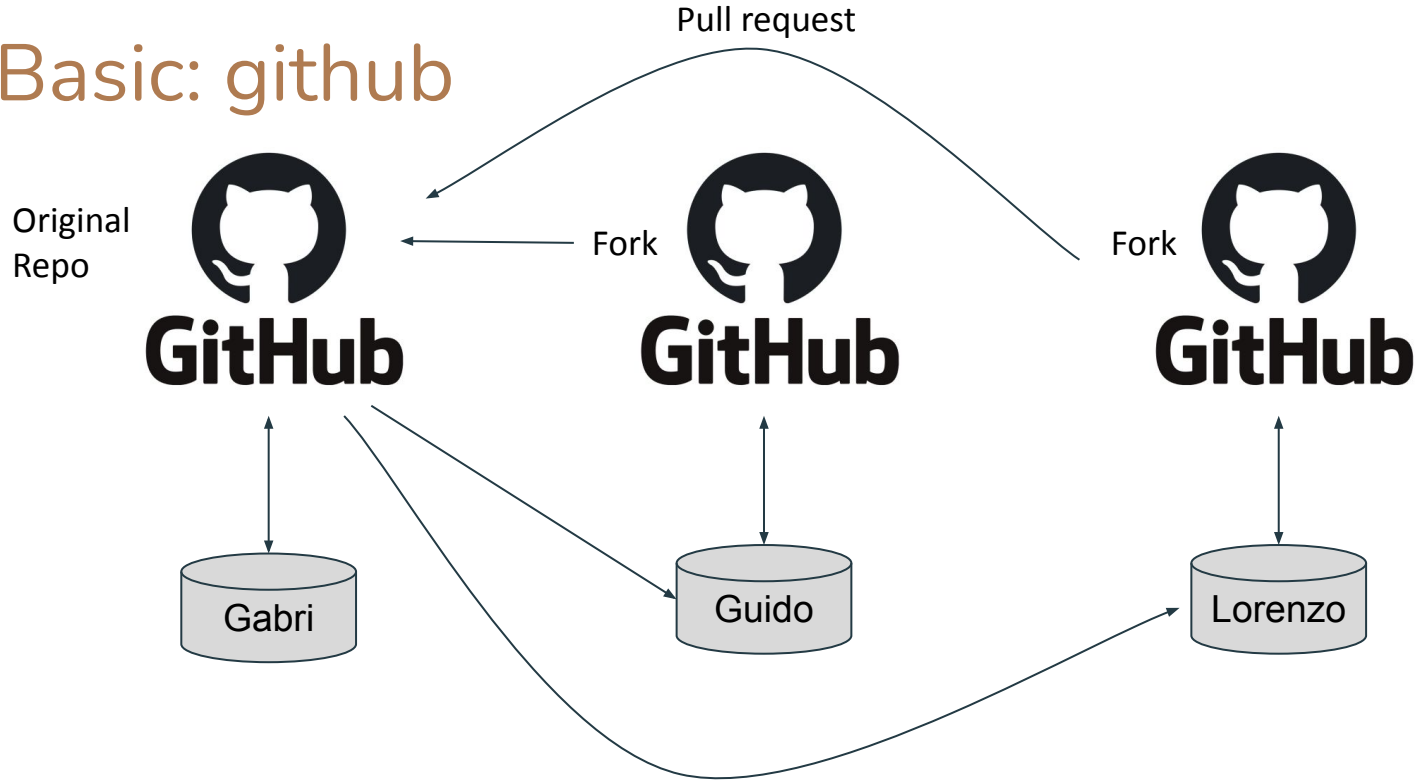
# Basic: git



# Basic: github















# Basic: github





# Pull requests

<input type="checkbox"/>	 5 Open ✓ 193 Closed	Author ▾	Label ▾	Projects ▾	Milestones ▾	Reviews ▾	Assignee ▾	Sort ▾
<input type="checkbox"/>	 <b>add CTrL integration</b> ✓ #561 opened 6 days ago by TomVeniat • Draft							 5
<input type="checkbox"/>	 <b>Standardization of Initial Weights (SIW) Plugin implementation</b> ✗ #527 opened 17 days ago by EdenBelouadah • Draft							 15
<input type="checkbox"/>	 <b>Added Deep SLDA Model implementation</b> ✓ #500 opened 23 days ago by tyler-hayes • Draft							 3
<input type="checkbox"/>	 <b>fix-agem. resolved memory filling and grad proj</b> ✗ #495 opened 24 days ago by JonasFrey96 • Draft							 10
<input type="checkbox"/>	 <b>Issue 212: Implementing forward transfer</b> ✗ #493 opened 25 days ago by ryanlindeborg • Draft							 4

# Issues

<input type="checkbox"/>	<span>🔔</span> 53 Open <span>✓</span> 269 Closed	Author ▾	Label ▾	Projects ▾	Milestones ▾	Assignee ▾	Sort ▾
<input type="checkbox"/>	<span>🔔</span> <b>bug with CIFAR10 default transformations</b> <span>Benchmarks</span> <span>bug</span> #575 opened 2 hours ago by AntonioCarta						
<input type="checkbox"/>	<span>🔔</span> <b>Replay not working anymore?</b> <span>Training</span> <span>bug</span> #574 opened 17 hours ago by AndreaCossu						2
<input type="checkbox"/>	<span>🔔</span> <b>Expand FZTH "models" chapter</b> <span>documentation</span> <span>models</span> #571 opened 21 hours ago by vlomonaco						2
<input type="checkbox"/>	<span>🔔</span> <b>How-To Tutorials on Gitbook</b> <span>documentation</span> #568 opened 2 days ago by vlomonaco <span>📄</span> 0 of 5						4
<input type="checkbox"/>	<span>🔔</span> <b>Confusion Matrix for W&amp;B</b> <span>Evaluation</span> #564 opened 3 days ago by AndreaCossu						
<input type="checkbox"/>	<span>🔔</span> <b>Check Offered Functionalities &lt;-&gt; Doc sync</b> <span>Feature - Medium Priority</span> <span>documentation</span> #558 opened 6 days ago by vlomonaco						
<input type="checkbox"/>	<span>🔔</span> <b>Uniform API for classic Benchmarks</b> <span>Benchmarks</span> <span>Feature - Medium Priority</span> <span>good first issue</span> #557 opened 6 days ago by vlomonaco						1
<input type="checkbox"/>	<span>🔔</span> <b>Expand Benchmarks Doc to Include Dataset Utils</b> <span>Benchmarks</span> <span>Feature - High Priority</span> <span>documentation</span> #551 opened 8 days ago by vlomonaco						4
<input type="checkbox"/>	<span>🔔</span> <b>Fix avl "core" module</b> <span>Feature - Low Priority</span> <span>core</span> <span>documentation</span> #548 opened 9 days ago by vlomonaco						
<input type="checkbox"/>	<span>🔔</span> <b>CORe50 Dataset and Benchmark Improvements</b> <span>Benchmarks</span> <span>Feature - Low Priority</span> #547 opened 9 days ago by vlomonaco <span>📄</span> 0 of 4						

# Discussion

Search all discussions

New

Top: All

Answered

Unanswered

New discussion

## Categories

View all

Feature Request




General

Ideas


Q&A

## Most helpful


Last 30 days


 vlomonaco	4
 lrzpellegri	2
 EdenBelouadah	1

- ↑ 2


 **Override some training parameters in `train` call**


AndreaCossu started 5 days ago in Feature Request · **Unanswered**

 13
- ↑ 1


 **Logging Validation loss in fine-grained level in Tensorboard logger.**


akashtadwai started 6 days ago in Q&A · **Unanswered**

 11
- ↑ 1


 **Visualising the last layer feature map with tSNE**


akashtadwai started 21 days ago in Q&A · **Answered**

 11
- ↑ 1


 **Create a Dataset from Tensors**


vlomonaco started 9 days ago in Feature Request · **Answered**

 2
- ↑ 1


 **Expand CORE50 support for Detection and Segmentation**

vlomonaco started on 15 Mar in Feature Request · **Unanswered**

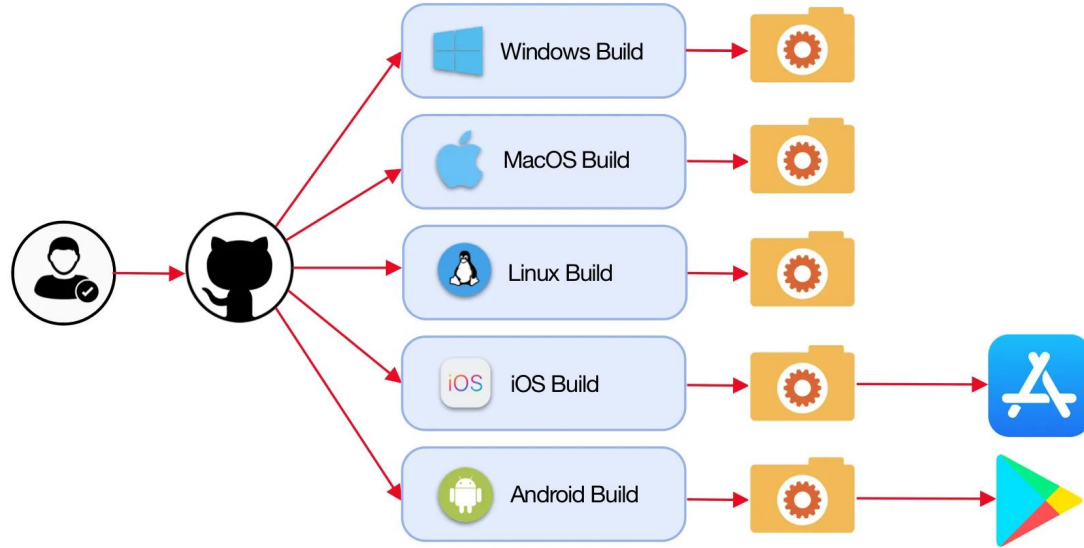
 3
- ↑ 2

 **How to convert one incoming example into a valid Avalanche-Mini-Dataset**

TheRisenPhoenix started 11 days ago in Q&A · **Answered**

 6

# Github Actions



# Github Actions

- Every interaction with github can trigger an action (push, pull\_request, comment, another action, ...)
- Github make a VM available to do (almost) everything I want
  - Not so bad spec (see next slide)
  - A lot of software preinstalled (see [here](#))
  - If I need certain software I can use a docker image!
  - I can use my own machines to host my github actions.
- Actions are “programmed” with .yml files inside the folder `.github/workflows`

# Github-hosted runners

- **Linux:** Ubuntu LTS 20, 18, 16
- **Windows:** Windows server 2019, 2016
- **Mac:** MacOS Big Sur 11.0, Catalina 10.15

Spec:

- 2-core CPU (3-core CPU for MacOS)
- 7 GB of RAM memory (14 GB for MacOS)
- 14 GB of SSD disk space

A lot of software preinstalled (e.g. python, conda, pip, mongodb, mysql, git, gcc...)

# But how much it costs?

- Free for public and open source repositories.
- Limited (depending on your plan) for private repository.

With our plan (Github pro):

- 3,000 minutes a month of computation
- 1GB of storage (for artifacts, etc..)

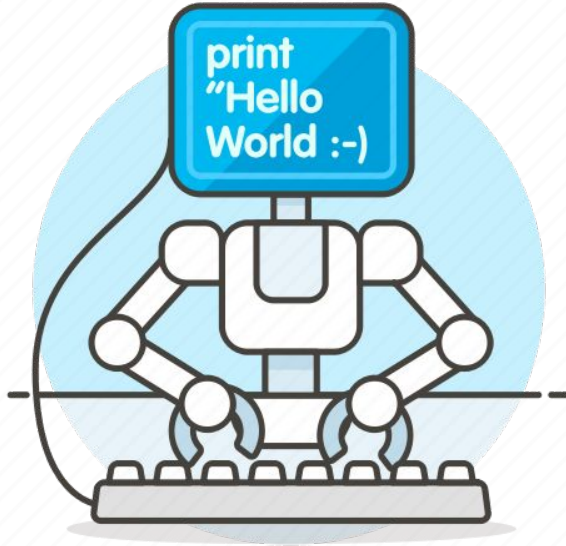
Note that 1min = 1min on Linux! On windows we have a x2 multiplier, on MacOS a x10 multiplier!

# Some (boring) notation

- **Workflow:** is an automated procedure that you add to your repository. Workflows are made up of one or more jobs and can be scheduled or triggered by an event.
- **Event:** is a specific activity that triggers a workflow.
- **Job:** is a set of steps that execute on the same runner. By default, a workflow with multiple jobs will run those jobs in parallel.
- **Step:** is an individual task that can run commands in a job. A step can be either an action or a shell command.
- **Actions:** are standalone commands that are combined into steps to create a job. Actions are the smallest portable building block of a workflow. You can create your own actions, or use actions created by the GitHub community.



# Hands-on: automatically publish API-doc



All the code is available here: <https://github.com/ggraffieti/github-actions-meetup>

# Hands-on: automatic unit test



Useful link: <https://docs.github.com/en/actions/reference/workflow-syntax-for-github-actions>

# Hands-on: PEP8 and comment PR

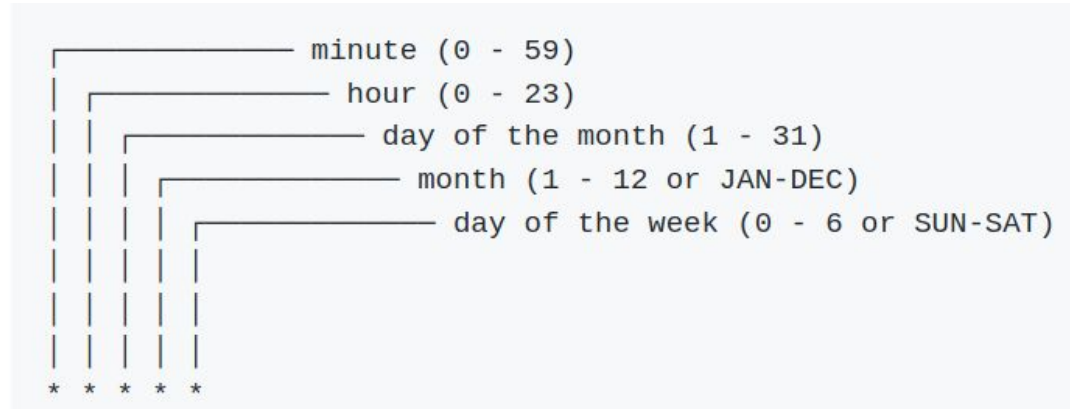


# Cron Jobs

Apart from being triggered by any interaction with Github, a workflow can also be scheduled to run periodically.

```
on:  
  schedule:  
    # * is a special character in YAML so you have to quote this string  
    - cron: '30 5,17 * * *'
```

# Cron Jobs



- \* = any value (\* \* \* \* \* = every minute of every day)
- , = value separator (2,10 4,5 \* \* \* runs at minute 2 and 10 of the 4th and 5th hour of every day.)
- - = range of values (0 4-6 \* \* \* runs at minute 0 of the 4th, 5th, and 6th hour.)
- / = Step values (20/15 \* \* \* \* runs every 15 minutes starting from minute 20 through 59 (minutes 20, 35, and 50).

Useful: 0 0 \* \* 0

# Advanced Actions (in Avalanche)



Link: <https://github.com/ContinualAI/avalanche>

That's all folks!