Fighting Git Workflows

April, 2019
Bristol

@ciura_victor

Victor Ciura
Technical Lead, Advanced Installer
www.advancedinstaller.com

Abstract

A medium size team, on a monthly release cycle, for a mature commercial product developed for almost 15 years.

About a year ago we decided to join the rest of the Universe and switched from SVN to Git. What could go wrong there ?

Centralized workflow, feature branch workflow, GitFlow, forking workflow, OneFlow, No flow ?! Oh my !
To rebase or not to rebase ? That is the question.

Prepare yourself for an equal amount of stories, gotchas and workarounds, both on the developer side of things, as well as the DevOps maintenance nightmare of CI/CD pipelines for testing, validation, daily builds and other necessary gears.

Please join me for a quick tour of a blatantly honest case-study on using Git / GitLab and related tools in a dynamic development team. Learn about the tools & scripts we built to help up navigate this treacherous land...

Spoiler: no unicorns here.

Intro

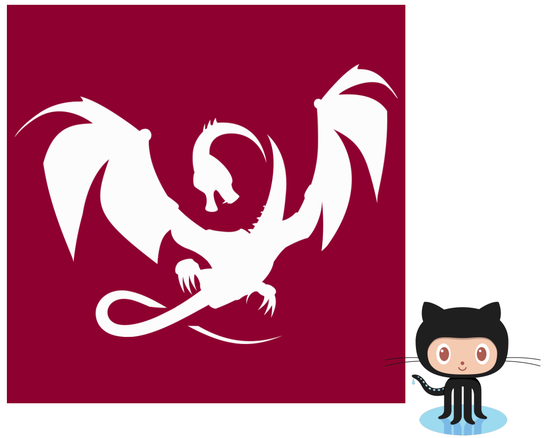
This session is first and foremost a blatantly honest case-study on using Git / GitLab and related tools in a dynamic development team and the struggles of switching from a centralized SCM system.

I'll present a path we explored as we learned what works and doesn't work for our team in terms of Git workflows, tools and accepted best practices.

I don't think our methodology is universally applicable to other teams and it isn't all perfect for us either, but some of the things we found out the hard way may serve others attempting similar changes.

Who Am I?

Advanced Installer



Clang Power Tools

@ciura_victor

Some context

Advanced Installer

www.advancedinstaller.com

- **16** year old code base, under active development
- **3+** million lines of C++ code
- **~146** Visual Studio 2017 projects (EXEs, DLLs, LIBs)
- **Windows-only** deployment
- **Monthly** release cycle

Some context

Advanced Installer

www.advancedinstaller.com

- **20** C++ developers
- **5** QA engineers
- **6** solution engineers / analysts
- **3** visual designers
- **2** technical writers

SVN mono-repo (15yo)

⬇

GIT repo + submodules

✅ Learn git basics

✅ Evaluate tooling options

✅ Prepare a migration strategy

😟 Learn about idiomatic git workflows

→ Fear Change
Change Fear ←

For 15 years...

Our team on SVN

development on `trunk`

seldom feature branches (*experiments*)

release branches (for easy *patches* 4.3.1)

code freeze period in each release cycle

Our team on SVN

Recurring issues:

imprecise planning/scheduling of features

=> feature flags 🤷

breach of code freeze period

very painful feature/change removal (*rollback*)

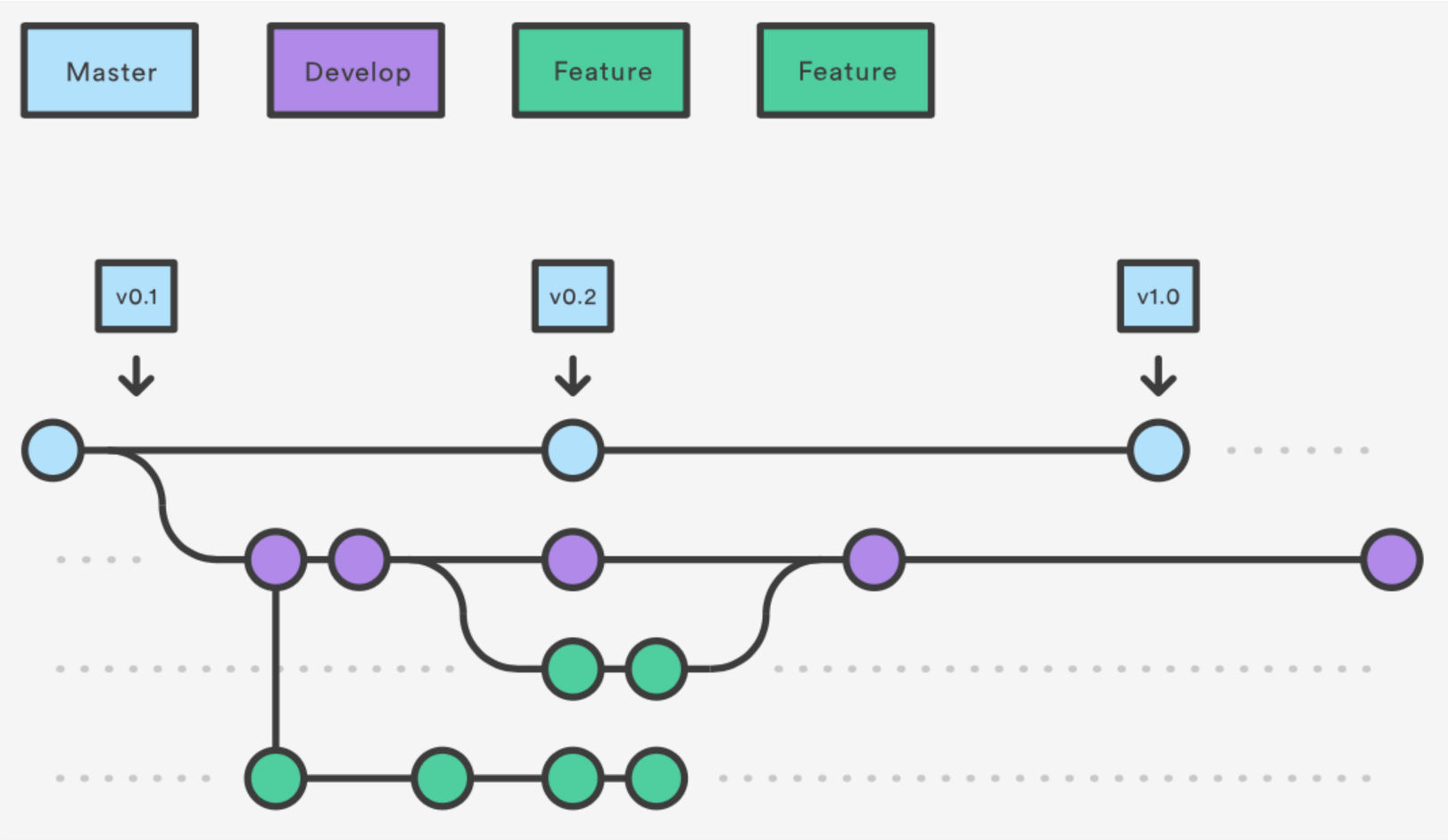
What we want:

(predictably) deliver high quality features on time

avoid painful *rollbacks*

A timid start down the right path...

Embrace feature branches



GitFlow

A timid start down the right path...

Two important decisions:

- when to integrate changes into `master` ?
- how to integrate them ?

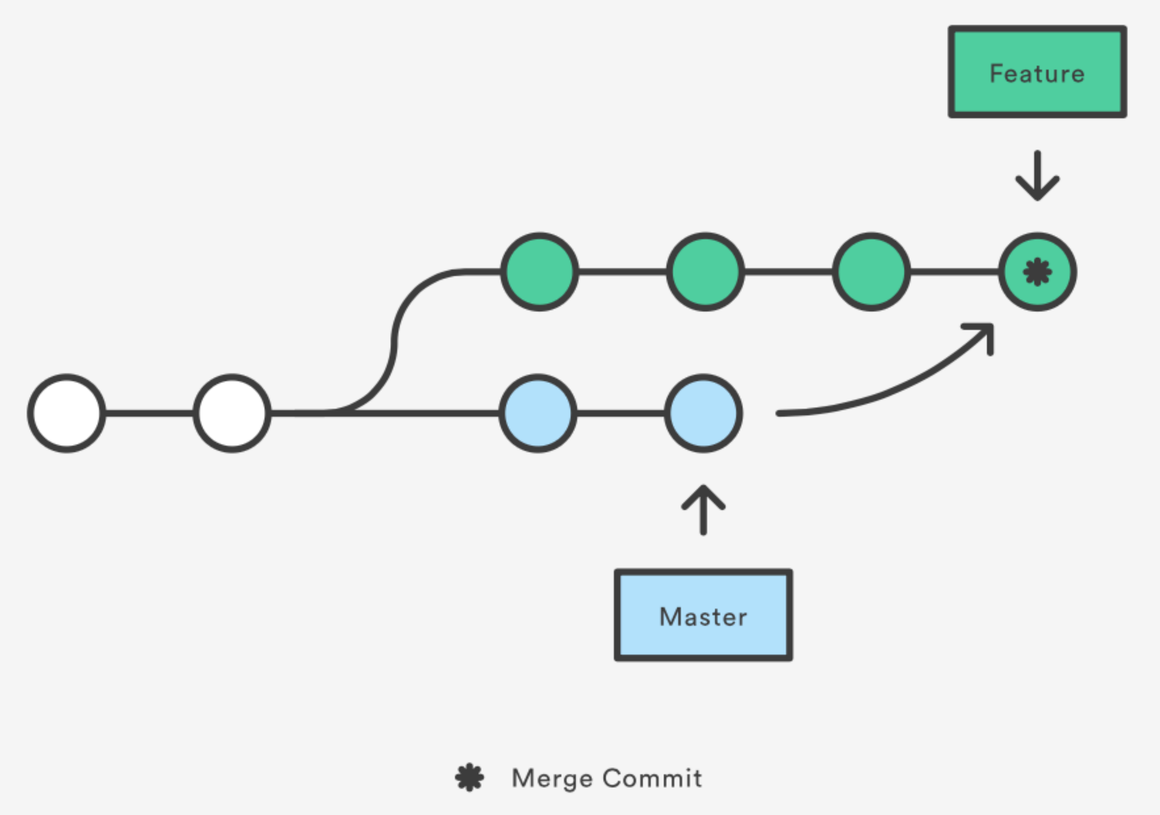
How to integrate changes
(from feature branches)

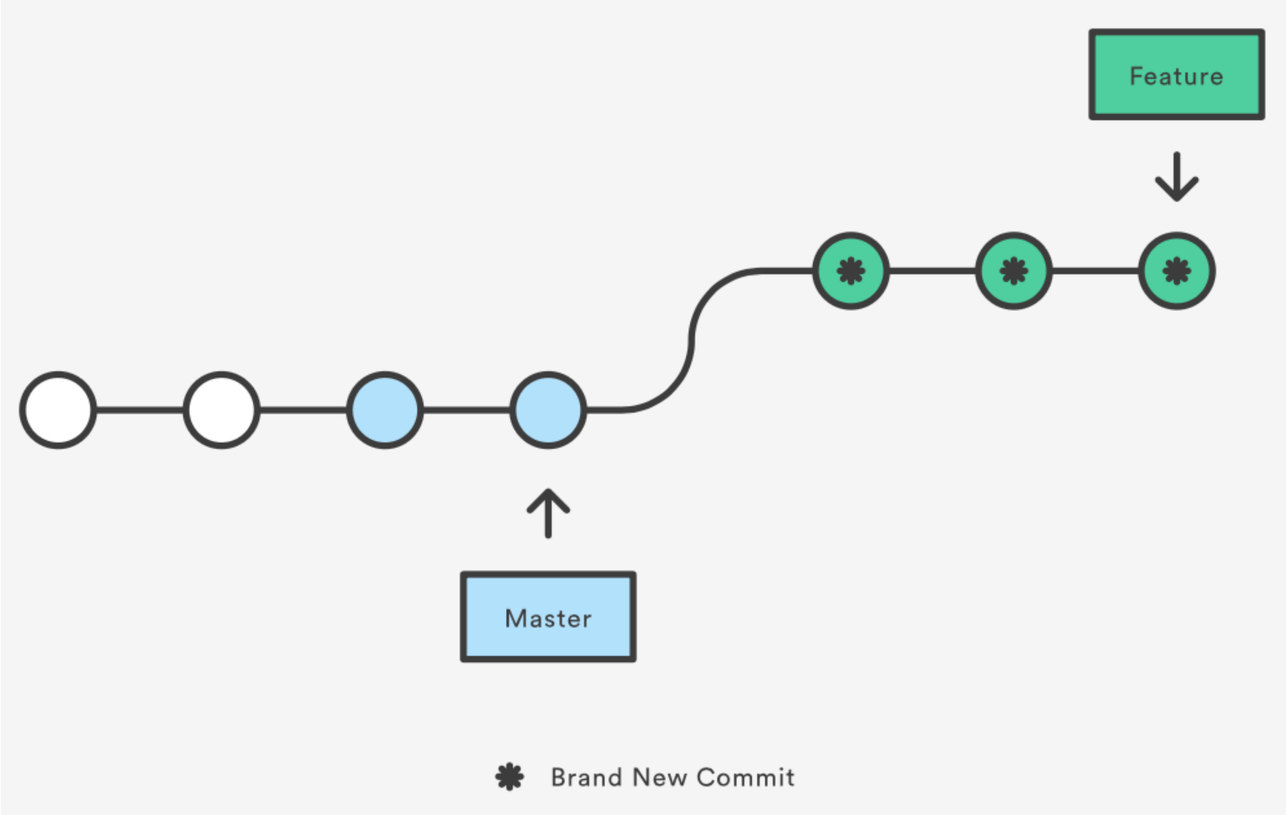
This being git, there isn't one answer to rule them all...

Merging vs. Rebasing

https://medium.com/@porteneuve/getting-solid-at-git-rebase-vs-merge-4fa1a48c53aa

Merging vs. Rebasing





https://www.atlassian.com/git/tutorials/merging-vs-rebasing

Merging

Pros:

- `Traceability:` **helps keeping information about the historical existence of a feature branch and *groups* together all commits part of the feature**

Cons:

- `History` **can become intensely** `polluted` **by lots of merge commits**

=> visual graphs can have many tangled branch lines, difficult to follow

https://www.atlassian.com/git/articles/git-team-workflows-merge-or-rebase

Rebasing

Pros:

- `History` **remains flat and readable. Clean, clear commit messages are encouraged.**
- `Reverting` **a single commit is easy.**

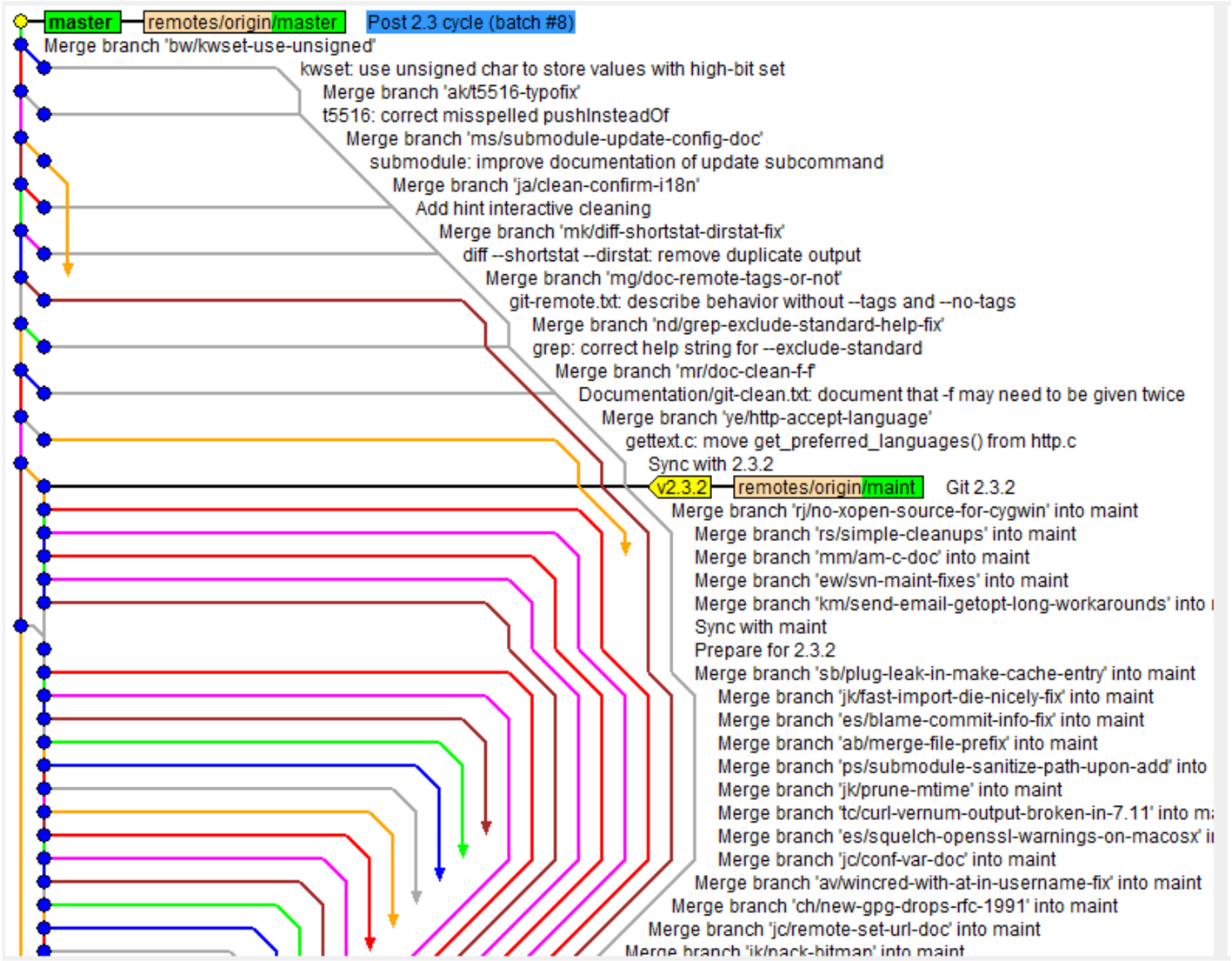
Cons:

- **Squashing the feature down to a handful of commits can *hide context.***
- **It's *more work*: using rebase to keep your feature branch updated requires that you resolve similar conflicts again and again.**
- **You need to** `force push` **at some point.**
- **Re-writing history can be dangerous.**

https://www.atlassian.com/git/articles/git-team-workflows-merge-or-rebase

Coming from SVN,
the natural choice, seems to merge

Merge Mania



http://www.tugberkugurlu.com/archive/resistance-against-london-tube-map-commit-history-a-k-a--git-merge-hell

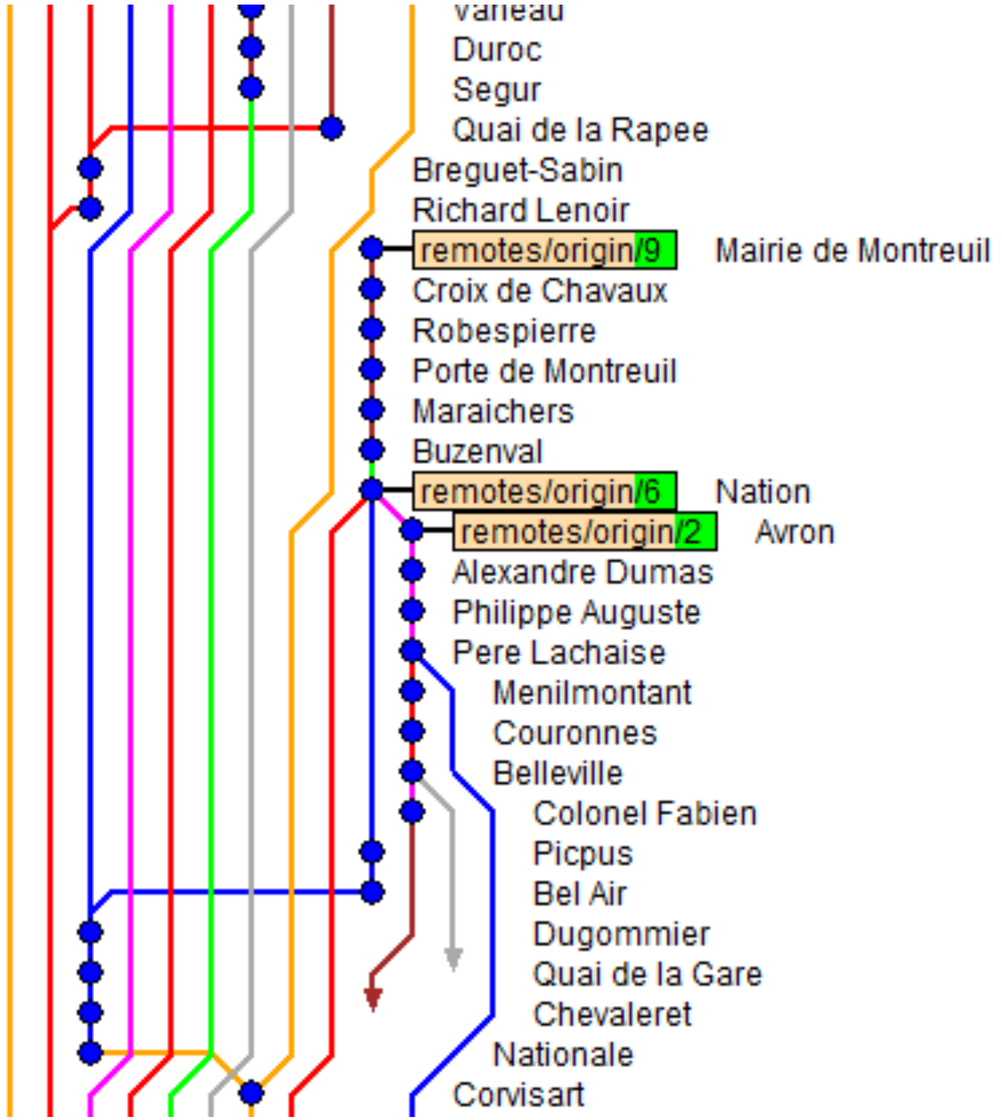
Paris Metro

Each metro **line** is represented by a branch.

A commit corresponds to a **station**.

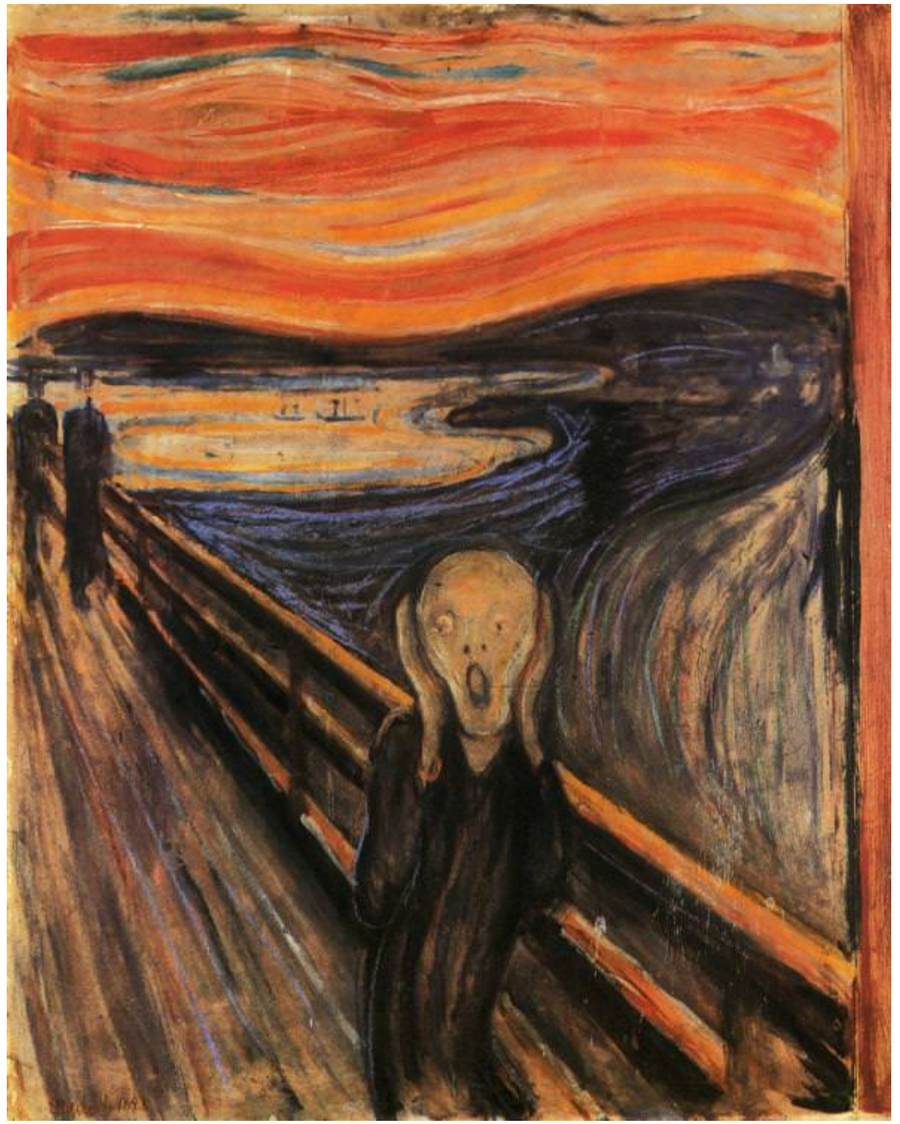
A merge is a **connection** between one or more lines.

300 metro stations from Paris



https://github.com/vbarbaresi/MetroGit

GIT REBASE



Edvard Munch (1893)

In the end we opted for
a clean, flat history

`git rebase`

When to integrate changes
(from feature branches)

When to integrate changes

We started an experiment:

6 months: workflow A
6 months: workflow B

Workflow A

First 6 months on `git`

Very similar to our old SVN workflow (min. disruption)

Approve MRs for next milestone (eg 4.6)
=> integrate into `develop/master` branch

QA on `develop/master` branch

Bug-fixes on multiple independent branches => `develop/master`

Workflow A
drawbacks

Has many of the old SVN workflow drawbacks

Bug-fixes on multiple independent branches => `develop/master`

Difficult to pull back integrated feature branches in case of QA issues before delivery

`develop/master` branch is not stable, up to release day

Release day can be pushed back (days), when 🔥 occurs

Workflow B

Next 6 months on `git`

Highly disruptive, but fits within *traditional* agile models...

QA on `feature` branch + *light* integration testing

Bug-fixes on `feature` branch => **often *rebases* needed**

Approve MRs for next milestone (eg 4.6) if it passes QA on time
=> integrate into `develop/master` branch

Workflow B drawbacks

QA on `feature` branches => *light* integration testing needed

Bug-fixes on `feature` branch => often `rebases` needed

If FeatureX doesn't pass QA on time => long-lived branch...

QA will pick-up testing on *deferred branch* and report bugs *weeks* after developer finished work on that component.

Contemplating a `git rebase` 🤔

When to integrate changes

We're currently on Workflow B,

but the experiment is far from over*
as we gain more experience with it every day....

* although its 6 months trial has passed

MR

⬇

code review

⬇

build pipeline

⬇

QA pass

⬇

merged

Teodor M

DevOps Engineer

Why should we do work that can be automated ?

GitLab

JIRA

GitLab Runner Pipeline

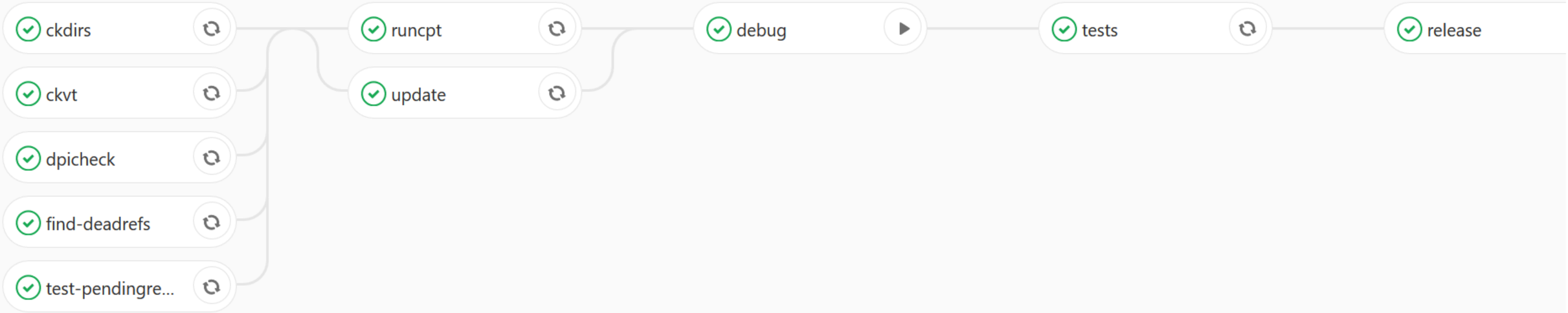
```
Issue: AI-15392
```

10 jobs from `improvement/15392-build-packages-from-repackager` in 90 minutes and 9 seconds (queued for 3 seconds)

latest merge request

`fc0c01ca`

Pipeline Jobs 10



GitLab Pipeline Scheduler

ai › advinst › Schedules

Scheduling Pipelines

The pipelines schedule runs pipelines in the future, repeatedly, for specific branches or tags. Those scheduled pipelines will inherit limited project access based on their associated user.

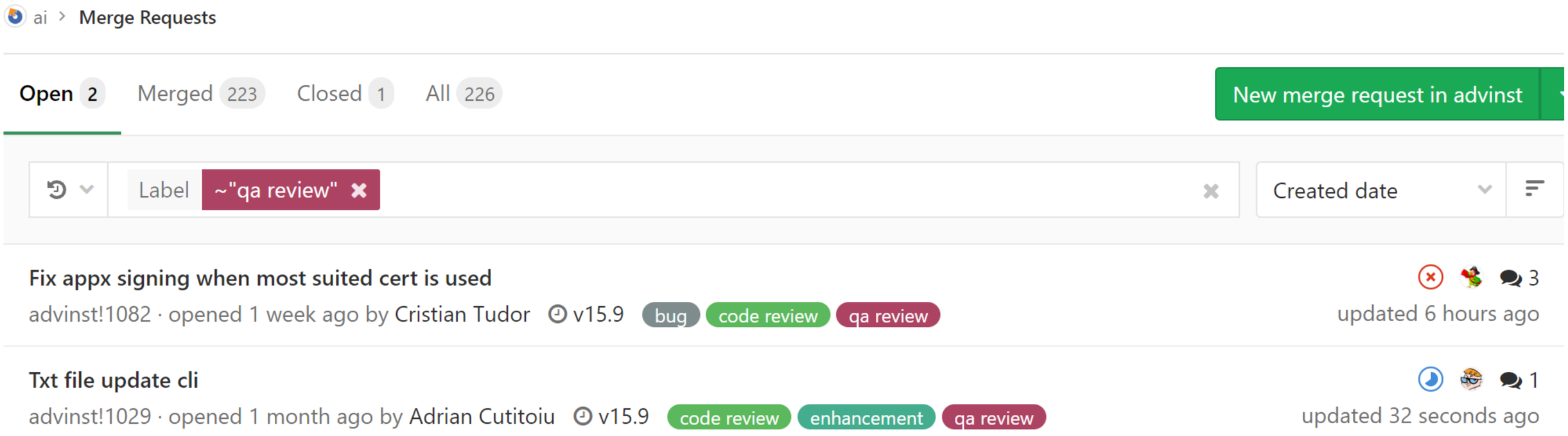
Learn more in the pipeline schedules documentation.

All 8 | Active 8 | Inactive 0

New schedule

Description	Target	Last Pipeline	Next Run	Owner	
[unpack] advinst release build	master	#19969	in 15 hours	Gitlab Bot	▶ Take ownership
[unpack] advinst debug build	master	#19968	in 15 hours	Gitlab Bot	▶ Take ownership
[rc] advinst release build	master	#19963	in 13 hours	Gitlab Bot	▶ Take ownership
[rc] advinst debug build	master	#19962	in 13 hours	Gitlab Bot	▶ Take ownership
[merge_requests] merge on master	master	#20266	in 41 minutes	Gitlab Bot	▶ Take ownership
[merge_requests] rebase on master	master	#20253	in 1 hour	Gitlab Bot	▶ Take ownership
[merge_requests] synchronize Jira tickets	master	#20265	in 41 minutes	Gitlab Bot	▶ Take ownership
[merge_requests] checks for approval	master	#20264	in 41 minutes	Gitlab Bot	▶ Take ownership

MRs with QA passed



MR behind master

ai › advinst › Merge Requests › !1029

Open Opened 1 month ago by **Adrian Cutitoiu** Edit Close merge request

Txt file update cli

Closes AI-13890

Request to merge
`improvement/13890-txt...` **into** `master`
The source branch is 124 commits behind the target branch

Open in Web IDE | Check out branch

Pipeline `#21346` **passed for** `d923acd3` **on** `improvement/13890-txt...`

Rebase Fast-forward merge is not possible. Rebase the source branch onto the target branch or merge target branch into source branch to allow this merge request to be merged.

Todo Add todo

Assignee Edit
Ciprian Burca
@ciprian.burca

Milestone Edit
v15.9

Time tracking
No estimate or time spent

Labels Edit
code review | enhancement | qa review

Lock merge request Edit
Unlocked

3 participants

MR rebased -> build pipeline

ai > advinst > Merge Requests > !1029

Open | Opened 1 month ago by **Adrian Cutitoiu** | Edit | Close merge request

Txt file update cli

Closes AI-13890

Request to merge `improvement/13890-txt...` **into** `master` | Open in Web IDE | Check out branch

Pipeline `#21448` **running for** `1df2a23d` **on** `improvement/13890-txt...`

Set by Gitlab Bot to be merged automatically when the pipeline succeeds | Cancel automatic merge

The changes will be merged into `master`

The source branch will be deleted

Todo | Add todo

Assignee | Edit
Ciprian Burca
@ciprian.burca

Milestone | Edit
v15.9

Time tracking
No estimate or time spent

Labels | Edit
code review | enhancement | qa review

Lock merge request | Edit
Unlocked

3 participants

MR Pipeline Failed

ai › advinst › Merge Requests › !1082

Open | Opened 1 week ago by **Cristian Tudor** | Edit | Close merge request

Fix appx signing when most suited cert is used

Close: AI-12635

Request to merge bugfix/12635-appx_sig... **into** master | Open in Web IDE | Check out branch

Pipeline #21409 failed for c6a4a2d5 on bugfix/12635-appx_sig...

Merge | The pipeline for this merge request failed. Please retry the job or push a new commit to fix the failure

Todo | Add todo

Assignee

Dorel Pîslan
@dorel.pislan

Milestone

v15.9

Time tracking

No estimate or time spent

Labels

bug | code review | qa review

Lock merge request

Unlocked

3 participants

MR was merged

ai › advinst › Merge Requests › !910

Merged Opened 3 months ago by **Dorel Pîslan** Edit Report abuse

Use detour in ComExtractor

Close AI-15199

Edited 2 months ago by Dorel Pîslan

Request to merge `improvement/15199-use_detour_in_comextractor` **into** `master`

Pipeline #21287 **passed for** `c978a56d` **on** `improvement/15199-use_detour_in_comextractor`

Merged by Gitlab Bot 15 hours ago Revert Cherry-pick

The changes were merged into `master` with `154c4d75`

The source branch has been deleted

Todo Add todo »

Assignee Edit

Mihai Udrea
@mihai.udrea

Milestone Edit

v15.9

Time tracking

No estimate or time spent

Labels Edit

code review enhancement qa review

Lock merge request Edit

Unlocked

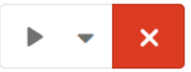
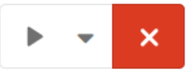
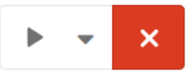
5 participants

Project Running Pipelines

ai › advinst › **Pipelines**

All 1,000+ | Pending 0 | **Running 5** | Finished 1,000+ | Branches | Tags

Run Pipeline | Clear Runner Caches | CI Lint

Status	Pipeline	Commit	Stages		
running	#17504 by latest merge request	improvement/… a7de5fae Update ribbon strings			
running	#17500 by latest merge request	task/gitlab-… 4c501fdd fix condition for pipeline curre…		00:48:01 1 hour ago	
running	#17471 by latest merge request	improvement/… a6eaae89 Add Build check for Signtool v…		21 hours ago	
running	#17326 by latest merge request	bugfix/15460… 3417548f Fix XML test search crash		00:01:58 5 days ago	
running	#17298 by latest merge request	improvement/… 8588bcac Change a description in wizar…		00:03:02 6 days ago	

Sometimes...

Status	Pipeline	Commit	Stages	
running	#20003 by latest merge request	improvement/… 54295640 Add support for reusing strin…		
running	#19988 by latest merge request	task/build-a… 6420584d new msbuild targets to skip re…		
running	#19987 by latest merge request	improvement/… f415ea75 Fix build		
running	#19982 by latest merge request	bugfix/11034… eb45ce69 Better fix		
running	#19981 by latest merge request	bugfix/13983… 1b4fa545 Fix progress		
running	#19980 by latest merge request	bugfix/13949… b9e2364f Build fails when having "Refre…		
running	#19979 by latest merge request	bugfix/5563-… 076f31dc Move FolderHasFiles method …		
running	#19978 by latest merge request	bugfix/15276… add4b6e9 Dont change install type in m…		
running	#19977 by latest merge request	bugfix/10736… f0c5fb65 Wizard crashes when a valid S…		
running	#19976 by latest merge request	bugfix/15491… 159f36a4 Cancel run in vm operation if …		
running	#19975 by latest merge request	bugfix/15275… aafaf735 Fix clang errors		
running	#19974 by latest merge request	improvement/… 3fd811f6 Fix clang build	00:43:22 1 hour ago	
running	#19973 by latest merge request	bugfix/13222… 9cc5d423 UI Improvement		
running	#19971 by latest merge request	bugfix/13487… a9d7bfbf Reversed changes made by m…		
running	#19776 by latest merge request	improvement/… 9c21dad8 Fix double quotes	02:14:36 18 hours ago	

Task Manager

File Options View

Processes | Performance | App history | Startup | Users | Details | Services

CPU
100% 3.59 GHz

Memory
7.6/31.7 GB (24%)

Disk 0 (C:)
18%

Disk 1 (D:)
2%

Ethernet
S: 584 R: 40.0 Kbps

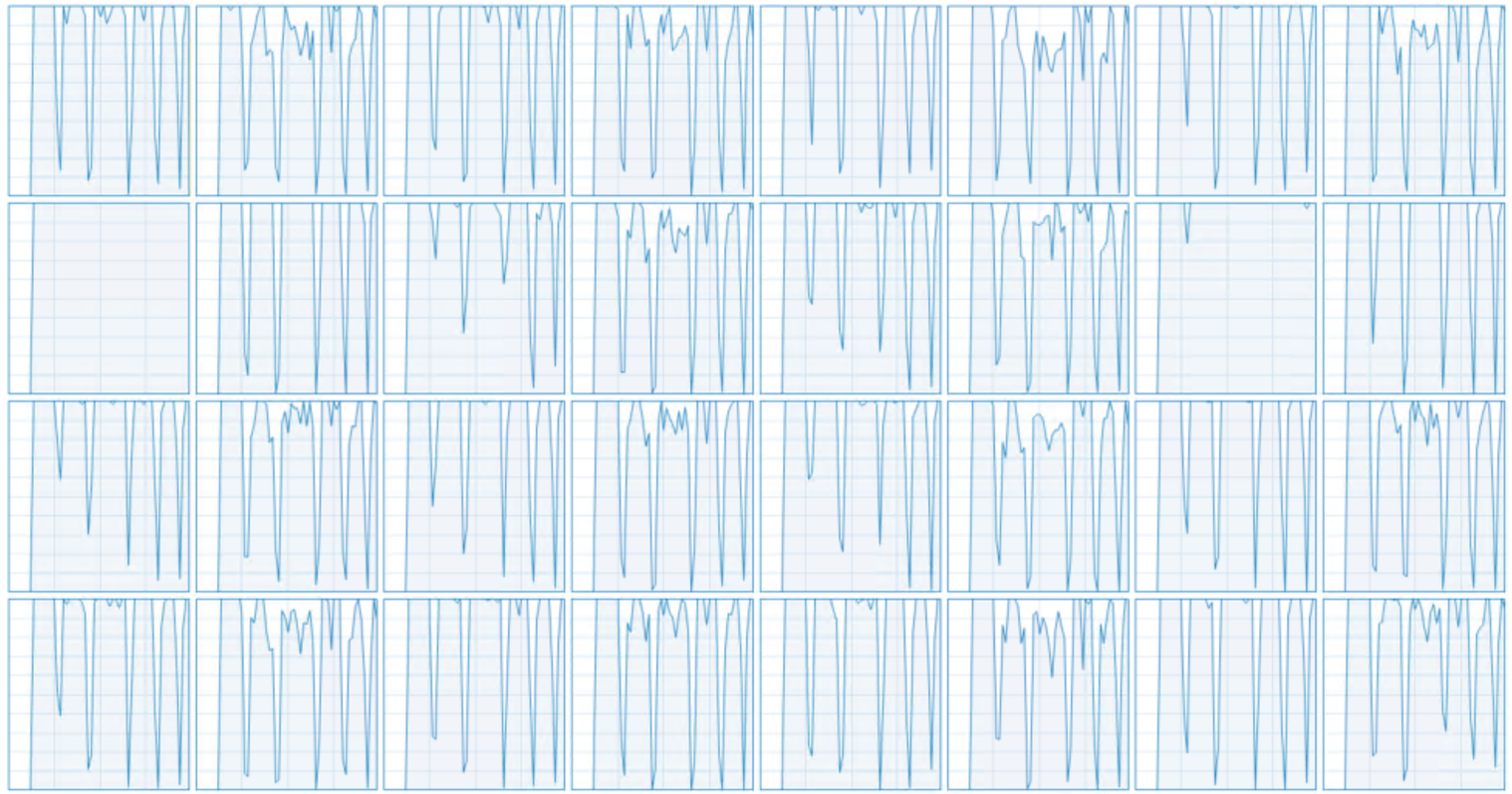
GPU 0
NVIDIA GeForce GT 730
1%

CPU

Intel(R) Core(TM) i9-7960X CPU @ 2.80GHz

% Utilization over 4 minutes

100%



Utilization	Speed
100%	3.59 GHz

Processes	Threads	Handles
284	3112	94174

Up time
0:00:41:59

Base speed:	2.81 GHz
Sockets:	1
Cores:	16
Logical processors:	32
Virtualization:	Enabled
L1 cache:	1.0 MB
L2 cache:	16.0 MB
L3 cache:	22.0 MB

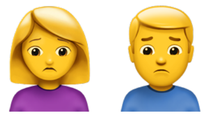
3 x

It's all good 👍

https://en.meming.world/wiki/Hide_The_Pain_Harold

Tools & People

This was not just about tools & processes,
rather about PEOPLE



Thanks to my colleagues
for their patience and willingness to *experiment*

I'm curious to learn about
your experiences...

Fighting Git Workflows

April, 2019
Bristol

@ciura_victor

Victor Ciura
Technical Lead, Advanced Installer
www.advancedinstaller.com