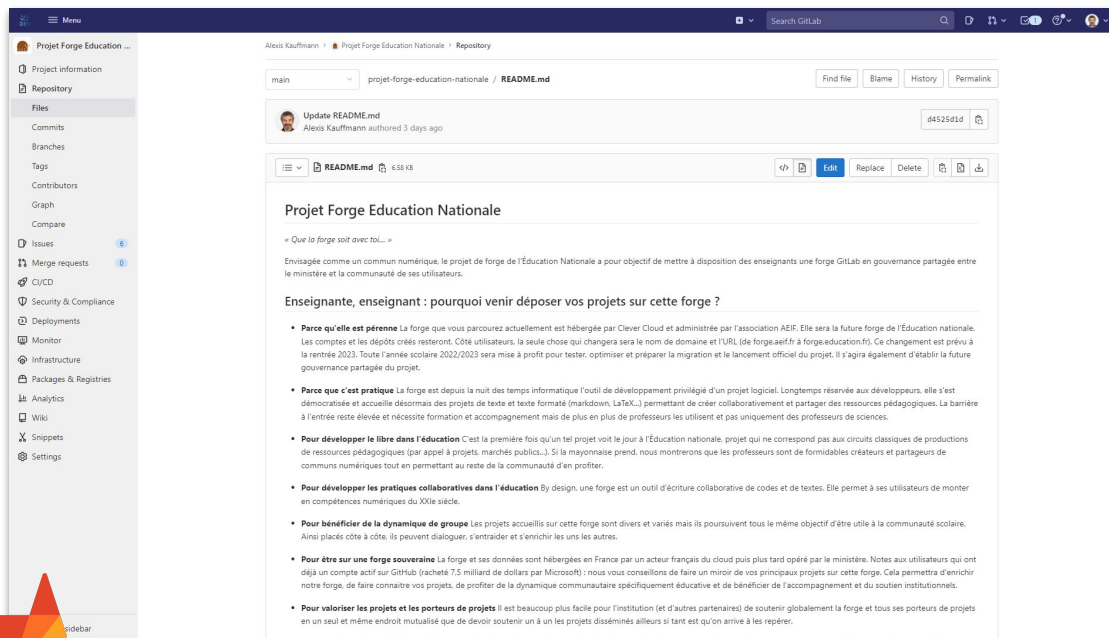


A Digital Educational Commons Forge

French Ministry of Education



Alexis Kauffmann
Math teacher
Founder of Framasoft.
Free Software and
Open Educational Resources
Project Manager
at the French Ministry of Education.



A GitLab forge for all teachers and students in France?

France has nearly a million teachers.

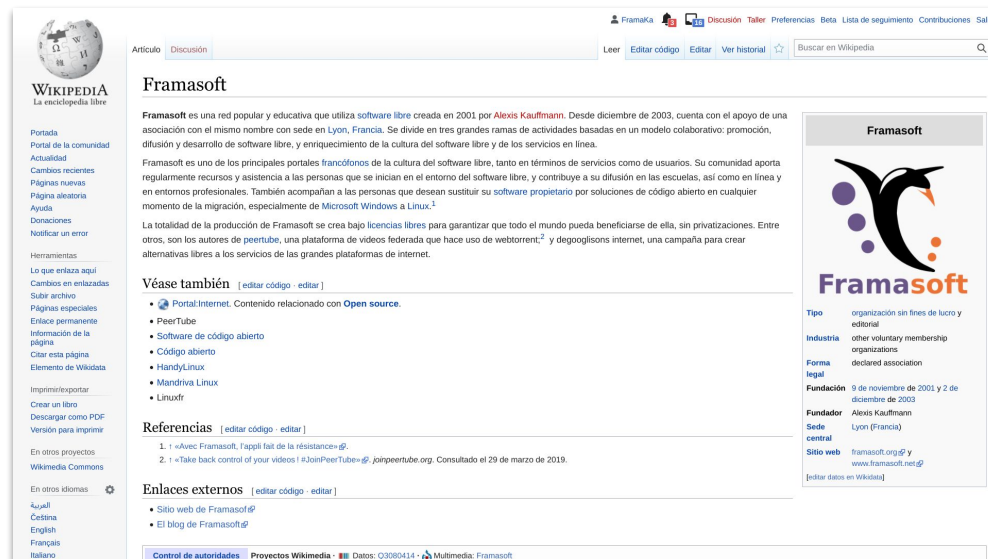
Among them, there are teachers developing Open Source softwares but there are also more and more teachers using markdown + mkdocs + pages to create and share educational content.

Yesterday all these projects were scattered over multiple forges including Microsoft's GitHub which is difficult to trust in the long term.

With the support of the French Ministry of Education, the "Forge of Digital Educational Commons" project aims to pool all these projects on a dedicated GitLab instance and invite the school community to participate.



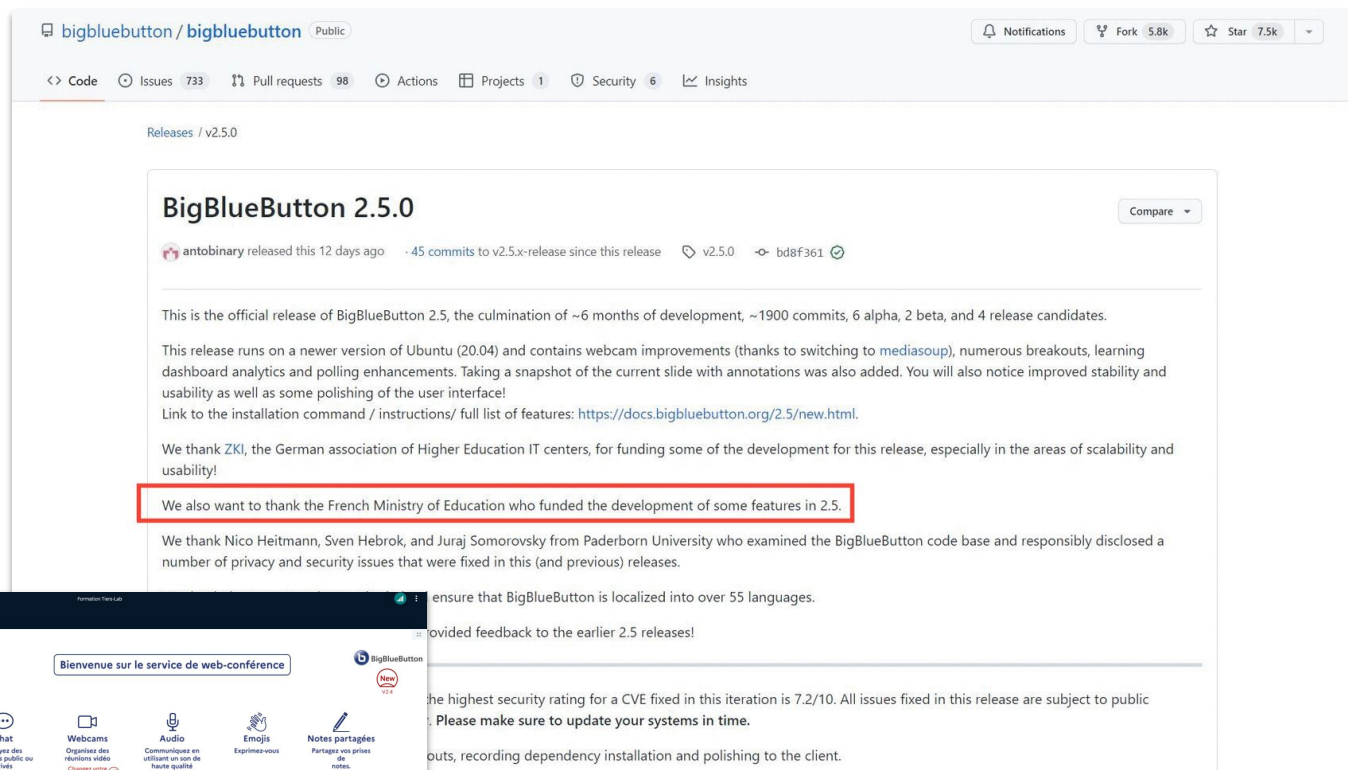
A few words about me...



The screenshot shows the Wikipedia article for Framasoft. The article describes Framasoft as a popular and educational network using free software, created in 2001 by Alexis Kauffmann. It mentions that since December 2003, it has received support from a association with the same name, based in Lyon, France. The article also notes that Framasoft is one of the main French-speaking portals for free software culture, offering services to users and contributing to the diffusion of free software in schools and professional environments. It also mentions that Framasoft creates free licenses to ensure that the world can benefit from it, without privatization. The article lists various projects and services offered by Framasoft, including Portal Internet, PeerTube, and various free software projects like Vandy Linux, Mandriva Linux, and Linuxfr.



"We want to thank the French Ministry of Education who funded the development of some features in 2.5."



bigbluebutton / bigbluebutton Public

<> Code Issues 733 Pull requests 98 Actions Projects 1 Security 6 Insights

Releases / v2.5.0

BigBlueButton 2.5.0

antobinary released this 12 days ago · 45 commits to v2.5.x-release since this release · v2.5.0 · bd8f361

This is the official release of BigBlueButton 2.5, the culmination of ~6 months of development, ~1900 commits, 6 alpha, 2 beta, and 4 release candidates.

This release runs on a newer version of Ubuntu (20.04) and contains webcam improvements (thanks to switching to mediasoup), numerous breakouts, learning dashboard analytics and polling enhancements. Taking a snapshot of the current slide with annotations was also added. You will also notice improved stability and usability as well as some polishing of the user interface!

Link to the installation command / instructions/ full list of features: <https://docs.bigbluebutton.org/2.5/new.html>.

We thank ZKI, the German association of Higher Education IT centers, for funding some of the development for this release, especially in the areas of scalability and usability!

We also want to thank the French Ministry of Education who funded the development of some features in 2.5.

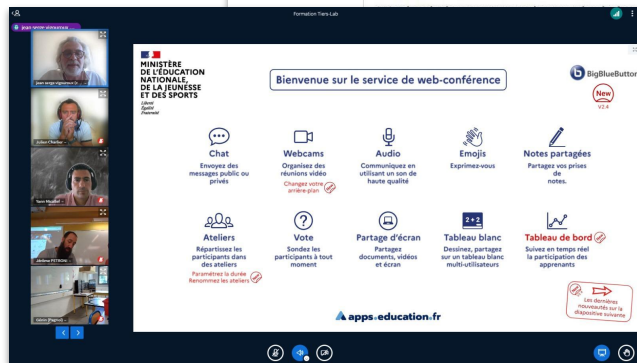
We thank Nico Heitmann, Sven Hebrok, and Juraj Somorovsky from Paderborn University who examined the BigBlueButton code base and responsibly disclosed a number of privacy and security issues that were fixed in this (and previous) releases.

ensure that BigBlueButton is localized into over 55 languages.

provided feedback to the earlier 2.5 releases!

the highest security rating for a CVE fixed in this iteration is 7.2/10. All issues fixed in this release are subject to public. Please make sure to update your systems in time.

outs, recording dependency installation and polishing to the client.



A forge to meet classic needs



A forge to develop new educational uses

The screenshot displays a web browser window with a French educational resource. The page is titled "Algorithme avancé des graphes" and contains a section on "Graphe bipartite" and "TP Couplage".

Problem Statement: On considère un chemin entre deux sommets x et y dans un graphe non-orienté pondéré. La largeur de ce chemin est le plus petit poids des arêtes présentes dans ce chemin. Le chemin vide de x à y est de largeur $-\infty$.

Exercises:

1. Prouver que l'arbre couvrant de poids maximal contient les chemins les plus larges entre toute paire de sommets.
2. Décrire un algorithme pour résoudre en temps $O(|S| + |A|)$ le problème suivant : étant donné un graphe non-orienté pondéré $G = (S, A)$, $x, y \in S$ et $W \in \mathbb{Z}$, est-ce que la distance de largeur d'arête entre x et y est inférieure ou égale à W ?
3. On suppose que la distance de largeur d'arête entre x et y est B .
 1. Prouver que la suppression d'une arête de poids inférieur à B ne change pas cette distance.
 2. Prouver que la contraction d'une arête de poids plus grand que B ne change pas cette distance. La contraction d'une arête (u, v) revient à identifier u et v , si cette contraction crée des arêtes parallèles, on ne conserve que l'arête de plus grand poids.

5 Travaux pratiques

5.1 TP Couplage

5.1.1 Graphe bipartite

On va considérer ici un graphe bipartite donné par une liste d'arêtes sous la forme $\{(source, but)\}$. Voici un exemple :

```
l = [(0,3), (1,3), (1,4), (2,3), (2,4)]
```

En commençant à numérotiser à 0, on peut donc supposer que le nombre de sommets est égal au plus grand entier apparaissant dans un couple plus un.

Question

Écrire une fonction `nombre_somets : (int * int) list -> int` qui calcule ce nombre.

Preuve

```
1 let rec nombre_somets l =
2   match l with
3   | [] -> 0
4   | (a,b)::q -> let n = nombre_somets q in
5     max (1 + max a b) n
```

Code Editor:

```
(* BEGIN NOMBRE_SOMETS *)
let rec nombre_somets l =
  match l with
  | [] -> 0
  | (a,b)::q -> let n = nombre_somets q in
    max (1 + max a b) n
(* END NOMBRE_SOMETS *)

(* BEGIN INTERSECTION *)
let rec intersection l1 l2 =
  List.filter (fun x -> List.mem x l2) l1
(* END INTERSECTION *)

(* BEGIN CONS_UNIQ *)
let cons_uniq t q =
  if List.mem t q
  then cons_uniq t q
  else t::q
(* END CONS_UNIQ *)

let rec couplage (l : (int * int) list) : (int * int) list =
  let rec subsequence (l : (int * int) list) : (int * int) list =
    let rec subsequence_aux (l : (int * int) list) : (int * int) list =
      match l with
      | [] -> []
      | (a,b)::q -> let n = nombre_somets q in
        max (1 + max a b) n
    in
    subsequence_aux l
  in
  subsequence l
```

Travaux pratiques

V.1 TP Couplage

V.1.1 Graphe bipartite

On va considérer ici un graphe bipartite donné par une liste d'arêtes sous la forme $\{(source, but)\}$. Voici un exemple :

```
l = [(0,3), (1,3), (1,4), (2,3), (2,4)]
```

En commençant à numérotiser à 0, on peut donc supposer que le nombre de sommets est égal au plus grand entier apparaissant dans un couple plus un.

Question V.1 Écrire une fonction `nombre_somets : (int * int) list -> int` qui calcule ce nombre.

Preuve

```
1 let rec nombre_somets l =
2   match l with
3   | [] -> 0
4   | (a,b)::q -> let n = nombre_somets q in
5     max (1 + max a b) n
```



Markdown + MkDocs + Pages

The left screenshot shows a web interface for editing a markdown file. The top bar indicates the file path: 'main / pratique / docs / N0 / 1-somme / sujet.md'. Below the path, there are buttons for 'Find file', 'Blame', 'History', and 'Permalink'. A user profile 'autre exemple' is shown with the text 'Francky SPOJ authored 6 months ago' and a commit hash '520a76d9'. The main content area shows the markdown file 'sujet.md' with a preview of the rendered content: 'author: Franck Chambon', 'title: Somme de deux entiers', and 'tags: - 0-simple'. Below the preview, there is a section titled 'Somme de deux entiers' with a description: 'Compléter la fonction ci-dessous pour qu'elle renvoie la somme des deux arguments a et b.' and an example of the function being used.

The right screenshot shows a documentation page for 'Somme de deux entiers'. The top bar indicates the page title 'e-nsi : Exercices pratiques'. Below the title, there is a search bar and a list of tags: 'Accueil', 'Tags', 'Parcours', 'Pour démarrer', 'À maîtriser', 'Guidés', 'Aventure', 'Mathématiques'. The main content area shows the title 'Somme de deux entiers' and a description: 'Compléter la fonction ci-dessous pour qu'elle renvoie la somme des deux arguments a et b.' Below the description, there is a section titled 'Exemple' with a code editor and a console. The code editor shows a Python function definition: 'def somme(a, b):' and a return statement: 'return ... # à compléter ici'. The console shows the output of the function: 'somme(10, 32) == 42' and 'somme(100, 7) == 107'. At the bottom right, there is a status bar indicating 'Evaluations restantes : 5/5'.

CI/CD

Mastodon, plutôt que Twitter

Pourquoi ?

📄 Un logiciel libre et *open-source* sans publicité et sans *tracking*

🌟 Des serveurs décentralisés, mais interconnectés

- Choix possible d'un serveur plus cohérent avec ses valeurs, ses centres d'intérêt
- Possibilité d'accéder cependant aux comptes sur les autres serveurs
- Possibilité de migrer sans difficultés vers un autre serveur ou d'utiliser une autre application Pleroma, Misskey
- Interaction possible avec d'autres applications : Peertube, PixelFed ...

👤 Des communautés plus respectueuses

Comment ?

1 Choisir son serveur

- On peut faire une recherche par mot-clé sur [instances.social](#)
- Bien regarder les conditions d'utilisation : chaque serveur a ses règles spécifiques

2 Créer son profil

- Faire une présentation avec le hashtag [#introduction](#)
- Mettre d'autres hashtags pour pouvoir être suivi-e par d'autres comptes

3 Suivre d'autres comptes

- Trouver des comptes avec les mêmes centres d'intérêt
- Regarder le fil local de votre serveur
- Regarder le fil global

- Faire une recherche avec un hashtag
- Utiliser un répertoire de comptes : [Trunk](#), [Fediverse](#)

Plus d'informations

- Voir [l'aide en ligne](#)
- Choisir [une appli pour téléphone](#)

main mindmap / mastodon-RAW.md

eysette update mastodon ✓

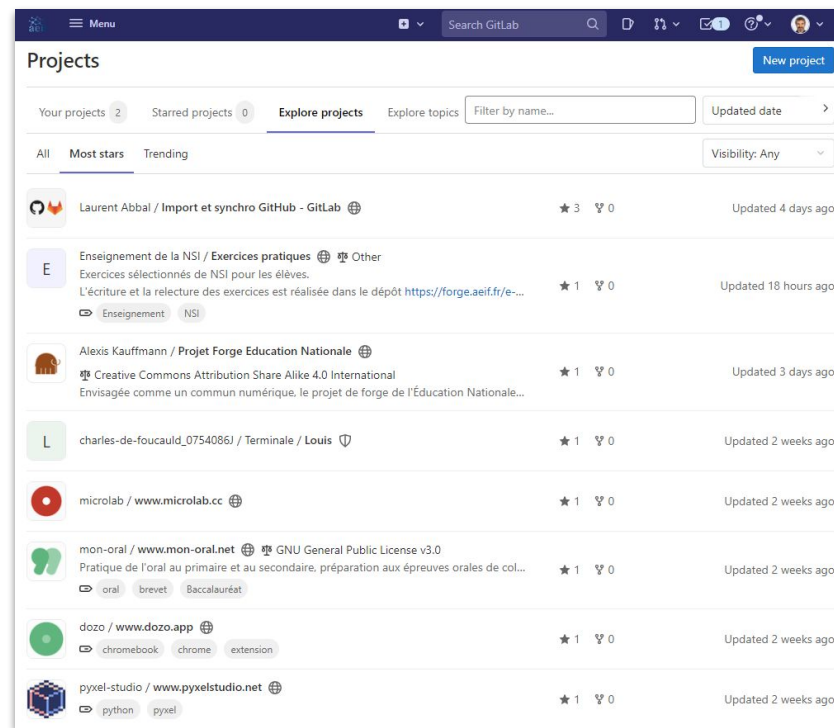
1 contributor

28 lines (25 sloc) | 1.4 KB

```
1 # Mastodon, plutôt que Twitter
2
3 ## Pourquoi ?
4 - Un logiciel libre et open-source sans publicité et sans tracking
5 - Des serveurs décentralisés, mais interconnectés
6 - Choix possible d'un serveur plus cohérent avec ses valeurs, ses
7 - Possibilité d'accéder cependant aux comptes sur les autres serveurs
8 - Possibilité de migrer sans difficultés vers un autre serveur ou
9 - Interaction possible avec d'autres applications : Peertube, PixelFed ...
10 - Des communautés plus respectueuses
11
12 ## Comment ?
13
14 - Choisir son serveur
15 - On peut faire une recherche par mot-clé sur [instances.social](https://instances.social/)
16 - Bien regarder les conditions d'utilisation : chaque serveur a ses règles spécifiques
17 - Créer son profil
18 - Faire une présentation avec le hashtag #introduction
19 - Mettre d'autres hashtags pour pouvoir être suivi-e par d'autres comptes
20 - Suivre d'autres comptes
21 - Trouver des comptes avec les mêmes centres d'intérêt
22 - Faire une recherche avec un hashtag
23 - Utiliser un répertoire de comptes : [Trunk](https://communitywiki.org/trunk/), [Fediverse](https://fediverse.info/)
24 - Regarder le fil local de votre serveur
25 - Regarder le fil global
26 - Plus d'informations
27 - Voir [l'aide en ligne](https://mastodon.help/)
28 - Choisir [une appli pour téléphone](https://mastodon.help/#MastodonOnSmartphones)
```



And here is our forge beta version
already in production



forge.aeif.fr



Digital strategy for education 2023-2027

A strategic vision of a shared public policy

Par ailleurs, les communautés d'enseignants (et d'autres acteurs de l'éducation) peuvent également être des lieux de construction de nouveaux outils. Des professeurs, notamment de NSI ou de SNT, sont en attente d'une « forge » qui leur permettrait de collaborer entre pairs et de partager du code informatique. Le ministère répond à ce besoin avec la mise à disposition d'une forge technologiquement souveraine et mutualisée à l'échelle nationale.

Numérique pour l'éducation 2023-2027

La vision stratégique
d'une politique publique
partagée

Janvier
2023

1 forge nationale pour accompagner
et favoriser la production et le partage
des communs numériques

"A national forge to
support and promote
the creation and
sharing of digital
commons"



FOSDEM

A little geek thing or a great potential?

strengths, opportunities:

- **institutional** support
- a project that carries **values**: openness, sharing, ethics, sovereignty, sustainability
- encourages teachers and students to **work together**
- increase in digital and computer **skills**
- by **gathering our forces** in the same place, it is easier to make community and to obtain means
- have a **large bank** of applications and educational content under open licenses available to the entire school community
- encourage teachers to share their content and promote them for this
- digital **sobriety** (light and open format, offline work, improvement of the existing rather than reinventing the wheel)
- a digital **commons** project, with shared policy and governance with users
- not very expensive (at least at first)
- *etc.*

weaknesses, threats:

- tool too complex to understand and use, do not take off, remain a little geek thing for a minority (great needs for documentation, training and community animation)
- fail to enlarge the circle of users
- convince great teachers who are already on forges, including GitHub, to migrate to our forge
- copyright (the productions of state agents belong to the agents or to the state?)
- dependency on GitLab and its evolutions (GitHub competition with Copilot & co)
- *etc.*



Mastodon: @lelibredu@mastodon.mim-libre.fr
Twitter: [@LeLibreEdu](https://twitter.com/LeLibreEdu)

Thank you for your attention

