

PLaTon

Platform for **L**earning **a**nd **T**eaching **o**nline

— *un outil pour apprendre et pour enseigner* —

DR MK

Platon

une initiative de l'Université Gustave Eiffel

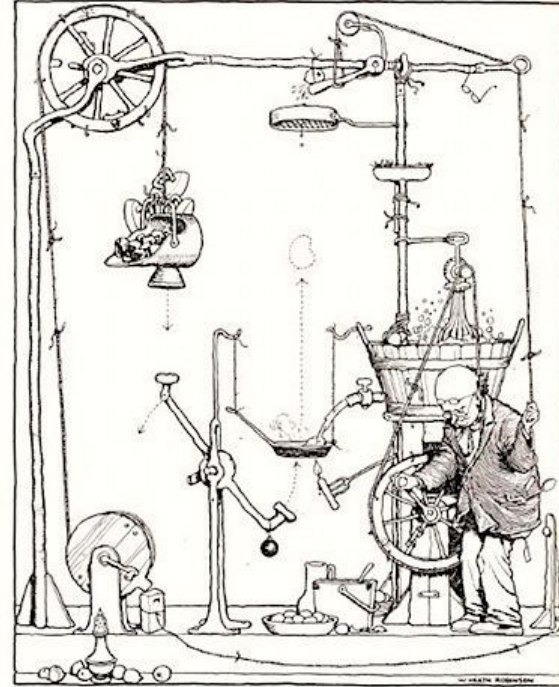
un développement logiciel fait par des apprentis et enseignants d'informatique et de mathématique.



Pourquoi un nouveau logiciel ?

Pour

- profiter des **avancées technologiques**
- solutions actuelles difficilement **extensibles**
 - Noyau en C (WIMS)
 - PHP (moodle)
- traiter des **besoins non couverts**
 - hybridation
 - scénarisation
 - editorialisation
 - exécutions sécurisées des codes d'élèves
- **ergonomie** perfectible des solutions actuelles
 - interopérabilité avec les LMS
 - capacité d'intégration d'outils tiers
 - compatibilité PC, smartphone, tablette...



Héritage de WIMS



- des **ressources pédagogiques** disponibles, réutilisables, restructurables, modifiables, partagées
- une volonté de proposer des **exercices répétables**
- une partie de la **philosophie d'édition** (faciliter la **création d'exercices** variés, le **partage**, fédérer une **communauté**)
- les **notions** d'exercices, de ressources, de classe, de niveaux de programmation

de grandes différences :

- organisation du **code**, **langages**, **capacités fonctionnelles** de la plateforme (ex : co-édition, mode de partage, processus de montée en qualité)
- WIMS a montré la voie. Notre objectif : **capitalisation** sur ce qui fonctionne, **changements technologiques** sur ce qui bloque

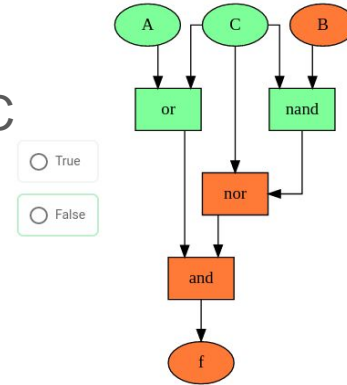
WIMS mène sa vie de son côté...

Platon pour l'apprenant



- exercices **aléatoires répétables** liés avec des **objectifs d'apprentissage**
- **activités guidées** par l'enseignant (présentiel / distanciel / asynchrone et synchrone)
- **pédagogie active** des apprenants (essai erreur feedback)
- **apprentissage personnalisé** (stratégies dynamiques, ontologie, recommandations)
- outils pour la **co-évaluation et auto-évaluation** (grilles critériées, ateliers)

Évaluer l'unique sortie f de ce circuit logique dont voici les valeurs des variables en entrées.
A : True, B : False, C : True.



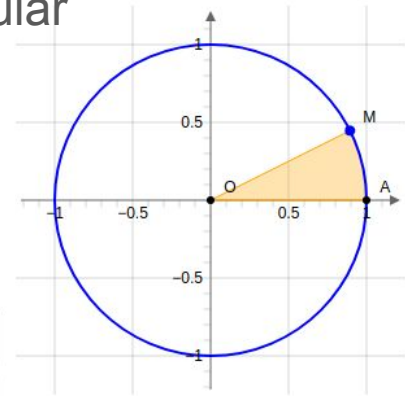
- True
- False

Exemples

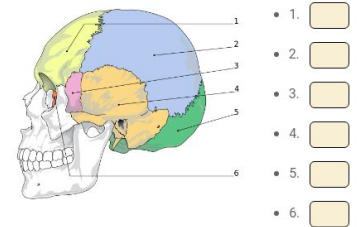
- les classiques QCM, Vrai/Faux, Numériques, etc, quizz, gift, AMC
- versions aléatoires (choix aléatoire, réponses mélangés, etc)
- Exercices de programmation:
 - tests de fonctionnement
 - comparaison a une solution
 - vérification de complexité, récursivité etc
- Exercices originaux basé sur un système de composants angular
 - <https://pl.u-pem.fr/components/input>

Angle

Placer le point M de sorte que l'angle (\vec{OA}, \vec{OM}) ait une mesure égale à $\frac{7\pi}{4}$.



Os de la voûte crânienne



- 1.
- 2.
- 3.
- 4.
- 5.
- 6.

- Os frontal
- Os pariétal
- Os sphénoïde
- Os temporal
- Os occipital
- Os ethmoïde

Moyenne des éléments d'un tableau

Écrire une fonction C **average_array** qui prend en argument un tableaux d'entiers (pointeur vers la première case) et sa taille puis retourne un flottant qui devra être la moyenne des éléments contenus dans le tableaux. Par convention, la moyenne d'un tableau vide devra être zéro.
Rappel: la moyenne est la somme de tous les éléments divisée par le nombre d'éléments.

```

1 float average_array(int* array, int size){
2     int i;
3     float res=0;
4
5     if (size == 0)
6         return 0;
7     for (i=0; i<size; i++){
8         res += array[i];
9     }
10    return res / size;
11 }
  
```

PLATON 2019-2020 en chiffres

150 107 corrections d'exercice en direct effectuées par les sandbox (24h/24 et 7j/7)
58 984 réponses parfaites des apprenants (note maximale 100/100 obtenue)

1 110 utilisateurs actifs

5239.0 h 35.0 m temps de travail cumulé des utilisateurs sur PLaTon

4.0 h 43.0 m temps de travail moyen pour chacun des 1110 utilisateurs

34 enseignants-éditeurs de ressources

1324 fichiers ressources pédagogiques d'extension .pl

18345 actions d'édition sur les ressources (ajout, suppr. ou modif. de fichier)

31 cours PLaTon créés

615 exercices auto corrigés déployés

88 activités (fiches de TP) déployées

5 relances manuelles des sandbox suite à blocage

2 universités sur le serveur MLV : U. Gustave Eiffel et U. Paris Saclay

PLaTon, au service des profs

The screenshot shows the PLaTon interface. The code in the editor includes:

```
1 extends = ../templates/editor.pl
2
3
4
5 generate== #|py|
6 max_syms = 3
7 max_factor_length = 3
8 alphabet, factor, viewer = Generator.factor(max_syms
9 alphabet = '.,'.join(List(alphabet))
10
11
12 title = Facteur
13 text ==
14
15 cette question, on considère l'alphabet  $\Sigma = \{a, b, c\}$ .
16 Donnez un automate reconnaissant le langage  $L1 = \{ \text{mots}
17 class="language" data-bbox="615 145 635 165">L1 = \{ \text{mots}
18 class="language" data-bbox="615 165 635 185">ayant ecc en facteur \}$ .
19 </p>
20 </endit>
21
22
23
24
25
```

The preview window displays the question: "Facteur. Pour cette question, on considère l'alphabet $\Sigma = \{a, b, c\}$. Donnez un automate reconnaissant le langage $L1 = \{ \text{mots ayant ecc en facteur} \}$. Il vous reste 5 tentative(s) avant de voir une solution !". Below the question are buttons: "Initial", "Non final", "Renommer", and "Supprimer". A state transition diagram is shown with states S0, S1, S2, and S3, and transitions labeled 'c'.

- les profs construisent leurs **triforces** (objectif / activité / évaluation)
- définir les **objectifs**, construire une **stratégie** de cours
- les profs **choisissent** / **modifient** / **fabriquent** les exercices (surtout pour les étudiants les moins experts dans le domaine ou en apprentissage)
- qui déterminent les **feedbacks** (déclenchement et contenu).
- qui **organisent les interactions** entre élèves, tuteurs, profs (websockets)
- l'évaluation est automatique et paramétrée par l'enseignant ou faite par l'enseignant et ou les pairs ou encore en auto-évaluation.

Nos forces

Réutilisabilité des éléments constituant un exercice ou une activité.

Formalisation du lien entre les concepts les exercices et les certifications.

Curation co-évaluation des ressources pédagogiques

Amélioration continue (Separation of concerns)

Communauté les cercles et le repository unique des ressources.

L'exercice du futur est déjà là...

L'**articulation technique à plusieurs niveaux** de la plateforme est un gage d'évolution des exercices à l'avenir.

Les **composants** qui proposent des interfaces de saisie

Un langage de description des interfaces : un **markdown étendu** qui allie simplicité d'écriture et puissance expressive avec un comportement responsive.

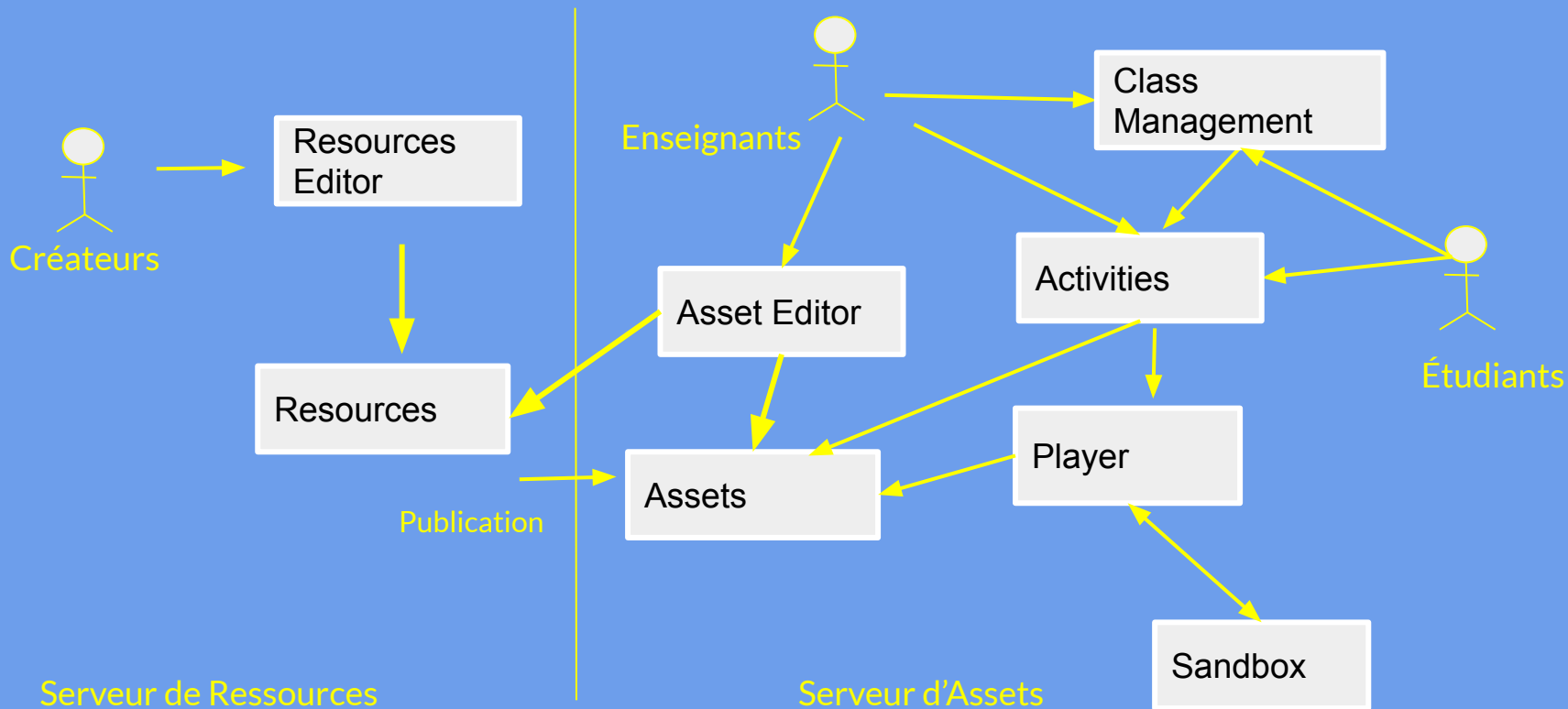
Un langage de programmation des évaluations : **Python3** avec une librairie de fonctions adaptées à chaque disciplines.



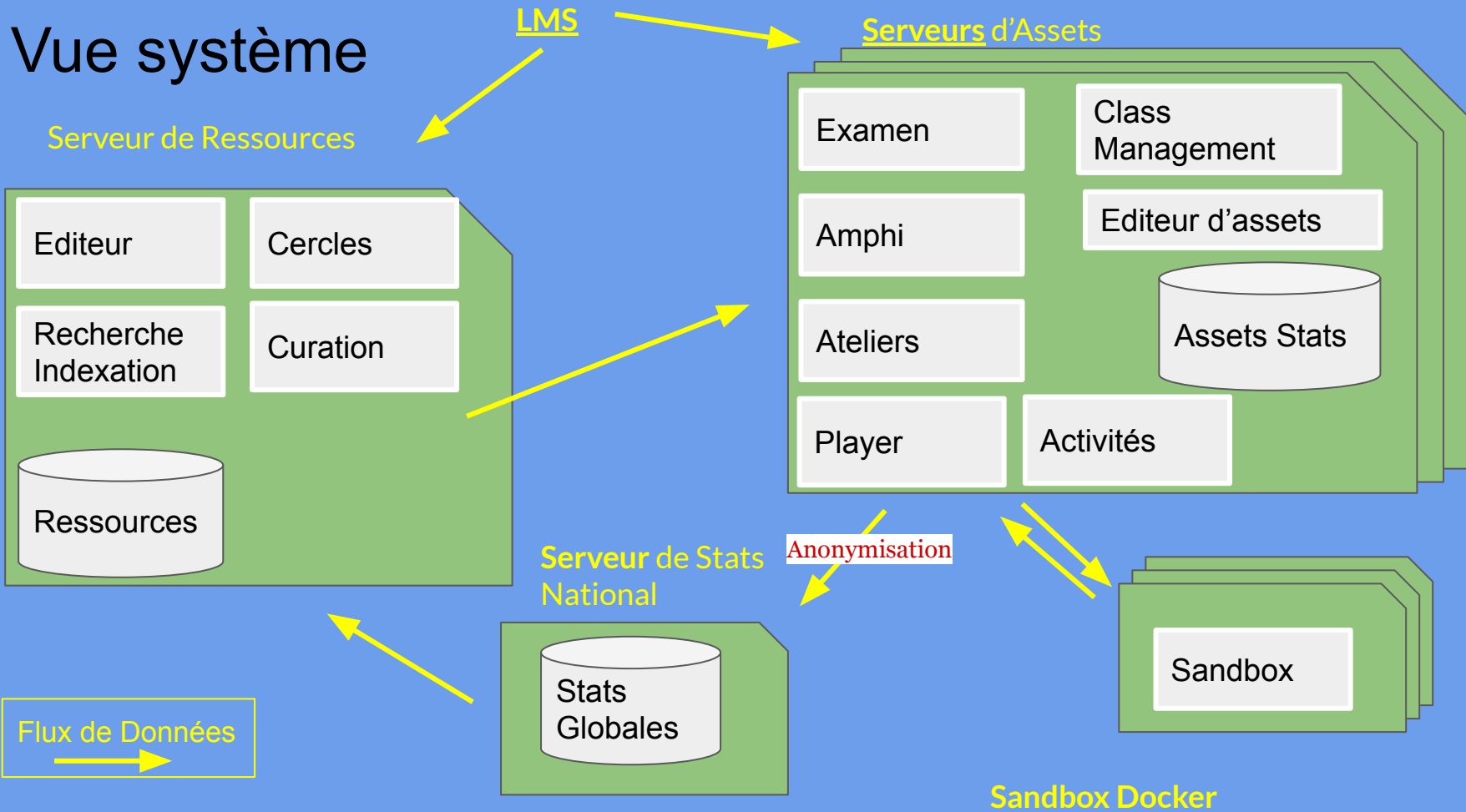
→ **La limite est votre imagination...**



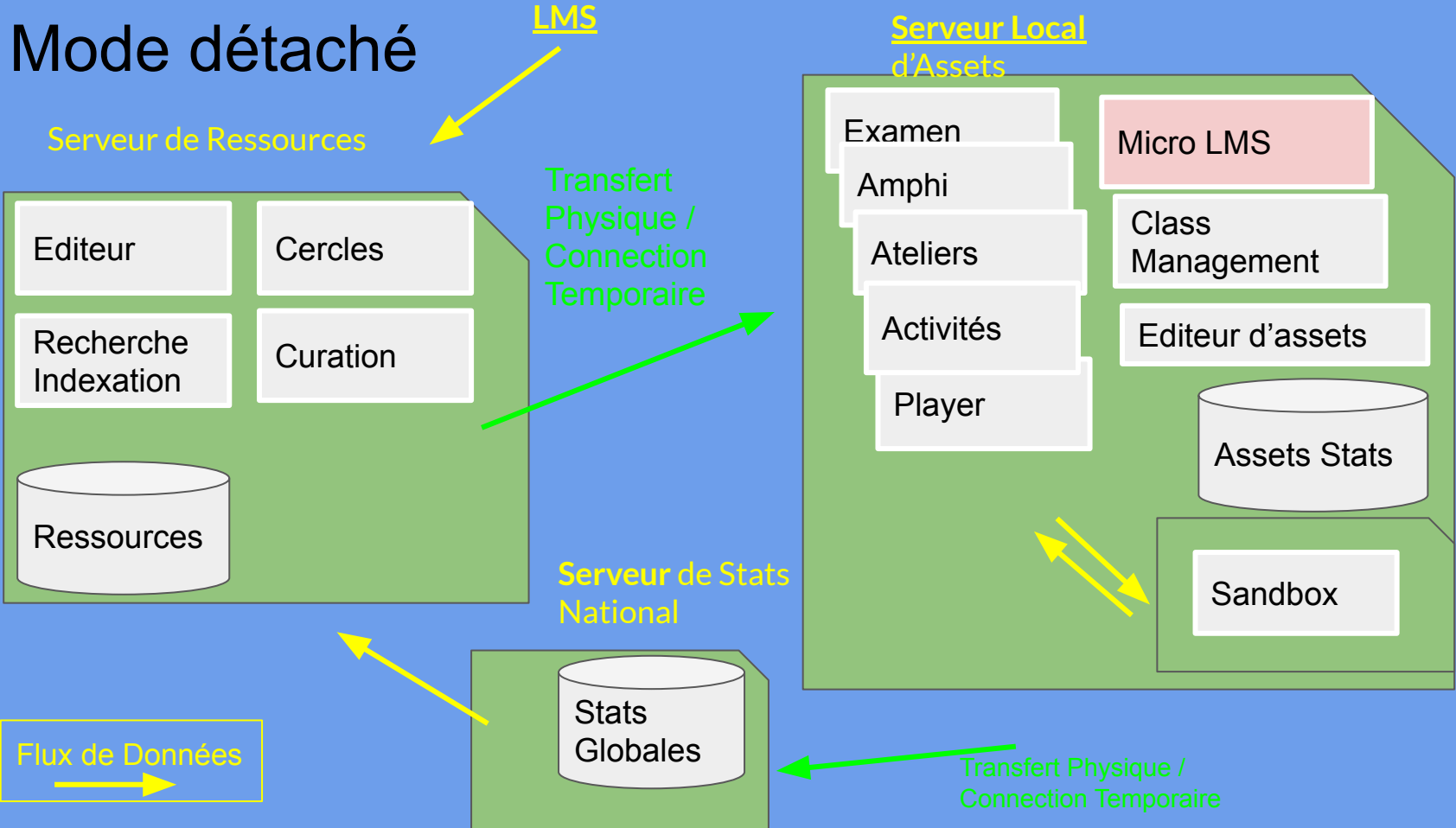
Vue utilisateur



Vue système



Mode détaché



Besoin Dev Ops

Dockerisation à la fois pour le déploiement mais aussi pour la gestion du cloud

- seule la sandbox est “dockerisé” (interne externe).
- le serveur d’asset doit le devenir (connecté et local)
- Gestion de la distribution de charge (Nécessité ...)

Micro LMS

- besoin d’un connecteur d’identification (CAS,etc)
- Connection data récupération d’information d’inscription (annuaire).
- Intégration dans un ENT

Philosophie de l'amélioration continue

licence libre,

chacun peut contribuer à la wikipédia.

L'équipe



Christophe, Adam, Quentin, Yohan, Killian, Magdalena, Dominique, Nicolas, Marie, Olivier, Anna,

Des absents : Mamadou, David, Etienne, Damir,

Vivement le retour à la normale.

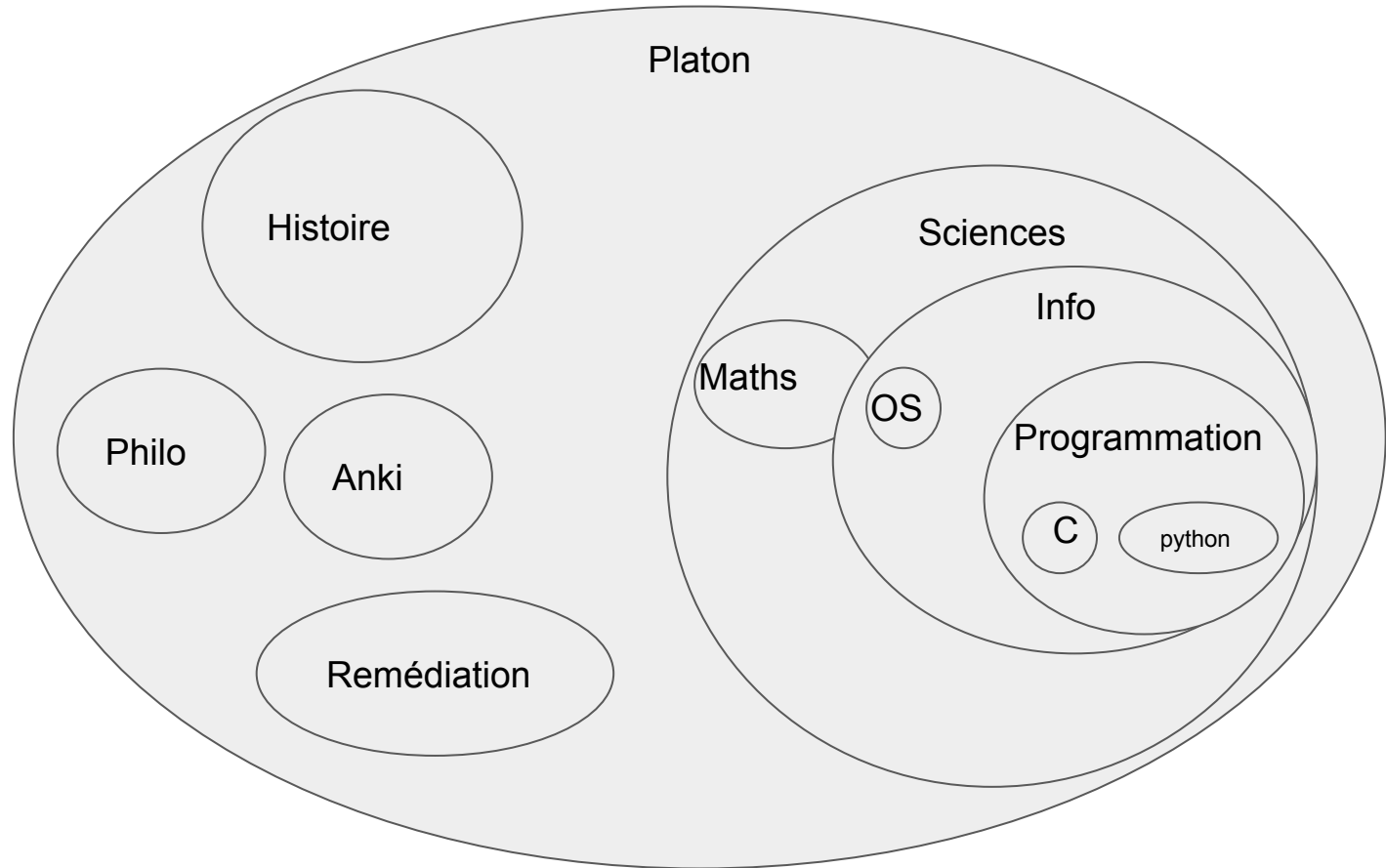




PLaTon

Vos questions ?

Les Cercles



Les Cercles

Un **président**, un **responsable scientifique**, un **modérateur** (trois rôles).

-> définir les objectifs du cercle, la politique -> le wiki

-> construire les outils spécifiques au cercle -> l'atelier -> les établis

-> gérer la communauté -> accueil organisation forum etc

Historique

- 2013 : Dans le cadre d'IDEA (PIA 2) création d'un outil de gestion d'ontologies
- 2014-16: Tentatives de développement dans moodle et avec PHP.
- 2017: Retour vers le Python et LTI (learning tools interoperability)
- 2018: Version de démonstration des principes.
- 2019: Sandbox, Composants, Langage, premières utilisations.
- 2020: Version 1.0 Assets/ Ressources, Markdown, Websockets, ...amphi, ateliers, Éditeur partagé, etc

PL-WIMS

WIMS-ED

WIMS-EVO

Historique

2013 : Dans le cadre d'IDEA (PIA 2) création d'un outil de gestion d'ontologies

2014-16: Tentatives de développement dans mon langage avec PHP.

2017: Retour vers le Python et LTI (learning tool interoperability)

2018: Version de démonstration des nouvelles fonctionnalités

2019: Sandbox, Composants, Langage de description de nouvelles utilisations.

2020: Version 1.0 Assets/ Plugins, Markdown, Websockets, ...amphi, ateliers, Éditeur partagé, etc

SARS-COVID-19

Historique

...

2019: Sandbox, Composants, Langage, ... dernières utilisations.

2020: Version 1.0 Assets/ Ressources, Markdown, Websockets, ...amphi, ateliers,
Éditeur partagé, etc

SARS-COVID-19

2020: Une version 0.7 fonctionnelle (voir chiffres).

Sandbox Version 3.1: choix des commandes, répartition de charge,
Composants

Feuille de route

Editeur d'assets.

Version asynchrone des applications et utilisation de Websockets, amphi, atelier, editeur, outils collaboratifs.

Base de Donnée spécifique (pour l'éditeur partagée et les cercles).

Nouvelle version des composants

Utilisation d'un serveur node.js pour le front.

Monté en charge, serveur d'assets/ressources.

Dashboards , exercices préparé, éditeur d'activités.

Platon et ESUP

- rentrer dans l'incubateur pour avoir de l'aide sur les aspects techniques
- inviter dans nos cercles
- financement des apprentis (besoin de 4 apprentis en continue, pour l'instant financer par UGE et un projet AMI), des enseignants chercheurs.

Pour atteindre l'objectif de vie, il faut des ingénieurs, des enseignants chercheurs et continuer avec au moins 4 apprentis pour développer la base logicielle.

Le budget annuel de PLATON est de 90K€ qui est financé par moitié par l'université Gustave Eiffel, nous cherchons à pérenniser un soutien externe qui est pour le moment fait par le projet.

La documentation

- Un tutoriel pour l'utilisation directe.
- Une documentation composant par composant
- Un site à la "stackoverflow" pour la communauté
- Un doc/wiki

PLaTon : une architecture logicielle

Dessins : serveur central, serveur d'assets, tout sous git, le front le back, partage d'édition,

La video a jour du développement (3 ans et pas 2 ans :-)

https://www.youtube.com/watch?time_continue=223&v=yecgE1SkWBI&feature=emb_logo

La syntaxe de Mamadou markdown+++