



# Workshop Le MOOC

## De la correction automatisée

C. Queinnec  
UPMC - LIP6

# Correction automatisée

## *Un cours (avec programmation) ?*

- des connaissances
  - théorie, documents (livres, manuels, guides, ...), ressources (sites, forums, ...)
- un savoir-faire
  - raisonner, programmer
- des vérifications
  - exercices, devoirs, examens
- un scénario pour tout lier

*La moitié environ des cours de licence utilisent un quelconque langage de programmation.*

## *Buts*

*C'est en programmant que l'on devient programmeur !*

- On programme au sein d'un environnement de programmation (documentation en ligne, aides diverses à la saisie, tests, . . .)
- On s'entraîne à programmer avec ces outils
- On s'imprègne du paradigme dominant « les programmeurs aiment tester ! »

## *Buts*

*C'est en programmant que l'on devient programmeur !*

- On programme au sein d'un environnement de programmation (documentation en ligne, aides diverses à la saisie, tests, . . .)
- On s'entraîne à programmer avec ces outils
- On s'imprègne du paradigme dominant « les programmeurs aiment tester ! »

La correction doit être automatisée et ainsi être

- professionnalisante
- inlassable
- uniforme

## *Première expérimentation*

2001-2006 : **LI101 Programmation récursive**  
(cours d'introduction à la science informatique)



- Livre de cours, d'exercices, annales corrigées et commentées, bandes son, video du cours
- CDrom avec environnement de développement
- QCM et exercices corrigés automatiquement pour travail personnel et autonome.

## *Deuxième expérimentation*

- 2001-2004 : **CFS** Contrôle final semestriel pour L3
  - Plusieurs centaines d'étudiants
  - Examen réalisé sur ordinateur
  - Examen rejouable : annales dynamique (E.Chailloux)
- depuis : 4 cours (licence et master) organisant régulièrement des examens sur ordinateur et corrigés plus ou moins automatiquement

# *Leçons*

- notation progressive (pas de tout ou rien)
- verbalisation du processus
- pas de question oui-non
- pas d'aléa non maîtrisé
- attention aux dépendances globales (détection de plagiat, compétition, etc.) qui passent mal à l'échelle



## *Conclusions sur cette aventure*

- Techniquement, ça marche !
- Avis mitigé des étudiants
  - pas de travail en groupe,
  - stressant,
  - machine ne corrige pas comme les enseignants
- Investissement conséquent pour les enseignants
  1. énoncé
  2. solution
  3. notateur
- préparation méticuleuse

## *Correction automatisée*

- Tests unitaires (en boîte noire ou blanche)
- par assertion
- par comparaison à une solution

# *Squelette d'un correcteur de question*

1. Vérification des fichiers attendus
2. Affichage des fichiers soumis
3. Boucle sur les tests
  - 3.1 Obtenir résultat étudiant
  - 3.2 Montrer résultat étudiant
  - 3.3 Normaliser résultat étudiant
  - 3.4 Montrer résultat étudiant normalisé
  - 3.5 Appréciation résultat (ici par comparaison)
    - 3.5.1 Obtenir résultat auteur
    - 3.5.2 Montrer résultat auteur
    - 3.5.3 Normaliser résultat auteur
  - 3.6 Détermination du gain final

et ajout de la verbalisation et détermination des gains

# *Infrastructure de correction*

Création d'une infrastructure (et d'un écosystème)

- faciliter l'écriture d'exercices
- multi-langages de programmation
- déploiement aisé
- passage à l'échelle
- robuste

## *Précautions techniques*

Programme d'un apprenant :

- une fonction C, Scheme, Caml, ...
- une classe Java
- une page PHP
- un script Shell, Perl
- etc.

## *Précautions techniques*

Programme d'un apprenant :

- une fonction C, Scheme, Caml, ...
- une classe Java
- une page PHP
- un script Shell, Perl
- etc.

Mais

- bogue
- boucle en temps
- sorties en excès
- bloqué
- et même malveillant

## *Précautions techniques*

Programme d'un apprenant :

- une fonction C, Scheme, Caml, ...
- une classe Java
- une page PHP
- un script Shell, Perl
- etc.

Mais

- bogue
- boucle en temps
- sorties en excès
- bloqué
- et même malveillant

*Une solution : le confinement*

## *Autres précautions techniques (suite)*

Programme d'un auteur d'exercices :

- tests unitaires
- inspection source (indentation, findbugs, ...)



## *Autres précautions techniques (suite)*

Programme d'un auteur d'exercices :

- tests unitaires
- inspection source (indentation, findbugs, ...)

Mais

- bogue aussi

*Une même solution : le confinement*

## *FW4EX : Infrastructure de correction*


- Définition d'une grammaire pour la description d'un exercice (XML)
- Définition d'un format d'empaquetage (.tgz)
- Définitions de protocoles REST d'interaction avec l'infrastructure

### Actuellement

- tourne depuis 2008
- 64 exercices ou examens (et JFP) déployés
- 36000 corrections depuis 2008
- 650 étudiants concernés
- plus de 1000 réponses pour certains exercices











# Exemple récent : JFP5


# JFP5

déconnexion

Historique Profil Documentation Exercices Progression

Seules les équipes ayant soumis au moins une fois apparaissent ci-dessous.

242		<b>Giteam</b> (2013-05-29 12:40:16Z)
158		<b>LesAmine9.0</b> (2013-05-29 13:32:32Z)
154		<b>ErrSeg</b> (2013-05-29 13:34:41Z)
115		<b>Coderz</b> (2013-05-29 13:22:05Z)
103		<b>Echelon 3</b> (2013-05-29 12:50:16Z)
81		<b>Je relève le Delphi</b> (2013-05-29 13:35:12Z)
79		<b>The Greasemonkeys</b> (2013-05-29 13:10:44Z)
77		<b>Let w in</b> (2013-05-29 13:28:23Z)
76		<b>PM</b> (2013-05-29 13:15:32Z)
61		<b>pixotters</b> (2013-05-29 13:23:35Z)



# Exemple récent : JFP5

# JFP5

déconnexion

Queinnec

Historique Profil Documentation Exercices Progression

- Voir la [liste des équipes](#)
- Afficher la [position des équipes](#)

Pour voir un rapport, cliquez sur la ligne correspondante. Pour voir à nouveau la progression, cliquez sur le titre JFP5.

## JFP5

	De	Note / Total	Date	Soumission	Rapport
1	Je relève le Delphi	75 / 330	2013-05-29 13:37:37Z (+19)	<a href="#">lgz</a>	<a href="#">rapport</a>
2	pixotters	61 / 330	2013-05-29 13:37:34Z (+4)	<a href="#">lgz</a>	<a href="#">rapport</a>
3	Coderz	105 / 330	2013-05-29 13:37:10Z (+4)	<a href="#">lgz</a>	<a href="#">rapport</a>
4	PM	59 / 330	2013-05-29 13:37:03Z (+9)	<a href="#">lgz</a>	<a href="#">rapport</a>
5	LesAmine9.0	182 / 330	2013-05-29 13:36:39Z (+33)	<a href="#">lgz</a>	<a href="#">rapport</a>
6	Echelon 3	6 / 330	2013-05-29 13:35:15Z (+3)	<a href="#">lgz</a>	<a href="#">rapport</a>
7	pixotters	58 / 330	2013-05-29 13:35:04Z (+5)	<a href="#">lgz</a>	<a href="#">rapport</a>
8	Coderz	105 / 330	2013-05-29 13:34:45Z (+3)	<a href="#">lgz</a>	<a href="#">rapport</a>
9	Je relève le Delphi	81 / 330	2013-05-29 13:34:18Z (+21)	<a href="#">lgz</a>	<a href="#">rapport</a>
10	PM	61 / 330	2013-05-29 13:34:17Z (+8)	<a href="#">lgz</a>	<a href="#">rapport</a>
11	ErrSeg	154 / 330	2013-05-29 13:34:14Z (+20)	<a href="#">lgz</a>	<a href="#">rapport</a>

## *Apports de l'infrastructure*

- la jouissance de voir les copies se corriger. . .

## Apports de l'infrastructure

- la jouissance de voir les copies se corriger. . .
- Moins de maintenance pour les auteurs
  - Mémorisation de toutes les soumissions d'étudiants
  - Mémorisation des toutes les rapports de correction et notes
- Indépendance vis-à-vis
  - des outils d'ingénierie pédagogique
  - des bases de données scolaires
  - des modes d'authentification
- Exercices réutilisables
- Protection vie privée
- Topologiquement souple (pare-feux, etc.)
- Décorable (*skinnable*)

## Apports de l'infrastructure

- la jouissance de voir les copies se corriger. . .
- Moins de maintenance pour les auteurs
  - Mémorisation de toutes les soumissions d'étudiants
  - Mémorisation des toutes les rapports de correction et notes
- Indépendance vis-à-vis
  - des outils d'ingénierie pédagogique
  - des bases de données scolaires
  - des modes d'authentification
- Exercices réutilisables
- Protection vie privée
- Topologiquement souple (pare-feux, etc.)
- Décorable (*skinnable*)
- Éternité pour les exercices

mais éco-système à bâtir au-dessus

## *Conclusions partielles*

- La correction automatisée est mûre !
- Elle est un peu utilisée
- Elle a des aspects assez techniques
  - confinement
  - passage à l'échelle
- mais elle sort des ornières ordinaires
- et apporte des données pour une réflexion sur son efficacité



# À propos des MOOC

## *État des lieux*

- UPMC : premiers cours en ligne en 2000
- derniers en licence en 2008
- cours = plan + transparents
- + quelquefois examen, corrigé ?, TD/TP (texte, corrigé ?) ? ?
- et, rarement, bandes-son, video, correction automatisée

## État des lieux

- UPMC : premiers cours en ligne en 2000
- derniers en licence en 2008
- cours = plan + transparents
- + quelquefois examen, corrigé ?, TD/TP (texte, corrigé ?) ? ?
- et, rarement, bandes-son, video, correction automatisée

*Que manque-t-il pour en faire des MOOC ?*

- texte et autres ressources de référence
- chapitrage absorbable
- l'évaluation (correction automatisée, *crowd marking*)
- l'organisation des communautés (apprenants, enseignants)

# *Apports du monde académique*

*Et s'il n'y avait plus que des MOOC ?*

- Diplômation sûre et authentique
- Encadrement (*coaching*)
  - groupes de travail d'étudiants
  - explicitation des cours
- Stimulation pour apprendre

*Corollaires :*

- unification des programmes d'enseignement
- valeur universelle de l'ECTS
- classement des universités par valeur ajoutée

## *Où sont les 19 millions d'autres ?*

- 1 milliards d'apprenants
- 20 millions en L1

*Il n'y a donc pas que des MOOC (au début) !*

- focalisation sur L1, M1 (les plus rentables (*sic*))
- limitation de la concurrence
- définition d'un programme d'enseignement de l'informatique
- lutter contre le mal français : « non écrit par moi ! »
- accroissement du temps pour les recherches

## *Devoirs(?) du monde académique*

- Reconnaissance des auteurs (réputation, rétribution, modèles d'affaire, etc.)
  - avant, pendant, après
- Régler la propriété intellectuelle (entre livres et programmes)
- et le calcul des temps de service des enseignants
- Besoin de moins de concepteurs de cours (professeurs ?)
- mais de plus d'animateurs
- et de plus d'investissement pour la réalisation de cours attractifs

## *Pour en savoir plus*

- FW4EX
- JFLA 2010 et bande-son
- CDroms sur C et Scheme

