La recherche reproductible au Labex Archimède

Denis Arrivault[1]

[1]Labex Archimède Aix Marxeille Université

9 Décembre 2014 / ProDev2014

Outline

Outline

Remerciements

Coauteur

Thibaut Surier : stagiaire L3 de juin à août 2014 au sein du Labex, a développé la plateforme interne.

Contributeurs

- **Yvon Stroppa** (LEO/UMR7322) responsable du développement de la plateforme Exec&Share.
- **Cécile Capponi**, **Sandrine Anthoine** et **François-Xavier Dupé**, chercheurs au sein du Labex Archimède. Ils sont notre comité utilisateurs sur ce projet.

Outline

Le Labex Archimède.

- 4 Laboratoires (I2M, LIF, LSIS, CPT) + CIRM.
- 4 Missions :
  - renforcer les collaborations inter laboratoires,
  - favoriser les échanges internationaux,
  - développer les échanges de compétences et les transferts technologiques,
  - améliorer les liens entre la recherche et l'enseignement au sein de l'AMU.
- 1 Cellule d'expertise en développement de logiciels. Projets en cours :
  - MACAON, traitement automatique de la langue, indexation morpho-syntaxique de textes.
  - LTFAT-PYTHON, bibliothèque d'outils temps/fréquence en Python.
  - GOOL, traduction de code objets (java, C++, C#, python).

Aix*Marseille université

La recherche reproductible, pourquoi ?

Mouvement qui prend de plus en plus d'empleur depuis 10 ans.

- Le calcul scientifique devient central dans beaucoup de disciplines.
- Des scandales ont éclatés (Avril 2013 en économie).
- Publications aux résultats incomplets ou faux.
- Difficulté à partager les codes et surtout les données. Culture de la publication papier.
- Reproductibilité très difficile quand on touche au big data ou au calcul haute performance.

La recherche reproductible, enjeux.

- Améliorer la fiabilité des résultats de recherche.
- Améliorer la traçabilité et la pérénnité des travaux.
- Favoriser la coopération et la réutilisabilité (échange de code / de données)

La recherche reproductible, comment ?

Dans les laboratoires.

- Garder la trace du cheminement computationnel (Versionnage).
- Généraliser et modulariser les traitements sur les données.
- Archiver les dépendences et les données initiales.
- Automatiser les tests.
- Faire du code robuste.
- Contrôler l'environnement d'excution (packager, abuser des CI).
- Documenter.
- Partager ses sources, scripts, résultats (forges, test dashboard).

Développement de qualité = meilleure reproductibilité.

La recherche reproductible, comment ?

Dans la communauté scientifique.

- Utiliser des normes de citations : DOI pour les produits, ORCID pour les auteurs.
- Anticiper la reproductibilité (les initiatives Science Exchange, HackYourPhD).
- Publier dans des revues qui prennent en compte la reproductibilité (The journal Biostatistics).
- Utiliser des outils qui intègre la reproductibilité ou la facilite (VisTrails, iPython).
- Publier son article avec les codes et les données (Exec&Share)

Outline

Demande du Labex.

- Proposer aux chercheurs une solution pour publier des articles de manière reproductible.
- Analyse des solutions existantes : Exec&Share.
- Proposition de stage L3 pour développer une maquette.

Analyse des Besoins.

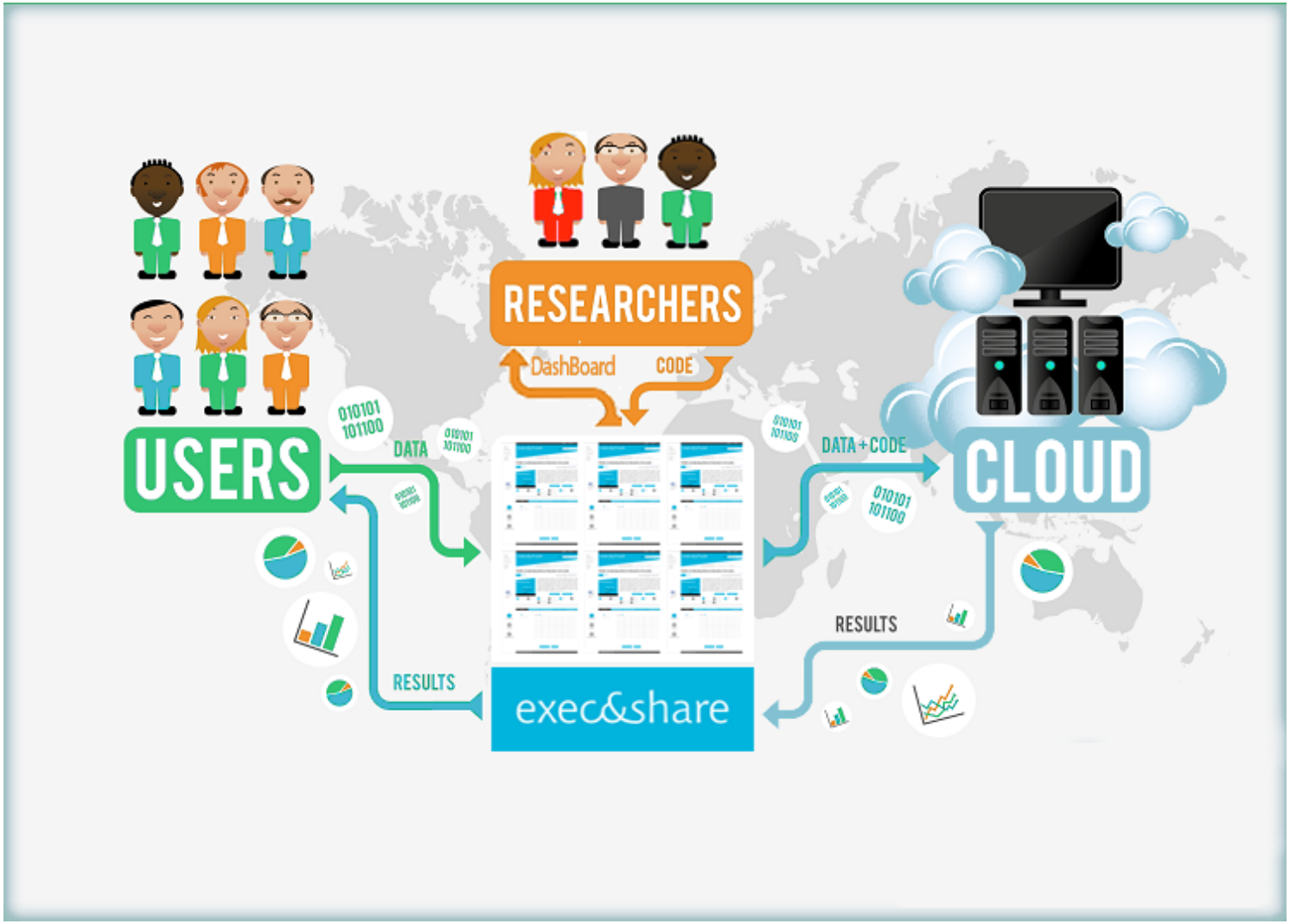
- Deux types d'utilsateurs de la plateforme : déposants et consultants.
- Constitution d'un commité d'utilisateurs avec double rôle.
- Organistation d'une séance de sensibilisation à la RR (Présentation d'Exec&Share puis discussion ouverte avec Yvan Stroppa).
- Séance de travail d'une demi-journée pour définir les besoins.

Analyse des Besoins.

Id	Description
Besoins Consultants	
cons#1	Pouvoir laisser un commentaire en ligne
cons#2	Pouvoir poser des questions au déposant
cons#3	Pouvoir télécharger le code , l'article et les données
cons#4	Recherche par mots clés, auteur, journal/conf
cons#5	Pouvoir accéder a tous les codes associés à un article et vice versa
cons#6	Récupérer les citations en format standard
Besoins Déposants	
Dep#1	Lien avec des plateformes existantes (Hal, Arxiv) où sont les articles. Choix entre upload le pdf ou lien existant
Dep#2	Extracteur automatique de métadonnées si lien avec Hal et Arxiv
Dep#3	Guider l'écriture du readme, avoir un « template » (standardiser le readme) (Pour obtenir la figure xx il faut faire ça...) (1 readme par code)
Dep#4	nArticle pour n codes
Dep#5	Un fichier exemple par article + guidage
Dep#6	Système de version pour l'article et pour le code
Dep#7	Notification automatique de modification de version dans Hal . . .
Dep#8	Statistiques (visualisations, téléchargements). . .
. . .	. . .

Outline

Présentation de la plateforme.



Présentation de la plateforme.

- Plateforme de dépôt de codes associés à un article.
- Le déposant doit construire un site compagnion où il précise les données d'entrées (format, nature, types) et donne son code.
- Le code est ensuite déployé sur une grille de calcul par un ingénieur dédié.
- Le consultant peut ensuite rejouer en ligne les exemples fournis ou demander une simulation sur ses propres données.

Exemple

Public debates driven by incomplete scientific data: The cases of evolution theory, global warming and H1N1 pandemic influenza. Serge Galam

Retours d'utilisation (Déposants).

- Utilisation complexe.
- Format des données d'entrée est très restreint (adapté à l'économétrie).
- Trop de champs à renseigner. Consommateur de temps.
- Manque de retour lors de la saisie.

En conclusion nos trois chercheurs veulent des choses simples. . . .

Outline

Choix techniques.

- Sobriété et simplicité d'utilisation.
- Développement en PHP.
- Base de donnée PostgreSQL avec ORM pour le mappage objet (Propel).
- Moteur de template Twig et Twitter Bootstrap pour le frontend.
- Déploiement sur une VM hébergée au LIF.
- Utilisation du GitLab du LIF pour versionnage et système de tickets.

Version 1.

- Premier "Sprint" de 3 semaines.
- Retours des utilisateurs.
- Mises à jour des besoins et prioritisation des tâches pour la V2.

Besoins Utilisateurs de la Plateforme de Recherche Reproductible

Besoins Déposants

Id	Description	Statut V1	Commentaires	Retours Utilisateurs	Prévisions V2	Priorité V2
Dep#1	Lien avec des plateformes existantes (Hal, Arxiv) ou sont les articles. Ou un choix entre upload le pdf ou lien existant	Ok	Pdf & url			
Dep#2	Extracteur automatique de métadonnées si lien avec Hal & Co	Nok		Non utile		1
Dep#3	Guider l'écriture du readme, avoir un « template » (standardiser le readme) (Pour obtenir la figure xx il faut faire ça...) (1 readme par code)	Nok	Upload d'un fichier texte seul	Oui. Il faut avoir le choix entre télécharger un fichier texte ou suivre un guide.	Thibaut	5
Dep#4	nArticle pour n codes	Nok	Codes attachés par conception à l'article	Il serait bien de pouvoir au moins renseigner l'url d'un code déjà téléchargé pour un autre article.		3
Dep#5	Un fichier exemple par article + guidage	Nok	Pas de guidage	Plutôt qu'un guidage proposer une aide avec un exemple.	Thibaut	5
Dep#6	Système de version pour l'article et pour le code	Nok		Il faut également pouvoir changer ou supprimer une version déjà téléchargée.	Thibaut	8
Dep#7	Notification automatique de modification de version dans Hal etc (voir aigle et ce qu'a fait Gerard Henry)	Nok		Abandonné		
Dep#8	Statistiques (visualisations, téléchargements)	Nok		Pas utile. Si besoin l'admin pourra toujours les fournir à la demande.		
Dep#9	Pouvoir paramétrer la personne contact (qui va recevoir les mails questions)	Ok	Le mail du compte	Pas utile. Le mail de l'utilisateur suffit.		
Dep#10	Une autorisation de publication explicite (cf Hal)	Ok				
Dep#11	Indiquer qu'elles sont les règles de protection intellectuel + aide a définir quel est le statut du code (licence, droit) avec cartouche par défaut	Nok		Il faudrait que la plateforme génère un exemple de cartouche pour les codes et données que le déposant pourra mettre dans ses fichiers.		5
Dep#12	Avoir un lien qui renvoi vers la liste des articles et codes	Ok				
Dep#13	Un identifiant formel (qui veut dire quelque chose) par article et code	Nok		Pas utile.	Thibaut	
Dep#14	Utiliser l'identifiant AMU	Nok		Serait bien au moins en V3.		2
Dep#15	Filtre anti-spam sur les commentaires	Nok				8
Dep#16	Comme Dep#11 pour les données.	Nok				5
Dep#17	Ajouter des champs facultatifs pour renseigner le type de licence codes et données.	Nok				5
Dep#18	Indiquer clairement les limites de tailles pour tous les téléchargements et inviter à contacter l'administrateur en cas de dépassement.					5

Besoins Consultants

Id	Description	Statut V1	Commentaires	Retours Utilisateurs	Prévisions V2	Priorité V2
Cons#1	Pouvoir laisser un commentaire en ligne	Ok		Ajouter un système anti-spam (Catcha)	Thibaut	
Cons#2	Pouvoir poser des questions au déposant (MP)	Nok		Par mail uniquement.	Thibaut	8
Cons#3	Pouvoir télécharger le code , l'article et les données	Ok				
Cons#4	Recherche par mots clés, auteur, journal/conf	Nok		Indispensable	Thibaut	13
Cons#5	Pouvoir accéder a tous les codes associés à un article et vice versa	Ok	Le vice versa dépend de Dep#4			
Cons#6	Récupérer les citations en format standard	Nok		Il faut au moins prévoir un champs pour que le déposant puisse mettre lui-même une citation de son article.		5
Cons#7	Pouvoir télécharger un article et tous les codes, exemples, données associés dans une archive tar.gz	Nok				5

Version 2.

- Le sprint a été racourci pour cause de rapport de stage.
- Encore des fonctionnalités non couvertes + bugs.
- Une V3 est nécessaire avant mise en production.

Un autre stagiaire. . .

Le site.

Besoins Utilisateurs de la Plateforme de Recherche Reproductible

Besoins Déposants

Id	Description	Statut V2	Priorité V2
Dep#1	Lien avec des plateformes existantes (Hal, Arxiv) ou sont les articles. Ou un choix entre upload le pdf ou lien existant	Ok	
Dep#2	Extracteur automatique de métadonnées si lien avec Hal & Co	Cancelled	1
Dep#3	Guider l'écriture du readme, avoir un « template » (standardiser le readme) (Pour obtenir la figure xx il faut faire ça...) (1 readme par code)	Nok	5
Dep#4	nArticle pour n codes	Nok	3
Dep#5	Un fichier exemple par article + guidage	Nok	5
Dep#6	Système de version pour l'article et pour le code	Ok	8
Dep#7	Notification automatique de modification de version dans Hal etc (voir aigle et ce qu'a fait Gerard Henry)	Nok	
Dep#8	Statistiques (visualisations, téléchargements)	Cancelled	
Dep#9	Pouvoir paramétrer la personne contact (qui va recevoir les mails questions)	Ok	
Dep#10	Une autorisation de publication explicite (cf Hal)	Ok	
Dep#11	Indiquer qu'elles sont les règles de protection intellectuel + aide a définir quel est le statut du code (licence, droit) avec cartouche par défaut	Nok	5
Dep#12	Avoir un lien qui renvoi vers la liste des articles et codes	Ok mais bugs	
Dep#13	Un identifiant formel (qui veut dire quelque chose) par article et code	Cancelled	
Dep#14	Utiliser l'identifiant AMU	Nok	2
Dep#15	Filtre anti-spam sur les commentaires	Ok	8
Dep#16	Comme Dep#11 pour les données.	Nok	5
Dep#17	Ajouter des champs facultatifs pour renseigner le type de licence codes et données.	Nok	5
Dep#18	Indiquer clairement les limites de tailles pour tous les téléchargements et inviter à contacter l'administrateur en cas de dépassement.	Nok	5

Besoins Consultants

Id	Description	Statut V2	Priorité V2
Cons#1	Pouvoir laisser un commentaire en ligne	Ok	
Cons#2	Pouvoir poser des questions au déposant (MP)	Ok	8
Cons#3	Pouvoir télécharger le code , l'article et les données		
Cons#4	Recherche par mots clés, auteur, journal/conf	Ok (Revoir la gestion de la case)	13
Cons#5	Pouvoir accéder a tous les codes associés à un article et vice versa	Ok mais bug	
Cons#6	Récupérer les citations en format standard	Nok	5
Cons#7	Pouvoir télécharger un article et tous les codes, exemples, données associés dans une archive tar.gz	Ok mais bug	5

Outline

Conclusion

- Compromis entre outil complet/simplicité d'utilisation pour le chercheur.
- Contraintes de dépôts : Un exemple, un code, un readme, un fichier de données. Procédure suffisante ?
- Gestion des droits d'auteur.
- Chaque chercheur a son propre besoin.

Questions. . .

- Y a-t'il des demandes dans vos unités ?
- Quels outils sont utilisés ?