



Déploiement logiciels

Journée Calcul
14 juin 2019
INRA

Déploiement logiciels

plan présentation

- Contexte plateforme Bioinfo
- Mise à disposition de logiciels
- Problématique du déploiement
- Différentes solutions techniques
- Support, formation utilisateurs
- Licences



ÉQUIPE DE 10 PERMANENTS (6.5 ETP)



COEUR DE METIER LA SEQUENCE



20 ANS DE VIE



ACTIVITÉS

- Prestation
- Accompagnement de projet
- Développement
- Formation

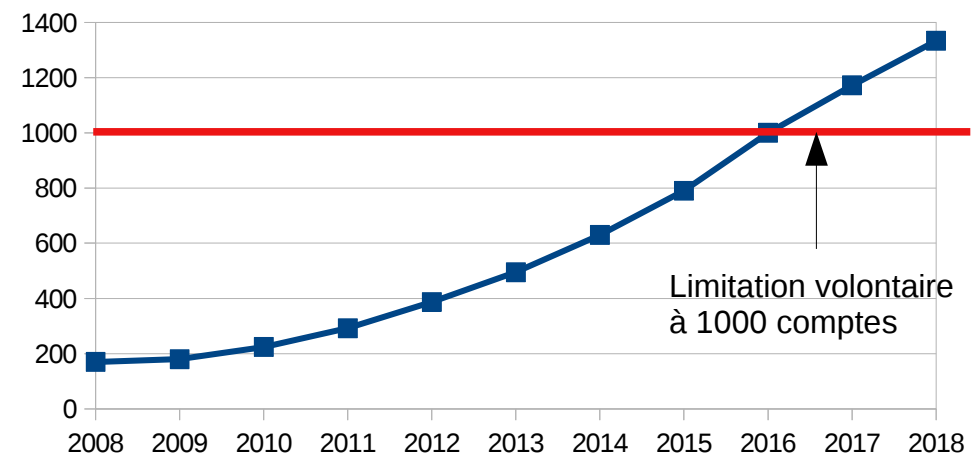
**LOGICIELS (>600)
BANQUES (>200)**

**SUPPORT :
2000 tickets annuel**

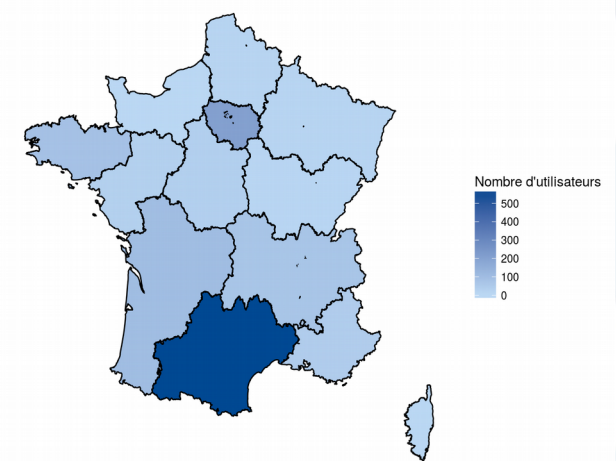
ÉQUIPEMENTS

- Cluster de calcul SLURM: 3.000 cœurs (6000 threads), 34 To RAM
- Stockage performant parallèle : 1 Po
- Stockage capacitif NAS : 2 Po
- Hébergement de machines virtuelles

Nb utilisateurs GenoToul Bioinfo



Nombre d'utilisateurs par région



Déploiement logiciels

retour d'expériences

- Communauté de **Bioinformatique** = **abondance** de logiciels
- Compilation, installation et mises à jour de logiciels à la **demande des utilisateurs** dans un répertoire partagé
- Ensuite on crée des **environnements** sous forme de **module** pour faciliter le chargement d'une version ou d'une autre
- Mise en place de **workflows** en ligne de commande (makeflow) ou graphique (galaxy)
- **Logiciels métier** : annotation, assemblage, nettoyage, alignement de séquences, statistique...
- **Outils de compilation** : gcc, icc, ghc, ldc, qt, dmd, open64, stack, swig, ocaml, rust, nim, llvm, cmake...
- **Utilitaires** : java, perl, ruby, python, R, go, cuda, blas, openmpi, singularity, conda, git...

Déploiement logiciels problématique

- **Plusieurs versions** pour un même logiciel
- Souvent avec des **dépendances** associées
- Souvent **mal documentés**, non standards, mal testés
- Problème de **portabilité** et de **reproductibilité** (en raison de l'évolution des systèmes)
- De **multiples paramètres** : l'architecture processeur, la version du compilateur, les flags utilisés...
- **Difficultés** à faire fonctionner les **nouveaux logiciels** sur une distribution stable
- Les **utilisateurs** sont **dépendants** de ce que les administrateurs mettent à disposition

Déploiement logiciels

solutions techniques

Gestionnaires de paquets (yum, apt, conda...)

Décrit les librairies nécessaires mais il est difficile de fixer les versions des dépendances logicielles

=> impact lors de la mise à jour des systèmes

Conteneur (docker, singularity...)

Virtualisation légère : accueille à la fois l'application et l'ensemble des logiciels dont il dépend (y compris la partie du système d'exploitation)

Boîte noire => aucune modification de paramètre possible

Nouveauté : guix

Solution de package logiciel reproductible qui cumule les avantages

=> nécessite un temps d'apprentissage non négligeable

Déploiement logiciels

licences

- Quasiment tous les logiciels mis à disposition sont issus de la **communauté Education Recherche** et sont pour la plupart sous licence libre
- Respect des 4 principes des **licences libres** (GPL) : utilisation, accès au code source, modification et obligation de publication si distribution
- Attention toutefois aux utilisateurs d'entreprises privées qui sont tenus de vérifier les licences des **logiciels académiques**
- Seulement quelques utilisateurs ont accès à des **licences** logicielles **payantes** mais ceux sont eux qui en font la gestion
- licence **fixe** : permet l'utilisation sur un serveur particulier (adresse MAC)
- licence **flottante** : fonctionne avec un **serveur de licence**; celui ci décompte le nombre de licences utilisé à l'instant "T"

Déploiement logiciels

support, formation

- Selon les **politiques** (centralisée, déléguée) et les **techniques** de déploiement choisies (package, conteneur)
- Il y a toujours besoin de **ressources humaines** pour mettre à disposition les applications dans un espace partagé
- Compilation, aide à la **soumission de scripts**, optimisation des paramètres, débogage
- Mise en place de **sessions de formations**, réponses aux tickets support, aide en ligne (FAQ)