

Développement d'un intergiciel de Grille de calcul - CiGri

Projet S10 - Groupe 10

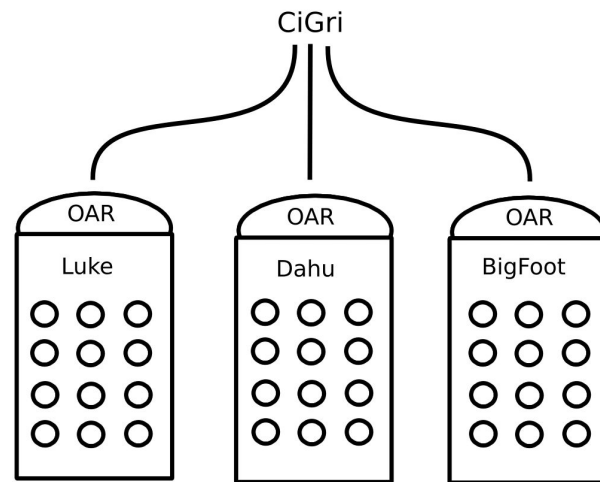


Les équipiers et leurs rôles

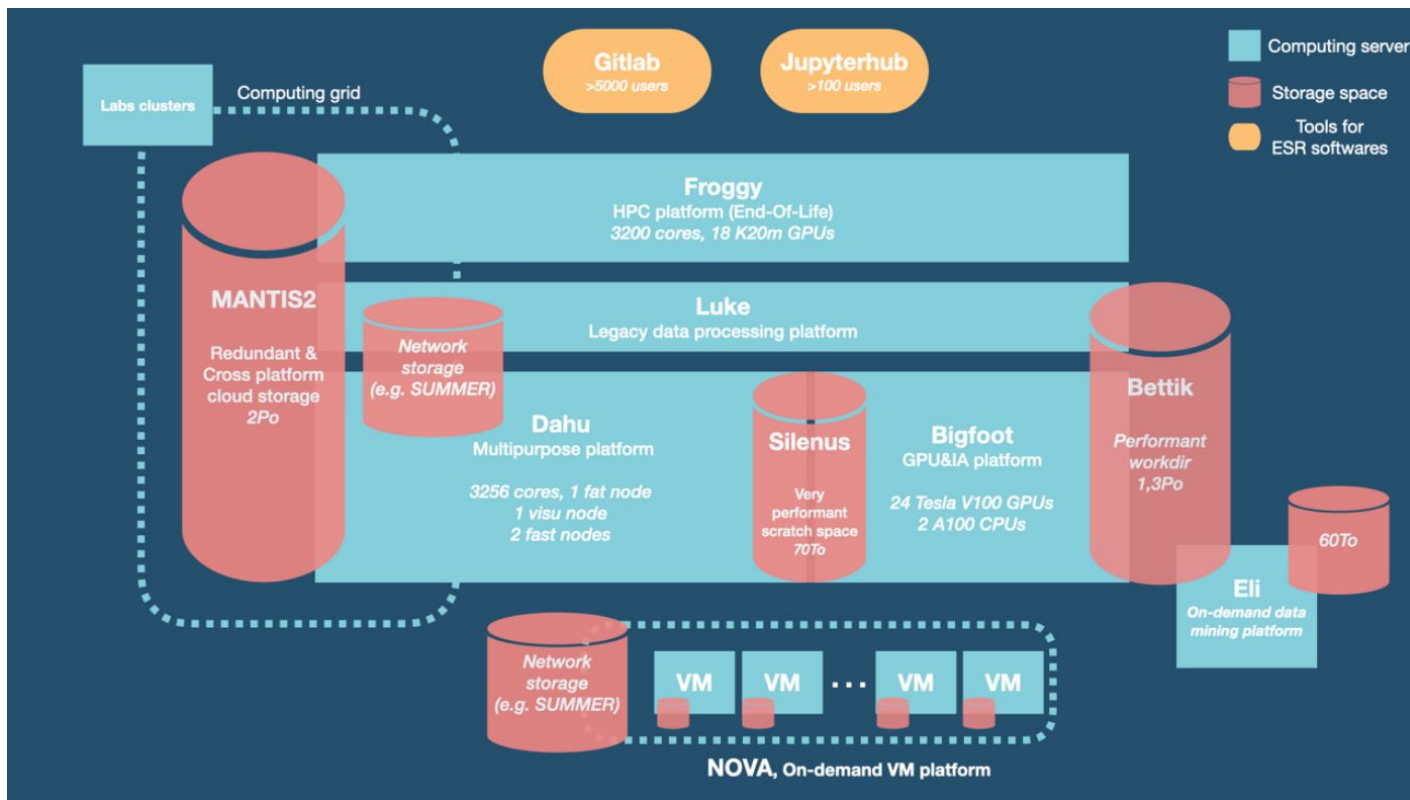
| Membres | Rôles |
|---------------------------|------------------------------|
| Killian GRICOURT | Chef de projet / Développeur |
| Antoine BONFILS | Scrum Master / Développeur |
| Diego CORREIA DE OLIVEIRA | Tech lead / Développeur |
| Tanguy DELAS | Communication / Développeur |

Le contexte, le sujet et l'objectif du projet

- Les scientifiques de l'agglomération grenobloise utilisent les clusters de calcul de l'organisation Gricad.
- L'une de leur interfaces pour utiliser ce service est le logiciel d'ordonnancement CiGri, développé par Bruno Bzeznik depuis 2007.
- Remise au goût du jour en 2024 de CiGri en le traduisant en python et en utilisant des technologies pérennes.

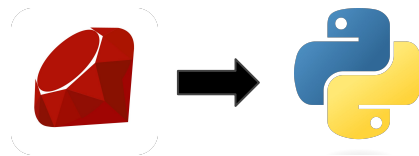


Le contexte, le sujet et l'objectif du projet

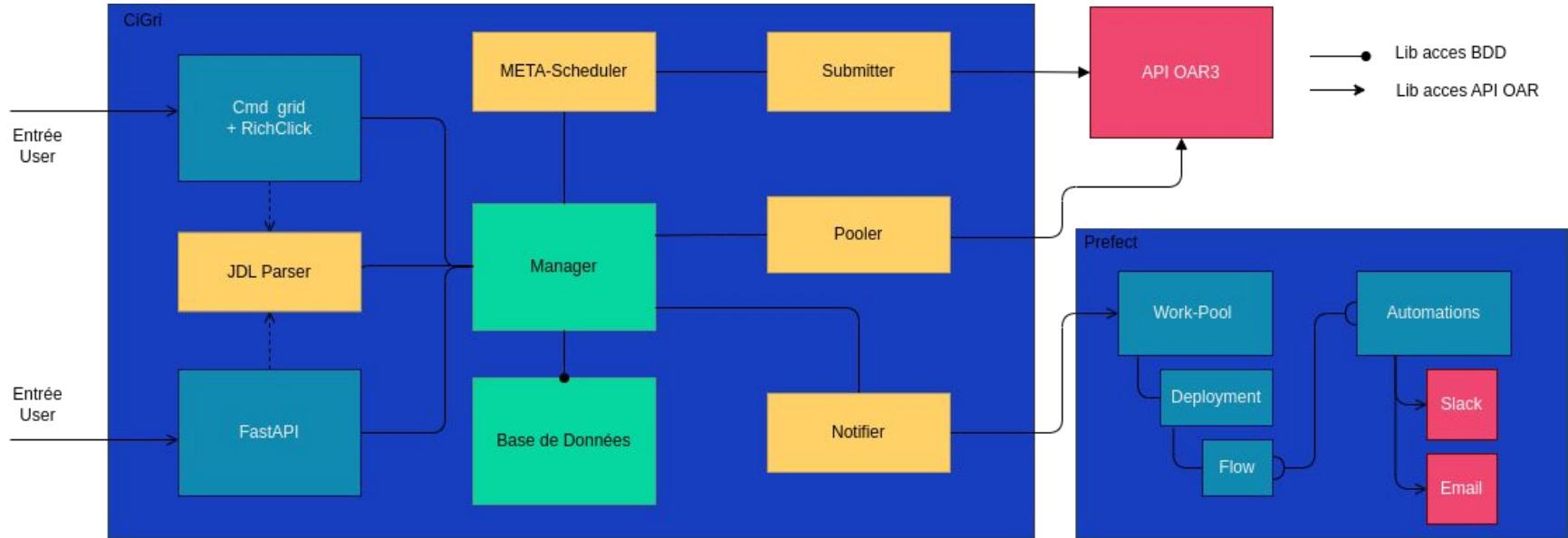


Les technologies utilisées

- Poetry
- Python
- FastAPI
- SQL alchemy
- Prefect



L'architecture du systèmes à réaliser



Le plan de travail

Liste des fonctionnalités

Ordre d'importance 1 = max

Difficulté 1 = min

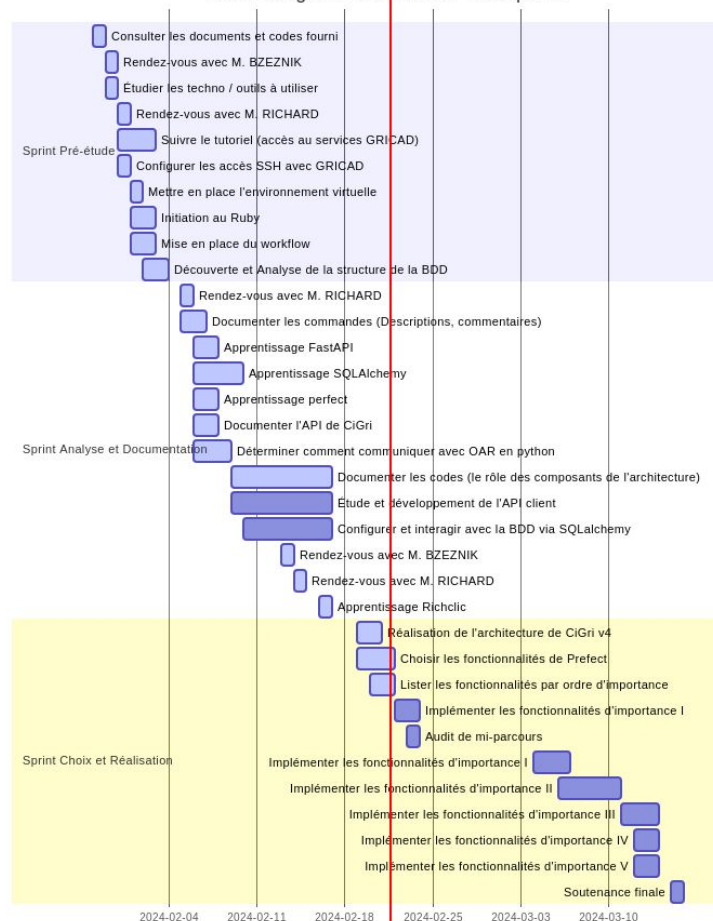
Commande CLI

| commande | Fonctionnalité | Importance | difficulté |
|-------------|--|------------|------------|
| gridcluster | Afficher les clusters disponibles sur Gricad | 2 | 1 |
| | Afficher des informations sur les clusters disponibles sur Gricad. | 4 | 2 |
| | Afficher des informations avancés sur les clusters disponibles sur Gricad. | 3 | 2 |

Back end

| Fonctionnalité | Importance | difficulté |
|--|------------|------------|
| Stocker les campagnes | 1 | 1 |
| Gestion de campagnes multiples | 1 | 2 |
| Création d'une liste de jobs à soumettre pour une campagne | 1 | 3 |
| Utilisation d'un scheduler sans famine | 1 | ? |

Notre Diagramme de Gantt - Groupe 10



Les difficultés

Prefect est un logiciel complet et indépendant, l'intégrer dans une architecture requiert une forte adaptation de l'architecture. Son intégration dans CiGri peut s'avérer incompatible sur certains points.

De plus, le manque de documentation n'a pas facilité la compréhension du code existant.

Ce que nous avons fait !

- Environnement de tests unitaires
 - Encapsulation des routes d'API OAR3 (+ client HTTP)
 - Gestion de l'authentification
-
- BEAUCOUP de documentation du code existant
 - BEAUCOUP de recherche et de réflexion autour des technologies à utiliser

Merci de nous
avoir écoutés