



HAL
open science

GITLAB CI/CD ET ENVIRONNEMENT R - CONSTRUIRE SON PROPRE PIPELINE

Jean-François Rey, Loïc Houde

► **To cite this version:**

Jean-François Rey, Loïc Houde. GITLAB CI/CD ET ENVIRONNEMENT R - CONSTRUIRE SON PROPRE PIPELINE. JDEV, Jul 2020, RENNES, France. hal-02899373

HAL Id: hal-02899373

<https://hal.science/hal-02899373>

Submitted on 15 Jul 2020

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

PRÉSENTATION

De nos jours, les outils de gestion de projets informatiques sont largement répandus, beaucoup d'entre eux fournissent une partie spécifique de la gestion du cycle de vie des projets ou sont des solutions tierces. Cela est particulièrement vrai pour l'**environnement R**.

Nous présentons ici notre solution utilisant **GitLab CE**, un outil open core et libre, qui nous permet de développer du code **R** collaborativement et d'automatiser la validation et la construction de **package R** ainsi que le déploiement d'applications **R Shiny**.

Cette solution locale dans notre laboratoire nous permet d'accélérer le développement et le partage de packages R sur différents OS et la soumission au CRAN en réduisant les erreurs possibles. Mais elle permet aussi la mise à disposition d'applications R Shiny immédiatement accessibles.

LES BESOINS

=> **Un outils de gestion de développement de package R et d'application R Shiny**

- Multi utilisateurs et collaboratif
- Dépôts privé et public + versionné
- Outils auto-hébergés
- Automatiser au maximum (gain de temps)

- Pipeline CI/CD personnalisé
- Multiple OS (Greenland avec R)
- Vérifier la régression de code
- Revu et qualité du code
- Facile à déployer
- Stockage des artefacts
- Reproductibilité

* Le plus important : moderne et simple à utiliser

GITLAB

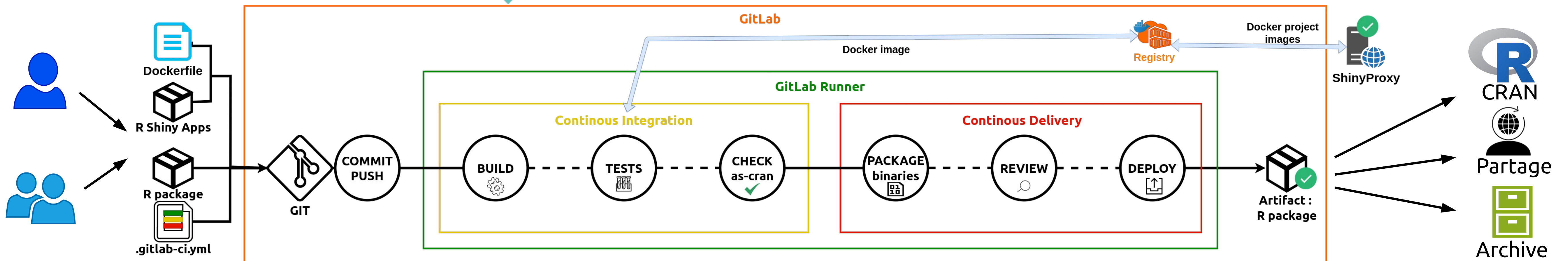


"GitLab is a single application for the entire software development lifecycle. From project planning and source code management to CI/CD, monitoring, and security." (<https://about.gitlab.com>)

- Community Edition (MIT licence)
- GitLab-Runner
- Docker registry

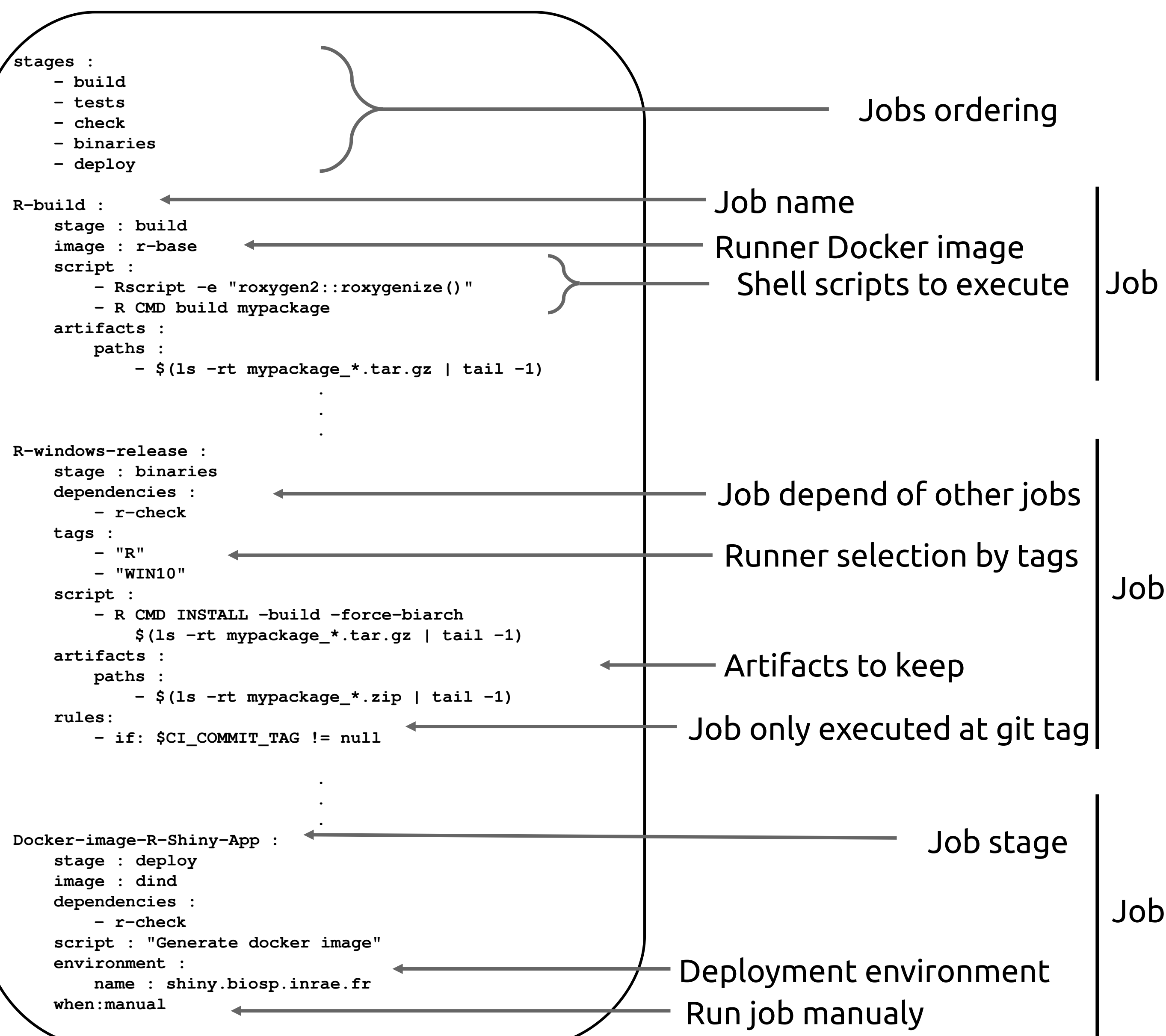
<https://gitlab.paca.inrae.fr>

ARCHITECTURE DU PIPELINE

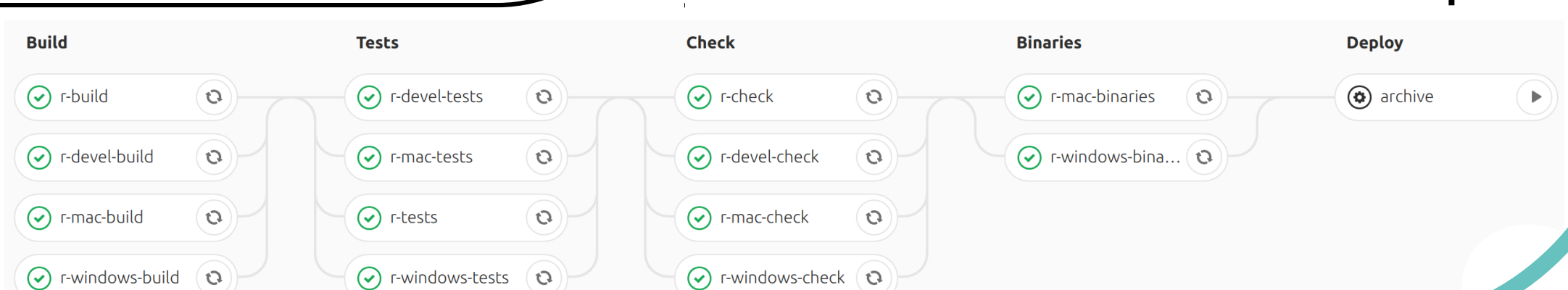


.gitlab-ci.yml

Orchestration CI/CD pipeline

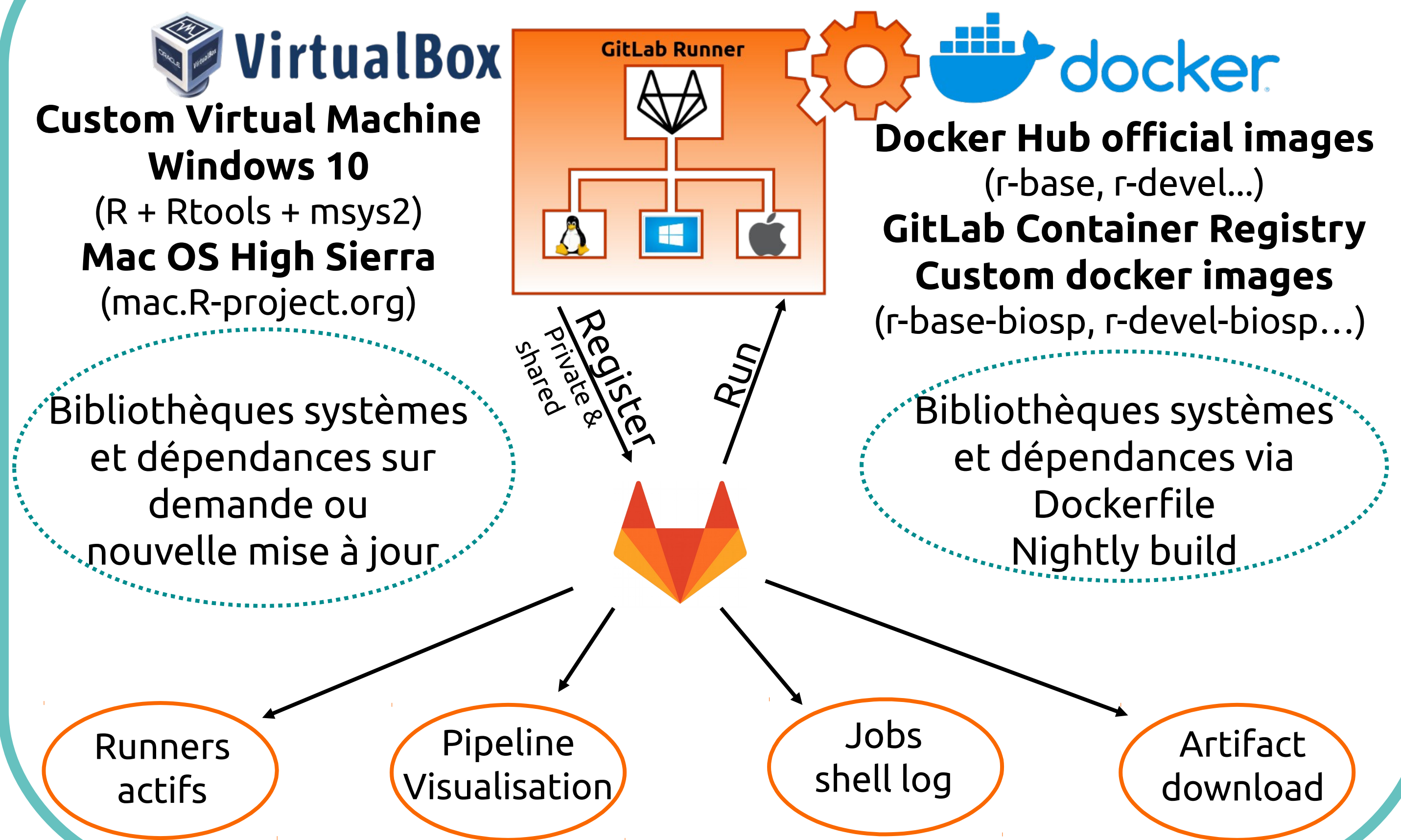


Visualisation du pipeline dans GitLab



GITLAB-RUNNER

Exécute les jobs et renvoie les résultats à GitLab



PERSPECTIVES

- Automatiser les Machines Virtuelles :
- Création
- Configuration
- Versionné/ snapshot
- Activer plus de fonctionnalités GitLab (web terminal, pages...)
- Utiliser Kubernetes (orchestration de container)
- CI/CD stratégies

Présentation vidéo : T4 REX



Centre
Provence-Alpes-Côte d'Azur -
Avignon

Jean-François Rey¹ and Loïc Houde¹

¹BioStatistique et Processus Spatiaux (BioSP), INRAE Avignon, France

biosp.org | jean-francois.rey@inrae.fr | loic.houde@inrae.fr | [@jfrey_official](https://twitter.com/jfrey_official)

Biostatistique



& Processus Spatiaux