Software Provisioning Infrastructure (UMD/CMD)

Mario David (LIP), João Pina (LIP), Carlos Manuel (LIP), Jorge Gomes (LIP), Samuel Bernardo (LIP), Pablo Orviz (IFCA)

EGI Repositories

The EGI Software Repository provides access for the middleware distributions together with some Community Repositories

Two main categories:

- The Unified Middleware Distribution (**UMD**): redistribution of traditional middleware
- **IGTF** Distribution of Authority Trust Anchors: packages with the trust anchor information

OS Support

- AlmaLinux9 (EL9) : **UMD-5**

Provider:

- IBERGRID since 2021

EGI Repositories

- Complete redesign of the service:
  - Current infrastructure based on Nexus Repository OSS
  - Release workflow implemented in GitHub and Jenkins pipelines.
- Software Quality assurance - Full-automated approach
  - **Faster release cycle** with higher automation and less manual intervention

EGI Repositories

New Functionalities:

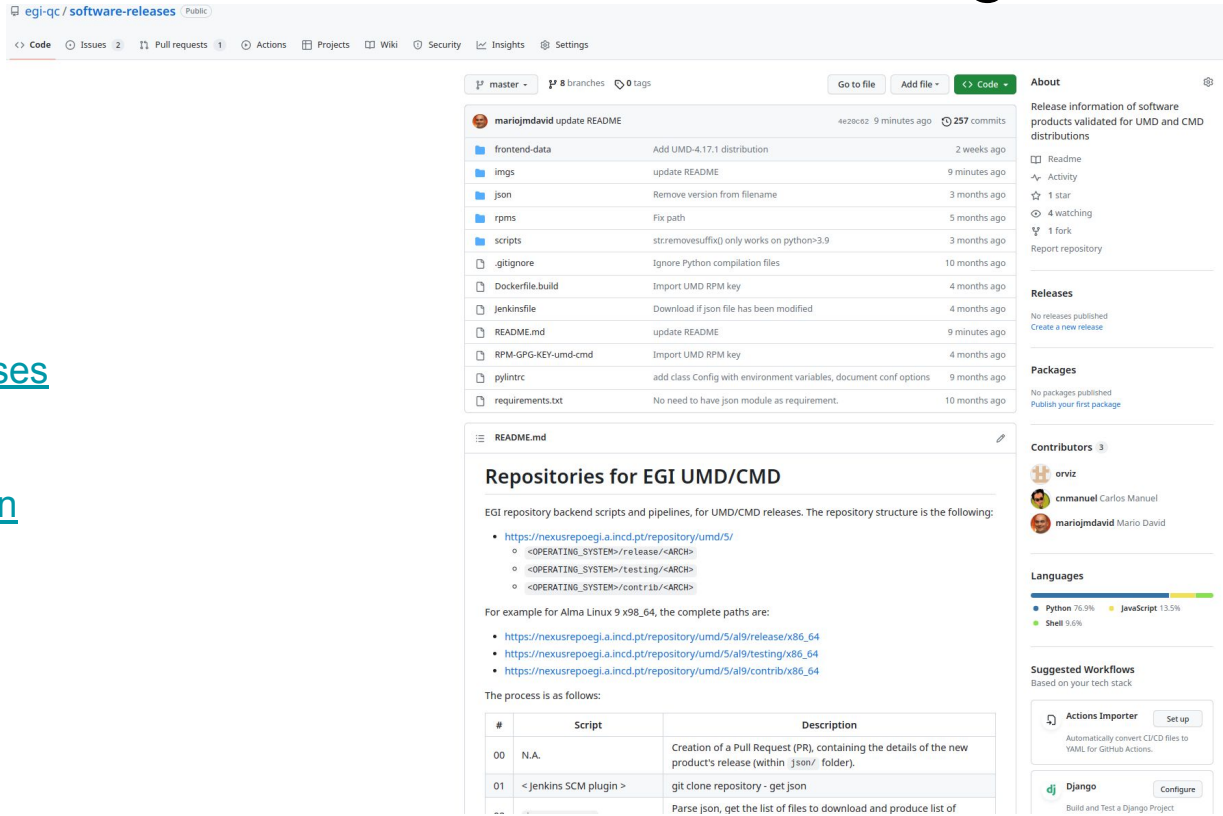
- IPV6 compliant
- New backend behind a proxy (HAProxy)
  - https://repository.egi.eu/sw/production/ (prod)
  - https://repository.egi.eu/software/umd/ (dev)
- High Availability (server side)
- Full automatic process until testing
- Testing -> production (manual git commit)

Tracking of the workflow all based on git

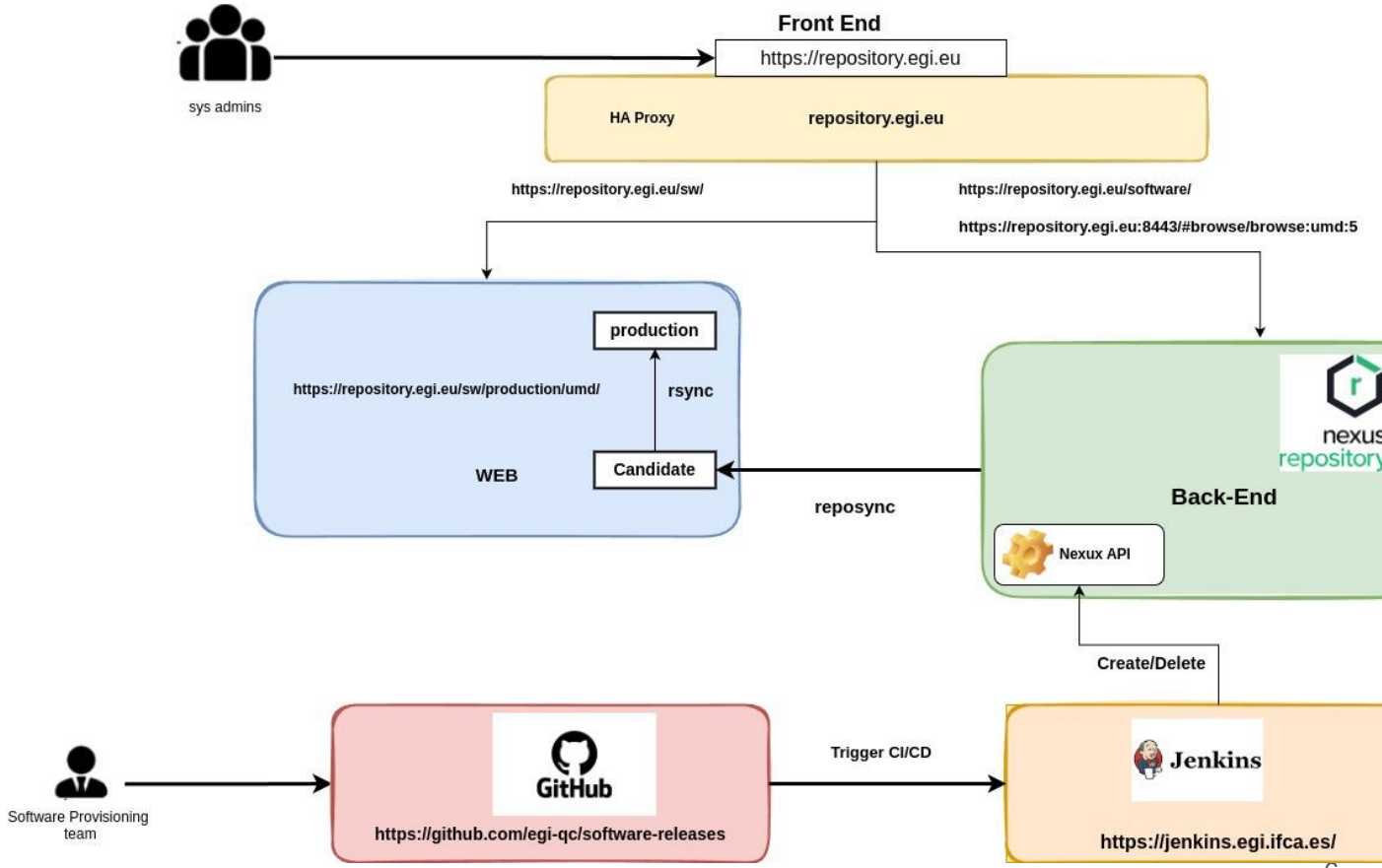
Software provisioning process:
https://github.com/egi-qc

Software Releases:
https://github.com/egi-qc/software-releases

Ansible playbooks for install software:
https://github.com/egi-qc/umd-verification



Architecture



Nexus

Yum repositories (new structure)

UMD-5 only

1. `release`: Main repository with all the packages (**= to production**).
2. `testing` : testing repository with rpm not fully validated in production environment. RPM belonging to release declared as production ready will be moved from testing to release.
3. `contrib`: repository for specific communities or special services. (**NEW**)

Dropped repositories: Staged Rollout and **Release Candidate**

- UMD-5 **Open for beta testers (September 2024)**
  - **https://repository.egi.eu/umd/distribution.html?id=5.0.0#5.0.0**
  - **Install instructions:**
  - **https://repository.egi.eu/umd/installation-notes.html?id=5**

Nexus

Yum repositories (new structure)

Production

```
[UMD-5]
name=UMD 5 main repository (Alma Linux 9)
baseurl=https://repository.egi.eu/sw/production/umd/5/al9/release/$basearch
enabled=1
priority=1
gpgcheck=1
gpgkey=https://repository.egi.eu/repository/umd/5/UMD-5-RPM-PGP-KEY
```

EPEL 9 required

UMD higher priority than EPEL

Priorities already part of DNF

Need crb almalinux9 enabled

No more base / updates >> release

Nexus

Yum repositories (new structure)

Testing

```
[UMD-5-testing]
name=UMD 5 nexus testing (Alma Linux 9)
baseurl=https://repository.egi.eu/sw/production/umd/5/al9/testing/$basearch
enabled=0
priority=1
gpgcheck=1
gpgkey=https://repository.egi.eu/repository/umd/5/UMD-5-RPM-PGP-KEY
```

Packages under testing but already validated

Nexus

Yum repositories (new structure)

External Contribution (new)

```
[UMD-5-contrib]
name=UMD 5 contrib (Alma Linux 9)
baseurl=https://repository.egi.eu/sw/production/umd/5/al9/contrib/$basearch
enabled=0
priority=1
gpgcheck=1
gpgkey=https://repository.egi.eu/repository/umd/5/UMD-5-RPM-PGP-KEY
```

Packages under testing but already validated: Caso

Nexus

Umd-5 first release (products list)

Arc 6.20.1

WN 5.1.0

UI 7.0.0

Dcache 9.2.25

Gfal2 2.23.0

Frontier-squid 5.9.2

Voms 2.1.0, voms-api 3.3.3, voms-client-java 3.3.4, voms-client-cpp 2.1.0

Apel 2.1.0

xroot 5.7.1

htcondor-ce 23.0

cvmfs 2.11.5

config-egi 2.6.1

egi-cvmfs 6.7.28

Davix 0.8.7

Ui 7.0.0

BDII 6.0.3

Site-bdii 1.0.8

UMD evolution

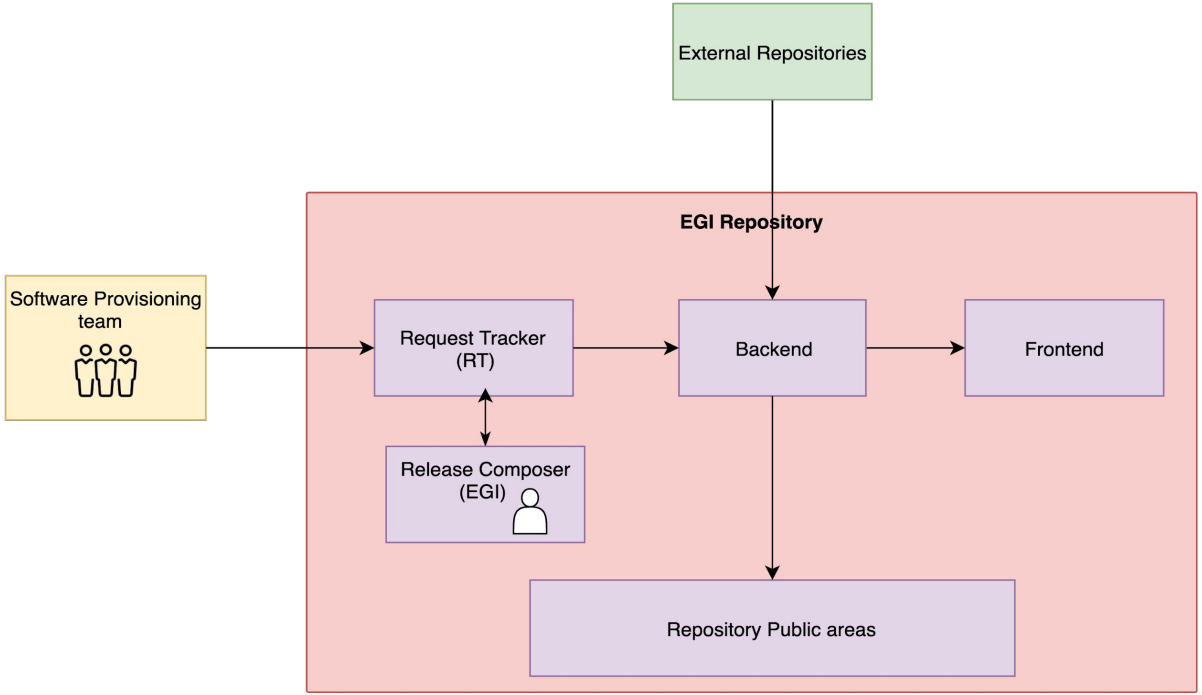
Future:

- Integration with SQAaaS from EOSC-Synergy
  - **Allow full pipeline tests**
- Allow it to be used by **external users**:
  - EGI / EOSC marketplace

End

Questions?

UMD-4 architecture (old)

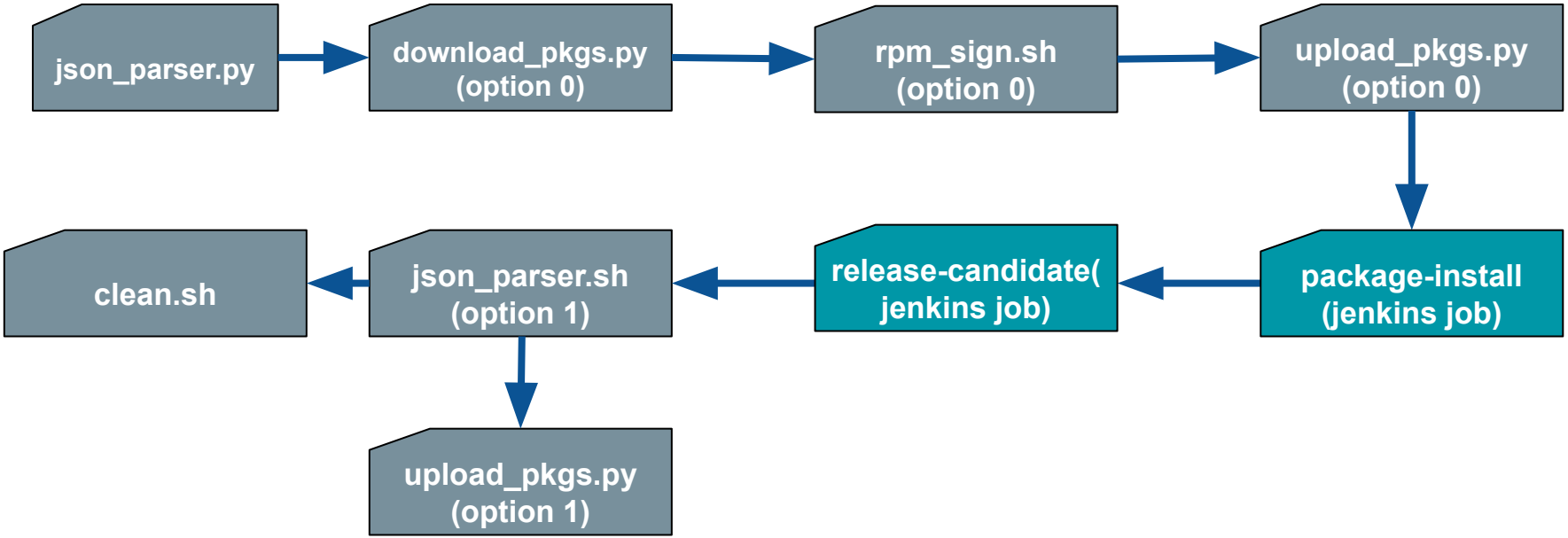


Old setup:

- No new functionalities for many years, since it's hard to implement them.
- Backend based on perl scripts with tight relation with RT.
- To many manual steps (RT fully manual) leading to a large delay in the releases of packages
- Frontend outdated which could lead to security issues (**already replaced by a simplified version**)

UMD/CMD new Workflow

https://github.com/egi-qc/software-releases



Approved the Release (Testing/Production repository)

Pipeline

https://github.com/egi-qc/software-releases

The pipeline is as follows:

1. `json_parser.py`: parse json, get the list of files to download and produce list of filenames (packages).
2. `download_pkgs.py` (option 0): download the packages to a temporary directory.
3. `rpm_sign.sh` (option 0): rpm sign each package.
4. `rpm_sign.sh` (option 0): verify signature of each package.
    a. a. verification of the packages
5. `upload_pkgs.py`: upload each package to nexusrepo.
6. `Package-install`: Validate package installations from testing repository and perform functional tests.
7. `release-candidate`: Install all packages in release repo together with the new packages from testing
8. `json_parser.sh`: Produce new json file as asset of the new release
9. `clean.sh`: clean temporary directories.