

FAIR Distribution & Deployment

Maarten van Gompel & Mario Mioldijk, KNAW HuC

Introduction (1)

- ▶ We are building a common research infrastructure
- ▶ Tools need to be **distributed** properly by tool developers
- ▶ Tools available as a service need to be **deployed** in the infrastructure by operators.
- ▶ De-coupling between **application provider** (distribution) and **infrastructure provider** (deployment)

This epic/shared service provides the embedding for this. Logical successor of the DevOps IG (RIP)

Introduction (2)

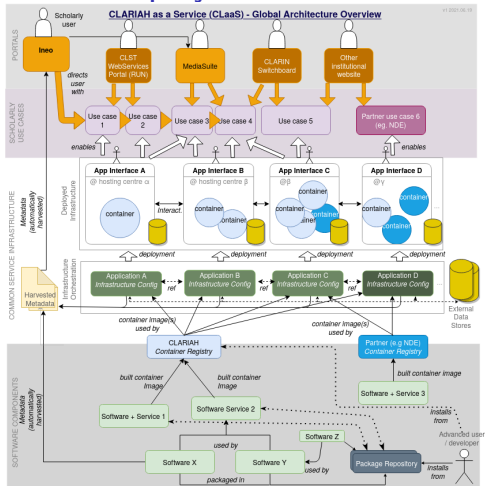
- ▶ **Distribution of tools**
 - ▶ How can CLARIAH developers publish their tools?
 - ▶ Facilitate installation for end-users and infrastructure providers
- ▶ **Deployment of tools**
 - ▶ Install a tool locally
 - ▶ Deploy a tool as a service in an infrastructure


Scope: aspects of distribution & deployment (1)

Broad scope, affecting all participants in CLARIAH:

- ▶ **Version control**
- ▶ **Packaging**
- ▶ **Containerisation**
- ▶ **Container Orchestration**
- ▶ **Infrastructure as Code** (decoupling)

Recap: From distribution to deployment



-  **A software component:** the source code of which is available as open source and hosted in a public version-controlled source-code repository (e.g. GitHub), where applicable the software is also packaged and stored in the following:
-  **A package repository** for the appropriate software ecosystem (e.g. Python Package Index, CRAN, CPAN, Maven Central, crates.io, npm) or target distribution (debian/ubuntu/redhat/alpine/homebrew). The “used by” connections between software components are usually mediated by such a package repository.
-  **An application deployment configuration (Infrastructure as code):** the source code of which is available in a version-controlled source-code repository (e.g. GitHub). This may consist of a Docker Compose and/or Kubernetes configuration. (by CLARIAH respectively a partner project (darker shade))
-  **Deployed docker containers:** or container **registries** storing the container images. (CLARIAH respectively partner project)
-  **Use case:** A scholarly scenario a user has. Multiple services may be required to fulfill the scholarly scenario. Services in turn can be reused in multiple scholarly use cases. Should be interpreted more broadly than the more technical use cases that would be more used to specific applications. (e.g. A.R.C.D.)

Scope: aspects of deployment (2)

Most is covered by existing WP2 tasks.

- ▶ **Security**

- ▶ Authentication and Authorization
- ▶ Tools to Data / Data to Tools
- ▶ Automated vulnerability scanning

- ▶ **Scalability**

- ▶ Load balancing
- ▶ Horizontal scaling

- ▶ **Monitoring**

- ▶ Service availability monitoring (from end-user point of view)
- ▶ Service usage monitoring
- ▶ Infrastructure monitoring

- ▶ **Workflows:**

- ▶ accommodating/pushing existing workflow solutions (DANE, NextFlow) within our infrastructure context

Out of scope

We focus on the shared *technical* dimension here, so out of scope are:

- ▶ Governance
- ▶ Service License Agreements etc. . .
- ▶ Hardware acquisition

Although we will give our technical input to the person and/or group who must cover the above.

User Stories

- ▶ **As a scholar, I** want to apply a processing tool on a (possibly large) data set in the CLARIAH infrastructure, either using computational resources provided by the CLARIAH infrastructure in order to be able to do quick and efficient processing on (large) data sets without needing my own infrastructure.
- ▶ **As a scholar, I** want to apply a processing tool on a large data set *within my own infrastructure* in order to take the tools to my (possibly restricted) data and work in my own secure environment.
- ▶ **As a scholar, I** expect to have access to low-level CLARIAH tools in industry-standard ways **in order to** use the tools in my own development setting.

Deliverables (1)

- ▶ **Technical Requirements**

- ▶ Software Requirements
- ▶ Infrastructure Requirements
- ▶ *Already worked on in 2021 and first version is available for further review (RFC)*
- ▶ Interoperability Requirements?

Deliverables (2)

- ▶ **Provisioning services with documentation (WP2)**
 - ▶ Docker Registry
 - ▶ Authentication & Authorization Provider (federated, single sign-on)
 - ▶ Version control platform for infrastructure as code
 - ▶ Version control platform for services/tools
 - ▶ Monitoring Solutions for services, usage and operations
 - ▶ Continuous Integration/Deployment
 - ▶ Research data store (storing results)
 - ▶ Computational resources for test drives and limited deployments
 - ▶ Support tools (wiki, servicedesk, maintenance tools) to support the end-users and IT-staff

Gaps and Challenges

- ▶ Focus points and scope reduction
- ▶ Coordinating this, transparently, over multiple institutes
 - ▶ Not a KNAW HuC only endeavour!
- ▶ Providing clear documentation
- ▶ Distribution and deployment solutions for distributed computing workflows (DANE, NextFlow)
- ▶ To find (test) end-users who are willing to take the jump and use it
- ▶ Legal terms and conditions

Questions and or Suggestions

- ▶ Please ask now! or
- ▶ Submit an issue at <https://github.com/CLARIAH/clariah-plus/issues>