



eurac
research

Technical Solutions for Reproducible Research

CLARIN Annual Conference 2019

Alexander König, Egon W. Stemle
<Firstname.Lastname@eurac.edu>

2 October 2019

- 1 Introducing the Problem
- 2 Possible Solutions
- 3 A Case Study
- 4 Challenges and Pitfalls
- 5 Conclusions and Outlook

The Problem

- By design, research results should be verifiable by other researchers to ensure good scientific practice and easily detect mistakes.

The Problem

- By design, research results should be verifiable by other researchers to ensure good scientific practice and easily detect mistakes.
- While this is a key feature of all research, it seems that this can often be a problem - especially in the Social Sciences and Humanities (SSH).

The Problem

- By design, research results should be verifiable by other researchers to ensure good scientific practice and easily detect mistakes.
- While this is a key feature of all research, it seems that this can often be a problem - especially in the Social Sciences and Humanities (SSH).
 - Primary data is often not available to fellow researchers.

The Problem

- By design, research results should be verifiable by other researchers to ensure good scientific practice and easily detect mistakes.
- While this is a key feature of all research, it seems that this can often be a problem - especially in the Social Sciences and Humanities (SSH).
 - Primary data is often not available to fellow researchers.
 - The analysis is based on software that is obscure, not easily available or (at worst) not properly identified in the scientific publication.

The Question

The Question

- **How can we make Linguistic Research more reproducible?**

Data Sharing

- Both the primary data and the tools being used have to be made available to fellow researchers.

Data Sharing

- Both the primary data and the tools being used have to be made available to fellow researchers.
- If the data and tools can be shared publicly this has the added bonus that they can be re-used for other research as well.

Data Sharing

- Both the primary data and the tools being used have to be made available to fellow researchers.
- If the data and tools can be shared publicly this has the added bonus that they can be re-used for other research as well.
- Good guidelines for how research data should be shared can be found in the FAIR Principles (<https://www.go-fair.org/fair-principles/>).

Versioning of Data and Tools

- Linguistic corpora are often "living data", which means they constantly keep being improved and added onto.

Versioning of Data and Tools

- Linguistic corpora are often “living data”, which means they constantly keep being improved and added onto.
- All versions of a corpus that have been the basis of a scientific analysis have to be available.

Versioning of Data and Tools

- Linguistic corpora are often “living data”, which means they constantly keep being improved and added onto.
- All versions of a corpus that have been the basis of a scientific analysis have to be available.
- The same is true for linguistic tools that are being used to process the data.

- 1 Introducing the Problem
- 2 Possible Solutions
- 3 A Case Study
- 4 Challenges and Pitfalls
- 5 Conclusions and Outlook

Versioning of linguistic corpora

- Most linguistic corpora are text-based or have a text component (and it's especially this component that is changing).

Versioning of linguistic corpora

- Most linguistic corpora are text-based or have a text component (and it's especially this component that is changing).
- An existing versioning software like subversion or git can be used to track changes in the primary data.

Versioning of linguistic corpora

- Most linguistic corpora are text-based or have a text component (and it's especially this component that is changing).
- An existing versioning software like subversion or git can be used to track changes in the primary data.
- To make the various versions available and have the changes be transparent, the data can be hosted on a Code Hosting Site like github or gitlab.

Versioning of linguistic tools

- In linguistic research handcrafted toolchains built out of a variety of separate programs are very common.

Versioning of linguistic tools

- In linguistic research handcrafted toolchains built out of a variety of separate programs are very common.
- Often, it will be difficult to rebuild such a toolchain exactly.

Versioning of linguistic tools

- In linguistic research handcrafted toolchains built out of a variety of separate programs are very common.
- Often, it will be difficult to rebuild such a toolchain exactly.
 - Some tools might no longer be available or cannot be found.

Versioning of linguistic tools

- In linguistic research handcrafted toolchains built out of a variety of separate programs are very common.
- Often, it will be difficult to rebuild such a toolchain exactly.
 - Some tools might no longer be available or cannot be found.
 - It might not be completely clear which specific version of a tool was used.

Versioning of linguistic tools

- In linguistic research handcrafted toolchains built out of a variety of separate programs are very common.
- Often, it will be difficult to rebuild such a toolchain exactly.
 - Some tools might no longer be available or cannot be found.
 - It might not be completely clear which specific version of a tool was used.
 - Some manufacturers do not keep older versions of their software available for download.

Versioning of linguistic tools

- In linguistic research handcrafted toolchains built out of a variety of separate programs are very common.
- Often, it will be difficult to rebuild such a toolchain exactly.
 - Some tools might no longer be available or cannot be found.
 - It might not be completely clear which specific version of a tool was used.
 - Some manufacturers do not keep older versions of their software available for download.
- One solution is to create a (Docker) container with a "frozen" version of the complete toolchain.

Versioning of linguistic tools

- In linguistic research handcrafted toolchains built out of a variety of separate programs are very common.
- Often, it will be difficult to rebuild such a toolchain exactly.
 - Some tools might no longer be available or cannot be found.
 - It might not be completely clear which specific version of a tool was used.
 - Some manufacturers do not keep older versions of their software available for download.
- One solution is to create a (Docker) container with a "frozen" version of the complete toolchain.
- Such a container can also be made available in a public container registry.

Versioning of linguistic tools

- In linguistic research handcrafted toolchains built out of a variety of separate programs are very common.
- Often, it will be difficult to rebuild such a toolchain exactly.
 - Some tools might no longer be available or cannot be found.
 - It might not be completely clear which specific version of a tool was used.
 - Some manufacturers do not keep older versions of their software available for download.
- One solution is to create a (Docker) container with a "frozen" version of the complete toolchain.
- Such a container can also be made available in a public container registry.
- Orchestrators such as Kubernetes can help fellow researchers to easily deploy such a container to reproduce the analysis.

- 1 Introducing the Problem
- 2 Possible Solutions
- 3 A Case Study
- 4 Challenges and Pitfalls
- 5 Conclusions and Outlook

The MERLIN corpus

- At the Eurac Research CLARIN Centre (ERCC) we made some first steps in implementing these ideas.

The MERLIN corpus

- At the Eurac Research CLARIN Centre (ERCC) we made some first steps in implementing these ideas.
- By now multiple corpora are versioned via git, the multilingual MERLIN corpus being the first.

The MERLIN corpus

- At the Eurac Research CLARIN Centre (ERCC) we made some first steps in implementing these ideas.
- By now multiple corpora are versioned via git, the multilingual MERLIN corpus being the first.
- The whole corpus is available on an on-premise gitlab installation.

The MERLIN corpus

- At the Eurac Research CLARIN Centre (ERCC) we made some first steps in implementing these ideas.
- By now multiple corpora are versioned via git, the multilingual MERLIN corpus being the first.
- The whole corpus is available on an on-premise gitlab installation.
- The different versions of the corpus are realized as git tags.

The MERLIN corpus

- At the Eurac Research CLARIN Centre (ERCC) we made some first steps in implementing these ideas.
- By now multiple corpora are versioned via git, the multilingual MERLIN corpus being the first.
- The whole corpus is available on an on-premise gitlab installation.
- The different versions of the corpus are realized as git tags.
- Tagged versions are also uploaded into a CLARIN DSpace repository.

The MERLIN corpus

- At the Eurac Research CLARIN Centre (ERCC) we made some first steps in implementing these ideas.
- By now multiple corpora are versioned via git, the multilingual MERLIN corpus being the first.
- The whole corpus is available on an on-premise gitlab installation.
- The different versions of the corpus are realized as git tags.
- Tagged versions are also uploaded into a CLARIN DSpace repository.
- The DSpace and the gitlab repository are pointing at each other, so users can choose their preferred way of obtaining the data.

The MERLIN corpus

- At the Eurac Research CLARIN Centre (ERCC) we made some first steps in implementing these ideas.
- By now multiple corpora are versioned via git, the multilingual MERLIN corpus being the first.
- The whole corpus is available on an on-premise gitlab installation.
- The different versions of the corpus are realized as git tags.
- Tagged versions are also uploaded into a CLARIN DSpace repository.
- The DSpace and the gitlab repository are pointing at each other, so users can choose their preferred way of obtaining the data.
- <https://gitlab.inf.unibz.it/commul/merlin-platform/data-bundle>

The MERLIN corpus

- At the Eurac Research CLARIN Centre (ERCC) we made some first steps in implementing these ideas.
- By now multiple corpora are versioned via git, the multilingual MERLIN corpus being the first.
- The whole corpus is available on an on-premise gitlab installation.
- The different versions of the corpus are realized as git tags.
- Tagged versions are also uploaded into a CLARIN DSpace repository.
- The DSpace and the gitlab repository are pointing at each other, so users can choose their preferred way of obtaining the data.
- <https://gitlab.inf.unibz.it/commul/merlin-platform/data-bundle>
- <http://hdl.handle.net/20.500.12124/6>

Outline

- 1 Introducing the Problem
- 2 Possible Solutions
- 3 A Case Study
- 4 Challenges and Pitfalls
- 5 Conclusions and Outlook

Some examples of problems - encountered and expected

Some examples of problems - encountered and expected

(?) How to handle non-public data? Hide the git repository? Protect it with a password?

Some examples of problems - encountered and expected

- (?) How to handle non-public data? Hide the git repository? Protect it with a password?
- (!) We used password-protected git submodules to keep the general description (README,CHANGELOG) still accessible to everybody.

Some examples of problems - encountered and expected

- (?) How to handle non-public data? Hide the git repository? Protect it with a password?
- (!) We used password-protected git submodules to keep the general description (README,CHANGELOG) still accessible to everybody.
- (?) Can non-local infrastructure like github be trusted with sensitive data?

Some examples of problems - encountered and expected

- (?) How to handle non-public data? Hide the git repository? Protect it with a password?
- (!) We used password-protected git submodules to keep the general description (README,CHANGELOG) still accessible to everybody.
- (?) Can non-local infrastructure like github be trusted with sensitive data?
- (!) gitlab is available as open source and can be installed locally, meaning the data will never leave the researcher's control.

Some examples of problems - encountered and expected

- (?) How to handle non-public data? Hide the git repository? Protect it with a password?
- (!) We used password-protected git submodules to keep the general description (README,CHANGELOG) still accessible to everybody.
- (?) Can non-local infrastructure like github be trusted with sensitive data?
- (!) gitlab is available as open source and can be installed locally, meaning the data will never leave the researcher's control.
- (?) Dockerfiles do not enforce consistent versioning of installed packages.

Some examples of problems - encountered and expected

- (?) How to handle non-public data? Hide the git repository? Protect it with a password?
- (!) We used password-protected git submodules to keep the general description (README,CHANGELOG) still accessible to everybody.
- (?) Can non-local infrastructure like github be trusted with sensitive data?
- (!) gitlab is available as open source and can be installed locally, meaning the data will never leave the researcher's control.
- (?) Dockerfiles do not enforce consistent versioning of installed packages.
- (!) One has to make sure to always pin specific versions of installed software packages. Relying mostly on the built images will make this problem less important.

- 1 Introducing the Problem
- 2 Possible Solutions
- 3 A Case Study
- 4 Challenges and Pitfalls
- 5 Conclusions and Outlook

How to ensure reproducibility?

- Reproducibility of scientific research will only become more important in the future.

How to ensure reproducibility?

- Reproducibility of scientific research will only become more important in the future.
- Especially with "living data", like linguistic corpora one has to take care to ensure that findings can be reproduced by keeping older versions available.

How to ensure reproducibility?

- Reproducibility of scientific research will only become more important in the future.
- Especially with "living data", like linguistic corpora one has to take care to ensure that findings can be reproduced by keeping older versions available.
- Standard IT tools like git and docker seem to offer an easy way to handle this.

How to ensure reproducibility?

- Reproducibility of scientific research will only become more important in the future.
- Especially with "living data", like linguistic corpora one has to take care to ensure that findings can be reproduced by keeping older versions available.
- Standard IT tools like git and docker seem to offer an easy way to handle this.
- Still they have to be used with care.

What role can CLARIN play?

- We see two possible ways in which CLARIN can help

What role can CLARIN play?

- We see two possible ways in which CLARIN can help
 - 1) Develop best practices and guidelines that can help researchers in ensuring the reproducibility of their research.

What role can CLARIN play?

- We see two possible ways in which CLARIN can help
 - 1) Develop best practices and guidelines that can help researchers in ensuring the reproducibility of their research.
 - 2) Help in setting up the necessary infrastructure, for example by hosting a trusted gitlab instance that can be used to host both data and toolchain containers

Thank you for your attention!

Comments? Questions?

Alexander.Koenig@eurac.edu