CAA
Computer Applications & Quantitative Methods in Archaeology

# Keeping it REAL

## Sustainability Challenges for Research Software in Computational Archaeology


**Lutz Krister Schubert**
CAA-DE e.V.
University of Cologne

**Florian Thiery**
CAA-DE e.V.
Research Squirrel Engineers Network


## Is Reproducible Science REAL?

Open Science and **Open Data** are of highest importance for modern publications to **make results FAIR and reproducible**. This implies, however, that any step involved in producing the data is equally open and accessible - this includes in particular any code used for analysis.

To this end, code must be treated in the same way as data, by making sure that all algorithmic processes published are

- **R**eproducible in the sense that the results can be achieved again with the same process and context
- **E**xecutable at any point in time (though not necessarily on any machine)
- **A**ttributable to the data and author at the stage of publication and
- **L**iteral in so far as that the algorithm is a sound and correct representation of the mathematical methods to be applied.

In general, scientists will generate their own code before reusing existing methods - mostly because of the complexity involved in (a) **understanding existing code** and (b) **adapting it to the given needs**. This means that code *for the same functionality* will be re-implemented times and times again, making it difficult to assess its "**L**iteral"ity.

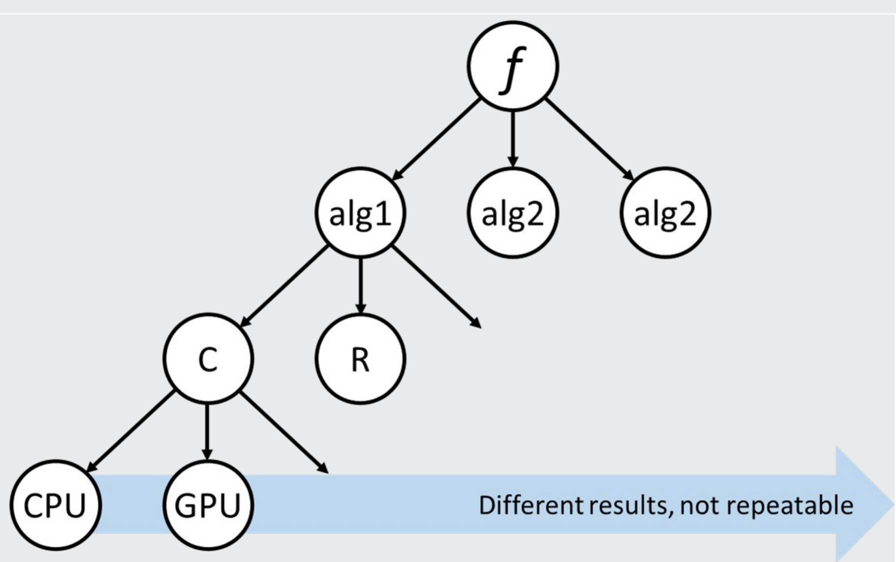


**Fig. 1**: Logical process from function to (running) code.

And even if two scientist have the same (mathematical) function in mind, they can choose different ways of realising and executing said function (cf. *Figure 1*).

**Links to Discover**

- The Turing Way: https://book.the-turing-way.org/reproducible-research/overview/overview-definitions.html
- https://littleminions.link
- https://sslarch.github.io

## Reproducibility is not REAL

The same ***algorithm*** can lead to completely different results depending on (1) choice of language, (2) data encoding, and (3) solver used (if needed). Similarly and even worse, the same ***code*** can vary in results basing on (1) hardware it is executed on, (2) compiler (settings) chosen, and (3) if solvers or AI is used, random coefficients.

To give a simple example, adding the number $^1/_{10}$ in **C** with `float` data type results in *1.0000001192092896* - reusing this value for further calculations increases the error accordingly.

Given that hardware, compilers and language develop over time. The exact same code will not produce the same results anymore after 5 years. For example, bit-precision has changed from 4-bit to 128-bit in the last 40 years and higher precisions are to be expected (cf. *Table 1*).

| | |
|---|---|
| **16 bit** | 3.143 |
| **32 bit** | 3.142857 |
| **64 bit** | 3.142857142857143 |
| **128 bit** | 3.1428571428571428571428571428571428 |

**Table 1**: precision comparison for the value of 22/7 with different bit-lengths.

## Being REAListic

**Research Data Management** and specifically **archiving research software** typically therefore chooses a simplistic approach: it virtualises the hardware and all components at the time of code execution. The whole virtual image is then stored for any reproduction purposes.

This ensures that the same results can be reproduced - to a degree. In particular where random parameters play a role in calculating the results, such as in Simulated Annealing, k-means or Monte Carlo methods, the same results can only be reproduced if the exact same random seeds are fed. In addition, effort, cost, storage space are excruciating, limiting applicability seriously.

## Rethinking what's REAL

Following the "Turing Way", reproducibility is strictly bound to the same data and same type of analysis (aka code). As noted though, very few algorithms, let alone codes, will create the exact same results unless under very controlled circumstances (cf. *Figure 2*).

What does that actually tell us though? First of all nothing else but that the code *executes*. After all

```
void main(){return rnd();}
```


**About the Poster**

deRSE25 at KIT, 25-27 Feb 2025

deRSE CONFERENCE KARLSRUHE 2025

Schubert, L. K. & Thiery, F. (2025). Keeping it REAL Sustainability Challenges for Research Software in Computational Archaeology. Squirrel Papers, 7(3), λ4. https://doi.org/10.5281/zenodo.14898291.


| | | Data | |
|---|---|---|---|
| | | Same | Different |
| Analysis | Same | Reproducible | Replicable |
| | Different | Robust | Generalisable |

**Fig. 2**: Reproducibility according to the Turing Way.
Note how the conditions are extremely controlled.

## Do we need it to be REAL?

As we can see, the strict ***Turing Way* definition** - and therefore the generally acknowledged definition of **FAIR and Open Science Code**, namely to be **fully reproducible** - does not lead to actual insights into the ***correctness*** (i.e. "**L**iteral"ity) of the code. We take it for granted by noting that code and data are made **accessible**.

However, what we are really interested is the scientific relevance and validity of a publication - in other words, whether the results apply to a wider range of conditions. This is immediately obvious with methods such as Monte Carlo or k-means which rely on ***stability over a series of random parameters***.

| | | Data | |
|---|---|---|---|
| | | Same | Same |
| Analysis | Same | Reproducible | Replicable |
| | Same | Robust | Generalisable |

**Fig. 3**: Modified Turing Way.
Note how the same conditions will lead to different interpretations of the results.

The implicit statement of such an analysis differs from ***Reproducibility*** and is closer to ***Replicability*** and ***Robustness*** of the ***Turing Way***. Yet, we need to bear in mind that ***data** and **analysis** have **not changed***. Instead, as depicted in *Figure 3*, they stay the ***same***, yet lead to ***different results***.

We therefore distinguish between different ***Interpretability*** of the results in terms of:

- ***Reproducible***: data, code and all environmental conditions are identical. Interpretation of the results is limited to the conditions applied
- **Replicable / Robust**: the results differ per execution, but stay within the expected tolerance. In other words, the ***interpretation*** of the results is unchanged.
- **Generalisable**: even if all conditions are varied, the results stay within the expected boundaries. Accordingly, the interpretation can only be that the assumption is ***universally applicable***.

Future research must clearly identify **the degree of tolerance the *interpretation* of their results allows.**