

A Conceptual Framework for Computational Reproductions: Formal Definitions and Epistemic Functions

Florian Kohrt¹, Filip Melinscak², Richard McElreath³, Felix D. Schönbrodt¹

¹ Department of Psychology, Ludwig-Maximilians-Universität München, Munich, Germany

² Faculty of Psychology, University of Vienna, Vienna, Austria

³ Department of Human Behavior, Ecology, and Culture, Max Planck Institute for Evolutionary Anthropology, Leipzig, Germany

Author Note

Florian Kohrt  <https://orcid.org/0000-0003-0374-5625>

Richard McElreath  <https://orcid.org/0000-0002-0387-5377>

Filip Melinscak  <https://orcid.org/0000-0001-8767-358X>

Felix D. Schönbrodt  <https://orcid.org/0000-0002-8282-3910>

FK was funded by the German Research Foundation (DFG SCHO 1334/6-1, awarded to Felix Schönbrodt) in the META-REP Priority Program (SPP 2317). FM was supported by the Austrian Science Fund (FWF) (grant DOI:10.55776/ESP133).

Correspondence concerning this article should be addressed to Felix Schönbrodt, Department of Psychology, Ludwig-Maximilians-Universität München, Munich, Germany. Email: felix.schoenbrodt@psy.lmu.de

Pre-print, version 2.1 from 2024-12-05. Not peer-reviewed yet, use at your own risk.

The authors would like to thank Daniel Krähmer, Laura Schächtele, and Julian Möbus for their helpful comments on earlier drafts of this paper, and Daniel Leising for the helpful description of VAST displays.

Please cite as:

Kohrt, F., Melinscak, F., McElreath, R., Schönbrodt, F. D. (2024). A conceptual framework for computational reproductions: Formal definitions and epistemic functions. *Zenodo*. <https://doi.org/10.5281/zenodo.10053573>

Abstract

Reproductions, which we define as “redoing activities” that work with the same data and aim to keep the computation as similar as possible, enable the verification of a scientific work. We seek to advance the understanding and conduct of reproductions by providing a conceptual framework and guidance to reproducers. In the first part, the epistemic function of reproductions and direct replications with regard to mistakes – and faults as a consequence – is discussed. In the second part, the logic underlying reproductions is formulated, accompanied by formal definitions of central terms. The third and final part will provide practical guidance on reporting reproductions. In short, a reproduction should involve checking the coherence of relevant instructions, performing another computation, and comparing the results for consistency. Evaluating the consistency means exploring the impact of computational choices on results before suspecting faults as a last resort. Computational choices are necessitated by obstacles such as underspecification and grounds for conflicting choices. Regarding faults, four cases are emphasized: (1) incoherent descriptions, (2) different data, (3) different source code, and (4) incorrect reporting of results. Separate from evaluating the consistency of results, reproductions can investigate their support for a claim. Finally, improper reporting practices can reduce the epistemic value of reproductions. Therefore, reproducers should also report the scope of their reproduction by detailing, among other things, which materials were consulted and to what extent any data were already preprocessed.

Keywords: error, mistake, fault, reproduction, replication, meta-science, conceptual framework, verification, formal model, guide, epistemic function, uncertainty

1. Introduction and Previous Research

The advent of open science in the past decade has given rise to many recommendations for improving the computational aspect of one’s research (Kohrt et al., 2024; Krafczyk et al., 2021; Mineault et al., 2021; Sandve et al., 2013; Wilson et al., 2017) or its independent verification (Ankel-Peters et al., 2024; Arguillas et al., 2022; Berkeley Initiative for Transparency in the Social Sciences, 2020; de la Guardia et al., 2024; Pilgrim et al., 2023; Stodden, 2015; Wilensky & Rand, 2007; Zhang & Robinson, 2021). However, researchers still face several challenges during such “redoing activities”¹ – scientific activities that redo some phases of a previous study. For example, when verifying the data analysis or the simulation published in a previous study, the study may contain multiple results and the researchers’ time and resources may not be sufficient to calculate all of them. Then, it is up to the reproducers to select an appropriate and indicative set of results to reproduce, even if the original study does not contain any hints in this regard. Furthermore, the data may be

¹ Similar concepts also go by the name *reanalyses* (Welch, 2019) and *repetitive research* (Schöch, 2023).

unavailable or only in preprocessed form. Similarly, the source code may have not (or only partially) been shared and may not run. Moreover, even if results are obtained eventually, they may differ from the original results, and one wonders whether the verification was successful. Finally, the authors are often no longer available to ask for clarification. This article sets out to improve on the status quo by advancing the understanding and conduct of redoing activities with an emphasis on the computational aspects.

In existing conceptual work on redoing activities, it has been noted that fields differ in how they call various redoing activities (Barba, 2018) and that many definitions confound the activity, the ability to conduct it, and its result (Penders et al., 2019). Plesser (2018) even speaks of a “confused terminology”, although – or maybe because – many unifying definitions are being proposed. For example, Heroux et al. (2018) call for a compatible usage of the terms “reproduction” and “replication” across disciplines, suggesting a focus on internal consistency for the former (“doing things right”) and a focus on external consistency for the latter (“doing the right things”). Others developed approaches to systematically categorize such activities along different dimensions (e.g., Dreber & Johannesson, 2024; LeBel et al., 2018; Nichols et al., 2017; Patil et al., 2016a; Schöch, 2023). In particular, Goodman et al. (2016) specifically differentiate the provision of details (“methods reproducibility”), obtaining the same results (“results reproducibility”), and drawing similar conclusions (“inferential reproducibility”). In a similar vein, Ulpts and Schneider (2024) suggest differentiating redoing (e.g., reproducing a previous study) from enabling (e.g., being transparent about the data analysis). They hold that researchers should state explicitly what is being enabled or redone and for what reason rather than relying on a particular terminology. To summarize, the existing literature on terminological issues suggests differentiating the act of enabling, the act of redoing, obtaining the same result, and reaching the same conclusion. This distinction is also echoed in the present article.

The reason for conducting a redoing activity relates to its epistemic functions, which also have been studied from a meta-scientific perspective. For example, Matarese (2022) details how changing irrelevant or relevant factors of a study allows for evaluating its reliability and validity. Schmidt (2009) specifically discusses five epistemic functions of redoing activities that collect new data: controlling for sampling variability, internal validity, and fraud, as well as generalizing results and verifying hypotheses. Bayarri & Mayoral (2002) highlight four goals of a replication: reduction of random error, validation (confirmation) of conclusions, extension of conclusions, and bias detection. Steiner et al. (2019) have developed a framework that allows for a causal interpretation of replication failures. López-ibáñez et al. (2021) also consider redoing activities that recreate digital objects like source code and mention that they are unlikely to contain the same faults, thus hinting at their function for detecting mistakes. Clemens (2017) is even more explicit in this regard: Redoing activities that use the original data set or sample and thus expect “materially the same results” (p. 327) can identify issues with measurement, coding, and data set construction, as well as scientific misconduct.

However, the epistemic functions of redoing activities that work with the original data (in the case of empirical works) and aim to keep the computation (e.g., a data analysis) as similar as possible (in the following, “reproductions” or synonymously “computational reproductions”) have received little systematic attention, and one is left with little guidance on what makes a result “materially the same”. An exception to this comes from LeBel et al. (2018), who propose four dimensions for evaluating the credibility of scientific findings, which

they call “method and data transparency”, “analytic reproducibility”, “analytic robustness”, and “effect replicability”. Like Nuijten et al. (2018), they propose a multistep approach to evaluating studies with redoing activities but also provide brief standardized workflows, including a 10% rule for comparing the original with the reproduced result. This rule balances rigor and usefulness as discussed, for example, by Patil et al. (2016b), for redoing activities that involve collecting new data. However, it falls short of accounting for ambiguity in the description of the data analysis. On this matter, Artner et al. (2021) propose to differentiate the correctness from the vagueness when evaluating the reproducibility of numerical results. Correct results can be reproduced without violating the data analytical descriptions in the study. The reproducibility is vague, however, if the descriptions allow for multiple correct results. This distinction allows putting results into the context of their description and is also reflected in the present article.

Following the lack of research on the epistemic function of reproductions, section 2 of this article introduces their verifying function and explains what can be learned from their conduct. Section 3 explains their underlying logic and defines central terms around them. Section 4 provides practical recommendations on their conduct and reporting and discusses its possible outcomes.

The challenges described initially relate to different issues when verifying the results of existing research:

- **Target selection** refers to which of the many results a redoing activity should focus on.
- **Issues with data and code** relate to their lack of availability, provenance, correctness, and executability.
- **Evaluation of results** is difficult when identity is expected but frequently not encountered. In the case of figures, standards for comparison may even be lacking.
- **Aggregation** is the issue of integrating and reporting multiple results, both on the level of individual studies and across them.

This article will focus on issues with data and code and the evaluation of results, though it will also touch on the other issues.

Note that we strive to use consistent language for identical concepts throughout this article – although any statements only pertain to the concepts rather than the labels we assign to them. For example, a scientific work that is the target of a reproduction attempt is referred to as the *original work*, its authors as *original authors*, and its results as *original results*. The counterparts to these terms are a *reproduction*, *reproducers*, and *reproduced results*.² Further, we use the terms *mistake* and *fault* according to international standards (International Organization for Standardization, 2017) by distinguishing a human action (a mistake) and its manifestation (a hardware or software fault).

² Importantly, a reproduced result is the one that corresponds to a particular original result, but is not necessarily identical to it.

2. The Verifying Function of Reproductions

In the following section, we describe where mistakes in the research process can occur, which redoing activities can detect the resulting faults, and how their combination can be insightful.

Research *producers* (i.e., the authors of an original work) commonly recognize and communicate their state of uncertainty about obtained results and try to quantify parts of it by making stochastic assumptions, for example, when inferring from a sample to a population and by providing confidence intervals. Similarly, most fields consider the effects of measurement errors, for example, using *structural equation models* in the social sciences (Hoffmann et al., 2021) or via *propagation of uncertainty* in physics. In the computing sciences, *interval analysis* can be used to investigate the limits of floating-point operations (Diethelm, 2012).

However, research *consumers* (i.e., an original work's readers) are confronted with more potential sources of uncertainty that the research producers have not considered or not reported, contributing to an information asymmetry between them (Young, 2009). A prominent example is *researcher degrees of freedom*, that is, the multitude of data collection and analysis options, which may call the robustness of a given result into question if only one path is reported (e.g., Botvinik-Nezer et al., 2020; Breznau et al., 2022; Huntington-Klein et al., 2021; Menkveld et al., 2024; Silberzahn et al., 2018). Equally, publication bias and questionable research practices might distort the results of a meta-analysis (Carter et al., 2019). As is the case with the uncertainty of research producers, research consumers can attempt to reduce their uncertainty, as is exemplified by robustness checks³ and studies that analyze publication bias (Franco et al., 2014).

Another instance of uncertainty among research consumers that is of interest to this article concerns the existence of faults within the computation of an original work.⁴ Redoing activities that deliberately redo some phases of a previous work as close to the original intent as possible while inheriting⁵ all remaining phases can reduce this uncertainty – we refer to this internal consistency check as “verification”. Verification is only one of many possible epistemic functions of reproductions – for example, in their review of the literature, Ulpts & Schneider (2024) also mention learning, training, and understanding. In the following, however, only the verifying function of reproductions will be detailed. In particular, we focus on two kinds of reproductions that both work with the original data and discuss a “direct replication” in comparison:

- Re-executing the original source code on the original data (“re-execution reproduction”). Usually, the methods section in the article is not considered here.

³ A multitude of names exists for similar concepts across disciplines, for example *multiverse analysis* (Steege et al., 2016), *sensitivity analysis* (Saltelli, 2008), *specification search* (Leamer, 1978), *vibration of effects* (Patel et al., 2015), *multimodel analysis* (Young & Holsteen, 2017), or Bayesian model averaging (Chatfield, 1995).

⁴ Note that we assume honest actors and always attribute to a mistake rather than fraud – though we don't expect maximal transparency.

⁵ In this context, inheriting a phase from an original work means that it is only performed once across original work and reproduction and its product feeds into the following phases and is thus identical for them.

- Re-implementing the source code and running it on the original data (“re-implementation reproduction”). Usually, it is only based on the methods section of an article, not considering the original source code.
- Collecting new data and re-executing or re-implementing the source code (“direct replication”). In the following, we only discuss the case where the source code is re-implemented.

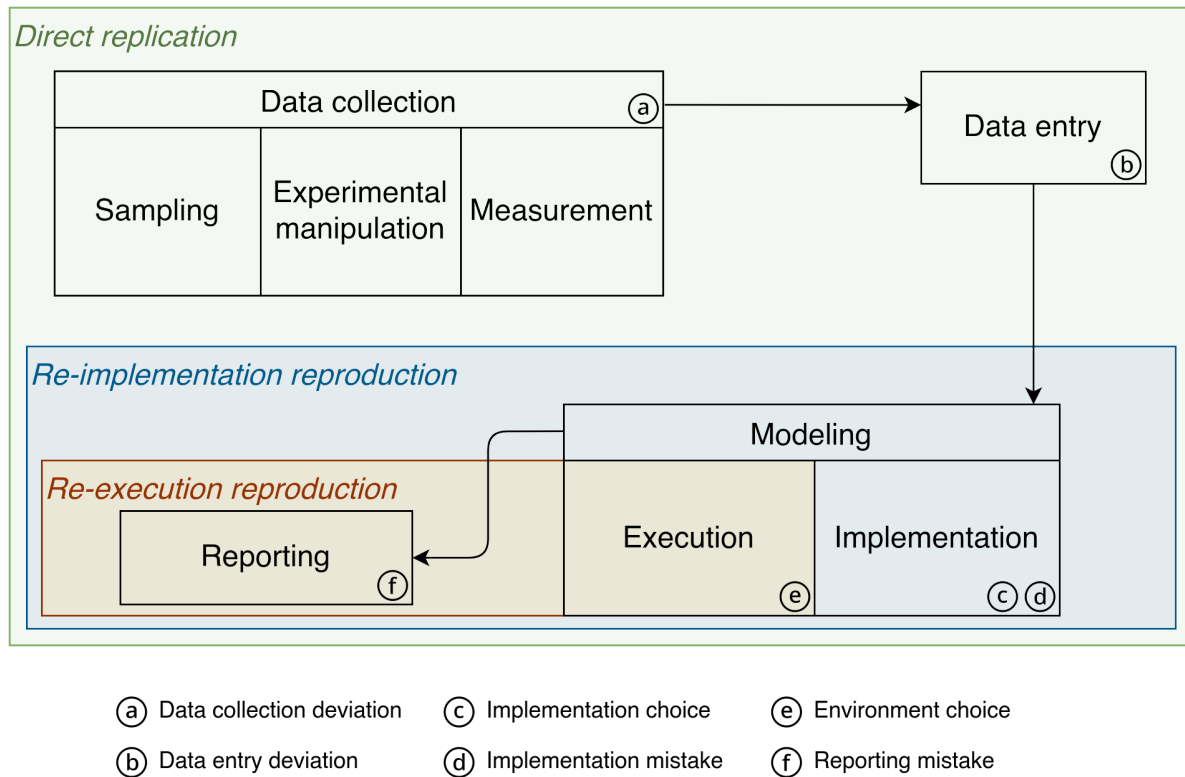
All three activities represent particular sets of phases that are redone out of all possible phases of a simplified research process, as detailed in Figure 1: data collection, data entry, modeling, reporting. Here, every phase covered by the colored area of a redoing activity is done again according to the description in the original work, while all remaining phases are directly inherited (i.e., not redone) from the original work. For example, for a re-implementation reproduction, one inherits the data collection and data entry of an original work, attempts to redo the model, and reports the results again. The data collection phase is further differentiated into sampling, experimental manipulation, and measurement. For the modeling phase, implementation and execution of a model are differentiated, which is important for some reproductions.⁶ Here, execution involves running an implementation within a computational environment.⁷ Of course, not all phases are part of every research process. Also note that these three redoing activities represent prototypical cases that may blend in reality. For example, a re-implementation reproduction might incorporate portions of the original source code rather than writing everything anew. However, in the following, we only consider the typical case.

⁶ Execution and implementation may also be inseparable, as is the case when using a pocket calculator.

⁷ By computational environment we refer to the hardware and software setup which is utilized while calculating the results.

Figure 1

Simplified Research Process With Its Possible Phases, Enclosed by Redoing Activities and with Associated Reasons for Differences in Results (a) through (f)



Note. Not all phases are part of every research process. (a) “data collection deviation” is a collective term that includes sampling variability, sampling bias, erroneous measurements, and other biases that can be introduced during data collection. Similarly, (b) “data entry deviation” is caused by all sorts of choices and mistakes, including decisions during digitalization of the data and typos.

While inherited phases should be identical between the original work and the redoing activity (i.e., are performed only once with their product feeding into the following phases), redone phases may deviate despite aiming for equality. For every phase, Figure 1 indicates associated reasons for differences in results as circled letters (a) through (f). These must be considered if, and only if, the respective phase is redone and the obtained result of a redoing activity is not the same.⁸ Both identical and non-identical results influence the reproducers’ state of knowledge about associated reasons by providing evidence for or against them. This simplified research process assumes that choices by the reproducers and mistakes by original authors or reproducers are solely responsible for different results obtained by reproductions. However, no choices are possible while reporting results, and no mistakes can happen during the execution of a model. Further, we differentiate an implementation mistake (made while implementing a data analysis or a simulation) from a reporting mistake (happening when writing up the results during reporting) and reproductions can detect both

⁸ By *same* we mean exact numerical identity, at the maximum precision that the representation of a result allows. Section 4 of this article introduces *consistency* as a measure of “sameness” that takes into account underspecified descriptions and conflicting choices.

of them,⁹ as will be detailed next. In general, mistakes may also happen earlier and result in deviations during the data collection or data entry, though they are not the primary focus of a reproduction. We use “deviation” as a generic term for the difference between an original and a redone phase, which can have all sorts of reasons, including statistical variability, biases, faults, and varying choices. Mistakes that happen later, that is, when interpreting the results, are also mostly ignored by reproductions, though we touch on them in section 4 of this article when discussing “inappropriate inferences”.

Figure 1 details the verifying function of certain redoing activities and helps to attribute differences in results to their possible reasons. For example, consider the case where the computation described in a work (e.g., a data analysis) has been re-implemented, resulting in a different effect size than originally reported. The re-implementation reproduction in Figure 1 is represented with a blue area covering three phases (implementation, execution, and reporting) that are redone, and consequently, four potential reasons for differences in results: environment choices (e) or implementation choices (c) by the reproducers, mistakes during the – original or redone – implementation (d), or reporting mistakes during manuscript preparation of the original work or the reproduction (f). The data are inherited from the original work, and thus, nothing during data collection or data entry can account for differences in results. Note that this allows for no statement whether the inherited phases were performed in a sensible and appropriate fashion. If the original authors made nonstandard or suboptimal decisions during data collection, an otherwise reasonable data analysis does not fit, and one would possibly criticize the original work, rather than the reproduction for this.

The same information is presented in condensed form in Table 1. It displays the state of knowledge about reasons for differences in results for three different redoing activities depending on the method of comparison. For now, only the comparison for identity is of interest. In the example, all columns with a large disk in the row *Re-implementation reproduction* → *Identity* → *Non-identical* are possible reasons for different results: reporting or implementation mistakes while conducting the – original or re-implemented – work, the environment choice, and an implementation choice (e.g., due to an insufficient description of the computation). If, in contrast, the re-implementation reproduction had resulted in identical results, all potential reasons indicated by a small disk can be considered unlikely. The equals signs of that row indicate that the re-implementation reproduction shares the sampling variability and any other data collection or data entry deviations with the original work.

⁹ More specifically, reproductions can detect faults that cause the computation to be at odds with its descriptions or, by extension, with any data or results the descriptions refer to.

Table 1

Redoing Activities and the State of Knowledge Their Result Implies About Potential Reasons for Differences

Redoing activity	Comparison	Outcome	Reporting mistake	Environment choice → different environment	Implementation mistake	Implementation choice → different implementation	Sampling variability	Other data collection deviation	Data entry deviation
Re-execution reproduction	Identity	Identical	•	•	=	=	=	=	=
		Non-identical	●	●	=	=	=	=	=
	Consistency	Consistent (but non-identical)	⊙	●	=	=	=	=	=
		Inconsistent	●	⊙	=	=	=	=	=
Re-implementation reproduction	Identity	Identical	•	•	•	•	=	=	=
		Non-identical	●	●	●	●	=	=	=
	Consistency	Consistent (but non-identical)	⊙	●	⊙	●	=	=	=
		Inconsistent	●	⊙	●	⊙	=	=	=
Direct replication	Consistency	Consistent	⊙	●	⊙	●	●	⊙	⊙
		Inconsistent	●	⊙	●	⊙	⊙	●	●

Note. The original and the reproduced result can either be compared for identity or for consistency (as described in section 4 of this article). The size of disks indicates the importance of various reasons for differing results. A small disk (•) indicates that there is no difference or that a reason is less important. A large disk (●) indicates a potentially important reason for non-identical or inconsistent results (but we do not know which one if a row contains more than one). A circle around a small disk (⊙) indicates that a reason is possible, but the caused difference is probably small in comparison to reasons with a large disk. Disks are red if results are likely non-identical or inconsistent due to a mistake and green otherwise. The equals sign (=) indicates that the phase containing this reason is inherited and, therefore, assumed to be identical. Therefore it cannot by itself be responsible for differences in results, while at the same time, issues in that phase cannot be detected.

Returning to the scenario where the re-implementation reproduction resulted in non-identical results, one cannot conclude which of the potential reasons with a large disk in Table 1 are responsible for the difference. To narrow it down further, a re-execution reproduction can be performed. Assuming that it leads to identical results as originally reported, reporting mistakes and environment choices common to the re-implementation and the re-execution reproduction can be considered unlikely because they have a small disk in the respective row of Table 1. Hence, implementation choices and implementation mistakes (either while conducting the original work or the reproduction) remain as potential reasons. Section 4 of this article discusses how these can be discerned by evaluating the consistency of results.

Most informative are cases where a large set of reasons can be ruled out (indicated by small disks), or a small set of reasons is identified (indicated by a large disk). If successful, we learn more from an re-implementation than from a re-execution reproduction, while it is the other way around if they produce non-identical results. The previous example also illustrates how different redoing activities can be combined to learn more about particular reasons for differences in results. Notably, no evidence against implementation faults can be produced by a typical re-execution reproduction, as it does not consider the description in the methods section. Because the term *reproduction* can also refer to a re-implementation reproduction, which differs in its verifying function, we recommend always indicating which exactly is conducted. However, when it comes to atypical cases, even this distinction is not sufficient, hence section 3 of this article introduces the concept of a reproduction's scope, which we also recommend to report.

Of course, by using Figure 1 and Table 1 to identify possible reasons for differences in results, one makes certain assumptions, which, if violated, invalidate the conclusions. This concerns, in particular, the inherited phases and whether their product is actually identical. For example, if reproducers obtain the source code from the original authors of a study, they operate on the belief that it is indeed the exact version that produced the study's result. Instead, the authors could have (unintentionally) sent an old version of the code. Only if data, computation, and reporting are kept together via executable research compendia (Gentleman & Temple Lang, 2007; Nüst et al., 2017) can one be reasonably confident that they stayed constant. As another example, data might have been preprocessed, either for usability reasons or, by necessity, due to resource limits. This leaves some mistakes undetectable during re-implementation reproductions, possibly without the knowledge of reproducers. For this reason, Table 1 should only be treated as a heuristic guide, providing evidence for or against particular types of mistakes, but not making definitive statements. These and other possibilities are considered in section 3 of this article.

3. Defining Reproductions

In this section, we sketch the logic of reproductions before formally defining reproductions and surrounding concepts by creating formalized displays.

The Logic of Reproductions

Up next, we first describe the logic of a reproduction: one possible goal, the procedure to meet that goal, and the preconditions. Next, we explain how the presence of faults is

determined. Finally, we describe two obstacles that are commonly faced during reproductions.

One goal of a reproduction can be to verify that the computation of a selected set of results corresponds to its specified choices, as defined in one or more descriptions. Examples of computations include data analyses but also simulations that do not require data. The computation itself is uncertain¹⁰ but constrained by its descriptions, the results, and, for empirical studies, the data. Therefore, a reproduction involves creating another computation using the descriptions, inheriting the data, and comparing the reproduced with the original results. The comparison is the main outcome of such a reproduction. It follows that the availability of at least one description, one result, and, if applicable, data are preconditions to performing a reproduction. Beyond the comparison, optionally, the support for the claim can be evaluated.

In section 2 of this article, the comparison for identity has been discussed. If the results are exactly the same, one can conclude that the original computation has likely been performed as described¹¹. If they do not match exactly, however, Table 1 lists mistakes *and* computational choices (i.e., environment choices and implementation choices) as possible reasons. Only mistakes should make a verification fail, because computational choices are necessitated by insufficient descriptions or a lack of resources (such as special software requirements), neither of which automatically threatens the internal consistency of a work.¹² Consequently, such obstacles should be taken into account: Rather than demanding identity, we suggest that results are called “consistent” if they differ no more than what is expected due to obstacles and given the methods standards chosen by the reproducers. This also provides an alternative to the suggestion by LeBel et al. (2018) to always allow for a difference of 10%.

Therefore, it is necessary to understand the reason(s) for the observed difference between the original and the reproduced result. Estimates for the difference introduced due to the reproducers’ computational choices can be considered, for example, by demanding that results only match up to a certain precision rather than exhibit exact numerical identity: If it is known that the result of a stochastic algorithm is robust up to the fifth digit across multiple runs, any larger differences may be attributable to faults.

In the following, actions that cause a computation to be at odds with its description are considered mistakes, manifesting as faults. A common mistake in the context of reproductions is writing incoherent descriptions. Other erroneous actions, such as choosing inappropriate statistical methods, might be considered mistakes in different contexts but are

¹⁰ For example, a graphical data analysis software leaves no trace after it has been used. But also source code is often not enough to determine exactly what has been computed, for example, due to changes induced by dependency versions or hardware.

¹¹ Technically, the original computation could still deviate from its description but accidentally give the same results as during the reproduction. If the domain of possible results is constrained (either discrete or for some other reason only certain values are possible), then this becomes more likely. In the special case of forensic reproductions where it is not acceptable to err, additional provenance might be required to continue at this stage.

¹² It is at the reproducers’ discretion to set the standards for any methods and thus define how to evaluate the internal consistency. Thus, for example, this includes the evaluation whether the choices made in the source code are consistent with the instructions in a methods section.

not the primary focus of a reproduction.¹³ Once faults are suspected, limited resources may prevent reproducers from investigating further, even if identifying them would be a valuable contribution. The identification typically involves the reproducers obtaining the same faulty result and a proper explanation. Alternatively, such an explanation might be provided by the original authors. In addition to the identification of faults, a reproduction can improve the scientific literature if it also reports the results after correcting all faults.

Obstacles can prevent reproducers from (initially) obtaining a result. In the following, two common types of obstacles are differentiated (although there may be others). First, the descriptions of the computational steps may be underspecified, providing too little information for the reproducers to know exactly what to do and making them choose any appropriate option. Second, although knowing what to do, the reproducers might be unable to follow the descriptions to the last detail, requiring them to deliberately deviate from the descriptions to obtain any result. In general, obstacles require the reproducers to make computational choices that potentially cause deviations from the original computation.

In case the original authors have shared the full source code, the reproducers might be able to directly identify faults as the reason for an inconsistency. However, if the source code has not been shared, underspecified descriptions and conflicting choices can complicate the identification of faults. This is because both have the potential to completely explain differences between results. For example, if the reproduction uses a different statistical software, at first glance, this might be considered the sole reason for a different result. Only by investing additional effort and attempting to assess how much difference between results the choice of software actually explains can one reasonably suspect or rule out faults as an additional reason for different results.

Box 1

Summary: Logic of Reproductions

- **Goal:** A reproduction checks whether a selected set of results from a computation can be obtained by following the descriptions and, if applicable, involving the data. To this end, we suggest comparing the original and reproduced results for consistency rather than identity.
- **Procedure:** Original and reproduced results are compared. If differences are observed, reproducers attempt to systematically explain them and attribute them to computational choices due to resolved obstacles or to faults.
- **Obstacles:** Obstacles can (initially) prevent reproducers from obtaining a result. These include underspecification and grounds for conflicting choices.
- **Mistakes:** Actions that cause a computation to be at odds with its description are mistakes, such as writing incoherent descriptions. They are either identified by closely studying the available materials, possibly with the help of the original authors, or they are suspected by ruling out other possible reasons for differing results. If identified, reproducers may provide corrected results.

¹³ However, such concerns can be raised if a reproduction also chooses to evaluate the claim(s) of the original work.

- **Results:** The reproduced results are consistent with the original results if their difference can be explained by the obstacles reproducers encountered.

Busy readers may skip the following formal definitions and immediately jump to their summary in Box 2.

Introduction to VAST Displays

In the rest of this section, we present formal definitions for central terms around reproductions. Starting with computations and descriptions, we delineate what reproductions are about at their core. We continue by defining scientific works and reproductions before exploring three reasons why a reproduced result can differ: underspecification, conflicting choices, and faults. Finally, we highlight the central importance of source code for reproductions.

Certain statements are accompanied by figures representing formalized depictions of our definitions using the visual argument structure tool (VAST; Leising et al., 2023). They start with the number of the VAST display and the specific number of that statement in square brackets. If a statement defines a particular term, the term appears in bold font. Parentheses are used for optional parts of the term, which can be left out for brevity. The statement itself is typeset in italic font. Any further terms (regardless of whether used for the first time or subsequently) are typeset in small caps and, if used for the first time, defined in their own indented statement directly following up. Examples or citations after the statement are in roman (non-italic) font again.

VAST comprises a set of rules for expressing the content of narratives more formally in a graphical display. In VAST, “concepts” are properties that may apply to objects to varying degrees and which are depicted by frames. There may be IF-THEN relationships of different qualities, and of different strengths, between concepts, depicted by arrows.

For example,

- IF concept D applies to an object, THEN this object is likely to be named (n) “dog”.
- IF concept D (named “dog”) applies to an object, THEN this implies (i) that concept A (named “animal”) also applies to this object.
- IF concept M (named “height measured in meters”) applies to an object to some degree, THEN one may deduce the extent to which concept C (named “height measured in centimeters”) applies to the same object, by way of a mathematical transformation (t).
- IF concept S (named “smokes”) applies to an object (e.g., a person), THEN this is a cause (c) for the fact that concept L (named “has lung cancer”) may also apply to the same object some time later.
- IF concept C (named “has a college degree”) applies to an object (e.g., a person), THEN it may be predicted (p) that concept P (named “has parents who have college degrees”) also applies to the same object.
- IF concept NA (named “has no alibi”) applies to an object (e.g., a person), THEN it may be reasoned (r) that concept G (“is guilty”) also applies to the same object.

The VAST system includes rules for expressing the extent to which a concept *IS* applicable to a given set of objects (depicted as a pentagonal shape with “IS”), and the extent to which a concept *OUGHT* to be applicable. In contrast to imagined concepts, measurable concepts are displayed with a thick black edge on one side. Finally, the VAST system also allows for specifying *who* holds a given set of views (“perspective”, shown as an ellipsis), and for treating any combination of the aforementioned elements as a higher-order concept of its own.

Computations

A reproduction's primary target is a result produced by a computation and reported in a scientific work, potentially together with data (in the case of an empirical study) and a derived claim (if available). It is also possible that one reproduction targets multiple results or claims.

[Fig. A9 → n_1] **Computation:** *A particular calculation of one or multiple RESULTS that is (if applicable) dependent on (1) specific DATA, (2) a specific COMPUTATIONAL ENVIRONMENT, and (3) specific CODE.*

[Fig. A10 → n_2] **Result:** *A product of a COMPUTATION. May be used as support for one or multiple CLAIMS.* For example, a result can be a singular number, a table, a figure, or a collection of such elements.

[Fig. A11 → n_2] **Claim:** *A conclusion drawn from one or multiple RESULTS.* For the purpose of a reproduction, the support for a claim can be found to be strengthened, weakened, or undetermined. It is at the reproducers’ discretion to decide which parts of a scientific work comprise a claim, as reproductions of single studies typically allow for a higher granularity than mass reproductions of multiple scientific works. For example, one statement by the original authors could be split into multiple claims by the reproducers.

[Fig. A8 → n_1] **(Empirical) Data:** *Records obtained from a measurement process.*

[Fig. A9 → n_2] **(Computational) Environment:** *The hardware and software setup in which a COMPUTATION is performed, for example, versions and (default) parameters of utilized software (dependencies), invisible states (such as of random number generators), and the architecture of hardware (such as the CPU or the specific pocket calculator).*

[Fig. A6 → n_1] **(Source) Code:** *A low-level DESCRIPTION of a COMPUTATION that can be executed by a computer.*

Put simply, a computation is the act of performing a calculation. Examples include

- using a pocket calculator and a *t*-distribution table to perform a *t*-test;
- using statistics software with a graphical user interface such as *JASP* (JASP Team, 2024) to conduct a Bayesian regression;
- executing an *R* script (R Core Team, 2024) to perform a permutation test; or
- running an agent-based simulation in *C*.

Descriptions

Descriptions of computations are central to reproductions, as they enable the verification of results (ICMJE, 2024; Livingston, 2020). In the following, we adopt a very broad notion of what constitutes a description:

[Fig. A3 → n_1] **Description (of a computation):** *Instructions on performing a COMPUTATION detailing COMPUTATIONAL CHOICES, possibly referring to specific DATA and RESULTS and mentioning a particular COMPUTATIONAL ENVIRONMENT.*

[Fig. A5 → n_1] **(Computational) Choices:** *Decisions related to a COMPUTATION.*

An obvious example of a description is the methods section of a journal article detailing the analytical treatment of data. However, one can also treat a preregistration, the supplementary materials provided with a journal article, or source code as a description of a computation. Treating source code as a (human-readable) description highlights its usefulness for understanding and reproducing the original work long after it stopped being immediately re-executable.

Scientific Works

[Fig. A11 → n_1] **(Scientific) Work:** *A compilation of DESCRIPTIONS, RESULTS, and, potentially, DATA and CLAIMS that relate to each other.* Other parts, such as theories or a description of the data collection, may also be part of a scientific work but are less relevant for reproductions. A prominent example is a journal article referencing publicly available open data. However, computational notebooks, blog posts, and other non-traditional research outputs can also qualify as scientific works.

Reproductions

Having established what the target of a reproduction is (the result of a computation) and how a particular computation is communicated (with one or multiple descriptions), it is now considered how a reproduction can reach its goal, that is, verifying that the computation of the result corresponds to its description.

[Fig. A4 → n_2] *Given one or multiple DESCRIPTIONS, all CHOICES LEFT UNSPECIFIED, and, potentially, DATA, one can infer a particular COMPUTATION.*

If this second (redone) computation gives a sufficiently similar result to the one reported in the original work, the verification is successful.

[Fig. A11 → n_9] A **reproduction** is a research activity with the goal of verifying that a COMPUTATION of RESULTS corresponds to its DESCRIPTIONS in the SCOPE. It entails (1) checking whether all DESCRIPTIONS in the SCOPE are COHERENT, (2) performing a COMPUTATION inferred from the DESCRIPTIONS in the DESCRIPTION SCOPE, using DATA in the DATA SCOPE (if applicable), and from UNSPECIFIED and CONFLICTING CHOICES, and (3) comparing the RESULTS. Optionally, reproductions can also assess (4) whether the original CLAIM is supported. It is possible that one reproduction verifies multiple computations.

[Fig. A11 → n_{10}] **Description scope:** *The set of DESCRIPTIONS that are used during a REPRODUCTION.* For example, a journal article might contain links to supplementary materials and source code. If the source code is re-executed, but its coherence with the article is not checked, the description scope solely consists of the source code.

[Fig. A7 → n_1] **Coherence of a description:** *None of the instructions in one or multiple DESCRIPTIONS conflict with each other.*

[Fig. A11 → n_4] **Data scope:** *Data sets commonly go through multiple preprocessing stages. The data scope describes which of these stages are considered during the reproduction. If all stages are considered this means one has obtained the PRIMARY DATA.*

[Fig. A8 → n_4] **Primary data:** *First digital representation of the RAW DATA (Gollwitzer et al., 2021).*

[Fig. A8 → n_5] **Raw data:** *Original records, that is the first non-volatile form of DATA.* In the case of electronic data collection, raw data and primary data coincide (Gollwitzer et al., 2021).

[Fig. A5 → n_3] **Unspecified choices:** *COMPUTATIONAL CHOICES that are not prescribed by the DESCRIPTIONS, filling gaps due to underspecification.* For example, if the original authors do not describe an algorithm's required parameters, these would be unspecified choices during the implementation concerning the source code. Similarly, if the version of a utilized software is not disclosed, that would be an unspecified choice during the execution concerning the computational environment.

[Fig. A5 → n_4] **Conflicting choices:** *COMPUTATIONAL CHOICES that are at odds with the DESCRIPTIONS.*

Of course, the results may simply differ because the unspecified choices varied between original work and reproduction. Also, reproducers may make choices conflicting with the descriptions out of necessity. For this reason, we recommend evaluating the consistency of results – rather than the identity – by exploring alternative unspecified choices and estimating the expected difference given any conflicting choices. Section 4 will go into more detail about that.

The broader the description scope of the reproduction, the more justified any conclusions drawn from the comparison become. For example, when the source code, the supplementary materials, or the preregistration are also considered, faults or their absence can be asserted with more confidence. Similarly, with a broader data scope, that is, fewer preprocessing steps, fewer faults can go unnoticed. In section 4 of this article, we will also consider other senses in which a reproduction is restricted to a certain scope worth reporting.

To summarize, a reproduction verifies whether a computation of results corresponds to its descriptions by comparing the reproduced result with the original results. A reproduction may also attempt to verify the results of multiple computations, possibly involving multiple claims. This way, a scientific work in its entirety may become the target of a reproduction. Notably, in our definition, a *reproduction* is the act of attempting a second computation after meeting the

preconditions (having available the necessary descriptions, data, and original results). Performing a reproduction does not imply that any reproduced results are actually obtained (because obstacles may prevent that) or how the compare with the original results. As a consequence, merely writing that “a work has been reproduced” is ambiguous regarding the outcome of the reproduction, which should always be mentioned in addition.

Differences Between Results

When comparing a reproduction’s result with the original result after performing the computation, one might observe a difference. According to Figure 1, this means that either the second computation deviates from the original computation (due to reproducers’ computational choices or an implementation fault) or there was a reporting fault.

When utilizing computers, there are many examples of minor deviations causing different results. Only changing the underlying hardware might already be enough to obtain a different result (e.g., see PyTorch Contributors, 2023; Stan Development Team, 2024), and in the case of hardware acceleration (GPU programming) and parallelization, even repeatedly using the same hardware may lead to a different result (Diethelm, 2012; Pham et al., 2020). Similarly, a difference may occur due to added debugging statements (Monniaux, 2008), a different operating system (e.g., Bhandari Neupane et al., 2019; Glatard et al., 2015; Scaria, 2018), different compilers (Hong et al., 2013), updated dependencies with fixed bugs, new features or different defaults (e.g., Van Den Bergh et al., 2023), different random seeds (Pham et al., 2020), or a different statistical software with new defaults (e.g., Herberich et al., 2010; Hodges et al., 2022; W.D., 2019).

Reasons for Differences

If the reproducers followed the descriptions of the original work, a different result raises the question of *why* the computations changed. In principle, we differentiate two broad reasons: Computations may have changed (1) due to the reproducers’ computational choices necessitated by obstacles or (2) because of mistakes, resulting in faults.

By obstacle, we mean everything that can prevent reproducers from obtaining a result (and that is not a fault). We discuss two common ones: underspecified descriptions (where the reproducers don’t know what to do) and grounds for conflicting choices (where the reproducers know what to do but cannot do it). They require that reproducers make additional choices, which may deviate from those of the original authors, leading to changing computations. Computational choices can manifest while implementing the source code and during execution in a particular environment, see (c) and (e) in Figure 1.

The uncertainty arising due to the possible existence of underspecification and faults has also been called “hidden uncertainty” (Auspurg & Brüderl, 2024, p. 3). In the following, we first focus on underspecified descriptions and conflicting choices before turning to faults.

Underspecified Descriptions

A first common obstacle and hence a possible reason for varying computations is underspecification, that is, the insufficient description of a computation. Whenever a part of the description seems underspecified to the reproducer in the sense that they can imagine

multiple appropriate ways forward, they cannot know which one was taken by the original authors. For example, articles published in journals that impose a word limit will not mention every aspect of the computation. In particular, procedures that are common in a field are not described in detail but are either referred to by a literature reference or are only mentioned by their name (e.g., “one-sided *t*-test” or “structural equation modeling”). If the reference or name is ambiguous (e.g., Weissgerber et al., 2018) or requires additional choices, the reproducers might have chosen to do something different than the original authors.

The methods section in journal articles is frequently found to contain less information than deemed necessary by the reproducers (e.g., Crüwell et al., 2023; Hardwicke et al., 2021; Seibold et al., 2021). Providing the source code of a computation can alleviate this problem (Ince et al., 2012), in the best case accompanied by information about the computational environment. However, studies frequently notice missing source code (e.g., Crüwell et al., 2023; Laurinavichyute et al., 2022; Sharma et al., 2024) or missing information about the computational environment (e.g., Herbert et al., 2021). Most requests directed at original authors to provide the source code remain unsuccessful (Krähmer et al., 2023). Of course, source code cannot be made available when using software that is exclusively controlled by a graphical user interface.

Underspecification might also be a deliberate, though misguided, decision by the original authors. We see two reasons for deliberate underspecification: First, leaving out details may be used as an abstraction device (Gervasi & Zowghi, 2010), signaling what is important about a computation.¹⁴ However, as this complicates further investigations, a recommended alternative is explicitly marking what is important rather than omitting what is not. Second, underspecification might be employed as strategic ambiguity (Frankenhuis et al., 2023), making it difficult to ascertain faults. Such ambiguities simultaneously decrease the worth of the scientific work to its readers, both in terms of falsifiability and as something to build on (Ince et al., 2012). For example, if the source code is withheld, it cannot be found to conflict with the methods section in an article. Then, different results might be (falsely) ascribed to the method’s stochasticity rather than faults. Also, a vague description generally allows for more conformant computations. Furthermore, such ambiguity raises the effort of reproductions, maybe preventing them in the first place.

Conflicting Choices

Sometimes, it is impossible to follow the description of a computation up to the last detail due to practical reasons. This is the case, for example, when reproducers do not have access to paid or discontinued software, lack sufficient computational resources, or do not have necessary domain-specific knowledge (e.g., how to use a particular programming language or where to find the relevant result among many lines of output). In these cases, either the reproducers do not proceed at all (e.g., Crüwell et al., 2023) or they deviate deliberately from the description, possibly causing a different result (e.g., Seibold et al., 2021).

¹⁴ For example, omitting the operating system and the specific hardware, because the result should be the same regardless.

Faults

Another possible reason for a differing result is a fault – either by the original authors or the reproducers themselves. In section 2, we differentiated faults by the phase of the research process the corresponding mistakes happen in, in particular implementation and reporting faults. What these two have in common is that they both cause the computation to be at odds with its descriptions or, by extension, with any data or results the descriptions refer to. This is the type of fault we consider in the following.

[Fig. A12 → n₂₄] *The COMPUTATION may not correspond to its DESCRIPTIONS.* For example, during the data analysis, the two experimental conditions might have gotten confused, effectively leading to the opposite claim.

Note that this does explicitly not cover faults that only manifest when applying the same code to other datasets, as the definition is dependent on the specific results and data mentioned in the descriptions. Also, from all the faults covered by this definition, there are four cases that we would like to specifically mention as they are probably not frequently thought of – for example, cases 2 through 4 are not covered by the heuristic guide from section 2. Reproducers might benefit from considering them as well if they try to identify the precise reason for different results.

[Fig. A12 → n₂₃] Fault Case 1: *The DESCRIPTIONS may not be COHERENT*, for example, if the source code contradicts the methods section of a journal article or the preregistration gives different instructions for the computation than the methods section.

[Fig. A12 → n₂₂] Fault Case 2: *The DATA REPORTED by authors may differ from the DATA UTILIZED to obtain the RESULT*, for example, because panel data in a repository were changed by their provider.

[Fig. A8 → n₃] **Reported data:** *DATA published alongside or referenced from a SCIENTIFIC WORK.*

[Fig. A8 → n₂] **Utilized data:** *DATA used during a COMPUTATION.*

[Fig. A12 → n₂₁] Fault Case 3: *The CODE REPORTED by authors may differ from the one that was EXECUTED to obtain the RESULT.* For example, even with version control, one might accidentally use a different version.

[Fig. A6 → n₄] **Reported code:** *CODE made available by its authors.*

[Fig. A6 → n₂] **Executed code:** *CODE run during a COMPUTATION.*

[Fig. A12 → n₂₀] Fault Case 4: *The REPORTED RESULT, referred to in the DESCRIPTIONS, may not correspond to the OBTAINED RESULT.* For example, for elaborate computations, the actual result has to be identified among many lines of output and subsequently transferred to the publication, which are two actions during which mistakes can happen.

[Fig. A10 → n₃] **Reported result:** *The RESULT given in an original WORK.*

[Fig. A10 → n₁] **Obtained result:** *The RESULT that was calculated during a COMPUTATION.*

Of course, researchers can make more kinds of mistakes than are covered here.

Notably, mistakes according to our definition do not extend to conceptual mistakes. Imagine that a researcher intends to analyze data from two independent groups with a dependent t -test. This is conceptually wrong and will not lead to meaningful results. But if the code correctly implements the (wrong) intention, according to our definition it is *not* a mistake in the context of reproductions, which aim to verify a computation.

Nonetheless, reproducers are encouraged to incorporate their concerns about substantive flaws in the computation or aspects of the study design and data collection into their appraisal whether the claim is still supported. Finally, they can also evaluate the rule of inference employed in the original work.

[Fig. A11 → n₅] **Rule of inference:** *Reasoning behind making a CLAIM based on a RESULT (or multiple RESULTS).*

The central importance of source code for reproductions

To summarize one of the key points up until now, scripted computations are in a unique position to facilitate a fruitful reproduction due to their dual nature: They are readable by humans and they are used to control computers, thus providing an accurate (though not complete¹⁵) record of a computation at a certain point in time. This sets them apart, for example, from statistical software with only graphical user interfaces, where a description of the performed manual steps is at a higher risk of being insufficient or inaccurate.

Checking the compatibility of source code with the (rest of the) descriptions in the scope is, therefore, a key activity that can be performed during a reproduction. One can discern three cases:

- The source code may clear up otherwise underspecified aspects of the descriptions.
- The source code may be partially (or completely) missing. In this case, it needs to be re-implemented.
- The source code may agree with or disagree with what is stated in the other descriptions. The latter represents a fault, and it needs to be resolved in order to obtain any result during the reproduction. Usually, statements from the preregistration take precedence over the implemented code.

With the definitions introduced in section 3, the difference between a re-execution reproduction and a re-implementation reproduction can be formulated precisely. First, they typically differ in their scope – the former usually considers the source code by inheriting (i.e., re-executing) it, while the latter usually does not consider the source code at all. Second, and as a consequence, the amount of unspecified choices is expected to be more pronounced for a re-implementation reproduction. However, because there are also atypical cases, merely stating that a re-execution or a re-implementation reproduction has been performed is generally not sufficient. For example, a re-implementation reproduction may

¹⁵ The execution of the source code requires human intervention, such as for providing the necessary hardware and software, compiling the source code, and invoking the program with correct parameters.

base its re-implementation on the original source code, or a re-execution reproduction might also compare the original source code to the methods description. Therefore we give specific recommendations in section 4 of this article on how to report a reproduction.

Box 2

Summary of Formal Definitions

In the preceding definitions, we introduced the *results* of a *computation* as the target of a reproduction. By *descriptions*, we refer to anything that characterizes a computation, such as parts of a methods section in a journal article or source code. Descriptions, together with results and, potentially, data and claims, are part of *scientific works*. During a *reproduction*, the descriptions in the scope are compared for coherence, another computation is performed based on them, and the results are compared for consistency. Underspecification, conflicting choices, and faults are all reasons for reproductions to produce different results than the original computation. Human actions that cause a computation to be at odds with its description, either during implementation or during reporting, are *mistakes* manifesting as *faults*. They can happen before and during the computation as well as while reporting the results. Four cases of faults are emphasized: (1) incoherent descriptions, (2) difference between the utilized and the reported data, (3) difference between the executed and the reported code, and (4) mismatch between obtained and reported results. Source code is of special importance for reproductions, as it is readable by humans and can be used to control computers.

4. Performing a Reproduction

In the following, building on the definitions, we sketch the reproduction procedure to verify that a computation of results corresponds to its descriptions in the scope. To recap (see Fig. A3 \rightarrow n₁), this means that the results obtained by following the instructions, potentially applied to the provided data in the specified computational environment, are sufficiently similar to the reported results. It, therefore, consists of (a) defining the scope of the reproduction in terms of the involved works, claims, results, data, and descriptions, (b) ensuring the availability of descriptions, results, and, if applicable, data, (c) obtaining results, potentially by resolving obstacles, (d) evaluating the results' consistency, potentially by assessing the expected differences in results, and (e) evaluating whether the claims are supported given the information acquired during the reproduction. We conclude with remarks on reporting the outcome of a reproduction.

(a) Defining the Scope

First, reproducers have to choose which scientific works, claims, results, descriptions, and, if applicable, data are part of the reproduction's scope:

- Works: Which scientific works are attempted to be reproduced? Reproductions can focus on individual or multiple works.
- Claims: Which claims within a work are being evaluated? Reproducers can focus only on a subset of all claims presented in a scientific work, for example, those that

appear in an abstract. Or disregard all claims and only focus on results if no claims have been made.

- **Results:** Which results are being evaluated as the reproduction's target? For example, this can be a particular figure, a table, or a set of numbers. There are many degrees of freedom for reproductions when selecting the evaluated results. One possibility is to focus on all results that (supposedly) support the claims chosen before.
- **Obtained data:** This only applies to reproductions of empirical studies. Which preprocessing steps have already been performed for the data obtained for this reproduction? In other words, how do the obtained data differ from the raw measurements recorded during data collection? In the best case, reproductions obtain primary data, but this is not always possible. Then, faults in the processing up to the point where the data are obtained are more difficult to identify.
- **Involved descriptions:** Reproducers may only focus on a subset of the available descriptions, whose coherence is checked and which informs their computation. The smaller the description scope, the fewer faults can be detected.
- **Inherited source code:** Of the descriptions involved, is any source code used for re-execution? Is it used as is, or is it being modified or extended?
- **Applied methods standards:** Which methods standards are followed when making unspecified choices, checking the coherence, estimating expected differences, and evaluating the appropriateness of specified choices? This will likely get more detailed and can be elaborated on over the course of the reproduction.

For reproducers performing mass reproductions of multiple scientific works, rather than evaluating all results, they can evaluate those that support claims that are mentioned in the title, the abstract, or the first table or figure, following Hardwicke et al. (2018) and Alipourfard et al. (2021).

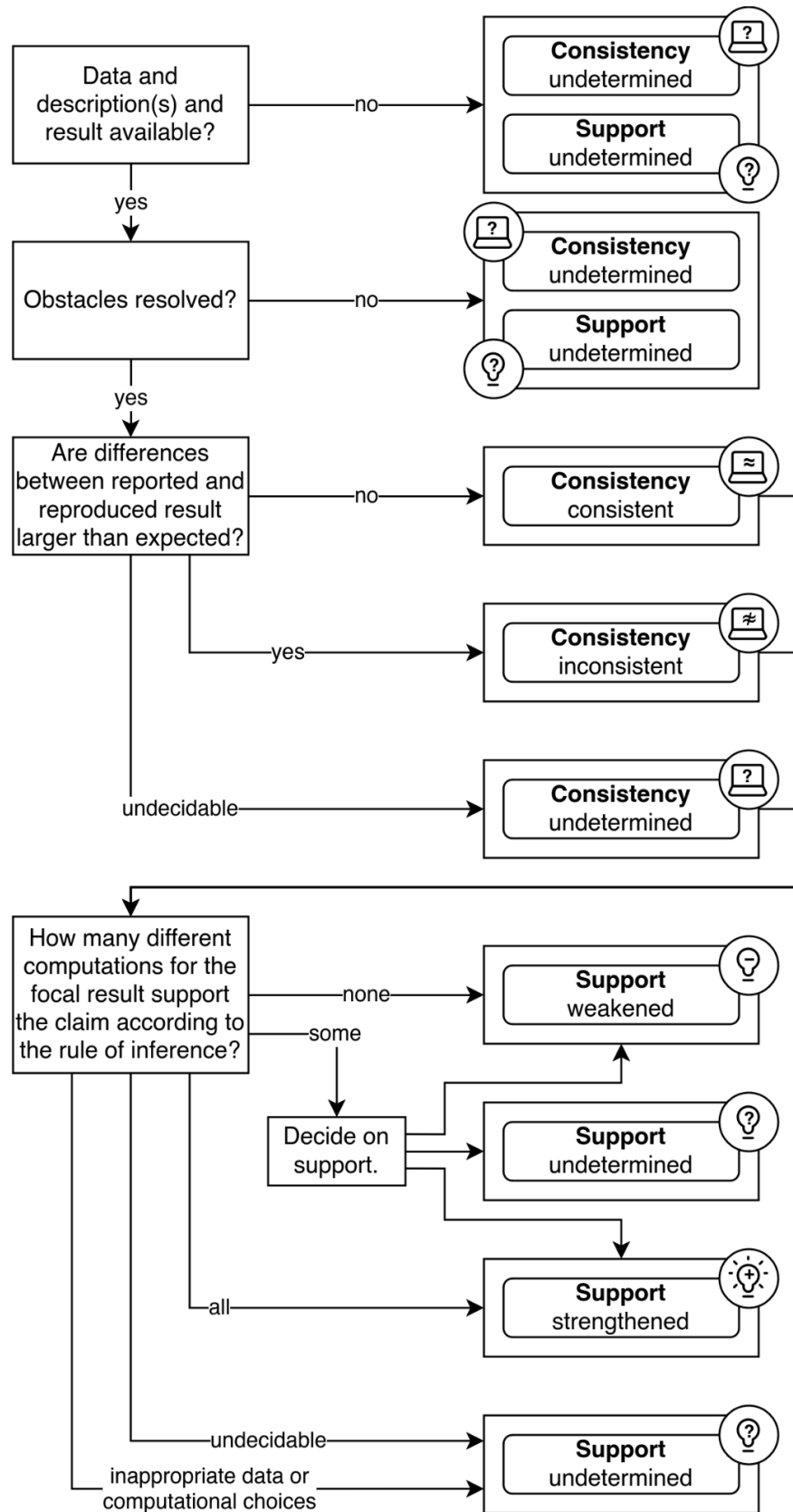
(b) Checking the Preconditions

If the descriptions or – in the case of an empirical study – the data in the scope are not available or the original results cannot be identified, the reproduction cannot be conducted. The original authors or any other relevant stakeholders should be contacted at this point to provide what is missing or the scope needs to be adjusted to what is available. Otherwise, the reproduction outcome for the affected results and claims is *undetermined* due to failing a precondition (e.g., lack of data).

From this step on, readers can compare the procedure with Figure 2. It depicts a flowchart for performing a reproduction in the case of one result and one claim. A more detailed version is available in the Appendix as Figure A1.

Figure 2

Summary Flowchart for Performing the Reproduction of One Result



(c) Obtaining Results

After ensuring the preconditions are met, reproducers need to be able to compute results for comparison. This requires them to understand the descriptions in the scope and, potentially, the reported data and to verify the coherence of the descriptions. Also, the reproducers must have the necessary resources to perform the computation, for example, in terms of knowledge, time, software, and hardware. Finally, they need to be able to perform the comparison, which includes identifying the results and knowing how to compare them. All of that can be obstructed by obstacles, which require reproducers to make a computational choice that is not evident from or contradicts the descriptions. Reproducers frequently encounter obstacles but may not acknowledge them as such, for example, when they use a different computer during execution than the original authors.

If reproducers identify faults, such as incoherent descriptions, they need to correct them before comparing the results. When reproducers encounter obstacles, in many instances a helpful strategy is to contact the original authors and ask for clarification. However, this may be impossible due to limited reachability and time constraints, or it might not help resolve the obstacle. In that case, another strategy is to start by making the most likely choice according to chosen standards in order to (initially) obtain any results. For example, if it is not clear whether a psychological study has performed a two-sample *t*-test for equal or unequal variances, psychologists might want to start with the latter (Delacre et al., 2017). However, choosing among multiple options might not be possible if the reproducers are limited by their resources. Hence, as a third strategy, reproducers might need to deviate deliberately from the descriptions, for example, by simplifying computations or using a different software. Finally, if the obstacles cannot be resolved, the reproduction's outcome concerning the affected results and claims must be regarded as *undetermined* by stating the respective obstacle.

(d) Evaluating the Consistency of Results

Once results have been computed, reproducers can evaluate their consistency with the originally reported results. For this, each original result is compared with the respective reproduced result. While LeBel et al. (2018), following Hardwicke et al. (2018), generally consider a result consistent if it matches within a 10% margin of error, we propose an approach that is aware of the obstacles faced during the reproduction and allows attributing differences to faults with more confidence, relative to a set of methods standards.

In the easiest case, the original and the reproduced result can be deemed consistent because they match exactly (i.e., at the maximum precision their representation allows¹⁶). In some instances, such an exact match can be obtained by using information provided by the original authors. For example, if asked about this, the original authors might argue that the reproducers misunderstood a description and provide them with a way of obtaining exactly the same result again, in which case they may regard it as *consistent*. On the other hand, the original authors might confirm the existence of a fault from which an inconsistency of a result may follow. Technically, in this case, the reproducers still need to eliminate other possible (and potentially more impactful) reasons for the difference, as detailed below.

¹⁶ For example, if a study reports a numerical result with four digits, all four digits need to match after the reproduced result has been rounded to the same precision.

If the results do not match exactly, it is still reasonable to consider them consistent if their difference can occur given how any obstacles were resolved by the reproducers. For example, if the difference between results can be explained solely by the fact that a different computer has been used (and it is not too large), most people would still consider them consistent. Then, it is up to the readers to decide whether the change between the two computations (here, a different computer) is critical to them. Therefore, reproducers can investigate whether the results differ no more than expected depending on how any obstacles were resolved. In the following, we describe two strategies for deciding on the consistency. It is recommended to try the first strategy if the amount of possible computations is finite and the second strategy if it is infinite.

The first strategy is that any choice to resolve underspecification can be **varied** to another appropriate option **until an exact match is obtained**. Remaining faults of the type discussed above (Fig. A12 \rightarrow n_{24}) can now be considered less likely and the results are deemed *consistent*. For example, imagine that the description of the original authors only mentioned a “*t*-test”. If the reproducers initially calculated a two-sample *t*-test for unequal variances and obtained a different *p*-value, a test assuming equal variances can be tried next – both analytical pathways are consistent with the underspecified description. If the reproduced *p*-value now matches the original one, the original and the (second) reproduced result can be regarded as *consistent* and faults can be considered less likely. If no match is found despite trying all appropriate choices, the reproduced and the original result can be regarded as *inconsistent*. Of course, this evaluation is highly dependent on the chosen methods standards.

However, there may be instances where it is impossible to try all possible (appropriate) combinations of choices. Also, sometimes the instructions cannot be followed, leading to conflicting choices. Therefore, a second strategy is to **estimate the impact of deviations** in terms of differences between results and determine whether the original and the reproduced result differ more than expected. For example, if the computing architecture is different – likely if a different computer is being used – reproducers can make an educated guess concerning the tolerable differences due to floating point arithmetic. Or, if no random seeds are reported, they can perform multiple computations with multiple random seeds and check whether the original result stands out from the reproduced results (compare Rahal, 2024). Empirical norm values for such standard cases are rarely available, for example, how much of a difference can be expected by changing a software package (for exceptions, see Jen et al., 2023; Letter, 2021; see also Keeling & Pavur, 2007; McCoach et al., 2018; Hodges et al., 2022). Again, the reproduced results can be regarded as *inconsistent* or *consistent* with the original result, depending on whether they differ more than expected or not (even if no exact match was obtained). Again, this evaluation crucially depends on the chosen methods standards.

If no strategy is feasible to decide on the consistency, it remains unclear whether the observed difference between reproduced and original result can be attributed to a fault or simply to a choice in the step of resolving an obstacle. In this case, despite having obtained a result, the consistency can be declared *undetermined*, stating all relevant obstacles.

Box 3

Possible Outcomes for the Consistency of Results

- **Consistent.** For example:
 - identical results
 - identical results after using information provided by the authors
 - identical results after trying other appropriate options
 - actual difference of results not larger than expected
- **Inconsistent.** For example:
 - out of options after getting non-identical results from all appropriate computations
 - actual difference of results larger than expected
- **Undetermined.** For example:
 - failed precondition (i.e., missing data, description, or result)
 - unresolved obstacles
 - estimation of expected difference is infeasible

Until this point, the reproducers made a separate consistency evaluation for every original result selected for the reproduction. Per original result, the reproducers either did not obtain a reproduced result (due to failed preconditions or unresolved obstacles), they obtained an exact match, or no exact match was obtained, and (if possible) they assessed which difference between the results is to be expected given the obstacles they resolved and the methods standards they chose. If the results are inconsistent or the consistency is undetermined, a reproduction can also explore why. This is an iterative process where the computation is modified until the same (faulty) result is obtained. It is most likely to succeed if the original source code is provided.

One can now revisit Table 1, which also displays the state of knowledge implied by evaluating the consistency of results. The example from section 2 of this article concluded with implementation choices and implementation mistakes as potential reasons for a different result. By evaluating whether the result of the re-implementation reproduction is *consistent* (rather than identical), one answers the question of whether the difference is possible given how obstacles were resolved by the reproducers and depending on the chosen methods standards. Suppose the results are found to be inconsistent. Then, one can say that the most important reason for differing results is an implementation mistake¹⁷ (indicated by a large disk), and any potential deviations due to choices concerning the implementation (or the environment) are negligible in comparison (indicated by a circle around a small disk). If, in contrast, the results are deemed consistent, one can conclude that implementation choices (and possibly environment choices) are the most important reason for differences, and any potential implementation mistakes (and reporting mistakes) probably are negligible in comparison – again, contingent upon the applied methods standards.

Table 1 also includes an example of evaluating the consistency of a direct replication's results. In addition to computational choices, the possible effects of sampling variability need

¹⁷ to the extent appropriate choices were covered by the methods standards

to be considered here, and the results are inconsistent only if both cannot plausibly explain their difference. Evidently, direct replications do have a verifying function of their own, as consistent results change the belief about reporting and implementation faults, which are

the two main threats to internal consistency discussed in this article. While their absence is not guaranteed, their consequences are small in comparison to computational choices and sampling variability. Inconsistent results, however, can have many possible reasons and a lack of internal consistency is only one of multiple explanations.

(e) Evaluating the Support for the Claim

If the reproducers have selected claims for evaluation, it is also of interest whether these are still supported, given the information acquired during the reproduction. Consistent results are neither a necessary nor a sufficient condition for a claim to still be supported. Evaluating the support is optional, although it is recommended. We will start discussing the simple case where a claim is backed by *one* result in the original work. Then, it is sufficient to evaluate the rule of inference for this one result, along with the data and specified choices that led to it.

First, one starts with the data. If the reproduction involved (empirical) **data**, the reproducers might find it inappropriate to support a claim. For example, the sampled population might be considered unfit to make a particular claim, the operationalization of a construct could be regarded as unsuccessful, or the reproducers might argue that the study design does not allow for causal claims. Also, the reproducers might not know whether the data are appropriate for the result to support the claim. In all of these cases, the reproducers can conclude that the support for the claim is *undetermined*, given the data. Otherwise, if the reproducers have no concerns about the data, the evaluation of the claim just continues.

Second, the reproducers might have issues with the original author's **specified choices** regarding a claim. For example, the reproducers might find that the utilized statistical methods deviate from best practices in the respective field or that inadequate preprocessing calls the claim into question. Also, the reproducers might not know whether the specified choices are appropriate for the result to support the claim. Consequently, in these cases, the support for the claim might be regarded as *undetermined* due to an inadequate computation. If the reproducers have followed up with an appropriate computation, they can continue evaluating the claim. In contrast, if the reproducers do not find the computation to invalidate the claim, they can continue with its evaluation.

Finally, it is necessary to check whether a claim is still supported given the **rule of inference** used in the original work. If this is undecidable because the rule of inference is unknown or unfit, reproducers can choose an alternative rule of inference. Otherwise, the claim's support has to be regarded as *undetermined*. The reproducers might have performed multiple computations per original (focal) result due to obstacles and hence obtained multiple results. In that case, all reproduced results for that particular original result need to be evaluated. If all of them support a claim or do not support it, its support can directly be regarded as *strengthened* or *weakened*, respectively. For example, if multiple results have been calculated until an exact match has been obtained (because of underspecification) and all of them support the claim, the support of the corresponding original result for the claim can be considered *strengthened*. In another example, suppose the reproducers have estimated an

expected difference. If even the most extreme results still within this difference support the claim, it can directly be considered *strengthened*. If only some reproduced results support the claim, no general rule exists and one has to decide on the strength of the support on a case-by-case basis. This is similar to evaluating the results of a robustness check or a multiverse analysis, although it is less systematic because one usually stops exploring computations after obtaining one consistent result.¹⁸ Of course, one may also conclude that the support for the claim is *undetermined*.

As mentioned earlier, a claim supposedly supported by a result that the reproducers could not compute at all (due to failed preconditions or unresolved obstacles) must be regarded as having *undetermined* support by stating the reason why no result was obtained. Moreover, even an exact match between the original and the reproduced result does not guarantee support for the claim: The original authors might have employed an inappropriate rule of inference or wrongly applied the rule. For example, the reproducers might write the following: “Although we could exactly reproduce the originally reported numerical result, the support for the claim is weakened because the employed rule of inference was applied incorrectly: In the methods section and preregistration, the authors announced to correct for multiple testing but actually did not do it. When the α level is correctly reduced, the result is not significant anymore.”

However, if a claim is backed by multiple results in the original work, all of them need to be considered. Because the rule of inference can be an arbitrary function that maps the results onto a claim, it might not be possible to consider the results individually in their support for the claim. Of course, such an “holistic” assessment of the results may be performed implicitly, making its criteria unknown to the reproducers. The case of multiple results is not covered by Figures 2 and A1.

As an example of multiple results, significant improvements in speed *and* accuracy might support the claim that an experimental condition improves performance. Consequently, the reproducers need to check that both speed *and* accuracy indeed improve significantly. In this example, if only one of the reproduced results shows a significant improvement, the support for the claim would be considered *undetermined*, and if neither shows significant improvements, the support for the claim would be considered *weakened*. In general, this depends on the logical function that maps the results onto claims.

Box 4

Possible Outcomes for the Support for a Claim by One Result

- **Strengthened.** For example:
 - all computations for the same (focal) result support the claim
 - some computations for the same (focal) result do not support the claim but the strength of the support is still sufficient
- **Weakened.** For example:
 - no computation for the same (focal) result supports the claim
 - some computations for the same (focal) result do support the claim but the

¹⁸ Of course, a traditional robustness check or multiverse analysis does not focus specifically on unspecified choices, but rather plausible alternatives to specified choices.

strength of the support is insufficient

- **Undetermined.** For example:
 - inappropriate data
 - inappropriate specified choices and no alternative conducted
 - obstacles not resolved
 - some computations for the same (focal) result do support the claim and the strength of the support cannot be determined
 - rule of inference undecidable and no alternative employed

(f) Reporting the Outcome

Reporting a reproduction should be more nuanced than just announcing its success or failure for at least two reasons. First, although reproductions should give very similar results, reproducers may underestimate the expected difference of results. Second, there is considerable variability in the activities that all counts as reproductions and similarly in the conclusions that can be drawn from them. Therefore, summaries of reproductions should (1) use careful language, (2) also evaluate the substantive claim, and (3) be transparent about their limitations.

We recommend a reproduction separately report the consistency of the results and the support for the claims. Consistent results mean that the original computation has largely been performed as described, as the most important reason for differing results are resolved obstacles. Faults may exist, but their impact on the results is negligible in comparison. Inconsistent results might have multiple possible reasons:

- The methods standards chosen by the reproducers did not cover the computational choice by the original authors.
- Aspects of the descriptions were meant differently by the original authors than interpreted by the reproducers.
- Faults in the original work or the reproduction.

Note that for the consistency evaluation, the original result is compared with the (reproduced) result of a computation that only includes corrections due to faults. When evaluating the support for the claim, however, one uses the (reproduced) result of a computation that also includes other corrections that reproducers deem necessary to decide on the support (such as changing the computation or the rule of inference, because the one originally proposed did not follow current best practice).

The evaluations of consistency of results and support for the claims critically depend on the reproduction's scope. For example:

- Interpreting the outcomes from a mass reproduction of multiple works depends on their inclusion criteria.
- Interpreting the outcome of a reproduction of a single work depends on which results and claims have been selected as targets.
- Interpreting the consistency evaluation of a result depends on the data scope, which descriptions were involved, the way any obstacles were resolved, and how the

expected difference was estimated. In turn, this depends on the methods standards applied by the reproducers.

- Interpreting the support for the claim depends on the results that were evaluated as potential evidence and on the methods standards that were used to judge the appropriateness of specified choices.

Therefore, we provide a set of recommended pieces of information that every reproduction should report, summarized in Box 5 and elaborated in Figure A2. Besides what has already been mentioned as part of the scope this also includes the reproducers' expertise, as differentiated by Garijo et al. (2013).

Box 5

What to Report in a Reproduction

- Reproducers' expertise
 - ☐ Author: The authors of the original work are among the reproducers.
 - ☐ Expert: The reproducers are familiar with the field and the methods.
 - ☐ Novice: The reproducers have basic knowledge about the field and the methods.
 - ☐ Minimal: The reproducers have little knowledge about the field and the methods.
- Scope
 - Works: Report any permanent identifiers of the investigated works.
 - Claims: Report which claims, if any, are being investigated.
 - Results: Report which results are being evaluated (mention page or table where the result can be found).
 - Data: To what extent has the obtained data been preprocessed?
 - Descriptions: Which descriptions are checked for coherence and inform the computation (select all that apply)?
 - ☐ Preregistration
 - ☐ Methods section
 - ☐ Supplemental material
 - ☐ Source code
 - ☐ Other (state them)
 - Source code: Which source code is inherited, that is, used for re-execution?
 - Methods standards: Which methods standards are followed when evaluating computational choices and deviations? Which unspecified choices are explored and how is the expected difference estimated?
- Preconditions (select all that apply):
 - ☐ At least one description was available
 - ☐ (for empirical work) A data set was available
 - ☐ The original results for comparison were available
- Obstacles: Were any obstacles encountered, like underspecification or grounds for conflicting choices and (how) were they resolved?
- Faults: Were any faults identified?
- Outcome for every evaluated result

- Reproduced result (with potential faults as covered by Fig. A12 \rightarrow n_{24} corrected)
 - ○ Consistent
 - ○ Inconsistent
 - ○ Undetermined (give reason)
- Outcome for every evaluated claim
 - Reproduced result (with all potential corrections)
 - ○ Support strengthened
 - ○ Support weakened
 - ○ Undetermined – Why?
- About the reproduction process
 - Were the original authors contacted (e.g., for clarification) and did they respond?
 - Are there any suggested improvements (e.g., better documentation, initial sharing of data)?

5. Discussion and Recommendations

The existing literature on redoing activities – scientific activities that redo some phases of a previous study – recommends differentiating between the act of enabling, the act of redoing, obtaining the same result, and reaching the same conclusion (Goodman et al., 2016; Ulpts & Schneider, 2024). Reproductions, which we define as redoing activities that work with the original data and aim to keep the computation (e.g., a data analysis) as similar as possible, have received little systematic attention. This article explicates their logic and provides a conceptual framework for describing their different flavors and understanding their epistemic function, that is, what one learns from their conduct.

In section 2, we presented a heuristic guide that details the epistemic function of reproductions in comparison to direct replications with regards to verification by discerning which redoing activity can provide evidence for or against which types of mistakes. We introduced implementation and reporting mistakes and explained that a typical re-execution reproduction cannot detect the former, though it can be combined with a re-implementation reproduction to provide evidence against reporting mistakes. The verifying function of redoing activities is their ability to detect mistakes that threaten the internal consistency of the computational aspects of a scientific work. One way of detection is by comparing the results for identity or consistency. As direct replications cover more types of mistakes, attributing differences in results to any one of them is more difficult.

Caution must be exercised, however, as the research process depicted in Figure 1 is simplified and the derived guide is dependent on certain assumptions, including linear progress through the various phases and the particular points where mistakes and choices by the reproducers can influence results. Also, the conclusions drawn from Table 1 require that data and code were indeed used by the original authors as reported (i.e., fault cases 2 and 3 are not present). Finally, one could also question the asymmetry with which this guide treats inherited phases as set. As a consequence, choices by the original authors during these phases are not considered as reasons for different results. For example, if missing values during data entry were recorded with "-999" and the original code correctly removes

these values, whereas the re-implemented code does not (expecting "NA"), according to this guide a different result would be attributed to a mistake by the reproducers whereas common sense would rather attribute this to the original authors' unconventional data entry choice.

In section 3, we sketched the logic reproducers can follow to answer the question whether a computation corresponds to its descriptions (e.g., methods section or source code) and, by extension, to any referred data or results – or has a fault otherwise. A reproduction involves checking the coherence of the descriptions involved, performing another computation, and comparing the results. Naturally, it requires the availability of *some* descriptions, results, and – in the case of an empirical study – data. At any point, reproducers may see no, exactly one, or multiple paths forward, depending on their interpretation of the descriptions – guided by the chosen methods standards – and any obstacles they encounter, such as underspecification and grounds for conflicting choices.

If the reproducers obtain a result, it may differ from the original result due to varying computational choices or because of faults. However, because a simple comparison for identity cannot discern them, we recommend evaluating the *consistency* by exploring the impact of computational choices on results before suspecting faults as a last resort. Results are deemed *consistent* if they differ no more than can be expected given how any obstacles were resolved, and *inconsistent* otherwise. For example, results can be demanded to only match up to a certain precision, given that a different optimizer has been used. Importantly, both consistent and inconsistent results are contingent on the methods standards the reproducers chose. The consistency can also be *undetermined* if no result was obtained or if it is infeasible to estimate the expected difference.

While the consistency is threatened only by faults that cause the computation to be at odds with its description, a reproduction can separately evaluate the support for any claims connected to the results, thus also considering the appropriateness of the data, the specified choices, and the rule of inference. The support for claims can be considered *strengthened*, it can be *undetermined*, or in some instances it may even be deemed *weakened*. Finally, reproducers should report the scope of their reproduction by detailing which descriptions were involved, whether source code was inherited, and to what extent any data were already preprocessed.

We demonstrated how underspecification reduces the epistemic value of studies. For example, if the data analysis in a study is underspecified, and a reproduction obtains results that are inconsistent with the original article, more uncertainty remains about the reason for the inconsistency: Either the reproducers made different implementation choices, or a mistake has happened. For researchers aiming to facilitate verification and allow informative reproductions, this might serve as an additional incentive to be as transparent as possible when communicating the results of one's research.

We discerned four special cases of faults, which can guide reproducers on where to look for the source of differences: (1) incoherent descriptions, (2) different data, (3) different source code, and (4) incorrect reporting of results. However, even for consistent results faults may be left if the difference they introduce is less than the difference caused by obstacles such as underspecification.

The approach described here is subject to some limitations. First, the consistency evaluation is heavily dependent on the chosen methods standards and how the expected difference is estimated. Second, a consistent result from a re-implementation reproduction does not guarantee that the implementations are free of any type of fault. This is because there can be faults that just don't manifest for the collected data or cancel each other out. Third, even identical results suggest but do not guarantee that the original computation has been performed as described. For example, if the domain of possible results is constrained, the original computation might contain a fault, but the reproducers might have gotten the same result through a different approach. The reproducers might even have made a mistake on their own, possibly the same as the original authors. Fourth, we treat the consistency of results as a binary variable which is only an approximation of a continuous reality. Fifth, in many instances we currently lack empirical norm values for standard cases of deviations and the resulting differences. Sixth, we do not discern between mistakes and fraud, that is, instances where the descriptions were written in a way to deliberately deviate from the actual computation conducted.

Taken together, we provide a conceptual framework to describe computational reproductions and their epistemic function of verification in detail. Revisiting the challenges described in section 1, the procedure we recommend addresses them in several ways: (1) First, by **providing standards**, common situations during reproductions can be dealt with in a consistent manner. For example, if the source code or the data have not been shared there is a defined way of dealing with it (i.e., changing the scope or reporting an undetermined outcome due to failed preconditions). If the reproduced results differ from the original results the determination of the outcome is not arbitrary. And assistance by the authors is welcome, but the proposed definitions do not depend on their availability. (2) We believe **checking the coherence of descriptions** is integral to reproductions and by treating source code as a regular description, we acknowledge that oftentimes it is not executable anymore but it can still provide valuable insight into the original computational choices. (3) By **scoping the outcome of a reproduction**, we are transparent about its limitations. Typical re-execution and re-implementation reproductions serve different purposes and the description scope makes that known. Similarly, a reproduction working with preprocessed data has its worth, if one keeps in mind that only faults after the preprocessing can be identified by them. Lastly, by being explicit about which results are part of the scope, it becomes obvious that a scientific work is typically not reproduced in whole, but that a reproduction always makes a statement about a particular result.

Finally, with the conceptual framework in mind one can make some recommendations for reproducers and original authors. If possible, reproductions should use the primary data, that is, the first digital representation of the original records, so fewer faults can go unnoticed. In addition, ideally all descriptions available to the reproducers should become part of the scope, thus checked for coherence with each other and informing the computation. Every deviation from this default should be specifically indicated when stating the scope. To ease the work of reproducers, original authors should include a “computation summary” in their appendix, detailing (1) the claims they make, (2) which results support these claims, (3) the rules of inference connecting claims and results (i.e., when is a claim considered strengthened or weakened), and (4) which are the most important claims and results.

We hope that our procedure for conducting and reporting reproductions helps reproducers to make the relevant choices, and also to achieve some standardization in the reporting of the

outcomes of reproductions. Such a common language would have the potential to improve and speed up meta-scientific research on the reproducibility of science and facilitate knowledge formation in the individual disciplines.

6. References

- Alipourfard, N., Arendt, B., Benjamin, D. M., Benkler, N., Bishop, M. M., Burstein, M., Bush, M., Caverlee, J., Chen, Y., Clark, C., Dreber, A., Errington, T. M., Fidler, F., Field, S., Fox, N. W., Frank, A., Fraser, H., Friedman, S., Gelman, B., ... Wu, J. (2021). *Systematizing Confidence in Open Research and Evidence (SCORE)*. SocArXiv. <https://doi.org/10.31235/osf.io/46mnb>
- Ankel-Peters, J., Brodeur, A., Dreber, A., Johannesson, M., Neubauer, F., & Rose, J. (2024). *A Protocol for Structured Robustness Reproductions and Replicability Assessments* (I4R Discussion Paper Series No. 143). Institute for Replication (I4R). <https://hdl.handle.net/10419/301917>
- Arguillas, F., Christian, T.-M., Gooch, M., Honeyman, T., Peer, L., & CURE-FAIR WG. (2022, July). *10 Things for Curating Reproducible and FAIR Research* (Version 1.1). Zenodo. <https://doi.org/10.15497/RDA00074>
- Artner, R., Verliefe, T., Steegen, S., Gomes, S., Traets, F., Tuerlinckx, F., & Vanpaemel, W. (2021). The reproducibility of statistical results in psychological research: An investigation using unpublished raw data. *Psychological Methods*, 26(5), 527–546. <https://doi.org/10.1037/met0000365>
- Auspurg, K., & Brüderl, J. (2024). Toward a more credible assessment of the credibility of science by many-analyst studies. *Proceedings of the National Academy of Sciences*, 121(38), e2404035121. <https://doi.org/10.1073/pnas.2404035121>
- Barba, L. A. (2018). *Terminologies for Reproducible Research* (Version 1). arXiv. <https://doi.org/10.48550/ARXIV.1802.03311>
- Bayarri, M. J., & Mayoral, A. M. (2002). Bayesian Design of “Successful” Replications. *The American Statistician*, 56(3), 207–214. <https://doi.org/10.1198/000313002155>
- Berkeley Initiative for Transparency in the Social Sciences. (2020). *Guide for Advancing*

- Computational Reproducibility in the Social Sciences*. <https://bitss.github.io/ACRE/>
- Bhandari Neupane, J., Neupane, R. P., Luo, Y., Yoshida, W. Y., Sun, R., & Williams, P. G. (2019). Characterization of Leptazolines A–D, Polar Oxazolines from the Cyanobacterium *Leptolyngbya* sp., Reveals a Glitch with the “Willoughby–Hoye” Scripts for Calculating NMR Chemical Shifts. *Organic Letters*, 21(20), 8449–8453. <https://doi.org/10.1021/acs.orglett.9b03216>
- Botvinik-Nezer, R., Holzmeister, F., Camerer, C. F., Dreber, A., Huber, J., Johannesson, M., Kirchler, M., Iwanir, R., Mumford, J. A., Adcock, R. A., Avesani, P., Baczkowski, B. M., Bajracharya, A., Bakst, L., Ball, S., Barilari, M., Bault, N., Beaton, D., Beitner, J., ... Schonberg, T. (2020). Variability in the analysis of a single neuroimaging dataset by many teams. *Nature*, 582(7810), 84–88. <https://doi.org/10.1038/s41586-020-2314-9>
- Breznau, N., Rinke, E. M., Wuttke, A., Nguyen, H. H. V., Adem, M., Adriaans, J., Alvarez-Benjumea, A., Andersen, H. K., Auer, D., Azevedo, F., Bahnsen, O., Balzer, D., Bauer, G., Bauer, P. C., Baumann, M., Baute, S., Benoit, V., Bernauer, J., Berning, C., ... Żółtak, T. (2022). Observing many researchers using the same data and hypothesis reveals a hidden universe of uncertainty. *Proceedings of the National Academy of Sciences*, 119(44), e2203150119. <https://doi.org/10.1073/pnas.2203150119>
- Carter, E. C., Schönbrodt, F. D., Gervais, W. M., & Hilgard, J. (2019). Correcting for Bias in Psychology: A Comparison of Meta-Analytic Methods. *Advances in Methods and Practices in Psychological Science*, 2(2), 115–144. <https://doi.org/10.1177/2515245919847196>
- Chatfield, C. (1995). Model Uncertainty, Data Mining and Statistical Inference. *Journal of the Royal Statistical Society. Series A (Statistics in Society)*, 158(3), 419. <https://doi.org/10.2307/2983440>
- Clemens, M. A. (2017). The meaning of failed replications: A review and proposal. *Journal of Economic Surveys*, 31(1), 326–342. <https://doi.org/10.1111/joes.12139>

- Crüwell, S., Apthorp, D., Baker, B. J., Colling, L., Elson, M., Geiger, S. J., Lobentanzer, S., Monéger, J., Patterson, A., Schwarzkopf, D. S., Zaneva, M., & Brown, N. J. L. (2023). What's in a Badge? A Computational Reproducibility Investigation of the Open Data Badge Policy in One Issue of Psychological Science. *Psychological Science*, 34(4), 512–522. <https://doi.org/10.1177/09567976221140828>
- de la Guardia, F. H., Seung, Y. S., Brodeur, A., Miguel, E., & Vilhuber, L. (2024). *Standardizing and Crowd-sourcing Analysis to Assess Reproducibility in Economics*.
- Delacre, M., Lakens, D., & Leys, C. (2017). Why Psychologists Should by Default Use Welch's t-test Instead of Student's t-test. *International Review of Social Psychology*, 30(1), 92–101. <https://doi.org/10.5334/irsp.82>
- Diethelm, K. (2012). The Limits of Reproducibility in Numerical Simulation. *Computing in Science & Engineering*, 14(1), 64–72. <https://doi.org/10.1109/MCSE.2011.21>
- Dreber, A., & Johannesson, M. (2024). A framework for evaluating reproducibility and replicability in economics. *Economic Inquiry*, ecin.13244. <https://doi.org/10.1111/ecin.13244>
- Franco, A., Malhotra, N., & Simonovits, G. (2014). Publication bias in the social sciences: Unlocking the file drawer. *Science*, 345(6203), 1502–1505. <https://doi.org/10.1126/science.1255484>
- Frankenhuis, W. E., Panchanathan, K., & Smaldino, P. E. (2023). Strategic ambiguity in the social sciences. *Social Psychological Bulletin*, 18, e9923. <https://doi.org/10.32872/spb.9923>
- Garijo, D., Kinnings, S., Xie, L., Xie, L., Zhang, Y., Bourne, P. E., & Gil, Y. (2013). Quantifying Reproducibility in Computational Biology: The Case of the Tuberculosis Drugome. *PLoS ONE*, 8(11), e80278. <https://doi.org/10.1371/journal.pone.0080278>
- Gentleman, R., & Temple Lang, D. (2007). Statistical Analyses and Reproducible Research. *Journal of Computational and Graphical Statistics*, 16(1), 1–23. <https://doi.org/10.1198/106186007X178663>
- Gervasi, V., & Zowghi, D. (2010). On the Role of Ambiguity in RE. In R. Wieringa & A.

- Persson (Eds.), *Requirements Engineering: Foundation for Software Quality* (Vol. 6182, pp. 248–254). Springer Berlin Heidelberg.
https://doi.org/10.1007/978-3-642-14192-8_22
- Glatard, T., Lewis, L. B., Ferreira Da Silva, R., Adalat, R., Beck, N., Lepage, C., Rioux, P., Rousseau, M.-E., Sherif, T., Deelman, E., Khalili-Mahani, N., & Evans, A. C. (2015). Reproducibility of neuroimaging analyses across operating systems. *Frontiers in Neuroinformatics*, 9. <https://doi.org/10.3389/fninf.2015.00012>
- Gollwitzer, M., Abele-Brehm, A., Fiebach, C. J., Ramthun, R., Scheel, A., Schönbrodt, F., & Steinberg, U. (2021). Management und Bereitstellung von Forschungsdaten in der Psychologie: Überarbeitung der DGPs-Empfehlungen: DGPs-Kommission „Open Science“ (beschlossen durch den Vorstand der DGPs am 26. 06. 2020). *Psychologische Rundschau*, 72(2), 132–146.
<https://doi.org/10.1026/0033-3042/a000514>
- Goodman, S. N., Fanelli, D., & Ioannidis, J. P. A. (2016). What does research reproducibility mean? *Science Translational Medicine*, 8(341).
<https://doi.org/10.1126/scitranslmed.aaf5027>
- Hardwicke, T. E., Bohn, M., MacDonald, K., Hembacher, E., Nuijten, M. B., Peloquin, B. N., deMayo, B. E., Long, B., Yoon, E. J., & Frank, M. C. (2021). Analytic reproducibility in articles receiving open data badges at the journal Psychological Science: An observational study. *Royal Society Open Science*, 8(1), 201494.
<https://doi.org/10.1098/rsos.201494>
- Hardwicke, T. E., Mathur, M. B., MacDonald, K., Nilsson, G., Banks, G. C., Kidwell, M. C., Hofelich Mohr, A., Clayton, E., Yoon, E. J., Henry Tessler, M., Lenne, R. L., Altman, S., Long, B., & Frank, M. C. (2018). Data availability, reusability, and analytic reproducibility: Evaluating the impact of a mandatory open data policy at the journal *Cognition*. *Royal Society Open Science*, 5(8), 180448.
<https://doi.org/10.1098/rsos.180448>
- Herberich, E., Hothorn, T., Nettle, D., & Pollet, T. V. (2010). A re-evaluation of the statistical

- model in Pollet and Nettle 2009. *Evolution and Human Behavior*, 31(2), 150–151.
<https://doi.org/10.1016/j.evolhumbehav.2009.12.003>
- Herbert, S., Kingi, H., Stanchi, F., & Vilhuber, L. (2021). *The Reproducibility of Economics Research: A Case Study* (Working Paper Series No. 853). Banque de France.
<https://publications.banque-france.fr/en/reproducibility-economics-research-case-study>
- Heroux, M. A., Barba, L. A., Parashar, M., Stodden, V., & Taufer, M. (2018). *Toward a Compatible Reproducibility Taxonomy for Computational and Computing Sciences*.
<https://doi.org/10.2172/1481626>
- Hodges, C. B., Stone, B. M., Johnson, P. K., Carter, J. H., Sawyers, C. K., Roby, P. R., & Lindsey, H. M. (2022). Researcher degrees of freedom in statistical software contribute to unreliable results: A comparison of nonparametric analyses conducted in SPSS, SAS, Stata, and R. *Behavior Research Methods*, 55(6), 2813–2837.
<https://doi.org/10.3758/s13428-022-01932-2>
- Hoffmann, S., Schönbrodt, F., Elsas, R., Wilson, R., Strasser, U., & Boulesteix, A.-L. (2021). The multiplicity of analysis strategies jeopardizes replicability: Lessons learned across disciplines. *Royal Society Open Science*, 8(4), 201925.
<https://doi.org/10.1098/rsos.201925>
- Hong, S.-Y., Koo, M.-S., Jang, J., Esther Kim, J.-E., Park, H., Joh, M.-S., Kang, J.-H., & Oh, T.-J. (2013). An Evaluation of the Software System Dependency of a Global Atmospheric Model. *Monthly Weather Review*, 141(11), 4165–4172.
<https://doi.org/10.1175/MWR-D-12-00352.1>
- Huntington-Klein, N., Arenas, A., Beam, E., Bertoni, M., Bloem, J. R., Burli, P., Chen, N., Grieco, P., Ekpe, G., Pugatch, T., Saavedra, M., & Stopnitzky, Y. (2021). The influence of hidden researcher decisions in applied microeconomics. *Economic Inquiry*, 59(3), 944–960. <https://doi.org/10.1111/ecin.12992>
- ICMJE. (2024, January). *Recommendations for the Conduct, Reporting, Editing, and Publication of Scholarly work in Medical Journals*.

- <https://www.icmje.org/icmje-recommendations.pdf>
- Ince, D. C., Hatton, L., & Graham-Cumming, J. (2012). The case for open computer programs. *Nature*, 482(7386), 485–488. <https://doi.org/10.1038/nature10836>
- International Organization for Standardization. (2017). *Systems and software engineering—Vocabulary* (ISO Standard No. 24765:2017). <https://www.iso.org/standard/71952.html>
- JASP Team. (2024). *JASP* (Version 0.19.0) [Computer software]. <https://jasp-stats.org/>
- Jen, M.-H., Varney, B., Lee, K., Arancibia, B., Qi, M., Taylor, L., Fillmore, C., Rickert, J., Stackhouse, M., & Rimler, M. (2023). *Key Considerations When Understanding Differences in Statistical Methodology Implementations Across Programming Languages – An Introduction to the CAMIS Project* (No. WP-077). PHUSE. https://psiaims.github.io/CAMIS/publication/white_paper.html
- Keeling, K. B., & Pavur, R. J. (2007). A comparative study of the reliability of nine statistical software packages. *Computational Statistics & Data Analysis*, 51(8), 3811–3831. <https://doi.org/10.1016/j.csda.2006.02.013>
- Kohrt, F., Zerna, J., & Scheffel, C. (2024). *Code Publishing Tutorial*. <https://lmu-osc.github.io/code-publishing/>
- Krafczyk, M. S., Shi, A., Bhaskar, A., Marinov, D., & Stodden, V. (2021). Learning from reproducing computational results: Introducing three principles and the *Reproduction Package*. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 379(2197), 20200069. <https://doi.org/10.1098/rsta.2020.0069>
- Krähmer, D., Schächtele, L., & Schneck, A. (2023). Care to share? Experimental evidence on code sharing behavior in the social sciences. *PLOS ONE*, 18(8), e0289380. <https://doi.org/10.1371/journal.pone.0289380>
- Laurinavichyute, A., Yadav, H., & Vasishth, S. (2022). Share the code, not just the data: A case study of the reproducibility of articles published in the Journal of Memory and Language under the open data policy. *Journal of Memory and Language*, 125,

104332. <https://doi.org/10.1016/j.jml.2022.104332>
- Leamer, E. E. (1978). *Specification searches: Ad hoc inference with nonexperimental data*. Wiley.
- LeBel, E. P., McCarthy, R. J., Earp, B. D., Elson, M., & Vanpaemel, W. (2018). A Unified Framework to Quantify the Credibility of Scientific Findings. *Advances in Methods and Practices in Psychological Science*, 1(3), 389–402.
<https://doi.org/10.1177/2515245918787489>
- Leising, D., Grenke, O., & Cramer, M. (2023). Visual Argument Structure Tool (VAST) Version 1.0. *Meta-Psychology*, 7. <https://doi.org/10.15626/MP.2021.2911>
- Letter, J. (2021, February 3). JASP Output Verified With Other Statistical Packages. *JASP - Free and User-Friendly Statistical Software*.
<https://jasp-stats.org/2021/02/03/the-jasp-verification-project/>
- Livingston, E. H. (2020). Study Design and Statistics. In AMA Manual of Style Committee, *AMA Manual of Style* (11th ed., pp. 977–1096). Oxford University Press New York.
<https://doi.org/10.1093/jama/9780190246556.003.0019>
- López-ibáñez, M., Branke, J., & Paquete, L. (2021). Reproducibility in Evolutionary Computation. *ACM Transactions on Evolutionary Learning and Optimization*, 1(4), 1–21. <https://doi.org/10.1145/3466624>
- Matarese, V. (2022). Kinds of Replicability: Different Terms and Different Functions. *Axiomathes*, 32(S2), 647–670. <https://doi.org/10.1007/s10516-021-09610-2>
- McCoach, D. B., Rifken, G. G., Newton, S. D., Li, X., Kooker, J., Yomtov, D., Gambino, A. J., & Bellara, A. (2018). Does the Package Matter? A Comparison of Five Common Multilevel Modeling Software Packages. *Journal of Educational and Behavioral Statistics*, 43(5), 594–627. <https://doi.org/10.3102/1076998618776348>
- Menkveld, A. J., Dreber, A., Holzmeister, F., Huber, J., Johannesson, M., Kirchler, M., Neusüß, S., Razen, M., Weitzel, U., Abad-Díaz, D., Abudy, M. (Meni), Adrian, T., Ait-Sahalia, Y., Akmansoy, O., Alcock, J. T., Alexeev, V., Aloosh, A., Amato, L., Amaya, D., ... Zwinkels, R. (2024). Nonstandard Errors. *The Journal of Finance*,

- 79(3), 2339–2390. <https://doi.org/10.1111/jofi.13337>
- Mineault, P., Nozawa, K., & The Good Research Code Handbook Community. (2021). *The Good Research Code Handbook*. <https://goodresearch.dev/>.
<https://doi.org/10.5281/ZENODO.5796873>
- Monniaux, D. (2008). The pitfalls of verifying floating-point computations. *ACM Transactions on Programming Languages and Systems*, 30(3), 1–41.
<https://doi.org/10.1145/1353445.1353446>
- Nichols, T. E., Das, S., Eickhoff, S. B., Evans, A. C., Glatard, T., Hanke, M., Kriegeskorte, N., Milham, M. P., Poldrack, R. A., Poline, J.-B., Proal, E., Thirion, B., Van Essen, D. C., White, T., & Yeo, B. T. T. (2017). Best practices in data analysis and sharing in neuroimaging using MRI. *Nature Neuroscience*, 20(3), 299–303.
<https://doi.org/10.1038/nn.4500>
- Nuijten, M. B., Bakker, M., Maassen, E., & Wicherts, J. M. (2018). Verify original results through reanalysis before replicating. *Behavioral and Brain Sciences*, 41, e143.
<https://doi.org/10.1017/S0140525X18000791>
- Nüst, D., Konkol, M., Pebesma, E., Kray, C., Schutzeichel, M., Przibytzin, H., & Lorenz, J. (2017). Opening the Publication Process with Executable Research Compendia. *D-Lib Magazine*, 23(1/2). <https://doi.org/10.1045/january2017-nuest>
- Patel, C. J., Burford, B., & Ioannidis, J. P. A. (2015). Assessment of vibration of effects due to model specification can demonstrate the instability of observational associations. *Journal of Clinical Epidemiology*, 68(9), 1046–1058.
<https://doi.org/10.1016/j.jclinepi.2015.05.029>
- Patil, P., Peng, R. D., & Leek, J. T. (2016a). *A statistical definition for reproducibility and replicability*. <https://doi.org/10.1101/066803>
- Patil, P., Peng, R. D., & Leek, J. T. (2016b). What Should Researchers Expect When They Replicate Studies? A Statistical View of Replicability in Psychological Science. *Perspectives on Psychological Science*, 11(4), 539–544.
<https://doi.org/10.1177/1745691616646366>

- Penders, B., Holbrook, J. B., & De Rijcke, S. (2019). Rinse and Repeat: Understanding the Value of Replication across Different Ways of Knowing. *Publications*, 7(3), 52.
<https://doi.org/10.3390/publications7030052>
- Pham, H. V., Qian, S., Wang, J., Lutellier, T., Rosenthal, J., Tan, L., Yu, Y., & Nagappan, N. (2020). Problems and opportunities in training deep learning software systems: An analysis of variance. *Proceedings of the 35th IEEE/ACM International Conference on Automated Software Engineering*, 771–783.
<https://doi.org/10.1145/3324884.3416545>
- Pilgrim, C., Kent, P., Hosseini, K., & Chalstrey, E. (2023). Ten simple rules for working with other people's code. *PLOS Computational Biology*, 19(4), e1011031.
<https://doi.org/10.1371/journal.pcbi.1011031>
- Plesser, H. E. (2018). Reproducibility vs. Replicability: A Brief History of a Confused Terminology. *Frontiers in Neuroinformatics*, 11, 76.
<https://doi.org/10.3389/fninf.2017.00076>
- PyTorch Contributors. (2023). *Reproducibility*. PyTorch 2.5 Documentation.
<https://pytorch.org/docs/stable/notes/randomness.html#reproducibility>
- R Core Team. (2024). *R: A Language and Environment for Statistical Computing* [Computer software]. R Foundation for Statistical Computing. <https://www.R-project.org/>
- Rahal, C. (2024). *On the Responsible use of Pseudo-Random Number Generators in Scientific Research* [Computer software]. <https://github.com/crahal/seeds>
- Saltelli, A. (Ed.). (2008). *Sensitivity analysis* (Paperback ed). Wiley.
- Sandve, G. K., Nekrutenko, A., Taylor, J., & Hovig, E. (2013). Ten Simple Rules for Reproducible Computational Research. *PLoS Computational Biology*, 9(10), e1003285. <https://doi.org/10.1371/journal.pcbi.1003285>
- Scaria, L. (2018). *A Framework To Evaluate Pipeline Reproducibility Across Operating Systems* [Master's Thesis, Concordia University].
<https://spectrum.library.concordia.ca/id/eprint/984061/>
- Schmidt, S. (2009). Shall we Really do it Again? The Powerful Concept of Replication is

- Neglected in the Social Sciences. *Review of General Psychology*, 13(2), 90–100.
<https://doi.org/10.1037/a0015108>
- Schöch, C. (2023). Repetitive research: A conceptual space and terminology of replication, reproduction, revision, reanalysis, reinvestigation and reuse in digital humanities. *International Journal of Digital Humanities*, 5(2–3), 373–403.
<https://doi.org/10.1007/s42803-023-00073-y>
- Seibold, H., Czerny, S., Decke, S., Dieterle, R., Eder, T., Fohr, S., Hahn, N., Hartmann, R., Heindl, C., Kopper, P., Lepke, D., Loidl, V., Mandl, M., Musiol, S., Peter, J., Piehler, A., Rojas, E., Schmid, S., Schmidt, H., ... Nalenz, M. (2021). A computational reproducibility study of PLOS ONE articles featuring longitudinal data analyses. *PLOS ONE*, 16(6), e0251194. <https://doi.org/10.1371/journal.pone.0251194>
- Sharma, N. K., Ayyala, R., Deshpande, D., Patel, Y., Munteanu, V., Ciorba, D., Bostan, V., Fiscutean, A., Vahed, M., Sarkar, A., Guo, R., Moore, A., Darci-Maher, N., Nogoy, N., Abedalthagafi, M., & Mangul, S. (2024). Analytical code sharing practices in biomedical research. *PeerJ Computer Science*, 10, e2066.
<https://doi.org/10.7717/peerj-cs.2066>
- Silberzahn, R., Uhlmann, E. L., Martin, D. P., Anselmi, P., Aust, F., Awtrey, E., Bahník, Š., Bai, F., Bannard, C., Bonnier, E., Carlsson, R., Cheung, F., Christensen, G., Clay, R., Craig, M. A., Dalla Rosa, A., Dam, L., Evans, M. H., Flores Cervantes, I., ... Nosek, B. A. (2018). Many Analysts, One Data Set: Making Transparent How Variations in Analytic Choices Affect Results. *Advances in Methods and Practices in Psychological Science*, 1(3), 337–356. <https://doi.org/10.1177/2515245917747646>
- Stan Development Team. (2024). *Reproducibility*. Stan Reference Manual.
<https://mc-stan.org/docs/reference-manual/reproducibility.html>
- Steege, S., Tuerlinckx, F., Gelman, A., & Vanpaemel, W. (2016). Increasing Transparency Through a Multiverse Analysis. *Perspectives on Psychological Science*, 11(5), 702–712. <https://doi.org/10.1177/1745691616658637>
- Steiner, P. M., Wong, V. C., & Anglin, K. (2019). A Causal Replication Framework for

- Designing and Assessing Replication Efforts. *Zeitschrift Für Psychologie*, 227(4), 280–292. <https://doi.org/10.1027/2151-2604/a000385>
- Stodden, V. (2015). Reproducing Statistical Results. *Annual Review of Statistics and Its Application*, 2(1), 1–19. <https://doi.org/10.1146/annurev-statistics-010814-020127>
- Ulpts, S., & Schneider, J. W. (2024). *A conceptual review of uses and meanings of reproducibility and replication*. MetaArXiv. <https://doi.org/10.31222/osf.io/entu4>
- Van Den Bergh, D., Wagenmakers, E.-J., & Aust, F. (2023). Bayesian Repeated-Measures Analysis of Variance: An Updated Methodology Implemented in JASP. *Advances in Methods and Practices in Psychological Science*, 6(2), 25152459231168024. <https://doi.org/10.1177/25152459231168024>
- W.D. (2019, August 31). Scikit-learn's Defaults are Wrong. *R y x, r*. <https://ryxcommar.com/2019/08/30/scikit-learns-defaults-are-wrong/>
- Weissgerber, T. L., Garcia-Valencia, O., Garovic, V. D., Milic, N. M., & Winham, S. J. (2018). Meta-Research: Why we need to report more than “Data were Analyzed by t-tests or ANOVA.” *eLife*, 7, e36163. <https://doi.org/10.7554/eLife.36163>
- Welch, I. (2019). Reproducing, Extending, Updating, Replicating, Reexamining, and Reconciling. *Critical Finance Review*, 8(1–2), 301–304. <https://doi.org/10.1561/104.000000082>
- Wilensky, U., & Rand, W. (2007). Making Models Match: Replicating an Agent-Based Model. *Journal of Artificial Societies and Social Simulation*, 10(4), 2. <https://www.jasss.org/10/4/2.html>
- Wilson, G., Bryan, J., Cranston, K., Kitzes, J., Nederbragt, L., & Teal, T. K. (2017). Good enough practices in scientific computing. *PLOS Computational Biology*, 13(6), e1005510. <https://doi.org/10.1371/journal.pcbi.1005510>
- Young, C. (2009). Model Uncertainty in Sociological Research: An Application to Religion and Economic Growth. *American Sociological Review*, 74(3), 380–397. <https://doi.org/10.1177/000312240907400303>
- Young, C., & Holsteen, K. (2017). Model Uncertainty and Robustness: A Computational

Framework for Multimodel Analysis. *Sociological Methods & Research*, 46(1), 3–40.

<https://doi.org/10.1177/0049124115610347>

Zhang, J., & Robinson, D. T. (2021). Replication of an agent-based model using the

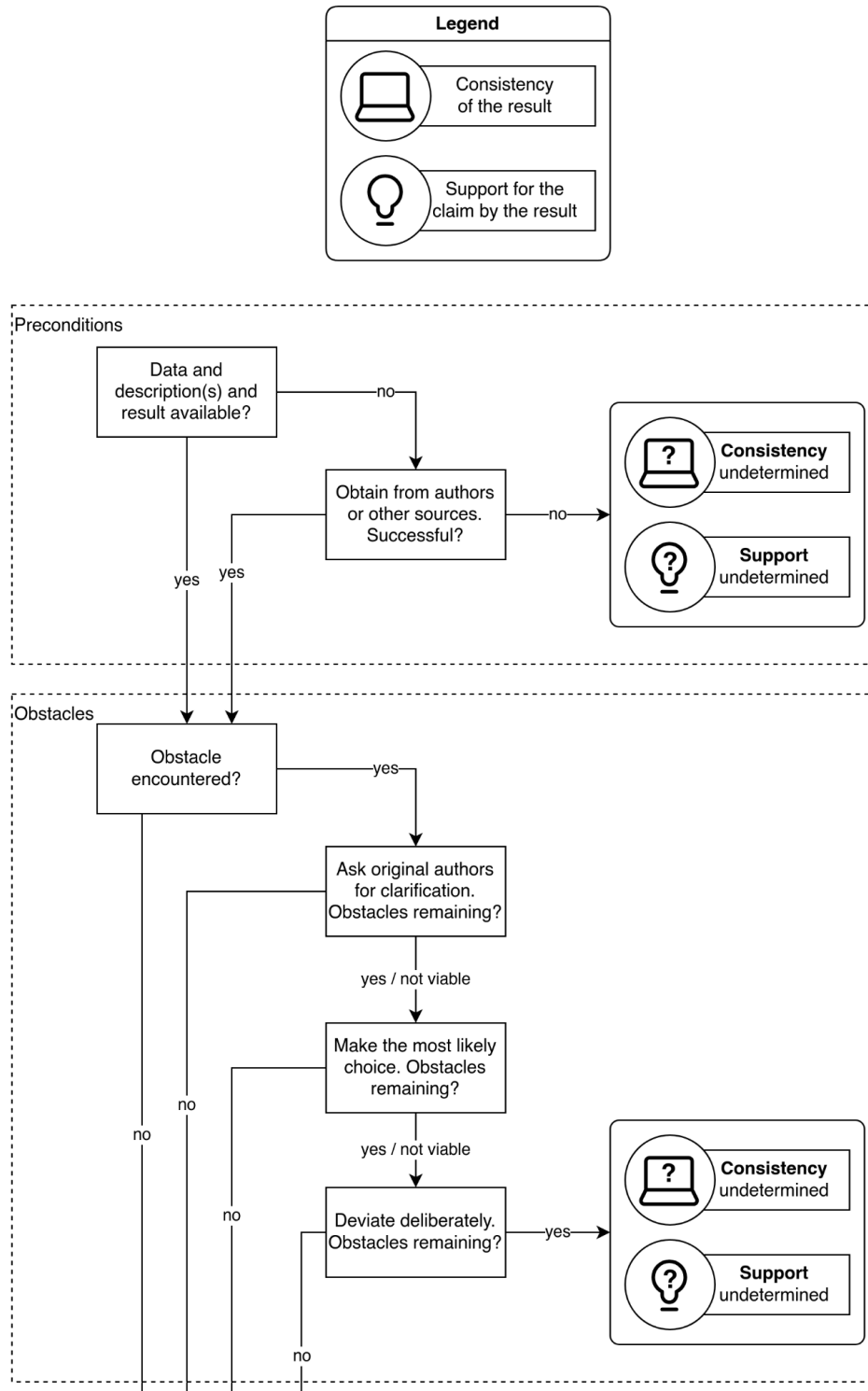
Replication Standard. *Environmental Modelling & Software*, 139, 105016.

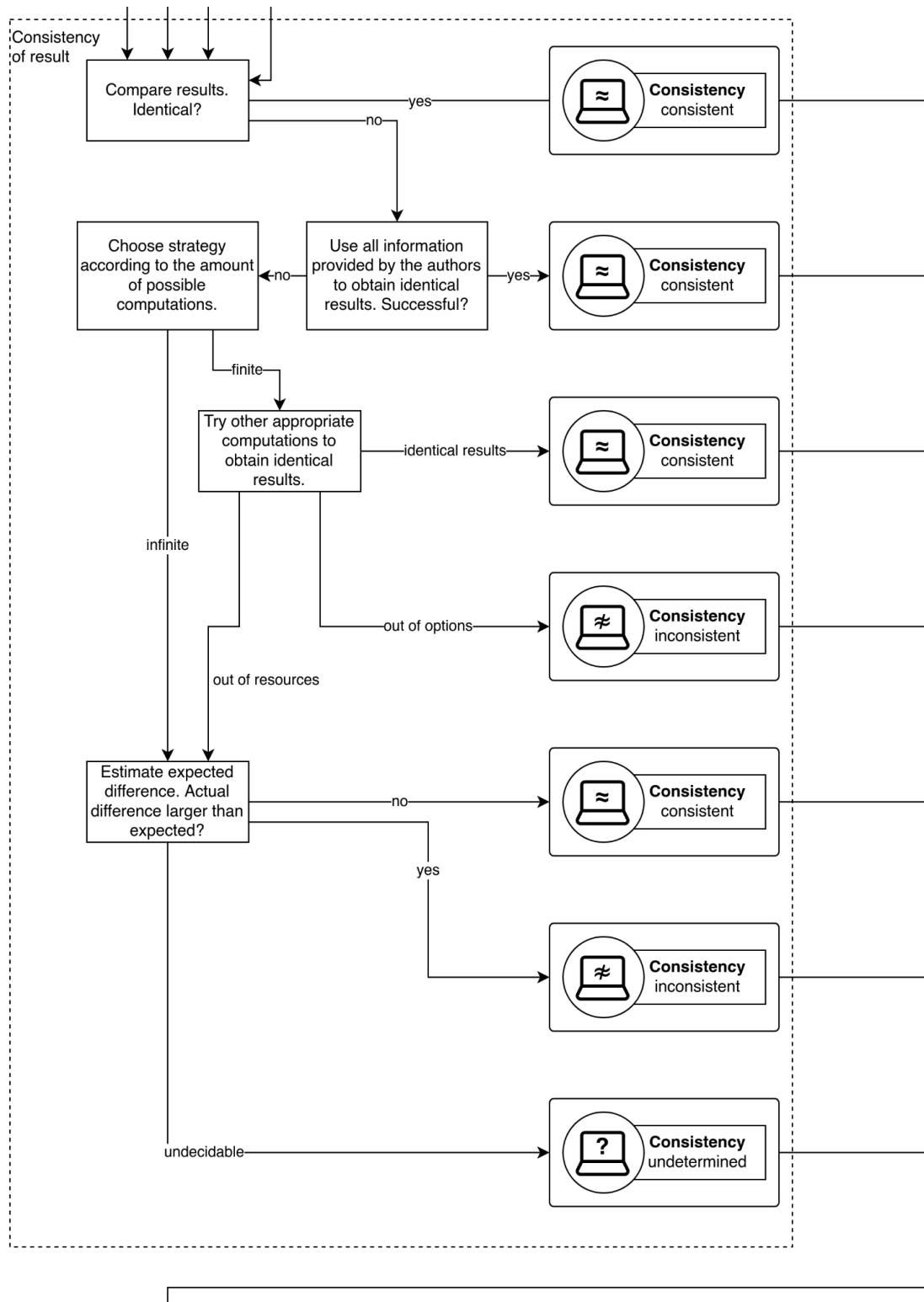
<https://doi.org/10.1016/j.envsoft.2021.105016>

7. Appendix

Figure A1

Detailed Flowchart for Performing a Reproduction of One Result





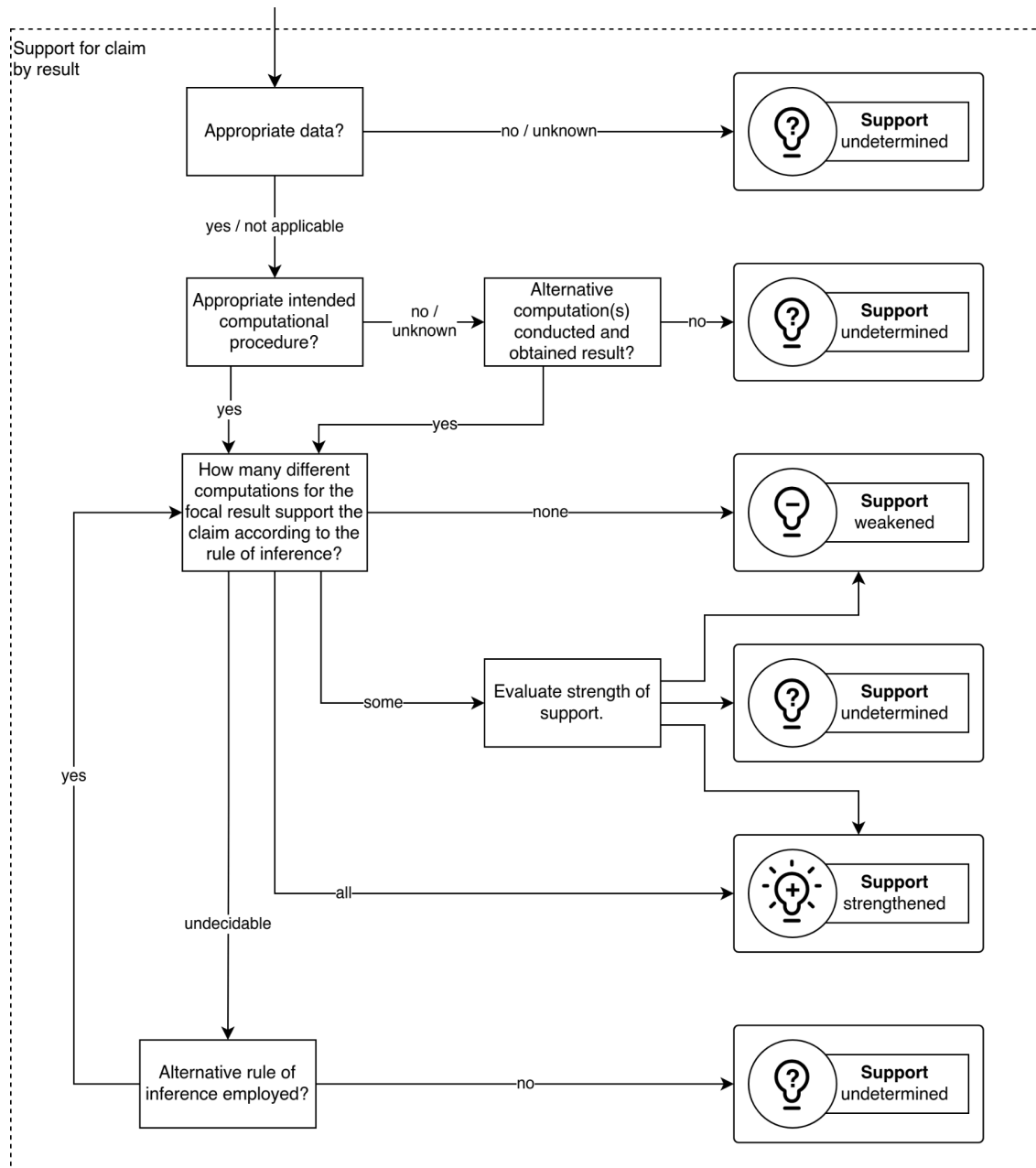


Figure A2

Reproduction Report

Reproduction report

Fill out one separate report for each individual result and each individual claim that is investigated within a work.

About you

Contact person

Who is filling out this report and when?

Reproducers' expertise

Indicate the level of expertise among the reproducers concerning the field of the original work.

- ☐ Author: The authors of the original work are among the reproducers.
- ☐ Expert: The reproducers are familiar with the field.
- ☐ Novice: The reproducers have basic knowledge about the field.
- ☐ Minimal: The reproducers have little knowledge about the field.

Scope and preconditions

Investigated work

Provide a reference to the original work, including any permanent identifiers such as a DOI.

Investigated claim

Which claim of the original work is being investigated? Provide an exact quote and, if possible, indicate the page number. Leave empty if only a result and no claim is investigated.

Investigated result

Which result is being investigated (supposedly supporting the claim above)? For example, this can be a particular figure, a table or a set of numbers. Describe its location in the work and, if possible, copy it here.

☐ The non-availability of the result prevents the reproduction from being conducted.

Obtained data

In the case of an empirical study: Which preprocessing steps have been already performed for the data obtained for this reproduction? In other words, how do the obtained data differ from the primary data, that is, the first digital representation of the raw data?

☐ The non-availability of the data prevents the reproduction from being conducted.

Involved descriptions

Which descriptions are consulted during this reproduction? Descriptions are instructions that describe the analytic treatment of data. For example, they can be part of the preregistration, the manuscript, any supplementary materials, or the source code.


Indicate for each involved description...

- where it was obtained from,
- whether it is checked for coherence with the other consulted descriptions, and
- whether it informs the computation during the reproduction.

☐ The non-availability of the descriptions prevents the reproduction from being conducted.

Inherited source code

Of those description involved, to what extent is source code inherited from the original work? In other words, is any existing source code used for re-execution? Is it being modified or extended?



Applied methods standards


Which methods standards are followed when making unspecified choices, checking the coherence, estimating expected differences, and evaluating the appropriateness of specified choices? Elaborate on this over the course of the reproduction where necessary.



Obstacles


Underspecification

Was the description of the computation insufficient, requiring choices left unspecified by the original authors? Which choices have been explored by the reproducers?



Conflicting choices

Were there grounds for conflicting choices? How did the reproducers deviate from the descriptions?



Other obstacles

Were there any other issues with the result, the data, or the descriptions that prevent a result from being obtained?



Faults

Were any faults found responsible for the original computation not corresponding to its descriptions, data, and result? (How) were they corrected?

(A specified choice that is deemed inappropriate by the reproducers does not qualify here.)



Consistency of result

Evaluate the consistency of the reproduced result:

- ☐ Consistent: The reproduced result is consistent with the original result.
- ☐ Inconsistent: The reproduced result is inconsistent with the original result.
- ☐ Undetermined: No statement regarding the consistency can be made, for example, due to obstacles or violated preconditions.

If the consistency is not *undetermined*, provide the reproduced result that led to this evaluation (that is, with corrected faults as described above).

If the consistency is *undetermined*, provide the reason below:

Support for claim from result

1. Appropriate data collection

Is the study design and the data collection (if applicable) meaningful and appropriate? Why (not)?

2. Appropriate computation

Is the computation described in the descriptions meaningful and appropriate? Why (not)?

Was an alternative (more appropriate) computation conducted? Which?



3. Claim follows from result

Provide the results from all explored computations. For example, in the case of underspecification, provide the results from all valid computational choices (if possible). If an alternative computation has been conducted (see previous question), only report those results.



Explain how many different computations for the same result support the claim according to the rule of inference employed by the original authors. In case this is undecidable (e.g., because the rule of inference is unknown), use an alternative rule of inference.

Evaluation

Consider all the different computations that have been performed for the same (focal) result to decide on the strength of support for the claim.

- ☐ strengthened: After conducting the reproduction, the support for the claim is strengthened.
- ☐ weakened: After conducting the reproduction, the support for the claim is weakened.
- ☐ undetermined: No statement regarding the support for the claim can be made.

If the support is *undetermined*, provide the reason below:

Communication

Were original authors contacted (e.g., for clarification), and did they respond?

Suggested improvements

Are there any suggested improvements (e.g., better documentation, initial data sharing)?

Time effort

How much time was spent on the reproduction in total working hours?

Figure A3

VAST Display 1 "Descriptions describe computations"

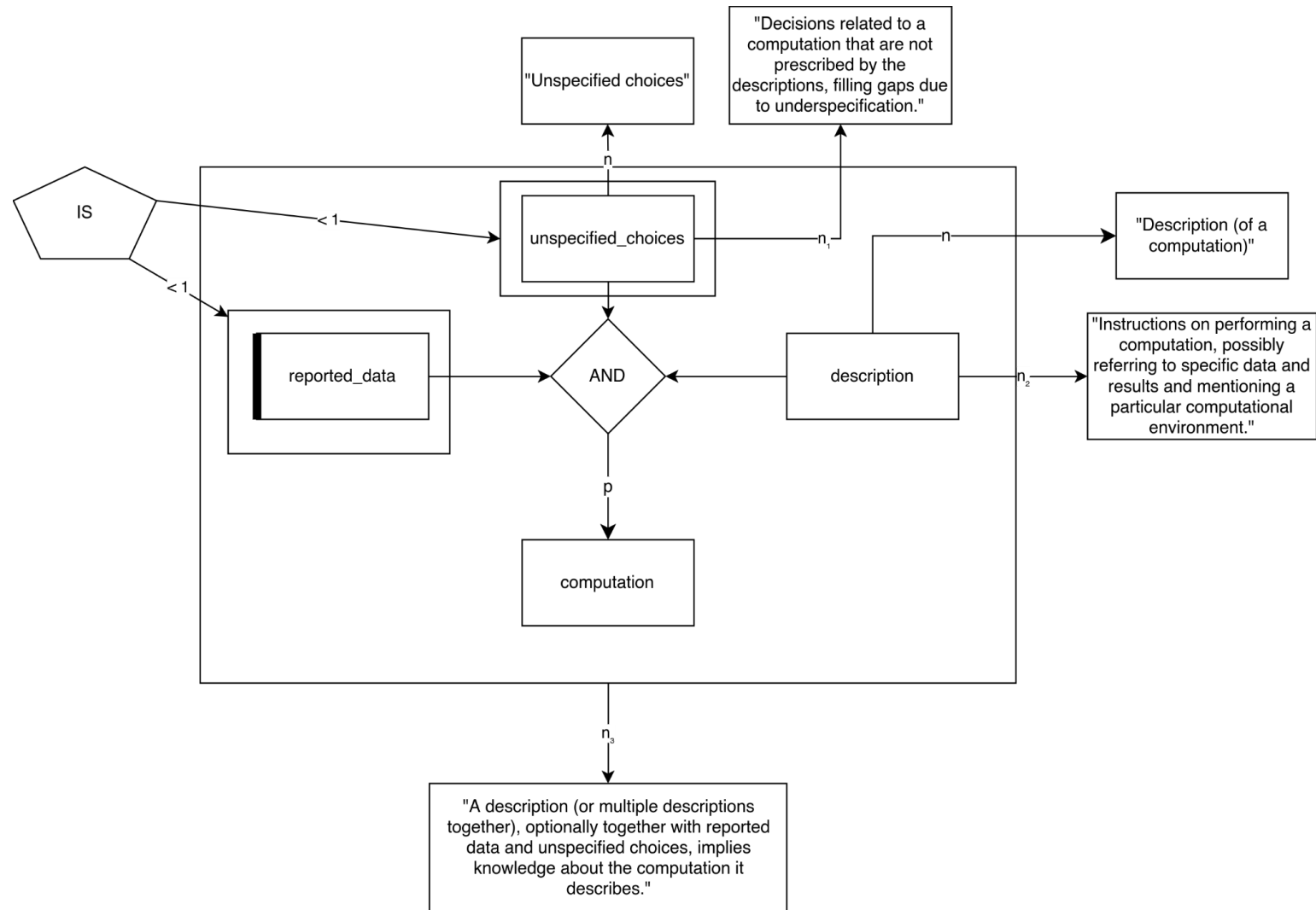


Figure A4

VAST Display 2 "Inferring computations from descriptions"

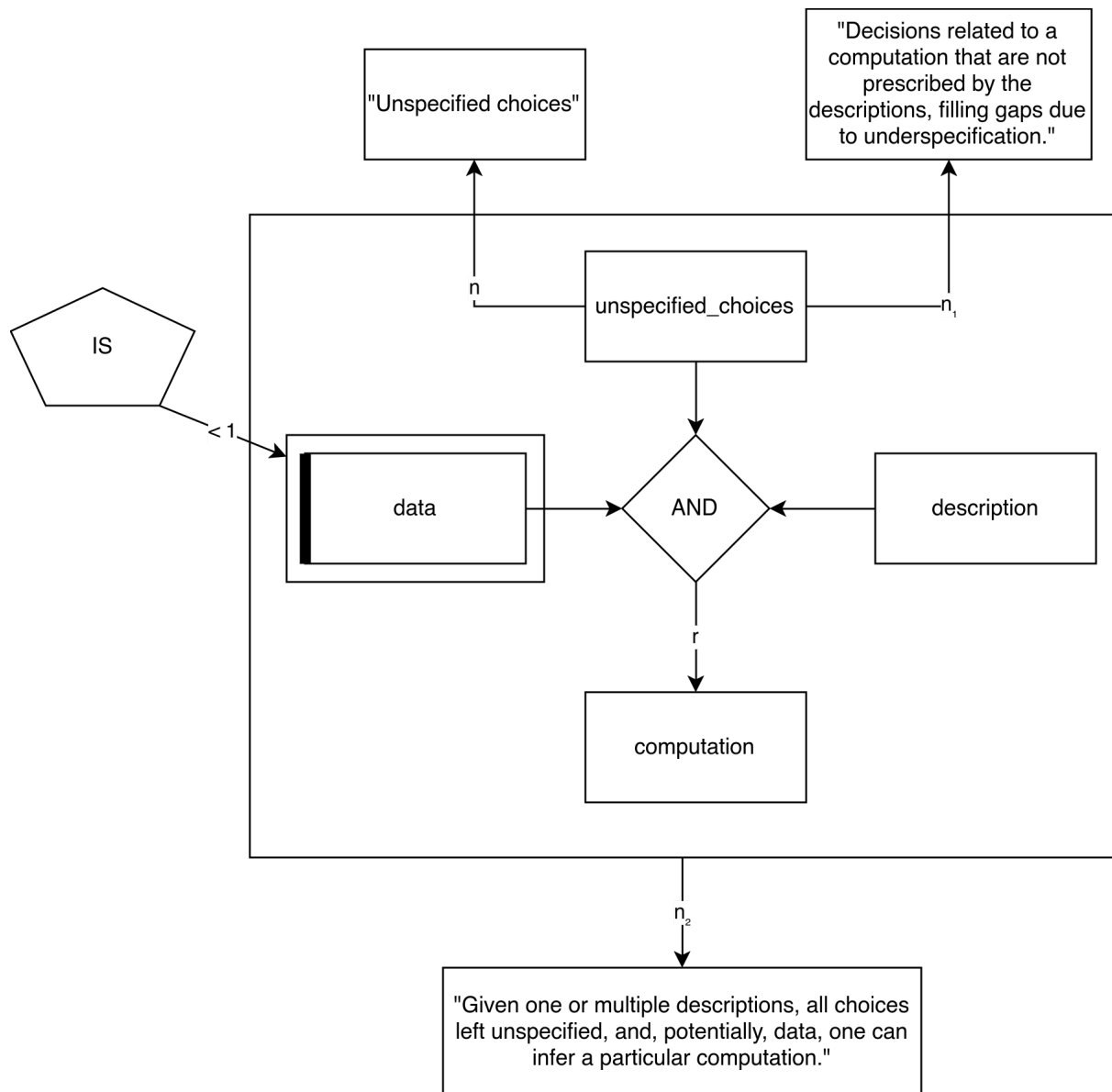


Figure A5

VAST Display 3 “Computational choices”

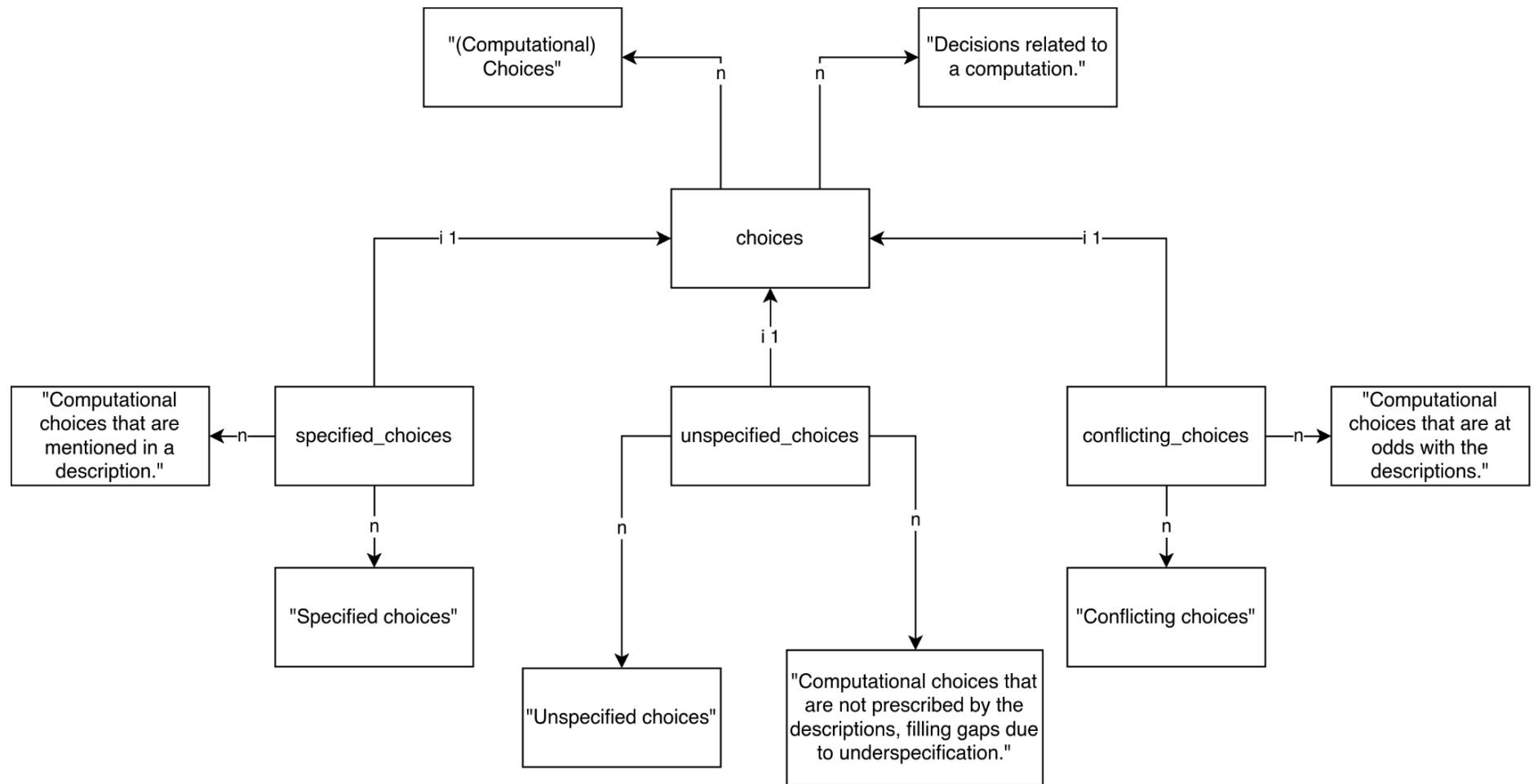


Figure A6

VAST Display 4 “Code”

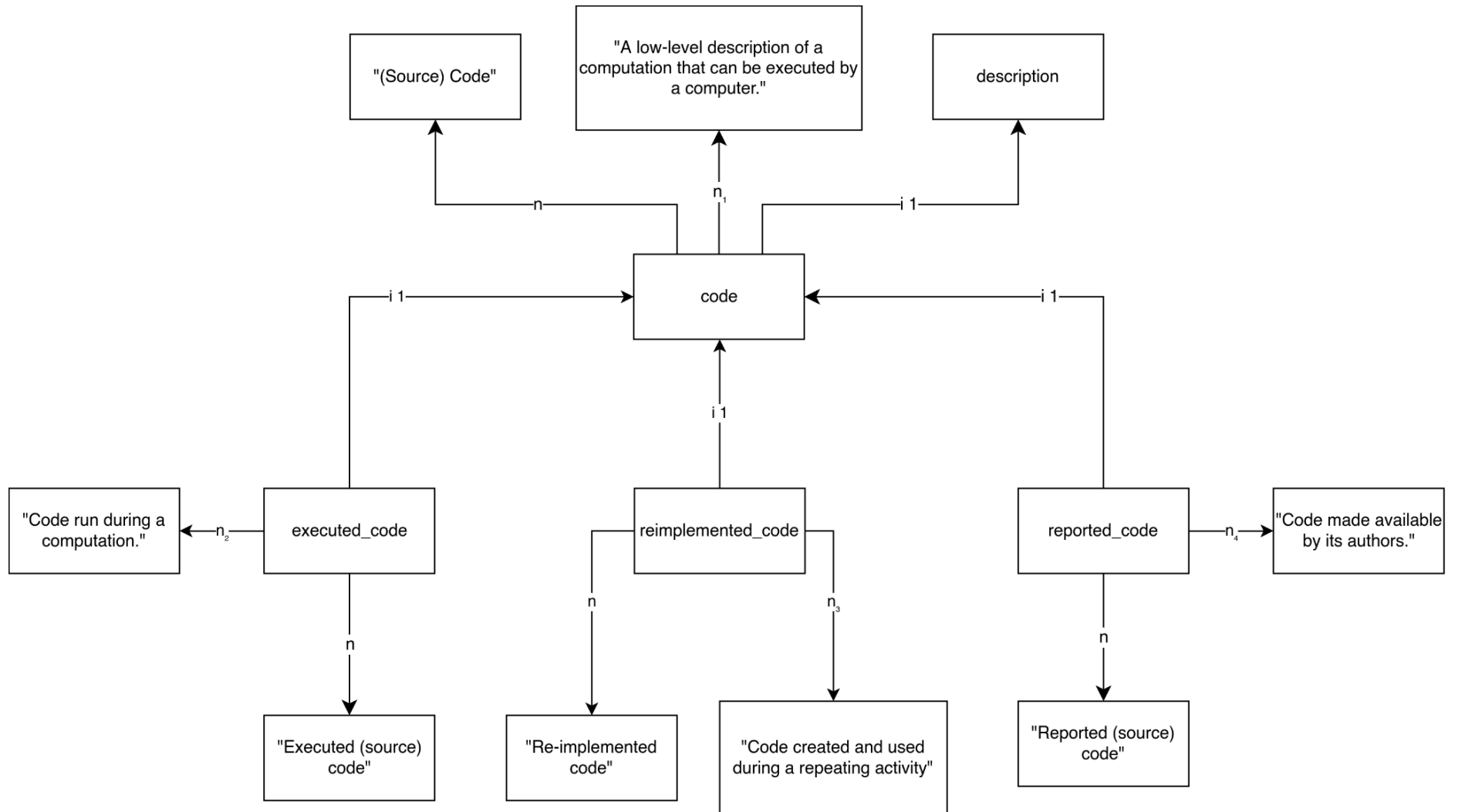


Figure A7

VAST Display 5 "Coherence of descriptions"

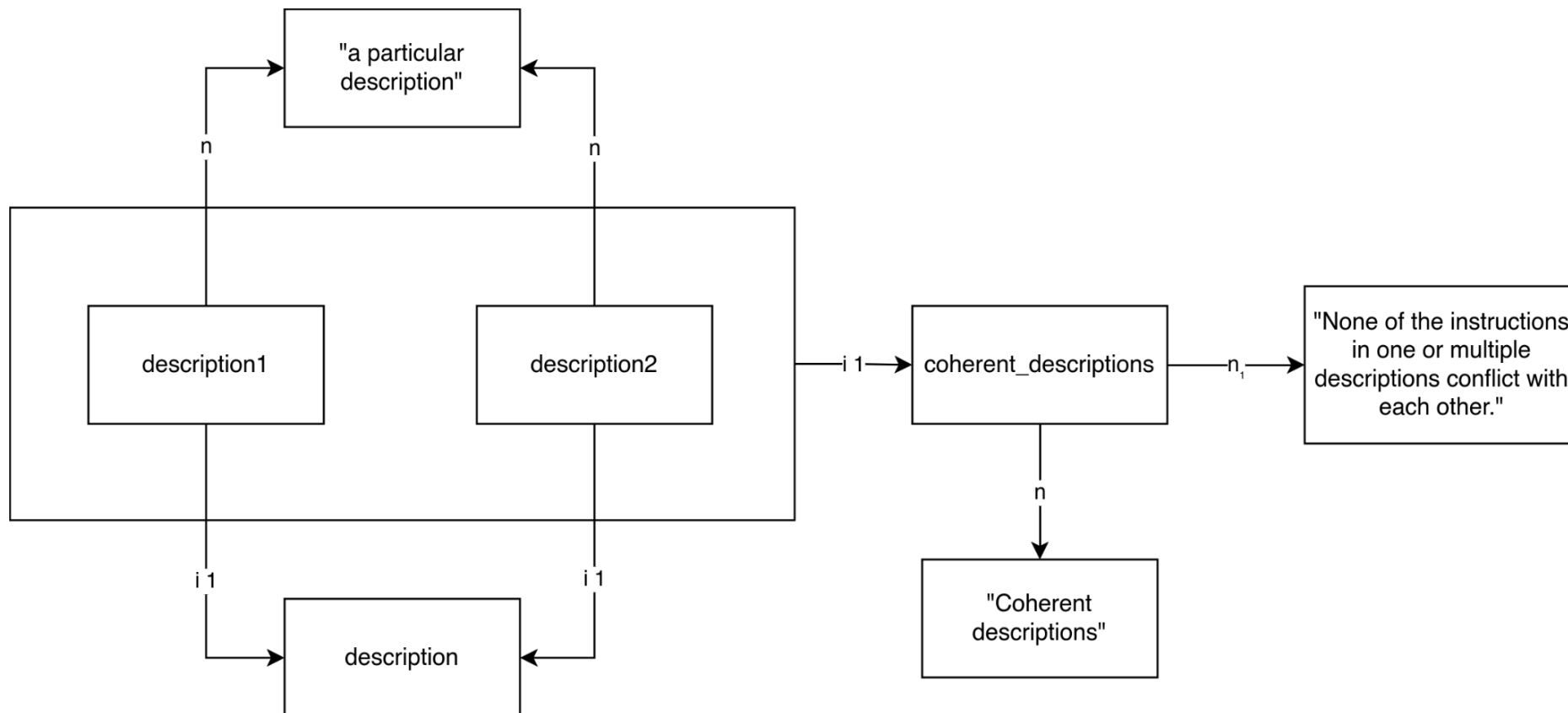


Figure A8

VAST Display 6 "Data"

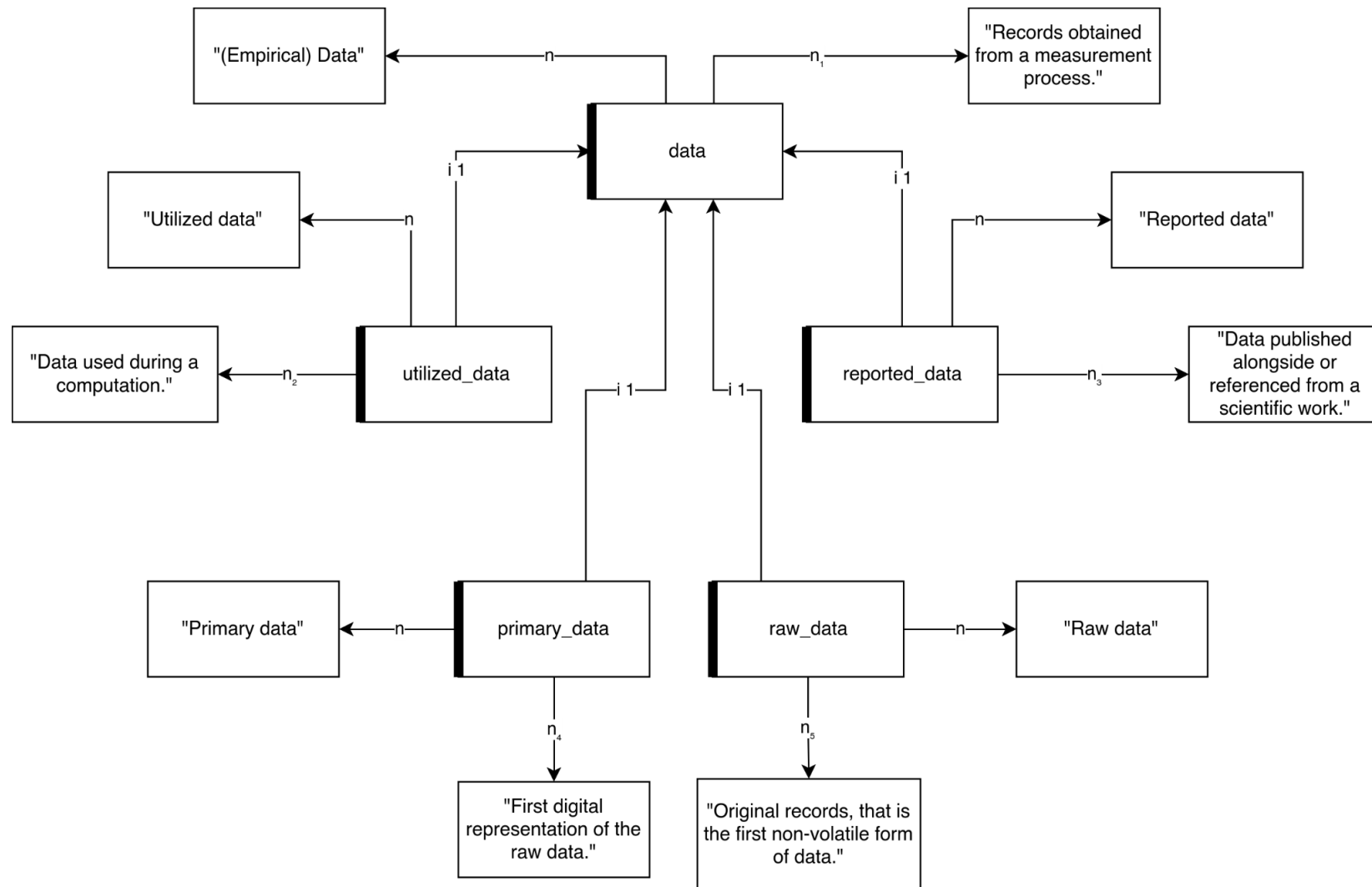


Figure A9

VAST Display 7 "Computation"

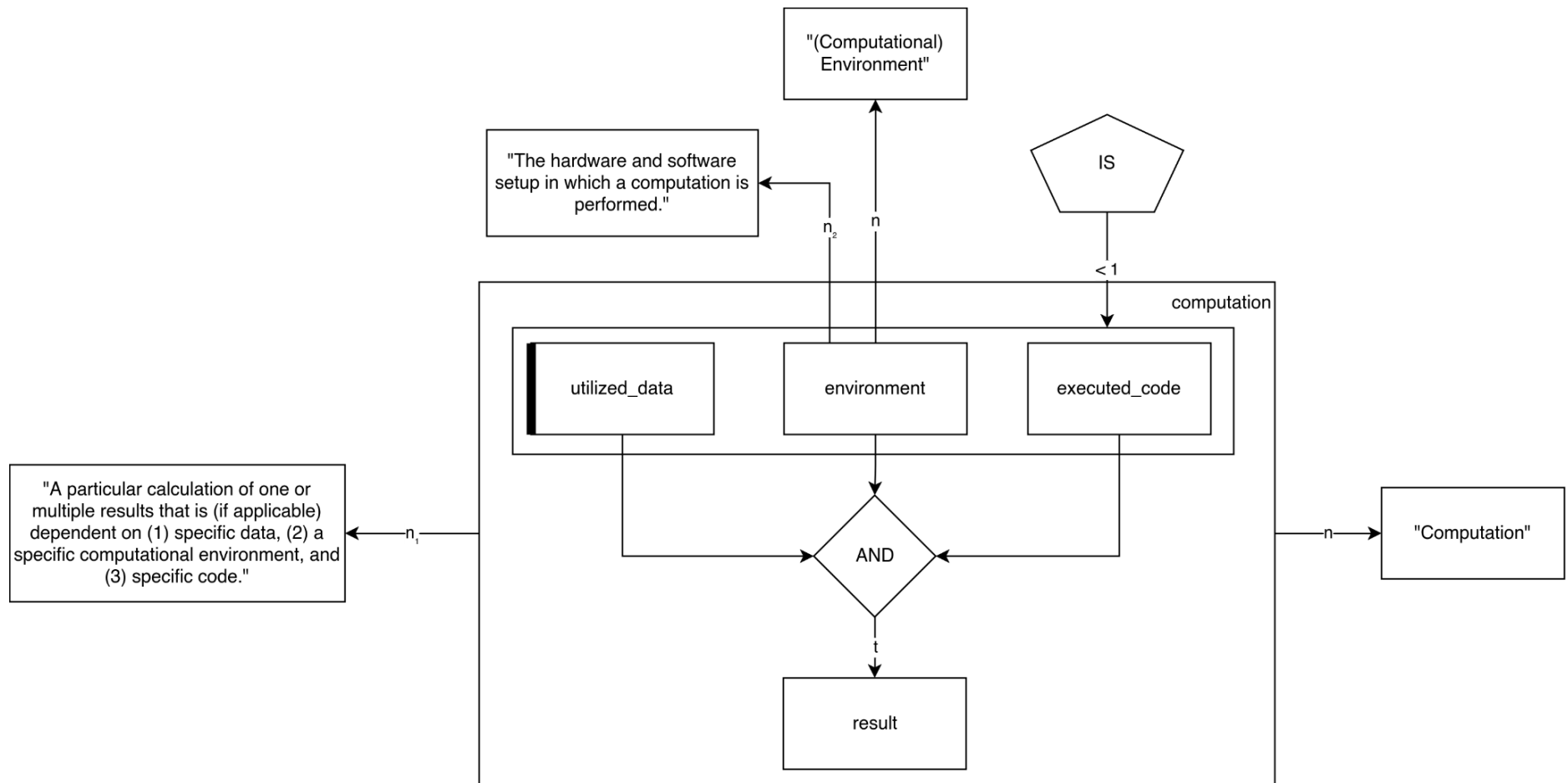
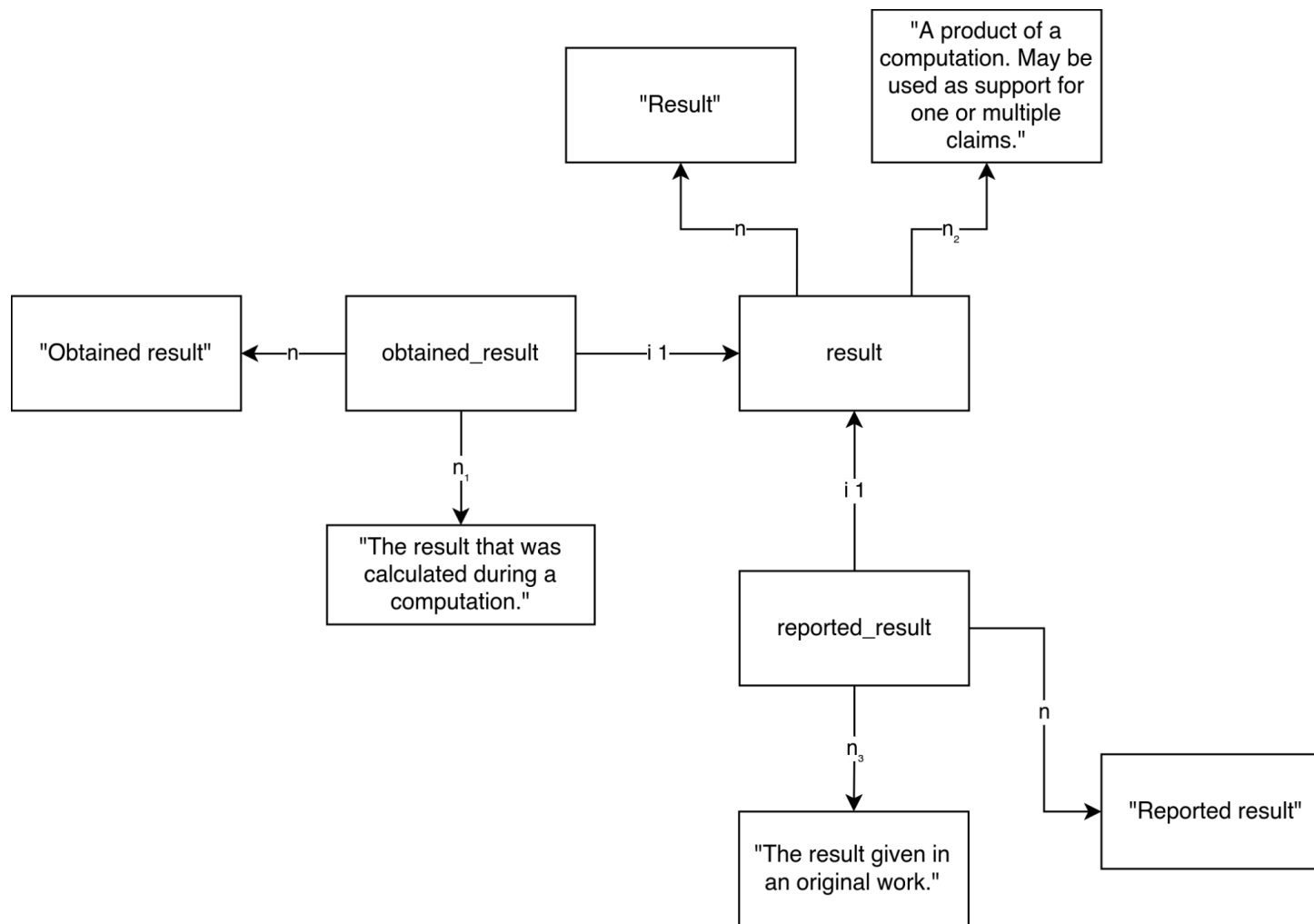


Figure A10

VAST Display 8 "Result"



VAST Display 9a “Re-execution”



Figure A12

VAST Display 9b "Faults"

