

## Conference Abstract

# Avoiding Conflicting Assertions: Approaches to Developing Consistent Test Implementations.

Paul J. Morris ‡

‡ Museum of Comparative Zoology, Harvard University, Cambridge, MA, United States of America

Corresponding author: Paul J. Morris ([mole@morris.net](mailto:mole@morris.net))

Received: 27 Mar 2018 | Published: 18 May 2018

Citation: Morris P (2018) Avoiding Conflicting Assertions: Approaches to Developing Consistent Test Implementations. Biodiversity Information Science and Standards 2: e25324.

<https://doi.org/10.3897/biss.2.25324>

## Abstract

What is a provider (or consumer) of biodiversity data to think when one quality assessment tool asserts that a particular problem exists in their data, while a different tool asserts that this problem is not present? Is there a problem with their data? Is there a problem with one of the tools? The Biodiversity Data Quality Task Group 2 is developing a suite of standardized descriptions of tests (validations, measures, amendments) of biodiversity data, implementations of which would be expected to provide consistent assertions about a particular data set so that input of identical data sets into two different test suite implementations will produce the same results (for some meaning of “the same”).

Development of standard test definitions is a big step in the direction of consistency. More is needed. Clear and detailed specifications for each test will help. For example, data might have suitable quality for global change analysis if collecting dates have a temporal resolution of one year or less. One implementer's test may check if the event date has a duration of 365 days or less, another might account for leap days, another might test if the data can be unambiguously binned into single years. For some data, each implementation will produce different assertions about the record. If the standard test specification states which of these meanings apply, then correct implementations should make identical assertions. To tell, however, if two implementations of a suite of tests will produce the same result for identical inputs we need two things, one is a set of tests (of the tests), the other is an understanding of what it means for results to be the same. It is expected that there will

be changes in the results of tests of scientific names over time, and that different authorities will have different opinions about that set of scientific names. One element of “the same” is an expectation that results will be the same when test implementations are run at the same time and with the same configuration, but not necessarily otherwise.

Consider tests at three levels: First, tests of the internals of a test, separate from the fitness for use framework (Veiga et al. 2017) or serialization of test results. At this first level, unit tests are very appropriate, but these are tightly coupled to the language of implementation and the unit testing framework, and to the internal details of the implementation. Unit tests are very effective for software quality control, but not particularly portable. Second, consider tests of the output of a suite of tests. At this level (of integration tests), we are tightly coupled to both the fitness for use framework and the serialization, and the meaning of “the same” is important. Different software implementations may be expected to have different orders of output for the same input, and human readable comments would be expected to vary (e.g. with internationalization). Identity of machine readable assertions but in varying orders should be tolerable, but this is not easily accomplished. Implementation at this level is difficult. Third, consider tests of the framework output of a particular test. Order becomes unimportant, only machine readable framework assertions can be considered, and this is probably the level to target for testing. Input data for tests could be synthetic, real, or modified real data. Real data has the advantage of being realistic, but it is difficult to find real data which contains single issues. Clean real data into which synthetic error conditions have been introduced is enticing for test purposes, but risks confusion with real data, so I propose some standard values for certain Darwin Core terms for identifying synthetic data.

## Presenting author

Paul J. Morris

## Funding program

ABI

## Grant title

Collaborative Research: ABI Development: Kurator: A Provenance-enabled Workflow Platform and Toolkit to Curate Biodiversity Data [1356438](https://doi.org/10.1371/journal.pone.0178731)

## References

- Veiga AK, Saraiva AM, Chapman AD, Morris PJ, Gendreau C, Schigel D, Robertson TJ (2017) A conceptual framework for quality assessment and management of biodiversity data. PLOS ONE 12 (6): e0178731. <https://doi.org/10.1371/journal.pone.0178731>