

Einige Anmerkungen zur „Handreichung zur Rezension von Forschungssoftware“ (Arch. Inf. 43, 2020)

Massimiliano Carloni

Zusammenfassung – In der Ausgabe 43 der Archäologischen Informationen haben HOMBURG U. A. eine Handreichung zur Rezension von Forschungssoftware veröffentlicht, die als Diskussionsbeitrag gedacht war. Der vorliegende Aufsatz wird als Teil der Debatte präsentiert, die der ursprüngliche Beitrag eröffnen wollte. Insbesondere werden nach einer allgemeinen Betrachtung der Handreichung einige Aspekte berücksichtigt, die evtl. weiterentwickelt oder diskutiert werden könnten: die Definition von Forschungssoftware; die Frage, an welchem Zeitpunkt ihrer Entwicklung Software rezensiert werden sollte (und wie häufig); die möglichen Anwendungen grundlegender Forschungsprinzipien und -paradigmen (Open Science; FAIR- und CARE-Prinzipien) auf Forschungssoftware; die Berücksichtigung von Funktionen für die kollaborative Forschung.

Schlagwörter – Archäologie; Archäoinformatik; digitale Geisteswissenschaften; Softwarebewertung; Datenmanagement; Metadaten; FAIR-Prinzipien

Title – Some notes on the “Recommendations for the review of archaeological research software” (Arch. Inf. 43, 2020)

Abstract – In issue 43 of the Archäologische Informationen, HOMBURG ET AL. published an article with recommendations for the review of research software, which was intended as a contribution open to discussion. The present article should be considered as part of the debate that the original contribution intended to initiate. In particular, after a general evaluation of the proposed recommendations, the paper takes into account some aspects that could be further developed or discussed: the definition of research software; when and how often research software should be reviewed; the possible applications of basic research principles and paradigms (Open Science; FAIR and CARE principles) to research software; the consideration of features for collaborative research.

Key words – archaeology; archaeoinformatics; digital humanities; software review; data management; metadata; FAIR principles

Einleitung

Der Diskussionsbeitrag „Handreichung zur Rezension von Forschungssoftware in der Archäologie und den Altertumswissenschaften“, der in der Ausgabe 43 der Archäologischen Informationen veröffentlicht wurde (HOMBURG U. A., 2020), ist unter mehreren Gesichtspunkten wertvoll. Erstens, weil er ein Systematisierungsversuch in einer Gattung, der Rezension, ist, die seit Langem durch „unausgesprochene“ Regeln bestimmt wurde, wie die Autoren selbst feststellen (HOMBURG U. A., 2020, 357). Dies betrifft nicht nur Softwarerezensionen, sondern das Genre der wissenschaftlichen Rezensionen insgesamt, das – anders als beispielsweise der Kommentar (s. z. B. MOST, 1999) – in der Wissenschaftsgeschichte keine besondere Beachtung gefunden hat (ein weiteres aktuelles Beispiel für Besprechungsrichtlinien, in diesem Fall für digitale Editionen, ist SAHLE, 2014).

Zweitens hat diese Handreichung das Verdienst, grundlegende Fragen auf den Tisch zu legen, die ganz allgemein die Rolle der neuen „digitalen Geisteswissenschaften“ betreffen: Was ist Forschungssoftware? Kann ein Programm als eine wissenschaftliche Leistung angesehen werden?

Können wir überprüfen, was eine Software tut, so wie wir die Methodik einer „herkömmlichen“ Veröffentlichung diskutieren? Darüber hinaus wird die Bedeutung von Open Source nicht nur als Verfügbarkeit von Software für eine Community hervorgehoben, sondern auch als Voraussetzung für die Möglichkeit, Forschungssoftware mit der Legitimität und Transparenz einer wissenschaftlichen Veröffentlichung (und in einigen Fällen sogar mit mehr als einer traditionellen Veröffentlichung) zu produzieren und verbreiten.

Schließlich ist dieser Beitrag auch eine wichtige Überlegung im Hinblick auf die Softwareentwicklung an sich. Die Autoren machen einen wichtigen Unterschied zwischen dem idealen Ziel eines Programms und seiner technischen Umsetzung. Diese Unterscheidung ermöglicht es dem Rezensenten, sich bei der Bewertung von Forschungssoftware zu orientieren und festzustellen, wo eventuelle Probleme liegen: in Programmierfehlern, in einer zugrunde liegenden technischen Einschränkung oder sogar in einem falschen Ansatz. In diesem Sinne bietet der Beitrag auch eine Reihe von „best practices“ für die Entwicklung, die sehr gut auf jedes Softwareprodukt, auch außerhalb des Forschungsbereichs, angewendet werden können.

In den folgenden Absätzen werden einige Aspekte vorgestellt, die in der Handreichung vielleicht noch erwähnt oder weiterentwickelt werden könnten. Die folgenden Beobachtungen gelten jedoch nur als Anregungen und Diskussionspunkte, denn die Handreichung ist in jedem Fall eine erhellende und fast vollständige Untersuchung aller Aspekte im Zusammenhang mit Forschungssoftware. Es ist zu hoffen, dass diese nicht nur zu einem bewussteren Verfassen von Rezensionen führt, sondern auch und vor allem, wie die Autoren hoffen, zu einer sorgfältigeren Bewertung und Berücksichtigung der wichtigen Rolle, die Software in der Forschungsarbeit spielt.

Was ist Forschungssoftware?

Das Ziel, das sich die Autoren gesetzt haben, ist sicherlich nicht unproblematisch; vielleicht liegt die Schwierigkeit manchmal in der Zweideutigkeit des Gegenstands. Können wir beispielsweise immer eine klare Grenze zwischen Forschungssoftware und anderen Arten von Software ziehen, z. B. digitalen Werkzeugen (HOMBURG U. A., 2020, 358), die von Forschern täglich verwendet werden? In einigen Fällen scheint die Unterscheidung ziemlich sicher zu sein (man denke z. B. an ein Textverarbeitungs- oder Tabellenkalkulationsprogramm, das für ein breites Publikum bestimmt ist), in anderen verschwimmt die Grenze (zur Schwierigkeit, eine eindeutig definierte Kategorie von „Forschungssoftware“ festzulegen, siehe z. B. SOCHAT, 2020, die von einem „gradient of features“ spricht). Die Autoren des Artikels selbst erwähnen später als Beispiel ein CAD-Programm, das aktiv von Archäologen genutzt wird, obwohl es hauptsächlich für ein Publikum von Architekten und Technikern bestimmt ist (HOMBURG U. A., 2020, 364), aber man könnte auch an ein Werkzeug wie Oxygen XML Editor (<https://www.oxygenxml.com> [6.2.2022]) denken, das von vielen im literarischen Bereich tätigen Wissenschaftlern genutzt wird, oder an Software zur Erstellung und Bearbeitung von Ontologien wie Protégé (<https://protege.stanford.edu> [6.2.2022]). Darüber hinaus könnte man sich fragen, ob es nicht sinnvoll wäre, ein Programm, das die Forschung als Teil seines Ziels beinhaltet, auch wenn sie nicht sein einziges Ziel ist, aus der Sicht der Forscher zu rezensieren, um zu verstehen, was dieses Programm in einem bestimmten Forschungsbereich beitragen kann.

Wann sollte eine Rezension verfasst werden?

Ein weiterer Aspekt, der Software stark von anderen Forschungsleistungen unterscheidet, ist ihre ständige Veränderung. Dies wird von den Autoren des Diskussionsbeitrags an mehreren Stellen erwähnt; es wäre jedoch interessant, dieser Frage einen eigenen Abschnitt zu widmen, insbesondere im Zusammenhang mit der Frage nach dem guten „Timing“ einer Rezension. Wann ist es sinnvoll und angemessen, eine bestimmte Software zu rezensieren? Können wir eine Software in Betracht ziehen, die sich zwar technisch gesehen im Betastadium befindet, aber dennoch seit langem im Umlauf ist und auch schon von mehreren Nutzern verwendet wird (in dieser Hinsicht sind die Konzepte des „perpetual beta“ und des sogenannten Bananenprinzips von besonderer Bedeutung; s. z. B. O'REILLY, 2005, 4; HAUSHERR, 1998)? Wann ist es angebracht, eine Rezension zu aktualisieren, weil das Programm in der Zwischenzeit neue Funktionen eingeführt oder seine Funktionsweise geändert hat? Es ist nicht unbedingt möglich, auf jede dieser Fragen eine genaue Antwort zu geben, auch weil einige von ihnen in den Bereich der Einzelfallbewertung fallen, aber es könnte nützlich sein, einige dieser Aspekte explizit zu machen, zumal die Zeit auch eine wichtige Variable ist, um die Nachhaltigkeit einer Software, das Vorhandensein einer aktiven Community (HOMBURG U. A., 2020, 365) und die Effizienz der von den Entwicklern geleisteten Unterstützung (HOMBURG U. A., 2020, 368) zu bewerten.

Forschungssoftware und Forschungsprinzipien

Die Autoren legen zu Recht die Grundpfeiler ihrer Überlegungen fest, indem sie im ersten Teil des Beitrags sowohl das Paradigma der Open Science als auch die FAIR- und CARE-Prinzipien erwähnen (HOMBURG U. A., 2020, 359-361). Sie betonen auch zu Recht die Notwendigkeit, diese Grundsätze über die Bereiche, in denen sie entstanden sind, hinaus auf Forschungssoftware (und ganz allgemein auf alle wissenschaftlichen Praktiken) anzuwenden (s. auch den sehr aktuellen Beitrag von KATZ, GRUENPETER & HONEYMAN [2021], der nach der Veröffentlichung der Handreichung erschienen ist). Vielleicht könnten einige Aspekte noch weiter durchdacht werden, insbesondere die Frage, wie CARE-Prinzipien auf die Programmierung und den Einsatz von Forschungssoftware angewendet werden können – die Autoren deuten das Thema an, entwickeln

es aber nicht weiter, und ethische Prinzipien spielen im anschließenden Fragenkatalog keine besonders wichtige Rolle. Darüber hinaus wäre es sinnvoll, die Verbindung zwischen Software und den FAIR-Prinzipien weiter auszubauen, und zwar nicht nur im Hinblick auf die Zugänglichkeit und Wiederverwendbarkeit der Software selbst, sondern auch in Bezug auf die Daten, die von dieser Software erzeugt werden. Obwohl dieser Aspekt an einigen Stellen behandelt wird, z. B. wenn es um die Interoperabilität und die unterstützten Ausgabeformate geht (HOMBURG U. A., 2020, 366), könnte man auch Fragen stellen wie: Können die Daten einfach in kleinere Dateien/ Ordner unterteilt und strukturiert werden, so dass sie auch von anderen Nutzern leicht eingesehen werden können? Folgen die erzeugten Daten über die verwendete Syntax oder das Format hinaus auch einem bestimmten Datenmodell, das allgemein verbreitet oder zumindest transparent und ausreichend dokumentiert ist? Ist es möglich, den erzeugten Daten schon durch das erstellende Programm Metadaten hinzuzufügen, in den Dateien selbst oder als separate Dateien, möglicherweise in einem Standardformat? Unter diesem Gesichtspunkt lässt sich auch beurteilen, inwieweit die Software die Einhaltung eines Datenmanagementplans im Rahmen eines langfristigen Forschungsprojekts mehr oder weniger einfach macht und ob die erstellten Daten bereits für eine Langzeitarchivierung geeignet sind.

Forschungssoftware für Teamarbeit

Ein Aspekt, dem in der Forschungssoftware nicht viel Aufmerksamkeit geschenkt wurde – wahrscheinlich, weil er auch einer der am schwierigsten zu implementierenden ist – ist die Möglichkeit der Zusammenarbeit zwischen den Nutzern. Auch im Bereich der Forschungssoftware wären z. B. Webanwendungen denkbar, die Teamarbeit ermöglichen, Änderungen bestimmten Mitgliedern korrekt zuordnen und dank bestimmter Algorithmen das Risiko von Überschneidungen oder Konflikten minimieren (einen Überblick über kollaborative Plattformen für die Forschung bietet BEZJAK U. A., 2018, Kapitel 2.7 „*Collaborative Platforms*“, in dem auch das Konzept von „*virtual research environments*“ erörtert wird). Vielleicht wäre es auch sinnvoll, diese Aspekte im Fragenkatalog zu erwähnen und gleichzeitig zu betonen, dass es sich dabei noch um eher seltene Features handelt. In jedem Fall könnte angegeben werden, ob es einfach ist, Daten von einem Gerät auf ein anderes

zu übertragen (nicht nur endgültige Ausgabedaten, sondern auch laufende Projekte, wobei alle wesentlichen Informationen und Einstellungen für die weitere Datenverarbeitung aufbewahrt werden): Sind zum Beispiel die Dateien für ein bestimmtes Projekt leicht auffindbar? Wird eine Kopie der Eingabedaten in den Projektdateien gespeichert (was die Übertragbarkeit von Projekten erleichtert)? Können Probleme entstehen, wenn Daten in der Cloud gespeichert werden (dies kann manchmal zu Kompatibilitätsproblemen und Datenbeschädigung führen)?

Einige weitere Anmerkungen

Ich füge eine Reihe kleinerer Bemerkungen zu bestimmten Punkten bei:

In der abschließenden Eckdatentabelle (HOMBURG U. A., 2020, 369) wäre es nützlich, bei Webanwendungen anzugeben, mit welchen Browsern (und welchen Versionen dieser Browser) die Funktionen der Anwendung vollständig und problemlos genutzt werden können.

Die Verfügbarkeit und Effizienz des technischen Supports (HOMBURG U. A., 2020, 368) könnte nicht nur aus der Sicht des Entwicklers, sondern auch aus der Sicht der Nutzerinnen und Nutzer von Bedeutung sein, da sie auf Probleme stoßen können, die nicht einfach von der breiteren Community gelöst werden können und die das technische Eingreifen der Entwickler selbst erfordern (z. B. Bugs), aber auch, weil die Endnutzerinnen und Endnutzer selbst neue Funktionen vorschlagen können, die dem Forschungsbedarf der wissenschaftlichen Gemeinschaft entsprechen (in diesem Fall ist die Möglichkeit, über eine spezielle Funktion in der Software oder über eine einfache E-Mail-Adresse Feedback zu geben, von grundlegender Bedeutung).

Man könnte sich auch fragen, ob es im Falle von einer Webanwendung möglich ist, eine Kopie der Daten lokal zu speichern. Dies hat auch eine Bedeutung für rechtliche Fragen im Zusammenhang mit dem Datenschutz (HOMBURG U. A., 2020, 366).

Literatur

Bezjak, S., Brinken, H., Clyburne-Sherin, A., Conzett, P., Fernandes, P., Görögh, E., ... Verbakel, E. (2018). *Open Science Training Handbook*. Genf: Zenodo, 4.4.2018. <https://doi.org/10.5281/zenodo.1212495>

Hausherr, T. (1998). Bananenprinzip. In H.-J. Schneider (Hrsg.), *Lexikon der Informatik und Datenverarbeitung* (S. 80). München: Oldenbourg. <https://doi.org/10.1515/9783486795523-005>

Homburg, T., Klammt, A., Hubert, M., Schmid, C., Schmidt, S. C., Thiery, F. & Trognitz, M. (2020). Diskussionsbeitrag: Handreichung zur Rezension von Forschungssoftware in der Archäologie und den Altertumswissenschaften. *Archäologische Informationen*, 43, 357-372. <https://doi.org/10.11588/ai.2020.1.81422>

Katz, D. S., Gruenpeter, M. & Honeyman, T. (2021). Taking a fresh look at FAIR for research software. *Patterns*, 2(3), 100222. <https://doi.org/10.1016/j.patter.2021.100222>

Most, G. W. (Hrsg.). (1999). *Commentaries – Kommentare*. Göttingen: Vandenhoeck und Ruprecht.

O'Reilly, T. (2005). *What is Web 2.0: Design Patterns and Business Models for the Next Generation of Software*. <https://www.oreilly.com/pub/a/web2/archive/what-is-web-20.html> [6.2.2022].

Sahle, P. (2014). *Kriterien für die Besprechung digitaler Editionen* (Version 1.1: unter Mitarbeit von Georg Vogeler und den Mitgliedern des IDE). <https://www.i-d-e.de/publikationen/weitereschriften/kriterien-version-1-1> [6.2.2022].

Sochat, V. (2020). *What is Research Software?*. <https://vsoch.github.io/2020/what-is-research-software> [6.2.2022].

Über den Autor

Massimiliano Carloni, Austrian Centre for Digital Humanities and Cultural Heritage, ÖAW Wien, schloss 2019 seine Promotion in Klassischer Philologie an der Scuola Normale Superiore in Pisa mit einem Kommentar zum „*Philippos*“ des Isokrates ab. Seine Hauptforschungsinteressen liegen in der Entwicklung digitaler Ausgaben antiker Texte und in der Verwendung semantischer Technologien und Metadatenstandards, um große Sammlungen unterschiedlicher Materialien zu beschreiben und zugänglich zu machen. Er interessiert sich insbesondere für graphbasierte Datenmodelle und Linked Open Data. Derzeit ist er Teil des Teams hinter ARCHE, dem digitalen Archiv für Forschungsdaten des ACDH-CH an der Österreichischen Akademie der Wissenschaften.

Massimiliano Carloni, PhD
Austrian Centre for Digital Humanities and Cultural Heritage
Österreichische Akademie der Wissenschaften
Sonnenfelsgasse 19
1010 Wien
Österreich
Massimiliano.Carloni@oeaw.ac.at
<https://orcid.org/0000-0003-2208-5108>