

# Richtlinie zur Entwicklung und zum Umgang mit Forschungssoftware am AWI

## 1. Einleitung

Für fast alle Forschungsdisziplinen ist Software ein unverzichtbarer Bestandteil der wissenschaftlichen Arbeit, sei es für Simulationen, zur Generierung, Verarbeitung, Analyse und Visualisierung von Forschungsdaten oder zur Steuerung von Forschungsgeräten und Experimenten. Viele WissenschaftlerInnen entwickeln eigene Software zur Beantwortung ihrer spezifischen Fragestellungen. Forschungssoftware<sup>1</sup> ist daher nicht nur ein Werkzeug im Prozess der Erkenntnisgewinnung, sondern stellt auch ein eigenständiges Produkt der wissenschaftlichen Arbeit dar. Zusammen mit den Forschungsdaten ist die zugehörige Software die Grundlage der guten wissenschaftlichen Praxis<sup>2</sup>.

Das AWI möchte einen nachhaltigen Umgang mit Forschungssoftware etablieren. Dazu soll die am AWI entwickelte Forschungssoftware im Sinne von Open Science<sup>3</sup> zugänglich gemacht werden, um die Nachvollziehbarkeit und Reproduzierbarkeit wissenschaftlicher Ergebnisse zu ermöglichen. Gleichzeitig soll die Nutzbarkeit von Forschungssoftware gefördert werden. Das AWI folgt dabei den [Empfehlungen zur Implementierung von Leit- und Richtlinien zum Umgang mit Forschungssoftware an den Helmholtz-Zentren](#)<sup>4</sup> und greift Impulse aus der Schwerpunktinitiative „Digitale Information“ der Allianz der deutschen Wissenschaftsorganisationen<sup>5</sup> auf.

Das Dokument wendet sich an wissenschaftliche ProjektleiterInnen, Software-Verantwortliche und EntwicklerInnen am AWI. Die Richtlinie berührt aber auch den Verantwortungsbereich der Stabsstelle Technologietransfer sowie der Rechtsabteilung. Ergänzt wird die Richtlinie durch eine Materialsammlung für EntwicklerInnen, in der einige der hier allgemein formulierten Punkte detaillierter im Hinblick auf die Praxis dargestellt sind und jeweils aktualisiert werden.

## 2. Entwicklungspraxis

Die Praxis von Softwareentwicklung und –dokumentation soll in Abhängigkeit von der Komplexität der Software und dem Nutzerkreis abgestimmt und festgelegt werden. Dazu werden Anwendungsklassen definiert:

---

<sup>1</sup> Zur Definition des Begriffs Forschungssoftware:

„In allen Phasen wissenschaftlichen Arbeitens wird heute Software genutzt, sei es zur Generierung, Verarbeitung, Analyse oder Visualisierung von Forschungsdaten. Wird diese Software hierzu im Rahmen der Forschung eigens entwickelt, spricht man von Forschungssoftware.“

(siehe [https://www.dfg.de/foerderung/info\\_wissenschaft/2019/info\\_wissenschaft\\_19\\_44/index.html](https://www.dfg.de/foerderung/info_wissenschaft/2019/info_wissenschaft_19_44/index.html))

<sup>2</sup> Deutsche Forschungsgemeinschaft DFG (2019): Leitlinien zur Sicherung guter wissenschaftlicher Praxis.

[https://www.dfg.de/download/pdf/foerderung/rechtliche\\_rahmenbedingungen/gute\\_wissenschaftliche\\_praxis/kodex\\_gwp.pdf](https://www.dfg.de/download/pdf/foerderung/rechtliche_rahmenbedingungen/gute_wissenschaftliche_praxis/kodex_gwp.pdf)

<sup>3</sup> Commission's Directorate-General for Research and Innovation (2019), Factsheet Open Science. Online verfügbar unter [https://ec.europa.eu/info/sites/info/files/research\\_and\\_innovation/knowledge\\_publications\\_tools\\_and\\_data/documents/ec\\_rtd\\_factsheet-open-science\\_2019.pdf](https://ec.europa.eu/info/sites/info/files/research_and_innovation/knowledge_publications_tools_and_data/documents/ec_rtd_factsheet-open-science_2019.pdf).

<sup>4</sup> Empfehlungen zur Implementierung von Leit- und Richtlinien zum Umgang mit Forschungssoftware an Helmholtz-Zentren (<https://doi.org/10.2312/os.helmholtz.008>) sowie Muster-Richtlinie Nachhaltige Forschungssoftware an den Helmholtz-Zentren (<https://doi.org/10.2312/os.helmholtz.007>)

<sup>5</sup> Handreichung zum Umgang mit Forschungssoftware (<http://doi.org/10.5281/zenodo.1172970>)

- Anwendungsklasse 0: Software mit geringem Funktionsumfang zum persönlichen Gebrauch, für die eine Weitergabe an andere nicht vorgesehen ist.
- Anwendungsklasse 1: Software, die von an der Entwicklung Unbeteiligten in festgelegtem Umfang genutzt und weiterentwickelt werden kann
- Anwendungsklasse 2: Software, für die eine langfristige Weiterentwicklung und Wartbarkeit sichergestellt werden soll, die aber keinen Produktcharakter<sup>6</sup> hat.
- Anwendungsklasse 3: Kritische Software und Software mit Produktcharakter.

Für die unterschiedlichen Anwendungsklassen möchte das AWI folgende Handlungsempfehlungen als Praxis etablieren, wobei mit steigender Anwendungsklasse die Entwicklung und Dokumentation strikter zu handhaben ist.

- Der Programmcode soll in einem Versionskontrollsystem (z.B. Git) verwaltet werden. Das zugehörige Repositorium soll sämtliche Bestandteile der Forschungssoftware enthalten oder referenzieren, die zu seiner Nutzung notwendig sind.
- Für die Forschungssoftware sollen Tests (einschließlich der dazu notwendigen Test- und Referenzdatensätze) vorbereitet werden, um die Korrektheit der Software prüfen zu können. Die Art und der Umfang der Tests werden dabei in Abhängigkeit von der Softwarekategorie angepasst.
- Stabile Versionen (Releases) der Software müssen für NutzerInnen eindeutig, z.B. über Versions-Nummern oder Tags im Versionskontrollsystem, gekennzeichnet werden.
- Die Forschungssoftware muss dokumentiert werden. Dieses umfasst die Dokumentation von Quelltext sowie die Installations- und Nutzerdokumentation.
- Der Programmcode soll nachvollziehbar implementiert und strukturiert werden. Dabei sind die in der jeweiligen Community geltenden best practices zu beachten.
- Wird die Forschungssoftware von einer Gruppe entwickelt, soll diese die Prozesse für die gemeinsame Entwicklung und Zusammenarbeit verbindlich festlegen und möglichst schriftlich dokumentieren, damit auch neu hinzukommende EntwicklerInnen die abgestimmten Verfahren schnell kennenlernen und anwenden können. Dieses betrifft Themen wie Dokumentation, Teststrategien, Entwicklungsmethoden, kollaborative Nutzung von Werkzeugen, Definition der Entwicklungszyklen und Kommunikationskanäle.

Zur Unterstützung der EntwicklerInnen stellt das AWI eigene Infrastruktur- und Beratungsangebote (wie Versionskontrollsysteme, Möglichkeiten der Softwarepublikation ...) zur Verfügung bzw. berät bei der Nutzung von entsprechenden externen Infrastrukturen (z.B. bei Helmholtz-übergreifenden Dienstangeboten).

### 3. Software als Ergebnis und Ausdruck von wissenschaftlicher Arbeit

Die Entwicklung von Forschungssoftware ist Teil eines kreativen Prozesses und ist in diesem Sinne ausführbares Wissen. Die Entwicklung von Forschungssoftware ist eine intellektuelle und urheberrechtlich geschützte Leistung. Das AWI unterstützt die Aktivitäten in der HGF und darüber hinaus, die Arbeit von SoftwareentwicklerInnen in der Wissenschaft in Evaluierungen zu erfassen und anzuerkennen. SoftwareentwicklerInnen werden ermuntert, die am AWI entwickelte Software in dazu geeigneten Journals<sup>7</sup> zu publizieren und damit zitierbar zu

---

<sup>6</sup> Unter Software mit Produktcharakter ist Software gemeint, die in einem produktiven Kontext genutzt wird (z.B. bei Expeditionen) und deren Funktionstüchtigkeit entscheidend ist.

<sup>7</sup> Z.B. Journal of Open Research Software, The R Journal, Computer & Geosciences, SoftwareX ...

machen. Im Gegenzug muss als Teil der guten wissenschaftlichen Praxis bei Publikationen die verwendete Software (soweit möglich) angemessen zitiert werden.

Bei größeren Softwareprojekten, die frei zugänglich gemacht werden, sollte über eine eigene Internetseite und dort installierte Monitorsysteme (wie zum Beispiel Matomo) die Anzahl der Webseitenbesuche und der Softwaredownloads sowie weiterer Informationen aufgezeichnet werden, um Rückschlüsse über Größe und Zusammensetzung der Nutzergruppe zu ziehen (wichtiger Parameter für den Helmholtz-Wissenstransfer).

### 3.1. Qualitätssicherung

Da die Qualität der Forschungssoftware in vielen Fällen unmittelbar Einfluss auf die Qualität der wissenschaftlichen Arbeit hat, kommt der Qualitätssicherung bei der Softwareentwicklung eine große Rolle zu. Die Umsetzung und Ausgestaltung einer guten Softwareentwicklungs- und Dokumentationspraxis erfordert dabei entsprechende personelle und zeitliche Ressourcen, die in der Planung von den Verantwortlichen zu berücksichtigen sind

### 3.2. Weiterbildung und Karrierewege

Die Qualität der Forschungssoftware hängt stark von den vorhandenen Kenntnissen und Fähigkeiten der EntwicklerInnen ab. Daher unterstützt das AWI die EntwicklerInnen in ihrem Bemühen um Weiterbildung und Vernetzung durch Bereitstellung von entsprechenden Angeboten im Institut bzw. zentrenübergreifend in der Helmholtz-Gemeinschaft (wie z.B. Software Carpentries) einerseits und durch die Ermöglichung der Teilnahme an solchen regelmäßigen Angeboten für die MitarbeiterInnen zu Programmier-Techniken, Methoden des Software Engineering und weiteren Aspekte der Softwareentwicklung andererseits.

Um längerfristig in den Arbeitsgruppen die notwendige Expertise für Softwareentwicklung aufbauen und halten zu können, sollten entsprechende Karrierewege geschaffen werden und die Leistungen von MitarbeiterInnen in der Softwareentwicklung gewürdigt werden.

### 3.3. Nachvollziehbarkeit von Ergebnissen, Zugänglichmachung von Software

Um die Nachvollziehbarkeit, Verifizierbarkeit und Reproduzierbarkeit von wissenschaftlichen Ergebnissen gemäß den Regeln der guten wissenschaftlichen Praxis zu gewährleisten, ist die Software zugänglich zu machen. Das AWI unterstützt die EntwicklerInnen von Forschungssoftware durch die Bereitstellung entsprechender Angebote im Haus (Publikationsrepository EPIC.awi.de) und der Beratung zur Nutzung externer Dienste.

So bietet der von der Europäischen Kommission finanzierte Dienst zenodo.org die Möglichkeit, wissenschaftliche Software mit einer konkreten Fassung (Snapshot, Version) als ZIP Archiv dauerhaft zu speichern. Auf zenodo.org veröffentlichte Software erhält einen zitierfähigen DOI. Durch entsprechende Schnittstellen können Releases auf github oder gitlab automatisiert auf zenodo veröffentlicht werden. Dabei soll auf zenodo unbedingt die Zugehörigkeit zum AWI unter Affiliation angegeben werden für eine Zuordnung der Software zum AWI. Die Metadaten der Publikation sind von den AutorInnen in das AWI Research Information System elements.awi.de zu importieren, um eine Erfassung zitierbarer und mit einem persistenten Identifier versehener Software für den Indikator Softwarepublikation zu ermöglichen, der ab dem Berichtsjahr 2022 im Rahmen der Programmorientierten Forschung in der HGF eingeführt wurde. Außerdem kann damit die so publizierte Software auch dem persönlichen CV hinzugefügt werden.

#### 4. Nachnutzung von Forschungssoftware

Die Entwicklung qualitativ hochwertiger Software erfordert den Einsatz entsprechend hoher Ressourcen (Zeit, Personal). Nachhaltige Softwareentwicklung soll daher eine Nachnutzung von Forschungssoftware ermöglichen, indem auf bereits vorhandene Lösungen aufgebaut werden kann.

Software ist das Ergebnis kreativer Arbeit und wird durch das Urheberrechtsgesetz geschützt. Für Software, die weitergegeben oder zugänglich gemacht wird (also Anwendungsklassen 1-3), muss daher die Nutzung durch Dritte mit einer entsprechenden Lizenz geregelt werden. Die Wahl einer geeigneten Lizenz hängt von einer Reihe unterschiedlicher Faktoren ab und sollte möglichst früh im Entwicklungsprozess betrachtet werden, um Einschränkungen oder Konflikte durch Einbindung fremder Software zu vermeiden.

Das AWI unterstützt daher grundsätzlich die Einräumung von Nutzungsrechten an Software gegenüber Dritten, wobei zwar im Sinne von Open Science eine offene Lizenz empfohlen wird, kommerzielle Verwertungsmöglichkeiten aber nicht ausgeschlossen werden sollten. Eine entsprechende Beratung zu den Optionen, Chancen, Risiken, Möglichkeiten der Unterstützung und frühzeitig zu bedenkender Faktoren kann und sollte in der Stabsstelle Technologietransfer angefragt werden.

Bei den offenen Lizenzen wird empfohlen, soweit wie möglich auf bereits verbreitete (und bewährte) Lizenzmodelle zurückzugreifen, die grob in zwei Kategorien unterteilt werden:

- **Copyleft:**  
Sie erlauben die Nutzung, Änderung und Weitergabe des Codes, wobei die Lizenzbedingungen unverändert mitgegeben werden müssen. Wichtigster Vertreter ist die GPL ("GNU General Public License")<sup>8</sup>, die ein striktes copyleft fordert: wenn man GPL-lizensierten Code in einem Programm verwendet, muss dieses ebenfalls unter die GPL-Lizenz gestellt werden. Eine Weitergabe nur eines kompilierten Programms ohne Zugang zum Quellcode ist nicht erlaubt. Damit sichert eine solche Lizenz, dass die Software weiterhin Open Source bleibt und ermöglicht allen Entwicklern den Zugang zu Weiterentwicklungen und Verbesserungen des Codes.  
Weniger strikt ist die „GNU Lesser General Public License“ (LGPL)<sup>9</sup>, die vor allem für Bibliotheken verwendet wird. Sie hat ein beschränktes Copyleft, so dass sie die Verlinkung der Bibliotheken sowohl mit Open Source Software als auch mit proprietärer Software ermöglicht.
- **Non-Copyleft:**  
Die auch unter dem Namen permissive license zusammengefassten Lizenzen dieser Gruppe erlauben, dass Weiterentwicklungen der Software unter anderen Lizenzbedingungen freigegeben werden als die Software selbst, solange das originale Copyright und Lizenz mitgegeben werden. Damit kann der Lizenznehmer veränderte Versionen der Software unter beliebigen Lizenzbedingungen weiterverbreiten, also auch in proprietäre Software überführen. Wichtigste Vertreter dieser Gruppe sind die MIT Lizenz<sup>10</sup> und die Apache Lizenz<sup>11</sup>, wobei die MIT Lizenz die größeren Freiheiten

---

<sup>8</sup> <https://opensource.org/licenses/GPL-3.0>

<sup>9</sup> <https://opensource.org/licenses/LGPL-3.0>

<sup>10</sup> <https://opensource.org/licenses/MIT>

<sup>11</sup> <https://opensource.org/licenses/Apache-2.0>

einräumt. Die Lizenz zielt vor allem darauf ab, den Urheber der Software im Fall von Fehlern in der Software zu schützen.

Im Zuge der weiteren Digitalisierung in der Wissenschaft kommt der Forschungssoftware eine hohe Relevanz zu. Insbesondere für die zunehmende zentrumsübergreifende Arbeit sollen die vorliegenden Richtlinien die Qualität der Entwicklung von Software dauerhaft sichern und die am Entwicklungsprozess beteiligten Mitarbeiter unterstützt werden, um weiterhin gute Forschung zu ermöglichen.