

Entwurf eines Metadatenrepositoriums zur Erfassung technischer Nachhaltigkeit von Forschungssoftware

Stephan Druskat

Institut für deutsche Sprache und Linguistik
Humboldt-Universität zu Berlin

stephan.druskat@hu-berlin.de

Helmholtz Open Science Workshop “Zugang zu und Nachnutzung von wissenschaftlicher Software”
HZDR, Dresden, 22. November 2016

Inhalt

① Vernachhaltung von Forschungssoftware

② Metadatenrepositorium

③ Fazit

Vernachlässigung von Forschungssoftware

Warum? Wie?

Vernachlässigung - Warum?

Verschiedene Motivlagen¹

- "Der edle Forscher"
- Proxy-Vernachlässigung
- Persönliche Gründe

Zusätzlich:

- Forschungssoftware vs. "Verbrauchersoftware"
- Interesse seitens der Förderer
(*Softwaremanagementpläne*)

¹Vgl. M. Turks Vortrag beim WSSSE3: <https://youtu.be/kHcL8WSpbhQ>

Vernachlässigung - Wie?

“**Zugang zu** und **Nachnutzung von** wissenschaftlicher Software”

Zugänglichkeit:

- Sichtbarkeit

Nachnutzbarkeit:

- Lauffähigkeit
- Anwendbarkeit
- Anpassbarkeit

Vernachhaltung - Wie?

“**Zugang zu** und **Nachnutzung von** wissenschaftlicher Software”

Zugänglichkeit:

- Sichtbarkeit
- **Sichtbarkeit von Nachnutzbarkeit**

Nachnutzbarkeit:

- Lauffähigkeit
- Anwendbarkeit
- Anpassbarkeit

Vernachlässigung - Wie?

“**Zugang zu** und **Nachnutzung von** wissenschaftlicher Software”

Zugänglichkeit:

- Sichtbarkeit
- **Sichtbarkeit von Nachnutzbarkeit**

Forschungssoftwaremetadatenrepositorium

Nachnutzbarkeit:

- Lauffähigkeit
- Anwendbarkeit
- Anpassbarkeit

Inhalt

① Vernachlässigung von Forschungssoftware

② Metadatenrepositorium

③ Fazit

Entwurf

Ziel:

- Repository, das Metadaten zu Forschungssoftware darstellt, auf deren Grundlage *Aussagen zur potenziellen Nachnutzbarkeit und weiteren Nachhaltigkeit* getroffen werden können
- Bereitstellung von **Maßen** über Nachnutzbarkeit und Nachhaltigkeit der Software

Herausforderungen:

- ① Operationalisierbare **Definition Nachhaltigkeit** (vs. Nachnutzbarkeit);
- ② Definition, Katalogisierung und Kategorisierung von **Nachhaltigkeitskriterien**;
- ③ Entwurf genauer, reproduzierbarer, manipulationssicherer und nachvollziehbarer **Maße**;
- ④ Entwurf von **Berechnungsalgorithmen** für Maße

State of the art

Code	Metadaten	Forsch.softw.	Forsch.softw.metadaten
GitHub	Open Hub	{Forschungsdatenrepos}	DiRT Directory
Bitbucket		Zenodo	SciencePAD
...		{Disziplinrepos}	EGI Applications DB
			...

Keine expliziten Daten zur
Nachnutzbarkeit/Nachhaltigkeit!

Definitionen

Drei Ziele:

- Sicherung der Existenz der Software
- Erhaltung ihres Produktivpotentials
- Schaffung und Erhalt ihres Potenzials für Weiterentwicklung und Adaption

Nachhaltigkeit geht über *Nachnutzbarkeit* (definierbar als: *technische/funktionelle Nachhaltigkeit*) hinaus, da auch nicht-technische Aspekte eine Rolle spielen (Menschen, Förderung, Community, Relevanz, ...).

Kriterien

- Erstellen/Erweitern von Kriterienkatalogen (vgl. CodeMeta, Kriterienbasiertes Assessment des Software Sustainability Institute, OntoSoft, ...)
- Evaluierung und Gewichtung der Kriterien (z.B. auch durch Crowdsourcing)
- Klassifizierung (Nachhaltigkeitsziele, Objektivität ↔ Quantifizierbarkeit)

Beispiele:

Lizenz, Dokumentation, Förderstatus, Sprache, Tests, Modularität, Entwicklercommunity (Personenstunden, Forks, Commits, ...), CI, Veröffentlichungen und Zitate, *Feedback*, ...

Maße

Kriterien können anhand ihrer **Objektivität** und **Quantifizierbarkeit** kategorisiert werden.

Daher: **Drei Maße für Nachhaltigkeit**

Typ	obj	quant	repr	manip	Beispiel
“Hart”	✓	✓	✓	✓	Lizenz, Förderstatus
“Halbhart”	✓		✓	✓	Buildsystem, Personalstruktur
“Weich”					Intuitives UI

obj: objektiv, *quant*: quantifizierbar, *repr*: reproduzierbar, *manip*: manipulationssicher

Ein Nachhaltigkeitsmaß erfasst ausschließlich (harte) technische Kriterien, die die Nachhaltigkeit beeinflussen.

Interaktivität

- **Begutachtung** der Software, **Erweiterung/Korrektur** von Kriterien
- **Bewertung/Beurteilung** spezifischer Metadatenpunkte (Beitrag zum “weichen” Maß)
- **Dokumentation** von Verwendung (Zitate!) und Entwicklung (Beitrag zu härteren Maßen)



7



Datengewinnung

- **Nutzereingabe** (Softwareherausgeber)
- **Harvesting** aus existierenden Repositorien (z.B. via GitHub und Open Hub APIs)
- **Crawling** von Coderepositorien (auch erste Verifizierung/Quantifizierung)

Inhalt

① Vernachlässigung von Forschungssoftware

② Metadatenrepositorium

③ Fazit

Fazit

- Potenziell geeignetes Werkzeug zur **Erleichterung des Zugangs** zu Forschungssoftware
- Macht **Nachnutzbarkeit sichtbar** und nachvollziehbar
- **Fördert** implizit **Nachnutzung** und Schaffung nachhaltiger(er) Software

Herausforderungen:

- Bestenfalls Verschränkung mit anderen Tools bzw. **Einbettung in Policies** (*Softwaremanagementpläne!*)
- Annahme durch die Community

Danke!

Danke für die Aufmerksamkeit!

`stephan.druskat@hu-berlin.de`

 @stdruskat