



POTSDAM INSTITUTE FOR
CLIMATE IMPACT RESEARCH

Von Closed zu Open Source

deRSE19

Lavinia Baumstark

5. Juni 2019

Warum Open Source?



Warum Open Source?

- + **Verbesserte Zuverlässigkeit der Software**
- + **Verbesserte Akzeptanz der Software**
- + **Erhöhte Sichtbarkeit der Entwickler:innen**

Von Closed Source ...

**Fokus: große, über viele Jahre hinweg, mit vielen Leuten
entwickelte Modelle**

➤ **Nie für Open Source gedacht**

Von Closed Source ...

Fokus: große, über viele Jahre hinweg, mit vielen Leuten entwickelte Modelle

➤ **Nie für Open Source gedacht**

Vorbehalte von unseren Entwickler:innen-Gruppen:

- **Falsche Nutzung durch Externe**
- **Falsche Interpretation durch Externe**

Von Closed Source ...

Fokus: große, über viele Jahre hinweg, mit vielen Leuten entwickelte Modelle

➤ **Nie für Open Source gedacht**

Vorbehalte von unseren Entwickler:innen-Gruppen:

- **Falsche Nutzung durch Externe**
- **Falsche Interpretation durch Externe**

Neue Entwicklungen:

- **Transparenz gewünscht**
- **Verstärkte Kooperation mit Externen**

➤ **Rahmenbedingungen nötig**

... zu Open Source



Potsdam Institute for Climate Impact Research (PIK) e.V.

Open Source and Open Data Policy

PIK supports the goals of the Berlin Declaration on "Open Access to Knowledge in the Sciences and Humanities"

Our mission of disseminating knowledge is only half complete if the information is not made widely and readily available to society. New possibilities of knowledge dissemination not only through the classical form but also and increasingly through the open access paradigm via the Internet have to be supported. We define open access as a comprehensive source of human knowledge and cultural heritage that has been approved by the scientific community.

In order to realize the vision of a global and accessible representation of knowledge, the future Web has to be sustainable, interactive, and transparent. Content and software tools must be openly accessible and compatible.

(<https://openaccess.mpg.de/Berliner-Erklaerung>)

Open Source

To further transparency, collaboration and code re-use PIK supports the publication of Open Source software developed at the institute, such as

- software tools,
- model code,
- contributions to existing Open Source projects that build the foundations for science.

Neue **Open Source Policy** am PIK:

- **Berliner Erklärung**
- **Checkliste für kleine Programme & Modelle**
- **Workflow für große Modelle (5 Schritte)**
- **Mitarbeit an existierenden Open Source Projekten geregelt**



1. Formale Kriterien überprüfen

Check-Liste für Modelle:

- ✓ Alle Entwickler:innen haben einen Arbeitsvertrag mit dem Institut
- ✓ Es gibt keine externen Entwickler:innen oder alle externen Entwickler:innen sind einverstanden





Check nach proprietären Produkten:

- ✓ Alle mit veröffentlichten Daten/Modulen sind frei zugänglich oder
- ✓ Die Bedingungen für eine Veröffentlichung von mitgelieferten Daten/Modulen sind mit den Anbietern geklärt (gilt auch für unvereinbare andere Open Source Lizenzen)

2. Modellentwicklung und -veröffentlichung

Strategie zur Modellentwicklung und -veröffentlichung:

- ✓ Es existiert ein Versionsmanagement
 -  GitHub: erlaubt die gemeinsame Entwicklung mit Externen
 -  GitLab: Entwicklung in privatem Repository möglich

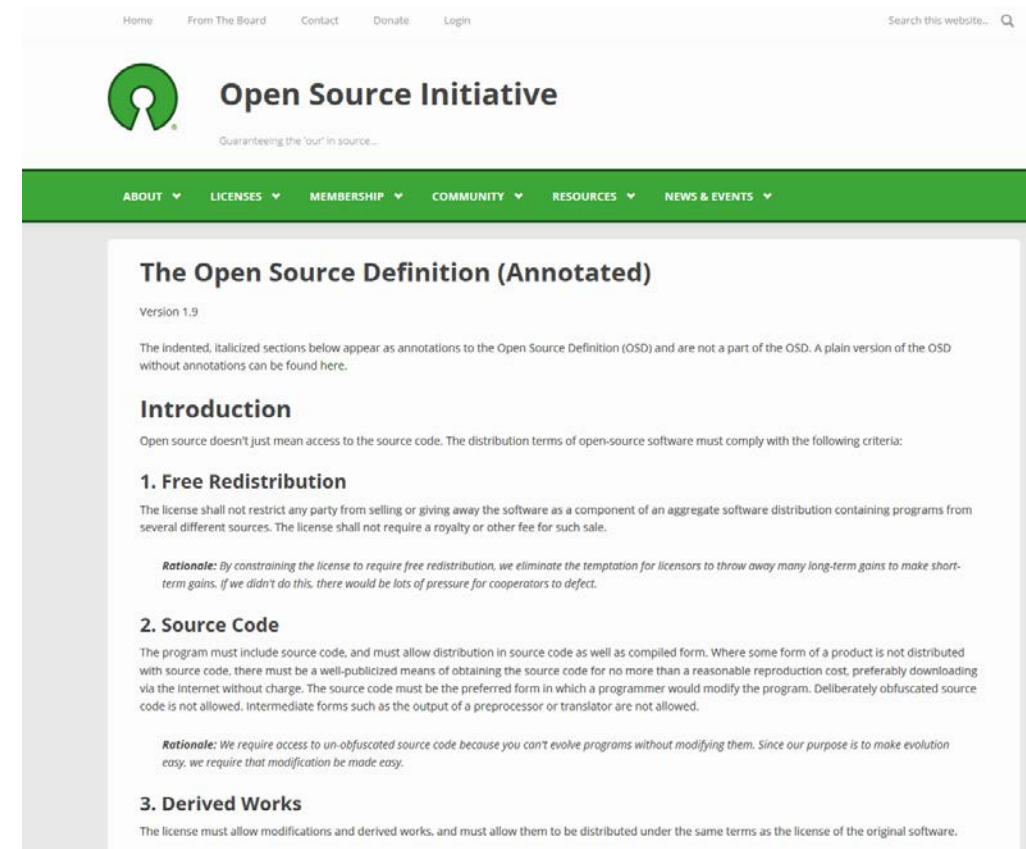
3. Lizenz wählen

Eine Open Source Lizenz bietet legale Rahmenbedingungen für die weitere Nutzung und Verbreitung von Software

Viele unterschiedliche Lizenzen mit unterschiedlichen Restriktionen:

© **Copyleft:** Weiterentwicklungen müssen unter den gleichen Bedingungen veröffentlicht werden (z.B. GNU Affero GPL)

freiere Lizenz: Weiterentwicklungen könnten auch proprietär werden (z.B. 2-clause BSD)



3. Lizenz wählen

Arbeitsschritte:



1) **Download** des vollen Textes der Lizenz und **Speicherung** im Hauptordner mit dem Namen LICENSE oder COPYING



2) **Copyright Notiz** hinzufügen



3) **Autoren** des Projekts in der copyright Notiz oder AUTHORS Datei

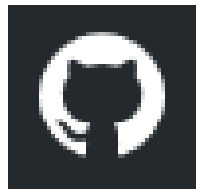


4) **Anfang jeder Quelldatei** des Projekts die copyright Notiz einfügen

3. Lizenz wählen

Contributors Agreement:

- Bedingungen, unter denen Beiträge zum Projekt gehandhabt werden
- alle Entwickler:innen und Besitzer der Rechte sich über die spätere Verwertung von Beiträgen einig
- Konflikte werden vermeiden



Bei Github kann bei jedem push request überprüft werden, ob ein Entwickler ein Contributor Agreement unterzeichnet hat (falls nicht, kann die Zustimmung über Github vorgenommen werden)

4. Dokumentation

- Herausforderung: Dokumentation und Quellcode konsistent halten
- Dokumentation sollte kein zu großer zusätzlicher Aufwand für die Entwickler:innen sein



4. Dokumentation

- Herausforderung: Dokumentation und Quellcode konsistent halten
- Dokumentation sollte kein zu großer zusätzlicher Aufwand für die Entwickler:innen sein



Arten von Modelldokumentation:

1. User manuals: Installationsanweisungen, einfache Anwendungsbeispiele, einfache „getting started“ für neue Nutzer:innen
2. Mathematische Modellbeschreibung
3. Modellimplementierungsdokumentation: Brücke zwischen Programmiersprache und menschlicher Sprache (z.B. Doxygen, Roxygen2, Sphinx)

➤ **Dokumentation als Teil des Quellcodes**



- Falls Dokumentation getrennt von Quellcode -> geeignete Lizenz

5. Finaler Check durch die Verwaltung

- ✓ Nutzungs- und Verwertungsrechte sind geklärt
- ✓ Vertragsbedingungen aller Beitragenden sind geklärt
- ✓ Projektbedingungen sind geklärt
- ✓ DOI vergeben um eine Referenzierbarkeit zu ermöglichen

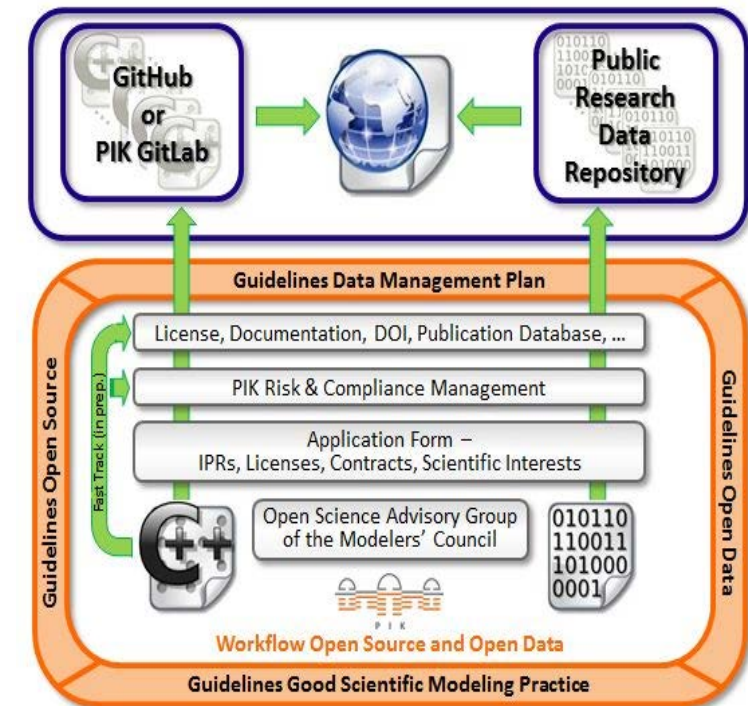
von Closed Source

...

zu Open Source

5 Schritte von Closed zu Open Source Software:

1. Formale Kriterien überprüfen
2. Strategie zur Modellentwicklung und –veröffentlichung
3. Lizenz wählen
4. Dokumentation
5. Finaler Check durch die Verwaltung



REMIND ...

REMIND momentan „nur“
transparent

Vorbereitungen, dieses Jahr
Open Source zu gehen

```
#####
#####
##### REMIND #####
#####
*** This is the REMIND model ("Regional Model of Investments and Development"),
*** developed by Research Department III - "Sustainable Solutions"
*** of the Potsdam Institute of Climate Impact Research (PIK)
*** https://www.pik-potsdam.de/research/sustainable-solutions/models/remind
#####
##### ATTENTION ##### ATTENTION ##### ATTENTION ##### ATTENTION #####
#####
*** There is no final decision about the REMIND license.
*** We publish the source code of REMIND to enable transparency and review,
*** but exclude operation of the model for research and commercial applications, distribution and any other use.
*** The source code may be copied for the sole purpose of reading.
#####
##### ATTENTION ##### ATTENTION ##### ATTENTION ##### ATTENTION #####
#####

* ##### CODE MERGED WITH FUNCTION singleGAMSfile (LIBRARY lucode) #####

##### R SECTION START (VERSION INFO) #####
*
* Regionscode: e4ac4a0fd71c9dc8bd5d87d752438894
*
* Last modification (input data): Tue Sep 15 14:59:22 2015
*
##### R SECTION END (VERSION INFO) #####

-----
*** main.gms: main file. welcome to remind!
-----

*** preliminaries:
-----
*** allow empty data sets:
$onempty
*** create dummy identifier to fill empty sets:
$phantom null
*** include unique element list:
$onuellist
*** include $-statements in listing
$dollardollar
*** include end-of-line comments
$ONeolcom
*** remove the warnings for very small exponents (x**=-60) when post-processing
$offdigit

-----
*** ..... basic scenario choices
-----
```

MAgPIE modell auf GitHub: <https://github.com/magpiemodel/magpie>



A screenshot of the GitHub repository page for 'Model of Agricultural Production and its Impact on the Environment'. The page shows the repository name, a description, and statistics for repositories, people, and projects. It also features a 'Grow your team on GitHub' banner and a list of repositories including 'magpie' and 'tutorials'. The 'magpie' repository is highlighted with a green pulse icon, indicating recent activity. The 'tutorials' repository is also listed with a green pulse icon. The page includes a search bar, navigation links, and a 'Sign up' button.

MAgPIE Dokumentation:

<https://rse.pik-potsdam.de/doc/magpie/4.1/index.htm>

MAgPIE Model Documentation | Version 4.1
created with goxygen 0.25.0

- Overview
- 09_drivers
- 10_land
- 11_costs
- 12_interest_rate
- 13_lc
- 14_yields
- 15_food
- 16_demand
- 17_production
- 18_residues
- 20_processing
- 21_trade
- 30_crop
- 31_past
- 32_forestry
- 34_urban
- 35_natveg
- 38_factor_costs
- 39_landconversion
- 40_biomass

MAgPIE - Modelling Framework

The *Model of Agricultural Production and its Impact on the Environment* (MAgPIE) is developed and used to assess the competition for land and water and the associated consequences for sustainable development under future scenarios of rising food, energy and material demand as well as production, climate change impacts and greenhouse gas mitigation and different land related policies (J. P. Dietrich et al. (2019)).

MAgPIE is a global partial equilibrium model of the land-use sector that operates in a recursive dynamic mode and incorporates spatially explicit information on biophysical constraints into an economic decision making process (H. Lotze-Campen et al. (2008)). It takes regional economic conditions such as elastic demand for agricultural commodities, technological development and production costs as well as spatially explicit data on biophysical constraints into account. Geographically explicit data on biophysical conditions are provided by the Lund-Potsdam-Jena managed land model (LPJmL) (A. Bondeau et al. (2007), Müller C. and Robertson (2014)) on a 0.5 degree resolution and include e.g. carbon densities of different vegetation types, agricultural productivity such as crop yields and water availability for irrigation. Based on the strong interaction with LPJmL, MAgPIE can also help to assess the consequences of climate change and increased CO₂ in the atmosphere on the land-use sector (Stevanovic et al. (2016)).

- MAgPIE - Modelling Framework
 - Authors
 - Bibtex format
 - Citation File Format
 - References

mit R-Paket „goxygen“

<https://github.com/pik-piam/goxygen>

MAgPIE Model Documentation | Version 4.1
created with goxygen 0.25.0

- Overview
- 09_drivers
- 10_land
- 11_costs
- 12_interest_rate
- 13_lc
- 14_yields
- 15_food
- 16_demand
- 17_production
- 18_residues
- 20_processing
- 21_trade
- 30_crop
- 31_past
- 32_forestry
- 34_urban
- 35_natveg
- 38_factor_costs
- 39_landconversion
- 40_transport
- 41_area_equipped_for_irrigation
- 42_water_demand
- 43_water_availability
- 45_climate
- 50_nr_soil_budget
- 51_nitrogen
- 52_carbon
- 53_methane
- 54_phosphorus
- 55_awms
- 56_ghg_policy
- 57_maccs
- 59_som
- 60_bioenergy
- 62_material
- 70_livestock
- 71_disagg_ivst
- 80_optimization

Food demand (15_food)

Description

The main function of the food demand module is to forecast the food demand from agricultural products, which is used in the module 16_demand.

Interfaces

- Food demand (15_food)
 - Description
 - Interfaces
 - Input
 - Output
 - Realizations
 - (A) anthropometrics_jan18
 - Definitions
 - Objects
 - Sets
 - Authors
 - See Also
 - References

Erfahrungen

- Mehr Interaktion mit externen Nutzern
- Mehr Wahrnehmung in der Community
- Neue Wahrnehmung als Open-Source-Model in der Öffentlichkeit
- Supportanfragen
- Nachfrage nach Workshops und Tutorials



Vielen Dank für Eure Aufmerksamkeit

rse@pik-potsdam.de

