



Scientific Open-Source Meetup

Как и зачем делать научный открытый код

Никитин Николай,
Руководитель направления открытого кода, с.н.с,
ИЦ «Сильный ИИ в промышленности»

Санкт-Петербург – 16.12.2022

Что такое открытый код?



Open Source — подход, разработки программ, при котором исходный код остается доступным и свободным для использования для всех.

Пишут и поддерживают open-source проекты:

- Разработчики-энтузиасты, исследователи. Поодиночке или в группах;
- Команды компаний или исследователей. Сопровождение проектов является частью (или полностью) должностных обязанностей.

Мотивация:

- Любопытство / хобби (“pet project”);
- Возможность набраться опыта, попробовать новый технологический стек;
- Open-source проекты служат демонстрацией навыков разработчика и команды; Такие проекты служат своеобразной «витриной»;
- Развитие проекта с целью последующей монетизацией разработок.

Открытый код в научных задачах



Проектирование и реализация проекта позволяет структурировать подходы - приучает держать "все в одном месте"



Готовый прототип проще и быстрее можно адаптировать под бизнес-задачи, чем «абстрактную» статью.



Open-source любят все - ученые, инженеры и разработчики.

Каждый ученый, который будет пользоваться вашим алгоритмом или моделью в своих исследованиях с высокой вероятностью вас процитирует.

ИТМО

Помогает подтвердить (в глазах рецензента в первую очередь) достоверность ваших исследований. А значит, статью с большей вероятностью примут.

Чем научный open-source отличается от «кода к статье»



Открытое ПО – это не только код! Чтобы им пользовались - нужны тесты, документация, примеры и общение с сообществом.

- **Примеры** должны быть понятны стороннему пользователю. Они должны показывать ему, как применить инструмент для новых задач – а не только тех наборов данных, которые использовали авторы.
- **Тесты** облегчают внесение изменений, а также в целом показывают что проект «жив». Сломанные тесты в основной ветке – плохой признак.
- **Общение:** открытые issue позволяют новым разработчикам влиться в проект, pull-request-ы – понять направление развития проекта. Не лишним будет и телеграм-канал.

Сценарий научной OS-разработки



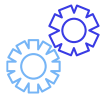
Идея проекта



Проверка идеи (например, через MVP)



Формирование команды



Разработка и
сопровождение проекта с
учетом особенностей open-
source



Подготовка материалов для
исследований, апробация
подходов и написание научных
статей (и не только)

Где хранить открытый код?



Bitbucket



gitee

gitee



GitLab



beanstalk

Beanstalk

Другие альтернативы : GitHub Alternatives

Подобные хранилища интегрированы с системами контроля версий, а также дополнительными инструментами непрерывного развертывания. Такой инструментарий позволяет эффективно работать над проектом команде из нескольких разработчиков одновременно.

При этом, код на этих платформах можно хранить и не показывая никому извне.

Немного рекомендаций (1/2)



- Разберитесь, как работать с git (commit, pull, push и т.д.);
- Оформите readme: напишите что делает ваш инструмент, для чего он нужен, оставьте контакты для связи;
- Организуйте процедуру code review (даже если вы работаете в одиночку, - это тоже можно делать);
- Тестируйте то, что вы создаете. Желательно делать это автоматически (см. Github actions);
- Подумайте как будете следить за задачами и их выполнением (делать это можно через github issues);

Немного рекомендаций (2/2)



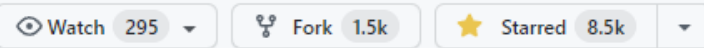
- Делайте так, чтобы установить модуль и запустить программу можно было как можно проще. Отчасти, в этом поможет хорошо оформленная документация и примеры.
- Чем полнее и точнее будет документация - тем проще вам будет сопровождать проект (это главное) и тем приятнее пользователям будет ставить ему звезды (это уже второстепенное).
- Если данные слишком большие или не хочется их показывать все в интернете, то сгенерируйте небольшой синтетический набор и сделайте примеры на нём - лишь бы было видно, что оно работает (и у пользователя была возможность это проверить).
- Изучите руководства - например "Как принять участие в работе Open Source проектов на GitHub. Краткое руководство для начинающих" и адаптируйте успешные практики под себя.

Истории успеха – открытый код, созданный учеными



Существует большое количество предметных моделей, которые были созданы не профессиональными программистами, а специалистами-предметниками. Равно как и хороших известных продуктов, созданными исследователями из компьютерных наук

Продукт: AutoML-фреймворк TPOT (Tree-based Pipeline Optimization Tool)



Создатели: Epistasis Lab, исследовательская группа в Седарс-Синайский медицинском центре (США)

Специализация группы - стык биологии, медицины и компьютерных наук

Продукт: PCSE (Python Crop Simulation Environment)



Создатели: исследователи из Вагенингенского университета (Нидерланды).

Специализация группы - сельскохозяйственные науки

Пример академического open-source для ИТМО

«теоретиков»

pycbc - Core package to analyze gravitational-wave data, find signals, and study their parameters. This package was used in the first direct detection of gravitational waves (GW150914), and is used in the ongoing analysis of LIGO/Virgo data.



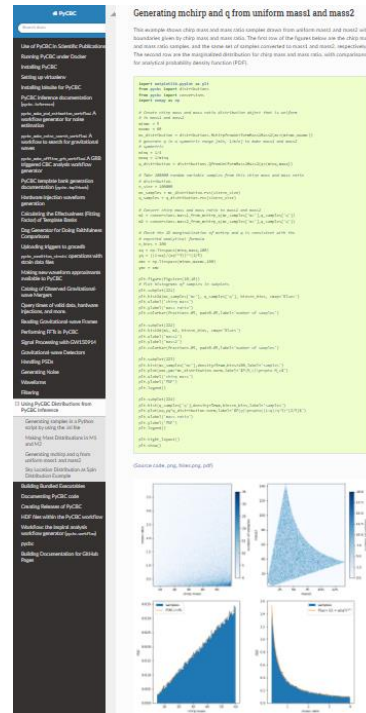
PyCBC Inference: A Python-based parameter estimation toolkit for compact binary coalescence signals

CM Biwer, CD Capano, S De, M Cabero... - Publications of the ..., 2019 - iopscience.iop.org

We introduce new modules in the open-source PyCBC gravitational-wave astronomy toolkit that implement Bayesian inference for compact-object binary mergers. We review the Bayesian inference methods implemented and describe the structure of the modules. We demonstrate that the PyCBC Inference modules produce unbiased estimates of the parameters of a simulated population of binary black hole mergers. We show that the parameters' posterior distributions obtained using our new code agree well with the ...

☆ Сохранить 99 Цитировать Цитируется: 142 Похожие статьи Все версии статьи (9) Web of Science: 80

[PDF] iop.org



Пример долгоживущего научного проекта



NEMO Ocean – модель для динамики океана, создаваемая консорциумом научных институтов. Разработается с 2003 года.

Сайт: <https://www.nemo-ocean.eu/>

About NEMO

NEMO standing for "Nucleus for European Modelling of the Ocean" is a state-of-the-art modelling framework for research activities and forecasting services in ocean and climate sciences, developed in a sustainable way by a European consortium.

NEMO is distributed under [a GNU GPL compatible free licence](#)

Репозиторий:
<https://forge.nemo-ocean.eu/nemo/nemo>

Регулярные встречи: <https://www.nemo-ocean.eu/>

NEMO at EGU 2020 and NEMO Users meeting

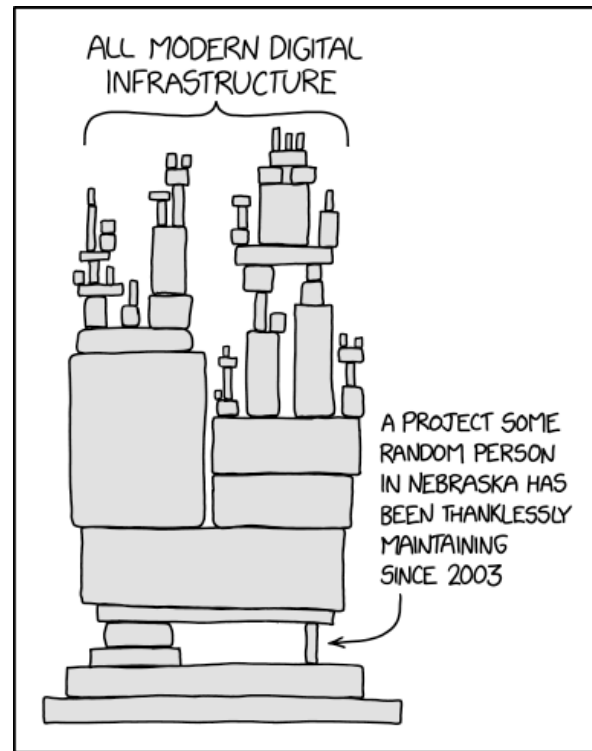
The EGU 2020 provisional program includes a session which might be of interest for you: **OS4.8/CL5.18/ESS11.2 Numerical modelling of the ocean: new scientific advances in ocean models to foster exchanges within NEMO community and contribute to future developments (co-sponsored by NEMO Consortium)**

Особенности открытой разработки

Просматривать исходный код и участвовать в разработке (потенциально) со временем будет большое количество сторонних разработчиков

Open-source проекты часто являются «лицом» компании перед потенциальными клиентами и сотрудниками. С НИИ и университетами всё точно так же.

Разработчики open-source решений формально не несут ответственность за ошибки появляющиеся при использовании их продукта (см. лицензию). Но пользователи не всегда так считают :-)



Из «Бомба Log4j и кризис опенсорса»

Почему без открытого кода не обойтись **ІТМО**

Два вида финансирования open-source проектов:

- Краткосрочный, продолжительностью на год/полтора;
- Долгосрочный, продолжительностью на период дольше двух/трёх лет

Возможные цели каждого из них:

- Небольшой завершённый MVP с хорошо оформленной документацией (как можно более подробной), проведенными экспериментами и опубликованными (1-3) научными статьями;
- Относительно крупный проект с командой вовлеченных в разработку научных сотрудников, с большим количеством (10-15) научных статей по теме и развитой системой сопровождения фреймворка. Формирование группы единомышленников в выбранной научной области

Особенности:

- Завершённый "изолированный" программный продукт практически не требующий сопровождения;
- Относительно крупный проект с большими накладными расходами по сопровождению;

Популяризация открытого кода (1/2)



Рассказывать про открытые разработки – легко и приятно. Слушать готовы многие.

Поэтому помимо написания кода при open-source разработке важным аспектом является его популяризация - формирования сообщества пользователей и контрибьюторов, которые возможно будут сопровождать проект в дальнейшем. Данный пункт актуален для уже набравших вес проектов, - когда есть что показать и о чем заявить пользователям.

Подойдут такие ресурсы как:

- Хабр, vc.ru;
- Medium, analyticsvidhya, Towards Data Science;
- Youtube и другие видеохостинги;
- Профильные чаты и конференции (например Open Data Science);
- Научные публикации и научные конференции

Популяризация открытого кода (2/2)



Мероприятия:

- HighLoad++: open-source трибуна – (<https://highload.ru/foundation/2022/news/1876>);
- YaTalks - промо новых Open Source проектов – (<https://yataalks.yandex.ru/ru>);
- Митапы «Научный opensource» - здесь и далее.

Журналы:

- Journal of Machine Learning Research (Q1) – секция открытого кода (формат 4-х страничных статей);
- Journal of Open Research Software (Q2);
- Journal of Open Source Software (без индексации).

Ресурсы по открытому коду



Сайты:

- https://habr.com/ru/hub/open_source/ - посты про открытый код
- <https://russiaos.ru/> - агрегатор мероприятий и событий

Чаты и каналы:

- <https://t.me/OpenSourceRu> - активное open-source сообщество в TG

Поддержка:

- <https://fasie.ru/> - Фонд содействия инновациям (гранты на открытый библиотеки)

Книги:

- <http://aosabook.org/en/index.html> - коллекция книг The Architecture of Open Source Applications

- Разработка открытого кода – win/win ситуация и для ученых, и для пользователей;
- Разрабатывать отрытый код непросто – поэтому лучше придерживаться сложившимся практикам;
- В то же время, поучаствовать в разработки решений с открытым кодом может любой;
- Популяризовать разработки с открытым кодом проще, чем закрытые разработки.

**Спасибо
за внимание!**

it'sMO *re than a*
UNIVERSITY

Ваши контакты

Анонс: программа менторства открытого кода



Для того, чтобы помочь ученым-создателям открытых библиотек, в этом году на мегафакультете ТиНТ стартует **программа менторства открытого кода**. Она реализуется в рамках проекта НИР магистров и аспирантов.

Кому хотим помочь: создателям открытых библиотек (в первую очередь с факультета ТиНТ).

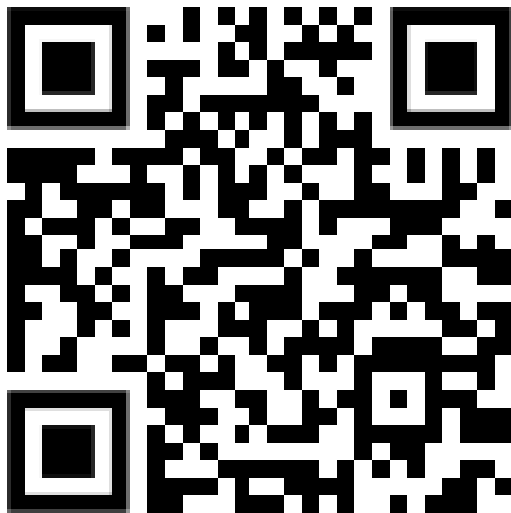
Кто может стать менторами: студенты и аспиранты ИТМО.

Каковы требования: быть студентом бакалавриата (начиная с 3-го курса) или магистратуры, иметь навыки и опыт разработки ПО на Python, готовность учиться лучшим практикам и помогать другим. Приветствуется также и участие аспирантов. Отбор кандидатов в менторы будет происходить по результатам собеседования.

Что получит менторы: опыт работы в команде разработки, прокачка своих навыков создания открытых проектов, опыт выполнения код-ревью. Участие в проекте менторинга оплачивается (от 20к/мес) и оформляется как полноценное трудоустройство.

Как подать заявку: присылать резюме Анастасии Ященко (yashchenko.anastasia.v@gmail.com) с темой “ТИНТ - менторство открытого кода - ФИО”

Программа менторства открытого кода



<https://aim.club/publications/mentoringprogram>