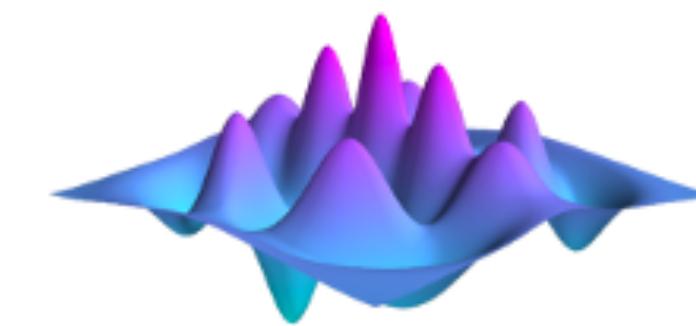


Open-source software for open science



QuTiP
Quantum Toolbox in Python



Nathan Shammah
Theoretical Quantum Physics Lab
Cluster for Pioneering Research
RIKEN, Saitama, Japan



2nd July 2019
CM Seminar
ICTP, Trieste

QuTiP: The Quantum *Physics* Simulator

The Quantum Toolbox in Python: A toolbox to study the **open** quantum dynamics of realistic systems.



Interactive Lectures @ ICTP, Leonardo Building

Tue 25th June - 11:45am, Seminar Room –

Driven-dissipative models in quantum physics

Wed 26th June - 11am, Seminar Room –

Quantum Open Source & Introduction to QuTiP

Thur 27th June - 9am, Computer Room –

Hands-on session on QuTiP's main features

Mon 1st July - 9am, Computer Room –

QuTiP stochastic solvers

Tue 2nd July - 9am, Computer Room –

PIQS and How to Build your Own Software

Wed 3rd July - 9am, Computer Room –

Extra meeting: SISSA/ICTP projects

Take a snapshot



<https://github.com/nathanshammad/interactive-notebooks>

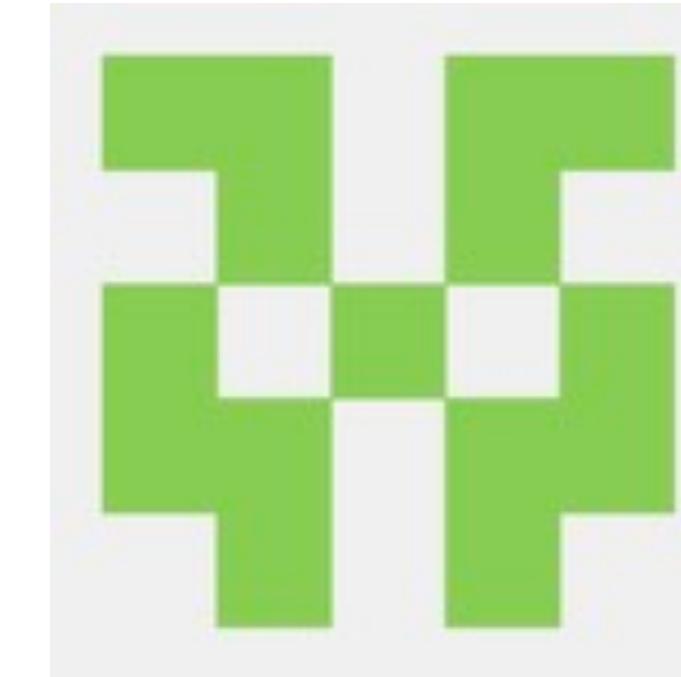
Acknowledgements and funding



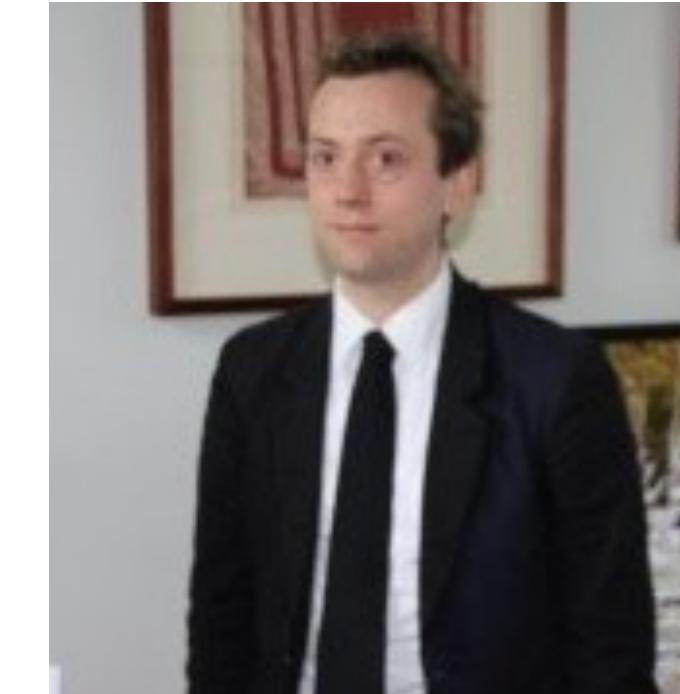
Shahnawaz Ahmed
Chalmers, Sweden
(RIKEN, Japan)



Alex Pitchford
Aberystwyth University
United Kingdom



Eric Giguère
U. de Sherbrooke
Canada



Dr. Neill Lambert
RIKEN, Japan



Prof. Franco Nori
RIKEN, Japan
U. of Michigan (USA)



日本学術振興会
Japan Society for the Promotion of Science



Japan Science and
Technology Agency



THE ROYAL
SOCIETY

JOHN TEMPLETON
FOUNDATION

NUMFOCUS
OPEN CODE = BETTER SCIENCE

Google
Summer of Code

UNIVERSITÉ DE
SHERBROOKE

PRIFYSGOL
ABERYSTWYTH
UNIVERSITY



@NathanShammah

GitHub: nathanshammah
LinkedIn: Nathan Shammah

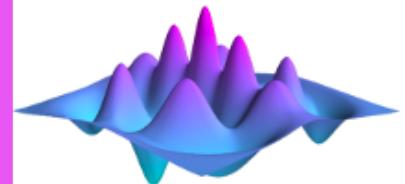
medium.com/quantum-tech

quantika.co

QuTiP: The project at a glance

The Quantum Toolbox in Python

Project Impact



QuTiP

Quantum Toolbox in Python

>600 citations (Google Scholar)

downloads 79k total (conda forge)

More info at <http://qutip.org/>

Timeline:

Inspired by the Quantum Toolbox in MatLAB.

2011-2012: QuTiP 1.0

Aug 2015: 100 citations

Aug 2016: 200 citations

Jan 2017: QuTiP 4.0

July 2018: QuTiP 4.3

Authors

Comp. Phys. Comm. 183, 1760–1772 (2012); ibid. 184, 1234 (2013).

Code



Robert J. Johansson
Rakuten Inc.



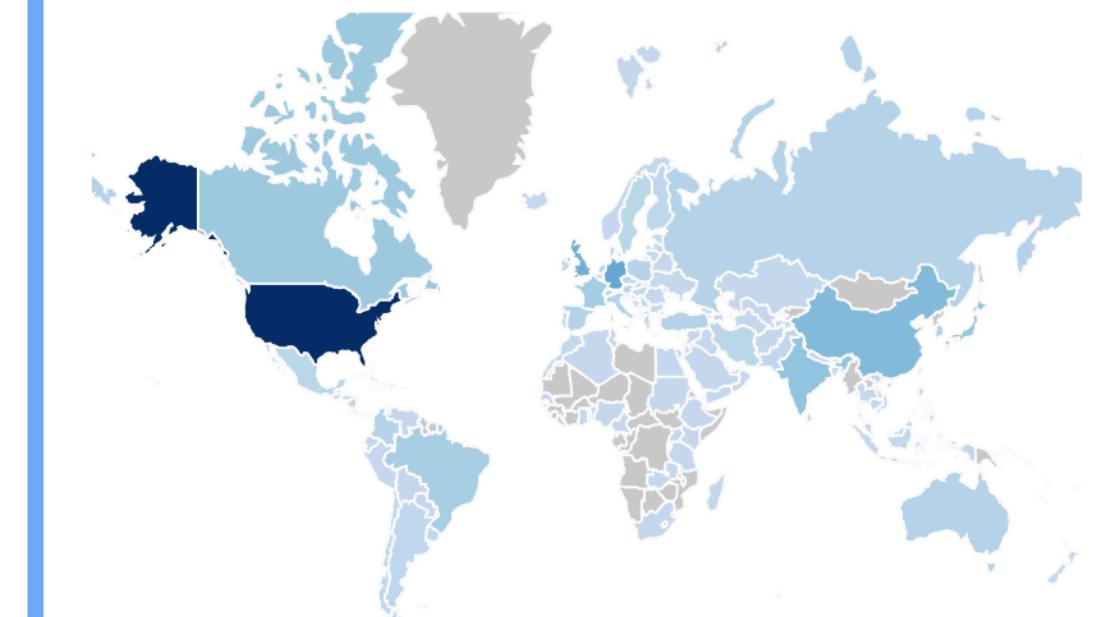
Paul D. Nation
IBM Q



Franco Nori
RIKEN / U. Michigan

Users

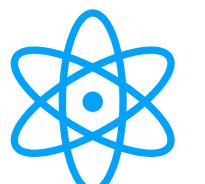
Distribution of 25k website visitors (2016)



Lead Developers



Alex Pitchford
Aberystwyth University



Éric Giguère
U. Sherbrooke



Arne Grimsmo
Université de Sherbrooke



Chris Grenade
University of Sydney

Contributing Developers

- Neill Lambert (RIKEN)
- Denis Vasilyev (Leibniz)
- Kevin Fischer (Stanford)
- Jonathan Zoller (Ulm University)
- Ben Criger (RWTH Aachen)
- ...
- Louis Tessler (RIKEN)
- Shahnawaz Ahmed (Chalmers)
- Nathan Shammah (RIKEN)

- GitHub: 44 contributors, 4k commits

License: BSD

(Berkeley Software Distribution)

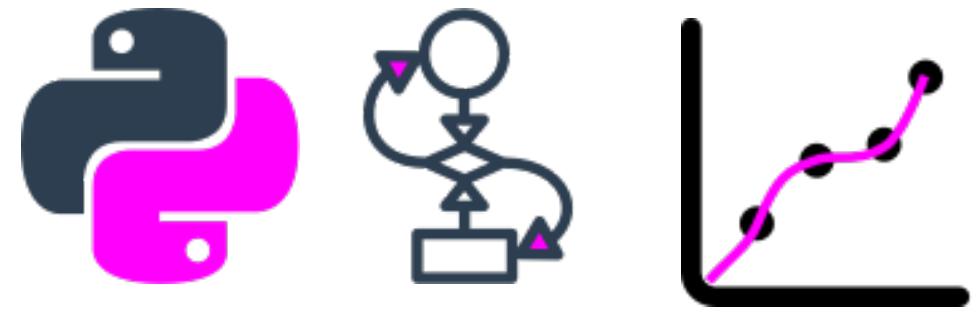
Style: PEP8 compliant

Libraries used:

- | | |
|----------|------------------------|
| • Scipy | • Matplotlib |
| • NumPy | • SymPy |
| • Cython | |
| | • Jupyter notebooks |
| | • Online documentation |
| | • Independent testing |

Open Source

A new era for open source



Open-Source Basics

Read. Download. Deploy.

Definition: You can **read** the source code (open-source ≠ free).

Examples of open-source: Linux, Android, FireFox, MySQL, LibreOffice, **Python**.

Open-source deployment is **accelerating** many end-industries applications.



Features for Developers

Learn. Debug. Deploy.

You can **learn** by reading the code and become a better developer.

You can edit the source code (licenses apply) and **collaborate** to existing projects.

You can submit **fixes** to bugs, propose improvements.



Open-source for Businesses

Flexible. Valuable. Growing fast.

Source is generally free but companies can charge for additional services.

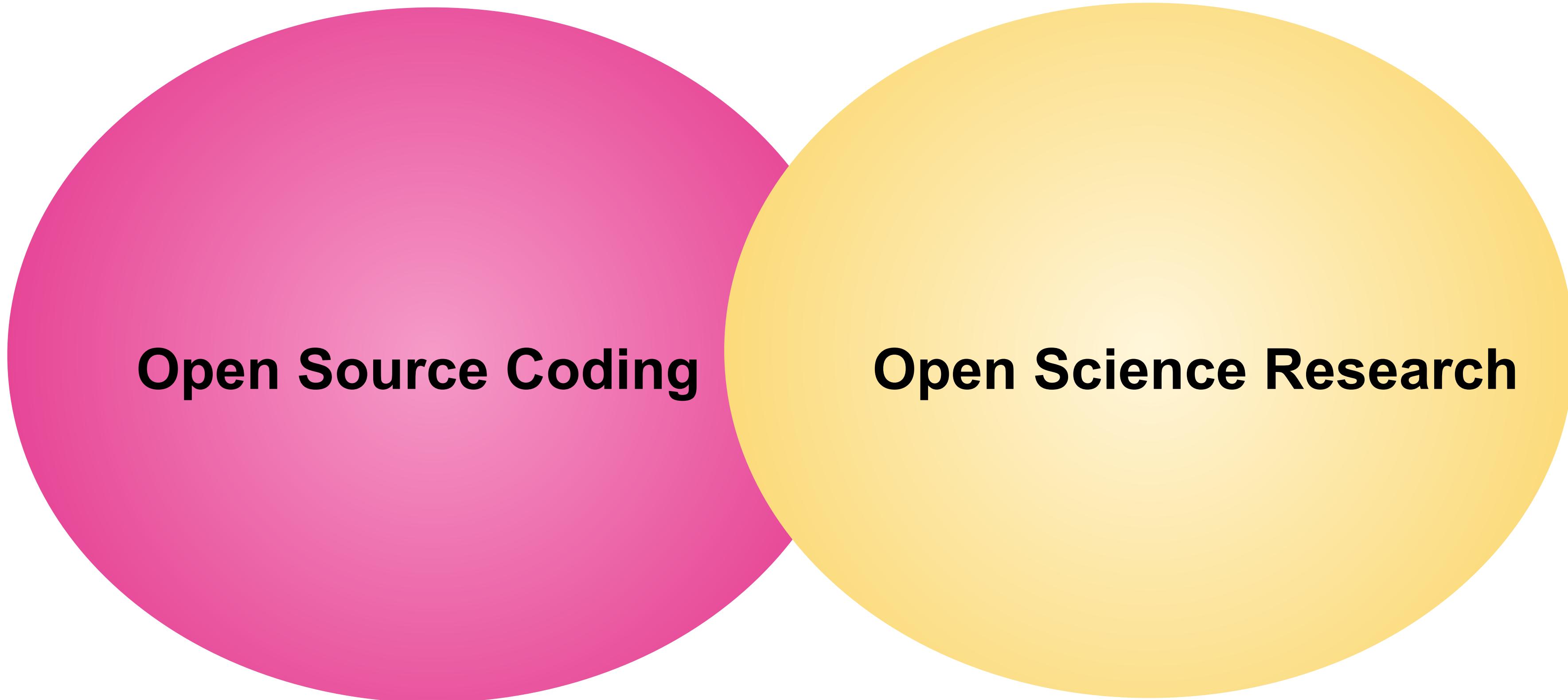
GitHub acquired for \$7.5 bln (2018). GitLab: \$100 mln in funding (2018).

Red Hat acquired by IBM for \$34 bln (2019).

Machine learning ('AI') is driving **fast**, pervasive adoption of open-source libraries.

Open source and open science

Aligned vision



Open source and open science

Aligned vision

Open Source Coding

Open Science Research

- Allow **access** to the research/project results, sharing knowledge.
- **Collaboratively** advance the field, building upon others' results.
- Coordinate large and delocalized **teams** working remotely.
- Make supporting data and code available for fast **reproducibility**.

Open Source for Open Science

A new movement in research strengthening science



Gaël Varoquaux

Tue 19 September 2017

[← Home](#)

gael-varoquaux.info

Beyond computational reproducibility, let us aim for reusability

under science | scientific computing | publishing | software | reproducible research

[Tweet](#) [G+](#)

G. Varoquaux: Group leader at INRIA in Paris.
One of the scikit-learn core developers.

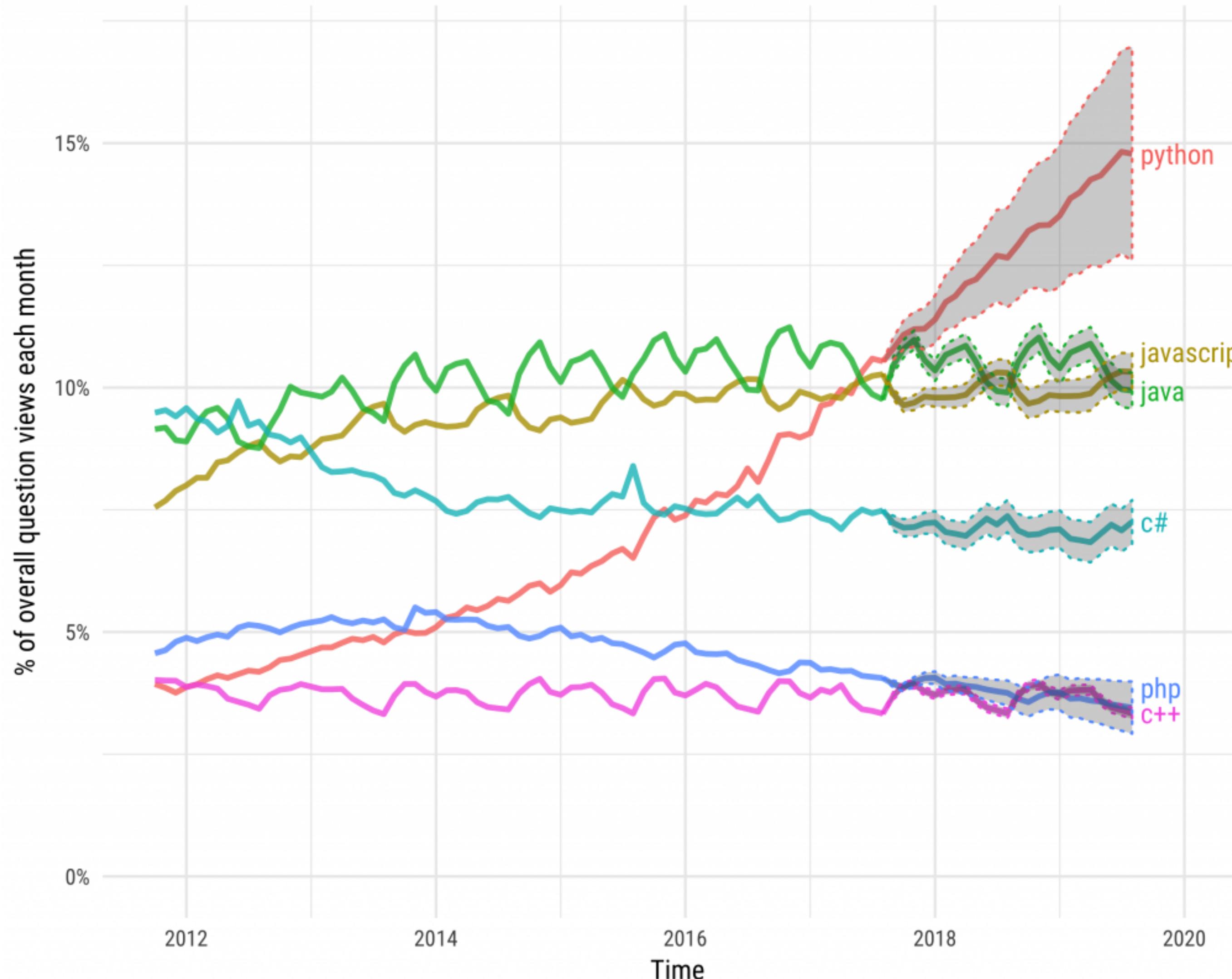
From reproducible data to reusable code.

The steady growth of Python

Empowered by a large open-source ecosystem

Projections of future traffic for major programming languages

Future traffic is predicted with an STL model, along with an 80% prediction interval.



Source: David Robinson

Python's strengths

A community-based programming language



Community

Resources

Tools

Python's strengths

A new community-based programming language



Community

PyCons
Workshops
Sprints
EuroSciPy



Python's strengths

A modular architecture for well-maintained libraries



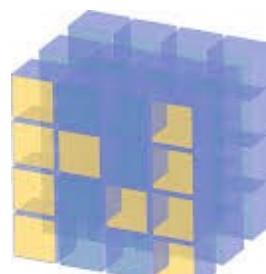
Resources

Libraries

Matplotlib

Scikit-learn

NumPy



NumPy



Python's strengths

Interactive Jupyter notebooks run an IDE in your browser



Tools

- Notebooks
- LaTeX comments
- Interactive code
- Jupyter



Below we give basic examples on the use of `qutip.piqs`. In the first example the incoherent emission of N driven TLSs is considered. In the two-level system ensemble is a subsystem coupled to another subsystem, a bosonic cavity. Similar considerations apply to the coupling to other subsystems (a single qubit, another two-level system ensemble).

In [1]:

```
import matplotlib.pyplot as plt
import matplotlib as mpl
from matplotlib import cm

from qutip import *
from qutip.piqs import *

import matplotlib.animation as animation
from IPython.display import HTML
from IPython.core.display import Image, display
```

1. N Qubits Dynamics

We study a driven ensemble of N TLSs emitting incoherently,

$$H_{\text{TLS}} = \hbar\omega_0 J_z + \hbar\omega_x J_x$$

$$\dot{\rho} = \mathcal{D}_{\text{TLS}}(\rho) = -\frac{i}{\hbar}[H_{\text{TLS}}, \rho] + \sum_{n=1}^N \frac{\gamma_n}{2} \mathcal{L}_{J_{-,n}}[\rho]$$

Ask help to the community – And help out

It's never too difficult to help



<https://www.nature.com/articles/d41586-019-02046-0>



TOOLBOX · 01 JULY 2019

How to support open-source software and stay sane

Releasing lab-built open-source software often involves a mountain of unforeseen work for the developers.

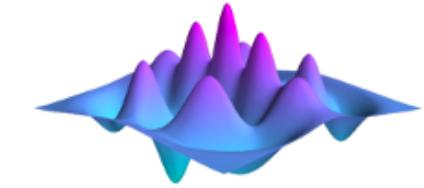
Anna Nowogrodzki

Contribute to an existing open source project

Multiple interaction channels

<http://qutip.org/tutorials.html>

Website:
Tutorials



QuTiP
Quantum Toolbox in Python

<http://qutip.org/docs/latest/>

Documentation User Guide
Documentation API: docstring comments
Documentation API: source code

<https://groups.google.com/forum/#!forum/qutip>

Help Group:
Ask (installation, physics) questions
Find *already answered* installation questions

<https://github.com/qutip/qutip>

GitHub:
Found a bug?
- Search answer in *already closed* issue
- Search *an open (unsolved)* issue on same topic
- Unknown? Open an Issue
Have a solution? Open a Pull Request (proposal for code modification of the official project source code)

<https://gitter.im/qutip>

Gitter Chat:
Join the conversation if you have other question

Create a new open source project

Use the tools of open source

<https://github.com/nathanshammah/opensource>

GitHub, Inc. [US] | https://github.com/nathanshammah/opensource/blob/master/README.md

A Guide to Building Your Open-Source Science Library

A cheatsheet to develop a scientific open-source library from scratch.

Zero to Library

Here you will find information to design, build, and release an open-source library to perform scientific research in Python from scratch to finish.

0 - Open Source for Open Science: Some information about the Python and open source ecosystem, and how they relate to open science are also given.

- StackOverflow

1 - Before Starting Coding: Setting up the working environment on your machine, including the tools you will need to write code efficiently.

- Sublime, Git, GitHub

2 - Developing your Project: A step-by-step guide with best practices for coding, and tips for making code development as effortless as possible.

- PEP 8, PEP 257
- jupyter notebook
- nbconvert

3 - Testing: Especially in software related to scientific research, at start, the destination is not always crystal clear. Code is written, optimized, reorganized. Unit testing is a crucial task to avoid getting lost in the process.

- nose2, pytest

Open Science through Open Source

The tools of open source make your code count

Code & Testing



GitHub



Travis CI

Documentation



Publication



zenodo

Open Science through Open Source: Making code count

The tools of open source

Code & Testing



Create code collaboratively; host it and perform version control



Enstrength code with independent testing of functions, from the cloud



Documentation

Self-generate a documentation for the library from commented functions

Freely host a library documentation with dedicated markup options



Publication

Install easily software on multiple platforms, ensuring update compatibilities

Keep data safe and track release control, with immediate DOI for bibliographic records

An [OpenAIRE](#)
+ CERN project

Python: Function, Class, Objects

[piqs/dicke.py](#)

```
def num_dicke_states(N):
    """
    Calculate the number of Dicke states.

    Parameters
    -----
    N: int
        The number of two-level systems.

    Returns
    -----
    nds: int
        The number of Dicke states.

    """
    nds = (N/2 + 1)**2 - (N % 2)/4
    return int(nds)
```

```
>> from piqs import num_dicke_states
>> print(num_dicke_states(30))
256
```

```
class Dicke(object):
    """
    A Dicke class defining an ensemble of spins

    Parameters
    -----
    N: int
        Number of two level systems

    """
    def __init__(self, N, H=None, emission=0.):
        self.N = N
        self.H = H
        self.emission = emission

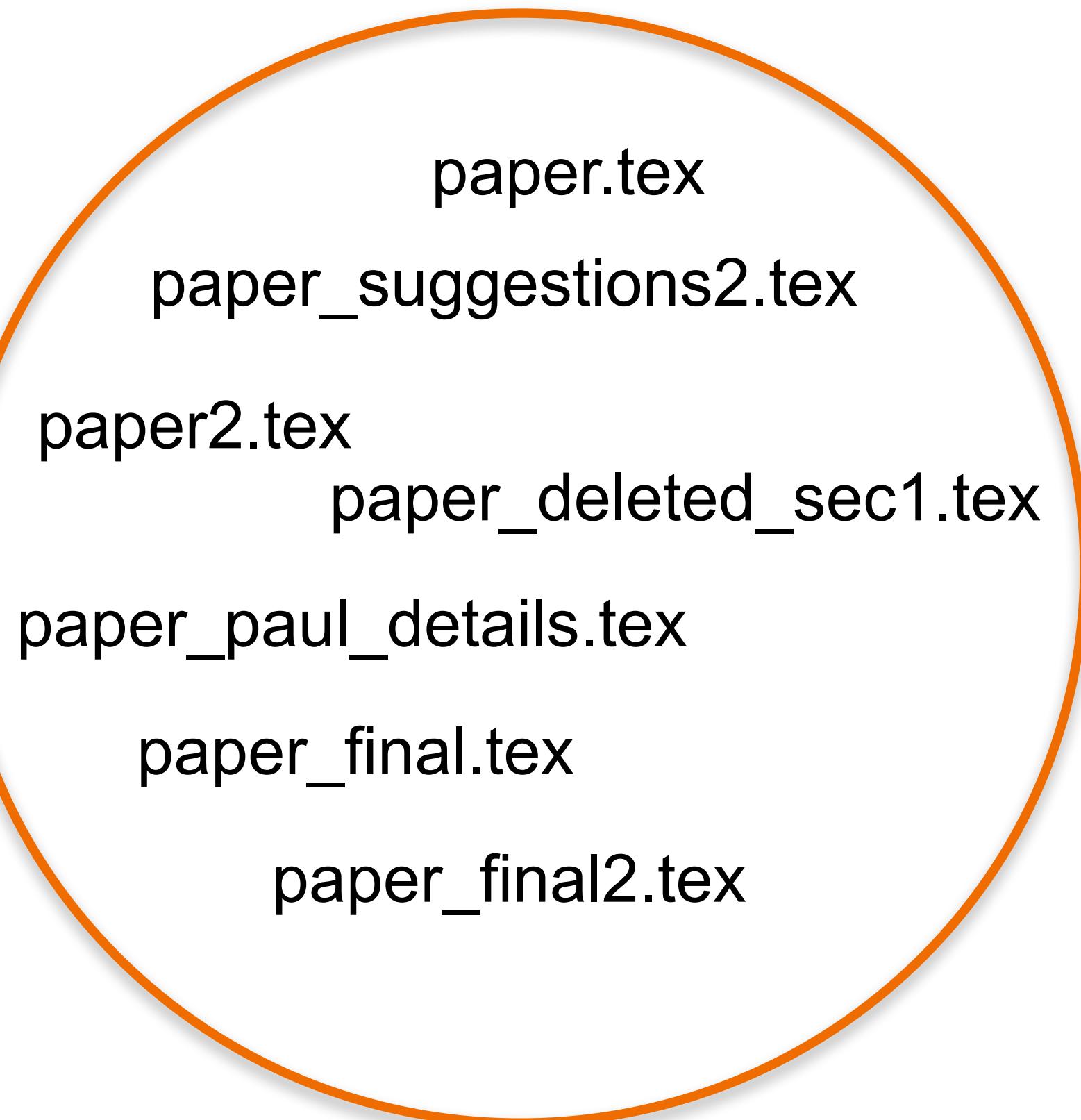
    def liouvillian(self):
        """
        Calculate the Liouvillian superoperator.

        .....
    
```

```
>> from piqs import Dicke
>> ensemble = Dicke(30, emission=0.1)
>> L = ensemble.liouvillian()
```

Git and GitHub

- Version control
- Collaborative code
- Online and open-source



	COMMENT	DATE
O	CREATED MAIN LOOP & TIMING CONTROL	14 HOURS AGO
O	ENABLED CONFIG FILE PARSING	9 HOURS AGO
O	MISC BUGFIXES	5 HOURS AGO
O	CODE ADDITIONS/EDITS	4 HOURS AGO
O	MORE CODE	4 HOURS AGO
O	HERE HAVE CODE	4 HOURS AGO
O	AAAAAAA	3 HOURS AGO
O	ADKFJSLKDFJSOKLFJ	3 HOURS AGO
O	MY HANDS ARE TYPING WORDS	2 HOURS AGO
O	HAAAAAAAAANDS	2 HOURS AGO

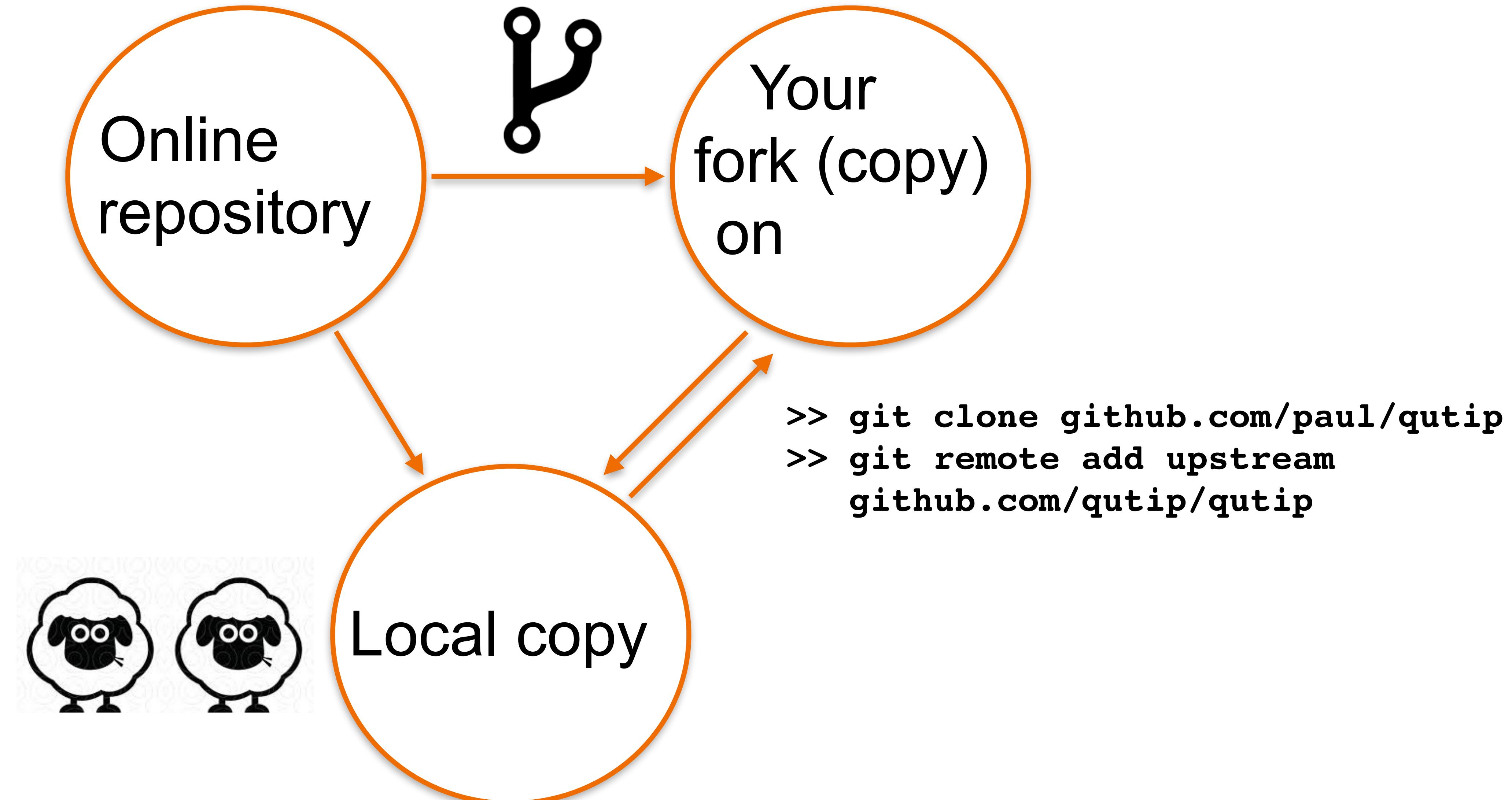
AS A PROJECT DRAGS ON, MY GIT COMMIT MESSAGES GET LESS AND LESS INFORMATIVE.

xkcd

Git: Setup

github.com/qutip/qutip

github.com/paul/qutip



Git: Use

```
>> git status  
On branch master  
nothing to commit, working tree clean
```

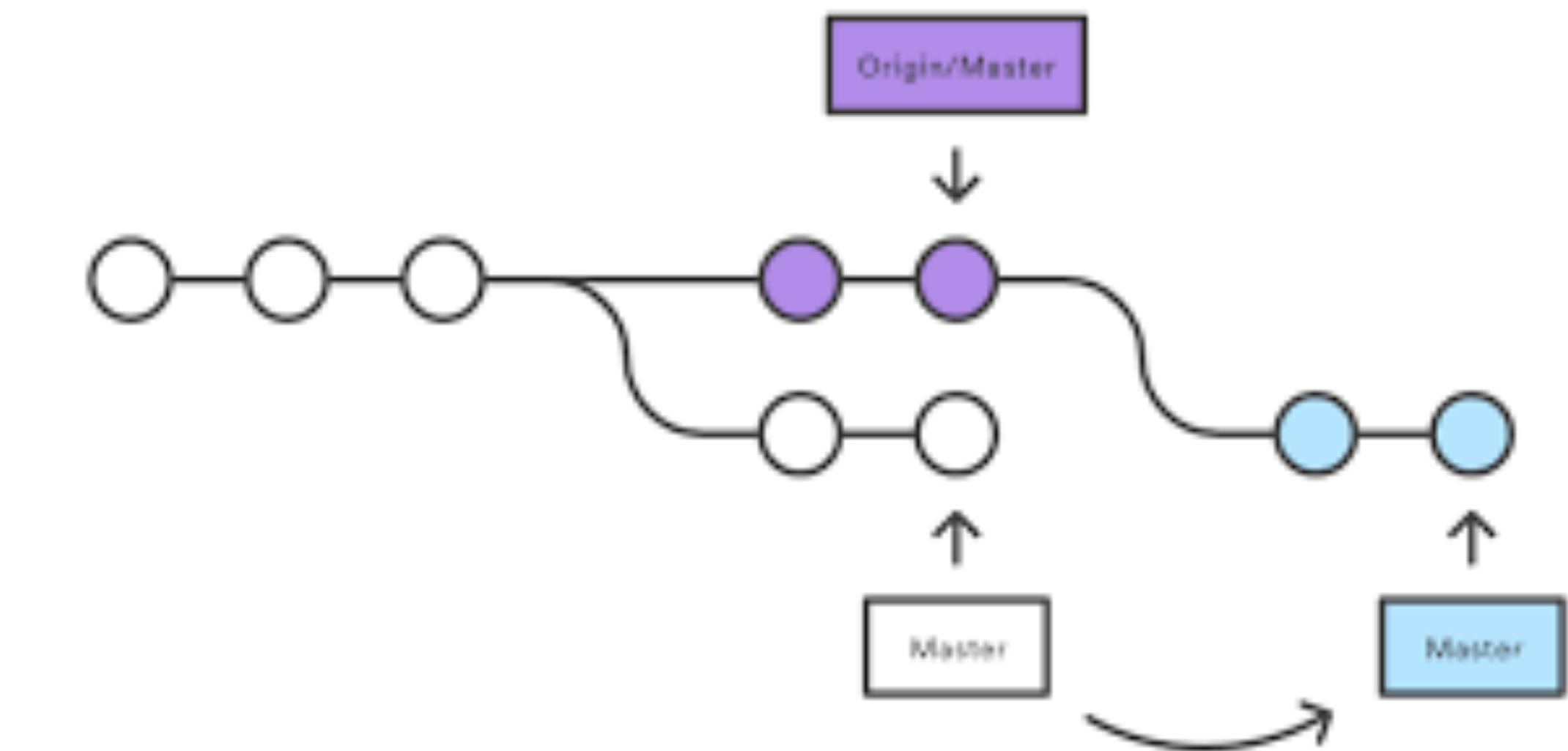
Branches

```
>> git branch checkout -b new_method  
Switched to a new branch 'new_method'
```

```
>> git branch checkout master  
Switched to branch 'master'
```

Changes

```
>> git status  
modified:   new_method.py  
>> git add .  
>> git commit -m "Added a new method"  
>> git pull upstream master  
>> git push origin new_method
```



- Open a Pull request online
- Review by others
- Merge

Documentation



Read the Docs



PDF LaTeX and more ...

A screenshot of the PIQS documentation page on Read the Docs. The top navigation bar shows the project name "piqs" and the version "latest". A search bar is present. The main content area has a header "CONTENTS:" followed by a list of sections: "Introduction", "Permutational Invariant Quantum Solver (PIQS)", "Integrated with QuTiP", "A wide range of applications", "Installation", "User Guide", "API documentation", "Developers", and "References".

- Introduction
- Permutational Invariant Quantum Solver (PIQS)
- Integrated with QuTiP
- A wide range of applications
- Installation
- User Guide
- API documentation
- Developers
- References

Docs » Introduction

Edit on GitHub

Introduction

Permutational Invariant Quantum Solver (PIQS)

PIQS is an open-source Python solver to study the exact Lindbladian dynamics of open quantum systems consisting of identical qubits.

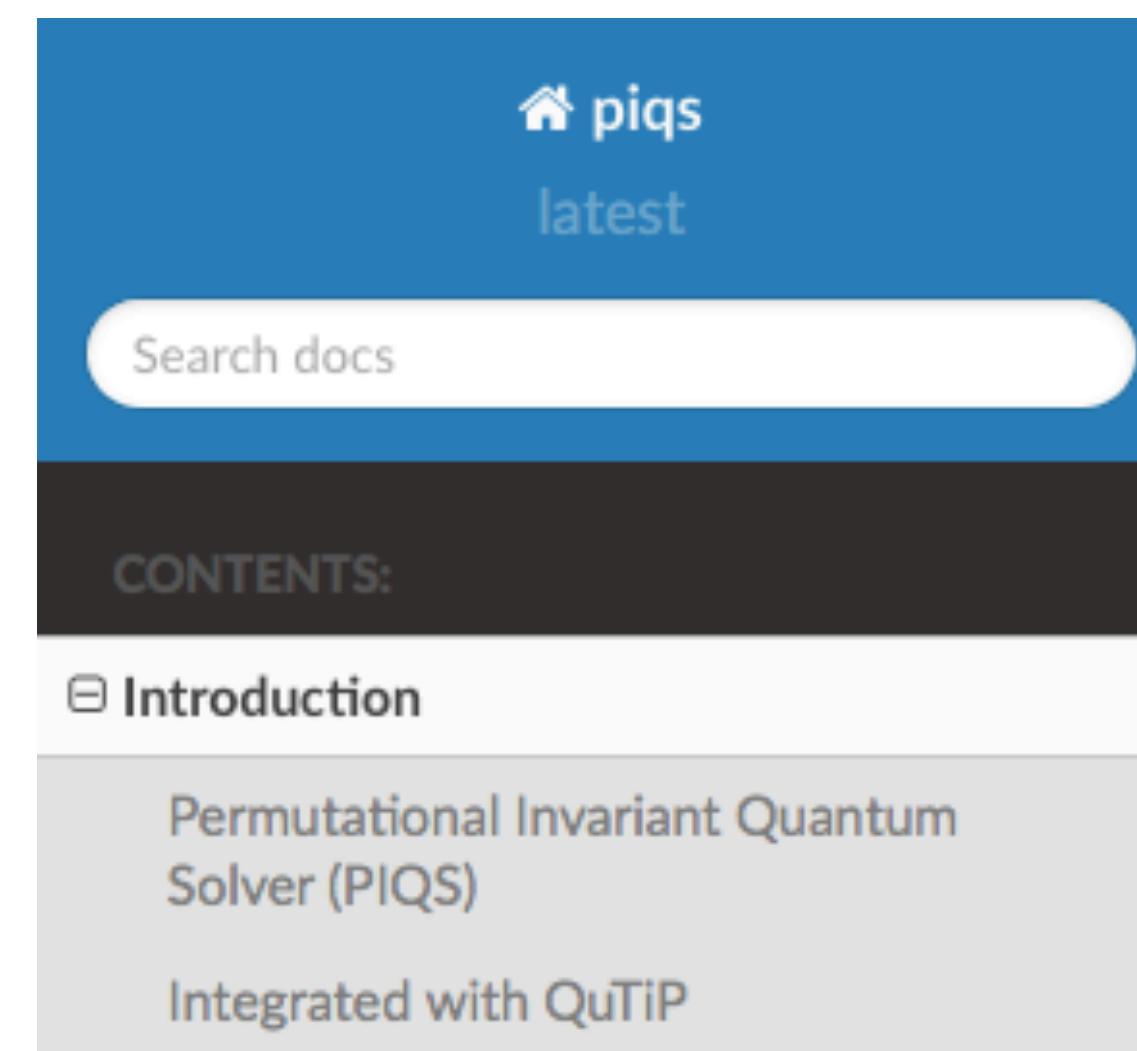
In the case where local processes are included in the model of a system's dynamics, numerical simulation requires dealing with density matrices of size 2^N . This becomes infeasible for a large number of qubits. We can simplify the calculations by exploiting the permutational invariance of indistinguishable quantum particles which allows the user to study hundreds of qubits.

Integrated with QuTiP

Documentation

Auto generate with Sphinx

```
>> sphinx-quickstart  
|-- doc/  
    |-- Makefile  
    |-- make.bat  
    |-- source  
        |-- conf.py  
        |-- conf.py  
        |-- index.rst  
        |-- intro.rst
```



Edit configurations

```
doc/source/conf.py  
# -- Project information  
project = 'piqs'  
  
# The short X.Y version  
version = ''  
# The full version  
release = '1.0'
```

Docs » Introduction

[Edit on GitHub](#)

Introduction

Permutational Invariant Quantum Solver (PIQS)

PIQS is an open-source Python solver to study the exact Lindbladian dynamics of open quantum systems consisting of identical qubits.

Generate documentation

```
>> make html
```

Read the docs - host online

Distributing package: pip, conda



conda-forge



Add setup information

piqs/setup.py

```
from setuptools import setup
setup(name='piqs',
      version='0.1',
      description='Permutational Invariant
                  ...)
```

meta.yml (recipe)

```
{% set name = "piqs" %}
{% set version = "1.0" %}

package:
    name: {{ name|lower }}
```

Make a package and upload

```
>> python setup.py register sdist upload
```

Install from pip

```
>> pip install piqs
```

```
>> conda build
```

Add to a personal channel or conda-forge

```
>> conda install -c www.github.com/paul
```

Peer-Reviewed Articles for Libraries for Quantum Physics

A list of journals

Community-based journals:

Quantum the Journal

SciPost and SciPost Physics

JOSS: Journal of Open Source Software (GitHub based)

“Professional publishers-based” journals:

Computer Physics Communications (Elsevier)

PlosOne (Plos)

SoftwareX (Elsevier)

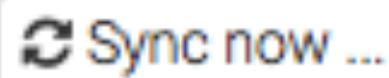
Publishing online: Release

github.com/nathanshammah/piqs

Permutational Invariance Quantum Solver for Lindblad open quantum system dynamics

The screenshot shows a GitHub repository page for `nathanshammah/piqs`. At the top, there are summary statistics: 142 commits, 7 branches, 1 release (which is highlighted with an orange box), and 2 contributors. Below this is a navigation bar with links for Code, Issues (0), Pull requests (1), Projects (0), Wiki, and Insights. Further down, there are tabs for Releases (which is active and highlighted with a blue box) and Tags. On the right side, there is a button labeled "Draft a new release" which is also highlighted with an orange box.

Publishing online: Zenodo

 GitHub Repositories (updated now) 

Get started

- ### 1 Flip the switch

Select the repository you want to preserve, and toggle the switch below to turn on automatic preservation of your software.

 ON
- ### 2 Create a release

Go to GitHub and [create a release](#). Zenodo will automatically download a .zip-ball of each new release and register a DOI.
- ### 3 Get the badge

After your first release, a DOI badge that you can include in GitHub README will appear next to your repository below.

DOI [10.5281/zenodo.8475](#)
(example)

Hands-on Session: Building a Sample Library

Python scientific project

In the lab will cover:

- Git
- Github
- Conda Environments
- Docstring comments
- Local unit testing
- Travis CI setup
- setup.py install
- pip install
- Distribution on test.pypi.org
- Building a documentation with Sphinx

You can find a sample library at
<https://github.com/nathanshammah/qictp>

Quantum Tech Newsletter



Nathan Shammah
Quantum researcher at RIKEN
Jun 25 · 6 min read

medium.com/quantum-tech

Nathan's Quantum Tech Newsletter: №11

Research Highlights

- Breakthroughs
- Reviews
- Divulgation

Tech News

- Startup creation, funding rounds
- Corporate Involvement
- Institutional Schemes

Bonus Links

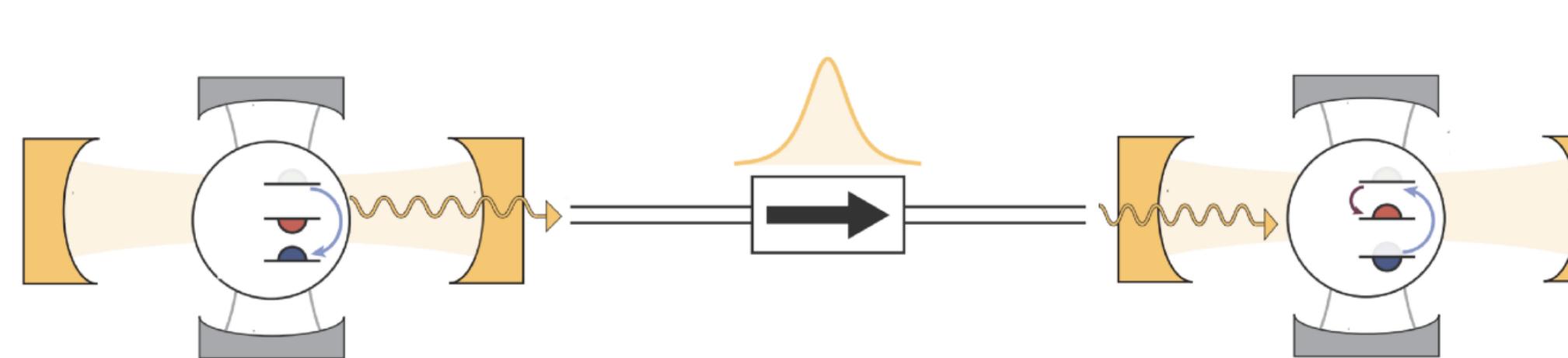
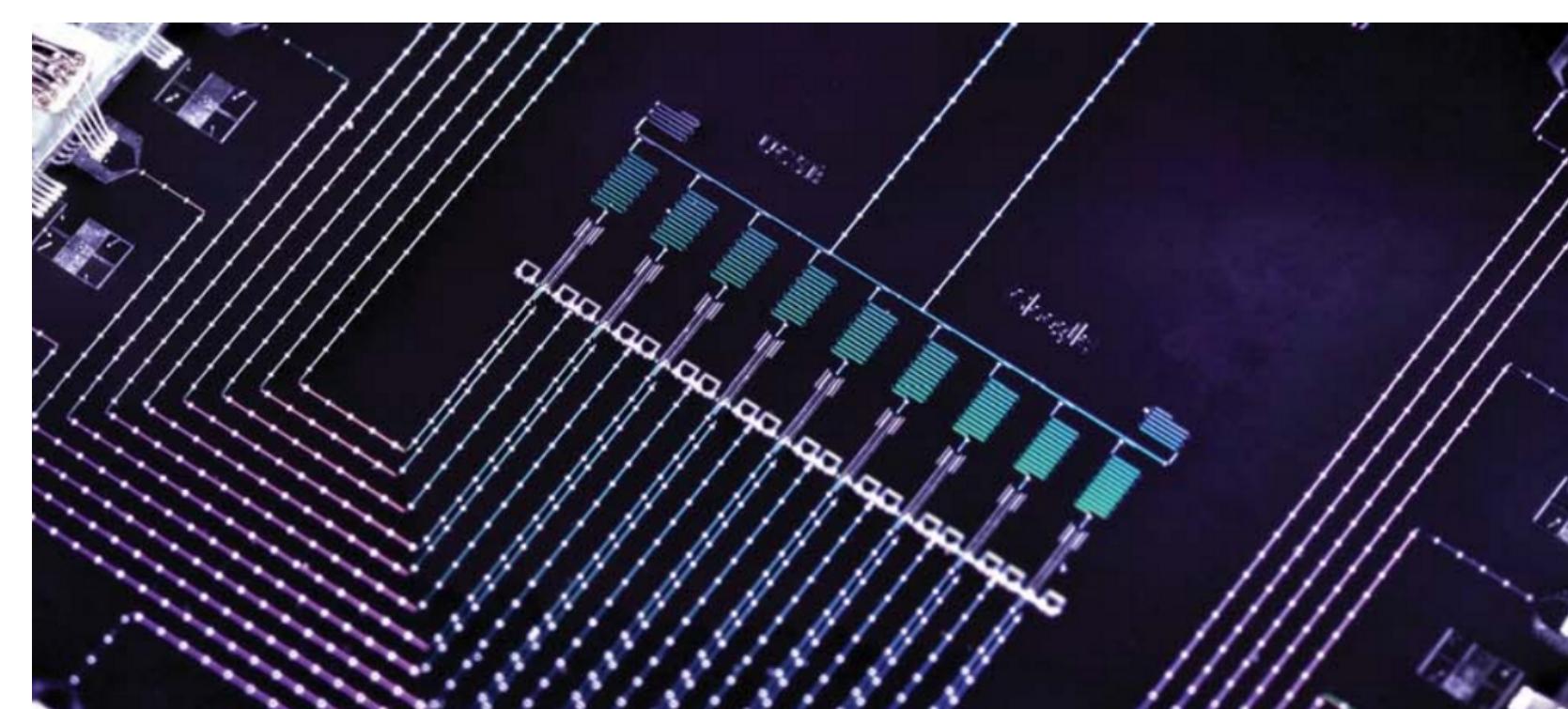
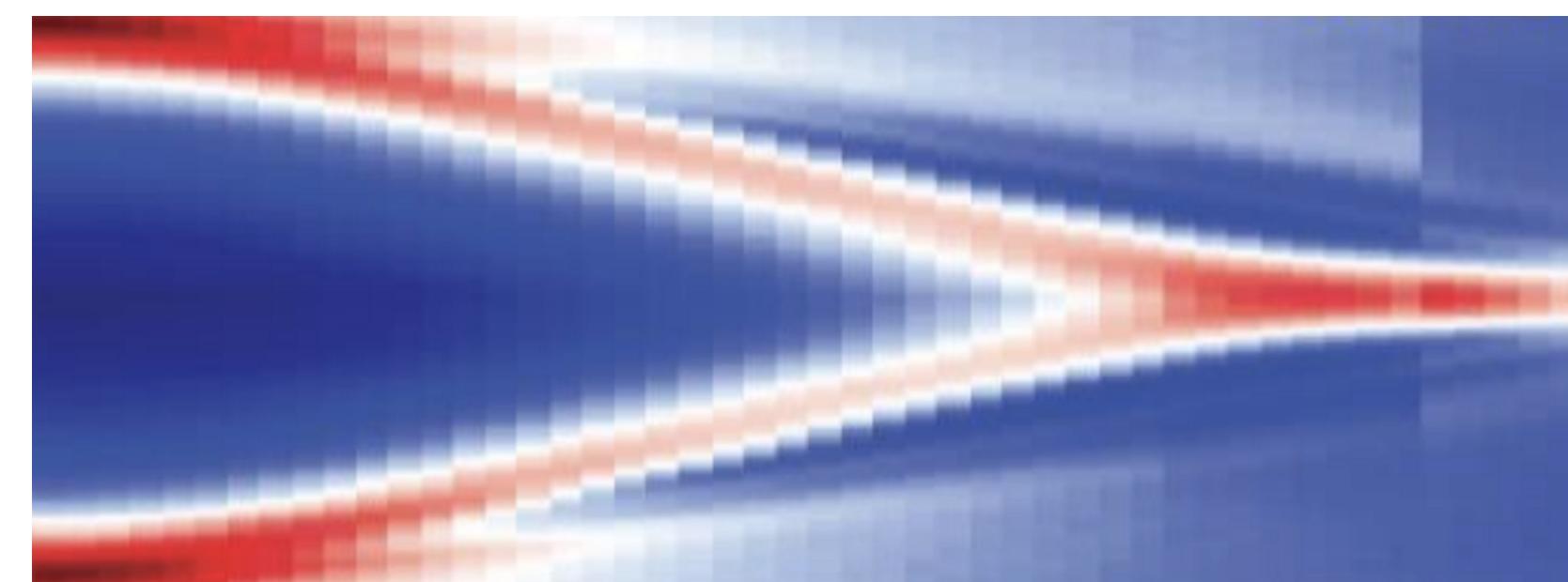
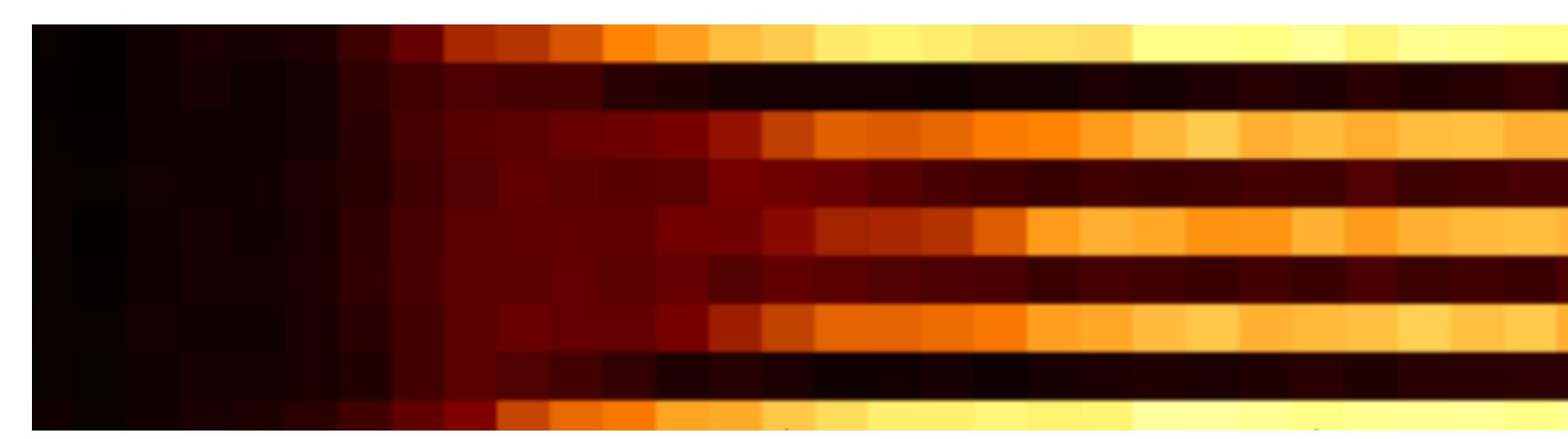
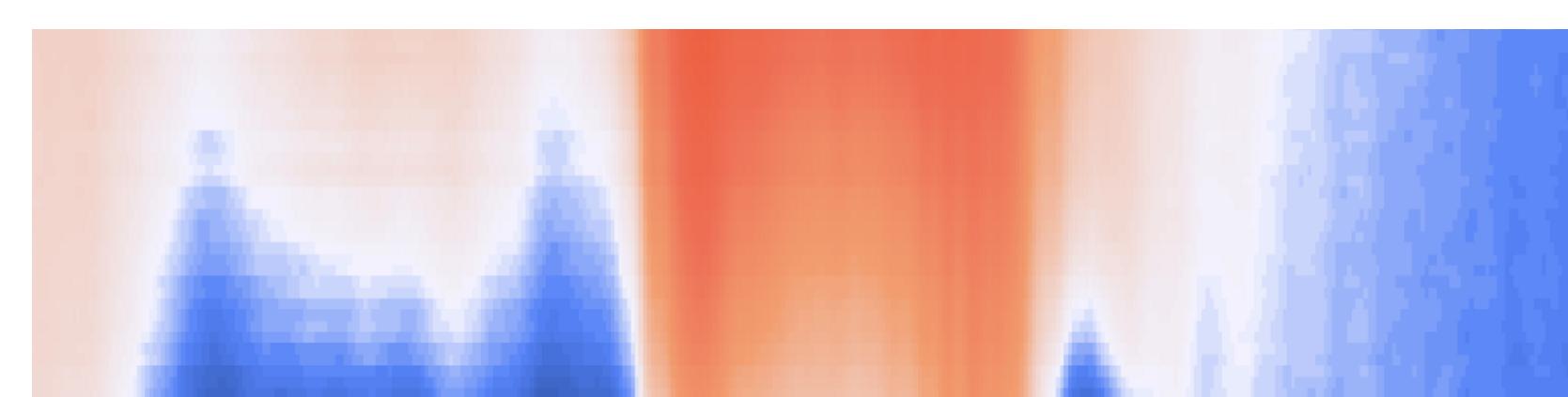
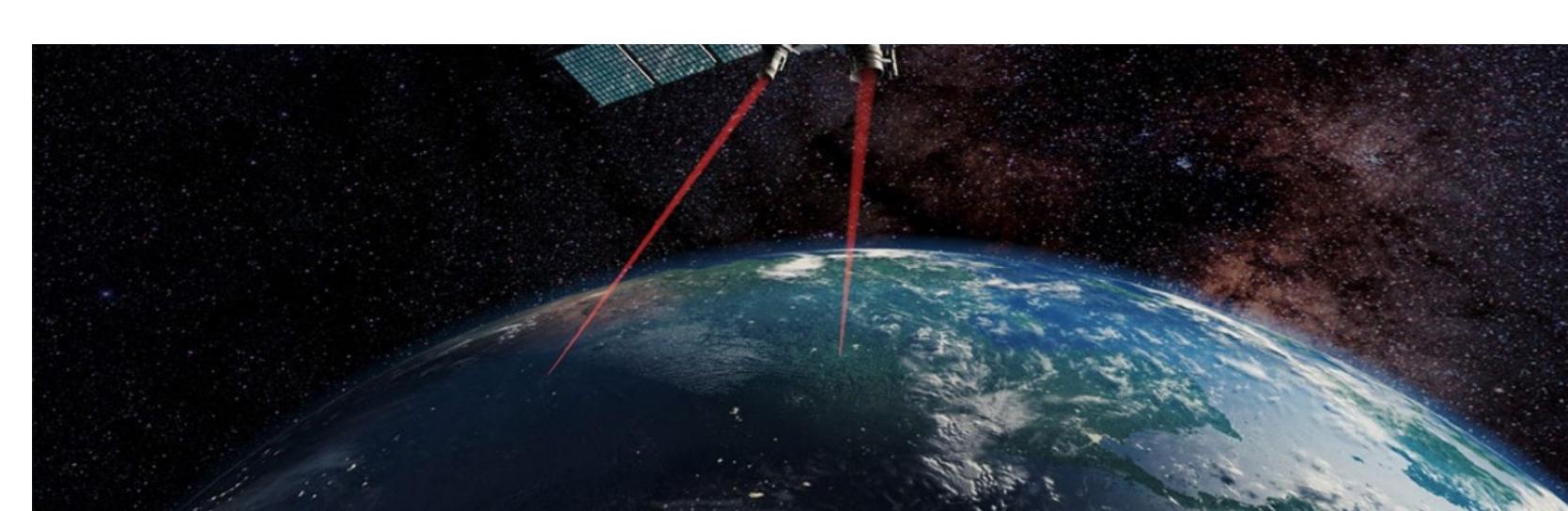
- Videos
- Long-read popular articles

Focus

- Space Quantum Communication
- Quantum Machine Learning
- Open-Source Quantum Tech
- Quantum Games

[Sign up: eepurl.com/c10FJz](https://eepurl.com/c10FJz)





Nathan Shammah
Quantum researcher at RIKEN
Jun 25 · 6 min read

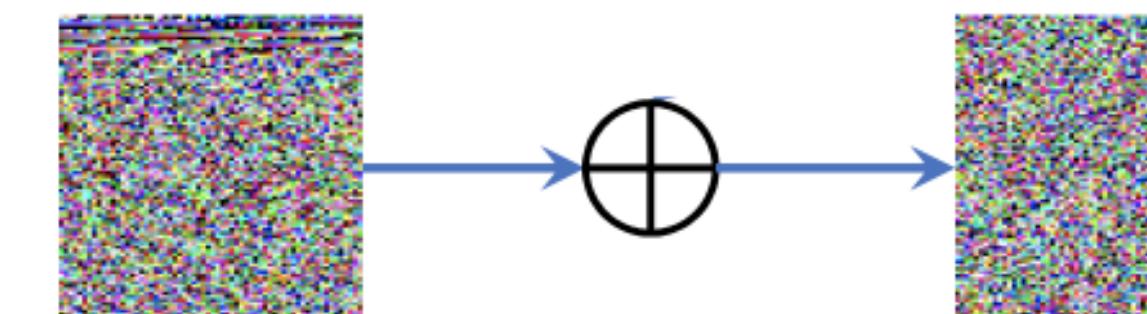
medium.com/quantum-tech



Nathan's Quantum Tech Newsletter: №11

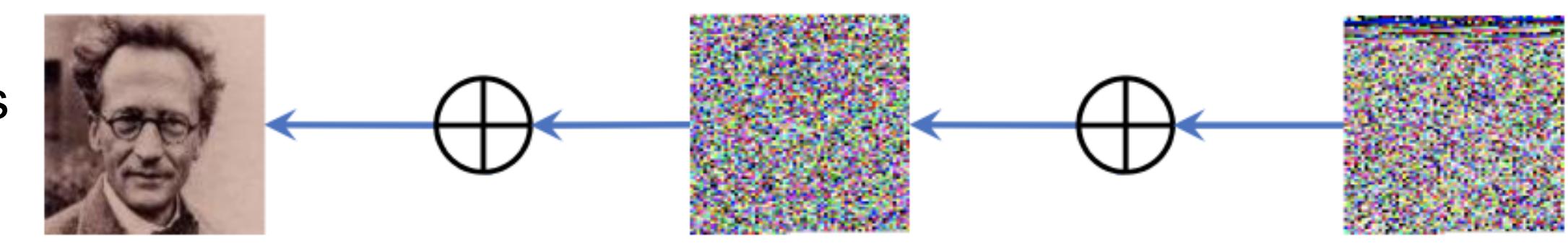
Research Highlights

- Breakthroughs
- Reviews
- Divulgation



Tech News

- Startup creation, funding rounds
- Corporate Involvement
- Institutional Schemes



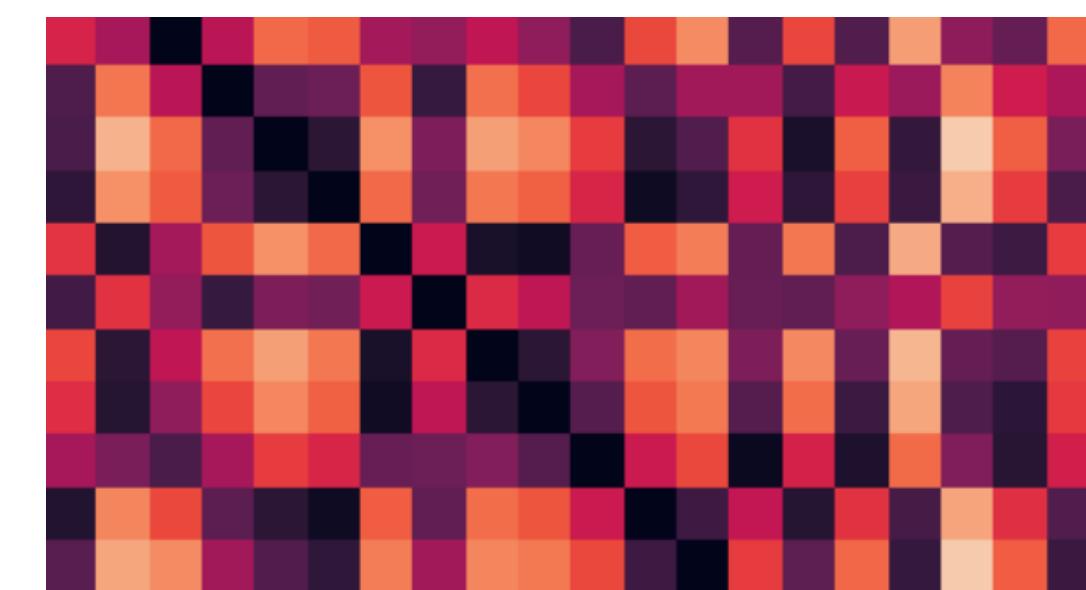
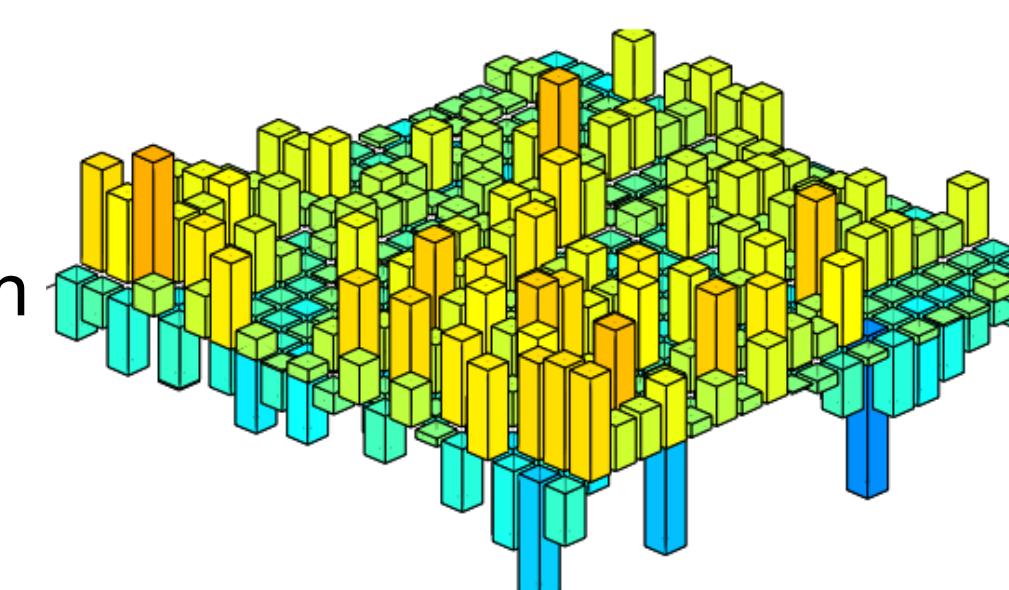
Bonus Links

- Videos
- Long-read popular articles

[Sign up: eepurl.com/c10FJz](https://eepurl.com/c10FJz)

Focus

- Space Quantum Communication
- Quantum Machine Learning
- Open-Source Quantum Tech
- Quantum Games



Thank you



@NathanShammah

GitHub: nathanshammah

LinkedIn: Nathan Shammah