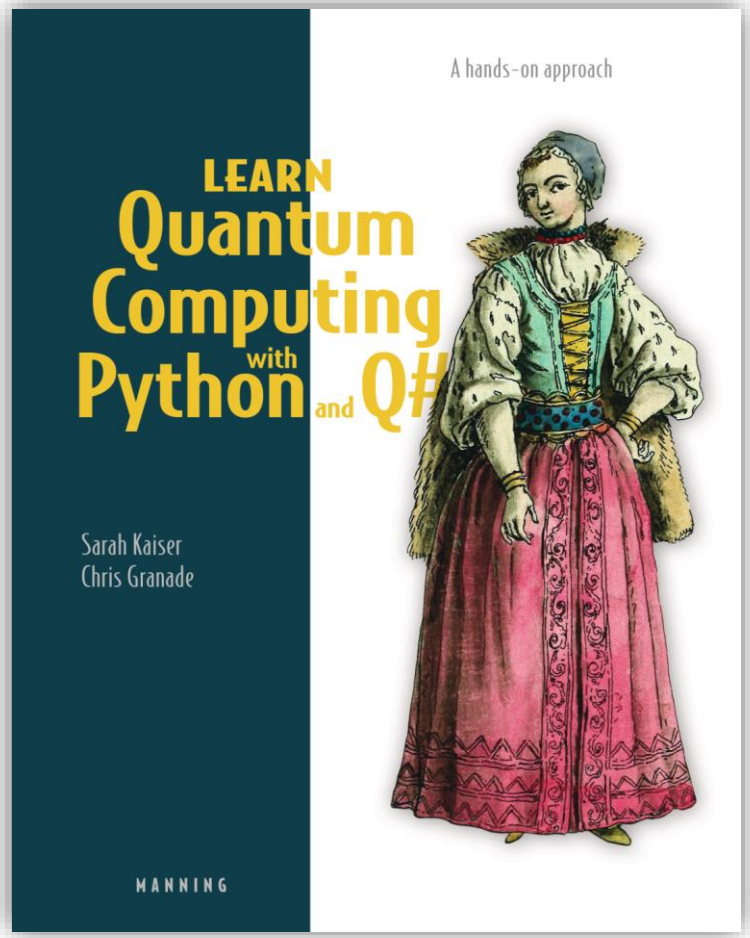
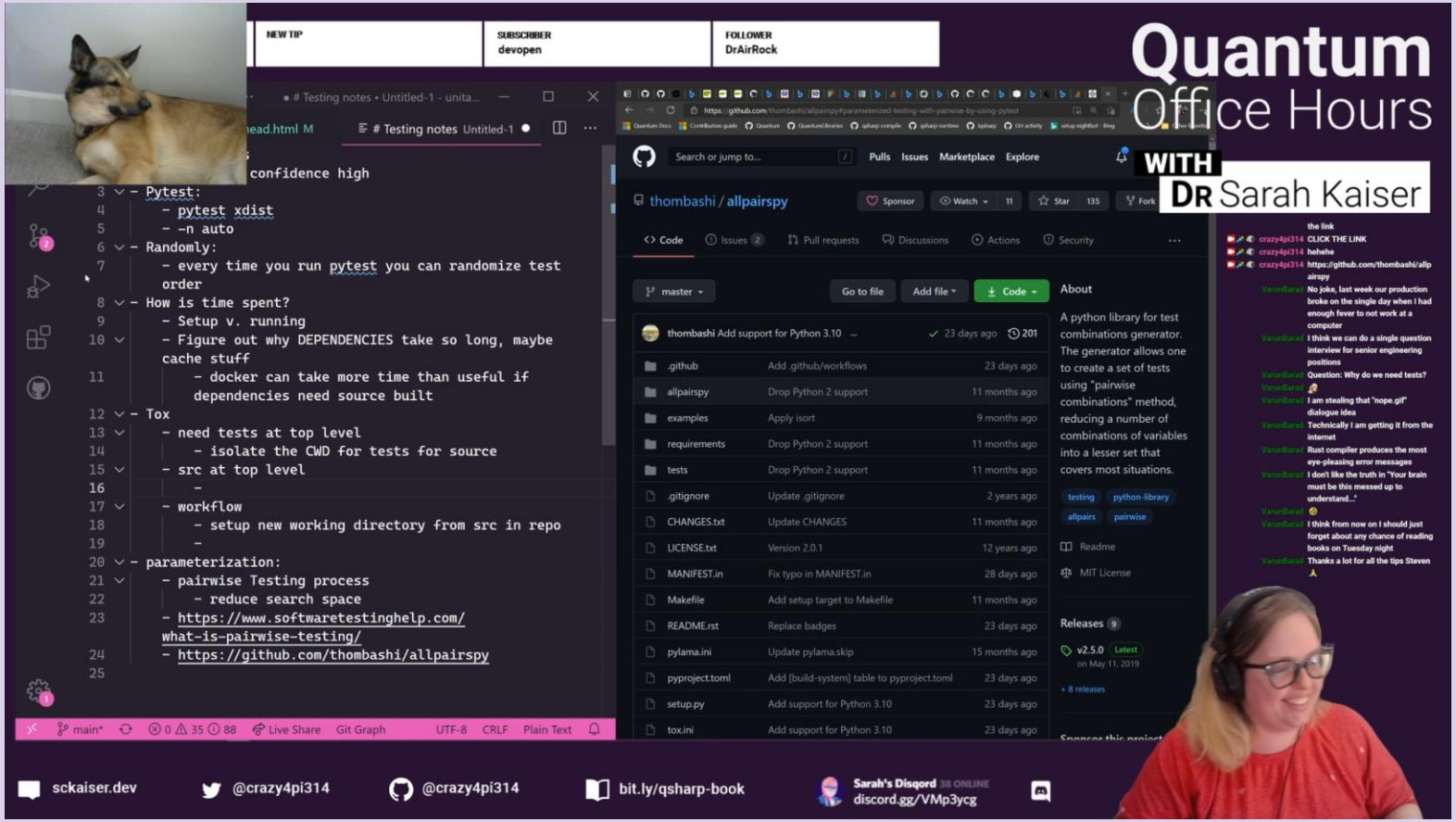
Everyone Can Be a Quantum Open Source Developer! 💖

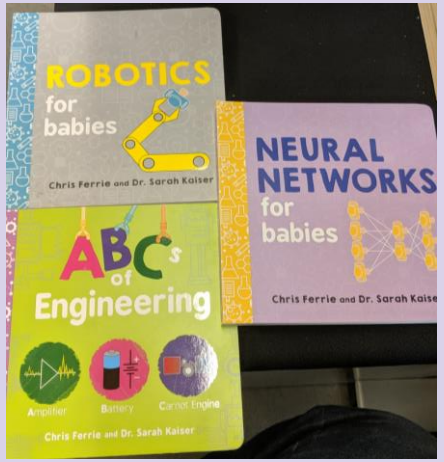
Dr. Sarah Kaiser | @crazy4pi314

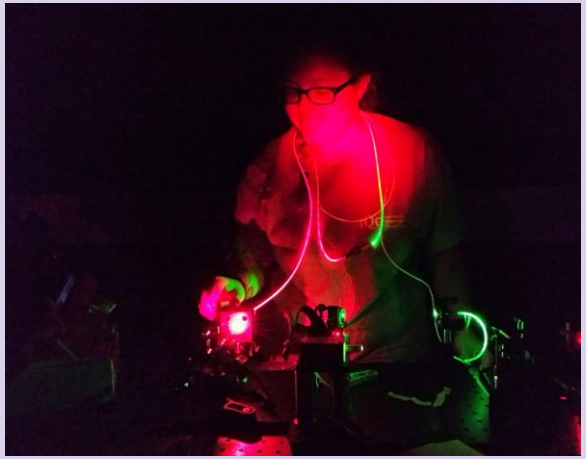
Technical staff and community lead at Unitary Fund

readme.md

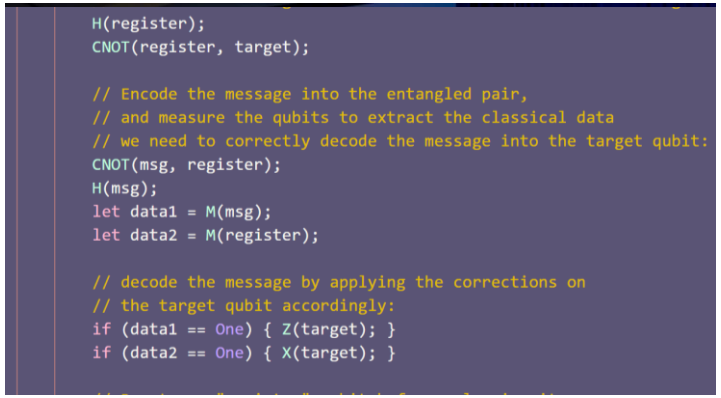

A hands-on approach
LEARN
Quantum Computing with Python and Q#
Sarah Kaiser
Chris Granade
MANNING


NEW TIP
SUBSCRIBER
devopen
FOLLOWER
DrAirRock
Quantum
Office Hours
WITH
DR Sarah Kaiser
- Pytest:
- pytest xdist
- -n auto
- Randomly:
- every time you run pytest you can randomize test order
- How is time spent?
- Setup v. running
- Figure out why DEPENDENCIES take so long, maybe cache stuff
- docker can take more time than useful if dependencies need source built
- Tox
- need tests at top level
- isolate the CWD for tests for source
- src at top level
- workflow
- setup new working directory from src in repo
- parameterization:
- pairwise Testing process
- reduce search space
- https://www.softwaretestinghelp.com/what-is-pairwise-testing/
- https://github.com/thombashi/allpairspy
thombashi / allpairspy
About
A python library for test combinations generator. The generator allows one to create a set of tests using "pairwise combinations" method, reducing a number of combinations of variables into a lesser set that covers most situations.
sckaiser.dev
@crazy4pi314
@crazy4pi314
bit.ly/qsharp-book
Sarah's Disqord
discord.gg/VMp3ycg


ROBOTICS for babies
Chris Ferrie and Dr. Sarah Kaiser
NEURAL NETWORKS for babies
Chris Ferrie and Dr. Sarah Kaiser
ABCs of Engineering
Chris Ferrie and Dr. Sarah Kaiser

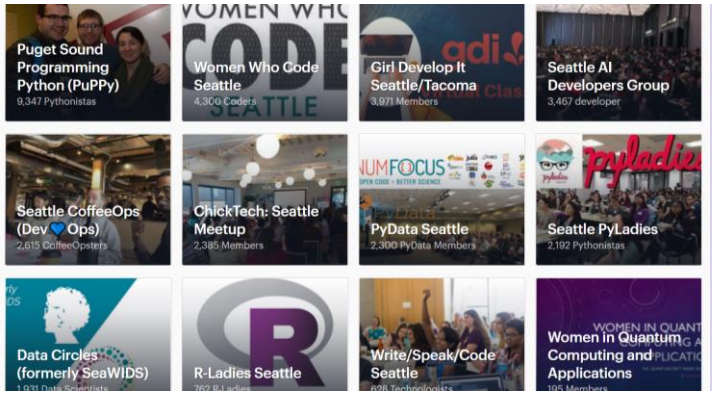


Quantum software developer + OSS advocate

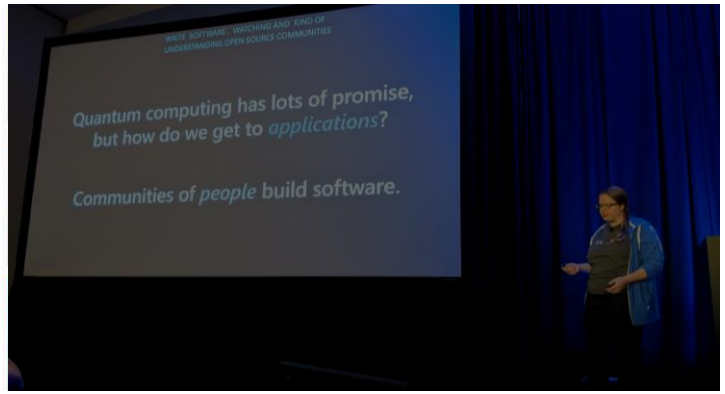


```
H(register);
CNOT(register, target);

// Encode the message into the entangled pair,
// and measure the qubits to extract the classical data
// we need to correctly decode the message into the target qubit:
CNOT(msg, register);
H(msg);
let data1 = M(msg);
let data2 = M(register);

// decode the message by applying the corrections on
// the target qubit accordingly:
if (data1 == One) { Z(target); }
if (data2 == One) { X(target); }
```

💻 I wanted to learn how to be a better programmer to help teach others



🤝 Global open source meetup groups were amazing, I wanted to help build one for quantum computing.



📣 I started giving demos of quantum programming at events and on Twitch. Also wrote a book on quantum programming

@crazy4pi314

Dr. Sarah KaisEEEKer #DefundThePolice
@crazy4pi314
That one time we sat here and talked about WINS: I got my first public PR approved on #Qsharp from @MSFTQuantum today 💖
Thanks again to @azureadvocates and especially @allinison for the best tech conference ever!!
Microsoft
4:18 PM · May 9, 2018 · Twitter for Android

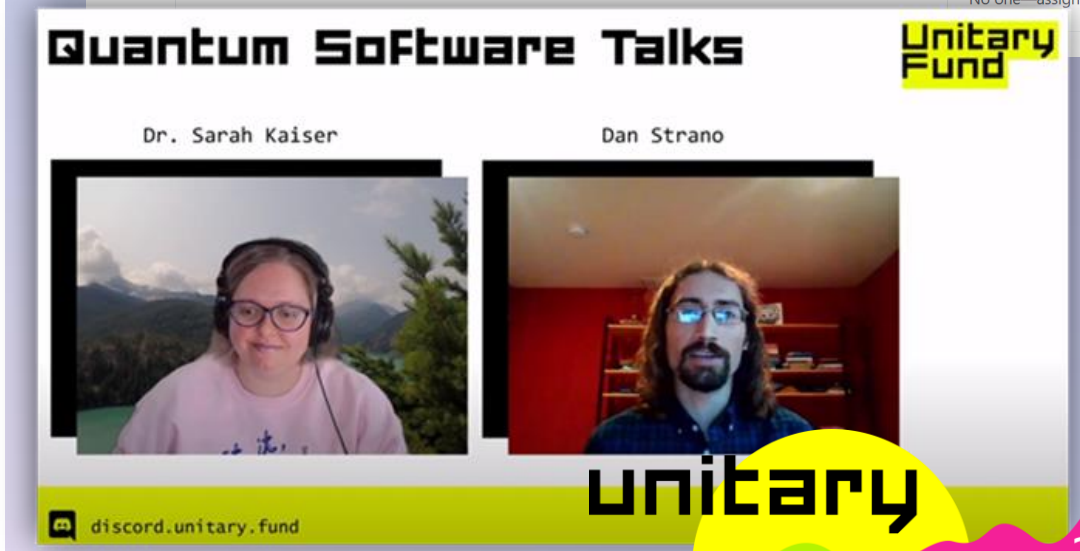
...and now I work here

Unitary Fund

because evolution is unitary

Creating a quantum technology ecosystem that benefits the most people.

https://unitary.fund



mitiq

unitary 2021 HACK

@crazy4pi314

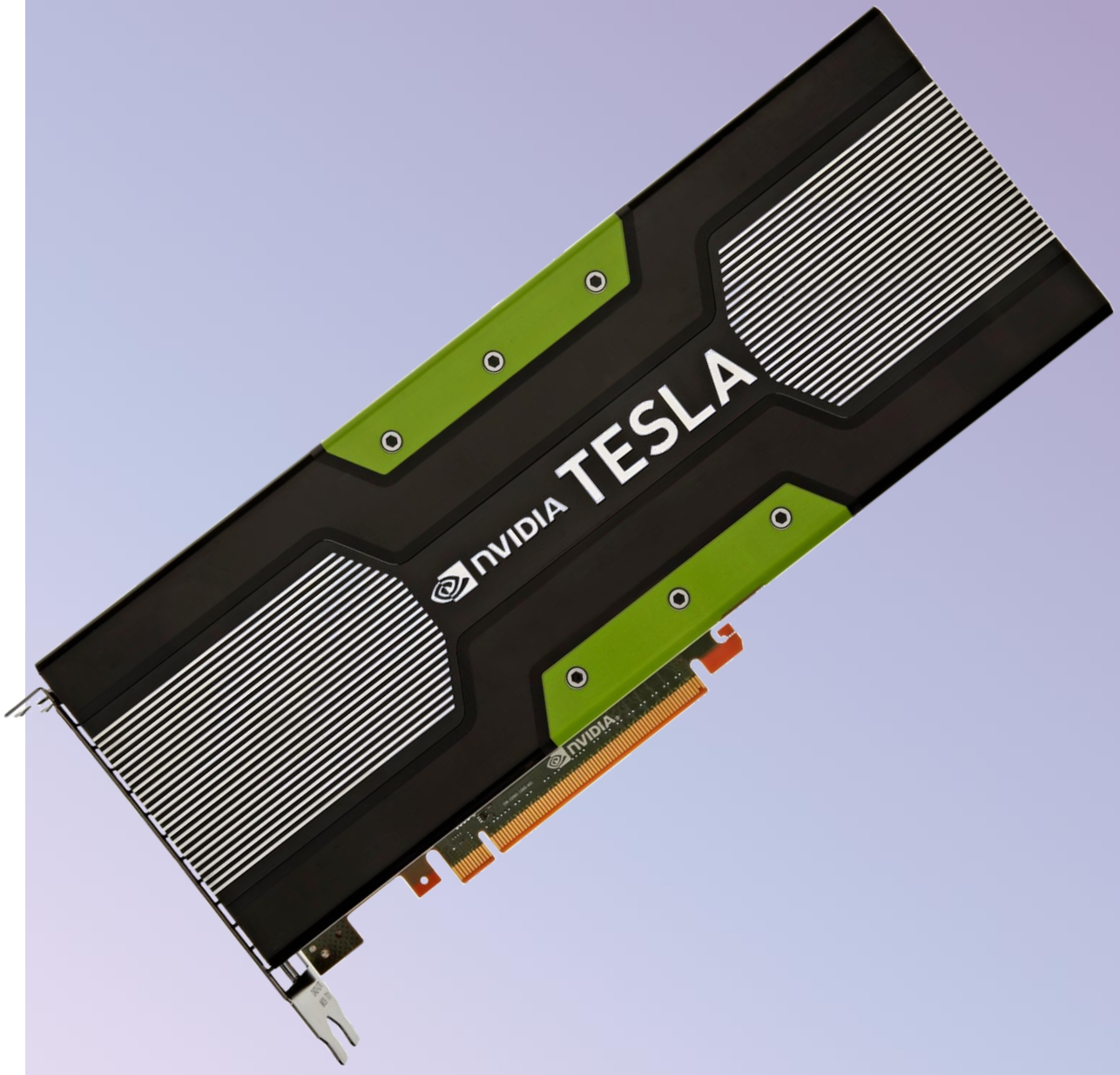
So…. what is quantum computing?

A question I ask myself frequently 🤔

Graphics cards are **hardware accelerators**.

GPUs are good at speeding up *some* highly parallelizable tasks:

- Rendering graphics
- Training + inferencing machine learning models
- Editing video and audio files
- Making your computer case hotter 🔥

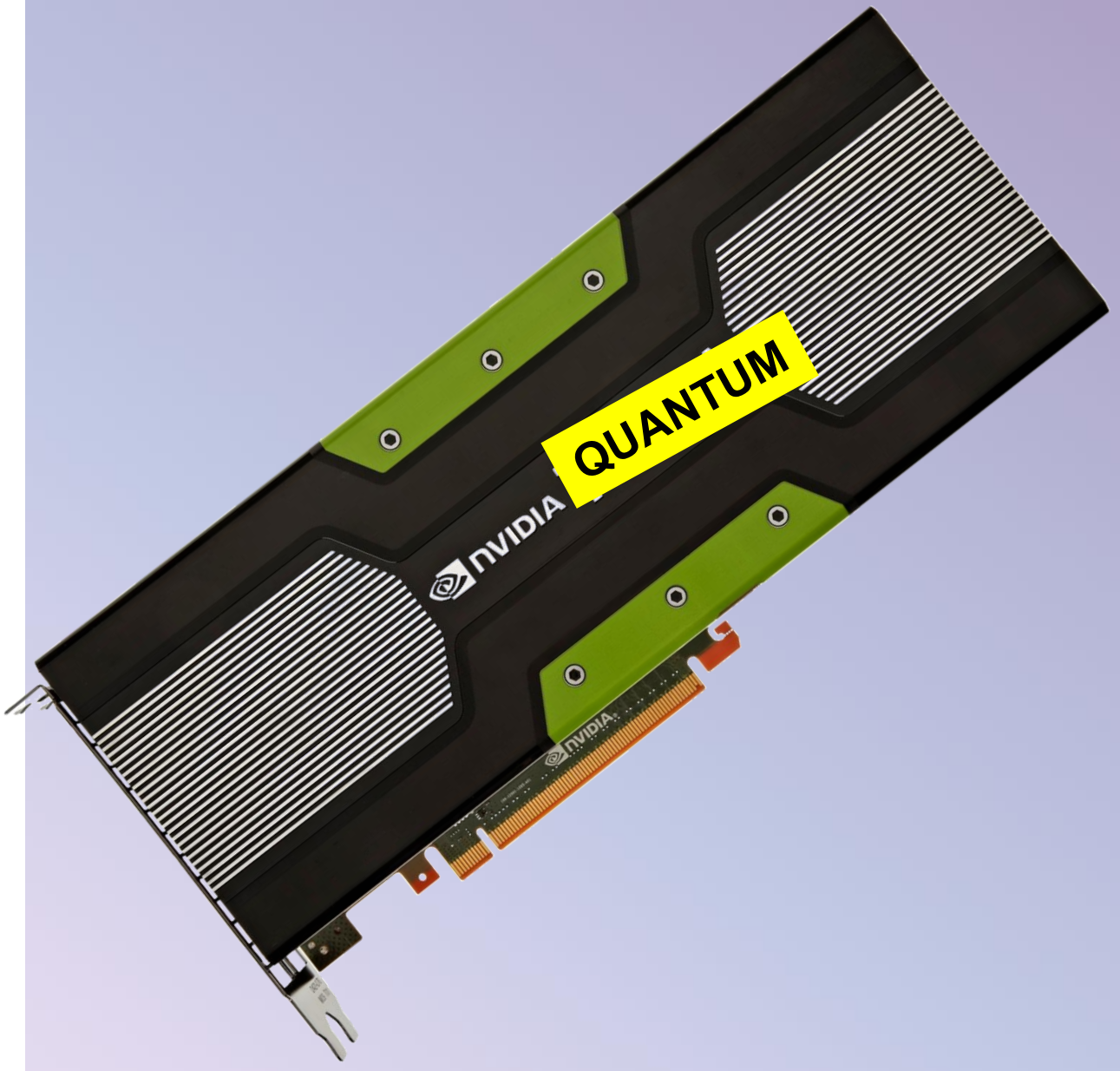


A quantum computer is a **hardware accelerator**.

Quantum computers are good at speeding up ??? tasks.

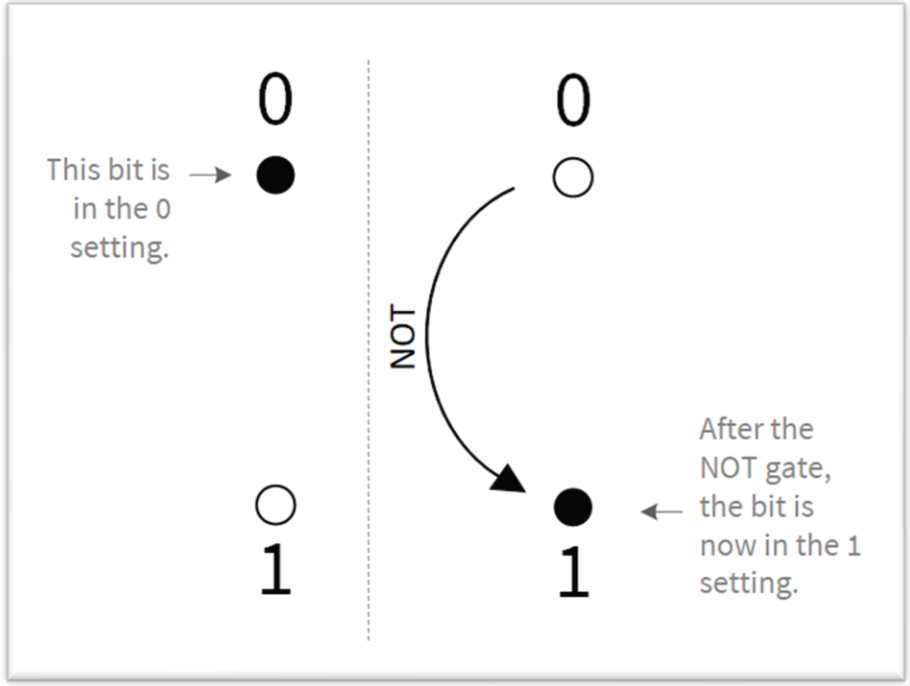
We know we can:

- Factor large integers
- Search unstructured data
- Simulate quantum systems
- …

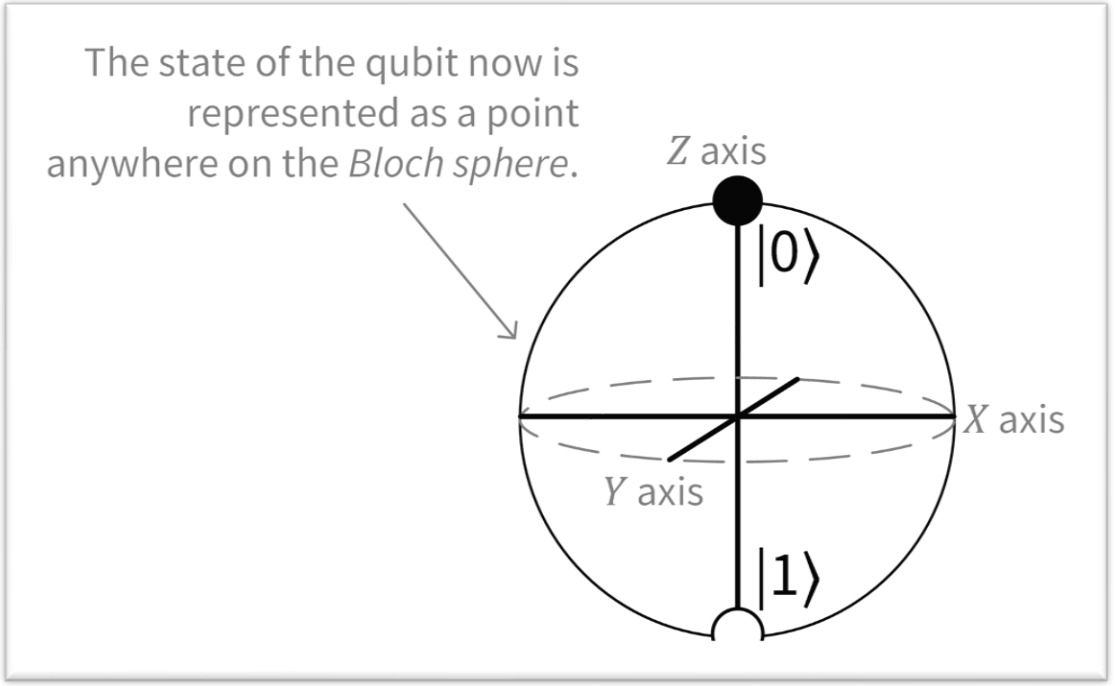


SIDEBAR:
Ok, but how is a quantum device different?

Normal (classical) bits



Quantum bits (qubits)



cgranade/**quantum-falsehoods.md**

NOTE: for informational purposes only, mental model only good for one qubit ☺

We **know** how to use hardware accelerators. We program them.

OpenCL: a framework for writing programs that execute across heterogeneous platforms.

OSS

CUDA: Industry/Hardware specific solutions for parallel computing.

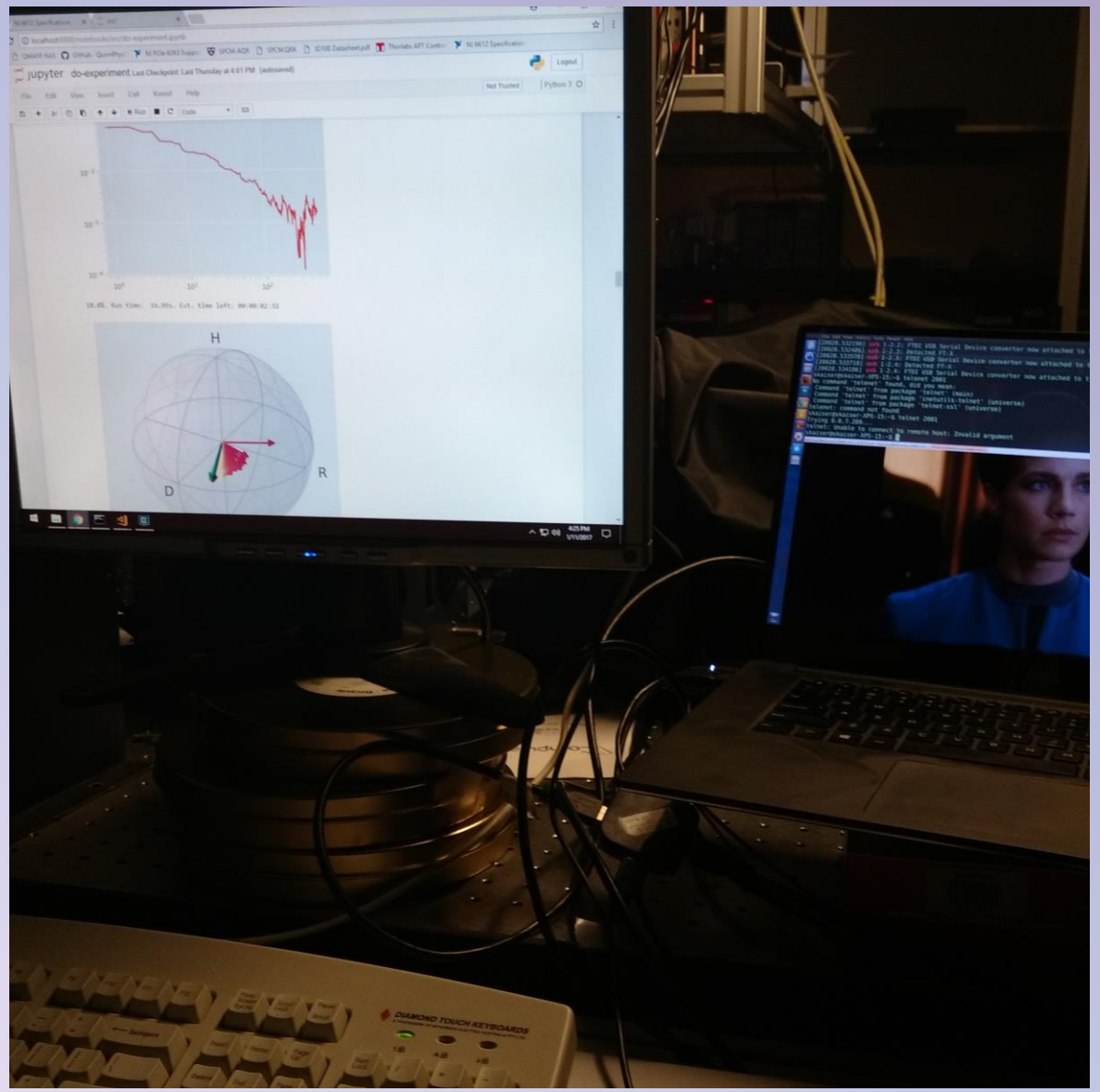
A quantum programming language for quantum computers??

OSS??

@crazy4pi314

Programming **concrete applications** helps us understand what to do next.

- How many qubits will we need?
- What are the right quantum algorithms?
- Who is going to develop it?



@crazy4pi314

Who can program quantum devices?

Full stack developers

Front-end developers

Machine learning scientists

Teachers

Data scientists

Dev ops

UI/UX designers

Physicists

Python developers

Chemists

Academics

Project managers

Mobile developer

Full stack developers

Front-end developers

Machine learning scientists

Teachers

Data scientists

Dev ops

You!

UI/UX designers

Physicists

Python developers

Chemists

Academics

Project managers

Mobile developer

Full stack developers

Front-end developers

Machine learning scientists

Teachers

Data scientists

Dev ops

EVERYONE!

UI/UX designers

Physicists

Python developers

Chemists

Academics

Project managers

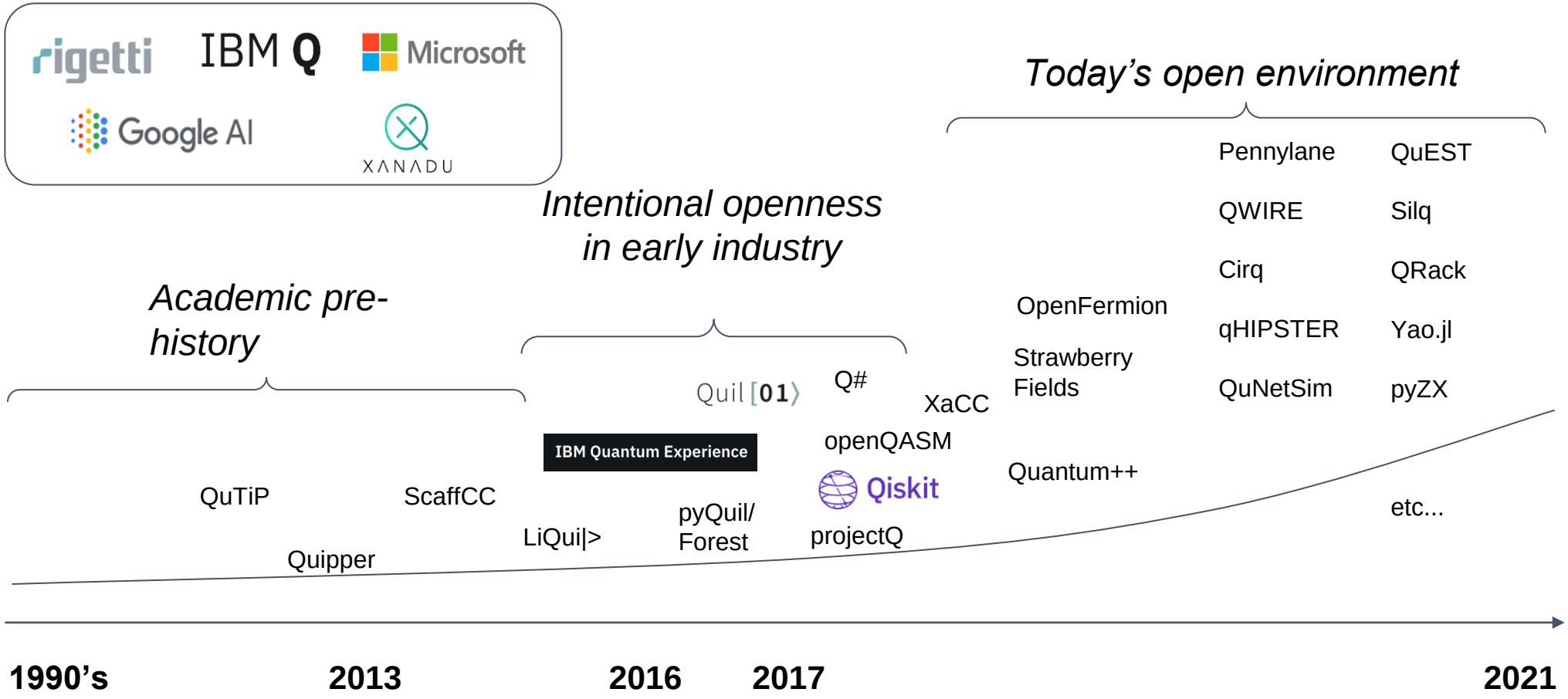
Mobile developer

Let’s program quantum computers.

With open source tools of course!

@crazy4pi314

The quantum ecosystem is **open**



* This timeline is representative, not precise

@crazy4pi314

The quantum stack

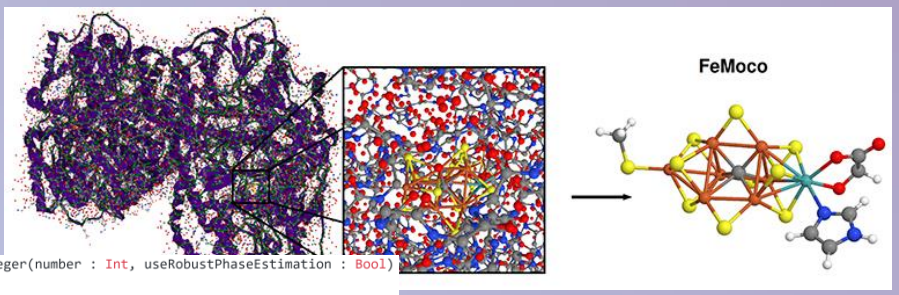
Quantum Applications — OSS

OSS — Libraries / Techniques

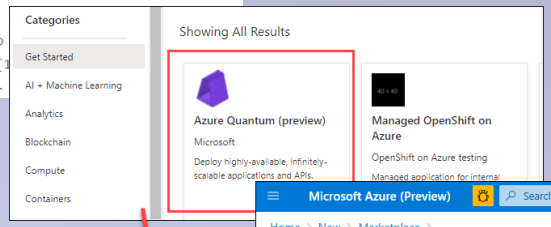
Quantum Language (Q#) — OSS

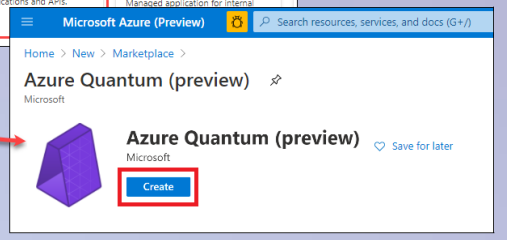
Intrinsic Operations

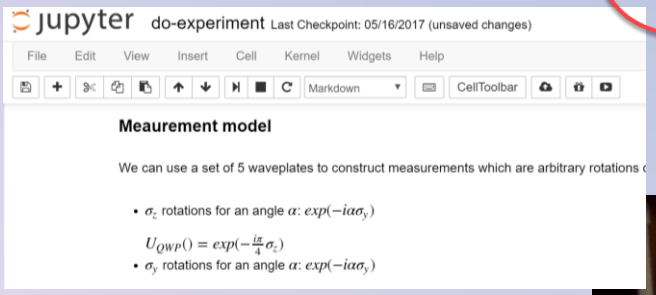
Quantum Hardware | Simulators — OSS

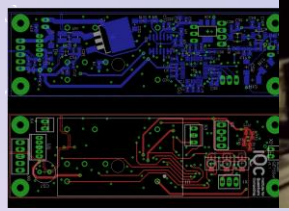



```
operation FactorSemiprimeInteger(number : Int, useRobustPhaseEstimation : Bool)
: (Int, Int) {
    // First check the most trivial case, if the provided number is even
    if (number % 2 == 0) {
        Message("An even number has been given; 2 is a factor.");
        return (number / 2, 2);
    }
    // These mutables will keep track of if we found the factors,
    // and if so, what they are. The default value for the factors
    // is (1,1).
    mutable foundFactors = false;
    mutable factors = (1, 1);

    repeat {
        // Next try to guess a number co-prime to
        // Get a random integer in the interval [1
        let generator = DrawRandomInt(1, number -
```



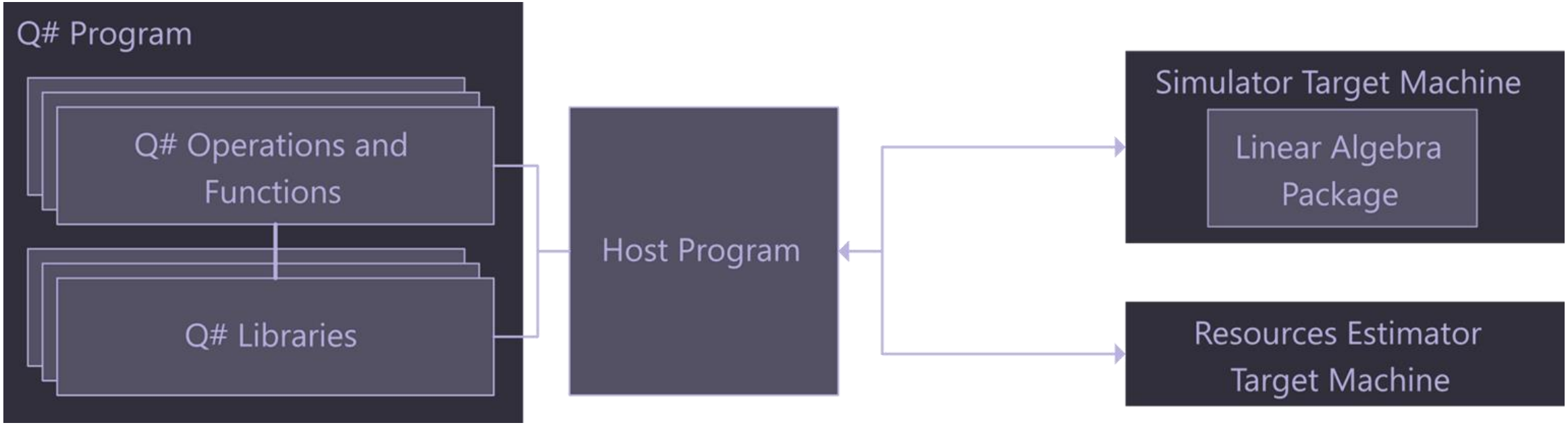






Meet Q# :
a domain-specific language for quantum programming

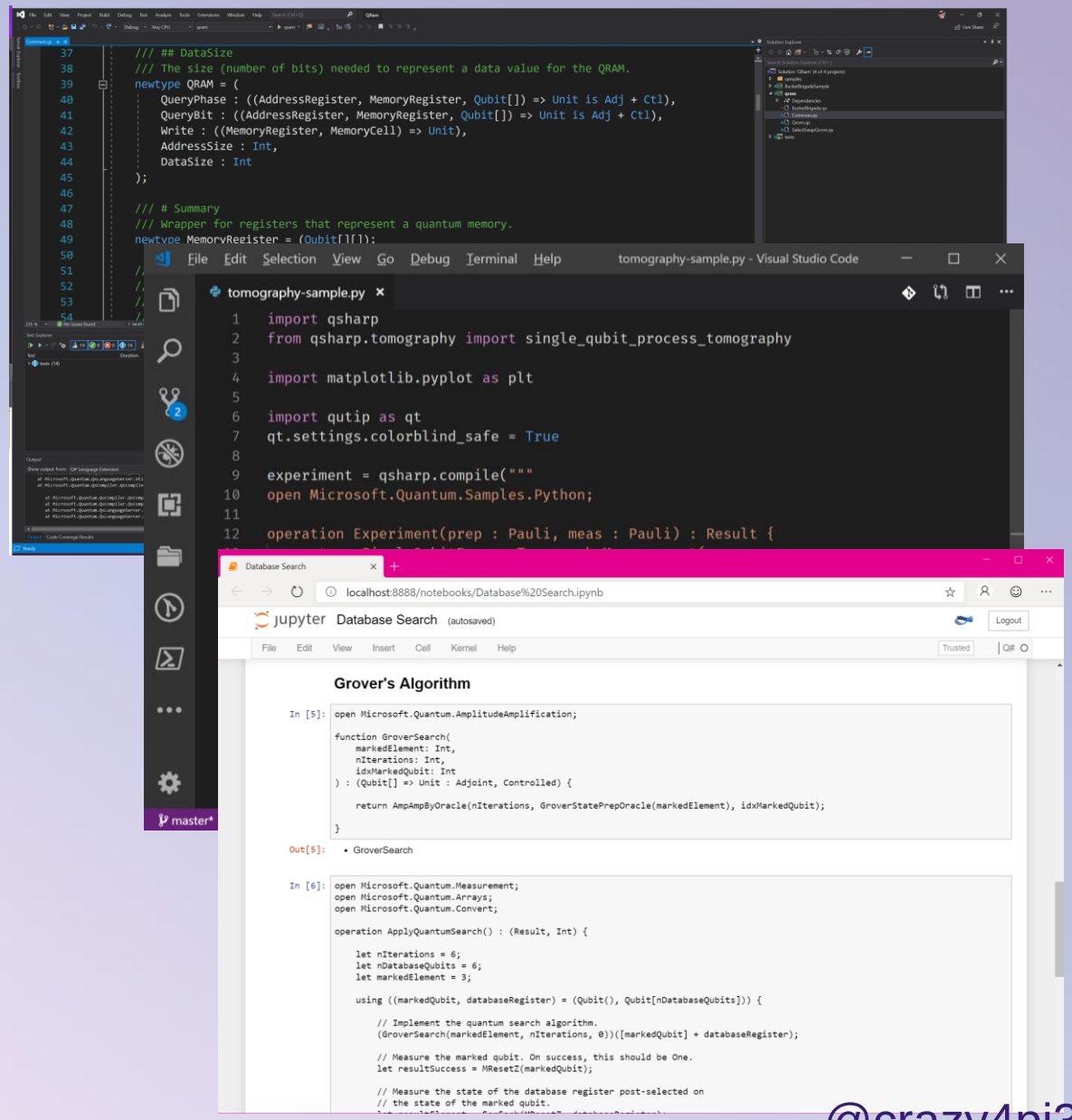
- Built on .NET Core, so works wherever you do
- Super extensible architecture
- Easy interop with Python, .NET languages, and more



Tools for quantum developers

The Quantum Development Kit offers lots of great tools for working with Q#

- Editor extensions and syntax highlighting and intelisense
- Great libraries to help bootstrap your applications
- Testing tools/examples
- Jupyter notebooks!
- Lots of built-in simulators and tools to build your own!
- Azure Quantum service to run your programs on hardware



@crazy4pi314

To the code!

https://bit.ly/osd-qsharp

Quantum computing is not magic, but we are ✨

Thanks for your attention 💖

TODO

Connect with me:

sckaiser.dev | @crazy4pi314

bit.ly/qsharp-book

My Co-Author Dr. Chris Granade:

cgranade/quantum-falsehoods.md

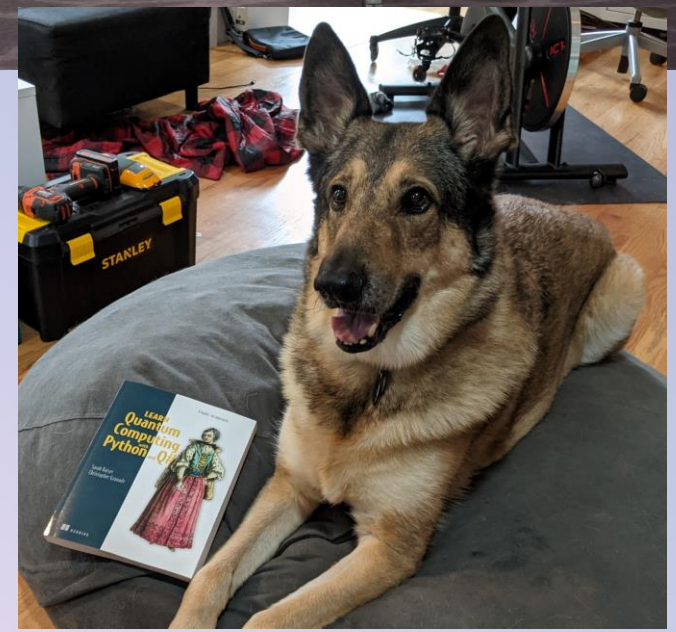
https://dev.to/cgranade

Quantum development kit:

docs.microsoft.com/quantum

Unitary Fund:

unitary.fund

Quantum open source foundation:

qosf.org