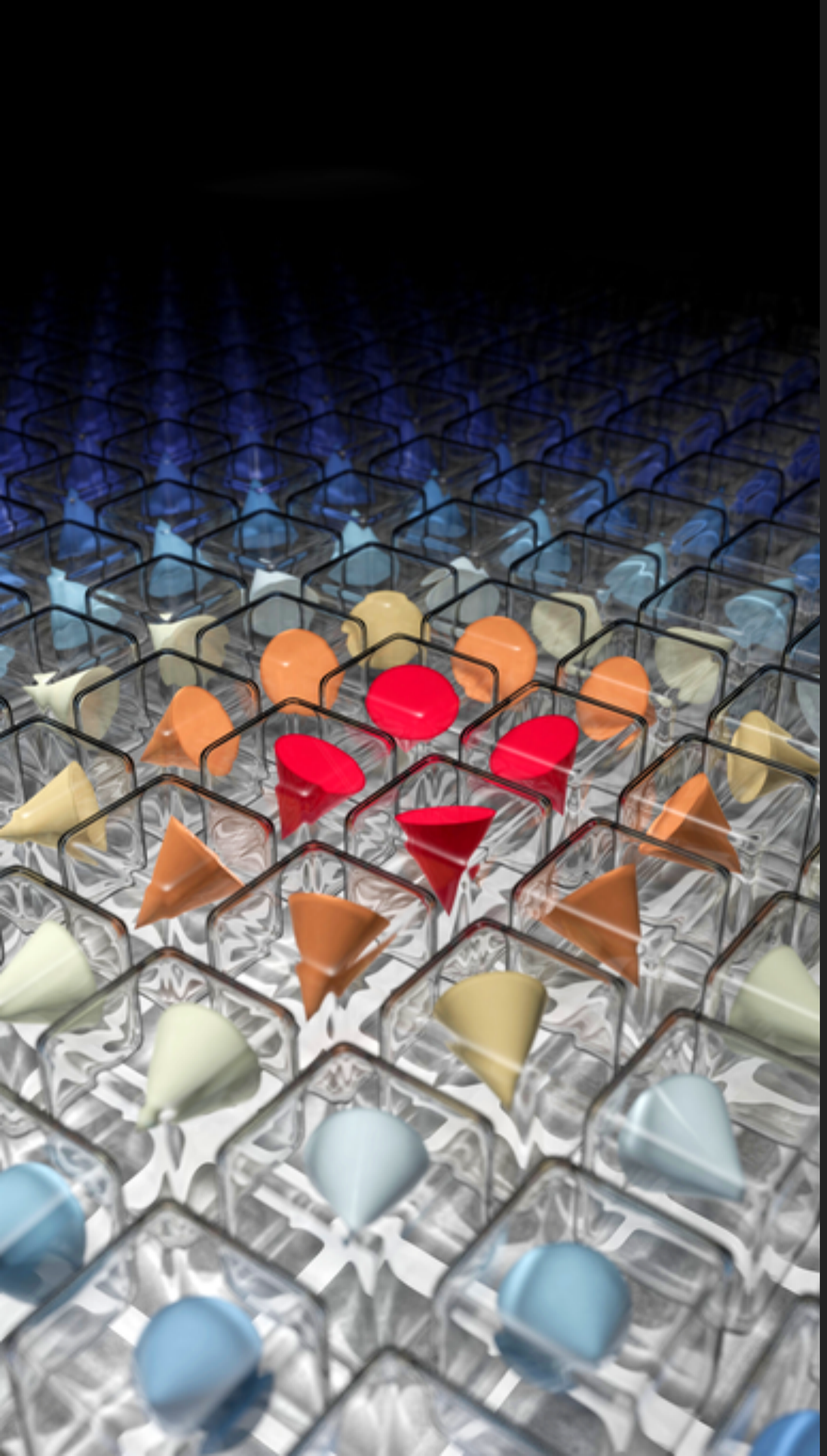
Marijan Beg[1,2] and H. Fangohr[1,2]

CASE STUDY: MICROMAGNETIC VIRTUAL RESEARCH ENVIRONMENT

[1]European XFEL GmbH, Holzkoppel 4, 22869 Schenefeld, Germany
[2]University of Southampton, Highfield, SO17 1BJ Southampton, UK

30 October 2018, OpenDreamKit Review Meeting, Luxembourg

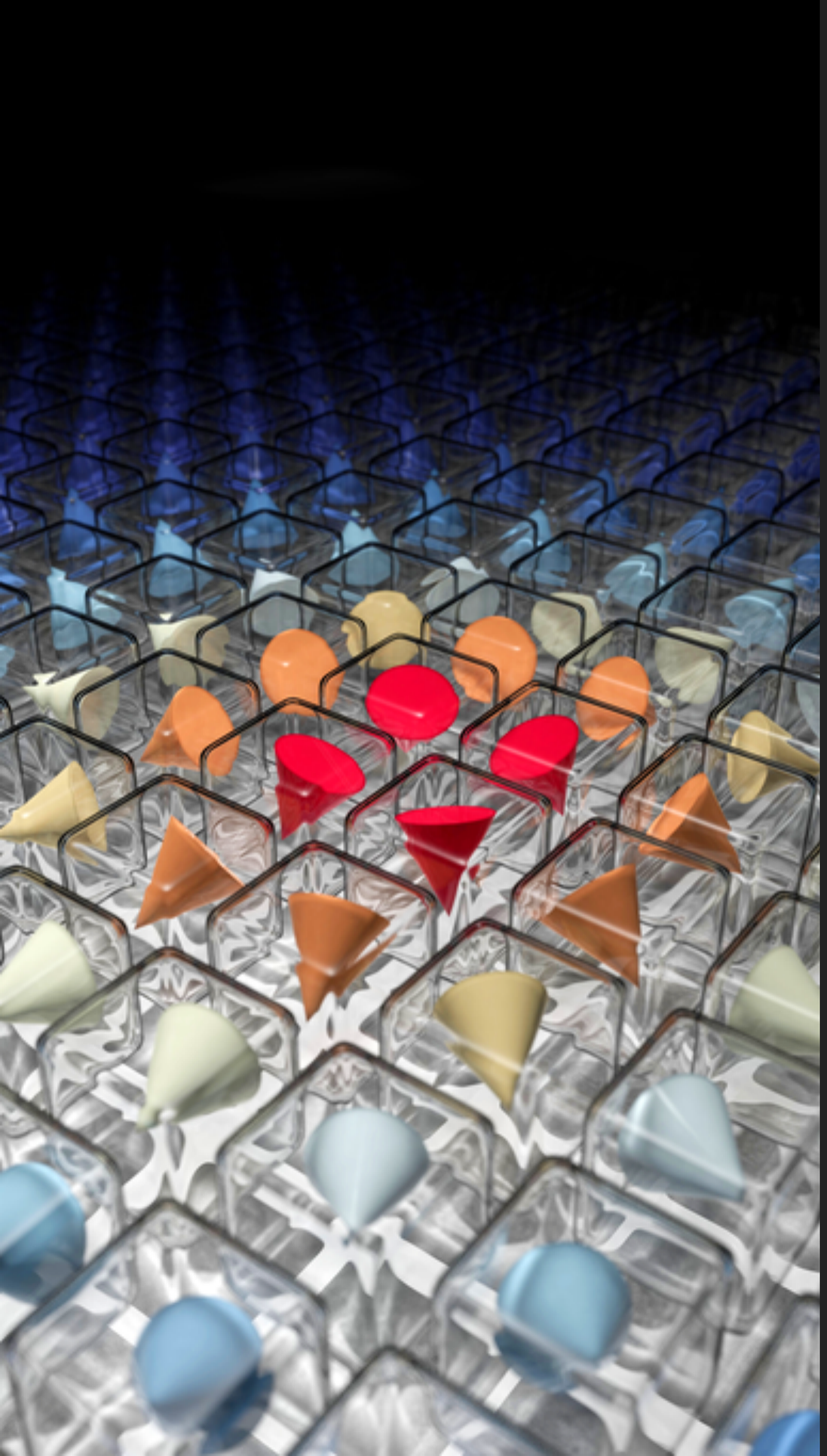


PART 0

OUTLINE

OUTLINE

1. Motivation
2. Micromagnetics and OOMMF
3. Typical computational workflow
4. JOOMMF Virtual Research Environment
5. Dissemination and evaluation
6. Summary



PART 1

MOTIVATION

COMPUTATIONAL SCIENCE

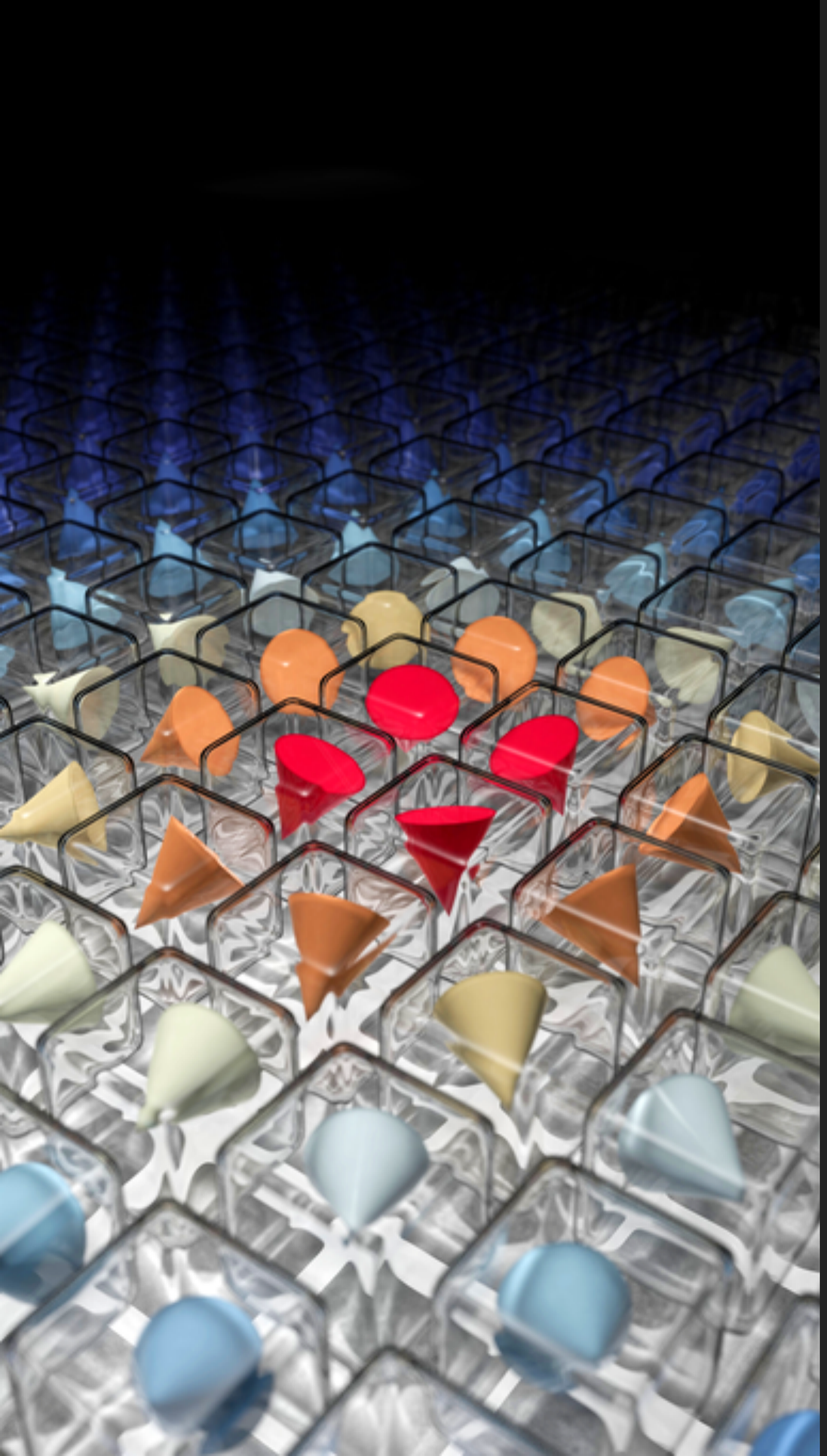
- Emerges as the **third pillar** of research and development in academia and in industry.
- Computational studies **complement experimental and theoretical studies**.
- At times **the only feasible way** to address research challenges, effective industrial design and engineering of various products and systems.

REPRODUCIBILITY IN COMPUTATIONAL SCIENCE

- Often results are **not reproducible**. Why not?
  - **Complex workflow**: simulation codes, configuration files, execution of many runs, post-processing, image creation, report/publication writing
- "Sharing data and software **gives away competitive edge**."
- "**Don't know how.**"
- "It is **too much work**."

GOAL: MAKE COMPUTATIONAL SCIENCE MORE EFFECTIVE

- Faster to use, more accessible, and more reproducible
- **Goal of the OpenDreamKit** project via a **Virtual Research Environment** (VRE)

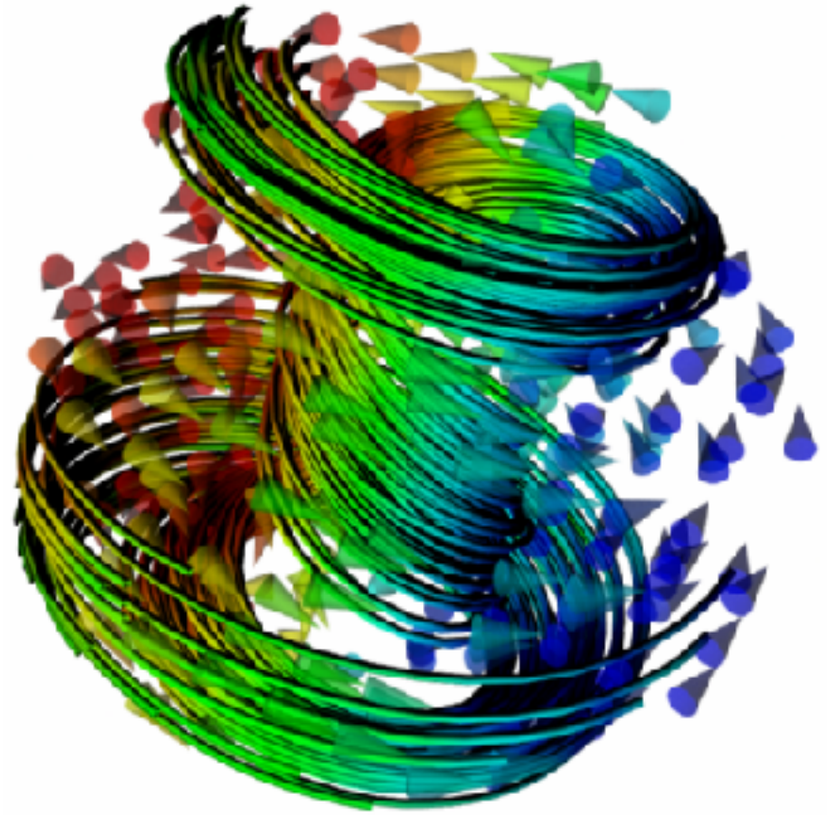


PART 2

MICROMAGNETICS AND OOMMF

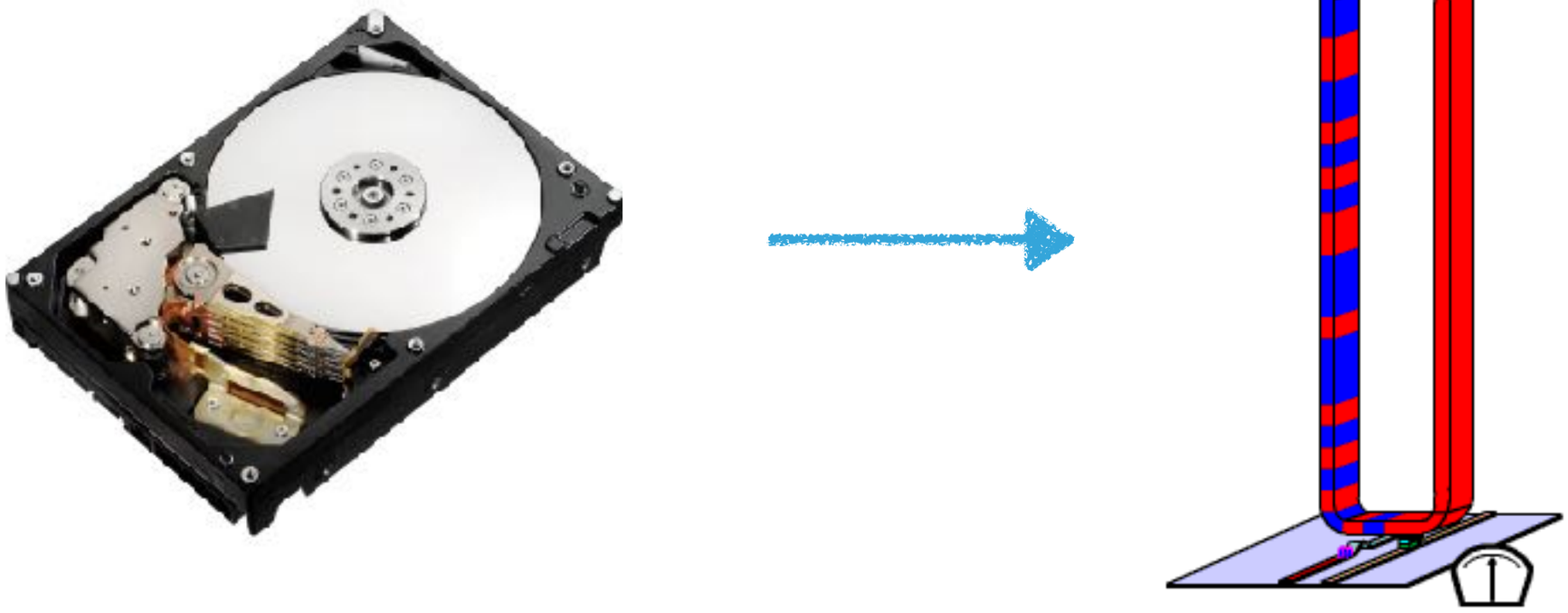
MICROMAGNETICS

"… is the study, modelling, and simulation of magnetic materials and their behaviour at the nanometer scale."



WHY DO WE NEED IT?

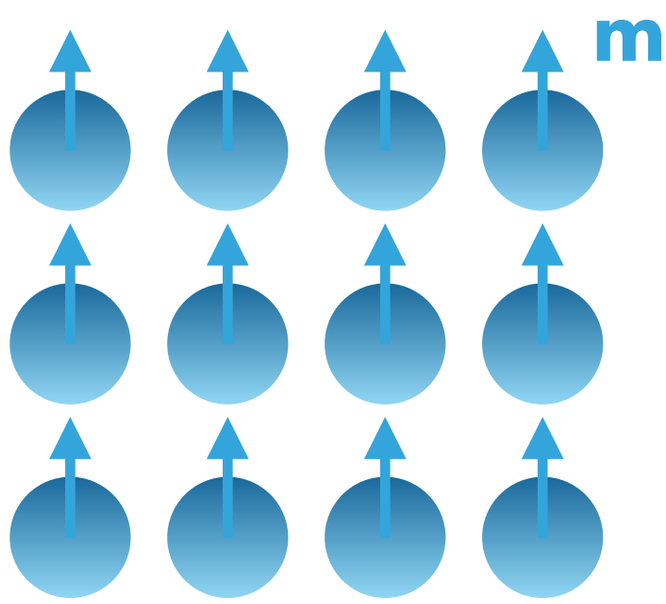
- In conventional semiconductor electronics and magnetic recording, **we already reached the physical limits**.
- We need **radically different approaches** for the development of future data storage and processing devices.
- One of the several possible candidates is **spintronics**.



MAGNETISATION

- The main unknown in micromagnetics is the **continuous vector field.**

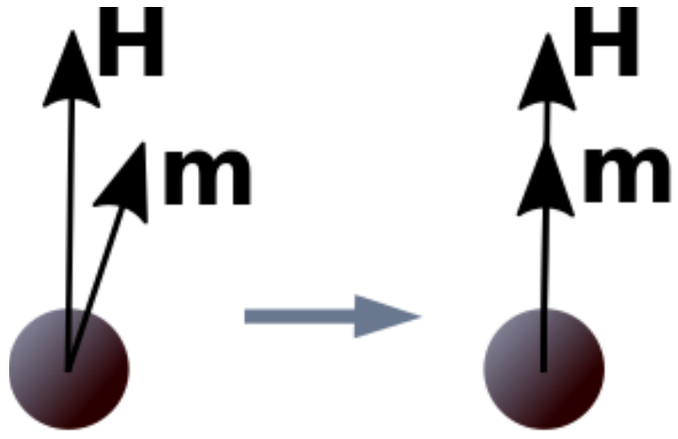
$$\mathbf{M} = \mathbf{M}(\mathbf{r}, t) \qquad \mathbf{M} : \mathbb{R}^3 \rightarrow \mathbb{R}^3$$



ZEEMAN ENERGY

- Tends to align magnetisation **parallel to the external field**.

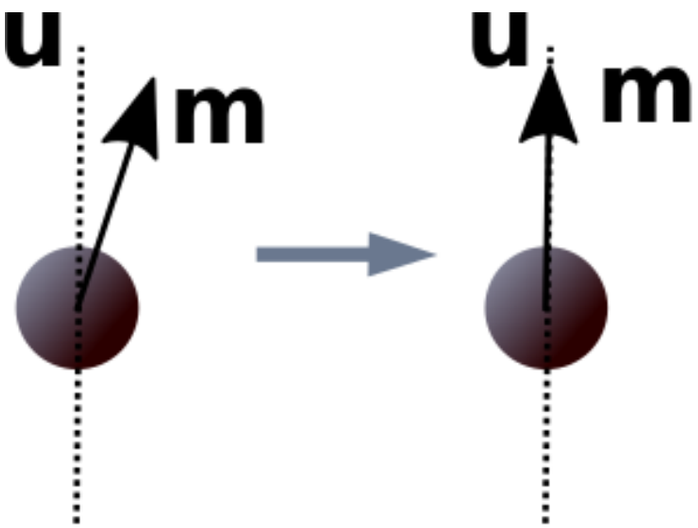
$$w_\mathrm{z} = -\mu_0 M_\mathrm{s} \mathbf{m} \cdot \mathbf{H}$$



UNIAXIAL ANISOTROPY ENERGY

- Wants magnetisation **parallel or antiparallel** to the anisotropy axis $\mathbf{u}$.

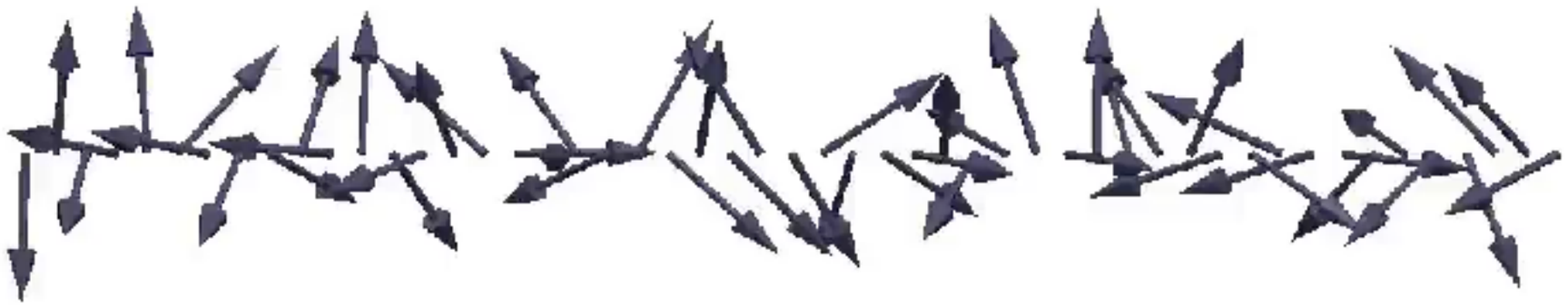
$$w_\mathrm{a} = -K(\mathbf{m}\cdot\mathbf{u})^2$$



EXCHANGE ENERGY

- Tends to align all magnetic moments **parallel to each other**.

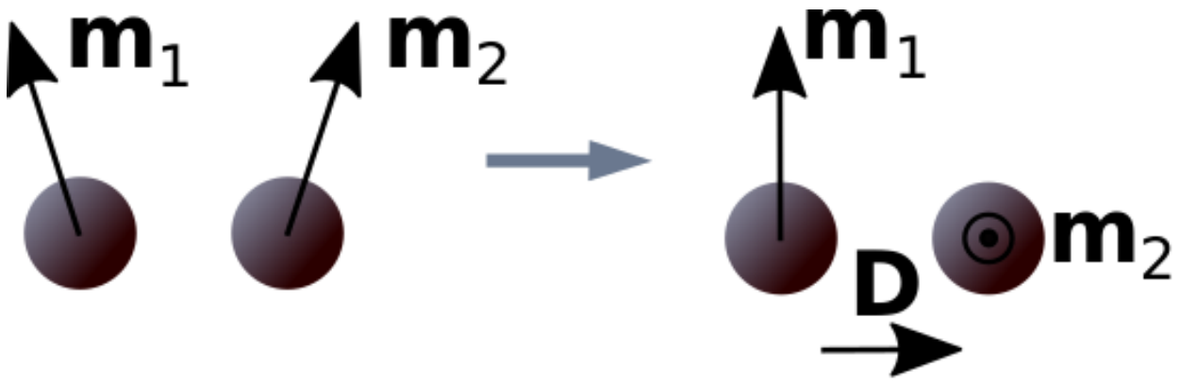
$$w_{\mathrm{ex}} = A[(\nabla m_x)^2 + (\nabla m_y)^2 + (\nabla m_z)^2] = A(\nabla \mathbf{m})^2$$



DZYALOSHINSKII–MORIYA ENERGY

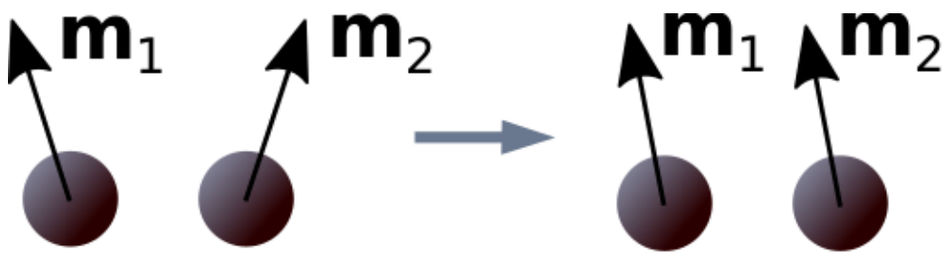
- Wants neighbouring magnetic moments **perpendicular to each other**.

$$w_{\mathrm{dmi}} = D\mathbf{m} \cdot (\nabla \times \mathbf{m})$$



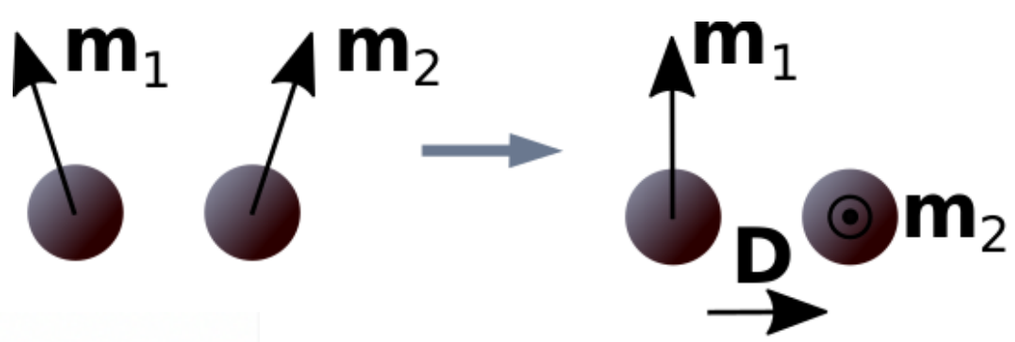
EXCHANGE AND DMI

- Exchange aligns all spins **parallel** to each other with no preferential direction.



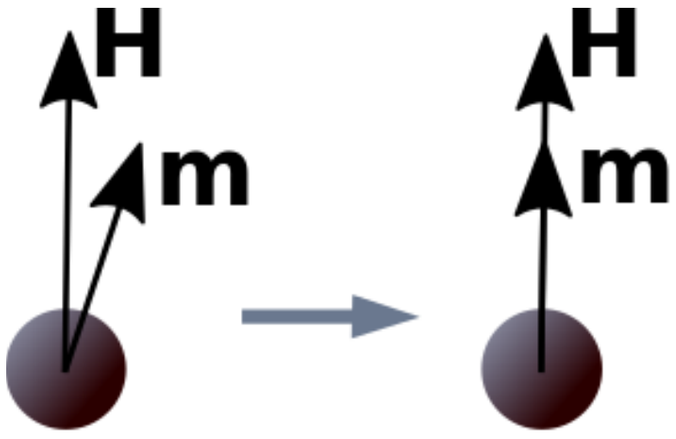
$$w_{\mathrm{ex}} = A(\nabla \mathbf{m})^2$$

- DMI wants neighbouring spins **perpendicular** to each other.

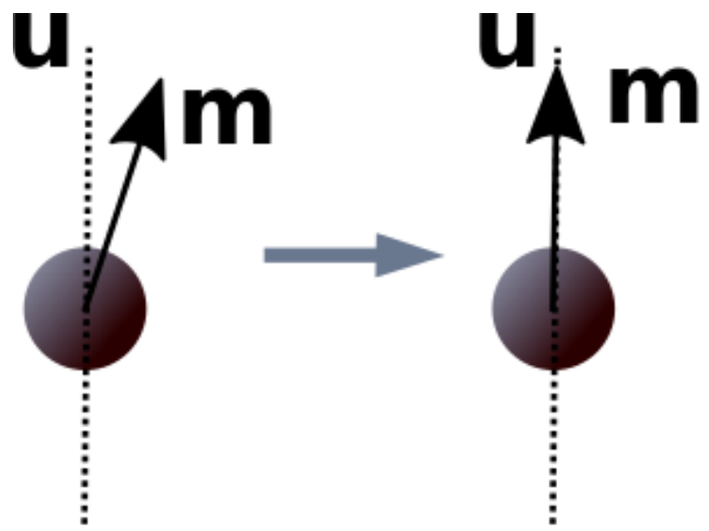


$$w_{\mathrm{dmi}} = D\mathbf{m} \cdot (\nabla \times \mathbf{m})$$

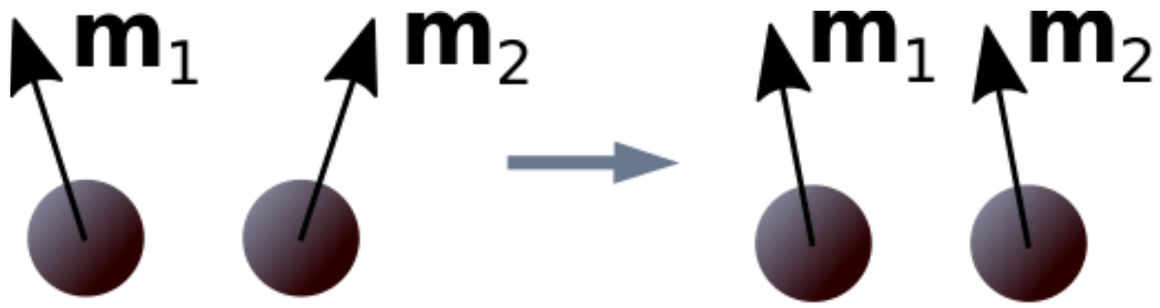
MORE COMPLICATED CASE



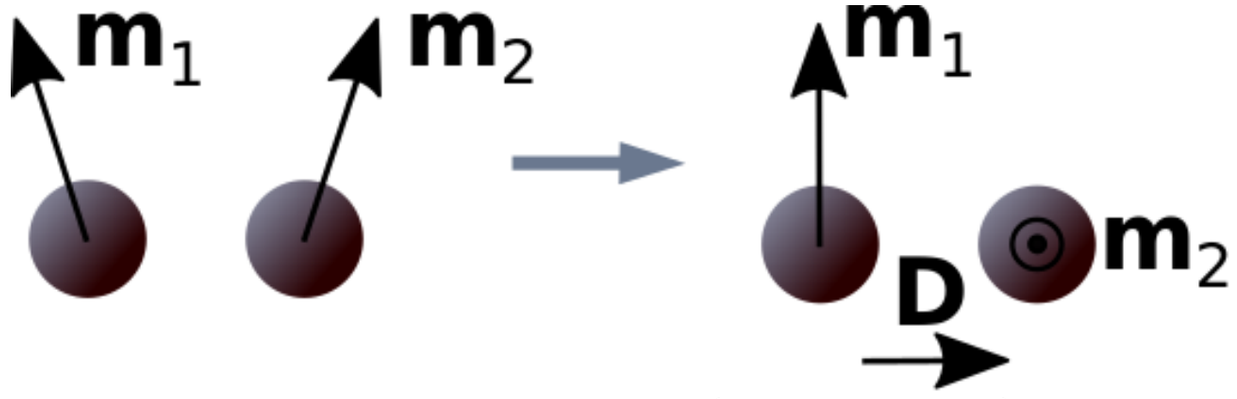
$$w_{\mathrm{z}} = -\mu_0 M_{\mathrm{s}} \mathbf{m} \cdot \mathbf{H}$$



$$w_{\mathrm{a}} = -K(\mathbf{m} \cdot \mathbf{u})^2$$

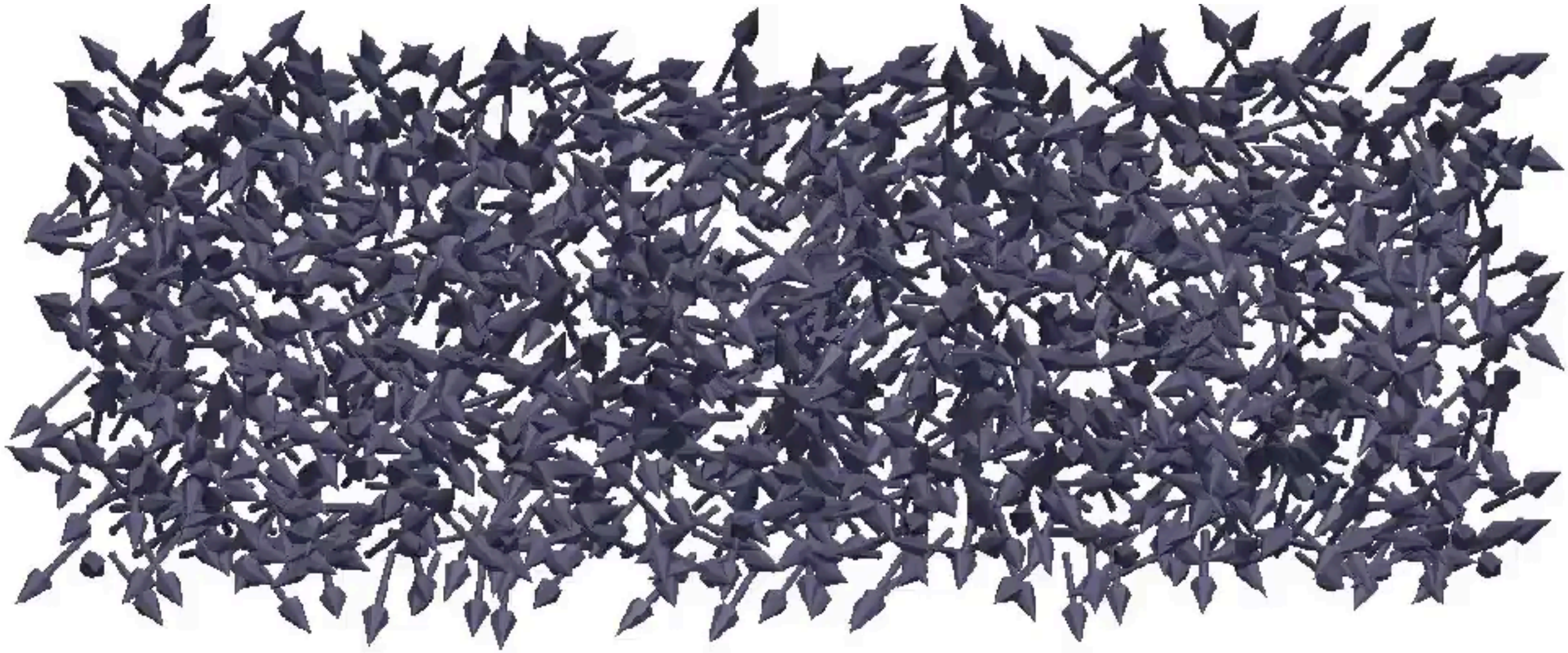


$$w_{\mathrm{ex}} = A(\nabla \mathbf{m})^2$$



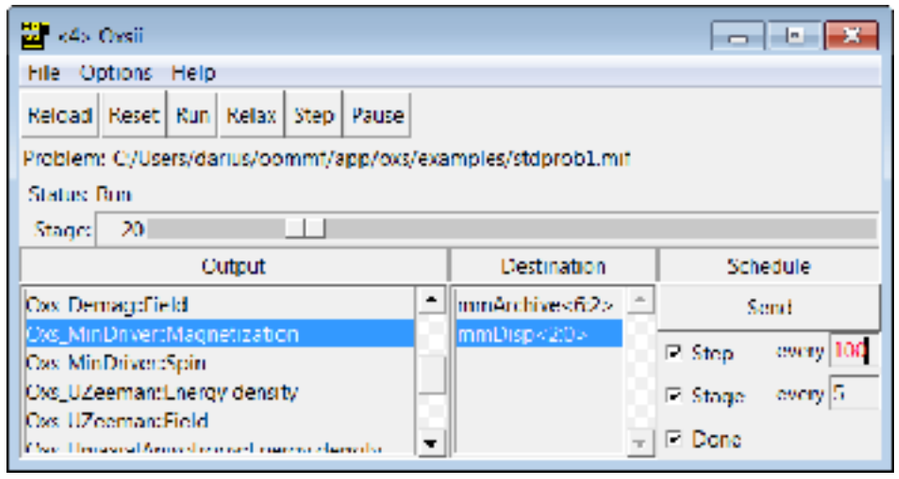
$$w_{\mathrm{dmi}} = D\mathbf{m} \cdot (\nabla \times \mathbf{m})$$

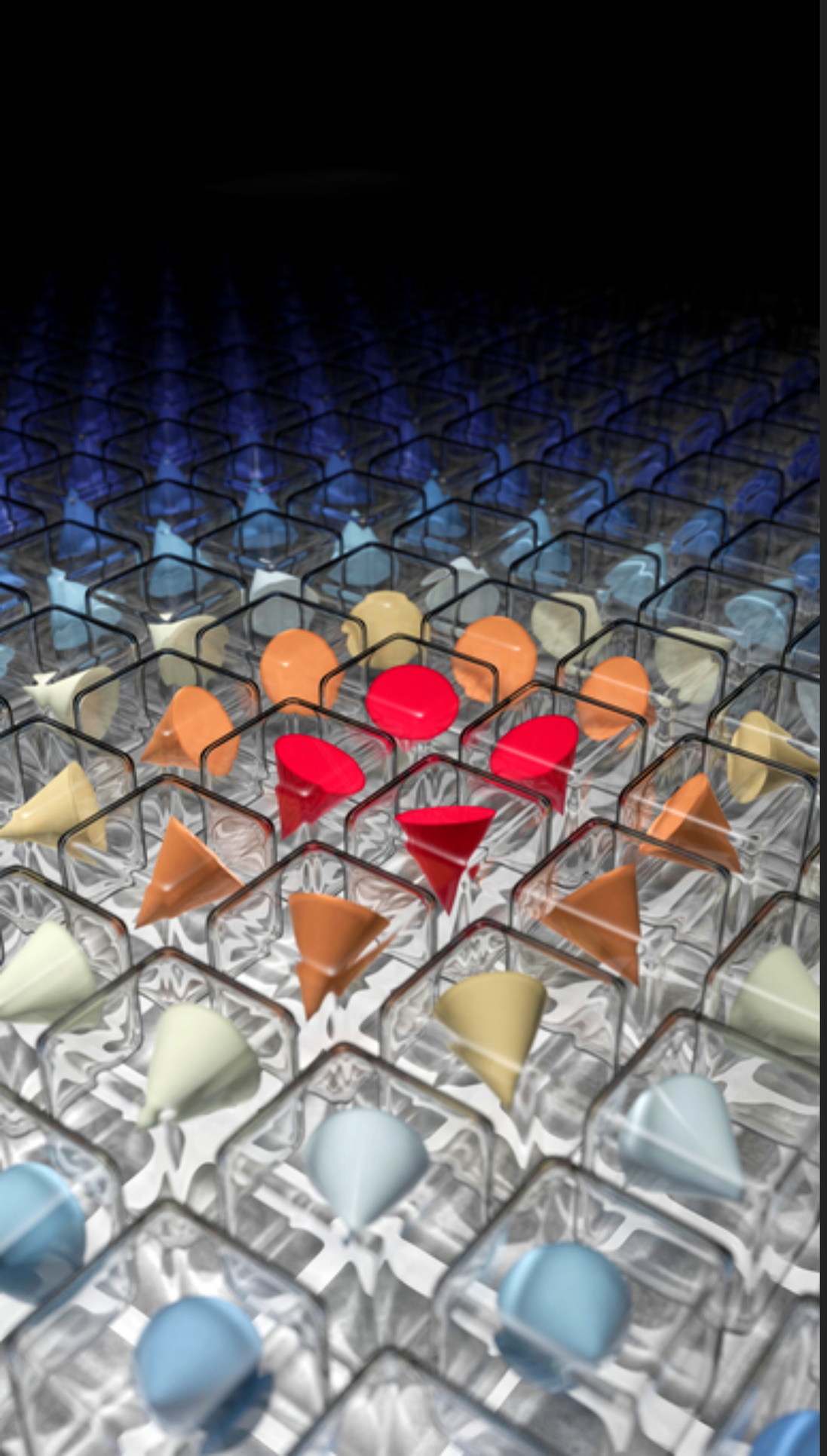
2D SAMPLE



OOMMF (OBJECT ORIENTED MICROMAGNETIC FRAMEWORK)

- Probably the most widely used micromagnetic simulation tool
- Developed at National Institute for Standards and Technology (NIST), US, since ~1998 by Michael Donahue & Don Porter
- Cited in over 2200 times scientific publications (likely more)
- Written in C++ & Tcl
- math.nist.gov/oommf/

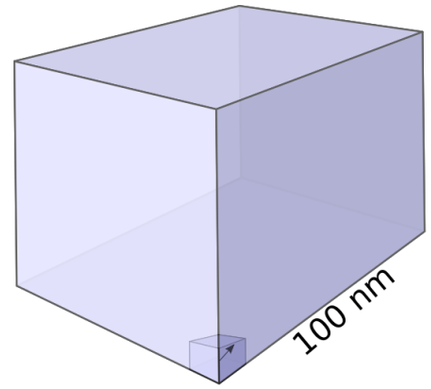


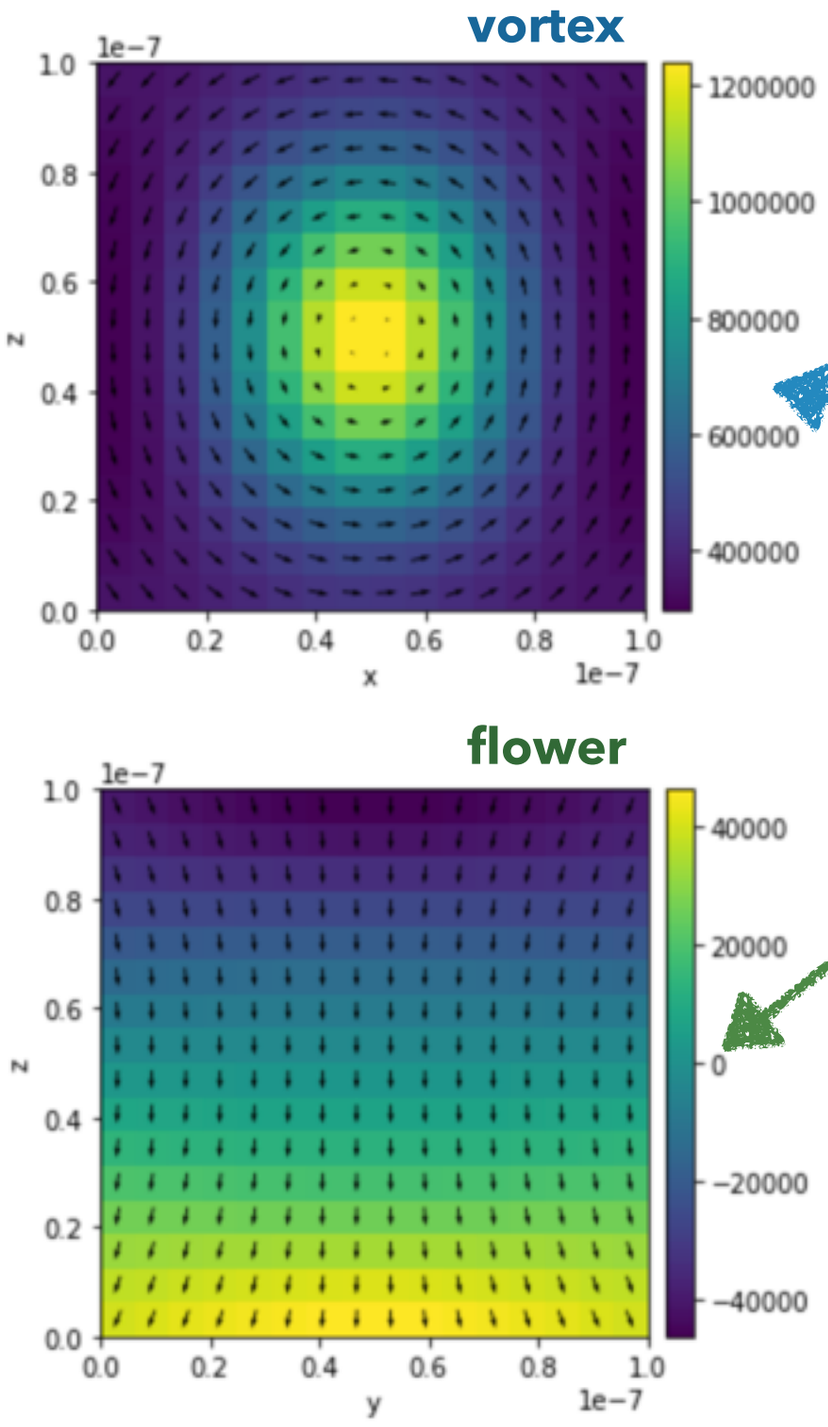


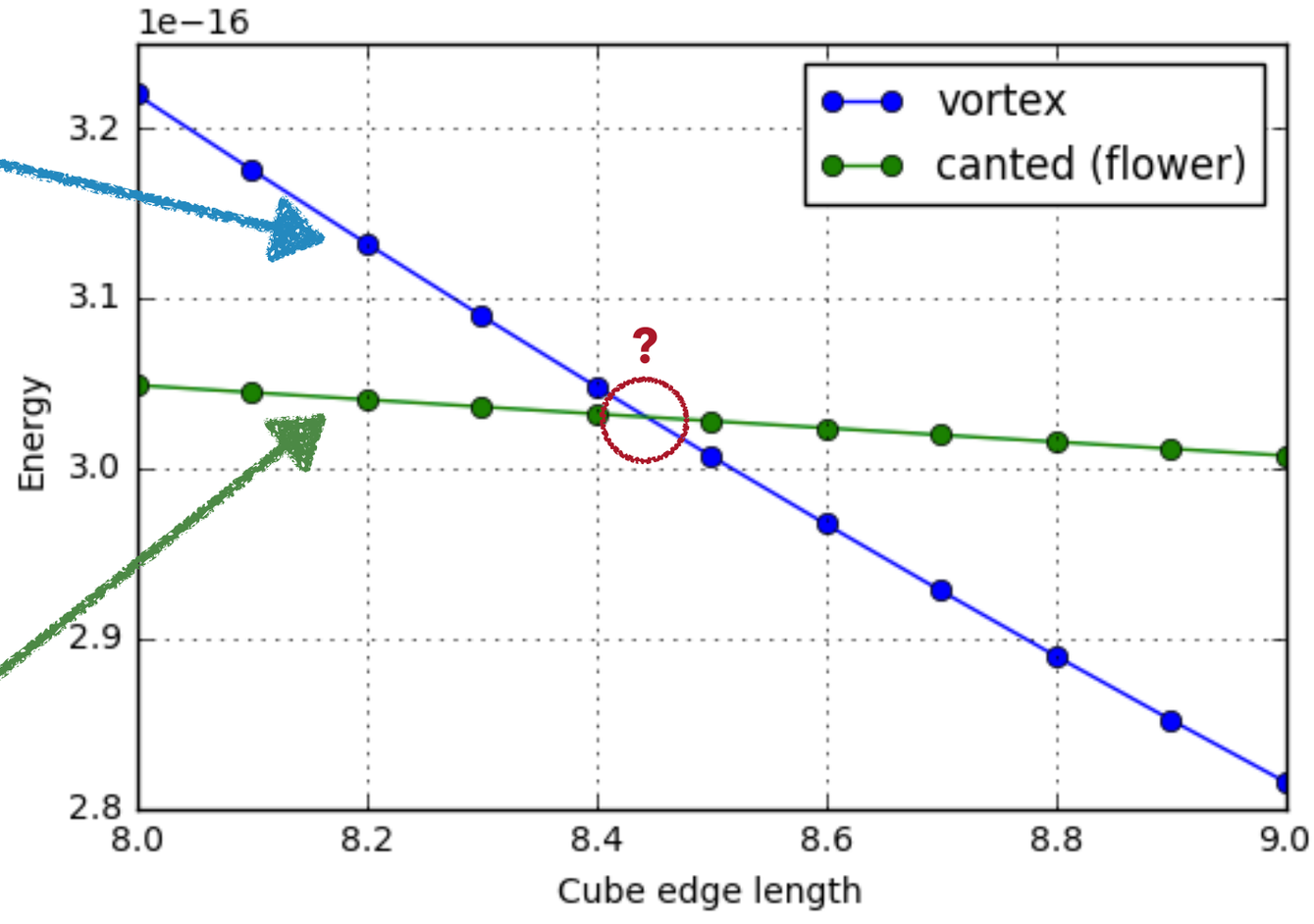
PART 3

TYPICAL COMPUTATIONAL WORKFLOW

RESEARCH QUESTION



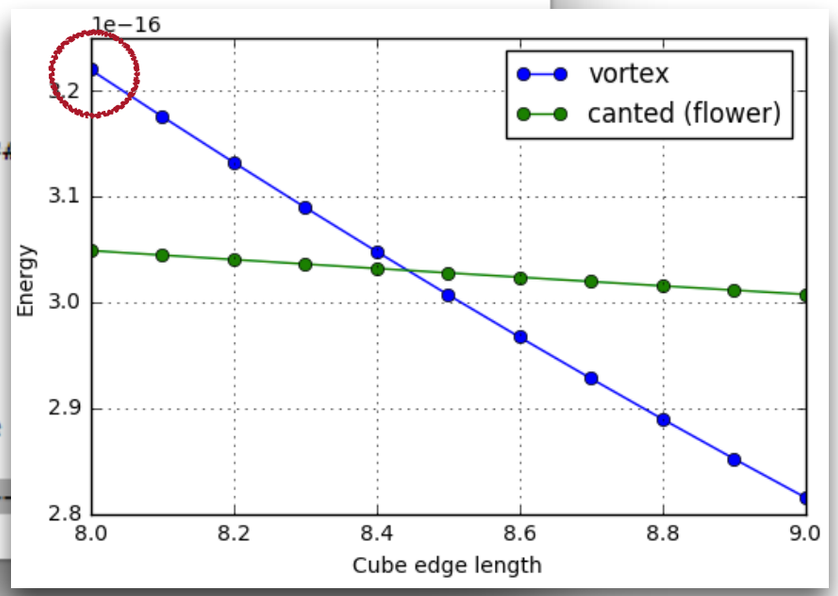




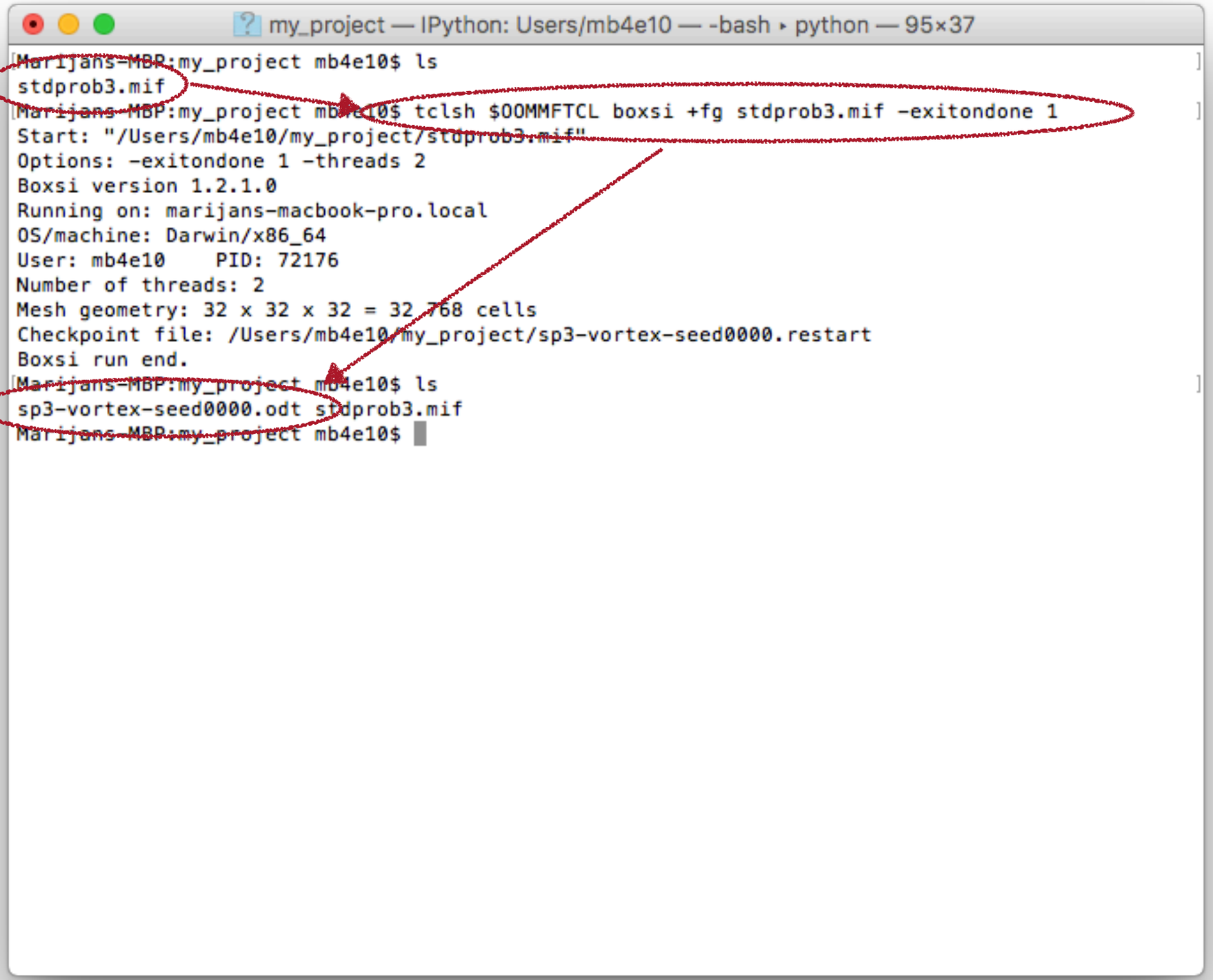
STEP 1: WRITE SIMULATION FILE

my_project — IPython: Users/mb4e10 — emacs -nw stdprob3.mif — 95×37

```
# MIF 2.1
# MIF Example File: stdprob3.mif
# Description: Sample problem description for muMAG Standard Problem #3

set pi [expr {4*atan(1.0)}]
set mu0 [expr {4*$pi*1e-7}]

Parameter seed 0
RandomSeed $seed  ;# Initialize seed to {} to get a seed
## value from the system clock.


########################################################################
# Simulation parameters

Parameter L 8     ;# Cube dimension, in units of exchange length
Parameter N 32    ;# Number of cells along one edge of cube

Parameter initial_state "vortex" ;# Initial state should be
## one of "uniform", "vortex", "canted", "cantedvortex", "twisted",
## "random" or "file <filename>"; in the last case <filename> is the
## name of a file to use as the initial configuration.

Parameter stop 1e-3

########################################################################
# Auxiliary variables:

# Work out Ms so magnetostatic energy density, Km=0.5*mu0*Ms^2,
# is 1e6 J/m^3
set Km 1e6
set Ms [expr {sqrt(2*$Km/$mu0)}]

# Arbitrarily set cube dimension to 100 nm, and compute cellsize
# exchange length based on parameters L and N.
-uu-:---F1  stdprob3.mif   Top L1     (Fundamental)----------
```



STEP 2: RUN SIMULATION



STEP 3: READ DATA

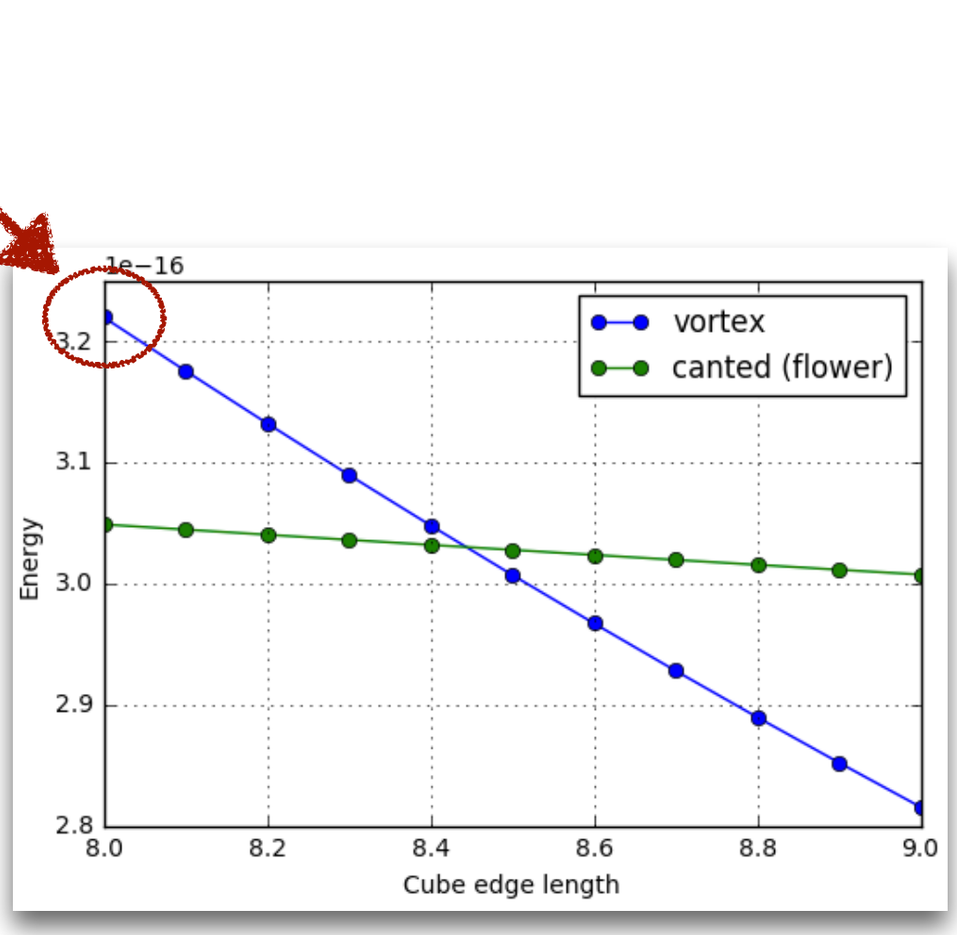
my_project — IPython: Users/mb4e10 — emacs -nw sp3-vortex-seed0000.odt — 95×37

```
# ODT 1.0
# Table Start
# Title: mmArchive Data Table, Wed Nov 16 20:54:28 GMT 2016
# Columns: {Oxs_CGEvolve::Max mxHxm} {Oxs_CGEvolve::Total energy} {Oxs_CGEvolve::Delta E} {Oxs\
_CGEvolve::Bracket count} {Oxs_CGEvolve::Line min count} {Oxs_CGEvolve::Conjugate cycle count}\
 {Oxs_CGEvolve::Cycle count} {Oxs_CGEvolve::Cycle sub count} {Oxs_CGEvolve::Energy calc count}\
 Oxs_UniaxialAnisotropy::Energy Oxs_UniformExchange::Energy {Oxs_UniformExchange::Max Spin Ang\
} {Oxs_UniformExchange::Stage Max Spin Ang} {Oxs_UniformExchange::Run Max Spin Ang} Oxs_Demag:\
:Energy    Oxs_MinDriver::Iteration {Oxs_MinDriver::Stage iteration} Oxs_MinDriver::Stage Oxs_\
MinDriver::mx    Oxs_MinDriver::my    Oxs_MinDriver::mz
# Units:             A/m                                J                          J            \
         {}                             {}                             {}                       \
             {}                             {}                             {}                   \
               J                             J                            deg                   \
                    deg                                 deg                            J \
                     {}                             {}                          {}              \
   {}                    {}                   {}
             0.00097778028256529097          3.2257415663518404e-16      0                      \
       353                             326                            7                         \
           340                             333                            680                   \
             5.4172367330709765e-17         1.780007679106069e-16          11.011344278380\
658                    90.000000000000014                         90.000000000000014            \
  9.0401021393867362e-17     670                          670                          0        \
           0.40912126717720015  4.6139202285652408e-17 -1.9801501518039851e-16
# Table End
```

```
-:---F1  sp3-vortex-seed0000.odt    All L1     (Fundamental)-----------------------------------
File mode specification error: (error "Buffer format not recognized")
```

REPEAT STEPS 1, 2, 3

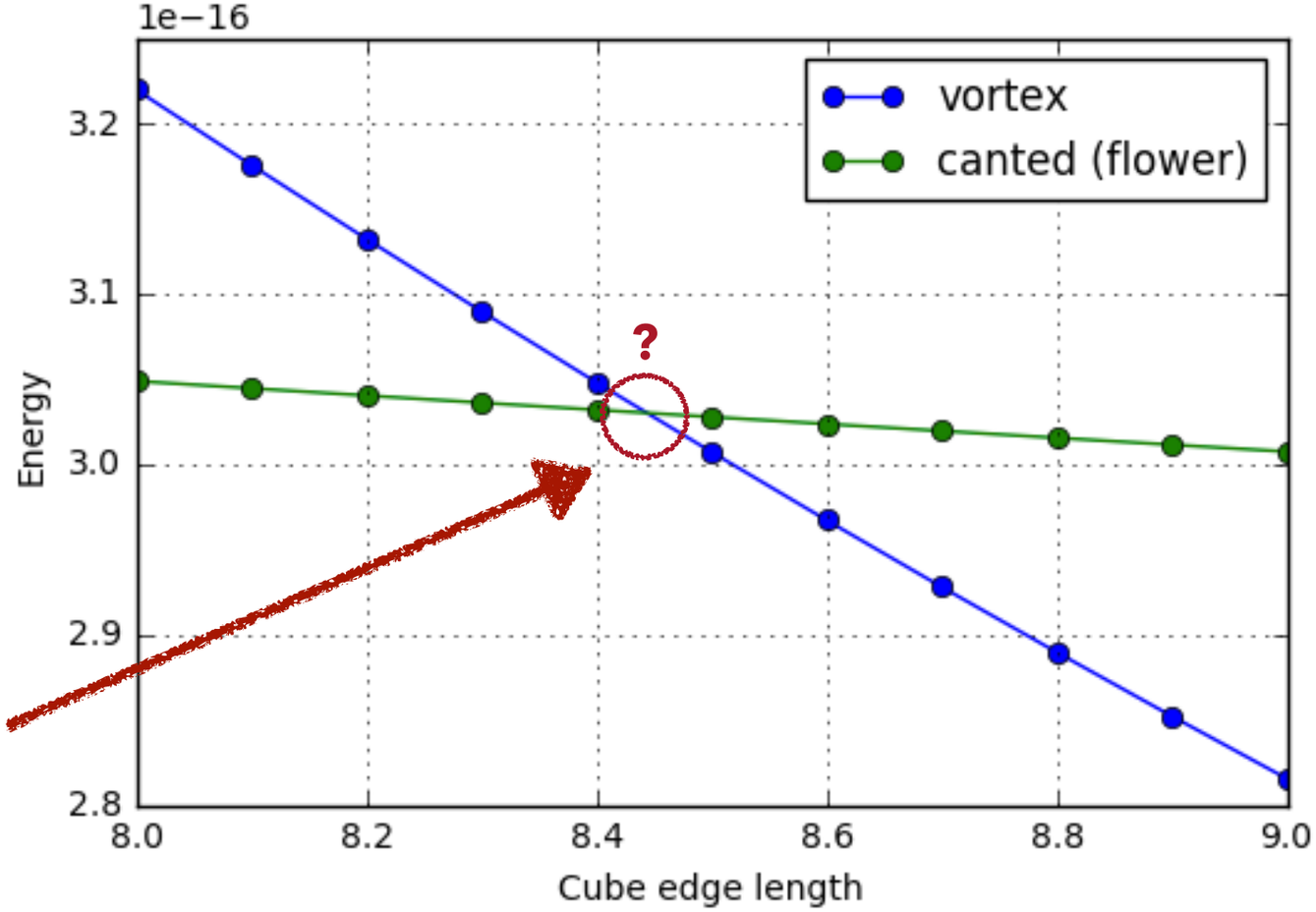
L	flower	vortex
8,0	?	3.23×10^{-16}
8,1	?	?
8,2	?	?
8,3	?	?
8,4	?	?
8,5	?	?
8,6	?	?
8,7	?	?
8,8	?	?
8,9	?	?
9,0	?	?



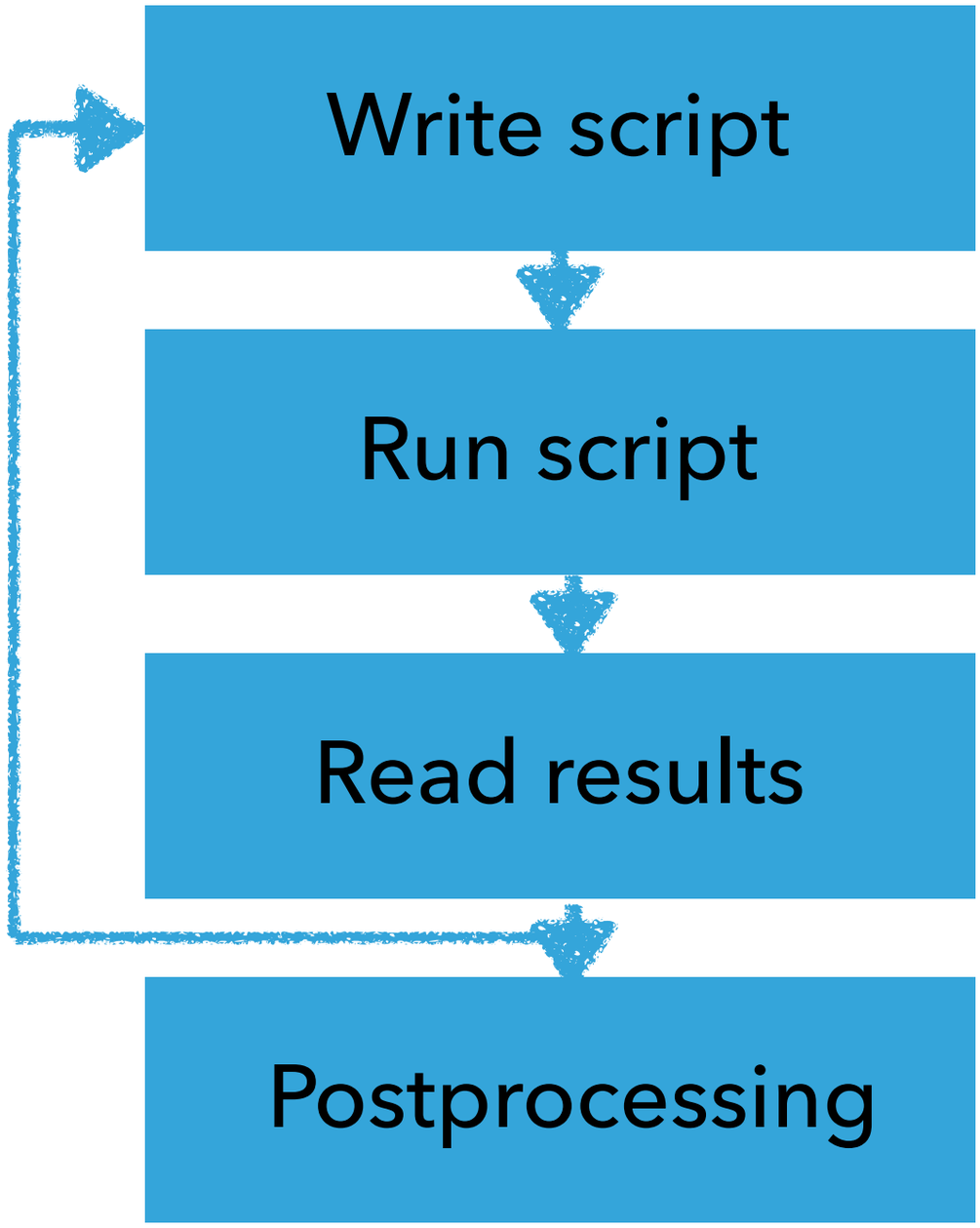
STEP 4: POSTPROCESSING

- Finally, we **plot the data and find the crossing**.
- We use separate plotting scripts or graphical user interface.

Find crossing
(usually “eye-norm”)



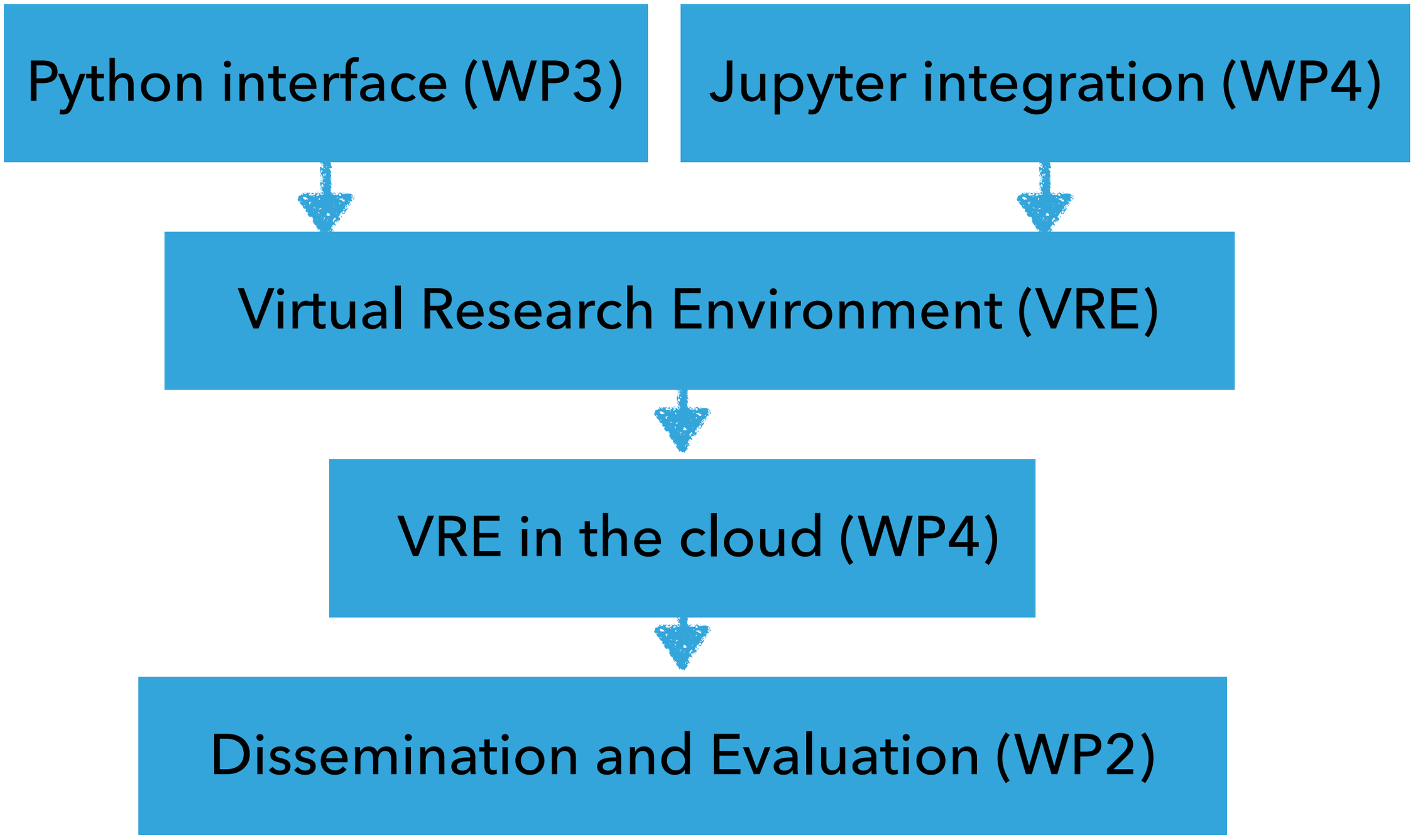
WORKFLOW



EFFECTIVE?

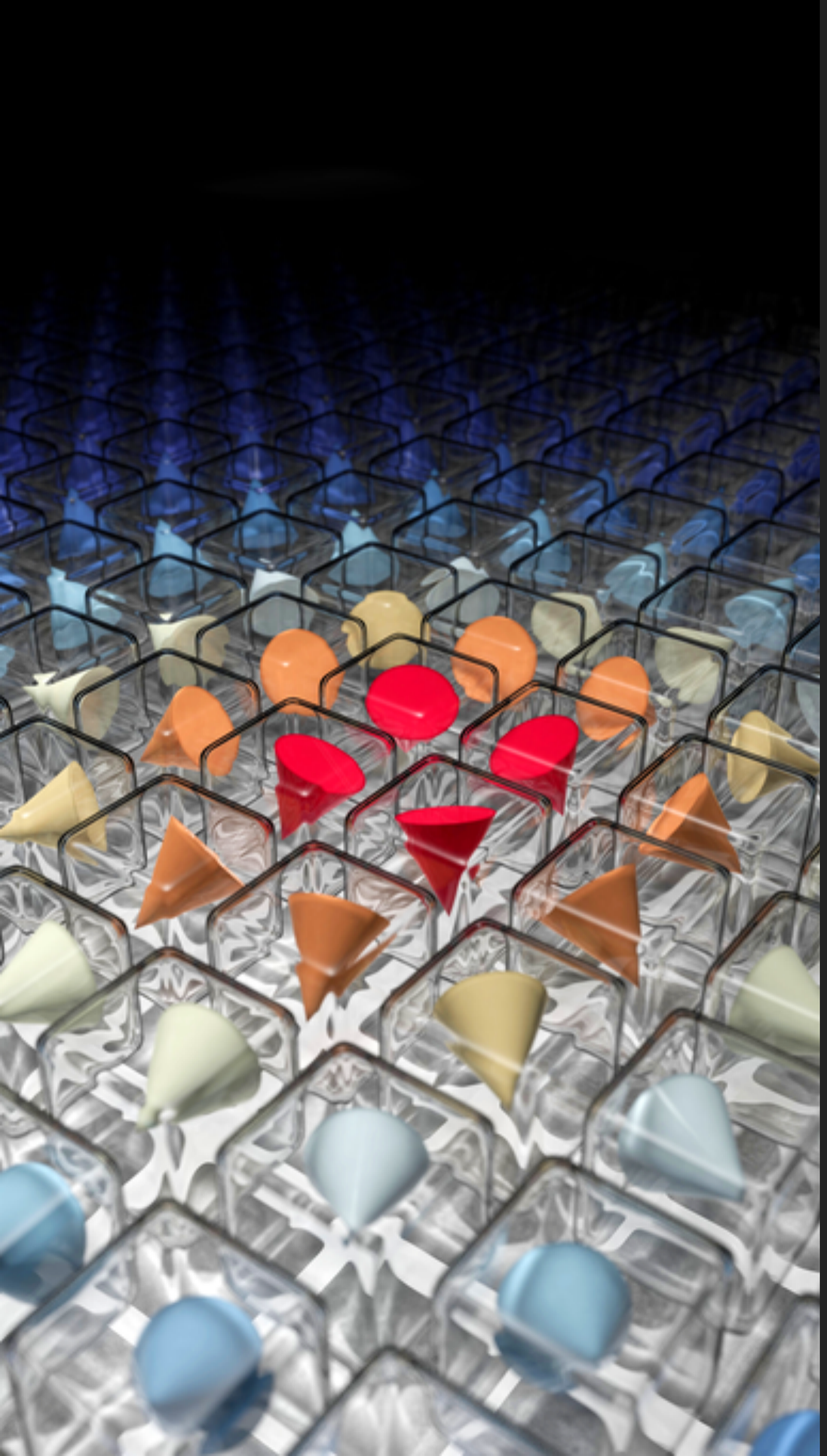
- **Time consuming**
- Many **manual steps**
- **Logging all steps** is very difficult
- **Difficult** to re-execute automatically
- **Reproducibility?**

OBJECTIVE: MICROMAGNETIC VIRTUAL RESEARCH ENVIRONMENT



COLLABORATION WITHIN OPENDREAMKIT

- **nbval** (Simula Research Laboratory)
  - Test Jupyter notebooks
  - Keep documentation up-to-date
- **k3d** (University of Silesia)
  - 3d visualisation inside Jupyter notebook
  - Interactive plotting
  - Coming soon…



PART 4

JOOMMF DEMO

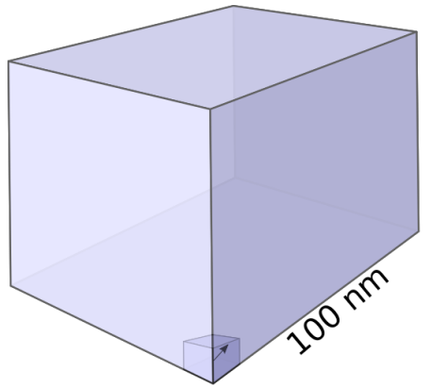
EASY TO INSTALL

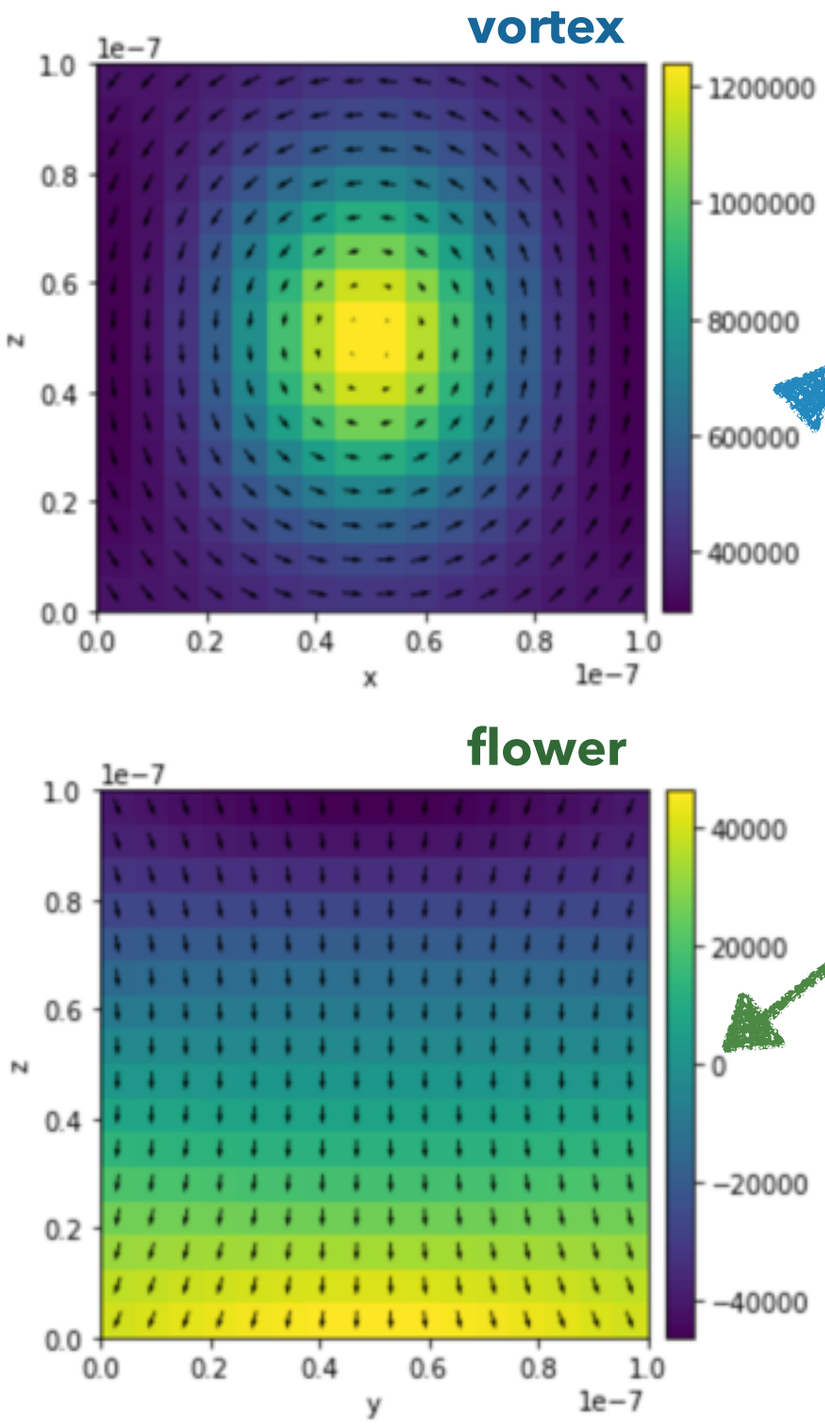
- **conda** (conda-forge)
  - `conda install –channel conda-forge joommf`
    - Provides OOMMF and all dependencies
- Supported on Windows, MacOS, Linux

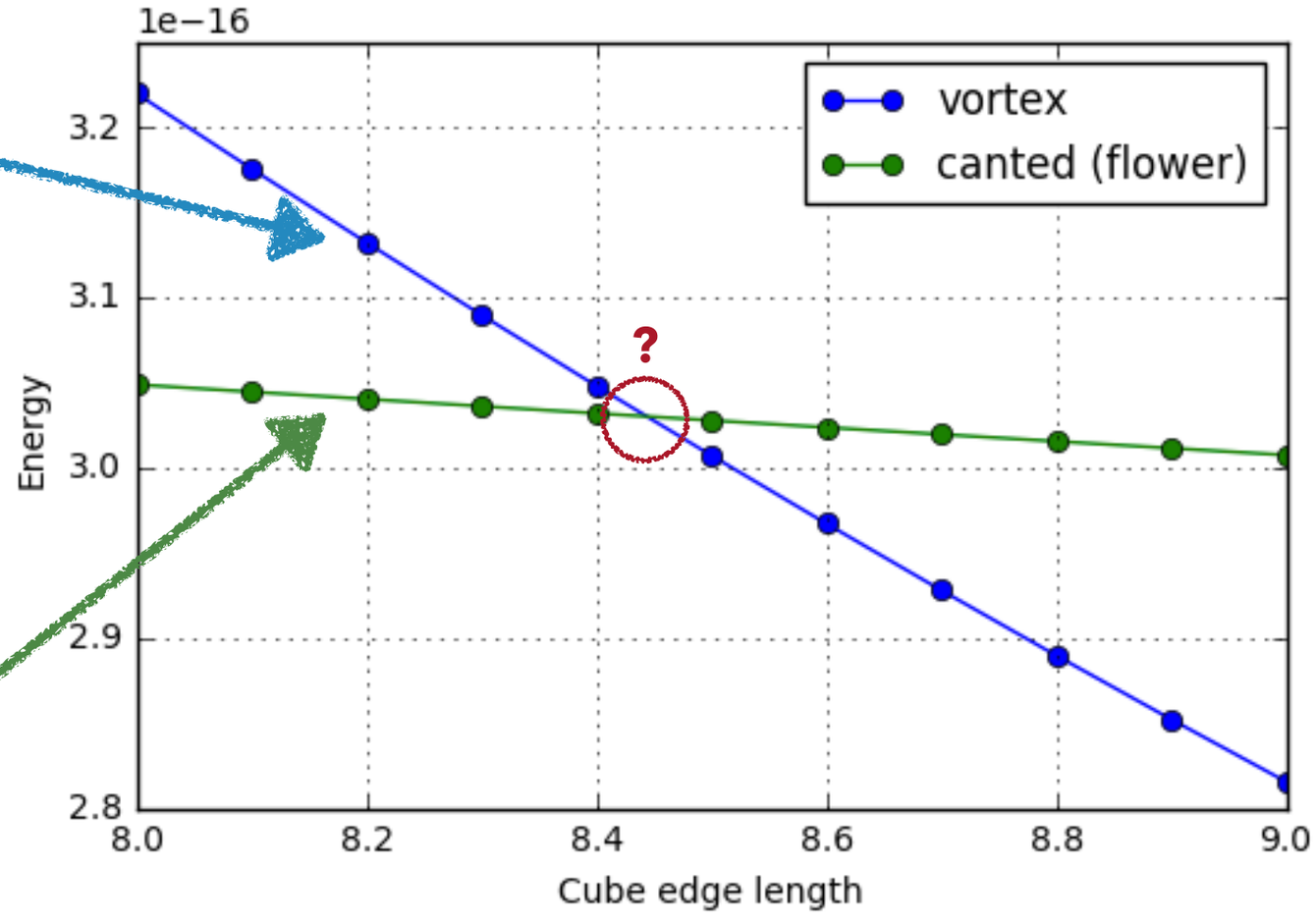
ALTERNATIVELY . . . JOOMMF IN THE CLOUD

- No need to install anything on the user's machine
  - All the user needs is a web browser and an internet connection
- No need to create an account
- No files are created on user's machine
- Anonymous free service
- Anybody in the world can run it
- Two possibilities:
  - https://tryjoommf.soton.ac.uk
  - Binder

DEMO: STANDARD PROBLEM 3



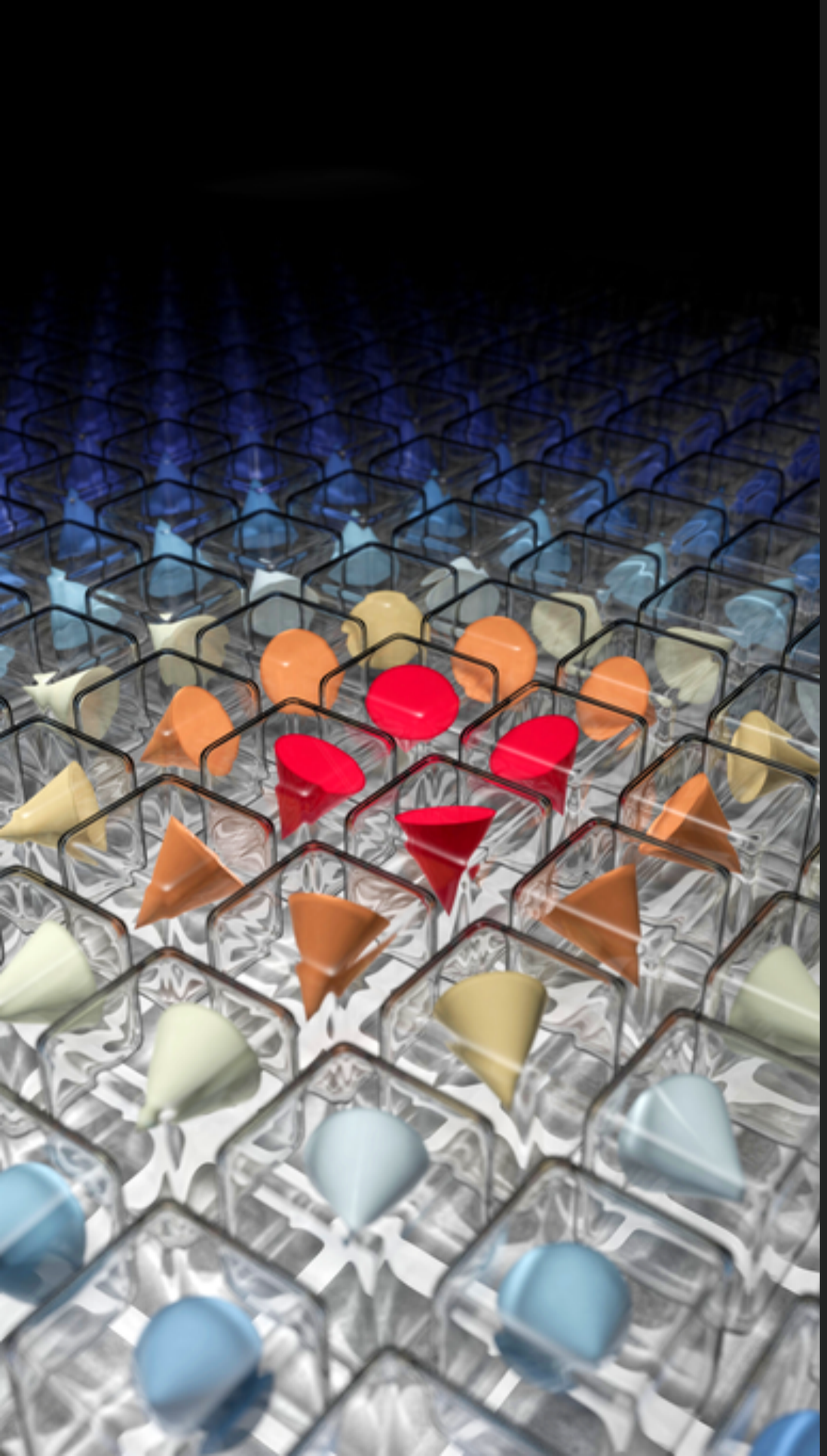




MICROMAGNETIC VIRTUAL RESEARCH ENVIRONMENT

- Executable document hosting text, equations, code, and results
- It replaces the complex workflow showed previously
- Easily executed again if any parameter should be changed

- Easy installation
- JOOMMF in the cloud:
  - https://tryjoommf.soton.ac.uk
  - Binder

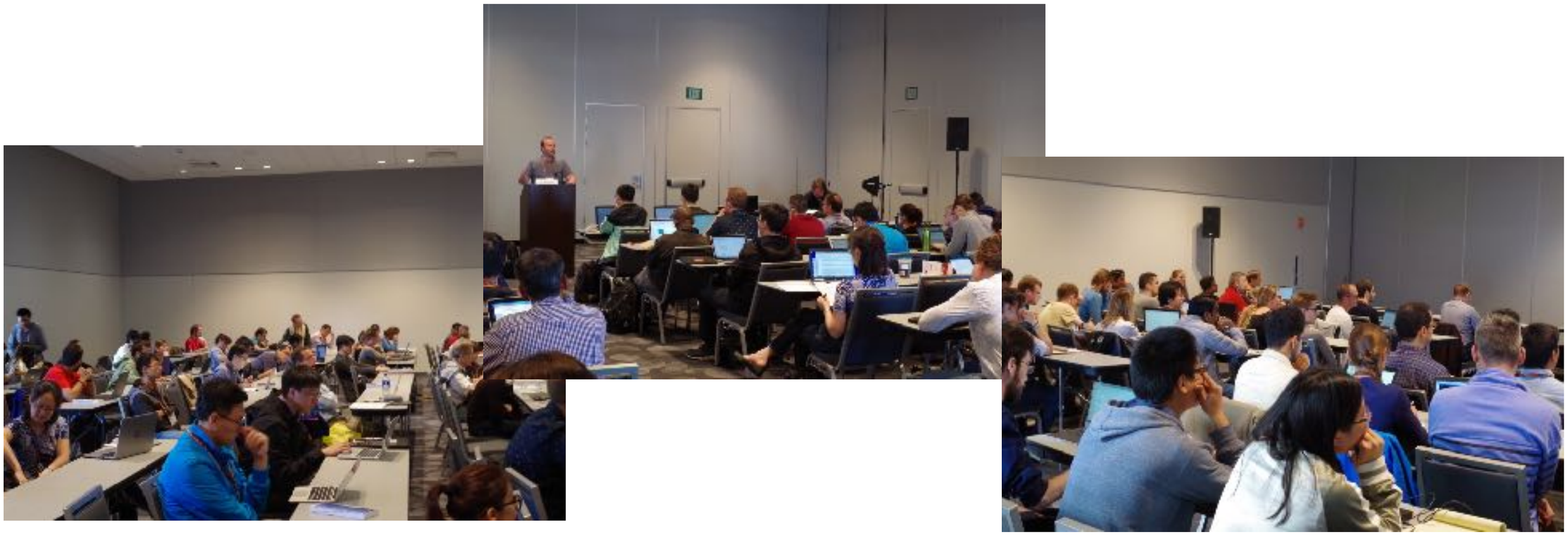


PART 5

DISSEMINATION

DISSEMINATION

- **Over 450 researchers from around 80 countries** have attended our workshops and tutorials at major international conferences
- **We invite feedback, feature requests** and provide support to researchers
- **Two university courses** are under development, based on JOOMMF



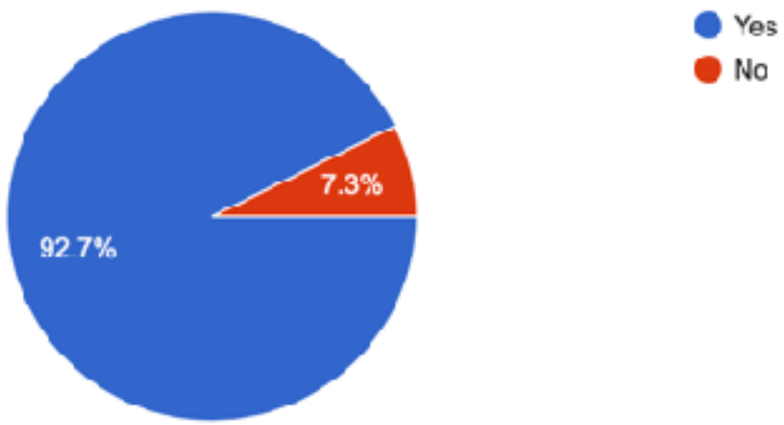
THREE TYPES OF DISSEMINATION ACTIVITIES

1. Walk-in sessions
2. Tutorials
3. Workshops

JOOMMF USERS

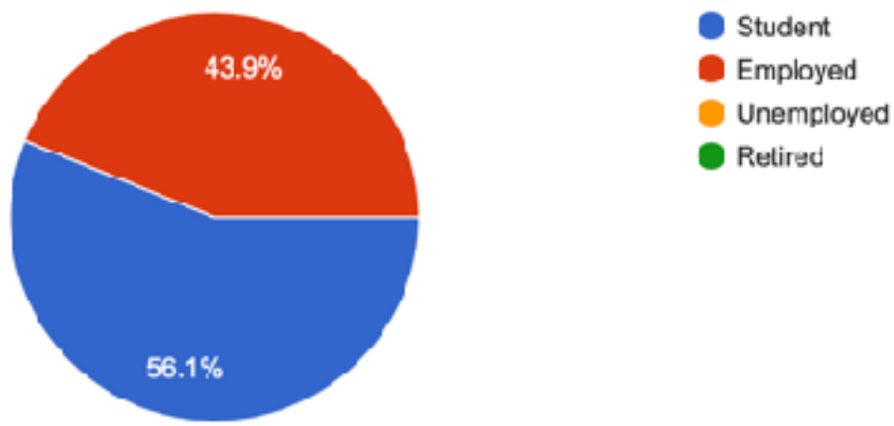
Are you active in the field of magnetism research?

41 responses



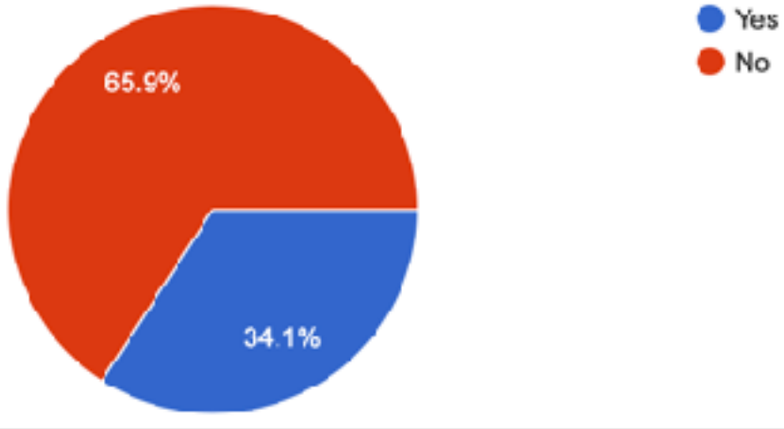
What is your profession?

41 responses



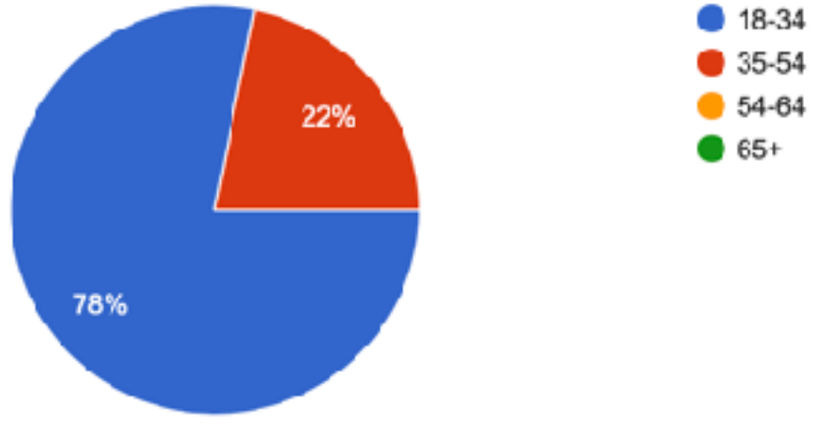
Have you ever performed a micromagnetic simulation?

41 responses



What is your age?

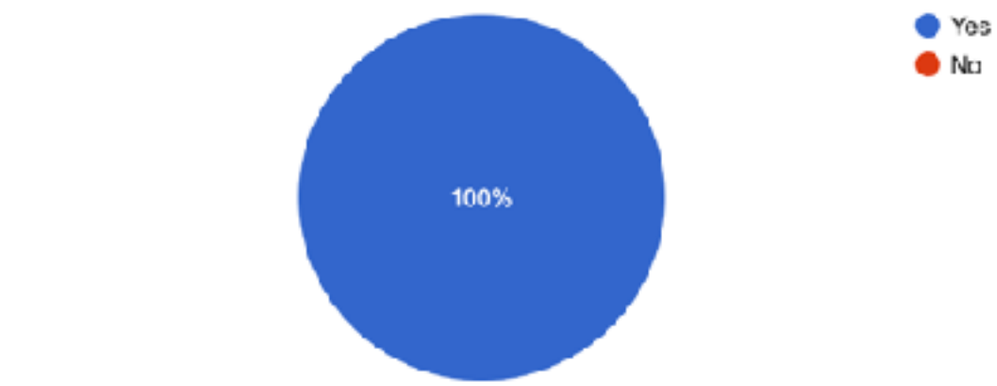
41 responses



PYTHON AND JUPYTER NOTEBOOK

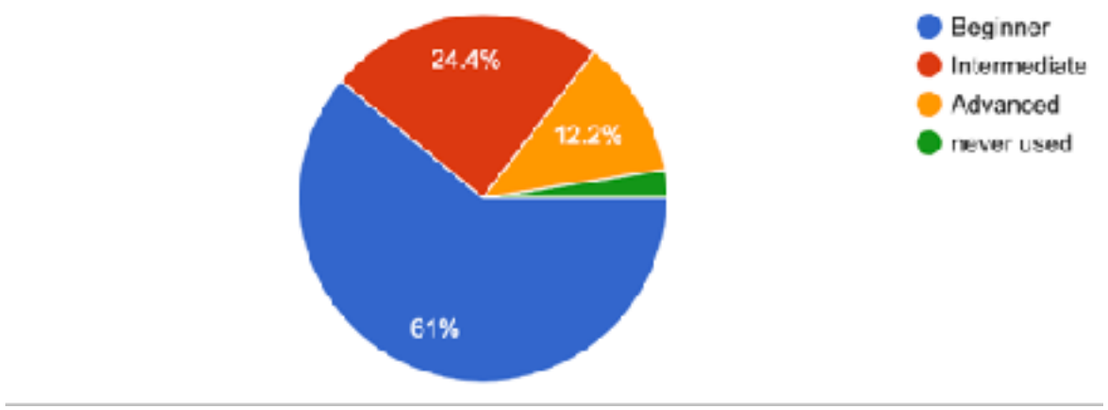
Have you heard about Python?

41 responses



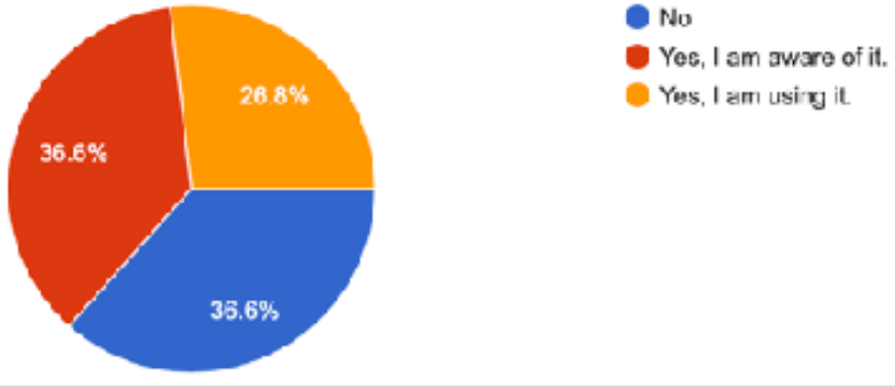
How confident are you with Python?

41 responses



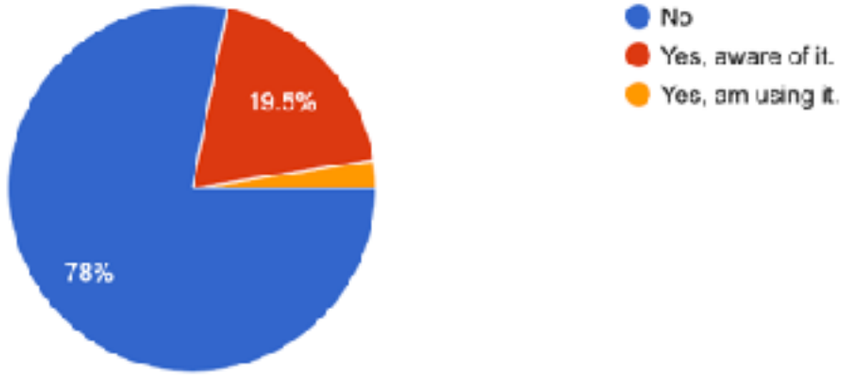
Have you heard of the Jupyter notebook? (It used to be called IPython Notebook.)

41 responses



Have you heard of "Docker"?

41 responses



WORKSHOP MATERIALS

- Introduction to **micromagnetics**
- Introduction to **JOOMMF**
- **Tutorials and exercises**
  - Include solved examples and exercises
  - Goal is to **reproduce results from a Nature Communications paper**.
- **Demo** on workshop materials during breaks

WHAT DO YOU LIKE IN JOOMMF?

I like the easy way to start up with simple problems
ease of use, excellent workshop
the combination of python and a well-established micromagnetic package enables very broad usabilities!!
user friendly
easy access, friendly surface
its clear and simple
what is amazing about it is that you can run and see everything in a single document
i have to try some other complicated geometries witch dinymics
Ease of use - interface
Very straightforward, and black box-style tools are nice to use
It seems to be quite easy to use
Very user-friendly and easy to use in comparison to writing/editing OOMMF MIF files
It is very user friendly and the simulations are nice
The possibility to document my work in the same file as the programming.
Seems straightforward and versatile to a large array of problems
easy for a beginner in simulations, easy to install
Python interface
Well presented
simple interface and very intuitive operations on Jupyter
I thought it was easy to use
conciseness, simplicity
Easy to use if you have basic programming knowledge.
Dynamics of spin polarization in FM/HM bilayer
The easy installation and notebook usage/output

WHAT DO YOU NOT LIKE IN JOOMMF?

Without any idea what to expect beforehand, I was a little disappointed to learn that it could not really simulate antiferromagnetism, unless it is at the cell level (though naively I don't know any systems with this physics). Also being a theorist, I wish for the option of setting up arbitrary or dimensionless units, for example, in terms of a single energy scale, or in terms of number of cells per length, rather than having to specify them.

But it still looks really interesting and useful for a zoo of other problems.

Not compatible with python 2.7 (so that i need to switch/reinstall anaconda before the workshop), but I guess python3 is the trend.

Not sure where I can get the documentation resources.

Not sure yet, I have to little experience with JOOMMF to answer this question

No yet.

nothing yet

Importing images for defining structure.
Complicated structures are harder to define compared to Mumax3.9.1.
There are no instructions regarding defining domain walls.
other than that, works great!

This workshop was excellent! Extremely informative! I am a beginner, a complete novice, and I found this workshop to be very helpful.

Nice workshop

Thanks for doing what you are doing!

It was an amazing workshop. Even though I am not familiar with python, the workshop thought me the basics of using the JOOMMF step by step. I wish there was more time! Thank you very much

unfortunatley the training during icm 2018 covers very basics

I found a weird bug in exercise 5 as pointed out to one of the people running the tutorial session

The lecture was nice

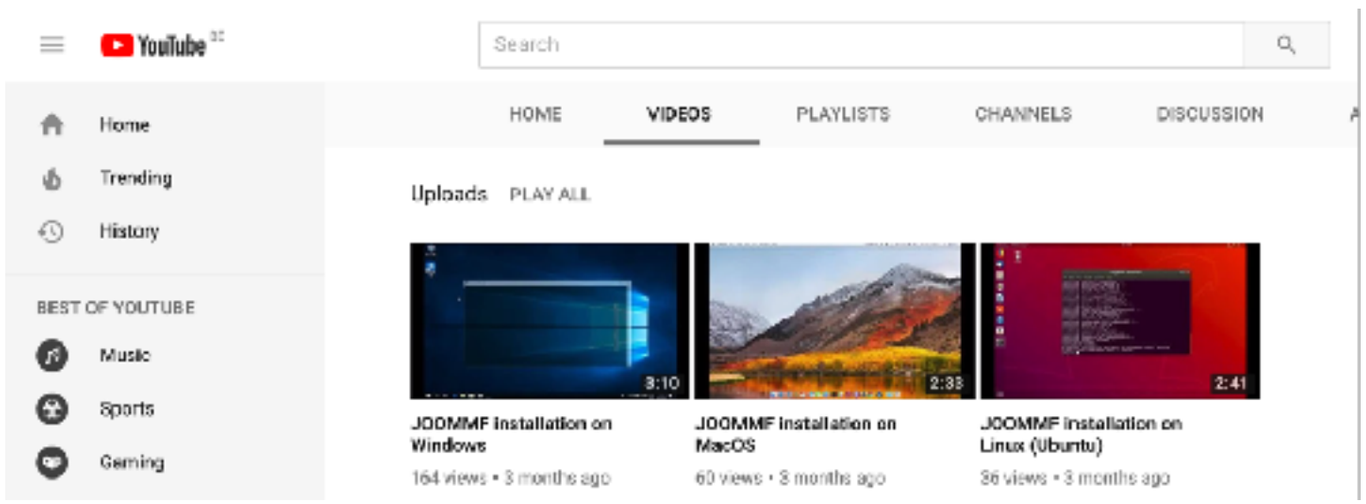
The workshop was really nice, the speed was perfect and I got a good overview of JOOMMF

I appreciated that it began at a basic level. Very nice.

Excellent speaker at the workshop; speed could have been a bit faster for me

PROBLEMS 1/3

- Registration
  - Not allowed to know the participants and cannot contact them before the workshop
  - We cannot give them installation instructions
- Usually the **first hour of the workshop** was wasted on installation
- Very **advanced steps required** for an average PC user (cloning repository, running commands in the command prompt,...)
- This problem was **now solved** with an **online JOOMMF VRE**
- We now have a **JOOMMF YouTube channel** with installation instructions.

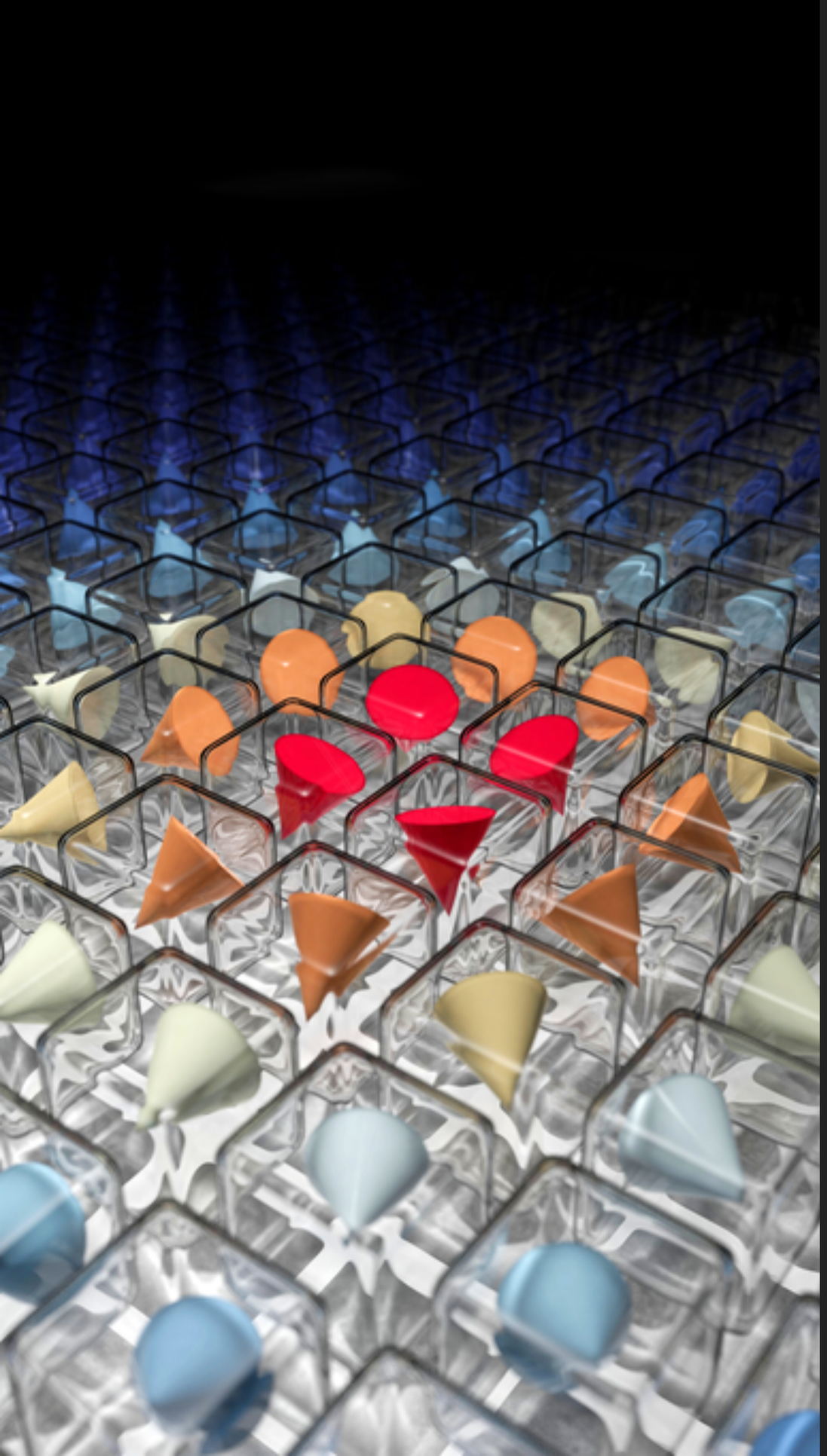


PROBLEMS 2/3

- Need to better understand our target audience: Physicists
  - not excited by implementation details (Python, Jupyter,…)
  - easier to learn plain OOMMF than JOOMMF prerequisites
  - some find Jupyter Notebook a strange environment
  - steep learning curve (Python, NumPy, SciPy, …)
  - some don’t see the benefit of a new way to do the same
- What can I do that I could not do so far? Nothing?
  - Why did you do it then?

PROBLEMS 3/3

- Academic integrity questions
  - Why are you using somebody else's tool
  - We emphasise several times that OOMMF is open-source
- JOOMMF in the cloud: Are you going to steal my work?



PART 5

SUMMARY

SUMMARY

- Ability to **drive micromagnetic simulations inside a Virtual Research Environment**
- **Integration with Jupyter Notebook**
  - **Rich media representation** of equations, meshes, fields
  - **Widgets** to explore data sets interactively in notebook
- Framework to include more micromagnetic computational solvers (for example mumax3, micromagnum, fidimag)
- **Use of the Python ecosystem** for computational and data science, including numpy, scipy, pandas, ...
- **Remote VRE execution** of simulation (TryJOOMMF and Binder)
- **Easier reproducibility**: Notebook contains complete simulation study

ACKNOWLEDGEMENTS

- Contributors: Ryan A. Pepper, Thomas Kluyver, and Hans Fangohr
- Financial support:
  - OpenDreamKit Horizon 2020, European Research Infrastructures project (#676541), http://opendreamkit.org,
  - EPSRC’s Skyrmion Programme Grant (EP/N032128/1),
  - EPSRC's Centre for Doctoral Training in Next Generation Computational Modelling, http://ngcm.soton.ac.uk (#EP/L015382/1), and
  - The Gordon and Betty Moore Foundation through Grant GBMF #4856, by the Alfred P. Sloan Foundation and by the Helmsley Trust.
  - This work has been supported by Polish MSHE grant no 6109/IA/621/2010/2011/2014/2016/2017