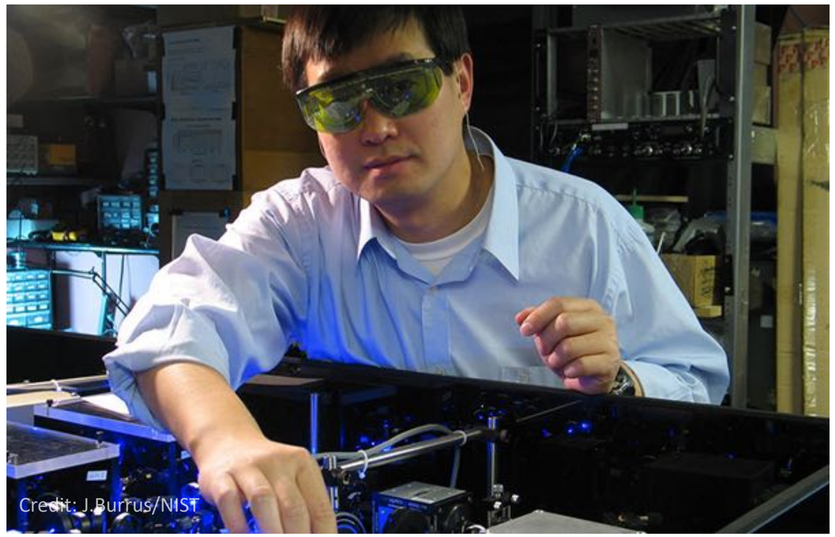
Snapshot of research software development practices in the NIST Materials Science and Engineering Division

Jon Guyer

Debra Audus, Jen Clark, Lucas Hale, Trevor Keller, Andrew Reid, Daniel Wheeler

NIST Mission

To promote U.S. innovation and industrial competitiveness by advancing measurement science, standards, and technology in ways that enhance economic security and improve our quality of life



NIST AT A GLANCE
Industry's National Laboratory

3,400+
FEDERAL EMPLOYEES

5
NOBEL PRIZES

2 CAMPUSES
GAITHERSBURG, MD [HQ]
BOULDER, CO

3,500+
ASSOCIATES

10
COLLABORATIVE INSTITUTES

400+
BUSINESSES USING NIST FACILITIES

16
NATL OFFICE FOR MANUFACTURING INSTITUTES

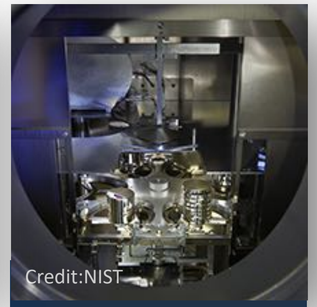
51
MANUFACTURING EXTENSION PARTNERSHIP CENTERS

U.S. BALDRIGE PERFORMANCE EXCELLENCE PROGRAM

NIST Laboratory Programs

Credit:NIST

Material Measurement Laboratory


Credit:NIST

Physical Measurement Laboratory

Credit:Shutterstock/ Dmitry Kalinovsky

Engineering Laboratory

Credit:Shutterstock

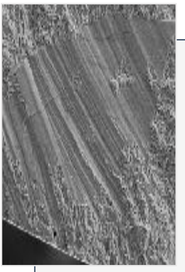
Information Technology Laboratory

Credit: Shutterstock/Italia Estro

Communication Technology Laboratory

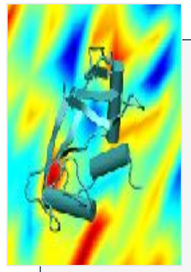
NIST Center for Neutron Research

MML Divisions

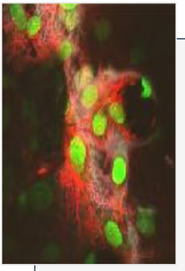


Applied Chemicals and Materials

Chemical Sciences

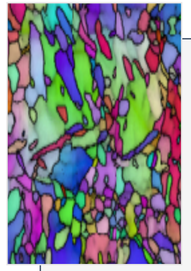


Biomolecular Measurement

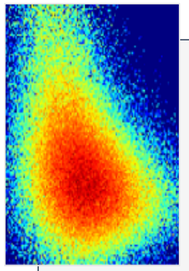


Biosystems and Biomaterials

Materials Measurement Science



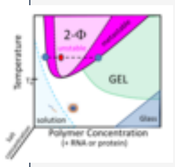
Materials Science and Engineering



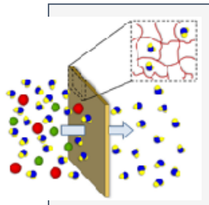
Office of Data and Informatics

Office of Reference Materials

MSED Groups & Centers

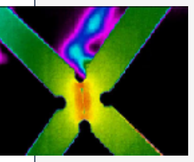


Polymers and Complex Fluids

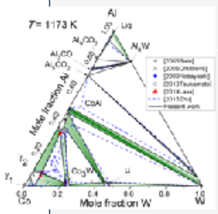


Functional Polymers

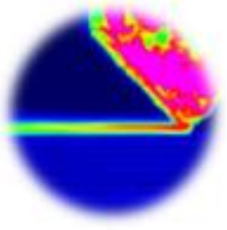
Functional Nanostructured Materials



Mechanical Performance

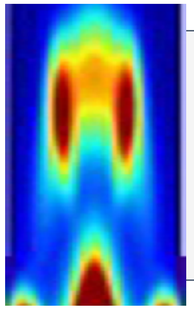


Thermodynamics and Kinetics



Polymers Processing

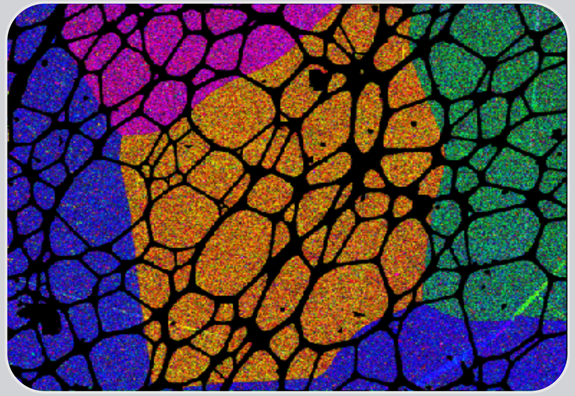
Center for Theoretical and Computational Materials Science



nSoft / nMat

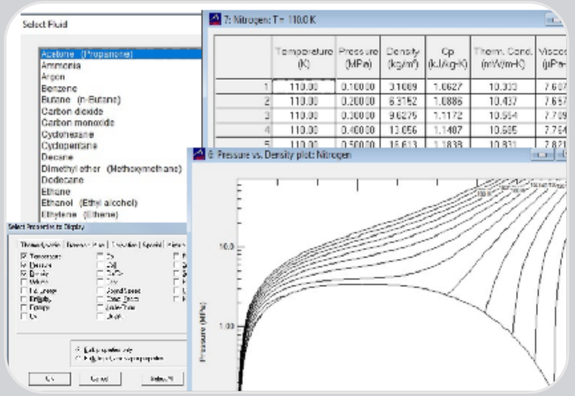
Applied measurement science

NIST MATERIAL MEASUREMENT LABORATORY

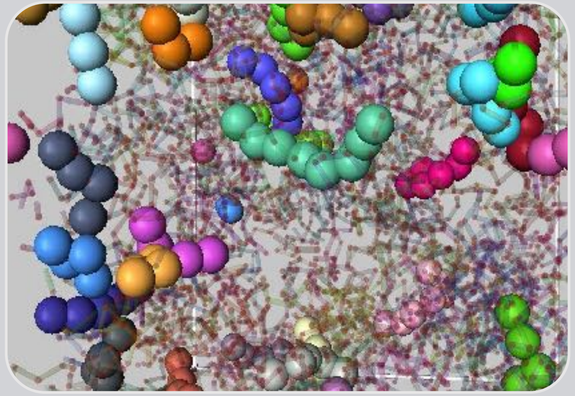


Characterize materials for the first time, or at new length scales

New instruments and methods

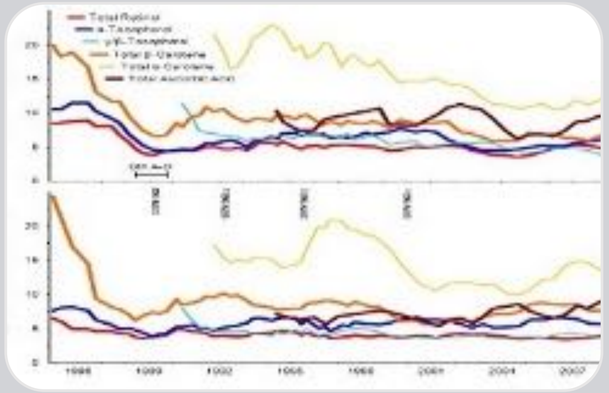


Validated data



Predictive models

Measurement services



Reference materials and data

- 1,200 Standard Reference Material products
- 100 Standard Reference Data products

User facilities

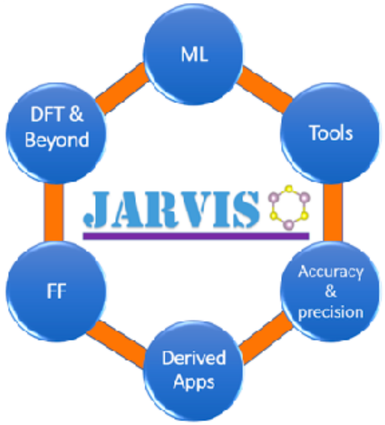
- National Synchrotron Light Source II at Brookhaven National Laboratory
- nSoft consortium at NIST Center for Neutron Research

Measurement quality assurance programs

- Cannabis
- Clinical measurements
- Dietary supplements
- Food nutrition and safety
- Marine environment

Software Products

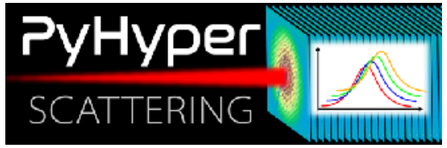
NIST MATERIAL MEASUREMENT LABORATORY

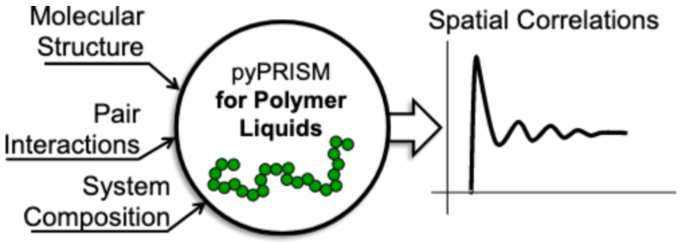


NIST AFL AUTONOMOUS FORMULATION LABORATORY

ZENO

p-RSoXR





WebFF

Survey

NIST MATERIAL MEASUREMENT LABORATORY

MSED Software Development Needs

Researchers in MSED write computer programs for tasks including control of instrument processes, automation of repetitive tasks, visualization and analysis of data, simulation of materials behavior, ...

These programs can be

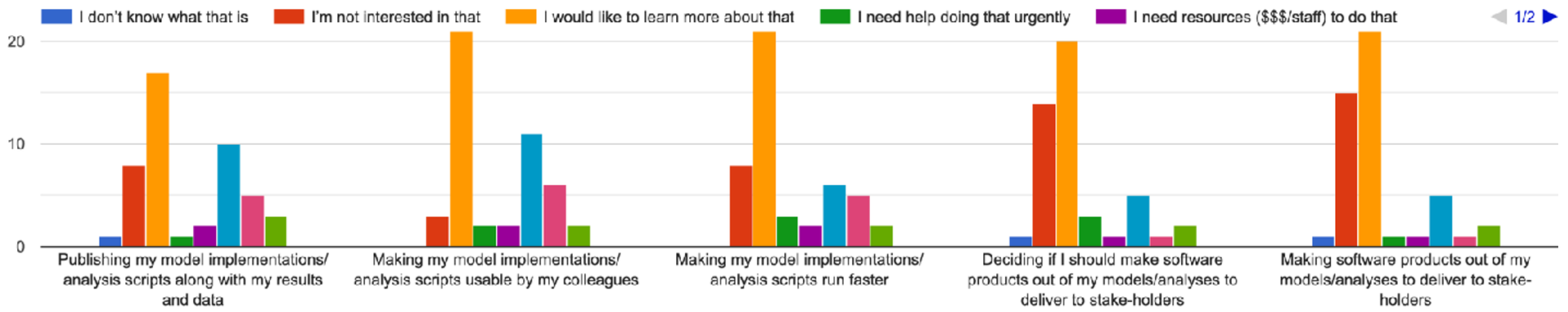
- of dubious utility even for the people who wrote them
- useful to their author(s), but impossible to share with others
- useful to a wider audience, but possibly in an embarrassing state
- regularly included in the course of research dissemination
- research products in their own right with substantial development effort to support them

The goal of this survey is to get a snapshot of where we are in MSED and to understand both where people need help as well as where they might be able to offer help to their colleagues.

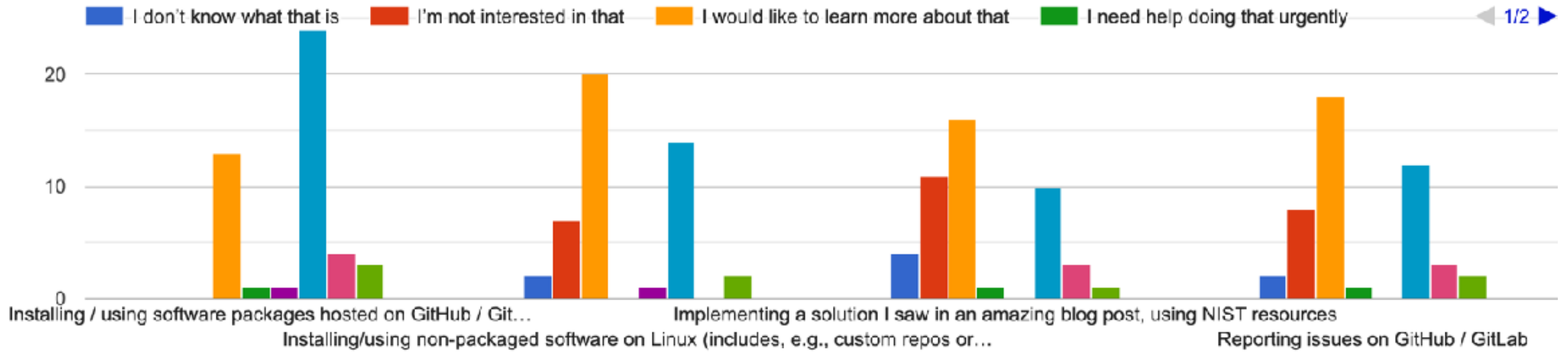
Spring 2023

Survey

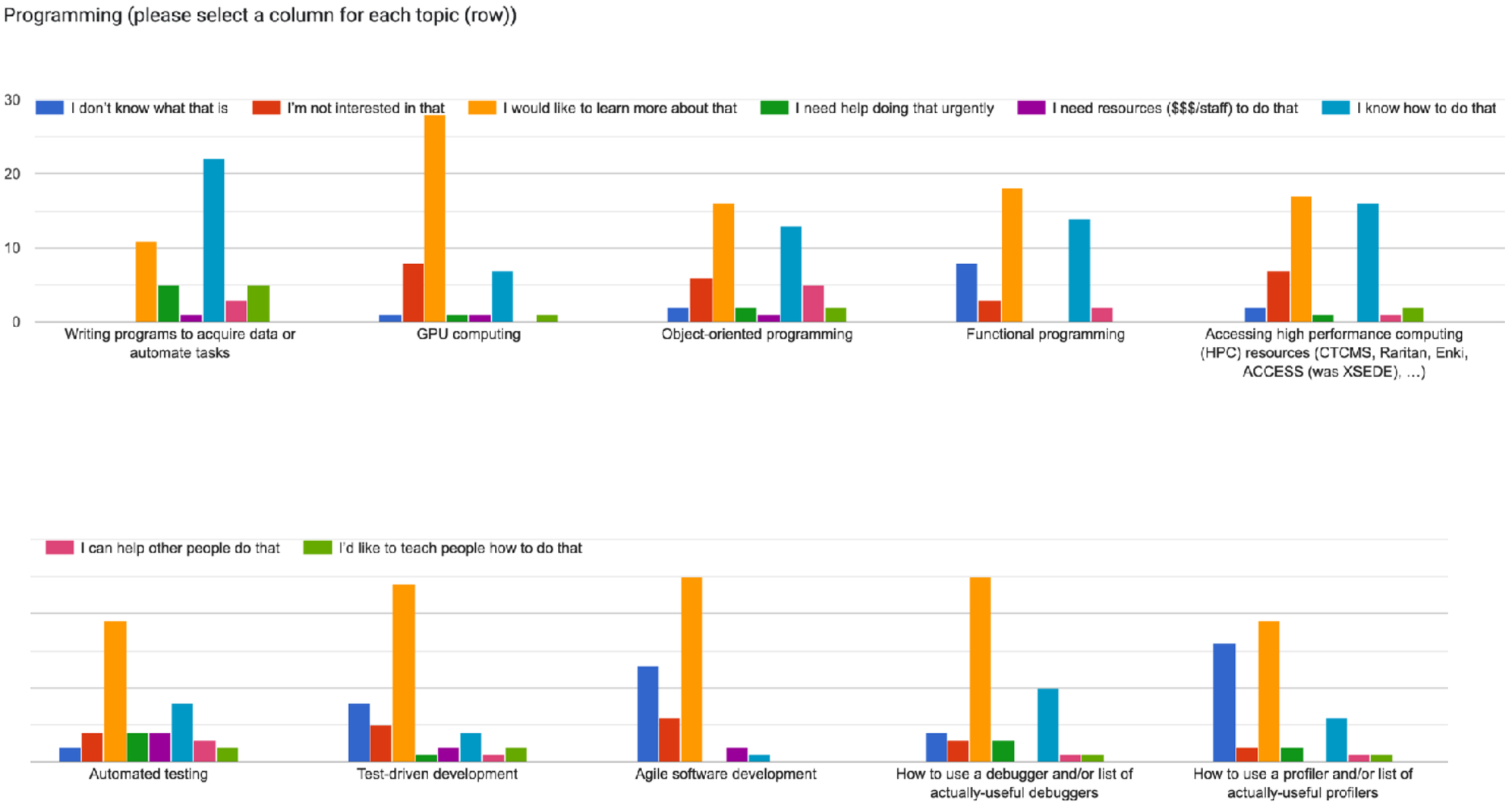
My Software (please select a column for each topic (row))



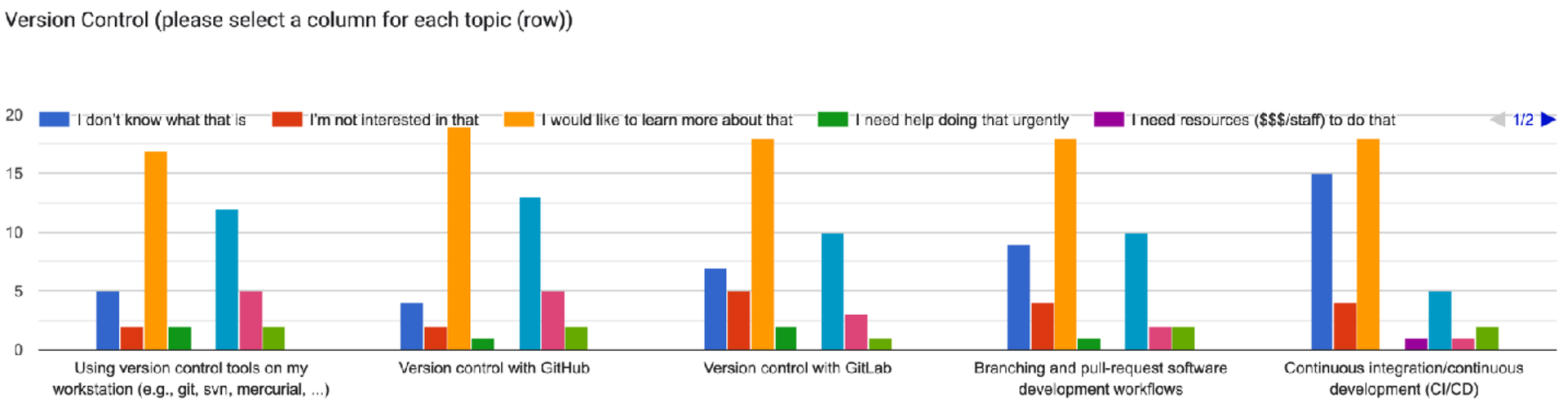
Other People's Software (please select a column for each topic (row))



Survey

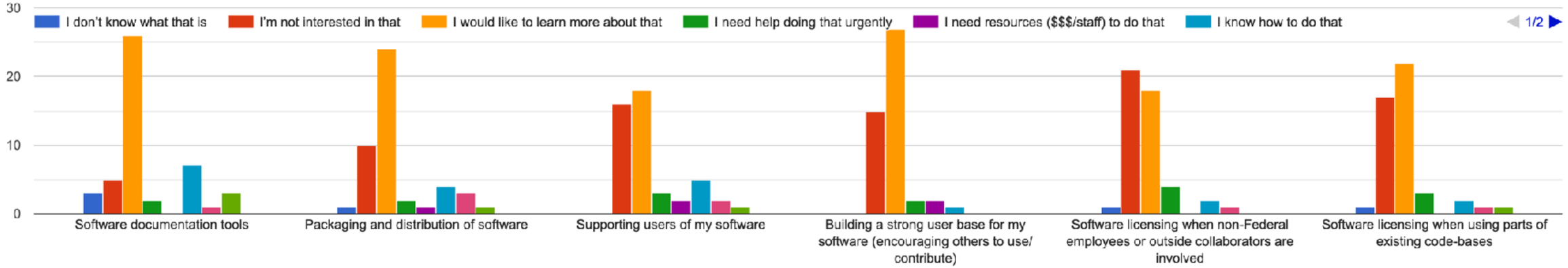

Programming (please select a column for each topic (row))
I don't know what that is
I'm not interested in that
I would like to learn more about that
I need help doing that urgently
I need resources ($$$/staff) to do that
I know how to do that
I can help other people do that
I'd like to teach people how to do that
30
20
10
0
Writing programs to acquire data or automate tasks
GPU computing
Object-oriented programming
Functional programming
Accessing high performance computing (HPC) resources (CTCMS, Raritan, Enki, ACCESS (was XSEDE), ...)
Automated testing
Test-driven development
Agile software development
How to use a debugger and/or list of actually-useful debuggers
How to use a profiler and/or list of actually-useful profilers

Survey

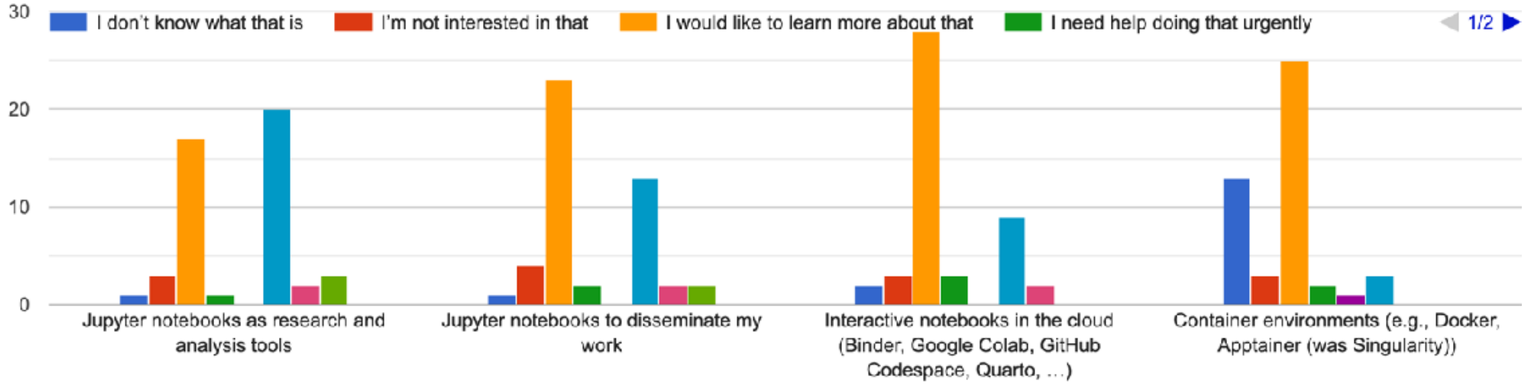

Version Control (please select a column for each topic (row))
I don't know what that is
I'm not interested in that
I would like to learn more about that
I need help doing that urgently
I need resources ($$$/staff) to do that
1/2
20
15
10
5
0
Using version control tools on my workstation (e.g., git, svn, mercurial, ...)
Version control with GitHub
Version control with GitLab
Branching and pull-request software development workflows
Continuous integration/continuous development (CI/CD)

Survey

Software Development (please select a column for each topic (row))



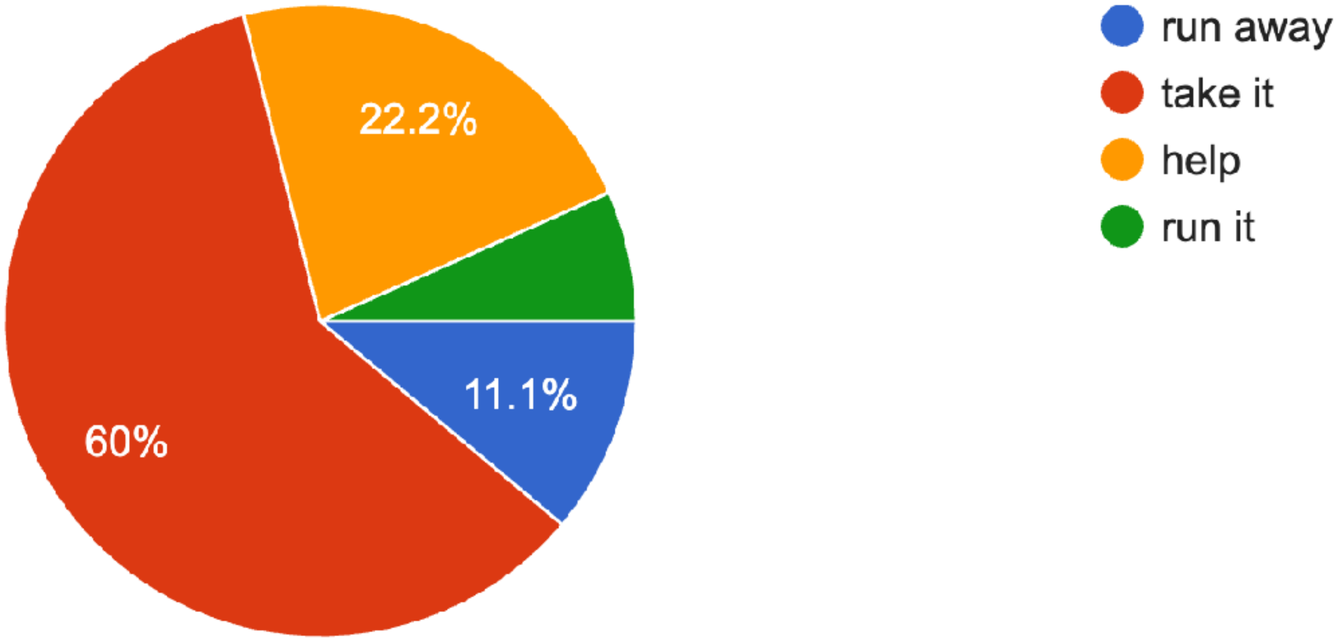
Reproducible Computing Tools (please select a column for each topic (row))



Survey

If there was a general / best practices coding bootcamp, I would

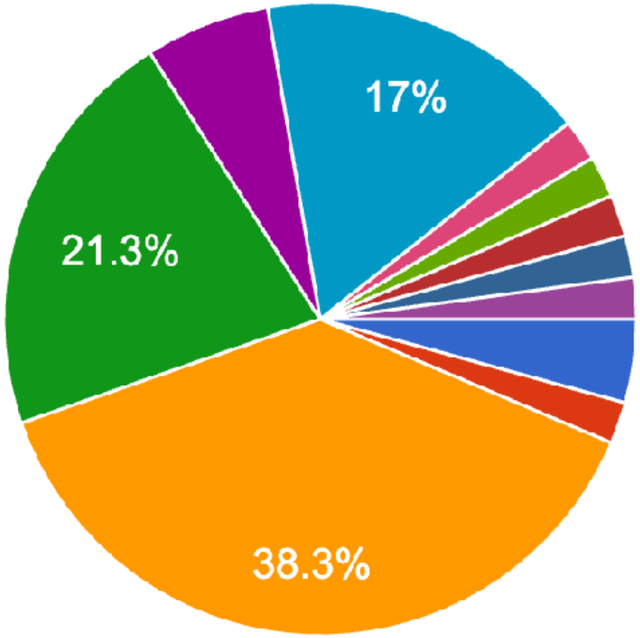
45 responses



Survey

When it comes to software, I identify as:

47 responses



- end-user (i.e. don't code and don't wa…
- new / novice coder (i.e. little/no experi…
- coder, new / novice package user (i.e.…
- coder & package user, new / novice c…
- package contributor, new / novice dev…
- developer
- Hacker, copy/paste from whatever I fin…
- developer of code for Molecular Dyna…

1/2

Interviews

Modeled on:
Data Practices Snapshot of the Materials Science and Engineering Division: Observations, Analysis, and Recommendations by Creuziger, Lau & Luecke
and ***CHIPs Project interview template*** by Lau, Reid & Keller

Pre-Interview Rhetorical Question

Consider your workflows and the research software that supports those workflows, including (but not limited to) spreadsheet formulæ, Fortran programs, Mathematica notebooks, Python scripts, Matlab libraries, Igor procedures, bash scripts, C++ libraries, glue code, Jupyter notebooks, LabVIEW graphical programs, PowerShell scripts, …

- Where are the research software programs used or written by your most recently departed postdoc or associate? Do you know how to run them?
- Consider a paper you wrote more than a year ago. How easily could you reproduce the analyses and figures from that paper? How easily could your stakeholders reproduce the analyses and figures from that paper?
- How and how often do you verify and validate the behavior of your research software? How do you communicate that verification and validation to the consumers of your research?

2024–2025

Interview questions

- Tell us what you're working on
- What are you excited about?
- What are you stymied by?
- What are your deliverables? E.g., insights, protocols, data, software, …
- How do your deliverables get to your stakeholders?
- If research software is an element of your deliverables, who makes strategic decisions about it? E.g., whether, when, and how to disseminate; staffing; how to support users; resources; legacy planning; …
- What's your workflow?
- How do you share your workflow with others?
- Where do you get your research software? (NIST site license, Open Source, stolen from grad school, write it yourself…)
- Do you know how to find existing tools (e.g., on JOSS, JORS, GitHub, GitLab, BitBucket, StackExchange, Slack, …)?
- In your ideal world, what would be the biggest enabler for your workflow?

Interview questions

- What interest do you have in:
  - Research Software Stand-down (e.g., Automation Stand-down, FAIR Data Stand-down)
  - Boot Camps? What topics interest you?
  - Sprints or Hackathons?
  - “Tried and true” processes, or codes, or _____ so you don’t have to reinvent things?
  - Somebody to just come and do _____ for you?

Interview highlights

- Lack of code fluency
- Continuity is catchup (needing to support things when authors move on)
- As experimentalist, have to be jack of every trade. Design experiment, take the data, analyze, … one researcher more hands-on, another is better at analysis, others are better coders, but need all
- Have to collaborate w/ large-scale MD or massively parallel DFT that aren’t on campus. But not cutting edge; boring to the external experts.
- Code works ATM, but if you look at it the wrong way, it crashes
- Documentation takes time from doing science
- Doing ML on a measurement that don’t have continuous access is challenging
- Not using latest and greatest AI model, but industry is using DoE models from the 1950s
- A lot of the data that’s acquired is from proprietary software
- Hard to know if vendor software has cut corners or if fitting functions are appropriate
- “Workflow forging” - smash tool to fit people or smash people to fit tool
- Coding with LLMs: both interest and skepticism

Interview highlights

- AFL needs the basics of a lot of things: networking, http, …
  - Onboarding is hard
  - Needs:
    - Documentation
    - Tutorials
    - Understanding the “cogs” of the software and the “cogs” of the robotics and the interface between them makes it particularly challenging
  - Brookhaven has 10x the staff for a comparable effort
- CTCMS
  - Lots of users don’t talk to us. Lots of weird cases but don’t ask.
  - Model where somebody reads a blog post and says “can we do that?” should work, but people don’t do this
- Technical debt
  - Research papers can be published and forgotten
  - Traceability to the SI (Standard Reference Materials®, Measurement Services, and Calibration Services) involves multi-decade commitments to stakeholders
  - Research software sits between these extremes
    - Tech support, security updates, does it still work?

Interview highlights

- Interviews were about software, not data, but…
  - Should this be in my OneDrive or GitLab or …?
  - What goes in data.gov?
  - Support for data management would be incredibly useful. Don’t need PhD in matsci; need BS in CSE.
  - Coordinating data across multiple instruments -> convert to .csv -> migrate -> aggregate
  - No NIST de facto standard LIMS
  - Can’t find data I took
  - Not everything is being recorded. Hard for somebody to pick up and reproduce.
  - Metadata trail not well captured
  - Video-rate data storage is a challenge
  - Need to crawl ResData and tell me what I did. How to curate backwards?

Interview highlights

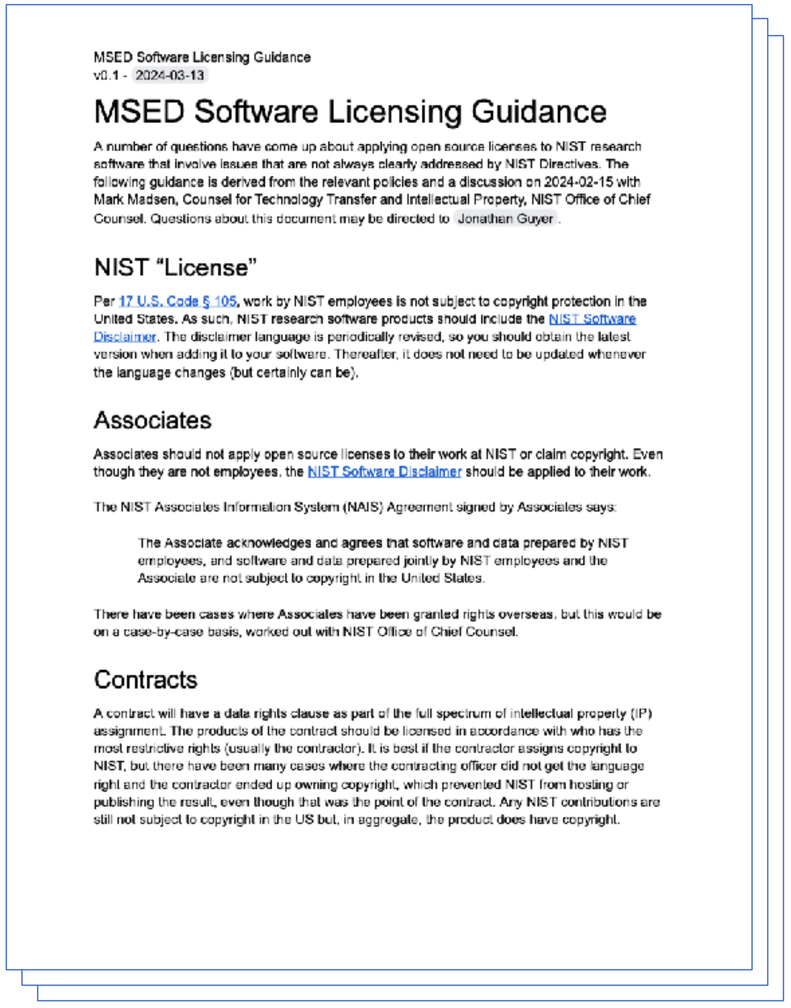
- Many want introductory training
  - Unaware of existing Carpentries training
  - or think Carpentries training will be too advanced
- Many asked for some form of Office Hours
  - How to run effectively?
  - Set time and place?
    - Historically, the same few people seem to come
    - Tend to be the ones who least need help
  - “Reverse Office Hours”?
  - Different topic each time?
- Boot camps: Depends on participants. Need to have a goal of what they want to achieve.
- Poster session or Show-and-Tell about how people are solving problems (rather than about their research)
- “Just come do it for me” not popular. Want to understand.

guyer@nist.gov

@guyer

Open Source Licensing

- Per 17 U.S. Code § 105, work by NIST employees is not subject to copyright protection in the United States
- Many questions from NIST staff as well as from the nascent Open Source Program Office at GWU
- Some topics not addressed by NIST Directives:
  - Associates?
  - Contracts?
  - Collaborators?
  - Journal of Open Source Software?
- Met with Counsel for Technology Transfer and Intellectual Property, NIST Office of Chief Counsel and drafted guidance



MSED Software Licensing Guidance
v0.1 - 2024-03-13

MSED Software Licensing Guidance

A number of questions have come up about applying open source licenses to NIST research software that involve issues that are not always clearly addressed by NIST Directives. The following guidance is derived from the relevant policies and a discussion on 2024-02-15 with Mark Madsen, Counsel for Technology Transfer and Intellectual Property, NIST Office of Chief Counsel. Questions about this document may be directed to Jonathan Guyer .

NIST "License"

Per 17 U.S. Code § 105, work by NIST employees is not subject to copyright protection in the United States. As such, NIST research software products should include the NIST Software Disclaimer. The disclaimer language is periodically revised, so you should obtain the latest version when adding it to your software. Thereafter, it does not need to be updated whenever the language changes (but certainly can be).

Associates

Associates should not apply open source licenses to their work at NIST or claim copyright. Even though they are not employees, the NIST Software Disclaimer should be applied to their work.

The NIST Associates Information System (NAIS) Agreement signed by Associates says:

> The Associate acknowledges and agrees that software and data prepared by NIST employees, and software and data prepared jointly by NIST employees and the Associate are not subject to copyright in the United States.

There have been cases where Associates have been granted rights overseas, but this would be on a case-by-case basis, worked out with NIST Office of Chief Counsel.

Contracts

A contract will have a data rights clause as part of the full spectrum of intellectual property (IP) assignment. The products of the contract should be licensed in accordance with who has the most restrictive rights (usually the contractor). It is best if the contractor assigns copyright to NIST, but there have been many cases where the contracting officer did not get the language right and the contractor ended up owning copyright, which prevented NIST from hosting or publishing the result, even though that was the point of the contract. Any NIST contributions are still not subject to copyright in the US but, in aggregate, the product does have copyright.

Open Source Licensing

tl;dr

NIST Software Disclaimer stipulates “The software developed by NIST employees is not subject to copyright protection within the United States.”

Associates? Covered by NAIS Agreement.
Associate agrees not to claim copyright.

Contracts? Covered by data rights clause of contract.
Can assign copyright to NIST, but often not done right.

External collaborators? OK for them to retain copyright under BSD, MIT, GPL, etc., alongside NIST Disclaimer

Publishing in JOSS? Needs to be handled case-by-case.
Not OK to assign copyright to self or to “Secretary of Commerce”.