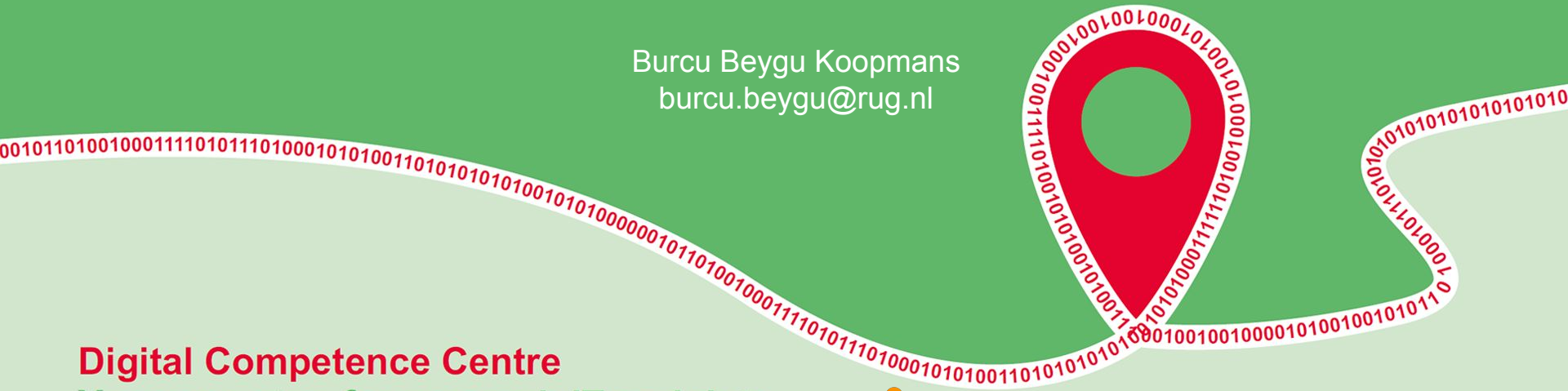
university of groningen

Financing Sustainable Research Software

TDCC-NES Bottleneck Project

Burcu Beygu Koopmans
burcu.beygu@rug.nl



Digital Competence Centre

Your one-stop for research IT and data

TDCC-NES
Natural & Engineering Sciences

university of groningen

TU Delft

CWI

Project Result

Dutch Thematic Digital Competence Centre for the Natural & Engineering Sciences (TDCC-NES)

Published April 14, 2025 | Version v1

Financing Sustainable Research Software

Beygu Koopmans, Burcu (Project leader)[1]

Contributors

Project leader: Beygu Koopmans, Burcu[1]

Project members: Hut, Rolf[2]; Williams, O.R.[1]

Related person: Vinju, Jurgen[3, 4, 5]

edu.nl/v6uvc

Motivation

- Developing research software is a fundamental aspect of research in the NES domain. Such software ranges from tools developed during PhD projects, to infrastructure services that support novel data processing pipelines, to software used in experimental setups for instrument control and data collection.

- These tools often play a critical role in advancing scientific discovery and can have long-term impact well beyond the lifespan of the original project.

- Despite their importance, however, research software frequently faces sustainability challenges, especially once project funding ends.

Motivation

Sustaining research software requires continuous human effort and dedicated financial resources

Goal

The project goal is to investigate and discover:

- Financial awareness of the long-term maintenance and sustainability
- Bottlenecks in the distribution of financial resources within national research institutions.
- The challenges which are faced with maintaining and disseminating software.
- Who develops and maintains research software?
- How developing/maintaining research software is included in career recognition
- Cost-benefit and risk analysis focused on human resources.

Methodology

- It is essential that the information comes directly from actual people who develop, maintain and/or manage scientific projects that involve software development.
- A first national survey and interviews with the Dutch NES community whose research involve developing and/maintaining research software.
- Survey:
  - Reach out the NES RS Community via TDCC-NES, Dutch DCC, Open Science, eScience Center, SURF and personal network
- Interview
  - Reach out a handful of director, manager and research group leader of Dutch NES Community

Data Collection

- The data was collected through a structured online survey designed in *Qualtrics* and distributed among researchers and professionals involved in research software within the NES domain of the Dutch research institutions as defined in the TDCC-NES roadmap.

- The survey remained open for responses between May 1st, 2024, and August 31st, 2024.

- A total of 197 responses were received.

Thematic Grouping of Survey Questions

Based on the intent and the context, 27 survey questions are grouped into 6 categories

Question Type
Participant profiles & roles
Context & use of the research software
Project & funding awareness
Budget allocation & financial sustainability
Human resources & maintenance responsibilities
Recognition

Thematic Grouping of Survey Questions

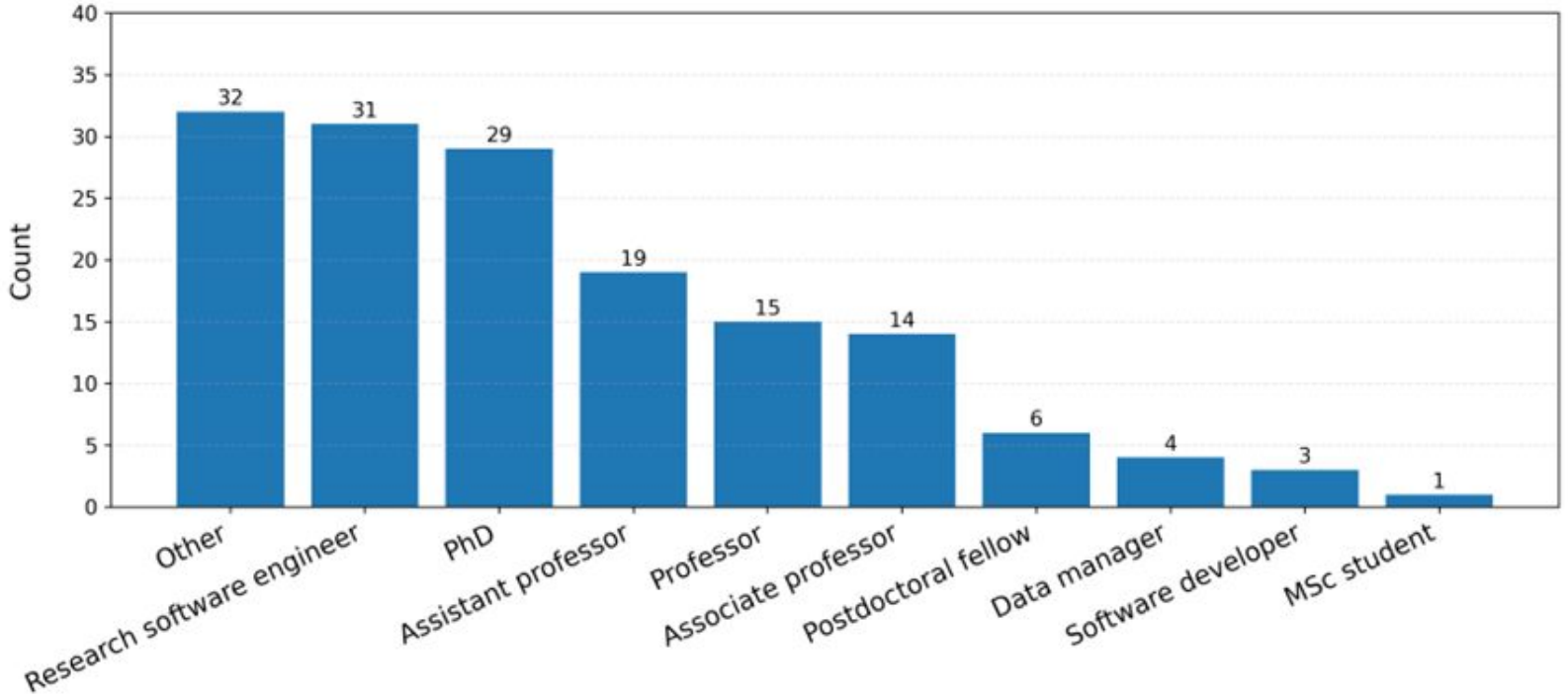
Based on the intent and the context, 27 survey questions are grouped into 6 categories

Question Type
Participant profiles & roles
Context & use of the research software
Project & funding awareness
Budget allocation & financial sustainability
Human resources & maintenance responsibilities
Recognition

TDCC-NES
Natural & Engineering Sciences

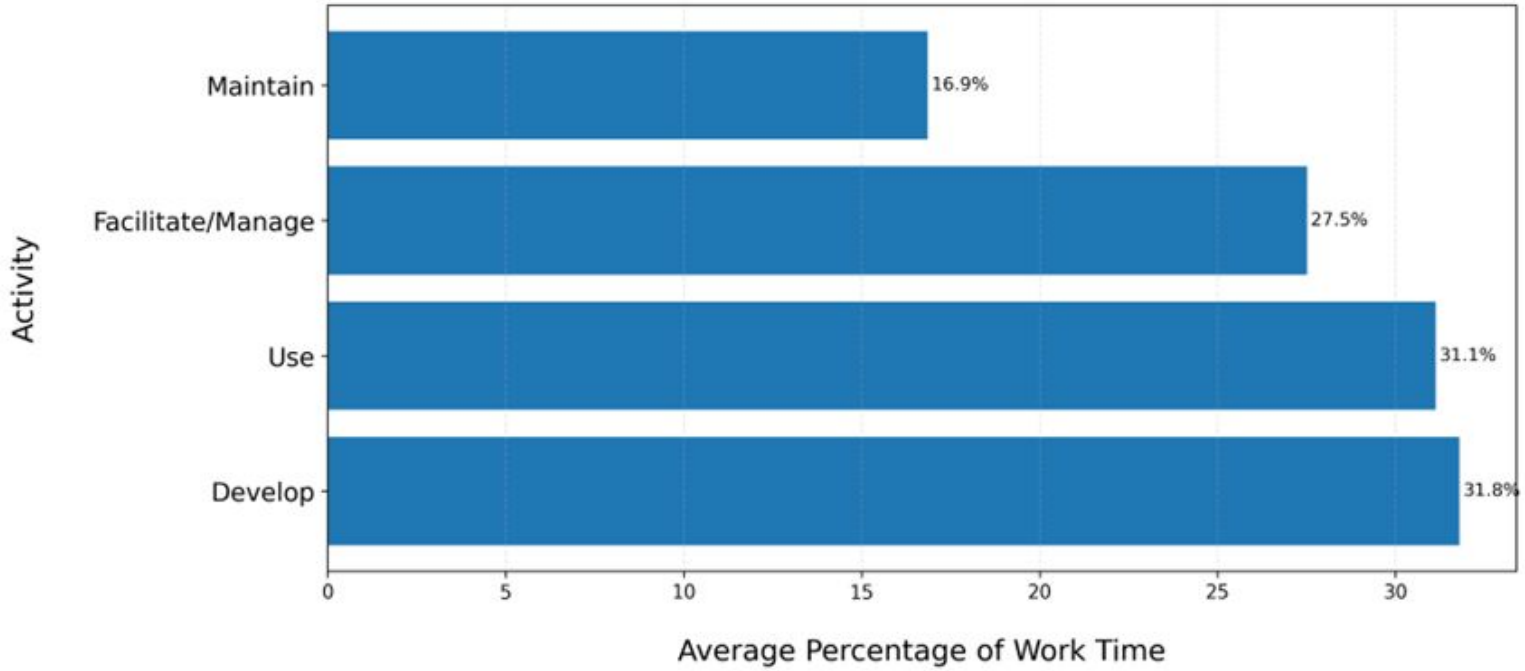
Participant profiles and roles

Participant profiles and roles - What is the job title



TDCC-NES
Natural & Engineering Sciences

Participant profiles and roles - How much time is spent on the following activities



TDCC-NES
Natural & Engineering Sciences

Who are the respondents?

Participant profiles and roles

- The majority of participants are affiliated with universities (%60) and represent a wide range of research fields.
- They often fulfill multiple roles—developing, using, maintaining, and managing research software simultaneously.
- While development and use occupy the largest share of work time, maintenance activities consistently receive the least attention, both in terms of time and formal role definition.

Context & use of research software

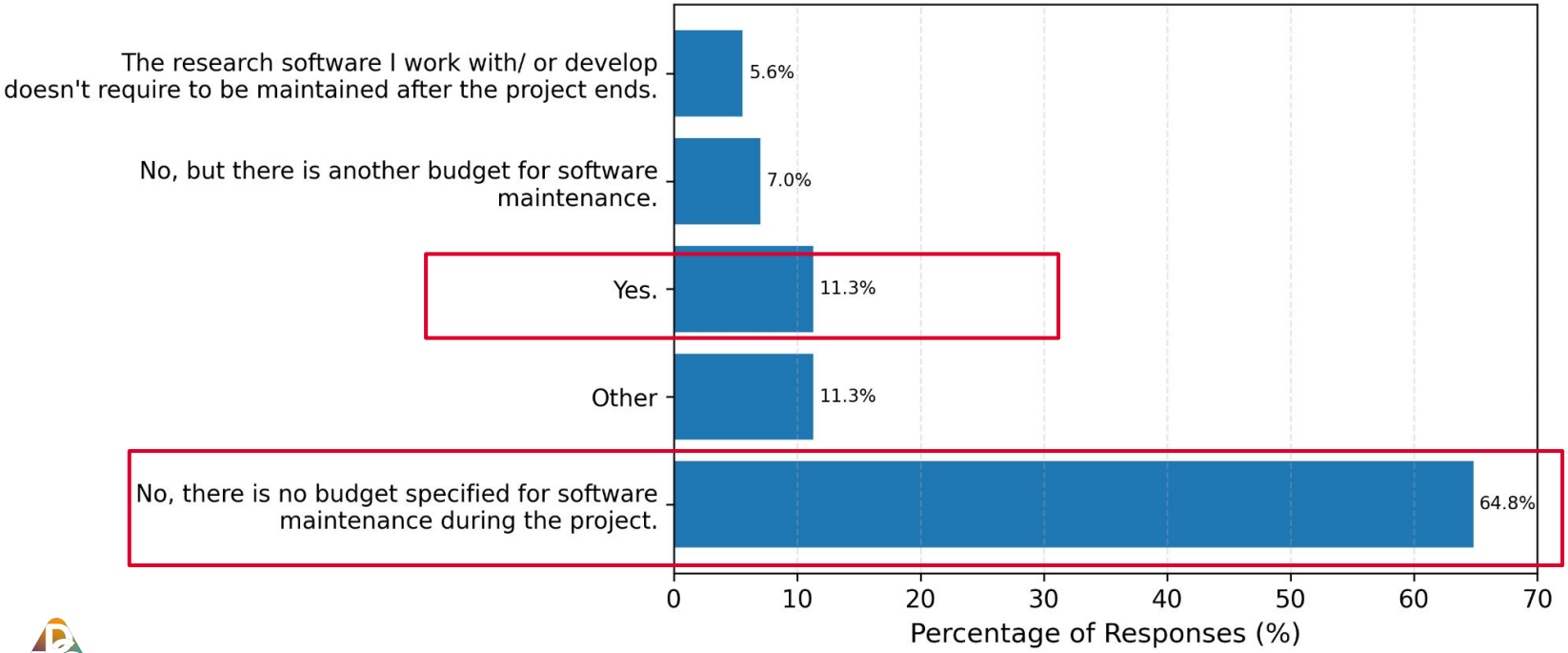
For whom the software is developed?

Context & use of the research software

- Most research software is developed with broader use in mind—by a research group or scientific community.
- It also covers multiple types, from specific tools to core infrastructure. This shows that software developed in NES is not purely individual, but part of collective research output.
- Astronomy & Astrophysics contributed significantly across all software types—most notably in Specific Methods and Community Contributions. This suggests that researchers in this field produce both specialized tools for domain-specific tasks and reusable software for broader community use.

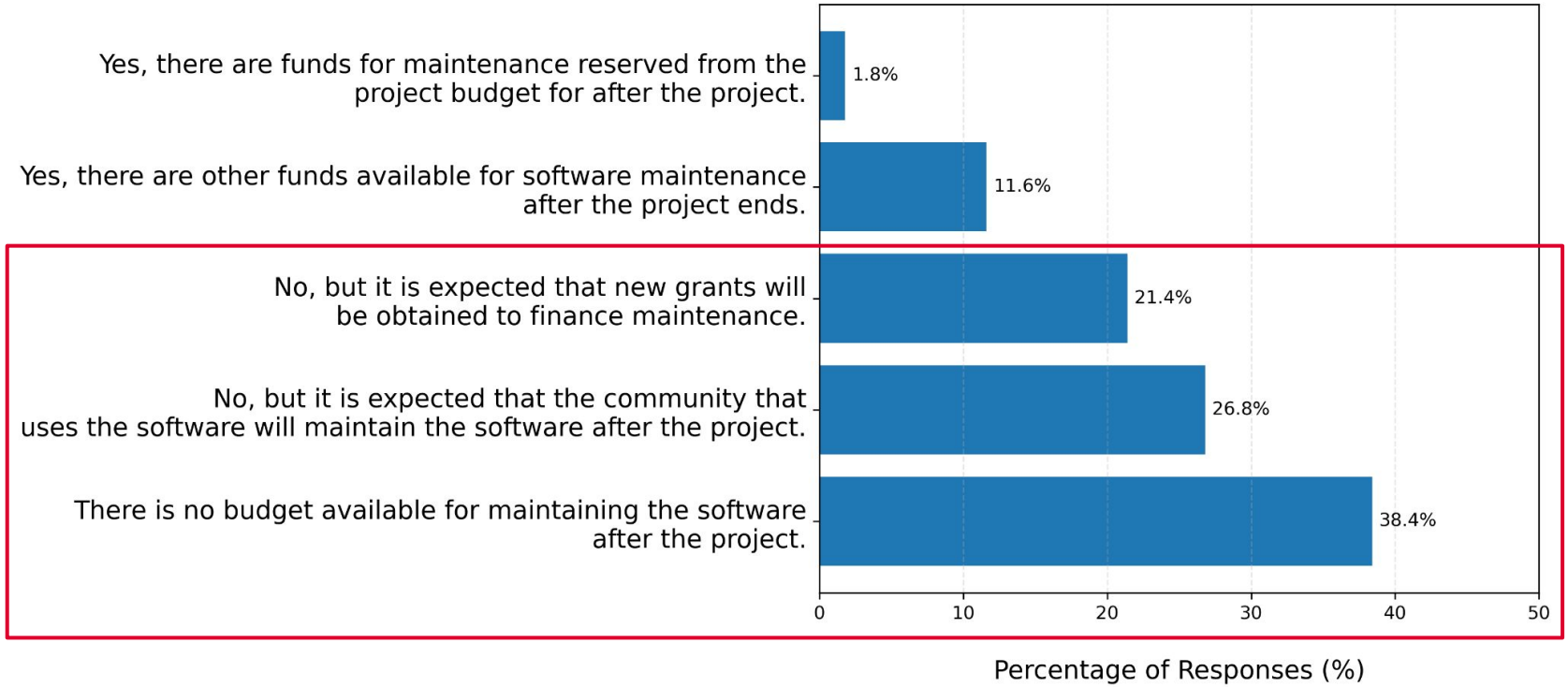
Budget allocations & financial sustainibility

Budget allocation & financial sustainability - Budget for maintenance during the project

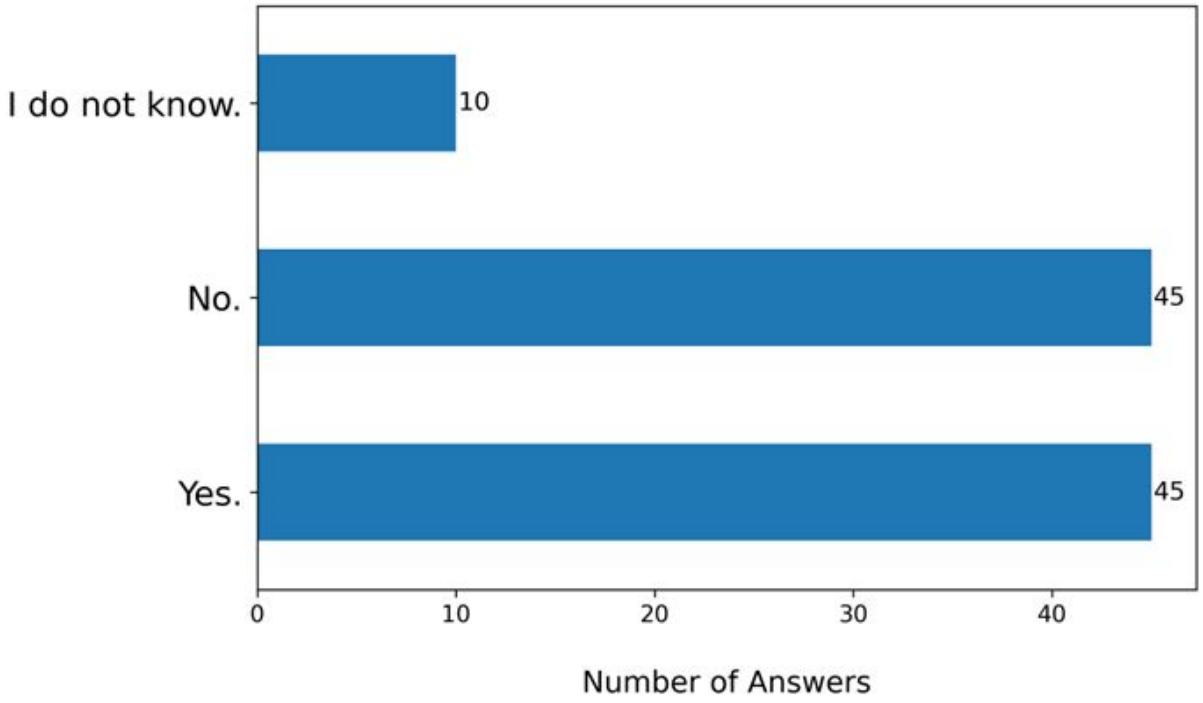


TDCC-NES
Natural & Engineering Sciences

Budget allocation & financial sustainability- Budget for maintenance after the project



Budget allocation & financial sustainability- Whether participants or their colleagues write hours for maintenance in financial systems



TDCC-NES
Natural & Engineering Sciences

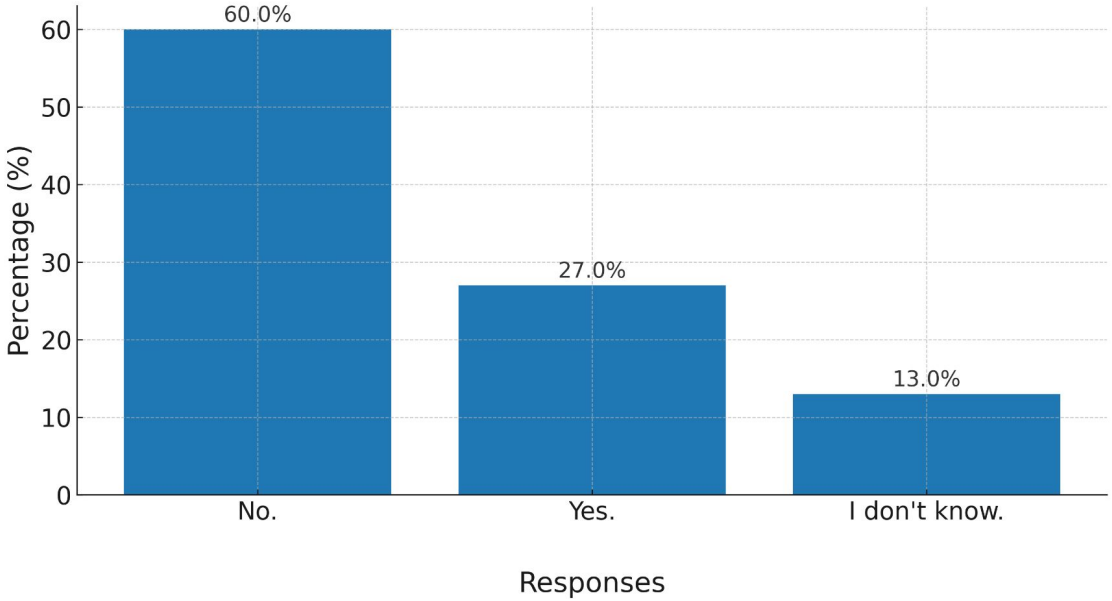
Conclusions

Budget allocation & financial sustainability

- Maintenance is rarely budgeted for—either during or after the project ends.
- Only a minority of respondents report having specific maintenance budgets, and even fewer indicate organizational-level funding.
- When funds are available, they are mostly used to reassign existing personnel rather than hiring new staff.
- Participants often do not know why maintenance is unfunded, reflecting both structural and communication issues. This finding reinforces the hidden nature of maintenance costs and the vulnerability of software after the end of a project.

Human resources & maintenance responsibilities

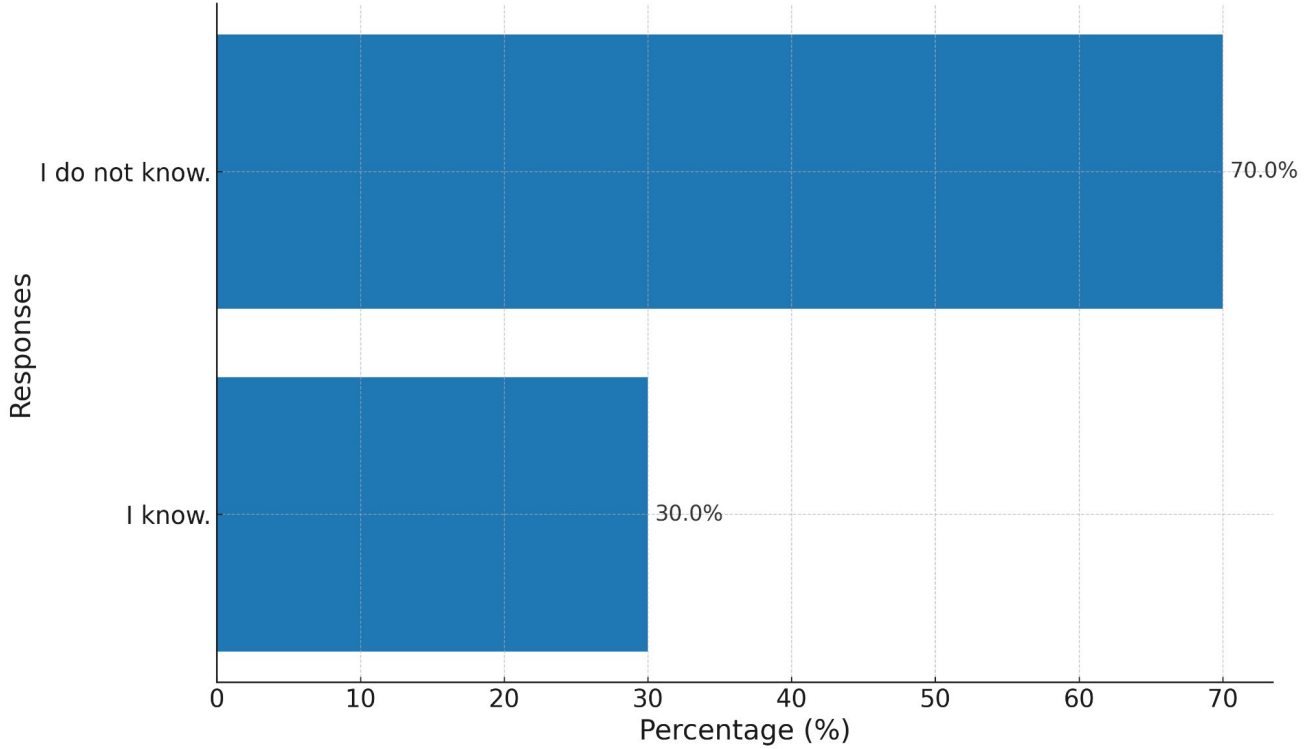
Human resources and maintenance responsibilities - If maintenance is part of participants' formal job descriptions



TDCC-NES
Natural & Engineering Sciences

Digital Competence Centre
Your one-stop for research IT and data

Human resources and maintenance responsibilities - Contracted time allocated to software maintenance



TDCC-NES
Natural & Engineering Sciences

Conclusions

Human resources and maintenance responsibilities

- Only a few participants report that software maintenance is explicitly part of job descriptions.
- The task often falls to PhD students, postdocs, or anyone available, rather than being formally assigned.
- Contractual time allocated to maintenance is minimal or unknown, yet actual time spent can be significant indicating a disconnect between practice and formal planning.
- Maintenance is performed, but informally and without structural safeguards, suggesting a reliance on goodwill and individual dedication.

Conclusions

Human resources and maintenance responsibilities

- Field-specific differences in maintenance workload planning also emerged. In Astronomy & Astrophysics, participants reported relatively high formal responsibility (35%) and significant actual maintenance time (32%). In contrast, participants in Computer Science & AI and Engineering & Technology reported no formal time allocation and little or no contract recognition, despite spending meaningful time on maintenance.
- This highlights a systemic disparity between disciplines, reinforcing the need for domain-sensitive policy solutions.

Recognition

Conclusions

Recognition

- Participants identify funding, sustainability, and lack of recognition as the biggest challenges to software maintenance.
- Majority report that development work is recognized more than maintenance in career decisions.
- Hour logging is inconsistent and, when tracked, is usually recorded under generic research posts, not maintenance-specific categories, making the work administratively invisible.

Conclusions

Recognition

- Measurement of software impact, when it occurs, is mostly informal and based on usage indicators like downloads or citations.
- A substantial number of participants report no evaluation or awareness of how the impact is assessed.
- The open comments confirm these trends, voicing frustration about short-termism, lack of career rewards, and structural neglect.

Final reflection

- Results point to organizational invisibility of development and maintenance of research software.
- The maintenance of software is widespread and time-consuming yet remains structurally unsupported.
- Recognition and funding are inconsistent, often relying on individual initiative and ad hoc arrangements.

Advice

- To bridge this gap, both research organizations and funders must integrate **research software development and maintenance into career systems, budgeting processes, and infrastructure planning**.

- Sustainability strategies must account for **disciplinary norms**, **institutional structures**, and **informal practices**, which vary significantly across the NES domain.

Advice

- Field-specific results reveal that these structural differences are not evenly distributed.
- In **Astronomy & Astrophysics**, participants reported relatively high levels of formal recognition and time allocation for maintenance tasks, alongside high actual time spent.
- In contrast, fields such as Computer Science and Engineering showed little to no formal assignment or contract time, despite ongoing maintenance work.
- These patterns suggest that any effective policy or funding response must be sensitive not only to the technical nature of research software, but also to the **disciplinary culture and institutional incentives** that shape how maintenance is assigned, resourced, and recognized.

Thank you!

Dutch Thematic Digital Competence Centre for the Natural & Engineering Sciences (TDCC-NES)

Published April 14, 2025 | Version v1

Financing Sustainable Research Software

Beygu Koopmans, Burcu (Project leader)[1]

Contributors

Project leader: Beygu Koopmans, Burcu[1]

Project members: Hut, Rolf[2]; Williams, O.R.[1]

Related person: Vinju, Jurgen[3, 4, 5]

edu.nl/v6uvc

Download the data and the project results!

TDCC-NES
Natural & Engineering Sciences

Digital Competence Centre
Your one-stop for research IT and data

Backup slides

Background

TDCC-NES Bottleneck Projects - Roadmap v1.0

Facilitating long-term software preservation and sustainability

Software is developed and maintained by humans

Financial aspect

TDCC-NES
Natural & Engineering Sciences

Background

TDCC-NES Bottleneck Projects - Roadmap v1.0 *

1. Community building and establishing online presence
2. Creating a Training Hub
3. Facilitating long-term software preservation and sustainability
4. Increasing interoperability and integrating tools and workflows for FAIR

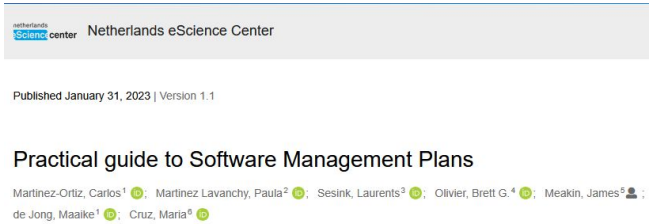
*TDCC-NES Roadmap v1.0

edu.nl/9yhgd

TDCC-NES
Natural & Engineering Sciences

Types of Research Software Taken into Consideration

- The goal of the project is not define different types of research software.
- We acknowledge that there will be examples that fall into multiple categories or none
- Software Categories according to eScience Center's Practical Guide to Research Software Management Plans



1. Software typically developed for a specific analysis
2. Software developed for a research project
3. Software developed or evolved to be mission critical

edu.nl/37mmy

TDCC-NES
Natural & Engineering Sciences

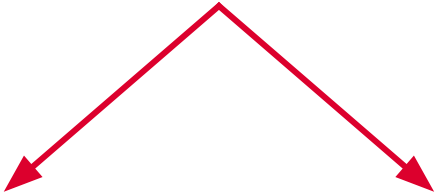
Future

- Given the findings;
  - **What do you think** could be a follow effort?
  - **What do you think** financial model can be improved such that research software development and maintenance tasks could be financed and managed more easily?
  - **What would you suggest** as solutions to financial managers, funding agencies, deans and directors?

Background

TDCC-NES Bottleneck Projects - Roadmap v1.0

Facilitating long-term software preservation and sustainability



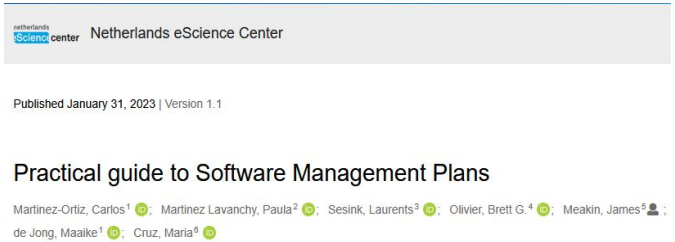
Training

Financial aspects

TDCC-NES
Natural & Engineering Sciences

Types of Research Software Taken into Consideration

- The goal of the project is not define different types of research software.
- We acknowledge that there will be examples that fall into multiple categories or none.
- Software Categories according to eScience Center's Practical Guide to Research Software Management Plans*



1. Software typically developed for a specific analysis
2. Software developed for a research project
3. Software developed or evolved to be mission critical

*Practical guide to Software Management Plans

Survey Structure

- The survey consisted of 27 substantive questions in a variety of formats, including *multiple-choice* (with some allowing multiple selections), *free-text* responses, *rating scales*, and *percentage-based distributions* .

- Participants could choose more than one option where applicable and were also able to provide their own answers in *free-text* form. If none of the listed options fit their situation, they could use the *Other* option to write a custom response.

- *Conditional branching* logic was used to tailor the questionnaire based on respondents' answers (e.g., only those involved in funded projects were shown financial-related questions), ensuring relevance and reducing survey fatigue.

Survey Structure

- As a result, not all participants will answer every question.
- Certain questions are only seen by participants based on previous responses.
- Funding-related questions are only shown to those working on funded projects.
- The survey may end early for some participants based on their responses.For example, participants who indicated no involvement in research software were automatically redirected to the end of the survey.

Survey Structure

- As a result, not all 197 participants answered all 27 questions.
- Each response was anonymized, with a unique *RecordID* assigned to allow consistent linking across questions for each participant.
- No personal data—such as names, addresses, email addresses, or the names of participants' organizations—was collected. At the beginning of the survey, participants were asked to give *informed consent* before proceeding.

Data Collection

- The survey was evaluated in advance, and the questions were reviewed by colleagues outside the project team to ensure clarity and relevance.

- Survey dissemination was conducted through multiple channels, including local Digital Competence Centers (DCCs), the TDCC-NES community, Open Science community and personal professional networks.

- An anonymous survey link was distributed via email across these networks to reach a broad and relevant audience. All the Dutch Universities and NWO I institutions were reached out.

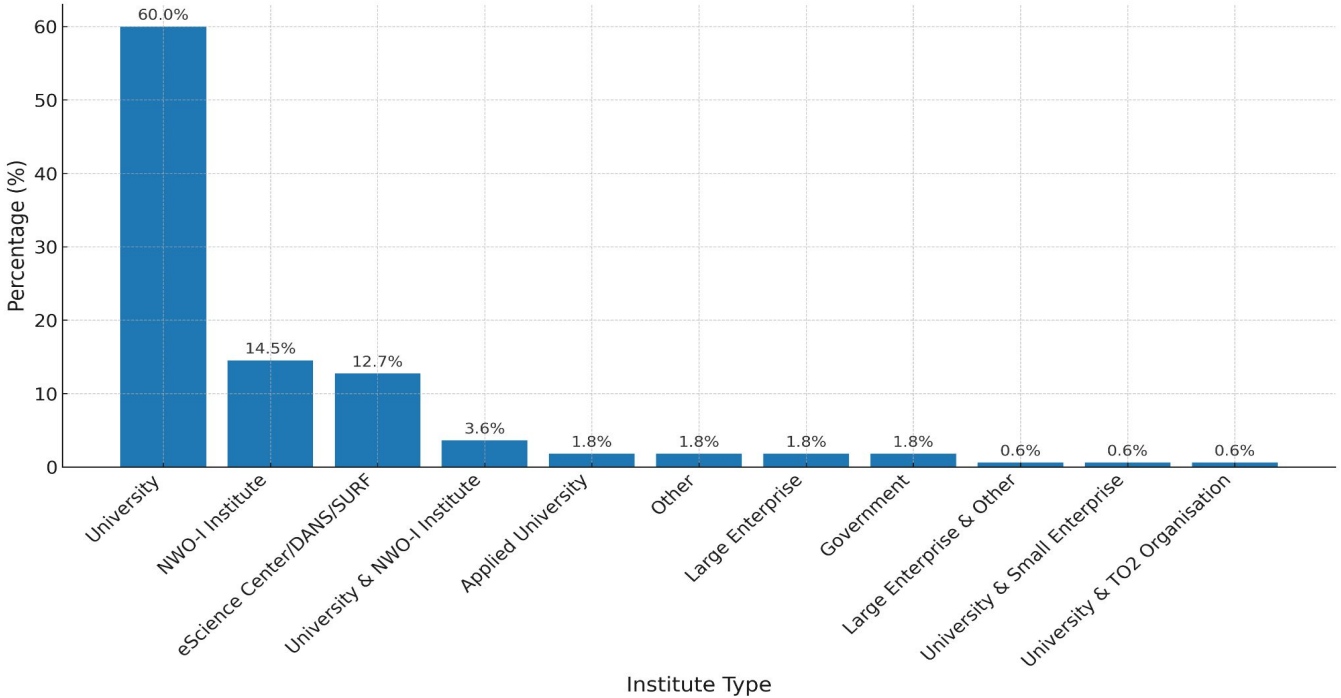
For whom the software is developed?

Context & use of the research software

- Most research software is developed with broader use in mind—by a research group or scientific community.
- It also covers multiple types, from specific tools to core infrastructure. This shows that software developed in NES is not purely individual, but part of collective research output.
- Astronomy & Astrophysics contributed significantly across all software types—most notably in Specific Methods and Community Contributions. This suggests that researchers in this field produce both specialized tools for domain-specific tasks and reusable software for broader community use.

Results

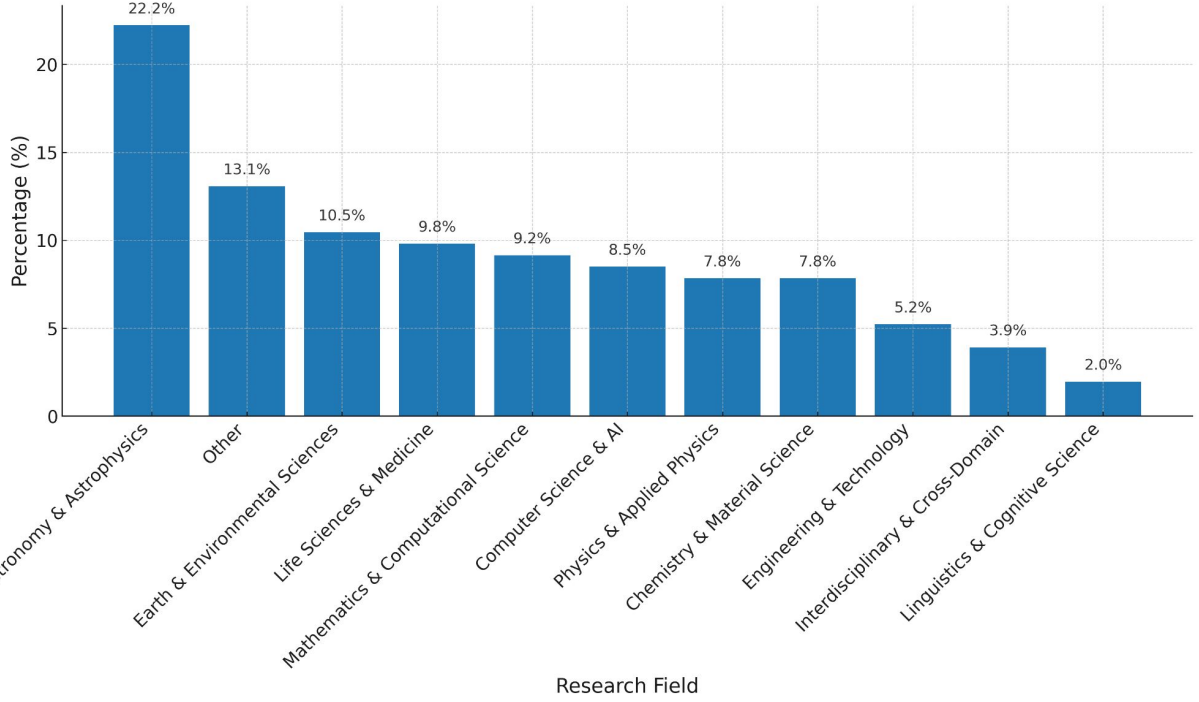
Participants profiles & roles- Type of institutes



TDCC-NES
Natural & Engineering Sciences

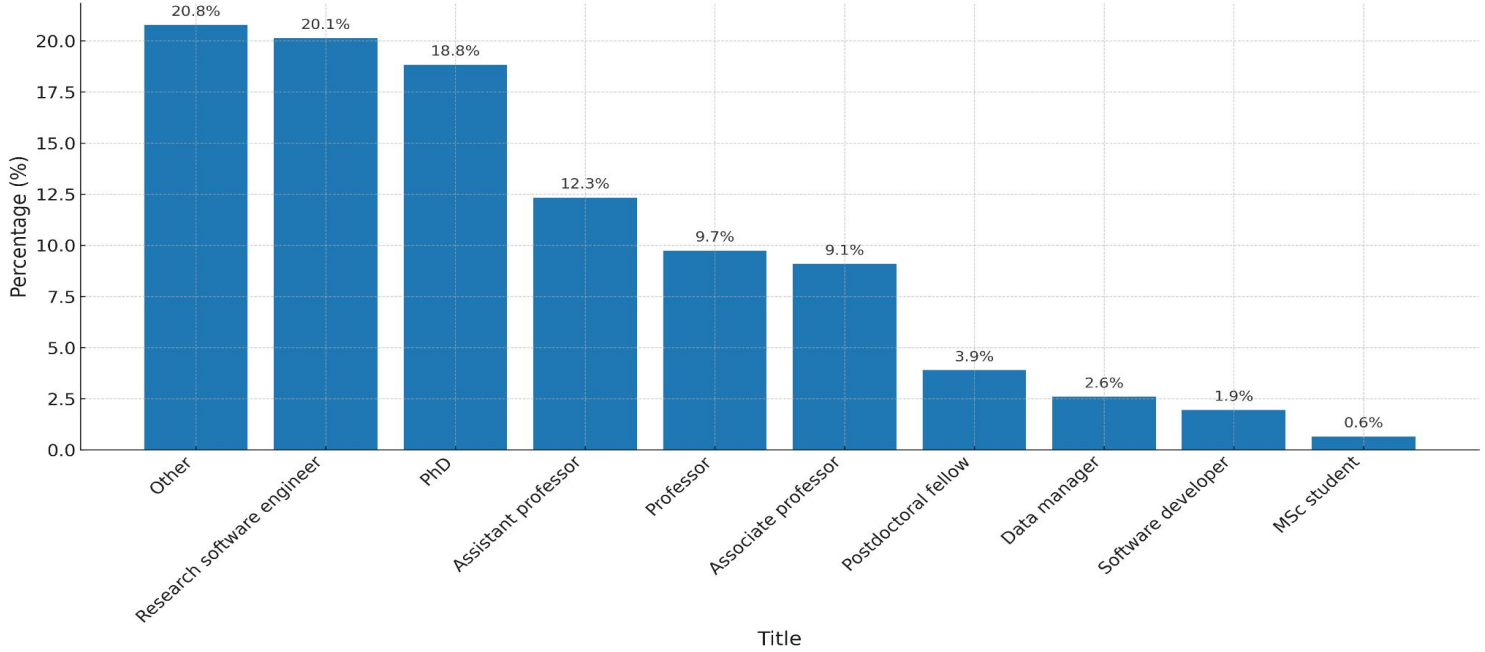
Results

Participants profiles & roles - Research field



Results

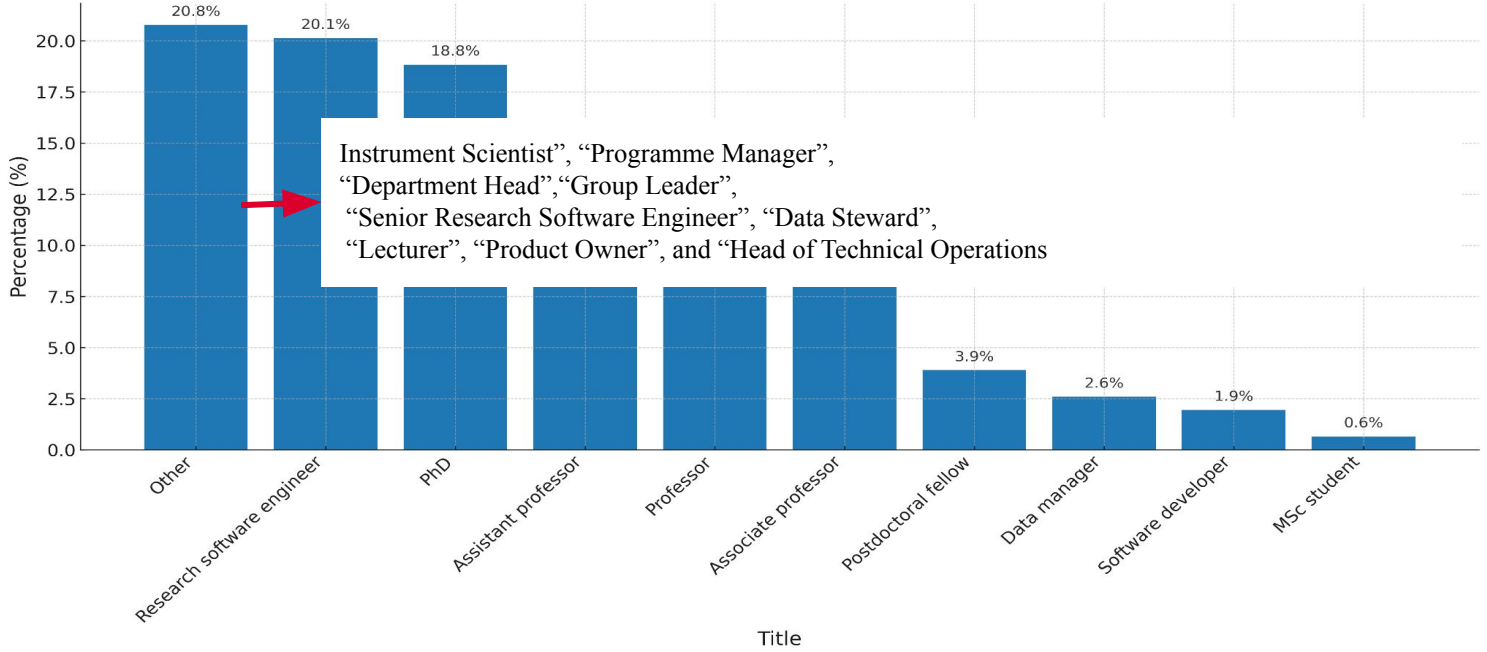
Participants profiles & roles - Job titles



TDCC-NES
Natural & Engineering Sciences

Results

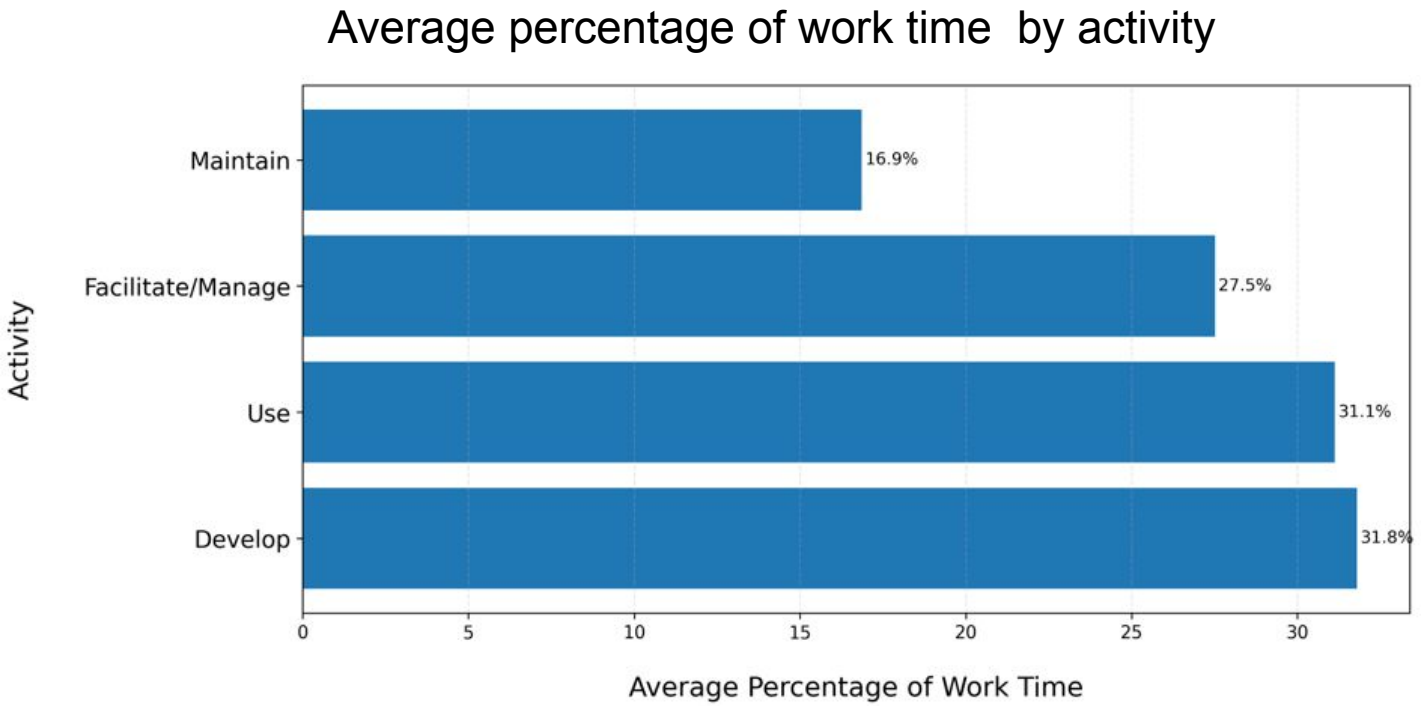
Participants profiles & roles - Job titles



TDCC-NES
Natural & Engineering Sciences

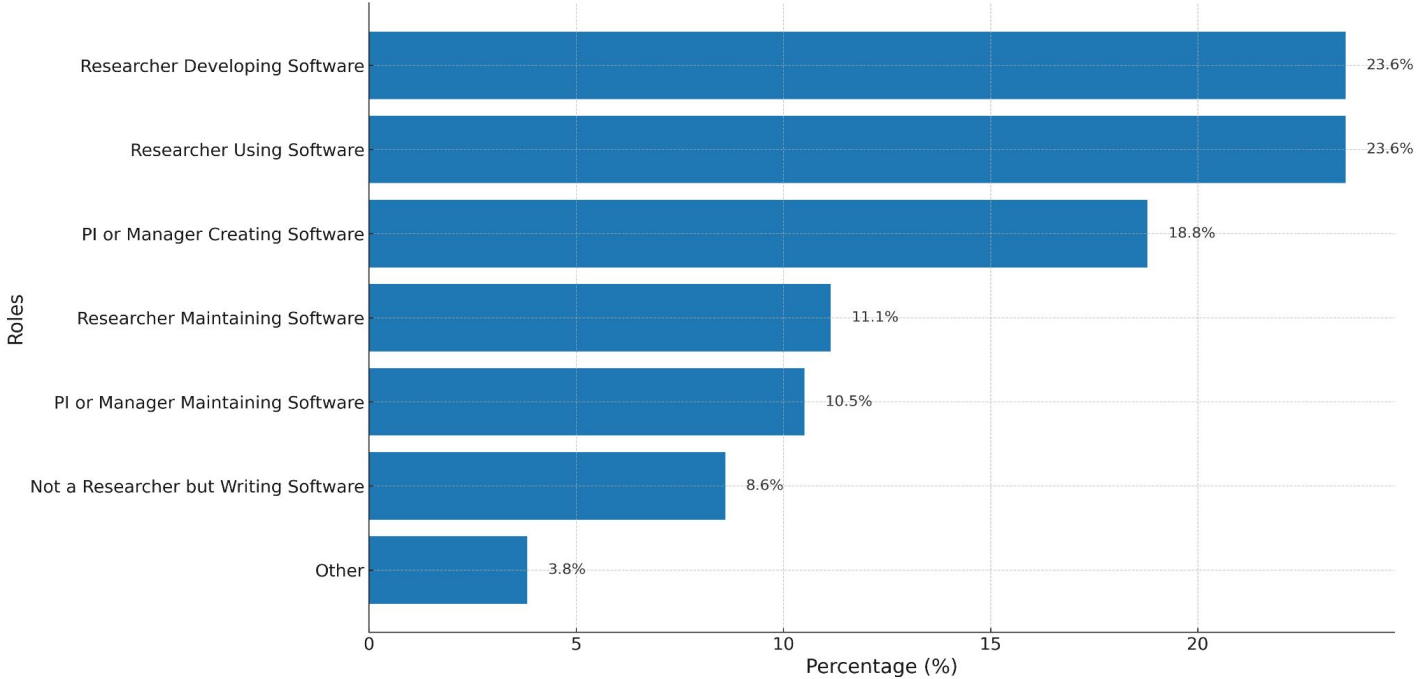
Results

Participants profiles & roles - Roles and time spent on activity



Results

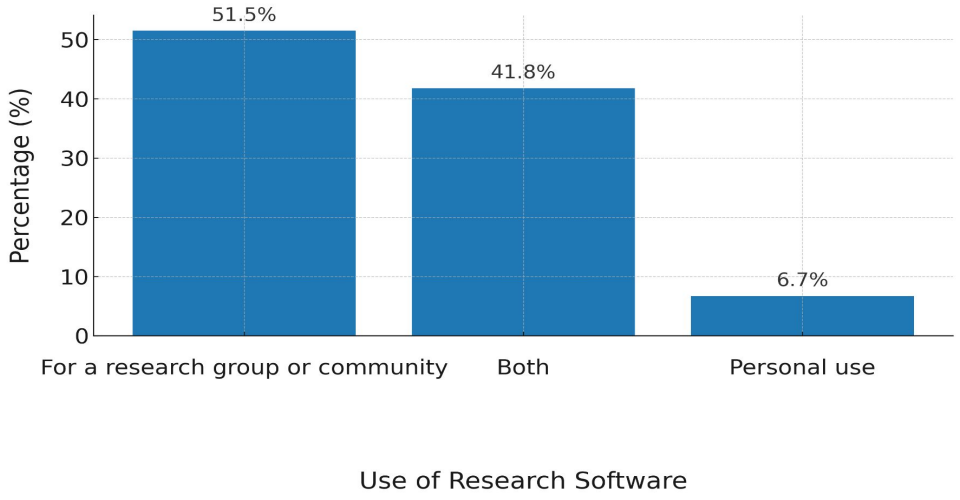
Participants profiles & roles - Roles and time spent on activity



TDCC-NES
Natural & Engineering Sciences

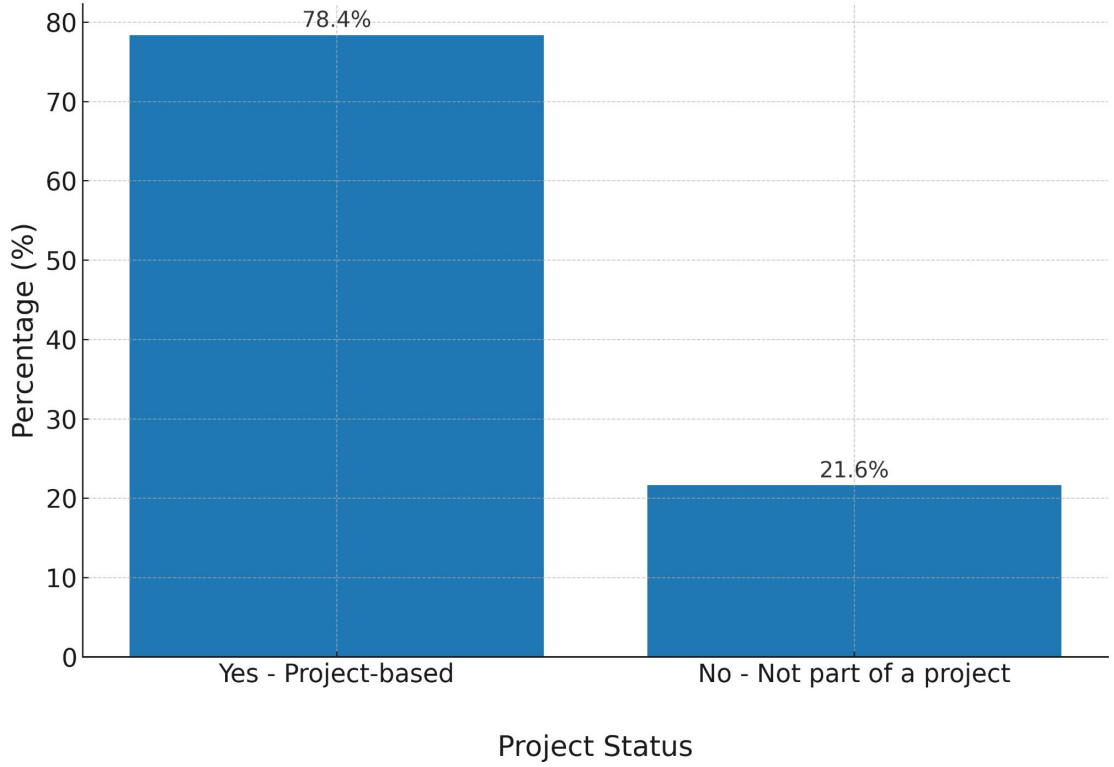
Results

Context & use of the research software - Intended audience



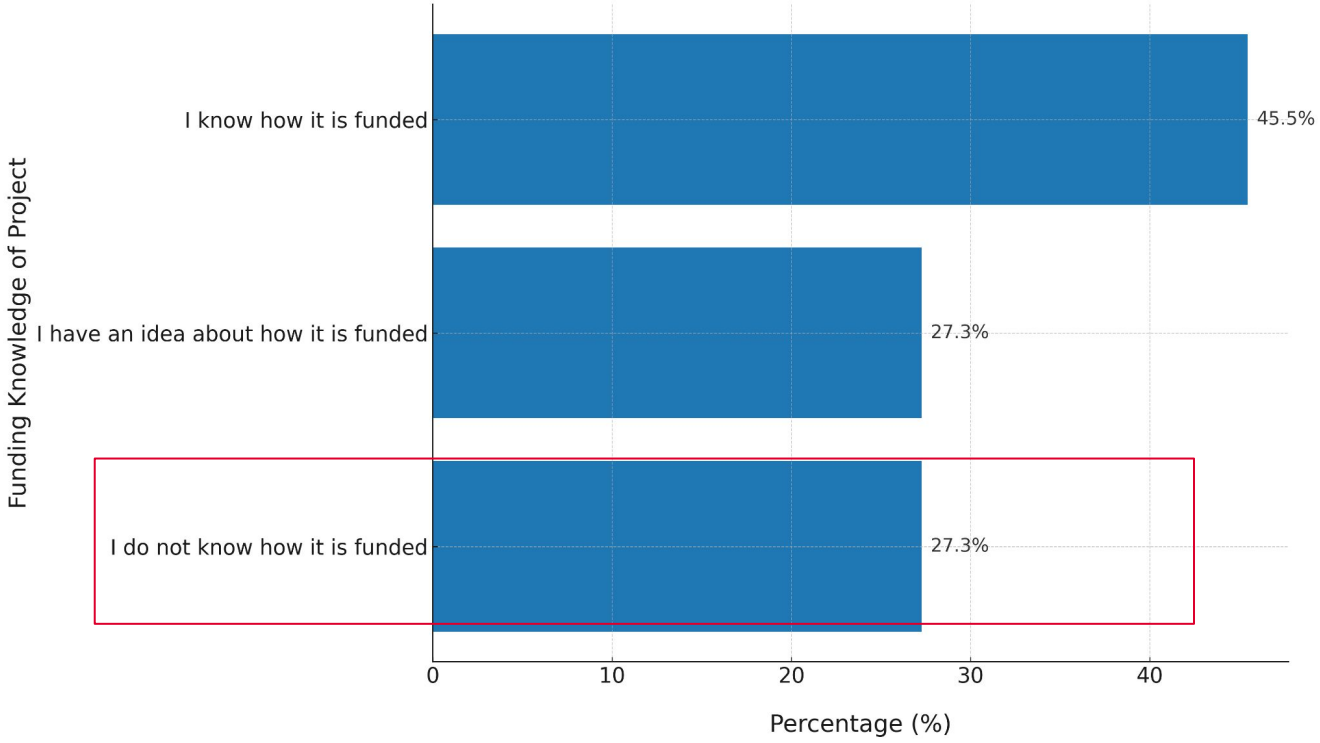
TDCC-NES
Natural & Engineering Sciences

Project & funding awareness - Is the research software part of a project



TDCC-NES
Natural & Engineering Sciences

Project & funding awareness - If the research know how the project is funded



TDCC-NES
Natural & Engineering Sciences

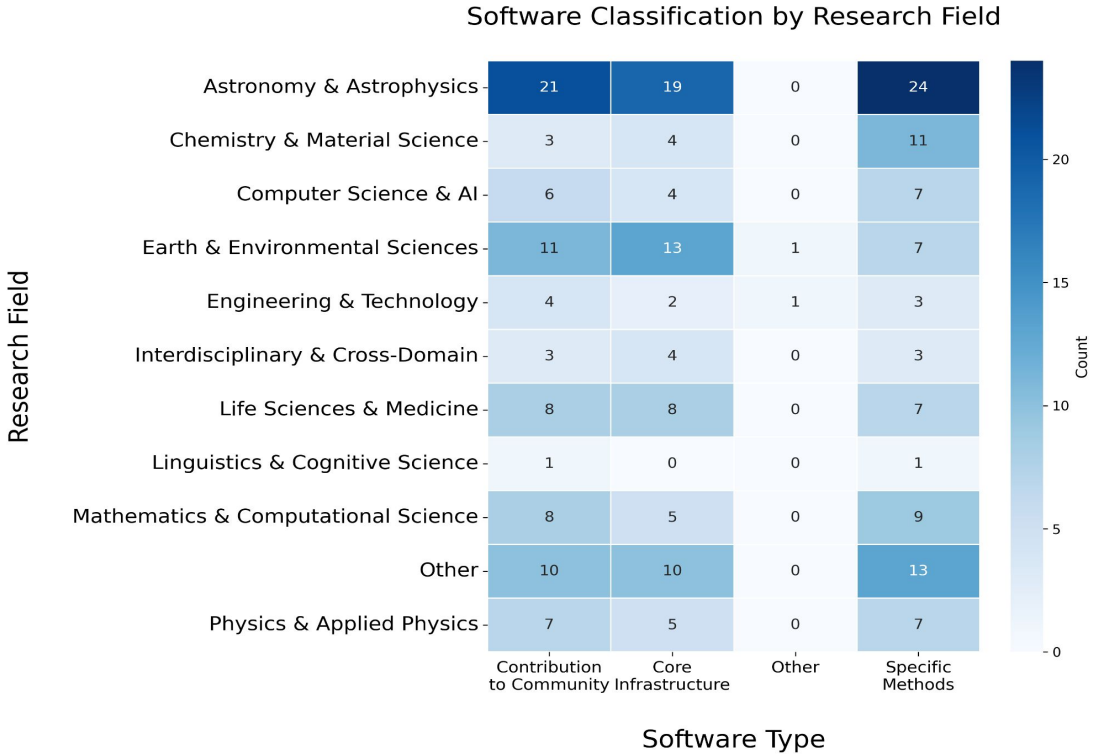
Conclusions

Project & funding awareness

- Software development is mostly embedded in funded research projects, but many respondents are unaware of how this funding is structured.
- Maintenance is often not taken into account.
- Funding sources are diverse, often described as “patchwork,” and lack consistent categories.
- A portion of respondents report no funding at all or contribute in their spare time.
- Grants named suggest dependence on NWO, EU, and eScience Center, but transparency and clarity are lacking.

Results

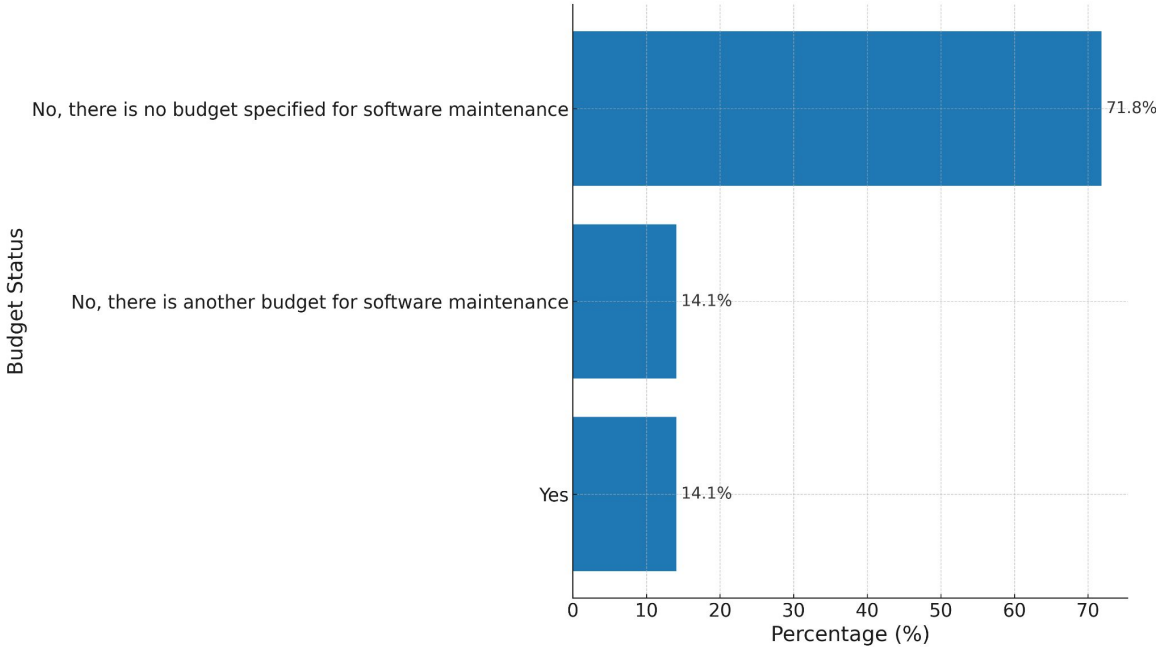
Context & use of the research software



TDCC-NES
Natural & Engineering Sciences

Results

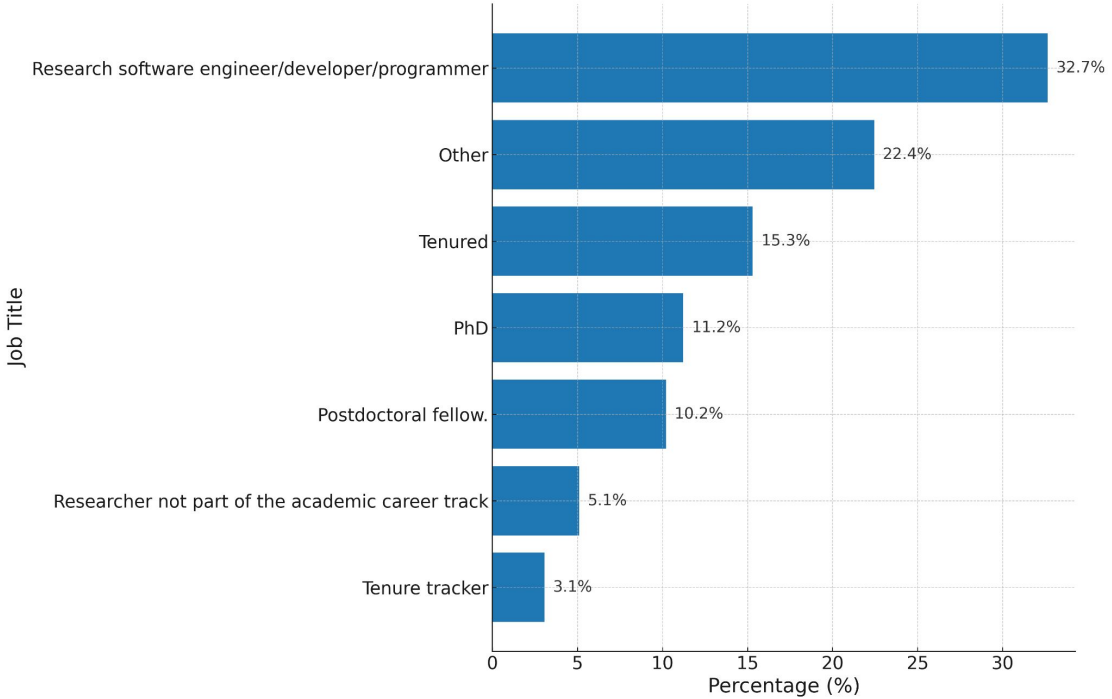
Budget allocation & financial sustainability- If there is budget dedicated for maintenance in the institution



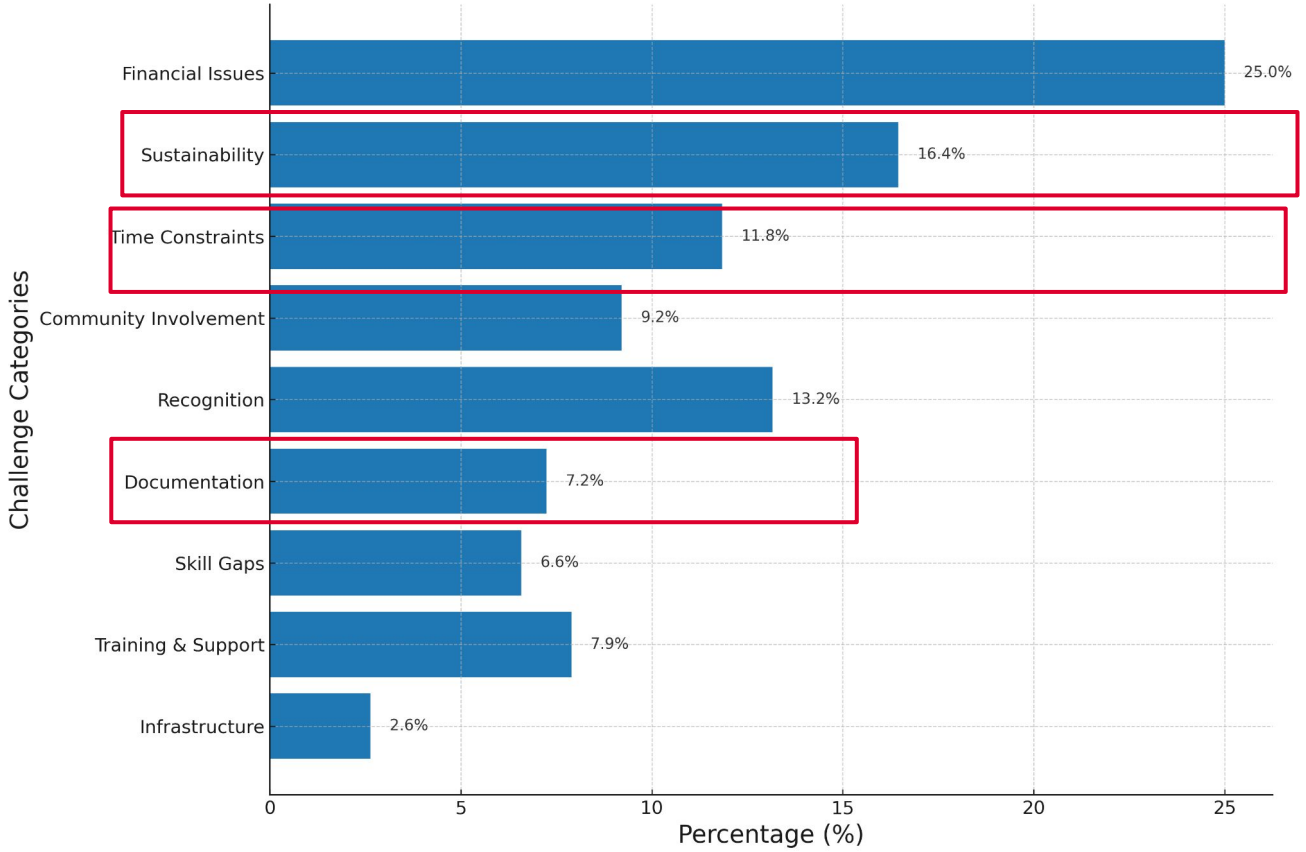
TDCC-NES
Natural & Engineering Sciences

Results

Human resources and maintenance responsibilities - Career stages of individuals responsible for software maintenance

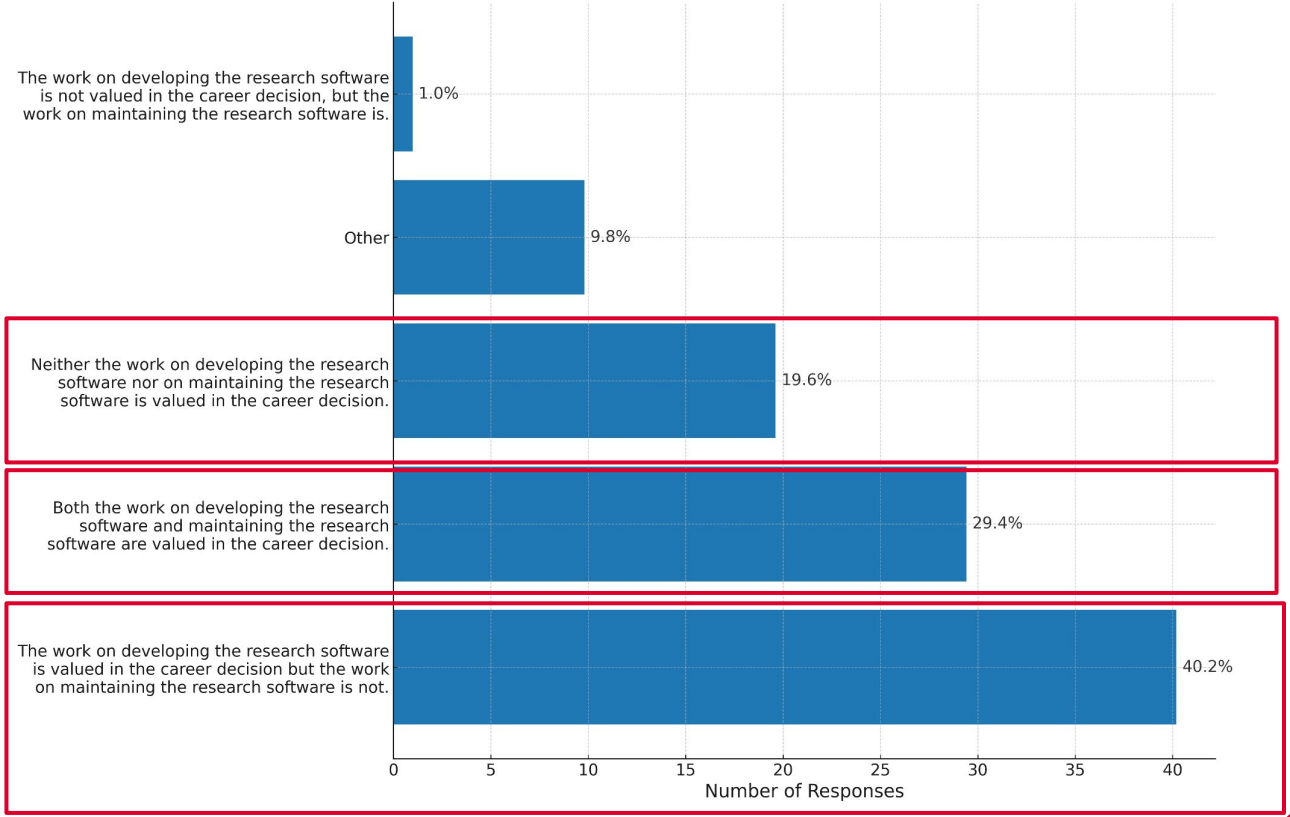


Recognition - What is the biggest challenge ?



TDCC-NES
Natural & Engineering Sciences

Recognition- If work on development and maintenance is taken into account in career evaluation



TDCC-NES
Natural & Engineering Sciences

Final reflection from participants

Participants emphasized serious structural issues around funding, continuity, and careers.

Many described the difficulty of securing long-term support for software maintenance, noting that funding often ends with short-term contracts (e.g., PhDs, postdocs) and leaves no succession plan. This mismatch between continuous maintenance needs and temporary funding was repeatedly called unsustainable.

Final reflection from participants

Respondents highlighted a lack of institutional recognition for software work.

Maintenance activities were often seen as undervalued compared to publications, and some felt that time spent on software even harmed their academic standing. These concerns were raised by individuals across roles — from PhDs to software engineers.

Final reflection from participants

Participants pointed out diverse situations not captured by the survey and broader structural barriers.

Several noted unique roles (e.g., freelancers, non-academic positions) or one-off software projects that do not fit standard assumptions. Others mentioned technical challenges, missing infrastructure, or risks tied to proprietary platforms. Despite frustrations, some respondents expressed appreciation for the survey and offered concrete suggestions for clearer roles, better handover processes, and improved support structures