CodeRefinery (Phase 1) Final Report

The report is drawn-up in agreement between NeIC as the project owner represented by Michaela Barth and the project manager Radovan Bast. It is verified through a steering group decision.

	Name	Partner/Activity	Date
From	Radovan Bast	UiT	2018-10-22
Reviewed by	Michaela Barth	NeIC	2018-11-13
Approved by	<SG members>		2019-03-04

Edition History:

Issue	Date	Comment	Author/Partner
1.0	2018-10-22	Report shared with steering group for comments.	CodeRefinery team
1.1	2019-03-04	Final report.	CodeRefinery team

Abstract:

The two-year CodeRefinery project has delivered 13 three-day workshops on research software development tools and workflows to ca. 400 researchers and students across the Nordics and deployed a continuous integration service as well as a source code repository hosting platform accessible to the Nordic research community. The project also managed to forge a coherent and functioning remote team.

Comprehensive information about the project		
Type of project	Delivery project	
Scope	Result	Development of training material and delivery of 18 workshops/events, as well as deployment of code hosting and testing services.
	Time	2 years
	Cost	2.75 FTE (which includes 0.25 FTE for SGAS support)
Documentation location	- https://drive.google.com/drive/folders/0B3i0ZR ReqpSxUW1BaWw4ZVJZd2s - https://coderefinery.org - https://github.com/coderefinery - https://wiki.neic.no/wiki/CodeRefinery - https://wiki.neic.no/int/CodeRefinery - https://neic.no/coderefinery/	

Table of Contents

1. Basic information

1.1. The project

Research heavily relies on software and too often research software is developed inefficiently, often without peer review, and without considering reproducibility and reusability. The CodeRefinery addresses these issues for the Nordic research communities with training workshops and infrastructure services.

CodeRefinery (phase 1) was a 2-year project which started in October 2016 and concluded in September 2018. During phase 1, the project has delivered 13 three-day workshops across the Nordics and 5 shorter events and taught over 400 students and researchers in the necessary tools and techniques to create sustainable, modular, reusable, and reproducible software.

The project has informal (and soon more formal) ties to The Carpentries and the Software Sustainability Institute and can be seen as an extension of The Carpentries to more advanced programming skills and the beginning of the formation of a Research Software Engineering network in the Nordics. The CodeRefinery project has developed a very fruitful informal cooperation with Aalto Science-IT which resulted in two workshops and collaborative lesson development.
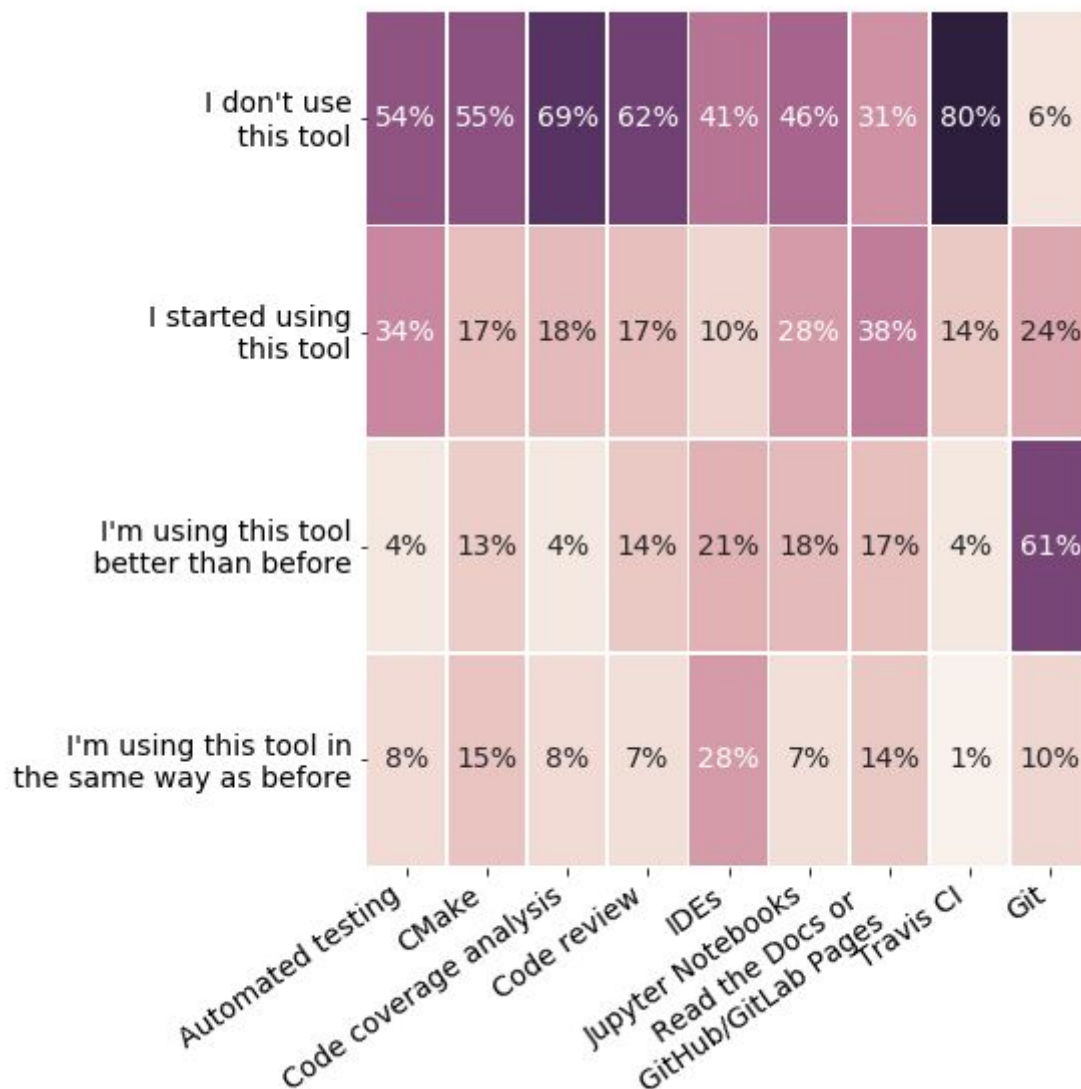
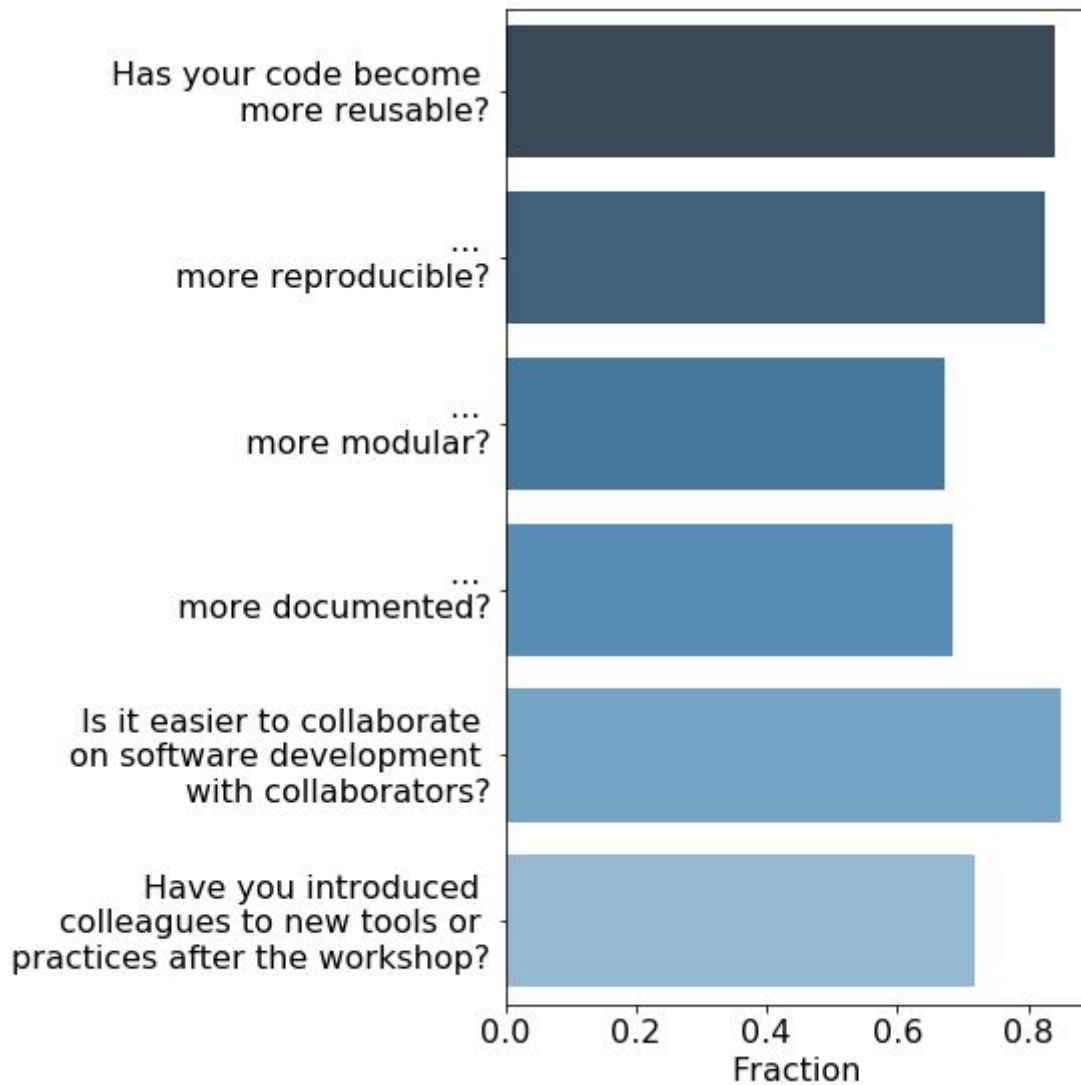## 1.2. Background and Business Case

The strategic result of phase 1 of the project was to enhance the productivity of researchers by improving the research software development and collaboration practices in the in the Nordic region for a significant number of students and researchers. The CodeRefinery project has also significantly contributed to increase the visibility of NeIC among the Nordic research community. In general, CodeRefinery has been motivated by providing added value to researchers, offer new services beyond national efforts, pooling competencies, and continuously strive to improve.

1.3. Summary

In phase 1, we had learned that students and researchers who write software to produce or analyze data are in critical need of the proposed training portfolio on software management, and currently only NeIC has the scale, network, and expertise to fill this need in the Nordics until these workshops become part of university curriculum. Phase 1 of the project could match this obvious need of the research community exceptionally well as shown by almost always fully booked workshops and the resulting surveys.

This project has demonstrated impact, which is measured through a post-workshop survey which is sent out to all former participants 3-6 months after attending a workshop: https://coderefinery.org/#impact. The two graphs below show how former participants use various software development tools after attending a workshop (left), and how their code and collaboration with colleagues has changed (right). These results are based on 71 survey responses obtained to date -- 48% of which are graduate students, 13% postdocs, 9% researchers, 9% assistant/associate/full professors and 19% other occupations (e.g staff scientists, scientific programmers):

	Automated testing	CMake	Code coverage analysis	Code review	IDEs	Jupyter Notebooks	Read the Docs or GitHub/GitLab Pages	Travis CI	Git
I don't use this tool	54%	55%	69%	62%	41%	46%	31%	80%	6%
I started using this tool	34%	17%	18%	17%	10%	28%	38%	14%	24%
I'm using this tool better than before	4%	13%	4%	14%	21%	18%	17%	4%	61%
I'm using this tool in the same way as before	8%	15%	8%	7%	28%	7%	14%	1%	10%

The project summary can be summarized with the following quote after the workshop at Aalto University, May 2018:
"CodeRefinery was the best course I have had. If you need some quote to support it for the future, just make something up from me." [sic]

2. Achievement of objective

2.1.   Result, delivery objects

The project has delivered 13 three-day workshops together with a number of shorter events and workshops: https://coderefinery.org/workshops/.

3-day workshops:
- Reykjavik, August 21-23, 2018
- Oslo, June 12-14, 2018
- Espoo, May 29-31, 2018
- Lund, May 15-17, 2018

- [Turku, March 20-22, 2018](#)
- [Trondheim, February 27 - March 1, 2018](#)
- [Espoo, December 12-14, 2017](#)
- [Linköping, November 7-9, 2017](#)
- [Aarhus, October 24-26, 2017](#)
- [Tromsø, June 19-21, 2017](#)
- [Copenhagen, May 9-11, 2017](#)
- [Stockholm, February 20-22, 2017](#)
- [Espoo, December 14-16, 2016](#)

Shorter workshops and other events:

- [Umeå, October 16, 2017](#)
- [Manchester, September 8, 2017](#)
- [Umeå, May 29, 2017](#)
- [Stockholm, May 19, 2017](#)
- [Oslo, April 6, 2017](#)

Some of the participants who had received the workshop training will integrate learning outcomes in the work of their research groups and thereby amplifying the result as shown by survey results in Section 1.3.

The continuous integration service use cases (Nordugrid ARC, Dalton, LSDalton, and DIRAC) are documented in https://coderefinery.org/ci/. We have to admit that the adoption of this service was slow since the launch of this service was late. Another possible explanation for the slow adoption of this service and lack of advertisement is that the team staff did not use or need this service for their own work.

The GitLab source code hosting platform https://source.coderefinery.org was launched one year into the project and within less than 12 months has grown to 150 users, 235 projects, and 27 groups, which is a success given that the service was neither heavily advertised nor (yet) used as part of the lesson material.

Compared to the project plan, we have not managed to deliver best practice guides as part of phase 1. Given the resources and the ambitious workshop plan we had to prioritize work on improving workshop material and organization of workshops instead of preparing online guides. We have moved this goal to the goals of the phase 2 project.

In addition to an overall benefit of increased competence in research software development among the workshop participants, this project also increased staff competence in teaching and coordination.

A significant result of the project is lesson material (https://github.com/coderefinery/) which is open, reusable, and will be continuously improved as part of phase 2 and onwards.

2.2. Time
The project kept its time objective.

2.3. Cost
The project kept its cost objective.

3. Project execution
The first important meeting was held in Linköping, Apr 4, 2016 (https://wiki.neic.no/wiki/Category:CodeRefinery_meetings) where the project was discussed and received very valuable input. Two persons in the room would later be part of the steering group.

The project start was 6 months later, Oct 1st, 2016, which is relatively early after the summer break and posed a challenge to assemble a team. The staffing of the project took time - as is typical for new projects - and was completed after the completion of the project plan, almost two months into the project.

The PM met the staff from CSC at a side-meeting of the NORDUnet conference 2016 in Helsinki 20-22 September. The first time where the entire team met was an in-person lunch to lunch meeting close to Arlanda, Sweden in November 2016. With this meeting we have very ambitiously started preparations for the first 3-day workshop which we have delivered only a month later in Helsinki. Within the following 20 months we would deliver 12 more 3-day workshops and few shorter events which was extremely ambitious.

We have started developing and testing the infrastructure services in 2017. We have deployed the GitLab service with the help of CSC staff without having even a mid-term service agreement in place since we wished to develop the service agreement and debug the service and backup at the same time to not lose precious project months. However, finding a project partner who would commit to hosting the GitLab turned out to be a challenge. After a lot of back and forth, the group around Martin Bech (DeiC) has offered to maintain

the service and this agreement is still in place today and this collaboration was and is a very good experience.

Due to staff changes from DK, the continuous integration service took time to launch and was operating for less than a year.

There were many meetings and events during the 24 project months but few stand out and were very important for the project:
- Research software engineer (RSE) conferences 2016 and 2017
- NeIC 2017 conference where we met Richard Darst who is an enthusiastic (informal) project partner ever since
- Team meeting in May 2018, together with Radek Lonka (NTNU) and the future project staff Anne Fouilloux where we have discussed and drafted a proposal for the phase 2 of the project and launched the Nordic RSE section.

The team met weekly via video (unless travel schedule prevented this or during vacation time). The project manager met the project owner weekly via video (again unless schedule prevented this).

4. Transferral of results

The main result of the project was the delivery of workshops and the recipient are the participants therefore a formal transferral of workshops to a "recipient" is difficult to document but we can refer to post-workshop survey results (https://coderefinery.org/#impact) and to feedback we have received during workshops (internal documents which are visible to the steering group: https://drive.google.com/drive/folders/0B3i0ZRReqpSxUEFNTmNRVURweTg).

The workshop lesson material (https://coderefinery.org/lessons/ and https://github.com/coderefinery) has not been formally transferred but it is publicly accessible and reusable under a Creative Commons license. It is the goal of the phase 2 of the project to find a sustainability model which will require to transfer the material maintenance from NeIC towards another entity or entities.

We were not able to transfer the continuous integration service yet due to its cost and/or lack of interest or resources from the project partners to commit to operating and maintaining such a service.

The GitLab service (https://source.coderefinery.org) has been transformed from a deliverable to a service level agreement with DeiC.

In general, we find that 2 years is too short to plan services, provision them, grow them, advertise them, grow a user community, and to form a finance model for transfering services and find a partner that has the hardware, the know-how, and the human resources to operate them. **Transferral of services is non-trivial in a non-uniform cross-border ecosystem.**

5. Collected experiences

5.1.   The project participants' experiences

[Sabry:] This project (to me) is a way to contribute to the community as well as a personal learning experience. This also helped me to build up a network across the Nordics with people with similar interests. One challenge I face is the unevenness in the level of knowledge of the participants. This is a natural thing in a project like this and more workshops we conduct I feel we are getting better at handling this. A thing I would like to have is a way to keep the project members committed for a longer time, in the order of two or more years.

[Thor:] From my point of view the project has been a great success. Every workshop has been highly appreciated by a large majority of participants, and it's apparent that the workshop material addresses a large training demand among PhD students and postdocs across a wide range of disciplines. I am also happy about how this project has matured, both in how practical issues are solved and how the skills of the project members has increased. I have certainly learned a lot. One negative aspect is that the financial situation of the project has been rather opaque. It's been unclear to me how much money is available and how it can be spent for workshop organization, instructor traveling and participation in conferences to present the project internationally. My feeling has also been that the amount of reporting and administration has been somewhat too large for a project with only 2.5 FTEs.

[Radovan:] Seeing the impact in our workshops first-hand was a very rewarding experience. Also the team spirit, good team communication, and our open discussions and constructive approach has been rewarding and inspiring and fun. The reporting load towards the organization has been in my opinion excessive for the size of the project. However, to be fair the paperwork would have been perfectly manageable if I did "only" the paperwork part but I wanted more: I wanted to actively participate in the workshop preparation and in the teaching and I wanted to "lead" by example, by taking and sharing

tasks, not by delegating and monitoring tasks. I have met a number of brilliant people through NeIC - friendships were formed and great learning opportunities were offered and taken and are very much appreciated. We are on to something and I appreciate that NeIC supports this important work.

5.2. The steering group's assessment

[Michaela Barth, SG chair representing NeIC as project owner:] During its short duration of only two years the phase 1 of CodeRefinery served as a NeIC flag ship project with good visibility and a broad reach and attractivity to many different disciplines. CodeRefinery was able to create very high impact with actually a quite minimal investment thanks to very engaged mentors, team members and a project manager who was able to motivate and convey the idea without letting anything important ever slip. All of them also believed in this project and their active contributions really made the project into the success it was. CodeRefinery phase 1 was also very appreciated and fully supported by the NeIC Board, so a follow-up project could be brought to life even in the complete absence of any remaining budget.
I'm sure that CodeRefinery, especially now with phase2 as the logical continuation, is on the best way to realize the systemic change we all want to see and sets a new state of the art for software creation within research.

[Rossen Apostolov, representing SNIC in the Steering Group]

"*Overall comments.*

The project has done exceptionally well and has met all objectives. It has successfully developed an advanced training program by applying modern best practices for user training. The program objectively meets the needs of the community and offers courses that are not covered by other similar initiatives. The courses have been highly praised by participants. Training has been delivered at full capacity. The team is highly motivated and dedicated to the project. The leadership has a clear vision for driving the project and has been able to execute the planned work successfully. The project has been able to start close interactions and collaborations with important, and much larger, initiatives in the same domain such as SSI (UK) and the Carpentries (US).

Recommendations.

1) Management activities have been run very well, and engagement of the team has been successful in delivering on promised results. However, responsibilities have not been consistently shared evenly among the team, which resulted in overload on the project

leader. It is advised that in future operations the full set of tasks (both technical and administrative) are appropriately shared among the whole team. Administrative assistant role has been planned for the second round of funding, which will considerably help. Still, the team leader and deputy should consider and work on optimal spread of effort.

2) Related to the last recommendation, (and this is only my speculation, and I might be wrong!) it is possible that some of the personnel in this first phase of the project might not have been comfortable with running "boring" admin tasks. If that has been the case, I would advise the national representatives to clarify the objectives and needs prior to hiring personnel.

3) Another topic, of which I have been very critical about throughout the project, is the unnecessary overuse of project management (PM) methodology. Applying best practices for PM is very important but standard "templates" need to be revised considerably to reflect _the size_ of the project. CodeRefinery is effectively only 3 people working full time. For such scale, a minimal amount of reporting is completely sufficient. The NeIC executive board should reconsider whether the full PPS process makes sense for projects of such size. My recommendation for the next phase (36 months) is to have a _short_ technical report around month 7/8 (i.e. Apr/May 2019), a mid-term review around month 18 (March 2020), and final review towards month 35/36 (Aug/Sep 2021). The team and its leader have shown very strong capabilities and competences. Thus, I don't think stricter monitoring is required."

5.3. The reference group's (use cases) assessment

This project was operating without a reference group due to the small size of the project and since the users of the project (workshop participants) are not organized into any obvious structure which could be used as a reference group.

6. Recommendations

Project start right after or soon after the **summer break** is problematic. Favor project start with some time-separation after vacation breaks to allow the partnering organizations to process the paperwork.

Face to face meetings matter. The first meeting should be early and makes later remote meetings and communication easier. In this project we were fortunate to meet often when organizing and teaching at workshops and this helped to improve the team cohesion.

All staff is working on many projects, not only CodeRefinery and team members cannot be expected to follow the team chat every day or restrict discussion to only one topic - therefore an efficient **threaded and asynchronous communication tool** and workflow is very important and the team has embraced a threaded chat solution.

Minimize the toolset: over the course of the project we have tried a number of tools but we have converged towards Google Drive and a threaded, asynchronous team chat, as well as GitHub. This combination turned out to be very effective and relatively intuitive. Meeting minutes as Google docs were easy to copy and adapt for the next meeting, retaining sticky items. There was no confusion where they can be found and which files to modify. Trello turned out as one tool too much. TypeForm was an excellent tool but it turned out to be problematic to store registration data outside of institutional servers. Zendesk is a very smooth tool for support request tracking but its pricing model does not match our situation where we have almost 10 "agents" who work only part time on the project and where we have only very few support questions.

Minimize meetings but provide a place to discuss. This was mostly the chat but we have also converged to a weekly opt-in video meeting schedule with meetings of most 60 minutes.

For the success of this project it is important that the staff originating from infrastructure providers is **well connected to the research community**. This was a challenge and a problem in this project as it is in other NeIC projects and probably one of the reasons why adoption of the infrastructure services among the research community is slow since many/most researchers in the Nordics simply do not know about these services. If project members have no stake in services, services are not likely to be successful.

Anchoring services within NeICs partners and getting long term commitments was difficult. Getting users on board without long-term commitments from NeIC partners was difficult. Getting long-term commitments without a large user base was difficult. The recommendation is that service commitments need to be part of the collaboration agreement. It is not enough if the collaboration agreement only focuses on FTE costs and the service part is then up to the team to locate and assemble.

Starting services and drafting service level agreements without creating an efficient and formalized **support structure** was a mistake. The missing support structure was at times absorbed by the PM which created strain for the PM and for the service users.

A persistent **project website** is important if the project is to persist. It makes probably very little sense to create new websites for every new administrative epoch (such as phase 2

of the project). We recommend to use **Twitter as an effective news channel** (this project might soon have more Twitter followers than NeIC).

Building a community is extremely non-trivial and we believe that the solution is not through new tools, but through **openness and through in-person events**. An example is the CodeRefinery mailing list, which we have launched, which many persons joined, but which basically nobody used. However we always had a spike of registrations and new Twitter followers after in-person events and meet-ups.

Draft a data/privacy policy early and communicate it transparently. We did this very late (only in phase 2) and it bit us since one of the tools we have used for registering participants had a data leak in 2018 which forced us to contact all affected users.

Minimize bureaucracy for small projects such as CodeRefinery. NeIC should not impose same structure to all NeIC projects, independent of its size. There should be less reporting "inwards" and to fewer administrative units, and more reporting "outwards", towards the user community.

Maximize transparency and document **success stories** publicly and visibly. Instead of filling out countless reporting spreadsheets, report results in one or very few places (ideally on the project website) and link to these results. These results can be referenced in reports and are also visible to stakeholders and the user community and most importantly can be edited in one place only.

The writing of the project report (this report) should be started very early in the project and edited during the project, not at the end. It would have been nice if the mid-term report used the same format as the final report or if it was the same document, simply evolving. The fact that both are disconnected documents with very different formats did not help the project or those who review it.

7. Other

8. References

Ref.no.	Document name/designation	Edition, date

1	Project directive: https://wiki.neic.no/w/ext/img_auth.php/5/5a/Collaborative_Infrastructure_for_Scientific_Software_Project_directive.doc	
2	Collaboration agreement: https://wiki.neic.no/w/ext/img_auth.php/6/6e/150904-Collaborative_Infrastructure_for_Scientific_Software_Collaboration_Agreement.pdf	
3	Project plan: https://wiki.neic.no/w/ext/img_auth.php/d/d0/CodeRefinery-project-plan-v1.0.pdf	
4	Mid-term report: http://cicero.xyz/v2/remark/github/coderefinery/report/master/mid-term.md/	
5	Phase-2 proposal: http://cicero.xyz/v2/remark/github/coderefinery/report/master/phase-2-proposal.md/	
6	Phase-2 project directive: https://wiki.neic.no/w/ext/img_auth.php/4/46/Coderefinery-project-directive-phase-2.pdf	
7	Presentations about the project: https://coderefinery.org/outreach/#presentations	
8	Articles and blog posts about the project: https://coderefinery.org/outreach/#articles-about-coderefinery	

Annex 1 – PPS Terminology

Decision points

During the lifespan of the project from startup to termination, a number of formal decisions must be made by the steering group. These fall into eight different types; which are numbered in the chronological order in which they are typically made.

DP1 – Decision point type 1; steering group decision to start the project, based on the project directive.

DP2 – Decision point type 2; steering group decision to continue, change or interrupt the project based on findings during the preparation phase. A project may have multiple DP2.

DP3 – Decision point type 3; steering group decision to approve the project plan developed during the preparation phase. Typically this is tied to a DP4 decision to start the execution phase.

DP4 – Decision point type 4; steering group decision to start the execution phase.

DP5 – Decision point type 5; steering group decision to continue, change or interrupt the project based on findings during the execution phase. A project may have multiple DP5.

DP6 – Decision point type 6; steering group decision to approve the result of a delivery, for example to end users. A project may have multiple DP6.

DP7 – Decision point type 7; steering group decision to transfer the responsibility for a delivery, typically to operations in a receiving organization.

DP8 – Decision point type 8; steering group decision to approve the final report and terminate the project.

Annex 2 – Glossary

Abbreviations

Term	Meaning	Comment/Link