Facilitating Collaboration in Machine Learning and High-Performance Computing Projects with an Interaction Room

Matthias Book*, Morris Riedel†, Helmut Neukirchen‡, Ernir Erlingsson§
School of Engineering and Natural Sciences, University of Iceland
†Double affiliation with Jülich Supercomputing Centre, Germany
Email: *book@hi.is, †morris@hi.is, ‡helmut@hi.is, §ere29@hi.is

***Abstract*—The design, development, and deployment of scientific computing applications can be quite complex, in particular when involving Machine Learning (ML) or High-Performance Computing (HPC). They require scientific and software engineering expertise and in addition HPC or ML knowledge. Often, such applications are however developed by scientists who are experts in their domain, but need support for the software engineering, ML, and HPC aspects. The cooperation and communication between experts from these quite different disciplines can be difficult though. We therefore propose to employ the Interaction Room (IR), a method that facilitates interdisciplinary collaboration in complex software projects. An IR uses annotated drawings to exchange information and stimulate discussion between project stakeholders, in order to improve common understanding and identify uncertainties, risks, and other aspects that are critical to a project's success early on. We suggest different drawing canvases and annotations that focus on different viewpoints and issues of the project. These canvases are specific to the project type, such as ML applications or HPC simulations.**

***Index Terms*—Collaboration, Interaction Room, Software Engineering, High-Performance Computing, Machine Learning**

I. Introduction

Developing scientific computing applications, in particular when they involve Machine Learning (ML) or High-Performance Computing (HPC), is challenging [1]: On the one hand, knowledge from an application domain is needed; on the other hand, ML or HPC – and ideally also software engineering – competence is required to write understandable, portable, verifiable and validatable, maintainable, extensible, efficient, and scalable code. Experience shows that scientists rarely master all these disciplines (application domain, ML/HPC, and software engineering). While scientific codes have a long lifetime, developer turnover tends to be high, particularly in academic environments, draining the project of domain knowledge, technical expertise, and awareness of implicit assumptions or limitations inherent in the code, thus threatening the sustainability of the scientific software [2].

When tackling a complex computational problem, domain scientists typically contact ML or HPC experts for assistance with ML models or to address issues concerning efficiency, scalability, or portability. At this point, the involved experts typically face the problem of understanding each others domains due to their very different backgrounds.

Recent research [3] supports our view that scientific computing software development is moving towards a broad collaborative (and even multi-institutional) approach.

In this paper, we suggest to adapt the *Interaction Room (IR)* method [4], [5] to facilitate , early in a project, the collaboration of experts from the scientific, the ML and/or HPC, and the software engineering domains, and thus design and implement ML or HPC applications more productively. An IR is a pragmatic, visual approach for establishing a common understanding of a project's fundamental requirements, a joint vision for its large-scale design, and an awareness of its most critical challenges.

Based on our experience with using the IR in enterprise projects [6], [7], and our observations of typical engineering struggles in ML and HPC simulation projects, we see additional potential in applying the IR method in such projects as well. Specifically, we needed to facilitate collaboration in the European Center of Excellence in Exascale Computing Research on AI- and Simulation-Based Engineering at Exascale (CoE RAISE).[1] The goal of CoE RAISE is to solve big engineering problems by using simulations and ML that will be able to exploit exascale HPC clusters.[2] In CoE RAISE, experts from various engineering projects work together with experts from the ML, HPC, and software engineering domains.

While we have already outlined in a position paper [9] how the IR method could be applied in pure simulation science HPC projects, we never had an opportunity to apply it in real projects, nor did we adapt it to ML problems. Therefore, we decided at the beginning of CoE RAISE to use the IR method

This project has received funding from the European Union's Horizon 2020 research and innovation programme H2020-INFRAEDI-2019-1 under grant agreement no. 951733 European Center of Excellence in Exascale Computing Research on AI- and Simulation-Based Engineering at Exascale (CoE RAISE).

[1] https://www.coe-raise.eu/

[2] Exascale computing [8] means to be able to scale computations so that the order of 10^{18} Floating Point Operations Per Second (FLOPS) is achieved. In 2022, the first exascale HPC clusters became operational in supercomputing centers: https://www.top500.org/lists/top500/2022/06/

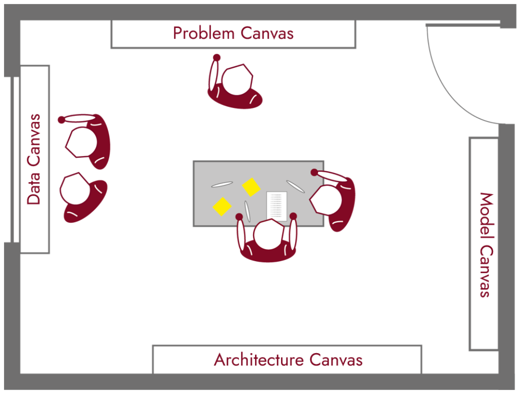

Fig. 1. Layout of an Interaction Room for ML projects

and investigate its applicability to ML and HPC projects. The main contribution of this paper is to present our adaptation of the IR methodology to ML and HPC projects and provide experience from applying it in real use cases from the CoE RAISE project.

In the following sections, we first introduce the IR methodology (Sect. II) and discuss how its elements can be applied to designing ML HPC solutions (Sect. III). We show how the method was used to facilitate communication between stakeholders in several use cases (Sect. IV). Finally, we summarize our gained experience with the method, and give an outlook on possible variations for other types of HPC projects, e.g. in pure simulation science projects (Sect. V).

II. Interaction Room Foundations

An Interaction Room (IR) [4], [5] is a room that is outfitted with several large analog or digital whiteboards (the so-called *canvases*) to visualize and facilitate discussion of key aspects of a software system (see Fig. 1): Each of the canvases is dedicated to representations of models of a particular aspect of the system, as described below. The key difference to other modeling techniques is that models in the IR are kept deliberately informal: The method's goal is not to create a detailed and complete specification, but to encourage stakeholders from diverse backgrounds to discuss those aspects that are essential to the project's success – the domain requirements, technical restrictions, aspects of particular value, and the most critical risks. They can be highlighted by attaching *annotations* to critical aspects (e.g., to pinpoint risks). The aim of the IR is not to make every stakeholder an expert in everyone else's domain, but to foster interdisciplinary discussion and understanding of these aspects already at an early project stage in order to prevent costly misunderstandings and oversights later on. However, the IR is not limited to the project start: the canvases can evolve throughout the project's lifecycle. This approach has already proven useful in several complex enterprise information system projects [6], [7] to create a better sense of joint project ownership among typically distinct stakeholder groups, and raise awareness of uncertainties and issues that were critical to the project's success.

III. The Interaction Room for Machine Learning (IR:ML)

The design of ML applications typically follows a path starting from understanding a particular scientific problem, via understanding the data and deciding on the needed ML models, to finally identifying suitable software libraries and hardware architectures (e.g., GPU clusters). This thought process poses different design and communication challenges than the design of information systems, and even within the HPC domain it differs between data science and simulation science projects. However, it does share the characteristic that stakeholders from diverse backgrounds need to work closely together. We therefore focus on adapting the IR method to data science ML applications in this paper, but give an outlook on a possible variation for simulation science projects in Sect. V.

A. IR:ML Canvases

We propose the following new IR canvases to support development for HPC-based ML applications:

Problem Canvas. The design of any scientific application starts with scoping the underlying scientific problem. Part of this initial project scoping is phrasing a precise research question, determining boundary conditions, clarifying assumptions and abstractions, and setting quality requirements such as accuracy or performance. It is also important at this stage to define how ML methods are expected to help solve the research question. All of these aspects are noted on the Problem Canvas as a reference for subsequent discussions.

Data Canvas. The focus then shifts to understanding the data that needs to be processed: What properties (e.g., in terms of data quality, but also data format, available metadata, volume, location, legal issues, owners) does the data have? How can the available data be accessed, and how can it be split into different datasets (e.g., for ML training, testing, and validation purposes)?

Model Canvas. The discussion next focuses on the ML models to use for the previously described data. (e.g., clustering, or classification using artificial neural networks or support vector machines, time series analysis, etc.). The Model Canvas invites stakeholders to consider opportunities for parameter optimization of the ML, adding neural architecture search to optimize the used neural network architecture, etc. Stakeholders also consider how to integrate HPC simulations with the ML models or to re-use models.

Architecture Canvas. Finally, stakeholders consider the technical implementation of the approach outlined on the preceding canvases: How can specific ML libraries (e.g., TensorFlow, pyTorch, Horovod) be used to implement ML models on specific HPC systems (e.g., CPU and GPU clusters)?

Ideally, these discussions would take place with the relevant stakeholders physically coming together in a room equipped with several large whiteboards, as shown in Fig. 1. This facilitates the most natural form of conversation, enabling stakeholders to collaboratively sketch and revise their plans, easily annotate it (see Sect. III-B), point to canvas contents, correlate content on adjacent canvases, and thus define and

Fig. 2. Annotations used in an IR:ML

discuss the problem, data, model, and architecture aspects while being visually and mentally immersed in them.

If such a physical IR setup is not feasible, e.g. for distributed teams, it is also possible to meet via videoconferencing and maintain the canvases in a shared digital whiteboard application. Working with the canvases live in a videoconferencing session, rather than filling them asynchronously as one might contribute to a shared text document, ensures that the canvases do not become mere information dumps but actual catalysts for discussion and mutual understanding between stakeholders with different backgrounds.

For similar reasons, it is important that the digital whiteboard is not just controlled by one person sharing their screen while having exclusive edit rights. Rather, all participants should be able to edit the whiteboard contents simultaneously, and navigate the views of the different canvases individually, just as they would be able to in a physical room. In the absence of physical gestures such as pointing to whiteboard contents to illustrate arguments in a discussion, it is ideal if the digital whiteboarding tool enables all participants to see what everyone else is currently pointing to.

In the distributed setting of CoE RAISE, we have found that MURAL[3] boards work great for this purpose as they facilitate live collaborative editing of a large virtual whiteboard, offering enough space to accommodate multiple canvases; allowing participants to pan and zoom around this space individually to satisfy their particular information needs, and at the same time showing everyone's labeled mouse cursor, enabling participants to point things out to each other. This makes for a very intuitive experience that transports most (even though not all) aspects of the physical IR into a virtual space.

Sketching the models on these canvases will likely not be a sequential process – rather, the interdisciplinary discussion and clarification of the above aspects will lead to an iterative refinement of all the canvases and thus a better understanding of the problem and the solution by all stakeholders before the actual implementation begins, as detailed later in Sect. III-C.

B. IR:ML Annotations

The purpose of an Interaction Room is to facilitate mutual understanding of stakeholders with very different backgrounds, such as application-domain scientists, ML or HPC experts, and software engineers. Interdisciplinary communication is enabled in two ways: First, by maintaining a very low contribution barrier – anyone can informally sketch their ideas on the canvases without adherence to a particular notation's formal syntax (as pointed out in the previous section). And second, by highlighting aspects of the project that are particularly crucial to understand and get right in order to ensure the project's success.

[3]https://www.mural.co/

This need of guiding participants' attention and helping them focus on the project's most critical aspects is addressed in the IR by so-called *annotations* – small symbols (stickers in a physical IR or image icons that can be dragged-and-dropped or copied-and-pasted in a virtual IR) that can be affixed by any participant to any canvas content in order to highlight particular value, risk or effort drivers, or key conceptual elements.

Figure 2 shows the annotations used in the IR for ML. From left to right, they denote:

- **Value**: A particularly critical element of the system that is considered valuable and needs to be of high quality for the project to achieve its goals.
- **Users**: A user-facing or user-controlled part of the system where high usability is important to achieve efficient and correct results.
- **Innovation**: A technical or conceptual innovation introduced into the system, i.e. a way in which the system is different from established approaches and thus requires special understanding or consideration.
- **Performance**: A particularly performance-critical part of the system.
- **Timing**: A particularly time-critical part of the system (e.g. in terms of real-time requirements, synchronization, or deadlines).
- **Security**: A particularly security-critical part of the system.
- **Automation**: A part of the system that is not accessible to user intervention.
- **Policies**: A part of the system exposed to particular legal, regulatory, or organizational constraints that all stakeholders need to be aware of.
- **Complexity**: A particularly complex part of the system that requires special scientific or engineering expertise.
- **Legacy**: A component that cannot be modified or adapted but needs to be integrated into the system as it is.
- **Under construction**: A component whose interface or behavior is subject to change.
- **External interface**: An interface to a system or component outside this project's control.
- **Uncertainty**: An aspect of the system that is still subject to unresolved questions or issues, i.e. not completely understood or defined yet.
- **Importance**: An aspect of the system that is important for stakeholders to take special note of (this can be used if the more specific annotations do not capture the essence of a stakeholder's comment).
- **HPC cluster**: An HPC hardware requirement, e.g. CPU, GPU, or other accelerators, or a specific HPC cluster.
- **Container**: Software that runs in a container (using, e.g., Docker or Singularity/Apptainer).

- **Data storage**: Special considerations concerning data, such as storage aspects for big data (e.g. volume, I/O speed, parallel I/O).
- **Compute-intensive**: A component or task that is particularly compute-intensive.
- **Machine learning**: ML-related considerations, e.g., a specific type of neural network to be used.
- **Software library**: A software library, in particular related to ML, e.g. TensorFlow.

Annotations do not only help stakeholders from diverse backgrounds to understand the most critical aspects of a canvas, but they also encourage stakeholders to reflect on which aspects of a project or system are particularly critical, risky, complex, important, uncertain, etc., and thereby help to focus the design and development effort on those parts of the project where it is most urgently needed.

C. Interaction Room Workshop Format

Filling and annotating the canvases of an IR is not a singular event, but a process that typically spans multiple iterations and might accompany a considerable part of a project's life cycle. Especially in early iterations (or in Unified Process [10] terms, in the Inception and Elaboration phases), IR workshops can be the main crystallization point for project knowledge. In later iterations/phases, the IR predominantly serves as a visualization of the overall project's scope and challenges that provides orientation and context for more formal in-depth specifications, where necessary.

A typical IR workshop focuses on one "primary" and several "secondary" canvases, and involves stakeholders from all disciplines participating in the project. While some stakeholders might have more to say on some canvases than others, no stakeholder group has "ownership" of a particular canvas, which rather represents one perspective on the whole project that is owned by all stakeholders.

The workshop is moderated by a so-called IR coach who is familiar with the IR method and the technological foundations, but not necessarily the details of the application domain. The moderator's role is to facilitate discussion between the stakeholders by maintaining a suitable level of abstraction (clarifying the big picture for all involved, but not getting lost in real-time problem solving or detailed specification activities that would be more suitably addressed in dedicated meetings of the relevant stakeholders), and by ensuring that any identified critical project aspects are highlighted with annotations, so the respective project knowledge is not lost.

The discussion in an IR workshop typically focuses on that workshop's designated "primary" canvas (in ML projects, this should initially be the Problem Canvas), whose contents are created or refined by all stakeholders together. As the discussion will typically also touch upon aspects relevant to other canvases, any knowledge, uncertainties, risks, etc. that belong on a secondary canvas should be noted there and put into the context there. However, the discussion should remain centered on the current primary canvas.

Once all canvases are filled in this way, subsequent workshops might also consider all canvases at once, and focus on cross-cutting concerns such as optimizations, refactorings, etc.

In addition to affixing annotations to canvas contents spontaneously as they are identified, it can also be useful to have dedicated "annotation rounds" where the IR coach invites the stakeholders to consider a whole canvas and annotate any aspects that they deem particularly important, uncertain, risky, etc., in order to make such "gut feelings" more explicit, which would normally not be shared by stakeholders, and use the opportunity to validate them in the larger group. If an annotation is found to be valid, a brief explanation is typically recorded alongside the visual icon; otherwise, the team may agree to remove it again.

Over the course of the project, the IR canvases become a high-level documentation of all relevant requirements, design, and implementation decisions that the team has agreed on. While not as highly formalized and complete as a dedicated requirements or design document, the IR has the benefit of also explicitly containing the team's meta-knowledge of the project, i.e. the awareness of particular issues, complexity, or importance associated with a requirement or design decision that would remain hidden in a formal specification despite having significant bearing on the project's success.

IV. Case Studies

The European Center of Excellence in Exascale Computing Research on AI- and Simulation-Based Engineering at Exascale (CoE RAISE) comprises nine HPC projects involving ML and simulation that are used as case studies: AI for turbulent boundary layers, AI for wind farm layout optimization, AI for data-driven models in reacting flows, smart models for next-generation aircraft engine design, AI for wetting hydrodynamics, event reconstruction and classification at the CERN HL-LHC, seismic imaging with remote sensing for energy applications, defect-free metal additive manufacturing, and sound engineering.[4]. They all involved combining ML and simulations where ML is then implemented in a case study-specific fashion into the highly customized simulation workflow.

In all of these use cases, the IR:ML was used to bring together experts from different domains and to foster their collaboration in order to reach a common understanding and identify issues using the IR:ML annotations.

A virtual IR was used with videoconferencing for audio communication and video for non-verbal communication and shared MURAL boards. At the start of the first IR workshop in each use case, the IR:ML method was presented by one of the authors with experience in the IR and another one from the authors as expert in HPC and ML, both took the moderator role asking stimulating questions to the experts from the other domains. The MURAL boards allowed all participants to fill the canvases in parallel, while the moderators took care of guiding the focus between the different canvases.

[4]https://www.coe-raise.eu/use-cases

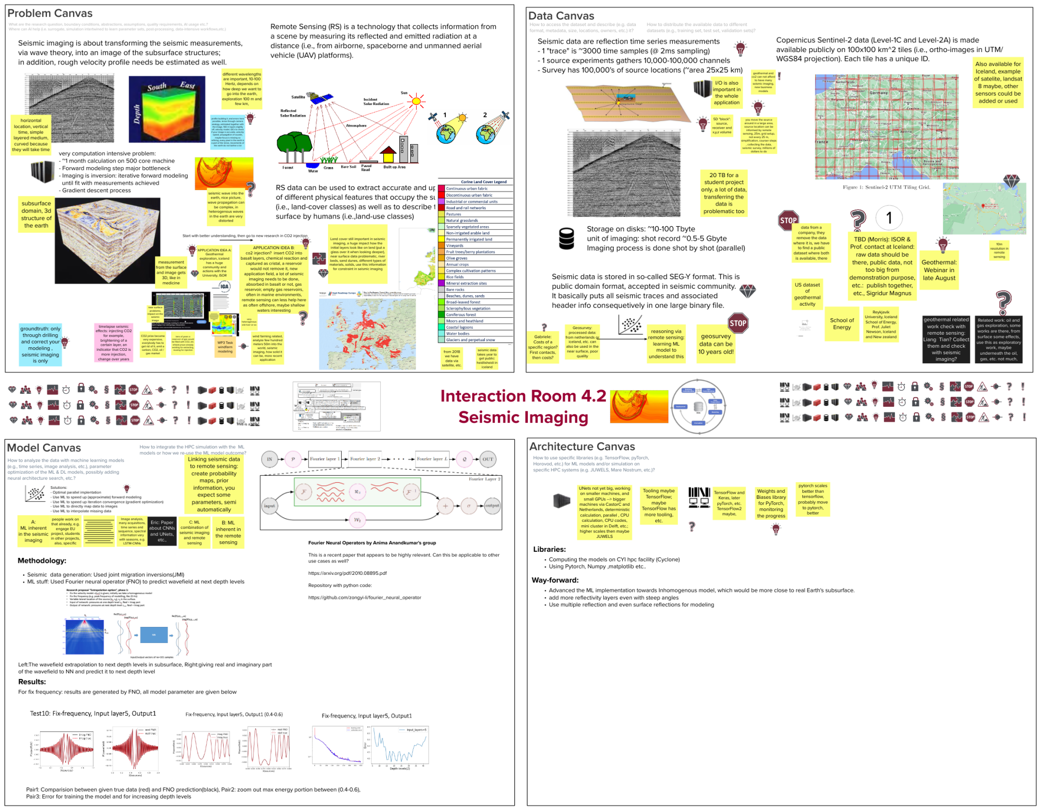

Fig. 3. Seismic Imaging use case: High-level view of all canvases in one MURAL board
Courtesy of Gabriele Cavallaro, Eric Verschuur, Nikos Savva, Jacob Finkenrath, and Naveed Akram

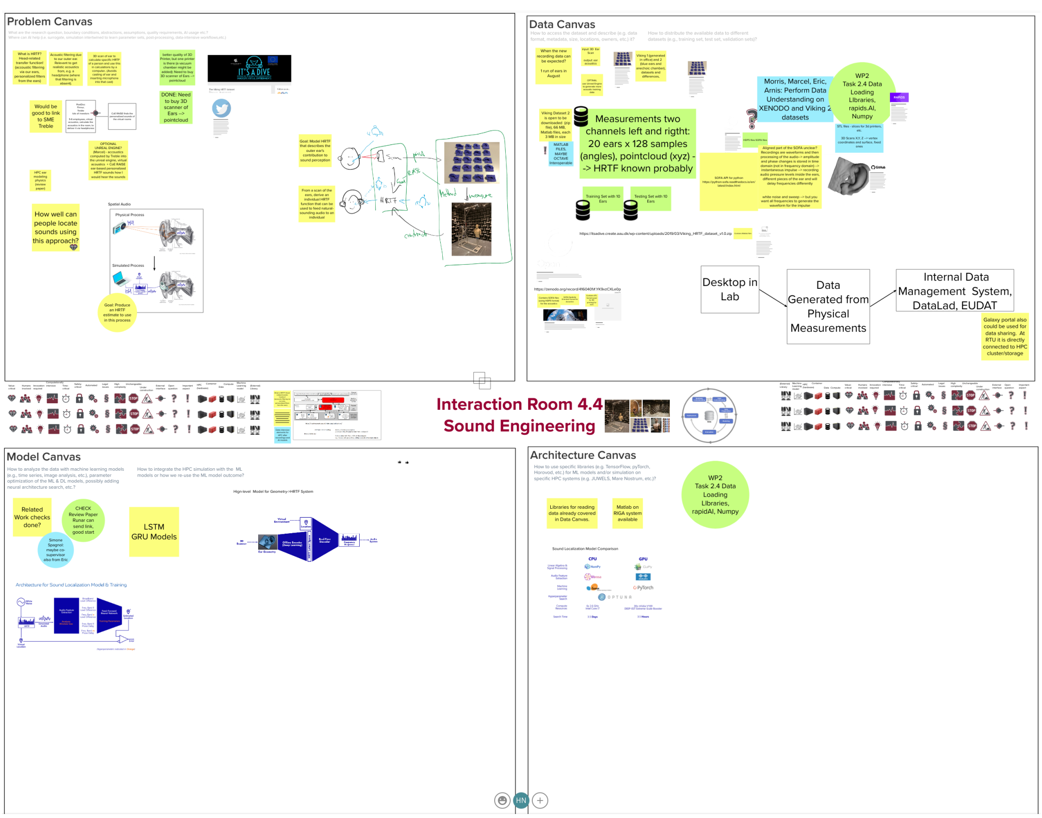

Fig. 4. Sound Engineering use case: High-level view of all canvases in one MURAL board
Courtesy of Rúnar Unnþórsson and Eric Michael Sumner

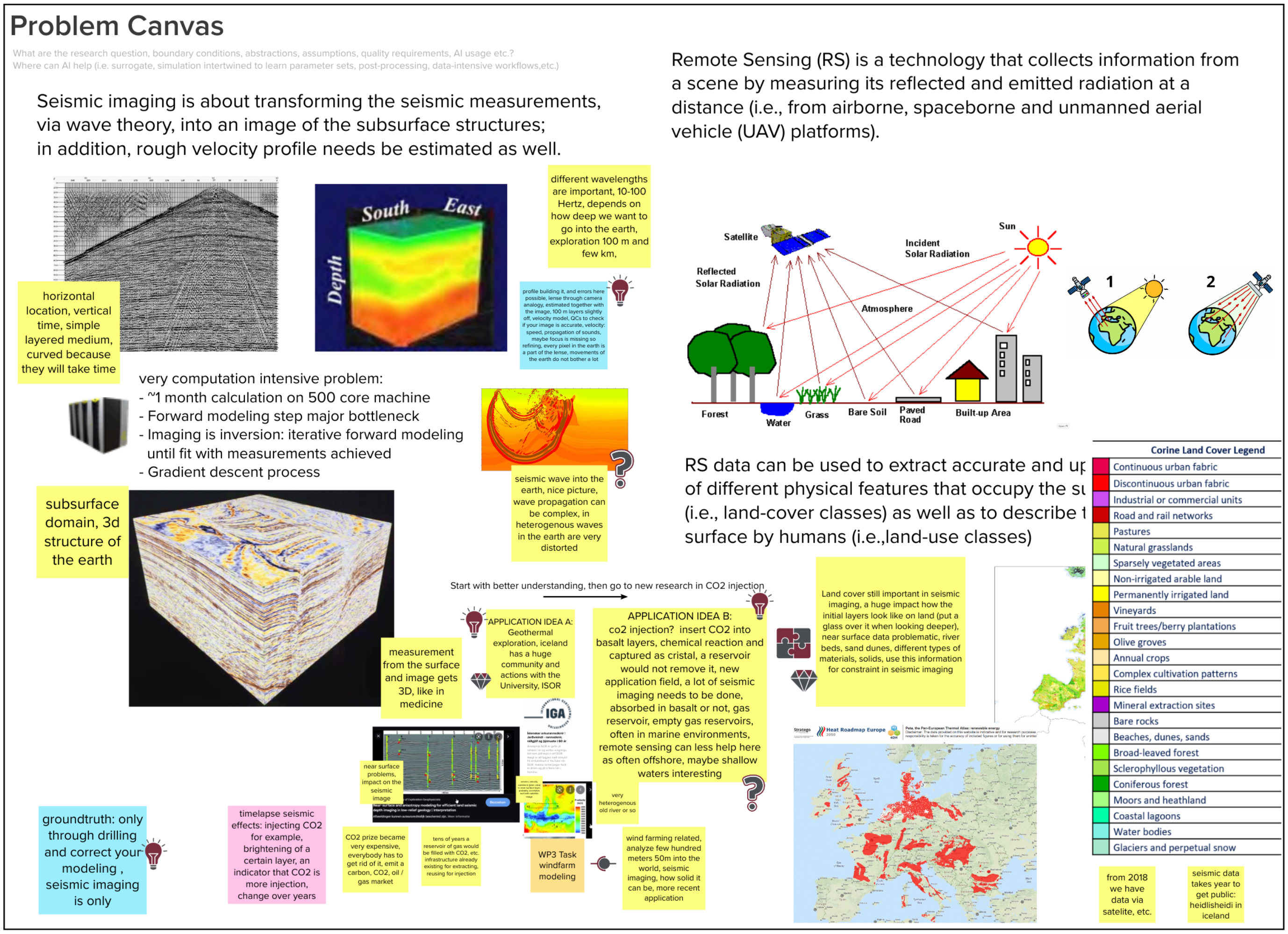

Fig. 5. Seismic Imaging use case: Problem Canvas

After the initial IR workshop, the extent of using the IR method varied in each use case, e.g. concerning the use of annotations, as the participants needed to get used to their meaning. Also, as the MURAL boards have no infinite canvas, but allow easy panning and zooming, the different teams made different use of available canvas space. Having a finite canvas helped to focus on the most important aspects.

As an example, an overview on the MURAL board with all four IR:ML canvases from the CoE RAISE use case "4.2 Seismic Imaging" is shown in Fig. 3, and from the use case "4.4 Sound Engineering" [11] in Fig. 4. The MURAL boards were subdivided into the four canvases described earlier, and multiple instances of the annotation icons had been placed between the canvases, so that they could be conveniently dragged-and-dropped. As an example of our experiences with the IR:ML method, we will report on the canvases created in the use case "4.2 Seismic Imaging" in more detail in the following subsections.

A. The Seismic Imaging Use Case

Seismic imaging ("geophysical tomography") uses intense acoustic sources at the surface of the Earth to transmit acoustic waves into the subsurface, where their reflected signals are recorded again with arrays of acoustic sensors at the Earth's surface [12]. As any tomography, this is a very computational intensive inverse problem based on wave theory that needs to be solved to determine reflection and refraction of the waves. The generated 2D and 3D subsurface models help to understand subsurface geology which is, e.g., used in volcanology, for finding oil and gas deposits, or for current climate research on CO_2 injection into basalt layers. This computational problem is solved by using iterative forward modeling until a fit with the measurements is achieved [13]. The idea in this use case is to use ML to speed-up convergence of this fit and to combine it with remote sensing [14] of the surface. Classification of remote sensing data (e.g., obtained from satellites [15] or drones) with respect to land cover is a classical ML application [16].

1) Problem Canvas: The IR workshop started with filling the Problem Canvas: the moderator asked questions to the domain experts who then started to describe the problems to be solved in this use case. The virtual shared whiteboard allowed the moderator to add information and annotations distilled from the ongoing discussion, while also enabling the other experts to add or revise information and annotations. This was

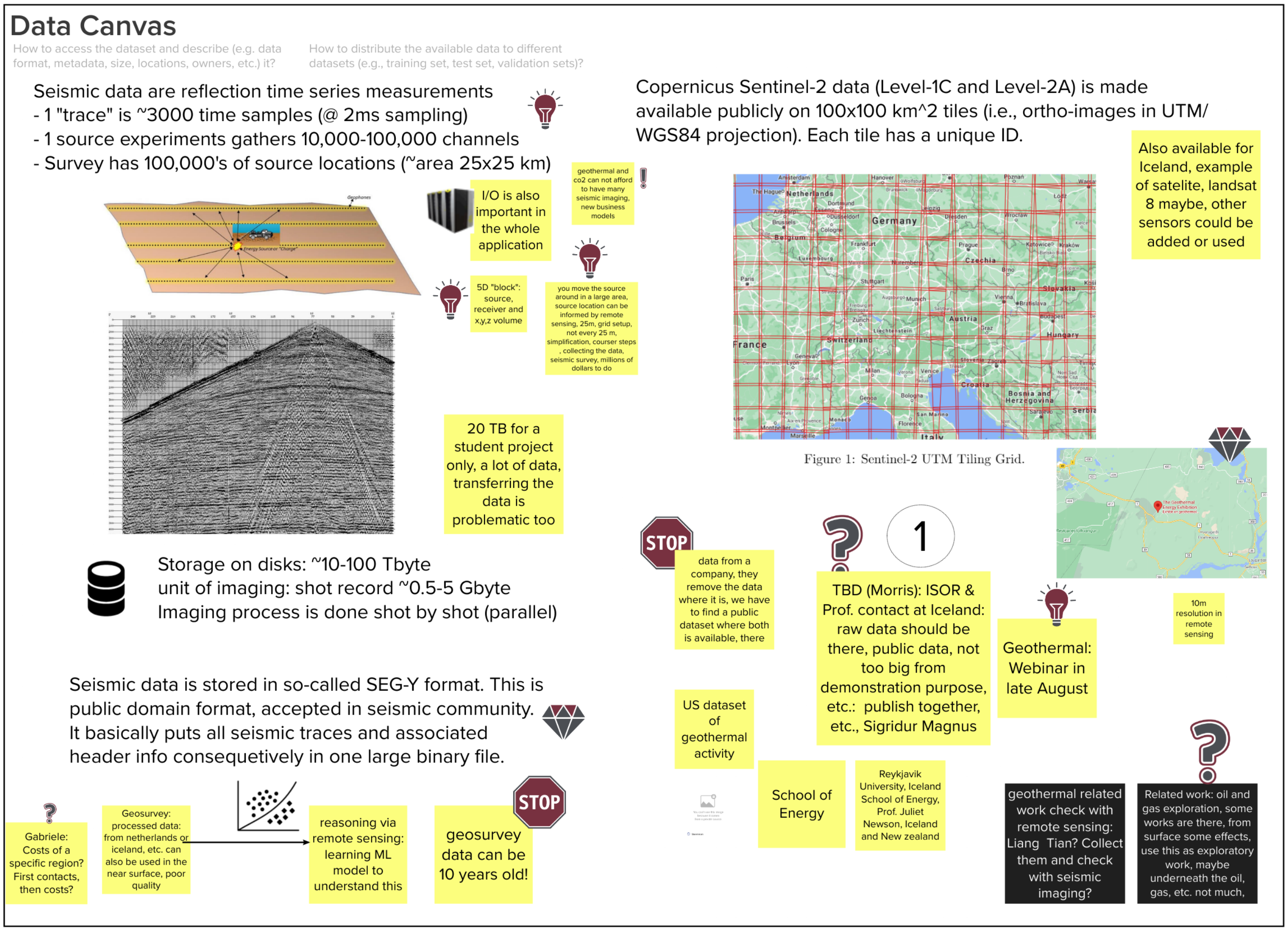

Fig. 6. Seismic Imaging case study: Data Canvas

an advantage of the virtual IR in comparison to a physical IR where only one person at a time can draw on a whiteboard.

The result of filling the Problem Canvas of the seismic imaging use case is depicted in Fig. 5. The left half is about seismic imaging, whereas the right half covers remote sensing.

Various annotations have been used, e.g. the Compute-Intensive Annotation (depicting an HPC cluster) has been added to the text explaining iterative forward modeling to solve the inverse problem. It can also be seen that the Value Annotation (diamond) and the Innovation Annotation (light bulb) were used together as innovation often creates value. A relationship to other work packages of the CoE RAISE project is marked via the External Interface Annotation.

As described in Section III-C, this was an iterative process and elements were added to the Problem Canvas as the result of a discussion in other canvases.

2) Data Canvas: Figure 6 shows the the Data Canvas that was created during the IR workshops. Again, the left half is about seismic imaging, whereas the right half deals with remote sensing.

As seismic imaging creates a lot of data, annotations for storage and HPC (because of I/O) have been added to this canvas. The Innovation Annotation (light bulb symbol) was also attached a couple of times, however, it is not always obvious how the annotated information refers to innovation. Open questions have been marked using the question mark symbol (Uncertainty Annotation) and data is annotated as valuable. The Legacy Annotation (stop sign) is attached to information concerning old data and data where some information has been removed by third parties.

3) Model Canvas: This canvas (Fig. 7) focuses on the ML models that might be suitable for the seismic imaging use case. It is remarkable that the project stakeholders did not include any content related to remote sensing on this canvas. The reason might be that using ML is already common in that field, and thus, not much needs to be discussed with respect to ML models. (Or it was simply forgotten to be added.)

The ideas for using ML for seismic imaging are annotated as innovative (light bulb symbol). No other annotations were used on this canvas.

4) Architecture Canvas: Finally, Fig. 8 shows the Architecture Canvas of the seismic imaging use case. Information concerning the HPC clusters to be used are marked using the HPC Cluster Annotation. The discussion concerning the

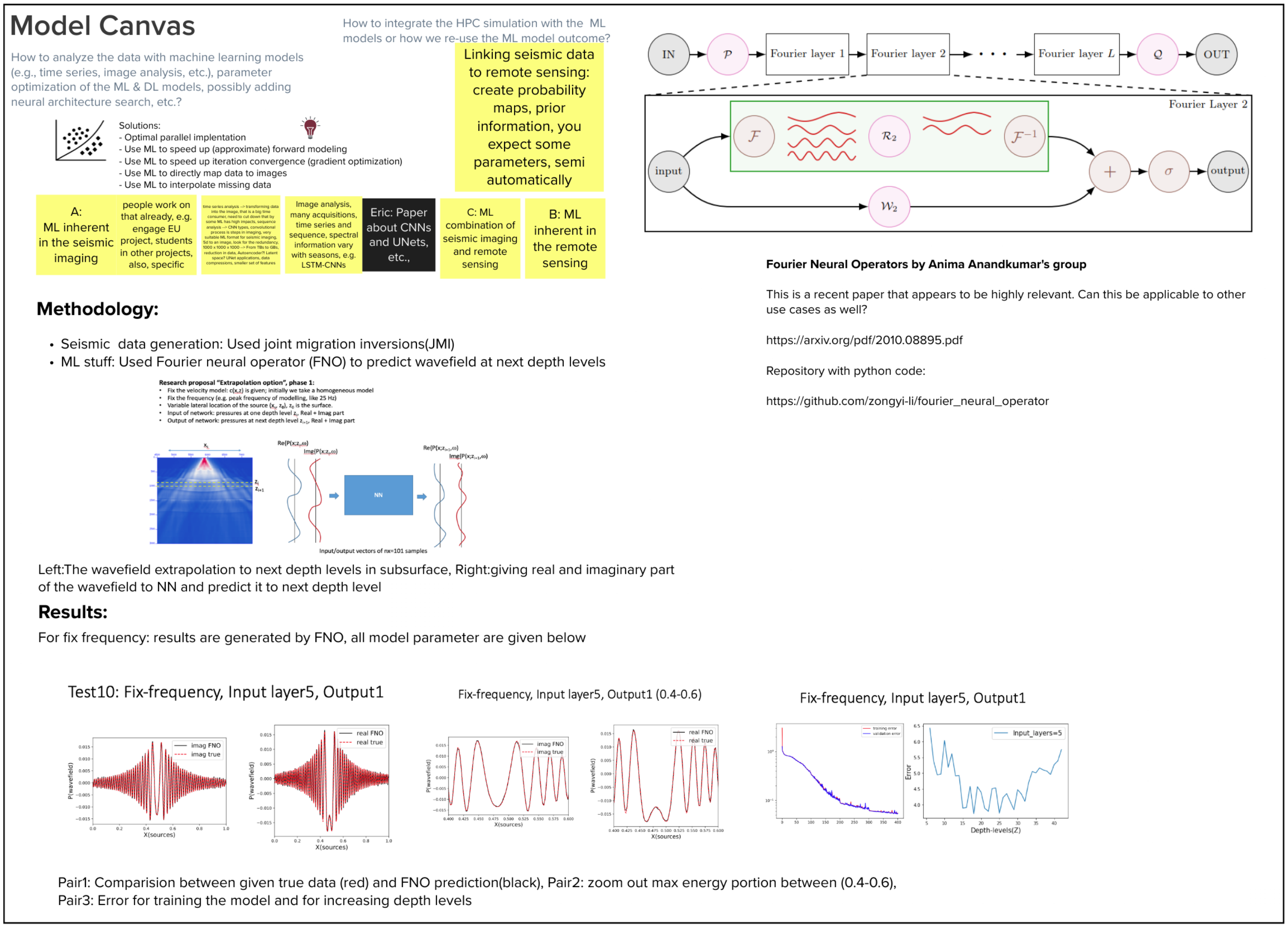

Fig. 7. Seismic Imaging use case: Model Canvas

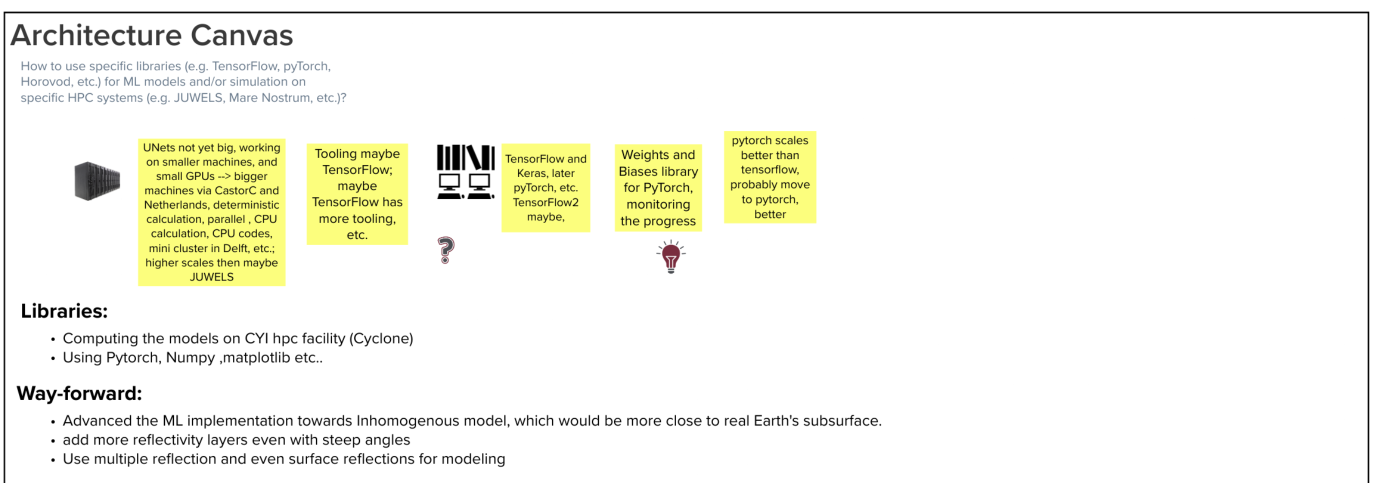

Fig. 8. Seismic Imaging use case: Architecture Canvas (empty bottom cropped)

ML frameworks and libraries to be used is marked with the Software Library Annotation, but as that discussion is not finished, the Uncertainty Annotation (question mark) has been attached as well. Monitoring the ML progress is considered innovative (light bulb symbol).

5) Discussion: In general, using the IR:ML benefited the CoE RAISE project in terms of fostering better communication of stakeholders, and more explicit externalization of discussion outcomes. Other projects where domain experts and ML experts had to work together without using an IR:ML typically just saw some unstructured discussions, resulting in the worst case even in an ultimate failure (for example, in another research project, a sub-task of introducing ML into a climate modelling code had to be cancelled after spending 4 PMs of a senior climate researcher and an experienced ML expert due to a lack of common understanding in addition to unrealistic expectations from project managers to that sub-task).

The extent of using annotations varied in the different CoE RAISE use cases (compare, e.g., Fig. 3 and Fig. 4). This suggests that the moderator should take care to encourage participants to add annotations spontaneously. For example, we did not explicitly use "annotation rounds" which would probably improve the usage of annotations.

Another explanation for varying usage of annotations could be that the meaning of the graphical annotations is not intuitive enough and therefore, participants did not remember the meaning and thus did not dare to attach annotations. For example, there are two symbols that resemble an HPC cluster (the Compute-Intensive Annotation and the HPC Cluster Annotation) that sometimes seem to have been used interchangeably. As this was the first time we applied this set of annotations, we will use observations from their usage in CoE RAISE to further tune the annotation set to the needs of HPC/ML project stakeholders. Also, while we did provide the annotations for easy drag-and-drop in our canvas templates (see the center left and center right in figures 3 and 4), we did not add text descriptions there. We hypothesize that adding text descriptions would already help with annotation uptake and proper use.

The fact that annotations were not used a lot in the Model Canvas suggests that the current set of annotations is not suitable for ML models. Therefore, further ML-related annotations need to be investigated. (The set of annotations was created by an HPC and ML expert together with an IR expert based on their experience, but without, e.g., formal evaluation of issues needing attention in earlier ML projects)

A further observation is that the Architecture Canvas is typically not as crowded as the other canvases. This might either mean that architecture aspects are straightforward (still, it was filled with useful information, so we do not recommend to abandon it – in particular, if a more heterogeneous architecture is used) or that more guidance is needed. Observing the further progress of the CoE RAISE use cases, and noting any architectural issues that might come up later, could be helpful in identifying aspects that require more guidance, e.g. by additional annotations.

Having a good moderator (preferably an IR coach who is experienced and familiar with the IR method, see Sect. III-C) has proven to be necessary to ensure the productivity of IR workshops, at least in early project phases. After our first moderated IR:ML workshops, we let the CoE RAISE teams collaborate further on the canvases without an IR coach, but this turned out less effective (in terms of surfacing new insights) than the moderated IR workshops.

V. Outlook: The Interaction Room for HPC in Simulation Science (IR:Sim)

In the preceding sections, we have described an IR format that is tailored to the needs of ML projects where research questions are answered by processing large datasets using ML techniques. However, HPC can also be applied to research questions that can best be resolved through complex simulations. In such simulation-science projects, the challenge lies in finding efficient ways for mapping a real-world problem to a mathematical model, and to map that model onto a supercomputing infrastructure. An IR can be useful to facilitate collaboration between the scientific and technical stakeholders of simulation projects as well, but we suggest focusing the IR:Sim canvases on a different set of perspectives [9]:

Problem Canvas. As for ML projects, the design of simulation projects starts with scoping the underlying scientific problem, e.g., forecasting the weather. Part of this initial project scoping is phrasing a precise research question, determining boundary conditions, clarifying assumptions and abstractions, and setting quality requirements such as accuracy or performance. All of these are noted on the Problem Canvas as a reference for subsequent discussions.

Real-World Canvas. The next step towards building a simulation solution that answers the defined research question is to understand the underlying real world processes, e.g. the physics or chemistry governing the weather. On the Real-World Canvas, domain experts conceptualize for themselves and for the technical experts which scientific processes exactly are relevant for their research question, what elements are active or passive components of the simulation, how their interplay is described by natural laws and formulae, etc.

Decomposition Canvas. Based on the understanding of the real-world structures, the HPC experts are next tasked with breaking the continuous world of the Real-World Canvas down into the components of a discrete simulation: Together with the domain experts, they identify suitable approximations for the formulae, and decompose the real world into chunks that are suitable for parallel simulation. This entails identifying necessary exchange of information between the chunks, adaptive refinement of the decomposition, etc.

Architecture Canvas. The final development step is the implementation and deployment of the simulation conceived in the previous step on a concrete HPC cluster. On the Architecture Canvas, HPC experts can visualize and discuss suitable communication strategies, necessary interconnect properties, efficient memory models, data storage, etc.

While we had no opportunity to conduct IR:Sim workshops yet, we hypothesize that they can proceed in similar fashion and yield similar benefits as the IR:ML workshops that we conducted for the use cases of CoE RAISE.

Gaining experience with the most suitable adaptations of the general IR format for different types of scientific computing projects, and tailoring the set of canvases and annotations for each, will be the primary focus of our ongoing research.

VI. Summary and Conclusions

We have presented an adaption of the Interaction Room (IR) method that is tailored to the challenges encountered in ML HPC projects by defining a set of canvases for visualizing the problem domain, datasets, ML modeling, and implementation of the solution on suitable HPC systems, as well as a set of annotations that are useful for highlighting typical critical aspects of such projects.

The canvases of the IR:ML help the stakeholders already early in the project to jointly discuss and visualize aspects of the HPC system that might otherwise not be discussed explicitly, because domain and technology experts may assume they are generally known, or because they do not realize these aspects need to be specified early in order to ensure the project is going in the right direction. Since the IR does not enforce any strict modeling syntax, it is not a specification tool but rather a catalyst for the interdisciplinary discussion between stakeholders from different backgrounds who can jointly identify the aspects of the project that are most valuable, most complex or risky, least understood, and thus most critical for the project's success.

From a software engineer's perspective, an IR is a pragmatic approach to capturing and organizing the knowledge elicited in the Inception and Elaboration phases of the Unified Process (UP). From a data scientist's perspective, the IR canvases can serve as a simple approach to record and discuss the findings of the phases in the CRoss Industry Standard Process for Data Mining (CRISP-DM) [17]. In either process model, the IR has the advantage of being a lightweight, informal documentation technique that emphasizes ease of interdisciplinary understanding over the resulting artefacts' formal correctness and completeness. As such, it can help teams to follow the UP or CRISP-DM models that would otherwise shy away from those process models' documentation needs.

While the IR canvases are obviously no substitute for the more formal, detailed and complete artefacts proposed by, e.g, the UP, they can serve for the stakeholders' orientation and highlight those aspects of the projects where the effort for creating a fuller specification is most needed. As such, it can help teams to spend their resources more efficiently.

Software sustainability [2], can be improved by archiving the canvasses that evolved throughout the project, as these capture knowledge and assumptions that is typically not documented anywhere else.

In our ongoing work, we are gaining further experience with the IR:ML in order to identify the most suitable informal notations to express and connect concepts on the different canvases, and evaluating the application of the approach in HPC engineering practice.

Acknowledgment

The authors like to thank Rúnar Unnþórsson, Eric Michael Sumner, Gabriele Cavallaro, Eric Verschuur, Nikos Savva, Jacob Finkenrath, and Naveed Akram for allowing us to reproduce the Interaction Room canvases of their projects.

References

[1] J. C. Carver, L. Hochstein, R. P. Kendall, T. Nakamura, M. V. Zelkowitz, V. R. Basili, and D. E. Post, "Observations about software development for high end computing," *CTWatch Quarterly*, vol. 2, no. 4A, pp. 33–37, 2006.

[2] T. Kehrer and B. Penzenstadler, "An exploration of sustainability thinking in research software engineering," in *Requirements Engineering for Sustainable Systems (RE4SuSy) 2018*, ser. CEUR Workshop Proceedings, vol. 2223. CEUR-WS.org, 2018.

[3] D. E. Bernholdt, M. Doucet, W. F. Godoy, A. Malviya-Thakur, and G. R. Watson, "A survey on sustainable software ecosystems to support experimental and observational science at Oak Ridge National Laboratory," in *Computational Science – ICCS 2022*, ser. LNCS, vol. 13353. Springer, 2022.

[4] M. Book, S. Grapenthin, and V. Gruhn, "Seeing the forest and the trees: focusing team interaction on value and effort drivers," in *Proc ACM SIGSOFT 20th Intl Symp on Foundations of Software Engineering (FSE'12)*, no. 30. ACM, 2012.

[5] ——, "Highlighting value and effort drivers early in business and system models," in *Proc 13th Intl Conf on Intelligent Software Methodologies, Tools and Techniques (SOMET'14)*. IOS Press, 2014, pp. 530–544.

[6] S. Grapenthin, M. Book, V. Gruhn, C. Schneider, and K. Völker, "Reducing complexity using an interaction room: An experience report," in *Proc 31st ACM Intl Conf on Design of Communication (SIGDOC'13)*. ACM, 2013, pp. 71–76.

[7] S. Grapenthin, S. Poggel, M. Book, and V. Gruhn, "Improving task breakdown comprehensiveness in agile projects with an interaction room," *Information and Software Technology*, vol. 67, pp. 254–264, 2015.

[8] J. Dongarra, S. Gottlieb, and W. T. C. Kramer, "Race to exascale," *Computing in Science & Engineering*, vol. 21, no. 1, pp. 4–5, 2019.

[9] M. Book, M. Riedel, H. Neukirchen, and M. Götz, "Facilitating collaboration in high-performance computing projects with an interaction room," in *Proceedings of the 4th ACM SIGPLAN International Workshop on Software Engineering for Parallel Systems*. New York, NY, USA: Association for Computing Machinery, 2017, p. 46–47. [Online]. Available: https://doi.org/10.1145/3141865.3142467

[10] I. Jacobson, G. Booch, and J. Rumbaugh, "The unified process," *IEEE Software*, vol. 16, no. 3, pp. 96–102, 1999.

[11] E. M. Sumner, M. Aach, A. Lintermann, R. Unnthorsson, and M. Riedel, "Speed-up of machine learning for sound localization via high-performance computing," in *2022 26th International Conference on Information Technology (IT)*. IEEE, 2022.

[12] T. H. Keho and P. G. Kelamis, "Focus on land seismic technology: The near-surface challenge," *The Leading Edge*, vol. 31, no. 1, pp. 62–68, 2012.

[13] D. Verschuur, X. Staal, and A. Berkhout, "Joint migration inversion: Simultaneous determination of velocity fields and depth images using all orders of scattering," *The Leading Edge*, vol. 35, no. 12, pp. 1037–1046, 2016.

[14] M. J. Canty, *Image analysis, classification and change detection in remote sensing: with algorithms for ENVI/IDL and Python*. Crc Press, 2014.

[15] J. Aschbacher, "Esa's earth observation strategy and copernicus," in *Satellite earth observations and their impact on society and policy*. Springer, Singapore, 2017, pp. 81–86.

[16] C. Paris, L. Bruzzone, and D. Fernández-Prieto, "A novel approach to the unsupervised update of land-cover maps by classification of time series of multispectral images," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 57, no. 7, pp. 4259–4277, 2019.

[17] C. Shearer, "The CRISP-DM model: the new blueprint for data mining," *Journal of data warehousing*, vol. 5, no. 4, pp. 13–22, 2000.