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

Matthias Book
University of Iceland
Reykjavik, Iceland
book@hi.is

Morris Riedel*
Jülich Supercomputing Centre
Jülich, Germany
m.riedel@fz-juelich.de

Helmut Neukirchen
University of Iceland
Reykjavik, Iceland
helmut@hi.is

Markus Götz*
Jülich Supercomputing Centre
Jülich, Germany
m.goetz@fz-juelich.de

Abstract

The design, development and deployment of scientific computing applications can be quite complex as they require scientific, High-Performance Computing (HPC), and software engineering expertise. Often, HPC applications are however developed by end users who are experts in their scientific domain, but need support from a supercomputing centre for the engineering and optimization aspects. The cooperation and communication between experts from these quite different disciplines can be difficult though. We therefore propose to employ the Interaction Room, a technique that facilitates interdisciplinary collaboration in complex software projects.

*Double affiliation with University of Iceland, Reykjavik, Iceland.

1 Introduction

Developing High-Performance Computing (HPC) code is challenging [2]: On the one hand, knowledge from an application domain is needed; on the other hand, HPC—and ideally also software engineering—competence is required to write understandable, portable, verifiable and validatable, maintainable and extensible codes that are both efficient and scalable. Experience shows that scientists rarely master all three disciplines (application domain, HPC, and software engineering). While scientific codes have a long lifetime, developers are likely to change in academic environments, draining the project of implicit assumptions and knowledge.

When tackling a complex computational problem, domain scientists therefore typically contact HPC experts at a supercomputing centre to get help with issues concerning efficiency, scalability, or portability. At this point, the involved experts typically face the problem of understanding each other's domains, due to their very different backgrounds. In this paper, we suggest to adapt the *Interaction Room* method [1] to facilitate the collaboration of experts from the scientific, HPC and software engineering domains, and thus design and implement HPC applications more productively.

2 The Interaction Room for HPC

An Interaction Room 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 models of a particular aspect of the system, as described below. The key difference to other modeling techniques is that models in the Interaction Room are kept deliberately informal: Its goal is not to create a perfect specification, but to encourage stakeholders from diverse backgrounds to discuss those aspects that are essential to the project's success – understanding the domain requirements, understanding technical restrictions, identifying aspects of particular value, and identifying the most critical risks. Fostering interdisciplinary discussion and understanding of these aspects at an
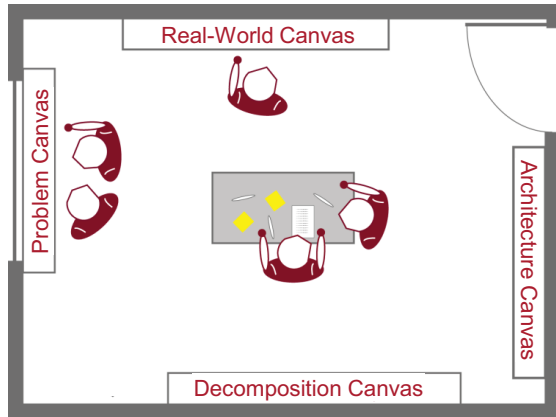
Figure 1. Layout of an Interaction Room for HPC projects

early project stage helps to prevent costly misunderstandings and oversights later on, and has already proven useful in numerous complex information systems projects [3].

The design of HPC systems typically follows a path from understanding a particular scientific problem to phrasing it in a parallel algorithm to implementing it on an HPC cluster.[1] This thought process poses quite different design and communication challenges than the design of information systems. We therefore propose the following new Interaction Room canvases for HPC systems development:

Problem canvas. The design of any HPC system 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 an HPC system 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.

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 coding begins.

3 Summary and Conclusions

The canvases of the Interaction Room force the stakeholders 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 Interaction Room 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 projects that are most valuable, most complex or risky, least understood, and thus most critical for the project's success.

Based on our industrial experience from applying Interaction Rooms in large enterprise information systems projects, we hypothesize that these insights can help the team to focus their subsequent specification, development and performance tuning effort better and thus develop high-quality HPC applications more efficiently. In our ongoing work, we are further refining the method, examining the most suitable notations for the canvases, and evaluating the application of Interaction Rooms in HPC engineering practice.

Acknowledgments

References

[1] M Book, S Grapenthin, and V Gruhn. 2012. 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)*. ACM.

[2] J C Carver, L Hochstein, R P Kendall, T Nakamura, M V Zelkowitz, V R Basili, and D E Post. 2006. Observations about software development for high end computing. *CTWatch Quarterly* 2, 4A (2006), 33–37.

[3] S Grapenthin, S Poggel, M Book, and V Gruhn. 2015. Improving task breakdown comprehensiveness in agile projects with an Interaction Room. *Information and Software Technology* 67 (2015), 254–264.

---

[1]We restrict our discussion to supporting the design of HPC applications in simulation science at this point. Adapting the concept for data science applications will be a topic of future research.