Chapter 80
The BioDynaMo Project:
Experience Report

Roman Bauer
Newcastle University, UK

Lukas Breitwieser
CERN openlab, Switzerland

Alberto Di Meglio
CERN openlab, Switzerland

Leonard Johard
Innopolis University, Russia

Marcus Kaiser
Newcastle University, UK

Marco Manca
CERN openlab, Switzerland

Manuel Mazzara
Service Science and Engineering Lab, Innopolis University, Russia

Fons Rademakers
CERN openlab, Switzerland

Max Talanov
Kazan Federal University, Russia

Alexander Dmitrievich Tchitchigin
Innopolis University, Russia

ABSTRACT

Computer simulations have become a very powerful tool for scientific research. Given the vast complexity that comes with many open scientific questions, a purely analytical or experimental approach is often not viable. For example, biological systems comprise an extremely complex organization and heterogeneous interactions across different spatial and temporal scales. In order to facilitate research on such problems, the BioDynaMo project aims at a general platform for computer simulations for biological research. Since scientific investigations require extensive computer resources, this platform should be executable on hybrid cloud computing systems, allowing for the efficient use of state-of-the-art computing technology. This chapter describes challenges during the early stages of the software development process. In particular, we describe issues regarding the implementation and the highly interdisciplinary as well as international nature of the collaboration. Moreover, we explain the methodologies, the approach, and the lessons learned by the team during these first stages.

INTRODUCTION

Most laboratories in computational biology develop their own custom software to carry out a specific simulation. These applications are monolithic, difficult to extend, and usually do not scale. Consequently, a lot of resources are spent in developing functionality that has already been created elsewhere. The BioDynaMo project has been started to close the gap between very specialized applications and highly scalable systems to give life scientists access to the rapidly growing computational resources.

Our project started as a code modernization initiative based on the simulation software for neuronal development Cx3D (Zubler & Rodney, 2009). Cx3D is a tool that can generate sophisticated structures based on simple rules defined by the computational scientist. Although Cx3D has a very compact code base (15 kLOC), it can perform complex simulations like "cortical lamination". However, the absence of modern software development practices such as automated tests and thus continuous integration, coding standards, and code reviews prohibits a sustainable development process.

The multidisciplinary nature of this initiative requires expertise from different backgrounds. In comparison with other simulation packages, this poses additional challenges in managing diverse collaborators, aligning project members, and defining unique selling propositions.

Aiming at a platform that will be used by a great number of researchers simulating large scale systems, software errors will have a huge impact. Therefore, we give an overview about software verification, different approaches, and why it has great significance for this project.

CODE MODERNIZATION

High performance and high scalability are the prerequisites to address ambitious research questions like modeling epilepsy. Our efforts in code modernization were driven by the goal to remove unnecessary overhead and update the software design to tap the unused potential caused by the paradigm shift to multi- and many-core systems. Prior to 2004, since performance was clock speed driven, buying a processor of the next generation automatically increased application performance without changing a single line of code. Physical limitations forced the CPU vendors to change their strategy to push the edge of the performance envelope (Sutter, 2005). Although this paradigm shift helped to improve theoretical throughput, sequential applications benefit only marginally from new processor generations. This puts additional burden on the application developers who have to refactor existing applications and deal with the increased complexity of parallel programs. As core counts are constantly increasing, the portion of unused computing resources will grow in the future if these changes are not applied. Furthermore, a modern processor offers multiple levels of parallelism that goes beyond the number of cores: it features multiple threads per core, has multiple execution ports able to execute more than one instruction per clock cycle, is pipelined and provides vector instructions also referred to as SIMD (Single Instruction Multiple Data).

Last year's Intel Modern Code Development Challenge was about optimizing sequential C++ brain simulation code provided by the Newcastle University in the UK. The task for participating students was to improve the run-time as much as possible. Using data layout transformations (array of structures AoS to structure of arrays SoA), parallelization with OpenMP, a custom memory allocator and Intel Cilk Plus array notation, the winner was able to improve the run-time by a factor of 320. This reduced the initial execution time on a Intel Xeon Phi Knights Corner Coprocessor with 60 cores from 45 hours

Related Content

Soil Cation Exchange Capacity Predicted by Learning From Multiple Modelling: Forming Multiple Models Run by SVM to Learn From ANN and Its Hybrid With Firefly Algorithm

Rahman Khatibi, Mohammad Ali Ghorbani, Rasoul Janiand Moslem Servati (2018). *Handbook of Research on Predictive Modeling and Optimization Methods in Science and Engineering (pp. 465-480).*

www.irma-international.org/chapter/soil-cation-exchange-capacity-predicted-by-learning-from-multiple-modelling/206762

Sentiment Analysis: Using Artificial Neural Fuzzy Inference System

Syed Muzamil Bashaand Dharmendra Singh Rajput (2018). *Handbook of Research on Pattern Engineering System Development for Big Data Analytics (pp. 130-152).*

www.irma-international.org/chapter/sentiment-analysis/202838

Fractal Coding Based Video Compression Using Weighted Finite Automata

Shailesh D. Kamble, Nileshsingh V. Thakurand Preeti R. Bajaj (2021). *Research Anthology on Recent Trends, Tools, and Implications of Computer Programming (pp. 232-252).*

www.irma-international.org/chapter/fractal-coding-based-video-compression-using-weighted-finite-automata/261029

Arts and Branches of Science Significantly Contributing to Cyber and Cyber Security: The West European and the Russian Views

Margarita Levin Jaitnerand Áine MacDermott (2018). *Cyber Security and Threats: Concepts, Methodologies, Tools, and Applications (pp. 977-994).*

www.irma-international.org/chapter/arts-and-branches-of-science-significantly-contributing-to-cyber-and-cyber-security/203544

Lattice Boltzmann Shallow Water Simulation With Surface Pressure

Iñaki Zabalaand Jesús M. Blanco (2018). *Analysis and Applications of Lattice Boltzmann Simulations (pp. 293-336).*

www.irma-international.org/chapter/lattice-boltzmann-shallow-water-simulation-with-surface-pressure/203093