

Web Platform for Sharing Modeling Software in the Field of Nonlinear Optics

Julia Dubenskaya^{1,*}, Alexander Kryukov¹, and Andrey Demichev¹

¹*Skobeltsyn Institute of Nuclear Physics, Lomonosov Moscow State University, Leninskie gory, GSP-1, Moscow 119991, Russian Federation*

Abstract.

We describe the prototype of a Web platform intended for sharing software programs for computer modeling in the rapidly developing field of the nonlinear optics phenomena. The suggested platform is built on the top of the HUBZero open-source middleware. In addition to the basic HUBZero installation we added to our platform the capability to run Docker containers via an external application server and to send calculation programs to those containers for execution. The presented web platform provides a wide range of features and might be of benefit to nonlinear optics researchers.

1 Introduction

Nonlinear optics is a rapidly evolving area of modern physical research and engineering with many important applications. Successful development of optical devices requires complicated modeling of physical processes that occur in components of devices. Many developers of application software have the intention to share their products among other researchers.

At the present time dedicated web resources are developed and widely used in the research community all over the world. There are a number of web resources that provide software on nonlinear optics [1–3]. Most of them are intended for commercial distribution of desktop applications. These applications need to be downloaded to the local computer, and then could be launched offline.

The architecture of an online web platform for nonlinear optics was proposed in [4, 5]. A distinguishing feature of the platform in question is that it provides web access to the already installed and configured software. A web platform of that kind will be called “a closed web platform”. It is easily manageable, but sometimes its fixed functionality is too limited. By contrast, a platform where users can install and share their own programs will be called “an open web platform”. In our studies we came across that there is a need for such a platform for nonlinear optics.

We developed a web platform for sharing users’ software in the field of nonlinear optics using HUBzero middleware [6] as a basis. The HUBzero middleware provides services for application software installation and includes a set of tools for simplifying the interaction between software developers, regular users and resource administrators. In particular, the HUBzero middleware comprises tools for transforming the application software into software as a service (SaaS).

*e-mail: jdubenskaya@gmail.com

The basic HUBzero installation provides a working web server and a local computational resource. By further customizing it is possible to set up access to GRID resources [7]. We added to our HUBzero installation the capability to execute applications in a specially prepared Docker container [8] which is launched on the external application server. From a technical perspective our platform is a single frontend for remote hardware resources where special middleware services are installed and run.

Thus, the presented open web platform is a user-friendly product with extended functionality that combines the power of the HUBzero middleware with the flexibility of the external Docker containers.

In the section 2, we will take a closer look at the architecture of the web platform. In the section 3, the operation of the web platform will be shown on a concrete example. In the Conclusion, we will summarize the accomplished work, and discuss practical applications and future improvements of the presented platform.

2 The web platform architecture

The HUBzero is a complex product that integrates many software and is based on various technologies. The basic HUBzero installation provides access to a powerful all-in-one server with rich functionality. In general, this functionality includes software lifecycle management, software launching, searching, access rights management, logging, software codes repository management, and so on. By further customizing it is possible to set up access to GRID computing resources, but the customizing process takes a lot of time, is difficult to perform and the type of external resources is restricted to GRID only.

We propose a lightweight solution for remote tasks execution. The basic idea is that in addition to the HUBzero server we deploy an external application server capable of receiving calculation tasks, running these tasks in a dedicated Docker container, and returning the result in a predefined form. A detailed description of the architecture and protocol of operation of such a server is presented in [9, 10]. This external application server is logically detached from the HUBzero platform, all the interactions are conducted via standard HTTP requests (REST API).

In order to run the calculation program to be shared on external resources a user should prepare two programs:

1. A frontend program to be installed and started via HUBzero. Upon starting, it receives input parameters from the user, sends a computational request to the application server, waits for the results to be received and displays them in a graphical form. Exactly this program can be shared among other users using standard HUBzero tools.
2. A calculation program to be run on external resources. The calculation program will be launched in the Docker container by the application server after receiving a request from the frontend program. Upon starting, it performs the requested calculations and returns results.

To ensure the transfer of a remote program to a Docker container a user needs to pre-allocate it in the image repository of the application server, while the frontend program call must contain the corresponding program identifier. Thus, running tasks on external resources is implemented using standard HUBzero tools, and the overall architecture of the system becomes quite flexible. As a result, the users can seamlessly mix launching basic HUBzero tools with running external Docker containers.

3 An example of using the web platform for nonlinear optics

As stated above, we used the HUBZero middleware to deploy a prototype of a web platform for the library of programs in nonlinear optics. As an example we pick up a program which calculates the

angle of phase matching [11] for a large number of crystal types stored in a database. The “angle of phase matching” is a geometrical feature which specifies conditions for nonlinear optical effects in crystal to become volume effects rather than point effects. Volume effect means that many atoms contribute to the effect.

This program calculates the angle of phase matching for the second-harmonic generation as a function of temperature, crystal parameters, and frequency of the incident radiation. The example program initially is a command-line script written in python, while its graphical interface is created by the special HUBzero module called the Rappture [13].

This program has been installed by its authors on our prototype of a web platform [12], after that the authors granted access to this program to a certain group of users. Further developments are considered to meet supplementary user needs. A frontend program concerning the use of Docker containers can be installed and shared in a similar way.

On the initial screen of the program there is a list of input parameters and a special button that should be pressed to perform the calculation. After the button is pressed the output results can be seen on the screen. In our example results are shown on a two-dimensional plot (see figure 1). The horizontal axis represents the frequency of the incident radiation and the vertical axis represents the calculated angle of phase matching. Holding a mouse pointer over the particular graph point a user can see the precise calculated value. Also there is a very convenient way to download all the results to the user’s local computer using the “Download” button located on the right side of the screen. After that a user can choose another set of parameters, run calculation one more time and download results again.

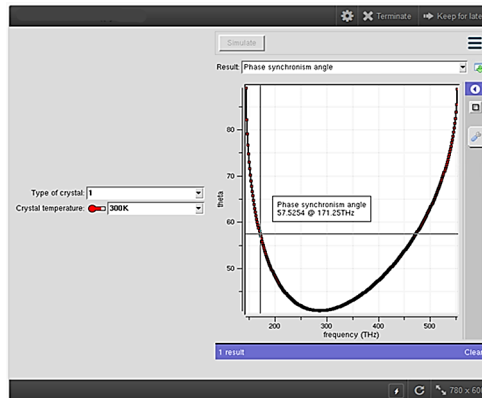


Figure 1. Screenshot illustrating the dependence of the angle of phase matching on the laser frequency

The above example shows that our HUBZero-based web platform is an open one, as users themselves can install new programs and share them among each other. This is an important difference from the already existing systems in the field of nonlinear optics. With the help of such a platform, users can share also the data, for example, the parameters of the crystals, which are used in the calculations.

It ought to be remarked that using Docker containers for external computation somewhat complicates the task of the author who wants to share the program. As for a regular user, nothing will change, and this is one of the most important features of the proposed solution.

4 Conclusion

Online web platforms is a fast-growing and promising area thanks largely to their usability and seamlessness. Unlike the solutions that offer a pre-installed set of programs, an open web platform allows its users to install their own programs and share them among others.

The HUBzero middleware is a very convenient tool for creating web platforms for scientific research, which was shown in this article by the example of a program for the angle of phase matching calculation. We also added to the basic HUBzero installation the capability to execute a calculation program on a specially created Docker container which is launched on the remote application server. This lightweight solution enhances the functionality of the platform.

As for the directions for our future research, we plan to improve our existing installation by increasing the computational capacity and providing some additional applications.

Summarizing, the proposed web platform provides shared access to the programs in nonlinear optics developed by the different users, which, in turn, stimulates wider use and faster development of that software.

Acknowledgement

The authors are grateful to the Russian Foundation for Basic Research (grant No 15-07-09309), with the financial support of which this work was carried out.

References

- [1] NLOsource, <http://www.nlosource.com/index.html>
- [2] RP Photonics, <https://www.rp-photonics.com/>
- [3] AS-Photonics, <http://www.as-photonics.com/index.php>
- [4] Polyakov S., et al., *Procedia Computer Science* **66** 602–608 (2015)
- [5] Prikhod'ko N., et al., *Journal of Physics: Conference Series* **681** (2016) 012052
- [6] McLennan M. and Kennell R., *Comput. Sci. Eng.* **12**, 48–52 (2010)
- [7] Thompson C. S., et al., *Proc. of the 2015 XSEDE Conf.: Scientific Advancements Enabled by Enhanced Cyberinfrastructure*, 36:1–36:7 (2015)
- [8] The docker software, <http://www.docker.com>
- [9] Polyakov S., et al., *Selected Papers of the 7th Int. Conf. Distributed Computing and Grid-technologies in Sci. and Edu.* **1787**, 557–560 (2017)
- [10] Polyakov S., et al., In *Proc. CSP2017*, *Journal of Physics: Conference Series* (2018), to be published
- [11] Ed. Trager F., *Springer Handbook of Laser and Optics* (Spr.-Verlag, 2012) 1694 pp.
- [12] Publishing of a tool, <https://help.hubzero.org/resources/173>
- [13] The Rapture, <http://nanohub.org/infrastructure/rapture/>