VIRTUAL ACCELERATOR: GRID-ORIENTED SOFTWARE FOR BEAM ACCELERATOR CONTROL SYSTEM

N.V. Kulabukhova[1], A.N. Ivanov[1], V.V. Korkhov[1], D.A. Vasyunin[1,2], S.N. Andrianov[1]

[1] *St.Petersburg State University, Universitetsky pr. 35, Peterhof, 198504, St. Petersburg, Russia, tel/fax: +7 (812) 428-71-59*

[2] *University of Amsterdam, Science Park 904, 1098 XH Amsterdam, the Netherlands*

The key idea of the Virtual Accelerator (VA) concept is the modeling of beam dynamics with the help of several software packages, such as COSY Infinity, MAD, etc., composed in pipelines and enacted on grid-enabled distributed computing resources.

The main use of the VA is simulation of beam dynamics by different packages with the opportunity to match the results (in case of using different solution methods for the same problem) and the possibility to create pipelines of tasks when the results of one processing step based on a particular software package can be sent to the input of another processing step.

The VA is considered as an information and computing environment and does not refer to real-time control systems. However, real-time control can be provided by connection to specialized software (e.g. Experimental Physics and Industrial Control Systems – EPICS). Such kind of VA is examined in [1] and [2] where authors emphasize on accelerator control development.

The general idea of the software implementation is based on the Service-Oriented Architecture (SOA) that allows using grid and cloud computing technologies and enables remote access to the information and computing resources. Distributed services establish interaction between mathematical models and a low-level control system.

The VA user interface allows getting solutions both from simulation models and from real accelerator machines. This approach gives researchers ability for system identification, parameter optimization, and result verification, which is impossible without computational models. The same approach to develop a virtual laboratory is discussed in paper [3] for nuclear physics applications.

The LEGO paradigm is used for the VA design. In terms of information technology it corresponds to object oriented design and component programming. Each object is represented as an independent component with own parameters and behavior. In paper [4] development of distributed computing systems based on this concept is examined in more detail.

Introduction

Contemporary control systems of complicated physics facilities, such as different accelerator complexes, thermonuclear reactors, etc., assume to use efficient scenarios to support operating mode. The development of workflows maintaining the work of different facilities is based on clear formalized mathematical models, describing appropriate processes on the one hand and effective software implementation on the other. The complexity of such facilities makes the models multicompound, and leads to the set of mathematical methods and formalizations. Variety of models and the computational complexity encourage to use distributed environment and appropriate methods, in particular Grid- and Cloud-technologies. It is necessary to distinguish two components: modeling (physical or mathematical) and software approach. The first one is to preliminary investigate (theoretically or experimentally) different effects of the installation. As the result the scenario for the control system of real facility if formed. The second component is responsible for the realization of the scenarios to achieve required operating modes.

In order to control large-scale accelerators efficiently, a control system with a virtual accelerator model was constructed by many facilities. In many papers by the notion of Virtual Accelerator an on-line beam simulator provided with a beam monitor scheme is meant. It works in parallel to real machine. The machine operator can access the parameters of the real accelerator through the client and then feed them to the virtual accelerator, and vice versa. Such a virtual machine scheme facilitates developments of the commissioning tools; enable feasible study of the proposed accelerator parameters and examination of the measured accelerator data. That is the common scheme of virtual accelerators used in different laboratories. Until now there is no virtual accelerator working without a real machine. Our goal is to construct a Virtual Accelerator application that can be used independently of any machine.

Virtual Accelerator construction

The Virtual Accelerator is considered as a set of services and tools enabling transparent execution of computational software for modeling beam dynamics in accelerators on distributed computing resources. Users will get the access to VA resources by unified interface including GUI on different platforms. Figure 1 shows the scheme of VA.

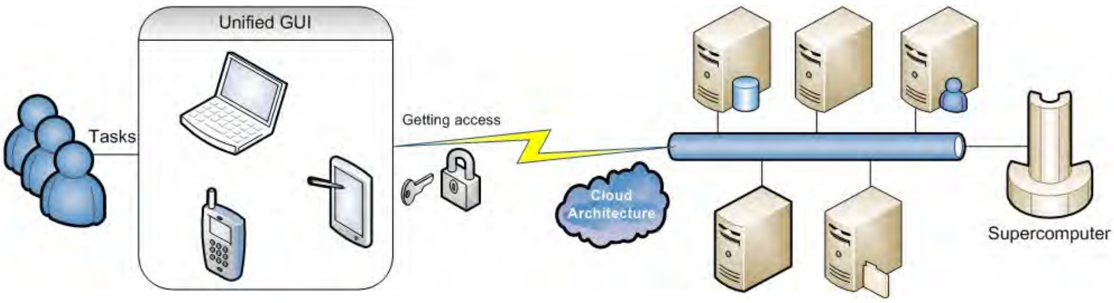

Figure 1: Schematic view of Virtual Accelerator environment

Virtual Accelerator as a computer model

Overview of the current literature allows us to formulate the following definition of the concept of Virtual Accelerator.

The key idea of Virtual Accelerator concept is the beam dynamics modeling by a set of several packages, such as COSY Infinity, MAD, etc., based on distributed computational resources, organized on Grid- and Cloud- technologies.

The main purpose of this environment is to conduct numerical experiments required to configure the actual physical facility during its start-up and ensure it is working properly in the relevant experiments. This purpose imposes certain restrictions on the resource potential of this complex.

Virtual Accelerator as a theoretical model

The main purpose of a virtual accelerator is to conduct computational experiments to simulate the beam dynamics using various software packages with the ability to compare the results of calculations (in case solutions of the same problem are obtained by various means), and the ability to create a task flow (solutions of one package can be used as input for the subsequent calculation in another package), see figure 2. In addition, an important part of the computational experiments are optimization problems sweeping over the possible parameters and system configurations in order to find the best option. In this case, technology of parallel computing and massively parallel computing systems can be used efficiently. Each of the above configurations may be executed independently of the others, along with them, together forming a parametric study of a given domain configurations of the accelerator.

The user has access to the resources of the virtual accelerator through a "single window" – a portal or some interface shell (for example, applications based on Java Webstart). In this interface the user selects a package or several packages in which wishes to carry out calculations, set the input data and parameters, and the task is run on the available computing resources. Access to resources can be provided on the basis of standard solutions used in the Grid-technology.

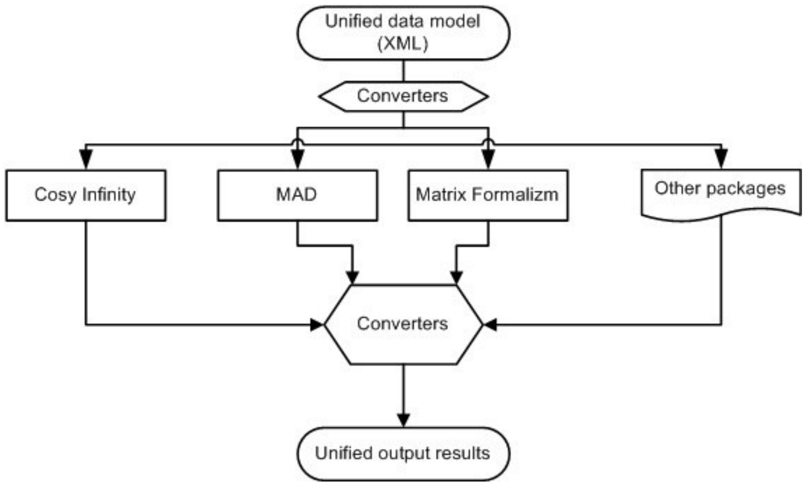

Figure 2: Computational experiment in VA

In this case, the resources to perform the calculations are taken from the common pool. At the time of peak load with a lack of resources a hybrid approach can be used, where the missing resources are taken from the Cloud [5]. Workflow management systems are required to control execution of workflows [6], as well as the conversion of data between the formats used by different packages must be taken into account. This requires the design of the experiment description language that can be translated into the language of each package for intermediate calculations.

Template using a Virtual Accelerator is as follows:

1. The user has access to the interface computer (authentication, authorization);
2. The user sets the initial conditions and parameters for the calculation. The options are:
   1) using "generic" description language that can be converted into a specific language used in the packages MAD, COSY, etc.;
   2) directly in the language of one of the packages, which will be used for calculations;
3. The user selects a package (or a set of packages) which will perform calculations;
4. The user instructs the system to run calculations using the packages and given initial data. This is done using either a dedicated resource (cluster, computer), which is selected manually or automatically selected resource based on information about requirements of the application.
5. After starting the calculation the user may wish to see the intermediate results. Depending on the abilities of packages, it may be possible to do it or not.
6. Most important thing is to track errors that occur. As practice shows, the most difficult is to figure out why something does not work. Carefully organized collection of error messages must be maintained. To collect this information, annotate data and results of computations so called provenance systems are used [7].
7. VA offers the possibility of organizing the flow of tasks – sequential running packages, where the next step uses the data obtained on the previous step. The means to convert data formats between packages must be provided.
8. Calculation results can be visualized by means of VA. It is particularly important to be able to visually compare the results of calculations of a problem in different packages.
9. A simple way to be able to restart the same calculation after a minor adjustment of parameters (which may be carried out after the analysis of the results).

It must be emphasized that the Virtual Accelerator is a modeling environment and is not directly related to the real accelerator control systems (eg, EPICS). However, the organization of communication between such systems is possible.

The task of running diverse software packages that have different requirements for the installed operating systems, libraries and other dependencies can be simplified by using the technology of cloud computing. In this case, the virtual machine images ready to set up and configured simulation package can be deployed on provided computing resources. In addition, the use of Cloud enables the experiment in case of lack of computing resources in the Grid, as well as in the mode of "urgent computing", when it is necessary to get the results to a pre-set time.

Conclusions

This paper presents a prototype Virtual Accelerator environment used for modeling beam dynamics with the help of a number of software packages on grid-enabled distributed computing resources. We present some design concepts, discuss usage scenarios and prototype implementation. Some modules such as global optimization tools, simulation and numerical algorithm are completely developed, other are in a progress. The future development of the research can be based on writing software using different parallel techniques and complete implementation of the described approaches.

Some approaches that were described above were tested in the distributed computational environment at the faculty of Applied Mathematics and Control Processes on the department of Computer Modelling and Multiprocessor Systems.

References

[1] P.C. Chiu, C.H.Kuo, Jenny Chen, Y.S. Cheng, C.Y.Wu, Y.K.Chen, K.T. Hsu, “Virtual Accelerator Development for the TPS”, IPAC’10, Kyoto, Japan 2010, WEPEB019, p. 2728, http://www.JACoW.org.

[2] C. Gulliford, I. Bazarov, J. Dobbins, R. Talman, N. Malitsky, “The NTMAT EPICS DDS Virtual Accelerator for the Cornel ERL Injector”, IPAC’10, Kyoto, Japan 2010, WEPEB022, p. 2734, http://www.JACoW.org.

[3] V. Korkhov, D. Vasyunin, A. Belloum, S. Andrianov, A. Bogdanov, “Virtual Laboratory and Scientific Workflow Management on the Grid for Nuclear Physics Applications”, Proc. of the 4th Intern, Dubna, Russia 2010, p. 153.

[4] N. Kulabukhova, A. Ivanov, V. Korkhov, A. Lazarev, “Software for Virtual Accelerator Designing”, 13th International Conference on Accelerator and Large Experimental Physics Control Systems: Proceedings of ICALEPCS2011, Grenoble, France, WEPKS016 p. 816.

[5] [5] Rudolf Strijkers, Willem Toorop, Alain van Hoof, Paola Grosso, Adam Belloum, Dmitry Vasuining, Cees de Laat, Robert Meijer, "AMOS: Using the Cloud for On-Demand Execution of e-Science Applications, " in IEEE e-Science 2010 Brisbane, Australia: IEEE Computer Society, 2010.

[6] A. Wibisono, V. Korkhov, D. Vasunin, V. Guevara-Masis, A. Belloum, C. de Laat, P. Adriaans and L.O. Hertzberger, WS-VLAM: Towards a scalable workflow system on the Grid, Proceeding of the 16th IEEE International Symposium on High Performance Distributed Computing, June 27-29, 2007, Monterey Bay, California, USA.

[7] Michael Gerhards, Volker Sander, Torsten Matzerath, Adam Belloum, Dmitry Vasunin, Ammar Benabdelkader. Proceeding WORKS '11 Proceedings of the 6th workshop on Workflows in support of large-scale science Pages 57-66, ACM New York, NY, USA ©2011.