

# SOFTWARE FOR VIRTUAL ACCELERATOR ENVIRONMENT

N. Kulabukhova\*, St.Peterburg State University

## Abstract

The article discusses appropriate technologies for software implementation of the Virtual Accelerator. 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. Control system toolkits EPICS (Experimental Physics and Industrial Control System), realization of the Graphical User Interface(GUI) with existing frameworks and visualization of the data are discussed in the paper. The presented research consists of software analysis for realization of interaction between all levels of the Virtual Accelerator.

## INTRODUCTION

In order to control large-scale accelerators efficiently, a control system with a virtual accelerator model was constructed by many facilities [1, 2]. In many papers by the words Virtual Accelerator an on-line beam simulator provided with a beam monitor scheme is mean. It works parallel with 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 [3]. Until now there is no virtual accelerator working without real machine. Our goal is to construct virtual accelerator application can be used independently from any machine.

## CONCEPT OF THE VIRTUAL ACCELERATOR

The model of Virtual Accelerator like any other model is standard. As it is shown on the figure 1 first comes the approximation model where all necessary options are determined. Further, the theoretical models are constructed. On them computational models are build. After that the time for testing and verifying comes.

The key idea of Virtual Accelerator (VA) concept is beam dynamic modeling by the set of several packages, such as COSY Infinity, MAD, etc., based on distributed computational resources, organized on Grid- and Cloud-technologies. Simulation beam dynamics by different packages with the opportunity to match the results (in case of using different resources for the same task) and the possibility to create the set of tasks when the results of using

\* kulabukhova.nv@gmail.com

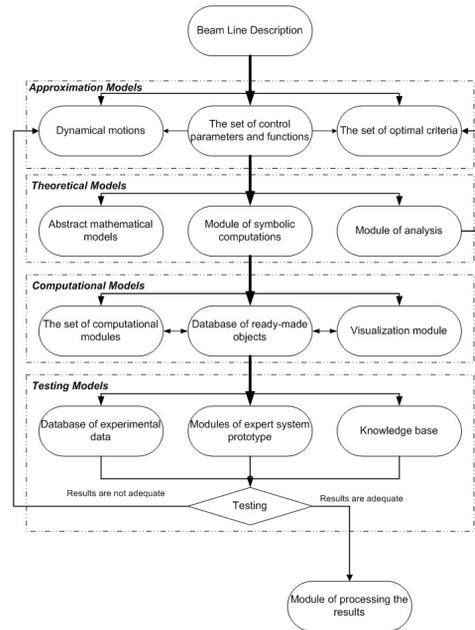


Figure 1: The total circle of computational experiment

one package can be sent to the input of another is the main use of VA [6]. Users will get the access to VA resources by unified interface including GUI on different platforms. Figure 2 shows the scheme of VA

The natural parallel and distributed structures of beam physics problems allow the use of parallel and distributed computer systems. But the usual approaches based on traditional numerical methods demand using the resources of supercomputers. This leads to the impossibility of using such multiprocessing systems as computational clusters. There are two classes of problems in beam physics which demand very extensive computer resources. The first class includes long-time evolution problems, the second is concerned with the computer realization of optimization procedures for beam lines. Examples of the first type of problem include multi-turn injection and extraction of the beam in circular accelerators. Usually, these problems do not consider space charge effects. For advanced applications it is essential to study beam dynamics in high-intensity accelerators. Such machines are characterized by large beam currents and by very stringent uncontrolled beam loss requirements. An additional difficulty of numerical simulation is connected with long-time beam evolution that requires the computation of hundreds of thousands or millions of turns. It requires the use of high-performance computers for beam evolution study. The problems of similar multi-turn evolution such as transverse stability with

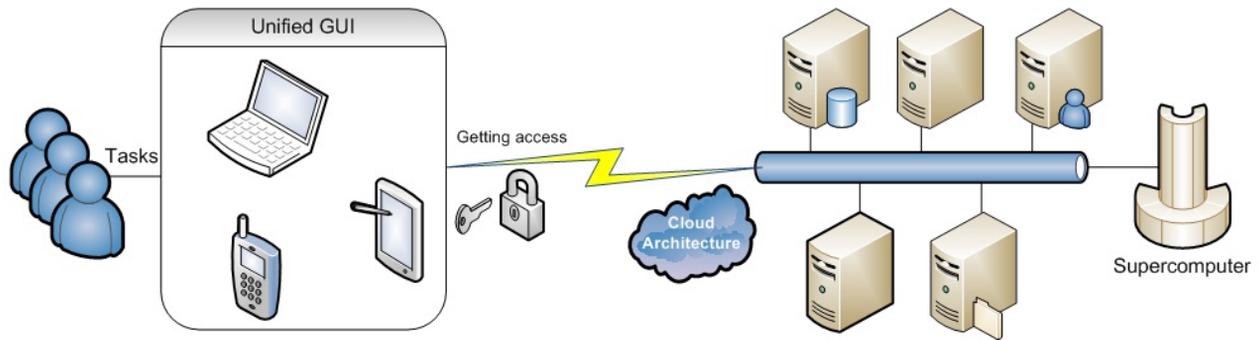


Figure 2: VA scheme

nonlinear space charge, uncontrolled beam losses due to space-charge-induced halo generation, etc. can also be mentioned. These problems are peculiar to modern high-intensity machines and require careful investigations of long-time evolution effects. From the computational point of view there are some problems related to the choice of models for beams with space charge, the presentation form of the beam propagator, and so on.

### Construction of VA

The common scheme of VA as a computational resource can be presented as following:

- Access to the VA resources by the authorization and authentication of users (symbolic password, graphical password, etc.)
- Specify initial data and settings for calculation. Interaction between user (unified GUI interface) and different packages (COSY Infinity, MAD, OptiM)
- Choice of the packages and simulation algorithms (more deeply in section Methods realisation will be discussed)
- Monitoring of the task states, data visualization and representation. Access to a logging system with error tracking.
- Access to the storage of user data that can be represented as simple remote file system with users directories and files.

Grid environment allows coordinated resource sharing and problem solving among groups of trusted users within Virtual Organizations. Such environments enable global distributed collaborations involving large numbers of people and large scale resources, and make data and computing intensive scientific experiments feasible. One of the important research topics in e-Science is to develop effective Grid enabled Problem Solving Environments (PSE), also called Virtual laboratories, for different scientific domains. Organizing software utilities (e.g. simulators, visualization and data analysis tools) as a meta experimental environment, a PSE allows a scientist to plan and conduct experiments at high level of abstraction [4].

### Methods realisation

It has already been said that Virtual Accelerator in itself can use different packages to calculate. Note that user can define the structure on one language (for example, on COSY Infinity notation) and select another program for calculation (e.g. OptiM). In this case conservation of system description will be made by the VA infrastructure invisible for user. And give the instructions to start the simulation. How it will be is shown on Figure 3.

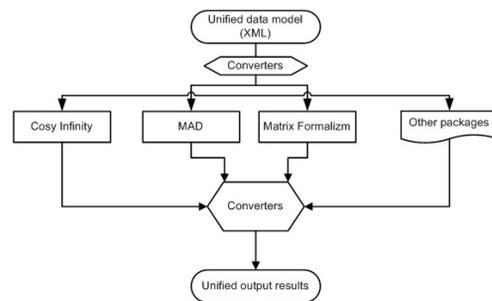


Figure 3: Scheme of unified access to different packages

## SOFTWARE FOR VA

### Client Application

As we mentioned above VA can provide unified access to it services both for web and desktop application. Not all services are required in each case. For example, for EDM-search project<sup>1</sup> it is necessary to include only electrostatic elements, optimization modules and related components. Other modules are enabled but available for the research purposes.

### Task Monitoring

For the calculation complex problems it is important to provide monitoring tools. Researchers at any time can have an opportunity to see progress of the task, restart or cancel it, save obtaining result and etc. This requirements provide

<sup>1</sup>St. Petersburg State University collaborates with Institute for Nuclear Research of Forschungszentrum Julich, Germany.

a complete integrated computing environment for composing, running, controlling and visualizing applications (see figure 4).

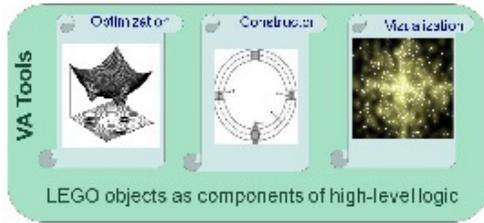


Figure 4: VA tools

### Connection to Physical Equipment

From our point of view model of Virtual Accelerator should consist of a simulation engine that reads in all relevant lattice settings from the accelerator control system and computes data corresponding to the real diagnostic data output from the real accelerator. Or, if it is no accelerator exist at that time, mathematical model will be given to compare the results. Here would ideally be a one-to-one map between all relevant control parameters and output data from the machine and its simulated counterpart. This model inherently assumes that the simulated lattice updates automatically any time the physical lattice settings are modified. If we speak about mathematical model, it means that at any time we can change settings and continue the computations This model should represent an extremely valuable tool to the beam physicist by providing a useful way to benchmark the simulation engine as well as providing a detailed description of the dynamics in the accelerator in real time. By saying real time, it means the time the system needs to get the problem, to analyze it and to give the message about it on the screen. It does not allow the beam physicist to directly use the simulation engine to guide beam operation because the virtual lattice settings are limited to those read in from the physical machine. This is the problem that led us to identify the three key requirements used in the design and implementation of our application (having mathematical model it is not a problem):

- The ability to simultaneously and independently operate two accelerators in real time: one real and one virtual.
- The ability to synchronize the lattice settings in both machines.
- The ability to control and display relevant data from both machines in one integrated graphical user interface.

To fulfill these requirements we use Experimental Physics and Industrial Control System (EPICS)[5] because it has followed a threetiered approach common to many high-level accelerator control and simulation systems. Figure 5 shows the basic layout of implementation. The three

tiers in this scheme include the distributed Input/Output Controllers (IOCs) serving data from the physical accelerator (bottom), the EPICS-DDS middle layer (middle), and the Machine Client application (top). That is exactly we need to our system.

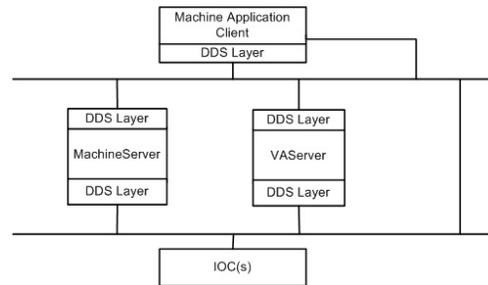


Figure 5: The Virtual Accelerator layout

### CONCLUSION

The future development of the research can be based on writing software using different parallel techniques and complete implementation of the described approaches.

### ACKNOWLEDGMENTS

Author thanks prof. Serge Andrianov and Andrey Ivanov for many useful discussions about Virtual Accelerator development.

### 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] H. Harada, K. Shigaki, F. Noda, H. Hotchi, H. Sako, H. Suzuki, K. Furukawa, S. Machida, "VIRTUAL ACCELERATOR AS AN OPERATION TOOL AT J-PARC 3 GEV RAPID CYCLING SYNCHROTRON (RCS), EPAC'06, Edinburgh, Scotland, 2006, WEPCH128, p. 2224, <http://www.JACoW.org>.
- [3] A.Shishlo, P. Chu, J. Galambos, T. Pelaia, "THE EPICS BASED VIRTUAL ACCELERATOR CONCEPT AND IMPLEMENTATION, Proceedings of the 2003 Particle Accelerator Conference, pp. 2366-2368.
- [4] V. Korkhov, D. Vasyunin, A. Belloum, S. Andrianov, A. Bogdanov, "Virtual Laboratory and ScientificWorkflow Management on the Grid for Nuclear Physics Applications, Distributed Computing and Grid-Technologies in Science and Education: Proc. of the 4th Intern, Dubna, Russia 2010, p. 153, <http://www.JACoW.org>.
- [5] C. Gulliford, I. Bazarov, J. Dobbins, R. Talman, N. Malitsky, "The NTMAT EPICS-DDS Virtual Accelerator for the Cornell ERL Injector, IPAC'10, Kyoto, Japan 2010, WEPEB022, p. 2734, <http://www.JACoW.org>
- [6] N. Kulabukhova, A. Ivanov, V. Korkhov, A. Lazarev, "Software for Virtual Accelerator Designing, ICALEPCS'11, Grenoble, France, 2011, WEPKS016, p. 816, <http://www.JACoW.org>.