ON THE APPLICATION OF THE HIGH-PERFORMANCE VIRTUALIZED COMPUTING INFRASTRUCTURE FOR PROCESSING LARGE VOLUMES OF EXPERIMENTAL MODELING DATA

Khashkovsky V. V., Bolotov M. V., Shkurko A. N. and Trotsenko R. V.
Southern Federal University, Russia
E-Mail: vkhashkovsky@sfedu.ru

ABSTRACT

The paper discusses approaches to organizing systems of experimental data analysis in terms of effective management of a high-performance computing infrastructure. The aim of the given work is to describe the organization of an efficient high-performance computing infrastructure for experimental modeling. To achieve these objectives, the analysis of data sources and processing stages, specific to the subject area, is performed. The approaches, based on key technologies, including the use of virtualization, are considered. For the approach based on the use of virtualization, the problems of the virtual infrastructure management and methods of its administration are considered. The conclusions tell about the coordination between the virtual infrastructure management policy and the management policy implementation resources based on development of specialized tools. The work performed is included into the basic part of the government task on "The Information and Algorithmic Support of Digital Control Systems, Autonomous High-Precision Navigation and Machine Vision in the Future of Aircraft: the Development of Theoretical Basis for Design, Algorithms, Methods of Efficient and Reliable Software Implementation, the Use of a High-Performance Computing Infrastructure for Experimental Modeling".

Keywords: virtualization, cloud computing, cluster technology, data processing, experimental data.

1. INTRODUCTION

While developing information systems and applying computing technologies for storing, sharing and processing data flows and arrays, we have passed several stages of improving its practical methods, from simple file storages through well-formalized statistical analysis methods to intellectual analysis methods, currently being developed. At the moment, the convergent nature of data flows and, correspondingly, data storages is increasingly oriented to a weak structure of data arrays by default, as only a little percent of data is accompanied by meta-information.

The traditional methods are usually based on time-tested mathematical tools; however, new methods are gaining their significance, mostly those connected to data mining, business intelligence, business analytics etc. It should be highlighted that the most of attention, for obvious reasons, is paid to numeric data processing. The so-called unstructured text information requires a human analyst to perform the analysis, and its significant volume raises a question on possibilities of and approaches to such analysis and defining the automated processing usage domain.

On the other hand, the application of intellectual data analysis (IDA) allows obtaining results in every application area that are characterized with new knowledge extracted from data arrays by detecting previously unknown, "hidden" and unobvious dependences between different-type data. Currently, there is the sufficient number of successful IDA applications, mostly in socially significant spheres, undoubtedly providing the important practical results. Upon that, on the one hand, IDA is oriented (at least with no theoretical limitations of this kind) to processing not only numeric but also text data (mostly unstructured). The subject of special concern is the application of IDA for processing experimental modeling data in scientific and engineering research. There are successful examples of such IDA applications described in publications, e.g. [1].

On the other hand, the peculiarity of IDA application in processing of experimental research results is that, first, the data is mostly numeric, making it possible to combine IDA and traditional mathematical methods; second, the volume of such experimental data raises the question of using a high-performance computing infrastructure and corresponding method modification, making it possible to obtain the processing results in a reasonable time.

2. DATA SOURCES AND PROCESSING STAGES

The modern literature [2] provides the quite simple scheme of data mining systems, that although describes all necessary data processing stages and allows building highly complex systems.

The area of use, that includes the experimental data processing, does not require the complicated methods of data pre-processing, making it in some way easier to form the basic data and work with data storages directly; other ways of obtaining the basic data, for instance, with search engines, is considered in [3]. It is to be mentioned that numeric data usually already contain their semantics, or metainformation, defining numbers in storage.

More than that, in data analysis systems, in particular, in the experimental data processing and similar scientific systems, the final stage of visualizing the data (or the processing results) can be emphasized. This stage is specifically named - the scientific graphics. On this stage, the visual representation of data analysis results is formed. The data mining results interpretation can be performed diagrammatically (graphs, diagrams, charts) or as the text, including the natural language texts. Such interactive visualization elements as dynamically changing graphic representation, scaling, data fusion, various animation and color changing allow perceiving the information more efficiently. Let us note that the certain form of result representation is subject-dependent, so the processing algorithm can determine the way results will be represented.

3. DATA PROCESSING APPROACHES

As noted above, for processing numeric data the traditional methods of mathematical statistics are quite efficient, however, the application of IDA or combined methods can lead to obtaining new knowledge thank to finding out previously unknown relations. Such IDA methods primarily include the machine learning methods, artificial neural networks, association methods and some of the others. These methods are distinguished by the high labor intensity, conditioning its usage in high-performance computing platforms.

On the other hand, nowadays high-performance computing platforms and their computing organization technologies rely on the reputable software libraries and programming paradigms, such as MPI [8], PVM [12] etc., also building conceptions like Grid systems [11] or cloud platforms. From the point of economically efficient usage, a cloud platform computation capacity significantly exceeds all alternate ways. The basis of the cloud platform building on the main IAAS (infrastructure as a service) level is introduced by virtualization tools (XEN- [5], KVM- [6], VirtualBox-, VMWare-based etc.) allowing using computation resources upon request (and as much as necessary) and, what is important, adjusting, or connecting new hardware units when necessary. There are many reasons to use virtualization. Pragmatically and economically, the best advantage is the possibility of server consolidation. Taking into account that the physical server hardware is rarely completely loaded, significant resources can be saved by transferring server systems to the virtual environment and storing them on the same physical server. The next step towards improving this idea is to store two identical copies of a VM on different physical servers and to balance the load between them. The most interesting thing here is the possibility of a “live” migration when the replication of a VM is performed without actually stopping it.

So, the task of organizing the experimental research data processing consists of three main parts: developing analysis algorithms, implementing these algorithms and deploying a virtual computing infrastructure that the processing is based on.

Developing data processing algorithms depends on a certain task and doesn’t concern implementation and a computing infrastructure generally. Otherwise, the implementation of algorithms depends on a utilized computing infrastructure building technology and, concerning the replicability of software, should use the maximally widespread software libraries of the system level, e.g. MPI, or be based on using universal software packages. The orientation of development to usage of MPI can be considered less efficient than more high-level modeling and processing packages, e.g. Matlab [9]. In particular, Matlab allows significantly speeding up the development by using a large number of library procedures. Then, the initial advantage of MPI, that is contained in the organizing of parallel interaction, is hardly significant, as Matlab offers tools for paralleling computations, mostly automated, speeding up not only development or debugging but also the modification, what is a serious advantage, as the experimental data processing makes modifying or correcting algorithms quite frequent needs.

Therefore, due to the actual advantage of Matlab with the parallel computations package, the main question is the deployment of the virtual computing infrastructure. With this approach, the virtualization is not necessary, as the parallel functioning of Matlab can be organized on the basis of a separate cluster. However, as the volume of calculations cannot be predicted, the virtualization allows adding necessary computing units in the simplest way – by copying existing units instead of configuring separate clusters.

4. VIRTUAL INFRASTRUCTURE MANAGEMENT

The virtualization of a separate computing unit (a cluster unit) itself is not a difficult task. After installing the basic OS, it is needed to install the virtual machine allowing creating guest machines; it actually makes no difference with installing software to an ordinary computer. The task of infrastructure management occurs, as the amount of cluster units grows, and each of them requires a random number of guest machines to be configured (also, for different objectives). These guest machines are actual computing units launching the components of Matlab modeling environment.

The basis of the virtual infrastructure management concerning creating, deleting or moving virtual machines between cluster units is the Libvirt [4] open software library. This library allows managing virtual machines directly, including those stored on other cluster units, accessible through a network. Libvirt is the basis of such known guest machine management shells as oVirt [10], OpenTask [7] or simpler management tools like Virsh or Virt-manager.

One important advantage of the Libvirt library is its openness to third-party applications, what allows using its guest machine management functionality by your own applications, included into the entire experimental data processing system and able, based on the initial data parameters, amount and possible processing time, to create additional or delete spare guest machines to be used by Matlab as workstations, taking the estimated time into account.

5. MANAGEMENT INTERFACE ACCESS

The given Libvirt library provides other applications with the default API and uses it by itself. Therefore, the utilization of this API is possible only by software means, from another application. So, the following access coordination is necessary. On the one hand, the Matlab modeling environment requires programs and algorithms described in the built-in Matlab language, oriented on utilizing inner possibilities of Matlab. On the other hand, Matlab programs can launch other OS programs and send them command string parameters but without their own access to the Libvirt API. So, for the virtual computing infrastructure management, the higher access level is required than the default API provides.

To solve this task, the software access tool was developed for managing virtual machines, allowing by launch sending certain commands (with or without parameters) with the command string, redirect these commands to corresponding Libvirt API functions, process and return results to an initial program.

Therefore, the general software tools interaction scheme for implementing the virtualized computing infrastructure management is the following.

Initial experimental research data to process is gathered and directed to a database. A type of database does not matter. This data must be accessible to a modeling environment performing the following analysis and processing. As the modeling environment, we consider the Matlab package with the processing paralleling functionality. The processing algorithm implementation itself must be written in the certain modeling environment (Matlab) in its corresponding language. The processing algorithm functioning is maintained by the modeling environment, so the algorithm has access to data structures provided by the modeling environment, on the one side, and to the software access tool for managing virtual machines, on the other side. In this sense, the processing algorithm determines the virtual machines separating policy, and the software tool implements this policy by accessing the Libvirt API.

6. CONCLUSIONS

Currently, despite the significant number of software systems of different sizes, applied in the data analysis; different, including specialized, methods of data processing, one of the most important tasks is the computing resources management. A relatively short but intense period of development of computing devices has led to the creation, development and practical application of a large number of different computation organization technologies. The basis of these technologies was always the performance that was being increased on both the hardware and software levels, with parallel computations and task-oriented computing systems. However, the high performance was the main but not the only criterion of computing system efficiency. Recently, one of the main discussion points is the efficiency (including energy efficiency) of computing resources, the convenience of adjusting and separating resources by request, reliability and fail-safety. To achieve these goals, the technologies of virtualization and virtualized resources efficient management are used. Such management is rather low-level and is possible only by accessing software libraries APIs directly. On the other hand, the solution of tasks, including those connected with the data processing, requires the usage of high-level processing tools, modeling environments, etc. that, though having access to necessary information, have no possibility to manage its distribution. As a result, the organization of efficient management is based on the application of auxiliary software tools allowing implementing the proper management policy by providing a higher level of interface than an API itself.

REFERENCES

[1] A.A. Barsegyan, M.S. Kupriyanov, V.V. Stepanenko, I.I. Kholod. Data Analysis Technologies: Data Mining, Visual Mining, Text Mining, OLAP. 2-е изд., перераб. и доп. - СПб.: БХВ-Петербург, 2007. - 384 с: ил. ISBN 5-94157-991-8.

[2] Jiawei Han, Micheline Kamber. 2006. Data Mining: Concepts and Techniques Second Edition. USA Elsevier Inc. p. 743.

[3] V.V. Khashkovsky, A.N. Shkurko. 2014. Modern Approaches to the Large Data Volumes Processing System Organization. // Известия ЮФУ. Технические науки - Таганрог: Изд-во ЮФУ. №8 (157). - 9 с. (с 241-250).

[4] The Libvirt open library official website // http://www.libvirt.org/;

[5] The XEN Official Website // http://www.xenproject.org/;

[6] The KVM Official Website // http://www.linux-kvm.org/page/Main_Page;

[7] The OpenStack Official Website // http://www.openstack.org/;

[8] The Open MPI Official Website // http://www.open-mpi.org/;

[9] The MATLAB Official Website // http://www.mathworks.com/;

[10] The oVIRT Official Website // http://www.ovirt.org/Home;

[11] Grid computing, from Wikipedia, the free encyclopedia // https://en.wikipedia.org/wiki/Grid_computing;

[12] The PVM Official Website // http://www.csm.ornl.gov/pvm.