Real-Time Visualization in the Grid Using UNICORE Middleware

Krzysztof Benedyczak¹, Aleksander Nowiński², Krzysztof Nowiński², and Piotr Bała¹,²

¹ Faculty of Mathematics and Computer Science, Nicolaus Copernicus University, Chopina 12/18, 87-100 Toruń, Poland
² Interdisciplinary Center for Mathematical and Computational Modelling, Warsaw University, Pawińskiego 5a, 02-106 Warsaw, Poland

Abstract. This paper presents real-time data streaming and client side visualization which was developed using UNICORE middleware. Described here IVIs system extends UNICORE capabilities and allows for streaming of the data produced by the running simulations. IVIs client provides user with advanced tools for data visualization and analysis. The developed software can be also used for remote steering and control. The important feature of the developed solution is utilization of the UNICORE security infrastructure which allows for data streaming without opening new holes in the firewall. The performance of the IVIs system has been measured and is presented. We have found that bandwidth is mostly limited by the network speed. Performed tests conformed that developed tools can be efficiently used for data streaming and on-line visualization and streaming on the grid.

1 Introduction

Existing grid middleware has been developed with the main focus on the batch type jobs. Therefore the real-time, on-demand data streaming doesn’t suit paradigm of any grid middleware. The reasons of such situation are well known: a large scale computations are performed as batch jobs which produce data postprocessed than on the local workstations. A job is submitted, then possibly queried for status and output is fetched to the local system for postprocessing and visualization.

The main disadvantage of the grid middleware is large communication overhead caused by the infrastructure which provides seamless access to the resources and ensures interoperability between different systems. This overhead results from transfer quite large amount of metadata and from need to invoke many metaoperations. Both of them are not needed for main computations but are mainly caused by the grid middleware itself.

As it has been observed, as systems get more universal and powerful, the grid middleware is increasing volume of additional communications and operations.
This is especially true for forthcoming XML based technologies. Grid tools become extended web services and invocation of even simple service job involves serious data preparation and processing.

Streaming of the data generated as a result of the remote computations which run on the grid is not trivial. In particular, there are two main problems to be solved. The first one is access to the job running in the batch system on the target computer which allows to fetch data, and the second one is performance which can be limited by the different components of the grid middleware making real time streaming unrealistic.

The first problem can be solved by the simulation of the interactive access using simple batch jobs. There was number of successful implementations of this approach, for example interactive access tools developed for the UNICORE [1]. Such approach assumes that each interactive command is submitted by a simple batch job, sometimes called service job. In this way user can mimic interactive access but middleware overhead is large. As consequence, the large amount of data cannot be transferred and this solution cannot be used for advanced visualization or real-life steering.

The performance required for the visualization on the grid is usually achieved by creation of the dedicated connection - directly from the system which produces data to the visualization client. This connection is initiated by the grid middleware but is realized using standard network technology bypassing grid security infrastructure. Therefore user cannot use this mechanism with firewalls and local networks.

This paper presents true real-time data access and client side visualization which was developed based on the UNICORE platform [2], however the basic middleware has not been modified. The UNICORE offers complete and ready to deploy solution. Developed extensions benefit from the UNICORE modular design and flexibility. The detailed description of the UNICORE middleware, its architecture and basic functionality can be found elsewhere [3,4].

2 Extensibility of the Core Unicore Software

The UNICORE middleware offers well known plugin infrastructure for the UNICORE Client. Basically, the most of the Client functionality is based on the plugins: some of them are distributed together with Client’s standard distribution and some are available as third party add-ons developed for the specific applications. The UNICORE plugin can be implemented using two schemes: Task and Extension. The purpose of the task is job creation and its submission to the target system. The task is incorporated in the UNICORE workflow and is part of the job created by the user and executed using UNICORE mechanisms.

The extension gives more freedom for developer. Especially it can use different communication schemas offered by the UNICORE Client and even allows for implementation of whole communication system from the scratch. This functionality has been used previously for example to access databases through their