EGI-InSPIRE

FIRE AND SMOKE SIMULATION VIRTUAL TEAM PROJECT REPORT

https://wiki.egi.eu/wiki/VT_Fire_Simulation

Date:	**14/09/2012**
Document Status:	**FINAL**
Dissemination Level:	**PUBLIC**
Document Link:	**https://documents.egi.eu/document/1341**

Abstract

This report is the final report of the EGI-InSPIRE Virtual Team project Fire and Smoke Simulation. The project started in December 2011, run for 6 months and published this final report in September 2012. The report describes the work that was carried out by the project, as well as collected requirements of Fire simulation community towards EGI infrastructure. The document reports the issues that have been identified by the project and describes follow up actions that can lead towards an EGI VRC on Fire and Smoke simulation.

II. DOCUMENT LOG

Version	Date	Comment	Author/Partner
0.1	09/08/2012	First draft with content outline	P. Kurdel / IISAS, J. Sebestyénová / IISAS
0.2	15/08/2012	Feedback on the content	G. Sipos / EGI.eu
0.3	03/09/2012	Updates based on the feedback	P. Kurdel / IISAS, J. Sebestyénová / IISAS
0.4	13/09/2012	Feedback on the report - TODOs	G. Sipos / EGI.eu
1	13/09/2012	Updates based on the feedback	P. Kurdel / IISAS, J. Sebestyénová / IISAS
2	14/09/2012	First public version in EGI DocDB	G. Sipos / EGI.eu

III. APPLICATION AREA

This document is a public report produced by the members of the 'Fire and Smoke Simulation' EGI Virtual Team project, run under the EGI-InSPIRE NA2 virtual team framework. Further information is available at https://wiki.egi.eu/wiki/Virtual_team.

IV. TERMINOLOGY

A complete project glossary is provided at the following page: http://www.egi.eu/about/glossary/.

TABLE OF CONTENTS

1   INTRODUCTION

In the last decade, fire science and fire engineering have grown thanks to the continuous progress of computational fluid dynamics (CFD), both in the field of theoretical modelling of physical phenomena and in the field of technological implementation of these models. The driver of model improvements is complex algorithms and the growing power of modern computers. However, not even the growing performance of commodity computers can keep up with the computing needs of some applications from the domain. CFD and numerical heat transfer are characterized by huge computational demand, far beyond the capabilities of current computers for certain use cases. Not only computing power, but also advanced networking and data storage systems are needed for these types of simulations. Correct parallel implementation of CFD techniques requires solution of nontrivial numerical problems.

This Virtual Team aimed to establish a fire and smoke simulation Virtual Research Community on EGI by

- Porting three types of parallel implementations of the FDS application to the European Grid Infrastructure.
- Identifying user communities for the ported application.
- Providing support for the communities to use the FDS application on EGI.
- Further developing the FDS application based on the feedback from the users.

The expected output of this project was:

- Parallel implementations of the FDS application on the European Grid Infrastructure, together with guides for users and software administrators.
- A European community actively using the FDS application on EGI.
- Support services provided for the FDS user community in multiple NGIs.

The required output for the project was foreseen to be achieved by the following tasks:

1. Contacting fire simulation teams within the NGIs and collecting input requirements from them for the EGI-enabled version of the FDS application
2. Porting three implementations of the FDS application to EGI according to the users' requirements: MPI application; OpenMP application; combined MPI-OpenMP application
3. Implementing different simulation use cases on EGI with the FDS application; analysing the results with the SmokeView visualiser tool
4. Developing training for fire and smoke simulation communities
5. Developing a marketing and communication plan to undertake the outreach to the fire simulation community at their events

At the beginning of the VT project (VTP) we planned to solve problems regarding fire simulation of human structures using Fire Dynamics Simulator (FDS) application. Experts from Slovakia (the leader of the VT) already had some experiences with running FDS on local cluster of the Institute of Informatics of Slovak Academy of Sciences (IISAS) in Bratislava. Slovak NGI members work with IISAS, so user support was provided to fire simulation experts. Right after the project started the VT members held a teleconference. As the result of this teleconference the scope and goals of the VTP have been broadened in order to take into consideration experiences of project partners from Portugal and Spain. Various kinds of fire simulations and applications tests were considered, dealing with tunnel fires, building fires, forest fires etc.

Re-specified VTP goals: After the re-scoping the Virtual Team aimed to

- Establish a Fire and smoke simulation Virtual Research Community on EGI and

- Specify requirements of integrating fire simulation applications with the EGI grid/cloud infrastructures
- Identify partners and roles that may lead to a consortium aiming to apply for funded EU project.

Different kind of fire simulations were considered, such as tunnel fires, building fires, as well as other fire simulation applications tests for other types of fires, e.g. forest fires. Fire and smoke simulation can be a priority topic in Mediterranean countries.

One of the main goals of EGI-InSPIRE is to collect user requirements and provide support for current and potential new user communities. In the scope of the VTP, we have prepared a questionnaire "Requirements of Fire simulation applications towards EGI grid environment". The survey should help us understand:

1. What are the key applications, software tools in the fire and smoke simulation area
2. Which one of these would benefit from integration with EGI/NGI services. We believe an application can benefit from EGI if it has at least one of these characteristics:
   - compute intensive (to run on a single machine/cluster)
   - data intensive (to run on a single machine/cluster)
   - needs to connect to databases or other resources at multiple sites
   - require hosting on remote resources (to provide a service for a large community)
   - require user friendly web interfaces (portal interfaces)
3. Who are using and who are supporting/developing these applications/tools now
4. Who and how many potential users are 'out there' who do not use the software because of the current limitations in it that EGI integration could resolve.
5. Would they use any third party app if that was integrated with EGI. If so, which one are these? The same that they list in point 1?

If we know the answers to these questions, then we could define the 'portfolio of smoke and fire simulation services' that is needed on EGI to attract and support scientific communities from this field. The questionnaire has been sent to fire simulation experts from 12 countries (Spain, Portugal, France, Greece, Czech Republic, Slovakia, Belgium, United Kingdom, Italy, Croatia, Romania, Poland)..

The VT received response for the questionnaire from 3 countries (Spain: Universitat Autònoma de Barcelona; Portugal: Forest Fire Research Centre, Coimbra, Universidade do Minho; Slovakia: Institute of Informatics, Slovak Academy of Sciences, Bratislava). The completed questionnaires are in the annex of this report, while a summary and analysis of these are available at https://wiki.egi.eu/wiki/Fire_Simulations_Requirements, and are provided later in the report (Section 2 and 3). Based on the received answers we have collected requirements and experiences of smoke and fire simulation scientists - what is needed on the part of EGI in order to help the scientific community from this field.

2 ANALYSIS OF SURVEY ANSWERS

A survey revealed a number of fire/smoke modelling applications. All of these applications already run on individual clusters, and tests for running some of them on grids have been done in the past or are still ongoing.

Key applications and software tools in the fire and smoke simulation area:

- Firestation
- pFirestation
- gFirestation
- Wind simulation
- FARSITE
- Fire Dynamics Simulator (FDS)
- Two-stage fire prediction scheme

The applications that will benefit from integration with EGI/NGI services:

1. *FDS*
   - compute intensive
   - data intensive
2. *Firestation*
   - compute intensive
   - requires user friendly web interfaces (portal interfaces)
3. *Two-stage fire prediction scheme*
   - needs to connect to databases or other resources at multiple sites
   - requires hosting on remote resources (to provide a service for a large community)
   - requires user friendly web interfaces (portal interfaces)

Types of applications used in fire simulation community:

- Pure batch mode
- Parallel applications with lots of communication
- Only transfer of input/output data to computer
- Frequent but small communication between computers
- Coupled applications (applications such as independent fluid structure)
- Parallel applications with low communication

Compilers needed to build the applications:

- gcc
- gfortran
- f90/95

More detailed requirements and information can be found in attached questionnaires, and may be further précised in future.

3 FIRE SIMULATION AREAS

Fire simulation researchers in Slovakia simulate fire in human-built structures (tunnel, house, room, car, car garage). Simulations using FDS are sufficiently accurate, so that on national level, there is a high level of acceptance of the research results. As an example, they were asked by the highest fire authorities to prepare a computer simulation of family house fire for court of law (justice) purposes.

A forest fire, similarly as other natural hazards, is a quite important problem that every year causes significant damages around the world. This kind of hazard provokes substantial losses from the ecological, economical, social and human point of view. Therefore, a quick response when an emergency occurs is crucial to minimize its effects. In this context, the accurate prediction of the propagation of forest fire is critical in order to use the available resources to fight against the fire in the most efficient way. The problem of predicting a forest fire is studied by the Spanish and Portuguese project partners.

3.1 Fire Dynamics Simulator (FDS)

Area leader: Slovakia

Fire Dynamics Simulator (FDS) is a model developed by National Institute for Standard and Technology (NIST, USA) that simulates fire and predicts its effects; in the package is included Smokeview, a graphical post-processor useful for FDS results analysis. FDS solves numerically a form of Navier-Stokes equations appropriate for low-speed, thermally-driven flow with an emphasis on smoke and heat transport of fire. FDS allows modelling of fire starting from a database of standard materials distributed with the program that can be modified by the user to introduce new materials, defined with their chemical and physical characteristics and with data obtained from experimental fire tests. Dynamics of fire is simulated through the parameters that characterize each material present in the simulation domain, each one with its characteristics of in-flammability and combustion or its reaction to fire. FDS can calculate and return as output, once the simulation has been set-up to effectively calculate the quantities of interest, all the values of the variables, both scalar and vectorial, computed in every cell of the domain and useful to understand the fire and to analyse its effects (concentrations of chemical species, distributions of temperature / pressures / gas velocities / smokes, visibility, ...). FDS can be used to model the following phenomena: Low speed transport of heat and combustion products (mainly smoke) from fire; Heat transfer between the gas and solid surfaces; Pyrolysis; Fire growth; Flame spread; Activation of sprinklers and heat detectors; Fire suppression by sprinklers. FDS is widely used by fire safety professionals. One of the applications is a simulation of fire and smoke transfer in human structures, for example for design of smoke control systems and sprinkler activation studies. FDS was also used in numerous fire reconstructions including the investigation of the World Trade Centre disaster.

FDS simulations with a real scenario represent long-time, computational intensive and memory consuming jobs. In order to find the programming model exhibiting the best performance and producing authentic true results for a given fire scenario, a great number of experiments, i.e. simulation runs with various input parameters, need to be fulfilled.

Unsuppressed tunnel fires involving Heavy Goods Vehicles (HGVs) can be extremely large. The simulation of an unsuppressed fire, which could not be conducted in the tunnel, provided a means of comparing conditions in a tunnel fire with and without an active suppression system. The tunnel fire tests (described in literature) showed that very large fires (20 to 60 MW) involving HGV fuel loads create highly turbulent conditions that vary from test to test, even for apparently similar test conditions. More information: http://www.cfd-online.com/Wiki/FDS, http://fire.nist.gov/fds

Fire Scenario:

In order to keep the computational domain within reasonable limits, only the 180m long test tunnel was included in the simulation (in the first phase). Basic layout of instrumentation in the test tunnel: We have constructed a two-lane road tunnel model with dimensions 10m x 180m x 7m with two fans located on the tunnel ceiling at the distance 50m and 140m from the left entrance of the tunnel. The fire source in the simulation was represented by burning of a flammable liquid in a pool with dimensions 2 x 3m placed in the distance 92m from the left entrance of the tunnel, 1.1m above the floor level. During the simulation, the fire did not spread along the tunnel; no other flammable obstacles were included in the simulation. The total duration of the simulation was 150s. The initial air temperature in the whole tunnel was set to 20°C. Various control devices were installed inside the tunnel in order to record mean values of gas phase quantities (soot volume fraction, visibility, temperature and carbon monoxide mass fraction) inside small testing cube-like volumes placed under the ceiling of the tunnel and at the level of human head The slices of gas temperatures, oxygen and carbon monoxide mass fractions were also recorded for several planes. The wall temperature of the tunnel ceiling was detected above the fire.

Input data required to run the model:

All of the input parameters required by FDS to describe a particular scenario are conveyed via one or two text files created by the user. These files contain information about the numerical grid, ambient environment, geometry of the problem modelled, material properties, boundary conditions and the fire itself. The input file should also contain information about the desired output quantities. All blockages in the FDS model have to conform to the computational mesh. For this reason, it is usually not possible for the FDS model geometry to be exactly the same as that of the physical model (unless very fine mesh is used). The irregular edge appearance could have been reduced by selecting a smaller cell dimension, but with a resulting rise in the number of cells and, hence, increased computational time. The grid cells can either be uniform in size (default mode) or they can be stretched in one or two of the three coordinate directions. It is possible to use more than one rectangular mesh in a calculation. This allows creation of an efficient computational domain for geometries which cannot be easily fitted into a single rectangular grid. It also allows using regions with different grid resolutions within one computational domain. Both the grid stretching and the use of multiple meshes allow the user to apply better grid resolutions in critical areas (e.g. near the fire) without unnecessarily increasing the demand for computational power by applying fine mesh to the entire computational domain. The use of multiple meshes is also required when an FDS simulation is to be run in parallel processing on more than one computer. The parallelization requires a decomposition of computational domain into computational meshes, which affect the simulation outputs. In order to use MPI version of FDS, computational domain must be decomposed into meshes and each one of them can be assigned to specific MPI process. Velocity values at the mesh boundaries are then averaged in order to maintain stability.

FDS source code has been designed to provide 4 programming models:

1. Sequential for a single CPU

2. MPI parallel – applied on distributed memory systems: To allow executing FDS as a single parallel job on a compute cluster the MPI (Message-Passing Interface) is used. The main strategy is, that the input computational domain is broken up into multiple meshes, and then the flow field in each sub-mesh is computed as an individual MPI process on the individual computer. MPI handles the transfer of information between these processes. Usually, each mesh is assigned to its own MPI process, but it is although possible to assign multiple meshes to a single process. In this way, large meshes can be computed on dedicated processors, while smaller meshes can be grouped together in one process running on a single processor.

3. OpenMP multi-threading – applied on shared memory systems: FDS supports multi-threading implemented through the OpenMP library, which enables to employ all available processors/cores on a given machine. The number of threads need to be specified by the environment variable "OMP_NUM_THREADS".

4. Combined model MPI and OpenMP which can be applied, for example, on multi-core cluster: The combination of MPI and OpenMP approaches enables a two-level parallelization: at first the computational domain is decomposed into meshes for distributed memory, and then, within each mesh the multi-threading approach on some selected code regions is applied for shared memory.

Realization of the FDS simulation on the cluster:

Several smoke transfer simulations in a road tunnel were performed using FDS and the impact of mesh decomposition of computational domain on the simulation results was under investigation. Simulations led to realistic smoke transfer behaviour and confirmed that parallel version of FDS is able to provide reasonable results.

- *FDS compiling and building*:
  - Simulations were realized using the FDS version 5.5.3
  - Compilers GNU v.4.1.2 and 4.4.0 (gcc, gfortran, OpenMP) and the Open MPI v.1.4. were used.
- *Simulation results*:
  The main goal of this work was to investigate the impact of partitioning the computational domain on the reliability of simulation results, and along with we struggled to find a FDS scenario exhibiting the best performance.
- *Simulation support*:
  In order to automate the process of a great number of simulation runs with a minimum of human intervention, for each of the configurations we have developed a fds manager script (written in Shell) which is responsible for the completion of the following actions:
  1. It accepts and checks the input arguments specifying the FDS input file, and the required cluster configuration (the number of nodes, the number of cores, and eventually the number of MPI processes and number of threads). Arguments are passed to the subsequent operations.
  2. It produces the corresponding job submission script which serves as an input to the Portable Batch System (PBS).
  3. It provides for the execution of the FDS simulation using the previously generated job submission script.
  To submit and start a batch job on a compute cluster the qsub command is used, to which the job submission script is passed as argument. A number of options on qsub allow the specification of attributes which may affect the behaviour of the job execution. When running the pure MPI version of the FDS, MPI processes are allocated by the (default) slots strategy, where one MPI process is mapped per processor-core. When running the hybrid MPI+OpenMP version, MPI processes are enforced to be allocated so, that each MPI process is mapped per node which has a sufficient number of cores available, free allocated cores are reserved to fork the OpenMP threads.

Requirements for running FDS simulations on the Grid:

- *Hardware*:

- high-performance, reliable and failure-free computing resources involving a sufficient amount of memory
- a storage resource to place output results
- *Installed software*:
  - Fortran 90 and C compilers including OpenMP library
  - MPI and MPI-Start
- *Grid middleware functionalities*:
  - the proxy credential delegation and renewal
  - the submission of parallel MPI and OpenMP jobs
  - monitoring the status of the submitted job
  - the job cancellation
  - the job output retrieval
  - optionally, the job perusal (real time output retrieval) enabling the inspecting the job output in real time
  - the basic data management services

In September 2012, Slovak NGI at IISAS planned to install (in the scope of SIVVP - see section 5.2) a new cluster, which will in part be provided for EGI. The above described tunnel fire simulation is planned to be run on it, too. We had problems with HW devices in SIVVP cluster, so its start of usage was delayed. We hope that our new resource will be integrated into EGI in October 2012. And as it runs SL6, we are still waiting for MPI in EMI-2 (UMD-2) and only then the FDS application will be running in its full extent. Gridification and test running were done in the scope of VTP. The FDS application is included in EGI AppDB.

Achievements:

All FDS models were compiled and executed on EGI infrastructure using Slovak NGI resources.

The sequential and OpenMP model were carried out within the "esr" and "voce", and parallel MPI models were carried out within the "mpi-kickstart" VO (only this VO provided the environment for executing MPI applications).

The submission of jobs was realized through the middleware EMI 1 (gLite) and the tool MPI-Start.

FDS models were successfully verified using as input simple fire scenarios in order to obtain results in a reasonable time period. The FDS application is included in EGI AppDB.

The FDS simulation of a real fire scenario represents a long-time job (the order of several days) which claims special conditions on the computing resource.

3.2 Forest fire simulation

Area leader: Portugal

An OGC-WS Framework to Run FireStation on the Grid:

Current geo-referenced computing and communication technologies are becoming robust and mature to be accepted as decision aid tools by the Civil Protection (CP) authorities in emergency calls. The CP authorities need fast and reliable tools to remotely access real time geo-referenced data and to estimate the risk spread well in advance, in order to lead the operational procedures with the active involvement of public and private organizations.

A forest fire is a typical example of a critical environmental disaster in Mediterranean countries, whose efficient fight requires the best use of the available computational and communication power.

FireStation is a standalone desktop simulator of the fire growth over complex topography, developed at University of Coimbra, which has been tested in the past by Portuguese fire brigades.

CROSS-Fire is a research project, funded by the Portuguese National Grid Initiative (NGI) and led by University of Minho in cooperation with the FireStation original team at University of Coimbra and the Portuguese CP Authority. It aims to develop a grid-based framework as a risk management decision support system for the CP authorities, using forest fires as the main case study.

The CROSS-Fire project aims to develop a grid-based framework as a risk management decision support system for the civil protection authorities, using forest fires as the main case study and FireStation as the standalone CAD application that simulates the fire spread over complex topography.

The CROSS-Fire initial tasks have been focussed on the development of a parallel version of the fire simulator engine (P-FireStation), and its porting into the EGEE grid environment (G-FireStation). The main contribution lies on the definition of an OGC-WS framework to enable a basic set of standard geospatial services that will allow FireStation to be interoperable with standard-based Spatial Data Infrastructures.

The P-FireStation version explores the inherent parallel environment offered by clusters at each site of the EGEE grid, to support larger data sets and to improve the accuracy of the predictions. This parallel version relies on the MPI protocol and supports larger data sets taking advantage of the MPI parallel I/O facilities.

The G-FireStation version integrates EGEE grid facilities, namely the gLite data management services and tools to access data, the AMGA gLite grid metadata catalogue to manage the simulation I/O data, and the WatchDog tool to monitor and provide data for the interactive control of simulations. The new approach will make G-FireStation more interoperable, allowing to access different data providers and publishing output data for further processing, following the guidelines of the EC CYCLOPS project.

The Open Geospatial Consortium Web Services (OGC-WS) are being adopted worldwide, as the technology to support the development of complex distributed applications over grid platforms, and to deal with data from many different sources, including live sensors.

The EC FP6 CYCLOPS project (Cyber Infrastructure for Civil protection Operative Procedures) adopted the EGEE grid platform to support the CP operative procedures aiming to refine EGEE standards and protocols within CP specific requirements. It has proposed an architectural framework built on top of the EGEE middleware gLite, to make use of the grid processing and storage capabilities, which include a set of geospatial services accessible through standard interface services. The approach has been tested and validated through the implementation of a grid-enabled prototype application, G-RISICO, for wild fire risk assessment, used by the CP authorities in Italy.

The FireStation is an integrated system aimed at the simulation of fire growth over a complex topography. It has been used by Portuguese fire brigades under the CP authorities in the past decade. The original standalone FireStation version integrated a module for wind field generation, as well as a module for the computation of the FireWeather Index, from the Canadian system. The software was originally written in MDL, a specific C language of Microstation with built-in subroutines for the design of window-based interfaces and generation of visualization elements both in 2D and 3D space. For the CROSSFire project, the frontend modules were separated from the backend ones, and a more accurate wind field estimator (using a Navier-Stokes solver called CANYON) was added. The main efforts in this project were to parallelize these FireStation backend modules and to port the resulting P-FireStation to the grid.

3.3 Forest fire simulation with two-stage prediction methodology

Area leader: Spain

Applying Grid and urgent computing solutions to forest fire propagation prediction:

Several models have been developed by researchers from different fields to represent and predict the fire propagation. These models require input parameters including terrain topography, vegetation

conditions and meteorological variables to produce precise and accurate predictions. Some of these parameters are uniform and static, but others have a spatial distribution and a temporal variation, and are difficult to know precisely their values beforehand.

So, a two-stage prediction methodology was developed. This methodology calibrates the parameters by applying artificial intelligence evolutionary techniques. In the calibration stage the actual propagation of the fire is observed and then the input parameters are calibrated. The values that best reproduce the actual propagation of the fire are used in the prediction stage.

A developed two-stage prediction methodology is independent from the propagation model and simulation kernel considered. The model itself appears as a black-box and the only point that must be considered are the input parameters required for each simulator. UAB (Universitat Autonoma de Barcelona, Spain) used FireSim and FarSite, and also have some experience with FireStation.

When the fire is burning time is critical and it is necessary to determine the future propagation of the fire beforehand. So, we face three main interrelated issues:

- The available time to provide a prediction is a critical point and seriously limits the number of iterations that can be executed on the calibration stage.
- The amount of computing resources determines the amount of simulations that can be simultaneously executed per iteration of the calibration stage.
- The quality of the prediction is directly related to the quality of the calibration, and the quality of the calibration depends on the number of iterations of the evolutionary algorithm and on the number of scenarios used per iteration.

Currently, the UAB team is working on the coupling of meteorological prediction models, such as WRF, and wind field models, such as Wind Ninja. These models become a key issue since the wind is a key factor on the prediction quality. However, such models are even more time consuming that the fire propagation model itself.

4 CONCLUSION AND NEXT STEPS

The VT collected information about fire and smoke simulation applications from those communities that could be reached through the NILs who responded to the VT project call. The response rate was lower than expected, resulting in a VT with members from three countries (Slovakia, Spain, and Portugal). While the survey that was setup by the VT reached researchers in 12 countries, responses came back only from those three that were directly involved in the VT. The low interest and response number indicate that either the size of the fire and smoke simulation research community is small, or that NGIs are not connected to fire and smoke simulation communities and cannot be used to reach such communities.

The information that the three survey responses provided highlight a number of applications with high level requirements for using EGI services. Based on the collected information and the described analysis the activities that started in the VT could continue in various ways.

4.1 IP projects

In the past Dr. Ladislav Hluchý (leader of Slovak NGI, IISAS Bratislava, Slovakia) initiated communication with partners from Spain and Portugal to prepare EU IP project proposal on fire simulation. Based on received information, 4 IP project proposals (with more than 50 potential collaborating partners) have already been submitted, all of the proposals received good rating (14 points), but the problem is the required budget. If project partners could in their proposals take into account the usage of EGI for their computations, the required budget might be smaller, leading to more successful proposals. Ladislav Hluchý will reopen negotiations with these partners with this suggestion.

4.2 SIVVP site for fire applications

In Slovakia, a national infrastructure project SIVVP (Slovenská Infraštruktúra pre VysokoVýkonné Počítanie - Slovak infrastructure for high performance and high throughput computing) runs between 2010 - 2015. In September 2012, the Slovakian NGI at IISAS planned to install a new cluster (in the scope of SIVVP) which will be connected to EGI. We had problems with HW devices in SIVVP cluster, so its start of usage was delayed. We hope that our new resource will be integrated into EGI in October 2012. And as it runs SL6, we are still waiting for MPI in EMI-2 (UMD-2) and only then the FDS application will be running in its full extent. The previously described tunnel fire simulation application will run on it. Gridification and test running were done in the scope of VTP.

This application with this site can be offered to fire and smoke simulation communities worldwide. The Slovakian NGI will provide support for the usage. Other applications that have been identified by the VT can also consider this site for use.

In section 3.1 described 4 types (sequential, MPI parallel, OpenMP, combined model) of test fire simulations using FDS run on EGI using resources of VO mpi-kickstart.egi.eu11, especially using resources of Slovak NGI included in VO mpi-kickstart.egi.eu11 . The FDS application is included in EGI AppDB.

4.3 Communication with NGIs

Before the VTP, EGI.eu sent an invitation to every NGI International Liaison to join the VTP. None of them accepted the invitation and it seems that NGIs could not adequately inform the fire simulation experts in their countries on the project proposal. Only after a direct invitation was sent by the Slovakian NGI to research groups they knew from earlier groups from Portugal and Spain joint the VT

through their NILs. Fire simulation groups exist in at least 12 countries. Invitation to collaboration through NGIs representatives will be renewed. We hope that after direct communication between NGIs and local domain experts (in respective countries) the aspiration for closer collaboration of fire simulation community will increase, leading in optimal case to a new Virtual organization and Virtual Research Community in the future. By our opinion, some of the experts from these countries could start to use grid infrastructure after they will receive more information and support from their National Grid Infrastructure providers. The Slovakian NGI will prepare a list of identified contacts within those NGIs and will connect them to their local NGIs to introduce them to each other and initiate local collaborations.

4.4 Application integration

One out of the three applications (FDS) that the VT identified as relevant applications for fire simulation has been integrated by EGI. The other two applications (Firestation and Two-stage fire prediction scheme) still need to be integrated with EGI through one of the grid middleware platforms or the federated cloud platform if EGI wishes to offer these to scientific communities. EGI could find effort for this integration using a new Virtual Team, through other EGI-InSPIRE mechanism (e.g. an NGI or a partner project that provides application integration services), or through a new EC project (See Section 4.1).

5 ANNEX

Questionnaire of the *VTP Fire and smoke simulation*
Requirements of Fire simulation applications towards EGI grid environment

The survey should help us understand:

a) What are the key applications, software tools in the fire and smoke simulation area?

b) Which one of these would benefit from integration with EGI/NGI services?

c) Who are using and who are supporting these applications/tools now?

d) Who and how many potential users are 'out there' who don't use the software because of the current EGI-specific limitations in it.

e) Would they use any third party application if that was integrated with EGI?

If we know the answers to these questions, then we could define the 'portfolio of smoke and fire simulation services' that is needed on EGI to benefit the scientific community from this field.

Grid Computing is seen as a means to provide access to distributed compute resources, instruments and databases as well as know-how. We must gain an understanding of the user community in order to get a precise picture of the requirements of the people and groups working in the field. [2]

The evolution of the Grid within the research community initially came from a 'globalization' of scientific practices and collaborations. This included:

- Information exchange within globally dispersed groups.
- Accessing and aggregating remote computing resources.
- Storing of data in large distributed data stores.

This led to the definition of the Grid, as "resource sharing and coordinated problem solving in dynamic, multi-institutional virtual organizations (VO's)." [1]

Virtual research communities (VRCs) are groups of like-minded individuals organised by discipline or computational model. VRCs typically have an established presence in their field (for example an ESFRI project, EIROFORUM laboratory or national research structure) and represent a well-defined scientific or research community. They can include one or more virtual organisations and act as the main communication channel between the researchers they represent and EGI. VRCs provide EGI with their technical and service requirements, which are then fed into the overall development of the infrastructure as a consumer-driven resource.

Virtual organisations (VOs) are groups of researchers with similar scientific interests and requirements, who are able to work collaboratively with other members and/or share resources (e.g. data, software, expertise, CPU, storage space), regardless of geographical location. Researchers must join a VO in order to use grid computing resources provided by EGI.

European Grid Initiative (EGI) / National Grid Initiatives (NGIs)

The **EGI Applications Database** (AppDB) stores information about grid-based computing tools for scientists to use. The scope of the database embraces all scientific fields, from resources to simulate exotic excitation modes in physics, to applications for complex protein sequences analysis.
The applications filed in AppDB are finished products, ready to be used. Using applications already enabled for EGI means that you, as a scientist, don't have to spend your research time adapting software to the grid.

Science gateways for users: A web-based science gateway is a portal that allows users to interact with grid resources via a web browser. Web-based gateways provide simple mechanisms to access tools and services from different platforms without software installation (Science Gateway to IGI [3], …)

A desktop application-based gateway is a client program installed directly on the end-users' workstations, with embedded capabilities to access grid resources and services (g-Eclipse - gLite and other grid middlewares client)

[1] I. Foster, C. Kesselman, S. Tuecke: 'The Anatomy of the Grid: Enabling Scalable Virtual Organizations' International J. Supercomputer Applications, 15(3), 2001.

[2] P. Lindner, T. Beisel, M. M. Resch, T. Imamura, R. Menday, P. Wieder, D. Erwin: GRIDWELTEN: User Requirements and Environments for GRID-Computing, 2003

[3] Science Gateway to IGI

5.1 Fire Dynamics Simulator (FDS)

Name:	L.Hluchy, P.Weisenpacher, M.Dobrucky
Affiliation:	UI SAV, Bratislava, SK

I. Domain specific questions:

1) What are the key applications, software tools in the fire and smoke simulation area? Which of the following applications do you use?

- ☒ Fire Dynamics Simulator (FDS)
- ☒ Smokeview
- ☒ WUI (Wildland Urban Interface) Fire Dynamics Simulator (WFDS)
- ☐ Fluent
- ☐ StarCD
- ☐ Smartfire
- ☒ PyroSim
- ☒ FARSITE
- ☐ Prometheus
- ☐ Canadian wildland fire growth model
- ☐ Coupled Atmosphere-Wildland Fire-Environment (CAWFE) model
- ☐ Coupled Atmosphere-Wildland Fire Large Eddy Simulation
- ☐ FIRETEC
- ☐ FIRESTAR
- ☐ WRF-Fire
- ☐ Weather Research and Forecasting model
- ☐ gWRF Grid enabled the climate and weather simulation WRF service
- ☐ Firestation
- ☐ pFirestation
- ☐ gFirestation
- ☐ Other

If other, please write which:	

2) Do you develop your own application?

No   *(Please, select from drop-down menu)

3) Would you use any third party application if that was integrated with EGI?

If so, please write which:	

4) Which of your applications would benefit from integration with EGI/NGI services?

Please, indicate the application name:	FDS

In case of more applications, please answer the question 4) as well as questions of sections **II.** and **III.** in a new copy of this questionnaire.

The application will benefit from EGI, as it is:

- ☒ compute intensive (to run on a single machine)
- ☒ data intensive (to run on a single machine)
- ☐ needs to connect to databases or other resources at multiple sites
- ☐ require hosting on remote resources (to provide a service for a large community)
- ☐ require user friendly web interfaces (portal interfaces)

Who is supporting this application/tool now?

Name:	L.Hluchy, V. Sipkova, M.Dobrucky, V. Tran
Organization:	UI SAV, Bratislava, SK
NGI:	Slovak

II. Selected Application dependencies:

5) Which computing system do you use for running your application?

- ☒ desktop computer
- ☒ workstation
- ☒ compute cluster
- ☒ Grid infrastructure
- ☐ supercomputer
- ☐ GPGPU
- ☐ Other

If compute cluster, how large is your system in terms of nodes, cores and memory:	128 cores, 6 GB per core
If other, please write which:	

6) Which operating system your application runs on? (Please choose all that apply and provide a comment)

☒	Windows	
☒	Linux/Unix	
☐	MacOS	
☐	Other:	

7) Which compiler/version do you need to build your application?
(Please choose all that apply and provide a comment)

☐	f77	
☐	f90/95	
☒	gcc	
☐	g++	
☒	gfortran	
☐	java	
☒	Other: Make, Shell	

8) Are any special third party libraries/software required to run your application? If yes, which?

Yes   *(Please, select from drop-down menu)

If yes, please write which:	Open MPI

III. Make an appraisal of your requirements:

9) Type of application:

- ☒ Pure batch mode
- ☐ Workflow/piped applications (one run after the other)
- ☐ (Online) controlled applications
- ☐ Observed applications (online visualization of application)
- ☐ Steered applications (online visualization of applications with feedback)
- ☐ Coupled applications (applications such as independent fluid structure)
- ☐ Parallel applications with low communication
- ☒ Parallel applications with lots of communication
- ☐ Controlled distributed applications
- ☐ Observed distributed applications
- ☐ Steered distributed applications
- ☐ Others: vectorized application

10) Requirement type:

- ☒ Only transfer of input/output data to computer
- ☒ Seldom and low communication between the computers, as in online control
- ☐ Seldom but intensive communication between computers as in online visualization
- ☐ Frequent but small communication between computers
- ☐ Frequent and intensive communication between computers

11) How dependent are your programs on network latency?   *In which area should the latencies lie?*

- ☐ Strong
- ☐ Less strong
- ☒ Normal
- ☐ Less
- ☐ Not at all

- ☐ Up to 10 microseconds
- ☐ Between 10 and 100 microseconds
- ☐ Between 100 and 1000 microseconds
- ☐ Between 1 and 10 milliseconds
- ☐ Between 10 and 100 milliseconds

12) How strongly dependent are the programs on bandwidth?

In which range should the bandwidths lie?

- ☐ Strong
- ☐ Less strong
- ☒ Normal
- ☐ Less
- ☐ Not at all

- ☐ Up to 1 Mbit/sec
- ☐ Between 1 Mbit/s and 10 Mbit/s
- ☐ Between 10 Mbit/s and 100 Mbit/s
- ☐ Between 100 Mbit/s and 1 Gbit/s
- ☐ Over 1 Gbit/s

13) In which area does the size of the input data lie?   *In which area does the size of the output data lie?*

- ☒ Up to 1 MB
- ☐ Between 1 MB and 10 MB
- ☐ Between 10 MB and 100 MB
- ☐ Between 100 MB and 1 GB
- ☐ Over 1 GB

- ☐ Up to 1 MB
- ☐ Between 1 MB and 10 MB
- ☐ Between 10 MB and 100 MB
- ☐ Between 100 MB and 1 GB
- ☒ Over 1 GB

14) Does your application use parallel I/O?

No   *(Please, select from drop-down menu)

15) Is a post-processing of the generated output data necessary?

Never   *(Please, select from drop-down menu)

16) Do you use debugging tools on high performance systems for your application?

Never   *(Please, select from drop-down menu)

17) Do you use performance profiling and analysis tools on the High Performance Systems for your application?

Never   *(Please, select from drop-down menu)

IV.   When using a GRID you expect ... *Please, fill in if you have no experiences with grid/cluster computing*

18) ... your application will be easier to handle.

I totally agree   *(Please, select from drop-down menu)

19) ... that your application can be used completely without changes.

I totally agree   *(Please, select from drop-down menu)

20) ... that you need to write a Batch-job script in order to use your application in this environment.

I totally agree   *(Please, select from drop-down menu)

21) ... that you need to adapt a program by using a particular Grid tool , in order to use it.

I don`t agree at all   *(Please, select from drop-down menu)

22) ... *that you need to rewrite your application by using a specific Grid-API, in order to use it.*

I don't agree at all   *(Please, select from drop-down menu)

23) ... *that you have to login per session ...*

☒ once
☐ on every computer that you wish to use in the grid
☐ each institution that is involved in the grid
☐ on every computer at each institution that is involved in the grid

24) ... *as an interface ...*

☐ a graphical user interface
☒ a console based interface
☐ both
☐ none of the above

25) ... *to know where the applications that you want to use can be found.*

I totally agree   *(Please, select from drop-down menu)

26) ... *to have to know where your input and output data is found.*

I totally agree   *(Please, select from drop-down menu)

27) ... *that you can explicitly choose the resources (hardware and software), which you wish to use.*

I totally agree   *(Please, select from drop-down menu)

28) ... *to have to select these again each time you use the Grid resources*

I agree less   *(Please, select from drop-down menu)

29) ... *that the Grid automatically selects the best resource for you*

I totally agree   *(Please, select from drop-down menu)

30) ... *that the Grid offers both options (automatic and user controlled search for resources)*

I totally agree   *(Please, select from drop-down menu)

31) ... *that in the usage of different types of resources, your application has to be recompiled for each.*

Partly   *(Please, select from drop-down menu)

32) ... *that each application compiled by you must be transported to each resource.*

I totally agree   *(Please, select from drop-down menu)

33) ... *a debugging tool, which you can use in the Grid environment.*

I don't agree at all   *(Please, select from drop-down menu)

34) ... *Performance Profiling and Analysis Tools for the tuning of your application in this environment.*

Partly   *(Please, select from drop-down menu)

35) ... *that the above mentioned tools can be used in the same way as in non-Grid environments.*

I rather agree   *(Please, select from drop-down menu)

5.2 Forest fire simulation (António Pina, UMinho Portugal)

Name:	
Affiliation:	

I. Domain specific questions:

1) *What are the key applications, software tools in the fire and smoke simulation area?*
Which of the following applications do you use?

- [ ] Fire Dynamics Simulator (FDS)
- [ ] Smokeview
- [ ] WUI (Wildland Urban Interface) Fire Dynamics Simulator (WFDS)
- [ ] Fluent
- [ ] StarCD
- [ ] Smartfire
- [ ] PyroSim
- [ ] FARSITE
- [ ] Prometheus
- [ ] Canadian wildland fire growth model
- [ ] Coupled Atmosphere-Wildland Fire-Environment (CAWFE) model
- [ ] Coupled Atmosphere-Wildland Fire Large Eddy Simulation
- [ ] FIRETEC
- [ ] FIRESTAR
- [ ] WRF-Fire
- [ ] Weather Research and Forecasting model
- [ ] gWRF Grid enabled the climate and weather simulation WRF service
- [x] Firestation
- [x] pFirestation
- [x] gFirestation
- [x] Other

If other, please write which:	Wind simulation

2) *Do you develop your own application?*

    Yes   *(Please, select from drop-down menu)

3) *Would you use any third party application if that was integrated with EGI?*

If so, please write which:	

4) *Which of your applications would benefit from integration with EGI/NGI services?*

Please, indicate the application name:	Firestation

In case of more applications, please answer the question 4) as well as questions of sections **II.** and **III.** in a new copy of this questionnaire.

The application will benefit from EGI, as it is:

- [x] compute intensive (to run on a single machine)
- [ ] data intensive (to run on a single machine)
- [ ] needs to connect to databases or other resources at multiple sites
- [ ] require hosting on remote resources (to provide a service for a large community)
- [x] require user friendly web interfaces (portal interfaces)

Who is supporting this application/tool now?

Name:	Departamento Informática
Organization:	Universidade Minho
NGI:	IBERGRID

II. Selected Application dependencies:

5) Which computing system do you use for running your application?

- ☒ desktop computer
- ☐ workstation
- ☒ compute cluster
- ☒ Grid infrastructure
- ☐ supercomputer
- ☐ GPGPU
- ☐ Other

| If compute cluster, how large is your system in terms of nodes, cores and memory: | |
| If other, please write which: | |

6) Which operating system your application runs on? (Please choose all that apply and provide a comment)

☒	Windows	
☒	Linux/Unix	
☐	MacOS	
☐	Other:	

7) Which compiler/version do you need to build your application?
(Please choose all that apply and provide a comment)

☐	f77	
☐	f90/95	
☒	gcc	
☐	g++	
☒	gfortran	
☐	java	
☐	Other:	

8) Are any special third party libraries/software required to run your application? If yes, which?

No   *(Please, select from drop-down menu)

| If yes, please write which: | |

III. Make an appraisal of your requirements:

9) Type of application:

- ☒ Pure batch mode
- ☐ Workflow/piped applications (one run after the other)
- ☐ (Online) controlled applications
- ☐ Observed applications (online visualization of application)
- ☐ Steered applications (online visualization of applications with feedback)
- ☐ Coupled applications (applications such as independent fluid structure)
- ☐ Parallel applications with low communication
- ☒ Parallel applications with lots of communication
- ☐ Controlled distributed applications
- ☐ Observed distributed applications
- ☐ Steered distributed applications
- ☐ Others: vectorized application

10) Requirement type:

- ☒ Only transfer of input/output data to computer
- ☐ Seldom and low communication between the computers, as in online control
- ☐ Seldom but intensive communication between computers as in online visualization
- ☒ Frequent but small communication between computers
- ☐ Frequent and intensive communication between computers

11) How dependent are your programs on network latency?   **In which area should the latencies lie?**

☐ Strong		☐	Up to 10 microseconds
☐ Less strong		☐	Between 10 and 100 microseconds
☒ Normal		☐	Between 100 and 1000 microseconds
☐ Less		☐	Between 1 and 10 milliseconds
☐ Not at all		☐	Between 10 and 100 milliseconds

12) How strongly dependent are the programs on bandwidth?

            In which range should the bandwidths lie?

☐ Strong		☐	Up to 1 Mbit/sec
☐ Less strong		☐	Between 1 Mbit/s and 10 Mbit/s
☐ Normal		☐	Between 10 Mbit/s and 100 Mbit/s
☐ Less		☐	Between 100 Mbit/s and 1 Gbit/s
☐ Not at all		☒	Over 1 Gbit/s

13) In which area does the size of the input data lie?   **In which area does the size of the output data lie?**

☐ Up to 1 MB		☐	Up to 1 MB
☐ Between 1 MB and 10 MB		☐	Between 1 MB and 10 MB
☐ Between 10 MB and 100 MB		☐	Between 10 MB and 100 MB
☒ Between 100 MB and 1 GB		☒	Between 100 MB and 1 GB
☐ Over 1 GB		☐	Over 1 GB

14) Does your application use parallel I/O?

Yes   *(Please, select from drop-down menu)

15) Is a post-processing of the generated output data necessary?

Sometimes   *(Please, select from drop-down menu)

16) Do you use debugging tools on high performance systems for your application?

Sometimes   *(Please, select from drop-down menu)

17) Do you use performance profiling and analysis tools on the High Performance Systems for your application?

Sometimes   *(Please, select from drop-down menu)

IV.   **When using a GRID you expect …** *Please, fill in if you have no experiences with grid/cluster computing*

18) … your application will be easier to handle.

Partly   *(Please, select from drop-down menu)

19) … that your application can be used completely without changes.

Partly   *(Please, select from drop-down menu)

20) … that you need to write a Batch-job script in order to use your application in this environment.

I totally agree   *(Please, select from drop-down menu)

21) … that you need to adapt a program by using a particular Grid tool , in order to use it.

Partly   *(Please, select from drop-down menu)

22) ... *that you need to rewrite your application by using a specific Grid-API, in order to use it.*

I agree less   *(Please, select from drop-down menu)

23) ... *that you have to login per session ...*

- ☐ once
- ☐ on every computer that you wish to use in the grid
- ☒ each institution that is involved in the grid
- ☐ on every computer at each institution that is involved in the grid

24) ... *as an interface ...*

- ☐ a graphical user interface
- ☐ a console based interface
- ☒ both
- ☐ none of the above

25) ... *to know where the applications that you want to use can be found.*

Partly   *(Please, select from drop-down menu)

26) ... *to have to know where your input and output data is found.*

I totally agree   *(Please, select from drop-down menu)

27) ... *that you can explicitly choose the resources (hardware and software), which you wish to use.*

Partly   *(Please, select from drop-down menu)

28) ... *to have to select these again each time you use the Grid resources*

Partly   *(Please, select from drop-down menu)

29) ... *that the Grid automatically selects the best resource for you*

Partly   *(Please, select from drop-down menu)

30) ... *that the Grid offers both options (automatic and user controlled search for resources)*

Partly   *(Please, select from drop-down menu)

31) ... *that in the usage of different types of resources, your application has to be recompiled for each.*

Partly   *(Please, select from drop-down menu)

32) ... *that each application compiled by you must be transported to each resource.*

Partly   *(Please, select from drop-down menu)

33) ... *a debugging tool, which you can use in the Grid environment.*

Partly   *(Please, select from drop-down menu)

34) ... *Performance Profiling and Analysis Tools for the tuning of your application in this environment.*

Partly   *(Please, select from drop-down menu)

35) ... *that the above mentioned tools can be used in the same way as in non-Grid environments.*

Partly   *(Please, select from drop-down menu)

5.3 Forest fire simulation with two-stage prediction methodology (Ana Cortés)

Name:	
Affiliation:	

I. Domain specific questions:

1) What are the key applications, software tools in the fire and smoke simulation area? Which of the following applications do you use?

- ☐ Fire Dynamics Simulator (FDS)
- ☐ Smokeview
- ☐ WUI (Wildland Urban Interface) Fire Dynamics Simulator (WFDS)
- ☐ Fluent
- ☐ StarCD
- ☐ Smartfire
- ☐ PyroSim
- ☒ FARSITE
- ☐ Prometheus
- ☐ Canadian wildland fire growth model
- ☐ Coupled Atmosphere-Wildland Fire-Environment (CAWFE) model
- ☐ Coupled Atmosphere-Wildland Fire Large Eddy Simulation
- ☐ FIRETEC
- ☐ FIRESTAR
- ☐ WRF-Fire
- ☐ Weather Research and Forecasting model
- ☐ gWRF Grid enabled the climate and weather simulation WRF service
- ☒ Firestation
- ☐ pFirestation
- ☐ gFirestation
- ☒ Other

If other, please write which:

2) Do you develop your own application?
   Yes   *(Please, select from drop-down menu)

3) Would you use any third party application if that was integrated with EGI?
   If so, please write which: Any fire simulator

4) Which of your applications would benefit from integration with EGI/NGI services?
   Please, indicate the application name: two-stage prediction scheme

   In case of more applications, please answer the question 4) as well as questions of sections **II.** and **III.** in a new copy of this questionnaire.

   The application will benefit from EGI, as it is:
   - ☐ compute intensive (to run on a single machine)
   - ☐ data intensive (to run on a single machine)
   - ☒ needs to connect to databases or other resources at multiple sites
   - ☒ require hosting on remote resources (to provide a service for a large community)
   - ☒ require user friendly web interfaces (portal interfaces)

   Who is supporting this application/tool now?

Name:	Ana Cortés Fité
Organization:	Computer Arquitectures and Operating Systems Department, Universitat Autònoma de Barcelona
NGI:	

II. Selected Application dependencies:

5) Which computing system do you use for running your application?

☐ desktop computer
☐ workstation
☒ compute cluster
☐ Grid infrastructure
☐ supercomputer
☐ GPGPU
☐ Other

If compute cluster, how large is your system in terms of nodes, cores and memory:	32 nodes with 4 cores each and 12 GB per node
If other, please write which:	

6) Which operating system your application runs on? (Please choose all that apply and provide a comment)

☐	Windows	
☒	Linux/Unix	
☐	MacOS	
☐	Other:	

7) Which compiler/version do you need to build your application?
(Please choose all that apply and provide a comment)

☐	f77	
☒	f90/95	
☒	gcc	
☐	g++	
☐	gfortran	
☐	java	
☐	Other:	

8) Are any special third party libraries/software required to run your application? If yes, which?

Yes   *(Please, select from drop-down menu)

If yes, please write which:	* Boost 1.4.6:
	  * NetCDF 4.1.1
	    * may need HDF5 support (sometimes easier to build on win32)
	  * GDAL 1.8.1
	    * NetCDF support built in
	    * PROJ.4 support
	  * zlib 1.2.3
	  * libcurl 7.21.4
	  * Qt 4.7.4
	    * QtGui
	    * QtCore
	  third-party -> third party needs, zlib, EasyBMP, and tnt

III. Make an appraisal of your requirements:

9) Type of application:

- ☐ Pure batch mode
- ☐ Workflow/piped applications (one run after the other)
- ☐ (Online) controlled applications
- ☐ Observed applications (online visualization of application)
- ☐ Steered applications (online visualization of applications with feedback)
- ☒ Coupled applications (applications such as independent fluid structure)
- ☐ Parallel applications with low communication
- ☐ Parallel applications with lots of communication
- ☐ Controlled distributed applications
- ☐ Observed distributed applications
- ☐ Steered distributed applications
- ☐ Others: vectorized application

10) Requirement type:

- ☐ Only transfer of input/output data to computer
- ☐ Seldom and low communication between the computers, as in online control
- ☐ Seldom but intensive communication between computers as in online visualization
- ☒ Frequent but small communication between computers
- ☐ Frequent and intensive communication between computers

11) How dependent are your programs on network latency? — **In which area should the latencies lie?**

- ☐ Strong
- ☐ Less strong
- ☒ Normal
- ☐ Less
- ☐ Not at all

- ☐ Up to 10 microseconds
- ☐ Between 10 and 100 microseconds
- ☒ Between 100 and 1000 microseconds
- ☐ Between 1 and 10 milliseconds
- ☐ Between 10 and 100 milliseconds

12) How strongly dependent are the programs on bandwidth? — **In which range should the bandwidths lie?**

- ☐ Strong
- ☐ Less strong
- ☒ Normal
- ☐ Less
- ☐ Not at all

- ☐ Up to 1 Mbit/sec
- ☐ Between 1 Mbit/s and 10 Mbit/s
- ☐ Between 10 Mbit/s and 100 Mbit/s
- ☒ Between 100 Mbit/s and 1 Gbit/s
- ☐ Over 1 Gbit/s

13) In which area does the size of the input data lie? — **In which area does the size of the output data lie?**

- ☐ Up to 1 MB
- ☒ Between 1 MB and 10 MB
- ☐ Between 10 MB and 100 MB
- ☐ Between 100 MB and 1 GB
- ☐ Over 1 GB

- ☐ Up to 1 MB
- ☐ Between 1 MB and 10 MB
- ☐ Between 10 MB and 100 MB
- ☐ Between 100 MB and 1 GB
- ☒ Over 1 GB

14) Does your application use parallel I/O?

No   *(Please, select from drop-down menu)

15) Is a post-processing of the generated output data necessary?

Always   *(Please, select from drop-down menu)

16) Do you use debugging tools on high performance systems for your application?

Sometimes   *(Please, select from drop-down menu)

17) Do you use performance profiling and analysis tools on the High Performance Systems for your application?

Sometimes   *(Please, select from drop-down menu)