A Microservices Approach for Parallel Applications Design: A Case Study for CFD Simulation in Geoscience Domain

Alexey Cheptsov
High Performance Computing Center Stuttgart,
University of Stuttgart
Stuttgart, Germany
e-mail: cheptsov@hlrs.de

Oleg Beljaev
Donetsk National Technical University
Computer Science Department
Pokrowsk, Ukraine
e-mail: oleg69@ukr.net

***Abstract*—Current geoscience applications face two major challenges – the integration with numerous diverse sensor devices and the use in real-time use case scenarios. Whilst the challenge of service integration is addressed by the concept of Cyber-Physical systems, which aims to incorporate sensor data in application workflows, the usage of High Performance Computers helps minimize the execution time to fulfill the real time scenarios requirements. However, the existing programming models do not allow scientific workflows to take advantage of both technologies simultaneously. This paper contribution offers an approach to encapsulation of workflow-based applications into services, which are flexible enough to run on heterogeneous, distributed infrastructures spanning over both industrial sensor services and parallel computing systems. The approach is demonstrated on a computational fluid dynamics simulation study of aerodynamic processes in complex underground mine ventilation networks.**

Keywords-Dynamic Simulation; Computational Fluid Dynamics; Microservices Architecture; Workflows; ChEESE.

I. INTRODUCTION

Geoscience applications rely largely on simulation, which is used to retrieve and investigate the state of the targeted complex dynamic systems and also to predict their behavior under certain conditions in the future. One of the typical geoscience simulation tasks is served by Computational Fluid Dynamics (CFD) – a technique that is used to study the behavior of liquids and gases in complex environments. The CFD technique can be used to model many safety-critical processes, such as, for example, the propagation of waves as a result of tsunamis, the distribution of volcanic plumes after an eruption, or the distribution of air and hazardous gases in underground ventilation objects like coalmines. As all the other CFD applications, these studies are based on complex mathematical models (like Navier-Stokes equation), which generally create a good deal of uncertainty for the simulation results and also require computationally expensive solution methods.

In practice, geoscience applications are often organized in workflows with several interconnected components, each implementing a specific part of the application logic and running on a dedicated resource of the distributed system. The computationally intensive parts of the workflow are usually executed on parallel High Performance Computing (HPC) resources, whilst the data acquisition happens on the sensor nodes. However, the workflow approach has several limitations. Firstly, the workflow management software requires quite a rich functionality of resource management, application scheduling, monitoring and other middleware that is related to the workflows execution (like Pegasus, as described by Chang et al. in [3]), which are difficult to provision on the production HPC systems. Secondly, the workflow-based specification of applications requires quite an extensive metadata schema, which might require substantial change from one execution scenario to the other. Lastly, the implementation of the control flow across the components that include parallelized applications, e.g. with the help of the Message-Passing Interface (MPI), is difficult due to the functional orientation of the latter. In other words, it is technologically difficult to build a workflow management system that would enable running applications of both types (event-based serial ones and functionally-oriented parallel ones) within the same control and data flow logic and on distributed heterogeneous resources.

This paper's contribution introduces an alternative approach, which allows MPI applications to be built in a service-oriented way, thus allowing for flexibility of data processing, as required by geoscience applications, whilst keeping a much lower management overhead than in the case of traditional workflow management systems. The proposed approach is facilitated by a Multiple Instruction Multiple Data (MIMD)-based programming model, which could be inheritably implemented in MPI-parallel applications. The use of the elaborated programming model is illustrated on an implementation of a CFD simulation application for underground coalmine ventilation tasks. The remainder of the paper is organized as follows: Section II gives an introduction of ventilation networks and simulation tasks for them. Section III elaborates a microservices based architecture and methodology for implementation of simulation applications. Section IV discusses the results that are obtained for the evaluation cases. Section V concludes the paper and discusses the main outcomes.

II. VENTILATION NETWORKS AS OBJECTS OF MODELLING AND SIMULATION

Fossil coal remains one of the most important energy sources, along with gas, oil, and regenerative energy technologies. In particular, it holds a leading position among the fossil fuels with proven reserves of over 1 Tera-Ton (as shown in [4]) worldwide. At the same time, the coal industry is one of the most dangerous and security-critical industry branches, due to the complexity of the coal mining process from the deep underground locations (up to many tens of kilometers under the surface) and considering the aspects of a high gas content in the obtained coal masses. Ventilation is the most important aspect of the security provisioning in underground production areas of coalmines (see Figure 1) – it aims to ensure the underground mining workers with the necessary amount of fresh air and also to dilute the hazardous gases (mainly CH_4 – marsh gas) that are emitted during the coal loosing, transporting and other technological activities of the mining process.

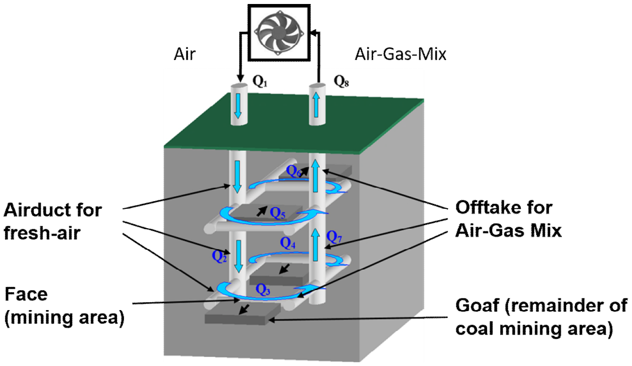

Figure 1. Structure of underground mine ventilation.

The degassing procedure is of especial importance for the safety of the mining process – the marsh gas concentration that exceeds the upper threshold can lead to vast exposures in the underground area with major human losses and injuries (see Figure 2). Historically, gas exposures have been the major reason of big catastrophes that have happened in the coal industry since the beginning of the industrialization era and up to nowadays (see [10]).

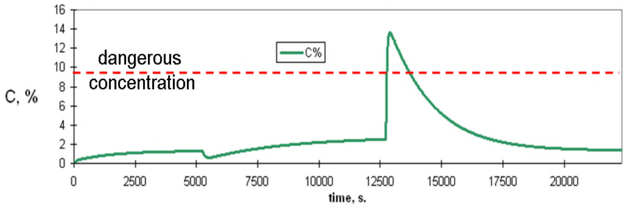

Figure 2. Analysis of march gas concentration in time.

The unpredictable nature of the gas emission is the major challenge for the operation of underground coalmines. Solving the challenge of air and gas distribution prediction requires a detailed knowledge of all dynamic processes that are happening in the elements of real industrial ventilation objects (see example in Figure 3). In most cases, the only possible chance to get insight into optimal planning of air distribution along the ventilation elements and plan the gas dilution actions is the use of modelling and simulation, coupled with the information coming from sensors, which are measuring airflow speed and gas concentration.

Typical CFD models of coalmine ventilation are based on the macroscopic definition of the multiphase flow based on Navier-Stokes equation, e.g., in the following general form for the air distribution in one branch of the ventilation network (e.g., as elaborated by Svjatnjy [5] for a 1-D approximation):

$$\begin{cases} -\dfrac{\partial P}{\partial \xi} = -\dfrac{2\rho}{F^2} Q \dfrac{\partial Q}{\partial \xi} + \dfrac{\rho}{F}\dfrac{\partial Q}{\partial t} + rQ^2 + r'(t)Q^2 \\ -\dfrac{\partial P}{\partial t} = \dfrac{\rho a^2}{F}\dfrac{\partial Q}{\partial \xi} - \dfrac{\rho a^2}{F} q \end{cases}, \qquad (1)$$

where P is the pressure, Q is the airflow, t is the time, ξ is the spatial coordinate, and the other values represent aerodynamic parameters and coefficients.

The gasflow distribution analysis is based on the data obtained from the sensors, which are fed in the transport equation (1) similarly to the airflow (for example, as described by Stewart et al. [6]). Given the insufficient coverage of the underground production areas by sensors, additional prediction models might be used, e.g., as described by the previous publication [7] for the goaf – a mine area that remains after the coal mining. The model of the whole ventilation network (see example in Figure 3) is built from the models of each of its elements/branches in the general form (1), extended by the boundary conditions in the nodes and following this hierarchical organization of the elements: approximation unit of numerical method → element/airway → section → network (see Figure 4). Further coalmine ventilation simulation tasks include reduction of the energy that is required for operation of the main mine fans (which usually consume up to 40% of the overall coalmine energy), as shown by Clausen [8].

The improvement of the quality of the CFD computational models has been the focus of many research activities in the last decade. In particular, lots of activities have been concentrated around integration of sensor data into the simulation process, for example, in the form of initial or boundary conditions for the mathematical models, serving the basis for the simulation packages. The "online" sensor data integration allows, among others, to specialize the generic models, i.e., adapt the model parameters to the specifics of the targeted simulation object. With the proliferation of the unified (Ethernet) networking standards (both wired and wireless) in the field of industrial and automated systems (like Industrial Ethernet's solutions PROFINET, Modbus, and others, as described by Kay et al. [1]), sensors have become a vital part of the distributed computing infrastructure and, most essentially, of their software applications. Such an infrastructure, which allows a seamless integration of the network-enabled data acquisition devices (such a CH4 – marsh gas sensor) with the "traditional" computation and storage facilities, is often referred to in the literature as Cyber-Physical Systems (CPS). Being initially elaborated for the automotive and industrial automation domains (as described by Broy [2]), the CPS concept is gaining an increasingly growing popularity for the other industrial and scientific areas, including the geoscience applications domain, as targeted by this paper.

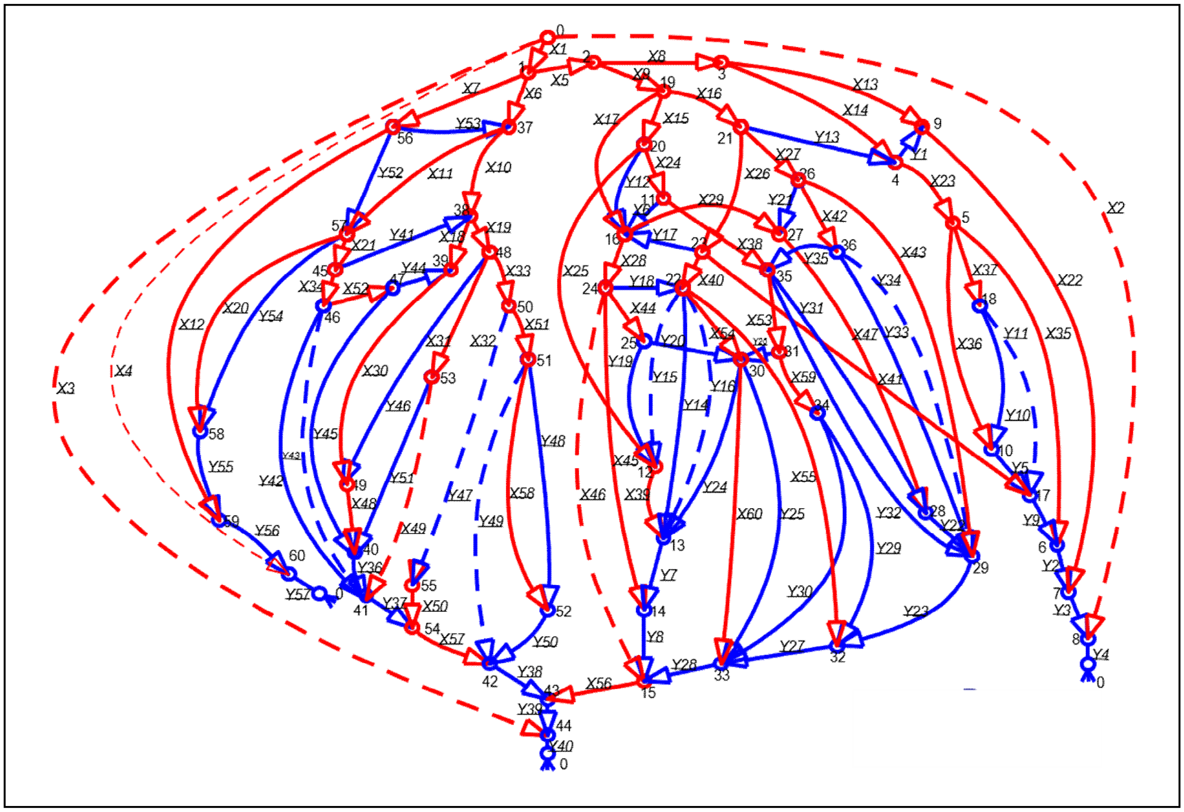

Figure 3. Illustration of real-complexity ventilation network with 117 branches and 61 connection nodes (coalmine South-Donbass Nr. 3 in Ukraine). The coloring of airflows is only used for better readability.

Nowadays, geoscience applications are represented by parallel software packages that require large-scale computing and storage facilities of HPC and Cloud infrastructures. Those applications are usually developed by means of MPI or other parallelization standards and have a limited ability to incorporate data from external (distributed over the communication network) sensors and other acquisition devices due to the following limitations:

- Heterogeneity of the CPS distributed infrastructure – many sensor devices are provided on the basis of a host system, whose architecture might differ from the typical HPC, Cluster, or Cloud environment, but still requires a seamless integration within the distributed application workflows. However, the standard parallelization approaches require a uniform infrastructure with the compute nodes of the same hardware architecture and performance class.
- Limited flexibility of the mainstream parallelization approaches to support distributed application scenarios – many parallel applications rely on the Single Instruction Multiple Data (SIMD) technique, which mainly targets densely built compute systems like HPC. However, the applications that are running on the truly distributed infrastructures (HPC + Cloud + remote embedded systems) have to be developed according to the MIMD approach, in order to allow different functionalities to be executed on different types of systems.

III. MICROSERVICE ARCHITECTURE FOR SIMULATION APPLICATIONS DEVELOPMENT

Ventilation networks analysis is a challenging process – the simulation software developers often face problems, some of which are listed below:

- Nonlinearity of the base equation system, which causes the need of applying a special numerical method, e.g., the Finite Differences, Finite Volumes, Discontinuous Galerkin, etc.
- Complex topological organization of ventilation networks.
- Complex hierarchical structure of ventilation elements involving several levels of control and regulation.

Although the mainstream simulation packages like OpenFOAM or ANSYS-CFX offer a rich development functionality which is sufficient for the implementation of the ventilation models, there are still numerous adaptations and optimizations necessary, which are very difficult to implement with general-purpose simulation tools. For example, it would be difficult to integrate the model for diffusion and filtration processes between the airways and gas emission sources in the ventilation section. However, the major disadvantage that remains is the inability of their use in distributed, heterogeneous hardware architecture environments. Therefore, novel approaches are required for the implementation of portable, scalable, and efficient simulation software.

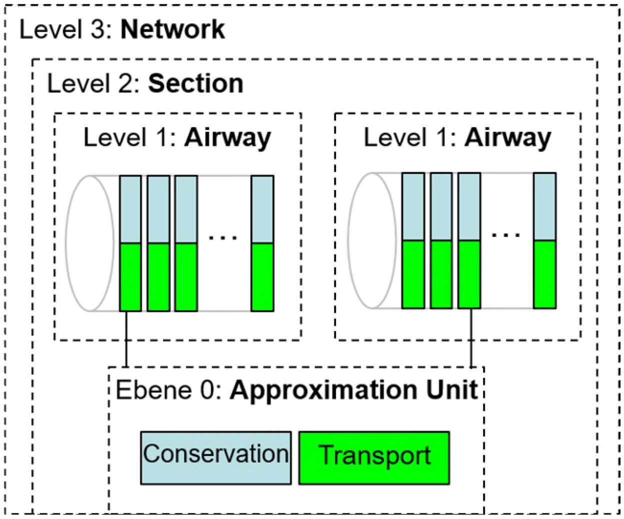

Figure 4. Hierarchical approach for composition of ventilation network models.

Object-oriented modelling and service-oriented platforms have been established in the last decade as an alternative to the traditional software development approaches. The actual trend in the service-oriented development goes in the direction of microservice (MS) architectures – a concept that is initially coming from the Internet-of-Things, Cyber-Physical Systems and Cloud domains. MSs are independent, isolated, portable software blocks/units, each implementing a part of a complex system, which can be decomposed according to functional, spatial or any other conditions. Each MS follows in its implementation the locality principle (as illustrated in Figure 5) – i.e., bearing responsibility for the assigned part of the complex system, according to the decomposition strategy. In order to reflect physical or informational connections of the real object, MSs can be bound together by a common data and/or control flow.

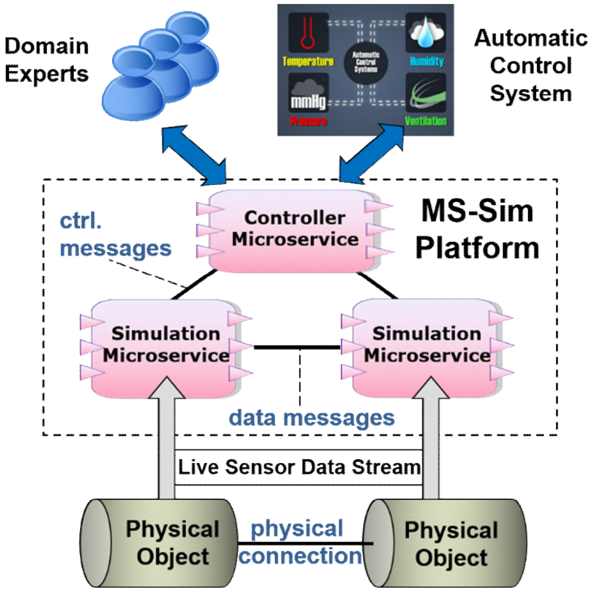

Figure 5. Microservice architecture for CFD simulation platform.

An example of a platform for execution of MS-based simulation workflows is presented in Figure 6 below. The connection between all the MSs in the system is provided by an external communication library, so that the MS-developers do not need to handle the data exchange explicitly – the data exchange performs asynchronously with the help of special buffers, used to flush the output data or read the input from the other MSs in the system, whenever required by the modelling algorithm. In case of an MPI-based implementation, every MS is executed by an independent MPI process, which is developed on the remote resource or a compute node. The MPI processes can run on a heterogeneous architecture, as they could be enabled by modern implementations like OpenMPI. Each simulation MS follows a command flow, as defined by the simulation logic, which can be implemented in an event-driven way, as shown by the listing in Figure 7. The command flow can be steered by a dedicated "master" MS, depicted as a "Controller" in Figure 5.

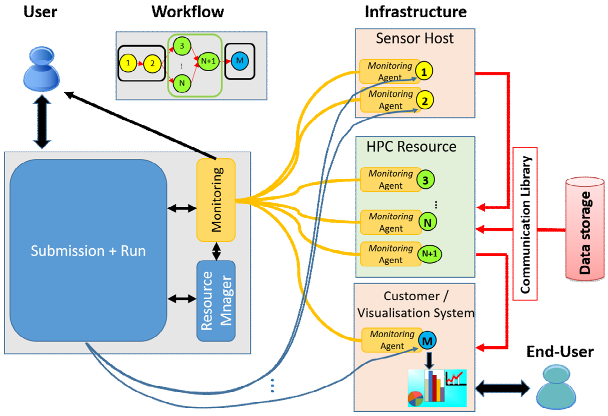

Figure 6. Execution platform design for microservice-based applications.

The communication between the MS happens with the help of the underlying communication library, e.g., by means of point-to-point or collective MPI calls to an MPI implementation. Results storing can happen either individually by every MS (e.g., in the Paraview format) or in the collective way using a consolidating database like ElasticSearch.

In fact, such functional decomposition-driven approach to the development of simulation software is not particularly new to the simulation of complex dynamic systems. For example, the Matlab/Simulink modelling package provides a module- (block-) based approach to construct a model from many smaller subsystems (functional blocks or submodels). However, this approach is inefficient when dealing with big, dynamic configurations of objects with a complex and variable network topology, like the targeted ventilation systems of coalmines. The main advantages of the MS-approach for application development are the separation of the computation and communication application logic during the development process, resulting in a decrease of implementation efforts, simplified implementation of the horizontally-scalable applications (just by replicating

```
void run() {
 bool finish = false;
 while (!finish) {
 // receiving command from the Master
   buffer_sync(Ports::command_flow);
   int command = get_buffer_value_stack<int>
                 (Ports::command_flow);
   switch (command) {
     case Commands::stop: {
       finish = true;
       stop();
       break;
     }
     case Commands::simulation: {
       simulation();
       break;
     }
     …
  }
```

Figure 7. Example of workflow event handling by microservice.

The services), easy implementation of hierarchical relationships between the services of different functionality levels (horizontal scalability), and the possibility to deploy a MS on any resource of the heterogeneous infrastructure. In the work that is presented in the paper, a modular architecture for development of CFD simulation applications based on a library of MS-components has been implemented. A simulation application is logically organized as a modular assembly of various MSs within a common architecture according to the hierarchical composition of the elements/services, as was previously depicted in Figure 4. Moreover, using the modular approach, the MS-based simulation application can transparently perform experimentation with different approximation schemes, discretization approaches, numerical solution methods, results validation techniques, etc.

IV. IMPLEMENTATION OF CFD VENTILATION STUDY WITH MICROSERVICES ARCHITECTURE

The above-presented microservices approach was used to implement a simple CFD study for a ventilation section consisting of 2 airways (Q_{FW}, Q_{VS}), a coal mining area (Q_S), and a goaf (q) with a gas emission source (q_m), according to the topology depicted in Figure 8. The airflow was enforced by a single mine fan, connected to the first airway (Q_{FW}).

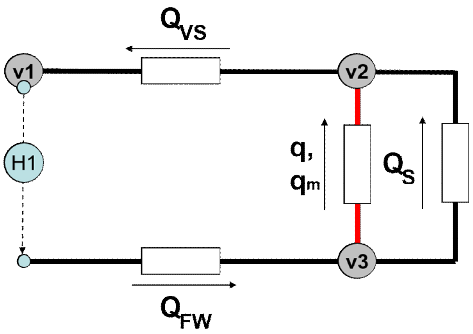

Figure 8. Test section structure.

Each element of the section is described by the base model (1). However, since the system (1) includes partial differential equations, the models development had to start from a lower granularity level – approximation elements of the numerical method of spatial discretization. In our case, the finite differences method was chosen for reasons of simplicity, that resulted in a set of k normal differential equations for approximation units in the general form (2):

$$\begin{cases} \dfrac{dQ_k}{dt} = \dfrac{F}{\rho} \cdot \dfrac{P_k - P_{k+1}}{\Delta y} - \dfrac{F}{\rho} r Q_k^2 - \dfrac{F}{\rho} r(t) \cdot Q_k^2 \\ \dfrac{dP_{k+1}}{dt} = \dfrac{\rho a^2}{F} \cdot \dfrac{Q_k - Q_{k+1}}{\Delta y} - \dfrac{\rho a^2}{F} q_k, \end{cases}, \qquad (2)$$

where the first equation represents the fluid transport in a (spatial) approximation element, and the second – conservation in approximation nodes, as elaborated in our previous publication [9]. These models form the bottom level of the ventilation models hierarchy (cf. Figure 4) and are used for the creation of the models of all upper hierarchical levels (airways/elements, sections, network), as shown in Figure 9. Connections between the models (or, precisely, their corresponding services) corresponded to the boundary conditions of the equations (2).

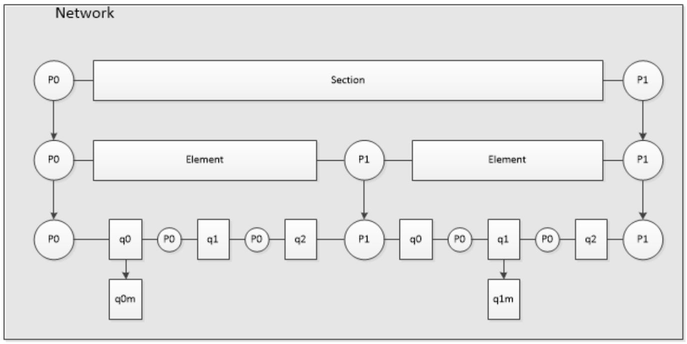

Figure 9. Hierarchical composition of modelling services.

The models were implemented with Open MPI as the underlying deployment and communication library. Three different scenarios were tested (A, B, C), as shown in Figure 10: an increase of fan pressure *P* (scenario A), an adjustment of global regulator *r* (scenario B), and a drop of fan pressure *P* (scenario C). The results are shown for flow in the outbound element (*Q*vs) and goaf (*q*), as well as marsh gas concentration *C*, which was defined as a ratio *qm/q*. All experiments (A, B, and C) showed results as expected to be by the physical experiments (e.g., performed by Svjatnyj in [5]). The service-based application workflow was steered by a dedicated supervising service – a manager, which performs the following functions:

- Initialization of microservices with the parameters that correspond to their respective physical object.
- Initiation of the iterative numerical solution process.
- Control of the solution readiness by means of polling the status of every individual "worker service".
- Instructing the services to store the results.

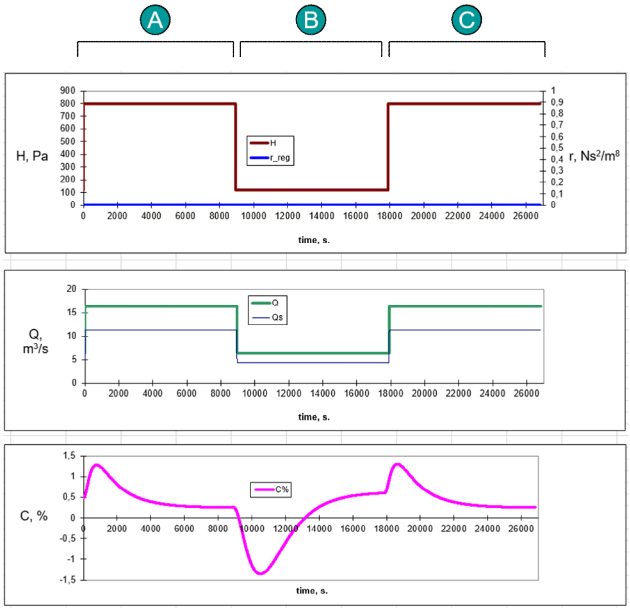

Figure 10. Modelling results.

The dynamic approach in which the services act as independent interactive components which are continuously running on the dedicated hardware and can be steered by a remote controller according to the specific application logic is particularly interesting for real-time control scenarios. In such scenarios, the services can incorporate the sensor data, make predictions for the future situation development, and instruct the control system about the probability of any potential risks appearance. On the other hand, the models can be optimized by adapting their parameters to best fit the actual mode of the controlled complex dynamic system.

V. CONCLUSION

The simulation technology is facing the challenges of application for new real-time scenarios that require a high flexibility of modelling tools in terms of the broader usage of the available infrastructure (data acquisition, storage, and processing devices). The rapid development of sensor networks has made possible a number of new innovative scenarios, for which the monolithic design of the existing simulation tools and workflow solutions on their top might be a big obstacle. Service-oriented platforms offer a promising vision of the future development of simulation tools by offering benefits of on-demand distribution and parallelization, which might be well supported by the underlying management platforms. Microservices are the technology that can, if not fully replace the workflow-based scenarios, have the potential to support and bring them to the principally new level of usability. The effort that was done on implementation of the ventilation scenario has revealed a high potential of microservices architectures in geoscience and other domains of science and technology.

Further research will concentrate, among other things, on elaboration of functional composition strategies for development of complex, assembled services for hierarchically-organized systems.

ACKNOWLEDGMENT

The work presented in this paper has become possible thanks to the support of the EU project ChEESE, which has received funding from the European Union’s Horizon 2020 research and innovation program under the grant agreement Nº 823844.

REFERENCES

[1] J. A. Kay, R. A. Entzminger, and D. C. Mazur, "Industrial Ethernet- overview and best practices", Conference Record of 2014 Annual Pulp and Paper Industry Technical Conference, Atlanta, GA, pp. 18-27, 2014.

[2] M. Broy, “Cyber-Physical Systems. Innovation through Software-Intensive Embedded Systems”, Springer, 2010.

[3] P. Chang et al., “Cyberinfrastructure Requirements to Enhance Multi-messenger Astrophysics”, In proc. Astro2020: Decadal Survey on Astronomy and Astrophysics, science white papers, no. 436; Bulletin of the American Astronomical Society, Vol. 51, Issue 3, id. 436, 2019.

[4] H. Andruleit et al., “BGR Energie study 2018 – data and development trends of German and global energy supplement”, Bundesanstalt für Geowissenschaften und Rohstoffe (BGR), Hannover, 2019.

[5] V. Svjatnyj, “Simulation of airdynamic processes and development of control system for underground mine ventilation”, PhD thesis 1985, (in Russian).

[6] C. Stewart, S. Aminossadati, and M. Kizil, “Use of live sensor data in transient simulations of mine ventilation models”, Mining Report 153 (4/2017), pp. 356-363, 2017.

[7] A. Cheptsov, “From static to dynamic: a new methodology for development of simulation applications”, chapter in Advances in Intelligent Systems: Reviews, Book Series, Vol.1, pp. 69-88, 2017.

[8] E. Clausen, “Mine ventilation in the 21st century – development towards adaptive ventilation systems”, Mining Report Glückauf 153 (4/2017), pp. 326-332, 2017.

[9] A. Cheptsov, “The system organization and basic algorithms of the simulation and servicing center for the coal industry”, in IEEE Proceeding International Conference Modern Problems of Radio Engineering,Telecommunications and Computer Science TCSET'2007, pp. 205-207, 2007.

[10] Wikipedia. Statistics of major catastrophes in coal mines, [Online]. Available from: https://de.wikipedia.org/wiki/Liste_von_Ungl%C3%BCcken_im_Bergbau/ 2020.03.24