CFD Modelling as an Integrated Part of Multi-Level Simulation of Process Plants – Semantic Modelling Approach

Marek Gayer[1], Juha Kortelainen and Tommi Karhela
VTT Technical Research Centre of Finland
Vuorimiehentie 3, Espoo, P.O. Box 1000, 02044, Finland, www.vtt.fi
Tel. +358 20 722 6246, Fax. +358 20 722 7052
ext-marek.gayer@vtt.fi, juha.kortelainen@vtt.fi, tommi.karhela@vtt.fi

Keywords: process simulation, CFD modelling, multi-level simulation, semantic modelling, plant equipment modelling

Abstract

In this paper, we present our computational fluid dynamics (CFD) modelling and simulation environment, which is designed to be suitable for integration to a large-scale system level simulation tools for industry process simulation of plants, such as Apros 6 software.

We discuss about the coupling possibilities of these simulations and further concentrate on the description of our pre- processing, fluid solver integration, and post-processing within the open-source Simantics integration platform. We present the semantic data model and ontologies for describing simulation models and their relations. We discuss about suitable open-source software component candidates, realisation of geometry, mesh, case configuration, boundary and initial conditions, solvers, and visualization parts.

Finally, we present our proposal implementation based on Eclipse Rich Client Platform. In addition, because we rely on open-source software tools, our proposals could be especially interesting for developments in the area of general process simulators, which might be eventually extended with more detailed 3D simulation models.

1. INTRODUCTION

1.1. Large-scale dynamic simulation of process plants and simulation information management

Dynamic-process simulation tools are used for example in the nuclear energy sector for planning, operator support and training, operation-state analysis, automation design and testing, safety analysis, and verifications in various stages of the power plant lifecycle. The advantages gained using these tools and methods can result in significant time and money savings, and improved safety.

The 3D flow simulation of an entire process plant is not possible for performance and complexity reasons. Instead, in dynamic-process simulation, we use 1D system codes for simulating the process of the plant eventually with co-operation or coupling with 3D simulation of the most important plant components, connected at inflow and outflow boundaries of the computational fluid dynamics (CFD) model. In large-scale dynamic-process simulation, there are also own models for the control and electrical systems.

CFD is used also for design and safety analysis. There is a clear need for combined analysis so that certain process components are computed using CFD and the rest of the process using large-scale process simulation.

In process simulation software systems, the simulation modelling flow is usually expressed using graphical diagrams. These environments also allow visualizing simulation data in diagrams by using monitors, animations, trends and various other, rather simple visualisation methods. Fig.1 depicts the environment of our software Apros 6, which is a 1D industry process plant simulator of plants.

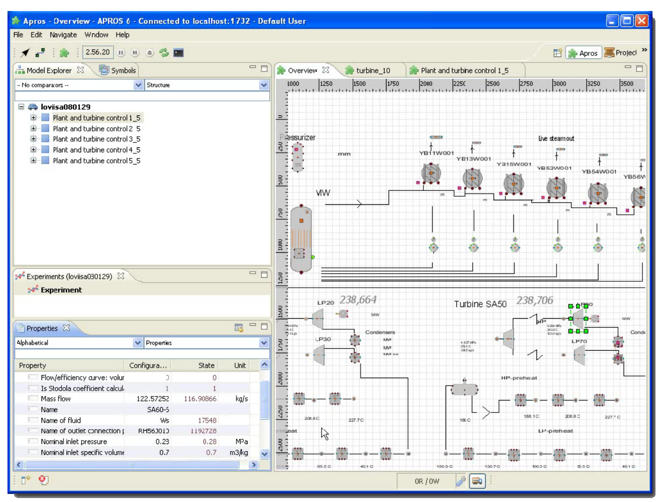

Figure 1. Apros 6 process simulator, integrated in Simantics Workbench.

The studied problem is industrially relevant because the need for multi-scale simulation tools is growing. It is

[1] This work was carried out during the tenure of an ERCIM "Alain Bensoussan" Fellowship Programme.

necessary to combine models of different levels of detail. Usually there are specific simulation tools for each detail level. By using combined approach, it is possible to make a set of multi-domain, multi-physics, and multi-scale models staying consistent even though the user is modelling from one perspective at a time.

Process simulators include Information management databases, which stores all the necessary information for performing the simulation. However, it is usually not possible to automatically co-operate with those databases and transfer corresponding data for the use in simulators outside the process simulations, such as existing 3D CFD tools. This significantly complicates the co-use of the simulators and it may be necessary to create separate models with different data storage mechanisms to be used in external simulation software applications, outside the process simulator. It would be a huge task to combine all these into one solver.

1.2. Coupling of CFD models with 1D process models

In coupling 3D CFD models with 1D process models, proper mapping of the mass and heat flow variables between the models is essential. At the inflow and outflow boundaries, the flow variables of the 3D models have to be reduced to 1D flow variables. In general, reduction a 3D flow variable to a 1D flow variable can be done in a straightforward manner by proper integration of the 3D flow field over the coupling surface and by proper matching of the 3D and 1D meshes. On the other hand, mapping the 1D flow variable to a 3D flow field is somewhat problematic because additional information is needed in order to generate the 3D spatial distribution from the 1D flow variable. In addition, local circumferential flow features at the mapping interface may cause problems for the numerical solving.

Similarly, as in coupling two different 3D codes, the numerical stability is an important issue because the coupling method is in practice often explicit. Therefore, numerical instabilities may rise if the coupling between the models is strong.

1.3. Our effort

We develop a platform for integration and coupling of process simulation models (1D) and more detailed CFD models (3D). For these purposes, it is necessary to establish not only optimal and intuitive data transfer and interfaces paradigms, but also pre-processing, post-processing, and computational solver integration to the environment. The latter are of primary concern and are of major topic of this paper. For this purpose, our efforts also deal with selecting and using optimal open-source tools to reach this goal. See Fig. 2 for an overview of 1D process / 3D CFD modelling connection. On the picture, it is indicated that upon selecting one of the component from the 1D process simulator (such as e.g. tank or pipe), we can change the modelling perspective in the software for of such a component to the full 3D CDF model, where we can make inspections, modifications and simulations.

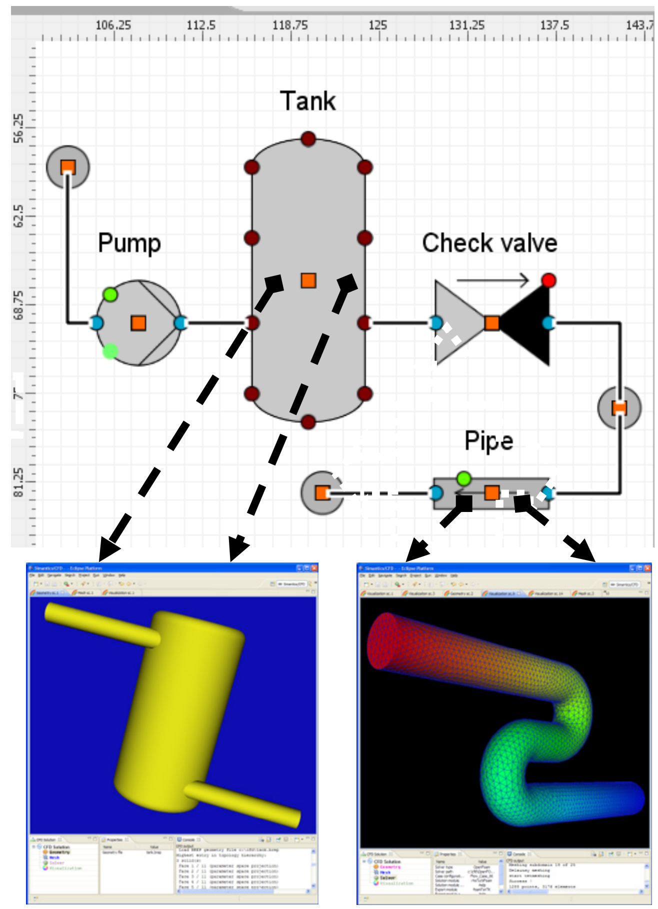

Figure 2. Illustrative schema of the link between 1D process simulator (Apros 6) and the 3D CFD model perspective.

2. RELATED WORK

The theoretical foundations of software platform Simantics, which we selected as implementation environment for process simulation and the CFD coupling, are in detail described in [1], [2], [3] and [4].

Several related methods and concepts have been demonstrated and documented separately. Using ontology-based 2D vector graphics, based on scalable vector graphics (SVG) was proposed in [5], a proposal of 3D models based on ontologies in large-scale process simulation can be found in [6], a marker based learning environment for detection of equipment in plants was described in [7]. The next generation of the large-scale process simulation software, developed at VTT and widely used in industry, Apros[2] [8], [9], is based on Simantics. In addition, the next version of BALAS[3] will be integrated to Simantics. BALAS is a steady state simulation package for pulp and paper processes developed at VTT Technical Research Centre of Finland, and several paper mills, engineering companies, and equipment manufacturers currently use it.

There has been some effort in integration between large-scale process simulation and CFD, such as a prototype level co-simulation solutions e.g. by FLUENT and Aspen Plus using CAPE-OPEN standard. This is a standard for communication between computational components in Process Modelling Environments (PME), suitable especially for sequentially based process simulators [10]. However, for the equation-based process simulators (such as Apros), the standard was found to be unsuitable for this kind of coupling [11], [12].

f CAPE OPEN has been used in implementation of Integration Toolkit for Aspen Plus and Fluent [13], [14], where in the bi-directional coupling, information of flow rate, temperature, pressure, and species components can be exchanged. The commercial applications have included chemical reactor, fuel cell system, coal-fired power plant, and natural gas combined cycle plant [15], [16].

Recently, component-based integration platform CHEOPS was implemented for chemical process modelling and simulation by using CORBA [17]. The simulation tools tested included Fluent, Aspen Plus, gPROMS and Parsival. As an application, dynamics of crystallisation was studied with multi-scale coupled simulation by using Fluent and Parsival [18].

Ontologies and semantic representations are recently used in wide area of engineering and industry applications. One of recent field is it's utilization in Building Information Modelling [19]. Also in Building Information Modelling, there is the desire of connecting simulations of different levels (CFD perspective of situations in the rooms vs. situation in whole building is a task to solve).

2.1. Open-source software tools and components

Our approach utilises as far as possible open-source software tools for the prototyping and delivery so that the major parts of the software proposal could be public and could be freely utilised also by others.

We have reviewed and practically experimented with several most advanced open-source CFD solvers and components for pre- and post-processing. The most appealing components candidates for CFD field according to our observations were: SALOME [20], OpenFOAM [21], Code_Saturne [22], Gmsh [23], NETGEN [24], TetGen [25], snappyHexMesh [21] and recently also Discretizer [26]. More detail reasoning can be found in [27], [28]. We also use open-source visualization and geometry processing libraries VTK[4] and OpenCascade[5]. The concrete selections and usage of the components in our proposal is covered in the next chapter.

3. OUR SOLUTION PROPOSAL

3.1. Semantic data model concept

The development of the Semantic Web[6] and its technologies has increased activity and interest on semantic data management and its applications. Technologies, such as the Resource Description Framework (RDF) [29], the Web Ontology Language (OWL) [30], SPARQL Protocol and RDF Query Language (SPARQL) [31], and Semantic Web Rule Language (SWRL) [32], introduce a set of basic technologies to describe the data, to model ontologies, to describe semantic database queries, and to model rules and complex restrictions into ontologies respectively. The overall concept these technologies introduce has inspired domains outside the Web technology to consider applying these methods for data management.

The fundamental principle of the Semantic Web, to include or map the meaning of the data to the data itself, is very attractive also from the system modelling data-management point of view. In the Semantic Web, one relatively simple data model is capable of describing practically all kind of data and knowledge, and data semantics allow computer-based reasoning on data, which, on the other hand, increases the usability and value of the data.

The Semantic Web project aims to developing technologies for improving the usability of the data and knowledge in the Web. Due to the fact that no-one can know what information can be found from the Internet, the basic assumption of the knowledge world is permissive; the Semantic Web is based on the open world assumption (OWA): if something is not specified in the data model it still can be correct, it is just undefined. This apparently simple assumption in the Semantic Web makes it difficult to apply the Semantic Web technologies for e.g. managing of modelling data of system simulation. This is due to the closed and well-defined nature of modelling domains. For setting restrictions and rules for modelling data, to e.g. enable automatic model validation, the use of OWA would make the domain ontology development a demanding task.

In system modelling, it is common that large amount of data is managed during the modelling process. An example of this is finite volume method, in which the modelled domain, e.g. the geometry of a pipe, is divided into a set of control volumes that fill the whole domain; this is called

[2] Apros software website: http://www.apros.fi

[3] BALAS software website: http://balas.vtt.fi

[4] Visualization Toolkit (VTK): http://www.vtk.org

[5] Open CASCADE website: http://www.opencascade.org/

[6] Semantic Web: http://www.w3.org/standards/semanticweb/

discretisation. The description of the discretised geometry, the mesh, contains usually large number of data, and for its representation, tables and vectors are the natural choices. In OWL, these data structures are missing, which also decreases its attractiveness for system modelling data representation.

3.2. Selected Approach

The selected approach in our work is based on the use of semantic data model and ontologies for describing the modelling data and its relations. In the semantic data model, all data is described using simple data structures, triples, which consist of a subject, a predicate, and an object; triples are also called statements [29]. With this simple data model, it is possible to describe complex data and its relations. In addition, the data model is very flexible and extensible. The data is described using ontologies, a kind of semantic vocabularies, which define concepts in different domain areas. Ontology in a semantic data model can be seen as a class library with a specified hierarchy and properties. Numerical simulation in general and system simulation especially are suitable for ontology-based modelling, due to their hierarchical and well-defined nature. The development of a domain modelling ontology is relatively straightforward procedure when the domain concepts are well known.

The advantage in using semantic data model related to present common methods is that all the data in the modelling database is described using the same simple data model. This model allows mapping of data from one domain to another so that all individual parts of data are captured just once. By using ontology mapping mechanisms, these data can form a network of data that describes, e.g. in our case, a complete process plant model. In addition to the data description mechanisms, the application of semantic ontologies enables the use of computer-based reasoning to the modelling data. Inclusion of e.g. domain modelling constraints and rules into the ontologies enables automatic model validation to some extent. And as the modelling constraints and rules are described also using the same data model mechanisms, they can be stored together with the modelling data. In traditional system modelling and simulation tools, the model validation information has been a feature of the software tool, not the modelling data.

3.3. Simantics Environment

There exist different kinds of solutions for modelling and simulation integration. Simantics platform[8] has a unique approach that combines semantic information modelling (ontologies) and simulation. Simantics has its own ontology description language called Layer zero. Layer zero has similarities with Web Ontology Language (OWL) but it has been specially designed for the description of engineering and system simulation ontologies, where the user is not just classifying the existing world but designing new products and production processes. These domains also consist of more complicated information structures and data types than some more traditional modelling targets.

In Simantics, plant information can be described and stored using a semantic knowledge database. In this environment, integration between different domain models can be effectively modelled and simulation tools can be configured based on existing plant design data, where semantic modelling in large-scale process simulation is used [33].

From the technological point of view, the platform applies the server-client architecture, in which the server consists of a semantic graph database, i.e., a triplestore (Simantics Core), and the client (Simantics Workbench) and its user interface framework are based on Eclipse[9] plug-in architecture.

3.4. Advantages of using CFD and ontology approach

Using of ontologies and semantic approach does not help us in obtaining better performance when computing CFD cases using numerical solvers. The reason for this is that these solvers are already implemented and does not count with ontological representations and cannot be easily modified to directly support semantic approach internally. Furthermore, due to performance reasons, data structures for codes inside these solvers are already chosen by its developers, and are suitable for underlying numerical methods, such as finite element or finite volume method.

The ontology representations provide advantages in other situations, such as easier integration and cooperation in different stages of CFD modelling. CFD modelling is used in process industry e.g. by equipment vendors during product development phase or by engineering offices during trouble shooting and safety analyses. Most of the engineering information during these stages is located in different plant design or information systems (such as CAD). CFD modelling process can then benefit on better integration into the design systems. Ontology approach provides us means for mapping information between the models that are different in nature. It also catalyses communication between engineers of different disciplines. Same goes also with linking simulators of different solving level of detail, such as in our case 1D process simulators and CFD solvers.

The CFD solvers differ in performance, requirement of the mesh quality and stability. While some solvers would fail to converge during simulated task due to mesh does not conforming quality requirements, such as conformance to Delaunay criterions or ortogonality of the mesh, others

[8] Simantics platform website: https://www.simantics.org/simantics

[9] Eclipse platform website: http://www.eclipse.org/

would still be able to continue, although with considerable performance and precisions penalties.

Another advantage of ontologies description and semantic approach is in possible obtaining of software and data abstraction, suitable for using of various types of meshing and other tools used in CFD modelling.

3.5. Modules and components for plant simulation and 3D CFD coupling

The CFD simulation process consists of the following three phases: pre-processing, solving, and post-processing. The pre-processing phase includes the definition of the flow domain geometry, the domain meshing, definition of boundary and initial conditions, as well as the parameters for the simulation. In the solving phase, the flow simulation is numerically solved. The post-processing phase can contain computation of additional dependent variables and visualization of the simulation results. In the following sections, the grouping of the software modules and components used in our approach are introduced and discussed. See Fig. 3 for an overview of the modelling.

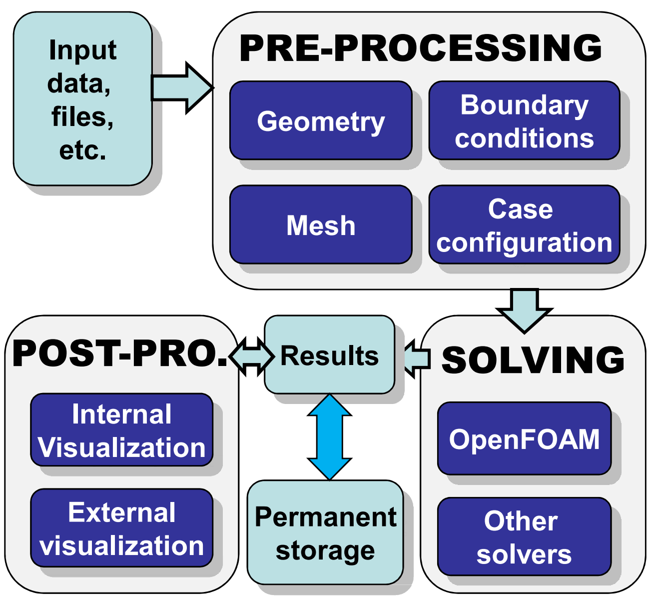

Figure 3. A schema of CFD modelling workflow and components used in our approach.

3.5.1. Geometry

For importing existing geometry e.g. from a CAD system we use Open CASCADE software library. It allows the geometry to be imported into the application in STEP, IGES, BREP format. The geometry can be edited in editors based on Open CASCADE, like SALOME. The imported geometry is then used in the subsequent modelling steps. Other formats, like STL are currently not directly supported, however in such particular cases, such as when format is simple, it is possible to implement easy conversion. In such case, it would be even possible for some simulation cases, to import the geometry model from modelling tools like Blender, Maya or Lightwave.

The Open CASCADE also provides solid geometry modelling features, which allow the imported geometry to be modified or to help to build an interactive geometry editor. This would be useful for e.g. simplifying the imported geometry for meshing. In the present version of the environment, this feature is however missing.

3.5.2. Meshing

Obtaining a quality mesh suitable for CFD simulations often remains a rather difficult task, namely if we request generating these meshes by using open-source tools. Finite volume method, which remains the most standard method in the field of CFD simulations, brings certain requirements of quality for the mesh and failing to meet those conditions can result in significant performance overhead, precision problems, or can even make simulation impossible due to instability and divergence of the solution.

When CFD is applied on industrial computation and complex geometries, the pre-processing phase and especially meshing becomes critical. For this reason, automatic meshing is an important requirement that would allow to make the meshing easy, intuitive, and fast, and would allow eventually changing of the modelled component shape fast. There are several algorithms available for automatic tetrahedral meshing of arbitrary geometry. These algorithms are robust and fast, especially compared to manual meshing. Currently, we use NETGEN meshing package, which we use as a custom command line tool created from the NETGEN libraries and integrated and launched from within our CFD environment. NETGEN provides an automatic tetrahedron mesh of a fair quality, which is sufficient to use with OpenFOAM. We found hexahedron meshes to be more performance optimal for CFD simulations, however there are not available open-source meshing routines for fully automatic generation of meshes on arbitrary structures. A possible integration of hexahedral meshers, like snappyHexMesh or Discretizer, remains for subjects of future work.

There are several strategies for managing mesh data in a semantic database. One is to use a fully semantic data model, in which all mesh details, i.e. nodes, cells, sub-meshes, and boundary patches, are described in the data model semantically. This approach would allow great flexibility in how the data is used, but the amount of the triple data and the limitations the size of memory and efficiency prevent it. Another approach is to semantically describe only the necessary features of the mesh, namely data tables that define the details, and boundary and initial condition data. This approach allows still good flexibility and requires only fraction of the resources compared to the fully semantic approach. In the present system, only necessary mesh data is managed semantically.

3.5.3. Solving

Currently, there are not many eventualities for open-source or free CFD analysis tools (especially for non United States residents, where several CFD packages are offered for free). OpenFOAM contains a C++ library, capable of numerical solution of partial differential equations. With this library, different solvers (including also those provided with the OpenFOAM distribution) can be built to solve various classes of problems in fluid dynamics and also other fields. We are using OpenFOAM for numerical solution of our CFD problems.

3.5.4. Case configuration and boundary and initial conditions

Our CFD environment is integrated into the Simantics database, which allows using ontologies to describe types of case configuration. In the user interface, we can define parameters for meshing, visualization methods, solver launching configurations and parameters.

OpenFOAM based solvers are configured by using its dictionaries, which are plain text configuration files, containing information's related to solving model, such as algorithm control, numerical schemes and numerical solution. However, in our proposal, instead of writing corresponding dictionaries manually, we can also use automatic transformation to generate dictionaries from our ontological representation. The settings are in this way also available through the user interface, by using the graph explorer and properties editor components, provided by using Simantics API. This way, we can configure basic parameters of OpenFOAM toolbox inside of our user interface.

Boundary and initial conditions for the simulation cases are part of the case definition for the OpenFOAM solver dictionaries, and therefore we configure them in the similar way.

3.5.5. Visualization

The flow field visualization in our environment is developed on VTK, The Visualization Toolkit [34, 35]. It includes high-level library routines for several visualization techniques, such as cut planes, iso-surfaces and streamlines. All visual components can be mapped with a field variable, such as flow velocity, pressure, or temperature. The use of this library remarkably decreases the effort for the implementation of visualization.

Beside of these integrated visualization possibilities, users can use several open-source visualization tools, such as possibilities with the generated data sets.

4. IMPLEMENTATION

The implementation of our proposal is realised within the Rich Client Platform (RCP) of Eclipse and is using Simantics. The RCP environment provides suitable application framework for appropriate user interface and also for integration and similar look and feel graphical user interface of Apros, BALAS, or of some general CFD based applications, such as ANSYS or SALOME. This implementation allows us to use pre-processing and post-processing capabilities for solved simulation cases. These possibilities include visualization of the geometry (Fig. 4), automatic generation of tetrahedron based mesh using NETGEN algorithm and visualization of results of the simulation using surface plots (Fig. 5), 3D cut plot visualization (Fig. 6) and streamlines inside the studied objects (Fig. 7).

5. FUTURE WORK

We would like to propose an ontology-based interface, which would allow connecting process simulators with various family of additional, both open-source and commercial CFD solvers, in a universal way.

Our semantic approach method between large-scale process simulation models (1D) and more detailed CFD models (2D/3D) will allow integration and coupling interfaces, communication, and synchronisation between the solvers. For this purpose, we will propose optimal data transfer and interfaces between 1D and 3D to allow control of simulation flow.

6. CONCLUSION

We have presented our pre-processing, post-processing and fluid solver environment for CFD simulations, which is a major part of a large-scale system level simulation integration of 1D process simulation of plants and 3D CFD modelling and simulation of selected plant components.

The base of our software proposal relies on using ontologies approach and semantic description, which is especially suitable for numerical simulation in general. We described advantages by using this semantic modelling and briefly describe our semantic integration environment Simantics, in which our proposal is realized.

Our CFD modelling environment proposal consists of software allowing geometry, meshing, case configuration, boundary and initial conditions, solving and visualization parts. The solving of simulation cases is realized by using an integrated OpenFOAM CFD code.

We have also briefly presented our proposal implementation in Eclipse Rich Client Platform and the Simantics integration platform.

Because we largely rely on open-source software components, our proposal could be interesting also in other, general 1D process simulations, where might be necessary modelling connection and coupling with more detailed 3D simulation models.

7. ACKNOWLEDGMENT

This project received financial support from Tekes (the Finnish Funding Agency for Technology and Innovation), and from Finnish industry. We would like to also thank Veli-Pekka Kuutti for his contribution of software development for this project, particularly for implementation of visualization methods.

REFERENCES

[1] T. Lehtonen, "Ontology-based diagram methods in process modelling and simulation," Master's thesis, Helsinki University of Technology, 2007.

[2] A. Villberg, "Design challenges of an ontology-based modelling and simulation environment," Master's thesis, Helsinki University of Technology, 2007.

[3] M. Luukkainen, "Use of 3D graphics for configuration and visualization of large scale process simulation: ontology-based approach," Master's thesis, Helsinki University of Technology, 2007.

[4] T. Kalajainen, "An access control model in a semantic data structure: Case process modelling of a bleaching line," Master's thesis, Helsinki University of Technology, 2007.

[5] T. Lehtonen and T. Karhela, "Ontology approach for building and visualising process simulation models using 2D vector graphics," in *SIMS Proceedings of the 47th Conference on Simulation and Modeling*. Finnish Society of Automation, SIMS - Scandinavian Simulation Society, 2006, pp. 141–146.

[6] M. Luukkainen and T. Karhela, "Ontology approach for co-use of 3D plant modelling and large scale process simulation," in *The 48th Scandinavian Conference on Simulation and Modeling (SIMS 2007)*. Linköping University Electronic Press, 2008, pp. 166–172.

[7] S. Prammanee, M. Luukkainen, T. Seuranen, and T. Karhela, "A marker-based mobile learning environment for a process plant," in *IADIS International Conference Mobile Learning*. IADIS, 2009.

[8] E. Silvennoinen, K. Juslin, M. Hanninen, O. Tiihonen, J. Kurki, and K. Porkholm, *The APROS software for process simulation and model development*. VTT publications, 1989.

[9] A. Niemenmaa, J. Lappalainen, I. Laukkanen, S. Tuuri, and K. Juslin, "A multi-purpose tool for dynamic simulation of paper mills," *Simulation Practise and Theory*, vol. 6, no. 3, pp. 297–304, 1998.

[10] The GCO Consortium, "CAPE-OPEN standards," 2003.

[11] M. Friman, "Modularisation issues in dynamic equation-based process simulation solvers - comparison of two different approaches," Master's thesis, Helsinki University of Technology, Department of Chemical Technology, 2003.

[12] E. Karlsson, "A process simulation model as part of a process plant life-cycle information environment," Master's thesis, Helsinki University of Technology, Department of Chemical Technology, 2004.

[13] S. E. Zitney, "Multiscale modeling and simulation of advanced power generation systems," in *DOE/NSF EPSCoR 2005 Conference*, Dearborn, USA, 2005.

[14] M. Osawe, "Fluent CAPE-OPEN COM/CORBA bridge and CO-compliant unit operation," in *2nd Annual U.S. CAPE-OPEN meeting*, Morgantown VW, 2005.

[15] M. Syamlal, S. Zitney, and M. Osawe. (2005) Using CAPE-OPEN interfaces to integrate process simulation and CFD. [Online]. Available: http://www.colan.org/CO%20Update/COUpdate07_Fluent_Article.html

[16] D. Sloan, "Power plant analysis using CFD and process simulation," in *2nd Annual U.S. CAPE-OPEN meeting*, 2005.

[17] G. Schopfer, A. Yang, L. von Wedel, and W. Marquardt, "Cheops: A tool-integration platform for chemical process modelling and simulation," *Int. J. Softw. Tools Technol. Transf.*, vol. 6, no. 3, pp. 186–202, 2004.

[18] V. Kulikov, H. Briesen, and W. Marquardt, "Scale integration for the coupled simulation of crystallization and fluid dynamics," *Chemical Engineering Research and Design*, vol. 83, no. 6, pp. 706 – 717, 2005, 7th World Congress of Chemical Engineering.

[19] P. Siltanen, S. Varjes, M. Ylikerälä, and A. S. Kazi, "Ifc and pmo for estimating building environmental effects," in *Proceedings of the 14th International Conference on Concurrent Enterprising, 23-25 June*, Lisbon, Portugal, 2008, pp. 515–522.

[20] A. Ribes and C. Caremoli, "Salome platform component model for numerical simulation," in *COMPSAC '07: Proceedings of the 31st Annual International Computer Software and Applications Conference*. Washington, DC, USA: IEEE Computer Society, 2007, pp. 553–564.

[21] OpenCFD Limited, *OpenFOAM - The Open Source CFD Toolbox - User Guide*, 2009, http://www.openfoam.com/docs/ .

[22] F. Archambeau, N. Mechioua, and M. Sakiz, "Code_saturne: a finite volume code for the computation of turbulent incompressible flows- industrial applications," *International Journal on Finite Volumes*, vol. 1, pp. 1–62, 2004.

[23] C. Geuzaine and J.-F. cois Remacle, "Gmsh: a three-dimensional finite element mesh generator with built-in pre- and post-processing facilities," *International Journal for Numerical Methods in Engineering*, vol. 79, no. 11, pp. 1309–1331, 2009.

[24] J. Schöberl, “NETGEN: An advancing front 2D/3D-mesh generator based on abstract rules,” *Computing and Visualization in Science*, vol. 1, no. 1, pp. 41–52, 1997.
[25] H. Si, *A Quality Tetrahedral Mesh Generator and Three-Dimensional Delaunay Triangulator*, 2006. [Online]. Available: http://tetgen.berlios.de
[26] B. Bergqvist. (2010) Discretizer, a free mesh program for CFD. [Online]. Available: http://www.discretizer.org/
[27] J. Kortelainen, “Meshing tools for open source CFD - a practical point of view,” VTT, Espoo, Finland, Tech. Rep., 2009, http://www.csc.fi/english/pages/lscfd/Documents/MeshingToolsForOpenSourceCFD.pdf .
[28] M. Gayer and G. Iannaccone, “A software platform for nanoscale device simulation and visualization,” in *IEEE International Conference on Advances in Computational Tools for Engineering Applications*. Zouk Mosbeh, Lebanon: Notre Dame University, Lebanon, 2009, pp. 432–437. http://www.gayer.ws/en/publications/gayer09actea/
[29] F. Manola and E. Miller, “RDF Primer,” The World Wide Web Consortium, http://www.w3.org/TR/2004/REC-rdf-primer-20040210/ , W3C Recommendation, Feb. 2004.
[30] S. Bechhofer, F. van Harmelen, J. Hendler, I. Horrocks, D. McGuinness, P. Patel-Schneijder, and L. A. Stein, “OWL Web Ontology Language Reference,” World Wide Web Consortium (W3C), Recommendation, 2004, see http://www.w3.org/TR/owl-ref/ .
[31] E. Prud’hommeaux and A. Seaborne, “SPARQL Query Language for RDF,” The World Wide Web Consortium, W3C Recommendation, Jan. 2008. [Online]. Available: http://www.w3.org/TR/2008/REC-rdf-sparql-query-20080115/
[32] I. Horrocks, P. Patel-Schneider, H. Boley, S. Tabet, B. Grosof, and M. Dean, “Swrl: A semantic web rule language combining owl and ruleml,” World Wide Web Consortium, W3C Member Submission, May 2004. [Online]. Available: http://www.w3.org/Submission/SWRL/
[33] A. Villberg, T. Lehtonen, T. Karhela, and K. Kondelin, “Applying semantic modelling techniques in large scale process simulation,” in *ALSIS ’06 Proceedings of the 1st IFAC Workshop on Applications of Large Scale Industrial Systems*. Suomen Automaatioseura, 2006.
[34] W. J. Schroeder, K. M. Martin, L. S. Avila, and C. C. Law, *The VTK User’s Guide*. Kitware, Inc., 2000, http://www.kitware.com .
[35] W. Schroeder, K. M. Martin, and W. E. Lorensen, *The visualization toolkit: an object-oriented approach to 3D graphics, 4th edition*. Kitware, Inc., 2006.

BIOGRAPHY

Dr. Marek Gayer is a research fellow of the Computer Simulation Technology team, which is a part of the System Research knowledge center of Technical Research Centre of Finland (VTT).

He obtained his Ph.D. degree in “Information Sciense and Computer Engineering” from Czech Technical University in Prague in 2006 after defending his thesis “Real-time Visualization Techniques for Modelling of Combustion and Fluids”. Prior to joining VTT, he was a research fellow of The National Research Council (CNR) at University of Pisa, Italy and Norwegian University of Science and Technology (NTNU) at Trondheim, Norway.

His current research interests include computer modelling and simulation (such as those based on finite element and finite volume method solvers), computational fluid dynamics, information technologies, open source software integration, software development and software platforms for research simulation, modelling and visualization projects.

MSc. Juha Kortelainen earned his MSc. degree in mechanical engineering from Helsinki University of Technology in 1995. He has been working as a research scientist at Helsinki University of Technology and the last eight years as a research scientist and senior research scientist at VTT Technical Research Centre of Finland. The main research focus of MSc. Kortelainen has been the application of multibody system dynamics for working machines and computational fluid dynamics of internal combustion engines. His current research interest includes application of semantic data model for modelling data management and integration.

Prof. Tommi Karhela is a Research Professor and leader of the Computer Simulation Technology research team at the System Research knowledge center of VTT. Karhela joined VTT 1996 and received his Ph.D. (Tech) degree at Helsinki University of Technology in 2002. Doctoral thesis was titled “A Software Architecture for Configuration and Usage of Process Simulation Models”. Prior to joining VTT, Karhela has worked short periods at CERN Geneva, Technology Center of Sandvik-Tamrock and Theoretical Research Center of Physics at Helsinki University. Karhela’s current research interests include process simulation, semantic data modelling and software architectures.

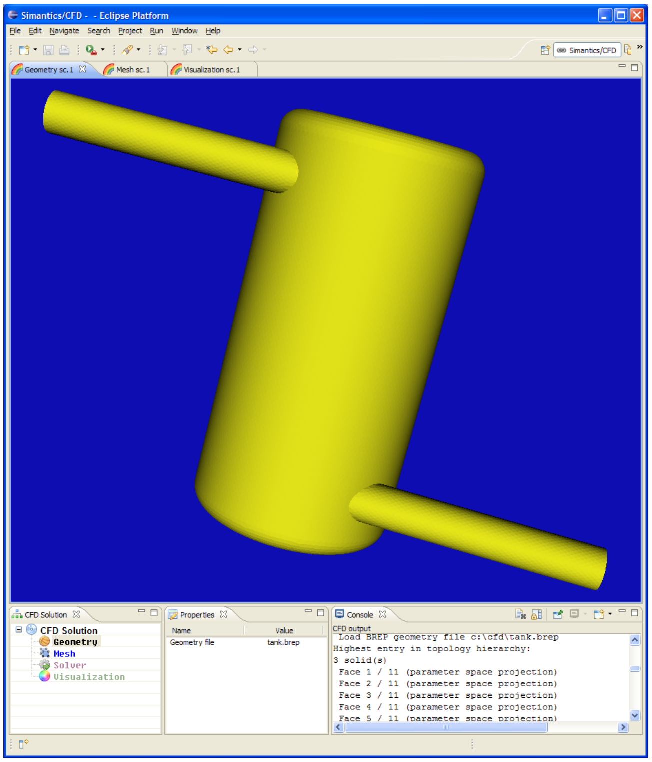

Figure 4. Visualization of geometry of a plant tank.

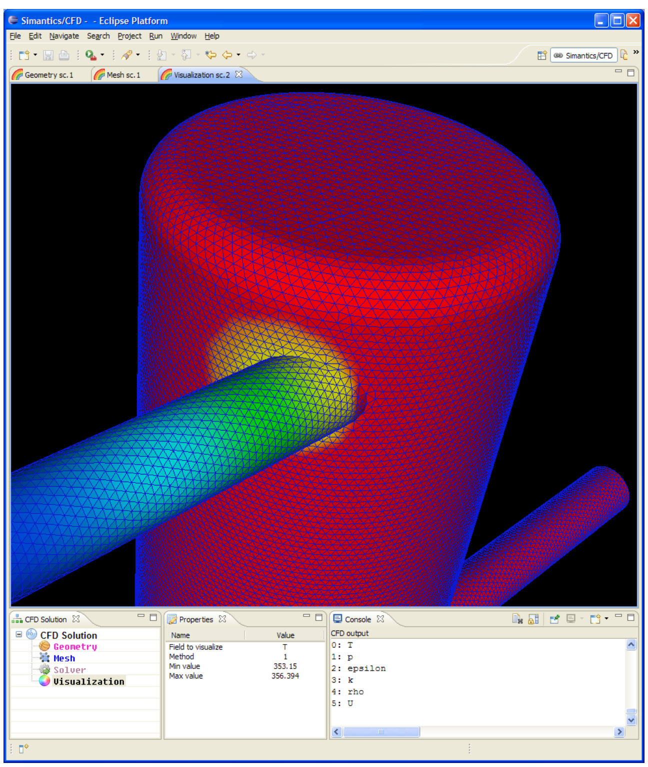

Figure 5. Mesh and surface plot of temperatures in the tank.

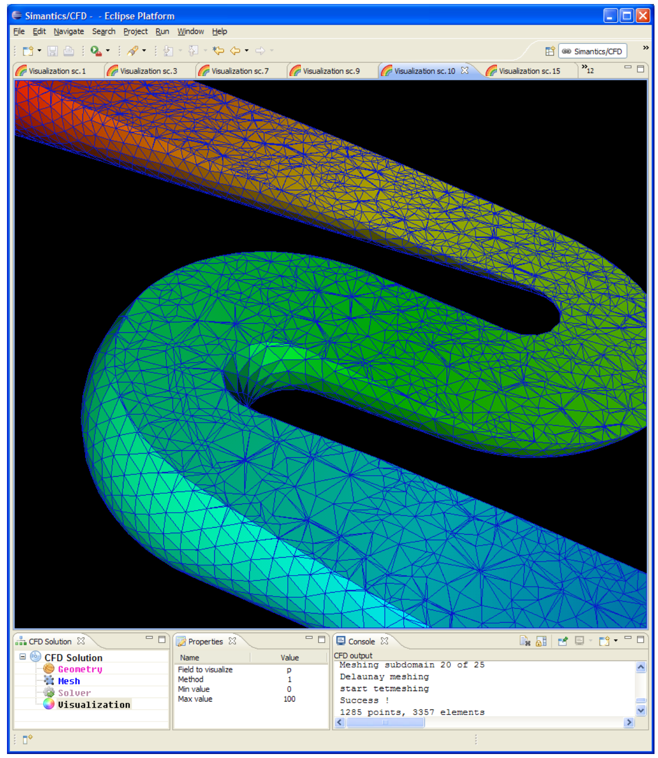

Figure 6. 3D cut plot visualization of pressures inside a pipe.

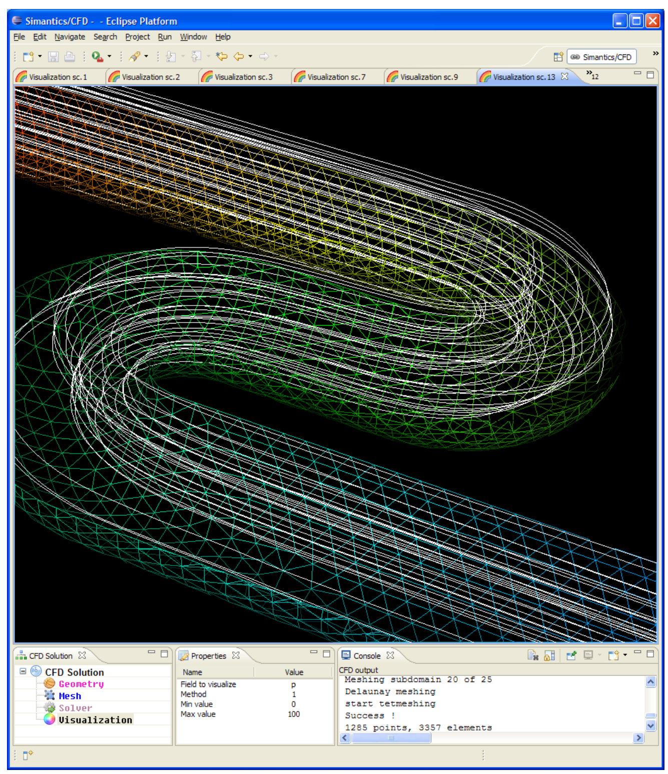

Figure 7. Stream lines visualization in a pipe.