International Environmental Modelling and Software Society (iEMSs)
2010 International Congress on Environmental Modelling and Software
Modelling for Environment's Sake, Fifth Biennial Meeting, Ottawa, Canada
David A. Swayne, Wanhong Yang, A. A. Voinov, A. Rizzoli, T. Filatova (Eds.)
http://www.iemss.org/iemss2010/index.php?n=Main.Proceedings

A Workflow-based Compliance Assistant for Facilities in EU Emission Trading System

Philip Joschko[1], Christian Schmitz[1], Bernd Page[1], Nicolas Denz[2]
[1]Department of Informatics, University of Hamburg, Germany, [2]ifu Hamburg GmbH
[page, joschko]@informatik.uni-hamburg.de, schmitzmedia24@web.de, n.denz@ifu.com

Abstract: The European Emission Trading System (ETS) has been elaborated as a market-based instrument for controlling greenhouse gas emissions. Selected sectors have to bring in Emission Allowances (EUA) for every emitted ton of CO2. A large part of EUAs is freely allocated to plant operators, but high emitters have to settle their lack of EUAs by buying EUAs from those with a surplus of allowances. Operators have to fulfil several tasks in time: Creating monitoring concepts, collecting initial data, participating in the allocation process, emission reporting, trading of EUAs, and refunding of used EUAs to national authorities. In this paper, we present the increments of our project "Emission Trade Assistant", which is funded by the German Federal Ministry of Education and Research. We present our draft of an assistant software built from distinct software components and sum up our experiences in modelling workflows of plant operators based on Windows Workflow Foundation and .NET 4.0. Our approach is promising to ease the duties of the facility operators, but it also raises high demands on an easy-to-use and highly dynamic workflow modelling framework. We summarize our discussions with operators, national authorities and accredited experts, before we provide requirements for this modelling framework. We will demonstrate that workflow-based assistants are highly suitable to support operators with certain tasks, to mediate between operators and various obligatory IT systems, and to take operators by the hand and guide them step-by-step through the ETS.

***Keywords**:* Software Assistant, Emission Trade, Environmental Informatics, IT-System for the Environment

1. INTRODUCTION

Since the 15th conference of the United Nations Framework Convention on Climate Change in Copenhagen, in Denmark 2009, the saving of natural resources as well as the avoidance of greenhouse gas emissions became a much discussed topic again. While the community of states was not able to agree on obligatory targets for a follow-up for the Kyoto-Protocol, the European Union establishes the idea of emission trading on its own since 2005. Emission Trading has been originally described in the Kyoto Protocol at the level of states, while the EU Emission Trading System (ETS) is working on the level of facility operators.

The following paper starts by explaining key concepts of Emission Trading and the ETS. We sum up the duties and working processes of the facility operators in scope of ETS from the point of view of applied computer science, we analyse the central workflows for the different duties of the facility operators and summarize our discussions with different participants in the EU ETS. We also give a short outline of the IT systems which are obligatory to use in the German ETS. In the third section, we illustrate our definition of an assistant software, before we present our draft of an assistant software particularly designed for facility operators in EU ETS in section 4. In section 5 we sum up our experiences in modelling central workflows of plant operators based on Windows Workflow Foundation and .NET 4.0. Section 6 finally contains our conclusion and the roadmap of the upcoming tasks.

This is a progress report of our two-year joint project between the Department of Informatics of the University of Hamburg and ifu Hamburg GmbH called “EmTrAs – Emission Trade Assistant”. The project is funded by the German Federal Ministry of Education and Research (support code FKZ 01LY0819 A/B).

2. EUROPEAN EMISSION TRADING

2.1 Idea of Emission Trading

The European Emission Trading System (ETS) has been introduced as a market-based instrument to help the EU in controlling greenhouse gas emissions. The ETS does not regulate where or how emissions have to be avoided, but it defines a Europe-wide upper emission limit (Cap) and enables the plant operators to freely trade their Emission Allowances (EUA). Thus it stimulates the market to find the most efficient and affordable way of emission reduction. For this, facilities of selected sectors with high carbon emissions, e.g. the energy sector, have to bring in EUAs for every emitted ton of CO2. In fact, a large part of required EUAs is allocated to plant operators by national governments free of charge, but high emitters have to settle their lack of EUAs by buying EUAs from plant operators with a surplus of allowances. Because the total amount of Europe-wide existing EUAs is limited, the EU is able to cap the total amount of carbon emission and reduce it step-by-step. This key instrument enables the EU to remain an outrider in climate protection, independently of the results in Copenhagen in December 2009. This mechanism is well-known and it exactly follows the theorem by Ronald Coase from 1937 (Wegehenkel [1980]) for which he earned the Nobel Prize in Economics in 1991.

2.2 Compliance Responsibilites of Emitters

However, from plant operators' view, the process of ETS is a little more complicated than this short review suggests. Several tasks have to be fulfilled in time to get in compliance: (a) creating and periodically revising emission monitoring concepts, (b) collecting initial data before taking part in ETS, (c) participating in allocation process to get EUAs for free, (d) detailed annual emission reporting, (e) getting diverse certificates by accredited experts, (f) trading of EUAs through a broker, in the stock exchange, or bilaterally with other operators, (g) refunding of used allowances to national authorities. Figure 1 shows a broad survey of these tasks. In some European states, e.g. in Germany, operators additionally are obligated to use several IT-Systems (see section 2.3) in order to simplify the processing for the authorities.

At the University of Hamburg, a research group at the Faculty of Economics and Social Science surveyed the behaviour of companies in selected European countries (Engels et al [2008]). They e.g. asked companies to rate their additional workload generated by EU ETS. While Danish and Dutch companies mostly stated a light or medium workload, many German companies perceived the additional workload to comply with the rules of EU ETS as rather heavy or very heavy (Engels et al [2008]). This might depend on problems with the obligatory IT Systems which have to be used by facility operators in Germany on the one hand, but on the other hand also on the Europe-wide rather complex legal and temporal structure of EU ETS. Because regulations and IT-Systems are in a constant change, operators have difficulties to bring routine into the 'jungle' of ETS. Thus a supporting software assistant could be one suitable approach to improve this situation.

We talked to different participants of the ETS, such as plant operators, national authorities, and accredited experts to identify the problems facility operators have with EU ETS. As a result of these interviews, we claim that flexibility is the most important requirement of an emission trade assistant software: Frequent changes in the ETS process require timely updates of the assistant software. The ETS workflow differs fundamentally from company to company. Some companies own more than 20 facilities, each of them implementing comprehensive material flows. In these companies, the environmental management departments would like to perform characteristic ecological balances via Environmental Management Information Systems. Before submitting obligatory emission reports, the characteristic measures are cross-checked and evaluated. Of course allowances can be shifted between the companies' facilities. Other companies only own a single combustion plant with one kind of material flow, mostly a fuel like heating oil or firewood. Because wood is a sustainable, renewable product, companies don't need to buy EUAs for emissions produced by the combustion of wood. Nevertheless, they need to report those emissions, too. Things are getting more complicated, if a company uses more than one kind of fuel, and become particularly difficult, if the fuel comes from a foreign country. In this case, special measuring instruments are needed, which must be examined regularly by accredited experts. In addition, the greater the amount of emissions, the more accurate the measuring equipment must be.

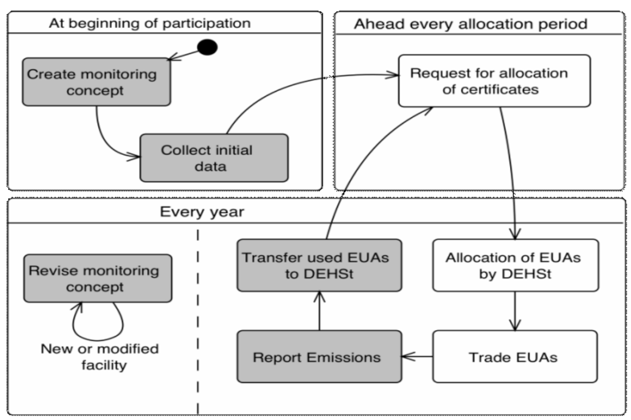

Figure 1. Workflow of EU ETS in Germany displayed as a UML state chart diagram. Gray rectangles are obligatory tasks, white rectangles are optional steps (from Page et al [2009])

2.3 Obligatory IT-Systems in German ETS

The German Emission Trading Authority (*Deutsche Emissionshandelsstelle*, DEHSt) commits facility operators to use distinct software systems for processing the tasks of ETS. Due to this obligatory status, an exhaustive Emission Trade Assistant software will have to provide interfaces to at least three software systems, which are described in the following.

The *Formular Management System* (FMS) is a web-based application, which can be customized for submitting data of specific application domains, but is mainly used in governmental settings. In the German ETS, the use of the FMS is mandatory for emission reporting, initial data collection, and allocation requests. A useful feature of the FMS web forms is their ability to save the form content locally on a client computer as an XML file, and to re-import this file into the form, since it provides an interface to connect external software systems like an Emission Trading Assistant to the FMS. The DEHSt publishes documentations for the XML, so that the connection of third-party systems is achievable.

The *Virtual Post Office* is a closed mail system, which provides encrypted, authenticated electronic communication (see KBSt [2009]). This client/server application differs from conventional email clients in three substantial aspects: First, it is a closed system. This means that only previously registered users can be addressed. Secondly, there are specific message types, e.g. for the transmission of an allocation request or the emission report. Thirdly, all communication between registered users is encrypted and signed with the help of a public-key-infrastructure based on obligatory electronic signature cards. Hence it is officially recognized like a handwritten signature by German Signature Act (SigG). The VPS client for facility operators is written in Java, but there is also a .NET-based derivate, which could be adjusted for the scope of ETS and used within our assistant software.

The *Emission Trading Registry* is a separate web-based application, which has to be used like an online banking system for transferring EUA instead of money. Every facility operator who participates in the emission trading scheme automatically obtains an account. All national ET registries of the participating EU member states are connected to the central European registry called Community Independent Transaction Log (CITL) and every national or inter-European transaction is processed by the CITL. Unfortunately, although accounts of the Emission Trading Registry keep allowances worth billions of Euro, they are protected in an inadequate way only with a password. This is one reason that unidentified persons were able to steal allowances at the beginning of 2010 from German accounts with a simple phishing attack. The use of TAN lists, as it is also used within common banking software, is advisable. From our point of view, this is a pre-condition for the integration into an Emission Trade Assistant, which has to be fulfilled by national authorities first.

3. SOFTWARE ASSISTANTS

When looking for an appropriate approach to support facility operators in the German emission trading scheme, the starting point was the question how and where informatics can help with its available instruments. As we have shown in the previous section, there are several tasks, which have to be fulfilled by facility operators and distinct assistant tools are needed to support these activities. There are three different relevant approaches: Office applications, Environmental Management Information Systems (EMIS) and Software assistants. Integrating the Emission Trade Assistant into an EMIS could especially ease the collection and analysis of data as part of facility operators' daily work. The focus of these frameworks is to support the collection, documentation, assessment, and management of enterprise environmental protection information. However, they are much to complex for most potential users by offering to many degrees of freedom and lacking a sophisticated knowledge of the addressed domain. Additionally, they do not provide tools for communicating with obligatory IT Systems like the FMS or the VPS. The concept of software assistants promises to be a more suitable approach. Since the term 'software assistant' is defined quite heterogeneously in computer science literature, we first state a definition, which narrows down the characteristic attributes of several literature sources (from Schmitz et al [2008]):

Definition: A software assistant is a distinct system with the purpose of supporting the user in carrying out complex and infrequent tasks. This is achieved by interacting with the user through dialogue prompts, forms, and help dialogues. Thus the processing of the given task is simplified by guiding the user step-by-step through a (pre-)structured workflow.

We can distinguish different types of software assistants. *Wizards* are used to setup systems and for data entry by means of dialogues and forms. They could e.g. be used for emission reporting. They also can support an import of data from office applications like Microsoft Excel. *Guides* offer instructions to deal with external system, which are not directly connected to the Emission Trade Assistant. We could use them for example, to show the user a responsible use of the Emission Trading Registry. *Rule-based assistants* are activated by a previously defined rule which implies that the user might need assistance.

Thereby not only single tasks like creation of the monitoring concept or the emission reporting can be supported by the assistant software, but it should support the whole process, by taking operators by the hand and guide them step-by-step through the complex emission trading system. Our approach of a step-by-step guiding software is promising to ease the duties of the facility operators in the complex process of emission trading, but it also raises high demands to an easy-to-use and very dynamic development framework for software assistants.

4. DRAFT OF AN ETS ASSISTANT

4.1 Software Components

From our interviews with participants of the ETS, we compiled a list of required software components.

First of all, we need a flexible *workflow engine*, which supports the processing of the ETS and can be adjusted to the affordances of individual companies. Another component is needed, which provides a *workflow overview.* To support the complex time course of the different duties the use of a timeline-based navigation seems to be appealing where all duties are positioned on a horizontal timeline with named icons. Thus the main process in the assistant workflow is mapped. This calendar management component should act as a *compliance guide*, which checks whether the user has fulfilled obligatory tasks, and in case of need, remembers the user to do so.

A *domain model* component keeps the domain specific data, like emission data or the material flows of the facility. A *persistence library* supports the storage of this model in a database (Busse et al [2008]). A *user management* tool is required, because in many companies, not all users are allowed to access all of the data. This is particularly important for companies with more than one facility and a central environmental management.

A *wizard framework* based on the workflow engine is needed to implement wizards for the different tasks. We can e.g. implement wizards for importing or exporting XML-Data from the German FMS or from spreadsheet applications, for collecting master data, emission data, data for allocation requests, or initial data. In all cases, an output has to be generated, which needs to be readable by the DEHSt FMS. An *assistant for monitoring concepts* could include a graphical tool for modelling the material flow of a plant as a *material flow net* (Möller [2000]) and offer text modules to describe the procedural method of the facility.

Because regulations of ETS are in constant change, an *update* component has to keep the workflow up to date, and has to provide relevant news in ETS with the aid of a *newsfeed* component. A context-sensitive *help and validation system* has to observe user action and, if necessary, provide hints or links to an archive with guidelines and corresponding laws.

A VPS *mail component* should be directly integrated in order to avoid system discontinuities since the user is obligated to send its emission report via the enclosed mail system (see Section 2.3). An integration of the Emission Trading Registry for providing access to EUA accounts and offering the possibility to buy and sell allowances via the Emission Trade Assistant is desirable, but because of security leaks currently not advisable.

Additionally, a future focus of our project is the development of analysis components that support the decision-making of facility operators, e.g. for buying and selling allowances.

4.2 Plug-in-based Architecture

As shown in the previous section, many distinct software components are required to support the processing of the EU ETS. However, only few companies need all of them, while some companies will need additional components to connect the Emission Trade

Assistant to their individual IT infrastructure. A plug-in-based architecture promises the flexibility needed to compose an individual solution for companies with different requirements.

In our project the open source plug-in framework Empinia (Schnackenbeck et al [2007], Empinia [2009]) is used to achieve these aims. The framework is developed at ifu Hamburg, the HTW Berlin, and the University of Hamburg. Its purpose is to provide a platform for the component-based development of interactive rich client applications under Microsoft .NET, comparable to Eclipse in the Java world. In contrast to the more general approach of Eclipse it focuses on the development of EMIS-related systems and components. Since this framework is designed for EMIS and has a growing community of participating developers, it allows for an easy coupling of the Emission Trade Assistant with other Empinia-based EMIS components. Therefore it will e.g. be possible to integrate components of EMIS tools developed at ifu Hamburg and the HTW Berlin. These tools allow to model, analyze, simulate, and visualize material and energy flows in terms of material flow networks (Möller [2000]), Sankey diagrams, and transaction-oriented simulation models. Coupling EmTrAs to modelling and simulation tools could e.g. ease the creation of a facility's monitoring concept, or allow facility operators to better estimate future emissions. Figure 2 illustrates a hierarchy of plug-ins for the Empinia-based Emission Trade Assistant and a basic shippable configuration (red marking).

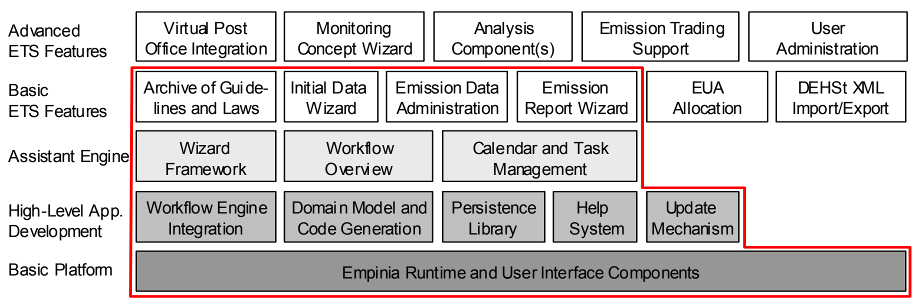

Figure 2. Plug-in based EU ETS Assistant. Application-independent components are shaded gray while ETS-specific components are white. (Figure in the style of Schnackenbeck et al. [2007].)

5. WORKFLOW MODELING AND IMPLEMENTATION

The most important requirement on the designed software assistant is a high flexibility and a straightforward update mechanism in response to the permanently changing regulations in ETS. Users should be able to adjust the workflow to their personal demands. This requirement can be satisfied if the workflow process is not hard-coded but dynamically loaded. Additionally, an easy manipulation of the workflow model and the corresponding dialogues should be possible via graphical and XML editors, and the applied workflow modelling language should support various selected workflow patterns: Sequences, parallel splits, exclusive choices, deferred choices, synchronization, synchronizing merge, cancel activities, cancel cases, implicit termination, arbitrary cycles and milestones (van der Aalst [2003], Workflow Patterns Initiative [2009]).

An ETS workflow consists of tasks like: requesting or importing emission data, automatic generation of obligatory emission reports for national authorities, interaction with obligatory systems like the VPS, and additional steps for interacting with an existing facility's IT infrastructure. Technically, these tasks are called workflow activities. An activity can contain a set of further activities, which enables hierarchical modelling. Furthermore, the user should be able to jump back to previous activities in the workflow, and – if possible – change the entered data. Re-entries in the workflow open the strict sequential processing and reduce the disadvantages of the automata approach by providing the user with more autonomy. The price for this flexibility is that previously followed branches and related data may become invalid when a re-entry to an earlier activity is performed.

To permit a straightforward integration with Empinia, we chose the Microsoft Windows Workflow Foundation 4.0 (MS WF) and .NET 4.0 (see MSWC [2009]) as a basis to implement the ETS workflows. MS WF 4 is easily extensible and includes a relatively stable visual workflow editor. This editor allows to model workflows not only within the Visual Studio 2010 IDE, but can also be embedded as a component into other applications. Hence, even end users of the Emission Trade Assistant could configure their own workflows. Workflow models and related assistant forms are persisted using the XML dialect XAML.

Modeling wizard assistants with a workflow framework is one of our primary objectives. A wizard will ask for user input by means of forms and process the input automatically. In the previous section, it was shown that this kind of assistant is needed for various tasks. To support the modeling of wizards we wrote a prototypical workflow extension called *WizardFramework* that supports re-entries into a workflow. Since wizard workflows are activities themselves they can be seamlessly embedded into larger workflows. Each call to a wizard form page corresponds to an activity of the newly defined type *ShowWizardPage*. We also provide extensions to model wizard workflows in the visual workflow editor, which automatically generates the corresponding XAML code. The wizard pages can be configured via the visual representation of the *ShowWizardPage* activity. This allows for an easy re-use of existing forms by parameterization in the editor (e.g. to set a domain-specific header and help text on a predefined form for entering addresses).

During the implementation of the wizard framework we found the solution to some problems particularly burdensome. Making it possible to return to a previous step within a wizard without having to interrupt and resume the workflow demanded the implementation of a custom control flow activity named *WizardSequence*. A common workflow sequence in MS WF can be processed only in one direction. The flow chart activity is more powerful but needlessly difficult to use for simple wizards since the execution order is modeled as a directed graph. As a compromise, our wizard workflow can be modeled as a sequence, but offers a built-in possibility to return to previous steps in the workflow. Another problem was to equip the *ShowWizardPage* activities with the ability to control the graphical user interface thread, and to suspend the workflow thread until a user interaction occurs. Although we were surprised by the effort of this work, it resulted in a tool that allows to integrate form-based wizards with automated processing of user data into workflow modelling.

The integration of a workflow engine adds to the Empinia platform for two reasons: As described above, MS WF and the additions of the wizard framework will provide a convenient way to visually build user friendly assistants that control Empinia plug-ins. Furthermore workflows allow to structure functionalities spread over different plug-ins in a task-specific way and thereby improve the understandability of the overall application (Simmendinger et al. [2007]). To embed workflow concepts into Empinia, we implemented a prototypical *workflow runner service* that provides a common interface to access workflows and workflow engines contributed by other plug-ins. We also implemented a simple integration plug-in for the MS WF engine, an ETS-specific plug-in containing a simple prototypical wizard workflow, and successfully invoked it via the service. Extensions will be necessary to support the proper synchronous and asynchronous execution of workflow threads and to provide a generic interface to observe state changes of workflow instances.

6. CONCLUSION AND OUTLOOK

In this paper we have presented our draft of ETS assistant software. We have started to define exemplary requirements in the context of ETS, which has been rated complex and time-consuming by facility operators. We described our definition of assistant software, and depicted necessary components needed in EU ETS. Particularly, we described our experiences in using Microsoft Windows Workflow Foundation 4.0 for implementing a key component called wizard framework and its integration into the plug-in platform Empinia.

In our joint two-year project, the requirements analysis will soon be extended to the needs of aviation operators, whose inclusion into EU ETS is about to become mandatory in the near future. These operators will be allocated specific aviation allowances that cannot be sold to conventional facility operators. Aviation operators will, however, be able to use 'normal' EUAs to balance their emissions. Another exception of aviation ETS is that even non-European airlines will have to own European aviation allowances, as long as their aircrafts approach EU countries.

Another aspect is the development of analysis components that support the decision-making of facility operators, e.g. for buying and selling allowances. Our focus is on simulating the process of Emission Trading by means of the simulation framework DESMO-J (Joschko et al [2010]). Plant operators will only be willing to use the software assistant, when it is stable and reliable, proves easy installation and use, integrates well into their specific IT landscape, and reduces the effort of their daily work.

ACKNOWLEDGMENTS

The presented research is funded by the German Federal Ministry of Education and Research (BMBF) with support code FKZ 01LY0819 A/B. The authors would like to thank the BMBF and the DEHSt for their support.

REFERENCES

van der Aalst, W. M., ter Hofstede, A. H., Kiepuszewski, B., Barros, A. P., Workflow Patterns, *Distrib. Parallel Databases 14, 1, Jul. 2003, (X)*, 5-51. , 2003.

Busse, T., Denz, N., Page, B., A Plugin-Based Framework for Domain Models and Persistence in Environmental Management Information Systems, in: Möller, A., Page, B., Schreiber, M.: *Environmental Informatics and Industrial Ecology (EnviroInfo 2008)*, Shaker, Aachen, 593-602, 2008.

Empinia, Internet: http://www.empinia.org, last access 06-29-09.

Engels, A., Knoll, L., Huth, M., Preparing for the ‚Real' Market: National Patterns of Institutional Learning and Company Behaviour in the European Emission Trading Scheme (EU ETS), in: *European Environment Nr. 18/5*, New Jersey, Wiley, 276-297, 2008.

KBSt, Standards und Architekturen für E-Government-Anwendungen - SAGA Version 3.0, Internet: http://www.cio.bund.de/cae/servlet/ contenblob/83726/publicationFile/5152/ archiv_saga_3_0_download.pdf, last access 30.06.2009, 2009.

MSWC, Microsoft Windows Workflow Foundation, Internet: http://msdn.microsoft.com/en-us/netframework/aa663328.aspx, last access 30-06-09.

Möller, A., *Grundlagen stoffstrombasierter betrieblicher Umweltinformationssysteme*. Dissertation am Fachbereich Informatik der Universität Hamburg, Projekt-Verlag, Bochum, 2000.

Joschko, P., Kruse, S., Page, B., "Modellierung und Simulation des EU-Emissionshandelsmarktes mit einer agentenbasierten Erweiterung zu DESMO-J", in Wittmann (Eds): *Proceedings of Workshop Simulation in Umwelt- und Geowissenschaften*, March 18-19, Shaker Verlag, Aachen 2010.

Page, B., Schmitz,, C., Joschko, P., Denz, N., "Design of an IT-Assistant System for CO2 Emission Trading", in Wohlgemuth, Page, Voigt (Eds.): Proceedings of the 23rd International Conference Environmental Informatics - Informatics for environmental protection, sustainable development and risk management Volume 2, Sep 10-12, Shaker Verlag, Aachen 2009, pp. 25-32.

Schmitz, C., Page, B., Rolf, A., Development of a Software Assistant supporting the CO2 Emission Trading System, in: *Proceedings of the International Conference Information Technology and Climate Change 2008*, 2008.

Schnackenbeck, T., Panic, D., Wohlgemuth, V., Eine offene Anwendungsarchitektur als Fundament eines Methodenbaukastens für betriebliche Umweltinformationssysteme, in: *Simulation in den Umwelt und Geowissenschaften*, Shaker, Aachen, 49-59, 2007.

Simmendinger, F., Cabac, L., Duvigneau, M., Knaak, N. (2007): Controlling OSGi Bundles with Petri Nets, in D. Moldt et al. (eds.): *Proc. Intern. Workshop on Petri Nets and Software Engineering (PNSE'07)*, pp. 220-225.

Wegehenkel, L., Coase-Theorem und Marktsystem, Mohr, Tübingen, 1980.

Workflow Patterns Initiative, http://www.workflowpatterns.com, last access 30-06-09.