EnviroInfo 2009 (Berlin)
Environmental Informatics and Industrial Environmental Protection: Concepts, Methods and Tools
 ISBN: 978-3-8322-8397-1

Design of an IT-Assistant System for CO_2 Emission Trading

Bernd Page[1], Christian Schmitz[1], Philip Joschko[1], Nicolas Denz[2]
[1]University of Hamburg, Department of Informatics
Vogt-Kölln-Str. 30, D-22527 Hamburg
[2]ifu Hamburg GmbH
Große Bergstraße 219, D-22767 Hamburg
[page, joschko]@informatik.uni-hamburg.de, schmitzmedia24@web.de, n.denz@ifu.com

Abstract

This paper examines the EU emission trading scheme with a focus on IT support tools from applied computer science. We claim that a software assistant is a suitable approach to help reducing the complexity of the European emission trading scheme from the point of view of facility operators. We present the preliminary design of an assistant system called EmTrAs (Emission Trade Assistant) that will guide facility operators through the 'jungle' of the emission trading system step-by-step. To allow for a timely adaptation of the assistant system to changing requirements (e.g. due to legal changes in the emission trading scheme) we consider an implementation based on workflow technology and component-based software engineering.

1. Introduction

The ongoing discussion about climate change deals with solutions to lower greenhouse gas emissions as well as business incentives to apply these solutions. While monetary aspects have traditionally guided the

economic consumption of resources, an *ecologically* responsible handling of emissions has for a long time been bound to legal conditions only. Emission trading strived to introduce market mechanisms into the environmental domain and is nowadays one of the key instruments of environmental protection in Europe.

The University of Hamburg and the ifu Hamburg GmbH have been co-operating in projects on environmental informatics for many years. In this paper, we outline our investigations into the European scheme for emission trading, from which we derive requirements on an *Emission Trade Assistant* (EmTrAs) software. In a two-year project funded by the German Federal Ministry of Education and Research (BMBF), we will analyze the CO_2 emission trading system from the point of view of applied computer science. The working processes and the duties of the participating roles (such as facility operators) will be identified. Alternative IT approaches will be examined in order to find appropriate solutions to support facility operators in their duties. We assume that especially software assistants are a highly suitable approach: On the one hand, specific assistants can aid facility operators in single tasks like initial data collection, certificate allocation, emission reporting, or trading of emission allowances. On the other hand, assistants based on concepts from workflow technology (see e.g. Dumas et al. 2005) can provide guidance through the complex overall process of emission trading.

Environmental Management Information Systems (EMIS, e.g. Wohlgemuth 2005) are on the rise to play an essential role in the planning and accomplishment of environmental protection. These frameworks provide tools to support the collection, documentation, assessment, and management of enterprise environmental protection information. Integrating an Emission Trade Assistant into an EMIS could especially ease the collection and analysis of data as part of facility operators' daily work. In our project the component-based open source EMIS framework Empinia (e.g. Schnackenbeck et al., 2007) will be used to couple the assistant to other EMIS components.

The following presentation starts by reviewing key concepts of the European emission trading scheme and the supporting German IT infrastructure in Section 2. In Section 3 we summarize related work with a focus on existing IT solutions. Section 4 − as the main part of the contribution − presents

requirements on our assistant software and elaborates on ideas related to its UI design as well as the workflow- and component-based implementation. In Section 5 we conclude the paper and describe upcoming project tasks.

2. Emission Trading

2.1 Concept of Emission Trading

Emission trading basically assigns a financial counterbalance to the emission of pollutants. Like costly natural resources force facility owners into a responsible consumption of raw materials, the artificial shortage of emission rights brings them to care about their emissions.

The limitation of emission rights (which is called *Cap*) by an authority is the most important mechanism to control the amount of pollution. The *Coase theorem* (Wegehenkel, 1980) implies that market participants will find the most efficient way to reduce emissions on their own, when they are dealing with a limited amount of pollution rights according to the laws of the free market. Thus, the most considerable reduction of emissions will be gained where it can be achieved with minimal costs (Schmitz, 2008). To gradually improve the environmental situation, the Cap can be reduced stepwise over the years. In 1968, the Canadian economist J.H. Dales firstly introduced the idea of freely tradable and limited pollution rights (see DIW Berlin, 2007). Although he only set up his model for water pollution, the same mechanism is used for air pollution in emission trading 40 years later. Today, the market-based emission trading scheme has become a key instrument of the EU to accomplish the objectives of the Kyoto Protocol.

The Kyoto Protocol itself contains an emission trading mechanism where so-called *Assigned Amount Units* (AAUs) are allocated to states. AAUs can be traded between states but cannot be bought by companies. In contrast, the European emission trading system (EU ETS) obligates selected business sectors with high emissions, such as power generation or combustion plants, to cash every emitted tonne with a so-called EU allowance (EUA). To prevent the economic system from being overcharged by the new cost type, every company taking part in EU ETS can request for freely assigned EUAs. The contingent of EUAs that a company receives for free depends on the business sector and the efficiency of the emitting facility. Companies that emit a larger amount of CO_2 than granted by their free EUAs have to buy additional emission rights from other facility operators with a surplus of allowances. By defining upper emission limits and by trading emission allowances (Cap and Trade) the market is expected to automatically find the most efficient and affordable ways to reduce emissions.

2.2 Process of EU Emission Trading

The process of emission trading is rather complex and varies in the participating countries. This paper exemplary deals with the German trading scheme, but the process is comparable to other countries. In Germany, the German Emission Trading Agency (DEHSt) as part of the Federal Ministry for the Environment is the main authority to administrate the national allocation of emission certificates.

The European scheme is based on allocation periods. Currently the second allocation period is in progress, which lasts from 2008 to 2012. The following third allocation period will last from 2013 to 2020. Before taking part in EU ETS, a facility operator must determine historical emissions for a given time period and have them approved by an accredited expert. Ahead the beginning of a new allocation period, the national governments settle National Allocation Plans (NAP) based on known data from facilities already taking part in EU ETS as well as data collected from new facilities. The NAP quantifies the overall certificate allocation for the period. It must fulfill the emission reduction targets (the Cap) and therefore be approved by the European Commission.

A broad survey of obligatory and optional tasks that facility operators must perform during the emission trading process is shown in Figure 1. To apply for free emission certificates, an operator

must submit an allocation request by a given deadline. The allocation request also has to be approved by a certified expert prior to submission.

The DEHSt annually transfers free allowances to facility operators. If the emissions of the facility differ from the freely allocated EUAs, a facility owner can buy missing EUAs from others, or sell a surplus of allowances respectively. EUAs can be traded (1) directly between facility operators, (2) over the counter (OTC) via a broker, or (3) on an emission trading stock exchange.

Facility operators have to submit their emission reports and monitoring concepts. Both items must be approved by a certified expert, who enters the verified data into the public emission trading registry. This registry allows every EU citizen to examine emissions and freely assigned EUAs. The final task of a facility operator in the annual emission trading process is to transfer the consumed EUAs back to the DEHSt. A facility owner who fails to transfer the amount needed to cap his emissions in time must pay a penalty of 100 Euro per tonne of CO2 at present. This penalty will presumably rise in the next allocation period.

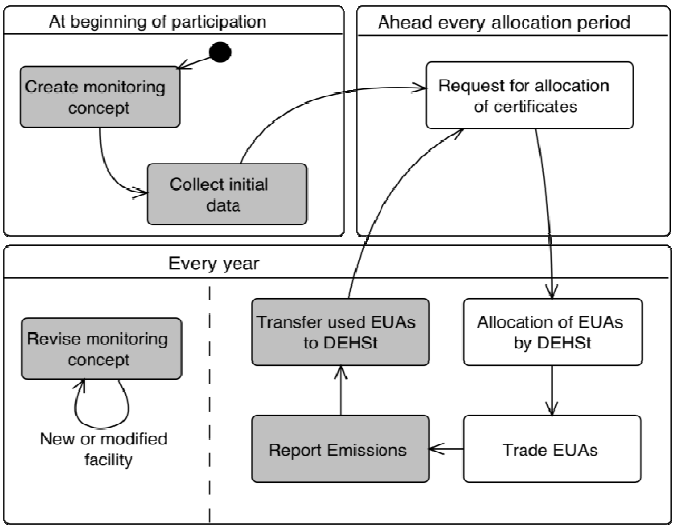

Figure 1: Workflow of EU-Emission Trading in Germany displayed as a UML state chart diagram. Gray rectangles represent obligatory tasks while optional steps are displayed in white.

2.3 IT Infrastructure in Germany

The DEHSt provides a software infrastructure which is legally binding for emission trading. Due to this obligatory status, our assistant software will have to provide interfaces that allow to enter and to communicate about emission-related data via the DEHSt systems. The three main components of the infrastructure are briefly introduced in the following.

The *Form Management System* (FMS) is a web-based application used by several German authorities, which can be customized for specific application domains. In the German ETS, the use of the FMS on the homepage of the DEHSt 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. The XML-based import and export is important since it provides an interface to connect external software systems to the FMS. To ease the connection of third-party systems, the DEHSt publishes detailed documentations for the XML interfaces of each web form type on their homepage.

The *Virtual Post Office* (VPS) provides encrypted electronic communication (see KBSt) and is also used by several German authorities. This Java-based client/server application differs from conventional email clients in two substantial aspects: Firstly, it is a closed system. This means that only previously registered users can be addressed. For each role in the EU ETS, a specialized version of the VPS exists, which can be downloaded free of charge. Secondly, there are specific message types like the transmission of an allocation request or the emission report. All communication between registered

users is encrypted and signed with the help of a public-key-infrastructure based on obligatory electronic signature cards. The electronic signature guarantees the authenticity and validity of the transferred data and fulfils the requirements of the German Signature Act (SigG). This means that it is officially recognized like a handwritten signature.

The *Emission Trading Registry* is a separate web-based application, which can be used like an online banking system. Instead of money, emission allowances (EUAs) are transferred from source to recipient accounts. Every facility operator who participates in the emission trading scheme automatically obtains an account. All national ET registries of the participating EU member states (currently 27) are connected to the central European registry called Community Independent Transaction Log (CITL). Every national or inter-European transaction is processed by the CITL. All national and European ET registries are public and can be inspected by every EU citizen via the internet.

3. Related Work

At the University of Hamburg, EU ETS is also subject of research at the Faculty of Economics and Social Science, where a research group surveyed the behavior of companies in Germany, the United Kingdom, the Netherlands, and Denmark. This study uncovered some facts, which are relevant for the objectives of our project as well: In interviews, companies were asked 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 partly depend on German mentality, but German bureaucracy certainly influences these results as well.

Another interesting fact is the willingness of facility operators to trade with EUAs. According to Engels et al. (2008) only 43% of German companies involved in EU ETS traded with EUAs in 2006, while this rate was significantly higher in the UK (74.2%), in Denmark (60%), and in the Netherlands (71.4%). This result is further evidence that the workload for German companies is high and that the legal and temporal structure of EU ETS in Germany is rather complex. A supporting software assistant could be one approach to improve this situation.

Before starting our project we reviewed and categorized existing IT-solutions that claim to support EU ETS. In discussions with certified experts, with the DEHSt, and with facility operators, we could not identify a popular all-in-one solution yet. Nevertheless, a number of tools support single aspects of EU ETS. Overall, we found less than a dozen of commercial and proprietary software tools. Only one open source software was found, which had already been discontinued due to changes in the German ETS process. Due to page restrictions, we omit the detailed review here and only present a summary of results.

Most reviewed tools exclusively focus on emission reporting and support no other aspects of EU ETS such as monitoring concepts or allocation requests. Even emission reporting is not supported consistently, as the output often does not comply with the FMS XML format required by the DEHSt. Some software manufacturers argue that the FMS format would change too frequently to keep interfaces up-to-date. Obviously, the effort to adapt a software system to changes in the FMS depends on the way the modelled workflows are implemented. In Section 4.2 we describe our vision to tackle this problem.

Most reviewed software systems are form-based and do not include assistant functionality. In particular, these tools do not guide the user through the complex legal and temporal structure of EU ETS, and do not notify him about impending tasks. The user must know exactly about upcoming tasks and their deadlines as well as the available obligatory software tools to perform them. We envision a comprehensive software assistant, that (1) helps the user to get a survey of outstanding tasks, (2) provides support and step-by-step guidance in solving them, (3) interacts with the provided software solutions of the DEHSt, (4) produces the correct output format of data, and (5) includes a straightforward update mechanism. To the best of our knowledge, a solution with this comprehensive functionality does not yet exist.

4. Design of a Software Assistant for Facility Operators

4.1 Specification of the Software Assistant

Based on the above considerations, we introduce a draft of an assistant system for supporting facility operators. Schmitz (2007) analyzed the central workflows for the facility operators' different duties in EU ETS to set up this draft. 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 (see Schmitz 2008):

Definition: A software assistant is a distinct (sub-) system with the purpose to support 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.

A guiding software assistant can help facility operators in coping with EU ETS most efficiently, when it covers all relevant duties in the complex process of emission trading. To map the whole process, all duties of facility operators must be modelled with their typical workflow. This includes the following tasks:

1. Administrative preparation: registration at DEHSt, Software installations (like VPS)
2. Creation of the monitoring concept and early verification through an expert.
3. Periodic revision of the monitoring concept, especially when the facility is changed and prior to each emission report.
4. Initial data collection: clarification if there is a need to participate in EU ETS, or if a voluntary participation provides advantages.
5. Allocation process: request for allocation and grant of the allowances.
6. Emission reporting: An adequate time scheduling is highly important to meet the given deadlines and avoid penalties for late submission or missing approval.
7. Trading with emission certificates.
8. Refund of used allowances to the DEHSt.

In this context, the facility owner is obligated to submit the necessary data for initial data collection (according to DEV 2012), for allocation requests (according to ZuG 2012) and for emission reporting (§5 TEHG) via FMS. According to Schmitz (2008) the assistant must appropriately map these forms and ease data entry with the following techniques: (1) Provide correlating introductory texts, help dialogs, and explanations for each field. (2) Define a logical course of data entry steps and a content-specific structuring of jointly or separately acquired fields. (3) Skip unnecessary fields, dialogs, and explanations that are not required for a specific facility, activity, or material flow. The assistant should also support the XML interfaces of the corresponding FMS forms to enable the export of the acquired data into the FMS.

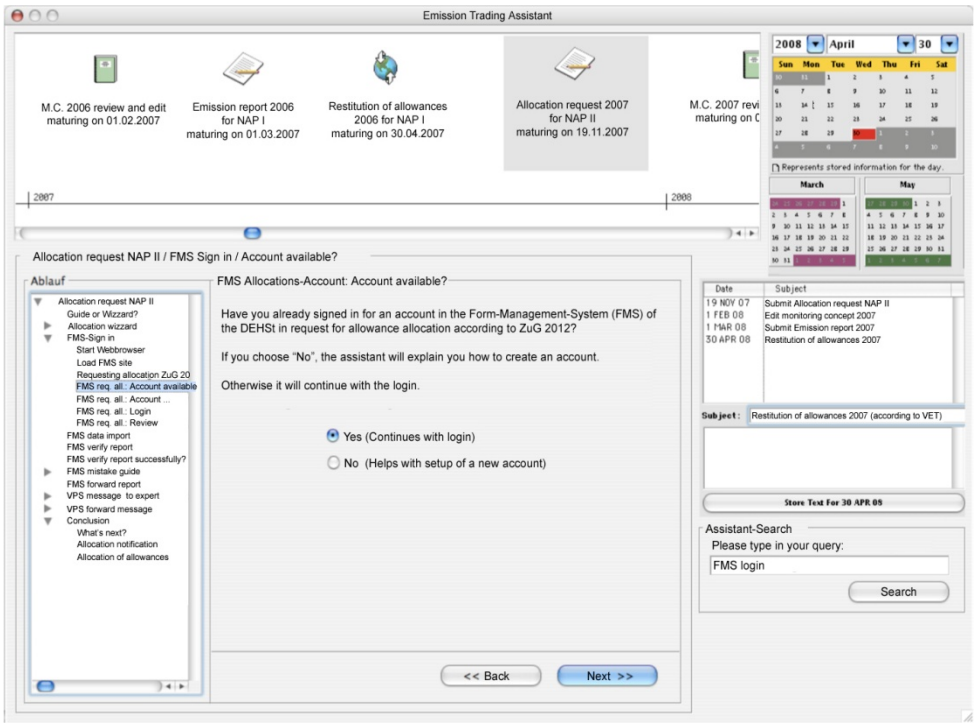

Figure 2: UI prototype of the Emission Trade Assistant: All sections and all graphical elements, including the hierarchical workflow navigation already exists, but do not provide functionality yet.

Figure 2 shows a mock-up of a possible graphical user interface (UI) design. The UI prototype combines different software tools that aid facility operators in fulfilling their EU ETS-related duties. A timeline-based navigation helps to handle the complex time course of emission trading with its overlapping cycles. All duties are presented with named icons and due-dates on a horizontal line. An additional calendar reminds the facility operator of close deadlines. When choosing a task from the timeline, the corresponding assistant will be shown in the central area of the UI. We expect especially two types of assistants to be useful in this context: While wizards are used during setup and data entry tasks, guides support the user with advice and instructions about how to perform certain activities outside the software assistant. Next to the currently active wizard or guide, the hierarchically structured overall workflow is shown in a tree view. Thus, the user is able to orientate himself in the workflow and to return to prior steps if necessary. A text input box will permit full text search through all wizards and guide dialogs.

After each step in a workflow, data must be validated and saved. In case of validation failure, rule based assistants could offer help to correct the entered data, e.g. by inserting additional workflow steps or by showing additional guides. This possibility enables re-entries into the workflow and complements the strictly sequential processing of automata-based approaches with more autonomy for the user. Importing data into the software assistant with the help of flexible interfaces to third party software (e.g. for database access) would also help to reduce work by re-using existing master data like e.g. addresses.

The assistant will provide a multi-user concept to distinguish between experienced and novice users as well. Other roles of the emission trading process such as accredited experts (verifiers) might also be considered. Depending on the user type, specific (parts of) guides and wizards could be displayed or hidden. User specific settings will allow for the configuration of individual dates or notifications, thus giving every user the individual preparation time he needs to perform a specific task.

4.2 Modeling the Workflows

One important requirement on the designed software assistant is a high flexibility and easy changeability in response to the permanently changing regulations and IT implementations (e.g. XML interface definitions) in EU ETS. This requirement can only be satisfied if the definition of the workflow and the corresponding dialogues are not hard-coded into the system. We will therefore use a visual or script-based workflow definition language as well as an interpreter system that dynamically loads the needed workflow definitions into the system and enacts the modelled workflows (workflow engine). One option is an XML-based workflow definition language like the XML Process Definition Language XPDL or the enhanced version XML Workflow Definition Language XWDL (WfMC, 2007).

We approach the decision for an appropriate workflow language and engine from different sides: Firstly, we develop a detailed model of the processes that an (idealized) facility operator must follow while taking part in EU ETS. This model will be refined and validated in interviews with German facility operators. Secondly, we will identify the workflow patterns (van der Aalst et al., 2003) that are relevant in EU ETS. Based on evaluations as provided by the Workflow Patterns Initiative (2009), it is straightforward to choose an appropriate workflow language and -engine that support the relevant patterns. Thirdly, we will identify practically applicable workflow engines that are compatible with Microsoft .NET as the target platform for implementation. One candidate is the Microsoft Workflow Foundation (MSWC, 2009).

4.3 Implementation with the Empinia Framework

Empinia (see Schnackenbeck et al., 2007; Empinia, 2009) is a software framework for the Microsoft .NET platform that was developed at ifu Hamburg, the HTW Berlin, and the University of Hamburg. Broadly speaking, the purpose of Empinia is to provide a platform for the component-based development of interactive rich client applications under Microsoft .NET; quite similar to the popular Java-based *Eclipse Rich Client Platform* (Eclipse RCP, 2009).

The main unit of modularization in Empinia is a *plugin*, i.e. a self-contained software component that can be added to or removed from an application at runtime (e.g. Schnackenbeck et al., 2007, p. 50). The runtime system consists of a small core with the only task to load and manage plugins. As in Eclipse, all application-specific functionality is provided by *extension* plugins that dock into *extension points* of others. Besides the runtime system, Empinia provides a set of general-purpose *services* (e.g. for object creation or the storage of application preferences) and a graphical *workbench UI*. Since all Empinia components are defined by general interfaces, it is quite straightforward to substitute the underlying implementations, e.g. to use different widget toolkits for the UI.

Empinia provides several components that can be employed for the implementation of the EmTrAs software. For example, an included framework for the model-driven development of business objects (Busse et al., 2008) will be used to implement the data model underlying the assistant. The framework includes generic mechanisms to persist business objects in databases or XML files, which will ease the linking of EmTrAs to external systems mentioned in Section 4.2. Domain-specific components providing EMIS functionality like material management or material flow analysis are under development as well (e.g. Schnackenbeck et al., 2007, Section 3.4).

Generally, the embedding of EmTrAs into existing IT infrastructures can be achieved in two ways: Interfaces to external systems can be developed as Empinia components and plugged into a standalone EmTrAs application. Alternatively, EmTrAs itself can be integrated as a plugin into any EMIS application built upon Empinia. This possibility is briefly sketched in Section 5.

The EmTrAs project is also expected to contribute new general-purpose software components to Empinia. One contribution will be the integration of a workflow engine into the plugin platform. Simmendinger et al. (2007), who presented a similar integration in the Eclipse context, argue that a workflow engine can impose an overall control flow upon an 'unordered' set of plugins and thereby improve the development and usability of plugin-based applications. We expect the workflow engine to be especially useful for the development of assistants for other Empinia-based applications.

5. Conclusions and Outlook

In this paper, we have presented a concept for a software assistant that will aid facility operators in fulfilling their duties related to the EU emission trading system. We have started to define exemplary requirements in the context of the German emission trading process, which has been rated complex and time-consuming by facility operators interviewed in a sociological study. Furthermore, we have presented first ideas to build an assistant that is easy to use by facility operators and straightforward to maintain and change by software developers. The applied methods will include assistant concepts like guides, wizards, and rule-based assistants, as well as workflow modelling and component-based implementation.

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 cap 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.

To model and implement the overall emission trading process, we will have to work through approximately 1000 pages of legal texts and have different industrial partners test our concepts. Facility operators will only be willing to use the software assistant, when it is stable and reliable, proves easy to install and to use, integrates well into their specific IT landscape, and reduces the effort of their daily work. To achieve this goal, iterative prototype-based software development (Pomberger und Pree, 2004) and agile project management (Scrum, e.g. Schwaber 2004) will be applied in cooperation with facility operators.

Another future focus is the development of analysis components that support the decision-making of facility operators, e.g. for buying and selling allowances. Due to the use of Empinia as a common technical basis, it will be possible to integrate components of the tools Umberto and e!Sankey developed at ifu Hamburg into EmTrAs. These tools allow to model, analyze, and visualize material and energy flows in terms of material flow networks (Möller, 2000) and Sankey diagrams. Complementary, it might be interesting to integrate the assistant into an existing EMIS such as Umberto. Another example is the simulation tool Milan for an integrated ecological material flow analysis and economic production simulation (Wohlgemuth, 2005), which is currently re-implemented with Empinia at the HTW Berlin. An integration of EmTrAs into this tool would allow facility operators to better estimate future emissions while at the same time planning an environmentally optimized production process.

Acknowledgement

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 for their support.

References

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

Busse, T., Denz, N., Page, B. (2008): 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, pp. 593-602.

Deutsches Institut für Wirtschaftsforschung (DIW) Berlin (2007): Der Handel mit Emissionszertifikaten http://www.diw.de/deutsch/produkte/publikationen/school/docs/diw_school_2004_02.pdf, p. 16, last access 07-23-07.

Dumas, M., van der Aalst, W. M., ter Hofstede, A. H. (2005): Process-Aware Information Systems: Bridging People and Software Through Process Technology, Wiley-Interscience, Hoboken.

Eclipse Rich Client Platform, Internet: http://www.eclipse.org, last access 06-30-09.

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

Engels, A., Knoll, L., Huth, M. (2008): 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, pp. 276-297.

KBSt (2009): 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.

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

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

Pomberger, G., Pree, W. (2004): Software Engineering: Architektur-Design und Prozessorientierung, Hanser, München.

Schmitz, C. (2007): Entwurf eines Assistenten-Systems für den CO_2-Emissionshandel, University of Hamburg, Department of Informatics, Bachelor Thesis.

Schmitz, C., Page, B., Rolf, A. (2008): Development of a Software Assistant supporting the CO_2 Emission Trading System, in: Proceedings of the International Conference Information Technology and Climate Change 2008.

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

Schwaber, K. (2004): Agile Project Management with Scrum. Microsoft Press, Redmont (WA).

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. (1980): Coase-Theorem und Marktsystem, Mohr, Tübingen.

WfMC, Workflow Management Coalition, XPDL Support & Resources, Internet: http://www.wfmc.org/standards/xpdl.htm, last access 08-20-07.

Wohlgemuth, V. (2005): Komponentenbasierte Unterstützung von Methoden der Modellbildung und Simulation im Einsatzkontext des betrieblichen Umweltschutzes. Dissertation at the University of Hamburg, Department of Informatics, Shaker, Aachen.

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