Supporting Collaborative Business through Integration of Knowledge Distribution and agile Process Management

Thomas Sauer
rjm business solutions GmbH
68623 Lampertheim, Germany
t.sauer@rjm.de

Kerstin Maximini, Rainer Maximini, Ralph Bergmann
University of Trier
Department of Business Information Systems II
54286 Trier, Germany
{kerstin.maximini|rainer.maximini|ralph.bergmann}@wi2.uni-trier.de

Abstract:
Independent from specific application domains, similar requirements can be identified regarding information needs in collaborative business. For coping with generality on the one hand and domain specificity on the other hand the Collaborative Agent-based Knowledge Engine (CAKE) is currently developed that combines CBR, agent, and workflow technology in an innovative way. A first prototype of CAKE is meanwhile available, and its application in the domain of geographical information management is presented in this paper. The challenge in this domain is the support of processes that cannot be fixed in advance and are subject of continuous change during enactment as well as the seamless integration in the existing IT environment of the participating company.

1 Introduction

The aim of research in CSCW (Computer supported cooperative work) [Gru94, CS03] since the 1980s is to improve the way human beings are working together in a certain working environment by means of information technology. In a collaborative working environment, support for communication, coordination, and cooperation must be integrated systematically with the business processes of the organization. Furthermore, collaboration support must not be limited to humans but must also cover automated decision support systems and automated knowledge sources in a seamless fashion. This demands for a much stronger link between CSCW and knowledge management research.

Of course, the design of a collaboration environment for an organization requires a very thorough analysis of a) the ways people are working together to achieve a certain business goal and b) the ways people are (or are supposed to be) interacting with the knowledge sources available in the organization. To implement such tailored collaboration support in
a cost-effective manner, flexible tools are needed that are generic in the sense that they can be configured to the individual need of the organization. Besides having to support the basic CSCW tasks communication, coordination, and cooperation, these tools must integrate knowledge management activities such as context-specific provision of knowledge and the use of best practices, too.

The aim of one branch of research conducted at the Business Information Systems II research group at the University of Trier is to develop such generic collaboration support tool. Within various R&D projects systematic analysis of the requirements of collaboration support have been performed. It turned out that similar problems occur in quite different application domains: fire services, software engineering, medicine, and geographical information management. In all these domains there is a need to coordinate collaborative activities jointly together with the knowledge required for enacting these activities. Further it turned out that the flow of these activities is very often quite dynamic and subject of change depending on the current context. Thus there is a strong need for a highly flexible CSCW solution. The authors have identified the most important requirements in each application domain in detail, generalised them to a more abstract level, and addressed them during the development of a domain-independent system called **Collaborative Agent-based Knowledge Engine (CAKE)**.

A first prototype of CAKE is meanwhile available, and currently the system is applied in several application scenarios to prove and evaluate it. How it is applied in the fire services scenario has already been reported [FMS05, FMMS05]. This paper describes CAKE and how it is applied to support the participants in a long-term eGovernment project on geographical information management, executed by the company rjm business solutions GmbH. Goal of this project is to offer facts and spatial information about monument protection to the public. A group of five students from the University of Trier uses CAKE to develop a supporting system called **GIS-DOKU** [BOP05] that will improve the data acquisition and publishing processes and will provide automated documentation and flaw detection.

In the following section the general architecture of CAKE is presented briefly. Section 3 demonstrates how CAKE is put into practice in the eGovernment project and evaluates its practicability. Finally, related and adjacent research is discussed in section 4 and a short discussion closes this paper.

## 2 The CAKE system

The CAKE system combines workflow technology with agent and CBR technologies following a new and innovative concept. CAKE introduces a light-weight workflow and agent management approach backed by CBR, allowing to leverage arbitrary knowledge sources and query mechanisms already available within an organisation. Hence, organisations can be effectively supported in performing their real world activities at as little effort as possible, with established tools and practises being integrated rather than replaced. Information access and information routing can be seamlessly integrated into collaborative actions be-
between humans, or between humans and machines through the inclusion of software agents that feed knowledge relevant to the required action from heterogeneous sources. New collaboration and knowledge distribution strategies can be developed as required using workflow representations that are easy to comprehend. Furthermore, CBR technology provides efficient retrieval facilities that are needed to design flexible applications.

As a key feature, CAKE offers sophisticated strategies for information access and information routing. These strategies address the challenge that in many domains, organisational knowledge is scattered over a rich set of heterogeneous information sources, and that individuals requiring the knowledge might not even be aware of these sources. Another key aspect covered by CAKE is the formal representation and execution of light-weight processes. On the one hand, providing the necessary knowledge during process execution requires a certain formal foundation, on the other hand process descriptions should be easy to comprehend by the end users, and should provide a high degree of flexibility.

An overview of CAKE’s architecture, showing the interactions of the three key technologies workflow technology, agent technology, and CBR technology is shown in Figure 1. The underlying data model and these key technologies are explained in more detail in the following sections. The last section describes, how this architecture has been used to implement the CAKE system.

![CAKE System Architecture](image)

Data Model. All components in the CAKE system use the same data representation. As CAKE should be applicable in several very different application scenarios, a flexible and unified data representation concept for all components is paramount. This is put into practise with a data model and corresponding data objects.
The **CAKE Data Model** describes all kinds of data that can occur in the system. It is an object-oriented model using specialisation and aggregation to define the data classes. Available data classes are atomic classes like boolean, integer, double, date, or time as well as compound data classes like aggregates, collections, and intervals. These system classes can be used to define an application specific data model. Each specialisation of a system class is called **user class** that again can be specialised or used like any other class. Each data class of the data model can be instantiated as a **CAKE data object**. Consequently, each data object has exactly one corresponding data class that defines the possible value(s) of the object. Additionally to the “main value” of a data object, each data object can have **properties** to store additional information about the object.

**CBR Technology.** The main purpose of the CBR technology is the similarity based retrieval of data objects. For this purpose a similarity model can be defined on top of the data model by combining and configuring similarity functions predefined in a similarity library. The library already contains about 30 similarity measures like Hamming Distance, Simple Matching Coefficient, or sigmoid function. More information about the similarity measures available is provided by Bergmann [Ber02]. The retrieval engine features a linear similarity-based retrieval but the architecture has been designed with more sophisticated retrieval strategies in mind, which will be available in the near future.

**Workflow Technology.** The goal of the workflow system is to model and execute processes to manage the collaboration between agents. In CAKE, processes are described using workflow definitions that are instantiated and executed in concrete situations. These situations are represented in the so called context of the process.

Each workflow definition consists of initial context values, a workflow characterisation, a set of task descriptions, and the control flow relationships between the task descriptions. The latter allow to arrange the tasks in sequence, in parallel, or in loops, but does not cover data flow at all; data exchange is realised by using the context. The workflow characterisation is a semantic description of the workflow definition represented with CAKE data objects. Thus, it enables a semantic CBR-based search for an appropriate workflow definition in a concrete situation. A task is technically realised as Java class and hence a task description mainly consists of a task name, the name of the Java class that implements the task’s functionality, and a set of parameters that are required for its execution, e.g. the name of the context keys to access and store information. One of the capabilities of a task is to trigger further workflow definitions, so hierarchical decomposition of complex activities is enabled.

At runtime, the workflow engine manager initialises a workflow instance of a workflow definition. Each workflow instance is executed by an own workflow engine (WE) that controls the interpretation of the workflow definition by executing the tasks in the specified order. Separating workflow definition and execution enables modifications on the workflow instances without doing changes on their workflow definitions. Beyond controlling the life cycle and control flow of tasks, each WE administrates the local context assigned to its workflow instance and facilitates the capturing, storing, and changing of context-based data.
Agent Technology. The CAKE agent framework provides a unified interface to couple external knowledge sources as well as user interfaces with the core system. Technically, the framework distinguishes between information agents and user agents. Information agents provide and may also change knowledge like search engines, databases, groupware calendars, human experts, data mining tools, classifiers, or dialogue strategy tools. User agents request knowledge and are in general interfaces to human system users, e.g. graphical user interfaces or natural language interfaces, but can also be other computer systems. Each agent is composed of a technology component, a wrapper, and a competence profile. The technology component is the external knowledge system that is made accessible via CAKE. To enable the communication between CAKE and different technology components, wrappers are used as mediators that fulfil two tasks: firstly, to map between the technology components’ ontologies and the CAKE domain specific ontology and secondly, to realise the technical interface. The competence profile is a CAKE data object characterising the agents’ competencies.

The flexibility of the workflows is carried forward to the agent framework by providing a dynamic set of agents whereas agents are able to enter and leave the CAKE agent society. Therefore, the agent architecture is able to manage heterogeneous agents that have completely different purposes, knowledge and capabilities [vEDA04]. When intending to enter the agent society an agent has to register its competences that are stored into a central repository in the CAKE system. Hence, when looking for information or for a special service the user agents send their request to the workflow system. Based on the request and further information that is probably available in context one or several agents are searched by using the CBR technology and the agent competences as case base. How the requests are forwarded to the retrieved agents and how the results are merged or adapted is completely managed by the workflow system.

Implementation. The CAKE System Architecture has been used as a model to implement the CAKE System. The CAKE System is realised in Java and the implementation of the core system is nearly finished. Currently, the front ends and user interfaces are under development. The web interface allows to hot-plug agents in distributed environments, to visualise the actual agent society with all registered and available agents, and to browse available workflow definitions. A user-friendly modelling environment is provided by the CAKE Editor.

3 Using CAKE for Corporate Knowledge Distribution and Workflow Support

Recently, changes to law according to the regulation of civil works in the German federal state of Hessen does no longer demand owners to seek explicit permission for most refurbishment tasks on buildings. Rather, planners and architects have to ensure themselves that they comply with all regulations that apply to the corresponding land parcels, including monument protection.

In Hessen, each building and site of historic interest (e.g. old marketplaces, or charac-
teristic quarters) is listed in the official monument register. For each monument, its exact location is recorded, as well as a rationale why the monument is subject to protection, and which type of protection applies. For instance, a building may be protected in a whole, or only parts of it may be of historic interest (e.g. the cladding, or an annex). As yet, these facts have been published in print only, leaving many planners and owners unaware of monument protection. This invites inappropriate reconstruction, damage, or even complete demolition of cultural heritage.

To overcome this, rjm business solutions GmbH has been assigned by the Monument Protection Agency of Hessen to conduct the long-term eGovernment project DenkXweb. DenkXweb provides a freely accessible Internet service to publicly access the monument register\(^1\). It presents the protection rationale as described above as well as a detailed map denoting location and dimensions of the protected building or site. As laid down by law [EF05], the latter is based on the official cadaster as provided by the surveying and mapping authorities. Thus, using DenkXweb, users can look up whether a building is a monument by itself, or whether according land parcels are included in site protections (which will have implications on any construction on these parcels).

Besides developing the Internet service, rjm business solutions GmbH and its associate company rjm medienservice GmbH provide data management services as well in order to combine monument register data with the geospatial information provided by the land cadaster. That is, internal tools and procedures are developed to create data displayed by the Internet service most cost-effective, but still as accurate as possible. During everyday work, handling these procedures have proven to be highly knowledge intensive and require the coordination of collaborative activities between the participants. The hardest challenges are shortly sketched in Section 3.1. To tackle them, a student research project has been conducted to determine a formal representation of the processes used by the companies so far, to develop strategies to better distribute the needed knowledge to the participants, to provide automated documentation and flaw detection, and to implement a supporting system based on CAKE that can easily and seamlessly be integrated within the IT environment already in use. Thus, the subsequent Section 3.2 discusses workflow modelling and information agent design, and Section 3.3 explains how CBR technology is used to retrieve appropriate workflows and agents. This section closes with an illustrative example in Section 3.4.

### 3.1 Challenges in everyday work

When collating and preprocessing data for the DenkXweb Internet application, staff members face several problems. The most engraving are shortly sketched below.

**Inconsistency of data.** Geospatial data is maintained by 41 regional authorities, each responsible for a district or a part of it. As of this writing, all authorities provide their data on CD once a year using separated files to describe map data in ALK\(^2\) format (i.e.

---

1\(^\)http://www.denkmalpflege-hessen.de/denkxweb

2\(^\)ALK = “automatisierte Liegenschaftskarte”, graphical representation of real estates
points and connections between them to describe the spatial information of buildings, land parcels, etc.) and meta information in ALB\(^3\) format (e.g. postal addresses of parcels). ALK data is always provided by the regional authorities themselves, while ALB information may be provided by an additional governmental service provider. When handing over data for the first time, a full snapshot is sent, otherwise incremental updates are delivered. After having combined map data and meta information, the location and dimensions of monuments and protected sites have to be transferred from a printed map, often collected manually by specialists in the fields.

Thus, data inconsistencies are quite common, e.g. map and meta information often cannot be merged because of missing or contradicting descriptions or different granularity of the data, or transferring monument related data can fail because of differences between digital and printed map editions. In all cases, inquiries to the authorities are required, and while waiting for answer, the area in question cannot be processed further. Since the issues raised may require large efforts in finding a solution, final resolutions may take up to half a year, making it difficult and time-consuming to resume work.

**Determination of the current processing status and open issues.** Several programs and databases are currently used to document flaws and to store the duration needed for each piece of work: whenever an employee works on the project, he or she enters start and end time as well as a short textual description in a custom time recording tool. A table in an internal Wiki system stores information which ALK and ALB files have been received on CD, but could not have been processed so far. Other Wiki tables show course-grained progress information on handling ALK and ALB, e.g. whether merging has been started, or whether monument related information has been transferred using a CAD application. Flaws noticed during the latter transfer are described in several Excel spreadsheets.

This heterogeneous data storage makes it difficult to determine the exact status of progress, and it is hard to derive documentation for the project stakeholders. In particular, the latter is required to explain inconsistencies between monument related data and the map, and the consequences of these inconsistencies. For instance, if monument related data and the ALK map representation differs, the Monument Protection Agency has to be informed whether there has been a trivial change (renumbering of index number), or if a site of historic interest is no longer included in the map, requiring (expensive) inspection on site. Besides, the exact status of progress is required to estimate efforts for future pieces of work. Thus, an automatic documentation and open issue collection would be helpful.

### 3.2 Workflow Modelling and Information Agent Design

*GIS-DOKU* [BOP\(^+\)05] handles the above mentioned problems by using CAKE as a basis. As a first step during its development, workflow definitions have been developed to describe the data acquisition and publishing processes for the closed area of a commune. The top-level workflow definition is shown in Figure 2, represented as *Event-driven Process Chain (EPC)*.

\(^3\)ALB = “automatisiertes Liegenschaftsbuch”, textual information on real estates
Each of the three process steps is defined by a workflow definition of its own. The workflow definition of the first process step “Import ALK and ALK data” is shown in Figure 3.

Additionally, special workflow definitions for open issue resp. flaw handling have been modelled. These are not part of each workflow instance, but the corresponding process steps are dynamically inserted in case of occurring failures, e.g., a detected inconsistency of ALK and ALB data. This approach leads in the end to completely different workflow instances, reflecting the fact that also in the “real world” each commune is processed differently: in one commune, the wrong ALK data may have been received, but after the data was re-sent from the authorities, the remaining data acquisition is without any errors. In another commune, the process step “Import ALK and ALB data” can be passed without any problems, but then the transfer of monument protection data leads to inconsistencies that must be handled. In the third commune, the worst case happens and all possible issues occur.

The next step was to initialise workflow instances, representing the processing status for each of the 426 communes in Hessen. Here, three cases have been distinguished:

**Communes where the data acquisition was already done.** In these cases, the top-level workflow definition was instantiated and the responsible workflow engines “know” that the workflow is already finished. It is important not to remove these finished workflow instances, because they are necessary for the agent that is responsible for the determination of the current processing status.

**Communes where the data acquisition had not been started.** In these cases, the top-level workflow definition was instantiated and the responsible workflow engines wait for the event “ALK and ALB data are available”.

**Communes where the data acquisition was already running.** These were the most complicated cases, because the current processing status had to be determined manually.
In case the employees of rjm business solutions GmbH are waiting for an answer from the authorities after a flaw had occurred, the corresponding additional process steps have been inserted.

As an integral part of GIS-DOKU, information agents are developed to connect to the information systems already deployed at rjm business solutions GmbH. For instance, an information agent polls the time recording system for changes, triggering an event to launch or continue a CAKE workflow instance when detecting appropriate changes. That is, whenever employees record that data has been requested from the authorities, or that ALK or ALB data have been received, this information will be forwarded to the CAKE system, allowing subsequent workflow management. The technical component of the information agent performs the database login and time recording data selection, as well as deriving meaning from the details an employee entered. Finally, the latter is expressed using the CAKE data model.

Further, there are information agents to monitor all active workflow instances in order to determine the overall work progress. Inconsistencies and other open issues detected during workflow enactment are collared and stored in a newly created GIS-DOKU database, allowing easy lookup and generation of documentation as required. In the long run, this database is intended to replace the Wiki pages and Excel sheets currently used.

3.3 Retrieving Workflows and Agents using CBR

When preparing data for the DenkXweb Internet service, deviations of the workflows above have to be expected. As described above, each commune requires an individual strategy in order to overcome ALK and ALB handling problems, and issues arising from preprocessing and transferring monument related data. In addition, external events have to be handled, like communal authorities voting against publishing data for certain areas (e.g. because residents have objected against protection).

Using CAKE, handling these deviations is supported well, since workflows instances can be altered using late and ad hoc planning. Using CBR retrieval, CAKE helps to find appropriate workflow definitions which can be incorporated as exception handlers as described above. That is, employees can tailor workflow instances easily as CAKE suggests appropriate workflow definitions for a given situation. Furthermore, the CBR engine helps to find appropriate agents to put these workflows into practise. That is, the numerous information agents can be leveraged best, and human team members can be selected based on their characterisation profile as well.

The latter guarantees that when events are triggered, tasks that may be enacted afterwards are shared among the most capable agents within the organisation. For instance, monument register data will have to be edited as well, and there are experts for maintaining and updating textual parts and assigned images. Other agents are experts for entering graphical information using the CAD tool, thus they can focus on these tasks without being obliged to revise monument descriptions.
3.4 Example

Employee Alice starts to prepare data intended to be displayed by the DenkXweb Internet service. She requests geospatial information for a district from the authorities, and because monument protection information is available on a per-commune basis, the employee requests individual ALK and ALB files for each commune within the district. However, two different authorities handle the data transfer, thus ALK and ALB are received from two different sources. Alice tries to merge both ALK and ALB parts, however they do not match: While ALK data has been received in fine-grained areas as requested, the meta information contained in the ALB files has been returned as a course-grained description for the whole district. She enters her efforts using the time recording system, and notes that the geospatial data could not be merged.

Thus, further data preparation have to be suspended as the second authority is asked to send the ALB data as required. Since requesting detailed geospatial data in this amount is a rather unusual request, the contact person at the authority has to find out what went wrong first, so it takes up three weeks before the corrected data is received by Alice.

Meanwhile, Alice continued with preparing another district. After two weeks, the Monument Protection Agency asks management if data preparation for the first district is already done. The executive logs into GIS-DOKU, and reads the current status as “Cannot edit, because ALK and ALB data do not match” derived from the entries made by Alice using the time recording system. Thus, he can provide this information immediately to the project stakeholder.

Finally, ALK and ALB data arrive, and this time, they do match. Since Alice is still working on another district, CAKE is used to look for another agent capable in resuming work on the first district. Charlie has been characterised as an experienced CAD user, too, so by using the CAKE retrieval engine, he is assigned to transfer monument related data based on the ALK map. Anyhow, when marking buildings as protected using the CAD application, he is unable to locate a large quantity of buildings in the ALK map. He uses the CAKE CBR engine to find an appropriate workflow for handling this situation, finding the workflow definition “Check ALK map”. The latter instructs him to run consistency checks first, and how to run them. Furthermore, the workflow definitions contains information how to contact authorities via e-mail if consistency checks do not resolve the situation.

Charlie performs consistency checks, but the buildings are still missing. So he calls the authorities via e-mail as described by the workflow, leading that the ALK map is not finished yet in that particular area. However, he is told that the area will be covered by an update available later that year. He adds this information using GIS-DOKU and continues with other tasks, allowing his colleagues to mark the buildings now missing as soon as the new map edition will arrive.
4 Conclusion and Related Work

Numerous systems have been designed before targeting the challenges of agent collaboration, organisation-wide knowledge management, and workflow support. Jennings et al. [JNF+00] point out the advantages of applying agent technology to workflow systems like flexibility, agility, and adaptability. From a more technical view, Loke and Saslavsky [LS01] emphasise further advantages: component based extensibility, event/exception-handling, remote installation, and support for mobile devices. All the above mentioned advantages are exactly addressed by the CAKE agent framework.

Very similar to CAKE is Workbrain [WWT97] that combines workflow and knowledge management, aiming to built up an Organisational Memory. For structural planning a CBR application is integrated and can be utilised by a human workflow planner before the respective workflow will be enacted. But a counterpart to the light-weight agent concept is missing, which enables CAKE to integrate itself with tools already deployed in the organisation. Another similar approach is presented by Tony et al. [TSP04]. Unlike a traditional, hierarchical workflow system (there are assignments and authorities to report to) their approach is a collaborative multi-agent based workflow system. But a counterpart to the CBR based agent and workflow selection is missing, which enables CAKE to react flexibly to changing situations.

Although the first prototype of CAKE is meanwhile available, a lot of work has still to be done: the user interfaces must be improved, more external knowledge sources must be made accessible, and more evaluations have to be performed.

Acknowledgements. The authors acknowledge the European Commission for funding AMIRA under grant number FP6, project IST-2003-511740.

Furthermore, they would like to thank Janette Birkner, Maryja Ognyanova, Jacques Pütz, Annabelle Rücker, and Yangshen Yao from the student research group for their valuable feedback to CAKE and for being the first real beta-testers of the system.

References


“Similarities - Processes - Workflows” on the Sixth International Conference on Case-Based Reasoning (ICCBR 2005), Chicago, Illinois (USA), August 2005.


