Adaptive Workflow Management in the Cloud – Towards a Novel Platform as a Service

Mirjam Minor, Ralph Bergmann, Sebastian Görg

Business Information Systems II
University of Trier
54286 Trier, Germany
{minor|Bergmann|goergs}@uni-trier.de

Abstract In this paper, a novel workflow platform as a service is presented[1]. The paper describes how existing components of adaptive workflow technology are integrated and extended towards a cloud platform on which individual users as well as small business enterprises may model, execute and adapt workflows without the need to possess an own workflow management system. As the adaptive workflow technology involves several case-based assistant systems, a novel user model concept is required to provide a trustable and usable solution. An example provides first insights into the application of the novel concept.

1 Introduction

Cloud Computing is a means to provide and use IT infrastructure, platforms and applications in form of *cloud services*, i.e. electronic services available on the Web [1]. Cloud services like Doodle[2] or Google Apps[3] are getting increasingly popular, as they are easy to use at a low-cost base. Private individuals as well as business enterprises adopt the novel technology swiftly. According to Gartner analyst Mark McDonald, "cloud services and virtualization are […] the top-two technologies for 2011 and are well-suited for this budget reality, as they offer similar service levels at lower budget costs."[4] Subsequent to the success of the light-weighted cloud services in supporting particular business application functions following the Software as a Service (SaaS) paradigm, first cloud computing research groups and companies are taking the initiative towards a deeper way of supporting business processes. Workflow Management as Platform as a Service (PaaS) is set up by YML-PC [12] a

[1] The work is part of the WEDA project (**We**b-based **d**esign, reuse, documentation and execution of **a**gile workflows) funded by Stiftung Innovation Rheinland-Pfalz under grant no. 974.

[2] A service for scheduling meetings or other appointments, see www.doodle.com

[3] For instance e-mail and calendar services, see www.google.com/apps

[4] based on responses from 2000 CIO's in the 2011 CIO Agenda survey by Gartner Executive Programs (EXP), see http://www.gartner.com/it/page.jsp?id=1526414

programming framework and cloud middleware in which several applications (of scientific computing) can run in parallel controlled by a workflow. A similar platform for scientific workflows is Wings Marbles[5]. The standard software vendor SAP provides some of its applications in a mobile service manager that extends SAP's business suite by Sybase Unwired Platform [9]. Using Sybase mobile technology, users can develop mobile applications including SAP's workflow management components [10], i.e., the mobile service manager is able to play the role of a PaaS. However, the initiatives described above are still in an early stage and the considered workflows have an inflexible structure. There is definitely a research gap for a general-purpose, adaptable workflow solution in the cloud implementing the PaaS paradigm. This paper aims at closing the research gap by **taking adaptive workflow management into the cloud implementing a workflow platform as a service.** A *cloud workflow* consists of *tasks* executed by a human participant or by a (cloud) service, a *control flow* that specifies the execution order of the tasks, and a *data flow* specifying the input and output data of the tasks. Adaptive workflow management means that the workflows may be adapted at run time concerning tasks, control flow, or data flow [8,7]. In order to create benefits for private individuals and small business enterprises, such a novel workflow platform as a service demands high usability and trustability. This is a true challenge as (I) adaptive workflow management requires sophisticated technology including assistant systems for the creation, execution, and adaptation of workflows and (II) cloud workflows involve many participants in different roles for creating workflows, starting workflows, executing tasks, inviting other participants to execute tasks, adapting workflows, sharing prototypes for workflows and tasks, and so on.

The paper is organized as follows: Section 2 provides an introduction to adaptive workflow technology and discusses how existing components can be integrated with the cloud scenario. In Section 3, a novel user model concept is introduced and tested in Section 4. Finally, Section 5 summarizes the paper and discusses some open issues.

2 Adaptive Workflow Technology

The intended workflow platform as a service integrates existing core technology for adaptive workflow management from the Cake system (Collaborative, agile knowledge engine) [5]. This includes components for modeling, execution, and structural modification of agile workflows at run-time [6], similarity-based retrieval of workflows [7,2] as well as automated workflow adaptation [3,4]. In the following, we will briefly sketch the core concepts of Cake and their integration with the novel platform:

- The Cake *modeling language* considers the adaptivity by providing a suspension mechanism using breakpoints, which prevent the workflow engine from overrunning tasks that are about to be modified (see [7]). The graphical notation of the original Cake modeling language is based on UML

[5] http://seagull.isi.edu/marbles/

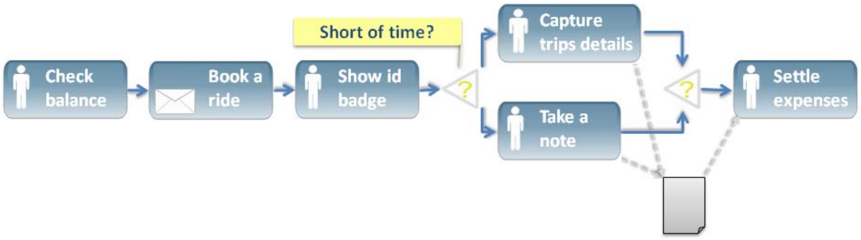

Fig. 1. Sample workflow in CFCN describing the transport of a disabled person.

activity diagrams. However, this notation seems too technically for the intended users of the platform. It has been re-designed with novel symbols for the same functionality and is now called Cake Flow Cloud Notation (CFCN). A sample workflow in CFCN that could be implemented in the platform is depicted in Fig. 1. It describes the workflow for a disabled person who gets benefits for rides to leisure activities [11]. It involves the disabled person, a cab service, and a home for the handicapped as a clearing house. The disabled person has a quarterly quota for such benefits. She starts the workflow with checking the balance (‘Check balance’). In case the quota has not yet been exhausted, she books the ride by sending an e-mail to a cab service (‘Book ride’). When entering the cab, she shows her ID badge to prove that she is eligible for benefits (‘Show id badge’). The cab driver then captures the trip details like name, date, and distance in a data object (‘Capture trip details’ with the attached file) or, if they are short of time, just takes a note (‘Take a note’ with the attached file). The file symbol is a placeholder standing for a database entry or simply a memo on the smart phone. After the ride has taken place, the cab driver settles the expenses (‘Settle expenses’).

- The adaptability of workflows leads to requirements for the workflow execution that cannot be met by traditional workflow enactment services. Cake provides an *agile workflow enactment service* that is able to deal with the adaptation of ongoing workflow instances (see [7]). This component can remain as it is to become part of the cloud platform.
- The creation of a workflow is supported by a repository of workflow prototypes. An *assistant component for workflow creation* based on our previous work on case-based workflow retrieval [7,2] is ongoing work.
- The adaptation of a workflow is supported by Cake’s *adaptation assistant*. The adaptation is performed in case-based manner, which enables the reuse of adaptation experience. We collect the experience from previous workflow adaptation episodes in a dedicated adaptation case base. An adaptation case stores the aim of the performed adaptation (change request description), the original workflow, the adapted workflow, as well as the difference of both workflows in terms of add and delete lists containing workflow elements. When a new adaptation request for a workflow occurs (as part of an initial workflow modeling activity or as consequence of an unforeseen event to a

workflow that is already under execution) this adaptation is performed in a case-based way, i.e., by automatic reuse of an adaptation case retrieved from the case base (see [3]). The adaptation assistant is to be integrated as a service provided by the platform.

- The graphical user interfaces have to be re-designed to fulfill the requirements of cloud users like extreme usability and trustability. Particularly, they have to be adapted to our new graphical language CFCN and as a different issue, they must be adaptedto be more suitable for use on mobile devices (e.g. smart phones. Further, a novel user model concept has been developed as a prerequisite for this re-design (compare Sect. 3).

3 User Model Concept

The user model concept is a means to maintain simplicity and trustability, which both are essential requirements in a cloud scenario. This is featured by a sophisticated management of access rights for groups and individual users. The basic idea is that every resource in the cloud platform (workflow, workflow tasks, and further resources) has an owner who is allowed to manage the access rights for the resource. The users may grant open access for the public to some of the resources. The authentification of the users is performed by a login with the email address following the practice of well-known cloud services like Google apps. Any user is allowed to invite novel users.

Fig. 2 depicts the user model concept for Cake citizens in the cloud in a UML class diagram. Cake citizens are either single users represented by an account (‘Account’) or groups of users (‘Group’) with several accounts as members. An account is identified by the email address and usually contains a password and a user name (‘CakeName’) to be displayed in the GUI, for instance when the user is assigned to a workflow task. If the user has also a Facebook account, the Facebook identifier may be stored in the Cake account as well. In case of occasional users, the account might consist only of the email address or the email address in conjunction with a Facebook identifier. Participants are groups or single accounts playing one or several roles in the Cake system:

- A participant may serve as an executor of a workflow task (HumanExecution), i.e., a single user or a group which has been assigned to the task in the traditional sense of a workflow participant or workflow role and is in charge of the execution of this task. Once the task is active, the assigned user receives it on her work list or - in case of a group - a load balancing module assigns the task to a group member and sends it to the according work list.
- A participant may have access to resources (‘Resource’) as an owner (‘accessRight.owns’) who serves as the contact for the resource and maintains the access rights for this resource, as a reader (‘accessRight.read’), as an author (‘accessRight.modify’) who is able to edit the content of the resource, or as an executor (‘accessRight.execute’) who is allowed to

execute the resource for example to start and stop a workflow instance ('WorkflowInstance').

Resources can belong mainly to workflow modeling and execution ('Data', 'Workflow', 'Task'), to workflow reasoning ('AdaptationCase', 'SemanticMetaData', 'Ontology'), or to administrative purposes ('Group'). The resources for workflow modeling and execution are split into resources at type level ('DataPrototype', 'WorkflowPrototype', 'TaskPrototype') and resources at instance level ('DataInstance' and so on). Tasks require an execution mode ('Execution') which is 'ServiceExecution' for automated tasks and 'HumanExecution' for human tasks. 'SemanticMetaData' refers to a semantic description of a particular resource and is depending on a particular ontology. We decided to inherit 'SemanticMetaData' from 'Resource' additionally as it might require own access rights. For instance, if a standard ontology from the Dublin core initiative[6] has been used to annotate a task resource, a group of advanced users might get the right to modify the annotated semantic meta data (e.g. to extend it with own terms) without being able to change the standard ontology of course, while other users might be only allowed to read and use ('execute') the annotated data for retrieval purposes.

4 Examples

In order to get a first impression on the feasibility of the proposed user model concept, the scenario on the transport of disabled persons described in Fig. 1 and a second scenario from the office domain are considered.

An instance of the user model concept in Fig. 2 for the scenario from Fig. 1 would consist of different groups of users: The group of disabled persons themselves, one group of staff members for each cab service involved, and a group for the team of the home of the handicapped. The home of the handicapped plays the role of a clearing house so that its staff may like to monitor ongoing processes. Additionally, individual groups may be built by a handicapped person as an owner for the participants involved into a single workflow, i.e., the particular taxi driver or cab service that has been called and the responsible person from the home of the handicapped. This group would get the access rights 'read', 'modify', and 'execute', for example, while further users like other handicapped persons could get the right to read the workflow in order to learn from the owner.

A second sample scenario is the administerial work from the office domain of a small to medium size company: An instance of the user model could consist of the following groups of users: The group of secretaries, different groups of staff members for each department including the according secretaries, the group of managers who are authorized to sign. In case of a workflow for ordering a new print-out paper or other office equipment, an arbitrary staff member would be allowed to start a workflow instance ('execute'). The secretary would be the owner of the workflow instance ('owns') who is allowed to 'read', 'execute', and 'modify' the workflow

[6] http://dublincore.org/

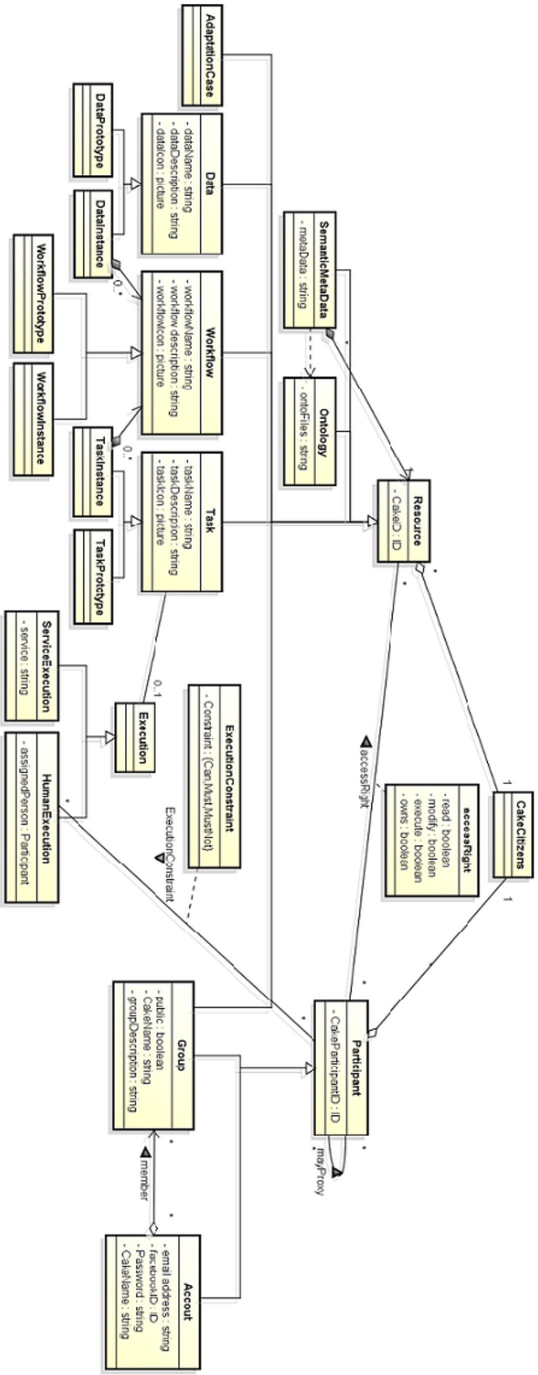

Fig. 2. User model concept for CAKE in UML class diagram notation.

except for the signing task, which can be executed and modified by the group of managers only. This means that she is able to insert or delete a sign task but not able to change its properties. In this sample, a part of a resource can have different access rights from the overall resource. Such a principle is well-known from configuring Web servers, for instance, where access rules for a sub-directory can be more restrictive than those for a super-directory.
The proposed user concept model fits well to both sample scenarios.

5 Summary and open issues

In this paper, ongoing work on a novel user model concept has been presented that allows to take adaptive workflow technology into the cloud as a platform as a service. The integration of existing components of adaptive workflow technology from the Cake system with the novel user model concept and a re-design of the workflow modeling language has been discussed briefly. First insights have been reported on applying the concept to a small scenario of workflow support for organizing the transport of disabled persons to leisure activities and to an office scenario. The next steps of our future work will be to test the user model concept with further use case scenarios and to implement the platform as a service. Especially challenging will be the extension of the future scenarios by integrating the case-based assistant systems. It is still an open issue to determine how access privileges for users can be handled. This may concern the access to a case base as well as the reuse of ontologies.

6 References

1. Baun, C., Kunze, M., Nimis, J. & Tai, S.: Cloud computing: Web-basierte dynamische IT-Services. Springer, 2009.
2. Bergmann, R. & Gil, Y. Retrieval of Semantic Workfows with Knowledge Intensive Similarity Measures. Proceedings ICCBR 2011. Springer, 2011.
3. Minor, M., Bergmann, R., Görg, S., Walter, K.: *Towards Case-Based Adaptation of Workflows*. Proceedings ICCBR 2010, pp. 421-435, Springer, 2010.
4. Minor, M. & Görg, S.: Acquiring Adaptation Cases for Scientific Workflows. In Proc. 19th International Conference on Case-Based Reasoning, 2011, accepted for publication.
5. Minor, M. Schmalen, D., & Kempin, S.: Demonstration of the Agile Workflow Management System Cake II Based on Long-Term Office Workflows. In Ana Karla Alves de Medeiros, and Barbara Weber, editors, CEUR proceedings of the BPM'09 Demonstration Track, Business Process Management Conference 2009 (BPM'09), September 2009, Ulm, Germany, volume 489, 2009.
6. Minor, M., Schmalen, D., Koldehoff, A. & Bergmann, R.: Structural adaptation of workflows supported by a suspension mechanism and by case-based reasoning. In 16th IEEE International Workshops on Enabling Technologies: Infrastructures for Collaborative Enterprises (WETICE 2007), 18-20 June 2007, Paris, France, pp. 370–375 (2007).
7. Minor, M., Tartakovski, A., Schmalen, D., Bergmann, R.: Agile Workflow Technology and Case-Based Change Reuse for Long-Term Processes. International Journal on Intelligent Information Technologies 4(1), 80-98 (2008).
8. Reichert M. & Dadam, P.: ADEPT flex—supporting dynamic changes of workflows without losing control, Journal of Intelligent Information Systems, Bd. 10, Nr. 2, S. 93–129, 1998.
9. SAP: "Mobile Service Manager: Powered by Sybase Unwired Platform", http://www.sap.com/solutions/mobility/sybase-unwired-platform/brochures/index.epx, Retrieved from the Web on June 6, 2011.
10. SAP: "Mobilizing SAP Applications for Use with BlackBerry Smartphones", http://www.sap.com/solutions/mobility/sybase-unwired-platform/brochures/index.epx, Retrieved from the Web on June 6, 2011.
11. Scheidegger, N., Briner, N. Vimalassery, D. Guggisberg, D., Sieber, P., Hahn, M. A., Meyer, R., Bertschinger, A., Habermayr, J., & Taaks, G.: Die Organisation des E-Business IX. Innovation: Fallstudien über die Bedeutung der Informatik und Telekommunikation zur Steigerung der Innovationskraft von Organisationen. Dr. Pascal Sieber & Partners AG, Bern, 2009.
12. Shang, L., Petiton, S., Emad, N. & Yang, X.: *YML-PC: A Reference Architecture Based on Workflow for Building Scientific Private Clouds*, in Cloud Computing, pp. 145–162, Springer, 2010.