Design and Deployment of Workflows in Cloud Environment

Anju Bala
Assistant Professor
Computer Sci. & Engg.Deptt.
Thapar University

Inderveer Chana
Associate Professor
Computer Sci. & Engg.Deptt.
Thapar University

ABSTRACT

Cloud computing is a new benchmark towards enterprise application development that can effectively facilitate the execution of workflows in business process management systems. Workflow applications often require very complex execution environments that are difficult to create otherwise. Integration of the workflow management systems for application deployment with any of the cloud platforms is a tedious task. Therefore, various cloud services are required to execute workflows along with workflow management software in cloud environment. In this paper, a comparative analysis of workflow design approaches has been discussed .This paper presents the challenges for implementing workflows in cloud computing and state of the art of workflow design along with a comparison of various workflow engines available. Design of workflow for a real time application has also been discussed with the help of a case study. The experimental results demonstrate how workflow for enterprise applications can be designed, validated, implemented and deployed on any major cloud infrastructures.

General Terms

Cloud Computing, Workflows, Workflow Management Systems

Keywords

Workflow, Workflow Engines, Cloud Computing, Workflow Management System

1. INTRODUCTION

With the promotion of the world's leading companies, cloud computing is attracting more and more attention for providing a flexible, on demand computing infrastructure for a number of applications [7]. Cloud computing has many unique advantages that can be utilized to facilitate workflow execution and scheduling. Workflow is concerned with the automation of procedures where documents, information or tasks are conceded between participants according to a defined set of rules. A workflow enables the structuring of applications in a directed acyclic graph form [9] as shown in Fig. 1, where each node represents the constituent task and edges represent intertask dependencies of the applications [3].A single workflow generally consists of a set of tasks each of which may communicate with another task in the workflow. Multiple workflows may further have multiple instances of workflow. A final and empirical interpretation is to consider these business processes as workflows that are supported by Workflow Management Systems [4].

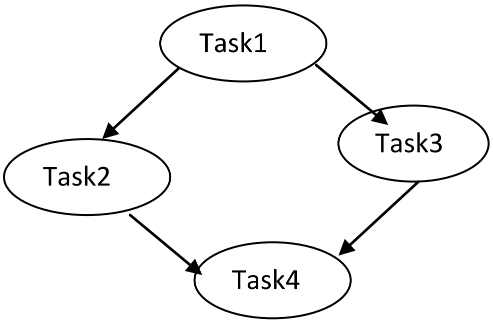


Fig. 1: Workflow Representation

The goal of workflow management is to make sure that the proper activities are executed by the right service at the right time [2]. The tasks are divided in to subtasks, roles, rules and processes to execute and observe the workflow, workflow system boost the level of production organization and work efficiency. There are various types of workflows like business workflow, abstract workflow, concrete workflow, scientific workflow and so on. These workflows are managed and coordinated by workflow management system, which provides the end user with the required data and the appropriate application program for their tasks. It allocate tasks to end-user based only on the performance of constraints like control flow, data flow, transition conditions or pre- and post-conditions.

According to the analysis of various workflow design approaches and challenges, the problem of workflow design, execution and deployment with cloud computing is formulated. It is difficult to integrate most of the workflow management systems with any of the cloud platforms to carry out the deployment of applications, so there is a need to follow an approach that can easily port applications built for one cloud platform to another. Applications should be built once and deployed several times on any cloud platform as per user's requirements. Hence, the aim of the paper is to design a workflow-based application and deploy it in cloud using a visual modeling development environment.

The rest of the paper is organized as follows. Section 2 presents the related work along with comparative analysis of various approaches. Section 3 discusses the challenges for implementing workflows in cloud. Section 3 identifies the comparison between various workflow engines. Section 4 presents the case study of workflows. Section 5 presents the implementation. Section 6 shows the experimental results and section 7 concludes the paper with a summary of our contributions.

2. RELATED WORK

Workflow Design is the core component of Workflow Management System. There are various approaches used for designing the workflows. Piotr Chrz et al.[19] proposed an Top-Down Petri Net-Based Approach that aims to Increase the soundness of resulting workflow nets and various powerful refinement rules are used. Irene Vanderfeestenm et al. [20] presented a product-based approach used to directly execute the product-based model. Josefina Guerrero García et al. [21] presented a model-driven approach to derive user interfaces of a workflow information system from a series of models. A graphical editor has also been developed. The editor is described and exemplified on a real-world case study for designing the user interfaces of a workflow information system. Dr. Mark Klein et al. [22] described a novel knowledge-based approach for helping workflow process designers and participants had better manage the exceptions. Lin Chen et al. [23] discussed a workflow design tool used for combining graphical process representation and ECA rules in controlling Grid workflow process. Integration adapter of the Grid Workflow system is also presented to facilitate the composition of all possible services. Macro Comuzzi [24] presented service based approach to deisgn monitoring process. Therani Madhusudan et al. [25] presented an Agent-based Process Coordination (APC) framework for Distributed design process management. This approach embeds autonomous agents in a workflow-based distributed systems infrastructure. It utilizes a centralized decision-making and task sharing approach to support design and activities. The comparative analysis of various workflow-designing approaches has been shown in table 1. The comparative analysis shows the various workflow designing approaches, features, language, environment and future work.

Table 1: Comparative Analysis of various Workflow Designing Approaches

Approach	**Features**	**Language used**	**Future Work**	**Environment**
Top-Down Petri Net-Based Approach [19]	It increases the soundness of resulting workflow nets	Petri Nets Based JAVA	Deadlock risks can be reduced.	Distributed
Product Based Approach [20]	It is the more flexible and dynamic support for workflow process.	Petri nets	Optimal strategies can be used	Distributed
Model-Driven Approach [21]	It is used for designing the user interface of WMS.	XML	Workflow Analysis Methods can be used.	Distributed
Knowledge Based Approach [22]	It is used to manage the exceptions	-	Needs to be implemented	Distributed
ECA Rule Based Design Approach [23]	It is used to solve the complicated business logic or scientific application problems	WSDL	Needs to apply artificial intelligence techniques	Grid Environment
Service based approach [24]	It is used to design the monitoring process.	Prom	Needs to be implemented in cloud	Distributed
Agent Based Approach [25]	It is used for flexible management of knowledge-intensive workflows	JAVA	Needs to be implemented	Distributed

There are various approaches for designing the workflows, but there is no any approach that can be deployed on cloud platform. Therefore, there is a need to use the approach in which the workflow can be designed, executed and deployed on various cloud platforms.

3. CHALLENGES FOR IMPLEMENTING WORKFLOWS IN CLOUD COMPUTING

Following challenges impede the implementation of workflows in cloud computing:

- Many workflow applications require the use of a distributed collection of computation, database and storage facilities. In the current development scenario, the workflow management requires the use of database servers. The coordination between workflow scheduling and database management system requires more accurate management of service usage and accurate estimation of service load [8]. Therefore, workflows for accurate management of services need to be implemented in cloud environments.

- Applications should be built once and deployed several times on any cloud platform as per user's requirements [10].So, workflow-based application can be designed for cloud computing.

- Although in reality some cloud services say, Google search is free for some cloud services for execution of business workflow applications. This means users may need to pay for what they use. Cost becomes a major concern for cloud workflow applications. Given that the attributes of both time and cost are involved, users should be able to decide on the fly to pay slightly more to reduce execution time or cost by allowing longer execution time as long as the final deadline can be met for workflow execution [16].Workflows can be used to meet the requirements of the user.

- Clouds allow users to acquire and release resources on-demand. This enables workflow systems to easily grow and shrink the available resource pool as the needs of the workflow changes over time [8]. Therefore, there is a need to migrate workflow executions to cloud computing platforms.

- Depending on the service, the platform support required by the cloud application service could also be a limiting factor. Apart from the above, there could be other challenges like security, regulatory compliance, data transparency [1].

4. TOOLS USED FOR DESIGNING THE APPLICATION

Various existing workflow tools used for designing the application have been compared which is based on workflow specification language, cloud platform, dependency mechanisms, event notification etc and have been presented in Table 2.

- Hamake [17] is a lightweight client-side utility that does not require installation and has very simple syntax for workflow definition.

- Oozie [16] is an open-source workflow/coordination service to manage data processing jobs for Apache Hadoop. It is an extensible, scalable and data-aware service to orchestrate dependencies between jobs running on Hadoop.

- Azkaban [18] is simple batch scheduler for constructing and running Hadoop jobs or other processes.

- Cascading [15] is a Data Processing API, Process Planner, and Process Scheduler used for defining and executing complex, scale-free, and fault tolerant data processing workflows on an Apache Hadoop.

- OrangeScape Studio [14] is ideal for designing process-oriented business applications. In the current scenario, the process is implemented in a visual modeling environment by creating workflows.

- JBOSS [13] tool is used for executing workflows with Octopus. The workflows implemented can be integrated in any cloud platforms. It supports Web AP1, JAVA**.**

- The Pegasus [12] encompasses a set of technologies that help workflow-based applications execute in a number of different environments including desktops, campus clusters, grids, and now clouds.

- YAWL [6] is an open source workflow management system. A workflow specification in YAWL is a set of process definitions,. YAWL is also used for Analysis of workflows and validation of workflows. It can be used for implementing multiple workflows.

Table 2: Comparison of Various Workflow Engines

Workflow Engines	Workflow specification language	Cloud Platform	Dependency mechanism	Event notification	Requires installation	Web service
Hamake	XML	Hadoop	Data-driven	No	No	Console/log messages
Oozie	XML	Hadoop	Explicit	No	Yes	Web Pages
Azkaban	Text file with key/value pairs	Hadoop	Explicit	No	Yes	Web pages
Cascading	Java API	Hadooop	Explicit	Yes	No	Java API
Orangescape Studio	MYSQL,Oracle,DB2	Google app Engine,Microsoft Azure,IBM	Data driven	Yes	No	Web pages
JBOSS	Java	Octopus	Explicit	Yes	No	Java API
Pegasus	Java	Eucalyptus, Amazon EC2,Open Nebula and so on	Explicit	Yes	Yes	Java API
YAWL	XML	None	Data driven	Yes	Yes	Web Pages

5. APPLICATION WORKFLOW DESIGN AND IMPLEMENTATION: A CASE STUDY

For designing the workflows, we present a case study on online banking. It is a service where people can perform financial transactions such as account-to-account transfers, deposit of cash and withdrawal of cash and so on.

5.1. Requirements of banking application

In study of online banking, the security is provided by giving a Personal Identification Number (PIN) to every customer to access his/her account. The system checks whether the PIN entered by the user is valid or not. It also takes care of the remaining balance in the customer’s account. If the amount entered by the user to withdraw the cash exceeds the current balance, the system gives a warning message else, it allows the user to withdraw the cash. Fig.2. shows a use case diagram of the functionality for an online banking application in terms of actors, their goals and dependencies between the use cases.

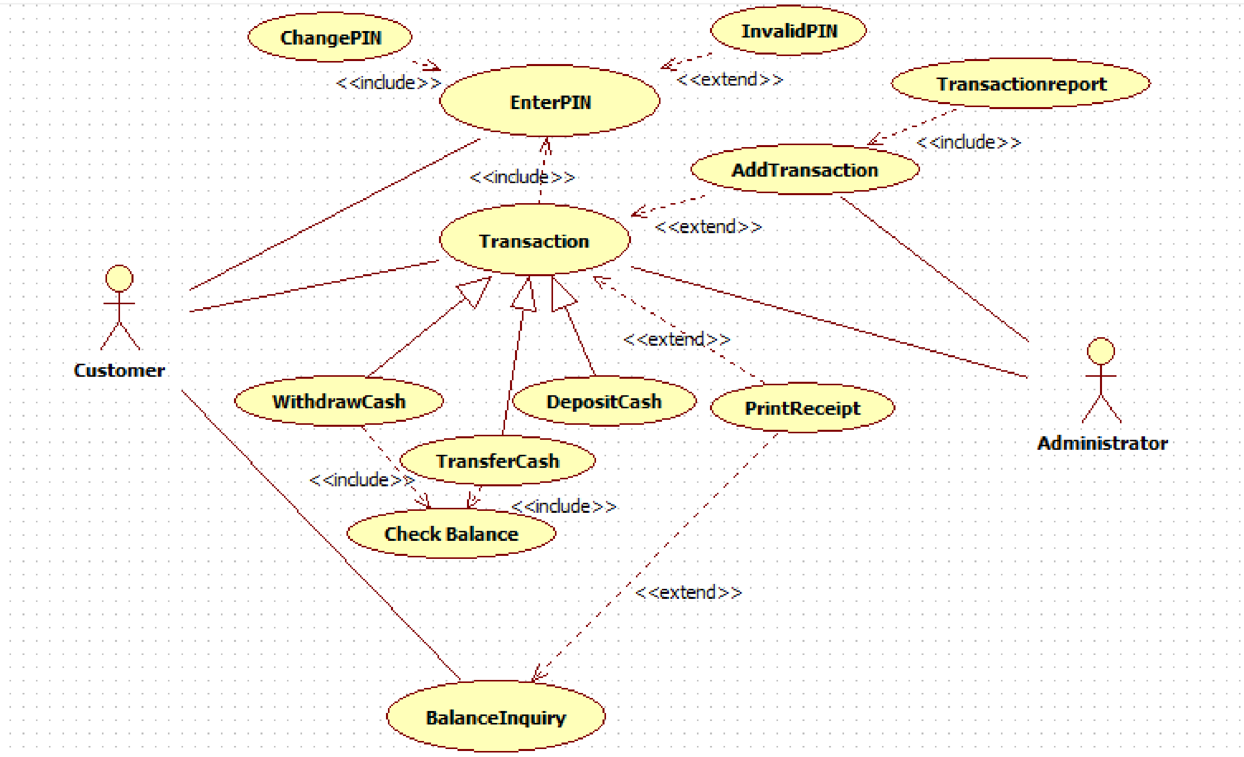


Fig 2: Use case diagram for Online banking application

5.2. Designing the Workflows

Above said application has been designed in YAWL. It is used to analyze the workflow specification. At any stage, the specification can be validated and or saved to a YAWL engine file (.yawl). The engine handles the execution based on the state of a case and its specification determining which events it should offer to the environment. Fig.3 shows the specification for the example application with no validation problems. Validation is used to test specifications against YAWL schema.

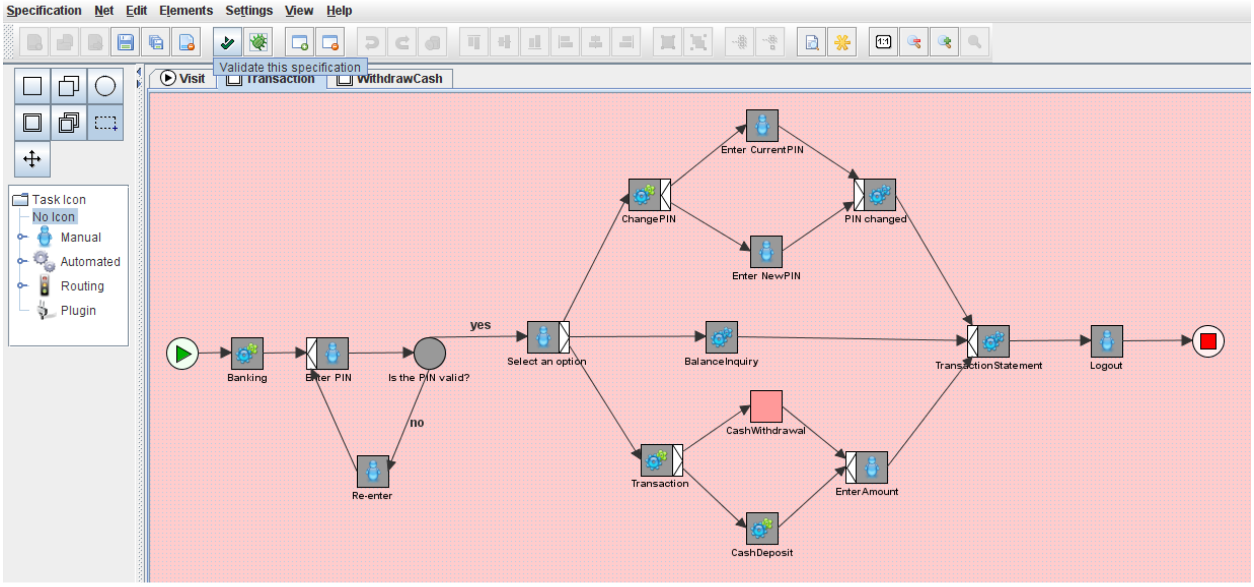


Fig 3: Designing the workflow specification against YAWL syntax

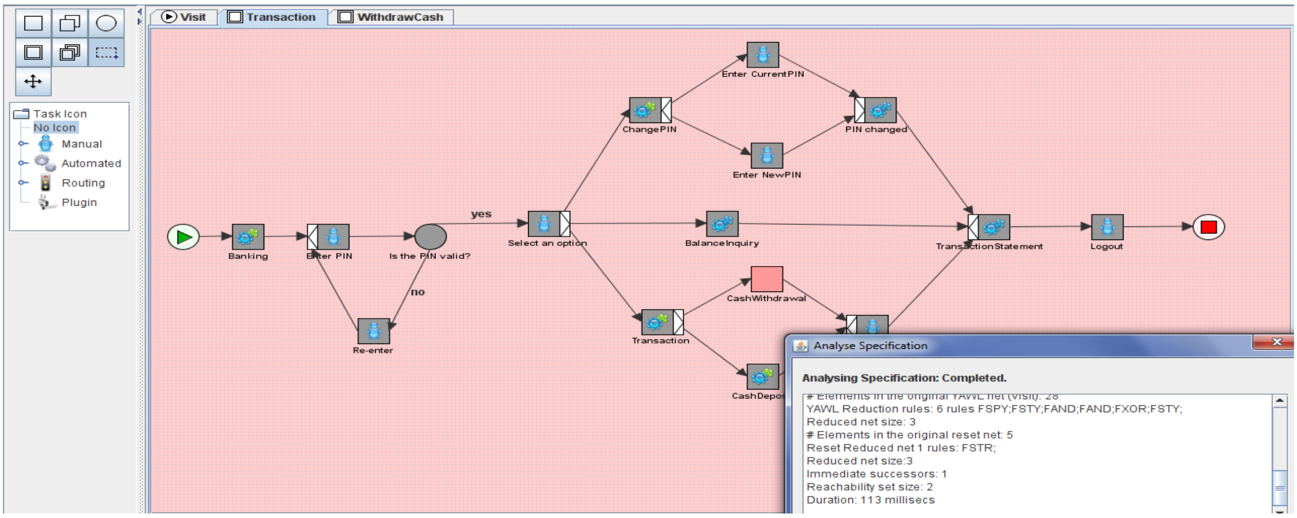
5.3. Implementation

The application with its four-tiered architecture is built on REST-style [10] service interface. OrangeScape Studio is used to implement this application. The packaged application is deployed on Google AppEngine.

6. EXPERIMENTAL RESULTS

Following results have been gathered after execution:

Case 1: The analysis of workflow allows the workflow designers to analyze the specifications. A number of potential problems with a workflow can be automatically spotted with analysis. The workflow specification for banking application is analyzed as shown in Fig.4.



Analysing Specification: Completed.

```
# Elements in the original YAWL net (visit): 28
YAWL Reduction rules: 6 rules FSPY;FSTY;FAND;FAND;FXOR;FSTY;
Reduced net size: 3
# Elements in the original reset net: 5
Reset Reduced net 1 rules: FSTR;
Reduced net size:3
Immediate successors: 1
Reachability set size: 2
Duration: 113 millisecs
```

Fig 4: Analyzing the workflow specification in YAWL

Case 2: A workflow applied to a banking application brings the details of that process into focus. Workflows are implemented using a visual modeling environment in OrangeScape Studio as shown in Fig. 5.

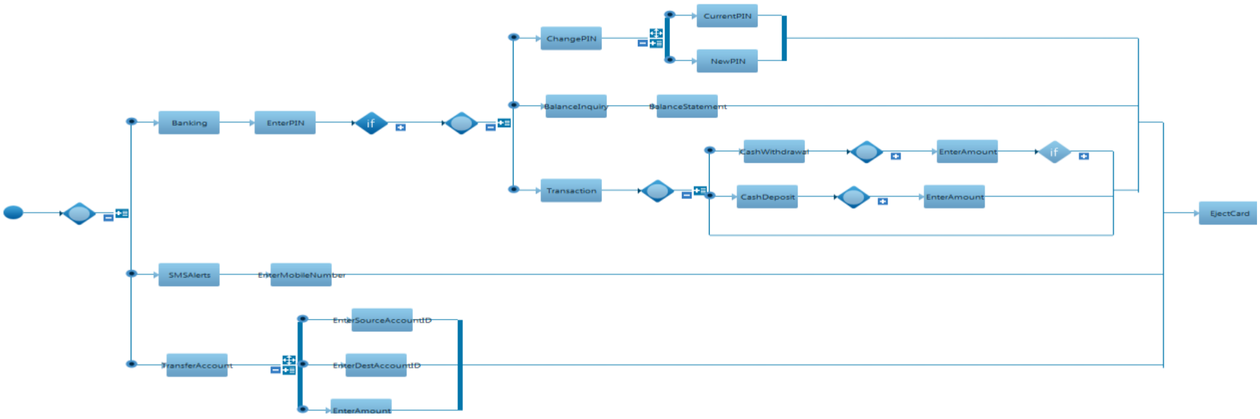
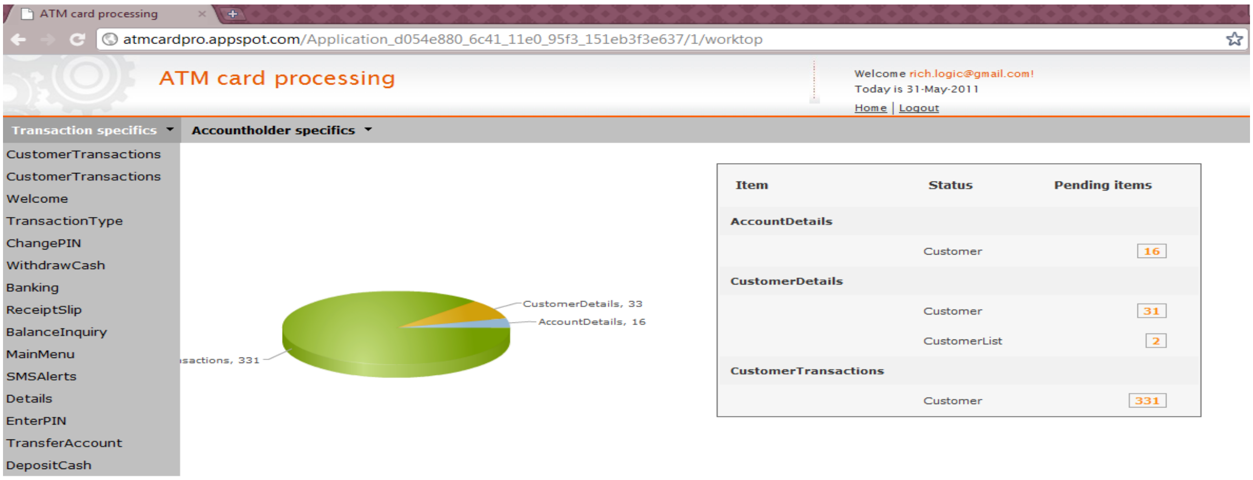


Fig 5: Implementation of the workflow specification

Case 3: The packaged application can be deployed on Google AppEngine by using a single-click deploy option in-built in the Studio environment as shown in Fig. 6.



atmcardpro.appspot.com/Application_d054e880_6c41_11e0_95f3_151eb3f3e637/1/worktop

ATM card processing

Welcome rich.logic@gmail.com!
Today is 31-May-2011
Home | Logout

Item	Status	Pending items
AccountDetails		
	Customer	16
CustomerDetails		
	Customer	31
	CustomerList	2
CustomerTransactions		
	Customer	331

Fig 6: Example application deployed on Google AppEngine

From the implementation, we conclude that these REST style web services can also be combined with various cloud-computing platforms. It can help many enterprises to design and deploy process-oriented applications based on workflow.

7. CONCLUSION

In this paper, we have surveyed the different workflow design approaches and tabulated their various parameters along with the future work. Existing workflow design approaches have not been currently deployed on cloud platforms. This paper discussed the challenges for implementing workflows in cloud computing and compared various workflow engines, which are currently available. In this paper, a workflow based real time application has been designed in YAWL and implemented in OrangeScape Studio that is deployed on Google App engine. The current workflow management system, YAWL can be extended with support for more workflow patterns. Some future extensions can also be possible by considering multiple workflows for enterprise applications. Applications supporting multiple instances of workflow can also be designed and deployed on some of the cloud infrastructures. Workflow management system needs to be used for diagnosis and immediate handling of disruptions such as resource failure occurring during the workflows execution.

8. REFERENCES

[1] A.V.Parameswaran, A.Chaddha, 2009 ,Cloud Interoperability and Standardization, SETlabs briefings.

[2]D.Cybok, 2008,Workflow Management for Grid Computing: A Grid Workflow Infrastructure,msg systems,Munich, Germany.

[3] E.Deelman, 2003, Mapping abstract complex workflows onto Grid environments, Journal of Grid Computing, 25–39.

[4] H.A.Reijers, 2002, Design and Control of Workflow Processes, PhD thesis, Eindhoven University of Technology, Netherlands

[5] H.Tan,W.M.P. van der Aalst,2006,Implementation of a YAWL Work-list handler based on the Resource Patterns, Proceedings of the 10th International Conference on Computer Supported Cooperative Work in Design, 1184-1189.

[6] K. Liu, Y. Yang, J. Chen, X. Liu, D. Yuan and H. Jin, 2010,A Compromised-Time- Cost Scheduling Algorithm in SwinDeW-C for Instance-intensive Cost-Constrained Workflows on Cloud Computing Platform, International Journal of High Performance Computing Applications, 445-456.

[7] M.Armbrust, A.Fox, et al., 2009, Above the Clouds: A Berkeley View of Cloud Computing, Technical Report No. UCB/EECS-2009-28, University of California, USA.

[8] R.Buyya, S.Pandey and D.Karunamoorthy, 2011, Workflow Engine for Cloud, Cloud Computing: Principles and Paradigms, New Jersey: John Wiley and Sons.

[9] R. Sakellariou and H. Zhao, 2004,A Hybrid Heuristic for DAG Scheduling on Heterogenerous Systems,The 13th Heterogeneous Computing Workshop (HCW 2004), Santa Fe, New Mexico, USA.

[10] W.Li and P.Svard, 2010, REST-based SOA Application in the Cloud: A Text Correction Service Case Study, IEEE 6th World Congress on Services.

[11] W. M. P. Van der Aalst, L. Aldred, M. Dumas, et al., 2004,Design the 16th International Conference on Advanced Information Systems Engineering(CAiSE 04), Riga, Latvia.

[12] Ian Taylor, Pegasus workflow management system, Available at http://pegasus.isi.edu/cloud/[Last accessed july2012]

[13]WorkflowEngine,Availableat, nftp://ftp.cordis.europa.eu/pub/transport/docs/summaries/ integrated_octopus_report.pdf/[Last accessed June 2012]

[14] Trial version of orangescape, Available online at https://www.orangescape.com/[Last accessed June 2012]

[15] Cascading workflow, Available online at http://www.cascading.org/[Last accessed June 2012]

[16] Yahoos workflow engines for hadoop, Available online at http://yahoo.github.com/oozie/[last accessed June 2012]

[17] Hamake workflow engine for hadoop, Available at http://code.google.com/p/hamake/wiki/HamakeManual[Last Accessed June 2012].

[18] Azkaban Workflow Engine, Available at http:// sna-projects.com/azkaban/ Azkaban/Last Accessed June 2012].

[19] Piotr Chrz¸astowski-Wachtel, Boualem Benatallah, Rachid Hamadi, Milton O'Dell and Adi Susanto, 2003, A Top-Down Petri Net-Based Approach for Dynamic Workflow Modeling, BPM 2003, LNCS 2678, 336–353.

[20] Josefina Guerrero García, Christophe Lemaigre, Juan Manuel González Calleros, 2008, Model-Driven Approach to Design User Interfaces for Workflow Information systems, Journal of Universal Computer Science, 3160-3173.

[21] Dr. Mark Klein, Prof. Chrysanthos Dellarocas, 2000, A Knowledge-Based Approach to Handling Exceptions in Workflow Systems, Journal of Computer Supported Collaborative Work, Special Issue on Adaptive Workflow Systems.

[22] Lin Chen Minglu Li Jian Cao Yi Wang, 2005,An ECA Rule-based Workflow Design Tool for Shanghai Grid, Services Computing (SCC'05).

[23] Macro Comuzzi and Irene T.P. Vanderfeesten, 2011, Product based workflow design for monitoring of Collaborating Buisness processes in CAiSE 2011, Springer, 154-168.

[24] Irene Vanderfeesten, Hajo A. Reijers, Wil M.P. van der Aalst, 2008, Product Based Workflow Support: A Recommendation Service for Dynamic Workflow Execution

[25] Therani Madhusudan,2005, An agent-based approach for coordinating product design workflows, International journal of science direct Computers in Industry 235–259.