A Workflow Execution Platform for Collaborative Artifact-Centric Business Processes

Sira Yongchareon[1], Kan Ngamakeur[2], Chengfei Liu[2], Sivadon Chaisiri[3], Jian Yu[4]

[1]Department of Computing and Information Technology
Unitec Institute of Technology, New Zealand
sira@maxsira.com

[2]Faculty of Science, Engineering and Technology
Swinburne University of Technology, Australia
{kngamakeur, cliu}@swin.edu.au

[3]School of Information Technology, Shinawatra University, Thailand
sivadon@ieee.org

[4]Auckland University of Technology
School of Computing and Mathematical Sciences, New Zealand
jian.yu@aut.ac.nz

Abstract. To execute an artifact-centric process model, current workflow execution approaches require it to be converted to some existing executable language (e.g., BPEL) in order to run on a workflow system. We argue that the transformation can incur losses of information and degrade traceability. In this paper, we proposed and developed a workflow execution platform that directly executes a collaborative (i.e., inter-organizational) workflow specification of artifact-centric business processes without performing model conversion.

1. Introduction

An artifact-centric process modeling approach has emerged to provide an alternative approach for specifying a business process. The approach focuses on describing how business-relevant key data entities, known as "*artifacts*", evolve in a business process [7]. IBM [5, 7] has developed an operational modeling framework which consists of three components: *artifacts*, *services*, and *associations* (between artifacts and services) and proposed a Guard-Stage-Milestone (GSM) approach to modeling artifact-centric processes [12]. Current workflow execution approaches require an artifact-centric model to be transformed to an executable activity-centric process language (e.g., BPEL) in order to run on existing workflow systems (e.g., in [9, 15, 17]). We argue that the model conversion incurs losses of information and affects traceability and monitoring ability of workflow [16], especially in a collaborative environment where the workflow span across multiple inter-business entities. We found several technical challenges such as executable model specification, workflow coordination, and data access/management that need to be addressed when developing a system to support execution of artifact-centric models in a distributed environment. To address the challenges, we developed an artifact-centric workflow execution platform for collaborative artifact-centric processes based on using view-based artifact-centric approach [2, 4] on service-oriented and event-driven architectures.

2. Artifact-centric Collaboration Execution Framework

An artifact-centric process model can be constructed using *artifacts*, *their life cycles* and *interactions* [4]. To achieve goals of a collaborative process, all organizations in the collaboration must develop and agree on a mutual contract for them to progressively operate towards the goals [3]. Here, we model artifact-centric processes by using the *Artifact-Centric Collaboration Model* [2] and the view-based approach presented in [1]. Two types of artifacts are used to model collaboration or the contract: (1) *local artifact* and (2) *shared artifact*. *Local artifacts* are owned and accessed by one organization to support the coordination between its local business processes and the inter-organizational processes. *Shared artifacts* are defined as a contract between participating organizations where it contains business stages to capture progress of the process toward the completion of the collaborative process. We illustrate architecture of our Artifact-Centric Collaboration (ACC) Execution Framework and its platform in Fig. 1 and Fig. 2, respectively.

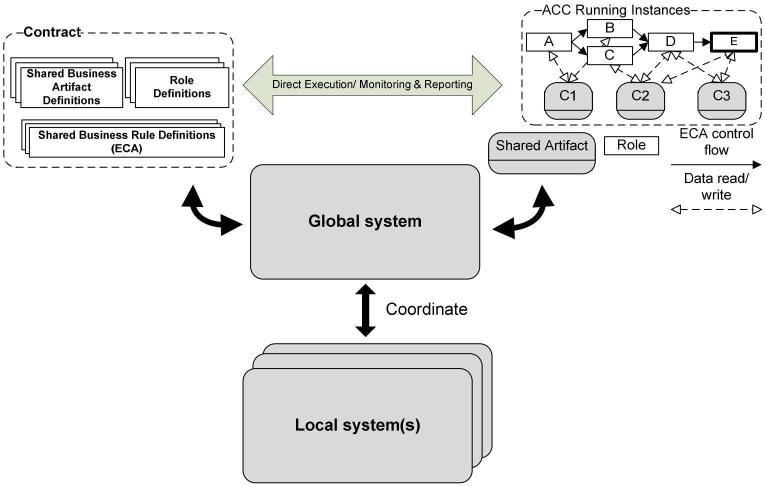

Fig. 1 Artifact-Centric Collaboration (ACC) Execution framework

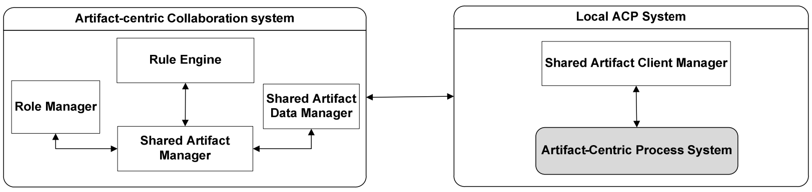

Fig. 2 Architecture of workflow execution platform for ACC

The platform utilizes event-driven and service-oriented architectures to design and implement the centralized controller to support distributed execution across organizations based on using a set of shared artifacts and shared business rules to govern interaction between organizations [2, 4, 13]. The platform comprises of an *Artifact-centric collaboration* (*ACC*) system and *local ACP* system(s). The local ACP system is designed to run in each participating organization. The ACP system is extended from the system presented in [8, 16] with a *Shared artifact client manager* to support execution of shared artifacts in a collaboration. The ACC system acts as a central controller and it consists of four components: *shared artifact manager*, *rule engine*, *role manager*, and *shared artifact data manager*. *Shared Artifact Manager* provides management functionality to ensure each contract running in the execution is created, managed and updated correctly. *Rule Engine* is to deliver functionality of rule evaluation. The rule engine serves as a central controller to coordinate internal and external operations of each component in the platform. *Role Manager* is to handle a task that is allocated to each role involved in a particular business process. *Shared Artifact Data Manager* performs a task of updating these shared artifacts. In a local ACP system, *Shared Artifact Client Manager* is designed to address communication between the central controller and local systems. Its main functionality is to receive and pass messages issued by the controller to a local system and also detect status of process execution of the local system and notify the controller regarding completion of a task or a session of the local system in a synchronized manner. A *coordination contract* is implemented in the ACC system for correct and consistent coordination between the global system and all local systems. Our platform is available at [19].

3. Related Work Discussion and Conclusion

This paper proposed a platform for executing collaborative artifact-centric business processes. Cohn and Hull [7] illustrated that IBM has used BELA tool to map an artifact-centric process model into an executable model (e.g., BPEL) that can run on IBM's WebSphere Process Server. Cohn et al. [10] proposed a system called Sienna to support execution of Finite-State-Machine lifecycles for artifacts. Barcelona [18] supports the execution of artifact models with Guard-Stage-Milestone (GSM). Sienna and Barcelona require ACSI Interoperation Hub [6] to support interoperation between enterprises. However, both of their systems are not publicly available at this stage, therefore we are not able test and evaluate them. G. Liu et al [9] proposed an artifact-centric workflow model, namely ArtiFlow, with a technique to translate ArtiFlow to a BPEL specification to run on a BPEL engine. Their prototype was developed and presented in [15, 17]. To support a dynamic runtime modification, Xu et al. [11] developed a hybrid model called EZ-Flow based on the ArtiFlow to gain advantages of both declarative and procedural natures. Compared with our approach, we execute an artifact-centric model without converting the model to an activity-centric model. Moreover, our system can record all running artifacts as specified in the artifact-centric model, therefore tracking can be achieved not only at the process level but also at the artifact level. Marinoiu et al. [14] developed AXART system to manage the updates of Active XML (AXML) models with embedded function/Web service calls. Compared with our work, artifacts, rules, and services are defined as separate components. Thus, those components are less coupled and can be more effectively managed. Russo et al. [20] proposed Data Centric Dynamic Systems to execute data-centric processes. The system uses a business rule engine to control an action that updates the state of data. However, their concept of the life cycle is not incorporated and there is no discussion on execution of inter-organizational processes.

In the future, we will improve our system prototype in several areas, e.g., run-time verification, exception handling, and change management.

References

1. Yongchareon, S., Liu, C.: Process View Framework for Artifact-Centric Business Processes. In: OTM 2010, LNCS 6426, pp. 26–43.
2. Yongchareon, S., Liu, C., Zhao, X.: An Artifact-centric View-based Approach to Modeling Inter-organizational Business Processes. In: WISE 2011, LNCS 6997, pp. 273-281.
3. Van Der Aalst, W.M.P, Lohmann, N, Masuthe, P, Stahl C, and Wolf, K.: Multipart Contracts: Agreeing and Implementing Interorganizational Processes, The Computer Journal, Vol. 53, No. 1, pp.90-106.
4. Yongchareon, S., Liu, C., Yu, J., Zhao, X.: A View Framework for Modeling and Change Validation of Artifact-Centric Inter-Organizational Business Processes, *to appear in Information systems* (2014), DOI: 10.1016/j.is.2014.07.004
5. Bhattacharya, K., Hull, R., Su, J.: A data-centric design methodology for business processes, Handbook of Research on Business Process Modeling (2009)
6. Hull, R., Narendra, N.C., Nigam, A.: Facilitating Workflow Inter-operation Using Artifact-Centric Hubs. In: ICSOC/ServiceWave 2009. LNCS 5900, pp. 1–18.
7. Cohn, D., Hull, R.: Business artifacts: A data-centric approach to modeling business operations and processes. IEEE Data Engineering Bulletin 32(3), 3–9 (2009)
8. Yongchareon, S., Liu, C., Zhao, X., Xu, J.: An Artifact-Centric Approach to Generating Web-Based Business Process Driven User Interfaces. In: WISE 2010, LNCS 6488, pp. 419–427.
9. Liu, G., Liu, X, Qin, H, Su, J., Yan, Z., Zhang, L.: Automated Realization of Business Workflow Specification. In: ICSOC/ServiceWave 2009. LNCS 6275, pp. 96–108.
10. Cohn, D., Dhoolia, P., Heath III, F., Pinel, F., Vergo, J.: Sienna: From Powerpoint to Web App in 5 minutes. In: ICSOC 2008. LNCS 5364, pp. 722-723.
11. Xu, W., Su, J., Yan, Z., Yang, J., & Zhang, L.: An artifact-centric approach to dynamic modification of workflow execution. In: Meersman, R., Dillon, T., Herrero, P. (Eds.): OTM 2011, Part I, LNCS 7044, pp. 256–273. Springer, Heidelberg (2011).
12. Hull, R., et al.: Introducing the guard-stage-milestone approach for specifying business entity lifecycles. In: Bravetti, M. (ed.) WS-FM 2010. LNCS, vol. 6551, pp. 1–24.
13. Yongchareon, S., Liu, C., Zhao, X.: A Framework for Behavior-Consistent Specialization of Artifact-Centric Business Processes. In: Barros, A., Gal, A., Kindler, E. (eds.) BPM 2012. LNCS, vol. 7481, pp. 285–301. Springer, Heidelberg (2012)
14. Marinoiu, B., Abiteboul, S., Bourhis, P., Galland, A.: AXART – Enabling collaborative work with AXML artifacts. In: Proc. VLDB Endowment (2010)
15. Zhao, D., Liu, G., Wang, Y., Gao, F., Li, H., Zhang, D.: A-Stein: A prototype for artifact-centric business process management systems, BMEI 2011, vol. 1, pp. 247-250.
16. Ngamakeur, K., Yongchareon, S., Liu, C.: A Framework for Realizing Artifact-Centric Business Processes in Service-Oriented Architecture. In S.-g. Lee et al. (Eds.): DASFAA 2012, Part I, LNCS 7238, pp. 63–78, Springer, Heidelberg (2012)
17. Zhao, D., Liu, G., Jiang, Y., Gao, F., Wang, Y.: The execution and detection of artifact-centric business process. In: IEEE International Conference on Computer Science and Automation Engineering, vol.4, pp.491-495. IEEE Press, Shanghai (2011)
18. Heath, III, F.(T.), Boaz, D., Gupta, M., Vaculin, R., Sun, Y., Hull , R., Limonad, L.: Barcelona: A Design and Runtime Environment for Declarative Artifact-Centric BPM. In S. Basu et al. (Eds.): ICSOC 2013, LNCS 8274, pp. 705–709. Springer, Heidelberg (2013)
19. Ngamakeur, K., Yongchareon, S.: ACC & ACP systems prototype (2014), https://sites.google.com/site/maxsirayongchareon/artifact-s
20. Russo, A., Diag, S.,Mecella, M., Patrizi, F., Montali, M.: Implementing and Running Data-Centric Dynamic Systems. In: the 6th IEEE International Conference on Service-Oriented Computing and Applications, 2013, pp. 225-232.