

# BPaaS: A Platform for Artifact-centric Business Process Customization in Cloud Computing

*Min Gao, Yuyu Yin, Xingfei Wang*

School of Computer, Hangzhou Dianzi University  
Key Laboratory of Complex Systems Modeling and  
Simulation of Ministry of Education  
College of Electrical Engineering, Zhejiang University  
Hangzhou, Zhejiang, P. R. China  
Email: {152050068, yinyuyu, 151050148}@hdu.edu.cn

*Lipeng Guo*

College of Computer Science  
Fudan University  
Shanghai, P. R. China  
Email: lpguo@fudan.edu.cn

*Lipeng Guo*

College of Computer Science  
Fudan University  
Shanghai, P. R. China  
Email: lpguo@fudan.edu.cn

*Ying Li*

College of Computer Science  
Zhejiang University  
Hangzhou, Zhejiang, P. R. China  
Email: cnliying@zju.edu.cn

*Zaidie Chen*

Hangzhou Housing Management Bureaus  
Hangzhou, Zhejiang, P. R. China  
Email: czd@tmsf.com

*Ying Li*

College of Computer Science  
Zhejiang University  
Hangzhou, Zhejiang, P. R. China  
Email: cnliying@zju.edu.cn

**Abstract**—As a new service paradigm of Software-as-a-Service (SaaS), Business Process-as-a-Service (BPaaS). BPaaS is used to build a cost-effective Business Process Management (BPM) system. Based on universal Artifacts, we develop a framework named SeGA (Self-Guided Artifact). In SeGA, a BPM system is capable of executing business processes from multiple clients, and responding query at runtime. what's more, by the template-based cascading data mapping method, entity to be synchronized with database automatically, so each BP instance can modify its process entity without worrying about the database access. In this paper, we conduct a deep research and implement a prototype system for Artifact process design.

**Keywords:** SaaS, BPaaS, Universal Artifact, SeGA

## I. MOTIVATION

A business process (BP for short) is a sequence of tasks that are performed by humans or computers to accomplish a business objective. In traditional settings, a BPM system always suffers from low adaptability, which means the BPM system is highly likely to be redesigned along with the change of environment or requirement [1].

In traditional business process modeling, activity is the “first-citizen” and the main focus is the control flow, such as resource planning and logistics, mainly aiming at business

management in general instead of software design or development. As a consequence, once any change occurs, the system has to pay the price.

Many miserable stories happen, and we will give an example of the BPM system that is used in the Hangzhou Housing Management Bureaus (HHMB for short). Note that HHMB has already gained rich experience in using BPM systems. Now, for the same bureau of Urumqi (the capital city of Xinjiang province), the administrative processes are similar. However, it differs from the case of Hangzhou in organization architecture and process complexity. Thus, it is a challenge to take advantage of the existing BPM systems. Although their tenants differ in terms of specific requirements, they do share the common requirement of administrative processes.

From the example upon, it is obvious to see the missing data is a key reason for hindering software design and management. The real world is not kind, for the absence of data, any change happened may disturb the executable activity sequence. This model is not scalable. Faced up with the change whether from data or organization structure, it is technically difficult to implement the change on the prototype system in practice due to the lack of coherent data design [3]. Honestly, there are more than one example like this. Processes in real world tend to be semi/loosely structured or unstructured, knowledge-intensive and usually driven by user decision and

data. These processes cannot be represented as a set of activities with predefined precedence relations. Availability and status of data objects are the main driver for the process progression. So to model and implement data-driven processes, a tight integration of processes and data is required. To overcome this limitation, we introduce the Artifact-based business process[4]. The artifact-centric approach[5] emerged as a foundational proposal for merging data and processes together. In Artifact-centric business process, the change rules could be temporary and non-schematic. Rules allow business processes to respond to situations flexibly with many more options.

Recently Software-as-a-Service (SaaS) has begun to be employed in business process manage. Cloud computing has undoubtedly fired the desire to provide BP execution as service or BPaaS[2]. By providing a set of pre-designed workflows on the SaaS application, a tenant can simply select and configure the most suitable one and use the business processes execution on-demand as a Service and thus brings a new service paradigm- Business Process-as-a-Service (BPaaS). BPaaS proposes a novel way to integrate the techniques of SaaS and BPM and enables multiple tenants (customers) to share common business process model and resources among each other. This has the disadvantage that multi-tenant models are often limited in terms of customizability: one model should fit the needs of all customers[6][7][8].

BPaaS can be regarded as a special SaaS that is used in business process management. SaaS is one of different ways to provide cloud services, through providing configurable interfaces in the Internet or Intranet. Similarly, BPaaS provides BP services for multiple users in a way of service. That is, it is not necessary for users to manage or control the underlying resources. Since the BPaaS provides resources through interfaces in the Internet or Intranet, the users can conveniently access the resources in the system via a browser, and do not need to invoke any low-level fine-grained component [9]. Note that, it is also an option to use the multi-tenancy technique to build a BPM system.

Consider real estate property management in China again. There are roughly 10 to 50 Housing Management Bureaus (HMBs) in each of 30 provinces for managing titles, permits, licenses etc. Every HMB currently runs and maintains its own BPM system[10]. For example, the BPM system for the HMB in a large city Hangzhou handles about 300,000 cases annually (with about 500 BP models)(Fig.1 and Fig.2). BPaaS can get an estimate 9% labor saving for Hangzhou HMB in managing and maintaining BPM systems[11] and is a great business opportunity in the software market in general.

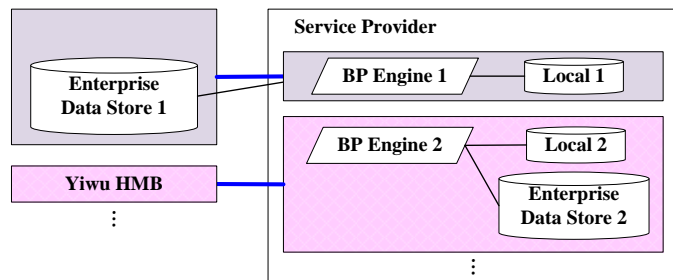


Fig. 1. Running Clients' BP Engines

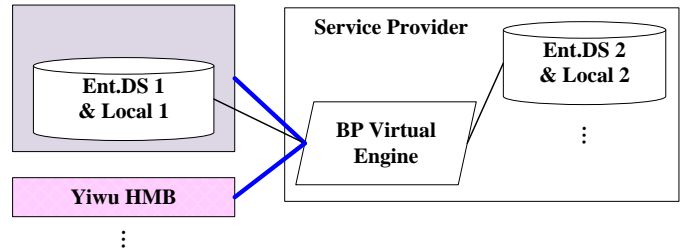


Fig. 2. Shared BP Virtual Engine

However, in most BPM systems (workflow systems), the data for process execution is scattered across databases for enterprise, auxiliary local data stores within the BPM systems, and even file systems (e.g., specification of process models). The interleaving nature of data management and BP execution and the lack of a coherent conceptual data model for all data needed for execution make it hard for (1) providing BPaaS (2) effectively support collaboration between business processes, due to an enormous effort required on maintaining both the engines as well as the data for the client applications. In particular, different modeling languages and different BPM systems make process interoperation one of the toughest challenges. In [12], Professor Su formulated a concept of a “universal artifact”, which extends artifact-centric models by capturing all needed data for a process instance throughout its execution. A universal artifact clearly captures all necessary data for execution and allows for a BP engine to process without any data outside of the universal artifact. In this paper, we conduct a deep research into the BPaaS model[13] and implement a prototype system for artifact process design.

## II. ARTIFACT

Artifact is a key, business-relevant conceptual dynamic entity that is used in guiding the operations of a business, and whose content evolves as it moves through those operations. It aims to provide a unified, end-to-end view of relevant entities and their possible evolutions.

### A. Maintaining the Integrity of the Specifications

An artifact type consists of two parts named information model and lifecycle model, aiming to provide a unified, end-to-end view of relevant entities and their possible evolutions.

- The information model is defined as a set of attributes, and includes all the data needed to capture business process goals and evaluate whether they are achieved, for the runtime state of a business process is determined by the current snapshot of all artifacts.
- Lifecycle model is the description of the allowed information model evolutions through the execution of a process. it specifies possible ways that an artifact might progress through the business but does not restrict the way to specify artifact lifecycles.

### III. ARCHITECTURE

The architecture of the target BPM system is shown in Fig. 1. The Designing layer is responsible for modeling, including Process Customizer, Entity Customizer and Data Connector. We will elaborate the three components below(Fig. 3).

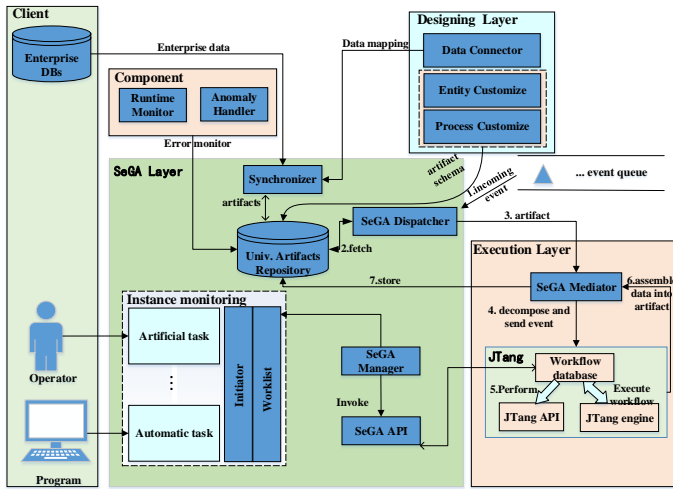


Fig. 3. The Architecture of the Target BPM System

Finally, complete content and organizational editing before formatting. Please take note of the following items when proofreading spelling and grammar:

#### A. The Entity Customizer

The Entity Customizer provides a series of tools to design schema and further saved in the repository in the SeGA layer. The Entity Customizer is responsible for data design. This module first imports the business process entity template of the corresponding process template, and the customer can modify the entity template and conduct attribute configurations during the designing process according to the specific business requirements. The process entity needs to consider three types of data: Enactment status which represents the current execution snapshot, resource usage and state needed for service execution and correlation between processes instances. The Artifact entity model in the BPaaS platform is a tree-shaped model[14], Each node represents an Artifact data. The following entity types are defined in this part:

**Artifact:** All the data nodes in this platform are of Artifact type. And entity of this type specifically refers to the one who has sub-nodes and has a one-to-one mapping with its instance. For example in Fig. 5, the Applicant Information of Public rental application form, It has sub-nodes such as name, age, but only one “applicant information” is allowed.

**Artifact: 1-n:** Artifact with a relationship of one-to-many. For example in Fig. 5, different from the “applicant information”, After instantiation, the “Family member information” can matching with several members for the Multi-member family .Entity of this one-to-many relationship is set to be Artifact 1-n type.

**Attribute:** Nodes of the Entity tree which represents inseparable metadata, such as “name”. Specially, the Attribute

attribute used to represent the uniqueness of the Artifact instance is called “Key”.

**Group:** Complex attribute type which stores as a metadata in the database, but in reality, it may be divided into multiple ones. For example in Fig. 4, in practical case, the “address” may be divided to “province” and “city” for use, but the separately stored attribute of province and city has no meaning in store, for this mean, the type “Group” is essential.

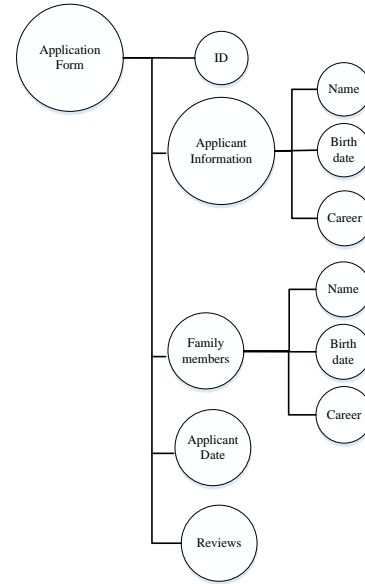


Fig. 4 Artifact Instance-Application Form

#### B. The Process Customizer

The Process Customizer adjusts the artifact lifecycle that is stored in the universal artifacts repository (UAR for short, in the SeGA layer) (Fig.6), and also makes service binding. Customers can customize the process by simply dragging the icon (such as the start node, task node, etc.) in the page according to his own need. And in service binding, by setting the read and write permissions of different fields in the business entity, the binding of the process task and the business entity is realized. The Process Customizer is in charge of setting basic activity attributes(introduction, merge mode, branch mode), as well as the attributes(conditional expressions) among different tasks. Each activity needs to specify which attributes to read and write.

#### C. The Data Connector

The Data Connector is a tool to define the mapping relationships in the entity-enterprise database (EDB for short), which are maintained by the Synchronizer in SeGA layer [14]. It includes two parts called DB-Template and EDB mapping and Entity-EDB mapping. Through the pattern mapping technology to establish relationship between the association automatically and add the mapping rules manually. Due to the different focus of business people and programmers, there are differences between the understanding of business entities and data of enterprise database, together with the heterogeneity of the data structure, it is difficult to establish a direct mapping

between the business entity and enterprise data, and thus can not guarantee the data consistency.

Below is the basic theory of cascade mapping(Fig.5). O represents cascading operations and M represents data mapping. The source mode includes business entity template  $S'$  and business entity  $S$ , target source mode includes data-base template  $T'$  and enterprise database  $T$ . In order to solve the problem of large differences in structure and attribute names of  $S$  and  $T$ , using the business entity template  $S'$  and the database template  $T'$  to achieve automatic data mapping of  $S$  and  $S'$ ,  $T$  and  $T'$ . between the two. In the  $S'$  and  $T'$  mapping, the use of pattern mapping technology can automatically complete the mapping of partial data, and meanwhile, manually modified mapping rules is allowed. In this way, we can establish the data mapping relationship from  $S$  to  $T$  by establishing the mapping relation between  $S$  and  $S'$ ,  $S'$  and  $T'$ ,  $T'$  and  $T$  respectively.

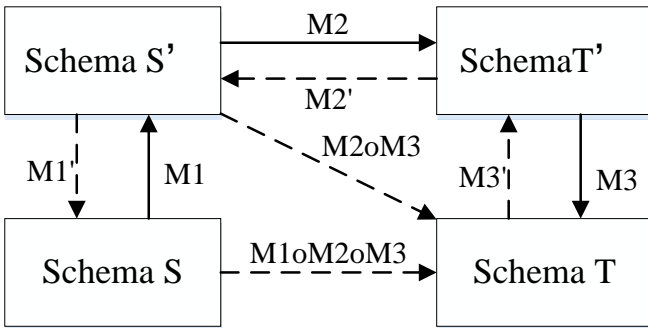


Fig. 5 cascade mapping

Based on the template-based cascading data mapping method[14], the data connector solves the differences in the understanding caused by the different points of interest by bridging the business template and the database template, and the part of the mapping work is automatically completed improving the efficiency of the mapping greatly. That is, Data mappings allow entity to be synchronized with database automatically. So each BP instance can modify its process entity without worrying about the database access[15][16].

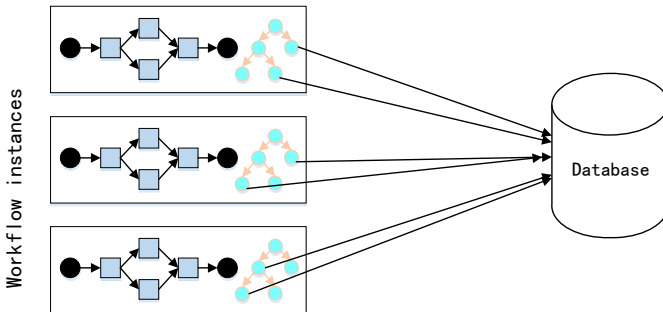


Fig. 6 data mapping

The SeGA layer acts as a Service Wrapper/Mediator [17]and separates data from the execution engine, which is a key advantage for BPaaS [1]. It consists of a SeGA dispatcher and a SeGA mediator. When an external event arrives, the dispatcher fetches the relevant universal artifact from a universal artifact repository, extracts the schema from the universal artifact and maps it back to the original

form(GSM[18][19] or EZ-Flow) restores the original artifact instance (GSM or EZ-Flow), and sends the external event, schema, and the original artifact instance to the mediator. When the mediator receives the event, schema, and the instance, it deposits the artifact schema in the appropriate location where the Barcelona/EZ-Flow engine will access, and passes the control over to the Barcelona[20]/EZ-Flow engine by forwarding the event. When the Barcelona/EZ-Flow engine receives the incoming event, it executes the next step and updates the artifact instances according to the schema deposited by the mediator; and outgoing events may also be sent directly from the engine if there exists task invocation during the execution. Once it completes, the mediator fetches the updated artifact instances, together with their schemas and states, and sends them back to the dispatcher. The dispatcher then maps the instances and schemas back to universal artifacts and stores into the corresponding repository.

The Execution layer consists of two, i.e., the SeGA mediator and JTang workflow system. The Execution layer is responsible for pure workflow management, not involving any data. The Task Manager invokes SeGA API and JTang API to connect the remote database. After loading data into the workflow system, SeGA invokes the JTang engine to execute the workflow. The JTang engine handles the event, updates the instance and sends the processed events. The JTang engine determines whether the next activity can be implemented according to the execution flow. The execution result will be received by the Task Manager component in the SeGA layer to decide which task is to perform next. There are two components that can control the task performance. 1) One is the SeGA Manager component that informs the program to execute the task. 2) The other is that people can start an initiator component to initialize the artifact instance. Besides, the Runtime Monitor component and Anomaly Handler component can handle the possible errors in the execution process in a timely manner.

#### IV. IMPLEMENTATION

When people use BPaaS in a real application, the proposed system provides templates for the process designer to specify their requirements. Based on the template-EDB and entity-EDB mapping, people can acquire the mapping between the schema and EDB table attributes. The proposed system stores such mapping relations into the dataset MySQL. We use the programming library JointJS1 to customize the BP template, which is displayed on the Web. We set the access permission to data for each activity according to two rules, i.e., read-only and write-only. Thus, people can acquire a complete business process, and store the web schema of each activity in the local database. We open the source code of BPaaS at <https://github.com/rendididi/SeGA>.

#### V. DEMONSTRATION

In demonstration, we set the process management of Urumqi Housing Management Bureau (UHMB for short) as a template. Due to the large population migration, we need a pre-step to make an investigation to confirm whether the residential address of the applicant changed or not. Further, different organizations have different configuration, which means

different data formats and sizes. So we need both Customize Entity and Customize Process component. We employ the following four procedures to implement the customization of BPaaS.

**Entity Customization.** The business experts choose the appropriate template to customize the entity, and send the request or data to the process execution engine. We can get the activity ID with some calculation, and after that, we can modify the artifact attributes of the template according to the requirement of the enterprise. For example, in the UHMB case, we need an extra attribute to confirm the current address of an applicant.

**Data Mapping.** This procedure selects the artifact database to map, and sets the basic information, such as type and address, to build the mapping relationship of DB Template and EDB. In detail, this procedure conducts the one-to-one mapping between an artifact property with a field in the database, and handles the automatic data synchronization in business process management. Also, the data mapping procedure modifies the processes for UHMB according to the template.

**Process Customization.** This procedure holds the position task modules to modify the processes of UHMB and sets the basic attributes. For each task in the process sequence, this procedure configures read and write permission for the attributes that are generated in the first procedure (Entity Customization). The system will return the corresponding forms according to the activity ID when the operators are responsible for different steps in the process steps.

**Process Execution.** In the Task Viewer, the operator generates an instance and sends the request to the process execution engine, which will return the activity ID with some calculation. According to the ID, this procedure generates the appropriate Web form. The system updates the database after the form is filled and submitted, and sends the form along with the ID to the execution engine. Meanwhile, the engine returns the ID of the activity that is to be executed next, and repeats the above procedure until the whole business process finishes.

The demonstration introduced in this paper is accessible via the link <https://youtu.be/ms2435UyeEI>.

#### ACKNOWLEDGMENT

The research in this paper is supported by the National Key Technology R&D Program under Grant (No.2014BAK14B04), Zhejiang Provincial Natural Science Foundation (No.LY12F02003), China Postdoctoral Science Foundation under Grant (No.2013M540492). Last but not the least, I would like to express my heartfelt gratitude to Professor Jianwen Su, who have instructed and helped me for this paper.

#### REFERENCES

- [1] Xu, W., Su, J., Yan, Z., Yang, J., Zhang, L. An artifact-centric approach to dynamic modification of workflow execution. In: Proc. 19th Int. Conf. on Cooperative Information Systems (CoopIS), pp. 256–273, 2011.
- [2] Y. Y. Sun, J. Su, and J. Yang. Separating execution and data management: A key to business-process-as-a-service (BPaaS). In Proc. Int. Conf. on Business Process Management (BPM), 2014.
- [3] F. Casati, M. Castellanos, U. Dayal, and N. Salazar. A generic solution for warehousing business process data. In Proceedings of the 33rd international conference on Very large data bases, VLDB '07, pages 1128–1137. VLDB Endowment, 2007.
- [4] K. Bhattacharya, C. Gerede, R. Hull, R. Liu, and J. Su. Towards formal analysis of artifact-centric business process models. In Proc. Int. Conf. on Business Process Management (BPM), Brisbane, Australia, September 2007.
- [5] Nigam A, Caswell N S. Business artifacts: An approach to operational specification[J]. *Ibm Systems Journal*, 2003, 42(3):428-445.
- [6] Chang, S.H., Kim, S.D.: A Variability Modeling Method for Adaptable Services in Service-Oriented Computing. In: Proc. 11th International Software Product Line Conference, , pp. 261–268, SPLC 2007.
- [7] Koning, M., ai Sun, C., Sinnema, M., Avgeriou, P. Vxbpel: Supporting variability for web services in bpel. *Information and Software Technology* 51, pp. 258–269, 2009.
- [8] Mietzner, R., Metzger, A., Leymann, F., Pohl, K.: Variability modeling to support customization and deployment of multi-tenant-aware software as a service applications. In: Proc. ICSE Workshop on Principles of Engineering Service Oriented Systems. Washington, USA, pp. 18–25, 2009.
- [9] R. Accorsi. Business Process as a Service: Chances for Remote Auditing. In *COMPSAC Workshops*, pages 398–403, 2011
- [10] F. Casati, M. Castellanos, U. Dayal, and N. Salazar. A generic solution for warehousing business process data. In: Proc. 33rd international conference on Very large data bases, VLDB '07, pages 1128–1137. VLDB Endowment, 2007.
- [11] L. Limonad, D. Boaz, R. Hull, R. Vacul'in, and F. T. Heath. A Generic Business Artifacts Based Authorization Framework for Cross-Enterprise Collaboration. In *SRII Global Conference*, pages 70-79, 2012.
- [12] Sun Y, Su J, Yang J. Universal Artifacts: A New Approach to Business Process Management (BPM) *Acm Transactions on Management Information Systems*, 7(1):3, 2016.
- [13] Pathirage, M., Perera, S., Kumara, I., Weerawarana, S. A multi-tenant architecture for business process executions. In: Proc. IEEE International Conference on Web Services (ICWS), pp. 121–128, 2011.
- [14] Y. Sun, J. Su, B. Wu, and J. Yang. Modeling Data for Business Processes. In Proc. Int. Conf. on Data Engineering, ICDE '14, 2014.
- [15] G. Liu, X. Liu, H. Qin, J. Su, Z. Yan, and L. Zhang. Automated realization of business workflow specification. In: Proc. ICSOC/ServiceWave Workshops, pages 96–108, 2009.
- [16] W. Xu, J. Su, Z. Yan, J. Yang, and L. Zhang. An artifact-centric approach to dynamic modification of workflow execution. In Proc. of the 19th International Conference on Cooperative Information Systems, CoopIS '11, pages 256–273, 2011.
- [17] Y. Sun, W. Xu, J. Su, and J. Yang. SeGA: A Mediator for Artifact-Centric Business Processes. In: Proc. Int. Conf. on Cooperative Info. Sys., CoopIS '12, pages 658–661, 2012.
- [18] R. Eshuis, R. Hull, Y. Sun, and R. Vaculin. Splitting GSM Schemas: A Framework for Outsourcing of Declarative Artifact Systems. In *BPM*, pages 259-274, 2013.
- [19] R. Hull, E. Damaggio, and et al. Business artifacts with guard-stage-milestone lifecycles: managing artifact interactions with conditions and events. In Proc. Of the 5th ACM International Conference on Distributed event-based system, DEBS 11, pages 51–62, 2011.
- [20] T. Heath, D. Boaz, M. Gupta, R. Vacul'in, Y. Sun, R. Hull, and L. Limonad. Barcelona: A Design and Runtime Environment for Declarative Artifact-Centric BPM. In Proc. of the 11th Int. Conf. on Service-Oriented Computing, ICSOC, pages 705–709, 2013.