Extracting Services from Legacy for Service-Oriented Business Processes: Challenges and Issues

Youcef Baghdadi[1] and Ricardo Pérez-Castillo[2]

[1]Department of Computer Science, Sultan Qaboos University,
PO Box 36 – PC 123, Al-Khod, Oman

[2]Alarcos Research Group, University of Castilla-La Mancha,
Paseo de la Universidad, 4 13071, Ciudad Real, Spain

Abstract. Non-agile enterprises slowly adapts to changing business requirements, which could be harmful and affects their competitiveness. Their business processes are rigid and could not respond to the changes. One of the solutions is to use a service-oriented business modeling by stressing out role of SOA and Web services. In this modeling, business processes are represented by specialized services having separated concerns. These are (i) controller service, (ii) state service, and (iii) worker services. This solution requires an approach to: (1) map the existing services of the organization or its partners, or (2) reverse engineering techniques to provide business objects, represented in the information system, as a set of common, consistent service providers.

1 Introduction

Non-agile enterprises slowly adapts to changing business requirements, which could be harmful and affects their competitiveness. Their Business Processes (BPs) are rigid and could not adapt to the changes. This agility is not possible if all the instances of a same BP run the same way because of the architecture of the existing Information System (IS) that supports the BPs.

One of the solutions is to reengineer the BPs by using a service-oriented business modeling stressing out the roles of SOA, and Web services in promoting agility. First, we define a BP a service that responds to Business Events (BEs) of different natures. It specifies a state that reflects the control flow and data flow, where changes in BEs are solely reflected in the state. The BP adapts to the state changes, which allows a dynamic realization. This service is itself realized by enterprise assets that play the role of service providers[1]. Then, we model a BP by a set of specialized services, having separated concerns. These are: (i) controller service, (ii) state service, and (iii) worker services. The controller service uses the state service to match and invoke the worker services that realize the required activities [2]. The worker services are themselves provided by shared, discoverable, reusable enterprise assets known as Business Objects (BOs). BOs model the enterprise service units that encapsulate activities and data; and provide them as services to all BPs.

Baghdadi Y. and Pérez-Castillo R..
Extracting Services from Legacy for Service-Oriented Business Processes: Challenges and Issues.
DOI: 10.5220/0004574100030011
In *Proceedings of the 1st International Workshop in Software Evolution and Modernization* (SEM-2013), pages 3-11
ISBN: 978-989-8565-66-2

This solution requires both reverse and forward engineering. This work limits to a reverse engineering that (i) transforms the IS into BOs, including the existing enterprise service portfolio into some of the specialized services, (ii) extract services from existing BPs (if any), and (iii) reusing partner and provider services, or (iv) all of them.

- The reverse engineering would modernize the IS so that it provides the required worker services.
- The mapping would map the existing service portfolio into specialized services, including the controller and worker services.
- The extraction would abstract the existing BPs.

This approach would have practical implications in terms of (i) compliance with SOA, and (ii) reuse of the existing assets of an organization to rapidly modeling, enacting, and execution of BPs.

The remainder of this paper is organized as follows: Section 2 provides some related work. Section 3 details the concepts of the BP modeling. Section 4 develops the reverse engineering approach. Finally, Section 5 discusses further development.

2 Related Work

Managing BPs as services has recently started making its way in IS by promoting BPs as compositions of loosely coupled services having separated concerns. This work concerns with two perspectives: (1) modeling BPs as services, (2) modernizing supporting ISs with respect to SOA. From service-oriented BP modeling, the academics have raised the importance of service-oriented as one of the top three issues to deal within BPs [3]. In [4], the author proposed a business model for B2B integration through Web services. The authors in [5] proposed an approach for designing BPs in service-oriented way, where a service composition process organizes the discovery, selection and assembly of business services to build BPs tailored to business designer's requirements. In [6], the author investigated how to extend Event-Process Chain (EPC) to come up with a new modeling language for service-oriented BP management.

From a general perspective of modernizing ISs and BPs, different approaches have been proposed [7]. In [8], the authors developed Marble to reengineer BPs within a BP archeology. This approach makes it possible to obtain BP models from source code and other software artifacts by reverse engineering. Efforts that concern with modernization for SOA have been reported in [9]. In [10], the author proposed to reverse engineer relational databases into services. In [11] the authors have defined a technique to wrap legacy applications for reuse in a SOA.

From the perspective of mapping enterprise services into a given service-oriented business modeling, there is a lack of approaches, though many authors such [12], [13], [14], [15],[16],[17],[18], [19] and [20] have attempted to classify services into taxonomy.

The drawback of these approaches is that they do not explicitly consider the reuse existing service portfolio of services in their service-oriented BP modeling neither have they reverse engineered the IS into BOs that provide reusable services.

3 Service-Oriented Business Process Modeling

Modeling is a representation technique that relies on first, a set of limited notations and concepts to represent relevant elements such as enterprise BPs and second, a set of rules that guarantee the proper use of these concepts.

A BP has a value expressed in terms of quality of services to customers. That is, a BP is a description of a service provided to customers upon their demands (of different natures). A set of coordinated activities realize the service. The IS supports these activities by providing them with data they consume.

A BP modeling captures the relevant properties with respect to the above-mentioned definition of BP. In our service-oriented modeling approach, specialized services are used as constructs to represent the different relevant elements of BPs.

A BP has a set of state values (including initial and final state values) that the BP modeler sets and changes over time when business requirements change. A state value reflects the execution BP.

The following are the related concepts used to model a BP with services and the relationships between them. These are 'use', 'has', or 'is' relationships.

3.1 Modeling Concepts

A ***service*** [14] may be defined from business and technology perspectives. In our modeling, we use web services and data services.

Our service-oriented approach for BP modeling emphasizes the separation of concerns that differentiate the activities of control and execution. Similarly, the data packaged into BOs are separated from the state of the BP. Therefore, we specialize a service into controller service, state service, and worker service.

The ***Controller Service*** (CS) is the central element of our modeling. It oversees a BP execution through its state. The CS deals only with the control and coordination of the BP. It invokes a State Service (SS) to retrieve the state of the BP; and accordingly invokes the respective Worker Service (WW) and updates the BP state when any of the WS terminates its job. The CS is invoked by an Initiator Web Service (IWS).

The ***Initiator Web Service*** (IWS) deals only with the initialization and starting of an instance of the BP.

The ***State Service*** (SS) is a data service that represents the structure of the state of a BP. It is used by the CS to retrieve the state of the BP.

The ***Worker Services*** (WWs) add value to a BP towards the achievement of its goal. WWs are provided by BOs. The CS according to the BP state invokes them. Inversely, WWs report the outcome of their execution to the CS.

WWs may use ***Data Se****rvices* (DS) if necessary to retrieve simple or integrate data from BOs. DSs retrieve or integrate the requested data from the BOs.

Business Object is "a representation of a thing active in the business domain, including its business name and definition, attributes, behavior, relationships, and con straints" (OMG). To represent BOs in a consistent way regardless of the needs of BP modeling, we insist on the monolithic representation of the activities and data of these BOs. Unfortunately, this representation is not happening nowadays as different representations (or images) of the same BO are developed depending on the needs of each organizational body, which leads to a limited view and inconsistency of BO across the whole organization.

3.2 Relationships between the Specialized Services

There are four types of relationships: association, specialization, realization, and use.

- ***Association Relationships*** indicate how the elements of a BP environment are associated with each other. For instance, a BP is associated with an event, a set of BOs, a set of states, including an initial state, and a final state.
- ***Specialization Relationships*** indicate the specific roles of some elements of the model. For instance, a service may be a CS, a SS, or a WW.
- ***Use Relationships*** show that some services use the capabilities of others. For instance, the CS uses both WWs and SS. The latter provides it with the state of the BP, whereas the former perform the required activity. The IWS uses the CS.
- ***Realization Relationships*** show that WWs are realized in the IS by the BOs, whereas the SS is realized by a specific data structure representing the state values.

4 Mining Services for Service-Oriented Business Processes

The common problem of service-oriented BP modeling is that most techniques discard existing services of organizations. Our approach proposes to transform an ***as-is*** BPs into ***to-be*** BPs by using different techniques as shown in Figure 1, namely (i): a mapping existing Enterprise Portfolio Services (ESP), (ii) reverse engineering IS, (iii) extract services from exiting BPs, or (iv) reuse Partner and Provider Services (P&P). In this approach, everything transforms to reusable services.

4.1 Service Taxonomy

The objective of service mapping is to identify services that are valuable, from business and IT perspectives. However, what constitutes a service has different meanings due to the lack of a standard classification. Several researchers attempted to provide taxonomy [15],[16],[17],[18], [19] and [20]. Taxonomy supports service-mapping process by clarifying the roles of the different types of services, which helps in understanding the role of each service. It also assists with the discoverability of services, which can further promote reuse. In addition, taxonomy helps to make organizational decisions like how to obtain a capability (build vs. buy vs. lease). To summarize the classification efforts of these researchers, services can be divided into three broad categories:

1. ***Conceptual Services*** (also called business services) service represents the core of software product requirements. They express organizational ideas, thoughts, opinions, views, or themes that propose software solutions to organization.
2. ***Capability Services*** provide an explicit business value, ranging from generic services, reused in multiple service-oriented applications to specialized services, part of a single service-oriented application. Capability services include: (i) *Process services* are services whose operations are directed by the BP definition. BPEL are example of such kinds of services. (ii) *Task services* (aka application/activity/capability) encapsulate business logic specific to activities or BP. A task service represents an atomic unit of process logic. (iii) *Entity services* (also called data service) represent one or more related business entities such as invoice. Entity service is considered a highly reusable service because it is agnostic to most BPs. (iv) *Utility services* contain logic derived from a solution or technology platform. Utility services expose capabilities of multiple applications within the organization, including migrated, reengineered legacy systems, and some Commercial Of-The-Shelf software (COTS). (v) *Hybrid services* contain logic derived from both BPs and applications. Hybrid services expose capabilities wrapped legacy systems and some COTS that may exists within the organization. (vi) *Partner services* are offered to/by external partners based on agreed terms, this type of services is known as Application as a Service (AaaS) or Software as a Service (SaaS).
3. ***Infrastructure Services*** are part of the organization supporting distributed computing infrastructure. These are: (i) *Communication services* transport messages into, out of, and within the system. Examples include publish-subscribe services, queues, and ESB; and (ii) *Auxiliary services* provide facilities for monitoring, diagnostics, and management activities of other services. These may include statistical information.

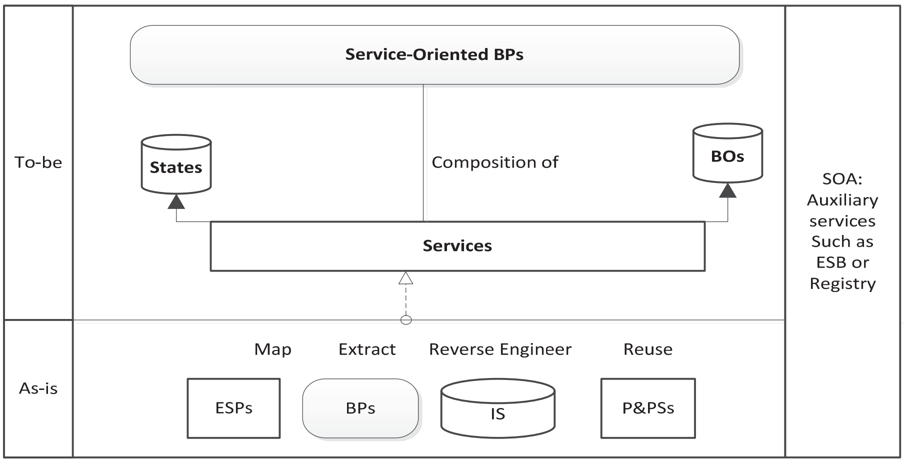

Fig. 1. Techniques to transform as-is BP into service-oriented BP.

4.2 Service to Business Process Mapping

Table 1 shows that any kind of existing services within the organization portfolio (first column) could be mapped into one of the services we use in our modeling: a CS, a WW, or a DS. It is worth noting that none of the ESP is mapped into a SS since SS is specific to our modeling. Infrastructure services map to SOA auxiliary services.

Table 1. Services-to-service-oriented BP Mapping.

Existing (ESPs)	**Refined Services**	**Service-Oriented BP**
Conceptual services		***CS***
Capability services	*Process service*	***CS***
	Task service	***WW***
	Entity service	***WW or DS***
	Utility service	***WW***
	Hybrid service	***WW***
	Partner services	***WW***
Infrastructure services	*Communication services (e.g., ESB)*	***SOA auxiliary services***
	Auxiliary services (e.g., Registry)	

This mapping might be automated if the organization could have a portfolio or services that have common meaning, which is the role of the BOs in our modeling.

4.3 Reverse Engineering LIS to Extract Services

When organizations do not have a portfolio of services, the reverse engineering of legacy IS (LIS) is required to first mine BPs, then extract services from these BPs to further reuse them in service-oriented BP modeling.

Most BPs in organizations are supported by their IS. The optimal BPM is therefore achieved when organizations additionally combine the management of their LIS [21]. The configuration management of LIS is particularly important since these systems undergo a considerable amount of changes during their lifecycles. Because of the evolutionary maintenance over time, new business knowledge and rules are embedded in LIS. This embedded business knowledge may not exist anywhere else [22]. The evolution of IS in isolation consequently affects the evolution of BPs (see scenario 1 in Fig. 2). In this case, it is necessary to discover and reconstitute the underlying BPs that are currently supported by LIS [23].

However, there are many organizations that currently carry out a vast amount of daily transactions through their IS without having ever done their own BP modeling. When these organizations deal with BP modeling for the first time, a recurrent method by which to attempt this modeling is the extraction of BP from LIS [24] (see scenario 2 in Fig. 2). This is owing to the fact that LIS is one of the few knowledge assets in organizations that can be used to attain an accurate understanding of the actual BP.

In both scenarios (Fig. 2), retrieving an up-to-date version of BPs from LIS allows organizations to take advantage of at least two main benefits:

– Benefits for BP modeling: BPs can always be up-to-date. Organizations may therefore conduct BPM by following the continuous improvement process [25]. This kind of BPM facilitates an agile adaptation of BP to meet the changes that occur in the uncertain environment. The rapid evolution of BPs allows organizations to maintain, or even improve, their degree of competitiveness [26].

- Benefits for LIS: LIS continue to be modernized on more occasions. A recent study by the SEI (Software Engineering Institute) states that it is first necessary to retrieve embedded business knowledge in order to modernize systems in line with the organization BPs [27]. Organizations can thus modernize their LIS whilst they align the new systems with their actual BPs. LIS is therefore evolved rather than being immediately retired and the ROI (return of investment) on such systems is improved.

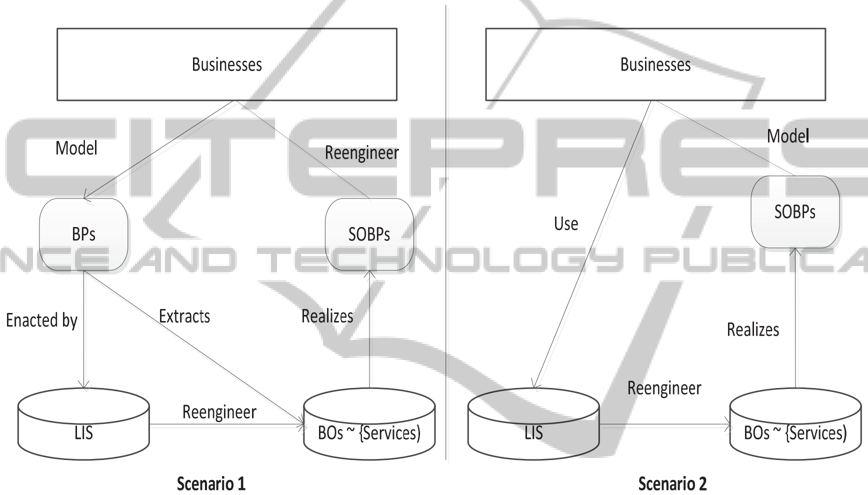

Fig. 2. Scenarios to discover and reconstitute BP embedded in systems.

Once we extract a BP from LIS, it is then possible to present it as a set of coordinated activities and data. The BP, its activities and data could easily be mapped into services. Indeed:

- A BP is mapped into CS
- Each activity output of an activity is mapped into state value
- Each activity is mapped into WW
- Each piece of data is mapped into data service

5 Conclusions

We have defined a service-oriented business process modeling, where the elements of business process environment are modeled as services, including controller service, state service and worker services. The worker services and state service are provided by real and artifact business objects respectively.

This modeling has required different techniques to move from as-is BPs to to-be BPs. We have categorized these techniques into: (i) reverse engineering the legacy information system and mapping the enterprise service portfolio, (ii) extracting services from traces of running business processes, and (iii) reusing partner and provider services. Then, we have sketched out the process of each technique.

This is a step towards moving SOA maturity towards the next levels, where services are part of the requirements and the solutions as business-related services and IT-related services respectively. This would promote integration, composition, flexibility, and agility in response to changing business events.

Although, we have presented approaches and processes for transformation techniques having a real impact on the way BPs should responsive to the changes in the business requirements by using web service-based SOA, this work has limitation. Therefore, we would consider that this work has presented the service-oriented business process as rather a roadmap towards research issues and questions related to transformation techniques than a definitive solution.

We need to develop tools and dig deeper in the different transformation techniques.

Further work would complete the close the cycle with the forward engineering phase by using MDD, as services define both the requirements and the solutions.

References

1. Cummins, F. A.: Building the Agile Enterprise: With SOA, BPM and MBM, Morgan Kaufmann, 2010.
2. Baghdadi, Y.: A methodology for web services–based SOA realisation, Int. Journal of Business Information Systems, 10(3), pp. 264-297, 2012.
3. Indulska, M., Recker, J., Rosemann, M. and Green, P.: Business Process Modeling: Current Issues and Future Challenges, Advanced Information Systems Engineering
4. LNCS, Vol. 5565, 2009, pp 501-514, 2009.
5. Baghdadi, Y.: A business model for B2B integration through Web services, in Proc. of IEEE Int. Conference on e-Commerce Technology, 2004. CEC 2004, pp. 187–194, 2004.
6. Cauvet, C. and Guezilian, J.: Business Process Modeling : a Service-Oriented Approach, in Proc. of Hawaii Int. Conference on System Sciences, 41st Annual pp. 1–8, 2008.
7. Stein, S.: Modelling Method Extension for Service-Oriented Business Process Management," Thesis, 2008.
8. Rahgozar, M. and Oroumchian, F: An effective strategy for legacy systems evolution, Journal of Software Maintenance and Evolution: Research and Practice, Vol. 15, no. 5, pp. 325–344, Sep. 2003.
9. Pérez-Castillo, R., de Guzmán, I. G.-R. and Piattini, M.: Business process archeology using MARBLE, Information and Software Technology, Vol. 53, no. 10, pp. 1023–1044, 2011.
10. Khadka, R., Saeidi, A., Idu, A. Hage, J and Jansen, S.: Legacy to SOA Evolution : Evolution : A Systematic Literature Review, Technical Report UU-CS-2012-006, 2012.
11. Baghdadi, Y.: Reverse engineering relational databases to identify and specify basic Web services with respect to service oriented computing, Information Systems Frontiers, Vol. 8, No. 5, pp. 395–410, Nov. 2006.
12. Sneed, H. M., Schedl, S and. Sneed, S. H.: Linking legacy services to the business process model, in IEEE 6th International Workshop on the Maintenance and Evolution of Service-Oriented and Cloud-Based Systems (MESOCA), pp. 17–26, 2012.

13. Al-Rawahi, N. and Baghdadi, Y.: Approaches to identify and develop Web services as instance of SOA architecture, In Proc. of ICSSSM '05. Int. Conference on Services Systems and Services Management, 2005, pp. 579–584, 2005.
14. Baghdadi, Y. and Al-Rawahi, N.: An architecture and a method for Web services design: Towards the realization of service-oriented computing', Int. Journal of Web and Grid Services (IJWGS: InderScience), 2(2): pp 119-147, 2006
15. Chesbrough, H. and Spohrer, J.: A research manifesto for services science, Communications of the ACM, vol. 49, no. 7, p. 35, Jul. 2006.
16. Gu, Q. and Lago, P.: Service Identification Methods : A Systematic review, LNCS 6481, pp. 37–50, 2013.
17. Lago, P. and Razavian, M.: Pragmatic Approach for Analysis and Design of Service Inventories, LNCS, Vol. 7221, pp. 44-53 pp. 44–53, 2012.
18. Marks, E. A. and Bell, M.: Executive's Guide to Service-Oriented Architecture, John Wiley & Sons, 2006.
19. Cohen, S.: Ontology and taxonomy of services in a service-oriented architecture, The Architecture Journal, Vol. 11, pp. 30-35, 2007.
20. Cho, M. J.. Choi, H. R. Kim, H. S. Hong, S. G. Keceli, Y and Park, J: Service Identification and Modeling for Service Oriented Architecture Applications, In Int. Conference on Software Engineering, Parallel and Distributed Systems, pp. 193–199, 2008
21. Erl, T., Taub, M. L., Hart, K. Mcfarland, J. and Young, T: SOA Design Patterns. 2009.
22. Jeston, J. and Nelis, J.: Business process management, Elsevier Publisher, 2012.
23. B. Paradauskas, B. Laurikaitis, A.: Business knowledge extraction from legacy information systems, Information Technolgy and Control, Vol. 35, No. 3, pp. 214-221, 2006.
24. Van Den Heuvel, W. J.: Aligning modern business processes and legacy systems : a component-based perspective, The MIT Press ©2009
25. Van der Aalst, W. and Reijers, H.A.,: A.J.M.M. Weijters, A.J.M.M, van Dongen, B.F., A.K. Alves de Medeiros, Song, M. and H.M.W. Verbeeka, H.M.W: Business process mining: An industrial application, Information Systems, Vol. 32, Issue 5, July 2007, Pages 713–732007.
26. Davenport, T. H.: Need radical innovation and continuous improvement? Integrate process reengineering and TQM, Strategy & Leadership, vol. 21, no. 3, pp. 6–12, Dec. 1993.
27. Weske, M.: Business Process Management: Concepts, Languages, Architectures, Springer, 2012
28. Lewis, G. A. Smith, D. B.: A Research Agenda for Service-Oriented Architecture): Maintenance and Evolution of Service-Oriented Systems, TECHNICAL NOTE CMU/SEI-2010-TN-003, no. March, 2010.