International Journal of Advanced Science and Technology
Vol.79 (2015), pp.37-48
http://dx.doi.org/10.14257/ijast.2015.79.04

Using a Reversed Affinity Analysis Technique to Validate Service Identification Approaches

Rana Yousef

Computer Information Systems Department, The University of Jordan
rana.yousef@ju.edu.jo

Abstract

Service identification (SI) is one of the challenging activities in developing service oriented models. Many service identification approaches have been proposed in literature ranging from top-down to bottom-up. However, the literature lacks enough research in the field of validating service identification approaches themselves. Most of the available techniques to validate SI approaches are theoretical in nature and uses case studies and/or examples to explain the approach's features. In this paper, we present a technique to validate the execution behavior of Service Identification approaches using a reversed affinity analysis method. This method is used to validate the granularity level, cohesion and low coupling of the identified services. The proposed technique is demonstrated using one of the SI approaches and is deployed on a real case study from the healthcare domain.

Keywords: *Service oriented architecture, service identification, affinity analysis, business process model, service validation*

1. Introduction

Service-oriented modeling and design consists of three main steps: service identification (SI), specification and realization of services [1]. The identification of services is concerned with determining the appropriate services to be implemented in a service-oriented architecture and defining which functions should be part of each service, most of the current approaches relay on business process descriptions to identify services [2, 3].

In the literature, most of the service identification approaches are evaluated by deploying them into practice. Some approaches are validated by showing their effectiveness in real life projects or by experimenting them in case studies. Other approaches only provide some examples to explain the proposed service identification method [3, 4].

In this paper, we use a reversed method of a BPA-driven service identification approach in order to validate the identified services, more specifically to validate the granularity level, cohesion and low coupling of the identified services. The BPA-driven SI approach used in our method is the Elementary business process and business Entity Affinity analysis Technique (EEAT) by Jamshidi *et al.*, [5]. This approach identifies groups of functions with the right granularity to satisfy low coupling, high cohesion, and low reuse cost principles for reusable software services based on affinity clustering.

The rest of this paper is organized as follows. Section 2 reviews the key related work in the field of validating service identification approaches. In Section 3, we present our technique for validating software services identified using SI approaches after reflecting on the EEAT method by Jamshidi *et al.*, [5]. A demonstration for this technique is presented in Section 4. Finally, Section 5 concludes the paper in addition to identifying directions for future work.

ISSN: 2005-4238 IJAST

2. Related Work

In this section, we present the most related work to our proposed validation technique, where some of the work done to validate service identification approaches is reviewed based on the systematic literature review conducted in [4].

The literature lacks enough research in the field of validating service identification approaches. Based on the systematic literature review conducted in [4] and reviewing more recent work in the field, it was found that no explicit validation technique is available to evaluate a SI approach.

Most of the validation techniques for SI approaches are only evaluations of proposed SI methods, where these approaches are validated by showing their effectiveness in real life projects [6, 7] or by experimenting them in case studies [8, 9]. Other validation techniques only provide some examples to explain the proposed service identification method [10, 11].

Some validation techniques explain how to use the proposed SI methods in order to improve their usability. Others evaluated their methods in terms surveys or comparisons.

Accordingly, in this paper, we provide an explicit validation technique to evaluate SI approaches through assessing the cohesion and low-coupling of each service. This will be accomplished by a reversed affinity analysis techniques to prove that the identified services group together all functions that create and update the same entities defining non-redundant building blocks (candidate services).

3. The Proposed Technique to Validate SI Approaches

In this section we present the new technique to validate SI approaches after briefly explaining the EEAT method by Jamshidi [5].

3.1. The EEAT Service Identification Method by Jamshidi et al [5]

Jamshidi, *et. al.*, [5] realized that the key activities that are needed to construct a quality based service-oriented solution is the identification of its architectural elements with the right granularity. Accordingly, they proposed a new process to identify and specify enterprise software services along with their architectural elements. They proposed a method for identifying service based on affinity clustering, Elementary business process and business Entity Affinity analysis Technique (EEAT). This technique identifies groups of functions with the right granularity to satisfy low coupling, high cohesion, and low reuse cost principles for reusable software services. Using the EEAT clustering technique, the tags of the matrix cells have priorities as C>U>D>R. The objective is to deduce groups of functions and entities that share create and update operations. Grouping together all functions that create and update the same entities defines non-redundant building blocks (candidate services).

3.1. Validating Service Identification Approaches: Satisfying the SI criterion of Jamshidi et. al. Approach [5]

In our proposed technique, we use the service identification approach of Jamshidi *et al.*, [5] to define our validation method, where we deploy the SI criterion used in their approach to inform the validity of services identified using a certain service identifications approach.

As we have mentioned previously, the SI criterion used in Jamshidi *et. al.*, approach is based on affinity clustering where the objective is to deduce groups of functions and entities that share create and update operations. This objective is accomplished by using the Elementary business process and business Entity Affinity

analysis Technique (EEAT) [5]. So, grouping together all functions that create and update the same entities defines non-redundant building blocks (candidate services).

Accordingly, our proposed validation technique aims to prove that the functionality of each of the candidate services identified using a certain SI approach creates and updates the same entities, and hence to inform the validity of candidate services.

The algorithm shown in Figure 1 defines our approach for validating the identified candidate services according to the EEAT technique.

Lines 1 - 4 of Algorithm I identify both the capabilities of each service, which provide us with the required functionality and functional boundaries, and the entities on which the capabilities act on. This makes our method applicable for SI approaches based on top-down methods such as business process decomposition, business functions, business functions business entity objects, goal-driven and component based. Where using these methods, capabilities and entities are tractable and can be easily identified. However, our proposed techniques must be amended to deal with SI approached based on bottom-up methods, such as existing supply.

Lines 5 – 13 of the algorithm builds the Validating_CRUD matrix using the information provided in the previous step. Then the boundary of each service is indicated as a cluster of the service capabilities and entities on which the capabilities act (Line 14).

Lines 15-21 requires that each service cluster is to be tested to ensure that it identifies groups of functions with the right granularity to satisfy low coupling and high cohesion according to Jamshidi et al [5]. This is accomplished by ensuring that the entities of the cluster are created and updated only by the functions within the same cluster. This is because the EEAT technique groups together all functions that create and update the same entities to define candidate services.

4. Demonstrating the Proposed Validation Technique Using a Case Study

In this section we demonstrate our proposed validation technique using one of the top-down SI approaches. The selected SI approach identifies services from an organization's Riva business process architecture [12]. This SI approach will be deployed using a real case study from the healthcare domain, the Cancer Care and Registration in Jordan. First, we provide a brief explanation of the selected SI approach to be evaluated, then we show the services identified using this approach when applied on the Cancer Care and Registration case study. Finally, we deploy our validation techniques on the selected SI approach.

Algorithm I: Validating a Set of Candidate Services According to the EEAT Technique

***Input**: the set of the identified candidate services generated using a SI approach $S=\{s_1, s_2, \ldots, s_i, \ldots, s_n\}$, $0 \le i \le n$*

***Output**: Validating_CRUD Matrix, result of validation*

Begin

1. *For each service s_i in S do*
2. *Find the Capabilities of s_i, $Cbt=\{cbt_1, cbt_2, \ldots, cbt_k, \ldots, cbt_x\}$, $0 \le k \le x$*
3. *Find the EBEs on which the capabilities of s_i act on, $EBE=\{ebe_1, ebe_2, \ldots, ebe_l, \ldots, ebe_y\}$, $0 \le l \le y$*
4. *End for*
5. *Build a Validating_CRUD matrix as follows:*
6. *for each capability cbt_k in Cbt do*
7. *set the cbt_k as the row of the Validating_CRUD matrix*
8. *for each essential business entity, ebe_l in EBE do:*
9. *set the ebel as the column of the Validating_CRUD matrix*
10. *Set the matrix cell as the relationship between cbt_k and ebe_l which is one of the CRUD functions (Create, Read, Update or Delete)*
11. *Exclude the points of interaction with other services*
12. *end for*
13. *end for*
14. *Indicate the boundary of each service as a cluster of the service capabilities and entities on which the capabilities act.$C=\{ c_1, c_2, \ldots, c_p, \ldots, c_z\}$, $0 \le p \le z$*
15. *for each cluster c_p in C do*
16. *If the entities of the cluster are created and updated only by the functions within the same cluster (i.e. the columns of the Validating_CRUD matrix only contain the C and U values within the cluster only, not exceeding the cluster boundaries) then*
17. *Service identification criterion is met for this cluster*
18. *Otherwise*
19. *Service identification criterion is not met for this cluster*
20. *End if*
21. *End for*
22. *If all clusters meet the service identification criterion, then*
23. *The set of services represents a valid set of candidate services*
24. *End if*

End

Figure 1. The Proposed Algorithm to Validate a Set of Candidate Services According to the EEAT Technique

4.1. The Selected SI Approach to be Evaluated: Riva-Based SI Approach [12]

The Riva method is used to identify an organization's business process architecture [13]. The Riva-based architecture is derived from an understanding of what business the organization is in, rather than its current structure or culture. So, once the architecture is understood, it becomes apparent what is required from the IT systems supporting these processes. Riva-based BPA was used in [12] as a starting point to generate software services for a SOA-based system.

In order to identify an organization's process architecture in Riva, the following steps should be taken [14]:

1. Agree the boundary of the organization.
2. Brainstorm the organizations' subject matter to identify Essential Business Entities (EBEs)

3. Classify these EBEs that have a lifetime which is handled by, or are the responsibly of, members of the organization as Units of Work (UoWs)
4. Draw a UoW diagram that depicts the dynamic relationships between UoWs.
5. Assume that for each UoW, there is:

a) a case process that handles single instances of the UoW; and
b) a case management process for dealing with the flow of instances.

6. Transform the UoW diagram into a first-cut process architecture; then, use the provided heuristics, to generate a second-cut process architecture.

The Riva-based SI approach, uses the second-cut process architecture diagram to identify Riva Process Architecture (RPA) Clusters. These RPA clusters are identified from the BPA diagram as the set of standalone CPs (have no Start, Request or Deliver relations) as well as the set of CPs and CMPs related together through the Start, Request and Deliver relations, but not with other clusters. The RPA clusters were proven to be suitable candidate services that satisfy SOA principles.

The method was deployed in [15] using the Cancer Care and Registration in Jordan case study [15, 16].

Figure 2 shows the Riva 2nd cut process architecture for the CCR case study, and Figure 3 shows the candidate services identified using the Riva based approach.

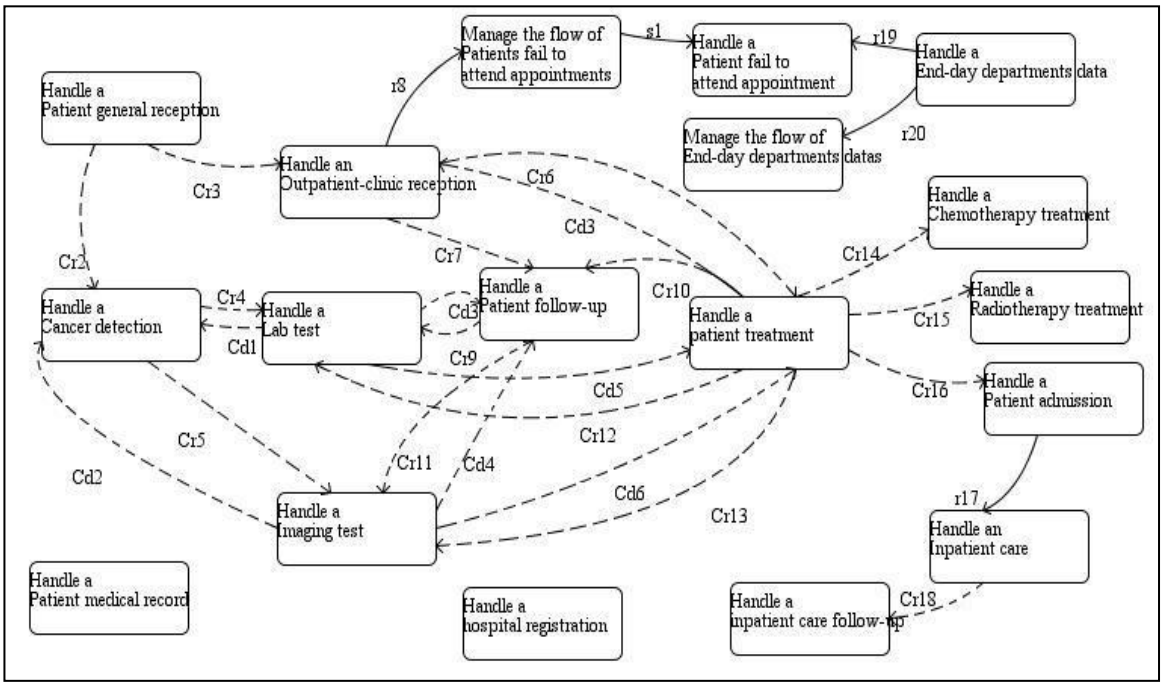

Figure 2. 2nd Cut BPA Diagram for the CCR Processes

The capabilities of each service can be derived from the process models associated with each case process and case management process. The entities are also tractable from the identified candidate services through the business process models [17]. The business process model for each case process and case management process can be found in [15].

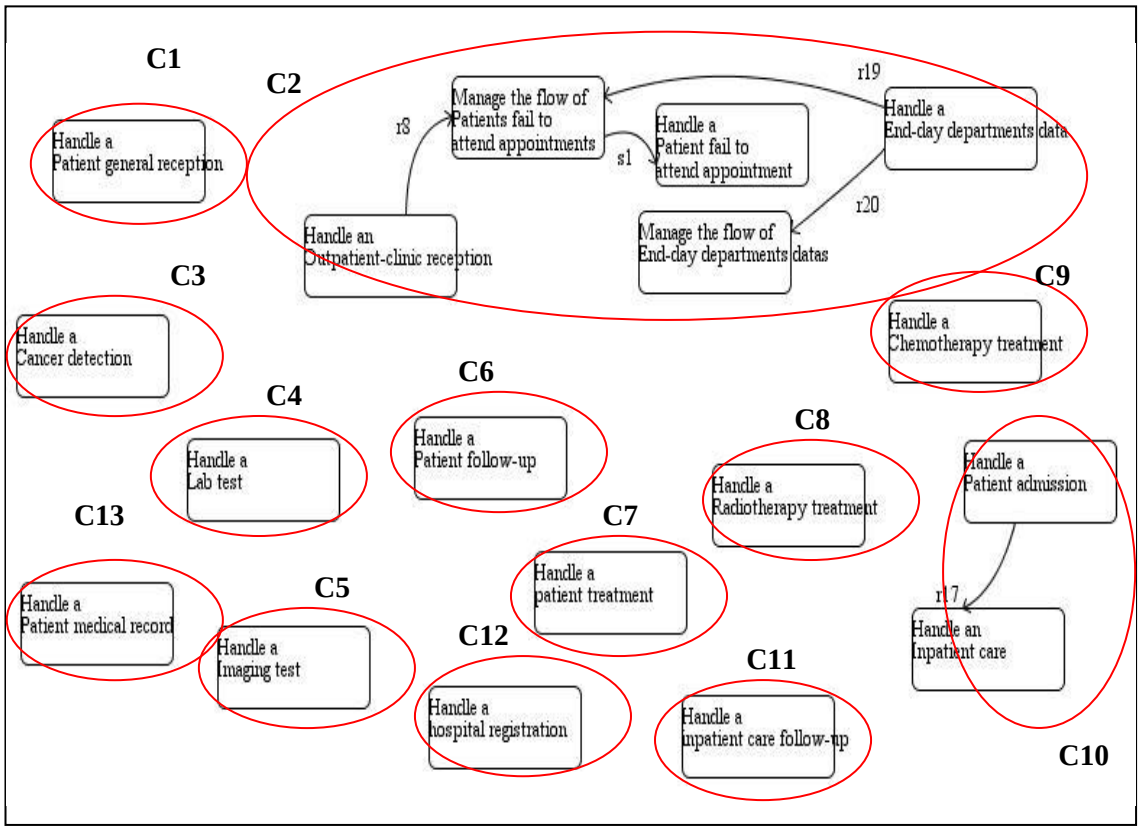

Figure 3. Candidate Services Identified using the Riva Method

Table 1 is the result of applying lines 1 - 4 of Algorithm I, where the first column in This table represents the candidate services identified using the selected approach, the second column represents the capabilities of each service which act as the functions required to build the Validating_CRUD matrix, the third column is the CRUD values which are entered as the matrix cell values and the last column represents the entities of relevant functions (capabilities). Also, we note in Table 1 that some capabilities act as points of interaction with other services, and hence they are not considered as performing any of the CRUD functionalities on the corresponding entities; they rather provide information of service relationships.

Figure 4 shows the result of applying the lines 5 – 13 of Algorithm I. A Validating_CRUD matrix is built using the information provided in the previous step. As can be seen from this figure, services are bounded in boldface borders in column A. Also, for each service, the cluster containing the capabilities and entities are bounded with a dashed boldface border (line 14 of Algorithm I).

Lines 15-21 of Algorithm I requires that each service cluster is to be tested to ensure that it identifies groups of functions with the right granularity to satisfy low coupling and high cohesion according to Jamshidi *et al.*, [5]. This is accomplished by ensuring that the entities of the cluster are created and updated only by the functions within the same cluster. This is because the EEAT technique groups together all functions that create and update the same entities to define candidate services.

Accordingly, the validation technique results in a confirmation that all of the identified services are candidate service, as no overlapping exists between service clusters and each service cluster creates and updates the same entities.

Table 1. Identified Candidate Services and the CRUD Values of its Capabilities

Candidate Software Service	***Capabilities (Functionality)***	***CRUD Values***	***Entities***
Cancer Detection	Book appointment	C	Appointment for cancer detection
	Check if patient is medically insured	R	Medical Record
	Check if patient in Data Base (DB)	R	DB
	Register patient details	Point of interaction to anther service	
	Order test	Point of interaction to anther service	
	Update patients file	Point of interaction to anther service	
	Book appointment for patient for an imaging test	Point of interaction to anther service	
	Refer patient to special combined clinic	Point of interaction to anther service	
Lab Test	*Check if patient is medically insured*	*R*	*Medical Record*
	Inform patient to visit doctor	C	Letter to check lab tests
	Add test results	Point of interaction to anther service	
Imaging Test	Book appointment for patient	C	Appointment for imaging test
	Check patient's appointment	R	Appointment for imaging test
	Check if patient is medically insured	R	Medical records
	Add and report results	Point of interaction to anther service	
	Inform patient to visit doctor	C	Letter to check imaging tests
Patient Follow-up	*Request admission from admission clerk*	*Point of interaction to another services*	
	Order test	Point of interaction to another services	
	Book appointment imaging department	Point of interaction to another services	
	Request another appointment	C	Appointment for follow-up

Table 1. (Cont'd): Identified Candidate Services and the CRUD Values of its Capabilities

Candidate Software Service	*Capabilities (Functionality)*	*CRUD Values*	*Entities*
	Send advises and instructions to patient	C	Advises and instructions letter
	Update patient file	Point of interaction to another services	
Patient Treatment	Review patient's history and investigations	R	Medical record
	Request admission	Point of interaction to another services	
	Order tests	Point of interaction to another services	
	Book appointment imaging department	Point of interaction to another services	
	Device plan for treatment	C	Plan for treatment
	Inform patient to visit radio	C	Letter to visit radio
	Book appointment for radiotherapy treatment	Point of interaction to another services	
	Inform patient to visit chemotherapy department	*C*	*Letter to visit chemo*
	Book appointment for chemotherapy treatment	*Point of interaction to another services*	
Radiotherapy Treatment	Check if patient has appointment	R	Medical record
	Check if patient is medically insured	R	Medical record
	Transfer patient	Point of interaction to another service	
	Add results	Point of interaction to another service	
Chemotherapy Treatment	Receive request for appointment booking	C	Chemotherapy appointment
	Check if patient has appointment	R	Chemotherapy appointment
	Check if patient is medically insured	R	Medical record
	Add results	Point of interaction to another service	
Patient Admission and Care	Check room availability	R	rooms
	Add patient to waiting list	U	Admission file
	complete paper work	U	Admission file
	Open admission file	C	Admission file
	Add notes to file	U	Admission file
	Update patient file	Point of interaction to another service	

Table 1. (Cont'd): Identified Candidate Services and the CRUD Values of its Capabilities

Candidate Software Service	*Capabilities (Functionality)*	*CRUD Values*	*Entities*
Inpatient Care Follow-up	Specialists review resident doctors' orders, diagnose patients and review old tests	R	Medical record
	Check patient financial state	R	Patient financial state
	Specialist makes appointment in outpatient clinic with patient	C	Outpatient clinic appointment
	Approve patient financial state	R	Patient financial state
Hospital Registration	Extract main details about cancer patients	R	Details about cancer patients
	Check if there is any contradictable data	R	Details about cancer patients
	Check if patient exist in DB	R	DB
	Add patient's details to DB	U	DB
	Check if primary tumor exist in DB	R	DB
	Add primary tumor	C	Tumor record
	Add any additional information	U	Tumor record
	Generate reports about cancer incidents in the hospital	C	Reports
	Add required details in JCR form	U	Reports
	Make copy of pathology reports and death certificate	R	Reports
Patient Medical Record	Find patient's file	R	Medical record
	Register file's details	U	Medical record
	Send patient's file	Point of interaction to other services	
	Open file	C	Medical record
	Save patient's file in library	U	Medical record

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T
1	Functions \ Entites	Appointment for cancer detection	Medical Record	Letter to check lab tests	Appointment for imaging test	Letter to check imaging tests	Appointment for follow-up	Advices and instructions letter	Plan for treatment	Letter to visit radio	Letter to visit chemo	Chemotherapy appointment	rooms	admission file	Patient financial state	Outpatient clinic appointment	Details about cancer patients	DB	Tumor record	Reports
2	Book appointment	C	R																	
3	Check if patient is medically insured		R																	
4	Check if patient in DB																	R		
5	Register patient details																			
6	Order test																			
7	Update patients file																			
8	Book appointment for patient for an imaging test																			
9	Refer patient to special combined clinic																			
10	Check if patient medically insured		R																	
11	Inform patient to visit doctor			C																
12	Add test results																			
13	Book appointment for patient				C															
14	Check patient's appointment				R															
15	Check if patient is medically insured		R																	
16	Add and report results																			
17	Inform patient to visit doctor					C														
18	Request admission from admission clerk																			
19	Order test																			
20	Book appointment imaging department																			
21	Request another appointment						C													
22	Send advices and instructions to patient							C												
23	Update patient file																			
24	Review patient's history and investigations		R																	
25	Request admission																			
26	Order tests																			
27	Book appointment imaging department																			
28	Device plan for treatment								C											
29	Inform patient to visit radio									C										
30	Book appointment for radiotherapy treatment																			
31	Inform patient to visit chemotherapy department										C									
32	Book appointment for chemotherapy treatment																			
33	Check if patient has appointment		R																	
34	Check if patient is medically insured		R																	
35	Transfer patient																			
36	Add results																			
37	Receive request for appointment booking											C								
38	Check if patient has appointment											R								
39	Check if patient is medically insured		R																	
40	Add results																			
41	Check room availability												R							
42	Add patient to waiting list													U						
43	complete paper work													U						
44	Open admission file													C						
45	Add notes to file													C						
46	Update patient file																			

Figure 4. The Validating_CRUD Matrix

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T
1	Functions / Entites	Appointment for cancer detection	Medical Record	Letter to check lab tests	Appointment for imaging test	Letter to check imaging tests	Appointment for follow-up	Advises and instructions letter	Plan for treatment	Letter to visit radio	Letter to visit chemo	Chemotherapy appointment	rooms	admission file	Patient financial state	Outpotient clinic appointment	Details about cancer patients	DB	Tumor record	Reports
47	Specialists review resident doctors' orders, diagnose		R																	
48	Check patient financial state														R					
49	Specialist makes appointment in outpatient clinic with patient															C				
50	Approve patient financial state														R					
51	Extract main details about cancer patients																R			
52	Check if there is any contradictable data																R			
53	Check if patient exist in DB																	R		
54	Add patient's details to DB																	U		
55	Check if primary tumor exist in DB																	R		
56	Add primary tumor																		C	
57	Add any additional information																		U	
58	Generate reports about cancer incidents in the hospital																			C
59	Add required details in JCR form																			U
60	Make copy of pathology reports and death certificate																			R
61	Find patient's file		R																	
62	Register file's details		U																	
63	Send patient's file																			
64	Open file		C																	
65	Save patient's file in library		U																	

Figure 4. (Cont'd): The Validating_CRUD Matrix

5. Conclusion and Future Work

In this paper a dynamic validation technique was presented based on the service identification criterion defined by Jamshidi *et. al.*, [5]. A reversed affinity analysis technique was used to evaluate the identified candidate services by testing that the functionality of the identified services creates and updates the same entities within the same group of capabilities. Accordingly, the presented validation technique is capable of evaluating the cohesion and low coupling of the identified services.

The validation technique was tested using the BPA-based SI approach described in [12] after deploying it on a real case study from the healthcare domain. The Results of validation were consistent with those of the SI approach.

One limitation of the presented validation technique is that it is not applicable for bottom-up SI approaches, where it could be difficult to identify functions and entities to build the Validating-CRUD matrix.

As a future work, we are trying to automate the proposed technique to provide a validation tool that inputs the identified services and produces the validating-CRUD matrix. We are also working in including all types of SI approaches to be applicable for our validation technique.

References

[1] Arsanjani, "Service-oriented modeling and architecture: How to identify, specify and realize services for your soa", Technical report, IBM, Software Group, **(2004)**.
[2] R. Huergo, P. Pires and F. Delicato, "MDCSIM: A method and a tool to identify services", IT Convergence Practice (INPRA), vol. 2, no. 4, **(2015)**, pp. 1-27.
[3] D. Bianchini, C. Cappiello, V. De Antonellis and B. Pernic. "Service Identification in Interorganizational Process Design", IEEE Transactions on Services Computing, vol. 7, no. 2, **(2014)**, pp. 265-278.
[4] Q. Gu and P. Lago, "Service Identification Methods: A Systematic Literature Review", Proc. ServiceWave, **(2010)**, pp. 37-50.
[5] P. Jamshidi, M. Sharifi and S. Mansour, "To Establish Enterprise Service Model from Enterprise Business Model", IEEE SCC, vol. 1, **(2008)**, pp. 93-100.

[6] F. Kohlmann and R. Alt, "Business-driven service modeling - a methodological approach from the finance industry", Int. Working Conference on Business Process and Services Computing, **(2007)**.
[7] L. Aversano, Cerulo and C. Palumbo, "Mining candidate web services from legacy code", Int. Symposium on Web Site Evolution, **(2008)**, pp. 37-40.
[8] R. Yousef, M. Odeh, D. Coward and A. Sharieh, "BPAOntoSOA: A generic framework to derive software service oriented models from business process architectures", ICADIWT, **(2009)**, pp. 50–55.
[9] L. G. Azevedo, F. Santoro, F. Baiao J. Souza, K. Revoredo, V. Pereira and I. Herlain, "A Method for Service Identification from Business Process Models in a SOA Approach", Enterprise, Business-Process and Information Systems Modeling, **(2009)**, pp. 99-112.
[10] Z. Yun, S. Huayou, N. Yulin and Q. Hengnian. A service-oriented analysis and design approach based on data flow diagram", Int. Conference on Computational Intelligence and Software Engineering, **(2009)**, pp. 1–5.
[11] M. J. Cho, H. R. Choi, H. S. Kim, S. G. Hong, Y. Keceli and J. Y. Park, "Service identification and modeling for service oriented architecture applications", Int. Conference on Software Engineering, Parallel and Distributed Systems, World Scientific and Engineering Academy and Society, **(2008)**, pp. 193-199.
[12] R. Yousef and M. Odeh, "A BPA-Based Approach for the Identification of SOA Candidate Software Services", The 5th International Conference on Information Technology, ICIT'11. Jordan, **(2011)**.
[13] M. Ould, "Business Process Management: A rigorous Approach", BCS, UK, **(2005)**.
[14] S. Green and M. Ould, "The Primacy of Process Architecture", 5th Workshop on Business Process Modelling, Development, and Support, in conjunction with the 16th Conference on Advanced Information Systems Engineering (CAiSE04), Riga, **(2004)**.
[15] R. Yousef, "BPAOntoSOA: A Semantically Enriched Framework for Deriving SOA Candidate Software Services from Riva-based Business Process Architecture", PhD thesis, The University of the West of England, **(2010)**.
[16] F. AbuRub, "Process Modelling for Business Process Improvement, with particular reference to healthcare processes", PhD thesis, The University of the West of England, **(2006)**.
[17] R Yousef, "Extracting Essential Business Entities from BPMN Models", International Journal of Computer Science Issues, vol. 11, Issue 4, no. 1, **(2014)**, pp. 131-134.

