



Towards a Collaborative, Interactive Web Services Composition Approach Based on an Intentional Group Recommender System

Meriem Kasmi¹(✉), Yassine Jamoussi^{1,2},
and Henda Hajjami Ben Ghézala¹

¹ Riadi Lab, ENSI, University of Manouba, Manouba, Tunisia
{meriem.kasmi, yassine.jamoussi,
henda.benghezala}@ensi.rnu.tn

² Department of Computer Science, College of Science,
Sultan Qaboos University, 36 Al-Khoudh 123, Muscat, Oman

Abstract. Nowadays, organizations are integrating human-computer interaction (HCI) into their information systems. This trend resulted in gathering developers and end-users in different tasks, particularly, the interactive Web services composition (WSC) task. Indeed, to search for solutions that go beyond their individual limited views, an increasing demand of collaboration among users has emerged. However, they are still facing uncomfortable situations especially when they are invited to select the appropriate Web service among functionally similar ones. More support is then needed to provide an effective composition. In this regard, a group recommender system providing the required functionality while considering the users' preferences, might be highly useful. In this paper, we present a holistic process spanning from capturing users requirements, constructing a global goal model "ColMAP" reflecting their intentions to performing a collaborative, interactive WSC. A step-by-step example illustrates the proposed process. We expect that this approach will pave the way for interactively, collaboratively engineered information systems.

Keywords: Interactive Web services composition · Collaboration
Group recommendation · MAP formalism

1 Introduction and Motivation

With the focus on improving the interaction between humans and computers, the human-computer interaction (HCI) has become one of the most thriving fields in nowadays' computing era. It has become even an integral part of information systems with the aim of helping their users (e.g. developers, project managers, customers, partners, or investors) to be more productive [23]. In fact, HCI growth has brought an unprecedented momentum to information systems by giving non-professional users the opportunity to interact with the different components. This interaction takes place on the basis of a service-oriented architecture (SOA) since it is considered as a reference architecture of information systems. SOA promises to achieve high agility of businesses as well as rich interoperability between distributed and heterogeneous applications.

Thus, different actors have henceforth the luxury of interacting with, information systems' components, mainly Web services by composing them in an interactive way. This process is called the interactive Web services composition (WSC) and builds upon an appealing investigation of approaches from HCI and SOA [4].

Unlike the manual WSC which is restricted to experts, the interactive WSC that belongs also to the end-user programming approaches, involves tech-novice users which lowers the cognitive barrier on all types of users [27]. But, one can notice that even an automatic composition is quite suitable for users with no technical background. Such concern is entirely appropriate. Still, it is also clear that full automation would miss the chance for users' to be productive and to break new ground by fine-tuning services to their needs [6]. Overall, the interactive WSC combines the advantages of manual and automatic modes and resolves their particular issues.

Yet, to search for solutions that go beyond their individual limited views, an increasing demand of collaboration among users has emerged. Consequently, these latter often operate in groups to accomplish tasks which are not feasible individually because of lack of expertise or time. Collaboration is then a necessity in organizations, as no one can possess all the required competences. Our research stresses then the interactive WSC performed in collaborative situations. Thereby, the involved stakeholders will be able to explore the differences constructively and perform an effective composition.

However, because of the abundance of available Web services, users are facing various uncomfortable situations even when collaborating with each other. To February 2018, the largest Web services repository, ProgrammableWeb.com¹, has registered 19,163 Web services. Within this wealth, many are functionally equivalent. This hampers the composition process and leads to users' frustration as they find it notoriously difficult to select the appropriate Web service. More support is then required to manage such an abundance of choices, as well as to guide users with no technical background to perform the most relevant alternative. Traditional Web service discovery approaches tried to solve this problem by relying on UDDI (Universal Description, Discovery and Integration) registries. Unfortunately, UDDI is no longer the choice, since the shutdown of public UDDI registries held by IBM, Microsoft, and SAP [15]. Consequently, Web service discovery approaches have moved toward using Web services search engines and public Web services repositories such as ProgrammableWeb.com. These alternatives turned out to be also ineffective as they are heavily dependent on correct queries from users [25]. Moreover, personalization in terms of mapping content to users' interests and preferences is absent. This raised new challenges for information systems engineers to find a solution that handles the problem of information overload, filters and efficiently delivers the most relevant Web services. It would be even desirable if this solution recommends Web services that align with users' interests without requiring the users to explicitly specify queries. In this regard, a group recommender system (GRS) providing the required functionality while considering the users' preferences, might be highly useful. In fact, recommender systems have become a widely adopted means to tackle the problem of information

¹ <https://www.programmableweb.com/>.

overload, e.g. products on Amazon [9] and movies on Netflix [10]. They were primarily developed to support individuals, but, with the growing need of group work, more elaborate recommender systems targeting groups came into the picture under the name of GRSs. Today, GRSs are used in real-world applications as well as in different research fields. To the best of our knowledge, there is no existing work in the literature which has exploited the opportunity of joining the group recommendation with the interactive WSC. While it is true that the work of Rong et al. [24] has exploited the idea of user group in the context of WSC, it has not provided group recommendations. It has rather identified similar users to apply the association rule mining technique in a “*collaborative-filtering fashion*” in order to rank the discovered Web services.

Hence, given the highlighted need for collaboration in interactive WSC and inspired by the high relevance of recommender systems, we intend in this paper to embrace the benefits of group recommendation in a collaborative, interactive WSC performed by information systems’ users. To address this challenge, we propose a new collaborative, interactive WSC approach based on an intentional GRS. We believe that this solution will create new opportunities to develop smart and personalized information systems. The suggested GRS capitalizes on a synergy between the functional needs deduced from a global intentional model “ColMAP” that we introduce and build in this paper, as well as the stakeholders’ preferences in order to discover relevant Web services. Our approach reflects a holistic process spanning from capturing users requirements, constructing a global goal-oriented model reflecting their intentions to performing a collaborative, interactive WSC assisted by a GRS.

The contributions of this paper include:

- (i) The consideration of multiple stakeholders and capturing their goals from their queries;
- (ii) The construction of a global goal-oriented model ColMAP aggregating individual retrieved goals and based on the MAP formalism [2];
- (iii) The automation of the Web services discovery step by a GRS in order to find the most relevant Web services;

The next section presents foundations for the WSC modeling adopted for this work and reviews existing modeling approaches. In Sect. 3, we expose our approach with an overview of its general phases then we detail these steps with a running example. Finally, we present conclusions and directions for future work in Sect. 4.

2 Web Services Composition Modeling

Diverse approaches to modeling WSC have been suggested in the literature. The work done in [12] provides a classification of these approaches into four main groups: industrial approaches (e.g., ones using WSBPEL² or BPEL for short), semantic approaches based on ontologies notably OWL-S³ to describe Web services,

² https://www.oasis-open.org/committees/tc_home.php?wg_abbrev=wsbpel.

³ <http://www.ai.sri.com/daml/services/owl-s/>.

quality-oriented approaches focusing on improving the quality of services and formal approaches using mathematic formalisms and models. These formal approaches include operational approaches such as Petri nets and intentional or goal-oriented approaches which are based on a representation of the WSC according to the service consumers' intentions. The first three groups focus on expressing Web services in a low-level manner through technical details (e.g., input/output parameters). Obviously, these latter are far from being understandable by end-users which represent an important category of actors in information systems. Moreover, in the analysis of complex organizations, it is more relevant to study first goals instead of functions. Business goals will be then realized by business processes which in turn will be implemented in the information system. Modeling business goals constitutes thus, a central activity of the business process as well as of information systems. Such recognition has led to a whole stream of research called Goal-Oriented Requirements Engineering (GORE). It is a subarea of Requirement Engineering, addressing the use of goals for requirements elicitation, specification, verification/validation and negotiation [11].

For these reasons, a high-level modeling of the WSC centered on goals appears to be the most suitable to our context. In this paper, we adopt an intentional modeling of WSC. Given that the intended WSC is performed in a collaborative way, we tried to find a collaborative intentional model to represent it. However much we searched in literature, we did not find any goal model that copes with our work. At the first glance, the work elaborated in [17] seemed to be suitable for modeling our collaborative composition in an intentional level. This work introduces a collaborative intentional model called CSRML (Collaborative Systems Requirements Modelling Language). It is in fact an extension of the i^* goal model [18] and aims at dealing with collaborative systems in which the awareness of the presence of other users as well as their actions are crucial. But, a closer look has revealed that we cannot choose such a model. In fact, collaborative plans evolve, and systems must be able to cope with any evolution e.g., involvement of a new actor, exclusion of an existing actor, change in actor's business process, etc. In CSRML, supporting such an evolutive, dynamic aspect is absent and can be done only manually by rebuilding the whole model.

Consequently, in this work, we argue for the idea of modeling individual intentions with a relevant goal model then aggregating these individual models into a collaborative one. In this way, any change that might occur can be easily embodied in individual model. Regarding the collaborative model, it will be automatically generated again by aggregation. Various goal models have been provided by the GORE community. The most popular ones are i^* [18], TROPOS [19], KAOS [20], the Goal-oriented Requirement Language (GRL) [21] and the MAP formalism [2]. These models share interesting features such as expressiveness and understandability but regarding the variability support, MAP outperforms all other goal models [26]. In fact, MAP gives a multitude of paths between two intentions. Each path corresponds to a strategy to achieve the target goal from the source one. This explicit representation of variability offered by MAP is missing in other goal models. MAP has been also successfully used in previous works on modeling Web services-based applications [13, 14]. We give in the following an overview of the MAP formalism, on which our intentional global model, ColMAP, is based.

2.1 MAP Overview

MAP was proposed by Rolland et al. [2]. Its main objective is to describe processes in an intentional level. Graphically, it is represented as a labeled directed graph. The nodes correspond to intentions or goals and edges are labeled with strategies. A strategy refers to the manner a target goal is achieved from the source one. This explains why this graph should and labeled with the name of the strategy. MAP offers the possibility to follow different strategies to achieve intentions. Figure 1 shows the metamodel of MAP. As depicted in Fig. 1, the fundamental elements of MAP are Intentions, Strategies and Sections. Each section is a triplet $\langle I_i, I_j, S_{ij} \rangle$ formed by one source intention I_i , one target intention I_j and only one edge S_{ij} linking them. A MAP model is then based on a multitude of sections. Three types of relationships exist between sections namely bundle, multi-thread and path. Figure 2 illustrates these three relationships.

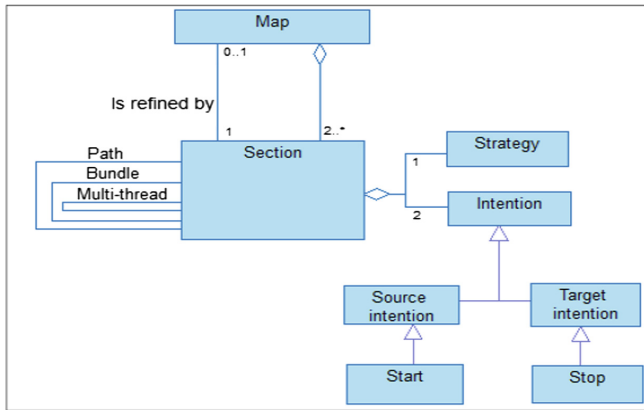


Fig. 1. The MAP metamodel

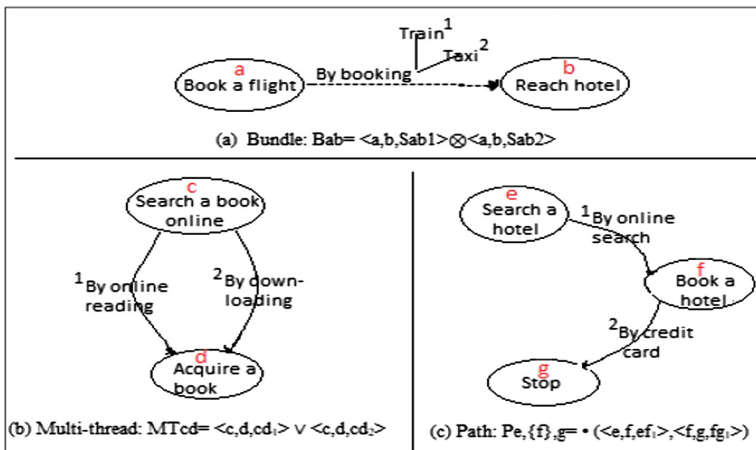


Fig. 2. Relationships between sections in MAP

In both bundle and multi-thread relationships, sections share the same source and target intentions. But in a bundle, only one strategy can be used to accomplish the target intention. Sections are then mutually exclusive in a bundle related by the logic function XOR using the operator « \otimes ». Graphically, a bundle is evinced by a dotted arrow. As for a multi-thread relationship, the target intention can be achieved from a source in many different ways. So, sections in a multi-thread can be executed simultaneously. They are related by the logic function AND/OR and represented by the operator « \vee ». Finally, a path relationship relies on a precedence/succession relationship. In this relationship, a target intention in a section, is a source one in another. It is represented by the operator “ \cdot ”. We note that it is possible to refine a section at level i into another MAP model at level $i+1$. The MAP model of level $i+1$ corresponds to a complex assembly of sections viewed as a unique section at level i .

In this paper, we adopt an intentional WSC modeling based on MAP. We specify each section within the MAP model by an Intentional Service Model (ISM) detailed in the following.

2.2 ISM Overview

Figure 3 was suggested by Rolland in [5] to abstract software Web services into intentional services. Since then, the ISM has been highly investigated. An intentional service corresponds to the intentional descriptors derived from users’ queries. ISM focuses then on the intention to accomplish rather than on the functionality to perform. As shown in Fig. 3, there are three different aspects to consider in the description of an intentional service, namely the intentional service interface, behavior and composition. First aspect reflects that an intentional service satisfies an intention, given an initial situation and terminating in a final situation. Intention corresponds to the intentional service id, whereas Initial Situation and Final Situation are the IN/OUT parameters. Second, the behavioral aspect is specified through pre and post conditions. They are initial and final states respectively describing the required state to start accomplishing an intentional and the resulting state after achieving it. Finally, the composition part defines intentional services as aggregate or atomic services. An Atomic service corresponds to an operationalized section i.e. it cannot be refined. An Aggregate service is associated with a high level section that should be refined to the lowest possible level.

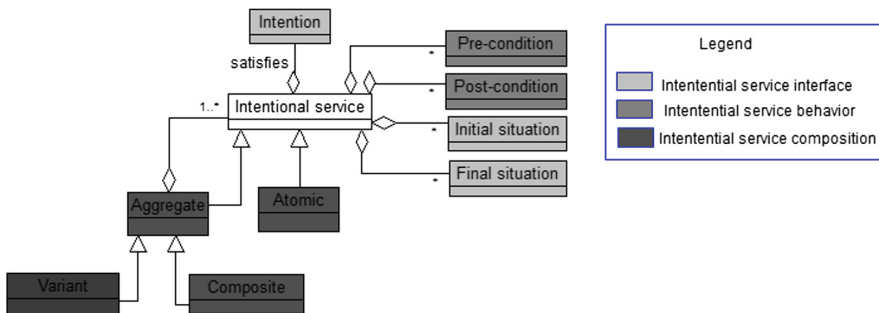


Fig. 3. The intentional service model

Aggregate services can be either Composite or Variant. Composite services can be generated according to a sequential, parallel, or iterative arrangement. As for Variant services, they express variability with alternatives and multi-choice.

3 The Proposed Approach

3.1 Approach Overview

In this paper, we propose a new interactive, collaborative WSC approach based on an intentional group recommender system. An overview of the proposed approach is shown in Fig. 4. It covers the four common WSC phases: the Planning phase, the Service Discovery phase, the Service Selection phase, and the Execution phase [8].

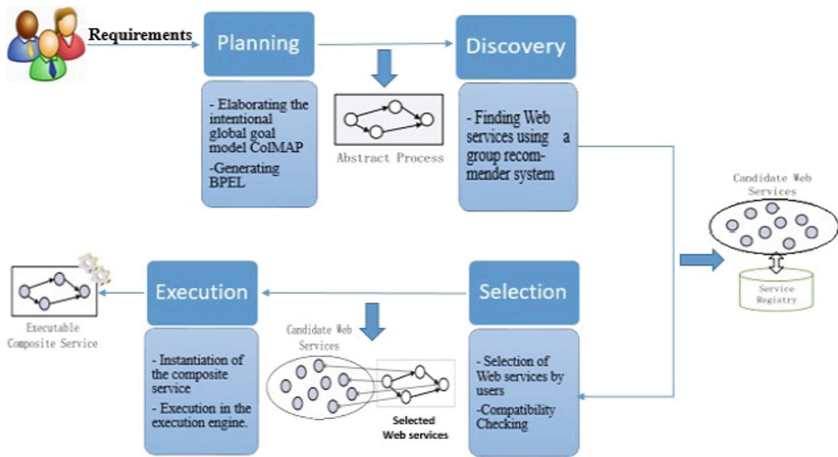


Fig. 4. Overview of the proposed approach

The planning phase determines the execution order of activities. In our approach, we opted for an intentional modeling of the WSC. Thus, to determine and coordinate activities, planning starts with a goal-oriented analysis of the stakeholders' requirements. This results in a set of partial individual intentions/strategies graphs for each stakeholder using the MAP formalism. Intentional services are then identified and specified for each section according to the ISM. Next, individual graphs are composed in a global goal model "ColMAP". Finally, a transformational operation from the intentional level to the operational level is carried out through the technique of model transformation. A BPEL abstract process model is produced.

The discovery phase finds the candidate services for each activity. In other words, it finds concrete components (Web services) that match the activities of the abstract process produced by the previous phase. Suitable Web services are located by searching the service registry. In our approach, this phase will be carried out by a GRS that considers two dimensions. On the one hand, it recommends Web services that match users' intentions derived from the intentional specification provided by ISM.

That is why, we call this GRS, an intentional GRS. On the other hand, this GRS recommends Web services according to each group preferences. Thus, recommended Web services will not only match their goals but also their interests inferred from their profiles which include explicit and implicit feedback (i.e. the provided ratings as well as the Web services invocation rates). It is noteworthy that GRSs have never been exploited in the SOC field.

During the third phase, users are invited to choose one Web service for each activity from the set of the recommended Web services. These latter will be provided in a clear and organized view to enable users to easily pick relevant Web services. The compatibility between selected Web services will be checked by a module for compatibility checking.

Finally, after all the required Web services are identified and bound to the corresponding activities, the composite service instance will be created and executed by the execution engine.

3.2 Detailed Process

We present details of the intended composition process spanning from capturing users requirements, constructing a global goal-oriented model leveraging all users' intentions to performing a collaborative, interactive WSC. The proposed process is illustrated with a step-by-step example from a clinical information system. In fact, with the move towards computerization, concepts such as "paperless hospitals" and "digital hospitals" are becoming widespread. Consequently, in hospitals, departments are increasingly using IT tools. We focus on the emergency department and give an example of a simplified patient care process in this department. Before, considerable time was wasted transmitting documents, namely patients' records from one stakeholder to another. In addition, some activities are distributed between departments which raised a data heterogeneity problem. Moreover, patients' records are written collaboratively by different stakeholders. It would be therefore useful to set up a collaborative solution in such a department. Applying our approach in such an environment is then quite appropriate.

Step 1. Pre-generating ColMAP. Every software development process starts with the phase of requirements analysis. This step results in a software requirements specification including functional and non-functional requirements. We note that non-functional requirements are out of the scope of this paper and that we focus mainly on functional requirements which are nothing else than users' intentions. Thus, unclear users' queries will result in improperly translating their intentions. However, we cannot expect from non-expert users to be always able to clearly express their intentions. Hence, we intend to provide them with a convenient interface showing intentions ranged by domain-specific categories e.g. travel, education, healthcare, etc. After successfully logging, users select intentions and the associated partial MAP models will be loaded. We note that MAP models should be already drawn by domain experts using an appropriate tool such as MetaEdit⁴.

⁴ <https://www.metacase.com/mep/> .

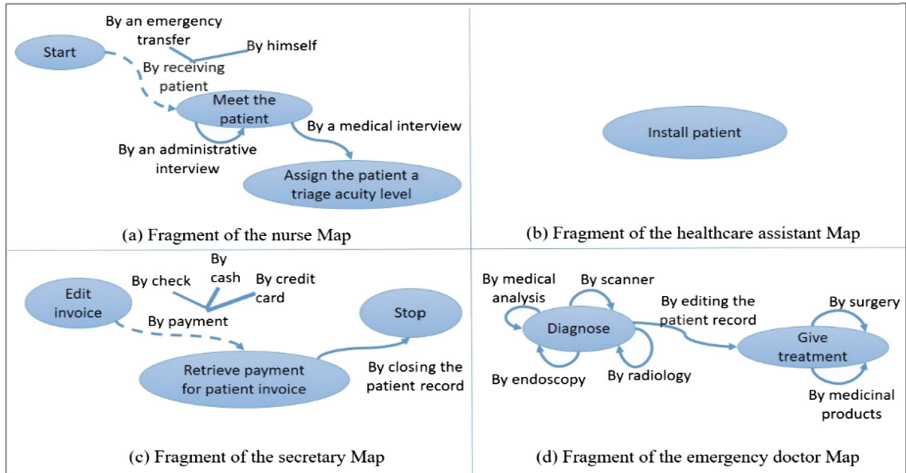


Fig. 5. Partial MAP models for a patient care scenario in a hospital emergency department

Figure 5 represents fragments of the partial MAP models of stakeholders involved in a scenario of patient care in an emergency department. This scenario gathers four roles: the Emergency doctor role, the nurse role, the secretary role and the assistant healthcare role. Once a patient arrives, a nurse will meet him/her to assess his/her triage acuity level. A healthcare assistant will then install him/her then the emergency doctor diagnoses and decides what treatment(s) to give. Finally, the patient will pay the cost for medical examinations he/she had.

Next, to elicit intentional services associated to each section of partial MAP models, we follow three rules [5]:

- R1. Associating every non-refined section to an atomic service. In the context of our example, we can associate 15 atomic services in total to the four individual MAP models of Fig. 5. For instance: The section <Edit invoice, Retrieve payment for patient invoice, By payment: By credit card> $\rightarrow S_{\text{pay: By credit card}}$
- R2. Identifying paths of the partial MAP models by applying an adaptation of the Mc Naughton and Yamada's algorithm [3].
- R3. Determining aggregate services by establishing the following guidelines:
 - Associate to sections in a bundle section an alternative intentional service. For example: $S_{\text{Meet the patient:}} \rightarrow S_{\text{Meet the patient:}} \text{ By himself } \otimes S_{\text{Meet the patient:}} \text{ By an emergency transfer}$
 - Associate to sections in a multi-thread relationship, a multi-choice intentional services.
 - Associate to sections in a path relationship, a sequential intentional service.

Step 2. Generating ColMAP. We define ColMAP as an extension of the MAP model in order to support collaboration. It is generated by composing partial MAP models through their potential dependencies. In the following, we give insight about the dependency concept and detail the ColMAP generation task.

Definition 1 MAP Models Dependency. Two MAP models are dependent if it exists at least one intention from the first which is dependent on one intention of the other.

Definition 2 Intentions Dependency. Let a section $\langle x, i, S_{xi} \rangle$ associated to an intentional service $Serv_{xi}$ from a MAP model $m1$ and a section $\langle j, y, S_{jy} \rangle$ associated to an intentional service $Serv_{jy}$ from a MAP model $m2$. i and j are dependent if it exists at least one intentional service $Serv_{ij}$ where: (i) $Serv_{ij}.Pre\text{-}condition.value = Serv_{xi}.Post\text{-}condition.value$ and (ii) $Serv_{ij}.Post\text{-}condition.value = Serv_{jy}.Pre\text{-}condition.value$. In a MAP model, all intentions forming a section are dependent.

The ColMAP generation task starts by determining dependencies between two MAP models then arranges these dependencies.

Dependencies Determination. This task takes as input two partial MAP models as well as their elicited intentional descriptors (intentional services). These descriptors are contained in .xmi files obtained by instantiating the ISM using Kermeta⁵. For instance, the .xmi file of the intentional service S_{pay} : By credit card is defined as:

```
<?xml version="1.0" encoding="ASCII"?>
<ism:Map xmi:version="2.0"
xmlns:xmi="http://www.omg.org/XMI"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:ism="http://ism/1.0"
xsi:schemaLocation="http://ism/1.0 ISM.ecore">
<service xsi:type="ism:Atomic"
intention="//@Intention.0"
pre_condition="//@Pre-Condition.0"
post_condition="//@Post-Condition.0"
initial_situation="//@Initial Situation.0"
final_situation="//@Final Situation.0"
id="Pay: By credit card "/>
<Intention description="Payment by credit card"/>
<Pre-Condition value="Invoice.Edit=true"/>
<Post-Condition value="Payment.State=true"/>
<resource name="Invoice"/>
<resource name="Payment"/>
<resource name="Card"/>
<Initial Situation input="//@resource.0 //@resource.2"/>
<Final Situation output="//@resource.1"/>
</ism:Map>
```

Dependencies are determined by mining the .xmi files of the different intentional services of the two MAP models. If any intentions dependency is found then these two MAP models are considered dependent. If exactly one intentional service is found for

⁵ <https://marketplace.eclipse.org/content/kermeta>.

this dependency, then a section joining the two dependent intentions is formed. Otherwise, a dependencies arrangement task must be carried out.

Dependencies Arrangement Let n : the number of intentional services found for the dependency determination task between the intentions i and j . Thus, n sections must be formed between i and j . These sections can be: (i) All in a bundle relationship, (ii) All in a multi-thread relationship or (iii) some of them in a bundle and others in a multi-thread relationship. To resolve this, we argue for a naming convention regarding intentional services in a bundle relationship. These latter must have exactly the same words before the “:” symbol for their ids. In the context of our example, the id of $S_{\text{pay:By credit card}}$ must be “Pay:ByCreditCard” and the id of $S_{\text{pay:By cash}}$ must be “Pay:ByCash”.

Thus, the intentional services for which this convention is validated are arranged in a bundle relationship while others join a multi-thread relationship. Figure 6 represents the generated ColMAP associated to the studied example.

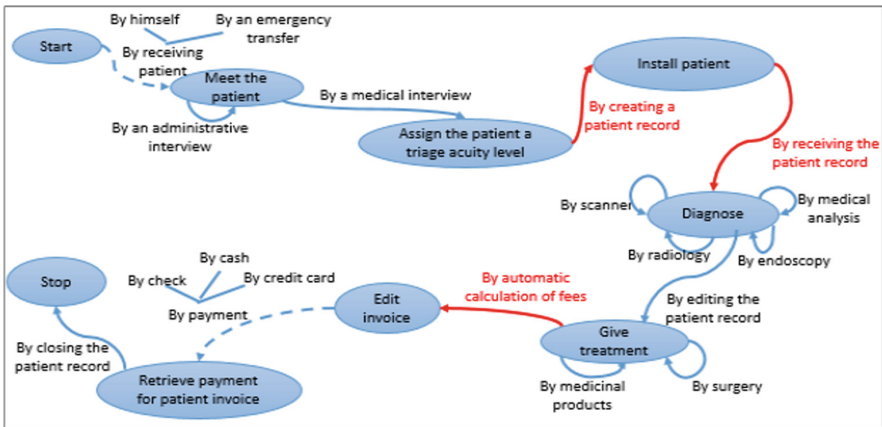


Fig. 6. The generated ColMAP model

Step 3. Generating BPEL. BPEL is a reliable standard in modeling and executing business processes. It is the most used language for orchestration in industry. With the advent of SOA, services are henceforth exploited to implement process tasks. Therefore, this justifies our choice for BPEL. Inspired by [7], we opt for a model transformation from the ISM to BPEL following the transformation rules reported in Table 1. The generated BPEL model is not executable. It is called an abstract BPEL model.

Step 4. Discovering Web Services Now that the planning phase is achieved, the Web services that operationalize intentional services must be discovered. However, a variety of potentially interesting Web services can match the required functionality. Therefore, the discovered set of Web services must be reduced. For this reason, we designed an intentional GRS. In fact, GRSs have proved to deal effectively with a large datasets and to retrieve useful items that are estimated to be collectively convenient for groups of users.

Table 1. Transformation rules from ISM to BPEL.

ISM Element	BPEL element
Atomic service	Invoke
Initial situation	Receive
Final situation	Reply
Pre-condition; Post-condition	Assign
Sequential composition	Sequence
Parallel composition	Flow
Iterative composition	while
Alternative variation	Switch, if, else
Multi-choice variation	Pick

To start, the system queries the Web services registry using keywords extracted from the .xmi files of the intentional services. These keywords are determined using the TF/IDF metric. Readers may refer to [1] for further details about this metric. TF/IDF allows us to filter out stop and low frequency words. Thereby, only meaningful words (i.e., keywords) having a high TF/IDF weight are maintained. As a result, a set of Web services will be returned. It is likely that a large number of candidate services will meet the requirements. Therefore, the GRS intervenes to reduce this number. It operates a two-level filtration.

In the first level, based on a semantic matchmaking between the .xmi files of the intentional services and the WSDL files of the returned Web services. This first filtration level generates a new reduced set to filter in a second filtration level according to users' preferences. To do so, the recommendation strategy as well as the approach to generating recommendations must be specified.

Current GRSs mainly falls into two categories based on the way recommendations are generated. The first category consists in aggregating individual models into group models. It merges individual user models into a group model and then generates recommendations using the aggregated group model. The second one aggregates individual recommendations already generated user, into group recommendation. Motivated by results of Berkovsky and Freyne [16] about comparing these two methods, we opt for the individual models aggregation method in this work. Individual models refer to rating matrices containing the users' ratings to the Web services they used. Obviously, such matrices are sparse due to the huge number of Web services compared to users. Thus, for pre-processing, we need to conduct rating matrix completion in order to estimate the individuals' preferences on the unseen items. For this, we adopt for matrix completion [22]. Next, the preferences (ratings and predictions) are aggregated with a specific aggregation strategy such as the average. We note that the aggregation strategy is not defined yet in this work. We intend to choose it during an experimentation phase.

Recommendations will be generated for each group of users performing the same role. In our example, given that we have four roles, we will have naturally four groups including different numbers of individuals performing the same role. The proposed GRS relies on a hybrid recommendation approach combining collaborative filtering

and content-based recommendation. In the former, users are recommended items that like-minded users preferred in the past whilst in the latter, they are recommended items similar to the ones they liked before. We adopt a hybrid approach since it helps to avoid shortcomings of both content-based and collaborative approaches and incorporates the advantages of these two methods. Therefore, the final set of discovered Web services is produced after this filtration according to preferences.

Step 5. Selecting Web Services. In a context of an interactive WSC, this step is done by the users through a dedicated interface presenting possible alternatives (Web services) for each elementary goal. Users are then invited to pick one Web service per goal. However, the Web services incompatibility is amongst the common issues that may emerge in the design and implementation of WSC. The WSC process is a coordinated and collaborative service invocation task including several multi-service workflows. These interacting workflows can be constructed using various emerging standards to manage the Web services compatibility. If two services are not compatible, a warning message will be displayed informing users of the problem and suggesting other compatible service(s) for the predecessor service.

Step 6. Execution of the WSC. Once the Web services are fully selected, an executable BPEL model will implement at the abstract BPEL model with the selected services. It will be deployed in a composition engine and the composition will be executed. Finally, users will be invited to rate the composition process. This explicit feed-back helps to refine the future recommendations.

4 Conclusion

In this work, we explored how an interactive WSC can be effectively combined with a GRS in collaborative scenarios. This kind of composition is receiving a lot of attention in recent years, as it enables end-users to perform an active role in the composition process. As for group recommendation, it has never been exploited before for suggesting Web services. Therefore, we offer a unique approach for collaborative, interactive WSC where the composed Web services do not only satisfy the business needs but also the collective “taste” of the group. Moreover, in this paper, we set the goal of bridging the gap between intentional and operational level by generating a global goal model from individual MAP models and matching intentional services with their software equivalent ones.

We are currently working on a prototype to validate it. Afterwards, we intend to conduct a user study which compares in a qualitative and quantitative way how the proposed solution improves the service composition process. We are planning also to extend to support a conflict resolution process when users of the same group do not pick the same Web service. Thus, we are considering the use of a module that resolves this conflict and promotes more successful collaboration between stakeholders.

References

1. Salton, G., Buckley, C.: Term-weighting approaches in automatic text retrieval. *Inf. Process. Manag.* **24**(5), 513–523 (1988)
2. Rolland, C., Loucopoulos, P., Kavakli, V., Nurcan, S.: Intention based modelling of organisational change: an experience report. In: *Proceedings of the 4th CAISE/IFIP 8.1 International Workshop on Evaluation of Modeling Methods in Systems Analysis and Design*, Heidelberg, pp. 1–36 (1999)
3. McNaughton, R., Yamada, H.: Regular expressions and state graphs for automata. *IRE Trans. Electron. Comput.* **EC-9**(1), 39–47 (1960)
4. Liang, Q., Bharadwaj, A., Lee, B.S.: Interactive and iterative service-composition-based approach to flexible information system development. *Int. J. Web Serv. Res.* **8**(4), 81–107 (2011). <https://doi.org/10.4018/jwsr.2011100104>
5. Rolland, C., Kaabi, R-S.: An intentional perspective to service modeling and discovery. In: *Proceedings of the 31st International Computer Software and Applications Conference (COMPSAC)*, Beijing, pp. 455–460 (2007)
6. Mehandjiev, N., Lécué, F., Wajid, U., Namoun, A.: Assisted service composition for end users. In: *Proceedings of the 8th IEEE European Conference on Web Services (ECOWS)*, Ayia Napa, pp. 131–138 (2010)
7. Driss, M., Jamoussi, Y., Moha, N., Jézéquel, J.M., Ghézala, H.H.B.: Une approche centrée exigences pour la composition de services web. *Revue des Sciences et Technologies de l'Information-Série ISI: Ingénierie des Systèmes d'Information* **16**(2), 97–125 (2011)
8. Claro, D.B., Albers, P., Hao, J.-K.: Web services composition. In: Cardoso, J., Sheth, A. P. (eds.) *Semantic Web Services, Processes and Applications*, pp. 195–225. Springer, New York (2006)
9. Smith, B., Linden, G.: Two decades of recommender systems at Amazon.com. *IEEE Internet Comput.* **21**(3), 12–18 (2017)
10. Gomez-Uribe, C.A., Hunt, N.: The netflix recommender system: algorithms, business value, and innovation. *ACM Trans. Manag. Inf. Syst.* **6**(4), Article no. 13, 19 p. (2015). <https://doi.org/10.1145/2843948>
11. van Lamsweerde, A.: Goal-oriented requirements engineering: a guided tour. In: *Proceedings of the 5th International Symposium on Requirements Engineering (RE)*, pp. 249–263. IEEE, Toronto (2001)
12. Ben Messaoud, W., Ghedira, K., Ben Halima, Y., Ben Ghezala, H.: Survey of web service composition. In: *Proceedings of the 5th International Conference on Information & Communication Technology and Accessibility (ICTA)*, pp. 1–7 (2015)
13. Aljoumaa, K., Assar, S., Souveyet, C.: Publishing intentional services using extended semantic annotation. In: *Proceedings of the 5th International Conference on Research Challenges in Information Science (RCIS)*, pp. 1–9. IEEE, Gosier (2011)
14. Rolland, C., Kirsch-Pinheiro, M., Souveyet, C.: An intentional approach to service engineering. *IEEE Trans. Serv. Comput.* **3**(4), 292–305 (2010)
15. Krill, P.: Microsoft, IBM, SAP Discontinue UDDI Registry Effort (2005). <https://www.infoworld.com/article/2673442/application-development/microsoft-ibm-sap-discontinue-uddi-registry-effort.html>. Accessed 22 Feb 2018
16. Berkovsky, S., Freyne, J.: Group-based recipe recommendations: analysis of data aggregation strategies. In: *Proceedings of the 4th ACM conference on Recommender systems (RecSys)*, pp. 111–118. ACM, New York (2010). <https://doi.org/10.1145/1864708.1864732>

17. Teruel, M.A., Navarro, E., López-Jaquero, V., Montero, F.: CSRML: a goal-oriented approach to model requirements for collaborative systems. In: Jeusfeld, M., Delcambre, L., Ling, T.W. (eds.) *Conceptual Modeling (ER) 2011*. LNCS, vol. 6998, pp. 33–46. Springer, Heidelberg (2011)
18. Castro, J., Kolp, M., Mylopoulos, J.: A requirements-driven development methodology. In: Dittrich, K.R., Geppert, A., Norrie, M.C. (eds.) *Advanced Information Systems Engineering (CAISE) 2001*, pp. 108–123. Springer, London (2001). https://doi.org/10.1007/3-540-45341-5_8
19. Bresciani, P., Perini, A., Giorgini, P., Giunchiglia, F., Mylopoulos, J.: Tropos: an agent-oriented software development methodology. *Auton. Agents Multi-Agent Syst.* **8**(3), 203–236 (2004)
20. van Lamsweerde, A., Letier, E.: Handling obstacles in goal oriented requirements engineering. *IEEE Trans. Softw. Eng.* **26**(10), 978–1005 (2000)
21. University of Toronto, Canada; GRL-Goal-oriented Requirement Language. <http://www.cs.toronto.edu/km/GRL/>. Accessed 21 Nov 2017
22. Kang, Z., Peng, C., Cheng, Q.: Top-N recommender system via matrix completion. In: *Proceedings of the 30th AAAI Conference on Artificial Intelligence (AAAI)*, pp. 179–185 (2016)
23. Carey, J., Galletta, D.F., Kim, Y., Te’eni, D., Wildemuth, B., Zhang, P.: The role of human computer interaction in management information systems curricula: a call to action. *Commun. Assoc. Inf. Syst.* **13**(1), 357–379 (2004)
24. Rong, W., Liu, K., Liang, L.: Personalized web service ranking via user group combining association rule. In: *Proceedings of the 7th IEEE International Conference on Web Services (ICWS)*, pp. 445–452 (2009)
25. Yao, L., Sheng, Q.Z., Ngu, A.H., Yu, J., Segev, A.: Unified collaborative and content-based web service recommendation. *IEEE Trans. Serv. Comput.* **8**(3), 453–466 (2015)
26. Driss, M., Moha, N., Jamoussi, Y., Jézéquel, J.-M., Ben Ghézala, H.H.: A requirement-centric approach to web service modeling, discovery, and selection. In: Maglio, P.P., Weske, M., Yang, J., Fantinato, M. (eds.) *ICSOC 2010*. LNCS, vol. 6470, pp. 258–272. Springer, Heidelberg (2010). https://doi.org/10.1007/978-3-642-17358-5_18
27. Cao, J., Fleming, S. D., Burnett, M.: An exploration of design opportunities for “gardening” end-user programmers’ ideas. In: *Proceedings of IEEE Symposium on Visual Languages and Human-Centric Computing (VL/HCC)*, pp. 35–42. IEEE (2011)