

TOWARDS SERVICE ORIENTED ENTERPRISE

Sodki Chaari^{1,2}, Frédérique Biennier¹, Chokri Ben amar², Joël Favrel¹

¹PRISMa, INSA de Lyon, France; Email: sodki.chaari@insa-lyon.fr. ²REGIM, Ecole Nationale d'Ingénieurs de Sfax, Tunisia

Abstract: To fit economical constraints, enterprises are more and more involved in collaborative strategies and require agile organization. To support such agile organizations, an efficient and agile IT system should be set. This can be partially achieved thanks to Service Oriented Architectures (SOA) that are able to implement opened and agile information system. The collaborative model we built on this approach is based first on the services exposed by each potential partner and then on a convenient assembly process used to build a cooperation process, designed as a services composition. This architecture impacts the enterprise own organization, leading to a service oriented enterprise (SOE). After describing the context we present the SOE framework and its components. Then the assembly process is described.

Key words: Service oriented architecture, collaborative business infrastructure, interoperability, semantic, policies, ontology

1. INTRODUCTION

Due to the economic pressure and to fit the agility level required by the changing environment, enterprises have to adapt their internal organization and to develop new business opportunities, often leading to Collaborative Business. These collaborative organizations, that can be seen as adaptive and on-demand virtual enterprises (Kwon et al. 2003) involve that enterprises integrate their business processes into a macro cooperative process to achieve common business goals in a more flexible and effective way. This trend has placed an increased emphasis on extending enterprise internal efficiency across their boundaries through inter-enterprise cooperation that is able to answer dynamically to user demands and deal with complex market

Please use the following format when citing this chapter:

Chaari, Sodki, Biennier, Frédérique, Ben amar, Chokri, Favrel, Joël, 2006, in International Federation for Information Processing (IFIP), Volume 207, Knowledge Enterprise: Intelligent Strategies In Product Design, Manufacturing, and Management, eds. K. Wang, Kovacs G., Wozny M., Fang M., (Boston: Springer), pp. 920-925.

requirements. These collaborative organizations involve designing agile and inter-operable information systems. To support the necessary openness, one can use Service Oriented Architecture (SOA) to implement efficiently the information systems. Coupled to ontology management, dedicated infrastructures can be built by combining services. Nevertheless, these approaches are often fragmentary so that services combination is taken into account only according to a technical point of view.

In this paper, after describing the current context, we propose a multi-level framework that can be used to improve service combination. Our approach, based on SOA, consists in a set of dynamically combined autonomous services that collectively provide a value-added service to users leading to a Service Oriented Enterprise organization (SOE).

2. CONTEXT

Setting a collaborative infrastructure to support C-Business involves paying a particular attention on inter-operability constraints. This leads to adapt the traditional information system organization to support the necessary openness. Emerging technologies such as the Service Oriented Architecture (SOA) (Baglietto et al., 2005), ontology (Yang et al., 2005) are generally perceived as core technologies to successfully deal with these challenges.

Introducing SOA for enterprises information system can reduce dramatically the funds required to start up a business. Services are readily available for integration and orchestration. Moreover, while each service has a machine understandable representation, discovering the convenient service could be achieved in a dynamic and automated way. Hence, enterprises will be able to form on-the-fly, project-driven and on demand alliances. As the number of services increase, a particular attention must be paid on an automated service combination.

Nevertheless, the service definition varies from one company to another as well as the granularity level of service, which can be associated to a whole workflow made of several tasks, or to a simple elementary task. This means that enterprise must “re-think” their business process organization to link them with these services. To reach this goal, enterprises can use an object oriented approach, modelling and decomposing recursively their business processes into a set of business objects that are turned into appropriate services either by adding them a convenient interface or by associating them to composed services. By this way, the IT support and the associated services are “prepared” both for an internal and an external use.

Moreover, bringing the necessary inter-operability involves also taking semantic constraints into account. This can be solved by using ontology. Basically, ontology is defined as an “explicit specification of a conceptualization” (Hu and Du, 2004). It clarifies a terminology shared by one community. Consequently, ontology can be worthy used to organise services and to allow an effective services discovery starting from requests and ensure a good management of available services area.

After discovering the convenient services, consistency controls must be implemented while orchestrating these services. Academic and industrial research has provided several approaches to reach this goal: (Medjahed et al., 2003) presents a framework for orchestrating atomic services semantically described (using non-functional properties such as function, category and quality) while (Fu et al., 2002) proposes a 3-step process including model checking, a service composition model (Mealy machines) and an automatic composition based on finite state machines. Of course, the automatic services composition can be seen as an “ultimate” goal, taking execution models into account. While (Berardi et al., 2005) presents a framework describing services’ behaviour as an external execution tree translated into finite state machines, (Hamadi and Benatallah, 2003) associates a Petri net to each service in order to describe its behaviour and defines some composition operators like sequence, choice, iteration, parallel with communication, discriminator, selection, and refinement.

Nevertheless, these approaches focus on particular technical points. To support dynamic service binding, leading to dynamic virtual organizations, a more complex framework is required.

3. SERVICE ORIENTED ENTERPRISE ARCHITECTURE

As we introduced the Service Oriented Enterprise as a set of on demand well-combined services, a particular attention must be paid on the service selection and combination to set a final cooperation process fitting SOE's goals. Many research works have focused on defining compatibility rules between services. Most of them define syntactic compatibility of input and output parameters and also compatibility rules based on semantic (Medjahed and Bouguettaya, 2005), on behaviour (Bordeaux et al., 2004) or on quality of service requirements (Cardoso et al., 2004). Distribution, dynamicity and heterogeneity of the services can prevent any attempt to understand their semantic. Thus, allowing efficient service sharing needs an adapted infrastructure that must provide flexible tools to organise discovery services, communication facilities, content description, and reconciliation of

incompatible syntax and semantics. Combining ontology with services will empower services with semantics and let us having an appropriate organization of our service area. The core of our model is the organization of service area into different service groups called *domain* (according to their functionalities) which are associated to dedicated ontology and their value-added. A domain is defined by a 3-tuple: $domain = (Id, Member, VS)$ where *Id* is the domain identifier, *Member* is the set of services suppliers whose publish their services in a particular domain and $VS = \{sv_1 \dots sv_n\}$ is the set of virtual services published in one domain. Then, each group is associated to ontology. As service can be related to different domains, ontology, relationships are set to improve the service searching process.

To fit these requirements, we propose to organise the services into a multi-level framework so that different compatibility rules can be set to control the service combination, taking into account both the enterprise policy, semantic constraints and operational assembly rules.

3.1 Framework organization

The framework we propose is based on 3 levels (figure 1):

- **The cooperative process level:** is associated to the enterprise policy description and is used to select the suitable services for the SOE creation. Different types of policies related to each service and global process will be also identified. We call services and policies sets the process schema (PS), described with a 3-tuple: $PS = (SD, PSD, PGC)$ where *SD* is the set of service domains. *PSD* is the set of policies related to every service domain. *PGC* represent the set of global policies associated to the SOE process. This PS schema supports the service selection and combination.
- **The virtual services level** is designed to improve the concrete service searching process, taking into account the different semantic domains and the related ontology. More precisely, this service is in charge of checking the compatibility of services belonging to different levels (process cooperation and basic concrete services). Virtual service is defined as: $vs = \{In, Out, Function, description\}$ where: *In* and *Out* are input and output variables respectively, *Function* = (*Functionality, synonyms*) where *functionality* is used to define what the service can do and *synonyms* presents the set of synonym functionalities, *description* indicates the set of descriptors which can be associated to this service.
- **The concrete services level** is associated to the real IT services. While virtual services are associated to “interface” components (as defined in

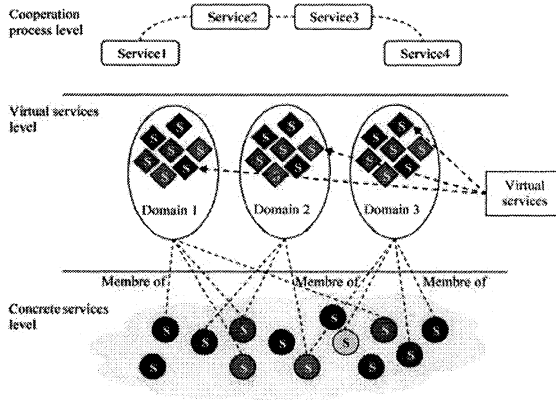


Figure 1. The three level of a service oriented enterprise

the SOA organization), we still have to select the convenient concrete service implementation. This checking process takes both rules and protocols to guide how basic services can be combined to achieve a complex service, fitting the interface requirements defined by the virtual service level.

3.2 Service combination control

Based on this 3 level architecture, our service combination process consists in the following steps:

- *Step 1*: identify the set of virtual services that will be chosen based on the process schema description of the service oriented enterprise and the domain ontology.
- *Step 2*: policies of each service domain extracted from the process schema are associated to the whole selected virtual services sets.
- *Step 3*: based on the set of virtual services defined previously, we try to match each virtual service with concrete services. This task is improved thanks to the different service domains classification. Consequently, the search and match process use syntactic, semantic, and policies compatibility rules. This process implements “vertical” compatibility rules since it verifies inter-levels compatibility and it provides a set of adapted concrete services associated to the virtual services.
- *Step 4*: based on last step output we check whether if the selected concrete services can be composed according to static, semantic and policy rules. We obtain numerous service sequences associated to execution plans. We call this compatibility checking “horizontal compatibilities”.

- *Step 5*: having several execution plans, this last step is used to select only one execution plan according to several criteria as global policy, QoS requirements...

4. CONCLUSION

In this paper, we present how a service oriented enterprise can support inter-enterprise cooperation. This architecture, based on the IT SOA, is organised into different levels (business objects used to model enterprise process are associated to virtual services and finally to concrete services) so that the global infrastructure is closely related to the enterprise business strategy and exhibit convenient agility abilities. We propose to add vertical and horizontal combination rules so that the orchestration process provides a consistent service combination.

5. REFERENCES

- Baglietto, P., Maresca, M., Parodi, A., and Zingirian, N., 2005, Stepwise deployment methodology of a service oriented architecture for business communities, *IST* 47:427-436.
- Berardi, D., Calvanese, D., and Mecella, M., 2005, Composition of Services with Nondeterministic Observable Behavior, in: *Proceedings of the 3rd conference on Service-Oriented Computing*, B. Benatallah, ed., Springer-Verlag, Amsterdam, pp. 43-58.
- Bordeaux, L., Salaun, G., Berardi, D., and Mecella, M., 2004, When are two web services Compatible, in: *Proceedings of the 5th conference on Technologies for E-Services*, M. C. Shan and U. Dayal, eds., Springer-Verlag, Toronto, pp. 15-28.
- Cardoso, J., Sheth, A., Miller, J., Arnold, J., and Kochut, K., 2004, Quality of service for workflows and web service processes, *WS* 1(3),pp.281-308.
- Fu, X., Bultan, T., and Su, J., 2002, Formal verification of e-services and workflows, in: *Proceedings of the workshop on Web Services, E-Business, and the Semantic*, C. Bussler, R. Hull, and S. McIlraith, eds., Springer-Verlag, Toronto, pp. 188-202.
- Hamadi, R., and Benatallah, B., 2003, A Petri net-based model for web service composition, in: *Proceedings of the 14th Australasian Database Conference*, Adelaide, pp. 191-200.
- Hu, H., and Du, X., 2004, Adopting Ontologies and Rules in Web Searching Services, in: *Proceedings of 1st symposium on Computational and Information Science*, J. Zhang, J. He, and Y. Fu, eds., Springer-Verlag, Shanghai, pp. 1047-1053.
- Kwon, Y., Lee, H. K., Lee, S., and Lee, J., 2003, The virtual enterprise: redefining the concept, *Lecture Notes in Computer Science*, volume 2713, Springer-Verlag, pp. 249-258.
- Medjahed, B., and Bouguettaya, A., 2005, A multilevel composability model for semantic web services, *TKDE* 17(7),pp. 954-968.
- Medjahed, B., Bouguettaya A. and Elmagarmid, A.K., 2003, Composing Web services on the Semantic Web, *VLDB* 12(4),pp.333-351.
- Yang, H., Li, Z., Chen, J., and Xia, H., 2005, Web services composition based on ontology and workflow, in: *Proceedings of the 6th conference on Advances in Web-Age Information Management*, W. Fan, Z. Wu, and J. Yag, eds., Springer-Verlag, Hangzhou, pp. 297-307.