Reuse of Service Concepts Based on Service Patterns

Wannessa Rocha Fonseca[1,2] and Pedro Luiz Pizzigatti Corrêa[1]

[1]*PCS / POLI / University of São Paulo, São Paulo, Brazil*
[2]*CEPROMAT, Cuiabá, Brazil*

Keywords: Service, Service-Oriented Computing, Reuse, Service Patterns, Electronic Government.

Abstract: In the process of service-oriented software development, one of the main tasks is to design services, since errors at this stage can propagate throughout the project. This paper proposes a service specification model in the public sector based on service patterns. The service pattern is an abstract service that represents a generic and reusable description. In this context, a lifecycle of service patterns is proposed, as well as the steps for specifying the service patterns. A case study shows an example of a service pattern in the e-government scenario.

1 INTRODUCTION

The adoption of service-oriented computing (SOC) may benefit the area of information and communication technology (ICT) as well as the business area. Some benefits in the area of ICT highlighted by Marks and Bell (Marks and Bell, 2006) are increased productivity, greater reuse of assets, agility and cost reduction of ICT. The business area may benefit from greater flexibility, faster time-to-market for business enterprise, rapid responses to changes in business and finally establishing a closer relationship between ICT service and business needs.

The public sector, in general, appears as a high potential scenario for service-oriented solutions use, especially due to the large number of existing applications, technological diversity, the need for interaction between these applications and the need for service quality management. However, the scenario described does not exempt the public sector from the challenges related to the paradigm of service-oriented development.

In the electronic government (e-government) context when the term *service* is used, it easily relates to the term *electronic service* directly provided to the citizen, through an end-user interface. In this paper, the term service is used to represent a software interface, provided by a public sector, to be consumed by applications from governmental institutions or non-governmental institutions. When the public sector provides an electronic service to citizens by means of an application, that application may be consuming a service (via Web Services) or not, it depends on the approach used to develop the application.

One of the challenges of lifecycle services is the activity of identifying services that support the business activities of the organization. The service identification is seen by Kang, Song and Baik (Kang et al., 2008), Arsanjani *et al.* (Arsanjani et al., 2008) and Boerner and Goeken (Boerner and Goeken, 2009), as a prerequisite for the successful implementation of a service-oriented architecture.

This paper presents research on the service specification, specifically in the government scenario, given that: i) the government scenario has been redundant in business rules in its various levels and sectors of government, and ii) there is the need to reduce effort and cost in designing service-oriented solutions. The goal is to have a service patterns repository for the area of government, and from these service patterns, new services will be created. The idea is that this model may support the reuse of the service concept in the e-government scenario.

There are five sections, besides this introduction. Section 2 presents the characteristics of service patterns. Section 3 work related to the topic of this research. Section 4 presents the proposed model for the reuse of service concept in the setting of e-government and the service pattern lifecycle. Section 5 presents case study. Finally, section 6 presents the paper's conclusions.

2 BACKGROUND

Generally speaking, patterns are reusable solutions for recurrent design problems (Li et al., 2009). The reuse of solutions that have already been devised and that worked in the past is a good practice in the development of systems, regardless of the paradigm this implies. Although Gamma *et al.* (Gamma et al., 2000) address the object-oriented paradigm, they point out that the best designers know they should not solve a problem based on basic principles or from scratch.

2.1 Service Patterns

The concept of service patterns used in this study is similar to that defined by Fki, Tazi and Dupuy (Fki et al., 2010) an abstract service representing a generic and reusable description. Besides this definition, service patterns must contemplate the description of atomic services and compound services, as well as the interactions between services. Thus, service patterns will be able to meet a government task or business process.

Drawing a comparison between services and service patterns, we can say that: i) while the goal of a service is to be part of a highly reusable inventory, the service pattern aims to be part of a highly traceable service repository, ii) While a service is represented by artifacts: specification of services, source code, Web Services Description Language (WSDL) and deployment code, the service pattern is represented only by the Service Pattern Description document, and iii) the reuse of a service is directly related to the amount of service invocations, while the reuse of a service pattern is related to the amount of times this pattern has been used as the basis for specifying new services. The service pattern is not implementable, but it is used as a reference for the service definition. A service pattern is intended to be more abstract than a service. A service interface describes how the service should be consumed, while a service pattern is used as a basis to define a new service.

Service orientation has its origins in past distributed computing design platforms and the influence of established design concepts, approaches and previously published design pattern catalogs. As illustrated in Figure 1, the Service-Oriented Architecture (SOA) design patterns are influenced by design patterns of different areas: object orientation, enterprise application integration (EAI), enterprise application architecture and software architecture. These patterns were certainly influenced by Christopher Alexander's original pattern language.

SOA design patterns are influenced by the design patterns of other areas. Similarly, see the influence of established concepts of SOA design patterns on service patterns and consequently on the specified services from the service patterns may be observed, which is the purpose of this work.

The goal of using service patterns is to assist the specification of new services from existing services in government, aiming at the reuse of already devised solutions. The goal of service-oriented computing is to have service inventories that can meet business processes. Therefore, based on the modelled business process the service patterns catalog should be consulted in order to locate service patterns that may be suitable for the business process.

3 RELATED WORK

Specifically regarding the use of service patterns, the works selected from a literature review on the subject are presented in Table 1. The work analysis supports the understanding regarding the proposed approaches and also the mapping of the following information:

- **Specific Application Domain –** If the patterns presented in the papers are meant to solve a specific business problem or a problem of service design that can be applied in various business areas;

- **Pattern Description –** If the paper presents a description scheme of service pattern. Nazih and Alaa (Nazih and Alaa, 2011) present patterns, but they do not provide a detailed description. It displays a list of generic service patterns and a diagram to represent the service abstraction layer and sequence diagram to represent the interaction between the actors of the systems.

In general, all of these patterns seek to support the reuse of solutions, both solutions to a service design problem, as well as to a specific business problem. Among the works presented in Table 1, those that are closer to the focus of this research are: Nazih and Alaa (Nazih and Alaa, 2011) and Li *et al.* (Li et al., 2009).

The following items are available in the e-PING (BRASIL, 2012) portal: a template for specifying services, called *Documentation Interoperability Services*, and a catalog of services provided by the government. The purpose of the catalog is to provide services that have already been created and made available by the government, aiming to the consumption of these services.
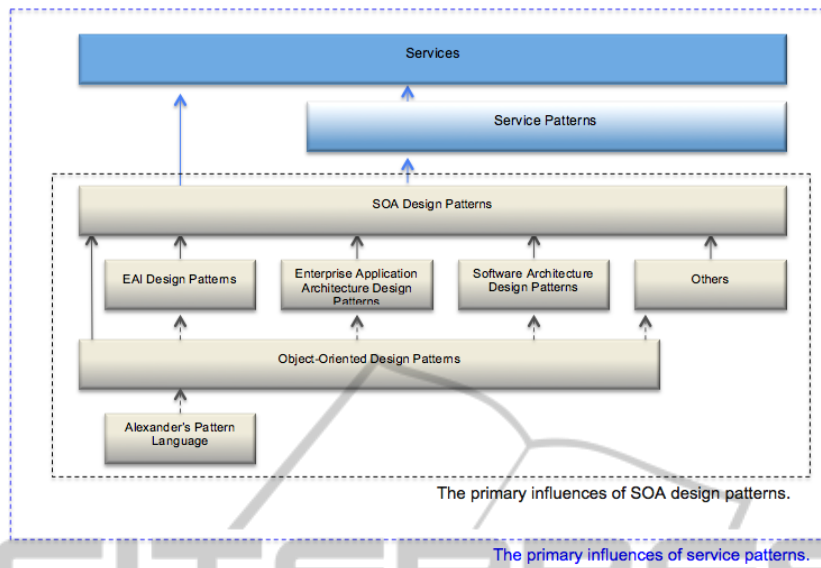
Figure 1: The Primary Influences of SOA Design Patterns - Adapted from Erl (Erl, 2009a).

Table 1: Patterns Related to Services.

Id	Approach	Specific Domain	Descri-ption
(Tchuta and Chunhua, 2011)	Presents a pattern for the creation of atomic services	No	Yes
(Nazih and Alaa, 2011)	Displays Generic Service Patterns for Enabled Public Healthcare Systems.	Yes	No
(Li et al., 2009)	Shows a reputation pattern as a solution to the trust problem in service provision in the SOC.	Yes	Yes
(Fu et al., 2009)	Approach based on service patterns to support service compositions, using ontology.	No	Yes
(Fki et al., 2010)	Presents an approach to service composition, based on the model of the users intention, domain ontology and a service pattern model.	No	Yes

4 PROPOSED MODEL

Figure 2 illustrates an abstract view of Service Specification Model for e-government (SSMe-Gov). The SSMe-Gov aims to:

a) Abstract the concept linked to a service and represent it as a service pattern;

b) Document the service pattern using the *Service Pattern Description* template;

c) Catalog the service patterns in the repository;
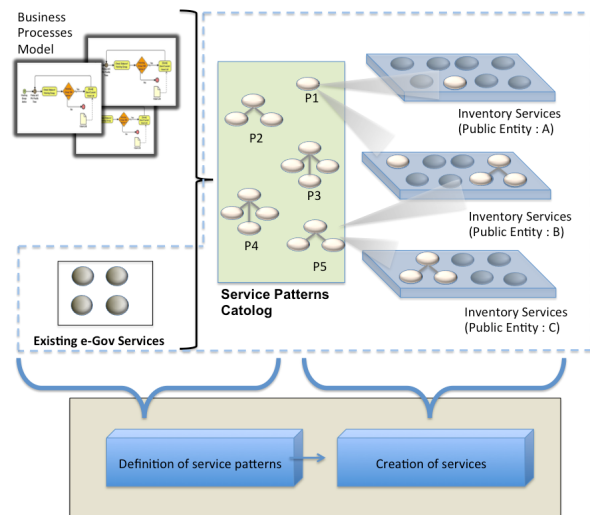
d) Develop new services from cataloged service patterns.

Figure 2: e-Government Service Specification Model Based on Service Patterns.

The model basically consists of two steps, defining the service patterns and creating services from service patterns. In this work will be discussed only the first step.
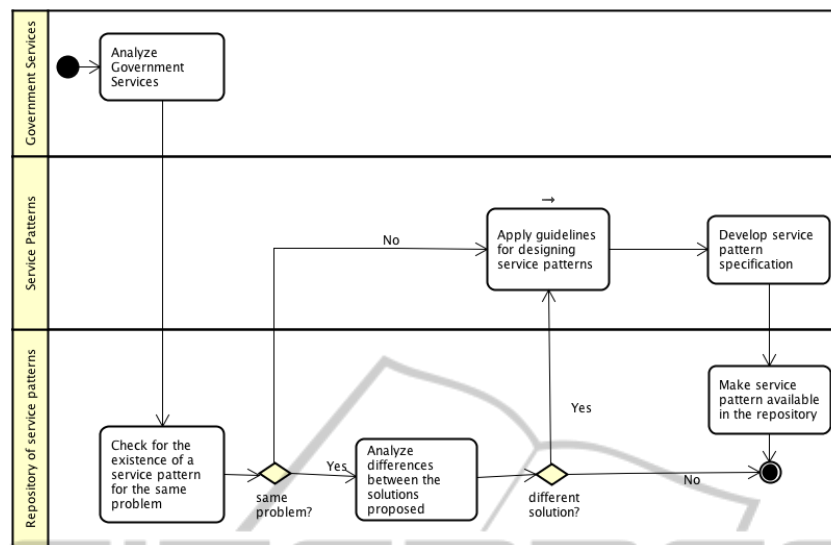
292

Figure 3: Activity to create service patterns.

4.1 Definition of Service Patterns

To define the service patterns the following steps are presented as a diagram in Figure 3.

- **Step 1.** Analyze the existing services in the government scenario.

  a) If an artifact of Service Specification exists, use this document for analysis;

  b) Otherwise, use the WSDL.

- **Step 2.** Check if a service pattern for the same problem already exists in the service patterns repository.

  a) If there is no service pattern to solve the same problem, you must apply the appropriate guidelines to define the service pattern;

  b) If there is a pattern to solve the same problem, the differences between the solutions proposed to solve the proposed problem should be analyzed.

- **Step 3.** Analyze the differences between the solutions proposed by the service pattern.

  a) If there is no difference between the solutions proposed to solve the same problem, one must consider that the candidate service pattern is already documented and the activity should be closed;

  b) In case of difference between the solutions proposed, the appropriate guidelines should be applied to set the service pattern. The new pattern should be documented highlighting the differences between the related patterns, so that

the solution is clear. There may be more than just one pattern in the service patterns repository to solve the same problem, as a problem can be solved in several ways.

- **Step 4.** Apply guidelines for developing service patterns.

  a) Check if the service meets the design principles proposed by Erl (ERL, 2009b)

   i) If the service meets the design principles, move on to the next step;

   ii) Otherwise, the services have to be remodeled.

  b) eliminate existing redundant and similar operations in service;

  c) Abstract the concept of service. Extract the basic concept of the service by eliminating the implementation details, data structure and communication technology.

- **Step 5.** Specify the service pattern.

  a) The pattern must be represented using the Service Pattern Description template;

  b) Represent the service pattern interface. The service pattern interface must only contain business information and not the information on technical control;

  c) An activity diagram that represents the logical operation of the activity must be prepared for each operation.

- **Step 6.** Make the service pattern available in the service patterns repository. The service patterns should be cataloged according to the business area to which the service is associated.

4.2 Representation of Service Patterns

In this work, the service pattern is identified by name and version. For a better understanding of the pattern, it should be described in more detail by presenting its purposes, described description of the problem as well as the solution to solve the mentioned problem. To represent the pattern, certain items must be previously defined: (i) the type of pattern (atomic or compound); (ii) which are the service patterns related to the pattern described, and (iii) which is the catalog the pattern belongs to. The service pattern description template does not intend to present either technical issues and or the internal processing logic of the service. The aim is to show the service pattern interface. In addition, the name of the operation and the UML activity diagram that represents the logic of the operation must be described for each operation of the service.

4.3 Service Pattern Lifecycle

The specification of the service patterns should be based on already designed government services, since the goal is to reuse the concept associated with a service. Figure 4 shows that the identified pattern must be documented and made available in the service patterns catalog.
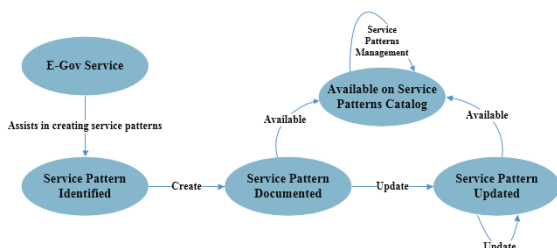

Figure 4: Service Pattern Lifecycle.

The model proposed in this paper is to set service patterns to represent a service at a higher level of abstraction, but it does not aim to set business process patterns. Other authors discuss the use of patterns related to the business process, such as Thom *et al.* (Thom et al., 2007), who addresses the reuse of patterns of workflow modeling tools for the business process and Thom *et al.* (Thom et al., 2009) presents patterns of workflow activities and the potential for their reuse in the context of business process modeling.

5 CASE STUDY

Based on the preliminary survey of government services, a study was conducted on the services related to the Electronic invoice (NF-e) and Electronic Cargo Bill of Lading (CT-e) projects.

5.1 Scenario Description

The replacement of the paper fiscal document for a national model of electronic fiscal document, gave birth to the NF-e project. The NF-e project (BRASIL, 2012b) has reached the status of a national system of electronic fiscal document, shared between the Federal States and the Internal Revenue Service of Brazil.

The goal is for the NF-e issuing company to generate and send an electronic file through the Internet, containing tax information on a commercial operation and also to perform other operations such as cancelling, and rendering NF-e numbering unusable, as well as conducting queries. The communication between the taxpayer application and the State Revenue Office is based on Web Services provided by the Electronic Invoice Reception System. The taxpayer application invokes a service provided by the State Revenue Office. There is a specific Web Services for each service offered. The services available for the NF-e are: 1) NF-e Batch Reception, 2) NF-e Batch Processing Check, 3) NF-e Cancellation, 4) rendering NF-Numbering unusable, 5) NF-e Current Status Check, 6) Service Status Check, 7) Taxpayer Register Check and 8) Event Log for the correction of NF-e information.

Other projects are currently using similar solutions to the NF-e for the processing of electronic documents between a taxpayer application and the State Revenue Office. An example is the CT-e, which is an exclusively digital document, electronically issued and stored in order to document shipping services. The CT-e Project (BRASIL, 2012a) emerged along with the electronic invoice project. This project has pretty much the same service structure, although the information input and output are different, but the concept of the NF-e and CT-e project services are similar. Services concerning CT-e are: 1) CT-e Batch Reception, 2) CF-e Batch Processing Check, 3) CT-e Cancellation, 4) rendering CT-e Numbering unusable, 5) CT-e Current Status Check and 6) Service Status Check. Another project that can also be cited is the Retail Electronic Invoice (NFC-e). One of the motivating factors for the emergence of the NFC-e project was the successful experience with the NF-e (Rehem and Oller, 2012).

After analyzing these projects and the services already created to meet those business areas, it becomes apparent that the service concepts specified for the NF-e project were reused in the project CT-e. Some factors that may have facilitated the reuse of concepts are: 1) the NF-e project has the service specification documented, 2) the institutions and managers responsible for the two systems were the same, making the sharing of experiences easier; and 3) NF-e is an example of a Brazilian model project, both in the area of government and other areas of application, because it somehow affects, directly or indirectly, several business areas in Brazil. In the case of specific projects within the government internal area (G2G) created solutions do not have the same impact and repercussion as the NF-e has had. These factors stimulate and facilitate the reuse of service concepts, but do not guarantee the concept reuse.

5.2 Examples of Service Patterns

In this case study, the creation of a service pattern from the NF-e Batch Reception service is illustrated. This service can be abstracted to contemplate receiving an Electronic Tax Document (DF-e –Documento Fiscal Eletronico) and represented as a *DF-e Batch Reception* service pattern. Considering that there are other types of Electronic Fiscais Documents that must be delivered to the Government, this service pattern have a potential for reuse. To demonstrate this possibility the *DF-e Batch Reception* service pattern established and documented, as illustrated in Table 2. Other patterns were also specified: DF-e Batch Processing Check, DF-e Cancellation, rendering DF-e Numbering unusable, DF-e CurrentStatus Inquiry and Service Status Check.

5.3 Difficulties

According to the proposed model, the creation of service patterns should be made from the existing services in the government, but most services have no documentation. This has been an obstacle to understanding the services. In the case study related to the Electronic Invoice presented in this article, the services had been documented. For other case studies conducted during the research the authors chose to select services from a business area of their knowledge, to solve the problem of lack of documentation. This was a difficulty encountered during the execution of the case study, but the purpose of this work is that, in the real scenario of government, the software architect who creates the

service also make the service pattern documentation himself based on the model proposed in this paper.

The services analyzed in the Government generally represent specific integration solutions and in most cases were not designed aiming at reuse. To solve this problem it was necessary to analyze the operations and to apply the principle of reuse capability of a service, according to Erl (Erl, 2009b).

Difficulty in establishing the level of abstraction of the representation of service patterns. Although there are works that address the use of service patterns, there is no established rule regarding the ideal level of abstraction for defining service patterns. Some patterns are highly abstract as to have only one name (title), others represent all details regarding the concept and technology. The Service Pattern Description template was created to solve the problem related to the level of abstraction necessary to represent the service, containing the necessary information to represent the abstraction.

To conduct the case study, a service survey was performed in public agencies in the State of Mato Grosso directly with the teams that had implemented the services, which demanded availability of the interviewee's agenda. Due to the difficulties encountered in the way of carrying out the survey of services in the State of Mato Grosso, the survey of services in other Brazilian states is being conducted via electronic questionnaire.

6 CONCLUSIONS

The lack of subsidies to promote the reuse of service-oriented solutions in the e-government scenario and the challenges inherent to the activity of identifying services in the service-oriented development cycle have been the motivating factors of this work.

A case study was performed in a real government scenario. The service patterns presented in the case study were defined based on the analysis of existing services in the Government. The case study demonstrated the feasibility of using service patterns as a way to support the reuse of service concepts.

The SSMe-Gov has been proposed. Besides the model, other items have been proposed the lifecycle of service patterns and a the primary influences of service patterns.

This work has also proposed a service patterns catalog. The creation of the catalog itself does not guarantee reuse, but it can support the reuse and design services. Successful reuse depends not only on technical issues but it is also closely related to issues

Table 2: Service Pattern Example - DF-e Batch Reception.

Name	DF-e Batch Reception
Description	Pattern service for the reception of messages from Electronic Fiscal Document (DF-e).
Keywords	Revenue, inancial, tax collection
Version	1.0
Contact	Project Manager – Revenue System (projectManager@nameinstitution.br)
Problem	The DF-e solution can be applied to facilitate the interactions between taxpayer and the government application. The DF-e is an exclusively digital document, issued and stored electronically.
Solution	The taxpayer delivers a batch of DF-e to the State Revenue Office. If the reception of the DF-e batch is successful, the State Revenue Office returns a protocol to the taxpayer application as a delivery receipt.
Type	Atomic Services.
Related patterns	DF-e Batch Processing Check
Catalog	Tax collection
Interface	<<ServiceInterface>> **DfeBatchReception** + dfeBatchReception(fiscalDocumentBatch : String) : String
Operation Name	DF-e Batch Reception
Operation Description	This method will be responsible for receiving delivery messages from DF-e batches and organizing them in the input line.
Diagram	

of management and organizational culture.

Future studies may consider developing an automated environment to manage the patterns catalog, defining mechanisms to search for patterns in this environment, and defining other sources to identify the service patterns, such as business processes. Change management of service patterns is also an important issue.

ACKNOWLEDGEMENTS

REFERENCES

Arsanjani, A., Ghosh, S., Allam, A., Abdollah, T., Ganapathy, S., and Holley, K. (2008). SOMA: a method for developing service-oriented solutions. *IBM Systems Journal*, 47:377–396.

Boerner, R. and Goeken, M. (2009). Service identification in SOA governance literature review and implications for a new method. In *2009 3rd IEEE International Conference on Digital Ecosystems and Technologies, DEST '09*, pages 588–593.

BRASIL (2012). Padrões de interoperabilidade de governo eletrônico - e-PING. Technical report, Brasília: Comitê Executivo de Governo Eletrônico. Documento de Referência Versão 2012.

BRASIL (2012a). Projeto conhecimento de transporte eletrônico (manual de integração contribuinte - padrões técnicos de comunicação). Versão 1.0.4c.

BRASIL (2012b). Sistema nota fiscal eletrônica (manual de orientação do contribuinte - padrões técnicos de comunicação). Versão 5.0.

Erl, T. (2009a). *SOA Design Patterns*. Prentice Hall PTR, 1 edition.

Erl, T. (2009b). *SOA: Princípios de Design do Serviço*. Pearson Prentice Hall.

Fki, E., Tazi, S., and Dupuy, C. S. (2010). Towards a user intention aware service composition. In *New Technologies of Distributed Systems (NOTERE), 2010 10th Annual International Conference on*, pages 113 –120.

Fu, J., Bastani, F. B., Yen, I.-L., and Hao, W. (2009). Using service patterns to achieve web service composition. In *Semantic Computing, 2009. ICSC '09. IEEE International Conference on*, pages 402 –407.

Gamma, E., Helm, R., Johnson, R., and Vlissides, J. (2000). *Padrões de Projeto*. Bookman.

Kang, D., Song, C.-y., and Baik, D.-K. (2008). A method of service identification for product line. In *Proceedings - 3rd International Conference on Convergence and Hybrid Information Technology, ICCIT 2008*, volume 2, pages 1040–1045.

Li, P., Xiangxu, M., Zhiqi, S., and Han, Y. (2009). A reputation pattern for service oriented computing. In *Information, Communications and Signal Processing, 2009. ICICS 2009. 7th International Conference on*, pages 1 –5.

Marks, E. A. and Bell, M. (2006). *Service-Oriented Architecture (SOA): A Planning and Implementation Guide for Business and Technology*. Wiley, 1 edition.

Nazih, M. and Alaa, G. (2011). Generic service patterns for web enabled public healthcare systems. In *2011 7th International Conference on Next Generation Web Services Practices (NWeSP)*, pages 274–279. IEEE.

Rehem, A. and Oller, N. (2012). Projeto nota fiscal eletrônica do varejo NFC-e.

Tchuta, L. and Chunhua, G. (2011). Atomic new service pattern. *International Journal of Software Engineering and Its Applications*, 5(International Journal of Software Engineering and Its Applications):1–20.

Thom, L., Reichert, M., and Iochpe, C. (2009). Activity patterns in process-aware information systems: Basic concepts and empirical evidence. *International Journal of Business Process Integration and Management (IJBPIM)*, 4(2):93–110.

Thom, L. H., Lau, J. M., Iochpe, C., and Mendling, J. (2007). Extending business process modeling tools with workflow patterns reuse. In *International Conference on Enterprise Information Systems, ICEIS, 9., Angers. Proceedings*. ICEIS Press.