

Platform Specific Extensions in the SOA-Pro Profile for the Description of Peer-to-Peer and Grid Services

G. Athanasopoulos, A. Tsalgatidou, M. Pantazoglou
Dept. of Informatics and Telecommunications
National and Kapodistrian University of Athens, Greece
{gathanas, atsalga, michaelp} @di.uoa.gr

Abstract

Developments in service-oriented computing have led to the emergence of heterogeneous service technologies moving well beyond Web services, such as Peer-to-Peer (P2P) and Grid services. Although all these technologies instantiate the de-facto service-oriented model, they introduce a number of technology-specific characteristics which hinder their integration and interoperability. A first step in overcoming this obstacle is to address the aforementioned heterogeneity at the design level. In this paper, we present our work towards enriching the upcoming SOA-Pro profile, which supports the design of service-oriented systems in a platform-agnostic manner, with the concepts necessary to model P2P and Grid services. We propose a set of extensions, which capture the specific features of P2P and Grid services, as they are seen in two prominent representative platforms, namely the JXTA technology for P2P services, and the WSRF framework for Grid services. The applicability and merits of our contribution are demonstrated through a case study based on the Purchase Order example.

1. Introduction

Service-Oriented Computing (SOC) is considered to be the up and coming trend in contemporary software engineering. SOC promises to revolutionize modern development approaches by leveraging interoperability and the seamless utilization of internet accessible functionality. However, the contemporary instantiations of the SOC paradigm, such as Web services, Grid services, and Peer-to-Peer (P2P) services, despite sharing some common features, they basically adhere to incompatible models and standards, and utilize distinct platforms and middleware to accomplish their

objectives. In such context, a first step towards supporting the development of systems composed of heterogeneous services is to address heterogeneity at the modeling and design level.

Existing approaches catering for the modeling of service-oriented systems include graphical languages such as BPMN. BPMN provides the necessary constructs to support the specification of business processes that can be easily mapped to contemporary orchestration languages such as BPEL.

Recently, another approach has emerged, namely the SOA-Pro profile, as a response to OMG's RFP (Request for Proposal) for a UML profile that leverages the description of service-oriented systems. The SOA-Pro profile supports the modeling of technology- or business-oriented services by providing technology-independent constructs capable of capturing generic traits and properties of service-oriented systems. The generality of SOA-Pro and the ongoing standardization efforts driving its specification, render it an appealing approach for modeling heterogeneous services.

Still, due to its platform-agnostic nature, SOA-Pro requires platform-specific extensions that will aid developers in describing specialized properties of contemporary types of services e.g. Grid and P2P services, as they are seen in their respective technologies. With respect to Web services and web service related technologies e.g. BPEL, even though SOA-Pro doesn't accommodate technology related concepts its close affiliation and alignment with the Web service model (as it is document within) do not dictate the provision of major extensions to the SOA-Pro profile.

In this paper, we present our work towards the provision of such platform-specific extensions on the proposed SOA-Pro profile, which accommodate the description of P2P and Grid services. The Grid service extensions have been primarily influenced by the

WSRF framework , which is the prominent paradigm for the provision of Grid services. The P2P service extensions on the other hand, have been primarily influenced by the work in the JXTA platform . In spite of that, the introduced concepts are technology-independent, and thus able to support the description of P2P services that are provided by various P2P platforms (e.g. JXTA , Gnutella and Edutella).

In the following paragraphs, we briefly present the P2P and Grid service paradigms. Consequently, we provide an illustration of the SOA-Pro profile with an emphasis on the concepts that have been used as a basis for our extensions. Then, we present our extensions and a case study which exemplifies their usage. Finally, we conclude by summarizing our work and presenting our remarks.

2. Grid and P2P service paradigms

Since their conception, Grid services have been considered by the service-oriented community as a stateful instantiation of the Web service paradigm . On the other hand, P2P services are an emerging type of service, which are provided and consumed by the peers of a P2P network . Both types of services adhere to distinct models, utilizing heterogeneous protocols and standards, as well as middleware, to support their description, discovery and invocation . In the following, we provide a brief overview of the characteristics of these two types of services.

2.1 Grid services

According to the WSRF specifications, a Grid service is defined as a *WS-Resource* which represents the implicit coupling between a Web service and a stateful computing resource that the Web service manages. This coupling is implicit, as Web service consumers do not directly access the resource state, or the methods that manage this state.

A resource is instantiated, manipulated, managed and finally destroyed as a result of a Web service operation invoked by a client, or, to be more precise, by the exchange of XML messages between the client and the service. The developer of the Web service may decide up to what extent the relationship between the Web service and the associated resources is exposed to the service consumer.

2.2 P2P services

P2P systems generally represent a contemporary class of distributed systems, whose primal objective is

to facilitate the sharing of resources residing at the edges of the network. Within the last decade, several paradigms of such systems and supporting platforms have emerged, including well known application paradigms such as Napster and SETI@Home , or platforms such as JXTA , Edutella and Gnutella .

The definition of P2P services has been a highly debatable issue. Each platform and system incorporates its own properties and features when it comes to defining a P2P service. We consider a P2P service as the *provision of resources or the execution of tasks by one or more (temporarily provider) peers on behalf of one or more (temporarily user) peers in a P2P network* . Peers within a P2P network can be organized within logical groups which usually dictate a set of services that are provided and consumed by the peers of the group.

3. The SOA-Pro Profile

The SOA-Pro profile is a response to OMG's request for a UML profile that supports the description of Service-Oriented Architectures . It has been built as a technology-neutral model that facilitates the description of business- and technology-oriented services. According to , models described in terms of the SOA-Pro profile, when enhanced with platform-related constructs, can be used as input to an MDA process that can lead to the description of platform-specific service-oriented systems.

As it is specified in , central within the SOA-Pro model (see Figure 1) is the notion of *Service*, which represents a capability offered by a set of participants using well defined terms and interfaces. Services are provided through specific ports of a *Participant* who may require the use of other services for its operation. A service required by a participant in SOA-Pro is represented by a *Requisition* element. In turn, participants represent actors which are specified according to the roles they play within a Service-Oriented Architecture (SOA), the services they provide, and the services they use. Hence, participants may be software components, organizations, or even individuals, providing and using services through specific ports.

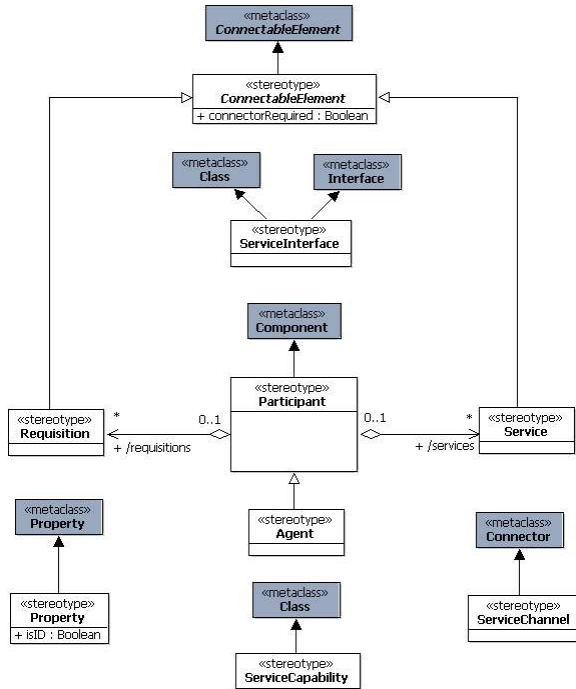


Figure 1: The SOA-Pro service model

The constructs provided by the SOA-Pro profile are generic enough to support the description of any type of SOA. Henceforth, appropriate customizations of those constructs need to be provided so as to be able to describe distinct types of services. In the following section, we introduce a set of extensions to the basic elements of the SOA-Pro profile, which facilitate the description of Grid and P2P service related elements.

4. P2P and Grid service extensions

The utilization of an MDA-based approach for the transformation of Service-Oriented Architectures described through the SOA-Pro profile, in Service-Oriented systems that comprise P2P and Grid services, necessitates the use platform specific concepts. Such concepts should facilitate developers in describing platform specific traits and properties of P2P and Grid services.

In the following subsections we present a set of elements that we have introduced as extensions to the SOA-Pro elements. Subsequently, based on the provided extensions we introduce a case study where we exemplify the use of the introduced concepts.

4.1 P2P Service Extensions

The proposed set of extensions catering for the description of P2P services is based on the core concepts of the P2P service paradigm and thus it includes the notions of *Peer* and *PeerGroup*. These extensions are illustrated in Figure 2 and are described next.

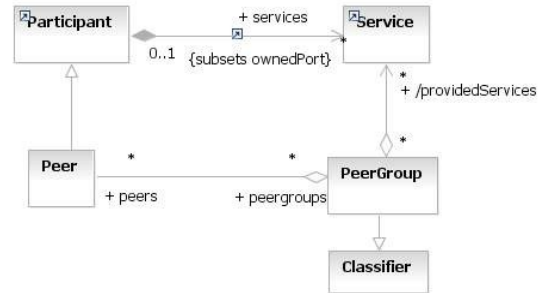


Figure 2: P2P service extensions

4.1.1 Peer. *Peers* are autonomous, independent software systems that communicate and collaborate with each other over a network. They share (provide and/or consume) resources such as files, computation power or storage space with each other, thus facilitating the utilization of resources which reside at the edges of the network. Peers may participate in more than one peer group.

The *Peer* construct extends the *Participant* element of the SOA-Pro profile, which represents a concrete component that can provide and/or consume services. Specifically, the *Peer* element constrains the *Participant* element by mandating that services provided by a peer can only be consumed by other peers in the same network.

4.1.2 PeerGroup. A *PeerGroup* logically aggregates peers which share some common features, e.g. they all provide a common set of services. Services offered by the peers of a peer group are normally called Group Services. A *PeerGroup* may also shape the frame for a set of peers towards various aspects, such as security, or the provision of specific functionality.

The *PeerGroup* element is introduced as an extension to the UML *Classifier* element from the Abstractions::Classifiers package of the UML v2.1 Infrastructure, which represents a set of element instances that share some common features. Specifically, the *PeerGroup* extends the UML *Classifier* by constraining the types of element instances to instances of the *Peer* concept.

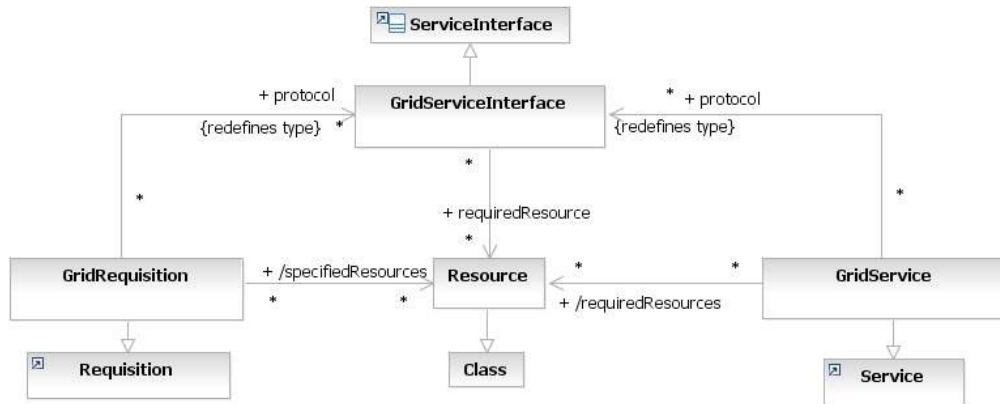


Figure 3: Grid service extensions

4.2 Grid Service Extensions

Figure 3 presents the set of extensions catering for the modeling of Grid services (or otherwise called stateful Web services). This set of extensions comprises the concepts of *GridServiceInterface*, *GridService*, *GridRequisition* and *Resource* which are discussed in the following paragraphs.

4.2.1 GridService: The *GridService* models the connection point via which one may access the capabilities of a Grid service. Actually, it is similar to the *Service* element of the SOA-Pro profile which it extends; it basically denotes the set of capabilities which are offered to its consumers as well as the resources that a consumer should provide to the Grid service in order to utilize its capabilities. A *GridService* instance can only be invoked if it has been associated to a *Resource* instance of the required Resource type.

4.2.2 Resource: The *Resource* concept represents a stateful entity, which could be either a software, or a hardware component. In a grid context, a *Resource* may be also considered as a set of properties the values of which are controlled by its associated *GridService*. Specifically, a *Resource* element specifies a type, whose properties are manipulated by the associated Grid service. It is instantiated, accessed and modified using the implied resource pattern that has been specified in .

It should be noted that the *Resource* element is a specialization of the *Class* element of the Core::Basic package of the UML v2.0 Infrastructure . In particular, the *Resource* element constraints the *Class* element by dictating that one can access the behavior and the properties of the corresponding resource only via the associated service.

4.2.3 GridRequisition: The *GridRequisition* element is a mechanism utilized by a *GridService* consumer in order to access the capabilities of a Grid service; it also provides the resources needed by a Grid service to perform its functionality. A *GridRequisition* element specifies the capabilities required by a Grid service client and the resources the client provides to be utilized by the Grid service. These properties are specified via the *GridServiceInterface*, which defines its type.

Similarly to the *Requisition* element of the SOA-Pro profile, the *GridRequisition* element can be considered as the point of interaction between a Grid service consumer and a Grid service provider. Indeed, the *GridRequisition* extends the *Requisition* with the ability to consume Grid services whose type is specified by the associated *GridServiceInterface*.

4.2.4 GridServiceInterface: A *GridServiceInterface* defines the way to interact with a *GridService*. It is used as a type or protocol for a *GridService* and a *GridRequisition* similarly to the way a *ServiceInterface* is used for a *Service* and a *Requisition* in the SOA-Pro profile. In addition, further to the features offered by the *ServiceInterface*, it addresses the specification of the resources required for utilizing the capabilities of a *GridService*.

5. Case Study

Having defined and described our extensions to the SOA-Pro profile, we proceed with a case study, in order to exemplify the need for and the usage of the concepts introduced above. The scenario of this case study is basically an extension to the Purchase Order paradigm that has been utilized in UPMS RFP and the SOA-Pro profile .

According to this scenario, a consortium of companies has decided to collaborate in order to produce a reusable service for processing purchase orders. The goals of this project include:

- The establishment of common means for processing purchase orders
- The processing and delivery of orders in a timely manner
- The minimization of the stock at hand
- The minimization of production and shipping costs

In order to foreground the need for the utilization of P2P and Grid services by this scenario, we have extended it by introducing another requirement within the presented list; this new requirements, raises the need for the *utilization of existing/available, probably heterogeneous, services and infrastructure*. This is described next.

Thus, we assume that the consortium is in possession of:

- a P2P infrastructure which interconnects all available warehouses, catering for inventory management. Each available warehouse in the P2P network is an autonomous peer providing the inventory management functionality as a P2P service.
- a Grid infrastructure as well as a set of Grid services catering for the planning of subcontracts and supplies, in order to support the current as well as the emerging production plans. These Grid services take as input the current production plans along with the list of materials and tasks that are necessary for the fulfillment of an order.

An illustration of the extended Purchase Order process is presented in Figure 4. As it can be seen in that figure, we have used the *ServiceContract* construct of the SOA-Pro profile to provide a specification of the roles, the interfaces and the choreography of the total process. The identified service contract is a first step towards the specification of the service interfaces.

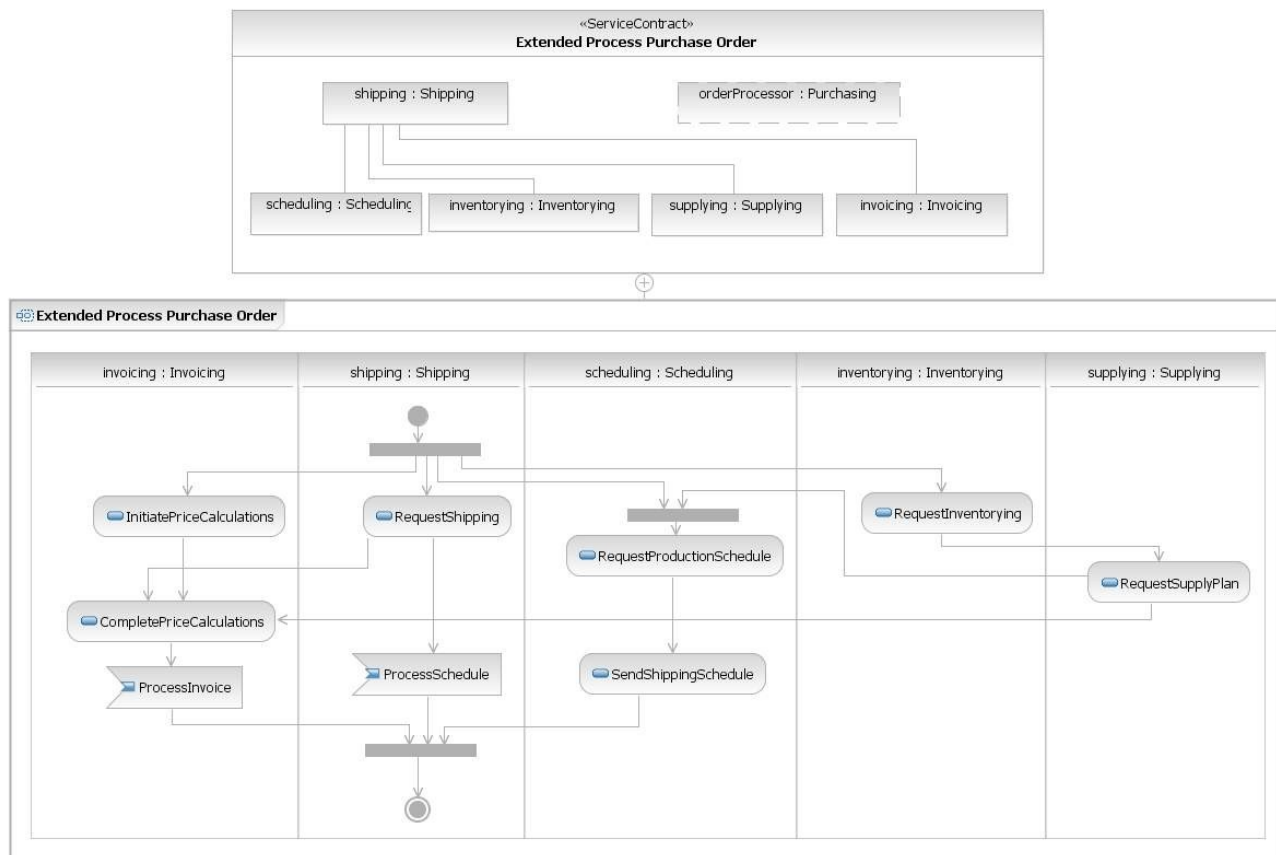


Figure 4: Extended purchase order process

As it is depicted in , the Purchase Order process initiates five parallel activities: one for managing production and shipping scheduling; another for price

calculation and invoicing; a third one for shipping the ordered products; a fourth one for performing the inventory checking; and a fifth one for the planning of

necessary supplies. Processing starts by initiating a price calculation based on the ordered products. However, price calculation can not be finalized since this depends on where the products are produced, the amount of shipping costs and the extra costs which incur from the ordering of missing supplies. At the same time, the order is sent to the inventory in order to identify the missing materials needed for its fulfillment, i.e. the ones that are missing and need to be ordered. The production scheduling is postponed until the process determines the supplies that need to be purchased and their related delivery timetable. Following that, the process calculates when the products will be available and from which locations. In parallel, the process requests a shipping schedule and then waits for it to be returned from the production scheduling. Once the production schedule is available, the invoice can be completed and returned to the customer, along with the shipping of the ordered goods.

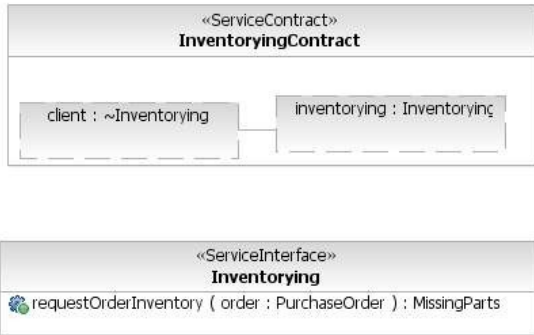


Figure 5: Inventorying P2P service contract

Among the roles specified by the extended purchase order service contract (Figure 4), there is a role catering for the inventory management titled “*Inventorying*”. This role is implemented by a P2P service with the same name, that the consortium has already in possession.

A service contract for that service along with the specification of its interface is illustrated in Figure 5. As it can be seen in Figure 5 the provided service provides an operation called “*requestOrderInventory*”, which accepts a purchase order and after checking the stock of needed parts within the list of the connected warehouses returns the missing parts for its completion.

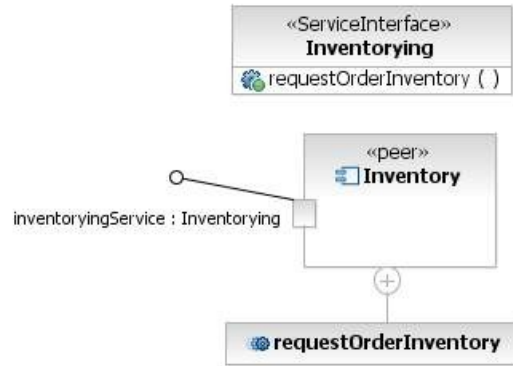


Figure 6: Inventory P2P service description

An implementation of the specified “*Inventorying*” service interface is provided by an “*Inventory*” peer of the P2P network that belongs to the consortium. The “*Inventory*” peer is part of a peer group called “*WarehouseGroup*”. Figure 6 provides an illustration of the peer which implements the specified service along with the group that this peer belongs to.

An additional role identified in the *ServiceContract* of the extended purchase order process is the one called “*Supplying*”. This role is implemented by the Grid service that the consortium has in possession. The service contract of this service along with its interface is specified in Figure 7.

As it can be seen, this service provides a method called “*requestSupplyPlan*”, which returns an estimated time schedule for their expected delivery, based on a list of missing parts.

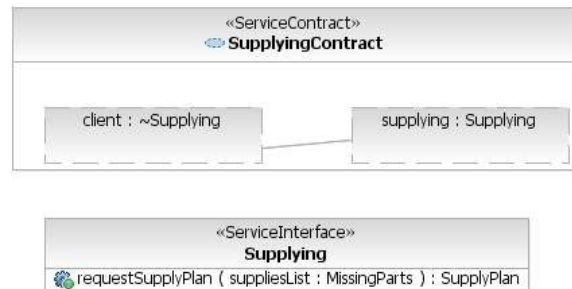


Figure 7: Supplying Grid service contract

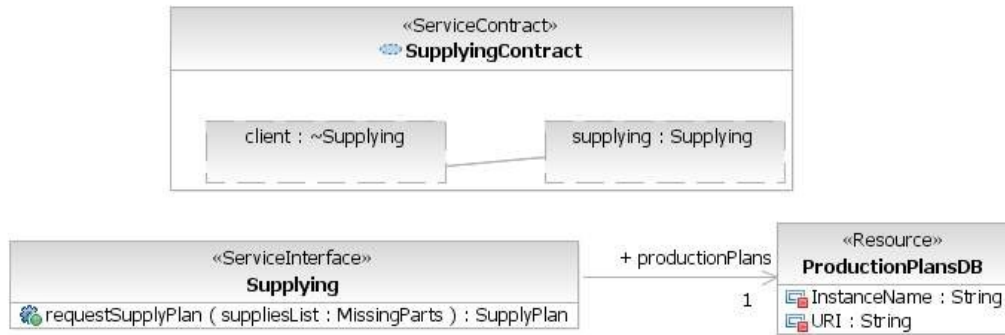


Figure 8: Supplying Grid service Interface specification

Since this service is implemented as a Grid service, the type of Resources that need to be associated with it for its completion are specified in . As it can be seen, this Grid service requires a reference to a database which contains the whole set of production plans that the consortium is responsible for. The properties of this resource are also illustrated in and include an instance name and a URI pointing to the associated database.

6. Conclusions

We presented a set of extensions on the basic constructs of the SOA-Pro profile (a UML profile for the description of Service-Oriented Architectures), which aim to assist a developer in describing Grid and P2P service related concepts when designing service-oriented systems comprising such heterogeneous services. Our main motivation was to leverage the modeling of heterogeneous service oriented systems by providing appropriated extensions, which can be also used as input to an MDA-based development approach that facilitates the specification of service-oriented systems comprising P2P and Grid types of services. This can be considered as a refinement of the SOA-Pro profile's goal that is to support the platform agnostic modeling of service oriented systems.

The provided extensions for Grid services were primarily influenced by the WSRF framework, the prominent paradigm for the provision of Grid services. The extensions for P2P services on the other hand, were derived by a thorough investigation of contemporary P2P service provision platforms such as JXTA , Edutella and Gnutella and they are generic enough to cater for the needs of most available P2P services, in a platform-independent manner. Nonetheless, required extensions for the generation of platform-specific models can be easily provided, due to the extensibility of the introduced concepts.

We would like to note that the provided concepts have already been applied as extensions on two other service models in order to cater for the needs of Grid and P2P services: a) on GeSMO , which is a conceptual model developed in the SODIUM project ; and b) on the conceptual model of the SeCSE project . Furthermore, we have already provided platform-specific extensions to cater for the needs of JXTA P2P services; the introduced concepts were provided as extensions to WSDL resulting a P2P service description language, called PSDL , that supports the discovery and invocation of JXTA P2P services.

Our future work includes the provision of further extensions to the elements presented above in order to facilitate the description of specific traits and properties of other P2P platforms such as Gnutella and Edutella and thus assist developers in specifying actual executable service compositions comprising such services. Another part of our future work is the extension of the introduced concepts in order to support the description of other types of services such as Jini services , Sensor services , UPnP services and any other emerging type of services.

In conclusion we need to state that due to the proliferation of several instantiations of the SOC paradigm i.e. Web, Grid and P2P services and the emergence of several new breeds of services e.g. Sensor services, we believe that the employment of a unified approach towards the development of Service-Oriented systems can be of great benefit to the developers of such systems. Therefore, this work can be considered as a step towards a unified approach in the engineering of Service-Oriented systems.

Acknowledgement: This work has been partially supported by the Special Account of Research Funds of the National and Kapodistrian University of Athens under contract 70/4/5829 and by the European Commission (under contract IST-FP6 004559 for the

SODIUM project and IST-FP6 511680 for the SeCSE project).

7. References

- [1] Athanasopoulos, G., Tsalgaidou, A., Pantazoglou, M.: "Interoperability among Heterogeneous Services", In proceedings of the 2006 IEEE International Conference on Services Computing (SCC), Chicago, USA, 18-22 September 06, pp 174—181
- [2] Tsalgaidou, A., Athanasopoulos, G., Pantazoglou, M.: "Interoperability among Heterogeneous Services: The case of integration of P2P Services with Web Services", To be presented in International Journal of Web Services Research, October-December 2008 , Vol.5 (4), pp 79-110,
- [3] Berre J. A., (eds), UML Profile and Metamodel for Services (SOA-Pro), OMG, Document – ad/08-05-03, Available at <http://www.omg.org/cgi-bin/doc?ad/08-05-03>
- [4] Amsden, J., Odell, J., UML Profile and Metamodel for Services (UPMS) RFP, OMG, Document - soa/06-09-09, Available at <http://www.omg.org/cgi-bin/doc?soa/2006-9-9>
- [5] White, S. A., Introduction to BPMN, White paper, IBM, May 2004
- [6] Gerke, J., Hausheer, D., Mischke, J., Stiller, B., An Architecture for a Service Oriented Peer-to-Peer System (SOPPS), in Praxis der Informationsverarbeitung und Kommunikation (PIK) 2/03, p.90-95, April 2003
- [7] Foster, I., Kesselman, C., Tuecke, S., The Anatomy of the Grid, Enabling Scalable Virtual Organizations, Supercomputer Applications, 2001
- [8] Czajkowski K., Ferguson, F. D., Foster, I., Frey, J., Graham, S., Sedukhin, I., Snelling, D., Tuecke, S., Vambenepe, W., The WS-Resource Framework version 1.0, Whitepaper, March 2004, available online at: <http://www-106.ibm.com/developerworks/library/ws-resource/ws-wsrf.pdf>,
- [9] Alves, A, Arkin, A., Askary, S., Barreto, C., loch, B., Curbera, F., Ford, M., Golland, Y., Guizar, A., Kartha, N., Liu, K.C., Khalaf, R., König, D., Marin, M., Mehta, V., Thatte, S., Van Der Rijn, D., Yendluri, P., Yiu, A., : Web Services Business Process Execution Language Version 2.0. OASIS Standard (2007)
- [10] Gong, L., Industry Report: JXTA: A Network Programming Environment, IEEE Internet Computing, May/June 2001, Vol(5) No(3), pp 88-95.
- [11] Nejd, W., Wolf, B., Qu, C., Decker, S., Sintek, M., Naeve, A., Nilsson, M., Palmer, M., Risch, T., EDUTELLA: A P2P Networking Infrastructure Based on RDF, Edutella White Paper, November 2001, available at <http://edutella.jxta.org/reports/edutella-whitepaper.pdf>
- [12] Ivkovic, I., Improving Gnutella Protocol: Protocol Analysis and Research Proposals, LimeWire Gnutella Research Contest, September 2001
- [13] Chinnici, R., Gudgin, M., Moreau, J.-J., Weerawarana, S., Web Services Description Language (WSDL) Version 2.0 Part 1: Core Language, W3C Candidate Recommendation, 27 March 2006, available at <http://www.w3.org/TR/wsdl20/>
- [14] OMG Unified Modeling Language (OMG UML), Infrastructure, V2.1.2, OMG, Document - formal/2007-11-04, Available at <http://www.omg.org/docs/formal/07-11-04.pdf>
- [15] Napster, Available at <http://www.napster.com>
- [16] SETI@Home, Available at <http://setiathome.berkeley.edu>
- [17] JXTA, Available at <http://www.jxta.org>
- [18] Edutella, Available at <http://edutella.jxta.org>
- [19] Gnutella, Available at <http://www.gnutella.com>
- [20] SODIUM project, Available at <http://www.atc.gr/SODIUM>
- [21] SeCSE project, Available at <http://secse.eng.it>
- [22] Colombo, M., Di Nitto, E., Di Penta, M., Distanti, D., Zuccalà, M., “Speaking a Common Language: A Conceptual Model for Describing Service-Oriented Systems”, in proceedings of the 3rd International Conference on Service Oriented Computing (ICSOC 2005), December 2005, Amsterdam, The Netherlands, pp. 48-60
- [23] Gibbons, P., Karp, B., Ke, Y., Nath, S., Seshan, S., (2003), IrisNet: An Architecture for a World-Wide Sensor Web, IEEE Pervasive Computing, Vol. 2 Num 4, Oct.-Dec. 2003, pp 22- 33
- [24] Jini Org. Available at <http://www.jini.org>
- [25] Newmarch, J. 2005. UPnP services and Jini clients. In Proceedings of the 2005 Conference on Information Systems: Next Generations (ISNG 2005). April 11-13 2005, Las Vegas, Nev.
- [26] Booth, D., Haas, H., McCabe, F., Newcomer, E., Champion, M., Ferris C., Orchard, D., Web Services Architecture, W3C Working Group Note, 11 February 2004, Available at <http://www.w3.org/TR/ws-arch/>