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. It 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 [8], and Peer-to-Peer (P2P) services [10], despite sharing some common features, they basically adhere to incompatible models and standards, and utilize distinct platforms and middleware to accomplish their objectives [1]. 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 [5]. BPMN provides the necessary constructs to support the specification of business processes that can be easily mapped to contemporary orchestration languages such as BPEL [9].

Recently, another approach has emerged, namely the SOA-Pro profile [3], as a response to the Request for Proposals (RFP) of OMG for a UML profile that leverages the description of service-oriented systems [4]. The SOA-Pro profile supports the modeling of technology- or business-oriented services by providing platform-independent constructs capable of capturing generic traits and properties of service-oriented systems. This submission attempts to leverage existing work by OASIS [27] and W3C to ensure conformance with existing reference models and Web Services platforms. The generality of the SOA-Pro profile and the ongoing standardization efforts driving its specification, render it an appealing basis for modeling heterogeneous services, as it can be easily enriched with platform-specific concepts where needed.

As the SOA-Pro profile is closely aligned with the Web service model [26], it can be easily mapped to Web service constructs. However, when it comes to modeling other types of services, such as P2P and Grid services, it requires enrichment with platform-specific constructs due to the wide range of discrepancies and diversities among such service types and the Web service model [1].

In this paper, we present our work towards the provision of 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 Web

Services Resource Framework (WSRF) [8], 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 [10]. In spite of that, the introduced concepts are platform-independent, and thus able to support the description of P2P services that are provided by other P2P platforms e.g. Gnutella [12] or Edutella [11].

In the following paragraphs, we briefly present the P2P and Grid service paradigms followed by 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 validate their usage via a case study. Finally, we conclude by summarizing our work and presenting our remarks.

2. Grid and P2P services

In the following paragraphs we give a brief description of the P2P and Grid services focusing mainly on their specific characteristics which necessitate the enrichment of the SOA-Pro with appropriate extensions to cater for their modeling.

2.1 Grid Services

Since their conception, Grid services have been considered by the service-oriented community as a stateful instantiation of the Web service paradigm [7].According to the WSRF specification, a Grid service is defined as a *WS-Resource* which represents the implicit coupling between a Web service and a stateful computing resource managed by the Web service. This coupling is usually referred as the implied resource pattern and it is implicit, as Web service consumers do not directly access the resource state, or the methods that manage this state. This implied association renders a clear departure of the WSRF model from the Web service model [13] with several ramifications on the Grid service conceptual model related to the management of the associated resource.

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 represent a contemporary class of distributed systems, which aim to facilitate the sharing of resources residing at the edges of the network. Peers of a P2P network can be organized within logical groups which usually dictate a set of services that are mutually provided and consumed by the peers of the group. Within the last decade, several paradigms of such systems and platforms have emerged, including well known application paradigms such as Napster [15] and SETI@Home [16], or platforms such as JXTA [17] , Edutella [18] and Gnutella [19].

The definition of P2P services has been a highly debatable issue. We adopt the definition by Gerke et al [6] who define 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.*

One of the main differences between P2P and Web services is their model of operation in the sense that the client of the former must be either a member of the P2P network or have access to the network through another peer (proxy peer), which is not the case with Web services. Another important difference between P2P and Web services is related to their description: Web services adhere to the WSDL service model (i.e. comprise a set of distinct operations) and utilize the SOAP message format, while P2P services adhere to a suppressed service model (where each service comprises a single operation), and employ proprietary message formats. Furthermore, concepts denoting the P2P network topology (e.g. peer, peer group, etc) are necessary for P2P service descriptions.

The aforementioned differences necessitate the enrichment of the SOA-Pro Profile in order to cater for P2P service modeling.

3. The SOA-Pro Profile

The SOA-Pro profile [3] is a response to the request of OMG for a UML profile that supports the description of Service-Oriented Architectures [4]. It has been built as a technology-neutral model that facilitates the description of business- and technology-oriented services. The models which are 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.

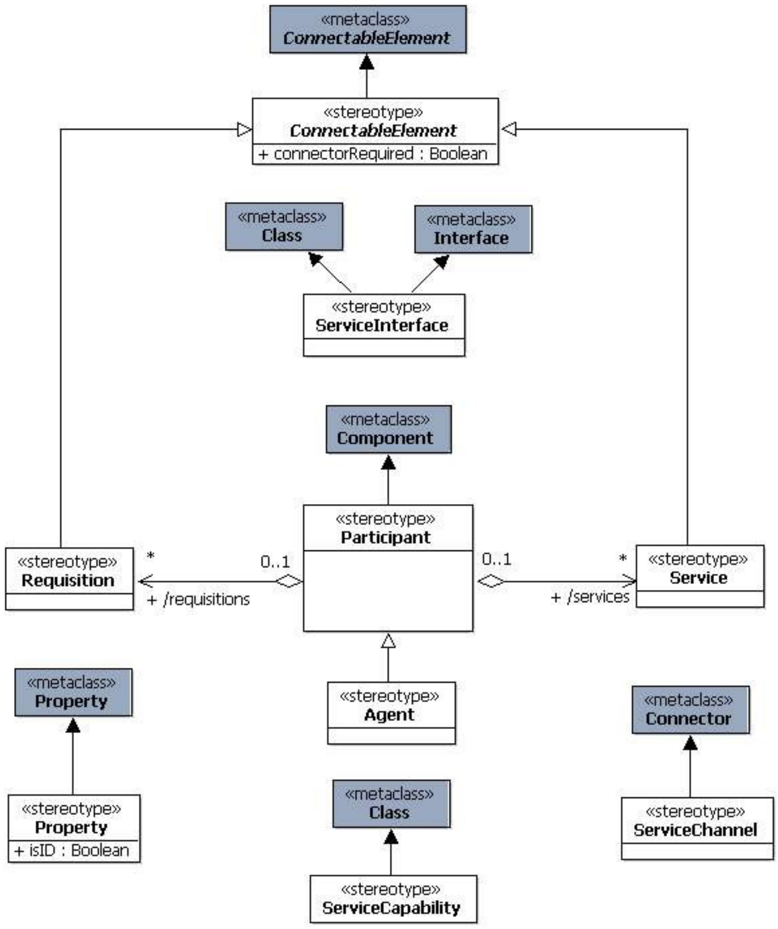

Figure 1: The SOA-Pro service model

As it is specified in [3], the notion of *Service* is central within the SOA-Pro meta-model depicted in Figure 1. This notion 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.

The constructs provided by the SOA-Pro profile are generic enough to support the description of any type of SOA at a high-level and from a technology-independent point of view. The specific traits and characteristics of particular types of services and the platform-specific concepts can be modeled by the enriching the SOA-Pro profile with appropriate elements. In the following section, we describe a set of extensions which we have introduced to the basic elements of the SOA-Pro profile in order to facilitate the modeling of Grid and P2P services.

4. P2P and Grid service extensions

The provided extensions are described next followed by a case study which exemplifies their usage.

4.1 P2P Service Extensions

The extensions catering for the description of P2P services are illustrated in Figure 2. They are based on the core concepts of the P2P service paradigm and thus they include the notions of *Peer* and *PeerGroup*. These concepts have been extracted from a thorough investigation of contemporary P2P service provision platforms (i.e. JXTA[10], Gnutella[12], Edutella[18]) and are described next.

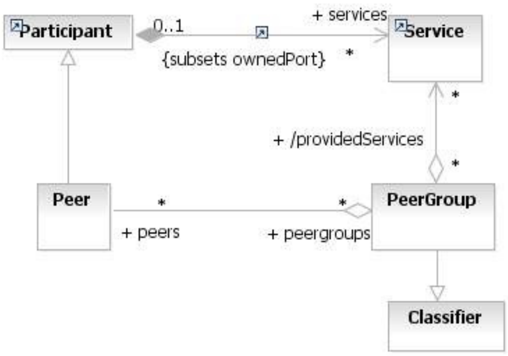

Figure 2: P2P service extensions

4.1.1 Peer. *Peers* are autonomous, independent software systems that communicate and collaborate with each other over a P2P 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 set the context 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 of the Abstractions::Classifiers package of the UML v2.1 Infrastructure [14], 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.

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). The introduced concepts accommodate the needs of the WSRF model which is the prominent specification of the Grid service paradigm. Specifically the notion of a *Resource* is included as a first level element within the profile along with appropriate associations with the *GridRequisition* and *GridService* constructs which support the implied resource pattern introduced in WSRF. These notions as well as the notion of *GridServiceInterface* 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.

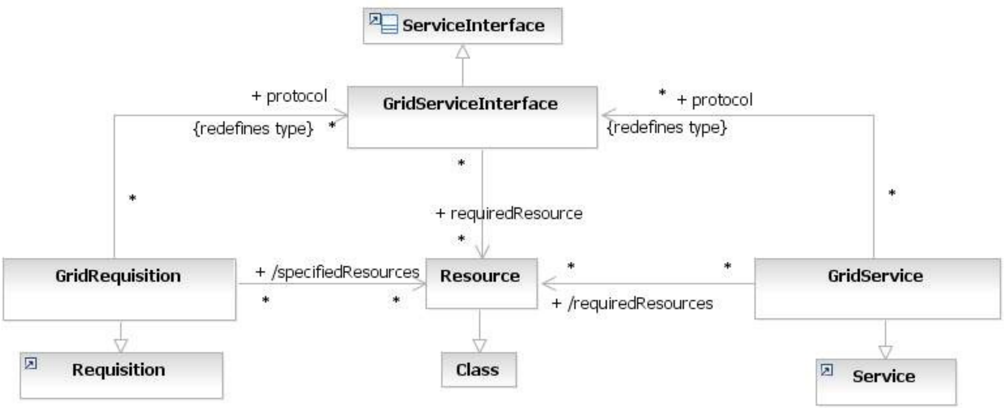

Figure 3: Grid service extensions

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* is also considered as a set of properties the values of which are controlled by its associated *GridService*. It is instantiated, accessed and modified using the implied resource pattern that has been specified in [7].

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 [14]. 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 the conjugate of the *GridService* element. It 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 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

This case study demonstrates the usage and validates the applicability of the concepts introduced above. The scenario is basically an extension to the Purchase Order paradigm that has been utilized in the UPMS RFP [4] and the SOA-Pro profile [3]. The extensions to this scenario include the use of Grid and P2P services, and has as follows:

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

The aforementioned consortium of companies is in possession of:

- a P2P infrastructure which interconnects all available warehouses and supports 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 supporting the planning of subcontracts and supplies while accounting for 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.

As it is depicted in Figure 4, 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.

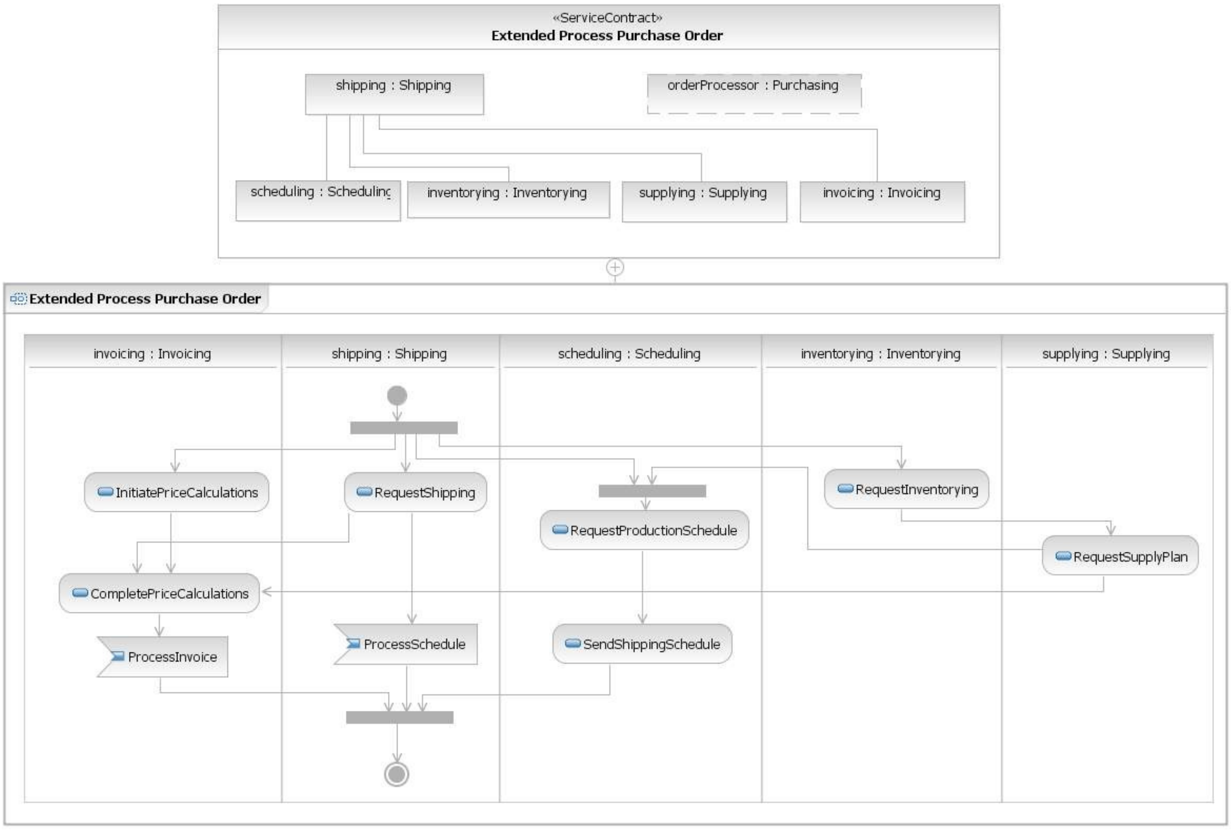

Figure 4: Extended purchase order process

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.

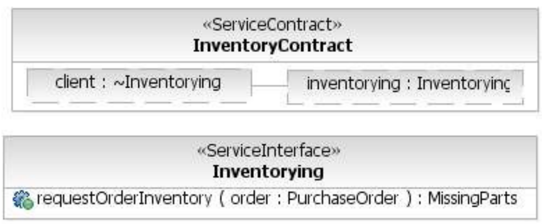

Figure 5: Inventorying P2P service contract

Among the roles specified by the extended purchase order service contract (Figure 4), there is one for the inventory management titled "*Inventorying*". This role is implemented by a P2P service (i.e. "*inventoryingService*" in Figure 6), 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.

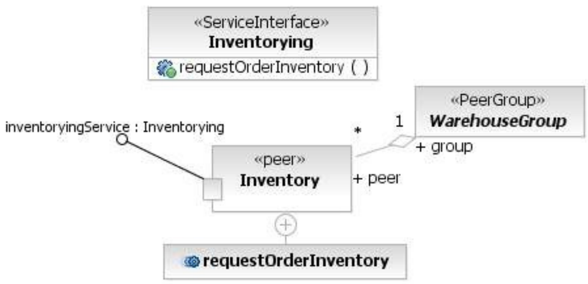

Figure 6: Inventory P2P service description

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 (see Figure 7), this service provides a method called "*requestSupplyPlan*", which returns an estimated time schedule for their expected delivery, based on a list of missing parts.

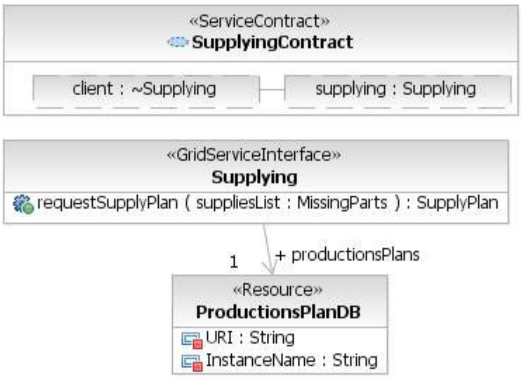

Figure 7: Supplying Grid service contract

Since this service is implemented as a Grid service provided by a participant called Supplier (see Figure 8), the type of Resources that need to be associated with it for its completion are specified in Figure 7. 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.

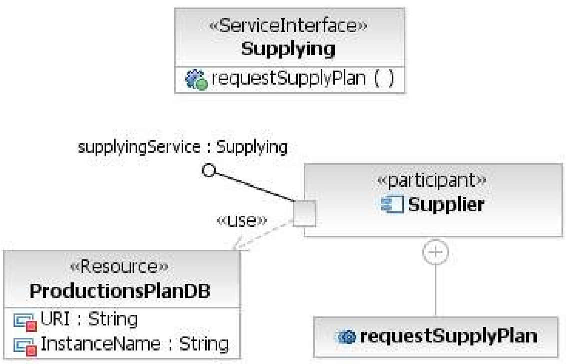

Figure 8: Grid service description

6. Conclusions

In this paper 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 assist a developer in describing Grid and P2P service related concepts when designing Service-Oriented systems comprising such heterogeneous services. The provided extensions for Grid services were primarily influenced by the WSRF [8] framework, the prominent paradigm for the provision of Grid services. The extensions for P2P services were derived by a thorough investigation of contemporary P2P service provision platforms such as JXTA [10], Edutella [11] and Gnutella [12] and they are generic enough to cater for the needs of most available P2P services, in a platform-independent manner. This approach is in alignment with our main motivation, which was to provide extensions that can be used as input to an MDA-based approach to produce platform-independent models which can later be used to generate platform-specific models of Service-Oriented systems comprising P2P and Grid types of services. The required extensions for the generation of platform-specific models can be easily provided, due to the extensibility of the provided concepts.

We would like to note that the provided concepts have already been applied as extensions to two other service models in order to cater for the needs of Grid and P2P services: a) on GeSMO [1], which is a conceptual model developed in the SODIUM project [20]; and b) on the conceptual model of the SeCSE project [22]. Both GeSMO and the SeCSE conceptual models were extended with the addition of the aforementioned concepts so as to facilitate the interoperability between these service types in terms of operations such as discovery, invocation and composition. Furthermore, we have provided platform-specific extensions to cater for the needs of JXTA P2P services; the introduced concepts were provided as extensions to WSDL [13] resulting in a P2P service description language, called PSDL [2], that supports the discovery and invocation of JXTA P2P services.

Our future work includes the development 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 [19] and Edutella [18] 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 [24], Sensor services [23], UPnP services [25] 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 and the emergence of several new breeds of services such as Sensor services or UPnP 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 [20] and IST-FP6 511680 for the SeCSE project [21].

8. References

[1] Athanasopoulos, G., Tsalgatidou, 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] Tsalgatidou, A., Athanasopoulos, G., Pantazoglou, "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., Goland, Y., Guízar, 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] Nejdl, 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] Christensen, E., Curbera, F., Meredith, G., Weerawarana, S., (2001) Web Services Description Language (WSDL) 1.1 W3C Note 15 March 2001, available at: http://www.w3.org/TR/wsdl

[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., Distante, 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/

[27] MacKenzie, C. M., Laskey K., McCabe, F., Brown, F. P., Metz, R., (eds) (2006) Reference Model for Service Oriented Architecture 1.0 OASIS Standard, 12 October 2006, Available at: http://docs.oasis-open.org/soa-rm/v1.0/soa-rm.pdf