



Proceedings of the  
Second International DisCoTec Workshop on  
Context-Aware Adaptation Mechanisms for  
Pervasive and Ubiquitous Services  
(CAMPUS 2009)

VOLARE: Adaptive Web Service Discovery Middleware for  
Mobile Systems

Panagiotis Papakos, David S. Rosenblum, Arun Mukhija and Licia Capra

6 Pages

# VOLARE: Adaptive Web Service Discovery Middleware for Mobile Systems

Panagiotis Papakos, David S. Rosenblum, Arun Mukhija and Licia Capra

Dept. of Computer Science  
University College London  
London WC1E 6BT, UK

{p.papakos, d.rosenblum, a.mukhija, l.capra}@cs.ucl.ac.uk

**Abstract:** With the recent advent and widespread use of smart mobile devices, the flexibility and versatility offered by Service Oriented Architecture's (SOA) makes it an ideal approach to use in the rapidly changing mobile environment. However, the mobile setting presents a set of new challenges that service discovery methods developed for non-mobile environments cannot address. The requirements a mobile client device will have from a Web service may change due to changes in the context or the resources of the client device. In a similar manner, a mobile device that acts as a Web service provider will have different capabilities depending on its status, which may also change dramatically during runtime. This paper introduces VOLARE, a middleware-based solution that will monitor the resources and context of the device, and adapt service requests accordingly. The same method will be used to adapt the Quality of Service (QoS) levels advertised by service providers, to realistically reflect each provider's capabilities at any given moment. This approach will allow for more resource-efficient and accurate service discovery in mobile systems and will enable more reliable provider functionality in mobile devices.

**Keywords:** mobile systems, semantic Web, context awareness, Web services, service discovery, quality of service metrics

## 1 Introduction

Service discovery is of vital importance in the architecture of a Service Centric System (SCS). The capability to dynamically choose an appropriate Web service for a given task is what gives SCSs their versatility. A typical Web service discovery approach in mobile systems is to use a broker to perform Web service matching and binding, given a semantic Web service request from the client [LCab06, Muk07]. This approach is often preferred in mobile systems, as it allows for a third party to act as a broker, which is useful in the resource-limited mobile device world. The client device initiates the search using a semantic request, which usually includes functional requirements, acceptable QoS levels and other metrics such as trust levels. These serve as the criteria to judge whether a service is appropriate for use in a SCS [Ben05].

From the provider side, Web services are advertised via schematic descriptions, which describe the functionality of the service and advertise the QoS levels that this service can provide [Ben05]. The broker will then choose the Web service best satisfying a service request. Currently, Web service discovery in mobile systems is based on architectures developed for conventional desktop systems [LCab06, Muk07]. However, such approaches do not account for the different challenges present in mobile applications. A conventional system, either a client or a service provider, will generally be dedicated to a set of tasks, with sufficient resources to execute that task. In this case, the resources or the context of the client or provider

device are less of an issue. However, unlike the typical desktop system, a mobile device's context changes dramatically during runtime. A mobile client's needs and priorities will change according to his environment and activities, and the Web service request must adapt accordingly to best satisfy these changing requirements. Additionally, a mobile device's resources tend to be limited, therefore proper prioritization of tasks and the services requested is essential to conserve those resources. A more adaptive approach is needed to account for the constant changes in the context and resources of a mobile device during the runtime of a SCS.

In a similar manner, a mobile device may also act as a Web service provider. In that case, the provider device's capabilities may change according to its resources and context as well, which may cause the real QoS levels it provides to be lower than the QoS advertised. This can lead to serious problems in the reliability and performance of a SCS using the provided service, such as bottlenecking. It may even lead to loss of trust if the provider keeps offering lower QoS than advertised. It is therefore important that the QoS levels advertised for each Web service adapts to reflect the current capabilities of the provider device.

This problem becomes more intense when a node acts as both provider and client. In that case, the tasks of requesting services to satisfy the device's needs and providing Web services to others must be balanced, according to the context and the available resources of the device. During runtime, different priorities may need to be given to the requester or the provider side according to the device's status, to ensure optimal operation.

In this paper we describe a middleware based approach called VOLARE, which can adapt the Web Service Discovery process to the mobile environment. This middleware can be used to adapt not only the Web service request of a client device, but also the advertised QoS levels of a service provider to better reflect its current capabilities, enabling more reliable service advertisements. This adaptation can happen dynamically, by identifying the variables of the service request and/or service description that will be altered, and the modifiers that may alter them. By using a modular approach, we also ensure that the existing services and service descriptions do not need to be altered to use the VOLARE middleware.

In this paper, we consider the context of a device to represent the environmental variables that affect the device, including time and date, surrounding devices and user preferences such as current activity, location, cost etc. As resources, we refer to the physical resources of the mobile device such as battery, data storage, network availability etc.

## 2 Related Work

There has already been extensive work in Web service discovery using brokers [Muk07, LCab06]. Such approaches do not focus on dynamic adaptation, since they were designed for systems where availability and resources are less of an issue. They are also not optimized for mobile networks, where the network layout and availability changes radically during runtime.

There has also been significant research on context-aware web services, specifically for mobile phone applications [LCap05, Rae05, She04]. While this work focuses on evaluating environmental variables that may affect the device's performance, or on how to use context data in an application [Rae05, She04], we instead focus on how to use that contextual information to directly adapt the QoS levels requested or advertised.

There have been similar efforts for adapting different policies for QoS thresholds that change according to contextual variables [LCap05] for many applications such as discovering resources. VOLARE focuses on Web service discovery. A key feature in the VOLARE approach is that the variables will be dynamically adjusted according to the changes in context

and resources, without the need to hard code specific policies within the service requests, or using only if-else statements. This allows for much greater versatility and intelligence in the adaptation process. One other difference is that our middleware will also focus on adapting the QoS levels advertised by the Web service provider as well. This ensures that the advertised QoS levels are realistic given the provider device's current status. It also allows the VOLARE approach to balance a single mobile device's Service Request and Provider functionality to improve performance and prioritize according to context and available resources.

A project that attempts a similar approach for web service discovery is PLASTIC[Aut07], however, it focuses on adapting the Service Level Agreements between client-server SLAs. It does not provide actual dynamic adaptation like VOLARE, neither does it provide adaptation of service advertisements, instead opting to adapt the service themselves to comply to requirements, which may not be feasible in the resource constrained mobile environment.

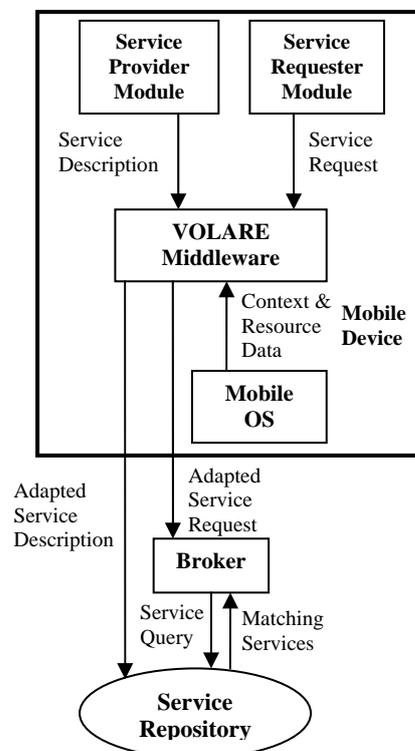


Figure 1: The VOLARE architecture

to hold the policy specifications.

The VOLARE architecture also allows adaptation in the case where a single node acts as both client and provider. The middleware will be able to balance and prioritize between these two functions according to the current status of the device, such as lowering the QoS levels advertised on the provider side when the context indicates that the client side requires a higher priority, and vice versa.

## 4 Example Scenarios

### 4.1 GPS Traffic Control Optimization

## 3 VOLARE Architecture

As illustrated in Figure 1, the VOLARE middleware will intercede between the mobile device OS and the Broker. It will monitor the Context and Resource Data given by the OS, and intercept any Service Requests from the Requester Module. Then the Middleware will adapt the Service Request to better accommodate the present needs of the device.

In a similar manner, any Service Descriptions advertised by the Service Provider module will also be dynamically adapted according to the context and resource variables, to ensure that the QoS levels of the provider match its current capabilities.

Using a middleware approach allows us to keep the adaptation functionality separate from the Web services themselves, and from the Broker module. The adaptation policies themselves are separate from the semantic descriptions or service requests (see section 4.1). This architecture ensures that the adapted service descriptions and advertisements are syntactically identical to unadapted versions, allowing for interoperability with non-VOLARE nodes. The adaptation should have no effect on the functionality of the broker module, or the service repository, however, a repository will be needed



One application of the VOLARE middleware is Traffic Control applications using mobile GPS devices in cars. In this scenario, each device uses Web services to transmit and request traffic data. The communication between mobile devices can be either P2P or through a centralized server. A typical runtime operation would be:

1. The car starts moving. The mobile device will switch to **Driving** mode.  
The **Service Requests** consist of requests about traffic information from surrounding devices, so that the optimal route is plotted.  
The **Service Provider** side consists of the device offering services that inform surrounding devices of its location and traffic status, at a certain refresh rate.
2. The car accelerates on the open road. The device switches to **Low Traffic** mode.  
The **Service Request's** QoS levels get adapted to require higher refresh rates, as the device must quickly acquire information about the surrounding traffic to plot the optimal path, since its location is rapidly changing.  
The **Service Provider** advertised QoS levels, such as the offered refresh rate, is lowered, since priority must be given to the increased activity of the Service Request component, and resources are limited.
3. The car arrives at a traffic jam, and its average speed slows. The device may also detect surrounding devices in **Heavy Traffic** mode, and thus switches to that mode as well.  
The **Service Request** is adapted to require lower refresh rate, since the location changes at a lower rate, and therefore rapid updates are not essential for route plotting.  
The **Service Provider** advertised QoS levels are raised, since there is lower emphasis on the service request side. Additionally, it is more critical to communicate its Heavy Traffic status to other devices so they may plot routes away from that location.

Note that this is an optimization problem scenario, where the adaptive functionality allows for more efficient choice of a service from a list of similar Web services. Using the VOLARE middleware ensures that the service provider and service requester module are given different priorities depending on the situation, so the functionality of the device itself is not affected.

A service description, advertising the capabilities of the Web service, would be:

```
<Web-service>  
  <name>TrafficService</name>  
  <wsdl>www.trafficServices.com/TrafficService</wsdl>  
  <bandwidth> 50 </bandwidth>  
  <refreshrate> 5 </refreshrate>
```

```
</Web-service>
```

To use VOLARE to support this service, we would just need to specify a policy:

```
<adaptation-policy>  
  <resources> Battery, CPU, NetworkAvaliability </resources>  
  <context> Date, Time, Location, Status </context>  
  <adaptation> bandwidth = bandwidth*NetworkAvaliability  
    <status value="Heavy Traffic"> refreshrate = refreshrate*(CPU+Battery)*1.5</status>  
    <status value="Low Traffic"> refreshrate = refreshrate*(CPU+Battery)*0.5 </status>  
  </adaptation>  
</adaptation-policy>
```

As can be seen, the adaptation policy itself is separate from the semantic description of the service offered, yet allowing adaptation of various variables within that semantic description. Note that VOLARE allows both gradual and if-else adaptation approaches, depending on the variable. The VOLARE middleware will intercept the Service Description, and use it's

contextual and resource data to adapt it, according to the specified policy. The policy itself can be adapted if necessary, and the system can support global policies.

#### 4.1 Venue Advertisement System

Another application of VOLARE would be in advertising commercial services, such as venues and gigs. In this case, the user moves around the city. He has set his mobile device to inform him of sites or attractions he may be interested in. The phone is in **Leisure** mode. The **Service Request** module will request attraction or venue information services according to the user's preferences. The QoS levels will be adapted to take advantage of the mobile device's spare resources, without compromising the device's functionality.

The **Service Provider** module may forward local restraint advertisements, including a personal ranking of each venue ascertained by the user's history. QoS levels will again depend on spare resources, but priority is given to the Requester module according to resources.

At lunchtime, the middleware will switch to **Lunch** mode automatically, according to the context data it has about the current time of day. The **Service Requests** will give priority to restaurant venue advertisements, and specifically to the types of restaurant that contextual data indicates that the user prefers. Such data can be collected by monitoring past user reactions to advertisements, for example if the user booked a table at a certain type of venue before etc.

The **Service Provider** module will operate in a similar manner to the Leisure mode, until the user makes a booking. At that point, priority will be given to the Provider Side.

This example demonstrates the use of adaptation in giving priorities to services according to changing content, depending on the context of the device. This will increase accuracy by binding services more likely to satisfy the user's requirements. Additionally, it optimizes the performance of the provider side by adapting according to the available resources.

### 5 Implementation and Evaluation

As part of this project a prototype implementation of VOLARE shall be developed to allow evaluation of the benefits provided by this approach. This prototype middleware will be integrated to a number of mobile devices, which will then use a broker based approach to discover and host a set of existing web services.

Evaluation will be done using existing commercial or academic Web services to produce plausible and convincing case studies. Comparisons between existing service discovery approaches and the VOLARE approach will be conducted to ascertain the differences in efficiency and accuracy of the service discovery process. Additionally, the test mobile devices themselves will be monitored to assess VOLARE's impact on resource efficiency

### 6 Conclusions and Future Work

In conclusion, the automatic adaptation of service descriptions in SOAs is an issue that has yet to be sufficiently addressed by the current state of the art. In this paper, we present a solution to this problem via the VOLARE middleware approach, which adapts service requests and service descriptions of web services according to the status of a mobile device. The middleware will monitor the context and resources of a mobile device and then proceed to dynamically adapt the Web service request or, if the device is a provider, the advertised QoS levels of the offered Web services accordingly. This will ensure more accurate descriptions of a client's current requirements and more realistic service advertisements from the providers. This in turn will increase service discovery efficiency and accuracy by choosing the most

appropriate services according to the current needs of the client. Additionally, with the adapted service requests there is a smaller chance of changing QoS requirements resulting in an already bound service bottlenecking the SCS.

VOLARE also enables providing more reliable Web service provider capabilities on mobile devices, by ensuring that the QoS levels advertised realistically represent the current capabilities of the device. This will in turn increase the overall reliability of SCSs that use that service, and limit the probability of loss of trust between provider and requester because of lower than advertised performance. Finally, by balancing the Requester and Provider functionality it will increase resource efficiency in mobile devices that act as both clients and providers.

There are several areas of this work that will be expanded in the future. In the case of Web service bindings that last for a longer amount of time, a system must be devised to ascertain how and when to switch to an alternative service as the requirements change. Prioritization between concurrent service requests or replies is also an issue that will present new challenges. Finally, the adaptation of composite services and workflows may present interesting research opportunities.

**Acknowledgements:** Panagiotis Papakos is funded by a studentship from the EPSRC. David S. Rosenblum holds a Wolfson Research Merit Award from the Royal Society. VOLARE is an outgrowth of the Dino project. The Dino project is partially supported by the EU Integrated Project SENSORIA (IST-2005-016004). We would also like to thank Sonia Ben Mokhtar and Michele Sama of University College London for their helpful discussions and feedback.

## 6 References

- [Aut07] Marco Autili, Paolo Di Benedetto, and Paola Inverardi. Context-aware adaptive services: The PLASTIC approach. In *Engineering of Software Services for Pervasive Environments (ESSPE '07)*, 2007.
- [Ben05] S. Ben Mokhtar, D. Fournier, N. Georgantas, V. Issarny. Context-aware Service Composition in Pervasive Computing Environments. In *Proceedings of the 2nd International Workshop on Rapid Integration of Software Engineering techniques (RISE'05)*. Heraklion Crete, Greece, September, 2005, LNCS 3943, pp 129-144.
- [LCab06] L. Cabral, J. Domingue, S. Galizia, A. Gugliotta, V. Tanasescu, C. Pedrinaci, B. Norton. IRS-III: A Broker for Semantic Web Services based Applications, In *Proc. of the 5th International Semantic Web Conference (ISWC 2006)*, Athens, GA, USA, 2006.
- [LCap05] L. Capra, S. Zachariadis and C. Mascolo, 2005. Q-CAD: QoS and Context Aware Discovery Framework for Adaptive Mobile Systems, In *Proc. of IEEE Int. Conference on Pervasive Services (ICPS05)*, Santorini, Greece, July 2005, pp 453- 456.
- [Muk07] A. Mukhija, A. Dingwall-Smith, and D. S. Rosenblum. QoS-Aware Service Composition in Dino. In *Proceedings of the 5th European Conference on Web Services (ECOWS 2007)*, Halle, Germany, 2007, pp 3-12.
- [Rae05] M. Raento, A. Oulasvirta, R. Petit, H. Toivonen. ContextPhone: A Prototyping Platform for Context-Aware Mobile Applications, *IEEE Pervasive Computing*, vol. 4, no. 2, pp.51-59, 2005.
- [She04] M. Sheshagiri, N. Sadeh, F. Gandon. Using Semantic Web Services for Context-Aware Mobile Applications In *Proc. Of the MobiSys 2004 Workshop on Context Awareness*.