Dynamic Value Webs in Mobile Environments Using Adaptive Location-Based Services

Peter Ibach[1], Gerrit Tamm[2], Matthias Horbank[1]

InterVal – Internet and Value Chains, Berlin Research Center on Internet Economics (http://interval.hu-berlin.de)
[1]Humboldt University, Computer Science Department, Rudower Chaussee 25, 12489 Berlin, Germany
{ibach, horbank}@informatik.hu-berlin.de
[2]University St. Gallen, Institute of Information Management, Müller-Friedberg-Strasse 8, 9000 St.Gallen, Switzerland
gerrit.tamm@unisg.ch

Abstract

Web Services appear promising to facilitate interoperability and enable Dynamic Value Webs – the on-demand aggregation of services even across enterprise boundaries. However, past experiences have shown interoperability limitations in practice mainly due to information asymmetries on the Web Services markets. This Paper describes a service-oriented architecture for building Dynamic Value Webs in mobile environments using adaptive Location-Based Services. Our concept addresses the following three predominant difficulties: (1) service adaptivity to changing conditions in mobile environments, (2) interoperability including higher levels of semantics, and (3) assuring trustworthiness to all affected parties.

1 Introduction

Web Services are modular self-describing software components [2]. Due to their standardization they may be used individually or may be aggregated to software bundles. Web Services standards facilitate Enterprise Application Integration, offer new ways of interconnecting both, consumer and business partners, and provide a comprehensive framework to enable the evolution of Dynamic Value Webs.

A lot of work on service-oriented computing refers to a vision that enables users to formulate abstract requirements, which are then performed by adaptive, self-configuring services. This implies in particular services that are able to find and to call other services automatically. Approaching this vision, software development based on Web Service standards is expected to exhibit advantages over classical software development. However, experiences have shown that this vision of future Web Services is still difficult to realize. As yet, the volume of transactions in the Web Service market has by far not reached the expected level. Information asymmetries and uncertainties among suppliers, aggregators, and potential customers are the predominant causes for the modest development and enforcement of Web Services [21].

By the help of strategies derived from information and institution economics (Chapter 2 and Chapter 3) the present paper elaborates economical (Chapter 4) and architectural (Chapter 5) requirements for Dynamic Value Webs in Mobile Environments using Adaptive Location-Based Services. The outlook and conclusion (Chapter 6) offer recommendations for governance in Dynamic Value Webs using Adaptive Location-Based Services. The results of this paper point out the significant positive impact of Adaptive Location-Based Services for the Dynamic Value Webs market.

2 Information Economy

Web Services are digital goods which can be aggregated to Dynamic Value Webs on the basis of a service-oriented architecture. Compared to physical products Web Services are thus very flexible. A fast growing market for Web Services was forecasted. Unfortunately, the market for Web Service has so far not lived up to the predicted expectations. Possible reasons for this relatively slow adoption may originate from the different perceived risks of the customers, which rise from a lack of knowledge about Web Services. Additional problems arise regarding configuration, display and communication on the aggregators' side [21]. However, a major problem is that potential customers do not know where their data is saved, what is done with it, how stable the service is, and which of the promised privacy properties it keeps.

The following section uses information economics as a methodical approach for the scientific analysis of the information problems described above and for the development of strategies to overcome these problems. The theory of information economics is part of the new institutional economics. It is based on the idea of classifying the detectability of the quality of a product on the basis of the different consequences of information asymmetries. According to that, several strategies will be developed, aiming at a reduction of the classified types of uncertainties in the context of quality evaluation. The characteristics of the Web Service bundle are attributed to different information economics based types. Then, the strategies recommended shall be deployed in order to facilitate the quality evaluation and the perception of the characteristics of the Web Service bundle. The following section describes the fatal consequences likely to occur if the actors on the Web Service market do not succeed in reducing information asymmetries.

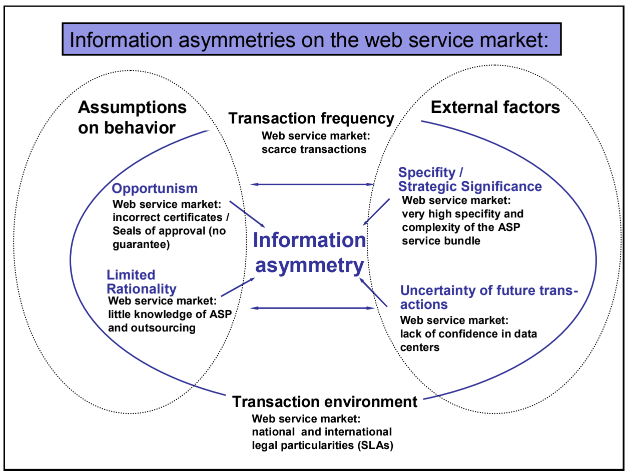

Fig. 1. Influence factors of the information asymmetry on the Web Service market

The information asymmetries prevailing between the supply and the demand side do not only cause high transaction costs, but can ultimately even result in a complete market failure. Fig. 1 relates and summarizes the main causes to the emergence of information asymmetries on the Web Service market [21].

3 Markets versus Hierarchies

Web Services use the infrastructure of electronic networks like the Internet, which introduces new combinations of business characteristics described as Internet economy [15]. Most differences compared to the 'old economy' are advantages in the reduction of transaction costs. Malone et al. characterize the results of using new technologies in economic processes: "By reducing the costs of coordination, information technology will lead to an overall shift toward proportionately more use of markets – rather than hierarchies – to coordinate economic activity" [11].

Most companies on the Internet concentrate on their core competencies for specified services. That leads to a strong division of labour which creates learning-curve effects and increasing economies of scale and scope. Compared with physical markets, a Web Service aggregator for Dynamic Value Webs faces relatively low search expenses, a better market transparency on the Internet, and moderate purchasing costs of elementary Web Services. It will therefore outsource a lot and set its boundaries very tight.

Transaction costs in electronic markets like the Internet are lower then in physical markets. Nevertheless, providers of digital goods, especially suppliers of value webs, must cope with the low willingness to pay for their products. Therefore, the Web Service supplier has to consider strategies to decrease the transaction costs on the supply side. An important aspect of lowering transaction costs is the presence of trust between business partners. Trust develops in partnerships over a long period of time in which the partners are content with the quality of supplied products as well as prices. Costs for searching, negotiating, integrating and especially monitoring do not apply here. However, other suppliers on the Internet might offer a better quality or a better price for the same elementary Web Service. Hence, even in long business partnerships, carriers of Dynamic Value Webs have to constantly watch new market developments.

Another method of lowering transaction costs is the increased speed of contracting between Web Service suppliers and carriers of Dynamic Value Webs. In the short run, one approach is to standardize elementary Web Services in order to make them easier to handle and easier to understand. In the long-run, complex automated electronic markets might be developed which are able to handle a lot of different Web Services very fast and cheap. Then, standardization has to turn into product differentiation.

In this context, two groups of products can be distinguished: Contract goods and exchange goods [1]. Products of the first group are complex and often need description and negotiation before purchase. The second group contains products with clearly delimited and simple properties. Here transaction costs are lower because no complicated negotiation is required. Standardizing a Web Service corresponds to the procedure of transforming a contract good into an exchange good. New approaches to speed-up the contracting process in the Web Service market are contractually assured specifications regarding, e.g., quality of service levels, policy assertions, or pricing models. The premise is that the properties of services are described, fixed, and understood by all interacting parties to build up Dynamic Value Webs. Following, we describe the principle advantages that Web Service standards provide for service coordination and composition and investigate the applicability to set up Adaptive Location-Based Services (Chapter 5).

4 Why Web Service?

Enterprise applications were initially developed on closed, homogeneous mainframe architectures. In the explosively growing heterogeneous landscape of IT systems in the 80's and 90's integration of intra- and inter-company business processes became one of the most important and most cost-intensive tasks of the IT economy. Due to missing or non-transparent standards many enterprises pursued integration by extremely expensive ad-hoc solutions. With the spreading of the Internet and the increasing importance of electronic business, open Internet-oriented solutions have emerged. Enterprise-internal monolithic software was broken into smaller, autonomous, and flexible components. This enabled the access to services not only enterprise-internally, but along the whole value chain to suppliers, distributors, and customers. We characterize this observation as a shift from rigid systems to flexible service-oriented architectures.

In service-oriented computing, resources are accessed via services. Services expose well specified interfaces and are the basic building blocks for flexible and efficient composition of more complex applications. Fundamental concept is the composition of systems by extensive reuse of commodity software/hardware components. Many approaches share this very general concept of compositionality (see Fig. 2).

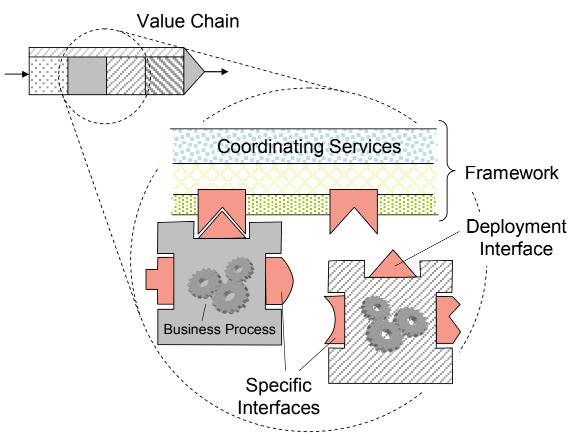

Fig. 2. Compositional architectures play an increasingly important role in value chains due to improved possibilities of interoperability, integration, composability, flexibility, reusability and thus increased efficiency at reduced total cost of operation

However, a number of differences – e.g., in wording, perception, implementation, and practical use – are indicating advantages of the service-oriented paradigm over previous approaches that were focusing on components, objects, modules, or other compositional entities. At the forefront, Web Services and Grid technologies are attracting a lot of attention accompanied by mixed opinions whether the expectations in reusability, composability, flexibility, maintainability, and return on investment that previous approaches have struggled with can finally be accomplished. See, for example, [5, 10] for the growing importance of Web Services in Enterprise Application Integration and [3, 6] for a detailed discussion of pros and cons comparing Web Services to preceding concepts like CORBA. Commonly, the following advantages are attributed to Web Services (still waiting for further empirical inspection):

- Improved degree of interoperability and flexibility (barrier-free computing) – across protocols, interfaces, programming languages, devices, connection lines, operation systems, platforms, enterprise boundaries, vendors, and service providers – through loose coupling based on the eXtensible Markup Language (XML).
- Service aggregation using choreography languages supports "two stage programming" including flow control, exception handling, and transactional processing.
- Integrated directory services such as UDDI or WS-Discovery.
- Enhanced protocols for propagation, discovery, and invocation of "lightweight" services for embedded devices with limited processing and communication capacity in ad-hoc networks.
- Asynchronous dependable messaging and security and privacy support for identification, authorization, access control, and secure data transmissions.

Web Services are intended to facilitate the application-to-application interaction extending established Internet technologies. Thereby Web Services and Grid concepts are converging, guided by the Open Grid Services Architecture (OGSA). Its goal is to overcome the two predominant challenges at the same time: uniform access to services *and* processing resources.

Web Services and Grid toolkits like the Globus Toolkit or the Emerging Technology Toolkit have helped establishing standards. Component-based software for embedded systems [13] and lightweight services [12, 20] expanded the domain to span from distributed client-server applications and globally networked e-business processes down to next generation heterogeneous embedded systems. These developments paved the way towards the general paradigm of service-oriented computing where all kinds of entities are providing, using, searching, or mediating services while efficiently exploiting available resources. Driving the widespread acceptance of the service-oriented paradigm, Location-Based Services might reveal the enormous economic potential of Dynamic Value Webs in mobile business.

5 Adaptive Location-Based Services

Location-Based Services (LBS) are services that utilize their ability of location-awareness to simplify user interactions and adopt to the specific context. With advances in automatic position sensing and wireless connectivity, the application range of mobile LBS is rapidly developing, particularly in the field of geographic, telematic, touristic, and logistic information systems.

However, present LBS are to a large extent incompatible with each other and unable to interoperate on location semantics. They are mostly bound to a specific technology reflecting the preferences of the service provider. Typically, proprietary protocols and interfaces are employed to aggregate the different system components for positioning, networking, content, or payment services. In many cases, these components are glued together to form a monolithic and inflexible system. If such a system has to be adapted to another technology, e.g., the change from GPS positioning to in-house WLAN or Bluetooth positioning, it has to be entirely reengineered. Due to the dynamic nature of mobile environments, available resources as well as achievable quality of service levels are incessantly changing. Thus, adaptivity – the ability of steady interoperation of variable resources under changeable connection conditions – becomes crucial for service end-to-end availability in mobile environments.

Let us consider a position sensing service, for example, a satellite-based GPS. If a mobile device moves from outdoor to indoor environments, the signal will likely become unavailable and position sensing will fail. Without the location information expected from this subservice, composite services depending on it will become unavailable as well. To arrive at seamless operation, on-the-fly switchover to an alternative position sensing service using a different technology is required. To choose from multiple possible position sensing services, the decision has to consider service availability, quality of service properties, and costs. In the near future, most mobile and wearable devices are expected to have multiple position sensing technologies at disposal, e.g., GPS, GSM, WLAN, and Bluetooth. Nevertheless, new technologies, like at present WiMax or RFID, are continuously emerging. Thus hardware devices and software components, their interfaces and architecture have to be able to deal with changing conditions to make mobile Location-Based Services highly-available.

From the perspective of a mobile user, the environment is ever-changing as he moves from one location to another. Adaptivity to location characteristics is essential for mobile service availability. In our approach, adaptivity of a composite Location-Based Service – we call these services Adaptive Location-Based Services (ALBS) – is accomplished by choosing the appropriate chain of subservices for composition (see Fig. 3). Prerequisites are general discoverability, interoperability and composability of subservices through standardized communication protocols and directory services.

Lots of research has focused on Location-Based Services combining the concept of location-aware computing with distributed geographic information services based on Internet standards, see, for example, [7, 8, 9, 14, 16, 17, 18, 19]. In particular, the Open Geospatial Consortium (www.opengeospatial.org) is focusing on Web Service standards for publishing, finding, and binding geospatial services.

In the Web Services standard, interoperable communication is accomplished by exchanging XML data over HTTP. But Web Services are not restricted to the WWW or a specific protocol. Rather, it is a promising solution for adaptive application synthesis in distributed, dynamically changing environments. The notion of Web Services goes beyond prescribed client-server communication. Emerging standards for directory services such as UDDI [22] or Web Services Choreography (e.g., BPEL4WS [4]) allow for dynamic discovery of services and composition of multiple Web Services to fully-fledged distributed applications.

Consider a Location-Based Service that requires some input, e.g., accurate position information or the user's choice of payment. The user might present these data to the LBS manually. Likewise, this information might be the result of a preceding Web Service which, for example, reads the geographic position from an attached GPS device. In case of payment, information about user's choice of payment could be sent to an accounting service which, for example, uses a direct debit authorization. For service composition it is not necessary to know how the accounting is actually performed or how the access to the GPS device is implemented, as long as one can trust the responsible Web Services. Authorization and trust will be fundamental for the success of Location-Based Services and Web Services composition. Moreover, protecting privacy regarding the user's trace of information is a severe issue. Further dangers of intrusion to take care of are service spamming, where undesired services are propagated, and service spoofing, where insecure services are offered under disguised identity. Ongoing developments in the Web Services Trust Language (WS-Trust) therefore accommodate a wide variety of security models. However, proper service specification as well as justification of specification's trustworthiness will remain a major challenge to ensure Web Service interoperability across enterprise boundaries.

5.1 Using Web Services for ALBS Implementation

We use Web Services standards to implement the appropriate selection of subservices and to process their composition. These comprise the service interface description in the Web Services Description Language (WSDL). In an interface description the *port type* specifies the service's request/response-behavior. A service instance is accessed through a *port*. Each port has to bind to a port type and has to support additional binding information, e.g., the used protocol.

For each application to be composed of subservices, a flow through *required* port types and *optional* port types guides the composition process. This flow can be specified using choreography languages (e.g., WSCL or BPEL4WS). Ongoing developments in Web Services Choreography incorporate far-reaching flow control where, for example, a group of services can be processed transactional or the flow may branch with respect to optional services availability or in case exception occurs. The composition process can be graphically expressed by a path through a network of accessible ports:

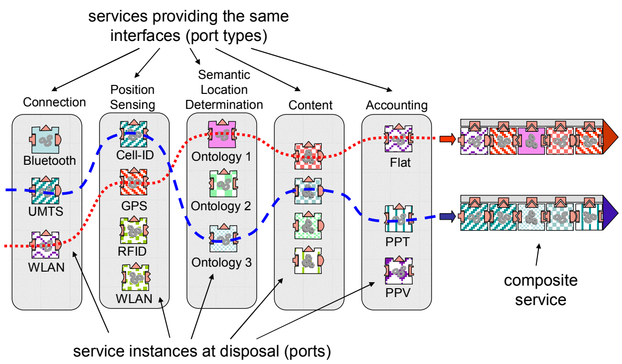

Fig. 3. Building Dynamic Value Webs by on-demand composition of Adaptive Location-Based Services

The composition process is triggered at service invocation. Whenever an ALBS is invoked, it is dynamically composed of suitable ports. Among the ports of each port type, the best match will be taken with respect to the specific context that determines availability and suitability of each port. Therefore, the Web Services Policy Framework defines general purpose mechanism for associating policy expressions with subjects. Thus, specific property assertions can easily be attached to ports and registered in the repository.

For successful composition, at least one port of each required port type has to be accessible. In the example presented in Fig. 3 there are five port types:

- **Connection**: Services of this type allow for access to available networks. This could be WLAN, LAN, Bluetooth, GSM, GPRS, or UMTS connections. Properties attached to these ports comprise information about bandwidth, costs, power consumption, encryption, or latency time.
- **Position Sensing**: This port type provides location information. Ports can be GPS-receivers, services based on stationary RFID-tags, or Bluetooth transmitters. Other services, e.g., based on WLAN-positioning or Cell-ID in cellular networks, are possible candidates as well. The properties should contain information about the accuracy of the location information. Extensions of position sensing services might be able to recognize direction, speed, and variance of movement. (WLAN and Bluetooth positioning base on signal strengths of different access points. For each position, these signal strengths exhibit a certain characteristics that can be translated into location information by a signal characteristics map. Since the signal strengths vary, the map needs periodic update. Nevertheless, coverage and accuracy of the positioning may be insufficient for some LBS. However, this way PDAs, laptops, or other mobile devices can locate themselves independently of GPS availability.)
- **Semantic location determination**: Information about location semantics is offered by this port type. Input is the context-specific sensor data (containing geographic location, RFID numbers, available services, or other specific characteristics that could be utilized to reason about the position). The response includes the semantic position according to a given ontology.
- **Content**: This port type offers content for a given geographic or semantic location. It receives a message with the location information which then is processed. The returned message contains information (text, pictures, audio, or video if requested) about the given location. To process the location information semantically, some common ontology is required (see Section 5.4. for further investigations on location semantics).
- **Accounting**: Accounting port type allows on-demand billing of services used.

5.2 Run-time Adaptation

The sequence chart (see Fig. 4) shows the message sequence of service interaction. The setup in this example consists of a service instance supervising the application control flow, the registry, e.g., an UDDI-implementation, two ports connecting to position sensing services (a GPS service and a WLAN positioning service), and two content ports. The example indicates how the composite service remains viable if some of its ports (here, the GPS positioning service) become temporarily unavailable, and how on-the-fly switchover to a replacement service (here, a WLAN positioning service[1]) takes place.

[1] Here, a connection switchover, e.g., from UMTS to WLAN connection, will probably occur. However, this is processed analogously and is not addressed in the Figure.

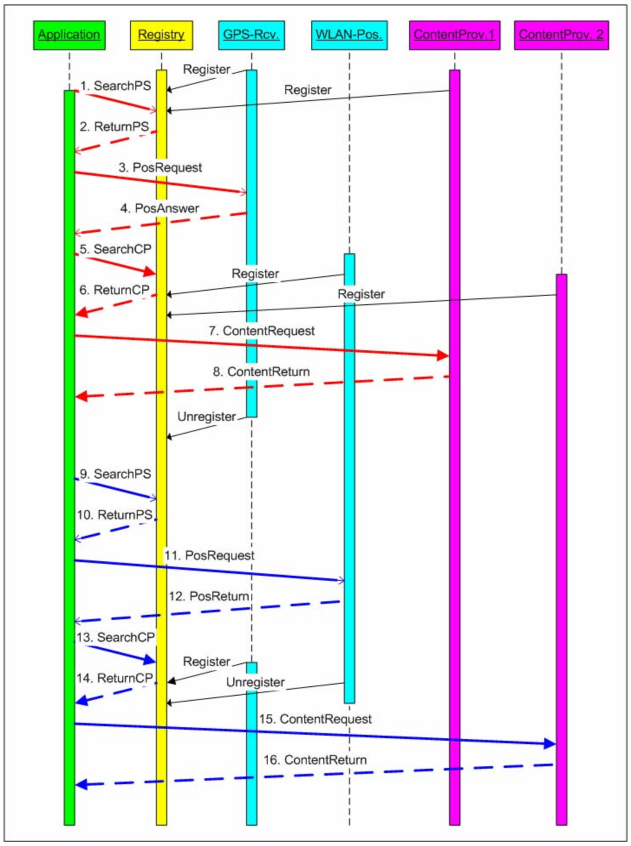

Fig. 4. Sequence chart of service interaction

The first sequence (messages 1-8) shows a service request using the available GPS service:

1. Search the registry for position sensing ports
2. Registry returns a GPS-receiver service port
3. Request position from returned port
4. GPS-receiver returns position
5. Search the registry for content port
6. Registry returns port of Content Provider 1
7. Request content from returned port
8. Content Provider 1 returns content data, e.g., a city map in which the building is located

Before message 9 is being sent, possibly the mobile user is entering a building, where the GPS device cannot receive the satellite signal and therefore unregisters its service from the registry. Supposing an in-house WLAN positioning service becomes available, the second sequence (9-16) shows the service request after this context change:

9. Search the registry for position sensing port
10. Registry returns port of WLAN-positioning service
11. Request position from returned port
12. WLAN-positioning service returns position
13. Search the registry for content provider port
14. Registry returns port of Content Provider 1 and Content Provider 2
15. Supposing semantic information is available that indicates the user is inside the building, Content Provider 2 providing corresponding content will be prioritized and requested
16. Content Provider 2 returns content data, e.g., a map that provides location-based guidance inside the building

As the sequence chart indicates, adaptivity results from context-sensitive service composition. Instead, the messaging behavior of each subservice remains independent of context changes. This is possible because ports of the same port type can be interchangeably replaced without interfering with the port's WSDL-prescribed request/response-behavior.

Traditional monolithic LBS typically do not provide this degree of context adaptivity (here, to switch to WLAN positioning in case the GPS becomes unavailable) without being explicitly designed for every possible change of interoperation. Furthermore, they hardly adapt to emerging technologies that were not foreseeable at design time. In contrast – provided that messaging behavior of new services remains compatible with the given type definition – ALBS can adapt to changing or newly emerging conditions without extra programming effort.

5.3 Using ALBS in Mobile Environments

The fundamental concept of the service-oriented paradigm is to enable uniform access to all resources via services – including mobile or embedded devices, and hardware resources, for example, GPS receivers or WLAN adapters. In dynamic environments, where network topology, connections, and bandwidth are unstable and connected devices may have limited resource power, this requires specific methods for service propagation, discovery, invocation, and processing:

- **WS-Discovery**: The Web Services Dynamic Discovery (WS-Discovery) standard defines a multicast protocol to propagate and locate services on ad-hoc networks in peer-to-peer manner. It supports announcement of both service offers *and* service requests. Efficient algorithms (caching, multicast listening, discovery proxies, message forwarding, filtering, scope adjustment, and multicast suppression) keep network traffic for announcing and probing manageable. Thus, the protocol scales to a large number of endpoints.
- **Lightweight Services**: For efficient invocation and processing of Web Services on embedded devices with limited processing and communication power, "lightweight" services utilize specific real-time protocols, programming languages, scheduling

algorithms, message queuing policies, or XML coding and parsing schemes.

In our example, the mobile device multicasts its request to the devices within local reach and collects the service announcements. The GPS receiver announces a service for position sensing and the WLAN adapter announces two services, one for position sensing and one for connection (see Fig.3). These ports are stored in the local registry cache. Retrieved entries from the global registry are cached as well. To locate a service, the discovery service is instructed to retrieve the corresponding port type. Additionally, the discovery service can look for certain assertions to be satisfied. Thus, the ALBS application communicates with local services the same way it does with remote services. All services are propagated, discovered, and invoked by standard Web Services protocols.

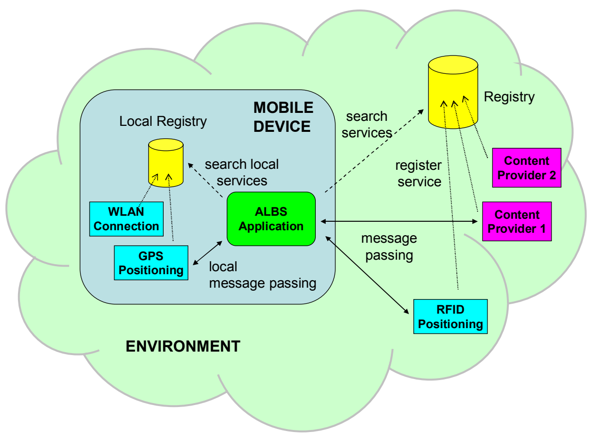

Fig. 5. The ALBS architecture allows for location-aware computing based on universal service-oriented communication

5.4 Semantic Location Information

For semantic interpretation we distinguish the following LBS classes: Location-Based Services can be provided by some immobile unit, e.g., a museum or a botanical garden. Typically such immobile units provide *stationary LBS* which are fixed to a given location. A common problem is to semantically detect the location and find or filter stationary services related to that location. For example, a user's movement in a museum can tell that he might be interested in information about a specific exhibition object (e.g., he moves to that object and then, looking at it, stops moving for some seconds). A location-aware device then could request the assigned service. Likewise, some immobile units may provide *general LBS* that are location-independently accessible but require a location parameter. Examples are a regional weather forecasting service or a service that processes queries like "where is the next subway station?" Regarding *mobile LBS*, the location is a parameter of the behavior of a mobile device. Imagine a user who travels with his laptop. If the laptop recognizes the availability of a specific LAN connection, it could conclude where it is located (e.g., in the user's office) and adapt its behavior (e.g., synchronize certain files and use encrypted messaging). Finally, *interdependent LBS* require multiple related location parameters, e.g., a people finding service that guides mobile users to meet at some intermediate place. All these cases demand for appropriate semantic interpretation of location.

Let us consider the following example of a mobile LBS in more detail: A user wants his mobile phone to automatically activate the hands-free speaking system inside a car or mute when inside a theatre. However, a cellular phone cannot tell from GPS coordinates or from its cell ID that it is inside a theatre. But if there is a service that translates the GPS coordinates or the cell ID to semantic location information like "this is a theatre" or "this is a place to be silent", the "mute feature" can be automated. The Resource Description Framework (RDF) addresses these issues and may be used to accomplish such communication.

For example, an extended position sensing service may return the semantic location "prater.theatres.berlin.de". To know how to act on this location information, a service assigned to it might return the following RDF message indicating that mobile phones and other possibly "obtrusive" devices should be switched off. Since the theatre is an "ambient" place in the following ontology, the device can understand that it should mute:

```
<rdf:Description about="urn://prater.
   theatres.berlin.de">
<rdf:type resource="urn://myontology.
   myID.de/Schema/theatre"/>
<rdf:type resource="urn://myontology.
   myID.de/Schema/places/ambient"/>
<t:Name>Prater</t:Name>
<t:DesiredCellPhoneActivity>Silent</
   t:DesiredCellPhoneActivity>
</rdf:Description>
```

To accomplish interoperability on higher levels of semantics, one has to agree on a suitable ontology which defines terminology and relations for each specific application area. A widely accepted ontology that models physical objects and their location is used in the Geographic Information System (GIS), standardized by the Open GIS Consortium. GPS position sensing together with geocoding services for visualization of geographic information enjoys growing popularity on mobile devices. Yet, seamless outdoor to indoor transitions, global scalability, and changeover to different services, e.g., providing different cartographic material, are usually not addressed. The Physical Markup Language of the EPC Network, standardized by the Auto-ID Center, is intended for product classification, but also allows for spatiotemporal annotations for object tracking and supply chain management. Moreover, the World Wide Web Consortium is extending the Resource Definition Framework to relate Web Content to its associated physical location. In all these attempts, however, expressiveness of location semantics is still in its infancy.

There are various ways to express such additional semantic location information. The device, therefore, not only must be able to access the ontology that is applicable, it moreover needs to know how to map this ontology to its decision alternatives. However, to the best of our knowledge, comprehensive ontologies widely accepted and suitable for broad semantic location processing are not available as yet.

5.5 Interoperability and Trust

Obviously, a major challenge is the appropriate standardization of the above service interfaces to enable interoperability across enterprise boundaries on one hand *and* to keep flexibility to allow further evolution and differentiation of the services on the other hand – a typical tradeoff between tight and loose coupling. Tight coupling, with all participating end points set up to interoperate properly might be easier to implement and more efficient in specific cases. However, this approach will likely result in poor reusability and adaptivity in case of varying and unforeseeable deployment conditions. Here, Web Services with their potential flexibility based on XML are expected to be advantageous.

In the above sections we have sketched how this might be approached in a restricted context, here, in the case of Location-Based Services. However, to generally succeed in Dynamic Value Webs it requires getting over serious hurdles:

- To increase reusability and adaptivity, service specification on higher levels of semantics is required. Not only is the interpretation of such specifications challenging by itself. It additionally gives rise to multiple ways of interpretation – probably resulting in unintended behavior.
- In case of independent actors, opportunistic or even malicious behavior comes into play. Means (e.g., reputation systems) for coordination and cooperation become essential to prevent the market of Dynamic Value Webs from failing.

Higher levels of semantics need to consider support for contracts, contract negotiation, and assessment of fulfillment or violation in case of discord. Possible solutions range from peer-to-peer feedback systems to (decentralized) authorities that may impose reliable reputations or sanctions on participants. One of the most severe issues might be the support for reputation systems that aim at establishing trust. Supporting prerequisites are: security, identifiability, authenticity, timeliness, correctness, and other issues of dependability.

Plug and play in a global scale of interacting software components and hardware devices is not only a problem of connectivity, but moreover of making interacting units flexible and interoperable at a semantic level. Interacting entities need to “understand” interfaces and specifications and adapt such that they are able to utilize each other’s abilities appropriately. Managing communication on higher levels of semantics is the key towards integration of information pieces, software components, business processes, and spreading devices in a global scale. Ultimately, the information infrastructure should carry the whole spectrum of human information exchange beyond people’s limited physical reach in space and time.

The issues to be solved spread manifold up from the physical layer, where the raw data is transferred, to the higher levels of semantics, where data is associated with meaning. This extends from the communication infrastructure and end-user devices (comprising, e.g, increased communication reach, coverage, bandwidth, and usability of devices) to high interoperability and seamless operation in dynamic and mobile scenarios. Finding desired services as well as assessing and assembling them to comprehensive problem solutions is highly demanding to cope with the information asymmetry arising from uncertainty, ambiguity, and complexity inherent to Dynamic Value Webs.

6 Outlook and Conclusions

We investigated the applicability of Web Services standards in the domain of mobile environment, in particular in the field of Location-Based Services. We described how to flexibly compose elementary services along typical value chains using Web Services technology and achieve high adaptivity at the composite service level. Further, we outlined how location information can be processed semantically by using ontologies. We anticipate that the methodology will be applicable to future context-aware computing in distributed, heterogeneous, and dynamic environments at great degree of interoperability. To our conviction, barrierless interoperability on location semantics will tightly interconnect physical and virtual spaces and have a catalytic impact on the markets of Location-Based Services and Mobile Commerce.

The use of open Web Services standards for Location-Based Services has its appealing strengths. And, as is turns out, Web Services standards are quite suitable to overcome a number of interoperability hurdles and to enable the evolution of Dynamic Value Webs. However, a number of desirable mobility characteristics were not originally provisioned in these standards and the integral Web Services support for mobility and ad-hoc adaptivity to changing conditions is still under development. Additional problems remain for future research to make

Dynamic Value Webs flourish. Among the most demanding we see widely accepted ontologies for semantic interoperability, contracts for dependable service specification and composition, and reputation system for justification of specification trustworthiness.

7 References

1. Alchian A. A., Woodward S.: The Firm is Dead; Long Live the Firm: A Review of Oliver E. Williamsons' The Economic Institutions of Capitalism. Journal of Economic Literature, Vol. 26, pp. 65-79, 1988.
2. Alonso, G., Casati, F., Kuno, H., Machiraju, V.: Web Services: Concepts, Architectures and Applications. Springer-Verlag, 2004.
3. Bloomberg, J., Schmelzer, R.: The Pros and Cons of Web Services. ZapThink Report, May 2002. www.zapthink.com
4. BPEL4WS, Business Process Execution Language for Web Services, Version 1.1, www-106.ibm.com/developerworks/library/ws-bpel.
5. Erl, Th.: Service-Oriented Architecture: A Field Guide to Integrating XML and Web Services. Prentice Hall, April 2004.
6. Gokhale, A., Kumar, B., Sahuguet, A.: Reinventing the Wheel? CORBA vs. Web Services. The Eleventh International World Wide Web Conference, Honolulu, Hawaii, USA, May 2002.
7. Hazas, M., Scott, J., Krumm, J.: Location-Aware Computing Comes of Age. IEEE Computer, 37(2):95-97, Feb 2004.
8. Hodes, T. D.: Discovery and Adaptation for Location-Based Services, Ph.D. Dissertation at UC Berkeley CS, January 30, 2003.
9. Ibach, P., Horbank, M.: Highly-Available Location-based Services in Mobile Environments. International Service Availability Symposium 2004, Munich, Germany, May 13-14, 2004.
10. Linthicum, D. S.: Next Generation Application Integration: From Simple Information to Web Services. Addison-Wesley, August 2003.
11. Malone, T. W., Yates, J., Benjamin R. I.: Electronic Markets and Electronic Hierarchies. Communications of the ACM, pp. 484-497, 1997.
12. Milanovic, N., Richling, J., Malek, M.: Lightweight Services for Embedded Systems. IEEE Workshop on Software Technologies for Embedded and Ubiquitous Computing Systems (WSTFEUS), Vienna, Austria, 2004.
13. Müller, P., Stich, Ch., Zeidler, Ch.: Components @ work: Component Technology for Embedded Systems. 27th Euromicro Conference, Warsaw, Poland, Sep 2001.
14. Peng, Z.-R., Tsou, M.-H.: Internet GIS – Distributed Geographic Information Services for the Internet and Wireless Networks, John Wiley & Sons, April 2003.
15. Picot A.: Zehn Eigenschaften der Internet-Ökonomie, http://www.competence-site.de, accessed January 28, 2004.
16. Pinto, H., Boas, N. V., José, R.: Using a Private UDDI for Publishing Location-Based Information to Mobile Users. Proceedings of the ICCC/IFIP 7th International Conference on Electronic Publishing (ElPub 2003), Guimarães, Portugal, June 2003.
17. Rao, B., Minakakis, L.: Evolution of Mobile Location-based Services. Communications of the ACM, 46(12):61-65, Dec 2003.
18. Reichenbacher, T.: Mobile Cartography – Adaptive Visualisation of Geographic Information on Mobile Devices. Dissertation submitted at the Institute of Photogrammetry und Cartography, Technical University, Munich, 2004.
19. Schiller, J., Voisard, A.: Location Based Services, Morgan Kaufmann, April 2004.
20. Schwan, K., Poellabauer, Ch., Eisenhauer, G., Pande, S., Pu, C.: Infofabric: Adaptive Services in Distributed Embedded Systems. IEEE Workshop on Large Scale Real-Time and Embedded Systems (in conjunction with RTSS 2002), Austin, TX, Dec 2002.
21. Tamm, G., Wuensche, M.: Strategies to reduce information asymmetry in Web Service Market. European Conference on Information Systems (ECIS), Naples, 2003.
22. UDDI, Universal Description, Discovery and Integration of Web Services, www.uddi.org.