Towards the Future Internet
G. Tselentis et al. (Eds.)
IOS Press, 2009

doi:10.3233/978-1-60750-007-0-203

The Service Web: a Web of Billions of Services

John Domingue[1], Dieter Fensel[2], John Davies[3], Rafael González-Cabero[4], Carlos Pedrinaci[1]

[1]Knowledge Media Institute, The Open University, Milton Keynes, UK
j.b.domingue@open.ac.uk, c.pedrinaci@open.ac.uk
[2]University of Innsbruck, Technikerstraße 21a, Innsbruck, Austria
dieter.fensel@sti2.at
[3]Next Generation Web Research, IT Futures Research Centre, BT, Ipswich, UK
john.nj.davies@bt.com
[4]Atos Research & Innovation, Atos Origin, Madrid, Spain.
rafael.gonzalez@atosresearch.eu

Abstract. SOA4All, a collaborative European research and development project, is pioneering advanced web technology that will allow billions of parties to expose and consume IT services online. Four complementary technical advances are being integrated to create a coherent and domain-independent service delivery platform. Service-oriented architectures and service-orientation principles are being used to support the development of complex services based on distributed and reusable components. Web principles and technology are used to provide an underlying infrastructure that allows the integration of services at a world wide scale. Web 2.0 is used to structure human-machine cooperation in an efficient, user-adapted and cost effective manner. And semantic technology is used to enhance service discovery, composition and execution.

Keywords: Service Web, Semantics, SOA, Web 2.0, WSMO

1 Introduction

The software industry has been experiencing a dramatic shift from products to services [1]. From a technical perspective, service-orientation is increasingly being adopted to allow applications to be created flexibly by linking loosely-coupled components, typically over a network. From a business viewpoint, fierce competition between vendors is forcing significant reductions in the price of software products. Companies are being compelled to embrace services to add value to product offerings and establish new revenue streams. Furthermore, the appearance of the internet as a disruptive platform has given rise to new business models such as Software as a Service (SaaS) and advertising-based revenue models (e.g., Google). In the light of these trends, business experts are suggesting that consideration should be given to 'servicising' products to provide added value, and 'productising' services so that they can be delivered more efficiently and at lower costs [1].

Large organisations such as Verizon already have systems based on approximately 1,500 web services [2]. However, the web contains only around 27,000 web services based on the Web Services Definition Language (WSDL) [3]. In consequence, service-oriented architecture (SOA) is largely still an enterprise-specific solution exploited by, and located within, large corporations. Because it minimises costs and maximises the potential market, the fully-automated delivery of services over the web appears to be a potential 'silver bullet' for delivering IT services.

We envisage that the combination of semantic technologies and SOA will lead to the creation of a 'service web'—a web that allows billions of parties to expose and consume billions of services seamlessly and transparently, where all types of stakeholders – from large enterprises to SMEs and singleton end users – engage as peers, consuming and providing services within a network of equals. However, as was highlighted in [4], SOA will not scale without:

- properly incorporating principles that made the web scale to a worldwide communication infrastructure;
- significant mechanisation of service lifecycle activities (which includes location, negotiation, adaptation, composition, invocation and monitoring as well as service interaction requiring data, protocol and process mediation); and
- a balanced integration of services provided by humans and machines.

In a service-oriented world, services must be discovered and selected based on requirements, then be orchestrated and adapted or integrated. If a web interconnecting billions of services is to be realised, the way this is done must clearly be both scalable and manageable.

In this paper, we present the principles and rationale behind the EU Seventh Framework project 'Service Oriented Architecture for All' (SOA4All) [5] based upon the integration of SOA, semantic, Web and Web 2.0 technologies. After describing how these 'technological pillars' will be integrated in SOA4All, we indicate how our approach would be instantiated within a telecoms application.

2 SOA and Service-orientation Principles

At the heart of our approach are SOA and service-orientation that embody a number of key principles [4], which we describe below.

Standardized Service Contract Principle. In order to make the description of service capabilities understandable to any interested party, the properties of a service – the service contract – should be compliant with some design standard. The service contract may include information to identify the services such as a URL; functional properties, such as the type of the input/output parameters; and non-functional properties, such as Quality of Service (QoS). Standardisation supports the global interpretability of services, resulting in an increase in the predictability of the service behaviour. The ability to predict the future behaviour of a service is a key mechanism to achieve scalability, since it facilitates the evaluation of the necessary computational resources required to enact a specific service. This mechanism enables the intelligent provisioning of resources.

Abstraction Principle. The abstraction principle dictates that the details of software artefacts that are not required for effective use should be hidden. Therefore all the information necessary to invoke the service is contained in the service contract. Conversely, all the knowledge of the underlying logic and platform technology should be buried completely. The abstraction principle enables replaceability which, when combined with fault isolation and recovery as outlined in [6], enhances scalability.

Reusability Principle. The reusability principle states that the functionality provided by services must be as domain- and context-independent as feasible, facilitating reuse [7]. As a direct consequence of the application of this principle, the logic of a service should be highly generic – that is, independent from its original usage scenario. The reusability principle is a key enabler for SOA infrastructures, since it makes possible the creation of huge libraries of domain-independent services that leverage the construction of new complex context-dependent services.

Autonomy Principle. The autonomy principle states that services should be able to carry out their processes independently from outside influences. The only way to affect the results of a service should be through the modification of the input parameters as specified in the service contract. Service autonomy increases reliability and, more importantly, predictability and fault isolation. As set out in [6], this leads to improved system scalability.

Statelessness Principle. The statelessness principle dictates that services should minimise resource consumption by deferring the management of state information when necessary [7]. This notion of statelessness has been taken to the extreme in the REST architectural style [8], which has also been successfully applied to SOA in recent years. Because state maintenance is one of the most resource consuming tasks in computer science, conformance with the statelessness principle is vital if the entire SOA infrastructure is to be scalable.

Discoverability Principle. The discoverability principle states that, to enable discovery by interested parties, services should be annotated with metadata. This principle is closely related to the standardised service contract principle.

Composability Principle. The composability principle requires services to be identified as effective composition participants, regardless of the size and complexity of the composition [7]. From a bottom-up perspective, this allows simpler services to be combined into larger services (c.f. reusability). Looked at top-down, service composition is an effective way to tackle the complexity of certain types of processes (c.f. abstraction). Because the ability to create new services easily is a key pre-requisite for the widespread take up of SOA, the composability principle is a core element within the definition of a service web.

Despite the original aim to support inter-organisational processes on the basis of SOA, it still remains almost exclusively an intra-organisational solution. To support the service web, we believe the principles underlying SOA must be revisited and extended with others coming from the web, semantic technology and Web 2.0 systems. We consider each of these in turn in the next three sections.

3 Enhancing SOA with Web Principles

The service web will lead to a global, dynamically-changing environment of services accessible for third-party usage. Services will undergo many changes within this environment, as resources currently do on the web, and the rate of churn will be very high. Users and resources will appear, disappear, and change location. Resources that initially are available free will transform to pay-per-use and vice versa. And services may occasionally be blocked, out of service, or inspected for antitrust violations. As a result, we believe that the transformation of SOA into an architecture comprised of billions of services requires the embodiment of the principles which made the web such a successful platform for the worldwide sharing of content.

Among the main principles underlying the web, its distributed and open nature is most relevant. The web is essentially a collection of distributed resources which, however, transparently appears to the user as a single entity where anybody can provide and consume resources. Distribution enables scalability but, when it comes to supporting the execution of processes, the absence of central control over many of the resources and services involved means other mechanisms must be provided to cope with changing conditions. Research in service-oriented computing already recognises the need for dynamic and adaptive processes within enterprises [9]. The service web will exacerbate this to a point where the existence of adequate means for self-configuring, self-adapting or self-healing processes will determine its success. In addition, aspects like the discovery and composition of services and the distributed and open nature of the service web will make it a very valuable yet technically challenging environment.

Another principle underlying the web is the use of a vendor-neutral interoperability platform that supports, and is based on, the integration of heterogeneous and proprietary solutions. A web of billions of services will require an advanced infrastructure that properly supports the integration of data and processes independently of their format, protocol or location. In fact, we foresee the creation of constellations of service ecosystems that span institutions, communicating seamlessly using the service web as a common interoperability platform.

Finally, a key feature of the web lies in the central role that users play. The ability to maintain this role in the service web will determine the market for services, which will in turn affect the uptake of service technologies on a web scale. For users to continue to play a central role as the service web evolves and grows, simple yet fully-fledged and customisable solutions must be created.

4 Enhancing SOA with Semantic Technologies

Standards for describing web services currently use syntactic (XML-based) notations such as WSDL. These descriptions are not amenable to applying automated reasoning, so specialist workers must be involved throughout the web service lifecycle. This causes numerous problems, the most significant of which is the lack of scalability. It simply isn't possible to maintain millions, let alone billions, of services to cope with environmental and context changes, discover new services, compose them or support their adaptation at runtime through human effort alone. Researchers studying the semantic web, semantic web services and more traditional artificial intelligence have shown it is possible to automate some of these tasks to a significant extent (see, for instance, work in OWL-S [10], WSMO [11], WSDL-S [12], semantic execution environments [13, 14] or planning [15]). Based on their findings, we decided to use semantic technologies as a core pillar of the service web.

By providing semantic descriptions of service interfaces and capabilities, the way is paved for simplifying and further automating the discovery of suitable services. In particular, services can be discovered based on the capabilities they offer rather than the low-level messages exchanged. Our approach to discovering suitable services is based on the notion of *goal* set out in WSMO. On the one hand, goals provide means for users to model explicitly their requirements (what to do); on the other, they serve as a natural abstraction over web services (how to do it). The notion of goal therefore enables the reusability of existing services while reducing the complexity for managing billions of services by providing additional layers of abstraction for guiding the discovery process [2].

This very distinction between goals and web services also enables what is usually referred to as late binding – that is, service selection at run-time [13, 14] – which has proven a useful approach for the management and execution of services [16]. From the management perspective, specifying processes in terms of goals provides further robustness and maintainability to the process models avoiding the creation of 'spaghetti solutions' that deal with the specifics of different service providers's implementations and using instead a single entry point. From an execution perspective, the infrastructure is given the possibility to choose the 'best' service taking into account contextual factors.

To achieve the level of flexibility required for the service web, we decided to extend the heuristic classification-based engine we use to dynamically select trusted services [17] to support more general contextual data such as location and QoS. To cope with the high dynamism and unreliability of the web – that services may disappear, for example – additional adaptive capabilities are required. To deal with changing conditions while also ensuring that service-level agreements are met, processes must be self-optimising. And to deal with unreliable services, they must also be self-repairing. To meet these requirements, we make use of:

- semantic descriptions of functional and non-functional properties of services;
- parametric design techniques [18] for supporting the (re)configuration of services based on contextual factors; and
- event-based opportunistic reasoning techniques based on the use of semantic spaces [19, 20].

By reducing the complexity for creating new services out of existing ones, full support for the creation of services within the service web will play a very important role in the uptake of service technologies. We have therefore decided to approach the creation of services through both manual and (semi-) automated processes. As is explained in more detail in the next section, this provides better support to users creating services. The (semi-) automated creation of atomic services will also be supported by intelligent recommenders that analyse service documentations and suggest semantic annotations. Furthermore, to ensure scalability, the creation of composite services will be supported by applying parametric design over process templates. This avoids the computational complexity of planning techniques [21].

Finally, throughout their lifecycle, services will benefit from the presence of formal semantic descriptions that enhance interoperability between humans and machines as well as between machines and other machines. Conceptual descriptions will bridge the gap between the user and the service descriptions and support the integration of information through semantic web technologies such as RDF(S) [22].

5 Enhancing SOA with Web 2.0 Technologies

The adoption of Web 2.0 technologies and principles within the service web will accelerate the take-up of service technologies and make it easier to integrate the user within the overall architecture as an extremely valuable source of information and computational power. To achieve this goal, the semantics we use are kept lightweight. User interfaces are enhanced with recommender systems that make them more intuitive to use, and the distinction between providers and consumers is blurred.

The use of lightweight semantic descriptions reduces computational complexity, making it easier for users to achieve their tasks. In particular, we use WSMO-Lite [23] and MicroWSMO [24] for the annotation of WSDL and RESTful web services. These semantic descriptions will allow efficient and scalable reasoning about services and will reduce the complexity for users. Note that, in this sense, the project intends to be protocol-neutral, an important feature given the split between RESTful and WSDL-based web services. Figure 1 shows the usage of each as reported by ProgrammableWeb [25], a website that maintains a comprehensive directory of web Application Programming Interfaces (APIs) [26] and the protocol(s) they support.

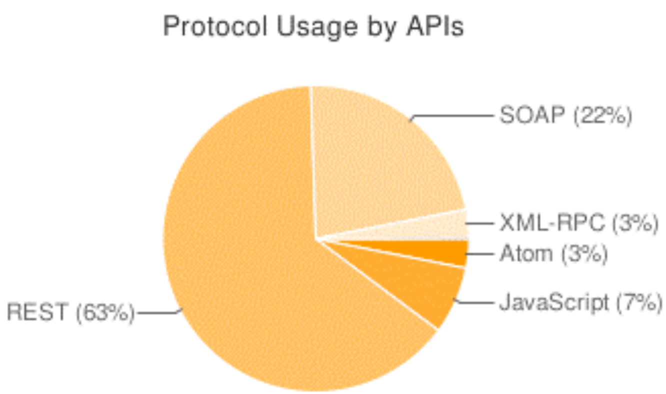

Fig. 1. Protocol Usage of Published Web APIs.

Users will be given the means to create lightweight service annotations by making use of recommender systems, but also through intuitive user interfaces such as mash-ups or pipes. Currently applied to data, these Web 2.0 solutions will be adapted to support the definition of simple service compositions and their publication. Through this simple mechanism, people will co-operate in building online communities around services. Simultaneously, such services will interoperate with one another to offer more sophisticated functionalities to the users in a largely automated way. Services will be combined in increasingly complex mash-ups that offer functions to support users both at home and work, helping them to perform their daily activities. This approach will blur the distinction between service consumers and providers, shifting the paradigm to one where users are active and are therefore contributing to the service web, rather than just being consumers of services. By incorporating human interaction and cooperation in a comprehensive fashion, tasks such as service ranking and mediation, that would otherwise be computationally expensive or even infeasible, can be addressed. In our view, the quality of the services available to the end-user can be increased significantly if it is transparent whether the 'engine' that abstracts services is completed by humans or machines.

Figure 2 summarises the impact of the four technological pillars discussed above on the service web of the future. By combining the semantic and social dimensions, Web 2.0 and semantic technology, we arrive at Web 3.0. By combining semantic technology and web services, we create semantic web services. And by combining Web 2.0 and web services, we enable the creation of service communities. Finally, as shown in Figure 2, when we combine all the advantages and enhancements offered by the four pillars, we arrive at the service web.

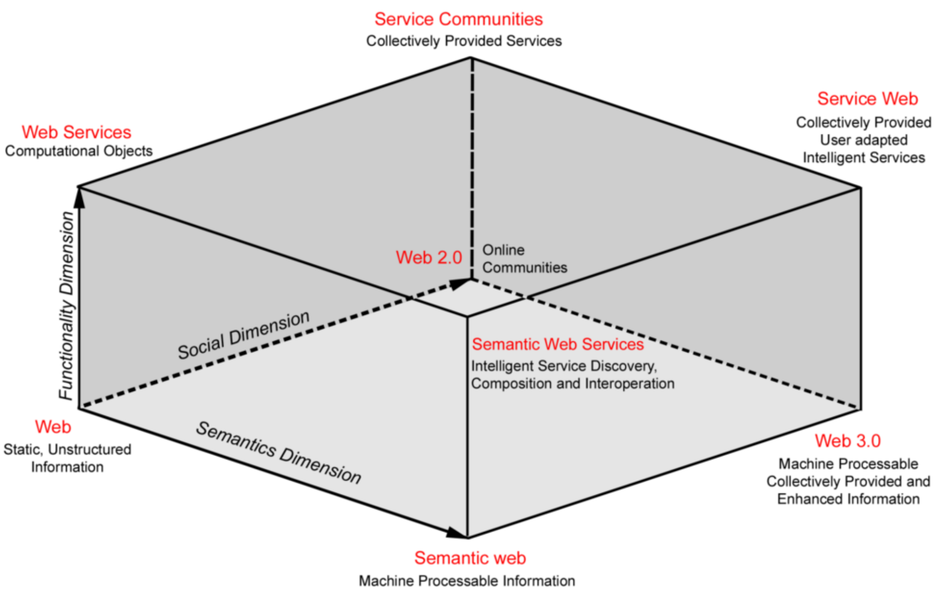

Fig. 2. Service Web technological pillars.

6 Application in Telecommunications: Telco 2.0 and Beyond

In this section, we will look at the potential impact of the emerging service web on the telecommunications sector. We start by setting out the business background, showing how the drive to an open service ecosystem is increasingly seen as inevitable and the implications for the telecommunications sector and beyond. We then explain the potential benefits gained by the deployment of SOA4All technologies in a specific telecommunications public SOA environment, BT's Ribbit services platform.

6.1 Business Background and the Drive to Open Service Ecosystems

With the increasing tendency of service providers of all types to publish services via the web and the emergence of Web 2.0 technologies, traditional telecoms companies (telcos) are being forced to evolve. Indeed, Figure 3 reveals that the telecommunications sector is among the leading areas for publication of functionality via the web. The key technological trends which demand telcos' immediate response come from Web 2.0 developments. Webcos – companies adopting Web 2.0 principles in their business models – are able to respond to changing demands and expectations in the marketplace by innovating at multiple levels. Their products are distinguished by the following key characteristics:

- **Web as platform**: The platform is no longer a specific server or application, but exists on the web ('in the cloud') and is characterised by the publication of capabilities rather than vertical applications (for example, the exposure by Google and Amazon of APIs via the web).
- **Architecture of participation (harness collective intelligence)**: designed to encourage users to take part, to share, to customise, to connect and even to participate in future product design (for example, user generated content – Flickr, YouTube, Delicious, mySpace, eBay, Amazon, social networks and user participative sites such as Wikis, blogs, Amazon reviews). The wisdom of crowds is a phrase used to refer to the harnessing of large numbers of end-user views and insights via these technologies.
- **Mash-ups**: as mentioned in the previous section, lightweight and rapid service/product composition in an open service ecosystem, leading to a larger number of applications developed more quickly and able to serve niche markets.
- **Long tail**: monetising the demand from the large number of highly diverse potential customers with non-typical requirements. Traditionally, telcos have focussed their efforts on a much smaller number of products and services addressing a larger market. In the Web 2.0 world, telcos can no longer ignore the revenues generated from the long tail.

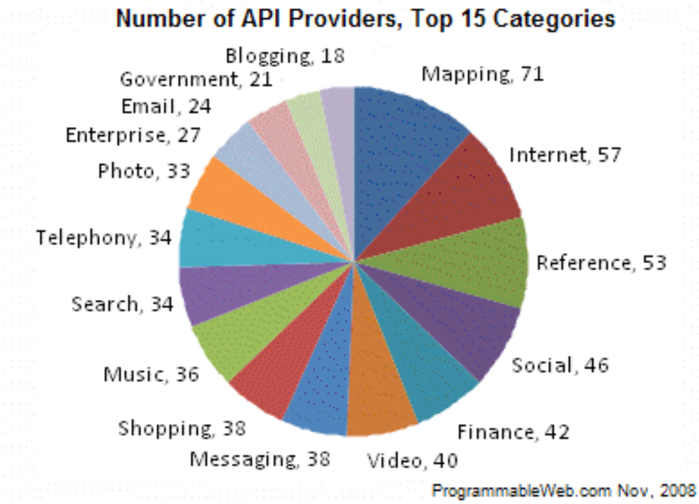

Fig. 3. Top 15 categories of Published Web APIs *(source [25])*.

Ribbit [27] is a wholly-owned US-based BT subsidiary and a voice-oriented web company ('webco'). Ribbit currently allows developers to consume voice services accessible via Adobe's Flex and Flash. The services that are exposed at the moment include voice calls, call routing, call management, third-party call control, voice and text messaging, speech-to-text, VoIP and contact management, with more to come in the near future. Currently, Ribbit users require detailed technical knowledge of the Flex and Flash programming languages in order to be able to access, combine and use web services.

In the SOA4All project, the use of contextual knowledge will support both the composition and provisioning of services in a customised manner. Contextual knowledge, includes, for example, knowledge about end-users and their preferences as well as about the previous experience of other users with respect to use of particular services. Using automation, we plan to shield service users from the complexities of creating such knowledge. BT will also take advantage of semantic descriptions of services in building (semi-) automated provisioning, composition, consumption and monitoring tools. The next-generation platform we envisage will also enable inclusion of third party services. In addition, it will address several key issues for BT's transformation, including: reducing time to market; enabling integration of third party services into BT's portfolio; increasing New Wave revenue and extending BT-wide SOA to the public web.

The possibility of increased competition from the 'Over The Top' (OTT) service providers – service and content providers that do not own the network they use – highlights a risk that telcos could become 'disintermediated' from the digital supply chain, consigned to the role of a commodity 'dumb' pipe provider. Webcos typically use advertising-based business models, whereas telcos collect revenues through usage-based billing. As these two sectors converge, the challenge for telcos is to reconcile the two different business models, finding ways to generate revenue from advertising, while continuing to offer billable services where appropriate. Finally, other changes are apparent such as the rise in virtual social networking, the roll-out of alternative access networks such as WiMax and the emergence of enterprise mashups that use Web 2.0-style applications alongside more traditional IT assets.

Considering all these aspects, by appropriately positioning themselves in the Web 2.0 world, telcos will continue to evolve and transform themselves from mere ‘dumb’ pipe providers to providers of ‘smart’ pipes (connections backed by QoS guarantees and service level agreements), platforms that support an open service ecosystem and a range of telco and third-party applications that run on such platforms. Telcos will not only need to create new services to address the needs of the long tail, but also allow third-party service providers to make use of telcos’ underutilised operations and business support systems capabilities to create new service offerings, thereby creating new revenue streams.

Figure 4 shows the long tail, representing untapped new business revenues, and the short head representing the conventional telco business revenues. New growth opportunities will arise from externalising current capabilities, allowing both the telco itself and third-party developers to address long tail demand by combining core services into more complex applications.

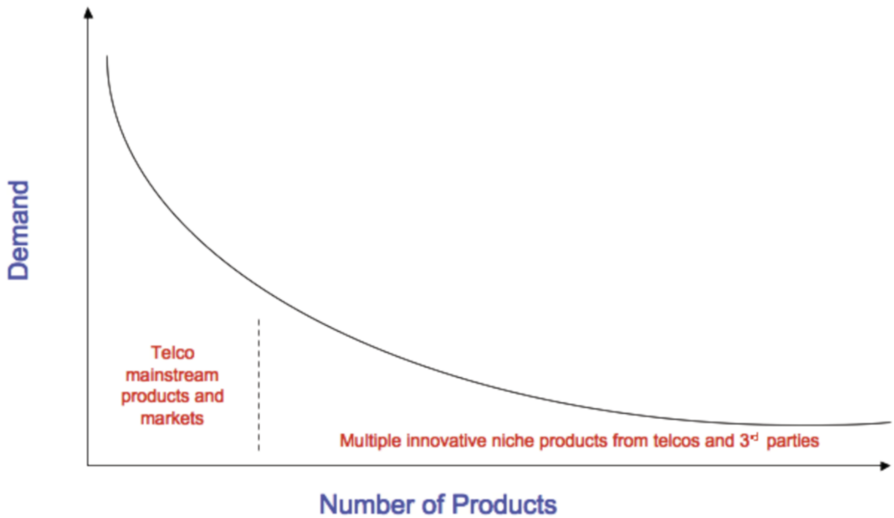

Fig. 4. Leveraging the long tail.

Telcos have historically been used to controlling the entire value chain from core network to end user. As discussed above, the long tail business model involves letting the market innovate, by using third-party developers to define how new services will be generated. Because this involves opening up the network, some loss of control is inevitable. In the future, the technologies employed will be telecom web services, representing an amalgam of telco services and internet services, enabling the telco to access long tail revenues via innovative niche applications from third party developers: the telco is taking the role of an aggregator.

6.2 Application of SOA4All Technology

The partners in SOA4All are investigating how BT – and, potentially, other telcos – can apply the technology being developed in the project's various core research areas to deliver next-generation web-based open services platforms. Below we describe the application of each technology area in turn.

Web 2.0

User-generated and maintained communities are an essential part of the case study, as well as encouraging ease of use and a low barrier to entry in utilising SOA. SOA4All technology based on Web 2.0 principles helps encourage this, specifically to:

- encourage users to work together to create new and innovative uses for BT services by providing an appropriate Web 2.0 community environment;
- provide facilities for users to share information about services and applications;
- allow users to tag, rate and comment on services for improved service discovery; and
- improve the usability of SOA, and facilitate the creation and management of new composed services, in a lightweight manner, with a low barrier to entry.

Related to the availability of Web 2.0-style tools for developers and users, the use of contextual knowledge will be key in supporting both the customised discovery and composition of services. The complexity underlying this will, however, be automated, so the process will remain hidden from the service user.

Service composition will use contextual knowledge about both the user domain, (for example, knowing the current user is an ISP targeting rural areas) and community knowledge (for example, knowledge of services tending to be used by similar ISPs) to suggest and customise solutions based on prior experience. As a result, the user will obtain more specific and more adapted proposals, which will simplify the selection of suitable services. This can be achieved by pre-defining each constituent service with a number of options, by parameterising certain functionality, and by providing a context-adapted service discovery and selection process according to pre-identified context dimensions.

Service provisioning will also be enhanced by the use of contextual knowledge. This will be achieved at run-time by integrating execution information, together with contextual knowledge such as user preferences and trust relationships (for example, which service providers does a given end-user trust). To do this, the service execution platform will integrate the information populated by the monitoring infrastructure with the contextual knowledge base, and will seamlessly invoke the context parameterisation engine as the need arises.

Semantics

BT will take advantage of the semantically-enabled improvements in provisioning, consumption and monitoring tools that are offered by SOA4All technology. Semantic descriptions for web services and goals will be built by BT and non-BT users in a straightforward way using purpose-built tools, allowing them to make available their services, or to discover and use the services of others, more efficiently. Similarly, semantics will also be important to enable new composition tools that will enable the creation of more complex services at lower cost.

Ontologies and semantic descriptions will form the basis for data, process and service representation in SOA4All. In the context of applying SOA4All technology to the future web-based telco platforms, telecommunications domain ontologies will also be used [28]. One example is NGOSS (Next Generation Operations Systems and Software), an industry-wide specification developed by the TeleManagement Forum [29] the purpose of which is to organise and guide the design and development of next generation operation systems in the telco domain.

NGOSS contains a set of frameworks, high-level architecture and methodology. It consists of four frameworks related to the different levels of looking at business:

- Business process framework (Enhanced Telecom Operations Map – eTOM)
- Enterprise-wide information framework (Shared Information and Data model – SID)
- Applications framework (Telecom Applications Map – TAM)
- Systems Integration framework (Technology Neutral Architecture – TNA)

Ontologies have been developed that capture telecommunication sector knowledge from NGOSS's standards. The SID model contains domain concepts related to market, product portfolio, customer, services, resources, the enterprise and supplier/partner, as well as common business terms called Core Business Entities (CBE) which are captured in the CBE Ontology (CBEO). The eTOM map defines a set of functional areas which serves as a reference classification for the business goals a process fulfils and which are captured in the business goals ontologies (BGO). The TAM map defines the typical IT systems map of telecommunication companies, and serves as a reference classification for a company's services map.

Services

As mentioned above, BT's current offering consists of Ribbit's services including voice calls, call routing, call management, third-party call control, voice and text messaging, speech-to-text, Voice over Instant Messaging (VoIM) and contact management, with more to follow shortly. As more services are made available, they will be included on the next generation platform.

In addition, there are also other business-to-business gateways and APIs that BT uses for interaction with its customers and suppliers, such as broadband provisioning [30] repair and diagnostic services. SOA4All technology also has the capability to describe these services in the same semantic framework (see, for example, [31]), allowing them to be combined with Ribbit services.

The third and final class of services are third-party services. One of the main aims of the case study is to promote the uptake of Ribbit by offering tools to encourage innovation in using and combining BT's services. SOA4All technology will make it easier not only to consume BT's services but also to combine them with other people's services to make new and interesting applications.

7 Summary

SOA4All will help to realise a world where a massive number of parties expose and consume services via advanced web technology. The outcome of the project will be a comprehensive framework and software infrastructure which will integrate complementary advances into a coherent and domain-independent worldwide service delivery platform. To achieve such a scalable and widely adopted infrastructure and framework, SOA4All stands on four main principles comprising: SOA, semantics, the web and Web 2.0. Our technologies will be proven in a number of real-world applications including the telco scenario outlined above.

From our point of view, we believe that the successful integration of semantic web and service-oriented technologies will form the main pillar of the software architecture of the next generation of computing infrastructure. We envision a transformation of the web from a web of static data to a global computational resource that truly meets the needs of its billions of users. Computing and programming will be positioned within a services layer thus putting problem solving in the hands of end-users through a truly balanced, cooperative approach.

Acknowledgements

This work has been supported by the EU co-funded IST project SOA4All (FP7-215219).

References

1. Cusumano, M.A.: The Changing Software Business: Moving from Products to Services. Computer 41 (2008) 20-27
2. Stollberg, M.: Scalable Semantic Web Service Discovery for Goal-driven Service-Oriented Architectures. Austria (2008)
3. Seekda home page. http://seekda.com (2008)
4. Benjamins, R., Davies, J., Dorner, E., Domingue, J., Fensel, D., Lopez, O., Volz, R., Wahler, A., Zaremba, M.: Service Web 3.0. DERI (2007)
5. SOA4All: SOA4All web site. http://www.soa4all.eu (2008)
6. Armstrong, J.: Making Reliable Distributed Systems in the Presence of Software Errors. (2003)
7. Erl, T.: SOA Principles of Service Design. Prentice Hall (2007)
8. Fielding, R.T.: Architectural Styles and the Design of Network-based Software Architectures. (2000)
9. Papazoglou, M.P., Traverso, P., Dustdar, S., Leymann, F.: Service-Oriented Computing: State of the Art and Research Challenges. Computer 40 (2007) 38--45
10. Martin, D., Burstein, M., J., H., Lassila, O., McDermott, D., McIlraith, S., Paolucci, M., Parsia, B., Payne, T., Sirin, E., Srinivasan, N., Sycara, K.: OWL-S: Semantic Markup for Web Services. http://www.daml.org/services/owl-s/1.0/owl-s.pdf (2004)
11. Fensel, D., Lausen, H., Polleres, A., de Bruijn, J., Stollberg, M., Roman, D., Domingue, J.: Enabling Semantic Web Services: The Web Service Modeling Ontology. Springer (2007)

12.Akkiraju, R., Farrell, J., Miller, J., Nagarajan, M., Schmidt, M.-T., Sheth, A., Verma, K.: Web Service Semantics - WSDL-S. http://www.w3.org/Submission/WSDL-S/ (2005)
13.Norton, B., Pedrinaci, C., Domingue, J., Zaremba, M.: Semantic Execution Environments for Semantics-Enabled SOA. it - Methods and Applications of Informatics and Information Technology Special Issue in Service-Oriented Architectures (2008) 118--121
14.Fensel, D., Kerrigan, M., Zaremba, M. (eds.): Implementing Semantic Web Services: The SESA Framework. Springer (2008)
15.Traverso, P., Pistore, M.: Automated Composition of Semantic Web Services into Executable Processes. 3rd International Semantic Web Conference (ISWC 2004) (2004) 380--394
16.Pedrinaci, C., Brelage, C., van Lessen, T., Domingue, J., Karastoyanova, D., Leymann, F.: Semantic Business Process Management: Scaling up the Management of Business Processes. Proceedings of the 2nd IEEE International Conference on Semantic Computing (ICSC) 2008. IEEE Computer Society, Santa Clara, CA, USA (2008)
17.Galizia, S., Gugliotta, A., Pedrinaci, C.: A Formal Model for Classifying Trusted Semantic Web Services. 3rd Asian Semantic Web Conference (ASWC), Bangkok, Thailand (2008)
18.Motta, E.: Reusable Components for Knowledge Modelling. Case Studies in Parametric Design Problem Solving, Vol. 53. IOS Press (1999)
19.Pedrinaci, C.: Knowledge-Based Reasoning over the Web. (2005)
20.Krummenacher, R., Simperl, E., Fensel, D.: Towards Scalable Information Spaces. In: Piskac, R., van Harmelen, F., Zhong, N. (eds.): New Forms of Reasoning for the Semantic Web, Vol. 291. CEUR-WS.org (2007)
21.ten Teije, A., van Harmelen, F., Wielinga, B.J.: Configuration of Web Services as Parametric Design. In: Motta, E., Shadbolt, N., Stutt, A., Gibbins, N. (eds.): EKAW, Vol. 3257. Springer, Whittlebury Hall, UK (2004) 321--336
22.Brickley, D., Guha, R.V.: RDF Vocabulary Description Language 1.0: RDF Schema. (2002)
23.Vitvar, T., Kopecky, J., Viskova, J., Fensel, D.: WSMO-Lite Annotations for Web Services. In: Hauswirth, M., Koubarakis, M., Bechhofer, S. (eds.): Proceedings of the 5th European Semantic Web Conference. Springer Verlag, Berlin, Heidelberg (2008)
24.Kopecky, J., Vitvar, T., Gomadam, K.: MicroWSMO. Conceptual Models for Services Working Group (2008)
25.Musser, J.: ProgrammableWeb: 1000 web APIs. http://blog.programmableweb.com/2008/11/03/1000-web-apis (2008)
26.ProgrammableWeb: Web 2.0 API directory. http://www.programmableweb.com/apis/directory (2008)
27.Ribbit home page. http://www.ribbit.com (2008)
28.Munoz, H., Pérez, N., Evenson, M., Szczekocka, E., Ksiezak, H., H.Kupidura, Belecheanu, R., Krysteva, P., Roman, D., Todorova, P., López, O., Cicurel, L.: Deliverable 8.1: Telecom Vertical Domain and Process Ontologies. SUPER (IST 026850) (2007)
29.Forum, T.: The TeleManagement Forum home page http://www.tmforum.org (2008)
30.BT Wholesale, B2B Gateway. http://www.btwholesale.com/pages/static/Applications/Orders/B2B_Gateway.html (2008)
31.Duke, A., Richardson, M., Watkins, S., Roberts, M.: Towards B2B Integration in Telecommunications with Semantic Web Services. Second European Semantic Web Conference, ESWC 2005, Crete, Greece (2005) 710--724