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

doi:10.3233/978-1-60750-007-0-183

From Software Services to a Future Internet of Services

Marco Pistore and **Paolo Traverso**

Fondazione Bruno Kessler
38100 Trento, Italy
[pistore,traverso]@fbk.eu

Massimo Paolucci and **Matthias Wagner**

DOCOMO EuroLabs
80687 Munich, Germany
[paolucci,wagner]@docomolab-euro.com

Abstract. Research in Service Oriented Computing has been based on the idea that software applications can be constructed by composing and configuring "software services", i.e., software utilities that can be used but that are not necessarily owned by consumers. A key aspect has however been dramatically underestimated in this research, namely the fact that – in most cases – software services are software components that provide electronic access to "real services" (e.g., a software service for travel booking allows us to access the actual service behind it, namely "the possibility of traveling"). Our claim is that the "Internet of Services" should focus on real services, rather than software services. In particular, we investigate the new role of Internet, which is a supporting infrastructure in the case of software services, but becomes a key enabler for real services, offering a unique capability to communicate in real time changes in real services and allowing for immediate reactions by service consumers. In the paper, we illustrate the project we are undertaking to demonstrate that Internet can become the service delivery platform of the future. We illustrate, in particular, the research challenges this vision produces in the areas of service usage, representation, engineering, and delivery, as well as the results we have already achieved.

1. Introduction and Motivations

So far, research on Service Oriented Computing [1] has been based on the idea that software applications can be constructed by composing and configuring "**software services"**, i.e., software utilities that can be used but that are not necessarily owned by users [2]. This idea has been posing interesting novel challenges for research, in that **software services are no longer under control of developers**. New paradigms are being investigated for the description of services and for the negotiation of Service Level Agreements between providers and users; novel methodologies and tools are being defined for engineering service-based systems – i.e., for selecting, composing and configuring software services – as well as for monitoring, managing and supporting the adaptation of services.

In the research on Service Oriented Computing, a key aspect has however been dramatically underestimated, namely the fact that in most cases software services are software components that provide an electronic access to **"real services"**. That is, we use a software service such as a travel booking service since it allows us to access the actual service behind it, namely the possibility of traveling. **The characteristics of**

real services are often very different from those of the corresponding software services. For instance, the duration of a software service (i.e., the time for booking a travel) is limited with respect to the duration of the real service (i.e., the actual travel). Software services are static and accessible anywhere and anytime, while the actual services are dynamic and context dependent, since they happen in the real world. Software services are rather independent and the constraints and conflicts among them are limited, while the actual services the user expects are usually related, and hence heavily constrained (it is easy to book two conflicting travels, but it is not possible to exploit both of them).

As a consequence, the concepts used to describe services and the approaches for "composing and configuring" them are radically different from those proposed for software services. It is insufficient to provide a technical description of the functional and non-functional aspects of services (interfaces, behavior, quality, security, and so on). The representation of real services must be based on a set of **key assets** that the services represent for their consumers and providers [3]. So, for instance, in the organization of a travel, aspects related to the duration and to the cost of the trip are much more relevant than the travel booking software service, since time and money are key assets we are well aware of. Such key assets have also the capability of providing those links between services that are neglected by software services. Indeed, assets such as time, money, social relations are fundamental in the organization of our lives, and are transversal to all the services we may exploit for our work, social life, free time, and so on. A similar situation also occurs in a business context, where the characteristics of the actual services (delivery, production, stocking …) are strictly related to the key assets of the company, and are hence much more relevant than the software interfaces that encapsulate them.

Our claim is that research on **Internet of Services should focus on real services** and on the key assets these services relate to. Indeed, Internet has a marginal role for software services – it is a convenient infrastructure for publishing, discovering, and executing software components. **Internet is instead a key enabler for "real" services**, in that it offers a unique capability to communicate to the user in real time the dynamicity of services and of their context (i.e., the cancellation of a flight can be immediately notified to the user), and to allow the user to react immediately to this dynamicity (i.e., re-scheduling the travel or looking for a train service). The effect will be similar to that in the traditional Internet, where the changes to the Web pages are seen immediately by everyone who connects to the Web: namely, the Internet of Services will allow users to live in and react to a world that changes at a higher and higher speed.

In this paper, we illustrate the project we are undertaking to demonstrate that Internet can become the service delivery platform of the future. The key ideas of the project, which we will describe in this paper, are: (1) to focus on real services instead of software services; (2) to describe these "real" services exploiting a small number of "core assets" that capture key concepts for service user and provider; and (3) to exploit the architecture of the Internet of Web pages we all know as a reference for the architecture of the future Internet of Services. The structure of the paper is as follows. In Section 2 we describe a motivating scenario based on [3]. In Section 3 we describe the project's objectives and approach. Finally, Section 4 concludes the paper.

2. Motivating Scenario: User-Centered Services

Mobile phones are becoming an essential tool in our life. They not only act as phones and media players, but more fundamentally they give us access to variety of services that we need in everyday life. A simple catalogue of such services includes those for travel activities (e.g., navigation and map services, ticket booking via mobile, SMS notifications of flight delays), social networking, personal assistance, entertainment, and so on. These services can be used to help us in managing a wide range of situations both in out private life and related to our business activities.

For example, even now it is possible to use the mobile phone to get information about movies by interacting with active posters equipped with RFID or 2D bar codes [4], to access Web services for booking movie tickets, and to exploit telco services for payment. We can store a movie event in the calendar and set up a reminder for it. We can share the information about the movie with the people in our contact list using telco services. And, finally, we can use navigation services to route us to the cinema.

Another example of the capabilities of mobile phones is when we receive an e-mail with an invitation to business meeting. We can save also this event in our calendar. If the meeting takes place in a different city or country, we can access Web services for organizing the trip (plan the itinerary, book and pay the travel tickets, make a reservation in a hotel, and so on). We can use the mobile phone to monitor and detect problems with our travel (for instance, Lufthansa provides a service that sends SMS notifications when our flights are being delayed or cancelled) and react to them (e.g., by re-scheduling the flight, or by informing the other participants of the meeting that we will not be able to attend). Finally, if we successfully reach our destination, we can receive information on how to reach the venue of the meeting, e.g., using local public transportation.

These scenarios rely on a set of already available services and applications (e.g., calendar, communications, map and navigation, context tracking, payment) that refer to different domains (traveling, personal activity management, entertainment, and so on), and show the possibilities offered by their composition. Unfortunately, while the number of available services is rapidly growing, each service is narrowly directed to solve a specific need of the user, and no attention is paid to how services may work together or to how the user may utilize these services in combination. As a consequence, at the moment the user is alone when he faces the problem of their composition.

- First, the user has to deal with different services, and consequently, with their specific interfaces, formats, and protocols.
- Second, the user has to manage by hand the composition of these services and information flow across them, copying data among services and “orchestrating” by hand the execution of these services.
- Third, the user has to continuously ensure the consistency of the information used by different services: if the user is not able to go to the movie, he has to take care of propagating the effects of this decision by means of relevant services, i.e., removing the event from the agenda, sending an alert to the friends, canceling the ticket on-line, and so on.

- Finally, it is entirely up to the user to identify conflicts and overlaps among the different personal activities that the user is managing through the calendar. For instance, the cinema and business trip scenarios, which we have presented independently, may be strongly interconnected: if the cinema and the business meeting are scheduled for the same day, it might be impossible for us to participate to both of them, or we may need to organize the travel to the meeting taking into account the constraints due to the cinema.

To solve these problems, we need to be able to compose and integrate a wide set of heterogeneous services, allowing the user to be in control, to be alerted when conflicts and inconsistencies emerge, and to be able to decide among alternative solutions. At the same time, we need to abstract away the technical aspects of the service implementations and the differences between the protocols and data formats.

We remark that, in the scenarios just described, there are of course software services – for buying tickets, for planning trips, for booking flights, and so on – as well as other "technical" services available on the mobile phone – for sending SMS, and emails, for managing the calendar and the appointments, and so on. The focus is however not on these software/technical services, but on the "real" services; that is, the scenarios are about being able to watch a movie, to attend a business meeting, to travel to this meeting, as well as about managing the overlaps and relations between all these activities.

3. Objectives and Approach

In order to investigate the vision of Internet of Services described in the introduction, and to address the scenario described in Section 2, FBK has recently launched a research project, which is being executed in collaboration with DoCoMo EuroLabs and other key partners in academics and industry. The ultimate goal is to demonstrate that Internet can become the (reference) service[1] delivery platform, and to realize all the necessary building blocks necessary for this demonstration. More concretely, the project pursues three kinds of objectives: (1) Theories, methodologies, and techniques that support different key aspects of Internet of Services. (2) A prototype platform that implements and integrates these theories, methodologies, and tools. (3) An experimentation of these results on key application domains.

The development of novel **theories, methodologies, and techniques** will be based on two main concepts, which we will better describe in the next sub-sections. First, the development of the Internet of Services will be based on a novel model of services which is based on "key assets". Second, the Internet of Services that we will build will have a structure resembling that of the Internet of Web pages.

3.1. Asset-Based Service Composition

Service composition in the Internet of Services requires managing a wide variety of services belonging to different domains – in Section 2 we have given some evidences

[1] From now on, by services we mean "real" services as described in the introduction.

of this. It is hence necessary to find the right concepts that allow for (1) describing all these services despite their high heterogeneity, and (2) allowing the user to be in control of the usage and composition of these services.

Addressing these two challenges at the same time is far from trivial. For instance existing Semantic Web Service approaches such as OWL-S [5] or WSMO [6] allow for managing a wide heterogeneity of services, but these techniques have been designed to provide a complete description of the services, of all their possible usages and of all their effects; as a consequence, the description of services they provide is too technical for allowing the user to keep control of the composition – higher level "semantic" concept are needed. Techniques like mash-ups [7] or Yahoo pipes [8] are nearer to what we need, but also in this case they are based on concepts such as data- and control-flows, which are still too technical and, once again, more adequate for software services that for real services.

Our claim is that, in order to be able to manage real services and to deal with the complexity described in Section 2, the concepts to describe the services should not relate to technical properties (inputs, outputs, data-flow, control-flow, and so on), but rather on the user assets that are affected by the services. In [3] we proposed the usage of a small set of core **"user assets"** for describing and organizing services. In particular we identified four resources:

- ***Time***, representing the temporal relation of user's activities, as well as the overlaps and conflicts among these activities;
- ***Location***, representing the (current and perspective) location of the user, the availability of services in these locations, as well as the necessity of traveling or moving to exploit services;
- ***Social relations***, representing other parties (family, friends, colleagues…) involved in the user activities;
- ***Money*** and other values, representing costs and assets involved in the user activities.

These assets are at the same time *enablers* for the usage of services and *constraints* for their exploitation. Indeed, being in a given location allows us to access services (e.g., entertainment services) that are available at that location, but it also forces us to organize travels in order to reach (services available only at) different locations. Our social network constraints our activities (e.g., obligations with our family reduce our freedom in undertaking other activities), but also provides help in case of problems (e.g., family or friends can help with activities we would not be able to undertake alone).

The mobile phone already provides well know applications for managing these resources, namely calendars, maps, contact lists, e-wallets, and so on. These applications, which are rather intuitive and easy to use also for a non-technical user, can be used to expose services to the user, and to let the user control the composition.

Consider for instance, the scenarios described in Section 2. Once the movie event has been stored in the agenda, we can use the calendar as the entry point for accessing additional actions and services that we can perform with the movie (see Fig. 1(a)), such as buying the ticket, looking to the trailer, plan the trip to the cinema, share the event, e.g., with our wife. When the beginning of the movie approaches (see Fig. 1(b)), the calendar in the mobile can alert us, navigate us to the cinema, or allow us to call a taxi. Similarly, when we add the business meeting to the calendar, the trip to the location of the meeting can be planned and booked through additional services accessible from the calendar. Moreover, the system can detect that there is an overlap between the travel and the movie (see Fig. 1(c)) and signal this problem to the user. If the user resolves this problem by moving the movie to a new date (see Fig. 1(d)), the system can take care of updating the reservation, and, in case this event is shared with our wife or with some friends, of notifying them the update, e.g., via SMS.

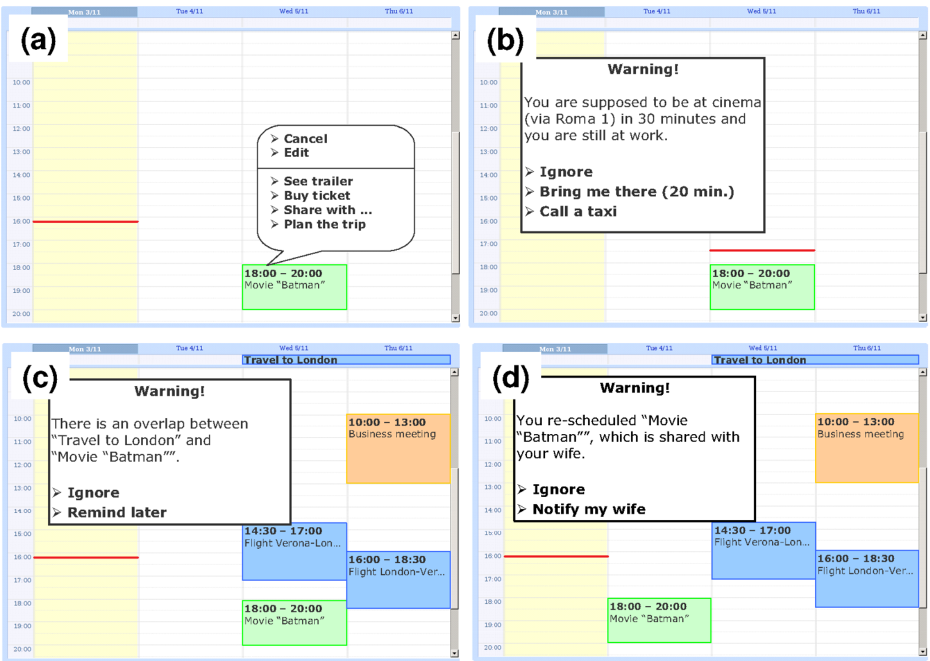

Figure 1. Service composition using the calendar.

This example shows how the time resource can be used to link and compose highly heterogeneous services such as those necessary for buying movie tickets, scheduling meetings, planning trips; and how the calendar can be used as a front end for controlling and orchestrating this composition. Similarly, we could use the other core assets – locations, social relations, money/values – for defining and identifying relations and compositions of services that are not time-related, and exploit the corresponding front-end tools – maps, contact lists, e-wallets – for managing these compositions.

We believe, however that the applicability of core assets as driving concepts for service description and composition in the Internet of Services is of broader applicability than the user-centered services available through a mobile phone. These concepts can indeed be applied also in business scenarios, even if in these cases the "assets" will be different, and specific to the specific business domain. So, for instance, in case of emergency management, a map of the territory provides the main view for planning and running emergency services – environmental sensors, reports on emergency events, location of emergency units, movement plans, and so on can all be represented, supervised, and "orchestrated" on this map. In the case of a financial company, the key assets and the main view to the business will probably be related to balances and charts of accounts; in case of manufactory, the production chain may offer the main view to the core assets. And so on.

3.2. Conceptual Architecture for the Internet of Services

In order to understand how to build the future Internet of Services, we will use as reference the current Internet of Web pages. More precisely, we will concentrate on four aspects that, in our opinion, contributed in a fundamental way to the success of the Internet we know, and we will investigate how these four aspects can be replicated in the Internet of Services. As shown in Figure 2, these four key aspects of the Internet of Web pages are:

- a powerful, easy-to-use **browser**;
- flexible tools for page editing and **content creation**;
- infrastructure tools (such as **Google**) for easing the access to the pages; and
- tools for modeling the content and knowledge in the Web pages (the **semantic web**).

In the following, we describe the corresponding aspects we foresee in the Internet of Services.

- **Asset-Driven Service Modeling.** In our vision, asset-driven service modeling will play the same role that semantic web plays in the classical Internet of Web pages [9]. Indeed, as shown in Subsection 3.1, assets capture the key aspects of services for provider and for consumer. Moreover, in our approach, assets are the glue among the different components on the Internet of Services. For this reason, a coherent and flexible modeling of assets is a fundamental element in the project. Novel methodologies and tools are needed to support the modeling of the key assets of services; it is also necessary to support the modeling and understanding of the processes and knowledge that these services need, and to associate these processes and knowledge to the service assets [10]. The methodologies and tools that we intend to adopt will allow the different actors involved in the provision and exploitation of services (e.g., service providers, end users, domain experts, knowledge engineers) to collaboratively work towards the modeling of asset-driven services. This, in spite of the different knowledge engineering and modeling skills of the different actors [11].
- **Service Delivery Infrastructure.** By "service delivery infrastructure" we mean all the mechanisms and tools that are between the service provider and the service consumer. In the classical Internet, this infrastructure includes for instance tools like Google, MySpace, YouTube, which facilitate finding contents relevant for us, or making our content easier to find. In the Internet of Services, the infrastructure tools need to address different purposes, the most important being **service composition**. In the case of software services, service composition is a design task performed by the service engineer (see, for instance, the approach described

in [12]). In the case of real services, the composition is dynamic, context-aware, user-centric, and asset-based; as a consequence, it stops being a design task, and becomes the most relevant functionality the deliver infrastructure should provide. New methodologies, techniques and tools are needed for this novel service composition which lives in the delivery infrastructure, and new reasoning services are needed which are able to take into account the heterogeneous knowledge used by service. In addition to service composition, the delivery infrastructure should support all the other operations necessary to use services. Namely, it should provide mechanisms and tools for the **enactment, monitoring, adaptation, management** of the delivered services.

- **Service Front End and User Interaction.** The goal is to design the equivalent of the browser for Internet of Services, that is, to develop a tool that allows the end user to access and exploit services. This requires the development of **intelligent interfaces** which support novel interaction paradigms, in particular in the mobile and situated settings; these interfaces will use the concept of service assets to drive the interaction of the user with the services. However, the effectiveness of the user interface depends not only on the quality of the technologies. The **user acceptance** of these technologies, in terms of usability, appropriation, personal and social impact, is as important, specifically if the goal is to deliver "real" services. We will exploit a value-based evaluation process [13] to assess user acceptance; in particular, we will assess the importance of using assets for modeling services for user acceptance.

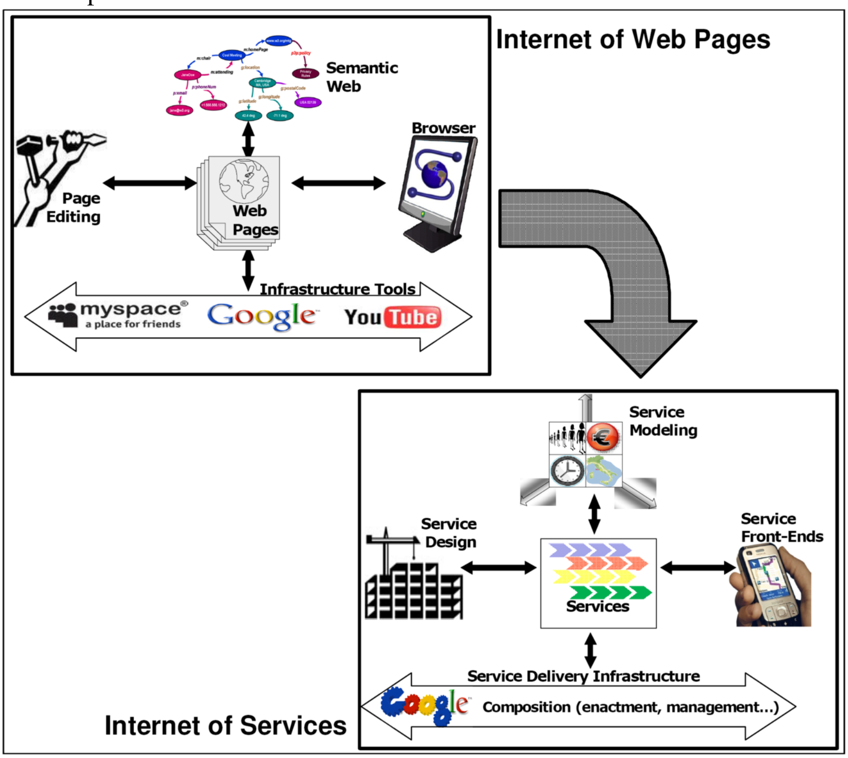

Figure 2. From the Internet of Web Pages to the Internet of Services.

- **Service Design and Engineering.** In the classical Internet, producing pages requires tools that are much more sophisticated than HTML editors: content managers, dynamic pages, Web application, Wikis, have been developed over time. Similarly, for the Internet of Services, there is a need for novel methodologies and techniques for designing and for managing the life cycle of the "real" services, as well as of the software components that encapsulate them. In particular, in order to be easy to monitor and to adapt to the specific user requirements and usage contexts, novel design and engineering principles and methodologies are needed, which should rely upon a deep understanding of user assets and requirements [14].

3.3. Integration and Evaluation

In addition to investigating these research challenges, the project will develop an **integrated platform** for the Internet of Services. This platform addresses two different requirements, namely to integrate the results of the different research lines just described in a coherent system, and to allow for a practical demonstration of the feasibility of the project vision.

Finally, the results of the project will be evaluated by applying the platform in two pilots based on different **application scenarios**. The first scenario is in the **user centered** domain**,** which concerns all applications where the services are used and combined according to the personal needs of (a certain class of) end-users. In this case, the person and her/his life is the center, the key assets are personal assets (like her/his personal agenda or social relations), and the services are organized and composed based on problems and opportunities related to such assets. The second scenario is in the **business centered** domain, where the focus is not the single user, but the definition of a complex business ecosystem based on the exchange of services among different types of actors. With respect to the previous one, in this scenario we not only have heterogeneous services, but we also have different actors with different key assets in which real services must be represented.

4. Concluding Remarks

In this paper, we introduced a project that we launched recently on the Internet of Services. This project relies on the key idea that, while most of the research in Service Oriented Computing has concentrated on "software" services, the future Internet of Services should focus instead on the "real" services that are behind this software. This shift requires re-thinking the current approaches for service representation, service engineering, service delivery, and service usage. In order to make it possible to address the new challenges deriving from this shift, in the project we adopt two hypotheses. First, we assume that a small set of core assets will be able to describe the key features of these "real" services, as well as to drive the service composition process, and to "orchestrate" the execution of these services. This hypothesis is based on previous results [3] obtained in the case of user-centric services accessible through the mobile phone, which show that "time" can be used as key asset in this context, and that the calendar can be used by the user to control the service composition. The second hypothesis is that the conceptual architecture of the Internet of Services will be similar to that of the Internet of Web pages, and hence we can learn from the key achievements in the latter in order to direct the research in the former.

The project is in its early stages, and most of the objectives and challenges described in Section 3 will be addressed in our future work during the five years of the project. During all our work, the validity of our hypothesis and the correctness and effectiveness of the results will be checked and challenged though the two application scenarios described in Section 3, but also thorough a continuous interaction and collaboration with the broader community of researchers and practitioners interested in shaping the future Internet of Services.

References

1. Papazoglou, M.; Traverso, P.; Dustdar, S.; Leymann, F.: *Service-Oriented Computing Research Roadmap*. 2006.
2. NESSI Strategic Research Agenda. Vol. 1. *Framing the Future of the Service Oriented Economy*. 2006.
3. Kazhamiakin, R.; Bertoli, P.; Paolucci, M.; Pistore, M.; Wagner, M.: *Having Services "YourWay!": Towards User-Centric Composition of Mobile Services.* FIS 2008.
4. Paolucci, M.; Broll, G.; Hamard, J.; Rukzio, E.; Wagner, M.; Schmidt, A.: *Bringing Semantic Services to Real-World Objects*. Int. Journal on Semantic Web and Information Systems 4, 2008.
5. *OWL-S Home Page*, http://www.daml.org/services/owl-s/2003/.
6. *Web Service Modeling Ontology (WSMO)*, http://www.wsmo.org/TR/d2/v1.2/.
7. Blake, B.; Nowlan, M.: *Predicting Service Mashup Candidates Using Enhanced Syntactical Message Management*. SCC 2008.
8. *Yahoo Pipes*, http://pipes.yahoo.com/.
9. *The NEON project*. http://www.neon-project.org/.
10. Di Francescomarino, C.; Ghidini, C.; Rospocher, M.; Serafini, L.; Tonella, P.: *Reasoning on semantically annotated processes*. ICSOC 2008.
11. Ghidini, C.; Rospocher, M.; Serafini, L.; Kump, B.; Pammer, V; Faatz, A.; Zinnen, A.; Guss, J.; Lindstaedt, S.: *Collaborative Knowledge Engineering via Semantic MediaWiki*. I-SEMANTICS 2008.
12. Marconi, A.; Pistore, M.; Traverso, P.: *Automated Composition of Web Services: the ASTRO Approach*. IEEE Data Eng. Bull. 31(3), 2008.
13. Not, E.; Leonardi, C.; Mennecozzi, C.; Pianesi, F.; Zancanaro, M.: *Beyond Usability: a New Frontier for User-Centered Design of "Future Internet" Services.* FIS 2008.
14. Penserini, L.; Perini, A.; Susi, A.; Mylopoulos, J.: *High Variability Design for Software Agents: Extending Tropos*. ACM Transactions on Autonomous and Adaptive Systems 2(4), 2007.