Composition of Interactive Service-based Applications by End Users

Abdallah Namoune[1], Usman Wajid[1],and Nikolay Mahendjiev[1],

[1] Manchester Business School, Booth Street West,
Manchester, M15, United Kingdom
{abdallah.namoune, usman.wajid, nikolay.mahendjiev}@mbs.ac .uk

Abstract. In this paper, we investigate web users' mental models of services, the underlying risks and benefits of service composition, and the problems anticipated while combining web services into final interactive applications. The study comprised three focus groups integrating group discussions and questionnaires, with a total of 35 participants, the majority without specialist programming skills. The results of the focus groups revealed a high degree of optimism towards service composition and consumption. However, several concerns, primarily related to personal privacy, trust, and technical difficulty, were highlighted during the focus groups. This paper discusses these concerns and proposes some ideas about how to address them.

Keywords: Web services, service composition, end user development, service-based applications.

1 Introduction

Service Oriented Architecture (SOA) technologies are becoming very popular on the Internet, especially in the form of independent services [1]. Their key benefit is reuse, indeed existing web services can be loosely coupled to produce new composite web services through the so called process of "service composition". Whilst only a small proportion of users, often with considerable computing knowledge and programming skills, can construct complex service based applications, the majority of online users are unable to exploit the advantages offered by SOA technologies and develop service-oriented applications tailored to one's needs. This difficulty can be linked to the complexity of the composition process which is carried out using advanced composition languages, and to the limited technical knowledge of ordinary users. In this respect, the research challenge lays in simplifying the composition process so that various services can be combined into interactive applications, and abstracting this process from unnecessary technical complexity. Such research promises to promote the consumption and reuse of web services, especially by ordinary web users. When creating such user-friendly service composition interface, we also need to consider user expectations regarding the trade-off between the costs of learning new tools and the benefits they expect to get from using them. For example, the spreadsheet interface hides aspects such as order of calculations and propagating updates, and minimises learning costs by using familiar metaphor of calculation tables and accounting books. The balance between costs and benefits is likely to differ for different groups of users and different target domains (e.g. [12,13]), yet we believe that identifying user attitudes and expectations towards service composition is a key to predicting successful uptake [12,13,21], hence it is the focus of the study reported in this paper.

Currently, end users can add web services as widgets/gadgets to their personal pages in a lightweight manner; this is particularly relevant to networking websites such as: Facebook [2] and personalized homepages such as: iGoogle [7] and myYahoo [16]. Users of these websites can select from a list of services and position them on their personal pages. The services are visually represented as independent windows and the users can interact with these services and customize their look and that of their personal pages. Although the widget-based model is simple and enables hosting different services together, it does not support service composition. Indeed, the web services, represented as widgets, are autonomous and do not interact with each other, thus restricting their usefulness for creating more complex assemblies. For instance, given a flight service, a car service, a hotel service, a card payment service, and an insurance service, users should be able integrate them to form a mini holiday organizer application. Service composition not only fulfils users' needs but also allows easy extension and customization of applications; thus, saving considerable time and resources.

Another advanced and rich approach to end user development of applications follows the mash-up based model. In this particular case, end users combine existing services and web feeds from multiple sources into a single web-based application using specialized mash-up editors, such as: Open Mashups Studio [18] and Yahoo!Pipes [23]. The major drawbacks of this approach relies in, firstly, the modelling skills needed to understand the data flow between services and secondly the strong emphasis on data aggregation while giving less importance to functionality aggregation.

Whilst the mash-up based model is complex and lacks flexibility, the widget-based model does not support any interaction between services offered by different service providers, This motivates the pressing need for more effective approaches to compose low-level services into interactive service-oriented applications by non-programmers. Easy to use and flexible service composition authoring tools that simplify the composition process should be offered. This is the main objective of the EC funded project, SOA4All [20].

Here we report on a study which aims to identify the balance between user expectations about costs and benefits of the SOA4All vision, and to chart users' concerns and background as relevant to this vision. It is worthwhile to note that this paper focuses on service composition and consumption by human actors and not by software agents. Focus groups were used as a self-contained method to conduct this study since no suitable prototype was available to evaluate at that stage. Focus group is an efficient technique used to collect qualitative data and generate concentrated information on a specific topic. It is argued to be better than user observation and individual interviews owing to the group interaction which provides detailed insights into opinions and experiences of participants [14].

This paper is organized into the following sections: Section 2 reviews the latest work on service composition. Section 3 provides a short description of the SOA4All project. Section 4 details the procedures carried out in the focus groups. Section 5 reports the findings of this research study. Section 6 presents a discussion about the findings and suggests various solutions to encounter the highlighted problems. Finally, Section 7 summarizes the paper.

2 Service Composition by End Users

Service Composition is broadly supported by two main approaches: workflow-based scripting of service components, and AI-based automatic composition of service components, reasoning with pre- and post-conditions. Further details are available elsewhere [8, 6, 19].

A large number of visual representations for service composition and interaction have been proposed with the purported aim to make the composition more user-friendly (e.g. Zenflow [10]). However, most of them are *ad hoc*, *i.e.* they use technology-led representations and metaphors, which are not derived from user studies. Only a few of them have been evaluated in terms of usability and cognitive effectiveness. For example, Lets Dance [24] has been evaluated using the framework of Cognitive Dimensions [4], but iterative testing and enhancement have not been documented in the related references. The framework of cognitive dimensions contains 14 principles describing aspects that are relevant to cognition [5]. It aims to evaluate the usability of interactive information artefacts (e.g. software applications) and non-interactive information artefacts (e.g. notations, programming languages) by non-specialists. Vitabal WS [9] is a version of an earlier visual language tuned to the needs of web service composition. It has been evaluated using the cognitive dimensions framework, yet it targets experienced web service developers and hence would have different characteristics from the service composition representations to be developed by SOA4All.

We believe that technology-led *ad hoc* visualizations will not work. Indeed opening up service use and development to people who are not professional programmers (we call them end users) requires the delivery of user interfaces that are task-oriented rather than technology-oriented, that is they should be tuned to the expected skills and foreseen tasks of our target users. Activities such as service construction and composition will involve non-trivial problem-solving in a context called *End User Development (EUD)* [22]. EUD research results provide an insight into the type of software interfaces and motivational factors likely to support end user activities.

Sutcliffe *et. al.* [21] see the trade-off between expected benefits and learning costs as a main determinant of uptake of an End User Development tool by its users. This has been extended to organizational context by Mehandjiev *et. al.* [13], who identify a number of risks and benefits for end users being involved with the development of software, including the construction of software services. These factors have then been used to underpin a number of quantitative studies in concrete domains, aiming to elicit the likelihood of uptake for end user development ideas in the specific context of that domain (e.g. [12]). The workshops reported here are an example of one such application of this approach to the target domains of SOA4All.

Several research studies have attempted to explore end user perception of software development, for example: McGill and Klisc [11] argue that end user developers of web development are aware of the associated risks and benefits and it is crucial to involve them in the development of approaches to minimise risks. Due to the difficulty of learning traditional programming languages, Myers *et. al* [15] report a number of studies aiming to elicit understanding of how people think about a particular task and design natural programming languages and environments that support the way end user developers are thinking. The generated data about user behaviour is used to build intuitive and usable programming environments. More recently, Namoune et. al [17] report on a user study in which potential problems of service composition are extracted when using a visual composition tool (although at its early stages of development). The main findings show that end users have difficulty connecting services together and understanding specialised service- related terms such as: operations, parameters, data types. Overall, review of available literature demonstrates that research in end user development of service based applications is very rare and most studies are in their infancy.

3 SOA4All

The research presented in this paper is a part of the ongoing work on SOA4All, an EC-funded project which aims to enable end-users from a variety of background to use web-services. In this respect, SOA4All aims at opening up services to the scale and accessibility typical for the WWW, and the surveys aimed to ensure that the tools produced by SOA4All will be accepted by the target groups of end users. This motivates our focus on acquiring end user perception of web-services, and then using this information to shape the tools and techniques produced by SOA4All.

SOA4All's approach to opening up services for everyone is based on the use of Web2.0 principles and state-of-art techniques for semantically tagging, retrieving and composing services. The developments on the technological front will result in addressing the specific needs of end users and allow them to implement innovative business models in order to address niche markets. In order to support the entire service lifecycle (service discovery to service consumption) SOA4All intends to provide a coherent and domain independent platform where a massive number of parties can expose and consume services. To facilitate in the development of such a platform, research within SOA4All involves clarifying the requirements as to how end users from a variety of backgrounds can not only interact with individual services but also compose different services to achieve their desired objectives. The requirement gathering process has been realized through a number of end user studies (focus groups) and the results of a subset of these studies are reported here.

The results obtained from the focus groups give a holistic view about the perception of target end-users. These results will be fed into to the design of SOA4All studio. SOA4All studio is envisioned as a rich web-based platform that will provide users with a unified view covering the whole lifecycle of services, including design-time, run-time and “post-mortem” analysis. It will provide the starting point for end-users when they commence to use SOA4All. In essence, the SOA4All studio represents a set of components to facilitate the composition of web-service based applications for novice users. The functionality offered by the studio will automatically help the end-users with the selection and placement of related web-services within the user interface.

The high-level view of SOA4All architecture is shown in Figure 1 (below):

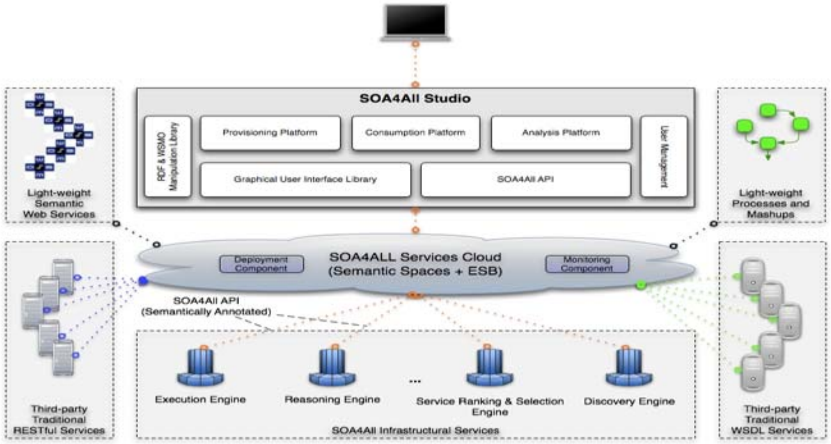

SOA4All Studio
RDF & WSMO Manipulation Library
Provisioning Platform
Consumption Platform
Analysis Platform
User Management
Graphical User Interface Library
SOA4All API
Light-weight Semantic Web Services
Light-weight Processes and Mashups
Deployment Component
SOA4ALL Services Cloud (Semantic Spaces + ESB)
Monitoring Component
SOA4All API (Semantically Annotated)
Execution Engine
Reasoning Engine
...
Service Ranking & Selection Engine
Discovery Engine
SOA4All Infrastructural Services
Third-party Traditional RESTful Services
Third-party Traditional WSDL Services

Figure 1: High level description of the SOA4All architecture.

4 Methodology

Three separate focus groups, involving 35 participants without programming skills (25 students and 10 academic and research staff) (range 19 to 40 years with a mean of 26 years) were undertaken within the Centre for Service Research at the Manchester Business School to acquire a better understanding of end users perception about web services, and the likelihood of uptake of user development. Each focus group lasted for approximately one hour; participant responses were recorded using audio recorders and questionnaires. The overall strategy was to first introduce participants to the topic of web services composition by end users through a presentation, followed by capturing their subjective judgment about the topic through a questionnaire, and finally discuss several issues in small groups. All participants were invited to perform these tasks:

1- Provide a definition of web services
2- Listen to a 20 minute presentation in which they were familiarized with web services and the concept of service composition; this was facilitated by examples
3- Fill in a service composition questionnaire

4- Discuss the potential risks and benefits of service composition and anticipate the composition-related problems; this was carried out in small discussion groups containing 5 participants each

5- Propose solutions to resolve the highlighted problems

4.1 Service Composition Questionnaire

The service composition questionnaire used in our study contains three main parts, as follows:

Part 1.

- My experience with Service Composition is (none 1-2-3-4-5 expert)
- I find web service composition interesting (disagree 1-2-3-4-5 agree)
- Please list the Service Composition languages and systems you are familiar with (or circle these examples: iGoogle, Facebook, Yahoo!Pipes, BPEL4WS, BPML, BPSS, OWL-S, WSCI, WSCL, WSFL, Semantic Pipes)
- How often do you compose services or build service based applications (daily – weekly – monthly – less often - never)
- What are your favourite service composition languages or systems?

Part 2.

Service composition by users (SCU)

- Is useful (disagree 1-2-3-4-5 agree)
- Is easy to achieve (disagree 1-2-3-4-5 agree)
- Brings about a more efficient way of conducting on-line activities (disagree 1-2-3-4-5 agree)
- Is unfeasible (disagree 1-2-3-4-5 agree)
- Is error-prone (disagree 1-2-3-4-5 agree)
- Can be used to break organisational rules and policies (disagree 1-2-3-4-5 agree)

Part 3.

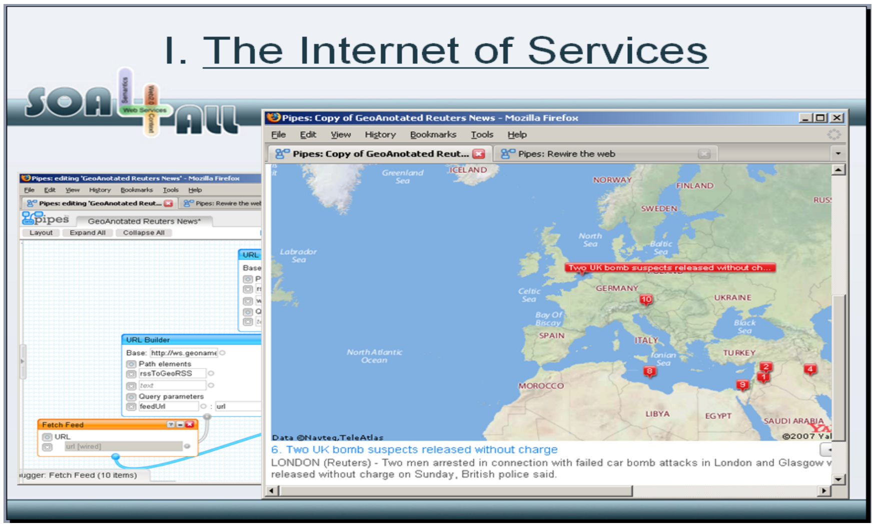
Please tell us your opinion about the following ways of encouraging and supporting Service composition by users (SCU)

- Examples of successful SCU can stimulate one to try it (disagree 1-2-3-4-5 agree)
- Recognising and rewarding SCU effort will make people more willing to try it (disagree 1-2-3-4-5 agree)
- Attending a training course could help people to start SCU (disagree 1-2-3-4-5 agree)
- SCU quality standards and testing will decrease risks (disagree 1-2-3-4-5 agree)

Although the questionnaire contains some questions which are difficult to assess at this stage, for example, it is practically hard to assess whether “composition is easy to achieve” without actually trying it, the principal aim was to drive first impressions about service composition and most importantly to check users’ acceptability of this innovative idea. In addition, the results will provide a reference point to advanced evaluation stages when end users perform composition using our composition authoring tool.

4.2 Introductory Presentation

The introductory presentation “The Internet of Services”, presented by one of the authors, aimed to introduce the concept of service and provide examples of service composition. It explained the difference between conventional services, software services and hybrid services, where human-performed services are enabled through software interfaces and services, such as buying a book through Amazon.com. The influence of current Web2.0 technologies was argued to enable end users to take part in the development of the web, and the idea is to move this influence to the internet of services. Following this, Yahoo! Pipes was used as a motivating example (Figure 2). Figures about the number of web services found were also reported (27.684 services and 7284 providers during the last 2 years), as suggested by the SEEDKA service crawler. Next, the motivation behind SOA4All was introduced to the attendees, with the project aiming to transform the current web of information into a web of services through which users of services could also become producers of applications, or what we call “Prosumers”.



I. The Internet of Services

Pipes: Copy of GeoAnotated Reuters News - Mozilla Firefox

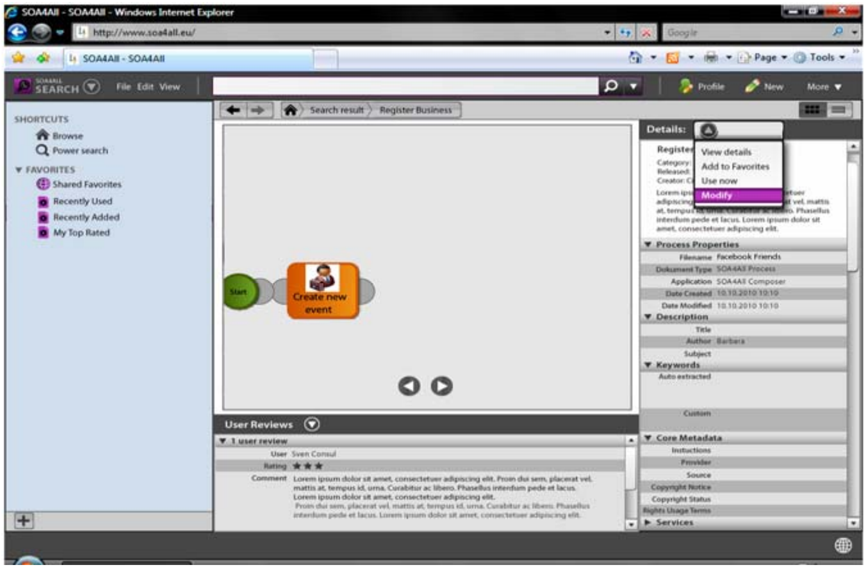
Pipes: editing 'GeoAnotated Reuters News' - Mozilla Firefox

6. Two UK bomb suspects released without charge

LONDON (Reuters) - Two men arrested in connection with failed car bomb attacks in London and Glasgow w released without charge on Sunday, British police said.

Figure 2: Yahoo! Pipes as a Stimulating Example

Then the scenario driving further discussions was introduced, the creation of a *Meet Friends* composite service. This hypothetical composite service allows a particular user to organise a meeting with friends at short notice. The Meet Friends composite service contains four services; service one fetches the address of friends from social networking sites (e.g. Facebook), service two finds out which friends are in the vicinity of the target venue, service three finds out weather and travel information for proposed meeting venue from a 3rd party, and service four sends out invites and directions using an SMS service. Finally, the presenter showed some mockups of a future authoring service composition tool (Figure 3). Participants were invited to ask questions related to aspects of the presentation before starting the focus groups.



Filename Facebook Friends
Dokument Type SOA4All Process
Application SOA4All Composer
Date Created 10.10.2010 10:10
Date Modified 10.10.2010 10:10
Title
Author Barbara
Subject
Auto extracted
Custom
Instructions
Provider
Source
Copyright Notice
Copyright Status
Rights Usage Terms

1 user review
User Sven Consul
Rating ★★★
Comment Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Proin dui sem, placerat vel, mattis at, tempus id, urna. Curabitur ac libero. Phasellus interdum pede et lacus. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Proin dui sem, placerat vel, mattis at, tempus id, urna. Curabitur ac libero. Phasellus interdum pede et lacus. Lorem ipsum dolor sit amet, consectetuer adipiscing elit.

Figure 3: A Mockup of the SOA4All Studio – a user-friendly composition tool under development in SOA4All

5 Results

The results of the three focus groups undertaken are divided into three main themes, as follows:

5.1 Web Services and Service Composition Perception

The pre-test questionnaires revealed that more than 85% of the participants considered themselves as not experts in terms of software and service development. 60% of the users specified that they have “never or less often” composed services or built service based applications. The qualitative analysis of the responses gathered in the focus groups showed that 25 user comments relate to service understanding. The results demonstrated diverse user understanding/definitions of services; these definitions varied between: features assisting users, solutions to issues, components of business process, offerings to customers, information provision, and execution of transactions. In general, users' definitions concentrated on two main aspects, (1) describing attributes/features of services such as: services are intangible and they have a back end, (2) describing specific interactions with users in the form of service consumption, such as: providing users with information, helping users, and delivering expertise.

Table 1. Service composition questions, rated between (1= disagree and 5= agree)

Service composition by users	**Mean answer**	**SD**
… I find web service composition interesting	4.20	0.76
… is useful	4.44	0.82
…brings about a more efficient way of conducting on-line activities	4.12	0.96
…is easy to achieve	3.32	1.19
… is unfeasible	2.26	1.18
… is error-prone	2.54	0.87
… can be used to break organisational rules and policies	3.50	1.08
Ways of encouraging and supporting Service composition by users		
Examples of successful SCU can stimulate one to try it	4.69	0.52
Recognising and rewarding SCU effort will make people more willing to try it	4.15	0.90
Attending a training course could help people to start SCU	4.38	0.77
SCU quality standards and testing will decrease risks	4.32	0.76

When asked whether service composition is interesting, 80% of users showed a high level of interest (mean = 4.20 /5, questions were rated on a five-point Likert scale where 1 corresponds to disagree and 5 corresponds to agree). Users also rated the usefulness of service composition high (mean = 4.44 /5), as well as the efficiency of service composition in promoting the accomplishment of online activities (mean = 4.12 /5). However, service composition by end users was regarded nor easy neither difficult (mean = 3.32 /5). In terms of error-proneness, fears were evident about the possibility of creating errors by ordinary web users (mean = 2.54 /5). Users concerns that relate to disruptive use of service composition (i.e. service composition can be used to break organizational rules and policies) were rated high (mean = 3.5 /5). Finally, 77% of the users disagreed or remained natural in regards to the question: "service composition by users is unfeasible" (mean = 2.26 /5).

In regard to user support, users agreed that successful examples (mean = 4.69) and training courses (mean = 4.38) could encourage people to be actively involved in the composition of services and development of service based applications. In summary, end users demonstrated a high level of interest and strongly agreed that service composition is useful and possible, but expressed uncertainty about the difficulty and potential misuse of service composition by the general public (Table 1).

5.2 Risks and Benefits

The discussion about the balance between risks and benefits is based on work [12,3,13,21] explaining the uptake of software development by end users (known as End User Development) as a rational economic decision based on the balance of perceived costs and perceived benefits of each user. The ongoing program of research in this area aims to analyse the factors which impact this perceived balance, and to discover organizational and technical strategies which aim to tip the balance in favour of the benefits, thus supporting the uptake of such technologies.

In terms of benefits, discussions in the focus groups mainly focused on the usefulness of reusing composition knowledge (40% out of all benefit responses), and the time users can save as a result of this (30% out of all benefit responses). Giving ordinary users control over service composition would empower them to produce various service oriented applications that can be tailored to their needs (15% out of all benefit responses), such as meta-search engines, thus saving them time and enabling them to obtain rich results.

In terms of risks, the biggest fear was about loosing control over personal information (8% out of all risk responses), especially when the effect is mediated through the effect of social interactions (e.g. your friends exposing information about you), or through the service provider (information aggregator), which may pass your personal information (e.g. phone number) to other sub-contracting services, which may or may not be bound to the data protection principles. Technical difficulty imposed by service compose was also amongst the biggest fears of end users (8% out of all risk responses). Errors in putting information together were also possible, especially when the composition is performed by inexperienced users and un-trusted third parties.

Moreover, users felt that services may no longer be there when they need them, and that any recommendation support for services may be biased to a set of services.

The participants also discussed what could be the social and organisational support for user-based service development. The following ideas emerged:

- “Go with the flow” – once everybody is doing it, people will join, mirroring success in other technologies;

- Non-trivial examples of successful use will also help (to sell benefits), this was felt quite strongly;

- Community-level control mechanisms such as feedback, etc. would ensure validation of services and, together with a validating body/watchdog may help to ensure the trust, which is considered vital for uptake of user-driven service composition.

5.3 Composition Problems

Although users favoured the idea of assembling services to formulate interactive applications that fulfils their daily needs, several service composition-related issues were raised, in particular:

- Services complexity: services are usually represented using their functional elements (operations and parameters) which are often not understood by ordinary web users.

- Services compatibility: users expressed frustration in regards to aggregating heterogeneous services from different service providers. How do they ensure the business services they are trying to combine together are technically compatible with each other?

- Composition steps: users agreed that it might be problematic to define the single steps required to combine services together and the order in which these services should be executed due to their lack of technical knowledge and skills. This issue becomes more complicated in the case of many services (for example: 100 atomic services).

- Other less aggravated user interface-related concerns evolved around the use of the service composition editor, for example: direct manipulation of web services (i.e. selection, deletion, etc) within the design space could be the main source of frustration.

In terms of technical support which can be provided by the composition editor, the following themes emerged:

- The difference between naïve and professional users was felt to lie partially in the awareness about the consequences of one's actions; this awareness should be supported;
- Full automation such as Google search results will frustrate owing to lack of control by the end users, a balance should be maintained;
- Tools should offer clarity of process in respect to building and using;
    - Context and personalization;
    - Reuse of designs.

6 Discussion

End users with no or little computing knowledge showed either no or basic knowledge of the technical aspects of services, i.e. they could not provide a technical definition of services. This result is expected as our target group has no specialist technical skills. Essentially, they perceived services as elements which deliver services (be it information, help, solutions … etc) to accomplish specified users goals. This view emphasises that services need to be abstracted from their technical complexity and presented in a way that efficiently describes their purpose/functionality, especially for ordinary web users.

Users showed a high likeability towards the idea of composing services into personalised interactive applications. This agrees with the current trends that end users are becoming proactive about developing the web. Users argued that service composition will save them time and enable them to develop applications on the fly and without the need to acquire considerable technical knowledge. Hence, it is important that end users are able to develop service-based applications without the need to learn programming languages and modelling notations.

To overcome the aforementioned problems, various tentative remedies that will form the functional requirements of a future visual service composition authoring tool –currently under development - are proposed in this section:

Promote service composition awareness: even though web users have experience adding autonomous services to their networking or personalised sites, the composition of services imposes a totally new and different challenge. Therefore, the composition editor should clearly communicate “the composition aspect” of services. Users’ awareness of the possibility to develop service-based applications should be elevated via the right amount of publicity to familiarize ordinary people with SOA technologies.

Simple service composition: this research aims to increase service reuse by ordinary users, it is therefore crucial to simplify service composition by hiding the technical aspects of services from users. Composition should be as easy as dragging and dropping a service into a design space, followed by creating connections between the selected services. No programming knowledge or expensive training should be required.

Guided service composition: users should be supplied with wizards, tutorials, and help messages to guide them through the composition process within an easy to use composition tool. This is particularly important to overcome the services compatibility and composition steps definition problems.

7 Conclusion

This paper reports on the results of three focus groups aiming to gauge end users’ perception of web services and their acceptability of service composition. Generally, users showed a high willingness to develop interactive service-oriented applications, but expressed fears that relate to the complexity underlying the composition process and to the knowledge required to build software applications. In future research, various composition design approaches of different complexity levels will be offered to accommodate end users with various skills and backgrounds within an easy to use online authoring tool, formally known as SOA4All studio.

Acknowledgments. This research work is supported by the EC funded project SOA4All. We would like to thank the students and academic and research staff of the University of Manchester for taking part in this study.

References

1. Alonso, G., Casati, F., Kuno, H., Machiraju, V.: Web Services: Concepts, Architectures, and Applications. Springer Verlag (2004)
2. FaceBook, http://www.facebook.com/, last accessed on 30th October 2009
3. Fischer G., Giaccardi E., Ye, Y., Sutcliffe A.G., and Mehandjiev N.: Meta-Design: A Manifesto for End-User Development. Communications of ACM, a Special Issue on End User Development (2004).
4. Green, T. R. G. (1989). Cognitive dimensions of notations. *In People and Computers V*, A Sutcliffe and L Macaulay (Ed.) Cambridge University Press: Cambridge., pp. 443-460.
5. Green, Thomas., Blackwell, A.: Cognitive Dimensions of Information Artefacts: A Tutorial, http://www.cl.cam.ac.uk/~afb21/CognitiveDimensions/CDtutorial.pdf (1998)
6. Hoffmann, J., Bertoli, P., and Pistore, M. Web service composition as planning, revisited: in between background theories and initial state uncertainty. In Proceedings of the 22nd National Conference on Artificial intelligence - Volume 2. A. Cohn, Ed. Aaai Conference On Artificial Intelligence. AAAI Press, 1013-1018 (2007)
7. iGoogle, http://www.google.com/ig, last accessed on 30th October 2009
8. Jinghai R, Xiaomeng Su. A Survey of Automated Web Service Composition Methods by: Semantic Web Services and Web Process Composition, Vol. 3387/2005, pp. 43-54 (2005)
9. Li, Karen Na-Liu. Visual Languages for Event Integration Specification. PhD Thesis, University of Auckland, Department of Computer Science (2008)
10. Martinez, A., Patino-Martinez, M., Jimenez-Peris, R., and Perez-Sorrosal, F. ZenFlow: A Visual Web Service Composition Tool for BPEL4WS. In Proceedings of the IEEE Symposium on Visual Languages and Human-Centric Computing. VLHCC. IEEE Computer Society, Washington, DC, 181-188 (2005)
11. McGill, T. and C. Klisc. End User Perceptions of the Benefits and Risks of End User Web Development. Journal of Organizational and End User Computing 18(4): 22-42 (2006)
12. Mehandjiev, N., Stoitsev, T., Grebner, O., Scheidl, S., Riss, U.: End User Development for Task Management: Survey of Attitudes and Practices. In Proceedings of IEEE Symposium on Visual Languages and Human-Centric Computing. Herrsching am Ammersee, Germany. IEEE Press. ISBN : 978-1-4244-2528-0 (2008).
13. Mehandjiev, N., Sutcliffe, A., Lee, D.: Organisational View Of End-User Development, in H Lieberman, F Paterno, and V Wulf, eds, End User Development, Human-Computer Interaction Series , Vol. 9, XVI, 492 p., Hardcover ISBN: 1-4020-4220-5 (2006)

14. Morgan, D. L.: Focus Groups as Qualitative Research. California, Sage Publications, (1997)

15. Myers, Brad, Pane, John F. and Ko, Andy, Natural Programming Languages and Environments. Communications of the ACM. (special issue on End-User Development). Vol. 47, no. 9. pp. 47-52 (2004)

16. MyYahoo, http://my.yahoo.com/, last accessed on 30th October 2009

17. Namoune, A., Nestler, T., and Angeli, A.D. End User Development of Service-based Applications. 2nd Workshop on HCI and Services at HCI 2009 Cambridge, (2009)

18. Orange Labs, Open Mashups Studio, http://www.open-mashups.org/ , last accessed on 30th October 2009

19. Papazoglou, M. P., Traverso, P., Dustdar, S., and Leymann, F. Service-Oriented Computing: State of the Art and Research Challenges. *Computer 40 (11)*, 38-45. DOI= http://dx.doi.org/10.1109/MC.2007.400 (2007)

20. SOA4All, http://www.soa4all.eu/ , last accessed on 30th October 2009

21. Sutcliffe, A., Lee, D., Mehandjiev, N.: Contributions, Costs and Prospects for End-User Development, Proceedings of HCI International, Lawrence Erlbaum Associates, Inc. New Jersey, USA (2003)

22. Sutcliffe, A. and Mehandjiev, N. 2004. Introduction: Special Issue on End User Development. The Communications of ACM, 47, 9, 31-32. (2004)

23. Yahoo! Pipes, http://pipes.yahoo.com/pipes/, last accessed on 30th October 2009

24. Zaha, Johannes Maria, Barros, Alistair P., Dumas, Marlon, ter Hofstede, Arthur H. M.: Let's Dance: A Language for Service Behavior Modeling. OTM Conferences (1) 145-162 (2006)