

User-driven Service Assembly (U-SAY)

Proposal for a **STREP** addressing

FP 7 Call Objective ICT-2007.1.2

Service and Software Architectures, Infrastructures and Engineering

Coordinator: Nikolay Mehandjiev, University of Manchester, Mehandjiev@acm.org

Technical Leader: Stefan Scheidl, SAP Research CEC Darmstadt, stefan.scheidl@sap.com

1. Vision and Rationale

People are increasingly exposed to using web services in a variety of contexts such as work, home, or leisure activities, and have to interact with this technology in a number of nontrivial ways. These interactions would often cross the boundary from setting parameters, such as the mode of payment, to activities which are in effect software development, such as devising information filtering rules in Yahoo Pipes, or even composing services to visualize different data sources side-by-side (“mash-ups”). These developments motivate the currently prevalent optimistic vision of an imminent future in which web services are written, composed and adapted by non-programmers, unleashing the creative energy which is driving the current user-generated web sites (Web 2.0).

The aim of U-SAY, like the aim of many currently running IST projects (S4ALL[5], OPUCE[6]) and many proposals in preparation [7], [8], is to bring us a step closer to this vision. What distinguishes U-SAY from these existing projects, however, is its concern with the actual problems faced by non-programmers when they develop software, and the development of software environments and tools to address these problems. U-SAY is building upon the theoretical advances and practical successes (such as spreadsheets and CAD systems) of a discipline called *End User Development (EUD)* [1], [2], [3], [4]. EUD appeared a quarter of century ago as a result of advances of PCs at the workplaces, its early optimistic visions drawing parallels between programmers and chauffeurs, suggesting user friendliness of software will soon increase to a degree where we would be able to “drive,” or develop our applications, without the help of specialists. This mirrors the optimistic mood of many current projects working in the area of “light-weight (or user-driven) service composition”. Taking stock of the progress a quarter of a century later, we are better positioned to see the complexity of the full picture. As stated by Fred Brooks, “there is no silver bullet”, because of the inherent complexity of software [9]. This is more so in a technically advanced domain such as Web Services, especially when we attempt compositions over and above the simple representation-layer nature of Web “mash-ups”.

EUD has achieved successes in important areas such as cognitive design of development/design environments, and AI tools providing automatic inference of user intentions. The two come together in the idea of Domain-Oriented Design Environments (DODEs). User-oriented DODEs will be the starting point of U-SAY, providing support for two other cornerstones: *formal mechanisms* to validate the quality of end-user service assemblies, and processes of *collaborative development* including users and programmers.

Formal mechanisms: EUD studies have demonstrated that the best domain-oriented end-user languages and metaphors are different from the technology-oriented control-flow and message-based models underpinning current web services. At the same time, we concur with Grady Booch that the future web of services should bring to the fore the millions of fine-grained software components (services with small “s”), in addition to the thousands of conventional large-grain Web Services (services with capital “S”) published today [10]. Both factors suggest that we should establish formal mechanisms enabling the discovery, consumption, assembly and publishing of a wider variety of software services which don’t fit the message-passing/RPC SOA model (such as the Yahoo-style filters), and their composition using alternative user-friendly mechanisms such as pipe-and-filter diagrams. This would require the formalisation of appropriate meta-data, including formal descriptions of the way services fit in service assemblies. This information will ensure the discovery of appropriate services and help users to avoid mis-configurations of service assemblies by using techniques from model verification and behavioural

validation. Using these formalisations, U-SAY will prepare a standardisation proposal to be submitted to W3C.

Collaborative development processes: The developments around “Web 2.0” and the experiences with EUD suggest a stronger focus on supporting the social aspects of service assembly in a manner similar to the patterns of collaboration found in OSS development. U-SAY aims to support the patterns of collaboration, negotiation and delegation which are frequently encountered within user groups and between users and developers [11], [12]. U-SAY will elaborate on possible support opportunities and provide the tools and meta-data necessary to support the following collaborative aspects of end-user service assembly: sharing of best practices, the emergence of super-users supporting others, delegation and collaboration with software developers.

This way, U-SAY will unite the three main strategies for supporting end user service assembly:

- to simplify access to service composition by providing domain orientation at the interface level using DODEs,
- to ensure the quality of the end user service assembly results by using formal service characteristics that support end users in discovering the appropriate services as well as in avoiding mis-configurations,
- to stimulate innovation and expertise sharing among users by providing an infrastructure for communication, collaboration and delegation between end users and between end users and professional developers.

2. Aim and Objectives

U-SAY’s **aim** is to develop the technology support needed to **enable end users to create or modify service assemblies** thus fulfilling their needs as service consumers and creators. This will include:

- *a new user-friendly service assembly model*, supported by
- *a domain-oriented design environment* and
- *a rich service description and discovery specification*, and embedded into
- *a method for collaborative software development* including users and software developers.

The following **examples** of service assembly applications are targeted by U-SAY:

- Configuration of a production process in a manufacturing or a chemical engineering company;
- Definition of business events triggering actions (such as an exceptional state of production triggering an alert).

The following **analogy** can be used to illustrate the aim of U-SAY:

- EXCEL is the paradigm of a domain-specific language for spreadsheet calculus;
- U-SAY aims at a comparable paradigm for end-user service consumption and assembly, i.e. programming.

To achieve its aims, U-SAY should address the following **challenges**:

- Service-oriented ERP systems employ a **variety of services** of different nature (e.g. data manipulation, event handling).
- This nature is relevant for
 - the **possibilities** and **constraints** for service assembly topologies (“technical concepts”), for example a service processing continuous streams cannot be combined with services processing singular events;
 - suitable **metaphors** (“application domain concepts”) enabling the end user to **understand** service assemblies (e.g. “pipe and filter” metaphor may be particularly suitable for the production domain but not for event composition; the “blackboard” metaphor may be suitable for event handling);
 - developing a shared understanding and shared expertise about effects and opportunities of alternative service orchestrations.

U-SAY will address these challenges by virtue of the following **objectives**:

- Identify and validate a set of suitable service composition **metaphors** and representations that can support end users in an arbitrary domain Identify a suitable **architecture description language** which can formalise the topology of connections between services in these user-oriented representations.
- Develop a **framework** for the development of **domain-specific service assembly environments** based on this architecture description language.
- Identify **socio-technical aspects** that limit or afford experience and expertise sharing as well as collaboration towards innovative service orchestrations, and suggest necessary functionality as well as its technological foundation in SOA structures.
- Explore possibilities and limitations of enriching services with suitable **metadata** regarding support for end-user service composition. This metadata will include the architectural suitability information based on the ADL formalisation, as well as information necessary to support the socio-technical aspects (recommendations, etc). This will be used to **extend existing standards** for service description and discovery (WSDL and UDDI).

The proposed project will produce the following **deliverables**:

- **Set of metaphors** and formal semantics validated by end users.
- **Programming language** (visual notation) with metaphors as primitives.
- Method for domain-specific adaptation of technical concepts.
- Prototype **development environment** with integrated design- and runtime.
- Set of functionalities to support interaction of user groups and user-developer communities in service assembly
- **Enhanced metadata structures** to inform the methods that implement the suggested functionalities
- Exemplifying **tools for collaboration support** in service composition (e.g. in the form of a network of services)
- **Validation** of the concepts found with the SME partners in the project

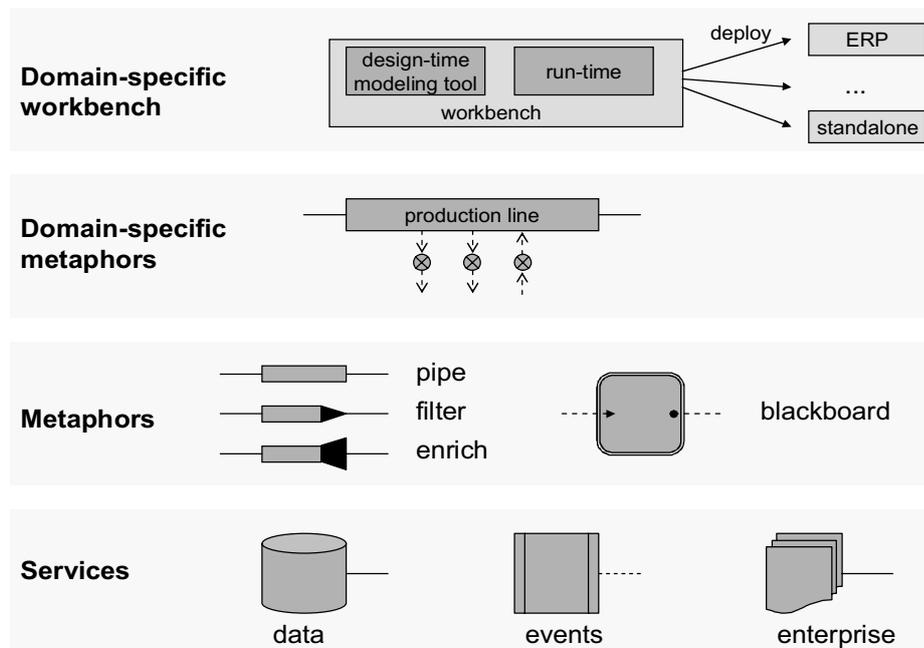


Illustration of conceptual layers

3. Relevance to the objective

Objective ICT-2007.1.2	U-SAY contribution
<p>Target Outcome (a) –</p> <p>1. Service architectures, platforms, technologies, methods and tools that enable context awareness and discovery, advertising, personalisation and dynamic composition of services.</p>	<p>U-SAY will</p> <p>1. provide a method, a technology and a tool to enable → service discovery, consumption, and assembly on the basis of a platform for domain oriented development environments supporting visual metaphors → a framework for the enrichment of services by semantic metadata on personal and community level</p>
<p>2. They should support flexible business models, provide for service management, and guarantee end-to-end quality of service.</p>	<p>2. propose new business models based on an ecosystem with the user community interacting with a consulting development community to achieve quality of service</p>
<p>3. They will cater for multiple component technologies and support vendor independence. Opportunities for standardisation should be exploited.</p>	<p>3. enable the procurement of services developed using different component technologies and architectural styles; and propose a new standard for service description and discovery, which includes architectural information and information supporting the new joint user-developer development</p>

<p>Target Outcome (b) –</p> <p>Service/software engineering approaches development processes, product lifecycle and tools for dynamically composed systems with dependable quality of service and reliability properties and promoting new open development paradigms with a higher degree of involvement of joint user and development communities.</p>	<p>U-SAY will</p> <ul style="list-style-type: none"> ■ provide an innovative service engineering approach using a collaborative process of joint user-developer service development and composition, allowing systems to be composed dynamically at the point of need with predictable dependability and reliability bringing a higher added value because of the opportunity to gain design feedback from the actual use of services via the end user community.
<p>Expected Impact – Improve competitiveness of enterprises and efficiency of organizations by</p> <ul style="list-style-type: none"> ■ Allowing the creation of dynamic services with guaranteed properties and new networked applications capable of interoperation across a wide variety of business domains and organisations of all sizes. 	<p>U-SAY service aggregation technology will</p> <ul style="list-style-type: none"> ■ furnish a framework integrating services from various application domains, thereby increasing interoperability of service aggregates and opening markets ruled by highly dynamic services also to small organisations without specific know-how in service technology
<ul style="list-style-type: none"> ■ Increased efficiency and productivity in software development and higher level of software reliability through novel service and software engineering tools and improved mastering of complex systems. 	<ul style="list-style-type: none"> ■ will realize an approach with a seamlessly integrating requirement specification, development and validation also for complex service aggregates. This will be done in conjunction with the ecosystem of collaborating user and development communities.
<ul style="list-style-type: none"> ■ New opportunities, notably for SMEs, through open and standard platforms and interfaces for: software and service development; middleware for resource sharing; and next generation operating systems. 	<ul style="list-style-type: none"> ■ The U-SAY proposals for extended UDDI and WSDL standards and related metadata enrichment middleware will foster resource sharing and leverage distributed computing to a new level.

4. List of Partners

- University of Manchester (UK)
- Siegen University (D)
- University of Trento (IT) [participation to be confirmed]
- SAP AG, SAP Research (D)
- Profactor (A)
- several SMEs (A, ...) [participation to be confirmed]
- HP Europe (UK or IT) [participation to be confirmed]

5. Work packages

- Project Management
- **Requirements discovery (suitable metaphors, representations and practices)**
- **Support for joint development activities and practices**
- **Support for architectural information in service discovery and assembly**
- **Design of DODE**
- **Integrated prototype development**
- **Prototype validation activities**
- Dissemination

6. Core Authors of this document:

- Nikolay Mehandjiev, University of Manchester
- Stefan Scheidl, SAP Research CEC Darmstadt
- Volkmar Pipek, University of Siegen

7. References

- [1] Mehandjiev N.; Sutcliffe A.G., Guest Editors, *Journal of Organizational and End User Computing*, vol. 18, no.4, pp. 43-65, ISSN: 1546-2234. Special Issue on "Technology Interaction Aspects of End User Development", 2006.
- [2] Lieberman H., Paterno F., and Wulf V., eds, *End User Development, Human-Computer Interaction Series*, Vol. 9 2006, XVI, 492 p., Hardcover ISBN: 1-4020-4220-5
- [3] Scheidl, S. Editor. *Proceedings of the Workshop on End User Development at Informatik'2006, Dresden, Oct 2006*,
- [4] Sutcliffe A.G., and Mehandjiev N., Guest Editors, *Communications of ACM*, a Special Issue on End User Development, September 2004
- [5] Jean, S. Sardes Project: S4All, funded by the ITEA program, information available from <http://sardes.inrialpes.fr/collaborations/collaborations/s4all.html>, last accessed 19th March 2007
- [6] OPUCE: Open Platform for User-centric service Creation and Execution, funded by F6 IST, information available from <http://www.opuce.tid.es/index.html>, last accessed 19th March 2007
- [7] Fensel, D. Service Web 3.0, FP7 project proposal. Presentation available from ftp://ftp.cordis.europa.eu/pub/fp7/ict/docs/events1-20070226-obj12-fensel_en.pdf, last accessed 19th March 2007

- [8] Cameron, A. GRIDWerk. FP7 project proposal. Presentation available from ftp://ftp.cordis.europa.eu/pub/fp7/ict/docs/events1-20070226-obj12-cameron_en.pdf , last accessed 19th March 2007
- [9] Brooks, Jr., F.P. 1986: "No Silver Bullet—Essence and Accidents of Software Engineering," Information Processing 86. H.J. Kugler, ed. Amsterdam: Elsevier Science Publishers B.V. (North Holland): 1069-1076. (Invited paper, International Federation of Information Processing (IFIP) Congress '86, Dublin, Ireland, September 1986.) Reprinted in Computer, 20, 4 (April 1987) 10-19. Also reprinted in The Mythical Man-Month, Anniversary Edition, Frederick P. Brooks, Jr., Addison-Wesley, 1995.
- [10]Booch G., Round Table Discussions, the University of Manchester, 22nd January 2007. unpublished
- [11] Pipek, V. From Tailoring to Appropriation Support: Negotiating Groupware Usage. Unpublished PhD Thesis, University of Oulu, Oulu, Finland, 2005
- [12] Wulf, V., Pipek, V., & Won, M.; Component-based Tailorability: Towards Highly Flexible Software Applications. Int. Journal on Human-Computer Studies, 2007, to appear