

# Towards Provenance Aware Design of Service Compositions: A Methodology for Analysing the Provenance Awareness in Service Designs

Paraskevi Zerva  
Department of Informatics  
King's College London, UK  
Email: paraskevi.zerva@kcl.ac.uk

Steffen Zschaler  
Department of Informatics  
King's College London, UK  
Email: szschaler@acm.org

Simon Miles  
Department of Informatics  
King's College London, UK  
Email: simon.miles@kcl.ac.uk

**Abstract**—Specifying and analysing *non-functional properties* (NFPs) is essential for driving architectural decisions and validating composite service designs. Only where NFPs have been specified can we choose between services with similar functionality that would better satisfy our non-functional requirements. Meanwhile, incorporating *provenance functionality* into service-oriented systems' design is becoming crucial for users, allowing them to query the generation methods and origins of the data the system outputs. This need is particularly evident in *compositions* of services, where audits of individual services do not provide a connected picture of the composition's processing history. Making provenance awareness (ability to answer provenance queries) an explicit NFP in composite service specifications would enable composite service designers to analyse whether they meet provenance-related requirements. In this paper, we discuss a framework for designing and analysing provenance awareness for service compositions. We envision this as a basis for analysing the impact of provenance on other NFPs such as performance and storage.

**Keywords**—formal specification; non-functional properties (NFPs); provenance awareness; service composition; Service-Oriented Architecture (SOA)

## I. INTRODUCTION

Service-oriented computing (SOC) [1] comprises both a software design philosophy and a system architecture, the service-oriented architecture (SOA), in which a collection of loosely coupled services communicate with each other through their published and discoverable interfaces. SOA aims to provide services which can be dynamically invoked and combined into new services by composing locally and/or remotely available services. Service composition is important when no single service is sufficient to satisfy all functional requirements. In cases where existing services have the same functionality but different non-functional properties (NFPs), the latter should be considered in order to best satisfy the overall non-functional requirements (NFRs). Composition according to NFRs is not a trivial task as there are dependencies between NFPs and properties of the execution environment that influence them. Where services are composed, the composition's NFPs will depend on those of the individual services composed as well as on details

of the composition. Formal modelling of NFPs and their dependencies is essential for enabling precise specification and analysis at design-time. Precise specification in NFPs is important, so that designers can ensure at design-time that their services will meet NFRs at run-time.

A relatively recent concern has been to provide answers to users' question's about the documented history of a service's processing and what process led to the data produced, the data's *provenance* [2]. In a service composition, *provenance questions* relate to the component service interactions during past execution, the data items exchanged and the dependencies between those items. Existing models and theories for NFPs do not support specification of provenance awareness, instead focusing on performance, reliability etc. A number of approaches have been proposed to implement provenance recording and providing provenance storage and querying functionality as part of service-oriented systems. However, this research does not address specifying provenance-awareness as part of a design, so that the provenance support can be analysed along with and in relation to other NFPs. Similarly, existing service specification languages do not provide a way to specify provenance awareness of composed services (see [3] for a detailed review). The latter is required as a formal basis for analysing the impact of provenance requirements on other NFPs during the composite services design process. We aim to address this gap by proposing a specification and analysis framework that will enable composite service designers to analyse at design-time in what ways a service-oriented system is provenance aware. This will allow designers to explore the relations and trade-offs between provenance awareness and other NFPs such as performance and storage.

The remainder of this paper is structured as follows. Section II presents our plans for our specification and analysis framework of provenance awareness based on a travel planner composite service scenario. We then present our perspectives and work for the future in Section III.

## II. RESEARCH GOALS AND APPROACH

In this section, we present our approach towards a specification and analysis framework of provenance awareness

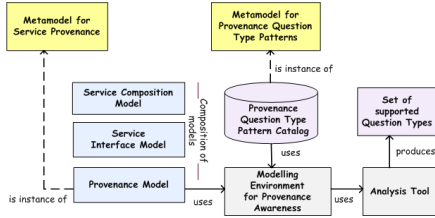


Figure 1. Specification and Analysis Framework of Provenance Awareness

for service compositions. We aim to provide a specification language for provenance awareness of composite services by identifying the set of required models, the *service interface*, *service composition*, and *provenance models*, and their interconnections, summarised in Figure 1. Those models provide concepts to specify 1) atomic and composite services (service interface and composition models) and 2) provenance data (provenance model). The *provenance model* of a composite service describes the provenance data that may be recorded and stored by executing that service. This model will be expressed as a templated provenance graph, inspired by an approach by Danger and Curcin [4]. Multiple provenance models (templates), specific to each composite service the service designers would like to model, will form instances given by a *service provenance meta-model*.

Finally, we require formal models of provenance questions. We abstract from these questions, defining a number of provenance question categories (*facets*) [3], which we plan to formalise as *provenance question type patterns*. Based on the latter, we will analyse a system’s provenance awareness, by checking at design time whether provenance data provided by the system (specified in the *provenance model*), is captured either by one or a composition of provenance patterns in our catalogue. We will build a modelling environment for provenance awareness, which will enable composite service designers to model their systems and then apply provenance-awareness checks. The result of this analysis would be the set of provenance question types that the composite service design can answer.

Figure 3 depicts a *provenance question type pattern* example for a particular *facet* [3] to identify service and provider identity. We use a simple scenario of composed services for booking a travel package (book a flight ( $t_1$ ), book a hotel ( $t_2$ ), book an attraction to visit ( $t_3$ )). An example provenance question for this scenario may be: were the providers of the atomic services of the travel planner composite service the same? In order to answer this, we need to analyse if the provenance data needed to answer the question, such as IP addresses of service providers and service names/URIs (captured by the pattern in Fig. 3), is provided by the composite service design. Those data cannot be simply provided through SLAs, as services may be dynamically selected at system runtime. The provenance model, depicted in Fig. 2 using a template extension to

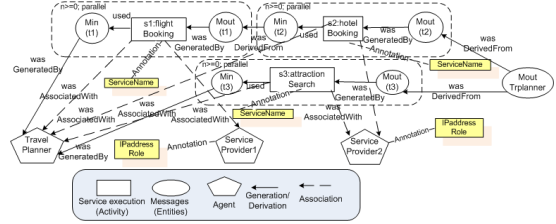


Figure 2. Provenance Model

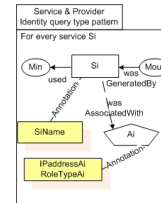


Figure 3. Provenance Question Type Pattern Example

W3C’s PROV, shows the provenance data that would be recorded by the composition.

### III. FUTURE WORK

In this paper, we have discussed our motivation and research methodology towards a provenance awareness analysis framework for composite service specifications. The next step towards achieving our research goals will be to formalize the *facets* through the means of provenance patterns. The challenge here will be to explicitly define those patterns based on the provenance data required to answer particular questions exhibiting those facets, along with the relations of provenance data captured by the different provenance patterns. In the future, we want to provide a number of such patterns to enable provenance-awareness analysis support for different provenance question types. We argue that composite service designers will benefit from the analysis results of our proposed framework, which can constitute useful feedback for balancing the trade-offs of provenance awareness with other NFPs they would need to consider for taking a final decision for their system’s design.

### REFERENCES

- [1] M. P. Papazoglou, P. Traverso, S. Dustdar, and F. Leymann, “Service-Oriented Computing: a Research Roadmap,” *Int. J. Cooperative Inf. Syst.*, vol. 17, no. 2, pp. 223–255, 2008.
- [2] S. Miles, P. Groth, S. Munroe, and L. Moreau, “PrIME: A methodology for developing provenance-aware applications,” *ACM TOSEM*, vol. 20, no. 3, pp. 1–42, 2011.
- [3] P. Zerva, S. Zschaler, and S. Miles, “Towards Design Support for Provenance Awareness: A Classification of Provenance Questions,” in *International Workshop on Managing and Querying Provenance Data at Scale (BIGProv’13)*, 2013.
- [4] R. Danger and V. Curcin, “D5.2 Application of the Provenance Model,” available: <http://www.transformproject.eu/>, 2012.