Chapter 9
Flexible Coordination Techniques for Dynamic Cloud Service Collaboration

Gary Creaner
Dublin City University, Ireland

Claus Pahl
Dublin City University, Ireland

ABSTRACT
The provision of individual, but also composed services is central in cloud service provisioning. The authors describe a framework for the coordination of cloud services, based on a tuple-space architecture which uses an ontology to describe the services. Current techniques for service collaboration offer limited scope for flexibility. They are based on statically describing and compositing services. With the open nature of the web and cloud services, the need for a more flexible, dynamic approach to service coordination becomes evident. In order to support open communities of service providers, there should be the option for these providers to offer and withdraw their services to/from the community. For this to be realised, there needs to be a degree of self-organisation. The authors’ techniques for coordination and service matching aim to achieve this through matching goal-oriented service requests with providers that advertise their offerings dynamically. Scalability of the solution is a particular concern that will be evaluated in detail.

INTRODUCTION
Service-oriented architecture (SOA) is an architectural style that allows for business processes to be implemented by integrating various services. These services can be thought of as software components. Cloud computing builds up on service architecture as the platform, providing cross-organisational, externally hosted services. Most current SOA implementations use web services as the technology platform based on message passing, a service registry and static service
Flexible Coordination Techniques for Dynamic Cloud Service Collaboration

description respectively. This approach is rigid in that it requires services publish the details of their functionality and how to interact with them to a registry. This information must then be used by the requestor to bind to and invoke the service in the way in which the provider has published, usually using WS-BPEL. This is a property that does not meet the flexibility requirements of cloud computing, in particular if flexible service brokering and mediation is required where offered and requested cloud services are matched dynamically through a cloud brokering service. Intermediaries such as brokers that bundle and customise offerings in response to dynamic needs will become more important in the cloud domain in the near future. Currently, a cloud service user or broker would have to completely define what services are to be used, the order in which they would be used and how the input and output is passed from one service to another in order to implement a full business process.

Current approaches to service collaboration (Pahl, 2002) are web service orchestrations and choreographies like WS-BPEL (orchestration) and WS-CDL (choreography). Both require the services used to be specified prior to the execution of the process. This kind of static process specification does not lend itself to a dynamic, flexible approach in which provided cloud services could be used as part of a process without prior knowledge of the service.

In order to overcome the limitations described, a framework is needed allowing services to be chosen dynamically at run-time. Our framework introduces a coordination space for providers to collaborate their activities in order to fulfill requests. The coordination space consists of a tuple space where requestors can deposit their requests and providers can take on requests according to their capabilities (Doberkat et al., 1992; Li & Parashar, 2005; Pahl et al., 2011).

Ontologies can be used to add semantic descriptions to web services (Pahl, 2005; Pahl, 2007). There are ontologies available which offer ways of describing services in terms of their functionality. These will be discussed and the way in which they could be used in the context of service matching will also be explored. Matching requested and provided web services is possible based on these ontological descriptions (Klusch et al., 2006; Sirin et al., 2003; Sycara et al., 2003; Nixon et al., 2007). However, these have not been integrated in a tuple space as their coordination platform. We integrate goal-based service matching into tuple space coordination in order to add flexibility and allow in-exact matches.

The proposed framework would change the service coordination model from a pull model to a push model, whereby requests are published to the coordination space and providers search for requests that they would be able to fulfill. This means that requestors would be able to focus more on the definition of their request rather than on the services that are provided to them.

The next section will introduce core technologies used and discuss related work. A use-case scenario will then be introduced. There, we will also discuss how a tuple space architecture could be used to implement a coordination space for web services. Afterwards, we will give some detail on how the matching of providers and requestors is performed. We discuss on the scalability tests which were performed on the architecture in order to evaluate such an approach. An evaluation of the work discussing possible limitations and ways of overcoming these limitations is also provided.

BACKGROUND

Tuple spaces are widely used to support coordination activities (Johannson & Fox, 2004; Li & Parashar, 2005; Nixon et al., 2007). We present their principles, an overview of the chosen platform, and some background regarding a semantic extension of tuple matching. We also review literature on service composition and coordination.