Dynamic Business Collaborations Through Contract Services

Surya Nepal, CSIRO ICT Centre, Australia
Shiping Chen, CSIRO ICT Centre, Australia

ABSTRACT

New applications have recently emerged within the domains of e-Health, e-Science, e-Research and e-Government that require the formation of dynamic collaborations between independent, autonomous business organizations for the duration of a project designed with a specific purpose. To successfully create and manage such collaborations, there is a need of a standard way to specify: (a) what resources are required, (b) who will contribute resources, (c) the type of access required to these resources, (d) agreement and obligations of the partners within the business collaboration, with the terms and conditions specified in the agreement, and (e) how to instantiate, maintain and terminate such business collaborations easily and in a well understood manner. The authors address these issues through the creation, negotiation and execution of an agreed electronic contract. First, this paper provides a framework for an electronic contract (e-Contract) by introducing a Web Service Collaborative Context Definition Language (WS-CCDL), which was developed in the context of dynamic business collaboration. Then, the authors illustrate its use with a universal (anywhere) connectivity service for a tele-Collaboration application in the context of e-Research domain. Both architectural design and implementation considerations are provided to highlight the feasibility and complicity of the technologies.

Keywords: Autonomous Business Organizations, Connectivity Services, Contract Services, Dynamic Collaboration, Télé-Collaboration

INTRODUCTION

Recently, there has been much interest in forming on-demand dynamic collaborations between multiple companies, who although competitors, wish to collaborate on occasion because of some mutual business interest (Yamazaki, 2004). For example, in the e-Research domain, a number of research institutes may formulate a joint project to solve a challenging scientific problem that is outside the capabilities of a single organization to solve. This project, as an example, requires the institutes to form a dynamic collaboration that lasts only for the duration of the project. Within this project and collaboration, each institution contributes its own unique resources to the collaboration that enable a platform for researchers towards finding the solution for the problem (Jrotka et al., 2006).
The idea of dynamic collaborations in this way, however, is not new. For example, the area of virtual organizations (Mowshowitz, 1994; Foster et al., 2001; Globus, 2011; King, 1991) explores mechanisms that enable entities from different organizations to collectively create a virtual enterprise for some mutual benefit. This goal is typically achieved through open service discovery, negotiation and execution based on Service Level Agreements (SLAs). In particular, the Grid computing community has made some important research contributions to the Virtual Organization (VO) literature (Foster et al., 2001). The Grid computing model for virtual organizations is based on resources that are usually intended to be “persistent”, i.e., open to discovery via the Internet in an open pool, and are expected to be available to an open pool of users on an ongoing basis for as long as required (Globus, 2011). While specific resources of the infrastructure (e.g., computational elements, storage systems, instruments and tele-Collaboration sites, etc.) may be taken offline or added on a dynamic basis, the resources (drawn from the pool) as a whole should remain available for as much as possible.

In contrast to the virtual organization model just described, there is a growing trend where virtual enterprises, built around their own transient business interests require their contributed resources to remain hidden from outside parties (Phillips et al., 2002; Handley et al., 2002). Resources are only revealed to a select group of participants. One example of this is the policy-based provisioning of resources, where resources are revealed only to trusted, collaborating partners by the agreed upon policies defined for the resources (Tsai et al., 2007). In situations such as this, open service discovery mechanisms cannot be used, as this resource information is deliberately hidden. Collaborations built around this concept are termed dynamic collaborations.

One of the key features of a dynamic collaboration is an on-demand contribution of resources from participating autonomous organizations. To establish such dynamic collaborations, all participants involved in the collaboration require a standard way to specify and/or agree: (a) what resources are required (Requirements); (b) who will contribute resources (Contributions); (c) how to access these resources (Access Polices). The participants may also have to explicitly add the corresponding terms and conditions to the technical specifications. Furthermore, contributed resources residing in different autonomous organizations bring a problem of interoperability between them, not the least of which includes differing access policies.

SOA concepts and Web Services technologies have proven to be successful in dealing with issues related to interpretability across autonomous systems. A stack of Web Services standards, such as WSDL, UDDI and SOAP, are defined to provide a platform-independent way of implementing services. Recently, resources such as storage and networking infrastructures, tools, software and data are implemented using Web Services technologies so that they too can be made available as services. For example, a concept of Software-as-a-Service (SaaS) is introduced for software (Ma, 2007) and Infrastructure-as-a-Service (IaaS) (Nepal et al., 2007) for storage and networking infrastructure. Therefore, we envisage that it is possible to define and share resources as services in the context of dynamic collaborations.

From the above discussion, we reach a conclusion that a contract that combines both technical specifications and legal declarations is a natural starting point to establish such dynamic collaborations. An electronically executable version of such a contract, whose notations and semantics are defined to be machine interpretable, is called an electronic contract (e-Contract) (Radha Krishna et al., 2005). The concept of an e-Contract, however, is not new. It has been used as a powerful concept (as well as a data model) for controlling and automating Business-to-Business (B2B) collaborations. The literature of e-Contract has been spread across various aspects and functionalities of e-Contract, including e-Contract representation (XML, ebXML, ECML, tpaML, RuleML).
Software Change Impact Analysis: An Approach to Compute and Prioritize Impacted Functions in Software Systems
www.igi-global.com/article/software-change-impact-analysis/126637?camid=4v1a

Control Architecture and Controllability
(2017). Model-Based Design for Effective Control System Development (pp. 75-108).
www.igi-global.com/chapter/control-architecture-and-controllability/179499?camid=4v1a