FAULT TOLERANT SERVICE INTEGRATION

Filipe Campos and José Pereira
Department of Informatics
University of Minho
Campus de Gualtar, 4710-057 Braga, Portugal
E-mail: {fcampos, jop}@di.uminho.pt

ABSTRACT

In the current economic environment, we face a continuously growing global market, which implies that several business areas are vulnerable to fluctuations, whether cyclic or not, such as the recent rise in the prices of the barrel of oil.

Due to this kind of unpredictable conditions, companies must adapt, in the best possible way, to ensure their survival. An intrinsic goal of a company is to make money and to be profitable. With this in mind, production costs must be reduced to the slightest value possible in order to maintain a financial margin that suffices to ensure the competitiveness of a company. Specially, in industry, production floor environments are getting more and more automatized, reducing the human interaction to the point of being almost invisible.

The growing interest in Service Oriented Computing (SOC) and Enterprise Application Integration (EAI) in a diversity of environments, and specially in production floor environments, has translated into a number of widely accepted standards ranging from data formats and the messaging infrastructure to a standard services portfolio which can be used to achieve interoperability among the many heterogeneous systems in existence. In that sense, new advances should try to:

- Take advantage of the existing standards;
- Aim at producing effectively useful protocols that attract the interest of companies, and, consequently, of standardizing consortiums or entities (Osrael et al. 2007).

A particularly interesting area of research and development is the coordination and composition of multiple services, or multiple instances of services, to achieve complex behaviors or provide additional guarantees. For instance, service coordination using the ubiquitous transaction processing paradigm has been addressed by the WS-AtomicTransaction and WS-BusinessActivity standards.

WS-AtomicTransaction and WS-BusinessActivity are both specifications to be used with the extensible coordination framework defined in the WS-Coordination specification in order to coordinate Web Services. WS-AtomicTransaction provides the type of coordination of atomic transactions that can be one of the following three agreement protocols: completion, volatile two-phase commit and durable two-phase commit. These protocols are to be used in applications that require consistent agreement on the outcome of short-lived distributed activities that have the all-or-nothing property. WS-BusinessActivity provides the coordination type for business activities through two specific agreement coordination protocols, BusinessAgreementWithParticipantCompletion and BusinessAgreementWithCoordinatorCompletion, that can be used to build Applications that require consistent agreement on the outcome of long-running distributed activities and, consequently, must deal with more complex failure scenarios than those that can be implemented using simple atomic transactions.

However, service integration in a production floor environment normally has particularly demanding dependability and timeliness requirements, that are not entirely satisfied or even compatible with transactional processing, and consequently, these standards. Although these requirements have traditionally been addressed by fault-tolerance techniques, which unfortunately rely on non-interoperable and mostly closed proprietary middleware solutions, this situation does not invalidate the study of WS-AtomicTransaction and WS-BusinessActivity, to analyze their functioning and how they exploit WS-Coordination (Campos and Pereira 2008).

A few attempts to address this gap exist, but cover only a very small subset of the fault-tolerance techniques and fall short as a general interoperability solution. The main challenges arise from the complexity of many fault-tolerant solutions, such as a view synchronous group communication protocol, but also from the subtle impact of service decomposition on the assumptions of such algorithms. In fact, one can even consider decomposing the major building blocks

of group communication themselves. Namely, gossip-based dissemination and consensus protocols, which do not provide fault tolerance and reliability guarantees for themselves.

A gossip-based communication protocol is inspired by the form of gossip in social networks, and also in the way viruses spread in a biological community, hence also being known as epidemic protocols. This kind of protocols provides a way to spread messages to a whole system and also to process acknowledgments in a distributed fashion, in order to avoid network congestion and nodes to be flooded with acknowledgments (Kermarrec and van Steen 2007).

Consensus is a fundamental problem in distributed systems to achieve fault-tolerance(Guerraoui and Schiper 2001), and it encloses the task of agreement among a set of processes or nodes in the presence of faults. Every process in the group proposes a value and the goal of a consensus protocol is for all correct processes, which are processes that don't fail during execution, to agree by choosing the same value that was previously proposed. Consensus has also been shown to have an impossible solution in some models of distributed systems.

Alongside with the evolution and increase of computing capabilities of devices, there is also a continuously growing trend of migrating processing tasks from servers and mainframes to these devices. The standardization of the Devices Profile for Web Services (DPWS) promotes interoperability, since it enables service-oriented communication to be performed between completely heterogenous devices located at a production floor environment without limitations due to the available computing power. But the interoperability is not only enable among devices, and the services they host, but also between these devices and the existing SOC systems. This enables the crumbling of the existing frontier separating Manufacturing Environment Systems (MES) and Enterprise Information Systems (EIS), allowing for more agile and harmonious integration within a single company, or even among several cooperating ones (Nguyen and Savio 2008), independently of the heterogeneity in computing power or complexity of the systems.

The goal of this project is to advance the state of art in service integration in a production floor environment by:

- Leveraging existing fault-tolerant protocols in a Service Oriented Computing context. This requires matching theoretical assumptions to actual environments; assessing the feasibility and adequacy of known protocols; and decomposing them into interoperable services.

- Addressing the impairments to fault-tolerance arising from Service Oriented Computing, namely, from composition strategies and very large number of components. This requires precise characterization of problems; and proposal of novel protocols to address them.

- Take full advantage of the standards provided by the DPWS stack to achieve fault-tolerance through gossip communication and consensus mechanisms built using protocols that must be simpler than existing heavyweight standards, such as WS-Coordination, for instance.

REFERENCES

Campos, F. and J. Pereira. 2008. “Gossip-based Service Coordination for Scalability and Resilience”. *Proceedings of the 3rd Workshop on Middleware for Service Oriented Computing (MW4SOC'08).*

Guerraoui, R. and A. Schiper. 2001. “The Generic Consensus Service”. *IEEE Transactions on Software Engineering* 27, No.1 (Jan), 29-40.

Kermarrec, A.-M. and M. van Steen. 2007. “Gossiping in distributed systems”. *ACM SIGOPS Operating Systems Review.* 41, Issue 5 (Oct), 2-7.

Nguyen, D. and D. Savio. 2008. “Exploiting SOA for Adaptive and Distributed Manufacturing with Cross Enterprise Shop Floor Commerce”. *Proc. of the 10th International Conference on Information Integration and Web-based Applications & Services (iiWAS2008).*

Osrael, J.; L. Froihofer; K. Goeschka. 2007. “On the Need for Dependability Research on Service Oriented Systems”. *Fast Abstract Proc. of the 37th Int. Conference on Dependable Systems and Networks (DSN'07).*

AUTHOR BIOGRAPHIES

FILIPE CAMPOS was born in Barcelos, Portugal and went to the University of Minho, where he studied systems and informatics engineering and obtained his degree in 2006. He worked for a year at Nokia Siemens Networks before starting his Ph.D. research in 2007 in the University of Minho. His e-mail address is : fcampos@di.uminho.pt and his Web-page can be found at http://gsd.di.uminho.pt/members/fjoc.