Chapter 7
Service Discovery Framework for Distributed Embedded Real-Time Systems

Furkh Zeshan
COMSATS Institute of Information Technology (CIIT), Pakistan

Radziah Mohamad
Universiti Teknologi Malaysia

Mohammad Nazir Ahmad
Universiti Teknologi Malaysia

ABSTRACT

Embedded systems are supporting the trend of moving away from centralised, high-cost products towards low-cost and high-volume products; yet, the non-functional constraints and the device heterogeneity can lead to system complexity. In this regard, Service-Oriented Architecture (SOA) is the best methodology for developing a loosely coupled, dynamic, flexible, distributed, and cost-effective application. SOA relies heavily on services, and the Semantic Web, as the advanced form of the Web, handles the application complexity and heterogeneity with the help of ontology. With an ever-increasing number of similar Web services in UDDI, a functional description of Web services is not sufficient for the discovery process. It is also difficult to rank the similar services based on their functionality. Therefore, the Quality of Service (QoS) description of Web services plays an important role in ranking services within many similar functional services. Context-awareness has been widely studied in embedded and real-time systems and can also play an important role in service ranking as an additional set of criteria. In addition, it can enhance human-computer interaction with the help of ontologies in distributed and heterogeneous environments. In order to address the issues involved in ranking similar services based on the QoS and context-awareness, the authors propose a service discovery framework for distributed embedded real-time systems in this chapter. The proposed framework considers user priorities, QoS, and the context-awareness to enable the user to select the best service among many functional similar services.

INTRODUCTION

According to the Internet of Things vision (Fleisch et al., 2005), the majority of devices will soon have communication and computation capabilities which users will apply to connect, interact and cooperate with the surrounding environment. In this dynamic environment, service-based systems will provide a good groundwork for a new type of real-world aware applications. In such an environment, efficiency will depend on the heterogeneous networked embedded devices and the challenge is how to discover the best real-world services for their integration into applications.

Heterogeneity in terms of programming language, operating platform and data management standards restricts the ability of devices to interact with each other: if two devices speak different languages, have different operating platforms or data management standards, then the data provided by one device cannot be interpreted correctly by the other. Therefore, in the dynamic environment of devices, heterogeneity restricts the understanding of the capabilities of the devices and the proper discovery and use of these devices.

In this regard, Service-Oriented Architecture (SOA) with the help of semantic web technology provides the base on which to properly address these restrictions as it enables different devices to work together by exposing their functionalities to others as services. The combination of the semantic web and ontologies (that present the concepts in a formal way by eliminating the terminological heterogeneity and enable the use of reasoning tools for knowledge discovery) allows the binding of data semantics along with the data for ease of sharing and correct interpretation.

In order to develop dynamic, flexible, distributed and cost-effective applications, service-oriented computing can be used. The web services can handle the complexity and heterogeneity with the help of ontology. Services are the entities which enable users to access the capabilities through pre-defined interfaces in accordance with the policies and constraints which are part of the description of that service (Estefan et al., 2008). Services are platform-independent and can be accessed through the internet. The most significant aspect of the web, due to which the overheads of companies have reduced and business is flourishing, is its role as facilitator in service outsourcing (Medjahed et al., 2003; Tsur et al., 2001). The service deployment model can be applied to any application component in order to make it a service. Services are well-defined, self-described and reusable software components that can be used over the web using the most silent and stable technologies such as the SOAP communication framework, Web Services Description Language (WSDL) and Universal Description Discovery Integration (UDDI) (which provides a mechanism to clients to find services (Bellwood et al., 2002)) (Curbera et al., 2002). A service is a set of related functions that can be accessed through programming over the web (Tsur et al., 2001). The key feature of the web services is that they are loosely coupled, allowing ad-hoc and dynamic binding and reusable software components and the key challenge is to find (discover) the best service for solving the particular problem.

With an ever-increasing number of functionally similar web services in the central repository, merely functional descriptions of web services are not sufficient for the discovery process. Therefore, we need Quality of Service (QoS) descriptions of services as an additional set of criteria to select the best service among many similar functionality services. However, the traditional UDDI lacks QoS descriptions, so it is difficult to rank the similar services by their functionality only. To solve this problem, some researchers have tried to add the QoS information during the service discovery process (Ran, 2003; Tran et al., 2009; Yao et al., 2008). However, the syntactic descriptions of QoS are not adequate, since the service providers and requesters may use different concepts, scales

Related Content

Applying AHP for Collaborative Modeling Evaluation: Experiences from a Modeling Experiment
Denis Ssebuggwawo, Stijn Hoppenbrouwersand Henderik A. Proper (2013). *International Journal of Information System Modeling and Design (pp. 1-24).*
www.irma-international.org/article/applying-ahp-collaborative-modeling-evaluation/75462

Review of Fault Tolerance Frameworks in the Cloud
Ajay Rawat, Rama Sushiland Amit Agarwal (2020). *International Journal of Information System Modeling and Design (pp. 79-99).*
www.irma-international.org/article/review-of-fault-tolerance-frameworks-in-the-cloud/259390

Components and Frameworks in the Cloud Era
Dino Konstantopoulos, Mike Pinkertonand Eric Braude (2012). *Software Reuse in the Emerging Cloud Computing Era (pp. 51-69).*
www.irma-international.org/chapter/components-frameworks-cloud-era/65167

Security in Service-Oriented Architecture: Issues, Standards and Implementations
Srinivas Padmanabhuniand Hemant Adarkar (2005). *Service-Oriented Software System Engineering: Challenges and Practices (pp. 292-316).*
www.irma-international.org/chapter/security-service-oriented-architecture/28960

A Preliminary Study on Adaptive Evolution Control Using Rank Correlation for Surrogate-Assisted Evolutionary Computation
Yudai Kuwahata, Jun-ichi Kushidaand Satoshi Ono (2018). *International Journal of Software Innovation (pp. 59-72).*
www.irma-international.org/article/a-preliminary-study-on-adaptive-evolution-control-using-rank-correlation-for-surrogate-assisted-evolutionary-computation/210455