Chapter 1.11
Peer-to-Peer Service Discovery for Grid Computing

Eddy Caron
Université de Lyon, France & CNRS – ENS Lyon – INRIA – UCBL, France

Frédéric Desprez
INRIA Rhône-Alpes, France & CNRS – ENS Lyon – INRIA – UCBL, France

Franck Petit
INRIA Rhône-Alpes, France & CNRS – ENS Lyon – INRIA – UCBL, France

Cédric Tedeschi
INRIA Sophia Antipolis – Méditerranée, France

ABTRACT

Within distributed computing platforms, some computing abilities (or services) are offered to clients. To build dynamic applications using such services as basic blocks, a critical prerequisite is to discover those services. Traditional approaches to the service discovery problem have historically relied upon centralized solutions, unable to scale well in large unreliable platforms. In this chapter, we will first give an overview of the state of the art of service discovery solutions based on peer-to-peer (P2P) technologies that allow such a functionality to remain efficient at large scale. We then focus on one of these approaches: the Distributed Lexicographic Placement Table (DLPT) architecture, that provide particular mechanisms for load balancing and fault-tolerance. This solution centers around three key points. First, it calls upon an indexing system structured as a prefix tree, allowing multi-attribute range queries. Second, it allows the mapping of such structures onto heterogeneous and dynamic networks and proposes some load balancing heuristics for it. Third, as our target platform is dynamic and unreliable, we describe its powerful fault-tolerance mechanisms, based on self-stabilization. Finally, we present the software prototype of this architecture and its early experiments.

1. INTRODUCTION

Any device in a computational grid provides some computing abilities (software components, scientific computing libraries, binaries …). They are for instance able to multiply matrices. Offered to the community, these abilities may be called *services*. Users around the world, may need to multiply some matrices while being unable to make it locally, for instance because the multiplication program is missing. It is then required for the user to use such a service in a remote mode. A critical prerequisite is the discovery of such a service. In other words, the user needs information about how to logon to a device providing the sought service (*e.g.*, protocol, address, ports), and how to use it (encoding of the matrices, location of the result …). The notion of service was generalized by the SOA reference model (Newcomer & Lomow, 2004), which is an attempt to define a standard architecture for using business computing entities. SOA is introduced as a paradigm for organizing and exploiting distributed capabilities that may be under the control of different ownership domains. The SOA standard describes a service as a mecha-nism to enable access to one or more capabilities, where the service is accessed using a prescribed interface and exercised consistent with constraints and policies as specified by the service description. This description is what the users are looking for. What is missing is a directory of services the user could consult to find what it needs.

The actual implementation and integration of a P2P service discovery system into computational grids require a set of protocols allowing servers, clients and the service discovery system to commu-nicate. Figure 1 illustrates the whole architecture: servers declare their services (the BLAS library (Dongarra *et al.*, 1990), complex simulations …). A client needs a service running under a Linux system and requiring 1 Gb of memory.

One important problem is to make such a service discovery possible within platforms emerg-ing today, which are large (gathering a high number of geographically distributed nodes), heterogeneous (in terms of hardware, operating systems, and network performance), and dy-namic (processors are constantly joining and leaving the system). Moreover, in such large environments, the number of services and requests

Figure 1. Integration of a P2P service discovery service to a computational grid: servers declare their services through a registration (application-level) protocol, clients express their needs through queries and the system sends a response (for instance, the address of a server satisfying criteria given inside the query) through a discovery (application level) protocol.
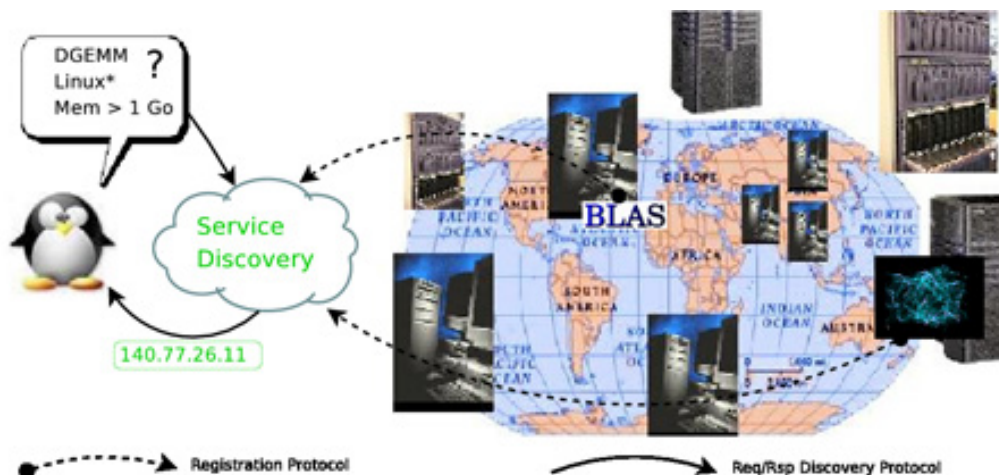
Related Content

QoS-Based Job Scheduling and Resource Management Strategies for Grid Computing
Kuo-Chan Huang, Po-Chi Shihand Yeh-Ching Chung (2012). *Grid and Cloud Computing: Concepts, Methodologies, Tools and Applications (pp. 1315-1331).*
www.irma-international.org/chapter/qos-based-job-scheduling-resource/64541

Providing Quantitative Scalability Improvement of Consistency Control for Large-Scale, Replication-Based Grid Systems
Yijun Lu, Hong Jiangand Ying Lu (2009). *Quantitative Quality of Service for Grid Computing: Applications for Heterogeneity, Large-Scale Distribution, and Dynamic Environments (pp. 91-111).*
www.irma-international.org/chapter/providing-quantitative-scalability-improvement-consistency/28272

Integrating Production Automation Expert Knowledge Across Engineering Domains
Thomas Moser, Stefan Biffl, Wikan Danar Sunindyoand Dietmar Winkler (2011). *International Journal of Distributed Systems and Technologies (pp. 88-103).*
www.irma-international.org/article/integrating-production-automation-expert-knowledge/55423

Optimizing Communication for Multi-Join Query Processing in Cloud Data Warehouses
Swathi Kurunji, Tingjian Ge, Xinwen Fu, Benyuan Liuand Cindy X. Chen (2013). *International Journal of Grid and High Performance Computing (pp. 113-130).*
www.irma-international.org/article/optimizing-communication-for-multi-join-query-processing-in-cloud-data-warehouses/102760

Security in Transnational Interoperable PPDR Communications: Threats, Requirements and Architecture Solution
Ramon Ferrús, Oriol Sallent, Cor Verkoelen, Frank Fransen, Keld Andersen, Christian Bjerrum-Niese, Jaakko Saijonmaa, Claudia Olivieri, Michel Duits, Anita Galin, Franco Pangalloand Debora Proietti Modi (2016). *International Journal of Distributed Systems and Technologies (pp. 41-60).*
www.irma-international.org/article/security-in-transnational-interoperable-ppdr-communications/168576