Chapter XV

A Framework for Intelligent Service Discovery

Robert Bram
Monash University, Australia

Jana Dospisil
Monash University, Australia

ABSTRACT

The claim of improved efficiency and reliability of networking technology provides for a framework of service discovery, where clients connect to services over the network based on a comparison of the client's requirements with the advertised capabilities of those services. Many service directory technologies exist to provide this middleware functionality; each with their own default set of service attributes that may be used for comparison and each with their own default search algorithms. Because the most expressive search ability might not be as important as robustness for directory services, the search algorithms provided are usually limited when compared to a service devoted entirely to intelligent service discovery.

This paper proposes a framework of intelligent service discovery running alongside a service directory that allows the search service to have available a range of search algorithms. The most appropriate algorithm may be chosen for a search according to the data types found in the search criteria. A specific implementation of this framework will be presented as a Jini service, using a constraint satisfaction problem-solving architecture that allows different algorithms to be used as library components.

INTRODUCTION

The availability, speed and reliability of networking technology validates a service discovery framework, where clients connect to services over the network based on the advertised capability descriptions (Pascoe, 2000) of those services. Services in this framework are computer-based interfaces to devices, applications, objects or resources that a client uses.

The key challenge for such systems is to enable clients to locate the service that best suits their needs, where best will be defined by the client using infrastructure provided by the framework's implementation. The client's requirement description must be compared with the advertised capability descriptions of the services to find the best match (Pascoe, 2000).

Comparison of a client's requirement description with a service's capability description requires an infrastructure with a language for describing service attributes. Descriptions can be formed using this language and used as search criteria. Descriptions might include service attributes, such as type of service, or quality of service in terms of cost, speed or accuracy of results.

Service discovery frameworks may be classified according to where the comparison takes place. The locality could be: at the client site, at the server site or at a third party search service called a lookup. Lookup is typically a directory-based process of locating or looking up a specific service or activating an agent capable of doing the job (McGrath, 2000). Each of these styles has a different profile in terms of network traffic and a combination of all three is possible (Pascoe, 2000).

When using a lookup, the client must be able to provide the lookup process with enough detail for the service to be located. This detail may be a specific address or identification or it may be data to form some matching criteria with which the lookup process may search and build a satisfier set of services that match the search criteria.

This paper begins by giving a useful definition of a directory service with an attribute language and search language. Most directory services are either distributed file system or networked file sharing mechanisms. This paper then expands upon that definition to form the concept of a service directory that allows sharing of electronic services by electronic clients. Java's Jini technology is explored as a perfect example of a service directory that uses capability descriptions and requirement descriptions with exact pattern matching as part of its search language.

Related Content

Multirate Techniques in Filter Design and Implementation
Ljiljana Milic (2009). *Multirate Filtering for Digital Signal Processing: MATLAB Applications (pp. 274-294).*
www.irma-international.org/chapter/multirate-techniques-filter-design-implementation/27218

A Generic Method for the Reliable Calculation of Large-Scale Fading in an Obstacle-Dense Propagation Environment
Theofilos Chrysikos, Stavros Kotsopoulosand Eduard Babulak (2013). *Integrated Models for Information Communication Systems and Networks: Design and Development (pp. 256-277).*
www.irma-international.org/chapter/a-generic-method-for-the-reliable-calculation-of-large-scale-fading-in-an-obstacle-dense-propagation-environment/79668

A Framework for Testing Code in Computational Applications
Diane Kelly, Daniel Hookand Rebecca Sanders (2014). *Software Design and Development: Concepts, Methodologies, Tools, and Applications (pp. 479-505).*
www.irma-international.org/chapter/framework-testing-code-computational-applications/77719

Reuse in Agile Development Process
Chung-Yeung Pang (2020). *Software Engineering for Agile Application Development (pp. 164-187).*
www.irma-international.org/chapter/reuse-in-agile-development-process/250441

Pluralism, Realism, and Truth: The Keys to Knowledge in Information Systems Research
John Mingers (2010). *Emerging Systems Approaches in Information Technologies: Concepts, Theories, and Applications (pp. 86-98).*
www.irma-international.org/chapter/pluralism-realism-truth/38175