Reuse and Improvement of Peersim Open Source Packages: A Case Study with Chord and Cloudcast

Mohamed Gharzouli, MISC Laboratory, Abdelhamid Mehri Constantine 2 University, Constantine, Algeria

ABSTRACT

This paper presents the reuse of two open source implementations of P2P protocols for Peersim simulator. Firstly, we will present the new implementation of Chord, which is considered as the most popular P2P protocol among the structured family of P2P networks. In this latter, the search requests generally use a Data Hash Table (DHT) to find a requested resource. This kind of systems is generally known as Structured and Informed Protocols. In this context, to improve the search mechanism of Chord, we modify its open source implementation for Peersim to introduce another type of resource search that supports semantic and blind requests. The main objective is to show how we have modified the open source code to achieve our goals. Also, in this paper, we will present how we have improved the display of results by the Peersim simulator through another open source package that implements Cloudcast protocol.

KEYWORDS

Chord, Cloudcast, Open-Source Software, Peersim, Reuse of Open-Source Code, TTL-Chord

INTRODUCTION

The use of open-source software has become inevitable in several research and development areas. In many cases, the economic factor is the most important motivation for developers to choose this kind of software. For instance, in the field of large-scale and distributed systems, building a Grid, Cloud or P2P platform is likely to be costly. For this reason, many developers of distributed systems prefer to use network simulators to test and validate their applications. Actually, developers generally opt for reusing open source network simulators, because they can adapt such source codes to their needs.

In this context, the performance evaluation and testing of P2P environments is a serious problem. This kind of networks poses many challenges because they are characterized by the high volatility and dynamism of connected nodes. A real analysis of P2P networks performances, require taking into account the existent protocols, equipment and realistic traffic models. This situation involves a lot of problems in terms of cost and complexity. For these reasons, many P2P tools have been developed to simulate this kind of environments (AnhDinh, Theodoropoulos, & Minson, 2008). In this context, several open sources implementations of existent P2P protocols have been proposed. These implementations are generally developed for a particular simulator. Among the P2P simulation systems, Peersim (Montresor & Jelasity 2009) is one of the most popular simulators in this context. It is written in Java and composed of two simulation engines, a simplified (cycle-based) one and event driven one. As open software, Peersim is used under the GPL open source license. Until now, the users of this simulator provide 24 additional packages developed for different kinds of P2P protocols.

DOI: 10.4018/IJOSSP.2016070103

Copyright © 2016, IGI Global. Copying or distributing in print or electronic forms without written permission of IGI Global is prohibited.
(These latter are generally classified as unstructured, structured and hybrid systems). Among the most popular P2P protocols, we find an open source code of Chord (Stoica, Morris, Karger, Kaashoek, & Balakrishnan, 2001), which is a structured P2P protocol based on the DHT mechanism (Data Hash Table). In this kind of protocols (like CAN (Ratnasamy, Francis, Handley, M., Karp, R., & Shenker, 2001) and Pastry (Rowstron, & Druschel, 2001)) resources are associated to a key produced by hashing, for example, the file name or the file content. DHT systems have scalable routing, the network topology is tightly controlled and files are placed at the precisely locations identified by their keys. The range of the values of the hash function constitutes the ID space. Each peer in the network is responsible for storing key belonging to a single partition of the ID space. Routing tables are stored locally on individual peers; including a list of other peers with addresses and range of keys they are responsible for (Bianchini, Antonellis, Melchiori, & Salvi, 2006). This kind of search mechanism is implemented by a kind of protocols known as Structured and Informed ones. It is generally characterized by the exactitude and the precision of the sent requests. To improve the research mechanism of Chord, we tried to implement a distributed algorithm for semantic Web services discovery by using an ameliorated version of Chord protocol called TTL-Chord (Gharzouli, Messelem, & Bounas, 2014), in which we have marked the research requests by a TTL (Time To Live). The main objective of TTL-Chord is to implement semantic and blind requests by hashing semantic concepts (input, output and goal) of a resource. Then, in this article, the main objective of our work is to present the different made modifications to implement this new version of Chord for Peersim simulator.

Also, among the main inconveniences of Peersim simulator is the representation and the display of the simulation results. Peersim does not provide a graphical interface and we need to use another program (like Gnuplot (Williams & Kelley, 2016)) to display and analyze the results. Through another open source package of Peersim, which is Cloucast, we describe how we have improved the display of Peersim simulator. It is important to clarify that we have improved the display of Peersim with many other protocols (like Pastry, Adam2 (Sacha, Napper, Stratan, & Guillaume, 2010), (Kademlia, Maymounkov & Mazieres, 2002), but we have chosen to present Cloudcast because in the implementation of this last even the packages of Peersim have been modified and compiled. This constraint created several difficulties especially for beginner users.

The rest of this paper is structured as follows: in section 2, we briefly introduce a background about the classification of P2P systems. Section 3 presents some examples of P2P simulators. In section 4, we describe the implementation of Peersim. Section 5 illustrates the implementation of Chord for Peersim simulator. In section 6, we present the first contribution that gives the different made modifications on Chord implementation. In section 7, we will describe the improvement of the display of the simulation results for the Cloudcast protocol. Section 8 concludes the paper and discusses further research.

P2P CLASSIFICATION

Peer-to-Peer networks are generally composed by equal nodes acting simultaneously as a client and a server. By exploiting the high autonomy of heterogeneous peers, we can make a system with a fully distributed control (Gharzouli & Derdour, 2014). There are many ways to classify P2P systems. However, there are two main classifications, which are based on two characteristics. The first one is related to the different levels of decentralization, and the second is based on the way peers are connected in the P2P network and on the way a protocol performs search of resources (Bianchini, Antonellis, Melchiori, & Salvi, 2006). The first classification has been proposed to distinguish between pure P2P systems and Hybrid P2P systems according to the presence or not in the network of special nodes having a coordination role in the resource discovery process (called generally Super Peers). The second classification makes the difference between unstructured and structured systems. In this last case, routing protocols can be blind, if no information about the location of the requested resource is available and therefore during the searching process other peers are randomly queried,
Hacker Culture and the FLOSS Innovation
www.igi-global.com/article/hacker-culture-and-the-floss-innovation/101204?camid=4v1a

Modding as an Open Source Approach to Extending Computer Game Systems
www.igi-global.com/article/modding-open-source-approach-extending/68149?camid=4v1a