

COIN

a distributed accounting system for peer to peer networks

This text describe the project I've ideated and developed for my Bachelor of Science degree at the Department of Computer Science of the University of Torino, Italy. During the project, mentored by Professor Giancarlo Ruffo, I worked on a decentralized accounting system for peer to peer networks.

Introduction

Internet is one of the new technologies which is changing our lifestyle. Every day millions of people exchange emails, browse the web, talk using chats etc... Today Internet technologies are used even for telephones with the growth of Voice Over IP (VOIP) technologies such as Skype.

This incredible usage growth has been supported by the quality and good design of the different low level technologies which the Internet rely on. IP, TCP and UDP are some of the hidden but fundamental technologies of the Internet.

Client Server and Peer to Peer Architectures

Almost every services widely used on the Internet (emails, browsing, etc..) is designed following a so called "Client / Server" architecture. This kind of architecture relies on a centralized component, called Server, which is accessed by lot of other hosts, called Clients, which want to use the service provided by the Server.

Only some years ago a new kind of architecture called "Peer to Peer" has been introduced. By eliminating or limiting the presence of centralized components, those architecture make each host which participate on the same level of the others.

| Properties | Client/Server Architectures | Peer to Peer Architecture |
|------------------------|------------------------------------|----------------------------------|
| Manageability | + | - |
| Data consistency | + | - |
| Extensibility | - | + |
| Fault-tolerance | - | + |
| Security | +/- | - |
| Resistance to lawsuits | - | + |
| Scalability | +/- | + |

The above table presents the different properties of the two different architectures. Client/Server system, as they rely on a centralized component, are usually easy to manage and keep consistent while its extremely hard on Peer to Peer Systems. For the same reason Client/Server systems are not easily extensible and fault tolerant as a decentralized system can be.

Security can be easily improved in Client/Server Architectures but implementing it on a Peer to Peer

architecture is extremely complex. Peer to Peer systems are also more resistant to lawsuits because the hosts involved are from different places on the world.

The main advantage of a Peer to Peer system is scalability which could be achieved easily.

The above considerations should mark how Peer to Peer networks are really good for some kind of applications.

The Free Rider Problem

Peer to Peer systems are based on the assumption that each peer who takes part in the system contribute as many resources as it utilizes from the other peers. Peers are expected to collaborate in order to achieve a common task.

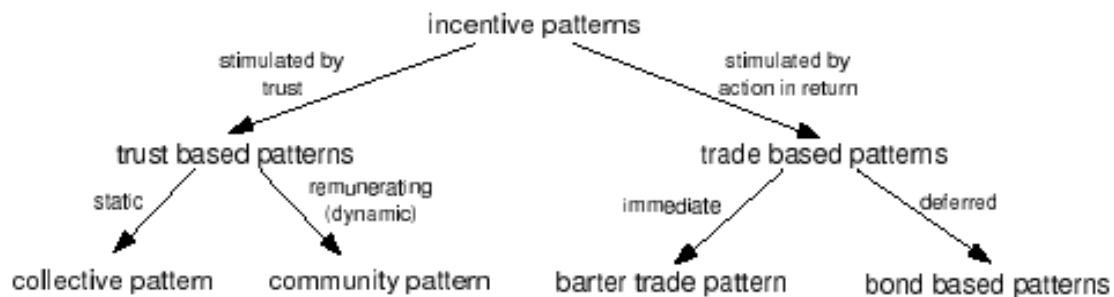
However collaboration is expensive for peers in terms of bandwidth, CPU, RAM, etc.. Collaboration could lower the user experience. Then it's unlikely that a rational peer will actively contribute to the system.

This is the so called "Free Rider Problem". A free rider is a peer who benefits from the efforts of other peers without contributing any resources or performing any tasks itself.

This is a serious problem because it could make the whole system useless if most peers were freeriders. It's needed a way to force peers to actively contribute to the system.

Incentives for Peer to Peer Networks

In order to overcome the free riding problem, peers need to be given appropriate incentives for cooperation with other peers. Different approaches have been created by researches for incentives which are described below.



Trust based incentives patterns: this kind of incentives are based on the trust between different peers. A community of peers is created and each peer can access to some services basing on the reputation it have in the community.

Trade based incentives patterns: this kind of incentives are instead based on the exchange of something in order to access to services. There are two approaches: *barter-trade*, where peers exchange services, and *bond-based*, where peers exchange a virtual or real money.

COIN - Introduction

During my project for the degree I developed a bond-based incentive system for a peer to peer network and I called it COIN. The main goals for the system were decentralization, efficiency, scalability, reliability and security.

In order to achieve those goals I decided to base my system on an overlay network. Overlay networks are some of the coolest new technologies for Peer to Peer networks which guarantee some good properties such as decentralization, scalability and fault tolerance. In particular I used the Pastry overlay network and the FreePastry implementation.

COIN - Protocol

COIN has been based on D. K. Hausheer work called PeerMint.

COIN has four main functions:

- balance request: used by peers to get their balance
- cash flow: used to exchange the virtual money between peers
- funding: used to convert real money into the system virtual money
- withdrawal: used to reconvert virtual money into real money

For details on this functions implementation please refer to my original degree thesis (in Italian)

COIN – Implementation

Basing on the FreePastry 2, an open Java implementation of Pastry, I've been able to realize a prototypical implementation of COIN where all main functions have been implemented. This implementation, however, does not contain all the security features of the protocol. It's just a demo which should convince you that such approach is possible.

Conclusions

Working on this project has been a wonderful experience. I really enjoyed designing and coding it.

I learned a lot: I've got comfortable with DHT, Overlay Networks and Pastry, I had to analyze and solve networks security problems, I designed an application level distributed protocol and a prototypical implementation and I also tested it.