On Exploiting Agent Technology in the design of Peer-to-Peer Applications

Steven Willmott, Josep M. Pujol and Ulises Cortés

Universitat Politècnica de Catalunya
Llenguatges i Sistemes Informàtics
Campus Nord, Módul C5-C6, C/Jordi Girona 1-3, Barcelona (08034), Spain
{steve, jmpujol, ia}@lsi.upc.es

Abstract. Peer-to-peer (P2P) architectures exhibit attractive properties for a wide range of real world systems. As a result they are increasingly being applied in the design of applications ranging from high-capacity file sharing and global scale distributed computing to business team-ware. The objective of this paper is to outline a number of areas in which Agent techniques for the management of *social problems* such as decision making or fair trading amongst autonomous agents could be used to help structure P2P actions. In particular we focus on approaches from mechanism design, argumentation theory and norms / rules and electronic institutions.

1 Introduction

Peer-to-peer (P2P) architectures exhibit attractive properties for a wide range of real world systems. As a result they are increasingly being applied in the design of applications ranging from high-capacity file sharing and global scale distributed computing to business team-ware.

In addition their benefits however, P2P systems also fundamentally change the networking paradigm used in an application often causing tensions with other application goals such as security, predictability, performance guarantees, billing and so forth. Some of these issues in particular arise due to the nature of *control*, *authority* and *ownership* typical found in peer-to-peer systems:

- It is no longer possible to know exactly who is participating in the system.
- Participants in the system may change over time appearing and disappearing without trace.
- There are generally no centrally controlled 'arbiters' available to make authoritative decisions.
- Nodes may not only fail - they may be actively trying to subvert the system.
- Nodes may not only behave maliciously by themselves - subgroups of them may do so in a coordinated manner.

Each of these problems is not only technical but also *social* in nature [12] - springing from the new found *autonomy* and *decision making power* of the peers (actors) in the system. Whilst standard distributed systems engineering provides some of the answers, much work relevant work can also be found in the Agent and Multi-Agent Systems literature. The objective of this paper is to explore how some of these more 'social' P2P network issues could be addressed using various paradigms / approaches from the Agent research community.

While there are already many good examples of the application of Agent technology to P2P systems (such as many of the papers in previous editions of the Agents and Peer-to-Peer Computing workshop itself), with the exception of studies in the area of reputation and coordination [3] and others) the majority of this work to date has focused primarily on *algorithmic* or *infrastructural concerns*. In order to broaden this debate, in this paper we look at a number of other areas of Agent technology which could also bring significant benefits but have not been extensively applied to P2P systems as yet, these are mechanism design, argumentation theory and notions of norms / laws and electronic institutions. The paper is organised as follows:

- Section 2 briefly outlines some often positively and negatively perceived properties of P2P architectures.
- Section 3 characterises typical P2P systems in terms of different potential types of agent systems.
- Section 4 provides a number of example potential areas in which different types of Agent technology might benefit P2P application design.
- Section 5 concludes the paper.

The paper is discursive in nature and is intended to act as a discussion starter rather than an in-depth analysis of the issues involved.

2 The Good, the Bad and the Anti-Social

While a certain amount of the interest in using P2P architectures in application development might be attributable to *hype* or *buzz*, they clearly also present key technical advantages beyond traditional client-server approaches for some applications. Some of the most visible of these advantages include:

- *Virtualised / transparent access to large-scale of distributed resources* - in particular computing resources. The SETI@Home search for extraterrestrial life program being one of the best known examples. [1]
- *Low configuration / low maintenance application deployment through Self organisation* - such as the easy to install and maintain teamware applications targeted by Groove Networks. [2]

[1] http://setiathome.ssl.berkeley.edu/
[2] http://www.groove.net/

- *High availability and fault tolerance* through replication, distribution or the extreme resilience of of power-law / scale-free topologies [3] such as the high-capacity content caching achieved by services such as AKAMAI and BitTorrent. [4]
- *Anonymity* for users and providers - such as the information sharing services provided by the Freenet system which obfuscates the provider of information by sharing it between many hosts. [5]
- *Explosive deployment and growth* through peer download and installation - such as the extremely rapid user growth exhibited by new services such as Skype (a new-entrant global P2P voice-over-IP telephony service). [6]

Each of these advantages could provide a *d*ecisive business advantage in certain types of applications - allowing an enterprise to exploit a market and/or establish itself in a way that would be impossible with conventional client server approaches.

Inevitably however, adopting a P2P paradigm may also subject the subsequent application to a number of less desirable properties. Specific, well documented issues include:

- *Management challenges* - the deployed application can no-longer be directly managed as a global whole (raising issues on how to guarantee Quality of Service, perform maintenance / updates or even monitor its size).
- *Network fragmentation* - the network may become accidentally or deliberately subdivided parts which are not interconnected causing fragmentation of a service. Although service may degrade gracefully (showing robustness) by functioning in the remaining sub-parts, a service provider may loose control or contact with some parts of the network.
- *Identity issues* - the identity of users or systems connecting to the system may not be known (potentially raising issues of accountability for actions, fraud, trust and - in non-free services - of billing).
- *Security / Subversion* - malicious users or systems may be able to connect to system and subvert it by exploiting the lack of centralised authorities to monitor or control actions by its users.
- *User/Provider conflict* - the objectives of individual users / nodes may conflict with the global objectives of the network (e.g. users of file-sharing systems such as Gnutella benefit when finding files they would like, however there is no obvious motivation other than altruism / reputation for serving files).

[3] Power law topologies are highly resistant to random errors (failures) for example, although they can be more sensitive against directed errors (attacks). The high clustering coefficient of power-law networks also favours the redundancy of connections while improving communication: network diameter and average path length grow as the log function of the size (number of nodes).

[4] http://bittorrent.com/

[5] http://freenet.sourceforge.net

[6] http://www.skype.com/

While the first two issues might be considered standard networking or distributed systems issues the later issues are increasingly *social* in nature - since they depend on the nature, objectives and eventual actions of individuals using the application. Such problems arise in many classes of systems which are distributed (in ownership and/or space) and open - both of which hold in many P2P systems. These issues therefore raise important questions about the design of P2P application that are generally not be present in their client-server approach. In some cases the issues may also make the choice of a P2P approach inappropriate.

The remainder of this paper is dedicated to looking at how we might use Agent technologies to help exploit some of the benefits of P2P systems whilst mitigating or managing the downsides.

3 Peer-to-Peer Systems as Agent Systems

Before addressing which areas of Agent technology might be applicable to P2P systems it seems worth spending some time examining the properties of P2P systems in terms of typical Agent characterisations.

Informally, nodes in a P2P network can be characterised as fulfilling many current definitions of Agenthood (see [14] for an overview) to a greater or lesser degree. Arguably more important than the properties of each peer as an agent however, are considerations of what type of *Multi Agent System* the application corresponds to.

In the general case one would expect that:

- Peers are *entirely autonomous:* each individual peer could act in any way - conceivably any code provided could have been entirely re-engineered by its owner/user.
- However, peers are *bound to a specific limited set of actions, protocols or messages* specified in the protocols defined for the application - that is they are limited to an agreed set of social conventions which may be broad or narrow.
- *Rational behaviour cannot be guaranteed:* an owner/user may have any number of *external* motivations for particular actions - some of which may not correspond to *rational actions* in the system itself. On the other hand rational action can be made more likely if:
  - Significant participation costs are involved (or in particular if irrational actions have direct costs).
  - Commitments made during participation can be enforced.
- *Cooperative behaviour cannot be assumed:* the motivations for actions of an individual user/peers are *unlikely to be primarily for the social good* - but primarily for that user/peers' own good. Obvious examples include the

phenomenon that systems such as Gnutella are dominated by so called *free riders* [1] - the assumption of benevolence cannot be made.
- *Out-of-band coalition formation and/or collusion is possible and likely:* in other words users/systems are likely to use additional communication channels invisible to application to coordinate their actions in groups when and where this is of benefit.
- *False name / identity participation is possible:* in other words users may create multiple identities (new P2P nodes) to participate in the system if this could lead to financial/other benefits - such as influencing market prices or manipulating trust/other social properties.

These properties unfortunately correspond to problems recognised as *some of the most complex and difficult to deal with* in the Multi-Agent research literature. An analysis of the Multi-Agent Systems literature would show that only a relatively small percentage of known results are directly valid for these conditions.

Hence for applications which fit the above profile, engineering coherent behaviour amongst users of the P2P application is likely to be very challenging. On the more positive side in some cases it might be possible to make some additional assumptions on average over the whole population:

- *Average rationality:* that *on average* nodes/users act rationality, although an individual may not (this could be justified for example in large market scenarios with many users).
- *Verifiable identity:* that false name / fake identity problems can either be excluded or at least made very rare (this could be justified in applications which directly tie application participation to some other verified identity mechanism such as digital certificates and/or corporate employee registrations).

In other applications it may be possible to reduce the impact of other characteristics. However in each application case it is important to capture the assumptions which do and do not hold since they may fundamentally affect the correct functioning of the system.

4 Where can Agent Technologies Help?

In order to illustrate how Agent technologies might help in P2P application design this section presents application examples coupled with a description of how particular techniques from the Agent literature might be used.

The descriptions are intended to be examples only; however we hope that they will be useful as food for thought for other potential domains / technology application. The examples are:

1. Mechanism Design *applied to* P2P trading systems.
2. Argumentation / Negotiation schemas *applied to* P2P-Social Choice problems.
3. Electronic Institutions, Norms, Rules, Policy Languages *applied to* context management in P2P ubiquitous computing problems.

Other diverse examples could include reputation and trust applied to social networking systems or Agent Communication Language semantics applied to interoperability amongst peers.

4.1 Peer-to-Peer Trading Systems and Mechanisms Design

P2P applications show great potential for trading or bargaining systems in which users are able to purchase / exchange goods, services, information or other items. Currently most well known Internet market applications such as EBAY for consumers and a myriad of Business-to-Business trading systems are strongly centralised in nature; however there are strong reasons why P2P approaches might be attractive: dis-intermediation of middle-players (potentially cutting out fees), privacy (since no one entity knows about all transactions), robustness and speed (reducing dependency on a single site) and others.

Unfortunately however, unlike in file-sharing systems where goods/services are provided at near zero loss to the provider, trading systems require real economic exchange. The subsequent potential for financial loss raises the obvious question: *how do we ensure that the rules in the application cannot be violated or abused by one or more parties to defraud other parties?* Standard responses to this such as requiring users to register their identities or the appointment of arbiters to regulate disputes quickly become unmanageable, can themselves be defrauded (e.g. by identity theft) or begin to recreate centralised elements that the application aimed to removed.

Although still in its infancy for strongly distributed systems the research area of mechanism design [8] which brings together theories from economics and multi-agent systems is extremely relevant here. Work by researchers such as Toumas Sandholm and David Parkes traces the boundaries of:

- Which properties market systems conforming to a particular combination of rules and protocols exhibit.
- Which strategies dominate under which conditions (and whether they are desirable strategies such as telling the truth about valuations for goods or not).
- The impact of potential market elements such as side payments (additional financial exchange occurring between agents in a market which is not openly carried out using the market itself).

4.2 Argumentation based Negotiation applied to Social Choice Problems

A specific example of mechanism design which does not necessarily involve financial exchange is in the solution of social-choice problems. In these applications, peers are required to provide opinions on a question such as the passing of a law, the truth of a statement, the election of a particular agent to a social role or something similar. This social choice generally needs to be made in such as a way as to respect various definitions of fairness, speed (termination) and proof against manipulation.

Mechanism design provides tools which analyse the properties of different types of voting systems (see [4, 5] for example). However, in cases where the full mechanism cannot be analysed, the decisions cannot be made by simply *weight of opinion* or the choices are not simple to enumerate (e.g. in deciding an arbitrary set of facts that is true) other techniques may also help. The challenge in such applications becomes: *how can I structure interactions between peers in order to reach a fair compromise?*

An emerging area which could be applied here is that of Argumentation theory [15, 10, 17] which provide methods and formalisms to structure dialogues and conversation rules amongst agents in order to ensure properties such as:

- Valid deduction of facts, commitments or other statements from assertions made by agents.
- Termination of the interaction.
- Allowing all agents to have their say - but preventing a vocal few drowning out discussion.

Papers range of different ways of formalising conversation rules [18] to applications in choice problems such as systems for guiding democratic debate.

4.3 Ubiquitous Computing: Institutions, Organizations, Policies, Rules and Norms

A further often cited application of P2P networked is the growing trend of viewing everyday objects (household appliances, mobile devices, clothes, vehicles or almost anything) as augmented communicating devices - each with its own identity and networked functionality. Generically known as *Ubiquitous Computing*, this vision relies on objects being fitted with small computational devices, network capability and behaviour patterns which provide their owners with additional benefits such as information, control, remote activation and so forth [11].

The potential explosion in such devices and the complexity of their interactions means that client-server architectures are generally expected to be overwhelmed and unable to provide reasonable management of such systems. In

particular challenges arise such as: *given so many devices - how can we ensure they do not clash with one another? How can we manage behavioural changes between (for example) home, office and street? How can ensure particular behaviour is enabled/disabled in situations when it may be dangerous?*

A concrete example might be visiting a neighbour's house bringing a wireless enabled device - and having it automatically interact with the house systems / objects [7] - changing its status to avoid clashes and being treated with care by the local network as a potential security threat (e.g. unbeknown to the owner it may have been infected with a computer virus). The resulting interactions are highly non-trivial and in particular dependent on the context of the interaction (home, office, street), the current world state (time, weather, malicious 3rd party network activity), the device owners' relationship and many other factors.

Whilst some relationships, rules and heuristics could be hard coded (such as a process for deciding whether a new device is permitted to access the network), as the number of rules and interactions grows the management problem looks set to explode. Several areas of Multi-Agent Systems research can provide useful tools in this respect:

- Techniques for modelling Norms, Rules, Laws an Electronic Institutions [9, 16].
- Coordination techniques based on declarative rules such as Shoham's Social Laws [20].
- Work on policy languages and models ([7] for example).

Each of these techniques describes collections of rules (from abstract to concrete), properties and other constraints on populations of agents: creating a tangible social context which systems are governed by. Most importantly the approaches are primarily declarative - allowing easy management of the rules and the derivation of guaranteed properties for certain combinations of rules. In a Ubiquitous Computing P2P system the approaches could be used to assign contextual roles to peers, tracking their obligations and rights over time as they change context (through motion or changes in the owner's attitude for example). Hence all devices are peers at one level (objects all have an identity and the ability to act) but are dynamically structured into organisational structures according to need.

This last scenario can also be seen as a superset of the previous two in that market mechanism rules and/or argumentation schemas are often seen as defining an institutional context (or applied in the context of a particular institution).

[7] See [6] for an office example.

5 Conclusions

The abstract for the panel discussion held at the P2P-Agents workshop in New York stated a number of challenges which the P2P systems (in particular for business) faced including *security, trust and reputation, representing business protocols, checking compliance, bootstrapping systems, and performance.* On closer analysis, P2P systems can arguably be recognised as having similar characteristics as some of the hostile environments described in the Multi-Agent Systems literature: peers are entirely autonomous, individual autonomy cannot be assumed, out-of band collusion between peers is possible and so forth.

However even under these conditions work in areas such as Mechanism Design and Norms/Institutions or structured Negotiation can potentially provide tools for engineering predictable applications by analysing:

- The motivations of actors.
- The implicit and explicit social rules and properties of the systems.
- Which types of behaviour can monitored/guaranteed and which not.
- The relationship between in-band actions/interactions (those which form part of the application) and those which might take place out-of-band (externalities or group collusion for example).

Work on reputation in Agent based P2P systems already follows these lines ([3, 13] and others) but we hope that the contents of the paper helps illustrate that there may also be other elements of Multi-Agent system's theory which can address these more *social* P2P application issues.

Acknowledgements

This work was partly supported by the European Projects ASPIC [8], Provenance (IST FP6-511085) and @lis technology net [9]. The work also owes much to discussions with colleagues in the Agentcities and openNet Initiatives.

Notwithstanding this the opinions expressed in the paper are those of the authors and do not necessarily reflect those of other project participants.

Particular thanks go to the other panellists in the session (Sonia Bergamaschi, Sandip Sen, Hector Garcia Molina) and the panel chair Munindar Singh.

[8] http://www.argumentation.org/
[9] http://www.alis-technet.org

References

1. Adar, E. and Huberman, B. A.: Free Riding on Gnutella Technical Report Xerox Laboratories, 2000, Internet Ecologies Area. http://www.hpl.hp.com/research/idl/papers/gnutella/gnutella.pdf
2. Atkinson, K., Bench-Capon, T. and McBurney, P.: PARMENIDES: Facilitating democratic debate. Third e-Government Conference (EGOV04), DEXA 2004, Zaragoza, Spain, September 2004. Published in: R. Traunmuller (Editor): Lecture Notes in Computer Science. Berlin, Germany: Springer.
3. Biswas, A., Sen, S. and Debnath, S.: Limiting Deception in a Group of Social Agents Applied Artificial Intelligence, 14:785–797, 2000.
4. Conitzer, V. and Sandholm, T.: Complexity of Manipulating Elections with Few Candidates. Applied Artificial Intelligence, 14:785–797, 2000. In Proceedings of the National Conference on Artificial Intelligence (AAAI), 2003.
5. Conitzer, V., Lang, J, and Sandholm, T.: How Many Candidates Are Needed to Make Elections Hard to Manipulate? Applied Artificial Intelligence, 14:785–797, 2000. In Proceedings of the Conference on Theoretical Aspects of Rationality and Knowledge (TARK), Bloomington, Indiana, 2003.
6. Chen, H., Finin, T., Joshi, A., Perich, F., Chakraborty, D. and Kagal, L.: Intelligent Agents Meet the Semantic Web in Smart Spaces IEEE Internet Computing, to appear, 2004.
7. Damianou, N., Dulay, N., Lupu, E. and Sloman, M.: The Ponder Policy Specification Language Lecture Notes in Computer Science, 1995.
8. Dash, R., Jennings, N., and Parkes, D.: Mechanism Design: A Call to Arms Computational, IEEE Intelligent Systems, November 2003, pages 40-47 (Special Issue on Agents and Markets).
9. Dignum, F.: Autonomous Agents with Norms Artificial Intelligence and Law 7:69-79, 1999.
10. Dung, P.: On the acceptability of Arguments and its fundamental role in the Nonmonotonic Reasoning, Logic Programming and N-Person Games Artificial Intelligence 77, 2, pp 321-358, 1995.
11. Fitzmaurice, G.W., Ishii, H. and Buxton, W.: Bricks: Laying the Foundations for Graspable User Interfaces CHI, pp 442-449, 1995.
12. Jennings, N. R.: Controlling Cooperative Problem Solving in Industrial Multi-Agent Systems using Joint Intentions Artificial Intelligence 75, pp 195-240, 1995.
13. Jurca, R. and Faltings, B.: Towards Incentive-Compatible Reputation Management. Trust, Reputation and Security: Theories and Practice, Lecture Notes in AI 2631, 2003, pp. 138-147.
14. Luck, M., McBurney, P. and Preist, C.: Agent Technology: Enabling Next Generation Computing January 2003, ISBN 0854 327886. http://www.agentlink.org/roadmap/index.html
15. Kraus, S., Nirkhe M. and Sycara, K.P.: Reaching agreements through argumentation: a logical model (Preliminary report) Proceedings of the 12th International Workshop on Distributed Artificial Intelligence, Hidden Valley, Pennsylvania, pp 233–247, 1993.
16. Panchoco, O and Carmo, O.: A role based model for normative specification of organised collective agency and agent interaction JAAMAS 6(2):145-183, 2003.
17. Parsons, S., Sierra, C. and Jennings, N.: Agents that Reason and Negotiate by Arguing Journal of Logic and Computation, 8, 3, pp 261–292, 1998.

18. Parsons, S., McBurney, P. and Wooldridge, M.: The mechanics of some formal inter-agent dialogues F. Dignum (Editor): Advances in Agent Communication. Lecture Notes in Artificial Intelligence 2922. pp 329–348. Berlin, Springer. g
19. Sandholm, T.: Automated mechanism design: A New Application Area for Search Algorithms In Proceedings of the International Conference on Principles and Practice of Constraint Programming (CP), 2003.
20. Shoham, Y. and Tenneholz, M.: On the synthesis of useful Social Laws for Agent Societies. In proceedings AAAI-92, pp 276-281, 1992