Towards a Policy-Based Distributed Data-Sharing Economy

Doctoral Consortium

Samuel R. Cauvin
Dept. of Computing Science, University of Aberdeen
United Kingdom
r01src15@abdn.ac.uk

ABSTRACT

Large scale data sharing is important, especially now, with more open societies of components such as Smart Cities and the Smart Homes creating data sharing ecosystems. We propose that access to data and knowledge be controlled through fine-grained, user-specified explicitly represented policies. Fine-grained policies allow stakeholders to have a more precise level of control over who, when, and how their data is accessed. We propose a realisation of a policy language, and supporting socially aware mechanisms, to facilitate data sharing in a distributed environment.

KEYWORDS

Policies; Peer-to-Peer; Reasoning; Data Sharing; Social Welfare

ACM Reference Format:
Samuel R. Cauvin. 2018. Towards a Policy-Based Distributed Data-Sharing Economy. In *Proc. of the 17th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2018), Stockholm, Sweden, July 10–15, 2018,* IFAAMAS, 2 pages.

1 INTRODUCTION

Data sharing is becoming an integral part of many aspects of our daily lives. With the emergence of data-driven technology that employ intelligent sensor devices in an environment, such as smart cities [2, 10] and smart homes [3], data exchange and data sharing has to be addressed. While data sharing can provide benefits and services to users, it is important to regulate it to allow users to retain control of their data. Not only is it important to give control to individual users, but to maximise the benefit to all users in data-sharing “ecosystems”.

Usually, data sharing is specified (and constrained) through the use of data access policies. These policies specify how data may (or may not) be accessed, changed and used. Traditional management of typical access policies tends to be centralised and, therefore, has a number of problems, such as information ownership and reliance on a central authority. This authority may allow the manipulation of these policies and answer queries regarding current policy settings for data.

We present a language to specify data access policies [5, 7, 9] and which is based on deontic concepts such as prohibition, permission and obligation. We equipped this language with fully distributed mechanisms to support participants making decisions on how they should go about sharing data, in a socially responsible manner. We consider scenarios in which a number of participants have data they want to share and data they want to obtain. To determine how data will be exchanged participants set up a collection of policies which control what data they provide under what conditions and what is expected in return. We envisage a data-sharing economy, where data can be safely exchanged; additionally, we consider participants also sharing data access policies among themselves

We aim to answer the following research questions:

Q1 What information/knowledge is needed to represent policies in a machine-processable, unambiguous, and compact fashion?
Q2 What mechanisms can we provide, using the information model and their representations (from Q1), to enable rational decisions about data sharing and policy-compliance?
Q3 Can our information model and their representations (from Q1) and mechanisms (from Q2) be sufficient to support data sharing in a distributed and secure fashion?

We considered a range of data sharing scenarios, and established the stakeholders involved, and the information and knowledge necessary to support various situations of data and policies being exchanged. We surveyed existing formalisms to represent and reason with deontic concepts [4, 8], and found these inadequate bearing in mind Q2 and Q3. We thus propose a purpose-built policy language which caters for our scenarios whilst supporting a distributed environment and socially aware mechanisms.

2 STATUS REPORT

Our approach utilises a peer-to-peer network [1, 6], in which participants (whether sensors, individuals, or companies) are peers, each of which is a self-interested party taking part in an economy where data is being exchanged. Each peer, upon joining the network, performs a bootstrapping operation by contacting a central server which issues them with a collection of neighbours and a unique (and signed) identifier. After this, all communication occurs between peers, through our mechanisms. As our mechanisms are designed to function in a fully distributed environment, we have created provisions to ensure compliance and security of our peers.

As we work in a fully distributed environment, each of our peers holds their own (possibly incomplete) information about other peers. Peers collect information as they interact with other peers, storing encrypted records of all interactions they take part in. This gathering of information expands our data exchange to include *meta-data*, that is, information other peers have collected.

In our solution, every peer defines a set of policies which determine how they will interact with other peers. These policies can be updated as time passes to reflect changes in the peers’ goals or knowledge about other peers. Our policies may express general regulatory statements such as, for example, “no drug records

Proc. of the 17th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2018), M. Dastani, G. Sukthankar, E. André, S. Koenig (eds.), July 10–15, 2018, Stockholm, Sweden.

and medical records can be obtained by the same party", or more specific, such as "I will only provide 10 records to each person".

Our mechanisms are designed, where possible, to maximise social welfare by ensuring interactions occur fairly between all stakeholders. Penalties in our solution are enforced by the mechanism of each peer, removing the need for any kind of regulatory body.

A typical exchange:

(1) Requestor (R) sends a data request for their most desired data (D) to Provider (P) with highest (estimated) probability of success
(2) P must determine what policies they should send out in response to this request by producing a number of offers. These offers are sets of policies from P's policy collection, for which P has estimated the profit to be acceptable
(3) P sends acceptable policy sets to R
(4) R must now determine if these policies are acceptable, and if so which is most profitable
(5) R sends transaction records to support a chosen policy set to P. The records are those which pertain to the conditions under which the chosen policy(s) are active.
(6) P determines if the current state and records from R permit access to D through the chosen policy set
(7) P writes the temporary transaction record to their set and sends a copy along with permitted data to R
(8) R stores the data and transaction records

While our solution provides peers with a number of mechanisms and controls, the above exchange can be simplified to four key mechanisms that our peers utilise. Our peers can:

- Estimate the profit of a policy (and by extension the profit of a set of policies)
- Estimate the cost to complete an action (and by extension the cost to achieve a specific set of conditions)
- Produce a set of relevant offers in response to a data request
- Determine if a given policy is active, and permits access by a peer to data

3 EVALUATION

We have established formal properties for some of our mechanisms (e.g., termination, correctness, complexity) and guarantees that social welfare increases when participants use our mechanisms. We developed software using Java and Prolog to simulate various scenarios adopting a peer-to-peer approach. Through our software we are able to run experiments with large numbers of peers, and collect a number of statistics including profit, expenditure, and volume of data exchanged. In our solution, peers all have their own data and goals to be achieved. Our software allows for fine-grained parameterisation, enabling us to explore a wide-range of scenarios.

We are currently planning an experiment using this simulation to measure the effects of our social welfare mechanisms on peer profit. We will do this by varying the ratio of selfish peers (with mechanisms disabled) and responsible peers (with mechanisms enabled), and measuring the effects on average peer profit and expenditure. We intend to do this with five cohorts of Selfish:Responsible peers: 50:50, 25:75, 75:25, 0:100, and 100:0. In addition to this ratio, we plan to vary the degree of connectivity of peers, to test the effects of peers having fewer options when attempting to find data.

4 FUTURE WORK

There are a number of avenues through which we can continue this work. The first that we intend to complete is the implementation, and subsequent experiments on, our new prototype. Following this, there are a number of place-holders for potential improvements to our mechanisms, some of which we will now outline:

We could make improvements to our policy profit calculations, to take into account elements of planning. For instance, the affect of obliging a policy to another peer on future interactions with that peer. This approach would use elements of practical reasoning, to relate the goals of a peer to actions they take. Doing so would allow peers to attempt to work towards favourable outcomes, but requires the implementation of new mechanisms to better predict the consequences of actions.

Another improvement using practical reasoning involves request chaining. This enables peers to exploit their knowledge to get access to data through other peers. For example, if $peer_1$ wants access to $data_1$, and $peer_2$ owns $data_1$ but will not permit $peer_1$ to access it. However, $peer_2$ will allow $peer_3$ to access $data_1$. Request chaining would allow $peer_1$ to dynamically create a policy that allows $peer_3$ access to something he wants, and in return $peer_3$ will obtain $data_1$ from $peer_2$ and send it to $peer_1$.

With all the information that our peers gather, we could enable them to make informed changes to their policy set in response to events. These policy changes can occur at any time, and can be in response to interactions, goal changes, and new knowledge about other peers and data. Implementation of so-called *meta-data* requests in our prototype could enable peers to gather information that they believe would be useful in making informed decisions about interactions. It would be interesting to see if access to extra information can provide an increase to profits.

We could also modify our policy language to allow for variables, and computations of those variables, within policies. This would allow for more complex and realistic policies to be set by peers. For instance, a policy could state "allow a peer access to n items of $data_1$, and in return oblige them to provide me with $2n$ items of $data_2$". This is not a policy we can currently express in our language.

REFERENCES

[1] John Buford, Heather Yu, and Eng Keong Lua. 2009. *P2P networking and applications.* Morgan Kaufmann.
[2] Andrea Caragliu, Chiara Bo, and Peter Nijkamp. 2011. Smart Cities in Europe. *Journal of Urban Technology* 18, 2 (2011), 65–82. https://doi.org/10.1080/10630732.2011.601117
[3] Richard Harper. 2006. *Inside the smart home.* Springer Science & Business Media.
[4] John-Jules C. Meyer and Roel J. Wieringa. 1993. Deontic Logic in Computer Science Normative System Specification. In *International Workshop on Deontic Logic in Computer Science.*
[5] Marek Sergot. 2001. A computational theory of normative positions. *ACM Transactions on Computational Logic (TOCL)* 2, 4 (2001), 581–622.
[6] Xuemin S. Shen, Heather Yu, John Buford, and Mursalin Akon. 2010. *Handbook of peer-to-peer networking.* Vol. 34. Springer Science & Business Media.
[7] Yoav Shoham and Moshe Tennenholtz. 1995. On social laws for artificial agent societies: off-line design. *Artificial Intelligence* 73, 1 (1995), 231–252.
[8] Georg H. Von Wright. 1951. Deontic logic. *Mind* 60, 237 (1951), 1–15.
[9] Georg Henrik von Wright. 1963. *Norm and Action: A Logical Enquiry.* Routledge and Kegan Paul.
[10] Y. Zheng, L. Capra, O. Wolfson, and H. Yang. 2014. Urban computing: concepts, methodologies, and applications. *ACM Transactions on Intelligent Systems and Technology* 5, 3 (2014), 38:1–38:55. https://doi.org/10.1145/2629592