[LHKMSt06] *Nicolas Liebau, Oliver Heckmann, Aleksandra Kovacevic, Andreas Mauthe, Ralf Steinmetz;* **Charging in Peer-to-Peer Systems based on a Token Accounting System;** 5th International Workshop on Advanced Internet Charging and QoS Technologies (ICQT'06), St. Malo, Frankreich, Juni 2006, S.

Charging in Peer-to-Peer Systems based on a Token Accounting System

Nicolas Liebau[1], Oliver Heckmann[1], Aleksandra Kovacevic[1], Andreas Mauthe[2], Ralf Steinmetz[1]

[1] Technische Universität Darmstadt
[2] Lancaster University

Abstract. Today, Peer-to-Peer applications are predominant on the internet when considered in terms of its traffic consumption. However apart from Skype, their commercial success is still very limited. This is due to the difficulties faced when trying to implement crucial functionality such as accounting and charging without violating the Peer-to-Peer paradigm. A fully decentralized accounting scheme based on tokens was presented by the authors last year. In this paper we analyse the interactions between token-based accounting and charging in order to enable peers to charge for their services. We present three different charging schemes using tokens as (1) pure receipts, as (2) Micropayment, and as (3) bill of exchange. These schemes are evaluated based on the provided security and the overhead traffic introduced into a Peer-to-Peer system.

1 Motivation

Apart from Skype, the commercial success of Peer-to-Peer (P2P) applications is negligible. Internet Service Providers believe that the future of P2P is very promising in the combination with Triple Play. They believe that one of the main drivers for Triple Play will be the strong interest of customers today in private content, which can be delivered efficiently using P2P applications. Besides this, other P2P applications have been envisioned whereby peers have to pay for services which they receive. However, it is still an open question for service providers regarding how to charge for the services they offer within a P2P system. A basic requirement for P2P business applications is a P2P architecture which supports commercial services. Often such a P2P architecture is provided by the manufacturer [1,2]. In this paper, we do not consider payment models which the manufacturer can use to charge the peers using the P2P platform. Instead, we focus on the P2P business applications whereby peers charge for their services delivered.

The requirements for an architecture suitable in supporting such business applications and related work about charging systems are summarized in Sect. 2. A core requirement is a reliable, trustworthy accounting mechanism that complies with the P2P paradigm [3]. We have developed a token-based accounting mechanism which fulfils these requirements (see [4]). A short overview is given in Sect. 3. In this paper, we present and analyse three charging alternatives which can be added to our token-based accounting scheme. These alternatives are presented in Sect. 4. In Sect. 5 we compare the different alternatives in terms of the transaction costs born by the peers. In Sect. 6, we draw the conclusions.

2 Requirements for P2P Business Applications

Peer-to-Peer business applications which offer service providers the possibility to charge money for their services have to fulfil several requirements. The fundamental mechanism needs to be able to determine supply and demand, both of which can be determined using the search functionality in P2P. Further requirements include pricing, metering and accounting, charging, billing and payment [5,6,7,8,9]. We will now present the process from pricing to the final payment together with the related work.

Pricing. If peer A wants a service from peer B, both peer A and B must first agree on the service (e.g. download a file) and its price. This price will be expressed in the form of a tariff. There are several options for determining a price, e.g. negotiations or auctions [10,11,12]. For a fair market the availability of price information is important. Price or tariff distribution is e.g. discussed in [13].

When A and B agreed on both service and price, the service will be delivered (e.g. the file will be uploaded by B to A). This period is called service session. During the service session, other functions mentioned above are also required. Several integrated frameworks in different fields of the Internet have been presented [14,15].

Metering. Metering is the process of objectively observing events happening within the P2P system. In P2P systems this is limited to local observation, although global observations are desired. Therefore, it is the responsibility of metering to provide measurements about crucial events (service delivery process, e.g. progress of file upload at B, download at A, time needed for service delivery, storage space, CPU power required, etc.) to the accounting system.

Accounting. By using information from metering, accounting creates receipts and may distribute these within the system for storage. Thus, receipts contain information about the events which the peers claim to have happened. It is the most objective information about service sessions available. Examples of accounting mechanisms for P2P systems are [16,17,4], see also next section.

Charging. Charging combines the accounting information provided, with the tariff which the transaction partners agreed upon and calculates the charge, the actual amount of money the service requestor has to pay to the service provider. Charging can be an ongoing process during the service session, an once only process at the end of the service session, or even an aggregating process over several service sessions. Examples for P2P based systems are [18,17]. The charging information is fed into a billing and payment system.

Billing and Payment. The billing functionality creates a bill which states, among with other information, the amount that the requestor has to pay to the provider. As money is something external to the P2P application, we also assume that the P2P application will use a billing and payment system which is external for the P2P application. The different options for payment are e.g. direct money transfer between bank accounts, online payment systems like PayPal [19], Micropayment systems like eCash [20].

Obviously, there are many alternatives for how a P2P plattform for business applications may be built. Examples include the projects MMAPPS [21] and P2P Yardsale [22]. Examples from other domains include [14,15].

3 The Token-Based Accounting System

The basic concept of our Token-based Accounting Scheme (TbAS) involves a service requestor paying tokens in return for a service provided. Tokens serve as receipts for services provided. Every token has an associated owner, i.e. only the owner may spend his tokens. Accordingly, service providers will collect foreign tokens from various service requestors. A service provider cannot respend foreign tokens he collected nut only exchange them in the so-called token aggregation process against new own tokens. This process of issuing new own tokens is fully decentralized and therefore follows the P2P paradigm. The exchange of tokens using a flexible exchange function enables the limitation of the number of tokens which a peer may possess. This allows the introduction of incentives for service provision within the P2P system. Further, behavior rules can be enforced by relating observed peer behavior with the number of new own tokens a peer receives in a token aggregation process. Next, the four building blocks of the TbAS are explained in more detail. For further details please refer to [4].

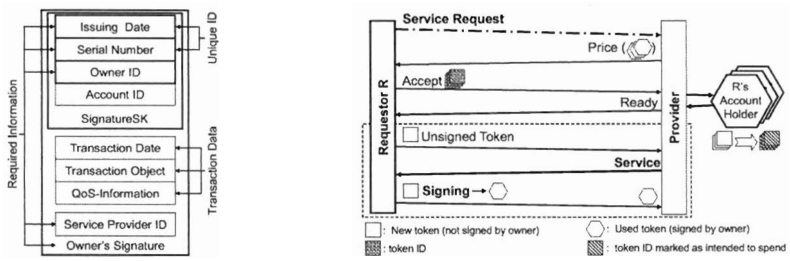
3.1 Token Structure

New tokens contain the owner's identification, e.g. the owner's public key, and a unique ID. To ensure integrity of this information and to prevent forgery of tokens, they are signed with the system's private key (SignatureSK) (see Sect. 3.3 and Fig. 1 (a)). The unique ID allows the detection of double spending. When the owner spends a token, he has to add the required accounting data, which includes the service provider, and then sign the token with his private key in order to achieve information integrity. The token structure is shown in Fig. 1 (a).

A token is not anonymous because its main purpose is to provide accountability in a P2P system. However, using the cryptographic scheme presented in [23], anonymity could also be achieved if desired.

3.2 Payment Process

The payment process of the TbAS is depicted in Fig. 1 (b). In order to prevent double spending for each peer in the P2P system there exists a set of third peers (the so called account holder set) which keep track of the tokens issued to a peer and tokens spent by the peer. Before a service session begins, the requestor discloses to the provider the IDs of the tokens the requestor intends to spend for receiving the service (see Fig. 2 (b)). Now, the provider can check if these tokens are valid. To avoid that the requestor double spends the tokens in a parallel transaction, account holders will mark these tokens as intended to be spent. Thus, double spending is not only detected but also directly avoided.



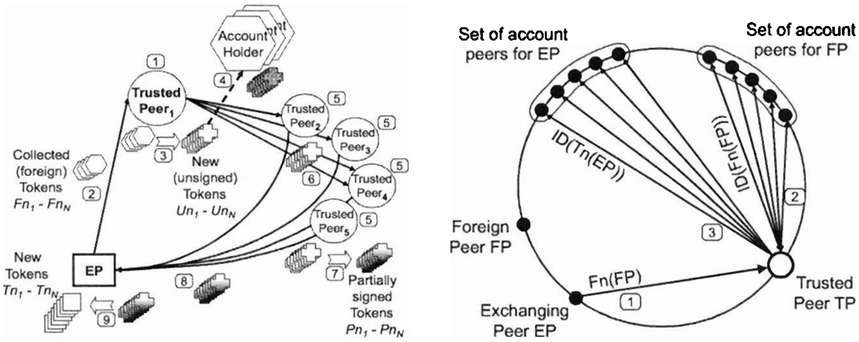
(a) Token Structure

(b) Payment Protocol

Fig. 1. Token Structure and Payment

3.3 Token Aggregation Process

After a peer has collected foreign tokens, it will have to exchange these foreign tokens against new own tokens in order to receive further services. The token aggregation process will determine the amount of new tokens the exchanging peer should receive and create a signature with the system's private key on the new created tokens to provide validity. The process is depicted in Fig. 2 (a). In order to create the system signature in a fully decentralized way, a subset of peers of the P2P system is selected as so-called trusted peers based on their reputation (the TbAS assumes that a reputation system is present within the P2P system). The exchanging peer (EP) sends its foreign tokens to a trusted peer (TP_1). TP_1 calculates the amount of new tokens to be created using the global aggregation function. It creates the new token (without system signature) and sends their IDs to EP's account holder set (see Fig. 2 (b)). The account holders update the list of tokens available to EP. Now TP_1 further chooses k trusted peers who create the system signature using Threshold Cryptography [24]. The system's private key is split into parts and each trusted peer owns one of these parts. k key parts are required to create a signature with the system's private key. Each trusted peer involved sends the tokens signed with the partial key back to EP, who reconstructs the final system signature. In this way, the system's private key is not compromised.


Account Holder
1
Trusted Peer$_1$
4
Trusted Peer$_2$
5
Trusted Peer$_3$
5
Trusted Peer$_4$
5
Trusted Peer$_5$
5
Collected (foreign) Tokens Fn_1 - Fn_N
2
3
New (unsigned) Tokens Un_1 - Un_N
6
New Tokens Tn_1 - Tn_N
EP
8
9
7
Partially signed Tokens Pn_1 - Pn_N
Set of account peers for EP
Set of account peers for FP
ID(Tn(EP))
ID(Fn(FP))
2
3
Foreign Peer FP
Fn(FP)
1
Exchanging Peer EP
Trusted Peer TP

(a) Aggregation Protocol (b) Double Spending

Fig. 2. Aggregation and Double Spending Protection

3.4 Token-based Accounting Scheme as Incentive Scheme

We have studied the use of TbAS in a file sharing scenario, whereby users pay one token per 1 MB of file size in order to receive the file. Whenever a peer does not have enough tokens to download another file, the peer exchanges foreign tokens collected against new own tokens. Each peer receives a specific amount of tokens upon entering the P2P system.

In [25], we examined the file sharing scenario for different aggregation functions. We have shown that the idea of using tokens as virtual currency (aggregation function set to $N = F$ (N =amount new tokens, F =amount foreign tokens)) will lead to market failure in the presence of altruistic peer. Altruistic peers provide much more services than they consume and therefore accumulate the majority of tokens in the P2P system. Accordingly, other peers do not have the possibility to redeem enough own tokens to be able to request further services. This problem can be solved by using weak or asymmetric incentives.

Now we consider the use of the accounting mechanism within a P2P market, where users pay actual money for receiving services. For these scenarios a charging system has to be added.

4 Charging based on Token-based Accounting Scheme

This section covers three possible alternatives for charging within a P2P application where users pay actual money for receiving services. Thus, we assume a P2P application providing the functionality as described in Sect. 2. Further, we assume that each peer owns a private/public key pair which enables it to provide legally valid signatures. This means that before a service session starts, the peers agree on the service to be provided and a tariff for calcualting the charge of the service.

4.1 Tokens as Receipts

Concept. The service requestor (A) will send one or several tokens to the service provider (B) as receipt(s) for delivered service. B can use these tokens to demand payment from A via a prior agreed billing and payment system. Each peer can request any amount of tokens using the token aggregation protocol. Tokens are not exchanged, only new ones are created.

Discussion. Here, tokens serve the same purpose as receipts created by transaction partners without having to be issued before. Receipts not issued must remain non-forgeable and double spending has to be detectable. This however does not have to be system wide but only between the transaction partners. Both are easy to achieve through the use of signatures and unique receipt ID. Thus with the TbAS, it seems unnecessary to issue receipts.

However, such issuing of receipts offers the possibility of decentralized control within P2P systems. It could be controlled who is allowed to participate in the P2P system. This can be used to exclude peers with a bad reputation. Further, the number of tokens available to a peer can be limited. This means that a peer can do only a limited number of transactions between two token aggregations. This limits the danger of misuse of the reputation system, as seen at eBay; A person could be well behaved until he has a very good reputation, then suddenly he starts to defraud his customers by not sending the purchased good. The person could continue this for some time until it becomes clear that he is a fraud. The limitation of the number of tokens available to a peer is possible, because peers aggregate tokens only after a transaction is completed to mutual satisfaction. To further limit possibilities of fraud, for higher valued services peers could agree on a higher amount of tokens. The enhanced functionality described is especially wise for P2P business applications, as there is no central instance which users could contact in case of fraud (as there is in eBay).

In order to make fraud limitation effective for both the service requestor and provider, TbAS has to be adapted so that both transaction partners must spend tokens for a transaction. Both the requestor when the service was received and the provider when he received the payment, must spend tokens. Otherwise, only service requestors could be excluded from the system. In P2P business applications, it cannot be assumed that each peer requests as much services as it provides. In oder to ensure that there are strong providers, an effective mechanism is also required.

It is apparent that this charging scheme also requires a fast payment scheme. Should the payment require e.g. several days to arrive at the service provider (as in eBay), the P2P business application is a lot less attractive.

4.2 Tokens as Micropayment

Concept: When using tokens as Mikropayments such as eCash [20], each token symbolizes a specific amount of money. Users use tokens to pay for receiving services.

Discussion: In comparison to existing Micropayment schemes tokens are not anonymous but can be modified to be (see Sect. 3.1 and Fig. 1 (a)). When using tokens as Micropayment protection against forgery and double spending is highly important. In the TbAS the forging of tokens is still possible under certain circumstances. However, it is highly unlikely (see [4]). Further, without a central bank it is not trivial problem to solve whom users should pay in order to receive the tokens neccessary for requesting services. A central bank to host the user's accounts and provide the token aggregation functionality would solve this issue. However, this compromises the P2P paradigm.

A solution without a central bank would require the cooperation of several banks with the (manufacturer of the) P2P system. A user would pay money to a participating bank which would in return create a certificate that entitles the user to the receive an amount of tokens. The peer (user) would present this certificate to a trusted peer for token aggregation in order to receive the tokens. It is important that the TbAS ensures that certificates are redeemed only once. The peer's account holders can save this information or a callback function with the banks is possible.

An advantage of using tokens as a micropayment scheme is that peers could exchange foreign tokens received against new own tokens by using token aggregation instead of exchanging them at the bank. This reduces transaction cost.

4.3 Tokens as Bills of Exchange

Concept: Tokens can also be used as a bill of exchange. A bill of exchange is a written order in which one person pays another person a specific sum on a specific date. It can be enforced easily without being subjected to defenses. In the past, the bill of exchange was a very important instrument for trading. Today, it used primarily in international trade. A token is worth the amount of money stated in it. Further information required for a bill of exchange (date of issue, drawee, receipient, due date) must be contained in the token.

Discussion: This concept is similar to the first alternative (Sect. 4.1), however the legal consequences here are much more strict. Therefore, this concept has higher requirements on the peers' signatures, because they have the potential of being accepted internationally.

As an extended concept, a token used as a bill of exchange could be transferred by endorsement to another peer. The old recipient would add the new receipient under the token and sign it. However, now double usage of the token must be avoided (the old recipient could still claim the money from the drawee if he keeps a copy of the token). Therefore, the drawee must be informed about a transfer. If he is not available, the drawee's account holder set must store the information.

Tokens as bill of exchange also offer the opportunity for peers to charge up the mutually "drawn" tokens.

Tokens as bill of exchange could be handled similarly as tokens used as receipts, because each peer needs to get as many tokens as he requires for the services he requests (Sect. 4.1).

When applying this alternative there is the danger of fraud by the transaction partner who has to deliver second (as explained in Sect. 4.1). Therefore, it is more secure to use several tokens in transactions if the service can be delivered in parts. To simplify the status quo of mutual debts, tokens with fixed amounts of money are preferable. Further, the control mechanisms presented in Sect. 4.1 should be applied also here.

5 Assessment

In order to evaluate if a charging scheme can be applied in practice, the two most crucial criteria are scalability and security. We will assess and compare the three alternatives based on these two criteria.

5.1 Security

The security of a charging scheme is measured by its ability to prevent double spending and token forgery. The main mechanism use to avoid double spending is the account holder set. As this increases in size, the system becomes more reliable. Its reliability depends on the probability distribution of the life time of the peers. This can be modeled using Markov Chains, however it has been shown e.g. in [26] that within a file sharing scenario, life time is not exponentially distributed. For new commercial applications, the life time distribution can only be determined based on measurement. These have to be conducted as soon as these applications have a sufficiently large user base. However, today we are only able to estimate the required quorum size based on Markov Chains.

In [4], it has been shown that the required quorum size t depends on the number of trusted peers in the system T and the percentage of bad peers (i.e. peers that try to defraud) among the trusted peers $1 - p_g$. The probability for a quorum consisting of only bad peers results in:

$$p(T, t, p_g) = \frac{\binom{T(1-p_g)}{t}}{\binom{T}{t}}$$

Tokens as Receipts. This alternative has the least security requirements compared to the other two alternatives. It is sufficient, if a defrauding peer must assume that double spending will be detected. Therefore, the account holder set can be kept small. An account holder set size $k = 6$ was selected for the traffic analysis. In order to calculate the quorum size, we assumed that 33% are bad peers and a probability of 0.1% that only bad peers are in the quorum which in results in $t = 7$.

Tokens as Micropayment. Here, strong peer IDs to enable enforceability of sale agreements is required. Further, this scheme requires very tight security against forgery and double spending as this is equipollent to creating money. Assuming that 50% of the peers are bad and a probability of 0.01% for at least one good peer in the quorum, a quorum size $t = 14$ is required. To prevent double spending, the account holder set size also needs to be increased to $k = 16$.

To make it more difficult to forge tokens, the initial tokens created from a bank certificate can contain information of this certificate and which can also be held by the account holders. Forging these initial tokens then becomes impossible. However, after received foreign tokens have been aggregated to new own tokens, it is much harder to prove if a token was illegally created. Here, the already known security mechanisms must be sufficient.

Tokens as Bills of Exchange. In this alternative, the transfer by endorsement is the most critical part because different scenarios for cheating exist here. First, receiver B transfers a token to receiver C. Then B agrees with the drawee A to be paid 50% of the amount of the token. A would save 50% and B gains another 50% and C would not be able to collect the money from A. As time stamps can be easily forged, it is hard to decide which happened first, the token transfer or the payment of A to B. In order to prevent such fraud, the account holder set must always note the actual holder of a token and each clearing of a token to remove it from its list. Accordingly, it is important that the account holder set is available and therefore its size needs to be increased to 16 as calculated above.

As token aggregation is primarily used for the limitation of fraud as in the "tokens as receipts" alternative, the quorum size is similarly configured. In order for an effective limitation of fraud in this scheme, it is required that the service provider sends an own token to the service requestor. These tokens are not allowed to be transferred as they are not bills of exchange.

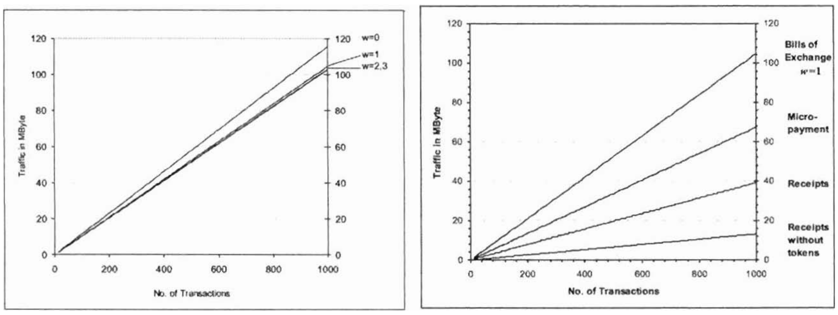
5.2 Scalability

In [4] the scalability of the TbAS was evaluated using measurements of our prototype based on JXTA [27], simulations and a worst case scenario analysis. It has been shown that the traffic overhead which TbAS introduces into a P2P file sharing system, where one token is exchanged for 1 MByte file size, is approximately one percent. The overhead traffic for the three charging alternatives is analysed by using the worst case analysis, considering the different required configurations of the TbAS. In order to compare the overhead of a charging scheme based on simple receipts without tokens, this alternative was also evaluated as shown in Fig. 3 (b).

Tokens as Receipts. We have evaluated the scalability of charging based on receipts with extended mechanisms in order to limit the possibilities of fraud (see Sect.4.1). We have assumed that peers exchange received foreign tokens in batches. The number of messages generated per transaction can be determined using $M(k) = 2s + 4k$ where k is the size of the account holder set and s is the amount of tokens used for the transaction. The number of messages generated by a token aggregation process can be calculated using $M(k,t,b) = \frac{ns}{b}(1 + 2k\frac{b}{s} + 2k + 2t)$, where t is the quorum size, n is the number of transactions that are considered by the aggregation, b is the batch size of aggregated tokens. In comparison to a file sharing scenario, double the amount of token aggregation processes will have to be executed, because both, the requestor and receiver use tokens in these transactions and have to aggregate them. We have assumed a quorum size of 7 and an account holder set size of 8. (see last section). The resulting traffic for a batch size of $b = 20$ is depicted in Fig. 3 (b).

Tokens as Micropayment. When using tokens as Micropayment, the traffic created by the TbAS is comparable to the traffic generated when tokens are used as an incentive in a file sharing scenario (see [4]). However, the system parameters have to be adjusted according to the security requirements (see last section). Accordingly, for the results presented in Fig. 3 (b) a quorum size of 14 and an account holder set of 16 was assumed.

Tokens as Bills of Exchange. The traffic overhead of this alternative is similar to the file sharing scenario of [4], however, the possibility of token transfer by endorsement has to be considered. By paying for a service with a token which the requestor received as bill of exchange, means that message sizes are larger but with fewer token aggregations. The effect of this coherence is shown in Fig. 3 (a), where w is the average number of transfers by endorsement.



(a) Charging using Bills of Exchange

(b) Traffic Comparison of the Charging Alternatives

Fig. 3. Generated Overhead Traffic

6 Conclusions

In most P2P systems today, some mechanisms which are required for business applications are still missing. These are crucial for effective accounting and charging functionality. In this paper, three alternatives for charging based on our Token-based Accounting Scheme [4] were presented. In general, the use of tokens has its advantages compared to using simple receipts. Especially, the number of tokens available to a peer can be limited. This can be used either as mechanism to resolve market failure or as a mechanism for limiting fraud possibilities, as the number of transactions which a peer may execute can be limited. In a P2P environment, this is especially important as the transaction partners are widely anonymous and therefore mechanisms which build trust are required. Therefore, identity management plays an important role as it is required in order to be able to identify defrauding peers clearly.

The additional functionality and control results in the generation of higher traffic overhead. As Fig. 3 (b) shows, in comparison to simple receipts without tokens, the overhead traffic generated increases by a factor 3 for token-based receipts, by a factor 5 for token-based Micropayments, and by a factor 8 for tokens as bills of exchange. In comparison to simple receipts without tokens, the additional traffic of approximately 26 kbyte for token-based receipts, 54 for token-based Micropayment, and 92 kbyte for tokens as bills of exchange was created. This traffic includes all transaction related traffic, but without key management. This is a worst case analysis based on measurements carried out with the JXTA based implementation of the TbAS.

The advantage of charging based on Micropayments or bills of exchange is the possibility for peers to charge up mutual debts and by doing so to save on banking fees. Further, especially when using tokens as Micropayment, peers receive their payment immediately. This means that customers can retrieve the requested service immediately and do not have to wait for a bank confirmation.

Security aspects become very important in P2P systems as soon as it involves real money. Therefore, it is questionable if users of banks would accept a Micropayment scheme which relies on a decentralized mechanism without a trusted third party. The presented charging scheme using tokens as micropayments can be considered secure, apart from the aggregation of foreign tokens for new own tokens, because after aggregation a new token cannot be traced back to the certificate signed by a bank.

References

1. : eDonkey2000. http://www.edonkey2000.com (2004)
2. : BBC integrated Media Player. http://www.bbc.co.uk/imp/ (2006)
3. Steinmetz, R., Wehrle, K.: Peer-to-Peer-Networking and -Computing. Informatik Spektrum **27**(1) (2004) 51–54

Liebau, N., Darlagiannis, V., Mauthe, A., Steinmetz, R.: Token-based Accounting for P2P-Systems. In: Proceeding of Kommunikation in Verteilten Systemen KiVS 2005. (2005) 16–28 (Received Best Paper Award).

5. Androutsellis-Theotokis, S., Spinellis, D., Karakoidas, V.: Performing peer-to-peer e-business transactions: A requirements analysis and preliminary design proposal. In: IADIS International e-Commerce 2004 Conference Proceedings. (2004) 399–404
6. Schoder, D.: Suitability of p2p for business transactions. In: Proceedings of the Peer-to-Peer Systems and Applications Daghstuhl Seminar, March 2004. (2004)

7. Gerke, J., Hausheer, D.: Peer-to-Peer Market Management. In: Peer-to-Peer Systems and Applications. Volume 3485 of LNCS. Springer-Verlag (2005) 491–507
8. Gerke, J., Stiller, B.: A Service-Oriented Peer-to-Peer Middleware. In: Preceeding of 14. Fachtagung Kommunikation in Verteilten Systemen 2005 (KiVS 05). (2005)
9. Hummel, T., Muhle, S., Schoder, D.: Business Models and Revenue Models. In: Peer-to-Peer Systems and Applications. Volume 3485 of LNCS. Springer-Verlag (2005) 473–489
10. Hausheer, D., Stiller, B.: Decentralized auction-based pricing with peermart. In: Proceedings of 9th IFIP/IEEE International Symposium on Integrated Network Management (IM 2005). (2005)
11. Lang, K.R., Vragov, R.: A pricing mechanism for digital content distribution over peer-to-peer networks. In: HICSS '05: Proceedings of the Proceedings of the 38th Annual Hawaii International Conference on System Sciences (HICSS'05) - Track 8, Washington, DC, USA, IEEE Computer Society (2005) 211.1
12. Reichl, P., Hausheer, D., Stiller, B.: The Cumulus Pricing model as an adaptive framework for feasible, efficient, and user-friendly tariffing of Internet services. Computer Networks, Elsevier **43**(1) (2003) 3–24
13. Heckmann, O., Darlagiannis, V., Karsten, M., Steinmetz, R.: A Price Communication Protocol for a Multi-Service Internet. In: Informatik 2001 - Wirtschaft und Wissenschaft in der Network Economy - Visionen und Wirklichkeit. (2001)
14. Stiller, B., Fankhauser, G., Plattner, B., Weiler, N.: Charging and accounting for integrated internet services - state of the art, problems, and trends. In: The Internet Summit (INET 98). (1998)
15. Briscoe, B., Darlagiannis, V., Heckmann, O., Huw, O., Siris, V., Stiller, B., Songhurst, D.: A Market Managed Multi-Service Internet. Computer Communications **26**(4) (2003) 405–415
16. Hwang, J., Aravamudham, P., Liddy, E., Stanton, J., MacInnes, I.: Charging Control and Transaction Accounting Mechanisms using IRTL (Information Resource Transaction Layer) Middleware for P2P Services. In: International Workshops for Quality of Future Internet Services and Internet Charging and QoS Technologies. (2002)
17. Hausheer, D., Gerke, J., Stiller, B.: A generic and modular accounting and charging system for peer-to-peer applications. In: 14. Fachtagung Kommunikation in Verteilten Systemen 2005 (KiVS 05). (2005)
18. Roscoe, T., Hand, S.: Transaction-based charging in mnemosyne: A peer-to-peer steganographic storage system. In: Revised Papers from the NETWORKING 2002 Workshops on Web Engineering and Peer-to-Peer Computing, London, UK, Springer-Verlag (2002) 335–350
19. : PayPal. (http://www.paypal.com/)
20. Schoenmakers, B.: Basic Security of the ecashTM Payment System. In Preneel, B., Rijmen, V., eds.: Course on Computer Security and Industrial Cryptography. Volume 1528 of LNCS. Springer (1998)
21. : Project "Market Management of Peer-to-Peer Services" (MMAPPS). http://www.mmapps.info (2004)
22. : P2P Yardsale Engine (Project Venezia) & P2P Yardsale Application (Project Gondola). http://venezia-gondola.jxta.org/ (3)
23. Chaum, D., Fiat, A., Naor, M.: Untraceable electronic cash. In: CRYPTO '88. Volume 403 of LNCS., Springer Verlag (1990) 319–327
24. Desmedt, Y., Frankel, Y.: Threshold cryptosystems. In: CRYPTO '89. Volume 435 of LNCS., Springer-Verlag (1989) 307–315

25. Liebau, N., Darlagiannis, V., Heckmann, O., Steinmetz, R.: Asymmetric Incentives in Peer-to-Peer Systems. In: Proceddings of AMCIS 2005. (2005)
26. Darlagiannis, V., Liebau, N., Heckmann, O., Mauthe, A., Steinmetz, R.: Caching Indices for Efficient Lookup in Structured Overlay Networks. In: Proceedings of Fourth International Workshop on Agents and Peer-to-Peer Computing, Springer (2005)
27. Sun Microsystems: Project JXTA. http://www.jxta.org (2004)