Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Jan Gerke, Burkhard Stiller

A P2P-Based Composed Service Market

TIK-Report
Nr. 230, August 2005

Institut für Technische Informatik und Kommunikationsnetze
Computer Engineering and Networks Laboratory

Jan Gerke, Burkhard Stiller
A P2P-Based Composed Service Market
August 2005
Version 1
TIK-Report Nr. 230

Computer Engineering and Networks Laboratory,
Swiss Federal Institute of Technology (ETH) Zurich

Institut für Technische Informatik und Kommunikationsnetze,
Eidgenössische Technische Hochschule Zürich

Gloriastrasse 35, ETH-Zentrum, CH-8092 Zürich, Switzerland

A P2P-Based Composed Service Market

Jan Gerke, Burkhard Stiller
Computer Engineering and Networks Laboratory (TIK)
Swiss Federal Institute of Technology (ETH) Zurich
CH-8092 Zurich/Switzerland
{gerke, stiller}@tik.ee.ethz.ch

TIK-Report No. 230

27th July 2005

Abstract

Service-oriented peer-to-peer architectures combine the benefits of service-oriented architectures and peer-to-peer networks, such as loose coupling and scalability. They also support service composition, the combining of existing services into new ones. Until now, only composition service strategies for a single peer acting as a service composer have been investigated. Therefore, this report investigates the market for composed services. It is shown that this market forms an oligopoly and, thus, requires incentive-compatible service pricing. In consequence, the vickrey procurement auction VETO has been developed determining the pricing mechanism in a fully distributed manner. It is proven that VETO is incentive-compatible and maximizes the market's welfare. Furthermore, the VETO protocol is specified and it is shown that this protocol prevents collusion within the auction. Still, VETO upholds the peer-to-peer paradigm by only relying on the peers which are taking part in the auction, no additional trusted entities or superpeers are required.

1 Introduction

Recently, service-oriented peer-to-peer (P2P) networks such as [8] and SOPPS [5] have received an increasing attention. They combine the advantages of both technologies, namely the loose coupling of service-oriented architectures (SOA) [9] with the robustness and scalability of P2P networks [12].

SOAs offer the possibility to combine existing services into new value-added services. This process of service composition is supported by a wide range of service interface description standards, such as WSDL [19], as well as initiatives for describing service functionality, such as the semantic web [10]. The ultimate goal of these initiatives is to enable automated service composition, *i.e.*, combining services into a new service, which fulfills automatically a given service specification without further human interaction. Early work in this direction exists in [2]. While most approaches to automated service compostition consider centralized SOAs only, [6] described and compared strategies for spezialized peers acting as service composers within P2P networks. However, this approach is still focussed on the situation of a single service composer.

Therefore, this report investigates the market for composed services in service-oriented P2P networks. This market consists of a service consumer and all service composers he contacts, in order to have a new service composed. Thus, while the normal service market is assumed to have perfect competition, the market for composed services is an oligopoly. Therefore, the pricing of composed services is crucial in order to enable a living market for composed services. One mechanism to perform pricing in oligopolies are auctions. If designed properly, they have the benefit that they can maximize welfare and reduce the chance of cheating (cf. [3]). In consequence, a vickrey procurement auction called VETO has been developed as a means to perform composed service pricing. VETO maximizes the welfare of the market for composed services, it is incentive-compatible, and it prevents collusion. Furthermore, the use of reputation systems (cf. [4]) to ensure security against expensive or fake service compostion request is discussed, versus the use of service composition fees set by the market.

The remainder of this report is structured as follows: Section 2 sketches the underlying service-oriented P2P network and the service model. Section 3 introduces the market model for the service market as well as the composed service market. Furthermore, it determines influences on welfare of the composed service market. Section 4 develops VETO to perform incentive-combatibel pricing in the composed service market. Section 5 discusses how the VETO protocol prevents collusion. Finally, Section 6 summarizes and concludes the report.

2 P2P-based Service Provision

Before investigating service composition in detail, it is necessary to describe the environment in which it takes place. Thus, the underlying service-oriented P2P network and roles being part of the system are outlined.

A P2P network is not a physical network, but it is built on top of the Internet (cf. [14]). This implies that *peers* are Internet hosts and *links* of the P2P network are end-to-end (e2e) connections through the Internet between such hosts. Thus, the set of hosts taking part in the P2P network and e2e connections between them form the overlay network on top of the Internet. It is assumed that connections provide e2e Quality-of-Service (QoS) guarantees when required by services, regardless of the technology used to provide these guarantees, *e.g.*, IntServ or DiffServ.

Every peer inside the network can provide to and request services from other peers. The term *service* is defined as the functionality, which is offered by a peer to other peers, and which can be accessed through input and output interfaces. [1] defines a service as "*an act or a variety of work done for others, especially for pay*". In the scope of this report, this definition is applied only to the envisioned technical system and its participants (the peers). Thus, a service is a piece of software or a software component operated by one of the system's participants. It fulfills one or several tasks on behalf of another participant of the system, thus carrying out work for him, for which he is paid for.

Every service S has properties $S' = (S'_1, \ldots, S'_n), n \in \aleph$ which describe characteristics of the service (cf. [2]), especially the task which is carried out by the service. These properties may include parameters and information from three different areas:

1. Functionality: A formal description of what the service should do (if specified by the service consumer) or what it can do (if specified by the service provider)
2. Quality: A list of QoS parameters which describe with which quality the service should be provided (if specified by the service consumer) or can be provided (if specified by the service provider). All parameters can have fixed values or can be described with value ranges.
3. Others: Who is providing/requesting the service.

Service properties are specified in *service descriptions*, where every property S'_x consists of a formal *property description* S''_x and a *property range* $S^{\circ}_x = [\underline{S}'_x, \bar{S}'_x]$. Property descriptions describe the meaning of properties according to a common semantic standard shared by all peers (cf. [10]). The property range describes the degree of fulfillment of this property in numerical values. Metrics generating these values are part of the property description.

A peer providing a service to another peer is acting as a *service provider*, while a peer which is using a service from another peer is acting as a *service consumer.* A single peer can take over both roles successively or even at the same time, if he provides a set of services to a set of peers and uses a set of other services from another set of peers. The service usage is always initiated by the service consumer, thus a service provider can not supply unrequested services to consumers or even demand payment for services delivered in such a fashion. Due to the dynamic nature of P2P networks, the duration of a service usage is restricted, i.e., it is not possible to rely on the availability of a certain service for weeks or months.

Two property descriptions *match* when they are semantically equal according to the common semantic standard. Thus, a property A'_x is said to *fulfill* another property B'_y, if and only if, A''_x matches B''_x their property ranges overlap, *i.e.*, $\underline{B}'_x \le \underline{A}'_x \le \bar{B}'_x$ or $\underline{B}'_x \le \bar{A}'_x \le \bar{B}'_x$.

In turn, a service S with properties $S' = (S'_1, \ldots, S'_n), n \in \aleph$ is said to *implement* a service description D with properties $D' = (D'_1, \ldots, D'_m), m \in \aleph$, if and only if, all its properties fulfill these properties specified in the service description, *i.e.*, $n = m$ and $\forall i \in [1,n]$: S'_i fulfills D'_i. Thus, a service S is called an *implementation* of a service description D, if and only if, S implements D. Vice versa, a service description D is said to *describe* a service S, if and only if, S implements D. It is assumed that exact service descriptions exist and have been published within the P2P network for all services offered by peers acting as service providers. However, service descriptions can exist without corresponding implementations.

If a consumer specifies fixed property values in a service description, an infinite number of different services can implement his service description. Thus, a *service class* is defined as a set of services which has specific common properties. Let $S = \{S_1, \ldots, S_n\}, n \in \aleph$ be the set containing all services and let D be a service description with properties $D' = (D'_1, \ldots, D'_n), n \in \aleph$. Then, $\widehat{D}$ is called a *service class* for a service description D, if and only if, $\widehat{D} = \{S_x | S_x \text{ implements } D\}, S_x \in S$. All services which have these properties are called *members* of the service class. Let C be the service class for a service description D. Then, a service S is called a *member* of C, if and only if, S implements D. Thus, every implementation of a service description is a member of the service class created by the description. Furthermore, every service description automatically creates a service class, though this class does not need to have any members.

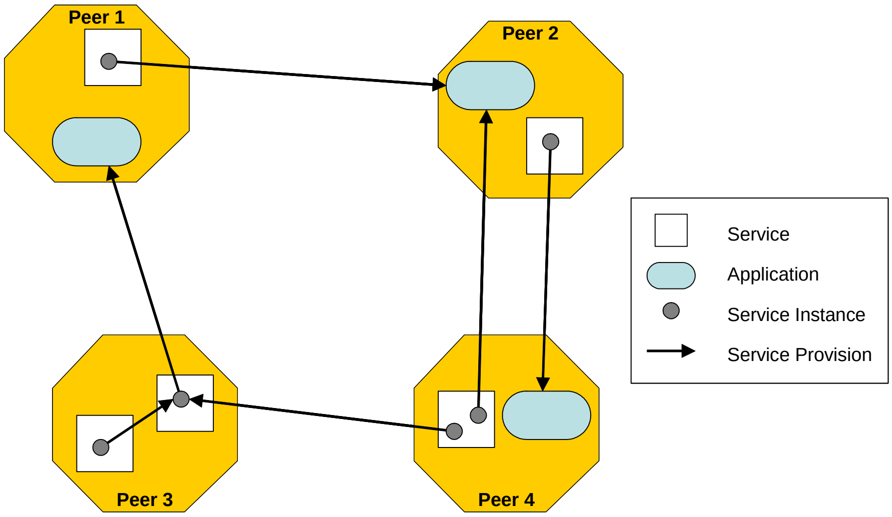

Figure 1. The Use Model [7]

Service usage follows a one-to-one relationship between service consumer and provider, i.e., neither do several service consumers use the same service instance, nor do several service providers together provide a service to a consumer directly. Several service consumers can still use the same service at the same time, but several service instances are created by the service provider and service usage takes place independently. Furthermore, service providers can use services from other service providers, in order to provide a new value-added service to a service consumer. This process of combining services is called *service composition*. A peer carrying out this process is said to play the role of a *service composer*. There is no direct relation between additional service providers and the service consumer. Examples of such service usages are shown in Figure 1.

3 Composed Service Market

Driven by the service market definition, enabled by the underlying P2P network and markets for composed services, influences on the welfare of the composed service market are discussed in turn.

3.1 Market Model

The service market consists of peers which offer services and use them, without composing new services on demand. It is assumed that the service market features perfect competition. While perfect competition is a theoretical concept, this assumption is still reasonable when looking on the requirements for perfect competition (cf. [11]):

1. *The industry is characterized by freedom of entry and exit.* This is one of the big benefits of the underlying P2P network. Apart from installing the P2P middleware, only a strong identity with the capabilities to sign and encrypt is required (cf. [7]).
2. *Buyers know the nature of the product being sold and the charges charged by each firm, i.e.*, the market is transparent. This is another benefit of the P2P network. Peers use its search functionality to look for services they want to use. The result of a search are descriptions of all matching services, including their charges. Furthermore, since the search is initated by the peer wanting to use a service, *i.e.*, the service consumer, it can be assumed that he knows the nature of the service he is looking for.
3. *All companies have a relatively small market share.* All peers are basically the same, *i.e.*, use the same P2P middleware and thus have the same marketing power. Furthermore, due to the low entrance hurdles of P2P

networks, the number of peers is assumed to be large and globally distributed. Thus, it can be assumed that each peer can only obtain only a small market share.

4. *All firms sell an identical product and all firms are price takers.* Obviously, not all peers offer the same service. However, if a peer was offering a service for a price higher than his own costs, another peer could offer the same service at a lower price. Again, this supported by the large number of peers, the available search mechanisms and the lack of special marketing. Thus, service prices are set by the market. Either a peer is able to offer a service for the market price or he is not able to sell the service at all, due to the lower-priced competition in the market.

However, the market model is clearly different when service composition is considered. Service composition itself is a complex process. Therefore, it is assumed that a small number of all peers decide to take on the role of service composer. Each of them carries out his own variant of service composition, using his own business secrets. Service composers act as brokers between service consumers and service providers. They take part in the service market as buyers and in the composed service market as sellers (cf. Figure 2).

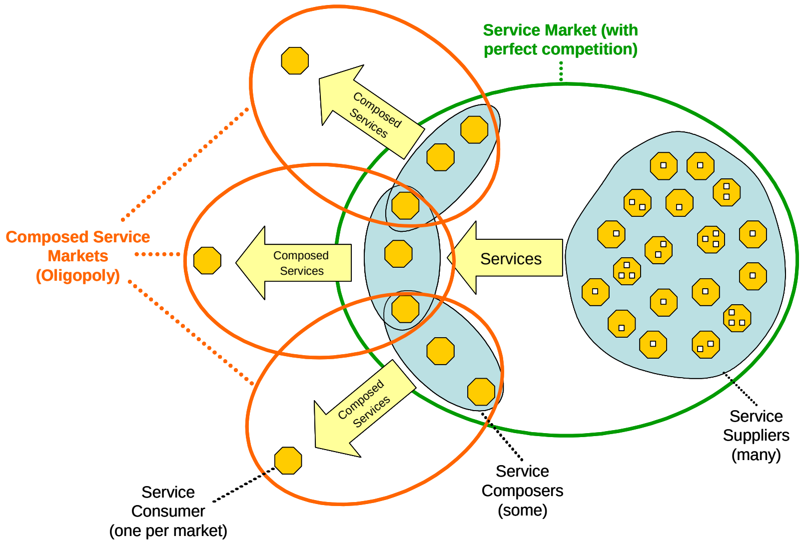

Figure 2. The Market Model

A new composed service market is created whenever a service consumer contacts a set of service composers with the request to compose a new service. Thus, there is a separate market for every new composed service. Due to the small number of overall peers acting as service composers, the market for composed services will not be in perfect competition but will be an oligopoly. Therefore, pricing is an important issue in the composed services market, as a price has to be found for a previously inexistent service. Table 1 covers key properties of these two markets.

Table 1: Comparison of the Two Markets

	Service Market	Composed Service Market
No. of Sellers	Many	Few
No. of Buyers	Many	One
Market Form	Perfect competition	Oligopoly
Traded Good	Basic Services	Composed Services
Persistence	Continuous	Created upon request
Prices	Fixed (set by the market)	Special pricing required

3.2 Social Welfare of Service Composition

The overall social welfare $W(N)$ of the service-oriented P2P network N is equal to the sum of the welfares of the markets running on top of it. Thus, if M_0 is the service market and $M_1 \ldots M_k, k \in \aleph$ are all composed service markets, then $W(N) = \sum_{i=0}^{\kappa} W(M_i)$. It is assumed, that all composed service markets are independant from each other, *i.e.*, every service which is part of a composed service is also available in the service market for all other composed services which require it. This is assumption is reasonable, since services can be offered by various peers and additionally a peer can provide the same service to different peers at the same time. Furthermore, it is assumed that the service composers are not different from peers just buying single services in the service market. This assumption is reasonable, due to the perfect competition in the service market. Thus, the overall social welfare can be maximized by independantly maximizing the welfare of the individual markets, *i.e.*, $max(W(N)) = \sum_{i=0}^{\kappa} max(W(M_i))$. Since the welfare of service market is maximized by the perfect competition (cf. [3]), the overall welfare can be maximized by maximizing the welfare of the composed service market.

As stated earlier, the composed service market is created dynamically, when a consumer requests a new service. Its participants are the consumer and all service composers he contacts. As stated above, the goal is to carry out service composition in such a way, that the overall welfare of this market is maximized. To this end, let M be a service composition market with participants $P = (P_0, P_1, \ldots, P_n), n \in \aleph$, where P_0 is the consumer and the other participants are the service composers he has contacted. In turn, the social welfare equals

$$W(M) = \sum_{i=0}^{n} W(P_i).$$

However, the consumer chooses only one composed service offered by one service composer, all other service composers have no welfare at all. Thus, if the consumer P_0 chooses the composed service S offered by service composer P_x,

$$W(M) = W(P_0) + W(P_x) + \sum_{i=0; i \neq x}^{n} W(P_i) \text{ holds.}$$

It is assumed that all costs of service composition are paid by service composers. They act as risk brokers. When a service composer receives a new service description from a consumer, he has to calculate the chance of finding a composed service, which fulfills this description and is chosen by the consumer. He has to compare this chance and his expected profits against estimated costs of composing the service. If a service composer is too optimistic in this risk calculation, he may be driven out of business by his costs of service composition exceeding his income.

For a single composed service market, $C_i(D)$ is defined as costs of service composer P_i to compose a service, which fulfills the service description D. These costs can be zero, when a composer decides not to start the service composition process. Otherwise, each composer has to pay these so-called *composition costs*, regardless of whether their composed service is the one which is chosen by the consumer (cf. [6]). The social welfare of a composed service market becomes

$$W(M) = W(P_0) + W(P_x) + \sum_{i=0; i \neq x}^{n} C_i(D).$$

The welfare $W(P_0)$ of the consumer equals the utility he receives from the composed service S minus the charge he has to pay for. He expresses his utility by a function $U(S)$, while his charge is calculated by applying the tariff $T(S)$ to the service.

The welfare of the composer P_x equals the revenue he receives from the composed service S minus the charge he has to pay for. He calculates his revenue by applying the tariff $T(S)$ to the service, which computes the price which he charges for the composed service. His costs are described through a function $C(S)$, which represents the sum of the costs of the individual component services used to build the composed service S.

Thus, $W(M) =$

$$(U(S) - T(S)) + (T(S) - D_x(D) - C(S)) - \sum_{i=0; i \neq x}^{n} C_i(D).$$

As this equation clearly shows, the welfare of the composed service market is independant of the tariff used to calculate the price of the service:

$$W(M) = U(S) - C[S] - \sum_{i=0}^{n} C_i(D)$$

This is a logical consequence of the fact that the charge for the composed service stays within the composed service market, thus not effecting its social welfare. The welfare is only effected by the service composers' ability to compose services which provide a high utility to the service consumer at low costs to the service composers.

4 Welfare-driven Procurement Auctions

It has been shown that the price of a composed service does not influence directly the welfare of the service composition market (cf. Section 3.2). Rather, its welfare (and thus the overall social welfare) depends on the utility of the service consumer and costs of service composers. Thus, in order to maximize the market's welfare, each service composers has to solve an optimization problem, namely to maximize the welfare of his composed service. He has to compose the service in such a way, that the consumer's utility is high, while his own costs for composing and delivering the service remain low in comparison. Since service composition is not a deterministic process, this optimization problem is solved by using composer specific heuristic strategies, which have been investigated in [6]. However, these strategies rely on knowing the consumer's utility function.

If a service composer knew the consumer's utility function he would be able to set a tariff which charged the consumer up to his utility, thus maximizing the composer's welfare (*i.e.*, his profit) at the expense of the consumer's welfare (*i.e.*, his benefit). While this would still lead to a maximization of the welfare of the composed services market, this unequal distribution of the welfare would make consumers reluctant to make their utility functions public. However, without knowledge of the consumer's utility function a service composer can not correctly build a composed service which maximises the welfare of the service composition market. Thus, the market needs to give two kinds of incentives to its participants, in order to maximize social welfare:

- It must provide incentives to the consumer to make his utility function public, *i.e.*, send it to composers along with the service description.
- It must provide incentives to service composers to price the composed service fairly, *i.e.*, not to charge the complete utility the service offers to the consumer, when the actual cost for the service composer is lower.

4.1 Vickrey Welfare Procurement Auction

Both incentives can be achieved by using a special form of procurement auction (cf. [12]) to set prices. In the standard procurement auction several sellers bid for a contract with one buyer and the seller with the lowest price wins the auction. In the service composition market the buyer is obviously the service consumer and sellers are service composers. However, a different rule is used to decide the winner of the auction. Different service composers will offer different composed services to the consumer. All of them fulfill the service description, originally sent by the consumer, *i.e.*, offer properties according to the specified property ranges. However, their properties can vary within these ranges, and, thus, the utility the consumer can receive from them also varies. Therefore, rather than deciding the auction by the lowest price it is decided by the highest welfare created for the consumer.

Then, the optimal choice for implementing the auction is analog to a second-price sealed-bid auction, also known as Vickrey auction [18]. In the classical Vickrey auction, bidders bid for an item by sending a sealed bid, which can not be seen by other bidders, to the auctioneer who chooses the highest bid to determine the winner, who finally receives the item. However, the price charged to the winner is equal to the second-highest bit. This kind of auction has two major benefits. Firstly, it is optimal for each bidder to bid his true valuation of the item, since it is not his own valuation of the item he has to pay when winning the auction. Secondly, the Vickrey auction is completed after only one round of bidding.

By adapting the concept of the Vickrey auction to a procurement auction for pricing composed services, a **V**ickrey w**e**lfare procuremen**t** aucti**o**n (VETO) has been developed. The acronym reflects the nature of it's distributed implementation (cf. Section 5.1 and Section 5.2), where participants of the auction are able to detect and proof cheating by others and can, thus, veto the auction's outcome.

In VETO, each service composer P_x bids by submitting his composed service S_x along with the tariff $T_x(S_x)$ used for calculating its charge. The winner is decided by choosing the composed service with the maximum welfare for the customer $W(S_x) = U(S_x) - T_x(S_x)$. The winner provides the composed service to the consumer, but receives a higher revenue than $T_x(S_x)$ from the consumer. Analog to standard Vickrey auctions, the consumer receives only the welfare of the second highest bidder. Thus, if P_1 is the service composer who has made the highest welfare bid and P_2 is the service composer who has made the second-highest welfare bid, P_1 receives the revenue of $R = U(S_1) - (U(S_2) - T_2(S_2))$ as shown in Figure 3.

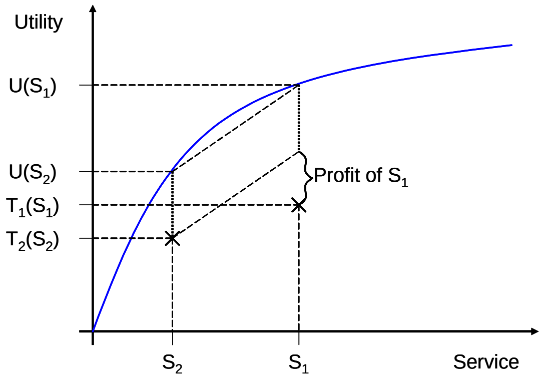

Figure 3. Calculating the Winner's Profit

4.2 Incentive Compatibility

When using VETO, the optimal bidding strategy for service composers is to bid their true valuation of their composed service, namely the sum of the costs of its component services. This leads to a maximization of the market's welfare. Thus, VETO is incentive compatible.

Proof: Apart from bidding the real valuation of the composed service, there are two other strategies, namely placing a higher or lower bid than the real valuation. If a higher bid (i.e., higher welfare and thus lower price for the composed service) is made there are three possible outcomes of the auction:

1. The service composer wins the auction and he would also have won by bidding his true valuation. In this case his revenue is determined by the utility he creates for the consumer and the second-highest bid. Thus, the revenue is the same as if he had bidden his true valuation.
2. The service composer wins the auction which he would have lost by bidding his true valuation. Again, his revenue is determined by the utility he creates for the consumer and the second-highest bid. However, since his true valuation is actually lower than the second-highest bid, he loses money on the deal, which would not have happened when bidding his true valuation.
3. The service composer loses the auction. In this case he would also have lost when bidding his true valuation.

If a lower bid (i.e., lower welfare and thus higher price for the composed service) is made, three possible outcomes of the auction exist:

1. The service composer wins the auction. His revenue is determined by the utility he creates for the consumer and the second-highest bid. It will thus be the same as if he had bidden his true valuation.
2. The service composer loses the auction which he would have won by bidding his true valuation. In this case he misses out on a deal which would have been profitable.
3. The service composer loses the auction which he also would have lost by bidding his true valuation. In this case, there is no difference to bidding his true valuation.

Thus, it has been shown that in all possible cases the service composer can never do better by placing a bid which is higher or lower than his true valuation. However, there are several cases in which he would do worse by placing a bid which is higher or lower than his true valuation. Therefore, the optimal strategy for a service composer is always to bid his true valuation of the composed service.

Thus, there is no need for the service consumer not to publish his utility function. The publication of this function in turn allows the service composers to maximize the welfare of their composed services. Finally, this leads to the maximization of the welfare of the composed service market (cf. Section 3.2).

4.3 Splitting the Market

Since the service consumer choses the service composers he contacts, it is also crucial that the service consumer can not benefit by splitting the composed service market. He is able to split the market by contacting several service composer sets independantly without informing them about other sets. This is prevented by making it a rule that the highest welfare bid in an auction automatically leads to a binding service level agreement which has to be fulfilled by the service composer as well as the service consumer. Furthermore, even without this rule, the service consumer can not benefit from splitting the market.

Proof: Let a service consumer contact several service composers with the same service request. Let him split the complete set of service composers into independant subsets. Let him carry out an independant auction for each subset and let him finally select the winning bid, which will lead to the highest welfare for him.

As shown previously, the welfare the service consumer receives from an auction is always equal to the second-highest bid in this auction. Thus, there are two cases:

1. Composers with the highest and the second-highest overall bid are in the same subset and, thus, their bids are in the same auction. In this case, the maximum welfare of the service consumer is equal to the second-highest bid. Thus, it is the same as if he had not split service composers into independent sets.
2. Composers with the highest and the second-highest overall bid are in different subsets and, thus, their bids are in different auctions. In this case, both bids will win their respective auctions. Thus, the best possible outcome for the service consumer is when the third-highest overall bid is also in one of these two auctions. Then, the maximum welfare of the service consumer will be equal to the third-highest overall bid. Thus, the service consumer's welfare will be equal or less as if he had not split the service composers into independant sets.

Thus, it has been shown that a service consumer can not get a higher welfare by carrying out several independent auctions for the same service in parallel.

5 Collusion

Vickrey procurement auctions have one drawback: They depend on a central auctioneer trusted by all parties. However, introducing a central entity such as a trusted auctioneer would destroy the fully distributed natore of the underlying P2P network. Neither can service consumers and service composers be trusted to act as auctioneers, because both have incentives to cheat: If a service composer was able to control the auction, he could change the bids of his competitors, thus always winning the auction and maximizing his profit. If a service consumer was able to control the auction, he could always introduce a false second-highest bid with a welfare near to the welfare bid of the winner. Thus, he could reduce the profit of the service composer to almost zero, while increasing his own welfare, once more leading to an unfair welfare distribution.

The optimal solution is to create a distributed and secure auction protocol to implement VETO. In order to be secure, this protocol has got to fulfill the following requirements:

- **Privacy**: During the auction, nobody who is not involved in the auction is able to read the submitted bids.
- **Integretiy of bids**: Nobody can enter fake bids into the auction, or change or delete submitted bids.
- **Non-Repudiation**: Nobody can step back from a bid after the auction, *i.e.*, the terms specified in the bid are binding for service consumer as well as the service composer who has won the auction.

Section 5.1 describes the distributed VETO protocol, while Section 5.2 discusses how it fulfills the security requirements. Section 5.3 describes how the protocol can be extended to prevent service composers from gaining insight into their competitors' cost structures after the auction. Finally, Section 5.4 discusses how composition costs can be covered and composers can be prevented from attacking competitors with fake requests.

5.1 The VETO protocol

The general idea of the VETO protocol is that service consumers act as auctioneers, but service composers can detect and also proof, if consumers try to cheat. This enables them to veto the auction's outcome with a legal authority.

As described in [7], it is assumed that every peer i has a strong identity, including a private key $priv_i$ and a corresponding public key $publ_i$ for encryption. Using a key j, every peer is able to perform the encryption function $enc_j(message)$ and the signature function $sig_j(message)$. The public key of a peer is known to all other peers and can be verified by them. It is assumed that all communication between peers is encrypted using public key-encryption, and thus, privacy is ensured. Furthermore, it is assumed that a legal authority exists, which enforces contracts signed with these technologies.

By integrating the VETO protocol into the P2P network, its fully decentralized nature can be mantained. The process consists of the following steps, which are also shown in the UML flowchart in Figure 4:

1. The service consumer P_0 describes the service S he seeks in a service description D.

2. The service consumer sends the description to several service composers $P_1 \ldots P_n, n \in \aleph$, along with his utility function u for this service and a specification of an auction deadline d, after which new bids will not be considered any more. He signs the complete message with his private key, thus sending $sig_{priv_0}(D + u + d)$.
3. Service composer P_i composes the requested service S_i, while maximizing its welfare W_i, *i.e.*, its utility minus the cost for composing and delivering the service. They create a description D_i of the composed service and place this description as well as the welfare associated to the composed service in a service offer O_i.
4. Every service composer encrypts his offer, using his own public key (cf. Section 5.2). Afterwards he signs the encrypted offer, using his own private key. He sends this encrypted and signed offer $sig_{priv_i}(enc_{pub_i}(O_i))$ to the service consumer.
5. The service consumer collects all incoming offers, up to the previously set auction deadline.
6. After the auction deadline, the service consumer sends collectively the complete set of offers he has received back to all service composers. He signs the complete set using his private key, thus sending $sig_{priv_0}(sig_{priv_1}(enc_{pub_1}(O_1)) + \ldots + sig_{priv_n}(enc_{pub_n}(O_n)))$.
7. Each service composer checks his previously received offers against the offers received from the consumer. Furthermore, he stores all offers for later use.
8. The service composer resends his own offer to the consumer, this time without additional encryption. He signs it with his private key, thus sending $sig_{priv_i}(O_i)$.
9. The consumer encrypts every offer O_i again using P_i's public key pub_i and compares the result to the first offer. If they do not match he knows and can proof that P_i is trying to cheat.
10. The consumer sends all unencrypted offers to all service composers. He signs the offers with his private key, thus sending $sig_{priv_0}(sig_{priv_1}(O_1) + \ldots + sig_{priv_n}(O_n))$.
11. Everyone can determine the winner and calculate the price the consumer has to pay, using the second-highest welfare bid.

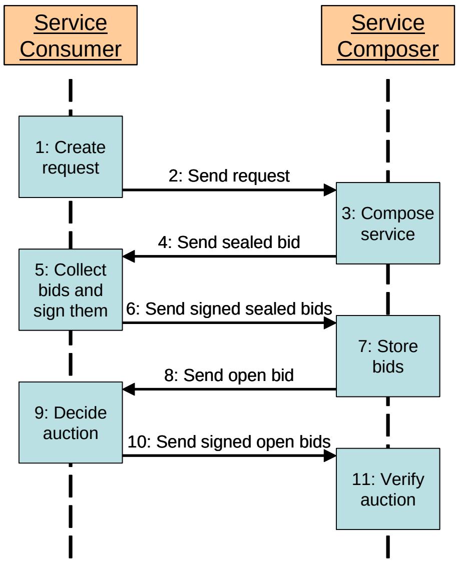

Figure 4. Interaction between consumer and composer

5.2 Discussion of Attacks

The most prominent way to cheat in Vickrey auctions is to create a fake second-highest bid which is just slightly lower than the highest bid. In the case of the Vickrey welfare procurement auction, by creating such a bid, a service consumer could secure all the welfare of the highest bid for himself. Alternatively, a competing service composer could create a fake second-highest bid to diminish the revenue of his competitor, in order to drive him out of business in the long run.

While a consumer or composer could still insert fake bids into the VETO auction before the auction deadline, he cannot benefit from this attack. Firstly, he has to have a fake identity for every fake offer, which requires a lot of effort. Secondly, he can not read those sealed offers sent to the consumer. Thus, he does not know how much welfare he has to offer to create a second-highest offer, which is just slightly lower than the highest offer.

If the consumer posessed a large number of fake credentials, he could still try to insert a large number of fake offers into the auction. After receiving the open offers from the real composers, he would only reveal the highest fake offer which is lower than the highest real offer. However, since the consumer also has to sign the complete set of open offers and send it to the composers, composers are able to detect, if offers are missing, which were originally part of the set of signed sealed offers.

When offers are revealed open after the auction's deadline, it would be possible for the service consumer to create fake second-highest offers or change existing offers. However, since he has signed previously sealed offers, service composers can easily detect such fraud. Using offers signed by the consumer, composers are able to proof the correct outcome of the VETO auction, including the correct determination of the highest and second-highest offers and their welfare.

Finally, offers signed by the composer also make it impossible to step back from the deal which is finalized with the auction's outcome. Using the signed offers, it is always possible to proof afterwards all terms of the agreement exactly. Thus, neither can the consumer refuse to buy the service or pay less, nor can the winning composer refuse to deliver the service for the offerd price or change its properties.

Thus, the security features integrated into the VETO protocol ensure the integrity of bids as well as their non-repudiation, the privacy is ensured by the underlying encryption used for all communication.

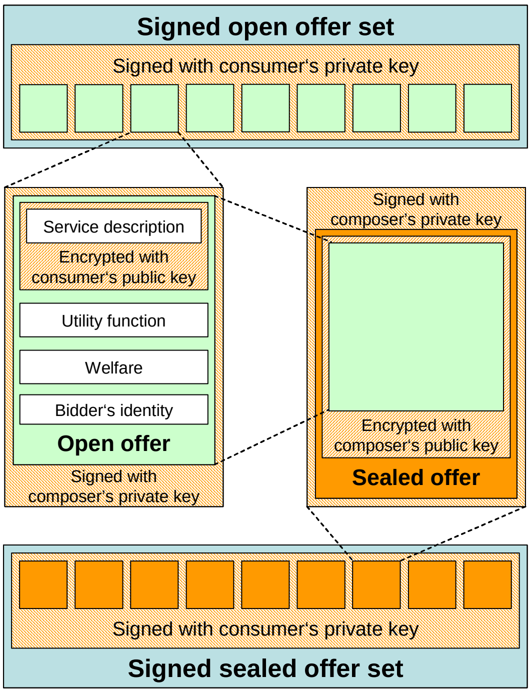

Figure 5. VETO protocol data structure

5.3 Hiding Cost Structures

A general problem of procurement auctions, where bidders are able to read bids of their competitors after the auction is over, remains to be solved: The ability to read bids of his competitors gives a service composer some insight into their internal cost structures, perhaps even in the techniques used to compose the service.

VETO solves this problem by making only a part of open offers readable to other service composers. The service description contained within the offer is encrypted with the consumer's public key. Thus, it is only readable to the consumer who can decrypt it with his private key. The only parts which remain readable to other competitors are the welfare a composer offers and his identity. This information is required to determine and proof the outcome of the auction. However, it does not enable a composer to draw conclusions about a competitor's internal cost structure of service composition techniques, as the offered service may be different from his own and, thus, he does not know how the welfare is created. The data structure of offers and offer sets is shown in Figure 5.

5.4 Controlling Service Requests

The service consumer decides which service composers to contact with a service request. However, if he has unlimited freedom to request services from service composers, two problems arise which are described in Section 5.4.1. Section 5.4.2 discusses technical approaches to solve these problems, *i.e.*, decentralized as well as local reputation systems (blacklists). However, reputation systems have drawbacks. Thus, Section 5.4.3 describes an alternative approach, in which the market itself solves the problems by setting composition fees.

5.4.1 Market Welfare and Request Attacks

First, as described in Section 3.2, the market welfare is influenced by the composition costs of service composers. So far it has been assumed that this cost is small in relation to the cost of the service. Thus, it is not considered explicitely in VETO. When a composer has composed a service, he has to pay the composition cost regardless of winning the auction. Thus, his composition cost does not effect the welfare he offers to the consumer. However, if a consumer contacts a large number of composers, *e.g.*, all composers within the P2P network, the combined composition costs of the composers effect the market welfare. If combined composition costs are higher than the welfare offered by the winning bidder, the overall market welfare becomes negative. In the remainder of this report, this problem is called the *composition cost problem*.

Second, unlimited freedom to request services creates the possibility to attack systematically competing service composers. A service composer A can send repeatedly different fake service requests to another service composer B, thus forcing B to compose a new service and pay the composition costs. Furthermore, A controls a fake bidder who always offers the same service for free. Thus, B never wins the auction but always has to pay the composition costs. If repeated often enough, A can drive B out of business by using this scheme. In the remainder of this report, this problem is called the *request flooding problem*.

5.4.2 Reputation

In order to address the request flooding problem, service composers can use decentralized reputation mechanisms to decide whether to process a service request (cf. [15]). A composer would only process requests from a consumer with a reputation he considers high enough. However, decentralized reputation schemes require a distributed and consistent storage of reputation information. Furthermore, it is difficult to ensure that reputation can not be created artificially, *e.g.*, by several peers giving repeatedly giving each other good ratings for dummy deals. In the past, even centralized reputation such as the e-bay reputation system have been compromized this way [4] (cf. also [16] and [20]).

Therefore, a simpler and more robust approach is to use centralized reputation mechanisms in the form of blacklists. Thus, a service composer would initially process all service requests from a consumer. When the consumer repeatedly requests services from him but he always loses the auctions against other composers with high welfare bids, he considers the consumer to attack him with fake requests. He puts him on his personal blacklist and does not process future requests from this consumer.

Decentralized or centralized reputation schemes can also be used to address the composition cost problem. If the reputation of a consumer is lowered by the ammount x each time a composer loses an auction and is raised by the ammount $y = z \cdot x, 1 \leq z \in \aleph$ each time a composer wins an auction, contacting the maximum number of service composers for a single service requests will lower a consumer's reputation. Thus, if he continues this behavior, in the long run an increasing number of service composers will refuse his service requests. An equilibrium is reached, when he contacts $z + 1$ service composers where the outcome of the auction does not change his reputation.

5.4.3 Composition Fees

However, using reputation schemes have drawbacks. First, especially the decentralized solution is complex and often vulnerable to attacks for artificially raising reputation. Second, centralized blacklists need time to react and thus a certain damage can be done before an attacker is blacklisted. Furthermore, a consumer might be wrongly blacklisted, because competitors just were better and the global picture is missing.

Therefore, a different scheme is proposed, which is based on the consumer paying a fee for the service composition process. Whenever a service consumer requests a service, he offers to pay a fee in return of service offers he receives. The consumer decides the height of the fee himself and signs request and offered fee with his public key. Thus, the service composers can later prove that the fee had been offered. They can also prove if they have sent an offer to the provider, using the signed sealed bids (compare Section 5.2). Furthermore, the "trustable transaction" scheme proposed in [13] can be used to send the fee as an unsigned token with the service request and signing it when an offer has been received.

Thus, the consumer cannot contact too many service composers because he has to pay them, which solves the composition cost problem. Neither can a service composer attack competitors with fake requests because he has to pay them., which solves the request flooding problem. The height of the payment can be set by the market, by service composers as well as service consumers.

Different service composers can set different composition fees. Some composers might decide to work for free, *e.g.*, to gain new customers when entering the market. Others might decide to only compose a service when the composition fee exceeds the expected composition costs. Ultimatevely, composers would process service requests when the composition fee exceeds the expected overall return from the composition, *i.e.*, *fee > composition_cost - chance_of_winning * (welfare - welfare_of_2nd_highest_bidder)*. However, since the different parts of this inequation are almost impossible to estimate, simpler strategies will be used.

Service consumers can influence the quality of offers they receive by altering the composition fee. The higher the fee, the larger the number of service composers willing to work for this fee, the higher the welfare which can be achieved when the consumer selects the right service composers.

6 Summary and Conclusions

It has been shown that it is reasonable to assume that the service market running on top of the presented service-oriented peer-to-peer architecture features perfect competition between service providers. In addition, there is a market for composed services. This market is only created by new service requests and is dominated by an oligopoly of service composers. It needs pricing mechanisms, in order to give the right incentives to market participants, such that the welfare of the market is maximized. Key influences on the market's welfare have been identified and the vickrey welfare procurement auction called VETO has been developed as a pricing mechanism. VETO has been specified as a fully distributed protocol, running between one peer acting as service consumer and any number of peers acting as service composers.

The welfare of the composed service market depends on the service consumer's utility function. While the market welfare could be maximized if the consumer made this function public, without any pricing regulation this could allow composers to price their service to keep most of the welfare for themselves. However, it has been proven that VETO provides incentives to bid true valuations of a service's welfare, thus making it possible to publish the utility function. The VETO protocol is secure against all envisioned attacks by the service consumer as well as service composers. Neither can insert fake bids and everyone can prove the correct (and binding) outcome of the auction. The completely distributed nature of VETO does not break the P2P paradigm at all, thus upholding perfect competition in the underlying service market. Furthermore, VETO does not allow for insights into the cost structure of competing service composers. Finally, attacks on competing service composers with fake requests as well as damaging the market by incurring overly high composition costs can be be prevented by introducing composition fees. Thus, by using VETO the welfare of the composed service market is maximized, as well as fairly distributed between service consumer and service composers.

Future work includes the implemention and evaluation of VETO in a P2P environment, especially investigating the use of different service composition strategies in the competitive VETO auction. Furthermore, service consumers sometimes do not have an explicit knowledge of their utility functions. Thus, it is of special interest to develop and evaluate tools which make the consumer's utility function explicit, by interpolating between the consumer's evaluations of different service property settings.

The authors would like to thank Oliver Heckman, Nicolas Liebau (TU Darmstadt, Germany) and Andreas Mauthe (University of Lancaster, UK) for the helpful discusstions on the subject of this report.

7 References

[1] American Heritage Dictionaries (Editor): The American Heritage Dictionary of the English Language, Fourth Edition; Houghton Mifflin, 2000

[2] M. Carman, L. Serafini, P. Traverso: Web service composition as planning; In ICAPS 2003 Workshop on Planning for Web Services, Trento, Italy, June 2003

[3] C. Courcoubetis, R. Weber: Pricing Communication Networks; Wiley, 2003

[4] R. Dingledine, M. J. Freedman, D. Molnar, D. Parkes, P. Syverson: Reputation; Digital Government Civic Scenario Workshop, Cambridge, MA, USA, 2003

[5] J. Gerke, D. Hausheer, J. Mischke, B. Stiller: An Architecture for a ServiceOriented Peer-to-Peer System (SOPPS); In Praxis der Informationsverarbeitung und Kommunikation (PIK), No. 2, 2003

[6] J. Gerke, P. Reichl, B. Stiller: Strategies for Service Composition in P2P Networks; In Proceedings of the ICETE 2005, Reading, UK

[7] J. Gerke, B. Stiller: A Service-Oriented Peer-to-Peer Middleware; Conference on "Kommunikation in Verteilten Systemen" KiVS'05, Kaiserslautern, Germany, in "Informatik Aktuell", Springer, 2005

[8] P. Haase, S. Agarwal, Y. Sure: Service-Oriented Semantic Peer-to-Peer Systems; In Proceedings of the Web Information Systems Engineering Workshop 2004, Brisbane, Australia, Springer, 2004

[9] Hao He: What is Service-Oriented Architecture?; In O'Reilly webservices.xml.com, http://webservices.xml.com/pub/a/ws/2003/09/30/soa.html, 2003

[10] J. Hendler, T. Berners-Lee, E. Miller: Integrating Applications on the Semantic Web; In Journal of the Institute of Electrical Engineers of Japan, Vol 122(10), 2002

[11] Investopedia Inc: Investopedia.com dictionary - Perfect Competition; http://www.investopedia.com/terms/p/perfectcompetiton.asp, 2005

[12] P. Klemperer: Auction Theory: A Guide to the Literature; In Journal of Economic Surveys v13, n3 (July), 1999

[13] N. Liebau, V. Darlagiannis, A. Mauthe, R. Steinmetz: Token-Based Accounting for P2P-Systems; Conference on "Kommunikation in Verteilten Systemen" KiVS'05, Kaiserslautern, Germany, in "Informatik Aktuell", Springer, 2005

[14] A. Oram (Editor): Peer-to-Peer: Harnessing the Power of Disruptive Technologies; O'Reilly, 2001

[15] Th. G. Papaioannou, G. D. Stamoulis: Effective Use of Reputation in Peer-to-Peer Environments; In Proc. of IEEE/ACM CCGRID 2004 (Workshop on Global P2P Computing), 2004

[16] F. Reichling: Effects of Reputation Mechanisms on Fraud Prevention in eBay Auctions; Stanford University Department of Economics Honor Thesis, 2004

[17] H. Varian: Intermediate Microeconomics: A Modern Approach; Norton, W. W. & Company, 2003

[18] W. Vickrey: Counterspeculation, auctions, and competitive sealed tenders; In Journal of Finance 16, 1961

[19] Web Services Descriptions Working Group: Web Services Description Language (WSDL) Version 2.0; http://www.w3.org/2002/ws/desc, 2005

[20] N. Wingfield: Elusive Figures: An EBay Merchant Disappears; In The Wall Street Journal Online, February 22, 2002