GridEcon: A Market Place for Computing Resources

Jörn Altmann[1,5], Costas Courcoubetis[2], George D. Stamoulis[2], Manos Dramitinos[2], Thierry Rayna[4], Marcel Risch[1], Chris Bannink[3],

[1] International University in Germany, School of Information Technology
Campus 3, 76646 Bruchsal, Germany
jorn.altmann@acm.org, marcel.risch@i-u.de
[2] Network Economics and Services Group (N.E.S.), Department of Informatics,
Athens University of Economics and Business (AUEB),
76 Patision Street, Athens, GR 10434, Greece
{courcou, mdramit, gstamoul}@aueb.gr
[3]Logica Management Consulting
Prof. W.H. Keesomlaan 14, 1180 AD Amstelveen, The Netherlands
chris.bannink@logica.com
[4]Internet Centre, Imperial College London
South Kensington Campus, 180 Queen's Gate, London SW7 2AZ, UK
t.rayna@imperial.ac.uk
[5] TEMEP, School of Engineering, Seoul National University
San56-1, Sillim-Dong, Gwanak-Gu, Seoul, South-Korea
jorn.altmann@acm.org

Abstract: This paper discusses the rationales for a Grid market and, in particular, the introduction of a market place for trading commoditized computing resources. The market place proposed makes computing resources from different providers substitutable through virtualization. This includes the definition of a spot and future market as well as the parameters that a market mechanism for computing resources should consider. The above market place is complemented by a set of value-added services (e.g. insurance against resource failures, capacity planning, resource quality assurance, stable price offering) that ensure quality for Grid users over time. The market place technology for all of the above services has been designed by the GridEcon project, contributing to a broader adoption of Grid technology and enabling a service-oriented knowledge utility environment.

Keywords: Commercial Grids, Grid Computing, Business Models, Grid Economics, Utility Computing, Market Mechanisms, Grid Market Place.

1 Introduction

The Grid, as we use it in this paper, is a system of interconnected, virtualized computing resources. Those computing resources can be located in a few data centers around the world (owned by large enterprises) or can be highly distributed (owned by many end-users or small and medium-sized enterprises). These virtualized hardware resources provide interfaces to execute software (e.g. applications or middleware).

There are many vendors that offer different kind of hardware virtualization software [14][15]. After the buyer purchased the right to execute the resources, it can upload an application or, first, a Grid middleware application and, then, an application on top of it. In our definition of the Grid, we explicitly exclude software resources or information resources.

During the past years, a few approaches have been undertaken to offer Grid technology for commercial purposes: The most successful approach is the one of Amazon, namely Amazon EC2 [1]. Besides this utility computing approach, there are three hardware vendors (HP [2], IBM [3], SUN [4]) and one more Internet company entered the market (Google [5]) that provides similar utility computing services. Since only these few players are in the market, the market structure for utility computing is an oligopoly.

In order to break the oligopoly market structure, GridEcon, an EC-funded project [10][11], offers market place technology that allows many (small) providers to offer their resources for sale. The effect of a market place for computing resources can be illustrated with the following example. If computing resources are scarce due to low supply, the market price for computing resources will be high. Enterprises requiring resources during high price periods will invest in additional equipment. These additional resources can then be externalized when they are not needed, thereby increasing the overall capacity available on the market. The income generated by selling resources in the Grid market will act as an incentive to sell spare capacity if a market place is available.

The GridEcon project designs the technology that is needed to create an efficient market place for trading commoditized computing resources. The market place allows every owner of computing resources to offer spare computing resources as a standardized virtual machine. The challenge is to design this standardized virtual machine, which can be traded on the market place easily and allows establishing a competitive market. The market mechanism used has been designed to be simple for participants to use, and also economically sound. The later is concerned with inducing the right economic incentives to participants and avoiding unwanted strategic behavior leading to market dominance by large players. The GridEcon project also designed a series of value-added services on top of the market place (e.g. insurance against resource failures, capacity planning, resource quality assurance, stable price offering), ensuring quality of the traded goods for Grid users. The market place technology and the value-added services contribute to a broader adoption of Grid technology and enable a service-oriented knowledge utility environment.

This paper is structured as follows: Section 2 describes the rational for a market place for computing resources. In Section 3, the requirements for a spot and future market are discussed in detail. This includes the definition of a spot and future market as well as the prerequisites for a market mechanism for computing resources. Section 4 concludes the paper.

2 Markets and Market Places for Computing Resources

In general, a market describes the entirety of all transactions between a buyer and a seller in all forms. This includes direct transactions between a buyer and a resource provider as well as the transactions performed using the market mechanism provided within the market place. A market for computing resources comes into existence if there is discontinuous demand and redundancy of computing resources.

A buyer of a (virtual) computing resource in our Grid market is an entity that purchases the right to execute an application on a computing resource. Similarly, a seller of a computing resource sells the right of using a computing resource for a certain period of time.

A market place is an environment, which supports buyers and sellers in carrying out their trading transactions between each other [13]. The rules of interaction between the players in the market place are set through the market mechanism. The market place helps finding trading partners more easily by storing information centrally and by offering procedures to facilitate the matching between supply and demand. It also makes sure that fraudulent transactions do not happen.

2.1 Rationale for a Market for Computing Resources

In general, a market for utility computing will only work if one of the following conditions are met: (1) the pattern of individual demand for resources show spikes; (2) the units of computing power that are needed are smaller than the purchase of a computer could provide. In addition to this, all of the following conditions exist: (3) adequate technology for implementing utility computing (e.g. definitions of standardized interfaces); and (4) none-constraining regulations.

The demand spikes, which have been mentioned in condition (1), could be a consequence of the type of business that the company performs. The business itself might bring uncertainty about the need for computing resources. An example for such a business could be a company that creates animated movies. When a movie has to be rendered, the demand for computing resources is very high, otherwise very low. To cover this uncertainty, until now, companies had to over-provision their IT resources, which is expensive.

If condition (2) applies, then, without utility computing, most users would not be able to afford computing resources because of the high cost of ownership and their sporadic usage patterns. Utility computing allows them to get access to a very large quantity of computing resources for a short time on a per-usage basis.

Condition (3) is a necessity in order to be able to substitute a computing resource of one provider with a similar computing resource of another provider. Without this condition, the effort to connect to another provider might be too high. In order to efficiently use utility computing, technology must be available that helps the provider to organize its tremendous amount of computing resources in an efficient way. It must be guaranteed that the management cost for using the Grid is not higher than administrating resources that are owned.

Government regulations mentioned in condition (4) can have a huge impact on the organization and efficiency of providing computing resources. These regulation issues must address areas such as data storage location, taxation, and access rights to computing resources. In order to make utility computing successful, a prerequisite is a set of supporting regulations.

If a market for computing resources is not available, an alternative solution is over-provisioning. That is, a user should have a permanent computing capacity that can meet the demand peaks, which in light of condition (1) are much higher than the average. However, besides the fact that this would defeat the purpose of utility computing, this is beyond the budget of many users and it causes economic waste of resource. Another alternative is not to meet all demand for processing resources, which consequently leads to missing an opportunity for getting additional revenues.

2.2 Economic Implications of a Grid Market Place

The rationale for a market place for computing goods is the current utility computing market. It is an oligopoly. The advantage of the few providers of utility computing (e.g. Amazon, HP, IBM, Google, Sun) in this oligopoly is that they have brand recognition and are trusted entities. However, these few providers offer their resources at a price higher than in a competitive market structure.

A market place provides an alternative to the existing oligopoly of utility computing providers in the market. If buyers and sellers accept (i.e. trust) the market place for executing their trades, it will increase the supply of computing resources in the market. Consequently, It will lower the price for buying computing resources in the market. Computing resources become even affordable to enterprises with low budget. All quality of service issues would be resolved by the market place. Similar to the case of stock exchanges, there could be more than one market place for trading computing resources.

The market place that we envision in GridEcon is an environment that allows SMEs to trade their resources. However, there may be also larger companies that will benefit from such a market place and its services [12]. For instance, it might be that large companies offer their spare capacity at the market place.

Eventually, we expect to see new business models arise. These business models will make entities act as brokers of computing resources to other companies, or offer other value-added services, complementing the market services.

2.3 Services for the Market Place for Computing Resources

In order to attract customers to the market place and get the market place concept accepted by users, the market place must offer a set of services that makes the use of the market place service convenient, secure, and less risky. This is the focus of the GridEcon project. The services, which we identified to be necessary for a market place environment, can be classified into core services and value-added services. The market place provider offers the market mechanism service and additional core services. These services are described in detail within the next subsection. The value-added services do not have to be offered by the market place provider but can be offered by independent service providers instead. The capacity planning service and the insurance service are briefly described in subsection 2.3.2. We are convinced, only if these services are present, a market place for commoditized computing resources will work.

2.3.1. Core Services of the Market Place for Computing Resources

Resource Redundancy

The market place might provide resource redundancy in order to achieve service reliability even if a resource provider dishonors his commitment. The market place might also provide extra resources in order to increase the probability of a liquid market in times when demand is not matched by supply. In these cases, the market place deals with the risk of resource unavailability and will ease the bootstrap of a new market place.

Monitoring of Computing Resource Offers

In order to assure quality of the good offered (i.e. to assure that customers are truthful in declaring their computing resource postings), the market place provider may probe randomly the offered, not leased computing resources by running benchmark programs on them. In case that the computing resources have been sold on the market place, the market place requests ratings from the buyer of the resources, using a reputation system. The information within the reputation system is private to the market place. It will be used to decide whether to allow resource providers to sell goods in the market place in the future.

Security

The market place has to provide a secure environment. All communications among the market participants and the market place has to be encrypted. The market place also has to ensure that no viruses are spread between machines that are traded on the market place. It also builds in protection mechanisms that blocks buyers from getting access to resources of the provider beyond the border of purchased resources.

Simplicity

The market place has to enable access to computing resources in a transparent and simple way, using an intuitive user interface. Any transaction on the market place has to be simple, including the integration of the resource into the existing IT infrastructure of the buyer.

Anonymity

The market place has to ensure anonymity of sellers and buyers. This service is necessary in order to hide the identity of large providers/sellers. If the identity cannot be hidden, buyers and sellers might circumvent the market place and make the market transaction directly. If anonymity exists, buyers and sellers cannot trade directly with each other and more competition is guaranteed on the market place. However, buyers must be given the option of bidding for resources that will be provided by a single provider.

Standardization of Computing Resources

In order to offer commoditized computing resources, the market place must be able to cope with different hardware types available at the sellers' premises. Therefore, the market place requires sellers to virtualize their hardware and to run "standardized" virtual machines with certain performance characteristics (as defined by the market place provider). The market place accepts only offers of computing resources that comply with those performance standards. This makes all hardware resources comparable and substitutable.

The standardized resources are classified in terms of quality (e.g. CPU speed, bandwidth, main memory, disk space). However, in order to abstract from the detailed specification of the performance characteristics of virtual machines, those "standardized" Grid resources are given abstract names such as GEUnit1, GEUnit2, or GEUnit3.

2.3.2. Value-Added Services to the Market Place

Capacity Planning

The acceptance of Grid computing also depends on how simple is it to make optimized planning decisions about computing resource purchases. Therefore, to achieve acceptance, a capacity planning service needs to be in place that supports market place participants (i.e. sellers and buyers) in their decision making process for selling and buying resources on the market place. The capacity planning service has also to help customers on how to optimally shape their demand and to find the appropriate resources for their applications. The precision of the prediction of the capacity planner is based on input parameters, such as the current load, the past load, the current demand, the market price of computing resources, and the existing computing capacity.

Insurance Contracts

Uncertainty about resource failures can also have an impact on the acceptance of the market place. Those market participants, who are risk averse, will not participate in the market place if there is uncertainty about resource reliability. To overcome this, an insurance service must be in place. The insurance service provides an insurance contract to buyers for occurrences of resource failures. In case of a failure of a resource, the insurance provider replaces the failed resource with a fault-free resource (in case it owns resources) or simply compensates the buyer with the amount of money specified in the insurance contract.

3 Requirements for a Spot and Future Trading

Stocks are traded in markets called stock exchanges (e.g. New York Stock Exchange). Though all exchanges used to require physical presence of traders and trading was performed by means of open outcry, most modern stock markets rely on automated electronic trading systems. For instance, NASDAQ is an electronic stock trading platform, where all trading is done by means of computer systems [6].

3.1 Definition of a Spot Market

In order to set the requirements for a spot market for computing resources, a general definition of spot markets and an example are given. In general, the spot market is a securities market, in which goods, both perishable and non-perishable, are sold for cash and delivered immediately or within a short period of time. Contracts sold on a spot market are also effective immediately. The spot market is also known as the “cash market” or “physical market.” Purchases are settled in cash at the price set by the market, as opposed to the price at the time of delivery. An example of a spot market commodity that is regularly sold is crude oil. Crude Oil is sold at the current prices, and physically delivered later [7].

The emergence of electricity wholesale markets is the consequence of privatization of the electrical power production companies. Like computation service, electricity is difficult to store (in large quantities), has to be available on demand, and (unpredictable) demand spikes may occur. Countries that have chosen to operate wholesale electricity markets where power companies offer their electricity output to meet the customers demand, have a number of mechanism to choose from. One model, which is used by the PJM [16] uses central scheduler to balance supply and demand and computes the market price, while the losses over the transmission network are also taken into account: At each network node a "shadow price" is computed, which reflects the cost of providing an additional MWatt-hour at this node.

Another model is that of conducting auctions in various time scales, i.e. auctions for yearly and daily provision of power, with additional spot market that resolves the need for accommodating short-term demand spikes. For example, this model is used by the European Energy Exchange (EEX) [17].

Finally, the Supply Function Equilibria has also been under investigation as the market mechanism of the power grids. It is also worth noting that there are several cases where regulators have intervened due to market failure, with the California market being the most prominent example [8][9].

3.2 Spot Market for Computing Resources

The spot market for computing resources also enables the trading of computing resources “as soon as possible”. It employs a bid and ask mechanism (i.e. a stock market-like double auction mechanism) that enables the trading of computational power. The underlying principle for this mechanism is that of a standard spot market: All parties publicly announce the maximum price they are willing to buy for and the minimum price they are willing to sell for. The spot bids (respectively asks) are put in the spot queue for bids (respectively asks). Matters are more complicated for our system’s spot market than in standard spot markets of storable commodities, since this good is non-storable and that the resource provisioning has to be transparent to the buyer.

3.3 Future Market for Computing Resources

The futures market for computing resources is actually a directory service containing the offers (respectively requests) for resources that are made available (respectively demanded) in a certain time interval. This index of offers and requests is searchable and visible to both bidders and providers. This market for futures complements the spot market. The futures and derivatives are contracts that denote the obligation of a buyer (respectively seller) to buy (respectively sell) at a certain agreed price.

3.4 Requirements for a Market Mechanism for Trading Computing Resources

After introducing briefly the spot and futures market for computing resources, we proceed to provide additional details regarding the unit that is to be traded in these markets (i.e. unit of trade), the format of spot market bids/asks and the futures market requests/offers, the matching algorithm to be adopted, and, finally, how bids and asks are routed in this system.

3.4.1. Unit of Trade

Prior to proceeding with the presentation of the GridEcon market mechanism, we define the unit that is to be traded in the Grid market place. Obviously, the unit must be suitable for the types of Grid applications currently existing or emerging. Computing resource providers offer different types of virtual machines (VMs) for leasing. It is expected that these resources be offered for a minimum desirable price and for a certain time duration within a specific time interval, depending on the providers' supply constraints. An assumption of our model for both the spot and the futures market throughout the paper is that time is discretized in time slots. Note that a virtual machine does not just correspond to a certain computational speed but rather to an entire configuration. The unit of trade, i.e. the VM, is defined through the CPU speed, the size of the main memory, and the size of the harddrive.

Depending on the nature of the tasks that consumers may wish to execute, their demand can be expressed in a multitude of ways. A general type of contract is specified by means of the number of VMs and the time duration. For instance, a company's Web server leases Grid resources when it is critically loaded. This type of consumer need can also be graphically depicted by means of a rectangle (see Fig. 1).

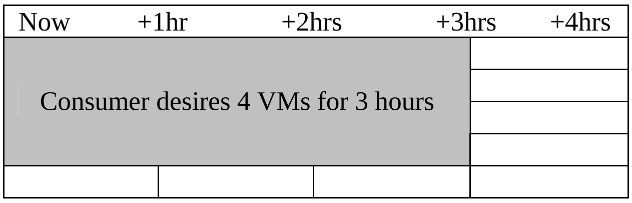

Fig. 1. A consumer's demand for 4 VMs over 3 hours is depicted as a rectangle.

The height of the rectangle denotes the number of virtual machines required at any time of the interval, while the width of the rectangle denotes the amount of time for which these machines are needed.

Another type of contract could be specified by means of computational volume, i.e. a total number of VMs must be made available up to a maximum deadline constraint, so that a certain computationally intensive task can be executed in time. As opposed to the previous case, only the total quantity of computational power is of interest, while the rate of computation provided at the various time epochs is not. This could be the case for data parallel applications. In this case, the consumer needs do no longer correspond to rectangles but rather to areas of rectangles, possibly with a maximum width constraint (i.e. deadline). Since this type of contract can be also expressed (with some effort on the consumer side) in the market place through the earlier type of contract, we will focus on the earlier contract type.

3.4.2. Format of the Bid/Ask

A bid in our system describes the resources required by the buyer. The resources are specified according to:

(1) The type of resource (VM) required,
(2) The quantity of resources (the number of VMs) required,
(3) The start time of the interval for using the resources (VM),
(4) The time duration of using the resources,
(5) The price expressed in €/min/unit, and
(6) The time limit until which the bid is valid. If the time limit is reached without the bid being matched, the bid is removed from the system.

In order to keep the definition of the bid as general and flexible as possible, instead of allowing only fixed values for the number of VMs and the time duration, we allow that the bid can specify whether these constraint values should be met with equality or $\leq$ or $\geq$. Our system also considers two additional constraints regarding the total expenditure and the total volume of computation (see the "Rectangle" column in Table 1). Therefore, since each bid is associated with a set of relation constraints, this allows a richer ontology of bids. A meaningful subset of this ontology is depicted in Table 1.

Constraints:	**Number of VMs**	**Time Duration**	**Total Expenditure**	**Rectangle**
Relations:	=	=	Redundant	Redundant
	$\leq$	$\leq$	Optional	Required
	$\geq$	$\geq$	Required	Redundant

Table 1: Meaningful combinations of constraints for bids.

There are two types of bids in our system, namely future and spot bids. Future bids (or equivalently requests) are the bids for which the start and end times are fixed. For instance, a request could be: "User X bids for 5 VMs of type A to be used for 5 hrs, starting at time 13:00, with price 0.5 €/min/unit". As opposed to requests, spot bids demand to utilize resources as soon as they are available. Spot bids are distinguished from requests by setting the start time to a specific value (e.g. 0) and by the fact that the start time and the end time are continuously moving as time passes. This is performed as long as the bid has not been matched (and up to the maximum time allowed by the expiry of the bid). For instance, a spot bid is: “User X bids for 5 VMs of type A to be used for 5 hrs, starting at time 0, with bid price 0.5 €/min/unit, and time limit 20:00”. In this example, the bid could be executed with a start time of 20:00 the latest.

An ask in our system describes the resources offered, which are specified by the following five parameters:

(1) The type of resources (i.e. VM) offered,
(2) The quantity of resources (i.e. number of VMs) offered,
(3) The start time and the end time of the interval when the resources are available,
(4) The price, expressed in €/min/unit, and
(5) The time limit for which the ask is valid (the expiration time of the offer). That means, the ask will be removed from the system after this time limit.

Similar to bids, there are also two types of asks, namely future and spot asks. Future asks (or equivalently offers) are those for which the start time and the end time are fixed instants in the future. For offers, the end time equals the start time plus the duration, while the time limit also has the same value by default. For instance, an offer looks like: “Provider Y offers for leasing 2 VMs of type A to be used for 8 hrs, starting at time 15:00, with a price of 0.2 €/min/unit”. On the contrary, spot asks offer resources that can be utilized as soon as there is demand for them. Such asks are distinguished from offers by setting the start time to a specific value (e.g. 0). Their main difference to offers is the fact that the start time and end time of asks are continuously moving as time passes. This continues as long as the ask is not matched (up to the maximum time allowed by the expiry of the ask). Therefore, spot asks are more flexible than offers. They offer service of a certain duration over a larger time interval. For instance, a spot ask is: “Provider Y offers 2 VMs of type A to be used for 3 hrs, starting at time 0, with an ask price of 0.2 €/min/unit, and a time limit of 19:00”.

3.4.3. Matching Algorithm

The matching algorithm defines how a bid (respectively request) in the spot market (respectively futures market) is matched by a set of asks (respectively offers). For simplicity reasons, it suffices to adopt a matching algorithm that passes the queues once. In our presentation below, we focus on the spot market. Indeed, the matching algorithm in the futures market is much simpler than that of the spot queue, since the time span of all requests and offers is fully specified, i.e. their start time and end time are decided upon their submission and cannot be changed subsequently, as opposed to spot bids/asks.

Trading in the spot market is performed by means of a continuous double auction mechanism. This is an extension of the standard spot market mechanism. Similarly to the standard mechanism, the spot bids and asks submitted by traders are placed in the *bid queue* and the *ask queue* respectively. Each queue is ordered according to the price and time of submission, with the *bid* queue being sorted in *decreasing* order of price, and the *ask* queue being sorted in *increasing* order of price. If two or more orders at the same price appear in a spot queue, then they are entered by time with older orders appearing ahead of newer orders. An order remains in the queue until it is removed by the system due to order expiration, removal by the user, or if a matching had occurred.

Moreover, the matching algorithm takes into account that spot asks may start providing resources at some later time than now, due to the flexibility associated with the provision of resources. Note that we refrain from adopting a combinatorial approach due to the high computational complexity.

The matching algorithm for the spot offers is as follows: It initially computes the candidate matches to an ask by means of creating a matrix. Each column of the matrix corresponds to a time slot (i.e. the time interval in which service can be provided). Each row corresponds to a provider that can offer service now, with the cheapest being on the top row. A cell of the matrix is marked if the provider can offer computing resources during this specific time slot.

Then, the algorithm attempts to perform a probabilistic matching. In particular, the algorithm starts with the cheapest ask and randomly fills some time slots of bids, so that the provider’s resource availability becomes zero. This means the cheapest ask is fully utilized. It then proceeds with the next cheapest ask and does the same. Note, after the second step, there might be slots allocated to two candidate providers. For these slots, each provider is assigned a probability of moving from this slot. A provider is moved to an empty slot according to a transition probability, which is larger for providers of this slot if they could serve a target slot where the number of providers that could serve the target slot is small.

The algorithm terminates when all the slots are assigned to some provider and thus a match is found. In case there are slots where there is no provider serving it, while there are not any slots with more than one provider, the algorithm has failed to compute a match. Due to the fact that we use a probabilistic matching algorithm, the algorithm can be repeated for a maximum pre-specified number of times until it terminates. If it fails, then it attempts to compute a match for the next time slot, i.e. for the time interval [Now + 1 slot, Now + 1 slot + service duration]. This is repeated until a match is indeed found or the algorithm fails for all time slots (i.e. entire duration) for which the bid is valid.

3.4.4. Routing of Bids and Asks

It is the responsibility of the matching module to be invoked periodically, prior to the beginning of the next slot, in order to compute matches and remove expired bids and asks from the bid/ask queue of the spot market and requests and offers from the futures directory of the futures market. The results of the matching procedure are subsequently passed to the scheduler, the reservation system of the provider, and the accounting system of the market place.

4 Conclusion and Discussion

In this paper, we discussed the rationale for a Grid market for leasing computing resources as well as the relevant key requirements. The GridEcon market place has one major advantage over existing utility computing services (e.g. Amazon’s EC2 service, Sun, HP, and IBM). It allows companies not only to access computing resources, but also to sell spare computing resources. However, in the GridEcon market place, not all providers need also to be consumers and vice versa. Furthermore, the low market power of the participant of the Grid market ensures that the price, though flexible, remains highly competitive. Therefore, not considering market lock-in (i.e. high switching cost), network externalities, or anti-competitive behavior of market leaders, a group of many small computing resource providers (i.e. any company with spare computing resources) could compete with IBM, Google, and Amazon.

The market place for computing resources has certain similarities to the electricity market place. Indeed, since the market price is directly related to demand and supply, it will provide incentives for companies to adapt their usage strategies (e.g. buy more and own less computing power; compute during the night only). Moreover, companies will adjust their in-house computing usage to the competitive market price. Since the market place allows reselling resources that have been purchased, a company can buy resources on the market for a longer time period and resell those resources that are not needed at a shorter time scale.

Finally, the Grid market opens opportunities for a wide range of services (such as insurances, and capacity planning). Those value-added services on top of the market place will provide functionality that addresses certain needs of users. Such additional services may also be developed in some of the aforementioned existing utility computing services. However, it is likely that these services will be developed in such a way that it ultimately increases the profit of the provider, which is of course detrimental to buyers of the computing resources.

Acknowledgement. This work has been funded by the European Commission within the context of the FP6 Project GridEcon, Grid Economics and Business Models, (FP6-2005-IST5-033634).

5 References

[1] Amazon, Amazon Elastic Compute Cloud (Amazon EC2), http://www.amazon.com/gp/browse.html?node=201590011, June 2008.

[2] HP, Utility Computing Services, http://h20338.www2.hp.com/enterprise/cache/308072-0-0-0-121.html, June 2008.

[3] IBM, IBM E-Business On-Demand, http://www-128.ibm.com/developerworks/ibm/library/i-ebodov/index.html , June 2008.

[4] Sun Grid, http://www.sun.com/service/sungrid/index.jsp, June 2008

[5] Google, App. Engine, http://code.google.com/appengine/, June 2008

[6] NASDAQ Stock Market, http://www.nasdaq.com, June 2008

[7] International Monetary Fund, http://www.imf.org/external/np/pp/eng/2005/092105o.htm, May 2008

[8] Joskow P. L.: Restructuring, Competition and Regulatory Reform in the U.S. Electricity Sector. In: Journal of Economic Perspectives, vol. 11, no. 3, pp. 119-138, 1997.

[9] Dekrajangpetch S., Sheble G.B.: Structures and Formulations for Electric Power Auctions. In: Electric Power Systems Research, vol. 54, pp.159-167, Elsevier, 2000.

[10] GridEcon, http://www.gridecon.eu, June 2008.

[11] Altmann, J., Courboubetis, C., Darlington, J., Cohen, J.: GridEcon – The Economic-Enhanced Next-Generation Internet. In: GECON 2007, LNCS vol. 4685. Springer, Heidelberg, 2007.

[12] Techcrunch, http://www.techcrunch.com/2008/04/21/who-are-the-biggest-users-of-amazon-web-services-its-not-startups/ , May 2008.

[13] Altmann, J., Ion, M., Mohammed, A. A. B.: Taxonomy of Grid business models. GECON 2007, LNCS 4685, Springer, Heidelberg, 2007.

[14] XenSource, Inc., http://xen.org/, 2008.

[15] VMware, Inc, http://www.vmware.com/, 2008.

[16] PJM, http://www.pjm.com/contributions/pjm-manuals/pdf/m04.pdf, June 2008.

[17] European Energy Exchange, http://www.eex.com/en/document/32353, June 2008.