Lehrstuhl für
Wirtschaftsinformatik

Information Systems
Management

No. 1

2005

Bayreuther Arbeitspapiere zur Wirtschaftsinformatik

Björn Schnizler / Dirk Neumann / Daniel Veit / Mauro Napoletano / Michele Catalano / Mauro Gallegati / Michael Reinicke / Werner Streitberger / Torsten Eymann

---

Environmental Analysis for Application Layer Networks

Bayreuth Reports on Information Systems Management

UNIVERSITÄT
BAYREUTH

ISSN 1864-9300

Die Arbeitspapiere des Lehrstuhls für Wirtschaftsinformatik dienen der Darstellung vorläufiger Ergebnisse, die i. d. R. noch für spätere Veröffentlichungen überarbeitet werden. Die Autoren sind deshalb für kritische Hinweise dankbar.

The Bayreuth Reports on Information Systems Management comprise preliminary results which will usually be revised for subsequent publications. Critical comments would be appreciated by the authors.

Authors:

Björn Schnizler / Dirk Neumann / Daniel Veit / Mauro Napoletano / Michele Catalano / Mauro Gallegati / Michael Reinicke / Werner Streitberger / Torsten Eymann

Information Systems and Management Working Paper Series

Edited by:

Prof. Dr. Torsten Eymann

Managing Assistant and Contact:

Raimund Matros
Universität Bayreuth
Lehrstuhl für Wirtschaftsinformatik (BWL VII)
Prof. Dr. Torsten Eymann
Universitätsstrasse 30
95447 Bayreuth
Germany

Email: raimund.matros@uni-bayreuth.de

ISSN 1864-9300

Environmental Analysis for Application Layer Networks

Björn Schnizler, Dirk Neumann, Daniel Veit
(University of Karlsruhe)
Mauro Napoletano, Michele Catalano, Mauro Gallegati
(Università delle merche Ancona)
Michael Reinicke, Werner Streitberger, Torsten Eymann
(University of Bayreuth)

Executive Summary.
CATNETS EU IST-FP6-003769 Project Deliverable D1 (WP1.1)
This paper is about Market Engineering for Application Layer Networks.

Document Id.	CATNETS/2005/D1/v1.0
Project	CATNETS EU IST-FP6-003769
Date	Revised Date
Distribution	public

CATNETS Consortium

This document is part of a research project partially funded by the IST Programme of the Commission of the European Communities as project number IST-FP6-003769. The partners in this project are: LS Wirtschaftsinformatik (BWL VII) / University of Bayreuth (coordinator, Germany), Arquitectura de Computadors / Universitat Politecnica de Catalunya (Spain), Information Management and Systems / University of Karlsruhe (TH) (Germany), Dipartimento di Economia / Università delle merche Ancona (Italy), School of Computer Science and the Welsh eScience Centre / University of Cardiff (United Kingdom), Automated Reasoning Systems Division / ITC-irst Trento (Italy).

University of Bayreuth
LS Wirtschaftsinformatik (BWL VII)
95440 Bayreuth
Germany
Tel: +49 921 55-2807, Fax: +49 921 55-2816
Contactperson: Torsten Eymann
E-mail: catnets@uni-bayreuth.de

Universitat Politecnica de Catalunya
Arquitectura de Computadors
Jordi Girona, 1-3
08034 Barcelona
Spain
Tel: +34 93 4016882, Fax: +34 93 4017055
Contactperson: Felix Freitag
E-mail: felix@ac.upc.es

University of Karlsruhe
Institute for Information Management and Systems
Englerstr. 14
76131 Karlsruhe
Germany
Tel: +49 721 608 8370, Fax: +49 721 608 8399
Contactperson: Daniel Veit
E-mail: veit@iw.uka.de

Università delle merche Ancona
Dipartimento di Economia
Piazzale Martelli 8
60121 Ancona
Italy
Tel: 39-071- 220.7088 , Fax: +39-071- 220.7102
Contactperson: Mauro Gallegati
E-mail: gallegati@dea.unian.it

University of Cardiff
School of Computer Science and the Welsh eScience Centre
University of Caradiff, Wales
Cardiff CF24 3AA, UK
United Kingdom
Tel: +44 (0)2920 875542, Fax: +44 (0)2920 874598
Contactperson: Omer F. Rana
E-mail: o.f.rana@cs.cardiff.ac.uk

ITC-irst Trento
Automated Reasoning Systems Division
Via Sommarive, 18
38050 Povo - Trento
Italy
Tel: +39 0461 314 314, Fax: +39 0461 302 040
Contactperson: Floriano Zini
E-mail: zini@itc.it

Contents

Chapter 1

Introduction

1.1 Motivation

The increasing interconnection between computers through the Internet has emerged the vision of Application Layer Networks. Application Layer Networks comprise an abstract view on overlay networks (e.g. Peer-to-Peer networks, Grid infrastructures) on top of the TCP/IP protocol. Their common characteristic is the redundant, distributed provisioning and access of data, computation or application services, while hiding the heterogeneity of the service network from the user's view [ERA+03].

A promising example for Application Layer Networks are Computational Grids. In the Computational Grid, computer resources such as processors or hard disks can be accessed in analogy to the power grid in a plug-and-play environment. A user has access to a reliable virtual computer, which consists of many heterogeneous computer resources. These resources are not visible to the user - such as a consumer of electric power is unaware of how the demanded electricity is being generated and thereafter transmitted to the power socket.

At the moment, most of the research done in the area of Application Layer Networks focuses in particular on the hardware and software infrastructure, such that from a technical point of view, "the access to resources is dependable, consistent, pervasive, and inexpensive" [FK04]. Nonetheless, there are still barriers preventing the deployment of large-scaled Peer-to-Peer networks or Computational Grids.

One of the key issues in building such networks is to determine which computer resources are allocated to which service and scheduled at what time. Most existing approaches such as Legion [COBW00], Condor [FTF+01], or Gnutella [FFRS02, KSTT04] employ optimization algorithms, which allocate and schedule resources based on static system specific cost functions [BAGS02, BAV04].

However, static system specific cost functions typically lead to economically inefficient allocations [BAGS02, BAV04]. These functions do not guarantee that those demanding users will receive their supplied resources who value them highest. Furthermore, these functions ignore the fact that users owning resources only have incentives offering resources, if they are adequately compensated. Compensation requires determining how the supplied resources are allocated among potential buyers and how the prices for the resources are set. However, for implementing economic efficient Application Layer Networks both described aspects are crucial.

Recently, the application of market mechanisms for allocating and scheduling services and resources in Application Layer Networks has been increasingly suggested [BAGS02, WPBB01b]. According to Hurwicz, markets can be an efficient institution to allocate resources (Pareto-) optimal [Hur72]. This is achieved by the interplay of demand and supply and due to the information feedback inherent to the price system. As such, the application of market mechanisms to Application Layer Networks as an allocation and scheduling mechanism is deemed promising.

1.2 CATNETS Scenario

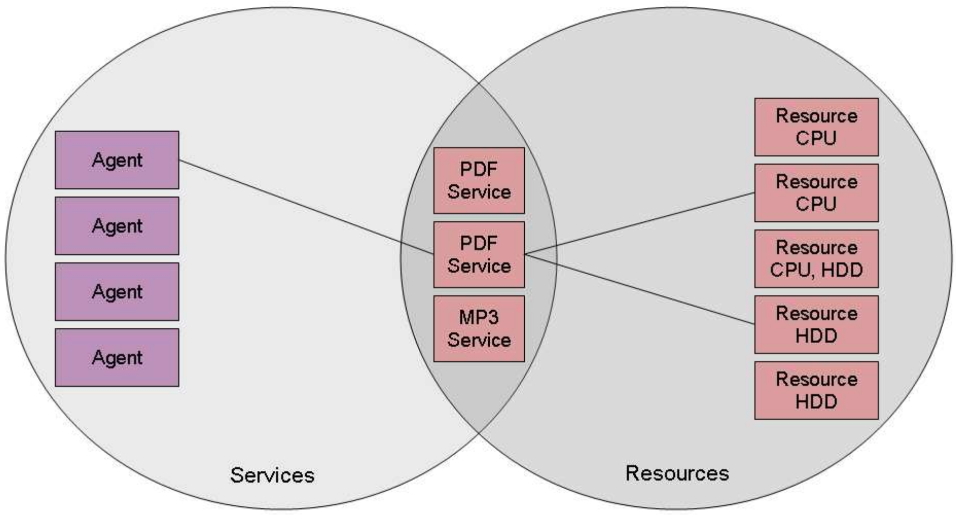

Figure 1.1: CATNETS Scenario: Service Market and Resource Market

The scenario that is envisioned in CATNETS is, that there is a set of services (e.g. services to create a PDF or to convert a MP3 file), a set of agents demanding these services for a specific job (e.g. an agent wants to create a PDF file), and a set of resources capable providing computational resources for executing these services (e.g. a processor, main memory, and a hard disk for creating the PDF file). However, an agent that is requesting a service is unaware of the resources the requested service requires to be carried out.

Figure 1.1 illustrates the CATNETS scenario. An agent is requesting a PDF creator service, which will be allocated to the agent. Furthermore, the required resources (CPU and hard disk) are allocated to the service. The service acts as a trading intermediary, i.e. the service knows what the agents are demanding and which resources are available for executing the services.

The objective of working package 1 (WP1: Market Engineering) in this project is to engineer a market place for such a scenario. This can be achieved by designing (interdependent) market mechanisms for allocating the services among the agents and for allocating and scheduling the resources among the services in two different markets: the Service Market and the Resource Market.

The difficulty in designing such markets is that the underlying mechanisms through which the participants act can have a profound impact on the results of that interaction [Jac02b]. For instance, in a sealed bid auction the bidders simultaneously submit bids to the auctioneer without knowledge of the amount bid by other participants. In contrast, all bids under an open cry auction are available for everyone to see. Thus, in a sealed bid auction mechanism the participants do not learn as much about the valuations of the other participants as in an open cry auction. The higher information feedback may affect the bidding behavior of the market participants and could therefore lead to different outcomes. Furthermore, the market infrastructure (e.g. ways of communication) as well as the business structure (e.g. trading fees) can also influence the behavior of the participants and therefore the outcome of the mechanism [WHN03, Neu04b].

The Market Engineering approach chosen in this WP manages these influences by means of a structured, systematic, and theoretically profound procedure of analyzing, designing, evaluating, and introducing electronic market platforms [WHN03].

Appropriating the methodology of Market Engineering – which will be introduced in Section 2 – the following objectives constitute the outline of this report: In Chapter 3, the Service Market will be analyzed and the requirements a mechanism for this scenario has to fulfill will be defined. In Chapter 4, the Resource Market will be analyzed and characterized, as well as the requirements for this market will be elicited. Finally, Chapter 5 summarizes the contribution and gives an overview on future work.

Chapter 2

Market Engineering Approach

In the general context of engineering, a design method refers to a way, procedure, or technique for solving an individual design problem[1]. These design methods can be either intuitive or discursive:

- Intuitive approaches involve creativity in the form of complex associations of ideas and aim at increasing the flow of ideas (e.g. using the Delphi method [PB84]). However, the results of intuitive approaches strongly depend on the designer's expertise, skills, and experiences. As such, intuitive approaches may fail to achieve suitable solutions for complex problems.

- Discursive approaches are strategies which decompose a complex design problem into several smaller, less complex problems. The design strategy intends to describe a step-by-step procedure to aid the designer in the matching of the unique problem situation along the overall design process with the available design methods.

The design of a market mechanism is a complex and interdependent task. Therefore, the approach of Market Engineering aims at the discursive, goal-oriented development of market institutions.

The Market Engineering Process Model is built upon the engineering design model proposed by [PB84]. In essence, the choice of the engineering design approach allows the explicit anchoring of two fundamental desiderata into the Market Engineering process model:

1. utilization of economic models in the design process and

2. the use of behavioral and cognitive models to determine the needs and requirements of the potential customers and stakeholders.

[1] Parts of this text are taken from [Neu04b]

The first desideratum is very powerful, as it allows the integration of economic modeling into the Market Engineering process. Market Engineering hence is not neglecting prior work on the field of economic design, but can incorporate it into the process. Economic design is essentially concerned with social effects in markets derived from the analysis of abstract resource allocation mechanisms. Analogous to engineering design that deduces solution principles on the basis of physical effects, Market Engineering can make use of the social effects[2] to derive solution principles. The second desideratum is also of great importance in the design of market mechanism, as it primarily addresses social issues with many different customers involved. As such, the engineering design template is additionally enriched by elements of the service development process developed in marketing [SJ89].

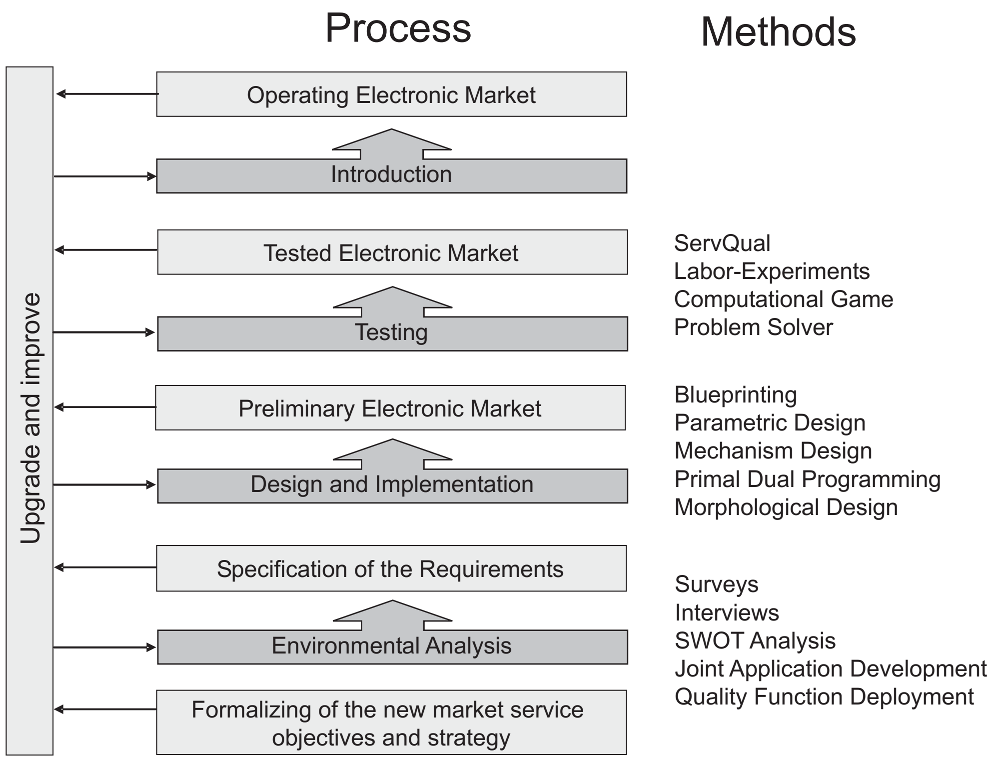

Figure 2.1: Stages of the Market Engineering Process

The corresponding Market Engineering process model and a selection of associated methods are outlined in figure 2.1 [WHN03] and briefly described in the following. The objectives and the strategy that governs the Market Engineering approach stand at the

[2]Social effects are here defined very broadly as all regularities that depend on specific interpersonal interactions.

outset of the Market Engineering process. In the first stage - the environmental analysis - the requirements of the new market mechanism are deduced. In the second stage, the new market mechanism is designed and implemented. Having implemented the appropriate market mechanism, it is tested upon its economic properties and its operational functionality in the third stage. At any stage of the Market Engineering process, there is a decision to be made whether to proceed with the next step or to repeat the prior one. The use of prototypes is again possible at any stage of the process, so that they are left out in the figure. The Market Engineering process not only structures the design process but also provides the designer with a whole array of methods that may support the individual sub-tasks. In the following each stage of the process is described particularly.

However, a detailed description of the single stages and the associated methods can be found in [Neu04b] and [Hol04].

2.1 Phase 1: Environmental Analysis

The environmental analysis stands at the outset of the market-engineering process. Basically, the term environment is used to describe the set of all individual circumstances in a market that are outside the control of the mechanism designer [Hur73]. Among others, these circumstances can be the number of potential participants, their characteristics of the participants (e.g., preferences, risk aversion), their social relations among each other, the characteristics of the resources, or the individual resource endowments [Smi89]. Since the performance of market mechanisms is inherently dependent on the underlying environment, the design of a market mechanism requires profound knowledge about the environment.

The environmental analysis is triggered by the specification of the objectives and the strategy of the designer. Objective of the environmental analysis stage is twofold: Firstly, the identification of a promising market segment for which a market mechanism is considered. Secondly, the analysis of the requirements potential adopters may have regarding the market mechanism. Corresponding with the engineering design process, this stage comprises two different phases: the environment definition and the requirement analysis.

2.1.1 Environment Definition

The central intuition of the environment definition phase is to pinpoint the environment for which subsequently the market mechanism is offered. The environment definition is clearly a marketing task. To systematize the phase more thoroughly, the environment definition is divided into three subsequent activities:

1. Market Definition,
2. Market Segmentation, and
3. Market Targeting.

Firstly, the relevant market is defined. In other words, the market designer is "carving out the arena in which it is going to compete for business" [FA03]. The market definition thus comprises information about the potential market participants, their geographical positions, their specific needs, and so forth. In short, the market definition[3] characterizes the demand-side for the market mechanism that is eventually determined by the definition of the trading object.

Secondly - having defined the relevant market - the designer divides the defined market into several segments. The division into segments is intended to disaggregate total demand into smaller pieces of demand. The smaller pieces of demand are ideally of homogenous nature such that it is easier to fine-tune the market mechanism to the needs of this market segment [FA03, Smi56].

Thirdly, the market segments are evaluated against each other. As a result, the target market consisting of one or more market segments is selected [HR86].

2.1.2 Requirement Analysis

Basically, the target market segment reveals the environment for which the market mechanism is intended. In order to gain potential agents as customers, the market mechanism must match with the particular needs of the agents. The requirement analysis phase consists of a thorough extraction of the potential customers' needs concerning the resource allocation problem and the environmental side-constraints [BCZ92, SSS00]. In other words, the requirement analysis seeks to describe the socio-economic environment consisting of the (potential) number of agents, their preference structure and risk attitude, the number of resources to be offered, their characteristics, and the agents' endowment. [Cra03] summarizes this as follows: "Good market design begins with a thorough understanding of the market participants, their incentives, and the economic problem that the market is trying to solve".

Common elicitation techniques for requirement analysis can be distinguished into three broad categories [BCZ92].

[3]The terms market definition, segmentation, and targeting have been established in marketing for a long time. As such, these terms are also used in this report, although the definition of the term "market" is not exactly corresponding with the institutional definition given here. What marketing literature addresses with the term "market" is the environment, whereas the market mechanisms are left out.

- Observation techniques
- Unstructured elicitation techniques
- Structured elicitation techniques.

In summary, the result of the requirement analysis ideally comprises the following aspects: a description of the socio-economic environment (agents, resources, and preferences), the legal framework, and a requirement list what objectives the market mechanism must attain, and lastly what properties it must have.

2.2 Phase 2: Design and Implementation

The second stage – headlined as design and implementation – comprises the actual design process. It commences when the design problem has been sufficiently specified. Obviously, the aim of this phase is twofold. Firstly, the conceptual design of an appropriate market mechanism on the blackboard, and, secondly, its transformation into a running software system. Following the engineering design process, the design and implementation stage is decomposed into four major phases being the conceptual design, embodiment design, detail design and implementation

2.2.1 Conceptual Design

The second stage – headlined as design and implementation – comprises the actual design process. It commences when the design problem has been sufficiently specified.

The conceptual design step hallmarks the peculiarity that distinguishes the Market Engineering from the service development process. Essentially the design problem is abstracted in a way that the design object is abstracted to its functions. As the design object is the market mechanism, it has the function to allocate resources, provide the customers with information, enforce the allocation, sue infringements and so on. The recourse to abstraction simply means "ignoring what is particular or incidental and emphasizing what is general and essential" [PB84].

Those functions are further divided into sub-functions reducing the overall complexity of the design problem. Then, the sub-functions are distinguished into important (i.e. main) or less important (i.e. auxiliary) functions. Important functions are tackled within the conceptual design phase, whereas the design of auxiliary functions is postponed to the embodiment phase.

Transferred to Market Engineering, the functions are also solved by means of social effects [Ore01]. Different than in engineering design, the entities that cause these effects are not form attributes of material, but a set of rules. For example, the designer may want to satisfy the function of an efficient resource allocation. In this case, the designer searches for market mechanisms that achieve this function. Suggestions from literature show that there are two methodologies available, which are concerned with the design of market mechanisms on the abstract level, namely primal/dual programming, and mechanism design [KP03]:

Primal/dual Programming [BO00]: The technique of primal/dual programming models the allocation problem as the primal the price determination as the dual problem. If the complementary slackness conditions are satisfied, the allocation at the given price is efficient. This approach can be translated into an iterative auction format. The auctioneer announces the standing highest price – the agents report their utility maximizing bid without considering the other agents' reactions. The auctioneer collects those new bids and determines the new provisional allocation. Then, the auctioneer verifies whether the complementary slackness condition is satisfied. If so the auction terminates with the given price and allocation, if not the price is raised. This approach – although mathematically elegant – requires the agents to report their bids myopically. This assumption is very rigid, as it assumes the agents to behave as price takers. If this assumption is violated the approach does not produce reliable results[4].

Mechanism Design [Jac02b]: Mechanism Design is a very elegant way to compute the optimal mechanism that attains the desired objective[5]. In essence, the mechanism that maximizes the objective function (e.g. social welfare) subject to the individual rationality constraint and the incentive compatibility constraint is the optimal mechanism [Led93]. Obviously, the mechanism design approach is very information intense, as it requires detailed information about the distribution of the environment. In the engineering practice, the mechanism design approach is accordingly of limited use[6].

Both approaches generate an abstract description of the market mechanism consisting of an allocation and price function.

The resulting abstract solution descriptions to all functions are aggregated into concepts. The concepts are furthermore supplemented by a calculation of profitability predicting the chances of the envisioned market mechanism in the competition. Finally, it

[4]One approach to heal primal/dual programming is to set incentives such that the agents behave as price takers [AM02].

[5]Alternatively, it is also possible to compute the outcomes of a given mechanism in a specific environment. This strategy is typically adopted by auction theory [Mil04]

[6]Other reasons can be found in [Neu04a]

is decided upon which concept – including abstract descriptions of the service enriched by profitability estimates – is further adopted.

2.2.2 Embodiment Design

While the conceptual design phase is concerned with the formulation of the problem and the search for abstract solutions, the embodiment design refines the abstract concepts to blueprints. A blueprint denotes a model that is more concrete than the concepts, but still independent of implementation details. In other words, the concept comprises at most a verbal description of the market mechanism. As such, many different blueprints can be found that realize the same concept. During the embodiment design this verbal descriptions are transformed into a model with sufficiently low level of abstraction that traditional design techniques may be applied in order to implement it: the concept becomes form [PB84].

Those blueprints comprise at least four main aspects:

Identifying processes One of the primary strengths of a blueprint is that it can visualize the processes necessary for providing the transaction service. A blueprint for the market mechanism aggregates the refined (sub-) functions of the conceptual design phase to processes. Apparently, the determination of a blueprint requires the specification of the interconnections among the (sub-) functions. Once, the processes are identified they can be analyzed upon fail points or time frames.

Isolating fail points The identification of the processes helps to analyze the critical process steps. In particular, the blueprint as analysis tool allows optimizing the entire process. Critical process steps for instance can be redundantly secured against failure by fail-safe processes or certain time-critical process steps can be automated [Sho84]. For any market mechanism the engagement of the customers is potentially critical. Generally the interface between customer and service system affects the customer satisfaction, which in turn strongly influences customer loyalty [JS95]. Unsatisfied customers may defect and will subsequently not be available to act as a co-producer of the service. This may exacerbate the quality of the overall service.

Establishing time frame A blueprint can also reveal information about the execution time, i.e. how long it takes to execute the single process steps. On the other hand, does the blueprint help to accelerate certain process steps. By identifying the potential bottlenecks the service system can be streamlined. Moreover, it is possible to establish reasonable time-of-service-execution standards.

Establishing cost analysis As soon as the processes and the corresponding time frames are established, it is possible to develop a thorough cost analysis. The cost analysis can also render the costs per any sub-process given a certain service system.

The main difficulty in embodiment design is that commonly "developers translate the subjective description of a need into an operational concept that may bear only the remote resemblance of the original idea" [Sho84]. As blueprints are, "[...] more precise than verbal descriptions of the service processes and therefore reduce ambiguity and the likelihood of misunderstandings that may originate from them" [HRRM00]. Furthermore, the technique of blueprinting almost prevents the designer from conceptual errors, because the blueprint allows " [...] the creation, study, and testing of services conceptually on paper before costly implementation" [HRRM00]. There are many methods that satisfy these four objectives of blueprints. The most common models that are presented in the context of blueprinting are flow diagrams or PERT (i.e. program evaluation and review technique) charts[7] [HRRM00, Sho82, Sho84]. The methodology Blueprinting tailored to embodiment design converting concepts into protocols is described in [Neu04a].

2.2.3 Detail Design and Implementation

The detail design phase starts out with the layout, which describes the central aspects of the system, but is still at a level that is not implementable. Detail design further refines the layout into a fully-fledged system model that is subsequently implemented. Apparently, this phase accounts for the software engineering effort in Market Engineering.

From the software development point of view, the precedent design phases of the Market Engineering process can be subsumed under the term requirement analysis. Different than the traditional sequence of interviews, or the use of Joint Application Development (JAD) meetings conducted by professional modelers, the Market Engineering process provides the designers with a systematic approach to collect design information from the experts. In other words, the precedent design phases of the market design process converts the activities of gathering, figuring out, and communicating what to build [HB95] into a closed discursive approach[8].

By doing so, the Market Engineering process supports the arguably most important step in the requirement analysis [DHDJ99].

Once the designer has a clear idea how the market mechanism will look like (by

[7]Broadly speaking, PERT charts visualize tasks, durations, and dependencies among task. Each chart starts with an initial node from which the first task(s) originates. The tasks are represented by arrows, which indicate the identifiers of the tasks, the durations, the number of people assigned to them, and sometimes even the names of the employees involved. The arrow points at another node, which identifies the start of another task, or the beginning of any slack time. Related techniques are CPM or GANTT charts.

[8]The Market Engineering process thereby follows the four phases of the requirement analysis, conceptual design, logical design, validation and formal specification proposed by [BCZ92, Zmu83].

means of the layout), an ordinary software development process can be started. State-of-the art approaches like the V-model [Som01, TH93] supplemented by methods such as the FUSION [CAB$^+$94] or Coad/Yourdon [CY91a, CY91b] and tools such as UML [OF99, RJB99] are available such that the software engineering process will not further be elaborated.

Detail design is, however, more than software engineering – detail design phase is also concerned with the concretization of the business rules. Up to this point, only the key data concerning the business rules such as target costs and price ranges for the market mechanism exist. Once the properties of the market are clarified, reliable pricing schemes can be developed.

The end of the detail design and implementation is reached, when the market mechanism is fully implemented.

2.3 Phase 3: Testing

Having implemented the market mechanism, it is tested. Stage 3 denotes, however, not testing in general but the final acceptance test before the market mechanism is rolled out. The inclusion of a separated testing stage may also account for the case that the designer has sourced the implementation task out. The designer will then only accept the software system if it passes their acceptance testing. Apparently, the inclusion of a testing stage does certainly not exclude testing along the entire design process; it rather provides the designer with decision support whether or not the system can be launched in the field.

Nevertheless, the term acceptance testing is used here in a broad sense meaning all activities "used in quality control operations to decide between acceptance and rejection of production lots based upon an inspection of selected items" [MCM75]. What is tested is the software quality, and the quality of the service. While the former testing checks the functionality of the service system, the latter refers to the outcome of the market mechanism in economic terms (such as efficiency). Thus, the testing stage comprises two different testing phases before release: functionality and concerning economic performance testing.

When the market mechanism passed through the functionality (acceptance) test, it is tested concerning its economic performance. Now the question arises, how economic performance can be evaluated? This question is extremely difficult, as the outcome is determined by the behavior of the agents and not by the mechanism. Milgrom summarizes the problems with behavior as follows: "Behavior is neither perfectly stable over time, nor the same across individuals, nor completely predictable for any single individual. Useful analyses must be cognizant of these realities" [Mil04]. In the discipline of economic engineering there are two approaches to address the evaluation problem of institutions:

- Axiomatic Approach
- Experimental Approach

The axiomatic approach imposes a couple assumptions upon human behavior and calculates equilibrium strategies. With those equilibrium strategies, the performance of the institution can be calculated.

Alternatively, the experimental approach exposes humans to the institution, who will autonomously form their strategy. To make the laboratory experiment comparable, the demand and supply situation is induced to the participating human by means of a monetary incentive scheme. Hence, the performance of an institution depends on the real social interplay among the agents.

2.3.1 Axiomatic Approach

Standard economic theory has bred out two standard paradigms with different assumptions upon agent behavior pervading economic modeling:

1. The first paradigm is bolstered by welfare economics. Agents are in these models assumed to maximize their individual utility no matter what the others do. In a market context this implies that the buying agents are price-takers, as they take the announced price as given and act accordingly.
2. The second paradigm pertains to game theory. Agents are assumed to choose their optimal strategy conditional on the rival agents' reactions.

The first paradigm usually assumes a behavior that can be described as myopic-best response. In myopic-best response the agents behave such that their utility is maximized, taking the behavior of all other agents as unchanged. Apparently, it is assumed that the own action has no impact on the other agents' reaction. Equilibrium is reached when no agent can improve its utility [KP03].

The second paradigm stems from game theory. Accordingly, the agents choose their utility-maximizing strategy conditional on the rival agents' reaction. An equilibrium is characterized by the fact that all agents play their best-response strategy to each other. This implies that a single agent cannot increase its utility by a unilateral deviation from the equilibrium [FT00]. Those strategies are easy to compute, as it involves no anticipation of the other agents' strategies [WMMRS03].

By the classification into game-theoretic and price taking behavior it implicitly raises the question, how complex a problem must be before the game-theoretic rationale loses its predictive power [KP03, Led93]. Accordingly, it is often assumed that the second paradigm is often used to reflect situation, where only a small number of agents are participating. The first paradigm may reflect agent behavior fairly well, when the number of agents is sufficiently large or when there is uncertainty concerning the preferences.

However, despite the nice properties of myopic-best response, this strategy has two major drawbacks. Firstly, myopic-best response is only meaningful for iterative auctions. Secondly, it is in many cases not rational, as there exists better strategies. Myopic-best response would thus neither maximize the agents' utilities nor attain allocative efficiency [MMORW04]. Nonetheless, myopic best response models work well when the number of agents is large.

Analytical Models It is principally possible to solve for these equilibria analytically. Analytical models rely on "stark and exaggerated assumptions to reach theoretical conclusions" [Mil04]. Typically, mechanism theory assumes that:

- Agents' valuations are well formed and describable in terms of probabilities,
- Differences in the valuations are fully reflected by differences in the information, and
- Agents maximize their utility and expect other agents to be utility maximizing entities as well.

Additionally, the models impose even more restrictive assumptions (such as symmetry of the agents etc.) upon the models to make them tractable [Mil04]. The conclusions of the analytical models may "fail miserably" when the assumptions may not reflect reality. In reality the strategic behavior and the environment is more complex than in the models. In those cases optimal (i.e. utility-maximizing) strategies are rarely found. As a consequence, analytical models can be used for developing intuition but not for evaluation of market mechanism. Furthermore, as [Mil04] noted, do state-of-the-art models only capture a very small subset of issues of the market mechanism.

Apparently, economic performance testing has to cope with the problem that the analytical derivation of the optimal strategies is impossible due to the inherent complexity the models want to capture. Restricting attention to ad-hoc strategies and their impact on the economic performance may, however, entail false conclusions, as these ad-hoc strategies can be arbitrarily far from the optimal strategy [RWMMO04].

Computational Methods One direct way to evaluate the performance of market mechanism is simply to the equilibria. For myopic best response models this is straightforward, as the agents' strategies can be easily computed with having to deal with bidding interdependencies. For the case of game-theoretic modeling this is more complicated, as one has to compute the optimal strategies in equilibrium. For computing Nash equilibria numerically, current computational game solver, such as Gambit, fail to compute even small problems in a reasonable amount of time [KP03]. This failure stems from the size of the strategy space, which is simply huge.

Another possibility is to reduce the strategy space and compute the equilibrium based on this strategy space. It still can happen that the problem is computationally intractable, as Wellman et al. report [WMMRS03].

Several different approaches for computational methods can be identified:

- Computational Game Solver: As aforementioned, Gambit is computational game solver for solving finite games. In essence, it uses the full pay-off matrix and successively eliminates strongly dominated strategies. Then, it performs a simplicial subdivision algorithm for finding at least one mixed strategy equilibrium to any n-person game [MMT00]. The major drawback of computational game solvers is, as aforementioned, that even small games cannot be processed within a reasonable timeframe [KP03][9].

- Evolutionary Tournament: Originally the interpretation of a Nash equilibrium aims at its evolutionary character. Accordingly, the Nash equilibrium denotes the end point of an evolutionary process, where the agents subsequently adjust their strategies. This evolutionary intuition can be formalized by replicator dynamics.[10] In replicator dynamics, it is the idea that not the individual agents learn but the society as a whole. Initially, a set of pure strategies and the proportion by how many agents of the population these strategies are adopted is given. A strategy that has proven its usefulness is replicated and thus stronger represented in the population. Once the proportions of the population playing a pure strategy remains constant and none of the strategies are extinct, a Nash equilibrium is reached [FL99]. Reaching such a fixed point means that all pure strategies are

[9] Kalagnanam and Parkes report about recent developments in the area of computational game solver [KP03]

[10] Replicator dynamics originate in theoretical biology, where they were intended to model shifts in the proportions of genes or species of different fitness.

equally performing well. Another, more plausible interpretation is that - in case the proportions converge - all agents are following mixed strategies according to the population proportions [RWMMO04][11]. It is, however, not guaranteed that the evolutionary tournament indeed converges. This is the major drawback of the evolutionary tournament method[12].

- Genetic Programming: Another evolutionary technique proposed for simulation of social processes is genetic programming [Pri97]. Basically, agent behavior is modeled as a mental process. This mental process corresponds to an internal evolutionary algorithm. Not the entire strategies from the other agents (as in the replicator dynamics example) but their decision processes are replicated. Broadly speaking, when the decision process is conceived as a tree of operations, than this tree mutates. The motivation [...] is to discover which kinds of strategies can be maintained by a group in the face of any possibly strategy alternative. If a successful alternative strategy exists, it may be found by the ŞmutantŤ individual, through conscious deliberation, or through trial and error, or through just plain luck. If everyone is using a given strategy, and some other strategy can do better in the environment of the current population, then someone is sure to find this better strategy sooner or later [Axe84]. As with the evolutionary tournament technique, it is also conceivable that evolutionary programming also fails to converge to a stable strategy.

- Agent-Based Simulation: Intelligent software agents are computational entities that can be viewed as perceiving and acting upon its environment and that is autonomous in that its behavior at least partially depends on its own experience. The use of autonomous interacting software-agents for representing economic agents has brought up the research area of Agent-based Computational Economics (ACE) [Tes02] building economies as independent evolving systems. Modelling individual strategies enables economists to study markets, market behaviour and its development over time and market microstructure under certain institutional and environmental rules. Agents applied in simulations normally use simple decision rules, learning algorithms, or statistical analysis to adapt their strategies. [Tes02] provides a detailed overview on ACE research and describes studies of market simulations in electricity and financial markets (for examples see also [AHL+96, MY01, Pri97, SHC96, WCW04]).

[11] In other words, a mixed strategy equilibrium is found. It can be verified whether this equilibrium is indeed a Nash equilibrium of the static game. In those cases the mixed strategy must be a best response to itself.

[12] From the theoretical point of view, replicator dynamics have been criticized as an inadequate representation of social processes. The main claim is that replicator dynamics reflect biological processes but not social processes. An overview over the criticism can be found at [Cha98]

2.3.2 Experimental Approach

The depiction of the axiomatic approach demonstrates that economic theory "is endowed with numerous theories, which are judged on logical completeness" [Dux95]. However, those (consistent) theories are developed in rather abstract scenarios. A check, whether those theories also hold in reality is a different question. Empirical tests and evidence can fill this gap by providing means to verify or reject those theories. However, there may be cases, where the available data are inadequate. This is where laboratory experiments can step in. Laboratory experiments are a fairly inexpensive method to generate data. The gist of laboratory experiments is their ability to create a test-bed, in which all relevant influencing factors of the environment can be controlled. In such a controlled environment it is possible to test theories or to discriminate between them [Smi94]. Furthermore, by the use of experiments also the effect of (new) mechanisms can be extracted, which are too complex to express them in a coherent formal model [MRS93][13].

Apparently, there exists a tool not only for testing theory but also for economic performance testing: Testbed experiments involve the actual implementation of a process. The purposes of a testbed are to determine if the process can be implemented and how it works once it is implemented. [. . .] More recently the terms "proof of principle" or "proof of concept" have been used to capture the motivation and interpretation of the research." [Plo94]. The use of laboratory experiments as engineering tool is not undisputed, though. The dispute stems from the fact that experiments abstract from reality by sketching a simplified view upon the (socio-) economic environment. As such, "[. . .] there is a question about the transferability of results from experiments" [LK98]. Two popular streams of arguments have developed discussion the pros and cons of experiments.

1. The first stream assumes that the agent behavior in laboratory experiments resembles behavior in real settings. This implies that the results of experiments can be directly transferred to real settings [EL93].
2. The second stream is a little bit more cautious, as it does not assume that behavior in experimental and real settings are alike. It basically perceives that a theory that does not hold even in a controlled environment of a laboratory experiment will also fail in more complex real settings. On the other hand, there is no guarantee that a theory that proves itself in an experimental setting will likewise do in

[13] Undoubtedly, there are many more reasons why experiments are important in economics. Smith for example mentions five additional arguments. For example experiments can help to explain the causes of theory's failure. Furthermore, experiments are adequate to distinguish the robustness of mechanisms. Last but not least experiments may serve as a source of empirical phenomena that needs explanation.

real settings. That means negative results can automatically be transferred while positive ones cannot.

What is undisputed is the fact that testbed experiments can reveal information about the way institutions work on the performance. As such, experiments are an indispensable tool of Market Engineering. It provides the only method that can currently capture the effect of the entire institution. As such, the experimental approach is currently state-of-the art in testing innovative institutions [Mil04].

Chapter 3

An Application for the Service Market

3.1 Environmental Analysis

When designing a market mechanism, one has to be aware of the underlying environment for which the market mechanism is to be designed. Thereby, the environment description embraces information about the potential market participants, their needs, the characteristics of the traded resources, and their endowments. Once the environment description is available, the designer can elicit the requirements for the market mechanism from the potential participants of that particular environment [Neu04b].

Corresponding to the engineering design process, the design stage comprises two different phases: the environment definition and the requirement analysis.

3.1.1 Environment Definition

The market for trading services is spanned around service agents as sellers [1] (e.g. a PDF creator service) and consumers as buyers (e.g. agents requesting a specific service). The services in form of sellers sell one or more specific services and are responsible for providing the traded services to the buyers and for procuring the required resources for the services on the resource market (cf. chapter 4).

For instance, if a buyer is allocated a MP3 converter on the service market, the selling service agent is responsible for providing the software (in form of the MP3 service) as well as the required resources (e.g. CPU, RAM, and hard disk for the conversion).

[1]From a technical point of view, a service can be interpreted as an agent selling different services during a specific time span.

The products traded on the service market are completly standardized. For example, an instance of a PDF creator traded once does not differ from a PDF creator instance traded at a later time.

The number of participants as well as the number of tradable services in a service market is difficult to determine. In practice, there there exists no comparable market at the moment. However, related markets (e.g. Seti@Home, Gnutella) as well as large scaled markets (e.g. XETRA) may be a clue for determining the number of participants and the number of different services:

Seti@Home Seti@Home was founded at the UC Berkley and enables users to participate in the search for extra-terrestrial intelligence by analyzing data from the Serendip radio telescope in Puerto Rico[2]. Seti@Home is a distributed system with a central coordinator. Seti@Home has more than 5.000.000 people participating in the network and more than 10.000 active users per day [Set05].

Gnutella Gnutella is a Peer-to-Peer file sharing system which allows users the worldwide access and provision of information (e.g. music files). [AH00] analyzed the network traffic and observed in a 24-hour period more than 35.000 connected peers.

XETRA XETRA (Exchange Electronic Trading) is a worldwide electronic securities trading system operated by the Deutsche Boerse Group in Germany. XETRA implements a centralized auctioneer and, in peak times, handles at the German Stock Exchange more than 500.000 orders a day [AG05].

Analyzing related systems and mechanisms gives a first impression of the number of participants and the number of tradable services. However, a detailed analysis has to be made in conjunction with the technical analysts in the CATNETS project.

Having analyzed the participants and the traded products of the service market, the requirements which an adequate market mechanism for this environment has to fulfill can be elicited.

3.1.2 Requirement Analysis

Ideally, the resulting market mechanism reflects the users' requirements. For the service market in the CATNETS scenario, the requirements stem from two areas: The theory of Mechanism Design and the underlying service market environment.

[2]See `http://setiathome.ssl.berkeley.edu/` for details (accessed: 04.02.2005).

Requirements stemming from the Theory of Mechanism Design

The Theory of Mechanism Design takes a systematic look at market mechanisms and their outcomes based on game theory [Jac02b]. Within the scope of Mechanism Design it is the goal to investigate a mechanism that is applicable in certain situations. Achieving this objective, the designed mechanism has to fulfill the following general requirements [Jac02b, Par01a, Neu04b]:

Allocative Efficiency: The mechanism should determine an efficient allocation. Assuming transferable utility among all participants, this is achieved if the total value over all participants is maximized. Allocative efficiency can be defined in an ex-post and ex-ante sense. Ex-ante efficiency takes preferences over expected allocations in consideration, whilst ex-post analyzes preferences over realized allocations.

Budget Balancy: The property of budget balancy is concerned with whether the mechanism requires payments from outside the system or not. A mechanism is budget balanced if all payments made to the mechanism are redistributed among the participants. Neither funds from the system are removed nor is the system subsidized from outcome. A weaker property is the concept of weak budged balance, i.e. net payments are made from the participants to the mechanism, but no net payment from the mechanism to the participants.

Incentive compatibility: A mechanism is required to be (approximated) incentive compatible. This is the case if all participants report their preferences truthfully. Participants may not have an incentive to untruthfully report their preferences in order to increase their individual utility. A mechanism is strategy-proofed if truthful revelation of the preferences is a dominant-strategy equilibrium.

Individual rational: A mechanism is individual rational, if any participant achieves at least as much utility from participating in the mechanism as without participating.

Computational tractability: Computational tractability considers the complexity of computing the outcome of a mechanism based on the agent's strategies [KP03]. With an increasing size of the message space the allocation problem can become very demanding. Computational constraints may delimit the design of choice and transfer rules.

Communication complexity: Communication complexity considers the minimization of the amount of communication that is required to converge on a desirable global outcome of the mechanism [San99]. With an increasing number of participants, it is required that the communication effort is minimized.

Requirements stemming from the service market environment

The requirements elicited from the mechanism design theory do, however, not fulfill all requirements for the service market in the CATNETS scenario. The underlying environment and its properties stem for more detailed and domain specific requirements as elicited in the following:

Simultaneous trading: The mechanism for the service market requires that multiple sellers and multiple buyers can trade and simultaneous. This can be realized by providing a double auction mechanism. Furthermore, providing a double auction installs competition on both sides and is thus deemed promising to yield an adequate allocation

Immediate service allocation: It is required that suitable buyer orders are matched immediately with suitable seller orders. This requirement can be realized by providing a continuous trading mechanism.

No partial execution: The mechanism is required to avoid partial executions of services. This bases on the fact, that a partial execution of a service is useless for a buyer.

3.2 Meeting the Requirements

As shown in the environmental analysis (cf. Section 3.1), services can be characterized as standardized resources. In this context, the use of auctions is an efficient way to allocate these standardized resource, as well as to determine its prices. An auction is characterized as a market institution with an explicit set of rules determining resource allocation and prices on the basis of bids from the market participants [MM87a].

Treating auctions for standardized resources, five common auction formats are usually analyzed which will be introduced in the following. Figure 3.1 represents a classification schema for these auction formats based on [WWW01].

3.2.1 Single Sided Auctions

Single sided auctions are mechanisms, where only buyers or sellers can submit bids or asks ($1 : n$ or $m : 1$ relations). The most prominent single sided auctions are the Vickrey Auction, the Dutch Auction, and the English Auction.

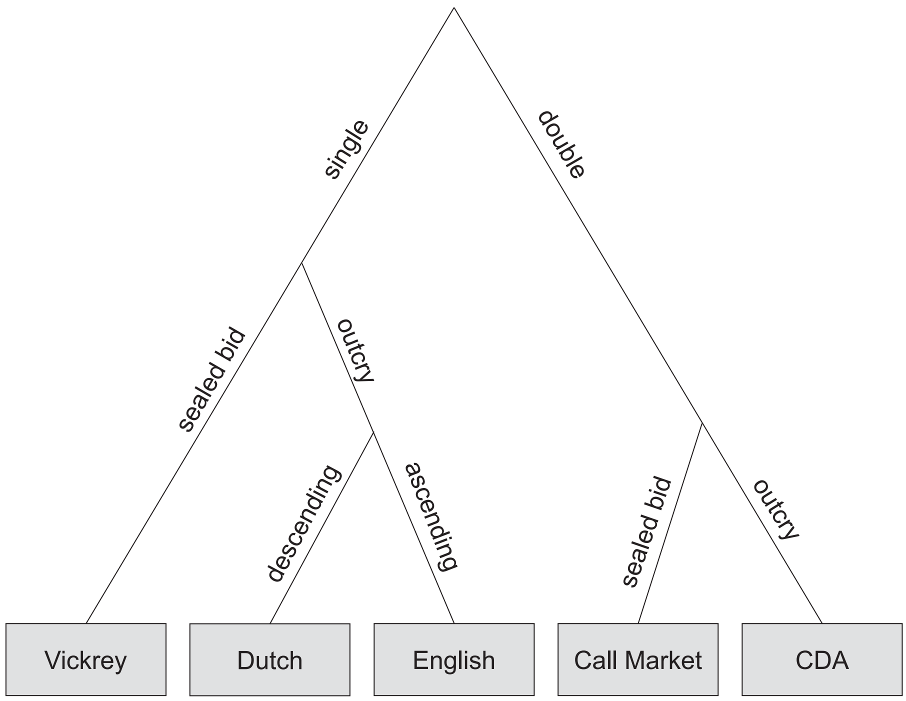

Figure 3.1: Classification schema [WWW01]

Vickrey Auction The Vickrey auction is a sealed bid auction. The bidders submit one single bid in a sealed envelope to the auctioneer. Finally, the bidder who submitted the highest bid is awarded with the item at the price of the second highest bid. It can be shown that truthfully bidding is a dominant strategy [GL77]. Thus, the English and the Vickrey auction are dubbed strategically equivalent, as their strategies are alike.

Dutch Auction In the Dutch auction, the auctioneer calls out a price and lowers this price incrementally as long as no bidder is willing to accept it. Once a bidder accepts the pealed price this bidder wins the auction and has to pay his bid. As a strategy in the Dutch Auction, the bidder has to define a price he is willing to accept [Mil89]. This price will be below his valuation in order to draw positive utility from the auction. Bidding below the own valuation is termed shading.

English Auction The English auction is the oldest and perhaps the most prevalent auction format [Kri02]. Central to the English auction is the auctioneer, who conducts the bidding process. More precisely, the auctioneer calls out a (low) price and increases that price incrementally as long as there are more bidders interested. When the second last bidder refuses to stay in the bidding process, the last bidder receives the item. Obviously, the price the winner has to pay equals the second highest bid (if necessary the price is one increment higher to beat the second highest bid). The dominant strategy of the English auction under private value settings can thus be headlined as pay up to your valuation.

Single Sided auctions are – from an economic point-of-view – well understood and applied successfully in different domains. However, single-sided auctions do not enable the simultaneous and continuous trading of multiple buyers and multiple sellers. Reviewing the defined requirements in section 3.1.2, these auction types cannot be applied for the service market in CATNETS.

3.2.2 Double Sided Auctions

Double-sided auctions or shortly double auctions are those auctions where competitive bidding takes places on both sides ($m : n$ relation). In comparison to traditional single-sided auctions, double auctions have received much less attention by modern economic theory.

For double auctions, where many buyers and many sellers compete against each other, it is difficult to game-theoretically model the strategic behavior of buyers and sellers [MM87b]. Thus, theoretical models typically focus on the very simple two sided institutions, where agents engage directly in bargaining over the terms of exchange.

The k-double auction is thereby the simplest form of a double auction. In essence buyers and sellers submit their bids in a sealed envelope to the auctioneer. The auctioneer forms from the individual bids demand and supply schedules and determines the prices where demand and supply are balanced. Using a given parameter $k \in [0, 1]$ a market clearing price $p = (1 - k)a + kb$ is chosen from the interval $[a, b]$ confining the range of all possible market clearing prices. Buyers with bids higher than this market-clearing price will then trade to those sellers, who submitted lower bids than the clearing price [SW02]. The k-double auction was initially introduced by [CS83]. Originally, they consider a bilateral-monopoly single unit-case with private values that are independently drawn from known uniform distributions [Fri91]. In essence they find linear Bayesian Nash equilibrium bidding strategies, which can miss mutually beneficially trades with a positive probability. These inefficiencies can be explained by the fact that both the buyers and sellers have an incentive to misrepresent their true preferences. In this context, [MS83] show that in environments with incomplete information some of those ex-post inefficiencies are inevitable [MS83]. However, [RSW94a] demonstrate that these efficiency losses decrease, as the numbers of buyers and sellers increase. This rise in efficiency occurs because each agent's strategy converges rapidly towards truthful reporting – in other words the agents become price-takers [RSW94a]. Accordingly, the increase of agents induces a price taking behavior and thus increases the efficiency of the double auction. [Wil85] introduces the many-buyer/many-seller double auction model as a game of incomplete information. He also comes to the conclusion that double auctions are incentive efficient in large markets with strategic traders [Wil85].

A key consideration in double auctions is the timing of the clearing process, i.e. the timing of determining the auction winners and thereby the allocation of the goods. Double auctions can be either cleared continuously (Continuous Double Auction) or periodically (Periodic Double Auction, Call Market):

Continuous Double Auction A Continuous Double Auction (CDA) is a double auction where buyers and sellers simultaneously and asynchronously announce bids and offers. Whenever a new order enters the market, the auctioneer tries to clear the market immediately. Thus, the CDA is advantageous especially in terms of immediacy.

Call Market A Call Market is a double auction with periodic uniform clearing, e.g. the auctioneer clears the market every fives minutes. All orders in a period are collected in an order-book and will be cleared periodically.

3.2.3 Summary

The review of the requirements elicited in section 3.1.2 implies, that the implementation of a Continuous Double Auction (CDA) fits the defined requirements for the CATNETS' service market.

Single sided auctions have the drawback to be either provider- or consumer-oriented and thus asymmetric. Double auctions enable providers and consumers to offer bids simultaneously and thus, the application of a double auction for the service market is deemed promising.

The requirements stemmed from the theory of Mechanism Design (cf. Section 3.1.2) can be (approximately) fulfilled by a CDA. It is possible to design an (approximative) efficient [3], budget balanced, individual rational, and (approximative) incentive compatible double auction mechanism. Furthermore, the computational and communication complexities of a CDA are manageable in large-scaled markets. For instance, the XETRA system at the German Stock Exchange also implements a (modified version) of a CDA [AG05].

Furthermore, the requirements coming from environment of the service market (i.e. simultaneous and continuous trading) can also be fulfilled by a CDA.

[3] [MS83] show that there cannot be any double auction (exchange) which is efficient, budget balanced, and individual rational at the same time. However, [PKE01] show that the efficiency of an exchange can be approximated while cleaving on the budget balance and individual rational properties.

3.3 Testing

Mechanisms can be evaluated and compared according to many types of criteria. In the following discussion, a set of common metrics for measuring the quality of markets is outlined. However, a detailed and formalized criteria catalogue requires the formalizing of a concrete market mechanism and is thus deferred to future work. Table 3.1 summarizes the metrics which base on literature surveys of [Neu04b, Par01a, Bud03].

Group	**Criterion**
Efficiency	Allocative Efficiency Informational Efficiency
Optimality	Revenue Maximization Correctness
Solution	Stability
Feasibility	Budget Balance Incentive Feasibility
Tractability	Communication Costs Computation Costs

Table 3.1: Evaluation criteria

3.3.1 Efficiency

Efficiency denotes the capacity to produce desired results with a minimum expenditure of energy, time, or resource [MW02]. Applied to market mechanisms, the criteria allocative efficiency and informational efficiency can be defined:

Allocative Efficiency In order to measure allocative efficiency in a competitive framework we have to evaluate the buyers and sellers surplus and payoffs. The Marshallian aggregate surplus is

$$S = \sum_{i=1}^{I} \phi(x_i) - \sum_{j=1}^{J} c_j(q_j) \tag{3.1}$$

It is a measure of welfare in a market, or more precisely, of the total utility generated by consumption less the costs of production. In order to compute it, we have to know agents' reservation prices and the actual transactions prices in every period. Buyer's payoff is given by the difference between his own reservation value v and the actual price p. Similarly, seller's payoff is the difference between the price p and his reservation value s. Overall welfare can be formalized as follows:

$$S = \sum_{i=1}^{I}(v_i - p_i) + \sum_{j=1}^{J}(p_j - s_j) \tag{3.2}$$

Competitive equilibrium results from the maximizing behavior assumption of agents. Overall welfare as defined in (3.2) is maximized in equilibrium and therefore the resulting allocation is efficient in the *Paretian* sense. The concept of Pareto efficiency is deeply rooted in the individualistic methodology on which competitive equilibrium analysis is built. Indeed a market allocation is Pareto superior to another one if the welfare of at least one agent in the economy has been improved without reducing the welfare of another agent. Under the assumption of stability of demand and supply curves it is possible to compute efficiency by evaluating equilibrium quantity x^* as the quantity in correspondence of the intersection of market curves. Thus, a measure of efficiency is the ratio between actual market quantity q_t and potential one x^*,

$$eff = \frac{q_t}{x_t^*} \tag{3.3}$$

Informational Efficiency Markets are informational efficient if prices reflect all relevant information available. Such a definition is linked to the rational expectation hypothesis, which states that agents do not make systematic errors in forecasting future prices. If such a condition holds, observed price dynamics can be informative about information held by traders. More precisely, information asymmetries among traders are needed for prices having an informational role. If this is the case traders can infer opportunities from prices and prices are affected by knowledge of trading opportunities. The foregoing dynamics can be reflected in prices as well as in bids and ask. In [Fri93a] the mean square deviation of actual price from rational expectation price is proposed as a metric for informational efficiency, in that it captures the tendency of price mechanisms to spread right information among traders. Another metrics is the bid and ask spread. If traders learn observing previous bids then informational efficiency would require a bid-ask spread shrinking to 0.

3.3.2 Optimality

Revenue Maximization An allocation is Pareto optimal if the welfare of at least one agent in the economy cannot be ameliorated without reducing the welfare of some other agent in the economy. A metric is the ratio of total payoffs actually earned by all traders - as stated in equation (3.2 - and theoretical payoffs that could be earned by all traders in equilibrium. In order to investigate the optimality of the mechanism (decentralized or centralized) we measure the distribution of profits across individuals which is [GS93]:

$$\sqrt{\frac{\sum_i (a_i - \pi_i)^2}{n}} \tag{3.4}$$

where a_i are actual payoffs and π_i are theoretical payoffs.

Correctness In a dynamic settings like ALN and a CDA or Call Market, the mechanisms involved have to be evaluated on the basis of the average number of times that a mechanism has approached the optimal allocation. This criterion is called "correctness" and according to [Par01b] can provide a more sensitive metrics of performance than efficiency.

3.3.3 Solution

Stability In a decentralized bilateral exchange setting, a measure of stability is the coefficient of convergence of prices to equilibrium values, that is defined as the ratio between standard deviation of actual prices and the predicted equilibrium price. The latter in turn is given by the intersection of demand and supply curves. Related to this metrics is the "time" of convergence i.e. time elapsed until the aforementioned ratio has shrunk to zero.

3.3.4 Feasibility

[Jac02a] argues that "A theme that comes out of the literature is that it is often impossible to find mechanisms compatible with individual incentives that simultaneously result in efficient decisions (maximizing total welfare), the voluntary participation of the individuals, and balanced transfers (taxes and subsidies that always net out across individuals)".

Incentive Feasibility In many mechanisms market agent interactions lead to strategic behaviour. If an agent knows that he can influence final price - and then his own payoff - he can report bids untruthfully [RSW94b]. Thus, mechanisms have to be evaluated with respect to the propensity of agents to report the truth values of goods in transactions. Actually, convergence to equilibrium can be attained also if traders use strategy of untruthful report their preferences, determining inefficiency (as a result some traders can stay out market also if they are willing to trade). [RSW94b] measure the foregoing feature, by evaluating the ratio of expected gains from trade across all agents in equilibrium and the expected gains from trade across all trades in equilibrium assuming that all agents behave as price takers. Incentive compatible mechanisms must be such that this ratio[4] to equals 1.

3.3.5 Tractability

Communication Costs [Hay45] works out the idea that in standard environments, the Walrasian mechanism is "informationally efficient," in that it realizes Pareto efficient allocations with the least amount of communication. The Walrasian mechanism involves only the announcement of prices, along with the allocation, which is much more economical than full revelation of agents' preferences. [NS01] examine problem communication establishing a lower bound of numbers of message needed to ensure efficient outcome. Thus a metric for this feature is the number of bids in transaction needed to approach the efficient outcome.

Computation Costs [Axt99] studies the complexity of exchange and establishes a lower bound of interactions between agents needed to ensure an efficient outcome. In this respect a metrics is the number of interactions which allows allocative efficiency. In general, this measure is exponential in the number of commodities and agents.

[4] A bidder's expected payoff is: $e = (expectedgain - expectedpayment) * probabilityofwinning$. [GO02]

Chapter 4

An Application for the Resource Market

4.1 Environmental Analysis

Having defined the environment for trading the services, the corresponding market for trading the resources has to be designed. The environment for designing a resource market can be compared to the environments found in the Grid, Peer-to-Peer, and Distributed Computing literature. These research areas - especially the research done in the area of Computational Grids - build the base for the following analysis.

4.1.1 Environment Definition

The participants in the resource market are those agents selling services in the service market as resource consumers (buyers) and a set of resource owners (sellers) offering computer resources.

Figure 4.1 outlines the resource market of the CATNETS scenario. Suppose a PDF service allocated a service on the service market and now requires computer resources for executing the service (e.g. a 700MHz CPU, 512MB RAM, and a 10GB hard disk). The agent with the instantiated PDF service acts as resource consumer (buyer) and the agents owning computer resource act as sellers.

In analogy to the service market, it is difficult to determine an approximated number of participants in a market for trading computer resource. At the moment, there is no existing commercial market place for trading such resources. However, there exist several test-beds and experimental evaluation studies which serve as clues. Table 4.1 lists exemplarily a selection of test-beds found in the Grid, Peer-to-Peer, and Distributed

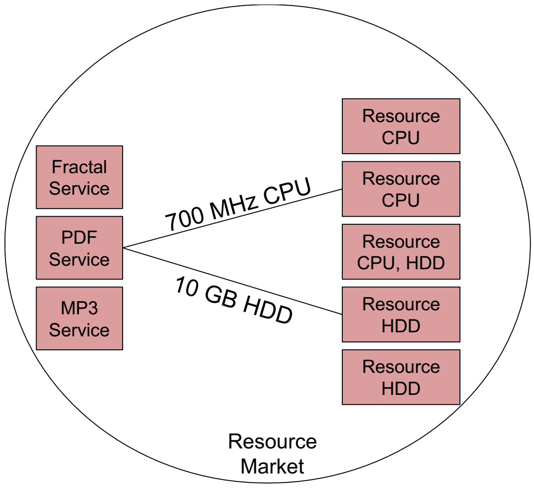

Figure 4.1: resource market

Computing literature.

Name	Resources	Users	Jobs/Day
Grid 2003 Project	700	102	1300
NorduGrid			1300
Conrell Center	512	512	800
San Diego	1152	468	250
PlanetLab	350	350	

Table 4.1: Grid and Computing test-beds statistics

Grid 2003 Project: The Grid2003 Project deploys a multi-virtual organization and application-driven Grid laboratory. The project sustained the production-level services for several months which were required by several different research institutes (e.g. for physics experiment, gravitational wave search) [FGG+04]. In peak times, the system handled 1300 simultaneous jobs.

NorduGrid: The NorduGrid[1] project provides a Grid infrastructure by adapting existing Grid middleware components (e.g. Globus Toolkit). In peak times, the NorduGrid handled about 1300 jobs per day.

Cornell Theory Center: The Cornell Theory Center is a high-performance computing center which supports scientific and engineering research projects. The center has

[1] See `http://www.nordugrid.org/` and `http://grid.uio.no/atlas/nglogger/nglogger_info/` for details (accessed: 10.02.2005).

over 500 CPUs and handled over 800 jobs per day[2] in peak times.

San Diego Supercomputer: The San Diego Supercomputer Center hosts a cluster with over 1100 processors. The average number of jobs executed per day was measured with 250 [EHY02]. The workload of this center served also as a benchmark for several scheduling architectures, e.g. the workload was used in [EHY02].

PlanetLab: PlanetLab is a distributed overlay platform to support the deployment and evaluation of planetary-scale network services [BBC+04]. It includes over 350 machines located in over 20 different countries.

Besides several test-beds, there exits simulation settings for evaluating the performance of different Grid components (e.g. scheduler, resource manager). For instance, [IF01] used in their setting 10,000 resources, 5,000 different users, and 100 jobs per day. [KDJN03] did a performance evaluation with 500 different users. [RS04] used 6,000 different resources for their simulation setting.

Similar to the analysis for the environment for the service market, the related systems and similation settings give a first impression of the number of participants and the number of jobs. Moreover, the number of participants in the resource market strongly depends on the number of participants (i.e. service agents) in the service market. Therefore, a detailed analysis has to be made in conjunction with the service market environment as well as with the technical analysts in the CATNETS project.

The transaction objects in the resource market are non standardized capacity-type objects. Capacity (e.g. processing power) not used in a time-period t is worthless at a later time-period $t + 1$. Furthermore, the same resources (e.g. CPUs) can differ in their capacities, e.g. a computing engine can be a high-performance processor capable of processing billions of FLOPS[3], as well as a low-performance processor of a mobile hand-held computer. Therefore, computer resources can be described by a set of possible attributes and its characteristics. For instance, a hard disk can be characterized by its quality attributes capacity (in Gigabyte (GB)), access time (in milliseconds (ms)), and data throughput (in bits per second (bits/s)).

The set and the number of possible different transaction objects in the resource market (e.g. resources like CPUs, hard-disks) are not specified in detail. Surveying the literature, a common set of possible resources can be identified. [CS01] introduce a base set of resource types that have to be supported within a resource management

[2]See `http://www.cs.huji.ac.il/labs/parallel/workload/` for details (accessed 05.01.2005)

[3]FLOPS stands for floating point instructions per second and is an approximate measure of a computer's processing speed.

mechanism: Computer resources (CPU, GPU), storage resources (memory, disk, tape), network resources (HPC switch, LAN, WAN), and miscellaneous resources (guest account, quota protocol). [BM02] and [KC02] abstract to the following five resources: processor, memory, storage, I/O, and network.

Following the above definitions of the environment and surveying the Grid literature, the simplified picture in Figure 4.2 can be adapted as a market place for the resource market [CFK04]. In essence, the market for trading computer resources (e.g. for trading in Computational Grid resources) is spanned around the resource owners as sellers (e.g. IBM or Sun with their computer centers), the resource consumers as buyers (e.g. scientists at universities, rendering or the biochemical firms) and some intermediaries (e.g. Condor, Gallop, Legion etc.). The intermediaries technically provide the resource management infrastructure for exploiting remote resources.

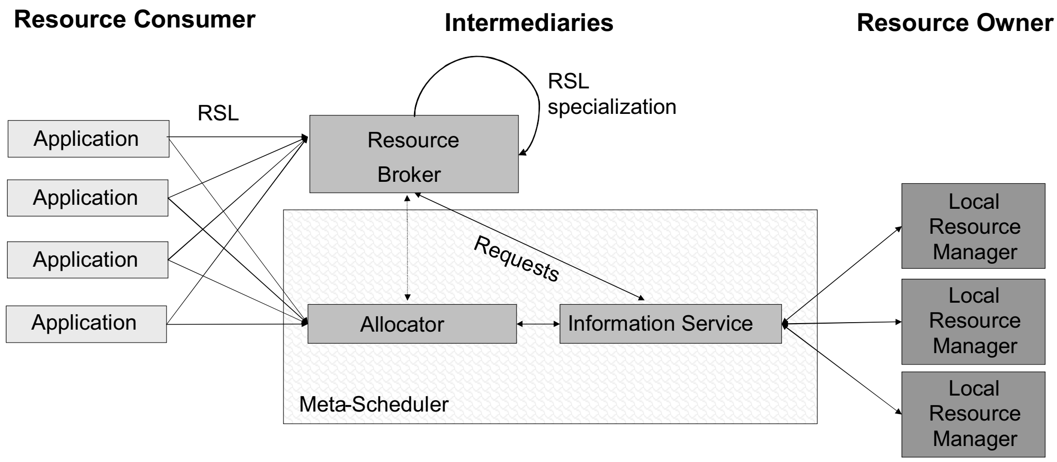

Figure 4.2: The market for Computational Grids

According to the resource management architecture proposed by [CFK04] the intermediary layer consists of three basic components:

Resource Broker: The resource broker components are responsible for resource discovery, selection, aggregation, and subsequently for the data and program transportation [CB02]. By transforming the resources to the consumers' requirements (which are specified for instance in the Resource Specification Language (RSL)) into a set of jobs that are self-reliantly scheduled on the appropriate resources (i.e. RSL specialization) and subsequently managed, the complexity of the resource market is concealed [VBW04]. For the market participants, the resource broker is apparently more of a black box. A Resource Broker may also include a matchmaking between multiattribute offers and request [Vei03].

Resource Information Manager: The resource information manager provides pervasive access to information about the current availability and capability of resources [CFK04].

Allocator: The allocator coordinates the allocation of resources at multiple sites. Obviously, the allocator and the information service assume the responsibility of (meta-) scheduling the jobs.

Based upon this view on the intermediary layer, the market mechanism for the resource market can be – in analogy to a market mechanism for the Computational Grid – sketched as follows: The transition from an intermediary layer to a mediated market mechanism is not too far. In essence, the scheduling performed by the resource broker can be shifted to the market mechanism. Instead of sending requests to the information service, the resource broker can translate the user requirements into bids. Those bids expressing demand and supply situation are subsequently cleared by the market mechanisms [SNW04].

4.1.2 Requirement Analysis

Despite the classical mechanism design requirements pertaining to the mechanism as described in section 3.1.2, the mechanism for the resource market must also account for the underlying environment. For the resource market the requirements stemming from the underlying environment are the following [SNW04]:

Simultaneous trading and immediate resource allocation: In analogy to the service market requirements, the market mechanism for the resource market has to support the simultaneous trading of multiple buyers and sellers, as well as an immediate resource allocation. Again, this can be realized by providing a continuous double auction mechanism.

Trading dependent resources: In the resource market, buyers usually demand a combination of computer resources as bundles to perform a task [SMT02]. This is based on the fact that computer resources (e.g. in the Computational Grid) are complementarities. Complementarities are resources with superadditive valuations ($v(A) + v(B) \leq v(AB)$), as the sum of the valuations for the single resources is less than the valuation for the whole bundle.

Suppose, for example, a buyer intending to render images requires hard disk, CPU, and main memory. If any component of the bundle, say the CPU, is not allocated to him, the remaining bundle has no value for him since the rendering cannot be processed without the CPU. In order to avoid the exposure risk (i.e.

receiving only one part of the bundle[4] without the other), the mechanism must allow for bids on bundles. Likewise, the seller can also express bids on bundles.

Support for multiattribute resources: Resources in the resource market are typically not completely standardized, as similar resources can differ in their quality. A hard disk can be characterized by its quality attributes capacity (in Gigabyte (GB)), access time (in milliseconds (ms)), and data throughput (in bits per second (bits/s)). For example, a rendering job requiring a minimum amount of GB 250 GB can be conducted by a 500 GB hard disk; however, it can not be executed by a 100 GB hard disk. As such, minimum quality requirements must be met, while similar resources of superior quality work as well. Hence, minimum quality requirements must be met, while similar resources of superior quality work as well[5].

Furthermore, buyers usually require resources only for a certain time span. Having conducted the computation, there is typically no further use for the resources. The exact timing of the computation is not always that important for the buyer. For instance, the buyer may be indifferent whether the job is performed at 10 a.m. or at 11 a.m., as long as the job is finished at a certain time, e.g. 3 p.m. Therefore, the market mechanism must allow (i) an advanced reservation (AR), which is a process of requesting resources for use at a specific time in the future and (ii) placing bids on time ranges [SIT00].

Partial execution on the sellers side: A seller of resources (e.g. an owner of a scaled server farm) is also willing to sell just a part of the offered resources. Thus, the possibility of a partial executing on the sellers' side must be given.

Co-Allocation: Scientific applications (e.g. genetic disease research) require more resources than any single seller could provide. As such, an allocation from multiple sites (or in short co-allocation[6]) is indispensable. In this case a set of resources is made available simultaneously by coordinating on-demand agreements across the required resources [CFK04].

4.2 Meeting the Requirements

A number of mechanisms have been proposed that attempt to solve the resource allocation problem in such a resource market or related architectures. Most of these mechanisms are central in nature in a way that the allocation problem is solved by a central

[4]A bundle denotes a combination of resources.

[5]The network congestion can also be included by the use of attributes. Therefore, network congestion metrics have to be defined and evaluated in future work, e.g. [Wol98], [Loe04]

[6]In this context, the term co-allocation is equivalent to the term co-scheduling.

entity using global optimization algorithms without the employment of prices. This central entity requires detailed information about the demand and supply situation in order to be effective. As information is dispersed among the buyers and sellers, central allocation algorithms may not enfold their full power, because this information requirement is not even closely met. Market-based approaches incorporate incentives for truthful information revelation by implementing prices.

Surveying the literature, various mechanisms for allocating computer resources adapt classical auction mechanism for standardized products as described in section 3.2. Furthermore, there exists a number of multiattribute, bundle, or combinatorial mechanisms, which can be used to trade dependent and non-standardized products. Both types of mechanisms and their applicability for the resource market will be discussed in the following.

4.2.1 Classic Auction Types

There exists a number of economic resource allocation systems which adapt classical auction mechanisms (e.g. English auctions) for Peer-to-Peer networks and Grid Computing infrastructures[7].

SPAWN: The SPAWN system provides a market mechanism for trading CPU times in a network of workstations [WHH+92]. SPAWN treats computer resources a standardized commodities and implements a Vickrey auction.

POPCORN: POPCORN [8] provides an infrastructure for global distributed computation [RN98, NLRC88]. POPCORN mainly consists of three entities: (i) A parallel programm which requires CPU time (buyer), (ii) a CPU seller, and (iii) a market which serves as a meeting place and matchmaker for the buyers and sellers. Buyers of CPU time can bid for one single commodity, which can be traded executing a Vickrey auction repeatedly.

OCEAN: OCEAN[9] (Open Computation Exchange and Arbitration Network) is a market-based infrastructure for high-performance computation, such as Cluster and Grid computing environments [ACD+01, PHP+03]. The major components of the OCEAN's market infrastructure are user components, computational resources, and the underlying market mechanism (e.g. the OCEAN Auction Component).

[7] A detailed overview of classical auctions in resource allocation systems is given among others in [KBM02].

[8] See `http://www.cs.huji.ac.il/labs/popcorn/(accessed:20.12.2004)` for details

[9] See `http://www.cise.ufl.edu/research/ocean/,(accessed:20.12.2004)` for details

In the OCEAN framework, each user (i.e. resource provider or consumer) is represented by a local OCEAN node. The OCEAN node implements the core components of the system, for instance a Trader Component, an Auction Component, or a Security Component. The implemented OCEAN auctions occur in a distributed Peer-to-Peer manner. The auction mechanism implemented in the OCEAN framework can be interpreted as a distributed sealed-bid continuous double-auction [ACD+01]. A trade is proposed to the highest bidder and the lowest seller. Afterwards, the trading partner can renegotiate their service level agreements.

G-Commerce: G-Commerce provides a framework for trading computer resources (CPU and hard disk) in commodity markets and Vickrey auctions [WPBB01a, WPBB01b, WBPB03].

Stanford Peers: The Stanford Peers model is a Peer-to-Peer system which implements auctions within a cooperative bartering model in a cooperative sharing environment [CGM02]. It simulates storage trading for content replication and archiving. It demonstrates distributed resource trading policies based on auctions by simulation.

Reviewing the requirements on the mechanism, it becomes evident that the previous described mechanisms fail to satisfy the requirements of a resource market. Especially the negligence of time attributes for bundles and quality constraints for single resources diminish the use of the proposed market mechanisms.

4.2.2 Combinatorial Auctions and Exchanges

Combinatorial Auctions and Exchanges comprise dependencies between resources and have attained a lot of attention in the last few years [BN03, Gra04, JV04, Par01a, SSGL02]. Combinatorial Auctions and Exchanges allow to submit logical concatenated bids like AND bids, OR bids, and XOR bids. The use of these auction types is crucial if resources are complementarities or substitutes. Due to their capability of guaranteeing ”all-or-nothing”[10] bids, their application for trading computer resources is deemed promising.

At the moment, there exists different combinatorial and bundle mechanisms which can be classified as sketched in figure 4.3:

Generalized Vickrey Auction: The Generalized Vickrey Auction (GVA) is a Vickrey-Clarke-Groves mechanisms (cf. Vickrey Auction) and implements a single shot

[10]For instance, a bid on a bundle {CPU, RAM, HDD} will only be executed if all components of the bundle can be allocated.

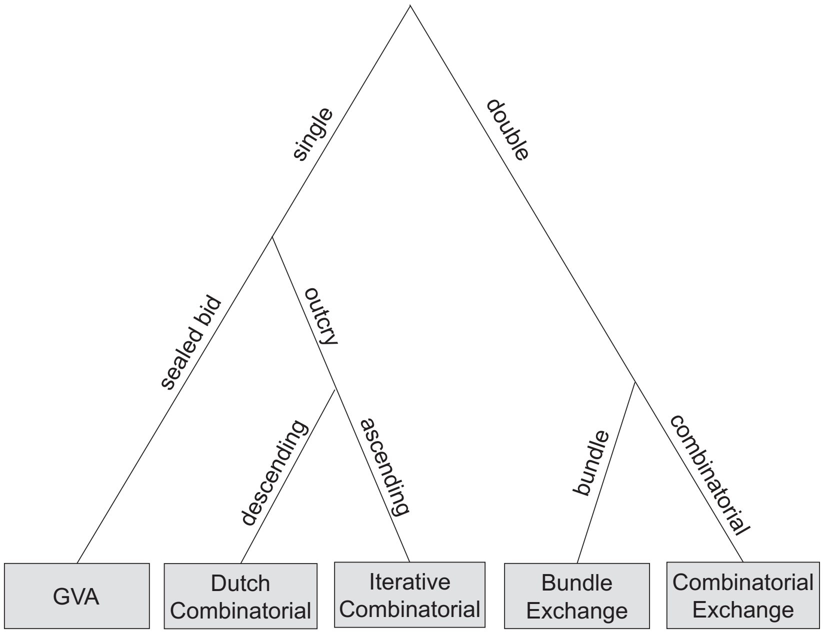

Figure 4.3: A classification of bundle and combinatorial auction types

and sealed bid second-price approach for combinatorial auctions. As the single item Vickrey auction, the GVA is strategy-proof, i.e. truthful bidding is a dominant strategy [GL77].
However, the mechanism is single sided and, thus, does not generate competition on both sides. Furthermore, the GVA treats the traded resources as standardized commodities and does not support quality characteristics. Besides, the computational tractability of the GVA is very high [Par01a].

Dutch Combinatorial Auctions: In a Dutch Combinatorial Auction, the auctioneer calls out a price for each bundle and lowers this price incrementally as long as no bidder is willing to accept it. An implementation of a Dutch Combinatorial Auction can be found in [BN03]. However, the auction is single-sided and does not support quality characteristics of resources.

Iterative Combinatorial Auctions: Iterative Combinatorial Auctions adapt English Auctions for the combinatorial case [AM02, Par99]. Iterative combinatorial auctions are the most prominent combinatorial mechanisms - however; their applicability on the resource market is not possible due to the neglect of multiattribute bids and multiple sellers and buyers.

Bundle Exchanges: Bundle exchanges are double auctions with support AND concatenated bids on bundles. In contrast to combinatorial auctions and exchanges,

these auctions types neglect the use of XOR bids. An implementation of a bundle exchange can be found in [GVW04].
Classical bundle mechanisms neglect the support of quality characteristics which disables their direct applicability for the resource market. However, with respect to the requirements for the resource market, a bundle exchange may be a base mechanism for future research.

Combinatorial Exchanges: Combinatorial exchanges are double auctions that support combinatorial orders. [PKE01] introduce the first combinatorial exchange as a single-shot sealed bid auction. [BN03, Bis04] propose an iterative combinatorial exchange based on a primal/dual programming formulation of the allocation problem. However, both approaches neither account for time nor for quality constraints and are thus not directly applicable for the resource market. However, the ability of trading dependent resources simultaneously qualifies a Combinatorial Exchange to meet most of the requirements on the Resource Market.

Reviewing the requirements on the mechanism, it becomes obvious that the previous described mechanisms fail to satisfy these requirements. Especially the negligence of time attributes for bundles and quality constraints for single resources diminish the use of the proposed market mechanisms. The introduction of time attributes redefines the allocation problem to a type of scheduling problem.

To account for time attributes, Wellman et al. model single-sided auction protocols for the allocation and scheduling of resources under consideration of different time constraints [WWWMM01]. Conen goes one step further by designing a combinatorial bidding procedure for job scheduling including different running, starting, and ending times of jobs on a processing machine [Con02]. Both approaches are, however, single-sided and thus do not create competition on both sides.

4.2.3 Summary

Reviewing the requirements for a resource market, it becomes obviously that none of the presented classical auction types and combinatorial mechanisms is directly applicable for the resource market. None of the mechanisms comprise at the same time simultaneous bids of multiple sellers and buyers, time and quality constraints as well as combinatorial bids.

The future work in the CATNETS project intends to tailor a mechanism for allocating computer resources by converting the aforementioned approaches into a combinatorial or bundle exchange that additionally incorporates time and quality constraints.

4.3 Testing

The metrics described in 3.3 are quite general and can be usefully exploited also for measuring the performance of the resources market. Thus, the reader is referred to that section for a list and an explanation of market performance metrics. In what follows, we simply discuss some problems related to the existence of a competitive equilibrium, that arise due to the very specific structure of the market for resources. In particular, the design of such a market must allow for the trading of dependent as well as for multi attribute resources. Put differently, resources in any bundle to be traded are *complementary goods* for buyers (e.g. a positive amount hard-disk storage capacity with a zero cpu is worthless for a buyer). Moreover, there are minimum quality requirements on them (e.g. because a service task requires an hard disk capacity of 100Gb *at least*).

We can formalize the foregoing requirements by saying that buyers in resources' market have the following *Leontief* utility function:

$$U_i(x_1, x_2, x_3, ..., x_n) = \min \{b_{i1}x_1, b_{i2}x_2, b_{i3}x_3, ..., b_{in}x_n\} \tag{4.1}$$

where $(x_1, x_2, x_3, ..., x_n)$ are resources included in a bundle of dimension n. Equation (4.1) simply states that *any* resource x_k in a bundle must be present in the non-negative proportion b_{ik} in order to give utility to buyer i. If those proportions are not respected a bundle is worthless to a buyer.

If buyers' preferences are like those specified in (4.1) problems may arise with the existence of a competitive equilibrium, and thereby with the use of all metrics based on it[11]. Indeed, Leontief preferences may lead to an individual demand function for resources' bundles like the one displayed in Figure 4.4

In the Figure, the agent i is willing to buy the bundle X_i at any price less or equal P_i whereas for any price above that threshold his demand is zero. This situation may occur if complementarities among resources are strongly enough to induce the agent to not substitute bundle X_i for another one with different resources proportions as the price goes down. Similarly, this *aversion for substitution* can induce the agent to prefer to not consume anything if the price for the preferred bundle is too high.

The situation represented in Figure 4.4 can be deleterious for the existence of a competitive equilibrium. Indeed, suppose every buyer in the resource market behaves like buyer i. Then, the situation displayed in Figure 4.5 may occur. As the figure shows, aggregate supply of bundles - i.e. the sum of individual supply functions - does not cross aggregate demand, that in this case is equal to n-replicas of the individual demand function of buyer i. Accordingly, the competitive equilibrium does not exist.

The foregoing problems can nonetheless be solved if buyers' individual characteristics - i.e. preferences and/or budget constraints - are heterogeneous enough. If that is the case the situation can be like the one depicted in Figure 4.3, where only four individuals

[11] For a thorough analysis of these problems see [CWG95].

are assumed to be present. In that picture, all individuals have the same preferences structure as agent i in Figure 4.4. Nevertheless, they are heterogenous with respect to the preferred bundle as well as with respect to the maximum price each one is willing to pay. Aggregating over this kind of individuals leads to an aggregate demand which is positive for different price-bundle couples (and not just for one as in Figure 4.5). Ideally, the more buyers are heterogeneous the more aggregate demand will tend to be a continuous function, thereby assuring the existence of a competitive equilibrium.

Summarizing, the structure of the resources' market - in which agents have Leontief Preferences over resources - is in general not an obstacle for the use of metrics described in section 3.3, granted that buyers' characteristics are heterogeneous enough. In this respect, measures of characteristics dispersion - as for example the variance of minimum quality requirements - can be complementary metrics to measure the performance of that market.

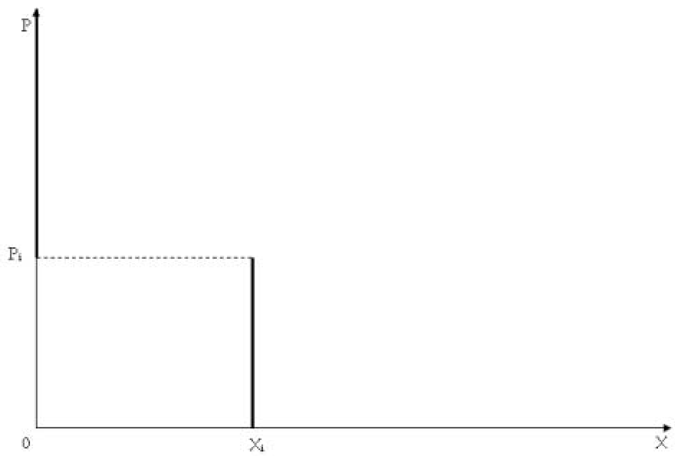

Figure 4.4: Example of Individual Demand when preferences are Leontief.

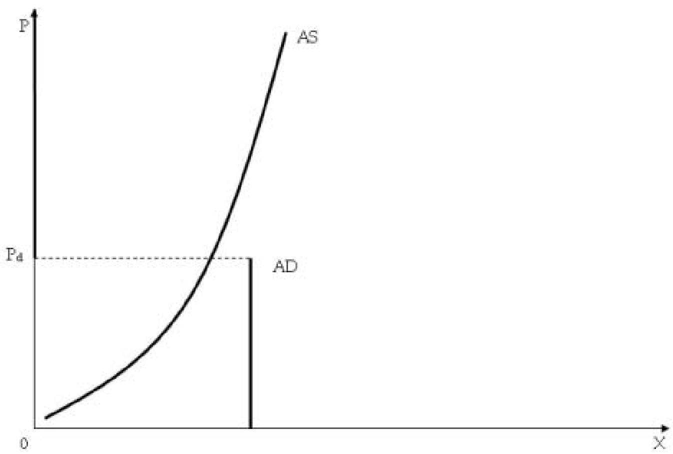

Figure 4.5: Example in which a competitive equilibrium may not exist if preferences are Leontief.

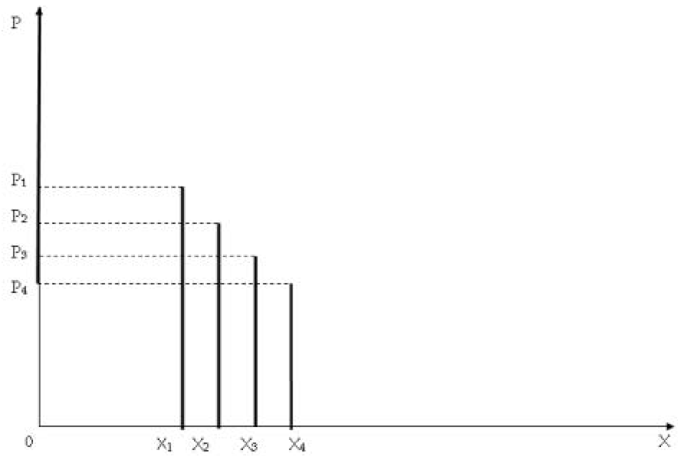

Figure 4.6: Example in which a competitive equilibrium can be restored if preferences of buyers are heterogeneous.

Chapter 5

Conclusion

The key issue in this work is the structured identification and definition of market mechanisms for the application layer networks. In completing deliverable 1 in work package 1 with this document the aim is to generate input for the following parts of the CATNETS project:

- Carrying out deliverable 2 in work package 1 that aims on the composition of the market mechanisms for simulation issues. Within this second part of the work package, specification and concepts for simulation and measurement of ALN scenarios are analyzed.
- Providing input for deliverable 2 in work package 3 that aims on the specification and implementation of services within the ALN middleware developed in CATNETS.
- Providing input for deliverable 2 in work package 4 that aims on the definition of the agent architecture for the simulator that will be implemented in CATNETS.

Hence, we provide a summary on the results of this report in Section 5.1 and an outlook on the next steps to be taken in Section 5.2.

5.1 Review of this Work

This work comprises several parts. Firstly, a centralized market perspective for application layer networks is drawn. Here, two markets are distinguished: a service and a resource market. The service market is used by customer agents that aim on purchasing a certain service in order to fulfil a tasks. The second market – the resource market – is the market on which the service instances trade the resources they need in order to operate the offered services (compare Chapter 1).

The approach which is chosen in both cases in order to design the market characteristics is the so called Market Engineering appraoch (compare Chapter 2). This approach is based on a process including several steps to define markets in a structured way. The process is accompanied by a set of well chosen methods that are consequently applied in the definition phase of the market mechanisms for ALNs in the CATNETS project. Next to defining and implementing as well as testing market mechanisms, this approach also employs means of experimental analysis and computerized simulation for the evaluation of market mechanisms. An important source is the work done in the domain of meachanism design and economic engineering in past years.

In the following, the Market Engineering approach is employed for the service and the resource market (compare Chapter 3 and Chapter 4). As a result of the analysis of the requirements for the service market, a specific type of a continuous double auction will be envisaged to implement CATNETS project. The result of the analyis of the requirements for the resource market provided to suggest a kind of bundle or combinatorial market mechanism to be implemented.

Within this report the focus lies on several issues: One central issue is the introduction of the Market Engineering approach. Consequently, this approach is applied to both markets in the CATNETS project. Finally, the classes of mechanisms which are aimed to be implemented for these markes have been identified.

5.2 Outlook on Next Steps

The next steps in work package 1 comprise several issues. In order to proceed with the detailed definition of market mechanisms for the service and the resource market, in depth knowledge in the field of mechanism design has to be gathered. Therefore, further literatur review is essentially. Within the ongoing work in this package, the market mechanisms will be defined formally and described in an implementation ready manner.

The output of this process will feed the definition and the impelementation of the simulator. Therefore, there will be taken a look on the economic metrics that allow the measurement of the quality of the chosen market mechanisms. Measuring efficiency is essential in order to vary market mechanisms further and to converge to high quality mechanisms.

The output of this report will also feed into the definition and implementation process of the service implementation in the CATNETS middleware. Here, more specific defintions of the market mechanisms are focussed and a close cooperation with the middleware work package is panned.

The next steps in this work package are the formal defintion and the in depth elaboration of economic market mechanisms that meet the outlined desiderata.

Bibliography

[ACD+01] Nachiket Acharya, Chaitanya Chokkaredd, Pinkeshkumar Desai, Rajesh Devrajan, Michael P. Frank, SriramKumar Nallan Chakravarthul, Murali Nagendranath, Pradeep Padala, Heeyong Park, Gargi Sur, Mark Tobias, Chi-Kin Wong, and BongJu Yu. The open computation exchange & auctioning network (ocean). Technical report, Department of Computer & Information Science & Engineering, University of Florida, 2001.

[AG05] Deutsche Boerse AG. Europe's premier listing platform. Fact sheet, Deutsche Boerse AG, 2005.

[AH00] Eytan Adar and Bernardo A. Huberman. Free riding on gnutella. *First Monday*, 5(10), 2000.

[AHL+96] W. Brian Arthur, John H. Holland, Blake LeBaron, Richard Palmer, and Paul Tayler. Asset pricing under endogenous expectations in an artificial stock market. Technical report, Santa Fe Institute, December 1996.

[AM02] Lawrence M. Ausubel and Paul Milgrom. Ascending auctions with package bidding. *Frontiers of Theoretical Economics*, 1(1):1–43, 2002.

[Axe84] R. Axelrod. *The Evolution of Cooperation*. Basic, New York, 1984.

[Axt99] Rob Axtell. The complexity of exchange. Technical report, Society for Computational Economics, March 1999.

[BAGS02] Rajkumar Buyya, David Abramson, Jonathan Giddy, and Heinz Stockinger. Economic models for resource management and scheduling in grid computing. *The Journal of Concurrency and Computation: Practice and Experience*, 14(13-15):1507–1542, 2002.

[BAV04] Rajkumar Buyya, David Abramson, and Srikumar Venugopal. The grid economy. In *Proceedings of the IEEE, Grid Computing*, 2004.

[BBC+04] Andy Bavier, Mic Bowman, Brent Chun, David Culler, Scott Karlin, Steve Muir, Larry Peterson, Timothy Roscoe, Tammo Spalink, and Mike Wawrzoniak. Operating systems support for planetary-scale network services. In *Proceedings of the First Symposium on Networked Systems and Design and Implementation, San Fransisco*, 2004.

[BCZ92] T.A. Byrd, K.L. Cossick, and R.W. Zmud. A synthesis of research on requirements analysis and knowledge acquisition techniques. *MIS Quarterly*, 16(1):117–138, 1992.

[Bis04] Shantanu Biswas. *Design of Iterative Mechanisms for Combinatorial Auctions and Exchanges*. PhD thesis, Computer Science & Automation, Indian Institute of Science, Bangalore 560 012 (India), 2004.

[BM02] Rajkumar Buyya and Manzur Murshed. Gridsim: A toolkit for the modeling and simulation of distributed resource management and scheduling for grid computing. *The Journal of Concurrency and Computation: Practice and Experience (CCPE)*, 14(13-15), 2002.

[BN03] Shantanu Biswas and Y. Narahari. Iterative dutch combinatorial auctions. *To appear in: Special issue of Annals of Mathematics and Artificial Intelligence on the Foundations of Electronic Commerce*, 2003.

[BO00] Sushil Bikhchandani and Josep M. Ostroy. The package assignment model. *Journal of Economic Theory*, 107:377–406, 2000.

[Bud03] Miroslav Budimir. *Informationsverarbeitung auf Wertpapiermärkten: Neue Marktformen, Design und experimentelle Evaluation*. PhD thesis, Lehrstuhl für Informationsbetriebswirtschaftslehre, Universität Karlsruhe, 2003.

[CAB+94] Derek Coleman, Patrick Arnold, Stephanie Bodoff, Chris Dollin, Helena Gilchrist, Fiona Hayes, and Paul Jeremaes. *Object-Oriented Development - The FUSION Method*. Prentice Hall, Englewood Cliffs, NJ, 1994.

[CB02] Madhu Chetty and Rajkumar Buyya. Weaving computational grids: How analogous are they with electrical grids? *Computing - Science and Engineering*, 4(4):61–71, 2002.

[CFK04] Karl Czajkowski, Ian Foster, and Carl Kesselman. *Resource and Service Management*, volume 2, chapter 18, pages 259–283. Elsevier, 2004. In: The Grid 2 - Blueprint for a New Computing Infrastructure.

[CGM02] Brian F. Cooper and Hector Garcia-Molina. Bidding for storage space in a peer-to-peer data preservation system. In *Proceedings of the 22 nd International Conference on Distributed Computing Systems (ICDCS'02)*, page 372. IEEE Computer Society, 2002.

[Cha98] Edmund Chattoe. Just how (un)realistic are evolutionary algorithms as representations of social processes? *Journal of Artificial Societies and Social Simulation*, 1(3), 1998.

[COBW00] Henri Casanova, Graziano Obertelli, Francine Berman, and Rich Wolski. The apples parameter sweep template: user-level middleware for the grid. In *Proceedings of the 2000 ACM/IEEE conference on Supercomputing*. IEEE Computer Society, 2000.

[Con02] Wolfram Conen. Economically coordinated job shop scheduling and decision point bidding - an example for economic coordination in manufacturing and logistics. In *Proceedings of the 16th Workshop on Planen, Scheduling und Konfigurieren, Entwerfen*, 2002.

[Cra03] Peter Cramton. Electricity market design: The good, the bad, and the ugly. *Proceedings of the Hawaii International Conference on System Sciences*, 2003.

[CS83] D. Chatterjee and W. Samuelson. Bargaining under incomplete information. *Operations Research*, 31(5):835–851, 1983.

[CS01] K. Czajkowski and V. Sander. Grid resource management protocol: Requirements. Ggf scheduling work group proposal, University of Southern California, John von Neumann Institut for Computing, 2001.

[CWG95] Andreu Mas Colell, Mark D. Whinston, and John R. Green. *Microeconomic Theory*. Oxford University Press, 1995.

[CY91a] P. Coad and E. Yourdon. *Object-Oriented Analysis*. Prentice-Hall, Englewood Cliffs, N.J, 1991.

[CY91b] P. Coad and E. Yourdon. *Object-Oriented Design*. Prentice-Hall, Englewood Cliffs, N.J, 1991.

[DHDJ99] Alan R Dennis, Glenda S Hayes, and Robert M. Daniels Jr. Business process modeling with group support systems. *Journal of Management Information Systems*, 15(4):115–142, 1999.

[Dux95] D. Duxbury. Experimental asset markets within finance. *Journal of Economic Surveys*, 9(4):331–371, 1995.

[EHY02] Carsten Ernemann, Volker Hamscher, and Ramin Yahyapour. Economic scheduling in grid computing. In *Proceedings of the 8th International Workshop on Job Scheduling Strategies for Parallel Processing*, pages 128–152. Springer-Verlag, 2002.

[EL93] D. Easley and J. Ledyard. Theories of price formation and exchange in double oral auctions. In Daniel Friedman and John Rust, editors, *The Double Auction Market: Institutions, Theory, and Evidence*. Perseus Publishing, Cambridge, MA, 1993.

[ERA+03] T. Eymann, M. Reinicke, O. Ardaiz, P. Artigas, L. Díaz de Cerio, F. Freitag, R. Messeguer, and D. Royo L. Navarro. Decentralized vs. centralized economic coordination of resource allocation in grids. In *Proceedings of the 1st European Across Grids Conference February, 13th-14th, 2003*. Universidad de Santiago de Compostela, Spain, 2003.

[FA03] Geraldine Fennell and G. M. Allenby. Market definition, segmentation, and positioning: An integrated approach. Working paper, 2003.

[FFRS02] Thomas Friese, Bernd Freisleben, Steffen Rusitschka, and Alan Southall. A framework for resource management in peer-to-peer networks. In Mehmet Aksit, Mira Mezini, and Rainer Unland, editors, *NetObjectDays*, volume 2591 of *Lecture Notes in Computer Science*, pages 4–21. Springer, 2002.

[FGG+04] I. Foster, J. Gieraltowski, S. Gose, N. Maltsev, E. May, A. Rodriguez, D. Sulakhe, and the other Grid2003 Project Members. The grid2003 production grid:principles and practice. In *Proceedings of the Thirteenth IEEE International Symposium on High-Performance Distributed Computing (HPDC13)*, 2004.

[FK04] Ian Foster and Carl Kesselman. *The Grid 2 - Blueprint for a New Computing Infrastructure*, volume 2. Elsevier, 2004.

[FL99] D Fudenberg and David K. Levine. *The Theory of Learning in Games*. MIT Press, Cambridge, MA, 1999.

[Fri91] D. Friedman. *The Double Auction Market - Institutions, Theories, and Evidence*, chapter The Double Auction Market Institution: A Survey, pages 3–26. Cambridge MA, Perseus Publishing, 1991.

[FT00] Drew Fudenberg and Jean Tirole. *Game Theory*. The MIT Press, Cambridge, MA and London, England, 7th edition edition, 2000.

[FTF+01] J. Frey, T. Tannenbaum, I. Foster, M. Livny, and S. Tuecke. Condor-G: A Computation Management Agent for Multiinstitutional Grids. In *International Symposium on High Performance Distributed Computing*, pages 55–67, San Francisco, CA, 2001.

[GL77] Jerry Green and Jean-Jacques Laffont. Characterization of satisfactory mechanisms for the revelation of preferences for public goods. *Econometrica*, 45(2):427–438., 1977.

[GO02] Jacob K. Goeree and Theo Offerman. Efficiency in auctions with private and common values: An experimental study. *American Economic Review*, 92(3):625–643, 2002. available at http://ideas.repec.org/a/aea/aecrev/v92y2002i3p625-643.html.

[Gra04] Peter Gradwell. Distributed combinatorial resource scheduling. Discussion paper, Department of Computer Science, University of Bath, 2004.

[GS93] S. Gode and K. Sunders. Lower bounds for efficiency or surplus extraction in double auctions. In David Friedman and John Rust, editors, *The Double Action Market: Institution, Theories , and evidence*, volume 15 of *Santa Fe Institute Series in The Science of Complexity*, pages 199–219. SFI, Addison Wedsley, 1993.

[GVW04] M. Grunenberg, D. Veit, and C. Weinhardt. Elektronische finanzmärkte und bundle trading. In *Proceedings der 66. wissenschaftlichen Jahrestagung des Verbandes der Hochschullehrer für Betriebswirtschaft,Graz*, pages 310–313, 2004.

[Hay45] F.A. Hayek. The use of knowledge in society. *American Economic Review*, 35:519–30, 1945.

[HB95] Karen Holtzblatt and Hugh R. Beyer. Requirements gathering: the human factor. *Communications of the ACM*, 38(5):31 – 32, 1995.

[Hol04] Carsten Holtmann. *Organisation von Maerkten - Market Engineering fuer den Wertpapierhandel*. PhD thesis, Economics and Business Engineering. Karlsruhe, Germany, University of Karlsruhe (TH), 2004.

[HR86] J. D. Hlavacek and N. M. Reddy. Identifying and qualifying industrial market segments. *European Journal of Marketing*, 20(2):8–21, 1986.

[HRRM00] Cengiz Haksever, Barry Render, Roberta S. Russell, and Robert G. Murdick. *Service Management and Operations*. Prentice-Hall, Upper Saddle River, NJ, 2000.

[Hur72] Leonid Hurwicz. On informationally decentralized systems. In C.B. McGuire and R. Radner, editors, *Decision and Organization*. North-Holland Publishing Company, Amsterdam, NL, 1972.

[Hur73] L. Hurwicz. The design of mechanisms for resource allocation. *American Economic Review*, 69(2):395–402, 1973.

[IF01] Adriana Iamnitchi and Ian T. Foster. On fully decentralized resource discovery in grid environments. In *Proceedings of the Second International Workshop on Grid Computing*, pages 51–62. Springer-Verlag, 2001.

[Jac02a] Matthew O. Jackson. *Encyclopedia of Life Support Systems (EOLSS)*, chapter Mechanism theory. Eolss Publishers, Oxford UK, 2002.

[Jac02b] Matthew O. Jackson. Mechanism theory. In *Encyclopedia of Life Support Systems (EOLSS)*, Oxford ,UK, 2002. Eolss Publishers.

[JS95] Thomas O. Jones and Earl W. Sasser. Why satisfied customers defect. *Harvard Business Review*, 73(6):88–94, 1995.

[JV04] Rahul Jain and Pravin Varaiya. Combinatorial exchange mechanisms for efficient bandwidth allocation. *Communications in Information and Systems*, 3(4):305–324, 2004.

[KBM02] Klaus Krauter, Rajkumar Buyya, and Muthucumaru Maheswaran. A taxonomy and survey of grid resource management systems. *International Journal of Software: Practice and Experience*, 32(2):135–164, 2002.

[KC02] Chris Kenyon and Giorgos Cheliotis. Architecture requirements for commercializing grid resources. In *Proceedings of the 11 th IEEE International Symposium on High Performance Distributed Computing HPDC-11 20002 (HPDC'02)*, page 215. IEEE Computer Society, 2002.

[KDJN03] H. Lim Choi Keung, Justin R.D. Dyson, Stephen A. Jarvis, and Graham. R Nudd. Performance evaluation of a grid resource monitoring and discovery service. *IEEE Proceedings - Software*, 150(4):243–251, 2003.

[KP03] Jayant Kalagnanam and David C. Parkes. *Auctions, Bidding and Exchange Design*. Kluwer Academic Publishing, 2003.

[Kri02] Vijay Krishna. *Auction theory*. Academic Press, San Diego, 2002.

[KSTT04] Ramayya Krishnan, Michael D. Smith, Zhulei Tang, and Rahul Telang. The impact of free-riding on peer-to-peer networks. In *Proceedings of the 37th Hawaii International Conference on System Sciences*, 2004.

[Led93] J. Ledyard. The design of coordination mechanisms and organizational computing. *Journal of Organizational Computing*, 3(1):121–134, 1993.

[LK98] Edna Tusak Loehman and D. Marc Kilgour. Introduction: Social design for environmental and resource management. In Edna Tusak Loehman and D. Marc Kilgour, editors, *Designing Institutions for Environmental and Resource Management*, New Horizons in Environmental Economics, pages 1–25. Edward Elgar, Cheltenham, UK, 1998.

[Loe04] C. Loeser. Demand forecasting in p2p and grid systems. In *Proceedings of the 10th International Conference on Cybernetics and Information Technologies, Systems and Applications (CITSA-2004)*, Orlando, Florida, 2004.

[MCM75] L. Daniel Maxim, Daniel E. Cullen, and John G. Mardo. Optimal acceptance test plans with grouping. *Technometrics*, 17(3):315–320, 1975.

[Mil89] Paul Milgrom. Auctions and bidding: A primer. *Journal of Ecnomic Perspectives*, 108(2):245–272, 1989.

[Mil04] Paul Milgrom. *Putting Auction Theory to Work*. Cambridge University Press, 2004.

[MM87a] Preston R. McAfee and John McMillan. Auctions and bidding. *Journal of Economic Literature*, 25(2):699–728, 1987.

[MM87b] R.P. McAfee and J. McMillan. Auctions and bidding. *Journal of Economic Literatur*, 25(2):699–738, 1987.

[MMORW04] Jeffrey MacKie-Mason, A. Osepayshvili, D.M. Reeves, and M. P. Wellman. Price prediction strategies for market-based scheduling. In *International Conference on Automated Planning and Scheduling,*, page forthcoming, 2004.

[MMT00] Richard D. McKelvey, Andrew McLennan, and Theodore Turocy. Gambit graphics user interface - an interactive extensive form game program. Documentation, 2000.

[MRS93] K.A. McCabe, S. Rassenti, and V. Smith. Designing a uniform-price double auction: An experimental evaluation. In Daniel Friedman and J. Rust, editors, *The Double Auction Market: Institutions, Theory, and Evidence*, pages 307–332. Perseus Publishing, Cambridge, MA, 1993.

[MS83] Robert B. Myerson and Mark A. Satterthwaite. Efficient mechanisms for bilateral trading. *Journal of Economic Theory*, 28:265–281, 1983.

[MW02] Merriam-Webster. Merriam-webster online. http://www.m-w.com/, 2002.

[MY01] H. Mizuta and Y. Yamagata. Agent-based simulation and greenhouse gas emissions trading. In B.A. Peters, J.S. Smith, D.J. Medeiros, and M.W. Rohrer, editors, *Proceedings of the 2001 Winter Simulation Conference*, pages 535–540, 2001.

[Neu04a] Dirk Neumann. Knowledge-based auction design. Working paper, 2004.

[Neu04b] Dirk Neumann. *Market Engineering - A Structured Design Process for Electronic Markets*. PhD thesis, Economics and Business Engineering. Karlsruhe, Germany, University of Karlsruhe (TH), 2004.

[NLRC88] N. Nisan, S. London, O. Regev, and N. Camiel. Globally distributed computation over the internet - the popcorn project. In *Proceedings of the 18th International Conference on Distributed Computing Systems, 26 - 29 May, 1998, Amsterdam, The Netherlands. IEEE Computer Society*, 1988.

[NS01] Noam Nisan and Ilya Segal. The communication complexity of efficient allocation problems. Unpublished Paper, November 2001.

[OF99] James Odell and Martin Fowler. *Advanced object-oriented analysis and design using UML.* Cambridge Univ. Press, Cambridge, 1999.

[Ore01] Shmuel S. Oren. Market based risk mitigation: Risk management vs. risk avoidance. In *White House OSTP/NSF Workshop on Critical Infrastructure Interdependencies*, Washington DC, 2001.

[Par99] David Christopher Parkes. ibundle: An efficient ascending price bundle auction. In *Proceedings of the 1st ACM conference on Electronic commerce (EC99)*, pages 148–157. ACM Press, 1999.

[Par01a] David Christopher Parkes. *Iterative Combinatorial Auctions: Achieving Economic and Computational Efficiency*. PhD thesis, Department of Computer and Information Science, University of Pennsylvania, 2001.

[Par01b] David Christopher Parkes. *Iterative Combinatorial Auctions: Achieving Economic and Computational Efficiency*. PhD thesis, Department of Computer and Information Science, University of Pennsylvania, 2001.

[PB84] G. Pahl and W. Beitz. *Engineering Design*. The Pitman Press, Bath, UK, 1984.

[PHP+03] Pradeep Padala, Cyrus Harrison, Nicholas Pelfort, Erwin Jansen, Michael Frank, and Chaitanya Chokkareddy. Ocean: The open computation exchange and arbitration network, a market approach to meta computing. In *In Proceedings of the International Symposium on Parallel and Distributed Computing 2003*, 2003.

[PKE01] David C. Parkes, Jayant Kalagnanam, and Marta Eso. Achieving budget-balance with vickrey-based payment schemes in exchanges. In *Proceedings of the Seventeenth International Joint Conference on Artificial Intelligence*, pages 1161–1168, 2001.

[Plo94] Charles Plott. Market architectures, institutional landscapes and testbed experiments. *Economic Theory*, 4(1):3–10, 1994.

[Pri97] Tony Curzon Price. Using co-evolutionary programming to simulate strategic behavior in markets. *Journal of Evolutionary Economics*, 7:219–254, 1997.

[RJB99] J. Rumbaugh, Ivar Jacobson, and Grady Booch. *The Unified Modeling Language Reference Model*. Addison-Wesley, Reading, MA, 1999.

[RN98] Ori Regev and Noam Nisan. The popcorn market Ů an online market for computational resources. In *Proceedings of the first international conference on Information and computation economies*, pages 148–157. ACM Press, 1998.

[RS04] Ali Roumani and David Skillicorn. Large-scale resource selection in grids. In *Proceedings of the First International Workshop on Grid Computing and its Application to Data Analysis*, 2004.

[RSW94a] A. Rustichini, M. Satterthwaite, and S.R. Williams. Convergence to efficiency in a simple market with incomplete information. *Econometrica*, 62(5):1041–1063, 1994.

[RSW94b] Aldo Rustichini, Mark A Satterthwaite, and Steven R Williams. Convergence to efficiency in a simple market with incomplete information. *Econometrica*, 62(5):1041–63, 1994. available at http://ideas.repec.org/a/ecm/emetrp/v62y1994i5p1041-63.html.

[RWMMO04] Daniel M. Reeves, M. P. Wellman, Jeffrey MacKie-Mason, and A. Osepayshvili. Exploring bidding strategies for market-based scheduling. *Decision Support Systems*, forthcoming, 2004.

[San99] Tuomas Sandholm. *Multiagent Systems: A Modern Introduction to Distributed Artificial Intelligence*, volume 1, chapter 5, Distributed Rational Decision Making, pages 201–258. MIT Press, 1999.

[Set05] Seti@Home. User statistics. Website, accessed: 12.02.2005, Seti @ Home website (http://setiathome.ssl.berkeley.edu/numusers.html), 2005.

[SHC96] K. Steiglitz, M.L. Honig, and L.M. Cohen. A computational market model based on individual action. In Scott Clearwater, editor, *Market-Based Control: A Paradigm for Distributed Resource Allocation*, Hong Kong, 1996. World Scientific. http://www.cs.princeton.edu/ ken/scott.ps (abgerufen am 11.02.2002).

[Sho82] G. Lynn Shostack. How to design a service. *European Journal of Marketing*, 16(1):49–63, 1982.

[Sho84] G. Lynn Shostack. Designing services that deliver. *Harvard Business Review*, 62(1):133–139, 1984.

[SIT00] W. Smith, I.Foster, and V. Taylor. Scheduling with advanced reservations. In *Proceedings of the International Parallel and Distributed Processing Symposium (IPDPS) Conference*, 2000.

[SJ89] Eberhard E. Scheuning and Eugene M. Johnson. A proposed model for new service development. *Journal of Services Marketing*, 3(2):25–34, 1989.

[Smi56] Wendell R. Smith. Product differentiation and market segmentation as alternative marketing strategies. *Journal of Marketing*, 21(1):3–8, 1956.

[Smi89] Veron Smith. Theory, experiment and economics. *Journal of Economic Perspective*, 72(5):151–169, 1989.

[Smi94] V. Smith. Economics in the laboratory. *Journal of Economic Perspectives*, 8(1):113–131, 1994.

[SMT02] Kumaran Subramoniam, Muthucumaru Maheswaran, and Michel Toulouse. Towards a micro-economic model for resource allocation in grid computing systems. In *Proceedings of the 2002 IEEE Canadian Conference on Electrical & Computer Engineering*, 2002.

[SNW04] Björn Schnizler, Dirk Neumann, and Christof Weinhardt. Resource allocation in computational grids - a market engineering approach. In *Proceedings of the WeB 2004, Washington*, pages 19–31, 2004.

[Som01] Ian Sommerville. *Software Engineering*. Pearson Studium, München, 6 edition, 2001.

[SSGL02] Tuomas Sandholm, Subhash Suri, Andrew Gilpin, and David Levine. Winner determination in combinatorial auction generalizations. In *Proceedings of the first international joint conference on Autonomous agents and multiagent systems*, pages 69–76. ACM Press, 2002.

[SSS00] Katrina Stanoevska-Slabeva and Beat F. Schmid. Requirements analysis for community supporting platforms based on the media reference model. *Electronic Markets*, 10(4):250–257, 2000.

[SW02] M. Satterthwaite and S.R. Williams. The optimality of a simple market mechanism. *Econometrica*, 70:1841–1863, 2002.

[Tes02] L. Tesfatsion. Agent-based computational economics: Growing economies from the bottom-up. Isu economics working paper no. 1, Department of Economics, Iowa State University, Ames, February 2002.

[TH93] D. S. W. Tansley and C. C. Hayball. *Knowledge-based systems analysis and design : a KADS developer's handbook*. Prentice Hall, New York, 1993.

[VBW04] Srikumar Venugopal, Rajkumar Buyya, and Lyle Winton. A grid service broker for scheduling distributed data-oriented applications on global grids. Technical report, grids-tr-2004-1, University of Melbourne, Australia, 2004.

[Vei03] Daniel Veit. *Matchmaking in Electronic Markets*. Springer Verlag, 2003.

[WBPB03] R. Wolski, J. Brevik, J. S. Plank, and T. Bryan. *Grid resource allocation and control using computational economies*, chapter 32, pages 747–772. Number ISBN: 0-470-85319-0. John Wiley & Sons, 2003. Booktitle: Grid Computing: Making The Global Infrastructure a Reality.

[WCW04] Ilka Weber, Clemens Czernohous, and Christof Weinhardt. Simulation of ending rules in online auctions. In Stefan Klein, editor, *Proceedings of the 11th Research Symposium on Emerging Electronic Markets (RSEEM)*, University College Dublin, Ireland, 2004.

[WHH+92] C.A. Waldspurger, T. Hogg, B.A. Huberman, J.O. Kephart, and W.S. Stornetta. Spawn: A distributed computational economy. *IEEE Transactions on Software Engineering*, 18(2):103–117, 1992.

[WHN03] Christof Weinhardt, Carsten Holtmann, and Dirk Neumann. Market engineering. *Wirtschaftsinformatik*, 45(6):635–640, 2003.

[Wil85] R. Wilson. Incentive efficiency of double auctions. *Econometrica*, 53(5):1101–1116, 1985.

[WMMRS03] Michael P. Wellman, Jeffrey MacKie-Mason, Daniel M. Reeves, and Sowmya Swaminathan. Exploring bidding strategies for market-based scheduling. In *4th ACM conference on Electronic commerce*, pages 115 – 124, San Diego, CA, USA, 2003. ACM Press.

[Wol98] Rich Wolski. Dynamically forecasting network performance using the network weather service. *Cluster Computing*, 1(1):119–132, 1998.

[WPBB01a] R. Wolski, J. Plank, J. Brevik, and T. Bryan. G-commerce: Market formulations controlling resource allocation on the computational grid. In *Proceedings of the International Parallel and Distributed Processing Symposium (IPDPS)*, 2001.

[WPBB01b] R. Wolski, J.S. Plank, J. Brevik, and T. Bryan. Analyzing market-based resource allocation strategies for the computational grid. *International Journal of High Performance Computing Applications*, 15(3):258–281, 2001.

[WWW01] Peter R. Wurman, Michael P. Wellman, and William E. Walsh. A parametrization of the auction design space. *Games and Economic Behavior*, 35(1-2):304–338, 2001.

[WWWMM01] Michael P. Wellman, William E. Walsh, Peter R. Wurman, and Jeffrey K. MacKie-Mason. Auction protocols for decentralized scheduling. *Games and Economic Behavior*, 35:271–303, 2001.

[Zmu83] R.W. Zmud. *Information Systems in Organizations*. Scott, Foresman and Company, Glenview, IL, 1983.

Die zunehmende Vernetzung von Rechners über das Internet ließ die Vision von Application Layer Netzwerken aufkommen. Sie umfassen Overlay-Netzwerke wie beispielsweise Peer-to-Peer Netzwerke und Grid Infrastrukturen unter Verwendung des TCP/IP Protokolls. Ihre gemeinsame Eigenschaft ist die redundante, verteilte Bereitstellung und der Zugang zu Daten-, Rechen- und Anwendungsdiensten, während sie die Heterogenität der Infrastruktur vor dem Nutzer verbergen. In dieser Arbeit werden die Anforderungen, die diese Netzwerke an ökonomische Allokationsmechanismen stellen, untersucht. Die Analyse erfolgt anhand eines Marktanalyseprozesses für einen zentralen Auktionsmechanismus und einen katallaktischen Markt.

ISSN 1864-9300