David Aubert and Arnstein Seljeflot Solli

Financial Derivatives Market for Grid Computing

Geneva, December 14, 2007

Master´s thesis

NTNU
Norwegian University of
Science and Technology
Faculty of Social Science and Technology Management
Department of Industrial Economics and Technology Management

David Aubert and Arnstein Seljeflot Solli

Financial Derivatives Market for Grid Computing

Geneva December 14, 2007

Norwegian University of Science and Technology
Faculty of Social Science and Technology Management
Department of Industrial Economics and Technology Management

CERN, European Organization for Nuclear Research
Directorate Services Unit
Technology Transfer Group

Abstract

This Master thesis studies the feasibility and properties of a financial derivatives market on Grid computing, a service for sharing computing resources over a network such as the Internet. For the European Organization for Nuclear Research (CERN) to perform research with the world's largest and most complex machine, the Large Hadron Collider (LHC), Grid computing was developed to handle the information created. In accordance with the mandate of CERN Technology Transfer (TT) group, this thesis is a part of CERN's dissemination of the Grid technology.

The thesis gives a brief overview of the use of the Grid technology and where it is heading. IT trend analysts and large-scale IT vendors see this technology as key in transforming the world of IT. They predict that in a matter of years, IT will be bought as a service, instead of a good.

Commoditization of IT, delivered as a service, is a paradigm shift that will have a broad impact on all parts of the IT market, as well as on the society as a whole. Political, economic and physical factors advocate a market for standardized computing resources supplied by multiple professional providers, benefiting from economies of scale. We argue for the trade of Virtual Servers as the standardized bundle of computer resources.

Continuous trade of homogeneous resources allows for scheduling market efficiency and liquidity, but may entail a risk of erratic, unpredictable prices. We therefore construct a complete, coherent Grid economy, consisting of both a spot market and a derivatives market. While the spot market is the trading place for the computer resources, the derivatives market aims to disperse the risk among those who are willing to invest in it.

Because the Virtual Servers are non-storable assets, normal arbitrage theory cannot be used to price derivatives contracts. We propose to solve this issue by creating storable swap contracts priced by an auction-based market, where we argue that the price process follows a geometric Brownian motion. Taking into account the absence of arbitrage in the swap market and the requirement for a complete market, we offer a theoretical framework for martingale pricing and hedging of derivatives written on swaps.

Contents

Appendices

Preface

This Master thesis is written at the Norwegian University of Science and Technology (NTNU). The authors are with this thesis completing our Master of Science degree in Industrial Economics and Technology Management, specializing within Accounting and Finance. The thesis was written through the Technical Student Programme at CERN, the European Organization for Nuclear Research, in Geneva, Switzerland, from March to December 2007. The topic was agreed on in collaboration with the Technology Transfer (TT) Group at CERN, in accordance with their strategic goals of disseminating technologies. Using the author's techno-economic background and insight TT wanted to explore the potentials of Grid Computing.

All errors and omissions in this thesis are the sole responsibility of the two authors.

Geneva, December 14th 2007,

David Aubert

Arnstein Seljeflot Solli

Acknowledgements

The work on this thesis has been challenging and educational. We would like express our gratitude to our supervisor at NTNU, Associate University Professor Snorre Lindset, for supporting our stay at CERN and giving us guidance throughout the process.

We sincerely thank Henning Huuse for his supervision and helpful comments and Jean-Marie Le Goff of the TT Group for his interest and discussions. The entire TT Group deserves a special acknowledgment for accommodating us and teaching us the field of Technology Transfer. The knowledge and experience gained through our operational work at TT has given us a good foundation for the development of this thesis.

Furthermore, we are particularly grateful to John Erik Sloper, without him we would not have been able to run our simulations on the Grid. For the speedy set up of our Virtual Organization "blaubert" and ensuring us acceptance on Grid sites, we would like to thank Nicholas Tackray, Antonio Retico, and Danilo Dongiovani. We are as well grateful to the Ben-Gurion University of the Negev, Israel; University of Patras, Greece; The Italian National Institute for Nuclear Physic; The Portuguese Instrumental Laboratory for Physical Particle Experiment; for letting us use their Grids. We would also like to thank Håvard Bjerke for the set up of a Virtual Machine for us at CERN.

A special thanks goes to Sidsel Hvaal for her constructive linguistic remarks. Thanks are also given to Katrine Aam Svendsen for her "programming hotline" support. We are also grateful to our parents who made this thesis possible.

Finally yet importantly, we thank the academic staff of the Industrial Economics and Technology Management Department at NTNU for 5 years of exceptional education and the department's support of our stay at CERN.

Permissions of Use

Figure 2.1illustrating the Large Hadron Collider is printed with authorized permission to use by the copyright holder CERN and (Mouche, 2006). Figure 3.1 illustrating the Grid is printed with authorized permission to use by the copyright holder CERN and (de Gennaro, 2003). For figures 2.1 and 3.1 we have obtained Permissions of Use by Ms Renilde Vanden Broeck with the CERN Press Office, 2007.11.15.

Figure 3.4 illustrating the GÉANT2 topology is printed with authorized permission to use by the copyright holder DANTE (Delivery of Advanced Network Technology to Europe). We obtained Permission of Use by Ms. Helen Martindale Public Relations Officer at DANTE.net, 2007.12.03.

Chapter 1

Introduction

This thesis is a study of the feasibility and properties of a financial derivatives market on Grid computing, which is a service for sharing computing resources over a network. In the following pages, we explain that a financial derivatives market is not only feasible, but depending on the underlying spot market may also become a necessity. A derivatives market offers the possibility of planning with a longer time perspective compared to spot market trade. In addition, some contract types give the holder the possibility of unbound profit combined with limited possibilities for loss. In this market, the most likely participants would be *risk mitigators* selling the risk of trading on the exchange market, as well as arbitrageurs and speculators taking opposite positions. In addition, functions such as *clearinghouses* and *market makers* provide stability and liquidity. This derivatives market is depicted in figure 1.1.

We propose *swap contracts* to be the connecting link between the exchange market and the derivatives market. The contracts in the derivatives market are then written with the swap contract as underlying. With this little trick, we may use the standard, no-arbitrage pricing framework to price the derivatives. We will also provide the hedging formulas and show that the writer of the Grid derivatives manages to hedge the contract sale.. We will, use the European swaption as a pricing- and hedging example and refrain from proposing new and exciting contracts.

The exchange market consists of both a *Continuous Virtual Server Exchange* (CVSE) and a special-purpose *Over-The-Counter* (OTC) market. The CVSE enjoys the greatest quantity of trading, while the OTC market supports the CVME. On the CVSE standardized bundles of computer resources, *Virtual Servers* are traded. Regarding the market participants, Virtual Server providers are large computer centers that virtualize their physical hardware and provide it to the highest bidder. The buyers consist of organizations that provide the *Hardware layer*, either internally, as a *User Integrator* and a *Software Integrator*, or externally by selling *Hardware-as-a-Service*. Figure 1.1 illustrates the exchange market.

This thesis is part of CERN Technology Transfer's dissemination of Grid technology, in accordance to the Technology Transfer group's mandate to disseminate inventions spinning out of CERN's fundamental research program. CERN is a large stakeholder and one of the largest investors in this new technology and wishes therefore to facilitate organizations and companies that are interested in adopting CERN's expertise and know-how. We will look at different aspects of the Grid and place this technology in a larger context, looking at its potentials beyond CERN's traditional mass-throughput computing use.

In part I we establish the need for Grid computing and look at trends pointing at a trading market for computer resources. Chapter 2, "CERN, at the Technology Frontier", describes the challenges this large fundamental research organization faces, with respect to

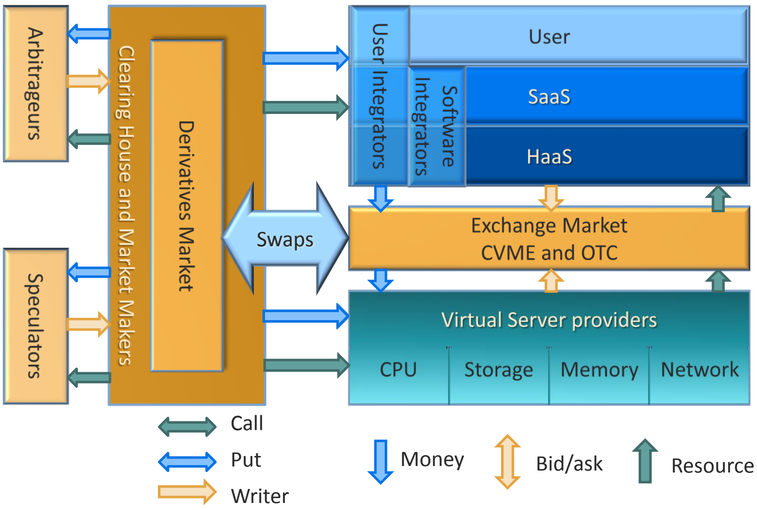


Figure 1.1: The complete Grid economy showing the buyers and sellers on the exchange market. It also illustrates the use of swap contracts as the connection between the spot and derivatives markets. The derivatives market, to the left in the figure, consists of the Risk Mitigators, Arbitrageurs, Speculators, and the supporting participants: clearinghouse and market makers.

processing and distributing data. Grid computing is a new technology in rapid development.

Much of the information and innovation regarding the commercial evolution of the Grid is being published on blogs and Internet articles. Therefore, to get the background information of the Grid several authors of such Internet references will have their say on the current commercial use and future prospects in chapter 3 "Grid Computing". At the end of chapter 3, the previous chapters will be summarized.

Part II outlines an exchange market and a derivatives market for Grid resources. In chapter 4 "Exchange Market" we propose a spot market based on a continuous auction on bundles of computer resources, which we have baptized Virtual Servers. We discuss the advantages and problems of such a market. The biggest problem is the possibility of large and erratic price movements, making it difficult for organizations to plan ahead. Chapter 5 "Derivatives Market" proposes to solve this issue with a Grid derivatives market, creating predictability. We show how to price and hedge contracts on Grid swaps and give a throughout example. We round off part II with a summary of chapters 4 and 5, before we finally conclude.

Part I

The Grid

Chapter 2

CERN, at the Technology Frontier

In this chapter[1], we take a brief look at the European Organization for Nuclear Research (CERN), one of the users and developers of Grid computing. We describe CERN's scientific purpose, that is, the study of the smallest components of the universe, and how this endeavor has lead to important discoveries and new technologies. The study of particles involves a huge amount of information that need to be stored, processed and communicated between the scientists, and this has lead to the development of Grid computing at CERN, an enabler of advanced sharing of computer resources. Dissemination of new technologies is an important part of CERN's activities. To explain the origin of the Grid we will give an overview its background in science.

This thesis is a part of CERN's effort to disseminate Grid computing, being an integrated part of the organizations mandate. First, we give an overview of the background for CERN and what this centre has become. Second, we look at the world's biggest instrument for particle research, the LHC. Third, we briefly describe the computer challenge associated with gathering all the data from this instrument. We then state that this problem is overcome by Grid computing, which is the topic of chapter 3.

2.1 CERN in the History Books

The European Council for Nuclear Research was founded in 1952 with the goal of creating a fundamental High Energy Physics (HEP) research organization in Europe. The main motive was to build up a Euopean stronghold in fundamental research. It was argued that only a large facility could keep the scientists in Europe, and the then prestigious atomic research was decided to form the scope of the laboratory. Nuclear science requires substantial funding, and few countries in Europe were capable of constructing large national laboratories on their own. Therefore, a joint European effort became the answer of how to re-launch the European research. Based on the work of the Council, the original twelve member states ratified the convention to establish the permanent organization European Organization for Nuclear Research (CERN) [2] in September 29th 1954 (Hermann, Krige, Mersits, & Pestre, 1987).

CERN, situated in Geneva, Switzerland, is the world's largest particle physics labora-

---

[1] The information in this chapter is mostly based on the "about CERN" section at the CERN website: www.cern.ch

[2] The acronym CERN is derived from the French translation of the name of the Council: Conseil Européen pour la Recherche Nucléaire. The acronym has been kept even though the organization's name does not contain the same letters.

tory (Bressan, Streit-Bianchi, & Marcastel, 2005), run by 20 member states[3]. 4, 500 scientists from the member states and 1, 700 from non-member states, from a total of 500 universities, study and work at CERN. These scientists represent a large specter of expertise, making the Organization the tone setting organization within fundamental research.

Since its inception, CERN has always been a leading actor in different fields. The laboratory strives to understand the composition of the matter that composes the universe, and has fostered three Nobel Prize Laureates in physics. The discovery of the W and Z bosons rewarded Carlo Rubbia and Simon van der Meer with the Prize in 1984, while the 1992 Nobel Prize was awarded to Georges Charpak for his invention and development of particle detectors. The Laboratory not only attracts Nobel Prizes, but has till now also attracted half a dozen Nobel Laureates (Hämmerle, Ménard, Sutton, & Gilles, 2004).

2.1.1 CERN Today

CERN is today a European as well as a world center of fundamental research, stretching far beyond its original *raison d'être*. To direct the future of the organization Dr. Robert Aymar, the Organization's Director General, has defined four missions for CERN. Among these, performing **research** and **dissemination of technology** are assigned a particular importance (CERN, 2004).

2.1.1.1 Technology Transfer

The undisputed main activity of CERN is fundamental research. However, all CERN research should, according to the missions of the Director General, be directly beneficial for society. It is stated in Article II.1 of the Convention (1953) Convention that "The results of its [CERN's] experimental and theoretical work shall be published or otherwise made generally available". The member states are increasingly anxious to have tangible results from the funds they contribute to CERN. Consequentially, CERN has created the Technology Transfer (TT) policy and established a TT group. "The CERN Technology Transfer activities are aimed at maximizing the technological and knowledge return to the Member States and promoting CERN's image as a Centre of Excellence for technology" (CERN, 2007d).

The CERN knowledge and technology dissemination is made possible using a set of different methods, including the publishing of articles[4]; licensing, i.e. selling the legal permission to exploit a patent; as well as through collaborative work with companies to develop technologies further.

The most famous spinout of CERNs research that has been disseminated is the World Wide Web. It was invented in 1990 by Tim Berners-Lee and Robert Cailliau in order to ease the sharing of information between researchers. The Web soon spread to other institutions in Europe, and three years after more than 200 Web servers were in use. Today, about 19% of the world's population consult and edit 25 billion existing Web pages (CERN, 2007a; WorldWideWebSize.com, 2007; InternetWorldStats, 2007). Other technologies spinning of CERN's research and benefiting society include technologies for renewable energy, such as solar power; green and safe Thorium reactor technology, radiation therapies, such as Hadron Therapy; Radiopharmaceutical compounds; and Medical imaging to mention a few

---

[3] The member states are: Austria, Belgium, Bulgaria, the Czech Republic, Denmark, Finland, France, Germany, Greece, Hungary, Italy, The Netherlands, Norway, Poland, Portugal, the Slovak Republic, Spain, Sweden, Switzerland and the United Kingdom. Many other countries, as well as the European Commission and UNESCO, have Observer status.

[4] CERN published about 1, 000 articles in various journals in 2006.

(Le Goff, 2007). However, CERN should not deviate from its original mission of fundamental research. Technology Transfer at CERN is therefore not comparable to commercially applied research, where the focus is on selling products or technology. The technologies and inventions commercialized from CERN are spin-offs of the fundamental research, or developments made to facilitate research. For CERN to make advances in HEP, challenges in many technical domains must be overcome. Cryogenics, vacuum, materials, and informatics are just some of the fields that the scientific personnel at CERN need to develop as intermediary steps towards the ultimate goal of studying elementary particles.

2.1.1.2 Tools for Performing Research

CERN's physicists use particle accelerators to perform experiments and study the basic building blocks of the Universe. Particle accelerators are machines that propel electrically charged particles to travel almost at the speed of light. Radio frequency pulses push the particles forward and powerful magnets keep them in the trajectory. After reaching the desired energy level, groups of particles are aimed at a collision point where detectors register every detail of the impact.

What the scientists seek are rare processes occurring in the interaction points, whose probability varies with the collision energy. This means that for physicists, the most important parameters are the beam energy and the number of interesting collisions, given by the intensity. As was stated at one CERN lecture by Gilardoni and Metral (2007): "The history of accelerator physics has been a 100 year long fight to get energy and intensity to such a level to study known and unknown particles and their interactions."

Searching for higher energies in order to understand how the Universe was created, has led to the building of larger and more powerful accelerators than the world has ever seen. Until the year 2000, four big particle-particle experiments were running at the Large Electron Positron (LEP) collider complex, an accelerator with 27 km circumference. In the spring of 2008, CERN scientists will complete the replacement of this giant machine with an even more powerful titan, the Large Hadron Collider (LHC).

2.2 Large Hadron Collider

The Large Hadron Collider, LHC, is situated in a 27 km circular tunnel, situated about 100 meters underground at the French-Swiss border. It is the world's largest scientific instrument and will, when it is switched on in the spring of 2008, be the most powerful particle accelerator ever built. The *material cost* of the LHC project is about €7.38 billion. The accelerator (3.03 billion), the computing (0.80 billion) and the experiments (3.55 billion[5]) are the largest cost drivers (CERN, 2007c).

The LHC is a particle-to-particle collider that will produce head-on collisions between two beams of hadron particles of the same kind, that is, either protons or lead ions. The particle beams will be accelerated in an ultrahigh vacuum, comparable to outer space, to prevent unwanted particle collisions (Group, 2005). The proton collisions generate, 10^{16} K, a temperature equivalent to a billion times the temperature of the centre of the Sun. The particle collision recreates about $1/10$ of a billion of a second after the Big Bang, i.e. the birth of the Universe[6].

---

[5] The LHC is a collaboration project where CERN's contribution to the experiments and the computing cost are accounted to about 20% (von Rüden, 2006; CERN, 2007c)

[6] Recreating the Big Bang using the LHC might seem like a bad idea, but both US and EU Specialist reports has deemed the LHC safe (Buszaa, Jaffe, Sandweiss, & Wilczek, 1999; Blaizot, Iliopoulos, Madsen, Ross, Sonderegger, & Specht, 2003).

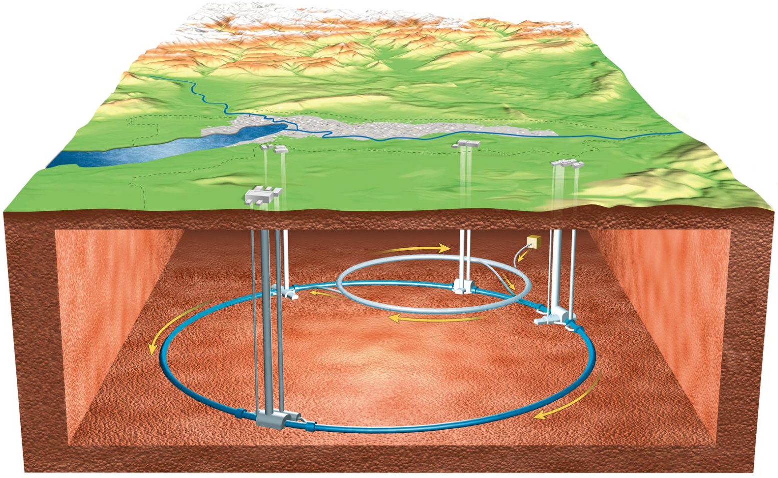


Figure 2.1: Artistic illustration of the Large Hadron Collider with the smaller, preliminary accelerators. copyright CERN (Mouche, 2006).

As the particle beams travel through the 27 km long ring, $7,000$ superconducting electromagnets, operating at extremely low temperatures, will guide them. With a temperature of $-271.3°C$ $(1.9K)$, it makes the conductor the coldest object in the Universe. Each beam will consist of nearly $3,000$ *bunches* of particles, where each bunch consist of 100 billion particles. Since the particles are minuscule, i.e. 10^{-10} m, the chance of two particles colliding is very small. When the two particle beams cross there will only be about 20 collisions among the 200 billion particles. However, the two particle beams will cross about 40 million times per second, generating roughly about 1 billion collision events per second combined for the four detectors.

These detectors convey as much information as the entire European telecommunications network does today. The information produced by the LHC will account for 1% of the information generation in the world (CERN, 2007b). This immense information production creates information technological issues that must be addressed.

2.3 Information and Computing Challenge

The collision event data from the detectors will be heavily filtered so that only about 100 events of interest per second will be recorded permanently. Each event represents a few Megabytes of data, so the total (continuous) data rate from the experiments will be in the order of 1 Gigabyte per second. The LHC will in total produce about 15 million Gigabytes of data annually, equivalent to about 3 million DVDs each year (CERN, 2007b).

In order to provide the necessary computing infrastructure for the LHC, computing power equivalent to more than $100,000$ of today's fastest standard PC processors is required. Even with a computer centre upgrade, CERN can only provide a fraction of the necessary resources. The solution is therefore to connect the computing resources of the world's particle physicists[7]. The Grid computing technology makes this collaborative solution possible. The computing Grid will allow thousands of scientists to access and analyze the LHC data in a seamless fashion, independent of their location.

The Worldwide LHC Computing Grid (WLCG) was launched in 2002 to integrate thou-

---

[7] 267 institutes and $4,603$ users in Europe in addition to 208 institutes and $1,632$ users elsewhere (Berlich, 2006).

sands of computers worldwide into one global computing resource, which will be used to store and analyze the huge amounts of data produced by the LHC. The WLCG operates the world's largest scientific Grid, depending in particular on two major science Grid infrastructures provided by Enabling Grid for E-SciencE (EGEE), and US Open Science Grid (OSG) as well as several national and regional Grid structures. EGEE alone has over 240 sites in 45 countries, including more than $41,000$ CPUs and about 5 Petabytes of storage[8](EGEE, 2007b).

CERN develops its own technologies, but constructs and implements them in close collaboration with industrial partners. In this manner technology and knowledge is transferred through CERN's purchases to benefit industry and society. As a result of this, the development of the Grid for the LHC at CERN and its collaboration with industry[9], has pushed this technology closer to the commodity market (CERN, 2007b). However, CERN is only one of the developers of Grid solutions, where others focus on the commercial market.

Chapter Summary:

This chapter has treated one of the big scientific users of Grid computing, CERN. It established that CERN is the leading organization within particle research with a goal of spreading CERN technology to societal benefit. We looked at the massive computer requirements for the start up of the world's largest machine, LHC. In chapter 3, we look closer at the Grid technology and its use outside CERN, and in part II, we propose a trading market and a derivatives market for buying and selling Grid resources.

---

[8] To monitor the Grid in real time, see: http://gridportal.hep.ph.ic.ac.uk/rtm/

[9] These partners include IBM, Hewlett-Packard, Intel, Oracle, SUN among others.

Chapter 3

Grid Computing

In chapter 2 we looked at the huge computing requirements demanded by the start-up of CERN's new particle accelerator. In this chapter, we look at the solution to this problem, namely Grid computing. First, we give an overview of what the Grid really is and how it is built. Second, we discuss benefits of using a Grid and we look at companies using Grid computing today. We propose a model of the grid-layered economics and describe some factors pushing the dissemination of Grid. Finally, we look at the prospects for Grid in the future, whereby we conclude that a trading market for Grid resources is probable. Different setups of such a market are examined in chapter 4, where we will further develop this idea by suggesting a financial derivatives market on the resources in chapter 5.

3.1 The Grid, what is it?

In this section, we give an overview of what Grid is and provide a discussion of the terms used for this concept. We show the layered architecture of the Grid, and its ability to decouple the hardware from the software.

3.1.1 The Grid explained

While the World Wide Web enables us to share information, the Grid enables us to share computer resources, such as processing power and storage. Based on the definitions of CERN (2007b), Oracle (2007), SUN (2007), Foster and Kesselman (2003), IBM (2007a), Hewlett-Packard (2007), Intel (2007) we define a Grid as follows:

Definition 1 ***Grid*** *is a service for sharing computer resources over a network, such as the Internet.*

The term Grid was introduced by the National Computer Science Alliance in 1997 wanting to transform the Internet and the Web into a powerful tool for advanced computational science and engineering: a prototype for the 21st century's distributed computing[1] infrastructure, with a name derived from the notion of the electrical power grid (Stevens, Woodward, DeFanti, & Catlett, 1997). The analogy to the power grid, where IT resources and services are provided as a metered service through a plug in the wall, is old. Fernando Corbató at MIT[2], envisioned in 1965 a "computer facility as a utility like a power company or water company." (Vyssotsky, Hill, Corbató, & Graham, 1965) and in 1969, Len Kleinrock at

---

[1] Distributed computing is according to www.whatis.com, any computing that involves multiple computers, that are remote from each other, where each have a role in a computational problem or information processing.

[2] The Massachusetts Institute of Technology (MIT)

UCLA[3] stated that " . . . we will probably see the spread of 'computer utilities', which, like present electric and telephone utilities, will service individual home and offices across the country." (UCLA, 1969). Such a global utility computing Grid is illustrated in figure 3.1.

Figure 3.1: Illustration of the Grid.Copyright CERN (de Gennaro, 2003).

3.1.1.1 One technology, many names

Grid computing is an emerging technology, which is used differently by various organizations and termed thereafter. The Science community, with CERN in the lead, and IT infrastructure vendors, such as IBM (2007a), Hewlett-Packard (2007), SUN (2007), Oracle (2007), and Intel (2007), use the term *Grid computing*, while other vendors use the terms Utility computing or Cloud[4] computing.

- *Utility computing* is according to IBM's Gayek, Nesbitt, Pearthree, Shaikh, and Snitzer (2004), Rappa (2004), Meyer (2003) and Accenture (2004), Willard, Joseph, and Lamy (2007) a financial outsourcing model, run by Grid infrastructure, for delivering specific IT services on a pay-as-you-go basis.

- *Cloud computing* is a service model for offering advanced Web Services over the Internet, build on a Grid architecture (Harris, 2007d; LaFollette, 2007). Companies using this term include Google, eBay, Yahoo, Amazon, Microsoft and IBM(Glider, 2006; IBM, 2007b).

Semantically, Grid-, Utility-, and Cloud computing are often used in an interchangeable manner and have been somewhat blurred by various marketing strategies. According to SUN's CTO, Greg Papadopoulos, all these technologies are about accessing resources and applications through a network, representing that computing move *from a good towards a service*. He states that "this transition is already well on its way under names such as Grid-, Utility- or Cloud computing" (Mitchell Waldrop, 2007). The Grid technology is, according to Accenture (2004), an infrastructure that forms the basis of this paradigm shift, and we therefore see these systems as a part of the term Grid computing.

---

[3] University of California, Los Angeles (UCLA)

[4] Cloud is a term used by network professionals when illustrate the Internet (About.com:Google, 2007).

3.1.2 The Grid layers, Hardware and Software layers

The Grid architecture can be described as being made up by layers. The Grid connects physical computer resources, using a network and makes them appear as a virtual pool of resources, the virtual Hardware layer. The software and applications that the users see and interact with, is the Software layer, which lies on top of the Hardware layer.

3.1.2.1 Hardware layer

The physical resources that form a Grid may be servers; PCs; processing units; storage units; clusters of computers; supercomputers; instruments such as telescopes; hand-held electronic devices; sensors, e.g. RFID; PDAs; home appliances, or any device that can be connected to a network. Since the network could be an Intranet or the Internet, the physical resources[5] in a Grid can be, and are in most cases, geographically distributed.

In order to share the physical resources between a set of users, the resources need to be organized and integrated. The software program that performs this is termed *middleware*. It creates the virtualized Hardware layer by using *virtualization* techniques (Group, 2006). Virtualization can make physical computer resources seem aggregated, emulated and partitioned. The *aggregation* gathers the connected physical resources enabling them to dynamically join and leave the Grid, making it seamlessly scalable.*Emulation* makes one resource appear as something else and "conceals" the physical characteristics of the resources, thereby improving the flexibility of the system. After the virtualization has aggregated and emulated the physical resources, they appear as one entity, which can be seen as a "virtual pool" of resources. The *partitioning* dynamically executes the priorities and boundaries for how much the different users can maximally consume of the virtual pool. The resources provisioned to a user from the pool can be dynamically and seamlessly scaled up or down, while it is being used. This allocation rules are given in accordance to the priorities of the organization for the situation when there is no idle capacity in the Grid and could be.

3.1.2.2 Software layer

The provisioned resources from the Hardware layer are for instance used by Virtual Machines. A *Virtual Machine* (VM) provides a complete environment in which an operating system and many processes for multiple users can coexist. The VM is thus a software implementation of a computer that can execute programs, giving a look and feel of using a physical computer (Smith & Nair, 2005). The VMs run independently of each other and if the software on one VM crashes, the other VMs run unaffected. Following from this we can say that the Hardware layer supports the layer that the user of the Grid will see and interact with, namely the Software layer. This layer includes all kinds of different operation systems and user applications such as the ones needed for science, engineering, business, finance, etc.

By aggregation, emulation and partition, virtualization creates a separation of hardware and software. It decouples the software from the physical resources on which they rely (Armijo & Nickolov, 2007), thus creating a Hardware layer and a Software layer.

From this, we can say that the Grid is a shared, network scaled, computer system. Grid computing enables the virtualization of distributed computing and data resources to create a single system image, granting users and applications seamless access to vast IT capabilities (IBM, 2007c, 2007b). IBM explains: "just as an Internet user views a unified instance of

---

[5] Depending on the what resources that are used and where they are located, names such as Enterprise Grids, Desktop Grids, Scientific Grids, Data Grids, Equipment Grids, etc. are sometimes used.

content via the Web, a grid user essentially sees a single, large virtual computer" (IBM, 2007c).

3.1.2.3 Users and Sharing

The Grid as well offers the possibility of cross-organizational work both by aggregating resources from each member and the possibility of accessing software through the network, from anywhere in the world (Accenture, 2004). The shared resources in the Grid can span multiple administrative and organizational domains, where users can create *Virtual Organizations* (VOs). VOs are organizations where the members work together via, for instance, the Internet, and are geographically spread while appearing to others as a single, unified organization (HighScalability, 2007). The Grid incorporates a Hardware layer and a Software layer, which the users see and interact with. This is illustrated in figure 3.2. In section 3.3, we take a look on who the users of Grid computing actually are, but first, in section 3.2 we discuss some of the benefits of using a Grid.

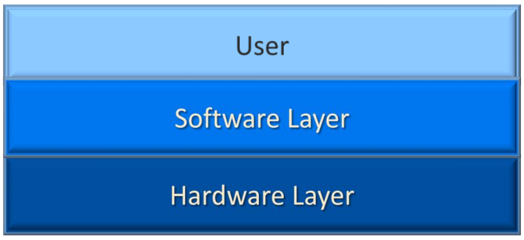


Figure 3.2: The Grid layers, Hardware layer and Software layer.

3.2 Benefits, why is the Grid interesting

Is the Grid just another type of supercomputer? According to Erwin Laure, from CERN and EGEE, a supercomputer exists to solve the most demanding computing problems, while the purpose of Grids are to federate computation and data, which makes for an effective tool for collaboration and dynamic reconfiguration (Harris, 2007c). The Grid provides more cost-effective utilization, fault tolerance, flexibility and scalability, than traditional computing systems.

3.2.1 Providing the necessary resources cheaper

One of the main benefits of Grid computing is that it allows reducing the cost of computing. While supercomputers require custom components, a Grid can be built using standard, inexpensive, off-the-shelf computer components. If there is need for more computer resources, the Grid can be dynamically scaled up adding more resources. In extreme cases, such as for the Large Hadron Collider, Grid computing is the only way to obtain the needed resources. The effect of the scaling means that the computers can be kept longer and updates can be done gradually by adding extra resources when needed. This is an ideal situation for start-ups, keeping their initial computer investments to a minimum, while having the option gradually to expand with an eventual growing market.

The pooling of resources also makes it easier to handle an extension of computer resources. In a traditional storage regime, the different hard drives are separate units. Adding extra storage requires that someone has to make decisions on which files to move and must also notify all the users and applications on where to find their data. When a new unit is added in a Grid, the Grid itself takes decisions concerning where to store data (Patel, 2007).

3.2.2 Improving ROI by sharing and using idle resources

In traditional computing, computers are all separate systems. Some hard drives get maxed out, thus reducing overall performance, while others are underutilized (Patel, 2007). Most servers only use 20% of their capacity; desktop computers only 5% to 10% (Meyer, 2003). Grid takes advantage of the unused computing capacity using virtualization to "pool" the shared resources. The idle capacity that was previously inaccessible is in this way made available for use. This reduces the amount of resources wasted and improves the utilization of the existing IT infrastructure, hence reducing the IT cost and the impact on the environment as well as the Return-On-Investment (ROI) (Wallom, 2007).

3.2.3 Access and Integration

The Grid enables sharing of geographically distributed data and computing infrastructure between users. Grid is therefore an effective method of providing unified access to computational resources (e.g. CPUs, storage systems, and data sources) irrespective of their physical location.

3.2.3.1 Access to software

With a Grid there is no need to download the data to the local computer, since it can be read and processed at its physical location through the Grid. A result of this, software can easily be altered, because it is only necessary to store one copy in the Grid, which can be accessed by all the users. This means that all the software can be updated from a single location. Large companies are therefore using Grid architectures to provide various business application and services (Strong, 2007). Employees can then access these business applications through a Web portal, instead of installing them locally. The whole Enterprise Resource Planning (ERP) system may be outsourced to ERP-vendors who offer a Web-interface on this service (Currie & Seltsikas, 2001).

3.2.3.2 Flexibility of access

Access to the Grid and its resources can be provided anywhere in the world where there is a broadband Internet connection. This is especially useful for devices with little or no internal processing power or storage, termed thin clients, such as PDAs and cell phones, enabled by the next generation of mobile networks, see section 3.3.4.1. These handheld devices can become gateways to the computer resources and data in the Grid (Kumar & Song, 2005; Mattmann & Medvidovic, 2006).

3.2.3.3 Collaboration

Geographically dispersed individuals, or groups across multiple administrative domains can create Virtual Organizations, mentioned in section 3.1.2.3, to ease collaboration. Regardless of location, Grid allows people, computing language, hardware and software architecture, to join efforts towards a common goal (Gerrity, 2007). This increases the flexibility of working outside the physical boundaries the organization.

3.2.4 Data safety, robustness and availability

In a traditional storage regime, all access to data is lost if the server goes down. In addition, these systems cannot withstand a simultaneous loss of more than one disk. If such a failure occurs, the data on those disks will be lost (Patel, 2007). To prevent this, separate copies for backup are made. Compared with ratios of 5-to-1 of backup required vs. original data, an advanced Grid based system only requires a ratio of 1.3-to-1. The data is split into many smaller parts, which are spread across the Grid. In addition to the reduced probability of simultaneous failures of geographically spread computers, any lost data can be restored by using the information from the other parts (Scheier, 2007). The lesser need for backup data, archive handling, and disaster recovery is one of the large cost savers of the Grid. SUN estimates that only 30% of all enterprise storage systems are used effectively (Wittmann, 2007).

The inherent property of the Grid is that it contains no single point of failure, which could cause the whole system to collapse. If one processing unit or resource goes down, the workload aimed at the failed unit is redirected, with little delay.

There are businesses that use Grid computing for one or more of the above-given reasons; we look at some of these segments and companies in section 3.3.

3.3 Commercial Use

A report by the Gartner Group, states that Grid computing has so far reached five to 20% of the targeted clients. Based on the classification of different technologies according to their media exposure combined with the technology maturity, Gartner estimates that Grid computing is 2-5 years from mainstream use (Claunch, Weiss, & McGuckin, 2007).

The Grid is an emerging technology and has therefore some challenges that have to be worked out before it will become mainstream. The 451 Group performed a study on Grid adoption in businesses, where one of the topics concerned difficulties. Among the biggest issues mentioned by the industry were licensing (Dornan, 2007), cultural resistance, security and the lack of standards (Wallage & Fellows, 2007).

3.3.1 Areas of application

The Grid is a general Information and Communication Technology (ICT) solution that can benefit any sector. Grids are either used to offer internal or external services. It has a wide range of application and is used in among other Banking & Finance; Insurance; Media & Entertainment; Transport & Logistics; Manufacturing; Retail; Oil & Gas; Pharmaceutical, as well as in the Web Industry (Gedling, 2007).

According to DataSynapse (2007), **Banking & Finance** is one of the largest commercial users of Grids, where all the major financial banks and institutions are using Grids. They are using Grids to reduce the processing time for portfolio pricing and hedging, as well as for financial and economic modeling. According to Micheal Yoo, Head of the Technical Council at UBS "everybody is doing the same thing; it is just about who is fast enough" "it is a new type of arms race... where last year's exotics are this year's commodities."(Yoo, 2007).

In **Manufacturing**, simulation of different scenarios, stress testing and design requires a lot of computing resources. According to IBM, Airbus saved €18 million on their A380 airliner project, by cutting lead time on the wings by 36% and eliminating re-entry of data because of improved collaboration with suppliers and subcontractors (IBM, 2006).

In the **Pharmaceutical** industry, Grids are used for computational chemistry, screening techniques, computational fluid dynamics, and finite element analysis. The Pharmaceutical sector was one of the first adopters of Grid computing. Today, according to Johnson & Johnson Pharmaceutical Research & Development, all the major Pharmaceutical companies have a Grid (Harris, 2007b). The kind of modulation the Grid offers reduces a drug's time to market by two years, saving $ 264 million in R&D cost, according to The Boston Consulting Group. In addition, the patent's market lifetime is extended by getting the product faster through the R&D. According to GlaxoSmithKline, a new drug can sell for about two to three million dollars a day, and one extra year in sale would be worth about $ 900 million (Cohen, 2003).

Very significant users of Grids also include the **Web Industry**. Companies such as Google, eBay, Yahoo, Amazon, and Microsoft use grid-like systems to offer advanced Web services externally. "No single computer could update millions of auctions in real time, as eBay does, and no one machine could track thousands of stock portfolios made up of offerings on all the world's exchanges, as Yahoo does" (Glider, 2006).

3.3.2 A Service Layer Business Model

We have now seen that the industry uses the Grid, we will now look at how they are using it. There is a trend in the market towards providing different types of Grid services on an on-demand basis. From the Hardware and Software layer point of view these services can be defined within a Service Layer Business Model. In this business model transactions are done between the two layers and the Users, both internally and to external organizations. Many companies from section 3.3.1 have their Hardware and Software layers in-house to serve the needs of their organization. We term these organizations *User Integrators*. We term organizations that offer their Software layer as an external service, using their in-house Hardware layer, *Software Integrators*. As well, vendors can offer the Software and Hardware layers independently, not having the other layer. Organizations offering their Software layer as a service provide *Software-as-a-Service* (SaaS). Organizations offering their Hardware layer externally offer *Hardware-as-a-Service (HaaS)*. Each of these vendor models is about achieving greater economies of scale and reduction of risk than any single company can achieve (Wardley, 2007b). These service layers are depicted in figure 3.3.

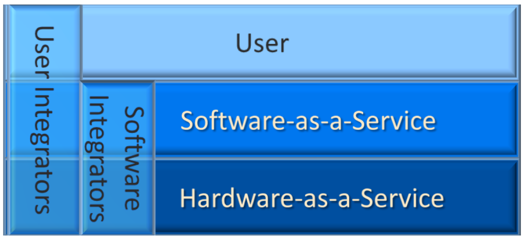


Figure 3.3: The layered economic service structure illustrates how Hardware-as-a-Service and Software-as-a-Service can be offered separately or internally by Software Integrators and User Integrators.

3.3.2.1 Hardware-as-a-Service

A Hardware-as-a-Service (HaaS) vendor, also called a *Grid host*, is someone who offers the Hardware layer, as described in section 3.1.2.1, and sell these to other companies. They are offering their resources on on-demand, pay-per-use basis. In the lead are large Web and IT companies, building supersized IT infrastructures, such as Amazon and SUN[6] and IBM (Willard et al., 2007). However, many smaller HaaS vendors are also emerging, such as 3Tera, Layered Technologies, Media Temple and Concentric (SuperbInternet, 2007). The buyers of these services are often Web companies envisioning themselves as the next YouTube, MySpace, or Google, requiring scalable solutions and given immediate provisioning, if they are successful. Entrepreneurs can therefore save costs when developing new Web services by buying the hardware as service from the HaaS Vendors, instead of owning and running their own IT infrastructure.

3.3.2.2 Software Integrators

A Software-as-a-Service (SaaS) Integrator is an organization or a company that provides software on-demand to its clients, using for instance pay-per-use, or ad-based revenue models. Google is a one example of a Software Integrator. Their office package Google Docs & Spreadsheets, as well as their photo editing tool Picasa, run on their own Grid and give the application user the software he needs wherever he is. The growth of such Web services spurred Microsoft change their business strategy to "software-AND-a-service" instead of only "software". The service strategy is to offer advanced Web software services to consumers, information workers, and IT staff (Ozzie, 2007).

3.3.2.3 User Integrators

An organization is a User Integrator if it provides all the services internally, both the Hardware layer and the Software layer to its members. Most companies that run Grids today are User Integrators, including pharmaceutical companies, and banking and finance.

Due to the growing importance of Information & Communication Technology (ICT) in the economy and the benefits of the Grid, as mentioned in section 3.2, there is a strong political push towards having more users run their applications on Grids.

3.3.3 Political Initiatives

The ICT is of key importance to the European Union (EU). 7% of the EU's gross domestic product, and half of the productivity gains in the EU's economy are due to ICT (EU, 2006; CORDIS, 2004). At the Lisbon European Council in March 2000 the Union set itself a new strategic goal for 2010: *"... to become the most competitive and dynamic knowledge-based economy in the world capable of sustainable economic growth with more and better jobs and greater social cohesion."* (EU, 2000). To support the Lisbon Strategy the EU initiated the "i2010" policy framework to increase its investment in ICT research by 80% (EU, 2005; CORDIS, 2004).

The Grid plays a vital role in EU's long term strategy of becoming the world leading information-economy. CORDIS (2007c) states that: "Grids will become the engine of the future knowledge-based economy and society". The EU countries therefore spent €841 million on Grid research and deployment (EU, 2006) in 2002-2006. These, however, are the initiatives of individual countries. The EU's main instrument for funding scientific research and technological development, and to meet the goals of the Lisbon Strategy, are the *Framework Programs (FPs)* (CORDIS, 2007a).

---

[6] Amazon's Simple Storage Server (S3) and Elastic Computing Cloud (EC2) Web Services offers a pay-as-you-go service for general purpose computing at $ 0.10 an hour, while SUN offers one CPU-hour per $ 1 (Martin, 2007; Harris, 2007d).

The European Commission's FPs are put in place to ensure EU's global leadership in ICT (CORDIS, 2007b). With the framework programs from 1998 - 2006, €280 million was spent on Grid technology development and pilot projects. The funding is even increasing. In the FP7 programme (2007-2013) €9.1 billion, 64% of the total FP budget, is allocated for ICT. The amount directly related to Grid is not yet known, but all the Grid projects from the previous FP program continue. In addition, EU grants €585 million for "Network and Service infrastructures" for 2007-08, and €600 million for an e-Infrastructure over the entire FP7 (EU, 2007), where much of this synonymous with Grid technologies. There are also numerous national and international Grid initiatives outside the EU[7], for instance ChinaGrid, the Japanese NAREGI, and the US TeraGrid (Gentzsch, 2007).

Where will this enormous commercial, political and technological push lead? It is a paradigm shift from computing as a good to computing as a service, which we argue will result in trading of computer resources, eventually leading to a professional market. In section 3.4 we discuss the business prospects and we build a trading market for Grid resources in chapter 4. First, we look at some requirements for such a market.

3.3.4 Technical Market Enablers

In order to have a market for trading of computer resources, two fundamental technical factors must be in place: it must be possible to transport large amount of data over the network, and open standards for communication between computers must be widespread.

3.3.4.1 Network Bandwidth

In the 80s, computers were expensive and large mainframes computers, accessed by thin client terminals by a network, represented the norm. In the late 80s, the price of computing power decreased and the network bandwidth became the bottleneck. Since then, the bandwidth technology has evolved rapidly, from 56 Kbps (Kilo bits per second) in 1985 to 155 Mbps in 1997 (Ahn, Black, & Effrat, 2007). Ten years later, 1 Gbps is not unusual, and the big laboratories communicate with a bandwidth of more than 10Gbps (DANTE, 2007), see figure 3.4.

The increase in network bandwidth has made it more interesting to again ship the data around so that computers no longer have to act as islands. Gilder Technology Report from 2000 states that "when the network is as fast as the computer's internal links, the machine disintegrates across the net into a set of special purpose appliances"(IEETA, 2003). This is exactly what the Grid enables and what a Grid market aims to do. It then depends on the availability of reliable high performance networking to connect the "pool" of distributed resources. Even private households obtain higher and higher speeds on both their cable and wireless connections, and will soon be capable of embracing all the possibilities Grid Computing offers. In order to run Grids with success, a bandwidth of at least 20 Mbps is needed (Casasús, 2007). This is not regarding the bandwidth that is needed to access and use the Grid, which is easily covered with today's ADSL connection. In addition to a well-developed cable network, mobile networks are reaching these speeds with technologies such as HSDPA[8], HSUPA, 4G/LTE. For instance the huge telecom companies AT&T, Verizon, and Vodafone are planning to move to 4G/LTE in 3-4 years and will be capable to offer speeds up to 100 Mbps (A1-telecom, 2007; Electronista, 2007). Further in the future speeds up to 380 Mbit/s can be offered with the MIMO technology (Group, 2007).

---

[7] For more Grid initiatives see EGEEs collaboration list (EGEE, 2007a).

[8] 85% of the Austrian population is already provided with 7.2 Mbit/s HSDPA networks (A1-telecom, 2007)

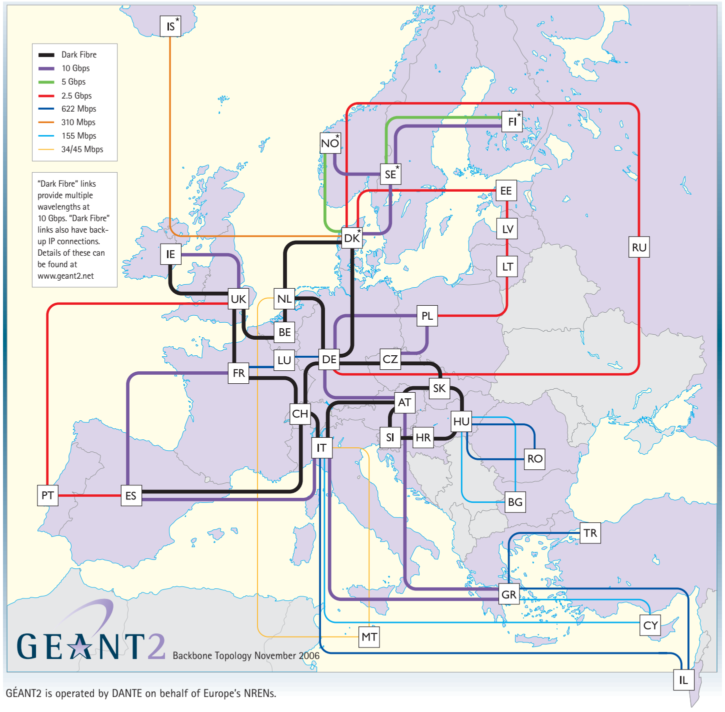


Figure 3.4: The different laboratories are connected through very high bandwidth capacity, via their National research network, in the science Grid: GÉANT2.Copyright DANTE.

Looking at history, Arpanet, a research network supported by the US Department of Defense's Advanced Research Project Agency, became the backbone of the Internet (Chetty & Buyya, 2002). Today's global science e-Infrastructures could be the next ICT infrastructure for society. As Arpanet was opened to the public, multi-gigabit research networks such as GÉANT2, Internet2, TEIN2 and SEEREN2 (GÉANT2, 2007) might someday as well be open.

However, it is not sufficient with fast network speed if the resources are unable to communicate with each other. Therefore, there is an industry push for open standards.

3.3.5 Open Source and Security

Mass adoption of Grid technology and data transfer between different organizations requires open standards. We look at the analogy of the World Wide Web from CERN as a testimony of this. According to one of the inventors of the Web, Berners-Lee (2007), the success of this technology depended on open technical standards for growth and innovation. Without open standards, the Web would probably not be the success it is today.

The Open Grid Forum (OGF) is an organization dedicated to developing open standards for Grid interoperability, solving the security issues, and accelerating Grid adoption. It is a community of users, developers, and vendors, involving more than 400 organizations from 50 countries. The members of OGF include large vendors such as Hewlett-Packard, IBM, Intel, Microsoft, Oracle, and Platform Computing (Ehrig, 2007). In addition, Google, Amazon, Ebay, Facebook, MySpace and Yahoo, among others, are using open-source software[9] in building their IT architectures. We see a joint effort among commercial actors to establish

---

[9] Open source software is computer software whose source code is available, under a license with relaxed or non-existing intellectual property restrictions, that permits users to freely use, change, improve, and redistribute the software in modified or unmodified form (TheOpenSourceInitiative, 2007).

open standards and open source towards a common protocol for inter-Grid communication. In addition, the network capacity and speed is moving towards the possibility of sending huge amounts of data over large distances. We will now discuss one direction the future ICT may take towards the construction of a commercial trading market.

3.4 Business Prospects, Supply & Demand

In this section, we look at trends and expectations on the future of Grid computing. We see how the increase in demand of computation resources results in a bigger commercial market. As well, outsourcing of computer services and the growth of professional actors will probably lead to increased trading of computer resources.

3.4.1 Commercial Predictions

From a market perspective, the Grid represents a growing billion-dollar business. The Gartner group estimates that Grid technology will enter mainstream adoption within 2010 (Claunch et al., 2007). This is supported by the Insight Research Corp., which predicts that the worldwide Grid spending will grow from $ 714.9 million in 2005 to approximately $ 19.2 billion in 2010 (Wellner & Weissberger, 2005). They examined Grid spending in 14 industry sectors and found that the majority of companies in the manufacturing and the financial service industries would invest in Grid computing in near future. As mentioned in section 3.1.2, virtualization is a technique in Grid, but it is also marketed as a separate solution. The market for virtualization is therefore also an indicator of the Grid market. An IDC report from July 2007 states that the market for virtualization services will grow from $ 5.5 billion in 2006 to $ 11.7 billion in 2011 (Harris, 2007a). These market estimations are however, based on a traditional use of Grids within an enterprise, i.e. a Work Grid. If the benefits of the Grid are realized in other domains, such as the creation of public- and home Grids, the adoption process would be much faster (Aubert & Solli, 2007).

3.4.2 Extreme Growth in Demand for Computing Power

Forrester Research states that while growth in IT budgets in 2007 has been stronger than expected, data center managers are still being asked to do more, even with pressure of cutting costs. A survey from 2006 by Winter Corp. shows that the growth is fastest among very large databases, where the biggest database was more than three times the size of the largest reported two years earlier. This increase in computer center size is also valid for the future. IBM projects that the world's information base, the raw material for databases, will be doubling in size every 11 hours(!) by 2010 (IBM, 2007d), however, this is not the amount of data that will be saved. External factors, such as governmental legislation are also pushing the need for processing. For instance, the US Food and Drug Administration is imposing new quality-assurance tools, known as Process Analytical Technology, that requires the pharmaceutical industry to dramatically increase the data collection and processing(Martin, 2007). There is therefore a dramatic increase in the processing and storage need. However, will it be greater than the general increase in computing power?

3.4.2.1 Outgrowing Moore's Law

It is important to emphasize that these rates of growth are unprecedented, even in the early days of the Internet boom. Growth in computer capacity has been known for the last 40 years to follow Moore's Law. This "law" [10] is based on Gordon Moore's techno-economic observation that the number of transistors on a chip would double every two years, resulting in a annual growth of about 42% (Martin, 2007).

SUN argues that there are two groups of companies: the ones that will grow at about the same rate as the Gross Domestic Product, who are not outpacing Moore's Law, and those whose processing needs are by far outgrowing Moore's Law. The companies that are exceeding Moore's law and driving exponential growth in both user demand and computing requirements are Web companies, such as Google, YouTube, etc. as well as large financial, energy, and pharmaceutical companies (Martin, 2007). For many organizations, computer power will be scarce and it will be more costly to satisfy all computing needs.

3.4.3 Towards Professional Trade

One solution to the problem with too little computing power is the emergence of large professional computer center serving multiple organizations. These centers profit from large-scale advantages and are therefore capable of providing computing resources to a competitive price as well as sustaining a near full utilization at all times. We look at the trends towards professional providers and why organizations wish to outsource their computing needs.

3.4.3.1 Professional Actors

"No IT staff has the time or expertise to lay out a data center for maximum efficiency" (Wittmann, 2007). A professional actor could, however, focus on reducing operational costs, while improving business continuity, scaling capacity to support business growth, and improving service levels (Martin, 2007).

IDC predicts that providers delivering Grid computing will become more professional as more customers see the benefits of buying computing power (Willard et al., 2007). The growing professionalism for delivering computing is according to the CTO of SUN Greg Papadopoulos a trend that boils down to efficiency at scale, i.e. real work performed versus capital expended, power consumed, space used and the number of people needed to keep it all working (Martin, 2007). This means that a professional provider of Grid computing services could double the scale and reduce the marginal cost of serving a new customer, or performing some unit of computation. As scale increases, so does efficiency, including the opportunity of co-design of cooling and power conditioning and distribution. Gartner Research backs up this trend of Grid-run cloud and utility computing services in a special report from of October 2007. Their research includes the insights of more than 40 analysts and more than 30 pieces of research. They assess 14 models for delivering IT that they say will completely transform the IT market in the next five years. Among which, according Wittmann (2007), the Grid is a key technology. The conclusion is according to Gartner, that this represents an irreversible trend toward the industrialization of ICT services. They argue that IT is progressively being delivered as a service, giving a broad impact on IT organizations, businesses, and IT vendors. (Da Rold, Margevicius, Cohen, & Bittman, 2007).

---

[10] Some argue that this has become a self-fulfilling prophecy, or a standard, that the chip makers feel obligated to reach to survive in the business (Ulaby, 2006).

3.4.3.2 Outsourcing

The trend of exponential growth in demand for raw computing power is the factor that makes outsourcing of computing interesting. SUN says that the trend is about a return to computing as a service and a shared infrastructure. The idea is old, as mentioned in section 3.1.1, but what is new is that there is an explosion in demand, a set of technological advances in high-speed networks, and a move towards open standards.

SUN, IBM, and Hewlett-Packard are believers of a large-scale shift to much of today's in-house computing loads into the Internet cloud, i.e. Grid-run cloud computing. As well, Google, Amazon, IBM, and Microsoft are, or are planning to, offer "in-the-cloud" computing services to consumers as well as businesses (Martin, 2007; IBM, 2007b).

Ross and Westerman (2004) states that the service-provisioning model of utility computing relies on Grid computing and allows the buying and selling of computing capacity. The main benefit for buyers is that the cost of computing is based on the actual resource use, rather than a fixed cost for capacity they only use during peak periods. IBM expects that for most firms much of the impact of the utility model will be on the extent and nature of outsourcing. By letting others take care of updating the computer farm, may allow firms to access state of the art technologies and technical skills with reduced costs. The on-demand capacity may allow firms of all sizes to invest less in computing capacity and should enhance management's ability to focus on strategic competencies.

There is a technology risk associated to Grid computing outsourcing. Firms may counter these risks by outsourcing key infrastructure components, those most demanding of reliability and security, to established vendors. However IBM says the growing number of vendors, such as SUN and Amazon, reduces reliance on a single vendor, should offset some technology risk. At the same time, outsourcing business processes and higher-level infrastructure services may allow them to spread the technology risk of their outsourcing (Ross & Westerman, 2004). It is probable that an internal Grid would be kept within the organization, in order to process data of key strategic importance, in parallel with outsourcing of normal computation. With respect to this, the development of standards is of key importance in order to allow communication between the internal and external Grid.

There is a balance between the economics of Grid-run utility computing and the need to control your own infrastructure[11] (Martin, 2007).

Gartner estimates that most of this move towards outsourcing will be concentrated in the next five years. A broad set of assessed delivery models, which is already used by early adapters, will become mainstream by 2012 and making traditional models obsolete (Da Rold et al., 2007).

3.4.4 Competition in Grid Resources Offerings

We have now seen that the large vendors and IT research & advisory firms estimate that IT industry will shift to delivering IT as a service, using a set of Grid service provisioning models. The companies providing these services will have to optimally manage cost factors such as space, power, cooling, maintenance, amortization of equipment and personnel (Martin, 2007). We now look at the factors that argue for multiple computer centers, instead of a few enormous ones.

Electricity is a key factor. A specialization of computer centers to Grid farms seems imminent, where the large Web companies are already paving the way. However, there are

---

[11] Why run a computer center yourself?, At a talk on "Commoditisation of IT", Simon Wardley characterized it as "Yak Shaving" (meaning no-sense) and that it is "common as chips" (in the sense that it is moving to a point where running a data center is not a part of the core competency of a company.)(Wardley, 2007a).

limits to how large a compute farm can become, with respect to how much electricity the electricity-grid can supplying to one single location. It is impossible to provide an unlimited amount of electricity to only one single location. According to the US Environmental Protection Agency, data center power usage will double by 2011 (Wittmann, 2007). Gartner predicts that half of the data centers in the world will not have enough electric power to meet the computing power and cooling requirements of the high-density computing equipment by the end of 2008 (Morgan, 2006). This is a physical factor to why several computer centers must be built geographically distributed and is a factor reducing the market entry barriers for new providers.

Cooling is a further factor. Because of cooling requirements, it might be interesting to locate the computer centers away from the users in favorable location or even climates. About 60% of the power consumption of a data center is attributed to cooling of components (Morgan, 2006). Dayne Sampson, Vice President of the fastest growing search engine Ask.com, gives an example of the electricity need for powering and cooling. He estimates that the five leading search companies, Google, Ask, Yahoo, Microsoft, AOL, have about 2 million servers, each dissipating 300 watts of heat annually, a total of 2.4 gigawatts. This is equivalent to half of what is needed to power the Las Vegas metropolitan area on the hottest day of the year (Glider, 2006).

The need for computer power as well as the increasing electricity consumption has prompted the big Web companies to start looking for places with comparative advantages as well as specializing themselves to get a competitive edge. Optical networks, which move data over vast distances without degradation, allow computing to migrate to wherever power is cheapest (Glider, 2006). Looking for such comparative advantages Google is building specialized computer farms of unprecedented proportion, using commodity components (Farber, Dignan, & Berlind, 2007). One is being built in The Dalles, Oregon. Google situated it near an abandoned aluminum plant and a 1.8 gigawatt power station, a source of inexpensive electricity. In addition to electricity and cold climate, Google has access to a 640 Gbps branch of the *Internet2* research network (Glider, 2006).

Another reason to believe that several companies should establish in the Grid resource business is that profitable industries attract more actors. Macro economic theory says that there will enter new companies in a market, given low entry barriers, as long as the price of the service is higher than the production cost. Microsoft and Yahoo are following the example of Google by building in Quincy and Wenatchee, Washington, where they take advantage of rock-bottom electricity prices and un-used fiber-optics laid down during the dot-com boom (Glider, 2006).

Governmental regulations for competition may also have an impact on increasing the number of Grid resource vendors. If one organization gets too much force, the international legislations will work to increase the competition and accommodate new entrance.

Based on the idea of specialization, professional actors with a dedicated computer-Grid-center could sell computer power cheaper than non-dedicated organizations. As more and more Grid-resource vendors emerge with the soaring growth in Grid computing provisioning models, it would not be efficient for organizations to have their own Grid or to cover their complete peak computing need with an in-house Grid, unless it is of strategic importance. We argue that this as a natural consequence will result in a market place for trading of Grid resources[12]. We look at how this market might be set up in chapter 4

---

[12] As Smith (1776) says "Monopoly is a great enemy to good management."

Chapter Summary:

We have now seen that the business prospects of Grid computing is entwined with the service delivering models such as utility and cloud computing. We have argued for the transformation of computing from a good to a service and a growth in outsourcing of computer resources to specialized computer centers.

Summary of Part I

Chapter 2 and 3 built the foundation for a trading market for Grid resources. We began describing one of the biggest users and developers of Grid computing, the European Organization for Nuclear Research (CERN), and its Technology Transfer policy to disseminate technology developed at CERN for the benefit of society. In order to study the universe's smallest building blocks, CERN uses the world's largest and most complex machine, the Large Hadron Collider (LHC). When the LHC is up and running in 2008, 1 Gigabyte of information will be retrieved from its experiments each second. Grid computing is the technology that makes it possible to transmit, store, processed and analyzed this enormous amount of data.

In chapter 3, we looked at what the Grid is and what it will become. Grid is a service for sharing computer resources over a network such as the Internet, and builds on the idea of offering computer resources as a commodity. As trends come and go, names like Cloud, - and Utility computing, which are service provisioning models using Grid, have come to compete with the term of Grid computing. Regardless of its denomination, the Grid effectively decouples the computing hardware from the software. The computer hardware resources are virtualized into a "pool" of resources, which forms a virtual Hardware layer. This layer is then partitioned into virtual bundles of resources to support the needs of the software, termed the Software layer. In this layer, Virtual Machines, i.e. an image of the software on a computer, offers an environment where users can run their applications on an on-demand basis.

Looking at the Hardware and Software layers from an economic point of view, we introduce a layered service business model for the transactions between these layers. Users might provide both the Software and Hardware layer internally, or buy one of these layers from external organizations as Hardware-as-a-Service (HaaS) or Software-as-a-Service (SaaS). We term the users that themselves provide and use both of these service layers themselves "User Integrators". An organization that has its own Hardware layer and sells software services, i.e. Software-as-a-Service, is termed "Software Integrator". Vendors of Hardware layer services offer HaaS, i.e. virtualized resources, for instance to organizations having a Software layer, but does not have their own Hardware layer.

Most organizations using Grid today are User Integrators. The most prominent sectors are Banking & Finance; Insurance; Media & Entertainment; Transport & Logistics; Manufacturing; Retail; Oil & Gas; Pharma, and the Web Industry. They use Grid computing to obtain resources inexpensively, improve the overall computer utilization, reduce the vulnerability of the system, and ease accessibility and collaboration. There also exist many political initiatives to increase the number of companies enjoying the benefits of the Grid.

The Grid is predicted to play an important role in the IT infrastructure of tomorrow. Exponential growth in processing needs is becoming commonplace and more and more companies are exceeding the normal capacity development following Moore's Law.

Once Grid computing is widespread, we envision a trading market for Grid resources.

Advances in broadband and open standard initiatives make such a market possible. We discussed outsourcing of computer resources to professional actors and argued that specialized organizations could benefit from economies of scale. In addition, electricity supply constraints and cooling requirements will limit the size of a single computer center. This constraint is one of the factors that open for competition amongst several Grid resource providers. The interest of governments in competitive markets and the opportunity of profit are two additional factors promoting multiple resource providers.

There are strong indications that there is a foundation to build a market for trading Grid resources. We have argued for the existence of both buyers and sellers, willing to trade Grid resources. In chapter 4 we look at how this trading could take place and in chapter 5 we lay the theoretical foundation a financial trading market for Grid resources.

Part II

The Grid Economy

Chapter 4

Exchange Market

A central problem in globally distributed computing, and within a company, is the allocation of computer resources. In the early days of mainframe computing there were few computers and distribution was not a big issue. The users simply agreed upon the allocation themselves, or a central manager gave different time slots of the mainframe computer to the different projects, based on their importance. When the number of computers increased, the central manager could not efficiently handle the increased complexity of finding the most suitable computer to each job. Automated techniques such as the primitive *Round-Robin*, where each incoming job is matched with the next available resource satisfying the job criteria, were implemented (Gomoluch & Schroeder, 2003). These distribution techniques resulted in an unsatisfactory allocation, as they did not take the importance of the different jobs into account. Sutherland's (1968) description of the bid on computer time slots at the Harvard University is one of the first economic based approaches in distributed computing within a single entity. Research on the use of economic based methods for intercompany allocation started, to our knowledge, with Buyya's (2002) thesis on Economic-based Distributed Resource Management and Scheduling for Grid Computing.

In this chapter, we look at global allocation of computer resources, in the sense that the utility of each user is maximized. This is contrary to maximizing the utility of a set of users with a common goal, such as one organization. The goal of this chapter is to analyze different factors for a Grid market and, based on these factors, make a sensible choice of market mechanism, also referred to as just "mechanism". First, we treat the market and define factors that influence the choice of mechanism both the parties involved and the resources traded. Second, we take a deeper look at auctions, one of the ways in which the mechanism could be arranged. Third, we review one of the auction approaches especially designed for Grid computing: the Multi-Attribute Combinatorial Exchange. Although this model takes into account most Grid aspects, we find this approach too detailed, and we finally propose a new method: the Continuous Virtual Server Exchange, CVSE. The issue with the latter method is that its price process may prove erratic. This problem, however, we will solve in chapter 5 with the introduction of Grid derivatives.

4.1 Markets

The study of open trading for computer resources dates back to the work of Buyya (2002). Before this, it was not possible to fully integrate computer resources and hence not feasible to buy extra temporary, sufficiently large-scale computing resources for a shorter or longer period.

For a Grid resource market to be interesting, there have to be organizations demanding more temporary computing resources than they already have, as well as organizations willing to sell excess resources. As stated in section 3.4, computer problems increase in size and complexity, and organizations are experiencing resource needs that are difficult to satisfy. As a result, a growth in specialized computer resource providers is expected, as stated in section 3.4. We therefore assume that computer resources are scarce and that there exist supply and demand for these resources.

In a multi entity Grid environment, the various users and providers have different goals, objectives and strategies. Find bilateral trading agreements may prove problematic. Markets constitute a competitive environment that naturally balances the common and diverging interest between parties (Tan, 2007).

Definition 2 *A* ***market*** *is a virtual or physical meeting point of supply and demand balanced by means of a mechanism. (Schnizler, 2007)*

Supply consists of the price and quantity of resources that the Grid providers are offering, while the demand is the users' willingness to pay for the different quantities of Grid resources. The market mechanism set up the rules on how to allocate and price the resources the demanders buy from the suppliers. We now look at the agents, resources and the other factors related to the choice of market mechanism. Based on Schnizler (2007), we make assumptions about them as well as their requirements towards the market mechanism. Our choice of market mechanism for the Grid market is a continuous double auction, called CVSE, presented in section 4.4. We look at how well this mechanism satisfies the requirements in section 4.4.3.

4.1.1 Agents

Definition 3 *An* ***agent*** *is one who is authorized to act for or in the place of another (Merriam-Webster, 2007).*

Since the agent represents and acts on behalf of the user, we use the word agent to describe both software agents and the real buyers and sellers of computer resources, be it an agency, a governmental department or a company. Mathematically: let $\mathcal{I}$ be the set of I agents, where $i \in I$ is an arbitrary agent.

4.1.1.1 Who Uses a Grid Trading Market?

The agents in the resource trading market consist of two groups: buyers of resources and sellers of resources.

Referring to the service layer model, figure 4.1, the buyers of Grid resources are agents, who provide Hardware-as-a-Service (HaaS), either internally or externally. HaaS may be an incorporated part of the value chain of a User Integrator or integrated in the service offered by a vendor offering Software-as-a-Service (SaaS), called Software Integrator. Some companies may sell HaaS to external organizations. These Hardware layer sellers offer HaaS to the organizations in the Software layer of the model. We define a buyer on the Grid trading market as follows:

Definition 4 *A* ***Grid resource buyer*** *is an agent that needs more Grid resources in his Hardware layer in order to provide HaaS externally or internally. He is able to attach a value to a resource, and is willing to give a financial compensation for it. Formally: let $\mathcal{L}$ be the sub-set of L buyers from the set of agents, $\mathcal{L} \subset \mathcal{I}$, of L, then $l \in L$ is a buyer of Grid resources.*

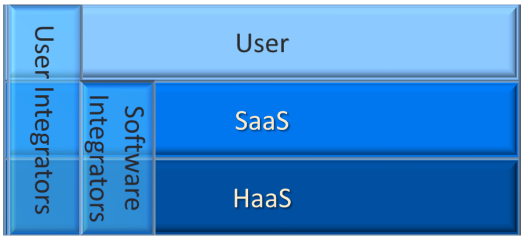


Figure 4.1: The layered service model illustrates how Hardware-as-a-Service can be offered either externally by vendors of HaaS, or internally to Software Integrators and User Integrators.

The sellers of resources could be computer centers of various sizes offering HaaS. We believe that it is most rational, and probable, to create a market with professional resource providers. In section 3.4.3, we stated that specialization leads to cost efficient processing and storage, which again leads to an outsourcing movement.

Definition 5 *A* ***seller of Grid resources*** *is an agent who owns a resource, has a reservation price, and is willing to trade this resource, i.e. offer HaaS, for a financial compensation. Let $\mathcal{M}$ be the sub-set of M sellers from the set of agents, $\mathcal{M} \subset \mathcal{I}$. $m \in M$ is hence a Grid resource seller.*

4.1.1.2 Assumptions About the Agents

We assume that there is a **high number of participants** in the Grid market and approximately an **equal amount** of each agent type, $M \approx L >> 0$. The number or buyers and sellers is an important factor when it comes to market design, because too small a market would have problems with the liquidity, making it difficult to buy or sell resources, while too big a market would have interaction problems. We do not believe that it is rational to talk about a global Grid market where all the providers and all the users are connected on one immense Grid. Such a market would require a lot of overhead in administrating the agents, allocating resources in a good manner as well as sending information over large distances. Instead, we envision smaller international markets, each covering all the resource providers and users in the countries belonging to that market. Even this market would have a high enough number of agents, probably several hundred, to provide the necessary liquidity.

We assume that the suppliers sell resources **regardless of the identity** of the buyers. The providers should be indifferent as to whether the buyers use the resources for e.g. medical research, commercial or military purposes. If this would not be the case, the providers would have done better by not participating in the market, because she is unable to control who receives the resources (Klemperer, 1999).

The **private information** criteria imply that the buyer has a single valuation of the resource based solely on her own preferences. This is opposed to common value where the traded object has some kind of true value, but each participant knows only part of the information regarding the object (Klemperer, 1999). The private information assumption is reasonable, as computer resources do not have an inherent value and one consumer's valuation of a resource is not affected by what the others believe it is worth.

All agents should be **utility maximizing**. The objective for any agent is to do the best she can in the market, i.e. to try to maximize her utility function (Nakai, 2002). We also assume that the agents are greedy[1] meaning they prefer more to less.

---

[1] *"Greed is good"* - Gordon Gekko, "Wall Street" (1987).

4.1.1.3 Requirements of the Agents to the Mechanism

The agents bring with them certain requirements that a market mechanism must satisfy. First of all, taking part in the market must not be a burden. The allocation must be both economical and allocation efficient, the market should reveal the agents' true preferences, and no one should be required to subsidize the market mechanism. We will discuss these requirements one by one.

Individual rationality says that the utility of the participants in the market should be equal or greater than if they did not participate. This should be true for both the resource users and the providers. If the mechanism does not support individual rationality, the market would not be viable as the number of users could be very low. For an agent to be interested in participating in a Grid market, it has to be **economically efficient**. Grid market design is therefore paramount (Tan, 2007; Schnizler, 2007; Nicolaisen, Petrov, & Tesfatsion, 2001; Miller & Drexler, 1988c, 1988b, 1988a; Buyya, 2002). Without an efficient market, the participants would get their computing resources elsewhere (Weng, Li, & Lu, 2007). Efficiency is the ability of the market to allocate the resources from the providers who offer them at the lowest price to the ones who value it the most, hence maximizing the total value over all agents. In order to have an efficient market, the mechanism must be **incentive compatible**. A mechanism is incentive compatible if, regardless of what others do, the agents show their true preferences, i.e. the strategy that maximizes their expected utility. If the market mechanism does not receive true signals from the agents, it cannot allocate the resources efficiently.

Several factors affect the utility of a market participant: the price, allocated resources, as well as the speed of getting the resources. The time taken from a need arises to its fulfillment is an important concept in Grid allocation, called **scheduling efficiency**. The value for the customer of a resource decreases with the length of the allocation time, from a maximum at immediate allocation, to zero when the allocation time reaches the minimum acceptable tolerance limit (Tan, 2007). This is depicted in figure 4.2 A mechanism is **budget-balanced** if it is a closed economy, i.e. if all money paid in to the mechanism from the buyers is redistributed to the resource sellers. Breaking this requirement means that someone from the outside has to fund a part of the allocation, or that the mechanism requires money from the agents.

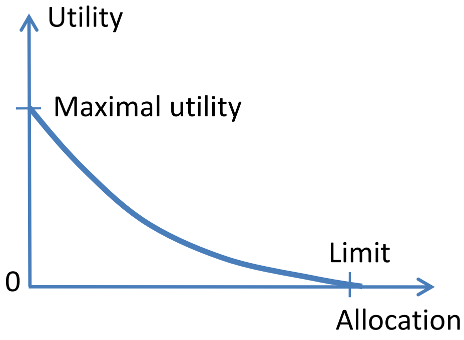


Figure 4.2: Illustration of the utility of a market participant.

4.1.2 Resources to be Traded

Many computational jobs involve the use of different resources such as applications, servers, clusters, processing units, storage, memory, networks, sensors, and other instruments. Many academics have debated what the most suitable trading unit for a Grid market would be. For instance, some follow a physical resource approach where each resource is traded separately (Weng et al., 2007), while others see the traded objects as elementary services where bundles of resources are sold (Schnizler, 2007). We agree that a bundling approach is the most appropriate in a Grid market. Such an approach prevents obtaining only a single resource, for example memory, and not the processing power. Figure 4.3 depicts the resource providers bundling basic resource into a greater logical unit.

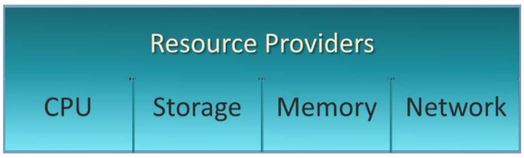


Figure 4.3: Physical resources are bundled into greater logical units.

4.1.2.1 Resource Factors that Determine the Market

The kind of resource traded greatly influences the agents' valuation. If the price is the only factor determining the value for the agents, the resource is said to have a **single attribute**. A resource is **multi-attributed** if other attributes, such as hardware limitations, also affect the agents' preferences. If an agent considers two Grid resource services to be equivalent, they are then characterized as **homogeneous** (Schnizler, 2007). The basic resources are relatively **heterogeneous**, for instance memory systems differ in attributes, such as size, latency, reliability, transfer rate, structure, access time, security and storage cost (Miller & Drexler, 1988c, 1988b, 1988a). The Grid middleware may aggregate and emulate basic resources to make up for some of the heterogeneity, such as size, by making them virtual.

We treat different resource bundling approaches in section 4.3 and 4.4. In the first section we accept that the Grid resources are heterogeneous and multi attributed. We find that this approach gives too many possibilities, making it difficult to allocate the resources. In the latter part, we reduce the alternatives and trade only in homogeneous, single attributed resources, Virtual Servers.

4.1.2.2 Requirements of the Resources to the Market Mechanism

In this section we will provide some of the most important requirements on the resources in a Grid trading market. These expectations state that the market mechanism must support a great variety of resource configurations in order to be fully functional.

It should **support heterogeneous resources**, as well as provide support for **different resource specifications**. This means that the market ought to offer functionalities for providing, for instance, both storage and computation with different qualities in for example the storage size and clock frequency.

The mechanism must support simultaneous allocation of multiple resources in the form of **bundles**, because the allocation of one type of resource may be worthless without another type of resource. For example, a computational unit without storage might be of no use for a customer. **Co-Allocation of resources** requires that it is possible to acquire resources from different providers in the best way possible, e.g. from the geographically closest providers.

The requirement of **substitutability of resources** demands that the mechanism only allocate one bundle of substitutable resources to an agent. A consumer might be indifferent of having one CPU running for four hours or four CPUs running for one hour, however, he would only like to utilize one of these options (Schnizler, 2007).

4.1.3 Other Factors for Choice of Market Mechanism

There are other factors, besides the agents and resources, which are important for the choice of market mechanism. We make assumptions regarding the market and give some requirements that it must support.

4.1.3.1 Assumptions on these Other Factors

Because the different companies have different usage-patterns of computer power, it must be possible to trade Grid resource services **both day and night**. The consumption being continuous and the computer resources non-storable, a pause in the trading means that the buyers would have to wait for the trading to start again, or for them to acquire their resources elsewhere. As the providers would not be able to sell during shut down periods either, the potential traded quantity in the market would be severely reduced. We therefore assume that the market is open around the clock, and will neither have start-up, nor end prices.

We also assume that a job can be **paused and resumed later**. This assumption prevents lock-in of jobs that take longer than expected. An agent might initiate to buy computer resources when she finds it appropriate and does not have to be afraid of potential subsequent price growth. If the job is not of high importance, she could wait for the prices to decrease, to continue the computation.

4.1.3.2 Further Requirements for the Mechanism

The **computational tractability** requirement demands that the mechanism calculate the winner and determine the price within a reasonable period (Schnizler, 2007). This means that the agents' tolerance limit of allocation time determines the maximum time the allocation process can take. Schnizler (2007) requires **automated resource allocation**, stating that no manual allocation should be necessary to match provider and demander. He also says that the mechanism must allow resource owners to announce their resources, and resource requesters to discover them. He calls this the requirements of a **double-sided mechanism**.

A consumer must be **guaranteed resource usage time**. This means that a user must be able to receive resources for the duration he desires. Schnizler (2007) says that the users should also be able to state in advance which time slots they would like to use and that the mechanism should support providers expressing their future resource offer. **Advanced reservation of resource** is a requirement supporting these plans of resource consumption and announcement.

Network quality is also a requirement that the allocation mechanism must consider, according to Schnizler. The mechanism must be able to take the given network bandwidth into account when allocating resources.

The problem is to find a market model and mechanism that support all of the requirements from the agents, resources and the other factors. There exists a variety of different market models, including fixed-price models, dynamic-price markets, and auctions, for details, see Buyya, Abramson, and Giddy (2000) and Buyya (2002). In an auction market, the prices are set dynamically in a decentralized manner. We prefer the auction model to the dynamic-price market because of the decentralized aspect, that is, the buyers and sellers in the market and not a central allocator determine the traded quantity and price (Vytelingum, 2006).

We summarize these factors, assumptions, and their requirements to the mechanism, in the table below

Factor	What/Who	Assumptions	Requirements to the market mechanism
Agents	Buyers: Internal or external providers of HaaS. Sellers: Resource owners.	High number of participants, Equally many buyers and sellers, Do not care with whom they trade, Valuation based on private information, Utility maximizers.	Individual rationality, Economical efficiency, Scheduling efficiency, Budget-balanced.
Resources	Computer resources.	Heterogeneous resources might be virtualized to be more homogeneous, Might be single or multi attributed.	Support of heterogeneous resources, Bundle approach, Co-Allocation, Resource substitutability.
Other Factors	Time, Allocation Complexity, Jobs, Set-up of the mechanism.	Trading both day and night, Jobs can be paused and resumed.	Computational tractability, Automated resource allocation, Double-sided mechanism, Guaranteed resource usage time, Advanced reservation.

4.2 Auction Theory

In this thesis, we take a closer look at auction models to allocate Grid resources. This type of market model is well defined and provides a valuable testing ground for economic theory. In addition, it has enjoyeda long experience period within both research and practice.

Definition 6 *A **Grid auction** is a mechanism, organized by an auctioneer, to distribute Grid resource services from the sellers to the buyers. The mechanism consists of determining the winner and setting the price.*

There are two main kinds of auctions: single-sided and double-sided. In single-sided auctions, many consumers bid on Grid resources from one provider, or many providers submit ask prices to one consumer for delivery of a resource. Single-sided auctions attain approximate economically efficient outcomes if it is a one-to-many or many-to-one relationship between the demanders and suppliers. This does not stick if there are multiple buyers and multiple sellers trading the same type of resource. The single-sided auction would then lead to inefficient allocation (Schnizler, 2007).

Auctions based on a double-sided mechanism allow multiple consumers to place bids on a resource in parallel with many providers submitting asks on the same resource. As the Grid market consists of multiple buyers and sellers, double-sided auctions, or exchanges for short, give results that are more economically efficient than its alternatives. In fact, there exists no other known trading system, where there are sufficiently many buyers and sellers, in which the agents would be better off in expectation, than in double-sided auctions (Klemperer, 1999).

4.2.1 Double-Sided Auction

The double-sided auctions treat the buyers and sellers symmetrically, letting buyers submit bids and sellers ask, see figure 4.4. The exchange might be arranged in different ways, either by letting the agents combine to a bundle different basic resources themselves, called

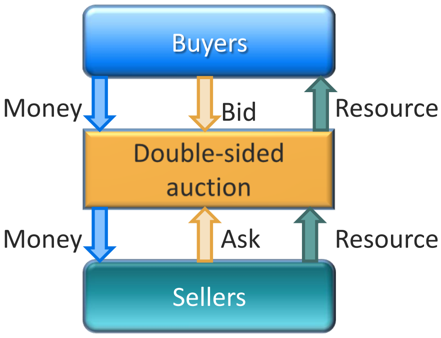


Figure 4.4: Figure showing the bids and asks from the agents, as well as the flow of money and resources in a double sided auction.

a combinatorial auction, or aggregate the resources before selling them, that is a single item auction.

Combinatorial double-sided auction is the simultaneous exchange of heterogeneous resources. It allows the buyer to specify the exact combinations of resources that maximizes his utility. The providers submit their asks on single Grid resource services and the auctioneer solves the possibly computationally demanding optimization problem of allocation. Section 4.3 gives a larger example of a combinatorial exchange, called a Multi-Attribute Combinatorial Exchange.

The **single item exchange** is a double-sided auction where the agents trade only homogeneous resources. Each consumer bids on one or many units of the resource and each provider states the quantity as well as the price for the resources he would like to sell. The auctioneer couples the overlapping bids and asks, distributes the resources and sets prices according to the auction's allocation- and pricing mechanism. In section 4.4, a single item exchange for allocating a single Grid resource, Virtual Servers, is further developed.

4.3 Multi-Attribute Combinatorial Exchange

The Multi-Attribute Combinatorial Exchange (MACE) is a market method for Grid resources proposed by Schnizler (2007). He considers a Grid market where the resource buyers specify, in detail, the resources they would like to buy. This whole section is based at the work of Schnizler (2007). We first look at how Schnizler describes the problem of how to allocate and price the resources, and then state how well the MACE conforms to the requirements set by the Grid market.

4.3.1 MACE Problem Description

In the Grid market, there are several discrete resources. Let X_j be a bundle consisting of one or more of these basic resources, for example it may consist of storage and memory. MACE assumes heterogeneous resources containing multi-attributes, meaning that the price is not the only factor determining the agents' valuation.

The mechanism allows the buyers to specify each quality requirement individually, for instance that the speed of the CPU in a certain bundle should be greater than 1 GFLOPS. A seller in MACE is able to specify the maximum offered quality characteristics.

MACE allows further specification of the time related to resource use. Each allocation period, let us say one day, is divided into discrete time slots. A buyer may wish to have a certain resource bundle over a time span of a certain number of time slots, while also specifying an earliest allocation time and an interval for the resource need. For instance, if the day is divided into 24 periods, he could request resource allocation of a total of 5 periods, with the earliest time at noon and the latest at eight o'clock. The seller makes a similar decision, but states only the starting and ending hour of the periods during which he offers a resource bundle. A buyer $l \in L$ expresses his maximum valuation, v_l of each bundle X_j. One seller, m, may be able to provide a certain part of the bundle, be it whole or only a fraction. Let v_m be the reservation price, that is, smallest amount a seller is willing to sell a bundle of resources X_j during one such time slot. There might be a difference between the value an agent attaches to the bundle and the price he offers or requests for it. The ask price a_m of a seller m for a resource is a result of the qualities and attributes, the start and end times, as well as the seller's true reservation price. The bid price b_l of a buyer also takes into account the composition of the bundle and the total period during which he would like to use it. In addition, when giving a bid, the buyer must specify the maximum number of co-allocations in each time slot. Buyers are not interested in having multiple bundles allocated to them when they only need one, hence they have to express possible substitutions among the bids.

There is quite a lot of information with which each seller and buyer must provide the exchange in order for it to allocate the resources in a proper manner. In most cases, computer programs specify all these factors and give an ask or a bid price. All these possibilities of specifications make it very hard for the allocation and pricing mechanism to decide on the winners. We describe the allocation and pricing in the MACE in section 4.3.2.

4.3.2 Allocation Determination and Pricing in MACE

We will not go into details of how MACE allocates resources, but simply give a brief overview of the allocation method. We refer the interested reader to Schnizler (2007). MACE allocate resources by solving the optimizing problem of maximizing the difference between the buyers' total bids subtracted by the sum of the ask prices when each period, bundle, buyer and seller are considered. The constraints to this maximizing problem includes that each buyer $l \in L$ can only be allocated one bundle X_j, and that each allocated bundle X_j is allocated within the total time defined by the buyer l. Each buyer can get a bundle from multiple sellers, and a seller can only sell a bundle to one customer. Other constraints include technicalities in relation to the earliest and latest allocation times, interdependencies among the Grid resources, as well as quality characteristics. This optimization problem is quite hard to solve and is the biggest drawback with MACE. The MACE allocation mechanism can be reduced to the known Non-deterministic Polynomial time (NP)-complete Capacity Allocation Problem (CAP), which means that the winner determination problem in MACE is also NP-complete. By assuming that the complexity class which can be solved in polynomial time (P) is different from the complexity class NP, the NP-complete problems cannot be solved in polynomial time with increasing input size[2]. The input size in a Grid context is the number of agents providing bids and asks. With a possibly huge number of bidders and askers, who are allowed to specify the resources, attributes, start, stop and total run time, an immense number of bundles is created. The determination of the winner would then take incredibly long[3].

Once MACE has solved the allocation problem, the pricing of the bundles is a much simpler task. The bundles that the agents trade are heterogeneous, and the auctioneer must

---

[2] Let n be the problem size. If the problem can be solved in a time that is a polynomial expression of n, for example $n^4 + 2n^2$, then the problem complexity is polynomial

[3] It is probable that a very big Grid must be installed to cope with such processing requirements.

set the discriminatory price. MACE uses k-pricing, where the price that a buyer has to pay is somewhere between his bid and the combined sellers' asks. As the winning bids and asks are seldom the same, their difference creates an aggregation surplus, β. A factor k decides how much each part should receive from the surplus. The surplus and the resources allocation are depicted in figure 4.5

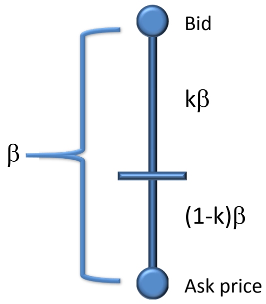


Figure 4.5: Figure showing the surplus and how the resources are allocated. $k\beta$ is allocated to the bidder and $(1-k)\beta$ to the seller.

4.3.3 Analysis of MACE

The biggest problem with MACE, among the requirements set by the agents, resources or other factors, is the computational tractability. As MACE is NP-complete, the calculation of the allocation would possibly take very long. This delay would also negatively affect the scheduling efficiency, since the users would not receive the resources immediately. The second biggest problem is that MACE does not satisfy the demand of economical efficiency. Schnizler's (2007) theorem 5.4, page 105, declares that MACE is "budget-balanced and individually rational," and from the impossibility theorems [4] We know that MACE is not economically efficient. More exactly, the incentive compatibility, which is necessary to have an efficient market, does not apply to MACE. This is because it would be better for an agent to provide a bid or ask price that differs from her true valuation or reservation price. The other requirements, however, are mostly satisfied. We will now present our alternative auction model to the MACE, called CVSE.

4.4 Continuous Virtual Server Exchange

We propose a Continuous Virtual Server Exchange (CVSE) together with a special supplementary resource market. The idea is that a Grid market consists of the combination of a large-scale homogeneous resources exchange with a supporting over-the-counter (OTC) market for exceptional resources. We find that these exchanges do not suffer from the major fault of MACE, the problem of computational tractability. However, by having a continuous market, we introduce the risk of severe price movements. How we may mend these risks is the topic of chapter 5 "Derivatives Market".

4.4.1 Factors

The most important factors that define the CVSE and the OTC markets are agents, resources and time. These two markets have different purposes and handle these factors differently.

---

[4] The impossibility theorems of Myerson and Satterthwaite, and Hurwicz Green and Laffont describes that it is impossible to design an exchange that is economically efficient, budget-balanced and individually rational.

While the agents are the same as in MACE, we provide an entirely different approach to what resources are traded, and the duration of the use of resources.

4.4.1.1 Agents

The agents on this market are approximately the same as in MACE. A buyer of resources, $l \in L$, would most likely be one of the following, cf. section 4.1.1.1 : a vendor of Hardware-as-a-Service (HaaS), Software Integrators or User Integrators.

A resource provider, $m \in M$, could sell homogeneous resources at the CVSE and/or special resources at the OTC market. A seller can participate on the OTC market if he has resources that are considerably better than the standardized resources, or if the resources have a kind of special feature. The only requirement for a seller to participate at the CVSE market is that he provide at least one CVSE resource, called Virtual Server.

4.4.1.2 Resources

Instead of letting each agent specify freely all the resources they would like to have, the CVSE limits the choices to a few standard Virtual Servers. A Virtual Server (VS) is a functional bundle aggregated from geographically separated basic resources, or as a partition from a larger resource, cf. 3.1.2.1. In this manner, a supercomputer could be split into multiple VSs of a certain specification, or numerous smaller resources could be joined to one VS with the same specification. A typical VS consists of storage, processing unit, memory, and communication possibilities. Knowing that it is difficult to find a VS that fits all participants' needs, we suggest trading multiple grades in the CVSE market. For instance, we may have the grades: *superior*, *normal*, and *inferior*. The difference between these groups could, for example, be the design of the resources or the way the different resources are connected. A high-grade classification could require that all the basic resources be situated at the same geographical location. We do not state any specific resource requirements for the different states, but simply state that the different grades should reflect the prevailing opinion, at each point in time, of what the different grades represent. They should thus be dynamic categories reflecting the technological evolution.

Similarly to what was mentioned in section 3.1.2.2 in terms of resources, a set of Virtual Servers can function as the Hardware layer in a Grid, supporting the Software layer. This means that the owner of the Software layer buys the Hardware layer as Hardware-as-a-Service. VSs can also be added and included in the Hardware layer of a Grid, providing the buyer with more computer resources.

Definition 7 *A* ***Virtual Server*** *is a bundle of physical resources provided as a service through a network, giving a functioning computing environment of a certain size. Let* X_j *represent a Virtual Server which offers at least the minimum requirements of a certain grade* $j = \{superior,\ normal,\ inferior\}$.

We design the CVSE market to handle a large quantity exchange of resources, and the OTC market to supplement it. In the OTC market, it is possible to buy resources that are highly heterogeneous and specialized, in order to add more storage or computer power to the VSs. The OTC might also open for trade of telescopes and other sensors, in order to increase the span of resources available to the buyers. Figure 4.6 depicts how the agents interact with the market mechanism in the exchange of Virtual Servers.

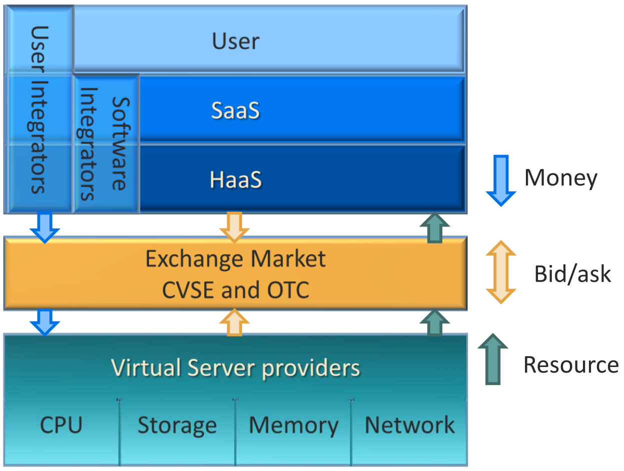


Figure 4.6: Figure showing how the sellers place ask price for Virtual Servers on the market, while the HaaS layer places bids on these resources. After the allocation is complete, money flows from the buyers to the sellers, while VSs go in the opposite direction.

4.4.1.3 Time

The Grid market needs to determine the duration in which the buyer has the resources at his disposal. The CVSE market standardizes this period in order to have perfectly homogeneous resources. The problem is then to find the right time slot. Too short a period will cause the overhead of sending jobs back and forth between the Virtual Servers to be immense. Too long a time slot would result in resource inefficiency, as a user must be certain to have need for the VS during the whole interval. In this thesis, we use one hour as a period whenever we need to state explicitly the duration. We do not claim that this is the correct length, we only emphasize that one single suitable duration must be found in such an exchange market. Again, we still assume that a job can be stopped and resumed at leisure. The agents might therefore have the resources during one half hour, and then pause before starting the job again, without any troubles.

In the OTC market, however, this duration limit should not exist. In this market, the resources should have a time length equal to the buyers' specifications. Here, the users should be able to buy storage for longer or shorter periods than one hour.

4.4.2 How the Trading is Done

In order give a complete overview of the exchange market, we also briefly treat the OTC. A similar process to MACE would probably solve the OTC allocation with great success. The only things that are different from the discussion under section 4.3 are the number of participants and bundling. In this market, as the trading consists of only single heterogeneous resources, there is no need for a bundle approach. It is not difficult to incorporate this in the MACE framework and we rely on this exchange for the OTC resources. In the OTC market, there are few agents, as the CVSE market is handling the gross volumes. Hence, there are fewer problem instances in the OCT optimizing problem compared to the full MACE market, and the time for the OTC auctioneer to find the optimal allocation would be much shorter.

There are several aspects to consider when designing a market mechanism for CVSE, and one single decision may affect the fulfillment of the requirements. One such crossroads is the decision of the allocation frequency.

4.4.2.1 Continuous Allocation and Pricing

An auction may allocate resources in two different ways depending on how long the auction should be open to match buyers and sellers. The Grid market can match the agents either periodically or continuously.

In **periodic allocation**, the auctioneer collects bids and asks during a fixed period and sorts them in increasing and decreasing order, respectively. He then matches the quantity of VSs desired by the highest bid with the quantity provided by the lowest ask(s). This process is repeated until there exist no bids that are higher than the remaining ask prices, and there, the price is set. Figure 4.7 depicts the determination of the winner and the price setting. The bidding-round ends with the auctioneer *clearing the market*, that is, invalidating the remaining un-matched bids and asks, before a new auction period may begin (Tan, 2007).

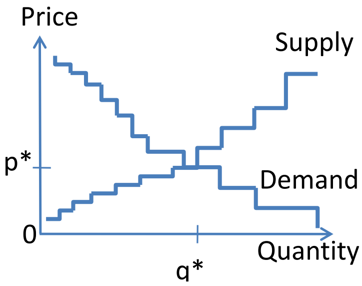


Figure 4.7: The auctioneer collects the bids and asks during a period, ranks them and finds the mechanism price, p^* and quantity, q^*.

There are several other problems with the periodic clearing approach. Uncertainty amongst the participants is one of them. The agents will not know, until the end of the period, if they were one of the winners. Thus, they cannot respond quickly to the changes in supply and demand (Tan, 2007). In addition, the allocation time is very long, making the mechanism scheduling inefficient because the allocation does not happen when the users need the Virtual Servers.

The **continuous allocation** method lets the allocation period approach zero. This means that the auctioneer allocates the VSs as soon as there exists a match between a bid and an ask.

Definition 8 *We define* ***Continuous allocation*** *as follows: during a short period* $[t, t + dt\rangle$, *maximum one agent can enter the auction.*

At any given time, the auctioneer keeps two queues, consisting of unallocated bids and asks, for each VS grade. A gap between the best bid and the best ask, i.e. the highest bid b_{max}, and the lowest ask, a_{min}, constitutes the bid-ask spread. A buyer thus has two choices: give a buy order, meaning a bid that is higher than the current a_{min}, or she could give a book order, that is, a bid that is not as high as the a_{min}. Only in the first case is an immediate allocation effectuated with a price of at least a_{min}. For the seller, her sell and book orders are symmetric to the ones of the buyer. This process is depicted in figure 4.8. At each point in time t when the auctioneer effectuates a buy, or a sell order, the resulting price, let us call it S_t, becomes the market price.

With such short time slots, the market mechanism cannot provide economical efficiency as it has problems with allocating the resources from the sellers with the lowest global reservation price to the buyers with the highest value. It has been shown, however, that

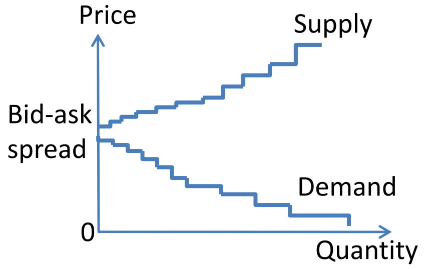


Figure 4.8: The auctioneer's book depicts the bid-ask spread, at time t. If a buy-order of a quantity enters, it is matched with the first seller(s) in the queue.

the continuous allocation typically converges to the competition equilibrium (Tan, 2007). In this equilibrium, the allocator is fully aware of the supply and demand curves of each participant, and can create economic efficient allocation. This convergence is not certain as the continuous model may also create large price fluctuations.

The main benefit of this approach, compared to the periodic allocation, is the immediate allocation, that is, once a match occurs, the resource transmission process starts. We agree with Tan (2007) that immediate allocation is fundamental for the flexibility of the Grid. We choose the continuous allocation method for the CVSE model and discuss shortcomings as well as other aspects of this model further in this chapter.

4.4.3 Analysis of CVSE

MACE's biggest problem is the **computational tractability**. This is, in CVSE, no longer an issue, as the continuous resource allocation problem could be solved in polynomial time. In fact, the worst case scenario would give a complexity of $O\left(n \lg(n)\right)$[5], where n is the problem size, as derived in appendix B. The most important thing, though, is that the problem is not NP-complete, as was the case with MACE. The auctioneer has a very simple problem to solve. Let us say that a buyer of resources has entered a bid. The auctioneer needs only to see if this bid is lower than the maximum of the asks, a_{max} and, if so, find the price and allocate the resources. Since CVSE does not need to solve an optimization problem, the allocation is solved momentarily each time an agent poses a bid or an ask. This immediate allocation is also an important factor for the **scheduling efficiency** requirement. By continuous matching of suppliers and demanders, the user receives the resources when he has the most use for them. He does not have to wait for an allocation period to be terminated, or for the auctioneer to announce the winner, in order to get the resources.

The agents' requirement of **economic efficiency** was not fulfilled in the MACE approach, nor is it not met in the CVSE market. Again, the different agents do not have the **incentive** to reveal their true valuation and reservation prices, as they may do better by bluffing. Numerous experimental and laboratory experiments have shown that, in many cases, the quantity and the price converge nonetheless towards the competition equilibrium. Still, there are no guarantees for convergence, and as high volatility in the prices may occur because of continuous trading, the price and quantity might not converge at all, causing inefficient allocation to take place (Tan, 2007).

In theory, the CVSE market can be **budget-balanced**, that is, that all payments from the agents are redistributed among the agents, cf. chapter 4.1.1.3. In the real world, however, the auctioneer and the other facilitators of the exchange must also somehow make money. A

---

[5] $O\left(n \lg(n)\right)$ the problem solving time does not increase with more than $\left(n \lg(n)\right)$, with increasing problem size n

weaker form of the budget-balanced property is more likely to be implemented in the Grid market, that is, that net payments are made from the agents to the mechanism, but no net payments go the other way (Schnizler, 2007).

The net payment to the auctioneer could conflict with the **individually rational** requirement. In order for the mechanism to be individually rational, the utility of participating for the agents should be equal or greater than not participating. If all participants have to contribute to the mechanism, then the non-winning participants will do better to refrain from participating. Instead of relying on the weaker form of the "individually rational" criterion, which states that the *expected utility* should be equal or greater than standing outside the mechanism, we solve the problem in a different way. We keep the participation in the CVSE free, but let the winners of the exchange pay transaction fees to the mechanism. This would reduce the winners' allocation surplus, but the individually rational criteria would presumably be fully satisfied. At the CVSE, only single attributed, homogeneous Virtual Servers are traded. The CVSE market alone does not meet the resource requirement of **heterogeneous resources** with **different attributes**. VSs merged with special resources, from the OTC market, create tailored resource bundles to fit the users' needs perfectly. It is important to state again that the special resource market only supplements the CVSE market. CVSE is much bigger, both in throughput and in market size, than the OTC market.

In the CVSE market, a customer is allowed to **co-allocate** VSs from many different providers, in order to create an even larger Virtual Server. The buyer is ensured that he only receives **one of the desired resource combinations**. Because the allocation is happening continuously, he knows immediately if he has been allocated resources. If the user also needs special resources, he would wisely buy these first, as he can reserve resources in the OTC. When he is certain about his allocation outcome, the buyer is free to buy the Virtual Servers on the CVSE.

A buyer can always buy Virtual Servers, given available capacity, by posing a high enough bid in the market, hence, the CVSE meets the requirement of **guaranteed usage time**. Both buyers and sellers can pose their bids and asks in CVSE, as is the requisite of a **double-sided mechanism**. The mechanism does not consider dynamic **network quality**, but relies only on static values, so we say that CVSE only partially fulfils the network quality requirement.

We believe that most of the requirements posed by the agents, resources and the other factors are mostly satisfied and that there are no major obstacles to creating the Continuous Virtual Server Exchange. The biggest problem, however, is that a continuous market might create huge price oscillations. These price dynamics are the topic of section 4.4.4 and we present a method to create predictability of future prices in chapter 5.

4.4.4 Price Dynamics

In a Grid market, there would be several factors driving the prices of the traded resources. Supply and demand are, of course, two factors determining the price, but there are also other forces affecting the prices. We separate between the market specific forces, due to the arrangement of the market, and external factors that only indirectly affects the prices.

4.4.4.1 Market Specific Forces

New agents entering the market may affect the prices in both directions. The number of suppliers may increase because different companies see that money can be made in the market. Macro economic theory says that as long as the prices are superior to the cost of producing the good, more and more companies will enter the market to offer their services.

This means that when the number of sellers increases, the prices decline. The opposite happens when there are new demanders entering, something that often occurs when the market is in the initial period. When companies notice that a Grid market offers flexibility, cost reduction etc, new buyers will enter. The big changes in the prices, due to internal factors, would probably happen because of **agents already in the market**. In a continuous market, there are always losers that have not been able to sell or buy computer resources. The "loosing" producers might consider reducing their ask price in order to sell resources, while the companies that have not been able to buy resources might consider increasing their bids.

4.4.4.2 External Factors

External factors are the price drivers that are outside the control of the agents in the market, affecting the agents' bid and ask strategies.

Increases in the **electrical power prices** would directly affect the producers, as the cost of computing would be higher. In fact, the price of a VS should, in the long term, be higher than the price of the electricity the VS uses, including both cooling and direct power consumption. Otherwise, the sellers would have a guaranteed loss on their VS sales.

Technological development would cause a price reduction of the VSs. Indeed, development would most certainly reduce the price of performing a same number of calculations. Such progress would lead to a revaluation of the specifications of the VS grades, and hence alter the prices of the different grades.

Macro economic factors could also affect the Virtual Server prices. Inflation, alteration in the foreign exchange, as well as changes in the nations' taxation policies would most certainly affect the Grid resource prices in either direction. A **destruction of companies** involved in trading may also be one direct external factor that would drive the prices in either direction. Natural or economical disaster and war are perhaps among the most devastating factors destroying companies, both on the supplying and the demanding side.

All these factors affecting the price lead to the conclusion that a computer resource market could be highly stochastic. Especially with the use of continuous market mechanisms, as CVSE is, the resulting prices could be very volatile. Price fluctuation creates problems for both sides of the agents in the market. The buyers do not have control over the possible cost incurred by computing in the future and the sellers cannot have long planning horizons for their income. Both sellers and buyers have financial risk and need a way to master this risk. In chapter 5, we present such a financial risk management instrument, namely derivatives.

Chapter Summary:

This chapter has treated the subject of creating a market on Grid resources. We looked at the potential agents in this market, both buyers and sellers, and provided their requirements towards the market mechanism. We also discussed the potential traded resources and other factors influencing this market. We found that a double-sided auction would be a good market mechanism and we looked at MACE and CVSE, two different exchanges. The properties of the MACE and CVSE are summarized below:

	CVSE	MACE
Resources	Standardized single-attributed VS	Multi-attributed resources
Allocation Mechanism	Single item	Combinatorial
Allocation Time	Immediate	Possibly long duration
Pricing Mechanism	buy & sell orders	k-pricing

As MACE has big problem with the processing needed for the allocation problem, we proposed the polynomial-time CVSE, a single item homogeneous continuous exchange. By analyzing the CVSE with respect to the mechanism requirements, we found that it satisfies most of them. CVSE's biggest problem, however, is the possibility of erratic and volatile price movements, both due to modeling and external factors. In chapter 5 we propose a derivatives market in order to create predictability.

Chapter 5

Derivatives Market

In this chapter, we propose Grid derivatives as a means of risk management. In addition to their ability to create a predictable future, derivatives are frequently used as instruments of speculation. First, we introduce the Grid swap. This instrument is dependent on the Virtual Server price process and creates the link between the spot market and the other derivatives. Second, we establish the theoretic framework for pricing and hedging the swap-based derivatives. Finally, we evaluate the viability of the derivatives market on Grid computing.

5.1 Grid Derivatives

We have now proposed a model for a Grid exchange market based on the concept of commoditization of computer resources. The traded asset is a Virtual Server (VS), X, with grade j, and price process S_t,

$$\begin{aligned} S_t : \text{price of the Virtual Server} X_j, \qquad & \text{at time } t \\ \text{with } j = \{\text{superior, normal, inferior}\}\,. \end{aligned} \tag{5.1}$$

As argued in chapter 4 "Exchange Market", the balancing of demand and supply, as well as the special physical attributes of computer resources such as non-storability, create uncertainty about the VS prices. This unpredictability represents a danger of price movements that negatively affect the agents. The spot prices, S_t, might depict erratic and extreme movements and thereby impose a significant risk if the agents trade solely at the Continuous Virtual Server Exchange, CVSE.

> *... risk by its nature has carried, and always will carry with it, the possibility of adverse outcomes. Accordingly, for globalization to continue to foster expanding living standards, risk must be managed ever more effectively as the century unfolds.* (Greenspan, 2002).

5.1.1 Financial Risk Management

Risk management includes activities dealing with recognition, assessment, management, and mitigation of risks. Strategies for risk management include transferring the risk to another party, avoiding the risk, reducing the negative effect of the risk, and accepting some or all of the consequences of a particular risk (Crockford, 1986). In our approach to a spot market, we introduce the CVSE for computer time and resource allocation. By doing so, we introduce a price risk because the agents do not know the future prices beforehand. To mitigate this risk and enable future reservation of the computer resources, we suggest a financial market where it is possible to buy and sell risks.

There have been hints of the use of financial risk management in computer sciences, see for example Smith, Foster, and Taylor (2000), Sulistio and Buyya (2004) and Darlington and Newhouse (2006), but to the best of our knowledge, no one has yet systematically treated the subject of international trading. Most of the scientific effort on computer risk management, or advanced future scheduling, has focused on management within a single company. We aim at building an international place to trade the financial risks connected with computer resource trading: a derivatives market.

5.1.1.1 Derivatives History

In finance, the agents manage risk by trading financial instruments or contracts, i.e. termed derivatives. The agents could vanquish all risk if they were able to forecast the future prices correctly. Such a precision forecast being highly unlikely, complete elimination of risk seems impossible. The solution is therefore to spread the risk. Lessons learned from financial markets suggest that financial derivatives, when well understood and properly utilized, are beneficial to the sharing and controlling of undesired risk. Uncontrolled exposure to market price risks could lead to devastating consequences for the Grid market participants, as was the case in the early electricity derivatives market. If the risks are properly dispersed, shocks to the overall economic system are better absorbed and less likely to create cascading failures that could threaten financial stability (Greenspan, 2002). In section 5.2, we introduce the Grid swap as a financial instrument for the agents to have complete knowledge of the price to pay or receive for the future Virtual Server use.

Economic risk has been managed using derivatives contracts for thousands of years, evolving from securing harvest payments described in the Code of Hammurabi in 1800 BC (Embrecht, 2003), to the Dutch tulip trade in the 1600 (MacKay, 1841), up to the establishment of the world's first future exchange, the Chicago Board of Trade[1], in 1848. The foundation of the Chicago Board of Exchange[2] coinciding with the publishing of the Nobel Prize winning fundamental pricing model of Black and Scholes, and Merton in 1973, the derivatives trade assumed its current form.

Both financial and commodity markets use derivatives as a key function for risk mitigation and transfer. Some commodity markets show that derivatives consist of the main part of the trade on the total market. Even in mature markets, the transactional value on the derivatives market is generally several times that of the spot market (Beaver, 2007). There has been an explosive growth in the use of financial derivatives, mainly due to more volatile markets, deregulations, and introduction of new technologies (Siems, 1997). This enormous trading on derivatives markets makes it interesting to propose such market on Grid resources.

5.1.1.2 What is Derivatives

Definition 9 *A **derivative** is an instrument whose price depends on, or is derived from, the price of another asset (Hull, 2005).*

We use Virtual Servers, treated in section 5.2, as the underlying assets for Grid swaps. Because of pricing issues, we let these Grid swaps serve as underlying for other derivatives in section 5.3. While the agents use VSs for trading of computer power, that is, one party

---

[1] www.cbot.com/

[2] www.cboe.com/

produce the VS and the other buys and uses it, derivatives are private contracts between two parties, where neither has to use or produce computer power. The sum of gains and losses on derivatives contracts must be zero; for any gain made by one party, the other party must have suffered a loss of equal magnitude (Jorion, 2005). The party agreeing to buy the underlying asset assumes a *long* position and the other takes a *short* position.

5.1.2 Who wants to trade risk

Participants in the derivatives market have different motivations for entering, and taking sides, in a trade. Primary participants, the ones who trade contracts directly in the derivatives market, consist of risk mitigators, speculators, and arbitrageurs. The market participants are not limited to one role, and a risk minimizer one day may become a speculator the next. The secondary participants, consisting of clearinghouses, and market makers, ensure efficient trading. Figure 5.1 depict the market participants.

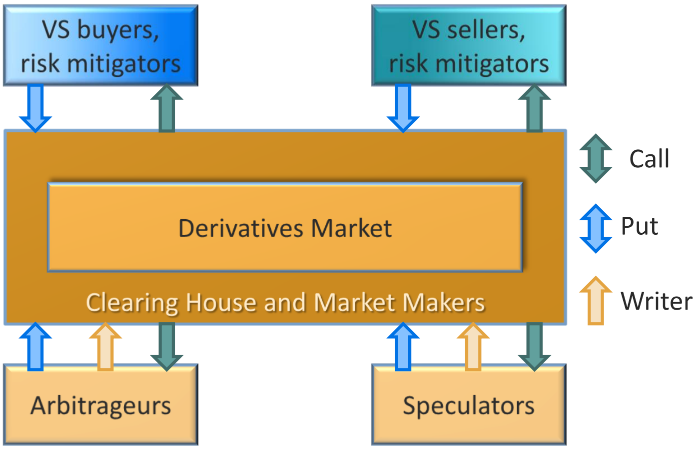


Figure 5.1: The derivatives market consists of risk mitigators; both buyers and sellers of the VSs; speculators; arbitrageurs; and secondary market participants such as market makers and clearinghouses.

5.1.2.1 Primary Participants

Risk mitigators are parties who seek to reduce the price risk related to the trade of the VSs, by entering into a derivatives transaction. They are, in general, organizations or individuals who trade derivatives to establish deterministic price levels, at some future time, for the VSs they intend to buy or sell on the CVSE. By using derivatives, they try to protect themselves against unfavorable price changes during a certain period (Association, 2006). One example of risk reducers in the Grid derivatives market is companies that offer computer resources to their customers. They might buy many computer resources on the exchange, partition them into smaller units, only to sell them to individual customers on subscription. These firms would then be subjected to risk since they have promised their customers the servers for a fixed sum while they themselves buy it for variable costs. As the price of the VS varies, the companies may have incentives to reduce their price risks using Grid derivatives. The risk minimizers are thus participants who would like to avoid price risk by fixing a certain cap on their future asset price. In fact, each buyer and seller of computer resources on the CVSE is a potential risk mitigator of the derivatives market.

Speculators position themselves in derivative contracts in anticipation of favorable future market movements of the VS prices. They have an opinion about the direction of the underlying prices that differs from that of the majority, and are willing to take on risk for a suitable compensation (Deng & Oren, 2006). Speculators may be traders who aim to generate profit, essentially by placing a bet on the movement of an asset. They are therefore willing to take the other side of a trade in return for higher expected profits (Beaver, 2007).

Arbitrageurs try to profit from inconsistencies in the pricing of an asset, such as the VS. They develop models to price various traded assets, and when they find price discrepancies, combinations of derivatives are used to make riskless profits. Arbitrageurs contribute to spread information in the market, thereby closing "loopholes".

5.1.2.2 Secondary Participants

Participants in a derivatives market are subject to numerous risks, such as counterparty risk, legal risk, and administrative risks. A **clearinghouse**, a neutral third party, becomes the counterpart to both the seller and the buyer in all transactions in the derivatives market and hereby reduces these risks (Beaver, 2007). It takes on the counterparty risk and frees the market participants from having to trust the opposite party. Beaver argues that a clearing function most certainly increases the trading volumes of derivatives as some market participants may be unwilling to take the above-mentioned risks.

The **market makers** ensure fast execution of buy and sell orders, and thereby provide the derivatives market with liquidity (Hull, 2005). A market maker for a given contract is someone who, when asked to do so, will quote both a bid and an ask price on the derivatives. The bid is the price at which the market maker is prepared to buy, and the ask is the price at which the market maker is prepared to sell. On a trader’s request, the market maker quotes bid and offer prices without knowing whether the trader wants to buy or sell the contract, and will give, of course, a higher ask than bid. This difference, called the bid-ask spread, is how the market maker makes money.

5.1.3 Contract Specifications

We discuss briefly two of the important aspects to have in mind when specifying the derivatives contract for a Grid market. These factors consider the trade-off between standardization and customization, as well as the importance of cash settlement of the derivatives. The reason why we do not provide clear guidelines for contract specifications is that we believe it is important to listen to the future users of the contract in order to put up rigid specification. We propose, where appropriate, further specifications for the derivatives.

5.1.3.1 Standardization

Standardization of contracts is important in order to maximize market liquidity, but the contracts need to be set up in the right way so as not to lead to inflexibility that does not satisfy any trading needs. There are several aspects that should be standardized for each derivative, for example the contract length, size, pay-off, and so on. In order to have a free derivatives market with the possibility to trade existing contracts, standardization of contracts is paramount.

5.1.3.2 Cash Settlement

In the CVSE, the sellers deliver the VSs to the buyers, that is, the spot market has delivery settlement. We can either have physical delivery or cash settlement in a derivatives market. For financial participants, such as speculators and arbitrageurs, delivery settled contracts are less appealing because there is a risk of having to take or make delivery of a physical commodity. This risk is especially pronounced if the market is illiquid and the writer is not able to find a counterpart to annul her positions. Cash settled derivatives markets have, therefore, a potentially higher level of liquidity than a delivery settled markets. Cash settlement seems to be the most attractive solution for Grid derivatives based on the physical asset. We now introduce a standardized, cash settled derivative, the Grid swaps, which is the only Grid derivative based solely on the spot market.

5.2 Grid Swaps

The swap is the first of the derivatives contract we will look at. In addition to be a good risk management tool in itself, we intend the swap to be the underlying asset for other Grid derivatives. First, we will discuss what a swap contract really is and what its purpose is. Afterwards, we discuss how to price the swap contracts and we find a mathematical expression for this price process.

5.2.1 What is a Swap Contract?

The asset in the Grid market is Virtual Servers, a non-storable consumption commodity. Agents use swaps in order to have a planning horizon and thereby reduce risk coupled to movements in the VS prices. A swap is an agreement to exchange cash flows in the future according to a prearranged formula (Hull, 2005).

Definition 10 *A **Grid swap** is a derivatives contract that specifies the obligation of one agent to exchange a fixed price with the Virtual Server price process provided by opposite party over a specified period in the future.*

The future aspect of the swap is in contrast to the spot agreement, which allows for buying or selling an asset for a period starting immediately. The exchange of cash flow happens over the delivery period, beginning at the future point T_b and ends at T_e. The party buying the VSs at a fixed price enters a long position, while the seller takes a short position. We let F_τ be the fixed price of a swap an arbitrary contract at time τ and $\frac{1}{T_e-T_b}\int_{T_b}^{T_e} S_u e^{T_e-u}du$, or $\bar{S}_{T_e-T_b}$ for short, be the average VS price during the delivery period forward priced to time T_e. Figure 5.2 illustrates one VS price process with the fixed sum F_τ and the variable cash flow $\bar{S}_{T_e-T_b}$.

The contract value of the long position at this time is:

$$\mathcal{X}_{swap}^{long} = (T_e - T_b)\left[\bar{S}_{T_e-T_b} - F_\tau e^{rT_e}\right], \tag{5.2}$$

And the short position

$$\mathcal{X}_{swap}^{short} = (T_e - T_b)\left[F_\tau e^{rT_e} - \bar{S}_{T_e-T_b}\right] \tag{5.3}$$

where r is the constant continuous risk-free interest rate. If the two swaps, long and short, have the same contract specifications and were entered into at the same time τ, then the pay-offs to the different parties are exactly opposite, see figure 5.3.

5.2.1.1 Risk Management Abilities

Both sellers and buyers of the swap can use the contract to increase the predictability for the future. Someone, who needs the underlying Virtual Server, may buy a swap contract at

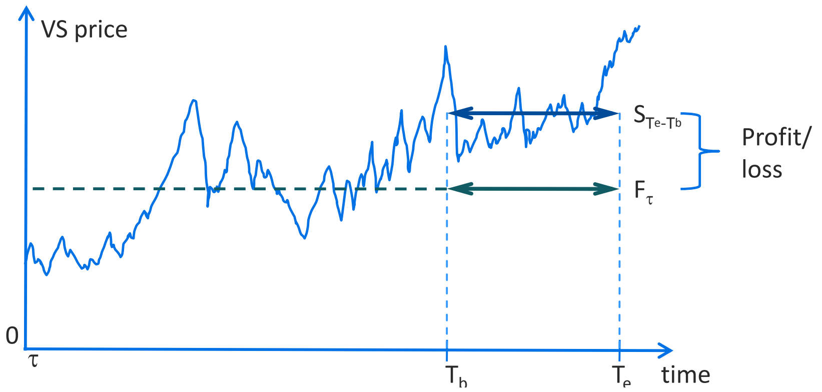


Figure 5.2: A price process of a Virtual Server showing the delivery period $[T_b, T_e]$ and the profit or loss when the fixed amount F_τ is exchanged for the variable amount $\bar{S}_{T_e - T_b}$. In this example, the party who bought the swap would have made a profit, while the seller would realize a loss.

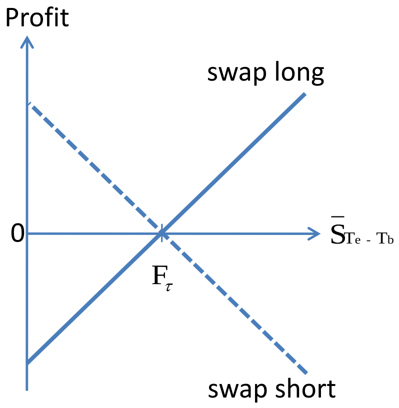


Figure 5.3: The payoff of the long and short positions in a Grid swap. When entered into at the same time, the pay-offs are exactly opposite.

time τ. The buyer has to pay the fixed price, F_τ, for the use of each hour during the delivery period of the swap, $[T_b - T_e]$. As mentioned, the swap should have cash settlement, that is, no Virtual Server changes hands due to the contract. The user could then buy computer power during the period, as he planned to do, and have the floating price remunerated by the seller.

Example 1 *A company goes, because of budgetary conditions, long* $10,000$ *swap contracts with delivery over a period of* 7 *days, or* 168 *hours, in three months. Today's swap price is* 100 *Norwegian cents per hour. The company buys the right to have the fixed flows from the CVSE over delivery for a total of* NOK 1.7 *million* $(10,000\ swaps \cdot 168h \cdot 100cents/h)$. *At time* T_e, *the* $\bar{S}_{T_e - T_b}$ *turns out to be* 90 $cents$, *and the opponent(s) pay(s)* NOK 1.5 *million to cover the company's outlays during the delivery. Even though the contracts turned out to lose money,* NOK $84k$ *when ignoring interests, they did exactly what they were supposed to: manage risk by creating deterministic prices for the Virtual Servers.*

The seller of the swap, let us say a computer farm, may have a similar motivation for wanting a predictable future. By taking a short position in the swap, the center receives the deterministic amount F_τ per delivery hour at time τ instead of the floating VS price. It could sell the VSs on the CVSE during the delivery period, receiving floating revenue, and then exchange it with the fixed amount. The computer farm would then create predictable incomes for a future period.

5.2.1.2 Fixed Price Evaluation

The question is how to determine a fair value on F_τ, when standing at time τ. At this time, the participants should agree upon the price that the buyers of the Grid swap are willing to pay, and the price the sellers are willing to accept for the future stochastic process S_t so that the expected value of the swap, at the time of the contract writing, is zero. If the prices S_t had been deterministic, we would not have any problems finding the F_τ. The price would simply have been the present value of the cash flow over the delivery period. Unfortunately, it is impossible to know the future values of the S_t. The normal approach of arbitrage pricing of derivatives is not applicable either.

Definition 11 *An **arbitrage** is a trading strategy that takes advantage of two or more assets being mispriced relatively to each other (Hull, 2005).*

That the arbitrage pricing, meaning the swap price, should be equal to today's spot price adjusted for some interest rate requires the possibility of *cash-and-carry*. This requirement terms that it should be possible to buy the underlying asset, hold it for a period and then sell it again in the market. If the *cash-and-carry* argument holds, possible arbitrages between the swap and the underlying would be exploited as soon as they appear, and the spot and derivatives contract should be correlated.

The Virtual Servers in the Continuous VS Exchange are, however, non-storable, and cannot be bought and held. Without this relationship, the spot price, S_t, only reflects the current state of demand and supply, and is independent of the Grid swap price process F_t (Skantze & Ilic, 2000). This result makes the arbitrage-pricing framework, termed the First Fundamental Theorem of Asset Pricing, unusable for finding a fair value.

As we are not able to price the Grid swap from the spot price using the no-arbitrage framework, how do we price these swaps and how do we proceed to find the swap price process? Inspired by the deregulated electricity market and other non-storable commodities with similar properties, we choose to let the market set the Grid swap prices by an auction. Then, the swap prices reflect the expected future value of the VSs.

5.2.2 Price Determination and Information

Before we look at the determination of the swap prices, let us introduce the notion of a filtered probability space, in order to model the derivatives market fully. The filtered probability space $(\Omega, \mathcal{F}, \mathbb{P}, \underline{\mathcal{F}})$, consists of the sample space, the minimal σ-algebra containing the open set on the sample space, the probability measure and the filtration of the σ-algebra. The sample space, Ω, is the set of all possible outcomes and one such set is an event and $\mathbb{P}$ is the probability measure of such event. The filtration $\underline{\mathcal{F}}$ is generated by $\mathcal{F}_t$. The interpretation of $\mathcal{F}_t$ might be the market information of all events happened up to time t (Embrecht, 2003).

5.2.2.1 Price Determination by Auction

We choose to model the Grid swap price F_t using an auction similar to the one in the CVSE. The goal of this double-sided auction is firstly to determine the fair price F_t of the swap at each point in time t.

Let the auction open with a discrete allocation of the swaps, say three month before the beginning of the delivery period. That is, the auction is open for a longer period, say a couple of hours, before the initial contract amount and price, F_0, are fixed. Afterwards, the auction could be a continuous exchange, as described in section 4.4.2.1. The market, with supply and demand of swaps, should set all the prices F_t according to its aggregated belief on the spot prices over delivery, and are globally set as,

$$F_t = \mathbb{E}\left[e^{-rT_e}\bar{S}_{T_e-T_b}|\mathcal{F}_t\right]. \tag{5.4}$$

$\mathbb{E}$ is the expected value given the filtration $\mathcal{F}_t$.

At the entry time, say τ, the expected value is equal F_τ and the value of the swap contract is zero. If the expectations about the spot prices over the delivery period are true, the F_τ will be exactly offset by the average spot price. As time goes by, new information enters which may increase or decrease the expectations of future spot prices. This expectation affects the prices of the newly signed contracts. In this way, the information at each point in time, t, gives the prices of the swaps, F_t. Equivalently, the prices reflect the information available at that time.

Example 2 *This next example features a computer center that likes to have a planning horizon of several months. They participate in the auction and get the knockdown on selling* $10,000$ *Grid swaps, each on one VS during a seven-day delivery period in three months time. The current price of the swap, say* $F_0 = 100$ *Norwegian cents, reflects the market's expectation that the average future spot prices amount to* 100 *cents an hour, over the delivery period. The value of the contract is hence nil. The next day, information about several new computer Grid farms opening in two months enters the market. Even though this news does not affect the spot market, because of the non-storability condition, the swap prices decrease as it is expected that these farms will push the prices down for the delivery period. Let us assume that the prices went down to* 90 *Norwegian cents. If the computer center would get rid of their contracts, all they have to do is go long* $10,000$ *contracts in order to neutralize the short position and realize the net loss of* $NOK\ 167k$ $((\,90 -\ 100)$*cents*$/h \cdot 168h \cdot 10,000$ *contracts*$)$.

We were, perhaps, a bit hasty when we established equation (5.4) and said that the prices reflect the information available. This equation is in fact due to the Rational Expectation and Efficient Market Hypotheses. In addition, these hypotheses help us on the way of forming a mathematical expression of the price process of the swap, which is helpful when we build other derivatives on the swaps.

5.2.2.2 Rational Expectancy and Efficient Market Hypotheses

Muth (1961) introduces the *Rational Expectancy Hypothesis* (REH), explaining how the agents anticipate future prices, to reach a systematic theory of fluctuations in an economy. He predicts that a market does not waste information and assumes that actors will in fact consider all available information, $\mathcal{F}_t$, while forming the expectations about future VS prices. Even though the expectations may not turn out correct, they will not divert systematically from the expected values. This deviation thus represents an unsystematic error of information ignorance and mistakes.

Nobel Prize laureate Lucas and Prescott (1971) develops the theory of Muth, and determines the competitive equilibrium by assuming that the actual and the anticipated prices have the same probability distribution, meaning that the price expectations are rational. In order to have fairness, the swaps should be priced according to the expectation of the S_t

during the delivery period. The *Efficient Market Hypothesis* (EMH) is a dynamic extension of the REH in the sense that it explains how the equilibrium results are maintained with changing information. "The Efficient Market Hypothesis is in essence an extension of the zero profit competitive equilibrium condition from the certainty world of classical price theory to the dynamic behavior of prices in speculative markets under conditions of uncertainty." (Jensen, 1978).

According to Friedman and Rust (1993) and Tan (2007), the continuous double auction model converges to the competitive equilibrium, resulting in highly efficient allocations. In the competitive equilibrium, the balance between offer and demand results in the market price of the swap. The auction forms the swap price based on all available information and expectations pertaining to what could possibly influence the spot price, already reflecting daily load curves, seasonality, etc.

The EMH assumes that the conditions of the market equilibrium can be stated in terms of expected returns, and that equilibrium returns are formed on the basis of, and fully reflects, the information set $\mathcal{F}_t$. This implies that trading systems based on information in $\mathcal{F}_t$ cannot have expected profits or returns in excess of the equilibrium expected profits or returns. This is the basis of the "fair game" efficient market models (Fama, 1970).

A market is efficient with respect to the information set $\mathcal{F}_t$ if it is impossible to make economic profit by trading on the basis of the information set $\mathcal{F}_t$ (Jensen, 1978). The basic idea is that competition drives all the information into the price quickly, or as Samuelson (1965a) says: "If one could be sure that a price will rise, it would have already risen." This property is formalized as absence of arbitrage.

5.2.2.3 Absence of Arbitrage

An arbitrage is defined in definition 11, and is a trading strategy that allows an investor to make a sure profit without taking any risk.

The meaning of absence of arbitrage (AOA) is not that arbitrage does not exist, but when the investors, or arbitrageurs, find an arbitrage possibility, it disappears almost immediately. The Efficient Market Hypothesis by Fama (1965, 1970) is concerned with whether or not prices at any point in time "fully reflect" the available information. Arrival of new information causes imperfection in the market, but every such imperfection is immediately arbitraged away (Mandelbrot, 1971). This happens if there is a large number of participants who trade and provide sufficient liquidity to the market. The economy is then in an arbitrage equilibrium, that is, there exists no self-financing trading strategy with no investment in $t = 0$ which gives, with probability 1, a value greater than zero at a future point in time, $t > 0$. We say that the swap market is a market satisfying AOA and we use this property in chapter 5.3 to price other derivatives on the swaps. First, let us describe the swap price process.

5.2.3 The Swap Price Process

We now establish a mathematical model, describing the price process of the Grid swap, in order to gain more insight into the swaps. A mathematical model also has a second motivation, which is that we will then be able to price other derivatives on this instrument and perform simulations of the swaps. The first step we take to reach a model is to identify that the process follows a random walk and holds the Markov property. Second, we assume normal distributed returns and let the process follow a Brownian motion. Finally, by allowing only positive prices, we obtain that the swap prices follow a geometric Brownian motion.

5.2.3.1 Markov Process

The random walk model makes a more detailed statement about the economic environment, and is an extension of the "fair game" efficient markets model. The "fair game" model states only that the conditions for market equilibrium are in terms of expected returns, but say little of the stochastic process generating them. The theory of random walks in asset prices involves two hypotheses: first, that successive price changes are independent and second, that price changes conform to some probability distribution. That is, the price differences are independent, identically distributed variables (Fama, 1965).

Both of these hypotheses are quite general and should be applicable to our market. They imply that the series of price changes has no memory and that the past cannot be used to predict the future in a meaningful way. This stochastic process is in fact a Markov process, where the entire distribution of the future prices relies on the current price only. The history of the variable, as the way in which the present has emerged from the past, are irrelevant, and only the price is now of any interest. We assume that price process of the swaps follow a Markov process.

Definition 12 *A* ***Markov process*** *is a stochastic process whereby the behavior of the variable over a short period of time depends solely on the value of the variable at the beginning of the period, not on its past history (Hull, 2005).*

Using the Markov process, we are able to model the dynamics of the Grid swap. During a small subsequent time interval dt, the price of the Grid swap changes to $F_t + dF_t$, where dF_t is a continuous infinitesimal change. We decompose this change in two parts. One is a predictable and deterministic part which gives the average growth rate, called the drift, μdt. The second contribution, the disturbance, $\sigma d\bar{W}_t$, models the random change in the Grid swap price. This noise is a continuous, stationary, stochastic process $\bar{W}$ multiplied with an amplifying factor, the volatility σ. From the drift and disturbance, we obtain the Stochastic Differential Equation (SDE)

$$\begin{aligned} &dF_t = \mu_t dt + \sigma_t d\bar{W}_t, \qquad t \in [0, T]\,, \text{with initial value} \\ &F_0 = f. \end{aligned} \tag{5.5}$$

5.2.3.2 Wiener Process and Brownian Motion

Within finance circles, discussion about what type of distribution fits the price changes has created a huge number of articles (Fama, 1970). Observing the stock market, Bachelier (1900) proposes a model implying normal distributed prices changes. He assumes that price changes from transaction to transaction are independent, identically distributed random variables with finite variances. He says that if transactions are uniformly distributed across time, and if the number of transactions per period is very large, then the Central Limit Theorem leads us to expect that the price changes will have a normal distribution. Mandelbrot (1963) raises serious doubts about the use of a normal distribution. According to Mandelbrot, Wesley Clair Mitchell offers, as early as in 1915, empirical evidence showing that asset returns, or price changes, do not follow a normal distribution, but rather a leptokurtic distribution. This "fat-tailed" distribution has been repeatedly observed in financial markets, among others by Fama (1965) and Carr, Geman, Madan, and Yor (2002), and is, today, an accepted fact. Much work has been done to propose alternative distributions, such as the stable distribution (Fama, 1965), hyperbolic distributions (Carr et al., 2002), which are often tested on and fitted to empirical market data (Burger, Klar, Müller, & Schindlmayr, 2004).

The normal distribution does not give the best fit in empirical tests on market data, but since no Grid market data yet exists to test possible distributions, we assume the applicability of the normal distribution. Using normally distributed returns in financial markets gives good results most of the time, except when extreme market conditions take place (Embrecht, 2003). We assume that the Grid swap prices follow a Markov process and have normally distributed price changes, implying that the random change $d\bar{W}$ in equation (5.5) becomes a Wiener process and the SDE in (5.5) becomes a Brownian motion.

Definition 13 *A standard* ***Wiener process*** $\bar{W}_t$ *is a stochastic process of continuous trajectory, almost surely, with*

- $W_0 = 0$
- $\forall t, s \geq 0, W_{t+s} - W_s$ *is independent of the* $\mathcal{F}_s$ *and is normally distributed* $\mathcal{N}(0, t)$ *(Lamberton & Lapeyre, 2000).*

Let $\bar{W}$ be a standard Wiener process, derived on a filtered probability space $(\Omega, \mathcal{F}, \mathbb{P}, \underline{\mathcal{F}})$. Now, let $\underline{\mathcal{F}}$ be generated by the Wiener process $\mathcal{F}_t^{\bar{W}}$. $\mathcal{F}_t^{\bar{W}}$ is the increasing subset of the σ-algebra $\mathcal{F}$ generated by $\bar{W}$ under the physical probability measure $\mathbb{P}$.

One practical problem is that prices following a Brownian motion can become negative. We solve this issue by the example of Samuelson (1965b), formulating normally distributed relative price changes $\frac{dF}{F}$. The prices then become lognormal and the SDE in (5.5) becomes a geometric Brownian motion[3] (GBM):

$$\begin{aligned} \frac{dF_t}{F_t} &= \mu F_t dt + \sigma F_t d\bar{W}_t, \qquad t \in [0, T]\text{, with initial value} \\ F_0 &= f. \end{aligned} \tag{5.6}$$

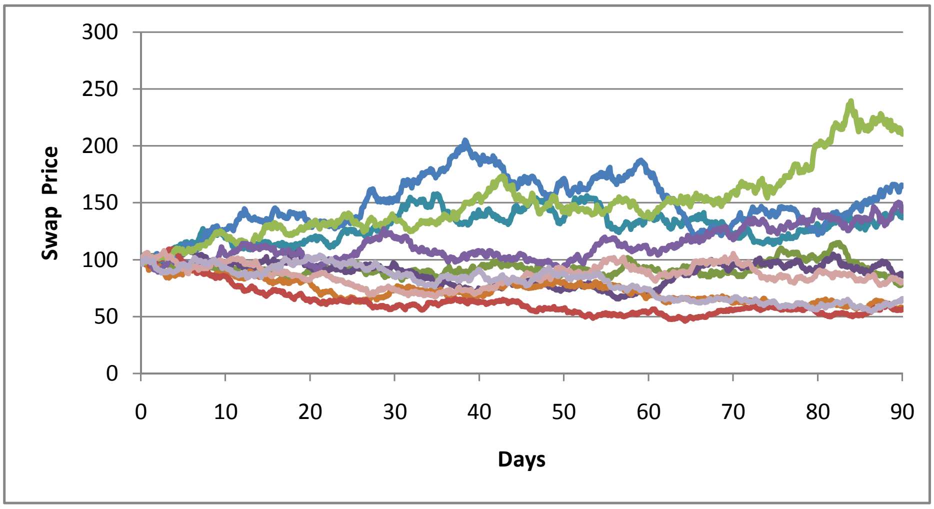


Figure 5.4: Simulation of ten different stochastic processes, with start value 100 Norwegian cents over a period of 90 days and a volatility of 80%.

We have now found a mathematical representation of the price process of the Grid swap F_t on the Virtual Servers, and we are able to simulate Grid swap price processes as depicted

---

[3] The geometric Brownian motion says that the distance travelled by a particle is proportional to the square root of time (Weron, 2000).

in figure 5.4[4] . These swaps are financial, storable assets, which we can use as underlying to value derivatives within the no-arbitrage framework. The derivatives that are written on the Grid swap are the focus of section 5.3.

5.3 Financial Instruments on Grid Swaps

We have looked at spreading the financial risk in the Grid exchange market using derivatives, and we have discussed the participants willing to take the opposite sides of these contracts. The issue of non-storability of the Grid spot contract was solved by creating a storable asset in the form of the Grid swap. We established the price process of the Grid swap, which enables us to construct all kinds of derivatives using the Grid swap as underlying, and value them. Now, we look closer at derivatives written on the Grid swap. Such a claim will in general have the form

$$\mathcal{X}_{derivative} = \Phi(F_t), \tag{5.7}$$

that is, it is defined by the pay-off function Φ on the price process of the swap F_t. In order for the market participants to buy and sell these contracts, they must be able to price them. In the following, we will show how to value any claim $\mathcal{X}_{derivative}$.

5.3.1 General Pricing and Hedging of Derivatives Written on Swaps

Let us consider a filtered probability space $(\Omega, \mathcal{F}, \mathbb{P}, \underline{\mathcal{F}})$ and let the filtration $\underline{\mathcal{F}}$ be defined by $\mathcal{F}_t = \mathcal{F}_t^{\bar{W}}$. Our Grid market consist of n risky swaps $F^1, \cdots, F^n$, which differs in some way for example in the length of the delivery period or by the expiration date. The F^i-dynamics under $\mathbb{P}$ are given by

$$\begin{aligned} dF_t^i &= \mu_t^i F_t^i dt + F_t^i \sum_{j=1}^{k} \sigma_t^{ij} F_t d\bar{W}_t^j, \qquad t \in [0, T_e], \\ F_0^i &= f^i \end{aligned} \tag{5.8}$$

where f^i denotes the value of the swap i at time $t = 0$, $\bar{W^j}$ is a k-dimensional standard Wiener process, and μ^i and σ^{ij} are adopted to the filtration representing the drift and diffusion respectively. We assume the existence of a risk free bond,

$$\begin{aligned} dB_t &= r_t B_t dt, \\ B_0 &= 1, \end{aligned} \tag{5.9}$$

where the rate of return r_t may vary over time. While the value of the bond enjoys an exponential growth at rate r_t, the Grid swaps fluctuate randomly.

The applicability of absence of arbitrage, which we introduced through the Efficient Market Hypothesis in section 5.2.2.2, implies that there exists one or more Equivalent Martingale Measures (EMMs), that is the set of EMM is non-empty, $\mathbb{Q}^* \neq \{\emptyset\}$. $\mathbb{Q}^*$ is the set of $\mathbb{Q}, \mathbb{Q} \in \mathbb{Q}^*$, $\mathbb{Q}$ and $\mathbb{P}$ are equivalent measures $\mathbb{Q} \Leftrightarrow \mathbb{P}$, i.e. $\mathbb{P}(A) = 1 \Leftrightarrow \mathbb{Q}(A) = 1$, for every $A \in \mathcal{F}$ (Björk, 2004).

Under $\mathbb{Q}$, the normalized processes $\tilde{F}_t^i = \frac{F_t^i}{B_t}$ are martingales,

$$\tilde{F}_t^i = \mathbb{E}^{\mathbb{Q}}\left[\tilde{F}_s^i | \mathcal{F}_t\right] \text{for} s \geq t. \tag{5.10}$$

---

[4] We performed these simulations on a Virtual Machine, using a version of the C++ program given in Appendix D

By applying Grisanov's theorem we transform the process from the physical probability measure $\mathbb{P}$ to a risk-adjusted measure $\mathbb{Q}$. The deterministic part of the F-dynamics is transformed from μ_t to the interest rate r_t, while the diffusion term remains unchanged. The F^i-dynamics in (5.8) are thus under a EMM $\mathbb{Q}$

$$\begin{aligned} dF_t^i &= r_t F_t^i dt + F_t^i \sum_{j=1}^{k} \sigma_t^{ij} F_t dW_t^j, \qquad t \in [0, T_e], \\ F_0^i &= f^i. \end{aligned} \tag{5.11}$$

This EMM is not necessarily unique, meaning that we are not sure to find a unique price of the derivative. For it to be truly distinct, the Grid market needs to be complete.

5.3.1.1 Completeness

The Grid security market is complete if we are able to replicate each cash flow from the claims at expiry, only by investing the premium of the derivative, Π, in some portfolio consisting of swaps, bond and/or other securities. In a complete market, we are able to hedge each claim, as well as to find a unique price for the claim.

According to Björk (2004), a market based on the described model is complete if, and only if, the number of Wiener processes are equal to the number of risky swaps, $k = n$, and if the volatility matrix σ_t is invertible *P-a.s.*[5].

Harrison and Pliska (1981) provide in their theorem a proof that in a complete, arbitrage free market, there must exist a unique martingale measure:

Theorem 1 *The second fundamental theorem states that the following statements are equivalent:*

- *The model is complete under some probability measure.*
- *The set of probability measures on the $(\Omega, \mathcal{F})$, $\mathbb{Q}^*$, is a singleton, meaning the EMM $\mathbb{Q}$ is unique.*

The question is whether we have a complete market. In order to provide a valuation and hedging framework, we restrict ourselves to a contingent claim on only one swap. Since we only have one risky asset and one driving Wiener process, $n = k$, the market is complete. In accordance with theorem 1, there exists a unique martingale measure $\mathbb{Q}$ and the claim $\mathcal{X}_{derivative}$ has a unique valuation.

5.3.1.2 Valuation

One of the issues the seller, often called the *writer*, of a financial instrument faces, is the valuation of the claim. The claim may be exercised at time T, while the price which the buyer has to pay to the seller is set at a prior date, let us say at $t = 0$. The problem is then to find a fair price Π_t of the claim $\mathcal{X}$ at the time t, at which the agreement is made.

We value the financial instrument consistently with the underlying swaps, using martingale pricing. This means that we construct a martingale under the EMM $\mathbb{Q}$ by normalizing the prices of the claim: $\tilde{\Pi} = \frac{\Pi}{B}$. At T, the price and the pay-off are equal, $\Pi_T(T, \Phi(F_T)) = \Phi(F_T)$. If the model is complete, we obtain the general valuation formula (5.12):

$$\tilde{\Pi}_t(T, \Phi(F_T)) = \mathbb{E}^{\mathbb{Q}}\left[\tilde{\Pi}_T(T, \Phi(F_T)) | \mathcal{F}_t\right]$$

---

[5] Probability-almost surely

$$\Pi_t(T, \Phi(F_T)) = B_t \cdot \mathbb{E}^{\mathbb{Q}} \left[\frac{\Phi(F_T)}{B_T} | \mathcal{F}_t \right]^6 \tag{5.12}$$

5.3.1.3 Hedging

After selling a derivative, the problem arises to know how the seller should cover or *hedge* his positions. When selling the instrument at time t, he does not know what the Grid swap prices will resemble in the future. If the derivatives writer does not hedge his position, taking what is known as a *naked position*, he will be exposed to a price risk. Hedging consists of taking positions in the swap market, the bond and/or other financial instruments, to lower the risk of disadvantageous swap price movements. The valuation and hedging problems are therefore closely related. The fact that we obtained a unique expression for the valuation, equation (5.12) of the claim, is a proof that it is possible, at least in theory, to hedge perfectly the derivative.

Pricing of the Grid derivatives is sensible to many factors, concerning both the underlying swap and the pricing model itself. These sensitivities are termed *the Greeks*, and are meant to help the writer handle these risks by identifying how the derivative price is affected by changes in the underlying, interest rate, the expiration time or the variance. By constructing a portfolio that is insensitive to small changes in one of these factors, the portfolio becomes neutral, corresponding to *the Greek* being zero (Björk, 2004). Let V denote the value of the portfolio, the different *Greeks* are the partial derivatives: Delta: $\Delta = \partial V / \partial F$, Gamma: $\Gamma = \partial^2 V / \partial F^2$, Rho: $\rho = \partial V / \partial r$, Theta: $\Theta = \partial V / \partial t$, Vega: $Vega = \partial V / \partial \sigma$.

If we let h^1 be the units of the risky swap and a h^0 denote the position in the riskless bonds, we may make a portfolio V_t delta neutral in the short period $[t + \Delta t]$ by letting:

$$\begin{aligned} h^1_{t+\Delta t} &= \Delta \text{ and} \\ h^0_{t+\Delta t} &= V_t - h^1_{t+\Delta t} F_t. \end{aligned} \tag{5.13}$$

$h^1_{t+\Delta t}$ and $h^0_{t+\Delta t}$ are $\mathcal{F}_t$ measurable positions, held until the next hedging of the portfolio. Such a neutrality of the portfolio last only for a short period of time, so in order to remain hedged against unwanted changes, it is necessary to rebalance the hedge. This is called a *dynamic hedge*, and is contrary to a static hedge which is set up only when the derivative is sold, not to be adjusted at any later point in time Hull (2005).

While the delta is the change in the portfolio price relative to the swap value, gamma is the sensitivity of value with respect to changes in delta. If the gamma is large, delta changes rapidly, and frequent changes in the portfolio must be done to keep it delta neutral. Rho is the portfolio sensitivity with respect to changes in the interest rate, Theta with respect to the time, and Vega hedging protects against variable volatility of the swap.

Another type of hedging strategy is using other derivatives as hedging instruments. By buying a derivative with similar properties to the one sold, the different sensitivity risks are effectively mitigated. Static hedges are often hedges with the use of other derivatives.

We are now able to value and hedge different types of derivatives on Grid swaps. We give a throughout example of how to price and hedge one type of financial instrument, options on swaps.

5.3.2 Options on Swaps

An option on a swap is a financial instrument giving one party, the holder, the right to buy or sell the underlying swap, at a future time, at an agreed price, the strike K. Call options

---

[6] With intermediary calculations: $\tilde{\Pi}_t(T, \Phi(F_T)) = \mathbb{E}^{\mathbb{Q}} \left[\frac{\Pi_T(T, \Phi(F_T))}{B_T} | \mathcal{F}_t \right] = \mathbb{E}^{\mathbb{Q}} \left[\frac{\Phi(F_T)}{B_T} | \mathcal{F}_t \right]$

gives the right to buy the underlying, while a put option gives the right to sell the swap. An option will only be exercised if it generates profit. An option is said to be *in-the-money* if exercising the option would give a positive pay-off. If the pay-off is zero, than the option is *out-of-the-money*.

The only downside the holder has is the price he paid for the option. At contracting, the writer receives the option price or a *premium*, Π, as payment for the opportunities the holder gets. The writer, however, has limited benefit, only the premium, and a possibly unbound downside.

Depending on whether or not the option can be exercised during a period or at a specific point in time, the options are classified as American and European, respectively. The last date at which the option can be exercised is termed maturity, T. We provide a full example of pricing and hedging with European type option on swap, the European swaption.

5.3.2.1 Valuation of the European Swaption

Options on swaps are called swaptions. A European swaption call[7] has for the holder the pay-off,

$$\mathcal{X}^{call}_{swaption} = \Phi(F_T) = \max\left[0, F_T - K\right]. \tag{5.14}$$

The put has the same pay-off except the K, and the F_T has switched place. The writer has the same pay-offs only with negative sign. Figure 5.5 depicts these pay-offs, including the premium.

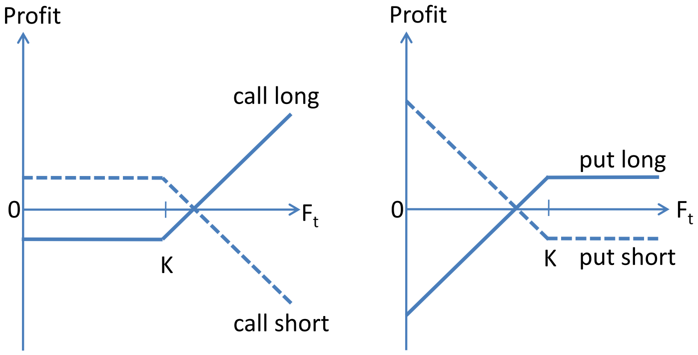


Figure 5.5: Pay-off diagram for the short and long call, and short and long put.

The pricing formula for an hour-priced European swaption is found by evaluating the risk neutral expectation, equation (5.12) inserted equation (5.14). We perform this calculation in appendix A and present the resulting formulas here. The European call swaption is priced by:

$$\Pi_t = c_t(F_t, T) = F_t\,\mathcal{N}(d_1) - Ke^{-r(T-t)}\mathcal{N}(d_2), \tag{5.15}$$

and the put swaption:

$$\Pi_t = p_t(F_t, T) = Ke^{-r(T-t)}\mathcal{N}(-d_2) - F_t\,\mathcal{N}(-d_1). \tag{5.16}$$

These formulas give the price per hour, meaning that we need to multiply by the number of hours in the delivery period $[T_b, T_e]$ to obtain the true price per contract. T is the maturity date of the option, F_t is the price of a swap at time t, while K is the agreed upon strike. r

---

[7] A call swaption is also known as a payer swaption, while a put option on a swap is a receiver swaption

is the constant interest rate for the whole period and at the beginning of the delivery period. d_1 and d_2 is:

$$\begin{aligned} d_1(F_t, K, T) &= \frac{\ln(\frac{F_t}{K}) + (r + \frac{\sigma^2}{2})\,(T-t)}{\sigma\sqrt{(T-t)}}, \\ d_2(F_t, K, T) &= d_1(F, K, T) - \sigma\sqrt{(T-t)}, \end{aligned} \tag{5.17}$$

where $\mathcal{N}(\,\cdot\,)$ is the cumulative normal function. We now perform a calculation of an example to show how these formulas work.

Example 3 *A computer center would like to have a planning horizon of three months. They anticipate a decrease in the Virtual Server price, but would also like to benefit if the prices would rise substantially. Let us say that it is normal in the derivatives market that the option expires one month before the underlying swap contract, so the swaption has an expiration date in two months,* $T = 60$*. The company would then buy, let us say,* $10,000$ *European put swaptions with strike* 95 *Norwegian cents,* $K = 95$*, and with a delivery period of one week,* $T_b = 90$ *and* $T_e = 97$*. Suppose that these three-month-Grid-swaps are traded at* 100 *Norwegian cents/hour,* $F_\tau = 100$*, and that they have a constant volatility* 80 *%,* $\sigma = 0.8$*. The continuous risk free rate of the bond is constant at five percent p.a.,* $r = 0.05$*. We find the price of one such European put using the pricing formula* (5.16)*:*

$$p = (9,83\, cents/h \cdot 168\; hours) = NOK\ 16.51,$$

and the total price the computer center pays for all $10,000$ *contracts is:* $NOK 165,000$*. Two months later, it turns out that the prices really have declined substantially and the swaps are now trading at* $F_{60} = 90$*. The computer center exercises the option and is able to sell the underlying swap contracts for* $K = 95$ *instead of* 90*. By paying the initial premium, the company has been able to postpone the risk management decision of selling the swaps. Had the swap prices moved in the other direction, say* $F_{60} = 110$*, the company would have done better by not exercising the option and selling swaps at the market price.*

5.3.2.2 Hedging of the European Swaption

The writers of European swaptions will often try to hedge their sale of the options. Although they are attracted by the premium, they have a risk of severe losses, even unbound in the case of European call, if the price movements are, in their point of view, adverse. In a delta hedge, small movements in the swaption price, due to movements in the underlying, are exactly offset by the delta position in the swap itself. For the portfolio to be delta neutral, we use the formula we derive in the appendix A. For a call swaption, the delta formula is:

$$\Delta_{\text{call}} = \mathcal{N}(d_1)\,, \tag{5.18}$$

and for a put swaption:

$$\Delta_{\text{put}} = \mathcal{N}(-d_2)\,. \tag{5.19}$$

We provide formulas for the other *Greeks* in Appendix A.

Example 4 *The writer of the European put swaption, used in example 3, wishes to hedge her short position. She would like to set up an initial delta hedge, in order to make the portfolio relatively insensitive to small changes in the swap prices. The writer manages to do this by going short* $\Delta = 0.365$ *swaps per option sold, in total,* $3,648$ *swaps. The money from both selling the swaption and the swaps are invested in the bond. Let us say* *that during a very short time, the swap price has declined to* 98 *Norwegian cents. The price of the option has now risen to* $17.77\ NOK$ $(\Pi(F_t = 98, K = 90) \cdot 168\ h)$, *which is the amount the writer would have had to pay to the holder if the option would have been exercised, yielding a net loss of* $(16.51 - 17.77)\ cents/h \cdot 10,000 =\ NOK - 12,600$. *The short swap position has a net gain* $NOK\ 12,260$ $(2\ cents \cdot 168\ h \cdot 3,648\ swaps)$ *because the writer has to pay less to repay the swaps borrowed in the market; in addition, she has some (insignificant) interest income on the bonds position. In total, she encounters a loss on about* $NOK\ 340$ *instead of a loss on* $NOK\ 12,600$ *(!) if un-hedged.*

In order to maintain delta neutrality over a longer period, the swap position must be changed. Since the delta is a derivative of the option price to the underlying, the neutrality only lasts for small changes in the swap price, depicted in figure 5.6. For larger price movements, the delta position would not neutralize the movements in the option price. Hence, for larger movements in the price, the delta needs to be recalculated and the portfolio rebalanced. The answer to how many rebalances are necessary to keep a delta neutral portfolio in a frictionless market is: the more, the merrier. In a real situation, the hedging is done under the real probability measure $\mathbb{P}$, as the hedging take place with real price processes. We perform the hedging as a simulation and hedge under the probability measure $\mathbb{Q}$, but the result is equivalent. By simulating many different swap price processes, we are able to analyze how well different delta strategies affect the success of the hedge. The main factors of such simulations are the variance of the end value of all the simulations and the maximum loss encountered in the worst possible simulation. These will show how well the process is controlled, as well as the worst-case scenario. Other factors, which also might be considered, are the mean profit of all the simulations as well as the maximum payoff of the most successful price path.

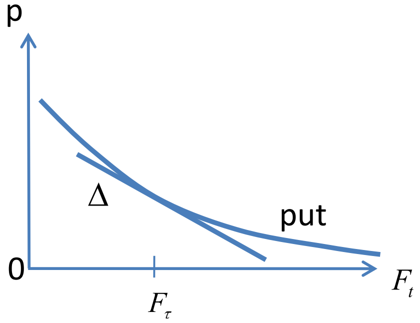


Figure 5.6: Relation between swap price and option price at time $\tau > T$. For the put option, an increase in the swap price would result in a smaller pay-off. The Δ is therefore negative.

We use a computer Grid to run the simulations. The example serves to illustrate two points: that we are able to perform calculations on the Grid swaptions, and that Grids can be used for high through-put calculations. If we had run the simulations on a single server, it would have taken approximately 40 hours wall-time, while using Grid computing with nodes, the time wall-clock time required is approximately 1.5[8] hours. We do not suggest using this much resources to calculate simple options, as we do with the European swaption, but for valuating very complex options on swaps, even larger Grids might be required to obtain the results fast enough. We implement the hedging strategies in the programming

---

[8] The wall clock time used by the Grid is not simply $40/31$ because the nodes used are not identical and some are slower than others. In addition, the Grid has some overhead of distributing the jobs, and in our case, some nodes are used to other jobs as well.

language C++ and provide the code in appendix D. We also provide, in appendix C, a *job description file* along with useful commands in order to run C++ executable programs successfully on a Grid using the *gLite* middleware. A computer-adapted (that is not Grid-adapted) version of the hedging program can be found on the accompanying CD.

Example 5 *Let us return to the swaption writer of example 4 with her problem of how to hedge the* $10,000$ *two-month European put swaption on the three-month swap. The most important factors to notice are the variance and the maximum loss. Both these values represent how much risk the writer takes on. Remember, she might encounter unbound loss, but only a finite gain. We simulate* $3,000,000$ *swap price processes and see what happens when we put up different delta hedging strategies. The result is depicted in table 5.3.2.2. First, a naked position is considered. Despite having a mean of almost zero*[9]*, the worst outcome scenario predicts a loss of* $64.2\, cents/h$*, see table* 5.1*. Not to hedge the swaption would in the worst case represent a loss of* $NOK1.08$ *million* $(64.2cents/h \cdot 168h \cdot 10,000swaptions)$*. The biggest gain for the writer is the premium that we, in example 3, found to be* $NOK165k$*. The variance also increases the writer's risk. The variance numbers should be seen as a dispersion of the pay-offs. A large number means that the pay-offs from the simulations will spread, while a small number represent a relatively grouped outcome. The naked position has a variance of* 188 *which represent considerable uncertainty about the outcome.*

If our writer performs only one hedge, that is, when selling the swaption she immediately delta hedges and holds this position until expiration, the variance has now improved to be 87*, again see table 5.3.2.2 This variance is still too high and represents a huge risk. The highest value that the swap reaches at the expiration of the option is 416.8 Norwegian cents whereas the swaption had a strike price of* $95\ cents$*. This figure, however, only appears to be good news. The maximum encountered loss is even worse than for the un-hedged position:* $NOK2.37million$ *(!) in total* $(141\, cents/h \cdot 168\, h \cdot 10,000\, swaptions)$*. In this case, the maximum loss can be blamed on the short position in the swap, taken at the beginning. When the swap price sharply increases, the swaption is worthless, so she does not have any loss on it, but she has to buy back the swaps from the market to make up for the short sell.*

Increasingly frequent rebalancing severely reduces the variance. We have calculated the different factors based on rebalancing once a month, ever 10 days, every day, each 2.5 *hours and so on, ending up with the extreme of rebalancing nearly every* 1.4 *minutes. In the latter case, the variance is reduced drastically to* 0.02 *and the maximum loss is only* $NOK\ 11k$*. Figure 5.7 illustrates the reduction in the output spread when the number of hedges increases, while table 5.3.2.2 gives an overview of the mean, variance, maximum loss and maximum gain in each hedging case.*

When looking at the mean in the table, we see that it is close to zero, but still negative for all cases. In an ideal case, it should be identical zero, c.f. the original replication argument of Black and Scholes. In Monte Carlo simulations, the simulated price is seldom exactly the same as the theoretical price. We suspect the difference to be a result of the number of calculations, the built-in random generator, and the approximate programming of the price process. In fact, when we run $10E^9$ *simulation with no hedging, the mean is reduced to* $8.0E^{-4}$*, hence the simulations give a good estimate of the option price.*

In reality, the problem with such frequent rebalancing of the portfolio is its cost. In the Grid market, as in any other market, it is impossible to buy and sell options to the same price. There are, of course, parties that facilitate the trading and would like to be compensated for the job they do, for example the clearinghouse and the market makers. This compensation

---

[9]

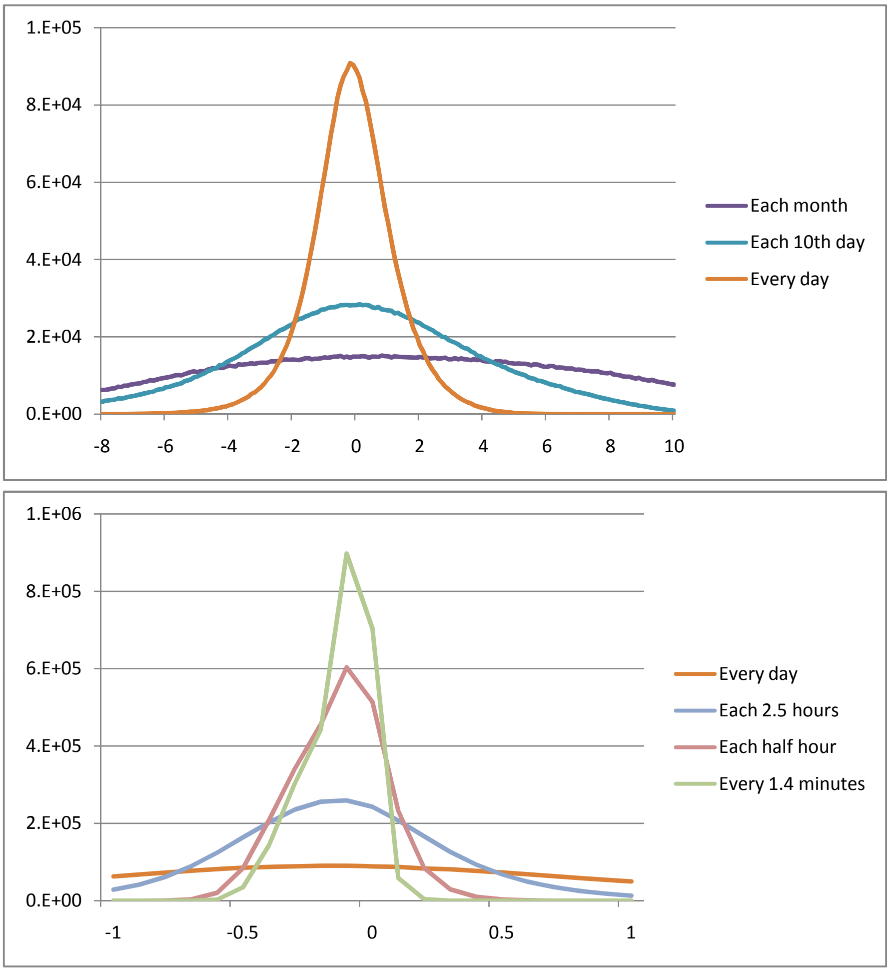


Figure 5.7: The figure at the top shows how the variance goes down with increasing number of hedges. Here, the longest hedging period is one month, while the shortest is one day. At the bottom figure, we have gradually increased the number of hedges. Note the difference in axis from the top figure. The pay-off span is much narrower, and the frequency is 10 times higher. Daily hedging is included also in the bottom figure as a reference. We see here a similar result, with relatively high variance for the more seldom hedges to a narrow peak for hedging seven times a minute. For this extreme hedging, the mode over the defined pay-off intervals has a frequency of almost 900,000.

results in incurred transaction costs for the trades concluded in the swap market, making it more difficult to perform continuous delta hedging. In such an environment, the writer would prefer not to set specific times to perform the hedging, but she would rather focus on the movement in the swap prices. If the prices are quite stable during a period, she sees no need to rebalance the portfolio.

We have now given arguments in support of the development of a derivatives market on Grid resources, and shown that it is indeed technically possible to construct such market. The question must now be addressed, is, once put into effect, would the market be able to last?

Number of hedges	Mean	Variance	Maximum Loss	Maximum Gain
Naked position	$-4.95E^{-2}$	$1.88E^{2}$	$-6.42E^{1}$	$9.83E^{0}$
Immediate hedging	$-7.85E^{-2}$	$8.70E^{1}$	$-1.41E^{2}$	$1.19E^{1}$
Each month	$-7.81E^{-2}$	$4.70E^{1}$	$-9.62E^{1}$	$12.1E^{1}$
Each 10^{th} day	$-8.23E^{-2}$	$1.71E^{1}$	$-4.21E^{1}$	$11.9E^{1}$
Every day	$-8.07E^{-2}$	$1.89E^{0}$	$-1.24E^{1}$	$7.72E^{0}$
Each 2.5 hours	$-8.08E^{-2}$	$2.08E^{-1}$	$-3.57E^{0}$	$3.16E^{0}$
Each half hour	$-8.06E^{-2}$	$3.24E^{-2}$	$-1.43E^{0}$	$1.23E^{0}$
Every 1.4 minutes	$-8.07E^{-2}$	$1.56E^{-2}$	$-6.38E^{-1}$	$5.53E^{-1}$

Table 5.1: Depiction of how the mean, variance, maximum loss and maximum gain are changed with different hedging strategies. Generally, an increasing number of hedges reduces the variance and makes the final pay-off more sure. In addition, the risk of large losses is also reduced, so is the maximum profit. The mean is almost unaffected by the hedging frequency.

5.4 Success Factors for a Derivatives Market on Grid Resources

Far from all proposed derivatives contracts become commercially successful. According to Vassdal (1995), only 30 percent of all futures contracts thrive. 70 percent of the initiated futures contracts are completely withdrawn from trading, or are traded in low volumes. In order for a derivatives market to exist, Cuny (1993) emphasizes that it must attract agents who want to use financial instruments for risk management purposes. Even though the Grid resource buyers normally take opposite position of the Grid resource sellers, the swap market needs more liquidity in order to be successful, enter the speculators. They are motivated by the possibility of profiting on the present risks. A successful derivatives market provides the risk mitigators with both a high-quality risk management instrument and a liquid market where these instruments can be traded at low cost (Cuny, 1993).

A market's successfulness depends on several factors in the underlying market, the contract specification, the properties, and the external factors of the market. These factors are derived from common factors for successfulness of futures markets, and we see that the Grid derivatives market do indeed fulfill many of these requirements. The presences of these elements do not guarantee success, but the absence of them will certainly reduce its probability.

5.4.1 Underlying Market Factors

In order to ensure greater probability of success with a derivatives market, the underlying market should contain homogeneous, standardized and expensive products. In order for the derivatives market to be viable, the underlying market should have a degree of uncertainty and volatility that is difficult to control with existing financial risk management solutions.

5.4.1.1 Uncertainty and Volatility in the Underlying Market

If the prices were stable, no one would ever have to face risk, since all price movements would be deterministic. A volatile and uncertain market would attract both risk managers and speculators. Price uncertainty could be leveled out by the use of derivatives and people with different beliefs about future prices would find this market an attractive place in which to speculate (Carlton, 1984). The Continuous Virtual Server Exchange, CVSE, probably contains both uncertainty and variability. This is due to the non-storability property of the resources, as well as the trading mechanism being a continuous double-sided auction.

5.4.1.2 Expensive Products

There would not be much of a risk if the price movements only insignificantly affected the agents. In order to have a viable derivatives market, the participants must be significantly affected by adverse price movements. Even though one Virtual Server by itself probably is not very expensive, the buyers would likely wish to require many of these. In addition, the delivery happens over a period, making the total risk of price movements quite severe.

5.4.1.3 Homogeneous Resources and Standardization

A derivatives market has a larger chance of viability if the products are homogeneous. If differences exist, these can be identified based on physical and not personal judgment (Vassdal, 1995). This requirement is fully satisfied for the Virtual Servers. VSs are standardized homogeneous resources based on some common requirements.

5.4.1.4 Not Possible to Reduce Risk with Other Existing Financial Instruments

The relation of underlying with existing contracts also affects the viability of a new derivatives contract. New contracts that allow risk management of previously unmanaged risks have a greater chance of being popular (Rausser & Bryant, 2004). If a trading place for computer resources is established, there are no other market segments where trade of similar goods is conducted. One could imagine buying or selling futures on stocks in the computer centers, perhaps combined with electricity, to provide some risk management, but this seems rather farfetched. We find that there does not exist any other way of financially reducing the risk of future price movements other than creating the Grid swap market.

5.4.1.5 Large Potential Number of Interested Participants and Industrial Structure

A large underlying market is important so that no one has the possibility to manipulate the underlying prices (Vassdal, 1995). It would be interesting for a speculator with a huge short position in derivatives, when these are *in-the-money*, to manipulate the swap prices to make the derivatives *out-of-the money*. If such manipulation should take place, this would result in severe loss of confidence in the derivatives market. With a large number of participants in the underlying market, each agent is too small to effectively affect the prices. Carlton (1984) argues that more firms are involved in buying or producing a good, the bigger the number of potential actors on the derivatives market, and hence greater chance for it to develop. It is difficult to say anything about the market size for Grid computing, as the market does not yet exist. Still, as there are no particular barriers to enter such a computing trading place, e.g. the cost of sending information is decreasing; computing exists in every company, department and home, the likelihood for a Grid spot market is high.

5.4.2 Design of the Derivatives Contract

While the ability of the underlying to create a foundation for a derivatives market is mostly exogenously given, or the result of the specifications in the spot market, the design of the derivatives contracts will directly affect the viability of the contract in a financial instrument market. There are plenty of examples of failures due to poor design of the contract. Rausser and Bryant (2004) finds that proposals emanating from the academy seem to fail more often than proposals from practitioners. We are therefore very prudent and desist from placing too many rigorous constraints. We will discuss, however, two important factors when designing derivatives contracts: the product specifications and the prevention of manipulation.

5.4.2.1 Identical Products

One of the success factors of contract specification is that the contract should be as identical as possible to what the risk mitigator would like to secure (Vassdal, 1995). For a standardized market, this is difficult, as it uses the resources differently, and so probably also has different risk management needs. For the Grid market, we propose trading the same resources, Virtual Servers, in both the spot and derivatives market, but with unequal usage periods. For the spot market, short time periods are suitable as these provide greater flexibility. While in the derivatives market, we need longer time periods because this reduces the number of contracts and hereby increases the liquidity.

5.4.2.2 Prevention of Manipulation

Another factor related to contract design is prevention of manipulation. In addition to having a large underlying market, the contract specification must also be set up so that no single party or group can control the delivered good. A common strategy for preventing corners and squeezes is to allow delivery of nonstandard grades of an underlying asset at a premium, or at discounted future prices (Rausser & Bryant, 2004). For the Grid market, we have suggested a multi grade resource. It might be interesting to build swap prices on only normal graded Virtual Servers, while the other grades are traded with a premium or discount, and hence reduce the chance of manipulations.

5.4.3 Market Factors

We present some market factors that increase the success rate. These are the absence of governmental rules and regulations, and a proper design of the market surroundings that ensures that the agents hold their commitments.

5.4.3.1 Government

There are many examples of governmental regulations that have destroyed existing futures markets. For instance, some coffee, cotton, butter, rice, tobacco and peanut futures were subject to such regulations (Vassdal, 1995). Carlton (1984) also empirically finds that successful contracts are based on non-subsidized commodities. It is not given how governments would react to an international trade of computer resource, but some degree of protectionism, taxation, information control, or other barriers are likely to appear.

5.4.3.2 The Commitments are Met

In order for an agent to voluntarily enter a contract agreement; he must be positive that the counterpart will keep her promises, explicitly stated in the contract (Rausser & Bryant, 2004). If the agents cannot be certain of this, derivatives trading will be very difficult. In a Grid setting, we recommend the use of a clearinghouse in order to separate the writer from the holder, and to eliminate the risk of default.

Chapter Summary:

This chapter started with the assumption of the existence of a functioning trading market for Grid resources. We argued that a derivatives market would attract participants with different motives. We introduced the notion of swap contracts and claimed that this would create a stable future for the risk mitigators. The pricing of these swap contracts was suggested to be done by way of auctions, and we argued that the swap price process would follow a geometric Brownian motion. We showed that it is possible to price and hedge other derivatives on these swaps and gave a complete example one such derivative, the European swaption. Finally, we presented different factors increasing the probability of a successful derivatives market, to conclude that a derivatives market on Grid resources should satisfy most of these factors.

Summary of Part II

While chapter 2 dealt with the immense computing need CERN faces and treated the solution to this problem, chapter 3 argued that the scarcity of computer resources is, and will continue to advocate a commercial trade of these resources through Grids. In chapter 4 "Exchange Market", we outlined an exchange for Grid resources on which we built the derivatives market in chapter 5 "Derivatives Market". For quick reference, we have illustrated the economic model that the two latter chapters jointly propose in figure 5.8. We fist discussed the agents that might be interested in a Grid resource trading market. We found that the potential buyers in the market were providers of *Hardware-as-a-Service* (HaaS) , either selling to external organizations, or providing the service internally, as the *User Integrators* and the *Software Integrators* do. We state that the sellers of Virtual Servers are professional resource vendors, such as huge computer centers. These agents are depicted in figure 5.8.

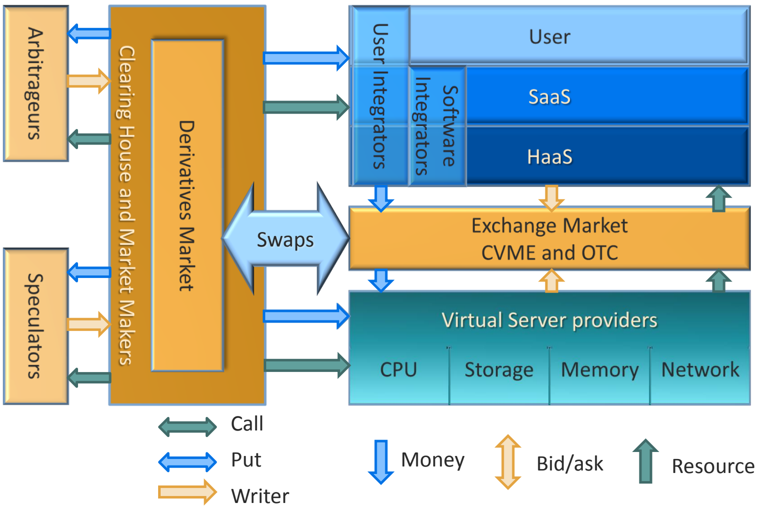


Figure 5.8: The complete Grid economy

We found that the best way in which buyers and sellers could meet to trade Grid resources would be through a double-sided auction. We developed two different, but related, markets: the *Continuous Virtual Server Exchange* (CVSE) and the *Over-The-Counter* (OTC) market. In the OTC market, resources with a special feature, be it high performance CPUs or a particular storage capacity is offered. We proposed that these special resources should be traded by periodic auctions, where the buyers specify minimum resource criteria and the sellers state the maximum. The auctioneer then proceeds to solve this resource allocation optimization problem.

For the CVME, the market mechanism is a bit different. First of all, it operates with continuous allocation in order to fully satisfy the requirement of *scheduling efficiency*. The bidders and sellers are able to obtain or sell resources immediately by posing a buy- or sell order, respectively. Second, only homogeneous, non-storable, single-attributed bundles of computer resources, called *Virtual Servers* (VS), are traded, making the auctioneer's allocation and pricing problem easy to solve. We found the allocation problem complexity of CVSE to be only $O\left(n \lg(n)\right)$.

The CVSE satisfies most of the requirements from the agents, resources and other factors. The biggest problem was that the market is not guaranteed to be *economically efficient*, and that it might provoke erratic, not-foreseen price changes. These *price oscillations* are due both the construction of the mechanism as continuous, market specific factors and external factors, and create difficulties for the agents who rely on stable environments. We therefore suggested the construction of a financial derivatives market on Grid computing.

The problem of creating a derivatives market is that the VSs are non-storable. It is, thus, impossible to exploit normal arbitrage theory. To cope with this issue, we constructed *swap contracts* as the link between the CVSE market and the derivatives market, depicted in figure 5.8. A swap contract is an agreement to exchange a future variable cash flow with a deterministic cash flow over a period, called the delivery period. We let the possible buyers and sellers, that is, the risk mitigators, speculators and arbitrageurs, buy and sell the swaps in an exchange. We stated, based on the *Rational Expectancy Hypothesis* and the *Efficient Market Hypothesis* that there was absence of arbitrage in the swap market, and that the price process can be described as a *Markov process*. We further assumed normal distributed price changes and let the price process follow a *geometric Brownian motion*.

We were then able to model other derivatives on these swaps, and we posed the general pricing and delta-hedging formulæ (5.12) and (5.13). We gave an example of a derivative on swaps, namely the *European swaption*. This gives the holder the possibility, but not the obligation, to buy or sell the underlying swap at strike K at the expiration date T. We derived the pricing formula for the call- (5.15) and the put- (5.16) swaption. We gave a large hedging example with the use of Monte Carlo simulation, calculated with the use of Grid computing, and showed how delta hedging makes the variance and max loss decrease with increased hedging frequency.

Finally we discussed common success factors for derivatives markets and saw how underlying market factors, such as product specifications, uncertainty, and the possibility of reducing risk with existing products, directly affected the applicability of derivatives. In addition, the design of the derivatives contract and external market factors, such as governmental regulations and the existence of a clearinghouse, affect the probability of success. We stated that a Grid derivatives market does indeed fulfill several of these criteria.

Chapter 6

Conclusion

In this chapter we summarize our findings, state what we believe to be our contributions to this field, and finally propose some subjects for further research.

6.1 Conclusion and Our Contribution

We have built on the expectation that a trading market for computer resources will emerge, but as mentioned in section 3.3 there are barriers such as licensing, standards, and culture. However, we view this as the future based on the strong trend of delivering IT as a service. Furthermore, a derivatives market on Grid resources seems both probable and viable. In a spot market for computer resources with unpredictable, erratic prices, the agents will request ways to manage their risk. We have constructed a complete, coherent Grid economy, consisting of both a spot market and a derivatives market. Given this market combination, the participants wishing to reduce their risks related to computer resource trade are able to find other parties willing to take on these risks.

To link the spot and the derivatives markets, we have suggested Grid swap contracts. Besides being a risk mitigation instrument, the swap allows for further construction of derivatives. With the introduction of this storable asset, that is the Grid swap, we build a theoretical framework for pricing and hedging of the derivatives contracts.

The future existence of a derivatives market depends, to a large extent, on how the underlying market will behave. Market trends indicate that IT will be transformed from a good into a service in a few years. We argue that a standardized bundle of computer resources, termed a Virtual Server, can be a good candidate for the traded service unit. However, continuous trading of homogeneous resources allows for scheduling efficiency and market liquidity, but may result in risk erratic and unpredictable prices. The suggested derivates market then proceeds to redirect the risk of erratic and unpredictable price pattern that could normally result from such a market, in the directing of those willing to invest in them.

The computer resource trading is founded on the Grid technology. We are convinced that the Grid has bright commercial business prospects. In fact, market analysis show that Grid computing is 2-5 years away from mainstream adoption. A number of commercial sectors have already implemented large Grids, but inn order for the different Grids to communicate with each other, open standards are needed. Information Technology is a driving economic force in society, which motivates the politicians to invest in R&D and promote the Grid technology to commercial actors. With the growing adoption of the Grid, the IT industry stands before an irreversible paradigm shift, which will greatly impact both the IT market and society in general.

We have employed our techno-economic knowledge on the Grid Computing technology to look at it from a different perspective than the traditional Physics and Engineering focus at CERN. For CERN, this thesis offers deeper insight into possible commercial aspects of the technology. Companies, especially start-ups working with Grid computing related to CERN, may take interest in this thesis as it will allow the reader to better understand the existing market and be aware of possibilities if a trading market ever appears. Such a market will be able to provide easy access to inexpensive computing, which is especially interesting to entrepreneurs. The innovators should be aware of the market risks, and know how to mitigate these. Thus, our thesis may serve as a tool for CERN to disseminate knowledge internally as well as externally.

6.2 Further research

We believe that further studies of the Grid economy would be interesting. For instance, performing simulations on these markets could provide important knowledge concerning the economic efficiency of the proposed markets, both for the derivatives and the underlying market. These data could later be compared to other mechanisms, in order to further assess the Grid economy. Laboratory tests, where people apply different strategies and learning abilities on the proposed markets, may also indicate how well these markets would perform in practice.

In addition, to make this market come true, exact specifications of the Virtual Servers are needed. It would be interesting to study which resources to include in this bundle, in order to satisfy as many parties as possible. The same specifications study could be done for the Grid swaps in order to find the optimal standard contract.

Finally, further studies might also be conducted into possible new derivatives contracts, especially designed for the need of the participants. For some sectors using Grid resources, exotic swaptions such as the American, look-back, swing or barrier swaptions, might be suited to cover a particular Grid resource requirement. The study of different types of derivatives contracts would provide a valuable addition to risk management or speculations on Grid computing.

With great excitement, we look forward to the day when the access to entire world's computer resources is just a click and an invoice away.

Appendices

Appendix A

Derivation of the European Swaption Put and Call Formulas

The goal of this appendix is to derive the pricing formulas for the European swaptions given in section 5.3.2.1 and the hedging formulas of section 5.3.2.2. First we derive the price process of the Grid swap, then, we derive the pricing and hedging formulas.

A.1 Price process of the Grid Swaption

We assume the swap price change follow a geometric Brownian motion

$$dF_t = F_t \mu dt + F_t \sigma d\bar{W}_t, \qquad \text{under } \mathbb{P}, \tag{A.1}$$

Where μ is the real drift of the swap, σ is the constant volatility of the returns of the swap price F, and $d\bar{W}$ is a Wiener process. By applying the Girsanov's theorem with

$$d\bar{W}_t = \frac{r-\mu}{\sigma}dt + dW_t,$$

Where r is the constant market rate of return, we obtain

$$dF_t = rF_t dt + \sigma F_t dW_t, \qquad \text{under } \mathbb{Q},$$

with initial value $F_0 = f$. We let $G_t = \ln F_t$ and use Itô's Lemma to obtain

$$\ln F_T = \ln F_t dt + \int_t^T \left(r - \frac{\sigma^2}{2}\right) du + \int_t^T \sigma dW_u,$$

and by integrating and solving for F_T, we have the following relation:

$$F_T = F_t \; e^{\left(r-\frac{\sigma^2}{2}\right)(T-t)+\sigma(W_T-W_t)}. \tag{A.2}$$

F_t is a exponential martingale under $\mathbb{Q}$.

A.2 Pricing of the European Swaption

The stochastic claim is defined by

$$\mathcal{X}_{swaption} = \Phi(T, x), \tag{A.3}$$

where Φ is the pay-off function of the derivatives. Given absence of arbitrage in the swap market, and a complete derivatives market, we may use risk neutral pricing:

$$\Pi_t(T, \Phi(x)) = \mathbb{E}^{\mathbb{Q}}\left[e^{-r(T-t)}\Phi(x)\right] \tag{A.4}$$

The pay-off function of the European call swaption, $\mathcal{X}^{call}_{swaption}$, is

$$\Phi(F_T) = [F_T - K]^+, \qquad K > 0. \tag{A.5}$$

The price of a European call swaption is

$$\begin{aligned} c_t = \Pi_t &= \mathbb{E}^{\mathbb{Q}}{}_t\left[e^{-r(T-t)}\Phi(F_T)|\mathcal{F}_t\right], \\ &= G_t(x), \end{aligned} \tag{A.6}$$

where

$$G_t(x) = \mathbb{E}^{\mathbb{Q}}\left[e^{-r(T-t)}g\left(xe^{\sigma(W_T-W_t)+\left(r-\sigma^2/2\right)(T-t)}\right)\right],$$

As $(W_T - W_t) \sim \mathcal{N}(0, T-t)$, the $G_t(x)$ may be evaluated by

$$G_t(x) = \int_{\mathbb{R}} e^{-r(T-t)}\Phi\left(xe^{\sigma\sqrt{T-t}\,u+\left(r-\sigma^2/2\right)(T-t)}\right)\frac{e^{-u^2/2}}{\sqrt{2\Pi}}du. \tag{A.7}$$

The equation (A.6) with (A.7), inserted the pay-off function for the European swaption call, A.5, becomes

$$c_t(F_t) = \int_{\mathbb{R}} e^{-r(T-t)}\left[F_t e^{\sigma\sqrt{T-t}\,u+\left(r-\sigma^2/2\right)(T-t)} - K\right]^+ \times \frac{e^{u^2/2}}{\sqrt{2\Pi}}du. \tag{A.8}$$

We would like to get rid of the $\max$ function in equation (A.8) in order to evaluate the integral. By noting that

$$[\cdots]^+ = \begin{cases} 0 & \text{if } \sigma\sqrt{T-t}\,u + \left(r - \frac{\sigma^2}{2}\right)(T-t) \le \ln\left(\frac{K}{F_t}\right), \\ \text{else} & F_t e^{\sigma\sqrt{T-t}\,u+\left(r-\frac{\sigma^2}{2}\right)(T-t)} - K. \end{cases} \tag{A.9}$$

We solve the first line in (A.9) with respect to u and obtain

$$\begin{aligned} \sigma\sqrt{T-t}\,u + \left(r - \frac{\sigma^2}{2}\right)(T-t) &\le \ln\left(\frac{K}{F_t}\right) \\ u &\le \frac{1}{\sqrt{T-t}}\ln\frac{K}{F_t} + \left(\frac{\sigma^2}{2} - r\right)\frac{\sqrt{T-t}}{\sigma} = d_2. \end{aligned}$$

Equation (A.8) gives us

$$\begin{aligned} c_t(F_t) &= \int_{-d_2}^{\infty} e^{r(T-t)}\left(F_t e^{\sigma\sqrt{T-t}\,u+\left(r-\frac{\sigma^2}{2}\right)(T-t)} - K\right)\frac{e^{-u^2/2}}{\sqrt{2\Pi}}du \\ &= F_t\int_{-d_2}^{\infty}\frac{e^{\sigma\sqrt{T-t}\,u-\frac{\sigma^2}{2}(T-t)} - \frac{u^2}{2}}{\sqrt{2\Pi}}du - Ke^{-r(T-t)}\mathcal{N}(d_2), \end{aligned}$$

where

$$\mathcal{N}(F_t) = \int_{-\infty}^{F_t}\frac{e^{-u^2/2}}{\sqrt{2\Pi}}du.$$

Changing the variable: $v = u - \sigma\sqrt{T-t}$, so that

$$\begin{aligned} c_t(F_t) =& F_t \int_{\left(-d_2-\sigma\sqrt{T-t}\right)}^{\infty} \frac{e^{-v^2/2}}{\sqrt{2\Pi}} dv - Ke^{-r(T-t)}\mathcal{N}\left(d_2\right) \\ =& F_t\,\mathcal{N}\left(d_1\right) - Ke^{-r(T-t)}\mathcal{N}\left(d_2\right). \end{aligned}$$

We thus obtain the formula for the price of an European swaption call, where

$$\begin{aligned} d_1(F,K,T) &= \frac{\ln(\frac{F_t}{K}) + (r+\frac{\sigma^2}{2})\,(T-t)}{\sigma\sqrt{(T-t)}}, \\ d_2(F,T) &= d_1(F,T) - \sigma\sqrt{(T-t)}. \end{aligned}$$

The derivation for the European swaption put is similar and we only state the result:

$$p_t(F_t,T) = Ke^{-r(T-t)}\mathcal{N}\left(-d_2\right) - F_t\,\mathcal{N}\left(-d_1\right). \tag{5.16}$$

With d_1 and d_2 defined in (5.17) We have now the pricing expressions given in section 5.3.2.1.

A.3 Hedging of the European swaption

The hedging formulas are easy to find. First, let us find the delta of an European call swaption. By taking the partial derivative of equation (5.15) by with respect to F_t and derivate by part we obtain:

$$\frac{\partial c_t(F_t,T)}{\partial F_t} = \mathcal{N}\left(d_1\right) + F_t\frac{\partial\mathcal{N}\left(d_1\right)}{\partial F_t} - Ke^{-r(T-t)}\frac{\partial\mathcal{N}\left(d_2\right)}{\partial F_t}. \tag{A.10}$$

We know that $\frac{\partial\mathcal{N}(d_i)}{\partial F_t} = n(d_i)\frac{\partial d_i}{\partial F_t}$. and that $n(d_2) = \frac{n(d_1)F_t}{Ke^{-r(T-t)}}$ Since $d_2 = d_1 - \sigma\sqrt{T-t}$, the derivatives of d_1 and d_2 with respect to F_t are equal. By inserting these relations into equation (A.10) we have:

$$\begin{aligned} \delta_{\text{call}} &= \frac{\partial c_t(F_t,T)}{\partial F_t} \\ &= \mathcal{N}\left(d_1\right) + F_t n\left(d_1\right)\frac{d_1}{\partial F_t} - Ke^{-r(T-t)}\frac{n(d_1)F_t}{Ke^{-r(T-t)}}\frac{\partial d_1}{\partial F_t} \end{aligned}$$

The formula for the delta of a swaption call is

$$\Delta_{\text{call}} = \mathcal{N}\left(d_1\right). \tag{A.11}$$

For a put, the derivation is similar and the result is

$$\Delta_{\text{put}} = \frac{\partial p_t(F_t,T)}{\partial F_t} = \mathcal{N}\left(-d_2\right) \tag{A.12}$$

For the other *Greeks* we only provide the results (these are adapted from Haug (2005a, 2005b)):

$$\begin{aligned} \Gamma_{call} = Gamma_{put} &= \frac{\partial\Delta}{\partial F_t} \\ &= \frac{n(d_1)}{F_t\sigma\sqrt{T-t}} \end{aligned}$$

$$\rho_{call} = \frac{\partial c}{\partial r} = (T-t)Ke^{-r(T-t)}\mathcal{N}\left(d_2\right)$$

$$\rho_{put} = \frac{\partial p}{\partial r} = -(T-t)Ke^{-r(T-t)}\mathcal{N}\left(-d_2\right)$$

$$\begin{aligned} Vega_{put} = Vega_{call} &= \frac{\partial c}{\partial \sigma} = \frac{\partial p}{\partial \sigma} \\ &= F_t n(d_1)\sqrt{T-t} \end{aligned}$$

$$\Theta_{call} = \frac{\partial c}{\partial T} = -\frac{F_t n(d_1)\sigma}{2\sqrt{T-t} - rKe^{-r(T-t)}}\mathcal{N}\left(d_2\right)$$

$$\Theta_{put} = \frac{\partial c}{\partial T} = -\frac{e^{r(T_e-T)}F_t n(d_1)\sigma}{2\sqrt{T-t} + rKe^{-r(T-t)}}\mathcal{N}\left(-d_2\right)$$

Appendix B

Complexity for the CVSE Allocation Problem

We here show that the allocation algorithm of the auctioneer in the case of CVSE has a complexity of $O(n \log n)$. Let him possess two sorted arrays, one with asks in queue, and one with bids. The bids and asks comes in with irregular frequency, but never more than one at the same time. The pseudo code for the allocation problem would be:

```
1. for all bids and asks do     //We only look at incoming bids, but similair for asks
2.     if bid > smallest ask in ask queue
3.         allocate the resources and update the ask-list
4.     else:
5.         insert bid in sorted bid-list
```

Let n be the number of bids and ask. We use asymptotic notation, where O is the upper bound complexity when the problem size increases to infinite. For the grid allocation problem, this would be if the number of bidders and sellers become infinite. Let us call the number of buyers and sellers n. We now take a look at the pseudo code to find the complexity of the allocation problem of CVSE.

First, let us look at line 2 to 5. Line 3 is effectuated only if the incoming bid is bigger than the smallest ask in the ask queue, as stated by line 2. The comparison and the eventual update in the queue is done in constant time, $(O(1))$, as these operations are standard constant time operations.

If the "if" in line 2 is not fulfilled, the program skips line 3 and moves on to line 4. It will effectuate line 5. How long time search and insert in a bid/ask-list takes depends on the size of the array. If there are few bidders in the queue, as it would be at the beginning of allocation, the time taken would be short. As the time passes, the number of bidders and sellers, and hence the size of the arrays, become large. In the worst case, that is if no allocation has ever been effectuated, the basic operations of search and insert of an unmatched bid would take $O(\log n_1).n_1$ represent the number of elements in the bid queue. In fact, the complexity of this line, taking both buyers and sellers into account is, in the worst case, $O(\log n)$.

The for-loop in line 1 would increase the complexity with a factor n because this "for" is effectuated n times. In fact The whole allocation problem of CVSE would have a worst case complexity of $O(n \log n)$, as stated in section 4.4.3.

Appendix C

Running Jobs on the Grid with gLite Middleware

We present here more information about the steps resulting in running jobs on the Grid. The Virtual Organization (VO) *blaubert* was created on our demand especially for the study of hedging of the European swaption on Grid resources. Then, this VO was added to a VOMS server (a Virtual Organization Membership Service server containing the credentials and the rights of the different VOs) and the different computer centers were encourage to allow the *blaubert* to use their computer resources.

We used gLite as middleware to submit our jobs to the Grid. gLite is a middleware developed by the international collaboration: Enabling Grids for E-sciencE (EGEE). A middleware is a necessary tool to build a grid and submitting jobs to the grid requires a special file containing job specifications. We give an example of a job description file used to submit several jobs to the grid, called job.jdl

```
// Job Description Language
[
    Type = "Collection";                              //multiple jobs at the same time
    VirtualOrganisation = "blaubert";                 //Our VO
    InputSandbox = {"script","Financial_Derivatives_Market_for_Grid_Computing.cpp","EuropeanSwaption.cpp","
        EuropeanSwaption.h","normal.cpp","normal.h"); //Input a compilation script and all the necessary
        files to compile
    StdOutput = "Std.out";
    StdError = "Std.err"
    DefaultNodeShallowRetryCount = 5;                 //Try max 5 times
    Nodes = [                                         //two nodes
        [
            Executable = "scirpt";
            OutputSandbox = {"information1.xls"};     //output results to information1.xls
        ],
        [
            Executable = "script";
            OutputSandbox = {"information2.xls"};
        ],
    ]
]
```

The script file has the command lines:

```
// the script file
//Compile, set rights, run program

#! /bin/bash
g++ Financial_Derivatives_Market_for_Grid_Computing.cpp EuropeanSwaption.cpp normal.cpp —o master.exe
/bin/chmod 777 ./master.exe
./master.exe
```

Several commands are used to communicate with the grid, where the most important are: Submit a job: *glite-wms-job-submit -output identifier.id -collection job.jdl*, where it says to do send the job *collection.jdl* to the grid and receive some job-tracking files in *identifier.id*

Check job-status: *glite-wms-job-status -input identifier.id*

Get job-output, in our case, the information.xls files: *glite-job-get-job-output -input identifier.id*

Appendix D

Implementation of Pricing and Hedging of European Swaptions in C++

We provide our implementation of the pricing and hedging of a European Swaption. We write the simulation code in C++, and for simplicity, we only show the *cpp* files and not the header files. We performed our calculations using a Grid, described in appendix C.

```
//*****************************************************************\\
//                                                                 \\
//          Financial Derivaties Market for Grid Computing         \\
//                                                                 \\
//          Purpose:                                               \\
//          Calculate pricing and hedging of European swaption     \\
//          to show how different hedging strategies affect the    \\
//          variance, maximum loss, maximum gain and the mean      \\
//                                                                 \\
//                                                                 \\
//          Programmed by:  David Aubert and                       \\
//                          Arnstein Seljeflot Solli               \\
//                          (C)2007                                \\
//                                                                 \\
//                                                                 \\
//                                                                 \\
//          version:        1.0                                    \\
//          Programmed for individual to simulate a Grid approach  \\
//                                                                 \\
//                                                                 \\
//*****************************************************************\\

// Financial Derivatives Market for Grid Computing.cpp : Defines the entry point for the console application.
//

#ifndef PI
const double PI=3.141592653589793238462643;
#endif

#include "stdafx.h"
#include <iostream>
#include <cmath>
#include <fstream>
#include <iomanip>
using namespace std;
#include <vector>
#include <ctime>
#include <string>
#include <sstream>

#include "EuropeanSwaption.h"
#include "normal.h"

int _tmain(int argc, _TCHAR* argv[]){
//int main(){

double optionInp, sigmaInp, F0Inp, KInp, rInp;
int iterationInp, nrOfTimesInp, flag;

cout << "Plase␣type␣in␣the␣Option␣expiration␣time,␣in␣days,␣from␣now␣\n␣and␣press␣the␣return␣button,␣(ex.␣
     60.0)␣" << endl;
cin >> optionInp;
cout << "Plase␣type␣in␣the␣option␣strike␣price,␣ex␣95.0"  << endl;
cin >> KInp;
cout << "Plase␣type␣in␣the␣current␣Swapt␣price,␣ex␣100.0"  << endl;
cin >> F0Inp;
```

```
cout << "Plase␣type␣in␣the␣volatility␣of␣the␣Swap,␣ex.␣0.8␣" << endl;
cin >> sigmaInp;
cout << "Plase␣type␣in␣the␣market␣interest␣rate,␣ex.␣0.05␣" << endl;
cin >> rInp;
cout << "Plase␣type␣in␣the␣number␣of␣simulations␣in␣each␣run␣\n␣you␣would␣like␣to␣perform␣(ex.␣100)␣\n(OBS␣OBS
      !␣We␣do␣not␣recommend␣more␣than␣100␣if␣it␣is␣not␣run␣on␣Grid)␣" << endl;
cin >> iterationInp;
cout << "Plase␣type␣in␣the␣number␣of␣times␣you␣would␣like␣to␣run␣the␣simulations␣\n␣(this␣is␣to␣simulate␣our␣
      Grid␣approach)␣␣" << endl;
cin >> nrOfTimesInp;

flag = 1;
    //Define option parameters
    const double& T_Option = (optionInp+0.0)/365.0; //time to maturity for the option

    //Define Swap parameters
    const double& sigma = (sigmaInp+0.0); //Assume volatile
    const double& F0 = (F0Inp+0.0); //Swap in t=0
    const double& K = (KInp+0.0); //Strike price option (and mayby the also the swap)

    //Define macro economic factors
    const double& r = rInp; //this is the intrest rate for the whole period

    //define div. variables
    int n_hedge; //# of redistribute the portfolio (# of hedges)
    double h; //a varialbe keeping track of the time between each hedging
    double H1; //the hedging position taken in the underlying swap
    double H0F; //the non hedged position
    double t = 0.0;  //time counter
    int hedging_strategies = 8; //the number of heding strategie
    double value; //final value of portfolio
    const double INF = 9999999.9;
    srand((unsigned)time(0)); //uses the time for seed in the random generator. We believe that the built-in
          generator is good enough for our illustrating purpose
    int position = 0; //use as a counting variable

    // each process having "partition" steps an we simulate "iteration" times
    int iteration = int(iterationInp); //this whole program is run several times on the Grid
    int partition = 60000;

    //text string to name the files
    string hele;

    //letting the time steps be the option expiration time divided by the numbers of partitions
    double delta_t = T_Option / partition;
    int length = 1001;  //length of the output vectors

    const double& half_sigma_sqr_pluss_r = 0.5*sigma*sigma + r; //in order to avoid calcuating later
    const double& r_T_option = r*T_Option;
    const double& r_min_half_sigma_sqpr = r-sigma*sigma/2.0;
    const double& sigma_sqrt_delta_t = sigma*sqrt(delta_t);
    const double& r_min_half_sigma_sqpr_delta_t = r_min_half_sigma_sqpr*delta_t;
    //Vectors:
    vector<double> Swap_price_process(partition);    //building a priceprocess with partition steps
    vector<double> Quadratic_mean(hedging_strategies); //helping variable for variance for each hedging
          strategy
    vector<double> Mean_calc(hedging_strategies); //a helping variable for mean for each hedging strategy
    vector<double> Mean(hedging_strategies); //mean of the different hedging strategies
    vector<double> Variance(hedging_strategies); // variance of the different hedging strategies
    vector<double> MinPayoff(hedging_strategies); // the minimum payoff
    vector<double> MaxPayoff(hedging_strategies); // the maximum payoff
    //vector<double> RepartitionHedge(hedging_strategies); // to show the repartition when nr_hedge = 60;
    vector<double> RepartitionSwap(length); // to show the repartition of swaps;

    //building a matrix which will consist of value-intervalls to show the effect of different hedging
          strategies
    vector < vector<double> > RepartitionHedge;
    RepartitionHedge.resize(hedging_strategies);
    for(int i=0 ; i < RepartitionHedge.size(); i++) {
        RepartitionHedge[i].resize(length);
    }
    double maxvalue = 0.0;

    //Value of the call swaption Black-Scholes-Merton, see appendix for derivation
    double value_option = price_european_swaption(F0,K,r, sigma, T_Option, t, half_sigma_sqr_pluss_r, flag);

    //startvalues for Hedging
    double start_hedge = delta_european_swaption(F0, K, r, sigma, T_Option, t, half_sigma_sqr_pluss_r, flag);
    double start_value_portfolio = value_option - start_hedge*F0;

    for(int nrOfTimes = 0; nrOfTimes < nrOfTimesInp; nrOfTimes++){
        cout << "Performing␣run␣nr:␣" << nrOfTimes+1 << endl;
        for (int j = 0; j < hedging_strategies; j++){
            for(int i=0 ; i < length; i++)
                RepartitionHedge[j][i] = 0;
        }
        for(int i=0 ; i < length; i++)
            Swap_price_process[i]= 0;

        t = 0.0;
        maxvalue = 0.0;

        //initialize hedging vector
        for (int l = 0; l < hedging_strategies; l++){
            Quadratic_mean[l] = 0.0;
```

```
    Mean_calc[1] = 0.0;
    MinPayoff[1] = INF;
    MaxPayoff[1] = -INF;
}

// This is the simulation to be done "iteration" times
for (int i=0; i<iteration; i++){
    value = 0.0;

    //simulate the swap path
    Swap_price_process = price_process(F0, r, sigma, delta_t, partition, r_min_half_sigma_sqpr_delta_t
          , sigma_sqrt_delta_t);

    //remembers the maximum traded end price for swap in order to find the worst outcome for value
    maxvalue = max(maxvalue, Swap_price_process[partition -1]);

    //insert in histogram for swap prices
    if(Swap_price_process[partition -1]> 400.0)
        RepartitionSwap[length -1] +=1;
    else
        RepartitionSwap[int(floor(Swap_price_process[partition -1]*10.0/4.0))] +=1;

    //Different hedging strategies
    for (int k = 0; k<hedging_strategies; k++){

        //determine which number to use for the hedge
        switch (k){
            case 0: n_hedge = 0;     break;
            case 1: n_hedge = 1;     break;
            case 2: n_hedge = 2;     break;
            case 3: n_hedge = 6;     break;
            case 4: n_hedge = 60;    break;
            case 5: n_hedge = 600;   break;
            case 6: n_hedge = 6000;  break;
            case 7: n_hedge = 60000;break;
        }

        if (n_hedge == 0){
            value = value_option - price_european_swaption(Swap_price_process[partition -1],K,r, sigma,
                  T_Option ,T_Option, half_sigma_sqr_pluss_r, flag);
            Quadratic_mean[k] = Quadratic_mean[k] + value*value;
            Mean_calc[k] = Mean_calc[k] + value;
            MinPayoff[k] = min(MinPayoff[k], value);
            MaxPayoff[k] = max(MaxPayoff[k], value);

            //insert in histogram for hedgeing
            if ( value <= -80.0)
                RepartitionHedge[k][0] +=1;
            else if (value >= 20.0)
                RepartitionHedge[k][length -1] +=1;
            else
                RepartitionHedge[k][int(floor(value*10.0)+800)] +=1;
        }

        else if(n_hedge == 1){
            value = start_hedge*Swap_price_process[partition -1]*exp(-r_T_option) +
                  start_value_portfolio - price_european_swaption(Swap_price_process[partition -1],K,r,
                  sigma, T_Option ,T_Option, half_sigma_sqr_pluss_r, flag);
            Quadratic_mean[k] = Quadratic_mean[k] + value*value;
            Mean_calc[k] = Mean_calc[k] + value;
            MinPayoff[k] = min(MinPayoff[k], value);
            MaxPayoff[k] = max(MaxPayoff[k], value);

            //insert in histogram for hedgeing
            if ( value <= -80.0)
                RepartitionHedge[k][0] +=1;
            else if (value >= 20.0)
                RepartitionHedge[k][length -1] +=1;
            else
                RepartitionHedge[k][int(floor(value*10.0)+800)] +=1;
        }

        else {
            h = T_Option/n_hedge;
            H1 = start_hedge;
            H0F = start_value_portfolio;
            for (int j=1; j<n_hedge; j++){
                t = j * h;
                position = int(t*partition/T_Option);
                value = H1 * Swap_price_process[position] + H0F * exp (r*h);
                H1 = delta_european_swaption(Swap_price_process[position], K, r, sigma, T_Option, t,
                      half_sigma_sqr_pluss_r, flag);
                H0F = value - H1*Swap_price_process[position];
            }
            value = (H1*Swap_price_process[partition -1] + H0F*exp(r*h))*exp(-r_T_option) -
                  price_european_swaption(Swap_price_process[partition -1],K,r, sigma, T_Option ,
                  T_Option, half_sigma_sqr_pluss_r, flag);

            Quadratic_mean[k] = Quadratic_mean[k] + value*value;
            Mean_calc[k] = Mean_calc[k] + value;
            MinPayoff[k] = min(MinPayoff[k], value);
            MaxPayoff[k] = max(MaxPayoff[k], value);

            //insert in histogram for hedgeing
            if ( value <= -80.0)
```

```
                        RepartitionHedge[k][0] +=1;
                    else if (value >= 20.0)
                        RepartitionHedge[k][length-1] +=1;
                    else
                        RepartitionHedge[k][int(floor(value*10.0)+800)] +=1;
                }
            }
        }

        //Files to keep the results in
        string file= "value";
        string xls = ".xls";
        string nr;
        stringstream out;
        out << nrOfTimes;
        nr = out.str();
        hele = file+nr+xls;

        //to Excell
        ofstream fichier;
        fichier.open(hele.c_str(), ios::out);

        fichier << iteration << "\n";
        fichier << maxvalue << "\n" ;

        for (int k = 0; k<hedging_strategies; k++)
            fichier << Mean_calc[k] << "\t" << Quadratic_mean[k] << "\t" << MinPayoff[k] << "\t" << MaxPayoff[
                 k] << "\n";

        for (int i = 0; i <= length-1; i++) {
            fichier << RepartitionSwap[i] << "\t" ;
            for( int k = 0; k<hedging_strategies; k++)
                fichier<< RepartitionHedge[k][i] << "\t";
            fichier << "\n";
        }
        fichier.close();
    }

    //read in, do calculations and keep the results in "information.xls"
    ifstream inFile;
    ofstream outFile;
    outFile.open("information.xls", ios::out);

    int iteration1 = 0;
    int temp_iteration;

    double hedge1 = 0.0;
    double swap1 = 0.0;
    double mean1 = 0.0;
    double quad1 = 0.0;
    double minimum1 = INF;
    double maximum1 =-INF;
    double maxSwap1 =0.0;

    vector<double> Quadratic_mean1(hedging_strategies); //helping variable for variance for each hedging
         strategy
    vector<double> Mean_calc1(hedging_strategies); //a helping variable for mean1 for each hedging strategy
    vector<double> Mean1(hedging_strategies); //mean1 of the different hedging strategies
    vector<double> Variance1(hedging_strategies); // variance of the different hedging strategies
    vector<double> MinPayoff1(hedging_strategies); // the minimum1 payoff
    vector<double> MaxPayoff1(hedging_strategies); // the maximum1 payoff
    vector<double> RepartitionSwap1(length); // to show the repartition of swaps;

    vector < vector<double> > RepartitionHedge1;
    RepartitionHedge1.resize(hedging_strategies);
    for(int i=0 ; i < RepartitionHedge1.size(); i++) {
        RepartitionHedge1[i].resize(length);
    }
    for (int l = 0; l < hedging_strategies; l++){
        Quadratic_mean1[l] = 0.0;
        Mean_calc1[l] = 0.0;
        MinPayoff1[l] = INF;
        MaxPayoff1[l] = -INF;
    }

    for(int i = 0; i < nrOfTimesInp; i++){
        string file= "value";
        string xls = ".xls";
        string nr;
        stringstream out;
        out << i;
        nr = out.str();
        hele = file+nr+xls;

        inFile.open(hele.c_str(), ios::in);

        //Read in the infile
        if(!inFile) {
        }
        else{
            inFile >> temp_iteration;
            iteration1 = iteration1 + temp_iteration;
            inFile >> maxSwap1;

            for (int k = 0; k < hedging_strategies; k++){
```

```
                inFile >> mean1 >> quad1 >> minimum1 >> maximum1;
                Mean_calc1[k] = Mean_calc1[k]+mean1;
                Quadratic_mean1[k] = Quadratic_mean1[k] + quad1;
                MinPayoff1[k] = min(MinPayoff1[k],minimum1);
                MaxPayoff1[k] = max(MaxPayoff1[k], maximum1);
            }
            for (int j = 0; j < length; j++){
                inFile >> swap1;
                RepartitionSwap1[j] = RepartitionSwap1[j] + swap1;
                for (int m = 0; m < hedging_strategies; m++){
                    inFile >> hedge1;
                    RepartitionHedge1[m][j] = RepartitionHedge1[m][j] + hedge1;
                }
            }
        }
        inFile.close();
    }
    //present in the file "information.xls"
    outFile << setprecision(4)<< "maxSwap␣\t" << maxSwap1 << "\n";
    outFile <<  "Mean" << "\t" << "Variance" << "\t" << "Max␣loss" << "\t" << "Max␣Gain" << "\n";

    for (int k = 0; k<hedging_strategies; k++){
        Mean1[k] = Mean_calc1[k]/iteration1;
        Variance1[k] = Quadratic_mean1[k]/iteration1- Mean1[k]*Mean1[k];
        outFile << setprecision(4) << Mean1[k] << "\t" << Variance1[k] << "\t" << MinPayoff1[k] << "\t" <<
              MaxPayoff1[k] << "\n";
    }

    outFile << "interval␣swap"  << "\t" <<"Number␣of␣occurences␣of␣value␣swap1"  <<  "\t"  ;
    outFile << "interval␣hedge" << "\t" << "0␣\t␣1␣\t␣2␣\t␣6␣\t␣60␣\t␣600␣\t␣6000␣\t␣60000␣" << endl;
    for (int i = 0; i <= length-1; i++) {
        outFile << i*4.0/10.0 << "\t" << RepartitionSwap1[i] << "\t" << (i-800.0)/10.0 << "\t" ;
        for( int k = 0; k<hedging_strategies; k++)
            outFile << RepartitionHedge1[k][i] << "\t";
        outFile << "\n";
    }

    //out to console
    cout << "The␣mean,␣variance,␣maximum␣loss␣and␣maximum␣gain␣is␣for␣the␣\n␣European␣swaption␣put␣";
    cout << "with␣start␣value␣" <<F0 << ",␣experiation␣in␣"<< optionInp <<"␣days,␣\n␣with␣start␣swap␣price␣"<<
          F0Inp <<",␣strike␣"<< KInp <<",␣volatility␣"<< sigmaInp<<",\n␣and␣interest␣rate␣"<<rInp <<",␣\n\n";
    cout << "Hedges␣" << "\t\t" << "Mean" << "\t␣\t"<< "Variance" << "\t" << "Max␣loss" << "\t" << "Max␣Gain"
         << "\n";

    for (int k = 0; k<hedging_strategies; k++){
        Mean1[k] = Mean_calc1[k]/iteration1;
        Variance1[k] = Quadratic_mean1[k]/iteration1- Mean1[k]*Mean1[k];
        switch (k){
                    case 0: cout << "␣Naked␣␣\t";        break;
                    case 1: cout << "␣Immediate␣\t";     break;
                    case 2: cout << "␣2␣times␣\t";       break;
                    case 3: cout << "␣6␣times␣\t";       break;
                    case 4: cout << "␣60␣times␣\t";      break;
                    case 5: cout << "␣600␣times␣\t";     break;
                    case 6: cout << "␣6,000␣times␣\t";   break;
                    case 7: cout << "␣60,000␣times␣\t"; break;
        }

        cout << left << scientific << setprecision(1) << showpos << Mean1[k] << "\t"<< left << Variance1[k] <<
              "\t"<< left << MinPayoff1[k] << "\t" << left << MaxPayoff1[k] << "\n";
    }
    cout << "\n␣And␣the␣price␣of␣the␣option␣is:␣" << fixed << setprecision(2) << value_option <<"\n";
    cout <<  "\n␣For␣more␣information␣of␣the␣repartition␣of␣the␣swap␣prices␣\n␣or␣repartition␣of␣hedging␣
         results,␣\n␣see␣the␣excel␣document␣named␣\"information.xls\"␣";

    cout << "\n␣\n␣Press␣any␣key,␣and␣than␣the␣enter␣key";
        string a;
        cin >> a;
    return 0;
}


#include "stdafx.h"
#include <iostream>
#include <cmath>
#include <fstream>
#include <iomanip>
using namespace std;
#include <vector>
#include "normal.h"

double d1 ( const double& Swap,
            const double& K,
            const double& r,
            const double& sigma,
            const double& time,
            const double& half_sigma_sqr_pluss_r) {
    double dOne;
    if (time <= 0.0)
    dOne = -100000; //in order to avoid fault with "double"
    else
    dOne =( log(Swap/K) + (half_sigma_sqr_pluss_r)*time)/(sigma*sqrt(time));

    return dOne;
};
```

```
double d2 ( const double& Swap,
            const double& K,
            const double& r,
            const double& sigma,
            const double& time,
            const double& half_sigma_sqr_pluss_r) {
double dTwo;
dTwo = d1( Swap, K, r, sigma, time, half_sigma_sqr_pluss_r) - sigma*sqrt(time);

return dTwo;
};


double price_european_swaption( const double& Swap,
                               const double& Kp,
                               const double& r,
                               const double& sigma,
                               const double& time_option,
                               const double& time,
                               const double& half_sigma_sqr_pluss_r,
                               int& flag){

    double a;
    double b;
                                   double pris = 0.0;
    if(flag == 1)
        pris = (Kp*N( -d2( Swap, Kp, r, sigma, time_option-time, half_sigma_sqr_pluss_r))*exp(-r*(time_option-
              time))- Swap*N( -d1( Swap, Kp, r, sigma, time_option-time, half_sigma_sqr_pluss_r )));
    if(flag == 2){
        //pris = Swap*N( -d1( Swap, Kp, r, sigma, time_option-time, half_sigma_sqr_pluss_r ))-Kp*N( -d2( Swap,
              Kp, r, sigma, time_option-time, half_sigma_sqr_pluss_r))*exp(-r*(time_option-time));
    a = Swap*N( d1( Swap, Kp, r, sigma, time_option-time, half_sigma_sqr_pluss_r ));
    b = -Kp*N( d2( Swap, Kp, r, sigma, time_option-time, half_sigma_sqr_pluss_r))*exp(-r*(time_option-time));
    pris = a + b;
    }
    if (pris < 0)
        pris = 0.0;

    return pris;
};

double delta_european_swaption(const double& Swap,
                               const double& Kp,
                               const double& r,
                               const double& sigma,
                               const double& time_option,
                               const double& time,
                               const double& half_sigma_sqr_pluss_r,
                               int& flag){

    double delta = 0.5;
    double k = 0.01;
    double dOne = d1( Swap, Kp, r, sigma, time_option-time, half_sigma_sqr_pluss_r);

    if(flag == 1){
    if (dOne > 0.0)
        delta = N(dOne) -1.0;
    else if (dOne < 0.0)
        delta = -N(-dOne);
    }
    if(flag == 2){
        if (dOne > 0)
            delta = N(dOne);
        else if (dOne < 0)
            delta = 1.0-N(-dOne);
    }
    return delta;
};


// create the price process vector, see appendix for derivation
vector<double> price_process(const double& F0,
                             const double& r,
                             const double& sigma,
                             const double& delta_t,
                             const int& partition,
                             const double& r_min_half_sigma_sqpr_delta_t,
                             const double& sigma_sqrt_delta_t){
    vector<double> prices(partition);
    double F_t = F0; // initialization
    for (int i=0; i<partition; ++i) {
        F_t = F_t * exp(r_min_half_sigma_sqpr_delta_t + sigma_sqrt_delta_t * gaussienne());
    prices[i]=F_t;
    };
return prices;
};


#include "stdafx.h"
#include <iostream>
#include <cmath>
#include <fstream>
#include <iomanip>
using namespace std;
#include <vector>
```

```
#ifndef PI
const double PI=3.14159265358979323846264 3;
#endif


double n( const double& x){

double norm = 1/sqrt(2*PI)*exp(-(x*x)/2);
    return norm;
};

// Univariate cumulative normal distribution
// from Hull
double N(const double& x) {
    double p  =  0.2316419;
    double b1 =  0.319381530;
    double b2 = -0.356563782;
    double b3 =  1.781477937;
    double b4 = -1.821255978;
    double b5 =  1.330274429;
    double ta =  1/(1+p*x);
    double norm;

if (x >= 0)
    norm = 1 - n(x)*(b1*ta+b2*pow(ta,2)+b3*pow(ta,3)+b4*pow(ta,4)+b5*pow(ta,5));
else
    norm = 1-N(-x);

return norm;
};
long double gaussienne(void) //calculate a standard gausien variable
{

    double g;
    double two;
    double one;
    g = 9999999999999.0;     //Utilise a NaN in the begining

    while ((g > 100000000) || (g < -100000000)){

    two = rand()/double(RAND_MAX);
    one = rand()/double(RAND_MAX);

    g = sqrt(-2.0 * log(one)) * cos(2.0 * PI * two);
    };
    return g;
};
```

List of Figures

List of Tables

List of Abbreviations

AOA	Absence of Arbitrage
bps	bits per second
CAP	Capacity Allocation Problem
CERN	European Organization for Nuclear Research
CPU	Central Processing Unit
CTO	Chief Technology Officer
EGEE	Enabling Grid for E-SciencE
EMH	Efficient Market Hypothesis
EMM	Equivalent Martingale Measure
ERP	Enterprise Resource Planning
EU	European Union
FP	Framework Program
GBM	Geometric Brownian Motion
GDP	Gross Domestic Product
HaaS	Hardware-as-a-Service
HEP	High Energy Physics
ICT	Information and Communication Technology
IT	Information Technology
LEP	Large Electron Positron
LHC	Large Hadron Collider
NP	Non-deterministic Polynomial
OGF	Open Grid Forum
OSG	US Open Science Grid
P	Polynomial
PDA	Personal digital assistant
REH	Rational Expectancy Hypothesis
ROI	Return On Investment
SaaS	Softwarre-as-a-Service
SDE	Stochastic Differential Equation
TT	Technology Transfer
VM	Virtual Machine
VO	Virtual Organization
VS	Virtual Server
WLCG	Worldwide LHC Computing Grid

Bibliography

A1-telecom (2007). HSDPA. http://www.a1.net/business/hsdpa. Retrieved 2007.10.03.

About.com:Google (2007). Cloud definition. http://google.about.com/od/c/g/clouddef.htm. Retrieved 2007.09.20.

Accenture (2004). Grid Computing, Harnessing the power of many to deliver high performance. http://www.accenture.com/NR/rdonlyres/B134ACE7-EBCF-49CE-855C-DB7C0158280C/0/grid_computing.pdf. Retrieved 2007.09.11.

Ahn, N., Black, J., & Effrat, J. (2007). Feature - open grid forum maintains focus on the 2010 goal. Stanford University Computer Science Department, http://cse.stanford.edu/class/sophomore-college/projects-01/distributed-computing/html/history.html. Retrieved 2007.11.28.

Armijo, B., & Nickolov, P. (2007). The 7 services virtualization lacks for utility computing. FishTrain, http://fishtrain.com/2007/11/15/the-7-services-virtualization-lacks-for-utility-computing/. Retrieved 2007.12.01.

Association, N. F. (2006). Opportunity and RISK, an educational guide to trading futures and options on futures. National Futures Association.

Aubert, D., & Solli, A. S. (2007). The Grid, Business Opportunities with CERN Open Source Software. Conference: Microcosm - CERN for Industry, 24th-25th of September 2007, Wroclaw Technology Park, by Arnstein Seljeflot Solli. Conference: Man Financial, Paris - 17 October 2007, by Jean-Marie Le Goff.

Bachelier, L. J.-B. A. (1900). Théorie de la spéculation. *Annales scientifiques de l'École Normale Supérieure*, *3*(17), 21–86.

Beaver, C. (2007). The design of an electricity derivatives market. BITS Ltd., www.bitscanada.com. , Retrieved 2007.11.30.

Berlich, D. R. (2006). Grid Computing: New Frontiers. National Grid Singapore, 2006 seminar, http://www.ngp.org.sg/seminars/. Retrieved 2007.08.26.

Berners-Lee, T. (2007). Testimony of Sir Timothy Berners-Lee. Hearing on the Digital Future of the United States: Part 1 - The Future of the World Wide Web, http://dig.csail.mit.edu/2007/03/01-ushouse-future-of-the-web. Retrieved 2007.09.20.

Björk, T. (2004). *Arbitrage Theory in Continuous Time* (2 edition). Oxford University Press.

Black, F., & Scholes, M. (1973). The pricing of options and corporate liabilities. *The Journal of Political Economy*, *81*(3), 637–654.

Blaizot, J., Iliopoulos, J., Madsen, J., Ross, G., Sonderegger, P., & Specht, H. J. (2003). Study of Potentially Dangerous Events During Heavy-ion Collisions at the LHC:Report of the LHC Safety Study Group. CERN, http://doc.cern.ch/yellowrep/2003/2003-001/p1.pdf. Retrieved 2007.07.22.

Bressan, B., Streit-Bianchi, M., & Marcastel, F. (2005). *CERN Technology Transfers to Industry and Society*. CERN TT Group.

Burger, M., Klar, B., Müller, A., & Schindlmayr, G. (2004). A spot market model for pricing derivatives in electricity markets. *Quantitative Finance*, *4*(1), 109–122.

Buszaa, W., Jaffe, R., Sandweiss, J., & Wilczek, F. (1999). Review of Speculative Disaster Scenarios at RHIC. CERN, http://doc.cern.ch//archive/electronic/hep-ph/9910/9910333.pdf. Retrieved 2007.07.22.

Buyya, R. (2002). *Economic-based Distributed Resource Management and Scheduling for Grid Computing*. Ph.D. thesis, Monash University, Melbourne, Australia.

Buyya, R., Abramson, D., & Giddy, J. (2000). An economy grid architecture for service-oriented grid computing. Proceedings of the 7th International Conference on Parallel and Distributed Processing Techniques and Applications, Las Vegas, USA, June 26-29, 2000.

Carlton, D. (1984). Futures markets: Their purpose, their history, their growth, their successes and failures. *Journal of Futures Markets*, *4*(3), 237–271.

Carr, P., Geman, H., Madan, D. B., & Yor, M. (2002). The fine structure of asset returns: An empirical investigation. *The Journal of Business*, *75*(2), 305–332.

Casasús, C. (2007). Mexico's CUDI Network. ICFA Digital Divide Workshop at Universidad Iberoamericana, Mexico City, 24 October 2007, http://indico.cern.ch/conferenceDisplay.py?confId=22909. Retrieved 2007.12.01.

CERN (2004). L'histoire du cern : un voyage de 50 années au coeur de la matière. CERN and Mannmade Productions. CERN Copyright Reference: CERN-MOVIE-2004-006.

CERN (2007a). Cern's public website. http://public.web.cern.ch/. Retrieved 2007.11.28.

CERN (2007b). Grid café. http://www.gridcafe.org. Retrieved 2007.11.30.

CERN (2007c). The LHC, Frequently asked questions. http://public.web.cern.ch/Public/Content/Chapters/AskAnExpert/LHC-en.html#q3. Retrieved 2007.11.30.

CERN (2007d). Technology transfer mandate. http://dsu.web.cern.ch/dsu/groups/TT/TT.html. Retrieved 2007.11.30.

Chetty, M., & Buyya, R. (2002). Weaving computational grids: how analogous are they with electricalgrids?. *Computing in Science & Engineering*, *4*(4), 61–71.

Claunch, C., Weiss, G., & McGuckin, P. (2007). Hype cycle for server technologies 2007. Gartner Research, ID Number: G00144727. Retrieved 2007.09.19.

Cohen, R. (2003). Grid Computing Projected Impact on North Carolina's Economy & Broadband Use through 2010. Rural Internet Access Authority Research Report, http://www.e-nc.org/pdf/grid_report_oct-28-03.pdf. Retrieved 2007.09.11.

Convention, C. (1953). Convention for the establishment of a european organization for nuclear research. Paris, 1st July, 1953, http://dsu.web.cern.ch/dsu/ls/conventionE.htm.

CORDIS (2004). Strengthening Competitiveness Through Co-operation, European Research in Information and Communication Technologies. EU's Community Research & Development Information Service, ftp://ftp.cordis.europa.eu/pub/ist/docs/strengthening-european-research-in-ict.pdf. Retrieved 2007.11.26.

CORDIS (2007a). ICT - Information and Communication Technologies. EU's Community Research & Development Information Service , http://cordis.europa.eu/fp7/ict/faq_en.html. Retrieved 2007.11.26.

CORDIS (2007b). Seventh Research Framework Programme (FP7). EU Publications Office, Community Research & Development Information Service , http://cordis.europa.eu/fp7/understand_en.html. Retrieved 2007.11.26.

CORDIS (2007c). What is a Grid?. EU Publications Office, Community Research & Development Information Service, http://ec.europa.eu/employment_social/knowledge_society/index_en.htm. Retrieved 2007.09.19.

Crockford, N. (1986). *An Introduction to Risk Management* (2 edition). Woodhead-Faulkner.

Cuny, C. J. (1993). The role of liquidity in futures market innovations. *Review of Financial Studies*, *6*(1), 57–78.

Currie, W. L., & Seltsikas, P. (2001). Exploring the supply side of it outsourcing: Evaluating the emerging role of application service providers. *European Journal of Information Systems*, *10*(3), 123–134.

Da Rold, C., Margevicius, M. A., Cohen, L. R., & Bittman, T. J. (2007). Alternative delivery models: A sea of new opportunities and threats. Gartner Research, ID Number: G00152244. Retrieved 2007.11.20.

DANTE (2007). Delivery of Advanced Network Technology to Europe (DANTE). DANTE Web site, http://www.dante.net/. Retrieved 2007.12.03.

Darlington, J., & Newhouse, S. (2006). A Market for Computational Services. Proposal to the e-Science Core Technology Programme. The London e-Science Centre and The CLRC e-Science Centre and The North West e-Science Centre and The Southampton e-Science Centre and The University of Wales at Swansea and The Astrophysics Research Institute at Liverpool and John Moores University, in collaboration with Computer Sciences Corporation, Numerical Algorithms Group, GridX, RealTime Engineering Ltd., Sun Microsystems Ltd., SGI Inc., Oracle.

DataSynapse (2007). Case study: Bnp paribas. http://datasynapse.com/case-studies-BNP-Paribas.aspx. Retrieved 2007.11.21.

de Gennaro, S. (2003). DataGrid: computing power and storage for next-generation scientific research. datagrid, le projet européen de grille de calcul et de données. BUL Collection.

Deng, S., & Oren, S. (2006). Electricity derivatives and risk management. *Energy*, *31*(6-7), 940–953.

Dornan, A. (2007). The End Of The Line For Licensing. InformationWeek, December 3, 2007 issue, http://www.informationweek.com/news/showArticle.jhtml?articleID=204400240. Retrieved 2007.12.06.

EGEE (2007a). Collaboration list. http://egee-technical.web.cern.ch/egee-technical/related-projects/list-projects.htm. Retrieved 2007.11.27.

EGEE (2007b). Enabling Grids for E-Science. http://www.eu-egee.org/. Retrieved 2007.11.23.

Ehrig, J. (2007). Feature - open grid forum maintains focus on the 2010 goal. iSGTW, International Science Grid This Week, October 17, 2007, http://www.isgtw.org/?pid=1000710. Retrieved 2007.11.28.

Electronista (2007). AT&T, Verizon, Vodafone to share same 4G network. http://www.electronista.com/articles/07/09/21/verizon.and.vodafone.4g/. Retrieved 2007.10.03.

Embrecht, P. (2003). The wizards of wall street : did mathematics change finance?. *Nieuw Archief voor Wiskunde*, *4*(1), 26–33.

EU (2000). European Council 23 and 24 march 2000 Lisbon, Presidency Conclusions. European Parliament, http://www.europarl.europa.eu/summits/lis1_en.htm. Retrieved 2007.11.26.

EU (2005). i2010 - A European Information Society for growth and employment. Commission of the European Communities, http://eur-lex.europa.eu/LexUriServ/site/en/com/2005/com2005_0229en01.pdf. Retrieved 2007.11.26.

EU (2006). ICT in FP7, At A Glance. European Commission, http://www.ehealthnews.eu/content/view/318/62/. Retrieved 2007.11.26.

EU (2007). ICT - Information and Communication Technologies, Work Programme 2007-08. European Commission, ftp://ftp.cordis.lu/pub/fp7/ict/docs/ict-wp-2007-08_en.pdf. Retrieved 2007.09.19.

Fama, E. F. (1965). The behavior of stock-market prices. *The Journal of Business*, *38*(1), 34–105.

Fama, E. F. (1970). Efficient capital markets: a review of theory and empirical work. *Journal of Finance*, *25*(2), 383–417.

Farber, D., Dignan, L., & Berlind, D. (2007). Google: Rewriting the book on data centers. ZDnet, December 5th, 2007, http://blogs.zdnet.com/BTL/?p=7254. Retrieved 2007.12.06.

Foster, I., & Kesselman, C. (2003). *The Grid 2: Blueprint for a New Computing Infrastructure* (2 edition). Morgan Kaufmann.

Friedman, D. P., & Rust, J. (1993). *The Double Auction Market: Institutions, Theories, and Evidence*. Westview Press.

Gayek, P., Nesbitt, R., Pearthree, H., Shaikh, A., & Snitzer, B. (2004). A web content serving utility. *IBM Systems Journal*, *43*(1).

GÉANT2 (2007). GÉant2 web site. http://www.geant2.net. Retrieved 2007.10.03.

Gedling, F. (2007). Grid in the commercial world. CERN Computing Colloquium, 21 November 2007, http://indico.cern.ch/getFile.py/access?resId=1&materialId=slides&confId=23570. Retrieved 2007.11.22.

Gentzsch, W. (2007). Grid initiatives: Lessons learned and recommendations. The Renaissance Computing Institute (RENCI), http://www.renci.org/publications/reports.php. Retrieved 2007.11.27.

Gerrity, J. (2007). The research network-grid link. *GRIDtoday*, *6*(12).

Gilardoni, S., & Metral, E. (2007). Accelerator physics - Introductory lecture. CERN 2007.07.05. CERN AB-ABP department.

Glider, G. (2006). The information factories. WIRED, Issue 14.10.

Gomoluch, J., & Schroeder, M. (2003). Market-Based Resource Allocation for Grid Computing: A Model and Simulation. 1st International Workshop on Middleware for Grid Computing, Co-located with Middleware 2003, Rio de Janeiro, Brazil Tuesday, June 17th 2003. Dept. of Computing, City University, London, UK.

Greenspan, A. (2002). International financial risk management. Speeches of Federal Reserve Board Members. speech before the Council on Foreign Relations, Washington, D.C.

Group, C. C. (2005). The Large Hadron Collider. CERN-Brochure-2005-002-Eng.

Group, T. . (2006). The business case for next generation IT architecture: Grid Computing Now!. Steve Wallage, Research Director, net meeting, 5. Dec, 2006, http://brighttalk.com/comm/gridcomputingnow/325a2ebf71-2473-335-2284#. Retrieved 2007.08.28.

Group, W. C. (2007). Future mimo-wlan. http://www.nari.ee.ethz.ch/wireless/research/projects.html. Retrieved 2007.11.28.

Harris, D. (2007a). Datasynapse, Capgemini Align Around Outsourcing. *GRIDtoday*, *6*(36).

Harris, D. (2007b). How Johnson & Johnson Took Its Grid Global. *GRIDtoday*, *6*(11).

Harris, D. (2007c). My day at sc07 ... and a whole lot of news. *GRIDtoday*, *6*(46).

Harris, D. (2007d). On the Origin of Utility Computing. *GRIDtoday*, *6*(38).

Harrison, M. J., & Pliska, S. R. (1981). Martingales and stochastic integrals in the theory of continuous trading. *Stochastic Processes and their Applications*, *11*(3), 215–260.

Haug, E. G. (2005a). Know your weapon part 1. Wilmott Magazine, May. Presented at Columbia University 2004 New York, Presented at CQF London., Updated version in Best of Wilmott 2, 2005 Wiley Publishing.

Haug, E. G. (2005b). Know your weapon part 2. Wilmott Magazine, July. Presented at Columbia University 2004 New York, Presented at CQF London, Updated version in Best of Wilmott 2, 2005 Wiley Publishing.

Hermann, A., Krige, J., Mersits, U., & Pestre, D. (1987). *Volume I History of CERN*. North-Holland Physics Publishing.

Hewlett-Packard (2007). Grid computing. http://h71028.www7.hp.com/enterprise/cache/125369-0-0-0-121.html. Retrieved 2007.11.28.

HighScalability (2007). Google Architecture. http://highscalability.com/google-architecture. Retrieved 2007.09.11.

Hämmerle, H., Ménard, C., Sutton, C., & Gilles, J. (2004). *Infinitely CERN Memories of fifty years of research*. CERN and Editions Suzanne Hurter.

Hull, J. C. (2005). *Options, Futures and Other Derivatives* (Sixth edition). Prentice Hall.

IBM (2006). Airbus UK gets new, collaborative ways to meet project demands, lower cost. http://www-306.ibm.com/software/success/cssdb.nsf/CS/CSAL-6VMLYC?OpenDocument&Site. Retrieved 2007.12.01.

IBM (2007a). IBM Grid computing. http://www-03.ibm.com/grid/index.shtm. Retrieved 2007.11.28.

IBM (2007b). IBM Introduces Ready-to-Use Cloud Computing. http://www-03.ibm.com/press/us/en/pressrelease/22613.wss. Press release, Armonk, N.Y. and Shanghai, China.

IBM (2007c). What is grid computing. http://www-03.ibm.com/grid/about_grid/what_is.shtml. Retrieved 2007.08.28.

IBM (2007d). Where is the Data Center Going? Top Trends Today. http://www-03.ibm.com/systems/virtualization/view/101807.html. Retrieved 2007.11.20.

IEETA (2003). Prospective Study on Grid Technology. Tech. rep. inforgenmed IST-2001-39013, IEETA - Instituto de Engenharia Electrónica e Telemática de Aveiro, Campus Universitário de Santiago, PORTUGAL.

Intel (2007). Grid Computing looking forward. http://www.intel.com/cd/ids/developer/asmo-na/eng/202676.htm. Retrieved 2007.11.28.

InternetWorldStats (2007). World Internet Stats 2Q 2007. http://www.internetworldstats.com/stats.htm. Retrieved 2007.11.19.

Jensen, M. C. (1978). Some anomalous evidence regarding market efficiency. *Journal of Financial Economics*, *6*(2/3), 95–101.

Jorion, P. (2005). *Financial Risk Manager Handbook* (3 edition). Wiley Finance.

Klemperer, P. (1999). Auction theory: A guide to the literature. *Journal of Economic Surveys*, *13*(3), 227–286.

Kumar, R., & Song, X. (2005). Gridlite: A framwork for managing and provisioning services on grid-enabled resources limited devices. Mobile and Media Systems Laboratory, HP Laboratories Palo Alto, HPL-2005-146.

LaFollette, G. L. (2007). I may have my head in the Clouds, but i have my feet well set on the ground. The CPA Technology Advisor, December 4, 2007,http://www.cpatechnologyadvisor.com/print/The-CPA-Technology-Advisor/I-May-Have-My-Head-In-The-Clouds–But-I-Have-My-Feet-Well-Set-On-The-Ground-/1$1853. Retrieved 2007.12.03.

Lamberton, D., & Lapeyre, B. (2000). *Introduction au calcul stochastique appliqué à la finance*. Ellipses-Marketing.

Le Goff, J.-M. (2007). Impact of CERN technology on society. Conference Microcosm - CERN for Industry, Wroclaw, Poland - 24 and 25 September 2007.

Lucas, R. E., & Prescott, E. C. (1971). Investment under uncertainty. *Econometrica*, *39*(5), 659–681.

MacKay, C. (1841). *Memoirs of Extraordinary Popular Delusions*. London: Richard Bentley, New Burlington Street. Publisher in Ordinary to Her Majesty.

Mandelbrot, B. B. (1963). The variation of certain speculative prices. *The Journal of Business*, *36*(4), 394–419.

Mandelbrot, B. B. (1971). When can prices be arbitraged efficiently? a limit to the validity of the random walk and martingale models. *The Review of Economics and Statistics*, *53*(3), 225–236.

Martin, R. (2007). The Red Shift Theory. InformationWeek, August 20, 2007 issue, http://www.informationweek.com/news/showArticle.jhtml?articleID=201800873. Retrieved 2007.12.01.

Mattmann, C. A., & Medvidovic, N. (2006). The grid lite dream: Bringing the grid to your pocket. Proceedings of the Monterey Workshop on Networked Systems 2006.

Merriam-Webster (2007). Agent. www.merriam-webster.com. Retrieved 2007.11.11.

Merton, R. C. (1973). Theory of rational option pricing. *The Bell Journal of Economics Management Science*, *4*(1), 141–183.

Meyer, T. (2003). Grid computing: Conceptual flyover for developers, what a developer needs to know to get started. IBM, http://www.ibm.com/developerworks/grid/library/gr-fly.html. Retrieved 2007.09.12.

Miller, M. S., & Drexler, E. K. (1988a). Comparative ecology: A computational perspective. In Huberman, B. A. (Ed.), *The Ecology of Computation*, pp. 51–76. Elsevier Science Inc.

Miller, M. S., & Drexler, E. K. (1988b). Incentive engineering for computational resource management. In Huberman, B. A. (Ed.), *The Ecology of Computation*, pp. 231–266. Elsevier Science Inc.

Miller, M. S., & Drexler, E. K. (1988c). Markets and computation: Agoric open systems. In Huberman, B. A. (Ed.), *The Ecology of Computation*, pp. 133–176. Elsevier Science Inc.

Mitchell Waldrop, M. (2007). Data center in a box. Scientific American Magazine.

Morgan, T. P. (2006). Gartner predicts half of data centers will run out of power by 2008. *The Four Hundred*, *15*(48).

Mouche, P. (2006). Overall view of the LHC. vue d'ensemble du LHC..

Muth, J. F. (1961). Rational expectations and the theory of price movements. *Econometrica*, *29*(3), 315–335.

Nakai, J. (2002). Pricing Computing Resources: Reading Between the Lines and Beyond. Tech. rep. NAS-01-010, National Aeronautics and Space Administration (NASA), Ames Research Center, Advanced Supercomputing Division, Advanced Management Technology, Incorporated.

Nicolaisen, J., Petrov, V., & Tesfatsion, L. (2001). Market power and efficiency in a computational electricity market with discriminatory double-auction pricing. *IEEE Transactions on Evolutionary Computation*, *5*(5), 504–523.

Oracle (2007). Oracle Grid Computing. http://www.oracle.com/technologies/grid/index.html. Retrieved 2007.11.28.

Ozzie, R. (2007). Microsoft, Financial Analyst Meeting 2007. http://www.microsoft.com/msft/speech/FY07/OzzieFAM2007.mspx. Retrieved 2007.09.11.

Patel, S. (2007). The clustered revolution: a paradigm shift in the storage industry. Continuity Central, http://www.continuitycentral.com/feature0492.htm. Retrieved 2007.12.01.

Rappa, M. A. (2004). The utility business model and the future of computing services. *IBM Systems Journal*, *43*(1).

Rausser, G. C., & Bryant, H. L. (2004). Contract market viability. Proceedings of the NCR-134 Conference on Applied Commodity Price Analysis, Forecasting, and Market Risk Management. St. Louis, MO.

Ross, J. W., & Westerman, G. (2004). Preparing for utility computing: The role of it architecture and relationship management. *IBM Systems Journal*, *42*(1).

Samuelson, P. A. (1965a). Proof that properly anticipate prices fluctuate randomly. *Industrial Management Review*, *6*(2), 41–50.

Samuelson, P. A. (1965b). Rational theory of warrant pricing. *Industrial Management Review (IMR)*, *6*(2), 13–31.

Scheier, R. L. (2007). Special report: 10 cool cutting-edge technologies on the horizon now, cleversafe's dispersed storage. Computerworld 23.08.2007, http://www.computerworld.com.au/index.php/id;1079326180;pp;5. Retrieved 2007.09.19.

Schnizler, B. (2007). *Resource Allocation in the Grid, A Market Engineering Approach*. Ph.D. thesis, Universität Fridericiana zu Karlsruhe, Studies on eOrganisation and Market Engineering 6.

Siems, T. F. (1997). Policy analysis, 10 myths about financial derivatives. CATO Institute.

Skantze, P., & Ilic, M. (2000). The joint dynamics of electricity spot and forward markets: Implications on formulating dynamic hedging strategies. Energy Laboratory, Massachusetts Institute of Technology. Publication no.: MIT EL 00-005.

Smith, A. (1776). An inquiry into the nature and causes of the wealth of nations. MetaLibri Digital Library 29. May 2007, http://metalibri.incubadora.fapesp.br/portal/authors/AdamSmith. Retrieved 2007.12.01.

Smith, J. E., & Nair, R. (2005). The architecture of virtual machines. *IEEE Computer Society*, *38*(5), 32–38.

Smith, W., Foster, I., & Taylor, V. (2000). Scheduling with advanced reservations. Proceedings of the IPDPS Conference. conference paper.

Stevens, R., Woodward, P., DeFanti, T., & Catlett, C. (1997). From the I-WAY to the National Technology Grid. *Communications of the ACM*, *40*(11), 50–60.

Strong, P. (2007). Interesting times for distributed datacenters. *GRIDtoday*, *6*(19).

Sulistio, A., & Buyya, R. (2004). A Grid Simulation Infrastructure Supporting Advance Reservation. Proceedings of the 16th International Conference on Parallel and Distributed Computing and Systems. MIT Cambridge, Boston, USA.

SUN (2007). , the Sun Grid Compute Utility. SUN Microsystems, http://www.sun.com/service/sungrid/index.jsp. Retrieved 2007.11.28.

SuperbInternet (2007). Grid Computing - Hype or Next Generation of Hosting. http://blog.superbhosting.net/2007/09/11/grid-computing-%E2%80%93-hype-or-next-generation-of-hosting/. Retrieved 2007.09.12.

Sutherland, I. E. (1968). A futures market in computer time. *Communications of the Association for Computing Machinery (ACM)*, *11*(6), 449–451.

Tan, Z. (2007). *Market-based Grid Resource Allocation Using a Stable Continuous Double Auction*. Ph.D. thesis, University of Manchester.

TheOpenSourceInitiative (2007). The Open Source Definition. http://www.opensource.org/docs/definition.php. Retrieved 2007.11.29.

UCLA (1969). UCLA to be first station in nationwide computer network. University of California, Los Angeles, Office of Public Information, Release, http://www.lk.cs.ucla.edu/LK/Bib/REPORT/press.html. Retrived 2007.08.28.

Ulaby, F. T. (2006). The legacy of moore's law. *Proceedings of the IEEE*, *94*(7), 1251 – 1252.

Vassdal, T. (1995). Er det mulig å etablere et futures-marked for laks?. Prosjektrapport nr. 5, NFR Prosjekt: Metoder for å redusere prisusikkerhet i laksemarkedene. Prosjektnummer NFR: 10487/110.

von Rüden, D. W. (2006). Grid computing - a new tool for science. ITU RRC-06 conference. CERN IT Department.

Vyssotsky, V. A., Hill, M., Corbató, F. J., & Graham, R. M. (1965). Structure of the multics supervisor. Joint Computer Conference, http://www.multicians.org/fjcc3.html. Retrived 2007.12.01.

Vytelingum, P. (2006). *The Structure and Behaviour of the Continuous Double Auction*. Ph.D. thesis, University of Southampton, School of Electronics and Computer Science Intelligence, Agents, Multimedia Group.

Wallage, S., & Fellows, W. (2007). Grids: the means to what ends?. The 451 Group at OGF20, May 2007, http://www.ogf.org/OGF20/materials/709/OGF20_GMB_William_Fellows.ppt. Retrieved 2007.11.28.

Wallom, D. (2007). Opinion - Grids can be green. iSGTW, International Science Grid This Week, 29 August 2007, http://www.isgtw.org/?pid=1000618. Retrieved 2007.12.01.

Wardley, S. (2007a). Commoditisation of IT. http://swardley.blogspot.com/2007/10/fowa-video.html. Retrieved 2007.11.28.

Wardley, S. (2007b). Competition, not greed, is good. BITS OR PIECES?, http://swardley.blogspot.com/2007/07/competition-not-greed-is-good.html. Retrieved 2007.11.20.

Wellner, R., & Weissberger, A. (2005). Report predicts grid market to hit $19 billion by 2010. *GRIDtoday*, *4*(6).

Weng, C., Li, M., & Lu, X. (2007). Grid resource management based on economic mechanisms. *The Journal of Supercomputing*, *42*(2), 181–199.

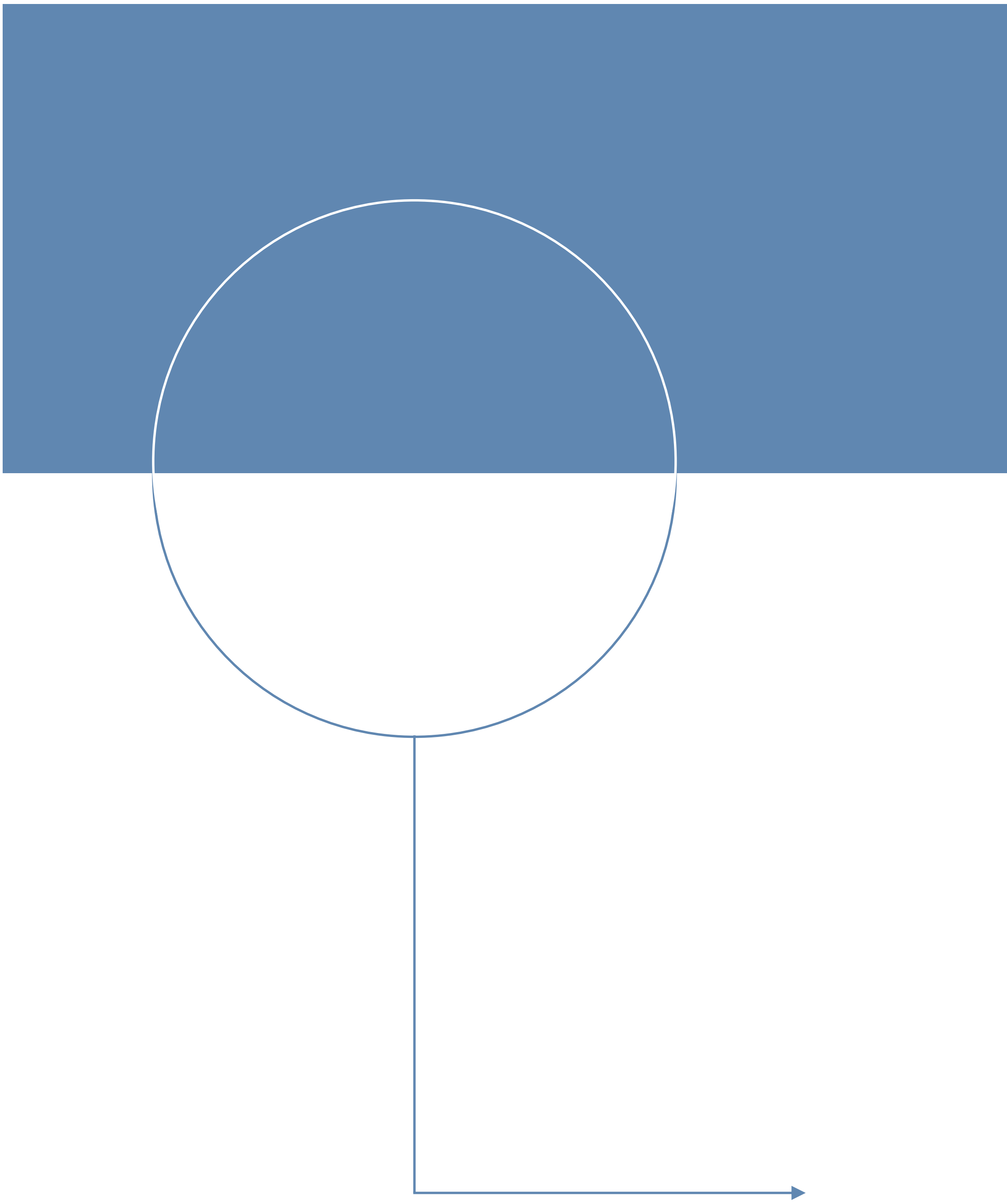
Weron, R. (2000). Energy price risk management. *Physica A: Statistical Mechanics and its Applications*, *285*(1-2), 127–134.

Willard, C. G., Joseph, E., & Lamy, L. (2007). An overview of compute - intensive utility computing. IDC white paper, http://www-03.ibm.com/servers/deepcomputing/cod/pdf/idcutilitycomputingwhitepaper.pdf. Retrieved 2007.12.01.

Wittmann, A. (2007). The cold, green facts. InformationWeek September 1, 2007, http://www.informationweek.com/hardware/showArticle.jhtml?articleID=201803326. Retrieved 2007.12.01.

WorldWideWebSize.com (2007). Daily estimated size of the World Wide Web. http://www.worldwidewebsize.com/. Retrieved 2007.11.19.

Yoo, M. (2007). Overview of high performance computing in the financial industry. Computing for Finance Conference,CERN 21 November 2007, http://indico.cern.ch/getFile.py/access?resId=0&materialId=slides&confId=23575. Retrieved 2007.11.22.



NTNU
Innovation and Creativity