Journal of International Information Management

Volume 7 | Issue 1 Article 7

1998

# Intermediation on electronic markets: The case of Java software elements

Peter Buxmann
*Institute of Information Systems*

Frank Rose
*Institute of Information Systems*

Wolfgang Konig
*Institute of Information Systems*

Johann Wolfgang Goethe
*University Frankfurt*

# Intermediation on electronic markets: The case of Java software elements

**Peter Buxmann**
**Frank Rose**
**Wolfgang König**
**Institute of Information Systems**

**Johann Wolfgang Goethe**
**Universität, Frankfurt**

## ABSTRACT

*This paper provides an introduction to the Java Repository - an intermediary for Java software elements. First, basic concepts for electronic intermediaries are derived from economic theory. Furthermore, the theoretic considerations are applied to the Java Repository and current and future developments of this special intermediary are presented, especially the use and implementation of Ecash to support financial transactions on the Java market.*

## INTRODUCTION

With the emergence of the programming language Java (Gosling, 1995) the Internet became a world-wide repository for programming solutions. The matching of programmers and users of Java software elements (classes and applets) is a necessary task in an electronic market which can be supported by electronic intermediaries. The Java Repository is such an intermediary collecting and documenting existing Java software elements. Programmers can register their resources online and consumers have the opportunity to find and obtain an appropriate problem solution using the Java Repository.

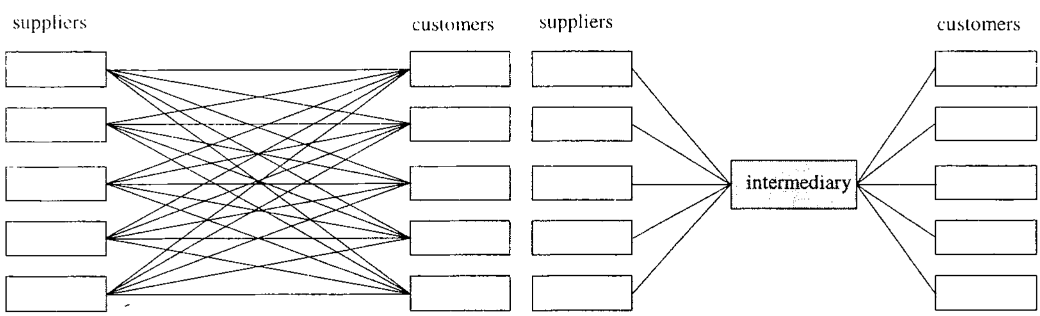
The aim of the current article is to introduce the Java Repository as an intermediary for Java software elements. In the analysis we derive some critical success factors for electronic intermediaries. Furthermore, we describe current and future developments of the Java Repository, especially the use of Ecash to support financial transactions on the Java market.

The second section shortly reviews the economic theory of intermediation. The third section discusses the theoretical implications for electronic intermediaries with regard to the Java Repository. In the fourth section we discuss the need of intermediaries on the Internet-age and try to predict the future development. The article closes with a short resume and with some considerations about the future development of the Java Repository.

## ECONOMIC FOUNDATION OF INTERMEDIATION

Already in the 19th century, Ricardo and Edgeworth pointed out that trade can increase a national economy's wealth without raising the amount of production. In general, trade can be supported by intermediaries which are agents to simplify and improve the matching between suppliers and customers. The intermediary's influence on the costs for contacts in this process were formally examined by Baligh and Richartz (1967) who analyzed an economy with $m$ suppliers and $n$ customers. Figure 1 shows the effect of intermediation. The intermediary reduces the number of possible contacts between all trading partners from $n{\cdot}m$ to $n+m$.

**Figure 1. Non-Intermediated and Intermediated Contacts**




Beside the matching of suppliers and customers, an intermediary can provide some additional services. Intermediaries offer a bundle of services, e.g., evaluation (Resnick, 1995) and categorization of products, trust (Bailey, 1996), quality assurance (Sass, 1984; Hänchen, 1985), and execution of market transactions (Hacket, 1992). A further source of additional value emerging from an intermediary is the development and enforcement of a standardized product description. The use of such a unified description can be understood in the light of transaction cost theory where different forms of coordination are studied. Products with a simple and standardized description are much easier traded on markets than those with a complex non-standardized description (Williamson, 1975). One reason is that a standardized product description simplifies the evaluation and comparison of alternative products. Another reason might be that in this case economies of scale can gain advantages for both suppliers and customers on the market.

## THE JAVA REPOSITORY: AN INTERMEDIARY FOR JAVA SOFTWARE ELEMENTS

In the current section we introduce the Java Repository, an electronic intermediary for Java software elements. We will present the Java Repository in the light of the theoretic discussion about electronic intermediaries from section 2.

### Concept of the Java Repository

The new possibilities for network programming now available with Java and its acceptance among programmers are the preconditions for future success and distribution. The Internet, especially the World Wide Web (WWW), is a communication infrastructure that connected nearly all Java programmers. So, the WWW permits access to a large number of highly specialized problem solutions programmed in Java that can be applied by programmers and users as well. For example, a programmer who has to implement a special algorithm for an optimization problem in a larger project has the following two possibilities: he can develop the algorithm himself („Make") or he can search for an existing implementation and reuse the program code in his own project („Buy"). Choosing the alternative „Buy", the programmer has to find an adequate problem solution under constraints in time and budget. The size and growth of the Internet, and therewith the growing amount of information makes searching for an available solution quite difficult nowadays. According to our considerations of section 2, we can identify the need for an intermediary matching demand and supply (Malone, 1987). The Java Repository[1] is such an electronic intermediary for trading Java software elements over the Internet, reducing the time for search and redundant production of information for the individual „consumer" of Java resources.

The Java Repository is implemented on an Oracle database connected to a webserver since an open infrastructure seems to be necessary (i.e,, to reach all Internet users and to tie up connections to as many partners for exchange as possible) (Steinfeld, 1996).

The user operates the service via HTML-forms and hyperlinks within dynamically generated HTML pages. These dynamic web pages are generated by about 100 PL/SQL procedures stored in the database. Each offer for a Java-applet or -class is stored in the database together with information about problem description, categories, keywords, internet-address (URL), and information about the programmer.

---

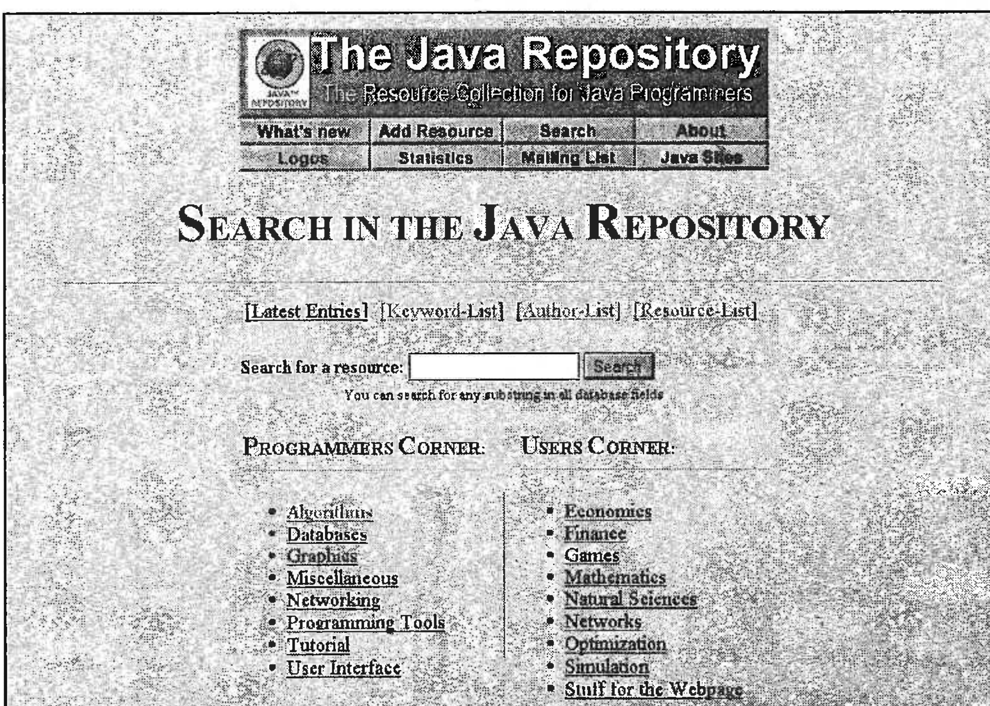
[1] URL of the Java Repository-- http://java.wiwi.uni-frankfurt.de

In section 2 we argued that an intermediary reduces the consumer's costs for product evaluation. For the reduction of these evaluation costs for Java resources registered in the Java Repository we have taken the following actions:

- development of a categorization for Java resources
- implementation of a search mechanism for all registered resources and
- standardized description of the Java resources.

The Java resources are categorized according to an *applicational view* („User's Corner"), including categories like simulation, optimization, finance, and a *programming view* („Programmer's corner") with categories like networking, databases, algorithms, graphics. The *applicational view* addresses the user looking for a ready-to-use program, the *programming view* focuses more on the needs of programmers looking for a solution for their special programming problem (cf. Figure 3). Additionally to the categorization all registered resources can be tied up with keywords, which allows a direct search for the entries.

**Figure 2. Search Page of the Java Repository with Search and Browsing Facilities**



The production of information necessary for evaluation of the resources is reduced by the standardized description of the available Java resources. The standardized description documents the object's slots and is based on:

- class hierarchy of the Java resources
- specification of the relationships between all classes used by a resource.

The unified description provides the programmer an easier evaluation of a certain Java-resource and comparison with other resources. The user interested in the ready-to-use applet can evaluate and test the running applet directly on the page of the entry in the Java Repository. The evaluation process of the resources registered in the Java Repository is further supported by a user feedback, i.e., the comments and ratings for resources left in the Repository by other users of the software. The possibility to rate and comment[2] on different resources and to compare with other solutions is an incentive for the seller to offer a program of high quality and to permanently improve the resource. The interaction between suppliers and customers, which can be initiated by the comments and ratings, is another feature that can easily be provided by electronic intermediaries.

## An Example for the Use of the Java Repository

In this section, we will give an example of the utilization of the Java Repository. Just imagine a programmer's job is the implementation of an algorithm to solve the Traveling Salesman Problem within a larger application.

Choosing the category *Optimization,* the repository offers ten solutions which handle optimization problems. Searching for the keyword *Tour*, the Java Repository provides two items with appropriate solutions: One applet solves the Traveling Salesman Problem by using well-known heuristics like 2-opt or 3-opt. Another applet is the implementation of the so-called COSA-algorithm[3]. Figure 4 shows the class relationship diagram of all classes of the COSA-implementation emphasizing the classes *Cosa, Population,* and *Tour,* which are the basic classes of the COSA-algorithm.

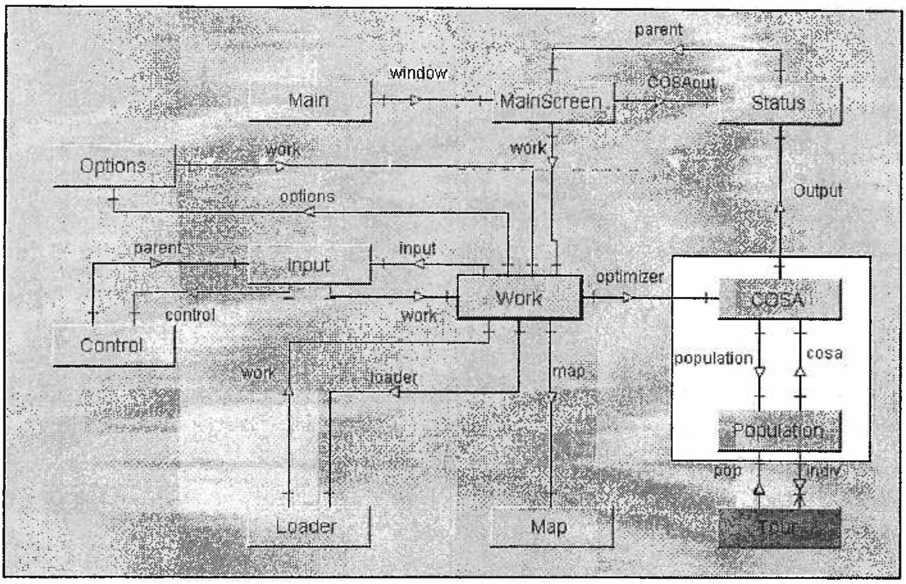
Furthermore, the user has access to the slots of the classes, allowing the investigation of the class's interfaces which are important for the reuse of a certain class. The remaining classes are implementations of the user interface and the graphical output.

---

[2] Users can leave their comments and ratings directly on the description page of the resource in the Java Repository.

[3] COSA is the abbreviation of Cooperative Simulated Annealing which integrates the basic ideas of genetic algorithms and simulated annealing to a powerful method for the approximate solution of optimization problems (Wendt, 1995). The programmer of the applet is Tim Stockheim, Frankfurt University.

## Figure 3. Class-Relationship Diagram of the Implementation of the COSA-Algorithm for the Traveling Salesman Problem[4]




The comments and ratings given by other users of the *Cosa*-class simplifies the choice of that certain resource and the integration of the class into another project. The experiences with the use of this resource can be added as comments and ratings to the resource in the Java Repository to help other programmers with their evaluation and to give feedback to the author.

The reuse of the COSA-implementation for a special optimization task in a larger project allows the programmer to concentrate on his original programming problem. One advantage of using the Java Repository can therefore be seen in time savings for software development.

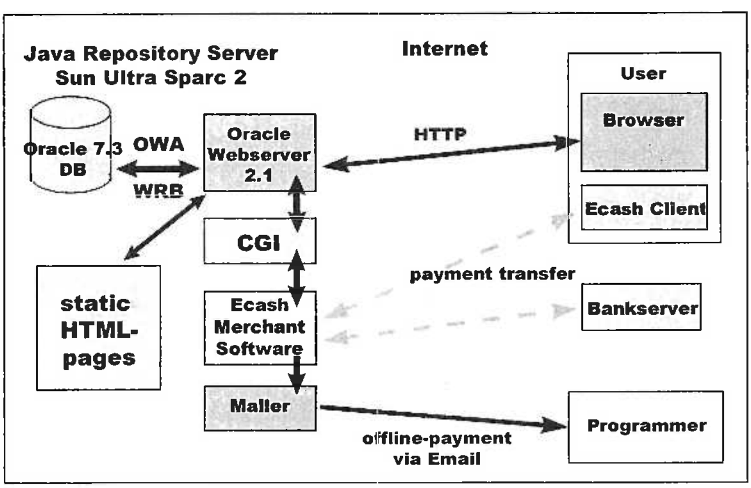
[4] The relationship-diagram was prepared with OEW for Java. OEW is the Object Engineering Workbench from Innovative Software GmbH, Frankfurt.

## Electronic Cash - A Payment System for Electronic Intermediaries

After a certain resource is found using the Java Repository the trading of that resource is supported by the Java Market@Java Repository. The financial transactions entailed in buying and selling source code can be handled completely over the Internet. This raises the question of an appropriate electronic payment method. Several electronic payment methods and systems for the Internet are currently discussed. For a detailed overview of different payment methods for the Internet refer to Wayner, (1996).

Since digital cash provides a high level of security and the opportunity of micropurchases (from 0.01 $ and up), which is especially necessary to build up a *pay per use* mechanism, we decided to implement the Ecash system[5] as the first electronic payment system (Kalakota, 1996; Lynch, 1996; Wayner, 1996). The Java Repository offers programmers the opportunity to distribute and account their Java resources in the Java Market, „consumers" can pay for the sources and download the source-code directly from the Java Repository. The electronic money is then transferred to the seller via encrypted e-mail after completion of the transaction. The whole transfer of the digital cash from the buyer over the Java Repository to the seller is completely free of charge, since all Ecash transactions are free of charge, and the Java Repository charges no fee either. The principle of using ecash and the integration with database- and the web-technology is shown in Figure 4.

### Figure 4. The Software and Hardware of the Java Repository




[5] The Java Repository currently uses the Ecash system (http://www.digicash.com/ecash/ecash-home.html) provided by Mark-Twain-Bank (http://www.marktwain.com), St. Louis, Missouri. The Java Repository runs a merchant account from Mark-Twain-Bank, buyers and sellers in the Java Market only require a buyers account.

## The Need for Intermediaries on Electronic Markets

This basic principle of intermediation discussed in section 2 applies to all different kinds of intermediaries. In the third section we have presented an intermediary for Java software elements. Intermediaries supporting transactions using information technology are often called electronic intermediaries (Sarkar, 1996). Since the emergence of the Internet leads to a decrease of information costs we often find the thesis that intermediaries will be displaced by direct contacts between suppliers and customers (Benjamin, 1995; Gellman, 1996). In the following, we will discuss this thesis and try to give a short outlook concerning the future of intermediaries in the Internet world.

Let us examine this thesis using the model of Baligh and Richartz we have presented earlier. In a system with $m$ suppliers and $n$ customers an intermediary can reduce the number of necessary connections from $n \cdot m$ to $n+m$. If we assume that all information costs at the edges have the same value we can simply show that intermediaries are still necessary as long as information costs are not zero. And even if the Internet reduces information costs, we can imagine that these will not become zero yet, if we consider the time we have to spend for searching a special information or offer.

But different structures of markets also affect the role of intermediaries. Matching supply and demand becomes more important in markets with a large and ever-growing number of different participants on the market. For example, in the automotive industry the number of possible firms to transact with is smaller than in the retailing sector, an industry where communication can primarily be based on Electronic Data Interchange (Bailey, 1996). Here the influence of the number of possible transaction partners on the advantages gained by intermediaries can simply be derived from the model of Baligh and Richartz. Moreover, the asset specificity of the products traded has an impact on the relative advantage provided by intermediaries (Bailey, 1996). The Java Market is a market with a large number of suppliers (programmers) and customers (users). So there is a high potential for an intermediary.

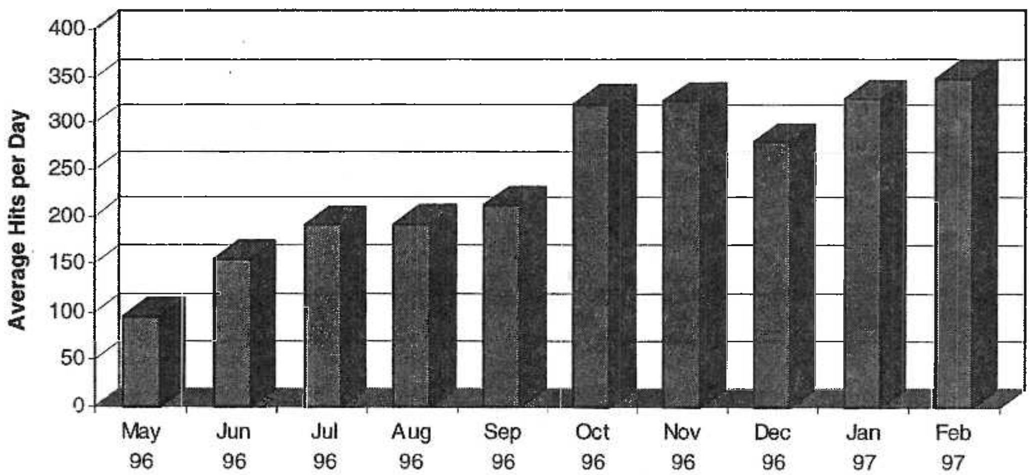
Beside the matching of suppliers and customers, we have discussed that intermediaries can provide some additional services. In the Java Repository we have implemented the following value-adding services:

- an evaluation and rating of Java software elements by users
- a categorization of Java software elements
- a standardized description of Java software elements
- implementation of an Ecash-system to support trading of Java software elements.

The following figure shows the development of the number of daily hits from May 1996 to February 1997.

We find an increase of the average number of hits per day from less than 100 in May 1996 to about 350 in February 1997 facilitated by providing additional services on the Java Repository.

**Figure 5. The Average Number of Hits Per Day**




If we consider the general question of future development of intermediaries on the Internet we cannot divide this discussion from the future development of electronic markets. One consequence of the emergence of electronic markets is that geographic distances between suppliers and customers become more and more irrelevant. This leads to a reduction of the number of markets. Blume and Goldstein (Blume, 1995) empirically confirmed this effect for capital markets. The reduction of the number of markets can lead to a reduction of the number of intermediaries, too, because we believe in world-wide networks there will be the need for only one electronic intermediary for a certain topic or market. Fur a further discussion and specification of different types of intermediaries confer to Hoffman (1996) and especially Sarkar (1996) who has introduced the term „cybermediary" for intermediaries in the Internet. From this discussion, we derive the conclusion that intermediaries, or at least one intermediary for one special market, are still necessary, even in the „Internet world".

## CONCLUSION AND FUTURE RESEARCH

The emergence and evolution of new intermediary services can be observed in many areas on the Internet. Information intermediaries come into being either if existing real world intermediaries provide their information services on the Internet or if new markets emerge from innovations. The Java Repository is one such intermediary to support the matching between suppliers and users of Java software elements over the Internet. The implementation of ecash is the first step to establish a Java Market where Java software elements can be traded over the Internet. Referring to economic theory we have shown that there will be no elimination of intermediaries. Matching of suppliers and customers is still necessary even if communication costs are decreasing. Furthermore, an intermediary can provide additional services that are valuable for its customers.

In addition to the Java Market we are currently implementing a Market for Ideas for new Java software elements, the Java Idea Futures. The Java Idea Futures provide developers and users of the Java Repository the opportunity to post offers or requests for software elements they are planning to program or wish to have implemented. So, offers or ideas can be evaluated by other users of the Java Repository and cooperation in the very early stage of software development is supported. We hope this will be another useful added value provided by the Java Repository.

## ACKNOWLEDGMENT

The authors would like to thank Thomas Golob, Oliver Christ, and Oliver Pfeiffer for massive support in the realization of the features of the Java Repository. Thanks to all researchers at the Institute of Applied Computer Sciences and from Innovative Software GmbH for discussions of the basic ideas and concepts for the Java Repository. Thanks to Tim Stockheim for providing the implementation of the *Cosa*-algorithm.

Special thanks to Sun Microsystems and Oracle Corporation for sponsoring the hard- and software of the Java Repository.

## REFERENCES

Bailey, J. (1996). The emergence of electronic market intermediaries. In *Proceedings of the 17th International Conference on Information Systems,* Cleveland, 391-399.

Baligh, H. & Richartz, L. (1967). *Vertical market structures.* Boston, Mass.: Allyn and Bacon.

Benjamin, R. & Wigand, R. (1995, Winter). Electronic markets and virtual value chains on the information highway. *Sloan Management Review,* 62-72.

Blume, M. & Goldstein, M. (1995). On the integration of the US equity markets. Working paper, University of Pennsylvania.

Brands, S. (1995). *Electronic cash on the Internet.* Proceedings of the Internet Society 1995 Symposium on Network and Distributed Systems Security. San Diego, California.

Cerqueiro, A. & Kosuge, Y. (1996, March). *Electronic payment systems on the Internet.* http://www.sloan.mit.edu/15.967/group02/pay_main.htm

Cosimano, T. (1996). Intermediation. *Economica, 63,* 131-143.

Farrell, J. & Saloner, G. (1985). Standardization, compatibility, and innovation. *Rand Journal of Economics, 16*, 70-83.

Frakes, W. & Fox, C. (1995, June). Sixteen questions about software reuse. *Communications of the ACM, 38*(6), 75-87.

Gellman, R. (1996). Disintermediation and the Internet. *Government Information Quarterly, 13*(1), 1-8.

Gosling, J. & McGilton, H. (1995, October). *The Java language environment - A white paper.* Sun Microsystems Computer Company.

Hacket, S. (1992). A comparative analysis of merchant and broker intermediation. *Journal of Economic Behavior and Organization, 18,* 299-315.

Hänchen, T. & v. Ungern-Sternberg, T. (1985). Information costs, intermediation and equilibrium price. *Economica, 52,* 407-419.

Hoffman, D., Novak, T., & Chatterjee, P. (1996). Commercial scenarios for the web: Opportunities and challenges. *JCMC, 1*(3). http://shum.cc.huji.ac.il/jcmc/vol1/issue3/hoffman.html)

Jones, T. (1994). Economics of software reuse. *IEEE Computer.*

Katz, M. L. & Shapiro, C. (1985). Network externalities, competition, and compatibility. *American Economic Review, 75,* 424-440.

Kalakota, R. & Whinston, A. (1996). *Frontiers of electronic commerce.* Addison Wesley.

Lilien, G., Kotler, P., & Moorthy, K. (1992). *Marketing models.* Prentice Hall International, Inc.

Lynch, D. & Lundquist, L. (1996). *Digital money.* New York: John Wiley & Sons Inc.

Malone, T., Yates, J., & Benjamin, R. (1987, June). Electronic markets and electronic hierarchies. *Communications of the ACM, 30*(6), 484-497.

Malone, T., Yates, J., & Benjamin, R. (1989, May-June). The logic of electronic markets. *Harvard Business Review,* 166-170.

Markus, M. L. (1987, October). Towards a „critical mass" theory of interactive media: Universal access, interdependence and diffusion. *Communication Research, 14*(5), 491-511.

McAfee, R. & McMillan, J. (1988). Search mechanisms. *Journal of Economic Theory, 44,* 99-123.

McQueen, J. (1964). Optimal policies for a class of search and evaluation problems. *Management Science, 10*(4).

Morgan, P. (1983). Search and optimal sample size. *Review of Economic Studies, L,* 659-675.

Morgan, P. & Manning, R. (1985, July). Optimal search. *Econometrica, 53*(4), 923-944.

Myerson, R. & Satterthwaite, M. (1983). Efficient mechanisms for bilateral trading. *Journal of Economic Theory, 29,* 265-281.

Resnick, P., Zeckhauser, R., & Avery, C. (1995). Roles for electronic brokers. *Toward a Competitive Telecommunication Industry.* Selected Papers from the 1994 Telecommunications Policy Research Conference, G. W. Brock (ed.). Mahwah, NJ: Lawrence Erlbaum Associates, 289-304.

Rivest, R., Shamir, A., & Adleman, L. (1978). A method for obtaining digital signatures and public key cryptosystems. *Communications of the ACM, 21*, 120-126.

Sarkar, M., Butler, B., & Steinfeld, C. (1996). *Intermediaries and cybermediaries: A continuing role for mediating players in the electronic marketplace.* http://shum.cc.huji.ac.il/jcmc/vol1/issue3/sarkar.html

Steinfeld, C., Kraut, R., & Plummer, A. The impact of interorganizational networks on buyer-seller relationships. *JCMC, 1*(3). http://shum.cc.huji.ac.il/jcmc/vol1/issue3/steinfeld.html

Wayner, P. (1996). *Digital Cash: Commerce on the net.* Boston et al.

Williamson, O. (1975). *Markets and hierarchies: Analysis and antitrust implications.* New York.