COORDINATION MECHANISMS AND SYSTEMS ARCHITECTURES IN ELECTRONIC MARKET SYSTEMS

S. Klein, T.J. Langenohl

Competence Center Electronic Markets, Institute for Information Management,
University of St. Gallen

Abstract

Computerized Travel Reservation Systems (CRS) are next to Computerized Financial Markets one of the most important and prominent applications of electronic market systems.

We want to introduce constitutive elements and concepts of electronic markets as emerging forms of organization and economic coordination. Within electronic markets and interorganizational systems, a wide variety of coordination or trading mechanisms, such as auctions, offer/accept and matching systems, are used. We will present a rough taxonomy of such coordination mechanisms. Aspects of implementing electronic markets will be discussed under the heading of systems architectures namely message passing systems. A particular focus will therefore be set on the comparison of different types of coordination mechanisms and enabling systems architectures. The example of CRS and in particular Galileo is used to illustrate different functional properties of coordination mechanisms and organizational contingencies for their application. Implications and requirements for the systems and application architecture will be discussed.

1. Electronic Markets: An Introduction

The term 'electronic market' has been used in various ways. Many synonyms of electronic markets characterize concrete forms of market transactions, e.g. electronic contracting between participants [17], electronic shopping [14] electronic execution of trading contracts depending on certain inputs of the involved actors. Examples for the application of such transaction forms are: raw material markets [20], travel and transport services [12, 7], commodity markets or financial and trading markets [25].

1.1. Market transactions and market phases

Market transactions between the different players are constitutive elements of a market. They can be coordinated in different ways. *A **market transaction** is a finite sum of interaction processes between members of different roles. The participants try in a certain period of time to settle their interests and expectations amongst themselves. The aim of a market transaction is* *a trading agreement between supplier and customer and the following settlement of goods and/or services.* In a market transaction, the following actors can be distinguished:

- The supplier and
- the buyer, who want to exchange goods and services. Especially in financial markets, actors frequently act as buyers and sellers.
- In some markets, intermediaries like dealers, brokers or market maker perform supporting functions to improve the efficiency of the transaction process.

The market place provides the (logical) location and infrastructure for the interaction of the market actors. Interaction processes in a market transaction can be combined into classes, so called phases of market transactions. [24]

Phase 1: Information phase

On the input side, information on available products and/or services, their specifications, suppliers and delivery terms are required. On the output side, potential customers have to be identified. In addition some general information on the market situation, branch or technological trends will be useful. This phase results in a list of potential partners in the market and their offer and/or demand.

Phase 2: Trading/matching

As soon as all relevant information has been evaluated, we can proceed to contact potential transaction partners and a matching pair of supply and demand (offer and bid) can be formed. Terms and conditions have to be fixed (terms of payment and delivery, warranties, additional services etc.). The legal basis is established for further transactions.

Phase 3: Settlement

In this phase the deal is settled, goods and services are exchanged for payments. This phase may consist of various derivated transactions that may require a full transaction process. In case of physical goods, possible transactions may be packaging, storage, shipping, insurance, customs clearance, etc. The physical exchange of goods is accompanied by financial and information flows. Now the physical transactions are carried out.

The following schema illustrates the sequence of process during a market transaction.

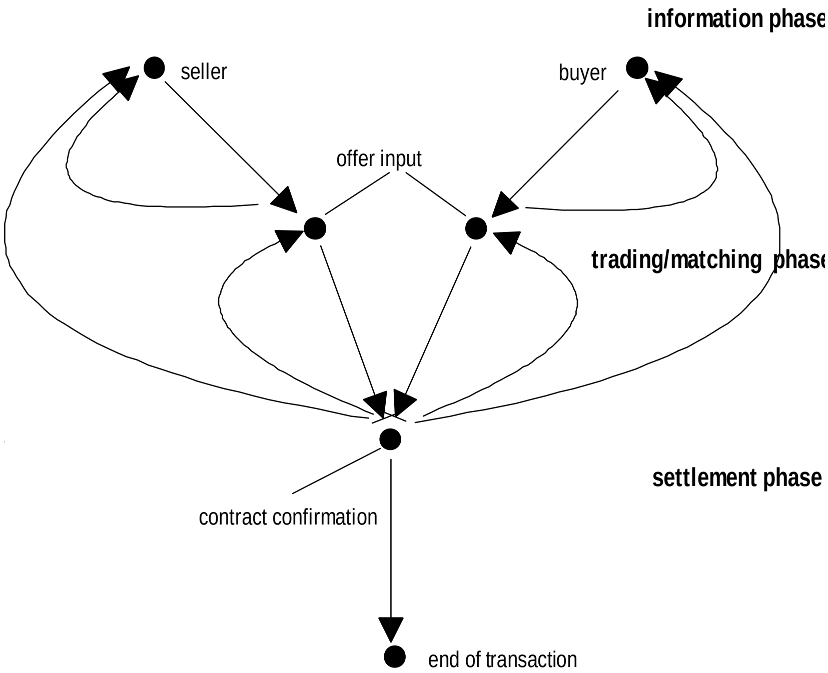

Fig. 1. Phases of a market transaction

1.2. A definition of Electronic Markets and Electronic Market Systems

Still, most existing systems have not realized electronic markets which support all phases of a market transaction. Bakos [1] defines an electronic market system as "an interorganizational system that allows the participating buyers and sellers to exchange information about prices and product offerings". For our purposes, this definition is too broad and unspecified. The exchange of information is only the first phase of a market transaction. Nearly all electronic newspapers and video text based systems would satisfy Bakos´ definition. Electronically contracting between suppliers and customers is more restrictive. Moreover, interorganizational systems are characterized as computerized systems between two or more organizations sharing data or other resources [2], excluding electronic shopping systems because the participants are single persons and not organizations.

Therefore, we define a*n electronic market as the totality of exchange relationships between market participants with potentially the same rights. All interaction processes between the actors are supported and automated by an electronic market systems - at least to the completion of the trading/matching phase*.

The term electronic market system represents a more technical point of view [8, 9, 19]. *An electronic market system is an information system, embedded in the context of an electronic market, in order to automate or to support electronically the interaction processes between the actors in a market. The information system consists of a communication network and can be distributed and/or open.* Openness and distribution aspects can be discussed from an organizational or technical point of view.

Electronic markets are hybrid systems, in which people and computers are working together in order to reach certain results. In the near future the vision of a fully automated market in all transaction phases seems not only unrealistic but also not desirable.

Within electronic markets and interorganizational systems, a wide variety of coordination or trading mechanisms are used. We will therefore present a rough taxonomy of such coordination mechanisms. In a separate chapter, aspects of implementing electronic markets will be discussed under the heading of systems architectures namely message passing systems. A particular focus will be set on the comparison of different types of coordination mechanisms and enabling systems architectures.

2. Coordination mechanisms

In electronic market systems, like in traditional markets, various mechanisms of coordination can be employed. They are matching supply and demand, offers and bids respectively. Depending on the degree to which a mechanism automates the allocation process and to the level of centralization, we distinguish three generic mechanisms. They represent ideal types of coordination based on different action patterns of the participants. Most electronic trade systems can be classified according to the coordination mechanisms or a combination of them. The first form, offer/accept (O/A) focuses on the selection of the "buyer", the automated matching (AM) emphasizes the comparison of offers and bids by an intermediary (system). The computerized auction mechanism (CA) represents an active, more or less dynamic process of trading and emphasizes the competition between the bidders and the aspect of price fixing.

2.1. Offer/accept coordination

The O/A mechanism is based on a continuous placement of offers, e.g. available seats, in an electronic order book, usually a database. The buyer can browse the order book and hit suitable offers. Hitting the offer involves a binding acceptance (commit) of offers at displayed conditions (hit and take), for example, a price set by the supplier or market maker. The priority rule is first come-first served.
The offer/accept mechanism can be best compared to the selection from an electronic catalogue. Its main advantage is to provide a high degree of transparency in the information phase of a market transaction. O/A systems provide some assistance for finding suitable offers such as retrieval or sorting algorithms, but in general they rely mainly on an active role of the buyer. CRS systems are among the most well known examples of this type of coordination.

2.2. Automated matching

An automated matching algorithm reduces the complexity of the selection process. Based on a preference profile, the system will scan huge amounts of data and select fitting matches. The function of a matching system is to take a sequence of buy and sell orders and transform it into a sequence of trades according to a predefined set of (informally) stated execution rules [22].
The order book contains orders which are (automatically) executed against incoming bids according to their fit of attributes. Rather than generating trades only based on price or price/time priority, intelligent electronic trading systems provide standards and protocols for extended product descriptions such as grades, qualities, delivery conditions, and,

corresponding, the representation of the traders´ utilities in terms of preferences and prices [18] The priority of bids and offers is produced through execution rules [10]. Prices are not dynamically adjusted in AM but trades are executed to the predefined price limits.

The traditional example of matching is the real estate market where an intermediary generates matches of buyers and sellers. Nowadays, automated matching gains increasing importance as an intelligent form of trading in commodity markets that require more complex matching than price, volume and time.

2.3. Auction

Auctions are often perceived as the ideal type of a market. Most electronic trading systems for financial or commodity items are based on auction mechanisms that automate the traditional role of a human auctioneer: it books buy and sell orders and matches them based on a price or price/time priority. Auction markets provide centralized procedures for the exposure of buy and sell orders to all market participants simultaneously [11]. Auctions can be distinguished according to variables like sequence (continuous - periodical), transparency of competing bids, or procedure (dynamic trading or submission)[23]. Typical areas of application are financial and commodity items, but also complex, difficult-to-price products like pieces of fine art (social creation of value).

The following list gives an overview of the respective features of the discussed coordination mechanisms:

Features Mechanism	Matching procedure	Intermediation and activities of the participants	Price discovery mechanism
Offer/ accept	selection from a (filtered) list of offers	electronic catalogue, reservation system, preselection of fitting offers, seller relatively passive, active selection of buyer	not during O/A, price adaptation during feed-back loops based on the scrutiny of buying patterns and competition
Automated matching	(automated) matching of buyer - seller profiles, preferences - product charac-teristics respec-tively	distinguished matching function, active role of human intermediary possible, a relatively passive role of buyer and seller (after having defined their respective profiles) possible	matching based on predefined price limits
Auction	matching based on price or price/time priority	market maker, coordina-tion of the auction process; active role of buyer and seller	depending on auction type, for many type price discovery during the auction is a core func-tion

Table 1: Comparison of coordination mechanisms

3. Systems architectures

The term information systems architecture (ISA) is ambiguous and is used in computer science and computer technology in various ways. ISA covers computer, software and network ar-chitecture. Computer architecture describes the fundamental organization of a computer. Terms like operating system-, compiler- and database systems architecture focus on construction characteristics of the respective system software category. All terms are a part of the systems architecture, they are connected and interact with each other. Nutt [21] defines systems architecture as: "A system's architecture describes the structure of interconnection of the parts of a system. The architecture specifies component interactions, and ultimately the details of the interface among interoperating components." Systems architecture is the description of the whole system consisting of computer hardware and system software. As electronic markets are open and distributed systems, systems architecture covers also the network architecture. *Systems architecture describes the interior construction, the infrastructure and the functionality of certain system components in open distributed systems. These components are the system software architecture with the underlying computer and the network architecture* (see also figure 2) [16].

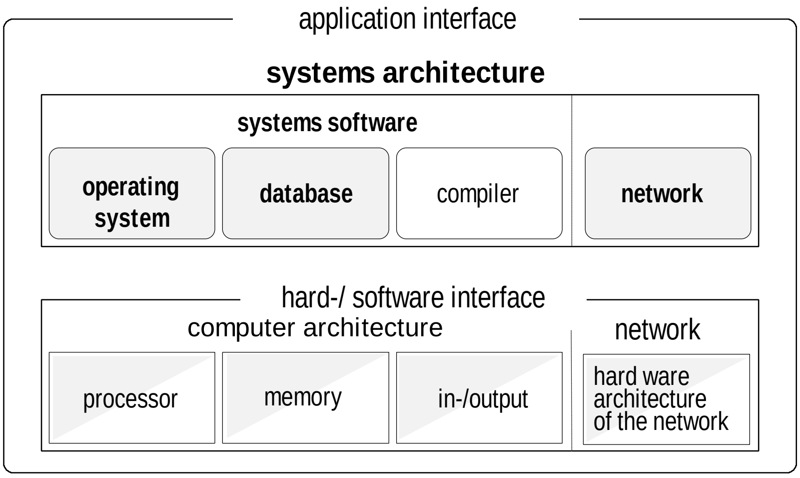

Fig. 2 Areas of systems architecture (the hatched parts of systems software)

In order to build a model of the systems architecture of electronic market systems we distinguish between several design options for market systems based on the information flow. The information flow reveals the connections among the processes in an electronic market system (static aspect). It shows which information a process has and which effects information has on other processes (dynamic aspect) [16]. So in a more abstract way each electronic market system can be described in form of processes and communication links between the processes. Systems which are characterized in such a way are so called message passing systems. Message passing models are used to describe - in distributed environments - the simultaneous execution of sequential processes. Beyond other methods, such as shared variables, message passing is the most important kind of interprocess communication. For the description of distributed systems several models of message passing exist than can be distinguished by the form of interprocess communication. In this paper we are following the work of Lamport and Lynch [15], who classify message passing models by the assumptions made about four separate concerns: network topology, synchrony, failure* , and message buffering. Different models do not necessarily describe different systems, they only represent different points of view of the same system.

* In this paper we make no assumptions about communication and process failures.

3.1. Network topology

The network topology describes which processes can send messages directly to which other processes. The topology is represented by a communication graph whose nodes are the processes, and where a link from process i to process j denotes that i can send a message directly to j. Most models assume an bidirectional graph, where a link joining two processes means that each can send messages to each other. On the other hand one can consider directed graph models in which there can be a link from i to j without one from j to i but not vice versa (unidirectional). In some models each process knows all other processes. In other models, processes have only partial knowledge - usually the identity of its immediate neighbors. The simplest models, embodying the strongest assumptions, are those with a completely connected communication graph, where each non-faulty process knows about and can send messages directly to every other non-faulty process in the graph.
James Martin [4] distinguishes the following six main topologies of communications architectures: mono-central, bicentral, multi-central, horizontal, horizontal-vertical and vertical-vertical structures. The mono-central architecture corresponds to the architecture of a host-system, a system which consists of multiple processes connected to a central host process. The bicentral structure is based on two central processes which other processes can used alternatively. The multi-central architecture is an expansion of the bicentral structure. Between the central processes no direct connection exists. The interconnections do not appear until the horizontal topology has been reached. In this case the processes are on the same level, it is a kind of peer-to-peer connection. The horizontal-vertical architecture is a combination of the mono-central and horizontal structure. In the vertical-vertical structure the central processes are not on the same level, they build a hierarchy of upper and lower processes.

3.2. Synchrony

Synchrony is an aspect to describe the coordination of cooperation and competition between processes. Especially in communication it is the coordination between the exchange of messages among the processes. In an asynchronous model no concept of real time exists. It is assumed that messages eventually delivered and processes eventually respond, but no assumption is made about how long this may take. In other models the concept of time is introduced in the way that upper limits on message transmission time and process response time are known (asynchronous interval concept). If a message is generated in response time to an event at a time t, it arrives at its destination at $t + \delta$, whereas δ is a known constant. The concept of interval response presupposes some form of real-time clock. The simplest type of a clock is a timer, which measures elapsed time; the instantaneous values of different processes' timers are independent of one another. In this configuration it is easy to detect failures as the assumption implies that a failure must have occurred if the reply to a message is not received within 2δ seconds of the sending of that message. Some models make stronger assumptions. The strongest assumptions one can make is a synchronous model in which the entire computation proceeds in a sequence of distinct rounds (synchronous time slices). At each round, every process sends messages based upon the messages that it received in previous rounds. In this model it is easy to simulate the use of synchronized clocks by letting begin each round $\delta + \varepsilon$ seconds after the preceding one.

3.3. Message buffering

In all message passing models, there is a delay between the time when a message is sent and when it is received. Such a delay requires some form of message buffering. In models with message buffering one can distinguish between finite and infinite buffers. Using finite buffering strategies, any link may contain only a maximum number of messages that have been sent over the link but not yet received. With infinite buffers, there may be an arbitrary number of unreceived messages in a link's buffer, and the sender can always send another message over the link. In reality each system has a finite buffer capacity, but this capacity may be large enough to make infinite buffering a reasonable abstraction. In buffering strategies it is possible for messages to be received in a different order than they were sent. Models with first-in-first-out (FIFO) buffering assume that messages received in the same order in which they were sent. In many asynchronous systems, interprocess communication presupposes FIFO buffering. In systems with timers and synchronized clocks a process does not send a message to another process until that the previous message to that process has been delivered, so FIFO buffering is not necessary. At the lowest level, real distributed systems usually provide FIFO buffering.

4. Computerized Reservation Systems as Electronic Markets

After we have introduced definitions and the constitutive elements of electronic markets, we now want to apply these constructs in a concrete application context. We therefore use the example of Galileo, one of the largest CRS in the tourist industry with the main focus on flight reservation. Galileo is a global information system owned by a consortium of airlines. It features global coverage of flights through cooperative arrangements with Covia and Gemini and gateways to other systems. Galileo links 23.000 terminals in 6000 travel agencies and 371 airlines, 62 of them with direct links the rest with an indirect connection via SITA.

CRS are among the most famous examples of inter-organizational systems (IOS) and in particular electronic markets. They have become a major marketing channel for the airlines and are responsible for about 80% of airline bookings [3]. After the systems' growth potentials have been exhausted, the CRS have become a field of intense competition. The product is time critical, relatively homogeneous and the complexity of the product description is low. In sum, it is an information and coordination intensive item [13].

The core functions of CRS are

- flight schedule information,
- reservation and
- booking of flights and special services such as special diets.

Seat assignment at the time of booking so that boarding passes can be issued is available in the US market, not however in most of the European markets.

4.1. Coordination aspects

We use a stylized transaction schema in order to distinguish different coordination aspects in Galileo.

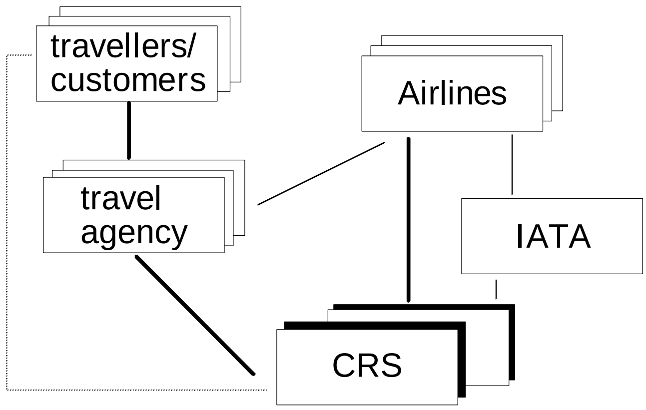

Fig. 3 Actors and transactions in Galileo

(1) Offer/ accept

Basically, Galileo is an O/A system where the customer is usually acting through a travel agent. Based on structured flight information of competing carriers, such as schedule, route, availability of seats and fares, customers select a flight (and seat) according to their preferences.

The CRS features a differentiated *accept*, mainly depending on the business practice concerning the "no show" factor and competitors policies:

- reservation which allows cancellation without costs, and
- booking where customers will be charged for cancellation or modifications in their itinerary.

Depending on the terminal used for booking, Galileo provides differentiated forms of confirmation.

(2) Extended matching: value added by travel agents

While CRS feature only limited retrieval and sorting options, e.g. the flight listings are usually based on departure time instead of costs, large travel agents provide extended matching functions based on detailed flight and customer information. General specifications and preferences about a flight, such as destination, time frame, price, route, number and location of stop overs/ change of aircraft, total time to reach destination, airline, aircraft type etc., and a customer profile, described in terms of class, smoking, seating, dietary preferences, etc. will be matched against available flights. Even business rules can be implemented that link position in hierarchy and conditions of the travel to class and medium of transport etc. The matching process can be automated using specific matching algorithms and customer databases.

In a situation of frequently changing prices and no or low cancellation fees, the automated matching can be extended until immediately before the flight in order to discover even better matches. The Covia-Galileo-Gemini CRS is the basic infrastructure for the Rosenbluth International Alliance (RIA), a global cooperation of travel agents. Several proprietary solutions have been built on the Covia-Galileo-Gemini system, the RIA Global Distribution

Network. RIA members can now access client profiles and itineraries directly through Covia-Galileo-Gemini, which is also the backbone for global consolidation of travel information for multinational clients [6].

(3) Adaptation of prices

While O/A and matching refer to the interactions of customer and travel agent with Galileo based on fixed prices, CRSs provide excellent information for the airlines to adapt their prices. In this case, the airlines take a more active role in the market process. Deregulation in the airline industry and the statistical analysis of demand patterns led to a multiplicity of fares particularly in the US market, up to the differentiation of last minute fares depending on the availability of seats. With the results, that during many flights only a few passengers had paid the same fare. Shrinking profits and huge losses*, however, have led to a return to a simple fare system (the fares depend on the time of reservation) with numerous exceptions and short-term adaptation of fares, such as seasonal specials. Recently the US Justice Department filed an antitrust suit that accused the eight largest American airlines of using a CRS to fix air fares. The airlines had used the CRS to signal and negotiate price changes [26].

The analysis of coordination aspects of Galileo has shown that electronic markets often combine several coordination mechanisms, depending on the patterns of interaction and roles of the participants (compare table 1). In the next step, we want to discuss implications for the design of systems architectures of CRS.

4.2. Applications - and systems architecture

CRS are in general on-line transaction processing systems (OLTP) and constructed in form of a client-server model. Nowadays high performance OLTP-systems exist which are able to execute 1.000 transactions per second [5], enough to build global CRS. A transaction T is an atomic unit of an operation, which is controlled by a transaction manager and consists of a finite number of actions ($T=a_1a_2a_3 \ldots a_n$). A common database for many users is a core feature of an OLTP-system. Other aspects are a high parallelism in transaction processing and a reliable updating of information. So it is possible for clients to use several resources (servers). The coordination between clients and servers is arranged by a transaction processing manager who controls and routes the accounting services. To realize those systems it is necessary to have a certain (or several) network topology(ies) with restrictions according to infrastructures of those networks.

4.2.1. Topology

In CRS several possibilities exist to construct a network architecture.

---

* The airlines that are organized within the IATA face a cumulated loss of 4.8 billion US$ for 1992 alone and a cumulated loss of 11,5 billion US$ since 1990. Neue Zürcher Zeitung Nov. 1 1993, 254.

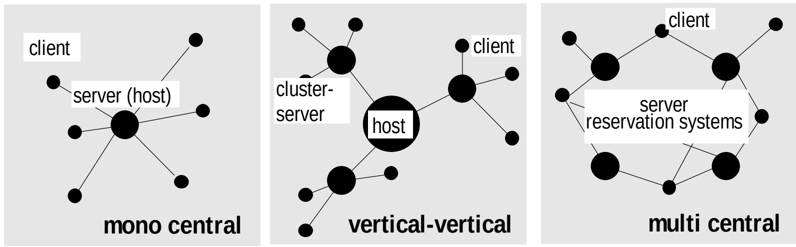

Fig. 4 Possible network topologies for reservation systems

The choice of a certain topology model (static aspect) for an airline reservation system depends on the regional aspects and the individual organization of the reservation system, in detail the structures of the associated travel agencies. The following static, bidirectional structures of a reservation system are possible.

- (1) central host (mono-central topology) and
    - (1a) directly connected diskless terminals, which have only in/output masks,
    - (1b) directly connected workstations, that have their own operating systems and the possibility for data storing and administration,
- (2) cluster wise connected workstations, which are connected with the central host of the CRS over an specially prepared workstation or a special server which is responsible for the cluster (vertical-vertical topology),
- (3) several CRS with the same rights to which customers have access to in form of terminals or workstations (multi-central topology),
- (4) the combination of the above network topologies.

Larger travel agencies in Switzerland, Kuoni and Danzas, have their own (database) host which is connected to the Swissair host. Swissair has a connection to the central host of Galileo in Denver, CO. Smaller agencies send their jobs and bookings directly to the Swissair host and from there to Galileo. These topologies are hierarchically combined configurations of type (2) - cluster wise connected workstations. The configuration of type (3) represents a further development stage of a CRS that offers the customers a connection to a cluster of CRS in a homogeneous way. The typical technical infrastructure of travel agencies in Galileo is a LAN token ring structure using OS/2-Gateway servers. The central system (central host) of Galileo is based on IBM hardware with the TPF-operating system for central booking services. The regional networks of the travel agencies and connections to other central CRS, especially Covia, are placed in a SNA-backbone network.

4.2.2. Dynamic aspects: Information flow

The communication between the supplier(s) and the customers characterize the dynamic aspects of the architecture of a CRS. Therefore, special mechanisms for synchronizing the communication have to be provided and supported by the operating system. The transactions between customers and the system (supplier) can be divided into three market phases: information, trading and settlement phase (see above). In the trading phase, CRS are based on O/A coordination mechanisms, so called fixed price systems. The exclusiveness of the services especially the booking of an aircraft seat makes it necessary to establish a synchronized

communication link (in form of synchronous timer or synchronous time slices) between the client and the server, holding the contingent of free aircraft seats. Therefore, the send and receive operations that have to be synchronized are based on blocked communication primitives. Both processes are blocked in their normal execution until the booking is finished. The customer process has to be a so called 'remote invocation send'-process. In this form of communication the receiver (server) confirms per message that the contents of client message, e.g. the booking of a seat, has been executed. Galileo provides different forms of booking functions that are based on these communication primitives: guaranteed booking with buying confirmation, super guaranteed booking with booking reference after finishing of the transaction and secured booking. In the last form the sold seats are directly marked on the seatplan of the airline and the travel agency gets a booking reference. In this case, the travel agent issues a boarding pass with valid seat number. An elder form of booking is called teletype, based on sending a telex to the respective airline. Teletype is a connectionless form of communication. It is an asynchronous message transfer by using mail boxes or ports, special operating system mail boxes.
The buffering of information depends on the way interprocess communication is used and which network typology is available. In a mono-central configuration all customers processes get their information out of a central organized buffer of the supplier. The supplier has only to make sure that the account on information is regulated in an ordinary way (a kind of process synchronization). If customers themselves have the possibility to buffer information (type 1b), message are placed in the buffer of the customers. In this case the supplier must be able to send simultaneous messages to all customers (broadcast) or a certain group of customers (multicast).
Functionalities in the Galileo system which do not require a kind of time critical execution are usually based on a asynchronous interprocess communication (asynchronous interval concept). This guarantees that processes on the server are not blocked through client processes.
In the information phase, customers need information about a flight tariff, date, time, etc. from location *a* to location *b* or they want do know something about special conditions like shortest flight-time, cheapest tariff, special airline etc. All these information is non time-critical. So it seems to be possible to locate these information on a database server of a travel agency and not on the central host of Galileo.
Another aspect in the information phase is the comfort offered by the reservation program to access the necessary information. For the user of such a system, it is important to have effective information retrieval functions that are also based on on-line transaction processing mechanisms. On the level of systems architecture these mechanisms are supported by special database functions and basic functions for resource and file management offered by the operating system.
The comfort of using a CRS depends on efficient and appropriated user interfaces. From the point of view of the customers, user interfaces have to be highly transparent to hide the details of implementation. In general modern user interfaces are graphical interfaces (GUI) with special masks and menus for an easy and comfortable handling of customers requirements to the system. The design and implementation of GUIs is not directly a problem that has to be solved

on the application architecture level and supported by certain systems architecture functions. The requirements of the coordination mechanisms with respect to masks and menus are ambiguous. It must be possible to support different forms of information retrieval e.g. in the extending matching mechanism in order to get suitable and temporal acceptable responses on a retrieval for smoking, seating, dietary references etc. Therefore it is necessary to have certain flexibility in adding new functions to and changing functions in an GUI. Portability and extensions in a heterogeneous and open system environments are a prerequisite for a relative independence of systems and network hardware (vendors).

The criteria mentioned in relation to the adaptation of prices (3.2) require several transactions for an adequate updating of price information in (distributed) databases. In this case, data control mechanisms must guarantee integrity and consistency. Problems of redundancy have to be solved and recovery functions for failures in databases have to be provided. The manipulation of data is connected to certain rights. The definition of rules for that manipulation is the task of the database administrator.

The following table relates coordination and architectures aspects:

coordination aspects	**systems architectures**			**application architectures**
	topology	**synchrony**		
		information	matching	
offer/accept	mono-central, vertical-vertical, multi-central	asynchron	synchron	OLTP-systems, (distributed) databases
extended matching			asynchron communication is also possible (RIA)	distributed databases on local servers of travel agencies for a complex information retrieval in the information phase
price policy	-	-	-	special rights for suppliers, price updating and integrity functions

Table 2: Architectural aspects of coordination mechanisms

5. Conclusions

Currently we observe increasing competition among CRSs. Several airlines have joint forces and shared the expenses to built CRS and to provide global coverage. Consequently, system functions and enabling systems architectures become the source of competitive advantages. Innovative designs of IS architectures are prerequisites of additional functions in this context of high performance OLTP. The discussed coordination aspects of CRS reflect crucial development trends for CRS: While initial CRSs provided only limited retrieval options and were biased in their presentation of flights, extended functionalities have been added during recent years. Systems and network architectures provide the technical infrastructure for global coverage and user friendliness.

As the example of RIA shows, travel agents will play a major role as intermediaries for the time being, especially if they succeed to add value for their customers in terms of extended retrieval and accounting functions. The division of labor among CRS and large travel agents has mainly organizational reasons. The complexity at least of parts of the travel business require local presence and know how, combined with access to global information. The usage of the

CRS infrastructure for proprietary applications of travel agents is a good example for the trend to increasing cooperation in the industry.

The close scrutiny of business practices of the airlines using CRS by national and international agencies (like the US Justice Department in the case mentioned above) indicates the potential impact of CRS on the market structure. On the one hand, CRS can be used for a strategy of high price flexibility of supply resulting in intense price competition and differentiated prices. Last-minute offers or even auctions could be implemented on CRS and used to increase the load factor. On the other hand, especially large players tend to prefer a more rigid price policy with only limited price competition. CRS thus have become powerful instruments of the respective airlines strategies, however, within the (inter)national regulations.

References

1. Bakos, J.Y.: A strategic analysis of electronic marketplaces, in: MIS Quarterly 15, pp. 295-309, 1991.
2. Barrett, S.: An IS* case: the closed loop scenario, in: Information and Management 10(5), pp. 263-269, 1985.
3. Bradley, C., Smith, D.: Networking Reservations, in: Network World, pp. 22-45, August 22, 1988.
4. Bues, M.: Offene Systeme - Herausforderung an das Informatik-Management. SVD/VDF-Tagung, Zürich, pp. 4-4. March, 1993.
5. Claybrook, B.: OLTP - Online Transaction Processing Systems. John Wiley Verlag, 1992.
6. Clemons, E.K., Row, M.C.,Miller, D.B.: Rosenbluth International Alliance: Information Technology and the Global Virtual Corporation, in: Nunamaker, Jay F., Sprague R.H. (Eds.): Proceedings of the 25th Hawaii International Conference on System Science, Vol. IV: Information Systems, Los Alamitos, Ca., pp. 678-686, 1991
7. Copeland, D.G., McKenney, J.L.: Airline Reservation Systems: Lessons from history, in: MIS Quarterly 12, pp. 353-370, 1988.
8. Dennis, S.G.P.: Automating financial markets, in: Data processing 26(1), Butterworth & Co Ltd, 1984.
9. Dewitz, S.K.: Contracting on a performative network: Using information technology as a legal intermediary, in: Stamper R.K., Kerola R., Lee R., Lyytinen K.J. (Eds.): Collaborative Work, Social Communications and Information Systems. Elsevier Science Publishers B.V. North Holland, 1991.
10. Domowitz, I.: The mechanics of Automated Trade Execution Systems, in: Journal of Financial Intermediation, p. 175f.
11. Garbade, K.: Securities Markets, McGraw Hill, p.426, 1982.
12. Gifford, D., Spector, A.: The TWA Reservation Systems (Case Study), in: Communications of the ACM 27(7), pp. 650-665, 1984.
13. Hooper, M.D.: Rattling Sabre - New Ways to Compete on Information, in HBR, pp. 118-129, May-June 1990.
14. Kimbrough, S.O., Isakowitz, I.: Perspectives in Electronic Shopping: On beyond Automated Order Entry. Center for Research on Information Systems, Leonard Stern School of Business. New York University. Working Paper Series, 1989.

15. Lamport, L., Lynch, N.: Distributed Computing: Models and Methods, Chapter 18, in: Handbook of Theoretical Computer Science. Elsevier Science Publishers B.V. North Holland, 1990.
16. Langenohl, T.J.: Systemarchitekturen elektronischer Märkte (systems architectures of electronic markets), (in print).
17. Lee, R.M.: International Contracting - A formal language approach. Hawaii International Conference on System Sciences. Kona, Hawaii, 1988.
18. Lee, H.G., Lee, R.M.: Intelligent Electronic Trading for Commodity Exchanges, in: EM-Electronic Markets, Vol. 3, No. 9/10, pp. 5-6, 1993.
19. Lee, R.M., Widmeyer, G.: Shopping in the Electronic Marketplace, in: Journal of Management Information Systems 2(4), pp. 21-35, 1986.
20. Lindsey, D., Cheney, P.H., Kasper, G.M., Ives, B.: TELCOT: An Application of Information Technology for Competitive Advantage in the Cotton Industry, in: MIS Quarterly 14. December, 1990.
21. Nutt, G.J.: Open Systems. Prentice Hall Verlag, 1992.
22. Reck, M.: Formally Specifying an Automated Trade Execution System, in: Journal for Systems Software, Vol. 21, pp. 245-252, 1993.
23. Reck, M.: A formal taxonomy of Electronic Auctions, (in print).
24. Schmid, B.; Elektronischer Märkte - Electronic Markets, in: Wirtschaftsinformatik, Vol. 35, No. 5, pp. 465-480, 1993,
25. Siegel, D.R. (Ed.): Innovation and Technology in the Market: A Reordering of the World's Capital Market Systems. Probus Publishing Company, 1990.
26. Tolchin, M.: U.S. Sues 8 Airlines over Fares - Computer Called Price-Fixing Tools, in: NYT, p. D1; D13. December 22, 1992.