Booking Method and System

Field of the Invention

The present invention relates to telecommunications. In particular, the invention relates to a method and system for booking a reservation in a booking system and synchronizing bookings in several booking systems comprising at least one booking systems; involving at least one service provider; a mediator service; a client, and at least one client terminal device that can be a mobile device and that includes a dialogue. In addition, the system comprises telecommunication connections that are used to connect the booking systems, the service providers, the mediator, and the client terminal device.

Background of the Invention

Services that are booked or used via the Internet are constantly increasing. The Internet enables one to use several on-line services such as services connected to banks, health services, travel agencies, vehicle maintenance, and so on.

The increasing popularity of mobile computing and communications devices introduce new challenges to services on the Internet. Mobile terminals are able to deliver information to users when needed and where needed. Users want ubiquitous access to information and applications from the device at hand. They also want to access and update this information wherever they happen to be.

It is important to notice, however, that not all the terminals will be mobile. Future services must be able to communicate with a large variety of terminal devices, both

those that are mobile and those that are not. Different terminal devices have very different capabilities.

The interoperability of different services and terminal devices requires standards on several levels. It is not enough to have, say, common communication protocols. It would be very important to share common concepts and understanding what a certain piece of data means in a certain context. However, it has been very difficult to agree on those issues, as there exist an enormous number of companies, organizations, and other actors in the field.

Many services must be able to manage bookings. They include for example booking appointments for health services; booking travel reservations for hotels, airlines, and rental cars; booking tickets for venues; booking appointments for vehicle maintenance; booking maintenance for apartments; and so on. It would be very useful, if those services could get information from one another. For example, if a customer is booking tickets for a concert, he or she might want to book a table in a restaurant also. It helps, if the restaurant's booking service gets basic information, like date and customer's name from the theater's booking system. Unfortunately, there have not been methods to exchange information between different kinds of booking systems.

There are many methods to exchange information between services. Speaking of services that include booking or calendar functions, information exchange often takes place as synchronizing booking or calendar entries. For that purpose, several important standardization efforts are going on. For example, SyncML is an industry initiative to develop and promote a single, common data synchronization protocol.

vCalendar is an exchange format for personal scheduling information. It is applicable to a wide variety of calendaring and scheduling products and is useful in exchanging information across a broad range of transport methods. A number of vendors have adopted the specification because it allows their products to exchange calendaring and scheduling information. vCalendar is an open specification based on industry standards such as the x/Open and XAPIA Calendaring and Scheduling API (CSA), the ISO 8601 international date and time standard and the related MIME email standards. The vCalendar format utilizes data normally stored within a calendaring and scheduling application, facilitating the cross platform exchange of information about items such as events and to-do's. An event is a calendaring and scheduling entity that represents a designated amount of time on a calendar. A to-do is a calendaring and scheduling entity that represents an action item or assignment. For instance, it may be an item of work assigned to an individual.

vCard automates the exchange of personal information typically found on a traditional business card. vCard is used in applications such as Internet mail, voice mail, Web browsers, telephony applications, call centers, video conferencing, PIMs (Personal Information Managers), PDAs (Personal Data Assistants), pagers, fax, office equipment, and smart cards. In addition to text, vCard information may include elements like pictures, company logos, live Web addresses, and so on.

As these examples show, there have been lots of efforts to build systems that can synchronize booking systems. A common problem with all of these existing solutions is that they do not provide common semantics for

different systems. For example, if an entry is tentative, different systems may interpret it in different ways.

Another problem is that booking systems have multiple different and usually quite complex user interfaces. If a customer wants to both make an appointment with a dentist and book a taxi to take him or her there, the customer needs to enter all the booking information to both booking systems in different ways.

One more problem is that it becomes challenging to manage client replies, if a client has been given a number of questions. For example, it makes sense to use SMS text messages to ask client which option he or she chooses, because in many countries, like in Finland, it is very common to communicate with SMS text messages and they create revenues to operators. However, if a client replies to several inquires by sending a number of text messages, it can be troublesome to find out, which answer corresponds to a certain question because the reply does not automatically include a reference to the question. Say, a service asks a client if he or she wants to reserve - in addition to a flight ticket - also a taxi and a hotel room, and the client replies "yes" to one question but "no" to the other, the service does not necessarily know which offer the client has accepted.

Objective of the Invention

The objective of the invention is to eliminate the drawbacks referred above or at least significantly alleviate them. The invention enables new kind of value adding services that are essential especially for mobile services.

It is a further object of the invention to provide a method and system capable of making booking type transactions involving at least one service provider and a plurality of users who are each communicating with a mobile telephone capable of receiving and sending short text messages.

It is a further object of the invention to provide a method and system capable of making booking type transactions between a plurality of service providers and a plurality of users who are each communicating with a mobile telephone capable of receiving and sending short text messages.

Brief Description of the Drawings

In the following section, the invention will be described in detail by the aid of a few examples of its embodiments, in which

Figure 1 represents one advantageous system in accordance with the invention;

Figure 2 represents a second advantageous system in accordance with the invention;

Figure 3 represents a third advantageous system in accordance with the invention;

Figure 4 is one advantageous example of a sequence diagram representing messages transmitted within a system in accordance with the invention;

Figure 5 is a second advantageous example of a sequence diagram representing messages transmitted within a system in accordance with the invention.

Figure 6 shows an example of the dynamic dialog matrix applied to a query and reply according to the invention.

Figure 7 shows the phases of the booking process in a preferred embodiment of the invention.

Figure 8 shows a matrix diagram corresponding to Example 2, according to a preferred embodiment of the invention.

Description of the Invention

The invention relates to exchanging and synchronizing information between booking systems and user terminal devices. The services may be for example booking appointments for health services; booking travel reservations for hotels, airlines, and rental cars; booking tickets for venues; booking appointments for vehicle maintenance; booking maintenance for apartments; and so on.

The booking system in accordance with the invention comprises at least one service provider booking system; at least one service provider; a mediator; a client; at least one client terminal device that can be a mobile device capable of receiving text messages, and that includes a dialogue; and telecommunication connections that are used to connect the service provider booking systems, the service providers, the mediator and the client terminal device to one another.

The service providers are those with whom clients want to make appointments, reservations, or other bookings and comprise the resources for the booking system to allocate. Service providers conduct business through service provider booking services. As used in this application, the mediator is a network based service available to the service provider booking services over the network that provides additional semantics, translation and

synchronization services needed for communication of the information needed for a client to complete a transaction with a service provider. The service provider booking services and the mediator are preferably applications operating on network servers such as the Internet or a private Intranet. In general, a system will comprise a plurality of service providers and service provider booking systems (implementing service provider booking services), but it is possible to have a simple booking system for only one service provider in which case the mediator and service provider could be tightly integrated into a single application.

Clients preferably include clients communicating on mobile telephones capable of receiving short text messages, such as Short Message Service (SMS) messages. Of course, a system that is capable of handling SMS messages will also handle other clients with greater capabilities. The mediator preferably communicates with mobile telephone clients through an SMS gateway, such as are operated by mobile telephone providers and a well known today. The mediator communicates with clients using dialogues. Dialogues are short messages which present information to the client and allow a simple reply. Dialogues preferably provide users with simple choices such as yes/no or to allow a selection from an ordered list. Dialogues can also be one way, such as to acknowledge a reservation. A transaction may typically involve a sequence of dialogues each involving a simple response. Dialogues involve asynchronous communication by messages. The system as described makes it possible to coordinate bookings among different service provider systems in order to fill a

clients need, for example coordination of an airline booking with transportation to the airport.

Figure 1 is a diagram of the simplest system comprising a single service provider booking system 100 for a single service provider, a mediator 102 communicating with the service provider over a network , and a user with a mobile phone having a dialogue entered thereon.

Figure 2 shows a plurality of service provider booking systems communicating with a mediator over a network.

Figure 3 shows a mediator named BookIT communicating with various service provider systems and users with telephone devices communicating dialogues.

A *reason based customer dialogue* is a desirable improvement from the client's point of view, because service providers can create their own dialogues in connection with each booking event. A dialogue is closely related to a certain booking situation. It becomes active automatically at the right moment, or the client can activate the dialogue as needed, or another entity in the system can send a message to the dialogue to activate it. The dialogue then sends an inquiry to another entity in the system or informs the client and possibly inquires client's choices. By means of this kind of dialogue, the client can make reservations in several booking systems using only one user interface. The dialogue connects to remote booking systems e.g. through the Internet or even mobile networks.

A mediator service can be capable of transmitting booking information between service provider booking systems. For example, after a booking is entered into an airline booking system, a taxi booking system can offer the

client a lift to the airport. In this application, a booking is an allocation of a single resource (either the airline booking or the taxi in the previous example), while a reservation is the union of the bookings for all of the resources for the same event (the airline booking plus the taxi booking in the previous example). The dialogue between the client, the mediator and the booking systems as well as stored customer profiles ensure that the client gets the reason based service he or she needs, not intrusive advertising.

A client can make reservations as well as confirm, change, and cancel them using many kinds of communication means, including but not limited to the Internet, e-mail, and mobile terminals. The client can also synchronize a calendar provided by the mediator or a service provider with a calendar in a terminal device using mediator's synchronization functions.

A service provider can remind clients to make reservations on a regular basis and thus increase customer loyalty. A mediator can help service providers to bring their booking systems together to provide more comprehensive services without extending their businesses unnecessarily. Because of internationalization, the mediator is able to support for example many languages, time zones, currencies, and data formats.

The system, including at least a dialogue, a mediator, a service provider, and a service provider booking system, can be on one of the following levels:

1. There is a predetermined set of dialogues in the system. Their content and the possible choices are set in advance. For example, if a client books a flight, a

dialogue always offers certain other bookings. Client's prior actions are not taken into consideration.

2. There is an unlimited number of dynamic or "intelligent" dialogues that are based on, for instance, a profile that a client has created himself or herself, usage history records, and client's location. Simple logic supports decisions. It is a low-level expert system.
3. The system is able to make decisions by itself and to support client's decision making. On this level, a dialogue may include a high-level expert system. It can act as an agent and negotiate with several service providers to get the best offer without client's direct involvement.

In one preferred embodiment of the method, a client books a service from a service provider. The booking may be carried out using a terminal that is connected to the mediator service. First, the client connects to the mediator service using a dialogue. The client inputs reservation inquiry to the dialogue that sends the inquiry to the mediator. The mediator inquires possible reservations from service provider's information system using concepts and terminology that those services are able to interpret. The inquiry is based on client's preferences. The client discloses some preferences that are related to the specific booking when he or she inputs reservation inquiry to the dialogue. In addition, the dialogue and the mediator service may have stored client's general preferences and use them so that the client do not need to input all the preferences each time.

Managing the inquiry and bookings is based on sophisticated state models. Each booking involves several phases that are described by states that track its status

through its life cycle. For example, when the mediator has inquired about a reservation from a service provider, the corresponding entry in each system has a state that the booking is pending but not confirmed. If the systems do not have common understanding what a certain state means, the mediator translates them. A preferred booking process including the phases and states is described in Example 1.

In addition to inquiring reservations from the service provider, the mediator is able to synchronize bookings in several service providers' systems. The synchronization is based on rules specified in the mediator service. For example, a rule can be that "if a client inquires booking for an airline ticket, inquire also bookings for taxis to the airport." Therefore, an inquiry from the client may be multiplied in the mediator service resulting a number of inquiries. The service providers answer to the mediator if they are able to provide requested service and they may add some additional information, like on seats or timing. The mediator combines gathered information and sends it to the dialogue that shows a simple list of options to the client. For example, the dialogue may show three options for a flight and ask if the client also wants to reserve a taxi that is actually already tentatively booked by the mediator. The client makes his or her decision by choosing the options from the simple list of alternatives. The dialogue sends information on client's choice to the mediator that confirms the bookings in accordance with client's choices and cancels the unnecessary reservations.

Figure 4 shows a sequence diagram of an inquiry CINQ1 originated by a client using a dialogue DINQ1 sent to the mediator. The mediator initiates the inquiry MINQ1

which corresponds to CINQ1 and DINQ1 to booking system 1 a service provider booking system. Ultimately an answer DANS1 gets back to the client offering a choice which is responded to with a selection CSEL1 resulting in a booking by the client on booking system 1. The mediator recognizes the potential need for a complementary service from booking service 2 and initiates an inquiry, MINQ2, to booking system 2, which ultimately results in a proposal including several choices, DANS2, returned to the client from which a selection, CSEL2, is made, resulting in a complementary booking on booking system 2.

The bookings can be done in other means as well, for instance, by calling the service provider with a telephone or by visiting on site the service provider's office. In that case the service provider may inform the mediator about client's bookings so that the mediator can inform the client on other options. For example, a dentist could tell the mediator that the client has booked an appointment so that the mediator may offer to book a taxi also.

Also, it is possible to add a reminder to the mediator service so that the mediator asks at certain time if the client wants to make a new booking. For instance, the mediator can send a notice to the client that it has been a year since the client last had an appointment with his or her dentist and ask if the client wants to make a new appointment. This notice can already include a few options for the appointment. The mediator has checked the client's calendar if he or she has allowed that so that the given options are convenient for the client. The dialogue shows the options in a simple and handy way. The client needs only to choose which option is the best for him or her or whether he or she wants to get new options or

postpone the booking. Figure 5 is a time sequence chart for such a situation where the original inquiry, MINQ1, was initiated by the mediator.

Example 1 - A Preferred Booking System

A preferred booking system according to the invention, is described below in terms of a system named BookIt.

BookIT is designed to interface between service provider booking systems and other parties over a network such as the Internet, and to end user clients equipped with mobile phones capable of receiving text messages. The former is preferably accomplished with a generic XML interface. BookIT supports vCard and vCalendar standards since they are used by all major booking and calendar systems.

BookIT communicates with mobile phone users using Short Message Service (SMS) via an SMS Gateway for asynchronous communication. BookIT uses the novel Dynamic Dialogue Matrix (DDM) for secure transfer and mapping of the SMS messages. The DDM is described further below.

A clear distinction needs to be made between a service provider booking process and BookIT Process. The former covers the standard booking only with time and resource reservation. The latter consists of booking, work, and financing. Both processes end to the same point. The BookIT Process consists of seven phases as follows:

Phases (Status handling)

The phases make a bond (rubber band) between the resources. In each of the BookIT Process' phases the data related to the booking will be amended to reflect the needs of the phase in question. For the statuses and values please see the underneath table.

The phases are described in more detail in the following discussion.

1. Filing

Filing means initialization of a BookIT Process and a booking process. As a result of the initialization an entry is inserted in the database w/basic information. It will not appear in a calendar since there is no scheduling information. It can be displayed in a separate task list of the owner as an open task.

2. Requesting

In the Requesting phase a booking request is sent to the resources required for the previously filed task. Since there is no scheduling, which in most cases will be essential, this phase may be executed together with the Scheduling phase.

3. Scheduling

Schedule is given to the owner and the resources. As a part and a result of the Scheduling the following data is needed:

a suggested start-time (ISO time-stamp w/time zone)
b suggested start-location (coordinates)
c suggested end-time (ISO time-stamp w/time zone)
d suggested end-location (coordinates)

4. Confirming

Time and location as it is accepted by the resources that have accepted. Data related to this phase:

a accepted start-time (ISO time-stamp w/time zone)

b accepted start-location (coordinates)

c accepted end-time (ISO time-stamp w/time zone)

d accepted end-location (coordinates)

By default the data is copied from the Planning phase.

In practice, if planned time is not needed, the same data structures can be used for this and status indicates the actual meaning of the data.

5. Working

The resources perform the booked task. Data related to this phase consists of different attributes and their values, which are related to the actual task. In addition, following static structures are needed:

a actual start-time (ISO time-stamp w/time zone)

b actual start-location (coordinates)

c actual end-time (ISO time-stamp w/time zone)

d actual end-location (coordinates)

e products used, extras, mileage, …

By default the data is copied from the Confirming phase.

1. Accounting

At this point all data stored in the data structures on previous phases is analyzed and processed for invoicing purposes.

Data related to this phase: Accounting data. To be defined separately.

7. Completing

The task has been completed. From the whole BookIT process point of view it is irrelevant whether the task succeeded or not. It is relevant to the Accounting phase, in which the financial actions to the organizer are handled. In this phase, housekeeping (database contents; temporary files, ...) is made in order to complete the BookIT Process.

The following table shows data available in each phase. Booking phase is in italics.

Filing	X					X	
Requesting	X	X				X	
Scheduling	X	X	X			X	
Confirming	X	X	X	X		X	
Working	X	X	X	X	X	X	
Accounting	X	X	X	X	X	X	
Completing	X	X	X	X	X	X	X
Phase/Data	Identi-fying	Resour-ces	Sugges-ted time	Accep-ted time	Task's work related	Accoun-ting	Closing

Phase Statuses, Values, and Transitions

The following table describes the phases, their statuses, and values along with transition to next logical phase based on the values gotten. In addition, corresponding vCalendar statuses are shown when applicable.

Phase	Satus	Next Phase	vEvent	vTodo
Filing		Requesting		
Requesting		Scheduling	Sent	Sent
Scheduling	Pending	Confirming	Needs action	Needs action
Scheduling	Scheduled	Confirming	Needs action	Needs action
Scheduling	Re-scheduled	Confirming	Needs action	Needs action
Confirming	Accepted	Working	Confirmed	Accepted
Confirming	Declined	Accounting	Declined	Declined
Confirming	Tentative	Accounting	Tentative	
Confirming	Delegated	Requesting	Delegated	Delegated
Confirming	Re-scheduling requested	Accounting or Scheduling		
Confirming	InProgress	Working		
Working	InProgress	Working		
Working	Delayed	Working		
Working	Started	Working		
Working	n % ready	Working		
Working	Ready	Accounting		
Accounting		Completing		
Completing	<Copied from phase before Accounting >	n/a		

Internal phases Paused, Re-started, and Canceled act as follows for all relevant phases at any point:

<Phase y>	Paused	<Status x>		
<Phase y>	Re-started	<Status x>		
<Phase y>	Cancelled	Accounting		

Figure 6 shows the work flow transitions from phase to phase. For conditions, see the table above. Also, please note that Canceled Status always leads to Accounting.

Confirming the (whole) Reservation

In order for the whole Reservation to be successful, all resources, which accepted the reservation, need to have the same scheduling. In addition, there will resources in different roles and data related to the working phase may vary even greatly.

The different statuses of the whole reservation are:

a "NoReplies" (0) for "No-one hasn't replied to the request made by the organizer"

b "NoDeclines" (1) for "Not all invitees have replied yet. The ones who have replied have accepted"

c "AllAccepts" (2) for "all invitees have confirmed"

d "SomeDeclines" (3) for "Some of the invitees have declined"

e "AllDeclines" (4) for "All of the invitees have declined".

The following decision table helps in evaluating the status of the whole booking. "Maybe" means that this condition only does not incontestably specify true or false result.

Booking Status \ Confirmations	No one answ-ered	No one accep-ted	Some Accep-ted	All accept-ed	No one dec-lined	Some dec-lined	All dec-lined
NoReplies	True	Maybe			Maybe		
NoDeclines	True	Maybe	Maybe	True	True		
NoAccepts	True	True			Maybe	Maybe	True
AllAccepts			True	True	Maybe		
SomeAccepts			True	Maybe	Maybe	Maybe	
AllDeclines		Maybe					True
SomeDeclines		Maybe	Maybe			True	Maybe

Based on the information and decision table above the organizer/application has to make the decision of what to do with the reservation. That can be an automatic decision made by the system based on pre-set rules or made by the organizer manually.

One major problem solved by the invention is the challenge of managing client replies, when a client has been given a number of questions and the client is using SMS text messages or similar technology in which a reply does not automatically include an explicit reference to the inquiry. The invention solves this problem using dynamic *dialog matrices*. An inquiry always includes some kind of receiver's address or identification. In the SMS text message case that is so called *B subscriber's* number. On the other hand, sender's *A subscriber's* number or Calling Line Identity (CLI), or similar identification is also attached to each text message. Therefore the client or B subscriber is usually easily able to answer a message using mobile device's answer or reply function. If a mediator service that sends inquiries to a client, uses different A

subscriber numbers in different inquires, it is possible to differentiate between answers based on which number the client is sending replies to. For example, if a mediator sends a client an inquiry "Do you need a taxi also?" using A subscriber number A1 and then inquiries "Do you need a hotel room?" from A subscriber number A2, client's reply to the first question goes to number A1 and the second answer goes to number A2. Using a dialog matrix, a mediator keeps track on inquires and answers. In the matrix, there is a column for each client and a row for each A subscriber number the mediator is using. Obviously, there could be a row for each client and correspondingly a column for each A subscriber number as well. After sending an inquiry from a certain A subscriber number to a client, the status and the reply is stored in the corresponding shell of the matrix. As a result, the mediator is able to find out whether the client has replied to a certain inquiry and what the answer was. Also, it is possible to use the matrix to collect information about clients' behavior and use it for example for marketing purposes. A mediator needs only a limited number of A subscriber numbers. A dialog matrix can also be used to find out which A subscriber numbers can be used when the next inquiry to a certain client is sent.

The use of the Dynamic Dialog Matrix as described above is illustrated in Figure 7.

The Dynamic Dialog Matrix is also a powerful but very simple security measure for authenticating a mobile phone user who has only the capability of sending and receiving messages. The problem is for a service to confirm a sender's identity. One way to try to identify the user is to check the sender's address. Normally SMS, e-mail, and other alike messages have the sender's address attached.

That address can be for example the sender's *A-subscriber*'s number or *Calling Line Identity* (CLI), or e-mail address or IP address. However, it is quite easy to falsify a sender address. From the service provider's perspective, the downlink from a service provider to a user is usually relatively reliable and it is hard for others to capture or change messages, but the uplink from a user to a service provider is much more vulnerable and it is not too difficult to give a wrong sender's address. A well-known solution to the above problem is to use encryption technologies to secure the communications, public-key infrastructures (PKI) being good examples. For instance, a user device can be equipped with a microchip, a secure SIM card in GSM devices for example, to encrypt messages using the user's private key. Then the service provider can be sure that the message is from the user, if it can be decrypted using the user's public key. However, this solution requires special devices that are not very common, inexpensive, or standardized so far. Relying on such a solution restricts the number of potential users significantly.

Using the DDM provides a novel solution. When the service sends a request to the mobile phone user, each request contains a different, preferably randomly chosen, reply number. Thus an acceptable answer is only the one that is sent to the correct reply address.

Example 2 - Use of the Dynamic Dialogue Matrix

This simple example deals with securing tickets on a morning flight tomorrow. The system sends a series of questions as SMS messages requiring a short response. Each message is earmarked so that its response can be identified

so the messages need not necessarily be sent or replied to in a particular sequence unless logic so demands (for instance, if the answer to one question affects the content of the next question).

A user whose phone number is ID = 0418 979 813 has requested the ticket. The system sends the following requests as individual SMS messages:

Please choose one of the following departure times:

6:00 a.m. , answer A

7:30 a.m. , answer B

8:15 a.m. , answer C.

If none of these is OK, answer D.

Sender: +358440844 027

Please choose ticket class :

First class , answer A

Business class , answer B

Economy class, answer C

Cheapest available, answer D

Sender: +358440844 011

Please choose:

Window seat , answer A

Aisle seat , answer C

Sender: +358440844 034

Please select the meal:

Vegetarian, answer A

Beef, answer B

Chicken, answer C

Sender: +358440844 003

The answers received from the customer to the preceding questions and several others were as follows:

' A ' to question with ref.no +358 440 844 027

' D ' to question with ref.no +358 440 844 011

' A ' to question with ref.no +358 440 844 034

' B ' to question with ref.no +358 440 844 003

' D ' to question with ref.no +358 440 859 751

' A ' to question with ref.no +358 440 844 277

' C ' to question with ref.no +358 440 841 368

From this, the service provider can find out that the customer chose:

- the first morning flight (= A),
- cheapest available ticket (=D),
- window seat (=A),
- beef for meal (=B),

and etc.

It is important to note with the matrix the customer can answer the questions in any order, and can even fail to answer some questions. If these are relevant, the system can urge for an answer. If not, the system can proceed without this information.

The above responses are shown on Figure 8 as a three dimensional matrix with customer numbers plotted on the X-axis, reply numbers are plotted on the Y-axis and answers plotted on the Z-axis. Our user with phone number 0418 979

813 is the left most user along the X-axis. The answers are plotted along the Z-axis corresponding to the reply numbers on the Y-axis.

Additional security can be achieved using semantic analysis. In the matrix shells, there can be information about the inquiry and what kinds of answers are acceptable. If an answer does not meet the criteria, it is rejected. For example, if the service provider asks the user to tell how many items are ordered, and the user answers "yes", then apparently the user did not know what the question was and the message was not an answer for the inquiry.

It is also possible that the service provider is actually a mediator and the "real" service provider is somewhere else. In that case, only the mediator needs to have the matrix-based system and the actual service provider communicates with the mediator using either the mediator's matrix-system or other secure means like a crypto-channel. For example, a car sharing system could be implemented in the following way: cars are located randomly around a city. When a user needs a car, he or she sends a message to a mediator to ask where the nearest car is. The mediator sends a message telling the car's location. That reply comes from a random address y'. When the user reaches the car, he or she sends a message to y' telling that the rental period begins and asking the mediator to remotely release the car's locks. This message is relatively reliable, because it is sent to the address that the user only knows. Therefore it constitutes a valid reason to release the locks and start billing. The communication between mediator and the car, on the other hand, is invisible to the user and outsiders. The car can be equipped with special devices and therefore remote commands

to release the locks etc. can be encrypted. Or, the communication between the car and the mediator could also be implemented using matrices. In either case, the mediator operates as a "firewall" between the user and the car disabling outsiders from unauthorized usage.

Although the present invention has been described in considerable detail with reference to certain preferred versions thereof, other versions are possible. Therefore the spirit and scope of the appended claims should not be limited to the preferred versions herein.

What is claimed is:

1. A method of a mediator carrying on a communication with a client terminal having a client identifier address, including:

a) initializing a communication with the client terminal, including associating a particular reply address to which a reply to a dialogue needs to be directed, including selecting the particular reply address from a multiplicity of addresses at which the mediator receives replies;

b) initiating a dialogue to the client terminal that includes the particular reply address;

c) receiving a reply at the particular reply address from the client terminal, the reply including the client identifier address; and

d) evaluating the reply using the client identifier address and the particular reply address at which the reply is received.

2. The method of claim 1, wherein evaluating the reply further includes analyzing the semantics of the reply.

3. The method of claim 1, wherein the communication pertains to a particular service provider.

4. The method of claim 1, wherein selecting the particular reply address from the multiplicity of addresses is a random choice.

5. The method of claim 1, wherein the communication includes a plurality of dialogue and reply exchanges with the client terminal and initiating includes associating different reply addresses with different dialogue and reply exchanges.

6. The method of claim 5, whereby evaluating the reply can proceed even when the different replies are received in a different order than the exchanges are initiated.

7. The method of claim 3, wherein initializing a communication is responsive to a set up request that identifies the client terminal and particular service provider.

8. The method of claim 4, further including storing the replies in a matrix having a first axis corresponding to client identifier addresses and a second axis corresponding to reply addresses.

9. The method of claim 7, wherein the mediator is simultaneously communicating with a plurality of other client terminals each having a different client identifier address, and wherein the method further includes tracking which of the multiplicity of addresses is currently available.

10. The method of claim 8, wherein initializing a communication further includes selecting the particular reply address from currently available addresses.

11. A mediator that controls communications between at least one service provider and a client terminal having a client identifier address, the mediator including:

a) a multiplicity of addresses at which the mediator is capable of receiving messages from the client terminal;

b) logic and resources adapted to

i) initialize a communication with the client terminal pertaining to a particular service provider, including associating a particular reply address to which a reply to a dialogue needs to be directed, the particular reply address being selected from the multiplicity of addresses;

ii) initiate a dialogue to the client terminal that includes the particular reply address;

iii) receive a reply at the particular reply address; and

iv) evaluate the reply using the client identifier address and the particular reply address at which the reply is received.

12. The mediator of claim 11, wherein the logic and resources to evaluate the reply further analyzes the semantics of the reply.

13. The mediator of claim 11, wherein the communication includes a plurality of dialogue and reply exchanges with the client terminal, and different reply addresses are associated with different dialogue and reply exchanges.

14. The mediator of claim 13, whereby the logic and resources are adapted to process replies to dialogues even when the different replies are received out of order from the different queries.

15. The mediator of claim 11, wherein the logic and resources adapted to initialize the communication includes a matrix having a first axis corresponding to client identifier address and a second axis corresponding to reply address.

16. The mediator of claim 11, wherein the logic and resources to initialize the communication is adapted to be responsive to a set up request that identifies the client terminal and the particular service provider.

17. The mediator of claim 11, wherein the logic and resources to select the particular reply address from the multiplicity of addresses chooses the selection at random.

18. The mediator of claim 11, wherein the client identifier address is chosen from the group consisting of a client's A-subscriber's number, Calling Line Identity, e-mail address and IP address.

19. A mediator for controlling a booking system having at least one service provider and a client terminal device for use by a client having a client identifier address, the mediator including:

a) a multiplicity of addresses at which the mediator is capable of receiving messages from the client terminal,

b) logic and resources adapted to

prepare at least one dialogue responsive to an inquiry pertaining to the service provider,

associate a particular reply address to each dialogue, the particular reply address being selected from the multiplicity of addresses,

send the at least one dialogue to the client terminal,

receive a reply at the particular reply address from the client terminal device including the client identifier address; and

evaluate the reply using the client identifier address and the particular reply address at which the reply is received.

20. The mediator of claim 19, wherein the logic and resources includes a matrix including a first axis indexed by client identifier address and a second axis indexed by reply address.

21. The mediator of claim 19, wherein the client identifier address includes an identifier chosen from the group consisting of a client's A-subscriber's number, Calling Line Identity, e-mail address and IP address.

22. The mediator of claim 19, wherein the logic and resources to evaluate the reply further analyzes the semantics of the reply.

23. The mediator of claim 19, wherein the inquiry is responsive to a service request that identifies the client terminal device.

24. The mediator of claim 19, wherein a plurality of dialogues are prepared responsive to the inquiry and wherein the logic and resources adapted to associate a particular reply address are adapted to select a different particular reply address to associate with each dialogue, whereby the logic and resources are adapted to evaluate replies even when different replies are received in a different order from different dialogues.

25. The mediator of claim 24, wherein the logic and resources includes logic and resources to track those addresses of the multiplicity of different addresses that are currently available and to assign the particular reply address at random from those addresses that are currently available.

26. The mediator of claim 24, wherein the at least one dialogue is a question which can be answered by making a selection of one item from an ordered list of choices wherein each choice has an ordinal position.

27. The mediator of claim 26, wherein the matrix further includes a third axis index for storing the ordinal position of the selection.

28. A method of a mediator booking resources for a client from a service provider, the client having a client identifier address and using a client terminal device to communicate with the mediator, the mediator performing acts including:

a) preparing at least one dialogue responsive to an inquiry;

b) associating a particular reply address to each dialogue, said particular reply address being selected from a multiplicity of addresses available for receiving messages from a client terminal device;

c) sending the at least one dialogue to the client terminal device;

d) receiving a reply to the at least one dialogue from the client terminal device, the reply including the client identifier address; and

e) evaluating the reply to the at least one dialogue using the client identifier address and the particular reply address.

29. The method of claim 28, wherein the act of evaluating the reply further includes analyzing the semantics of the reply.

30. The method of claim 28, further including storing the reply to the at least one dialogue in a matrix including a first axis indexed by client identifier and a second axis indexed by reply address.

31. The method of claim 28, wherein the act of associating a particular reply address to each dialogue is performed so as to select the particular reply address at random.

32. The method of claim 28 wherein the inquiry is responsive to a service request from the client terminal device.

33. The method of claim 28, wherein the client identifier address includes an identifier chosen from the group consisting of a client's A-subscriber's number, Calling Line Identity, e-mail address and IP address.

34. The method of claim 28, wherein the act of preparing the at least one dialogue further includes preparing the at least one dialogue into a form that can be answered by making a selection from an ordered list of choices wherein each choice has an ordinal position in the list of choices.

35. The method of claim 28, wherein the at least one dialogue includes a plurality of dialogues such that different particular reply address is associated to each dialogue, whereby evaluating the reply can proceed even if replies are received in a different order than dialogues are sent.

36. The method of claim 34, wherein the matrix includes a third axis indexed for storing the ordinal position of selections, and wherein the method further includes storing the selection to the at least one reply along the third axis.

37. The method of claim 34, wherein the client terminal device includes a mobile telephonic device and the dialogues include SMS messages.

38. A network server programmed as a mediator to perform the method of claim 28.

39. A network server programmed as a mediator to perform the method of claim 34.

40. A method of a mediator authenticating a client, the client using a mobile telephonic device capable of sending and receiving SMS messages and having a client's calling line identifier number, the mediator performing acts including:

a) assigning a unique reply address to an SMS dialogue from a multiplicity of available reply addresses;

b) sending the SMS dialogue to the client at the client's calling line identifier number; and

c) authenticating the client if a reply to the SMS dialogue is received at the unique reply address.

41. The method of claim 40, wherein the unique reply address is assigned at random from among the multiplicity of available reply addresses.

42. The method of claim 40, wherein the mediator includes a network server programmed to perform the method and wherein the method further includes storing the reply in a matrix including a first axis indexed by client calling line identifier number and a second axis indexed by reply address.

43. The method of claim 40 wherein the authenticating act further includes analyzing the semantics of the reply.