

5

10

15 “System and method for processing complex queries”

20 TECHNICAL FIELD

The present invention relates in general to the field of data processing. More specifically it concerns the handling of complex queries which here means that each query involves to search for and to retrieve from a database a multiplicity of items to be associated together to present to the user sets of said items combined together in a logical way.

One non limitative application is the processing of search requests in the field of the travel and tourism industry when a user wishes to book a bundle of services, such as a combination of air travel services with hotel accommodation and/or car rental which are very different components by nature.

30

BACKGROUND

Today's search systems are well efficient for processing requests for one type of items. In the travel industry field, search engines are now widely used and return travel solutions are sent in reply to an end user request. This is the case for engines dedicated to the airline industry such as low fare search engines providing the user with a set of low price flight solutions to be displayed on a friendly graphical interface.

35

The success of these techniques led the industry to propose computerized tools for enlarging the scope of the searches, for returning packages of products. This is particularly the case in the travel industry where the customer often needs a bundle of services when traveling. Products to be packaged can include flight tickets, hotel  
5 nights, and tourism activities.

Figure 1 depicts how a state-of-the-art technique builds product packages. A plurality of data repositories Da, Db and Dc are used and each stores travel products of one type. By way of example, Da, Db and Dc respectively contain information about air seats, about hotel rooms and about car rental services. A server S is used as a central  
10 node of the system. It receives user requests and returns travel packages solutions from end users over a network such as a wide area network preferably the internet. Upon receipt of a request, the server S communicates a plurality of individual queries each targeting one search engine Ea, Eb, Ec respectively associated with one of the repositories Da, Db, Dc. Local and independent searches are performed at the level of  
15 the search engines Ea, Eb, Ec and products are returned to the server S. The later server then aggregates the products to construct travel packages made of a plurality of products.

Such a search system is complex and involves multiple hardware/ software computer resources.

## 20 SUMMARY

The invention aims at overcoming at least some of the drawbacks of current techniques for processing complex queries.

One technical problem the invention intends to solve is to optimize the processing of complex queries where multiple items of different natures need to be  
25 searched and combined to produce a response to a user request. The invention provides an efficient solution for searches based on heterogeneous data and triggered by complex queries.

In a more specific manner, the present invention relates to a method for processing a data search based on a complex query for at least one set of combined  
30 elements matching search parameters, at least two of the combined elements being of different element types. Said method preferably comprises the query processing steps of:

- i) obtaining definition data of elements of a least two different element types, said definition data having heterogeneous data structures, said definition  
35 data comprising at least one element fare for each element,

- ii) converting said definition data into converted definition data sharing one single generic data structure,
- iii) receiving the complex query at a search engine,
- iv) at the search engine, finding elements of at least two different element types whose converted definition data matches the query and combining said elements, the step of combining comprising:
  - associating element fares of said elements,
  - building at least one fared set of combined elements based on association of element fares
- v) and returning at least one fared set of combined elements.

The at least one set of claims combined elements is/are preferably the best set(s) with regard to evaluation parameters such as, but not systematically, the price of the set. In the later case, the best set(s) is (are) the cheapest one(s).

Thanks to the invention, searches can be performed on the basis of very heterogeneous data sources. Heterogeneous data structures here means that at least two of the element types are defined by data of different data structures. The presentation of the data and/or the data fields describing the elements of said element types are thus different from one source of data to at least another and this means that no single search engine is normally capable of searching among these heterogeneous sources. Whereas current systems cope with this problem by multiplying the search engines (one for each source) and by implementing post-processing means to aggregate the results obtained from the plurality of engines, the invention dramatically saves computer resources since one engine is sufficient. In addition the engine directly produces element associations without requiring a post step of aggregation.

And the results returned by the engine are much more reliable. This is because the engine directly works with elements described by definition data of the same data structure so that the search and the combination of the elements are integrated within the same process.

Further optional features which may be used cumulatively or alternatively are introduced hereafter:

- the method comprises the steps of storing the definition data in a repository and of making the search engine access the repository.
- the step of combining comprises checking if the association of element fares complies with combinatory rules.
- the complex query includes data defining a plurality of components derived from the search parameters and for which at least one element is to be searched for.

- the step of combining comprises:
  - for each component, retrieving a set of matching element,
  - associating matching element fares,
  - checking if the association of the matching element fares complies with the combinatory rules,
  - if the association complies with the combinatory rules, validating the association of the matching element fares,
  - if the association fails to comply with the combinatory rules, invalidating the association of the matching element fares.
- the method comprises repeating the preceding steps with at least a further association of matching element fares.
  - it comprises defining a plurality of components derived from the search parameters and for which at least one element is to be searched for, and defining, for each component, at least one element type for which at least one element is to be searched for.
  - the following steps are processed:
    - executing the query processing steps for a plurality of queries to build associations of matching elements,
    - upon receipt of a request of an end user, retrieving at least one of said associations which matches the request,
    - checking if the at least one of said associations complies with the combinatory rules,
    - if the at least one of said associations complies with the combinatory rules, validating the at least one of said associations,
    - if the at least one of said associations fails to comply with the combinatory rules, invalidating the at least one of said associations,
    - returning to the end user the at least one of said associations if it is a validated one.
  - the following steps are executed:
    - executing the query processing steps for a plurality of queries to build associations of matching element fares,
    - building associations of matching elements based on the associations of matching element fares;
    - checking if the associations of matching elements comply with the combinatory rules,

for each of the associations of matching elements, if it complies with the combinatory rules, validating said association, and it fails to comply with the combinatory rules, invalidating said association,

storing the validated associations or at least some of them; selected according  
5 to evaluation criteria, such as the price, to store only the best validated associations,  
upon receipt of a request of an end user, retrieving at least one of said best validated associations which matches the request,  
returning to the end user the at least one of said best validated associations.

- the definition data comprises fare data, availability data and application rule  
10 data.

- the element types include at least two of: air travel element type, railway travel element type, accommodation element type, car rental element type.

- the method is performed as a result of execution of computer software by at least one data processor, the computer software being stored in a non-transitory  
15 computer-readable memory medium, which here comprises any computer readable memory such as ROM or RAM memories.

The present invention further relates to a system for processing a data search based on a complex query comprising means for executing the method of the invention.

20 According to a preferred embodiment, the search engine is a low fare search engine.

This system may comprise computer software stored in a non-transitory computer-readable memory medium that is executed by at least one data processor that comprises part of the system.

25 The present invention also relates to a computer program product storing a computer program comprising instructions adapted to perform the method of the invention.

#### BRIEF DESCRIPTION OF THE DRAWINGS

30 The present invention will now be described in details with reference to the drawings for the purpose of illustrating the preferred embodiment.

Figure 1 illustrates an example of a state-of-the-art system for aggregating results of queries.

Figure 2 shows a preferred embodiment of one aspect of the invention related to the  
35 conversion of definition data.

Figure 3 depicts an example of conversion of definition data.

Figure 4 is a diagram illustrating steps of the method.

Figure 5 shows a checking step.

Figure 6 illustrates an embodiment of the invention with the construction of a database of associations of elements.

5 Figure 7 shows an alternative to the embodiment given in figure 6.

#### DETAILED DESCRIPTION

The invention can be implemented with computer hardware and software means. It can include a server side where the processing of data takes place. This server side -  
10 which may be comprised of single or plural computer devices- preferably communicates via network resources with at least one remote user device such as but not exclusively a desktop computer, a smart phone or the like. Communication networks used for the data transmission may include local area networks and/or wide area networks such as the internet.

15 A few definitions are given hereafter:

- "end user request" or request here means a search request sent from one device of the system and intending to obtain results corresponding to search parameters. In a preferred embodiment the request is based on search parameters at least partially inputted by a user -such as a remote user communicating with other  
20 parts of the system via a network- at a user device. In one preferred application, the request is for travel arrangement solutions comprising a plurality of items (see hereafter "component") to be associated.

- "query" here means a message sent to a search engine and specifying search criteria. Depending on the working context queries can be based or derived from  
25 requests of end users or be generated automatically at the server side without being directly linked to an actual traffic of user requests.

- "component" here means an item of an association of items used to form a solution in reply to a user request. In the case of travel arrangements, the components are typically segments each satisfying one portion of the travel arrangement defined by  
30 the search parameters. By way of example, a complex travel request may include an outbound travel component, an accommodation component and an inbound travel component. One component may include a set of items which are grouped such that they form a single entity to be searched for. For example, consecutive hotel nights may consist in one single component.

- "element type" here means a generic class definition of elements sharing the  
35 same nature or structure. In the field of the travel and tourism industry, an element type

is typically a sort of services provided to a customer such as air travel services, railways travel services, hotel accommodation services, car rental services, lunch services, and any other service a customer may need during a travel period. Element types are used to classify elements which are the unitary bricks used to build a complete solution matching the complex request. In a preferred embodiment elements are products that can be combined together to fulfill the needs expressed by the user in the request. By way of example an element can be: an air travel leg (corresponding to one flight between an origin and a destination), a car rental service, a hotel (or other accommodation) night (or a set of nights), a taxi reservation, a travel segment other than an air segment (railways, bus ...). Components can match with one element type or plural element types. For example a travel between an origin location and a destination location, corresponding to one component of a request can lead to a search for flights and for trains matching the search parameters.

The detailed description of a preferred embodiment is given hereafter in the case of travel products.

We hereby show how a typical travel product is structured in the air industry field. The airlines industry adopts a quite complex definition of the travel products to be ticketed. This definition is organized around a fare definition concept which not only gives price amount information about a travel product but also details about the application of the fare, availability indication on top on more general data (origin and destination, cabin class and service level...).

One aspect of these data is the application rules an example of which is provided hereafter: within each rule, there are *categories* that are designed by a number. A category describes the conditions in a given area. For example, we have: 1 → Eligibility... 19 → Children discount... Etc. Inside a category, there are items called *Records* 2 hereafter also simply called records. Each of them is composed of a set of elements called matching elements (loc 1, loc2, effective and discontinued dates ...) that define which record 2 is applicable for a given trip. A record 2 contains a string of records 3. Each record 3 defines a given set of restrictions for the category.

Here is an example:

- *Rule AF01*

- o *Category 1 – Eligibility*

- *Record 2*

- *Between Paris and London*

- *String with 2 records 3 linked with OR*

- o *Passenger must be a soldier*

- OR
- *Passenger must be a clergyman*
- *Category 12 – Surcharges*
  - *Record 2*
    - *Between Paris and London*
    - *String with 1 record 3*
      - *If passenger flights in Business class  
he will pay a surcharge of 2000\$*

5

10

This shows the complexity of definition of the elements when they are made of air travel products defined with fares. In addition the data structure and thus the element types may be different from one airline carrier or travel provider to another. And other travel elements (such as railways tickets, car rental or hotel rooms) have different definition data structures so that one technical problem is to work with the various data structures within the same process of response to a query.

15

20

Figure 2 depicts a first parts of the invention corresponding to a conversion stage. D1, D2 and D3 represent data repositories - the number of which is not limited – and containing definition data of elements. By way of example Air travel elements are stored in D1, Hotel room elements are stored in D2 and car rental elements are stored in D3. The definition data of the elements stored in D1, D2 and D3 have different data structures.

For example, Air travel elements may be of an element type defined by:

25

- fare data
- availability data
- rules
- TLA data, i.e. data coming from low-cost carriers
- OAG data, i.e. flight data

Hotel room elements of D2 source may be of an element type defined by:

30

- one or plural fare data
- hotel and hotel rooms definition data
- rules

And car rental elements of D3 may be of an element type defined by:

35

- one or plural fare data
- cars definition
- rules

All these elements correspond to heterogeneous items which do not share a generic data structure. Typically the data structure of each element type is specified by the administrator of the provider of the considered element.

According to the invention, a conversion of the definition data of the elements is performed. A converter as depicted in figure 2 may be used for that purpose. It is configured to output converted definition data so that all the elements are described in a single generic data structure.

According to a preferred embodiment, the generic data structure comprises:

- identification data describing the element type (such as air, car, hotel...) and possibly a brand associated to the element.
- At least, one element fare set composed of:
  - Fare data (fare amount)
  - Availabilities
  - Application rules
- Application rules may be divided in two categories:
  - internal rules which tell the conditions of application of an element fare; for example, rules may include the age of the requestor ;
  - external rules , which are conditions of element fare application constrained by the presence or non-presence of another kind of element in the association to be built (e.g. a hotel fare which requires to have flown on a specific airline). The external rules specify whether the combinability is possible with:
    - such element,
    - such brand,
    - such fare,
    - or a combination of these criteria.

The external rules are used to check the validity of association between various elements. They are hereafter called combinatory rules. In one embodiment the combinatory rules are stored in the same memory area as the other application rules. In an alternative embodiment, they are stored at a different level. They can be administered by a person differing from the provider of the elements. For example, combinatory rules may be defined by the administrator of a web based entity such as an online travel agency.

If a combinatory rule is not filed in the generic format, the converter will adapt it the same way it adapts the rest of the element definition data.

For combinatory rules, the generic data format can be similar to the category 10 of rules as defined by IATA (International Air Transport Association) for combinability of air fares.

Here is a further exemplified case of combinatory rules for a car rental element  
5 having a promotional fare:

- Combination with hotel elements

“The fare can only be used if car transportation is from or towards an IBIS® or MERCURE® hotel.

- Combination with flight elements

10 “The fare can only be used if car transportation is from or towards an AIR FRANCE® flight arriving between 8:00 am and 11:00 am or between 14:00 pm and 17:00 pm.

The results of the conversion step performed by the converter may be stored in an appropriate repository such as the cd database depicted in figure 2 and containing  
15 converted definition data of elements (cd<sub>1</sub>, cd<sub>2</sub>, cd<sub>3</sub>).

Figure 3 illustrates a further example of the definition data conversion step. In this case, hotel elements are subject to a format transformation from the structure depicted in the table of figure 2 to a converted format in which as many converted elements are created as there are lines in the table.

20 Turning now to figure 4, an embodiment is illustrated with a data flow starting from the receipt of a user request to the provision of results.

The end user – such as a potential traveler – enters search parameters by means of a computer device. Preferably this step uses a graphical user interface displayed via a browser. Not all the search parameters useful for processing the  
25 request need to be entered by the user. Some parameters may be determined by the system based, for example, on default values or on values deduced from the environment of the user request (e.g. an origin location may be defined as the user device location determined by a satellite positioning system or an IP address).

In one embodiment, the end user connects to an airline or other provider-website  
30 and can use a product selection button to choose components he wishes to book. Once a product is selected, the graphical user interface requests the user enters more details on the corresponding component. By way of example, if the component is an air product, it leads the interface to request the departure and arrival cities as well as the dates or time windows. Then the user is offered to add another component. Similarly,  
35 each new component involves the input of particulars for this part of the request.

For illustration purpose, consider a request from an origin to New-York city with accommodation, then the input of the request would involve three components: air (outbound), hotel, air (inbound), corresponding to two element types.

Preferably the request is transmitted over a communication network to a server sideNext, the definition of the relevant element types and the search parameters are used to define a query to be processed by a search engine to search for matching elements. The engine may be a conventional search engine such as a fare shopping engine used in transactional environments to return priced products. More precisely the engine may be a low fare search engine aimed at returning low price solutions. Thanks to the invention only one engine – and of a conventional kind – is sufficient to process the query which is however complex since it involves plural element types.

To search for matching elements, the engine accesses converted definition data obtained according to the previously described conversion method.

The processing of the queries is preferably driven by the element fares of the elements. As previously indicated, each element is defined by general data (identification of the element; general definition data...) and by at least one (but often plural) element fares. The association is here based on the element fares. Upon receipt of a query, candidate elements are searched for each component of the query. For the found elements, the system starts an association mechanism in which element fares of the found elements are associated. The system discards associations that failed (when their rules are not compatible) and generates successful element fares associations. The later associations are then used to build sets of combined elements to which a fare is assigned deriving from the association of the element fares. Such sets of combined elements are thus called fared sets of combined elements.

It should be noted that one element can be the source of series of element fares associations. For example, one element having one element fare A and one element fare B lead to associations based on A and or B.

In one embodiment, the best elements found for each component are combined.

However other association policies may be pursued.

Associations of elements must comply with the combinatory rules. In a preferred embodiment, this check is done dynamically with the association process: starting from one matching element fare (corresponding to an element included in a set of elements matching one component of the request), a second matching element fare (corresponding to an element included in a set of elements matching another component of the request) is considered. Their combinability is checked by accessing the combinatory rules. This check is depicted on figure 5.

If the association is invalidated, it is withdrawn from consideration by the engine which continues its search.

If the association is validated, the engine then tries to combine the association with another matching element fare of an element corresponding to still another  
5 component of the request. Again combinatory rules are checked and the process continues until a complete association of matching elements fitting the whole request has be determined and checked against the combinatory rules.

Returning to figure 4, the output of the process consists in at least one validated association of elements complying with the search parameters.

10 Another flow is illustrated in figure 6 for processing the user request.

According to this embodiment, associations of elements are built during a construction stage. They are stored for future use. This corresponds to a pre-process in which massive queries are generated at the server side without direct link to an actual user request.

15 The queries are processed by an engine which can be the same as the engine described in reference to figure 4. At end user transaction time, another engine will parse the database with all cheapest associations, and discard the ones invalid due to non-combinability.

The stored associations are accessed by the second search engine upon receipt  
20 of a user request. Matching associations are then confronted to the combinatory rules and non-complying associations are invalidated. Validated associations are returned to the user. From a user point of view, the response time is decreased thanks to this embodiment. A drawback is that all calculated associations may be invalidated by this post-filtering.

25 A variation of the embodiment of figure 6 is given in figure 7. Associations of elements are built in the same manner but they are immediately checked against the combinatory rules. This leads to validated associations then stored for future use. Upon receipt of a request, the search engine accesses the validated associations and tries to find associations matching the search parameters of the user request. Such  
30 matching validated associations can be returned to the user. From a user point of view, the response time is also decreased thanks to this embodiment.

Although illustrative embodiments of the present invention have been described in detail with reference to the accompanying drawings, it is to be understood that the invention is not limited to those precise embodiments and that changes and  
35 modifications may be effected therein by those in the art without departing from the scope and spirit of the invention.

CLAIMS

1. Method for processing a data search based on a complex query for at least one set of combined elements matching search parameters, at least two of the  
5 combined elements being of different element types, said method comprising the query processing steps of:
- i) obtaining definition data of elements of at least two different element types, said definition data having heterogeneous data structures, said definition data comprising at least one element fare for each element,
  - 10 ii) converting said definition data into converted definition data sharing one single generic data structure,
  - iii) receiving the complex query at a search engine,
  - iv) at the search engine, finding elements of at least two different element types whose converted definition data matches the query and  
15 combining said elements, the step of combining comprising:
    - associating element fares of said elements,
    - building at least one fared set of combined elements based on association of element fares,
  - v) returning at least one fared set of combined elements.
- 20 2. Method according to the preceding claim, comprising the steps of storing the definition data in a repository and of making the search engine access the repository.
3. Method according to any of the preceding claims, the step of combining comprising checking if the association of element fares complies with combinatory  
25 rules.
4. Method according to any of the preceding claims, wherein the complex query includes data defining a plurality of components derived from the search parameters and for which at least one element is to be searched for.
5. Method according to the two preceding claims in combination, the step of  
30 combining comprising:
- for each component, retrieving a set of matching elements,
  - associating matching element fares,
  - checking if the association of the matching element fares complies with the combinatory rules,
  - 35 - if the association complies with the combinatory rules, validating the association of the matching element fares,

- if the association fails to comply with the combinatory rules, invalidating the association of the matching element fares.

6. Method according to the preceding claim comprising repeating the steps of claim 5 with at least a further association of matching element fares.

- 5        7. Method according to any of the preceding claims comprising defining a plurality of components derived from the search parameters and for which at least one element is to be searched for, and defining, for each component, at least one element type for which at least one element is to be searched for.

8. Method according to claim 3, comprising:

- 10        - executing the query processing steps for a plurality of queries to build associations of matching element fares;,  
          - building associations of matching elements based on the associations of matching element fares;  
          - storing the associations of matching elements,  
15        - upon receipt of a request of an end user, retrieving at least one of said associations which matches the request,  
          - checking if the at least one of said associations complies with the combinatory rules,  
          - if the at least one of said associations complies with the combinatory  
20 rules, validating the at least one of said associations,  
          - if the at least one of said associations fails to comply with the combinatory rules, invalidating the at least one of said associations,  
          - returning to the end user the at least one of said associations if it is a validated one.

25        9. Method according to claim 3, comprising:

- executing the query processing steps for a plurality of queries to build associations of matching element fares,  
          building associations of matching elements based on the associations of matching element fares,  
30        - checking if the associations of matching elements comply with the combinatory rules,  
          - for each of the associations of matching elements, if it complies with the combinatory rules, validating said association, and if it fails to comply with the combinatory rules, invalidating said association,  
35        - storing the validated associations,

- upon receipt of a request of an end user, retrieving at least one of said validated associations which matches the request,

- returning to the end user the at least one of said validated associations.

10. Method according to any of the preceding claims wherein the definition data  
5 comprises fare data, availability data and application rule data.

11. Method according to any of the preceding claims wherein the element types include at least two of:

- air travel element type,
- railway travel element type,
- 10 - accommodation element type,
- car rental element type.

12. Method according to any of the preceding claims performed as a result of execution of computer software by at least one data processor, the computer software being stored in a non-transitory computer-readable memory medium.

15 13. System for processing a data search based on a complex query comprising means for executing the method of any of the preceding claims.

14. System according to the preceding claim wherein the search engine is a low fare search engine.

20 15. The system as in any of the two preceding claims, comprising computer software stored in a non-transitory computer-readable memory medium that is executed by at least one data processor that comprises part of the system.

16. Computer program product storing a computer program comprising instructions adapted to perform the method of any of claims 1 to 12.