

# **PERSONALIZATION OF THE SEARCH PROCESS IN TOURISM**

Doctoral Thesis

Dipl.-Inf. Sven Döring  
Faculty of Applied Computer Science  
University of Augsburg

Doering@Informatik.Uni-Augsburg.de  
© Copyright 2008. All rights reserved.

Examiners: Prof. Dr. Werner Kießling  
Prof. Dr. Bernhard Möller

Day of oral examination: 09.05.2008

## Abstract

The combination of travel and tourism represents the leading domain for applications in B2C e-commerce. Thus, it deserves highest attention. Since most people only have a very limited number of vacation days each year, they have learned to be more demanding about their trips. More and more they ask for better-personalized travel products instead of standard packages designed by tourist operators. Due to insufficient search engines and the lack of personalization, however, arranging a trip on current online travel portals is often not as easy as it should be. Even for rather straightforward scenarios, searching and booking a suitable travel package can be tedious and might often take longer than 1 hour. In order to provide good sales experiences and custom-tailored products similar to the ones competent human travel agents can offer, a personalized search approach for online travel portals has been overdue for some time. This thesis, therefore, presents a novel personalized search process delivering travel products exactly tailored to customers with respect to their situations and preferences.

In a first step, a novel model for the search process in electronic commerce will be introduced. A deep personalization of the search will be provided by dividing the process into four stages, namely *Preference Analysis & Modeling*, *Search Interface*, *Query Processing*, and *Presentation*. The main part of this thesis, will then apply the new model to the tourism domain, i.e. each step of the search will be examined in the context of tourism. A situation model adequately adjusted to the tourism domain will then provide each stage of the search process with additional situational knowledge. Based on this, several essential components for a domain specific search in tourism will be introduced accordingly: a new preference constructor dealing with typical price-quality tradeoffs, a smart preference elicitation process supporting customers who have to find an optimal departure airport, the composition and evaluation of database queries supporting the interplay of individual and global preferences, and an appropriate adaptation of search interface and product presentation. Moreover, by using preference search technologies as underlying basis for the search itself, best alternatives can be delivered in case there is no perfect match.

Several novel software components for a personalized search process in tourism have come into existence in the context of this thesis, e.g., the personalized prototype COSIMA<sup>T</sup>. The interplay of these components with existing preference components will be examined and evaluated by means of numerous use case scenarios at the end of this work. It will be demonstrated that by a proper combination of these components, custom-tailored travel products with respect to preferences and situations can be found and presented to the customer in an intuitive, fast and more comfortable manner than before.



## Acknowledgments

This work would not have been possible without the valuable support of several people I am particularly grateful to. At the chair of databases and information systems of the University of Augsburg (Germany) my doctoral adviser, Prof. Dr. Werner Kießling, provided me with an inspiring view into a world full of preferences. I would like to thank him for his support and good advice during my research.

I am deeply grateful for the incentive and joyful work with my colleagues in the department. In particular, I would like to thank Stefan Holland and Stefan Fischer for a smooth start into the *Preference World*, Timotheus Preisinger, Markus Endres, and Alfons Huhn for the good cooperation during various projects. I am also especially grateful for the support of Prof. Dr. Bernhard Möller and Anna Schwartz.

The work at FORSIP, the Bavarian Research Cooperation for Situated, Individualized and Personalized Human-Computer Interaction, offered me insights into various interesting research fields. I would like to thank all the people at FORSIP, I had the pleasure to work with.

Special thanks are due to Alex Mangold for a last-minute, linguistic review of my work.

Finally, I am very grateful to Stefanie Leistner for her support and patience.



# Contents

|  |           |
|--|-----------|
| <b>1 Introduction.....</b>   | <b>9</b>  |
| <b>2 Travel Search in Tourism .....</b>                            | <b>13</b> |
| 2.1 Customer's Experience with a Typical Travel Portal.....        | 13        |
| 2.2 An Overview of Existing Online Travel Portals.....             | 16        |
| 2.2.1 Expedia.....   | 17        |
| 2.2.2 Travelocity.....   | 18        |
| 2.2.3 TUI .....  | 18        |
| 2.2.4 Traveltainment and '5vorFlug'.....                           | 19        |
| 2.2.5 Summary .....  | 20        |
| 2.3 Objectives of this Thesis .....                                | 21        |
| <b>3 Foundations of Preferences Revisited.....</b>                 | <b>25</b> |
| 3.1 Modeling Preferences .....                                     | 25        |
| 3.1.1 Base Preferences.....  | 27        |
| 3.1.2 Complex Preferences.....                                     | 30        |
| 3.1.3 SV-Semantics.....  | 32        |
| 3.2 The Preference Framework.....                                  | 34        |
| 3.2.1 Preference Search.....                                       | 35        |
| 3.2.2 Situated Preference Model and Preference Repository.....     | 36        |
| 3.2.3 Personalized Presentation of Query Results.....              | 37        |
| <b>4 Tailoring a Personalized Search for Tourism .....</b>         | <b>39</b> |
| 4.1 Personalization of the Search Process in E-Commerce.....       | 39        |
| 4.1.1 Common Preferences in Electronic Commerce.....               | 39        |
| 4.1.2 Design Principles for a Personalized Search Process.....     | 42        |
| 4.1.3 Search Model.....  | 43        |
| 4.2 Situation Modeling for the Search Process in Tourism.....      | 45        |
| 4.2.1 Situated Entity-Relationship-Model for Tourism.....          | 46        |
| 4.2.2 Tourism-Related Preference Repository .....                  | 48        |
| 4.3 Tradeoff Preference Constructor.....                           | 49        |
| 4.3.1 Definition of the Preference Constructor.....                | 55        |
| 4.3.2 Complexity and Performance Considerations.....               | 61        |
| 4.3.3 Quality Valuation .....                                      | 62        |
| 4.4 Smart Preference Elicitation.....                              | 64        |
| 4.4.1 Preference Elicitation Based on Information Integration..... | 66        |
| 4.4.2 Smart Preference Elicitation in Conclusion.....              | 75        |
| 4.5 Advanced Preference Query Processing .....                     | 76        |
| 4.5.1 Preference Query Expansion Approach.....                     | 79        |
| 4.5.2 Algorithm and Complexity .....                               | 92        |
| 4.6 The Concepts in Retrospect.....                                | 94        |

---

|   |            |
|---|------------|
| <b>5 Preference Based Components for Tourism.....</b>                       | <b>95</b>  |
| 5.1 History of COSIMA .....   | 95         |
| 5.2 Novel Components for a Deeply Personalized Search in Tourism.....       | 97         |
| 5.2.1 Airport Finder and Travel Recommender.....                            | 97         |
| 5.2.2 Personalization of the Search Interface: Visualization Component..... | 99         |
| 5.2.3 Advanced Preference Query Processing: COSIMA <sup>T</sup> .....       | 100        |
| 5.3 Putting the Pieces Together: Interplay of Components.....               | 101        |
| 5.4 Typical Use Case Scenarios.....   | 103        |
| <b>6 Achievements and Related Work.....</b>                                 | <b>109</b> |
| 6.1 Search Process Model.....   | 109        |
| 6.2 Situational Influences.....   | 111        |
| 6.3 Integration of Heterogeneous Information Sources.....                   | 112        |
| 6.4 Search Process in Tourism.....  | 113        |
| <b>7 Summary and Outlook.....</b>   | <b>117</b> |
| 7.1 Summary of this Thesis.....   | 117        |
| 7.2 Future Work.....  | 118        |
| <b>Bibliography.....</b>  | <b>121</b> |
| <b>Appendix A.....</b>  | <b>135</b> |



# 1 Introduction

The travel and tourism industry makes up a considerable branch of our economy. According to the *World Travel & Tourism Council*<sup>1</sup>, almost 11% of the world wide GDP is represented by travel and tourism. Besides, travel and tourism is the leading application field in B2C electronic commerce; it represents nearly 50% of the total B2C turnover ([Wer03]). The travel industry and its products have rather specific features, which makes them predestined for online distribution: the product is a confidence good and has a short-living nature, customer decisions are mostly based on information, and the industry is highly networked, based on the world-wide cooperation of different types of stakeholders ([Wer03, Ben06]).

Low-cost airlines like Ryanair<sup>2</sup> or EasyJet<sup>3</sup> have gained huge significance in recent years. They strongly rely on their online portals in order to sell tickets. Nevertheless, there are still a lot of people who prefer to consult a human employee in a travel agency instead of using the internet for booking or organizing a journey. Although the travel industry already stands for an important part of electronic commerce, there is still a lot of room for improvement. The ability to conduct a task such as searching is absolutely central for travel websites ([Ben06]). However, due to technical problems, incomprehensible interfaces, and insufficient search engines, average customers are often over-strained when arranging a vacation on the internet. This might be one reason why there is still little evidence of electronic markets leading invariably to lower search costs ([ÖK03]).

Since most people have only a very limited number of vacation days each year, they have become more demanding about their trips. More and more they ask for better personalized travel products instead of standard packages designed by tourist operators. This might be one reason for the emerging concept of *dynamic packaging*. A vacation package combines hotels with flights and/or rental cars in one price. Customers become their own travel agent and build their travel packages themselves. This has been introduced to the tourism industry as an approach to achieve a competitive advantage providing customers with flexible travel packages ([CL07]). The authors also state that product packages should be customized based on the requirements specified by customers, who, in addition, are no longer content with just finding the lowest price. Customers are looking to take control and **identify the perfect trip** ([Gro07]). But this even worsens the problems mentioned above, since existing online portals

---

1 [www.wttc.travel](http://www.wttc.travel)

2 [www.ryanair.com](http://www.ryanair.com)

3 [www.easyjet.com](http://www.easyjet.com)

like Expedia<sup>4</sup> are sufficient for simple problems like inquiring a flight from A to B with specific hard constraints. More complex problems, on the other hand, require more intelligent systems ([Wer03]).

Besides tourists traveling during their vacation, there are business travelers who also need human-friendlier search engines in order to reduce their valuable time for finding and booking a business trip on the internet. Travel expenses make up some of the largest expenses in companies, second only to labor and IT. At the same time, they are an excellent target for a cost reduction which can be induced by optimizing individual processes such as searching and booking. The selection of the 'right' hotel, for example, is insufficiently described by price alone, as choosing a suitable room for a business trip is a non-trivial task. A number of a hotel's characteristics (e.g., facilities) as well as its location must be taken into consideration ([NMT06]). In addition, there are also several criteria related to the travelers and their optimal accommodation with respect to individual preferences, e.g., the quality of the hotel. Note that this comprises only one part of the travel package; taking other parts such as flights and rental cars into account will make the decision-making process even more complex.

Only recently, the travel and tourism industry has learned to take advantage of search engines as an important marketing argument in promotion. There was a widespread advertising slogan of Fly.de<sup>5</sup>, a German online portal for flights (see Figure 1.1): "Fliegst Du schon oder suchst Du noch?", which means something along the lines of 'Are you flying or still trying (to find an appropriate flight)?'.



Figure 1.1. Fly.de advertising slogan

The idea of buying travel packages via online travel portals is simple. Customers would like to have at least the service they would have when directly contacting a human employer in a travel agency. This means that customers want to be treated individually, according to his or her wishes and situation. Generic, 'one-size-fits-all' search engines will not suffice in tourism. Therefore, good search engines for tourists have to be adapted to this domain. Furthermore,

4 [www.expedia.com](http://www.expedia.com)

5 [www.fly.de](http://www.fly.de)

knowledge about the travelers' preferences and their choice behavior is required ([SBA01]). Attentive user interfaces and personalization have been emphasized as important factors for better searches that really support customers ([WR04]).

In a recent joint study of PhoCusWright (see [Hsi05]), questionnaires were sent to companies of the travel and hospitality industry. The 95 survey participants represented a mixed group of CEOs, directors, managers, vice presidents, and other travel, tourism, and hospitality professionals. Almost a quarter of the participants (24%) expect search engine optimizations to become the technology with the most important impact over the next 5 years.

In this thesis, a preference based approach will be presented in order to improve the search process of online travel portals. It aims to provide a novel personalized search process, which will be able to deliver custom-tailored products with respect to customers' preferences and situation. In the following, the search process in tourism will be examined in some detail. The impact of tedious and frustrating empty-result-effects as well as the lack of suitable preference models will be demonstrated by means of a compact market analysis. Thereafter, a novel approach for a more customer-friendly search process will be presented.



## 2 Travel Search in Tourism

The overall tourism information space is huge ([PF03]). Average customers are often overstrained arranging a vacation on the internet due to technical problems, incomprehensible interfaces, and insufficient search engines ([ÖK03]). In the following section peculiarities and challenges of the search process in tourism are presented by means of a typical travel scenario.

### 2.1 Customer's Experience with a Typical Travel Portal

As already stated, tourists have learned to demand travel packages, which have been selected and arranged to meet their individual expectations. These packages should be tailored based on each customer's preferences and situation. However, a personalized selection process of fitting products and packages is quite challenging, as the following example will demonstrate. Business traveler Mark has to travel to London for a week in order to visit project partners and potential customers. Therefore, a package consisting of flight, hotel, and rental car is necessary. Mark's company is located in Augsburg in the southern part of Germany. His needs and preferences are as follows:

1. In order to represent his company in a positive manner, a premium class car **must** be rented.
2. The accommodation **should** be in the northern part of London because of the proximity to a great deal of partners.
3. On account of good experiences in the past, the airline **should** be British Airways.
4. The limit of \$1.500 **must not** be exceeded due to the company's policy.

Points 2 and 3 describe preferences which **should** be matched, while 1 and 4 denote hard constraints which **have to** be matched. This is a reasonable scenario, since preferences are usually understood as wishes in the 'real world'. If there is no perfect match, people are not always but often prepared to accept alternatives. Note that the first three points describe wishes regarding individual aspects of the journey. By contrast, the last point represents a global constraint regarding the price of the complete package.

For the example scenario the popular online travel portal of Expedia is used. Choosing the option 'Book together & save! Flight+Hotel+Car' allows Mark to book a travel package consisting of all necessary parts (see also Figure 2.2). The web form of Expedia is offering the following possibilities:

- Itinerary with departure airport and destination
- An option to fill in two different destinations
- Dates and times of travel
- Number of rooms and travelers
- An option to use a hotel only for one part of the trip

Mark fills out the web form with dates, number of rooms and travelers, and his destination airport London. Munich is chosen as his departure airport.

The system now returns a list of possible combinations and corresponding total prices. Since Mark is familiar with Expedia, he knows the option 'Hotel map view', which shows the locations of hotels on a map of the city. Subsequently, he picks an adequately located hotel in the northern part of London. After this, he manually opts for the desired flight with British Airways. In the next step, he chooses a premium class rental car. Only now Mark may recognize the violation of his hard constraint regarding the price limit.

Consequently, Mark has to find an acceptable solution. He chooses combinations of cheaper hotels and flights. In either case he has to curtail softer constraints (preferences) in order to match the hard price limit. After a while Mark remembers the smaller airport of Nuremberg which is also close to his location in Augsburg. As a consequence he has to plan the trip all over again. Finally, after evaluation and comparison of his results, he chooses an acceptable solution. However, he was not aware of Stuttgart Airport, which is almost as close to Augsburg as the airports of Munich or Nuremberg.

This rather simple example scenario demonstrates a number of problems:

- After opting for a travel package, there are only a few search criteria left to be specified by customers in current travel portals. Many criteria that may be important to customers cannot be specified. For instance, a preferred airline and the quality of a hotel or a rental car cannot be specified in the search form provided by Expedia after opting for 'Book together & save! Flight+Hotel+Car'.
- In most existing systems, constraints and preferences are not modeled explicitly, but remain implicit in selections made by the customer. What is more, most of the search engines support only the specification of hard constraints and solutions are only shown if all constraints are matched. Compromises have to be made manually by the customer if the empty-result-effect occurs. Most customers would like to see alterna-

tives if there is no perfect match. Often the entire search has to be repeated, causing a very tedious and frustrating search process. It is not uncommon to spend more than 1 hour organizing a trip as described above.

- The lack of preference modeling in current systems does affect tourism even worse, since travelers typically have a wide variety of preferences. As demonstrated above, even in simple scenarios there are preferences regarding individual aspects of the journey, for example the flight, and there are preferences about global constraints, e.g., for the total price of an entire package. Today, even important travel portals like Expedia do not allow the customer to specify global constraints such as the total price. Furthermore, there are preferences which **should** be fulfilled and hard constraints which **must** be fulfilled, respectively. Due to missing preference modeling, customers are often left alone to manually configure their trip in countless search sessions.
- In practice, customers are not aware of all constraints until they see them violated. For instance, Mark may not think about a preference for a non-stop flight until a system offers him a connection with one or more intermediate stops. Customers would like to add newly recognized preferences in an easy manner instead of repeating the entire search process over and over again.
- Travel portals often force customers to make a decision based on incomplete knowledge. For example, most systems require the specification of a departure airport. In the scenario above, Mark was not aware of the airport in Stuttgart. Therefore, he possibly missed a flight that would have matched his preferences better. Often a customer just wants to start from an airport which provides a flight to the destination, is close to his or her location, and is reachable in a cheap way. In order to specify an appropriate airport they have to use other information sources like online trip finders. Taking such preferences of customers into consideration exceeds the capabilities of current systems by far because it requires the integration of heterogeneous information sources.

A personalized search process for online travel portals is necessary in order to provide a good sales experience similar to the one competent human travel agents can offer. Customers' preferences have to be taken into account to offer personalized services. Preferences and constraints can be different, depending on the customer's situation. For example, a business traveler may have other preferences than a family father. As demonstrated, the modeling of situations and wishes in tourism is a non-trivial process (see Figure 2.1). It is even harder to deliver an optimal result with respect to the customer's situation and wishes, since there are a huge number of dependencies. For instance, the optimal flight might depart on Monday, while the preferred hotel is not yet available, or the best combination of hotel, flight, and the car might exceed the price limit.

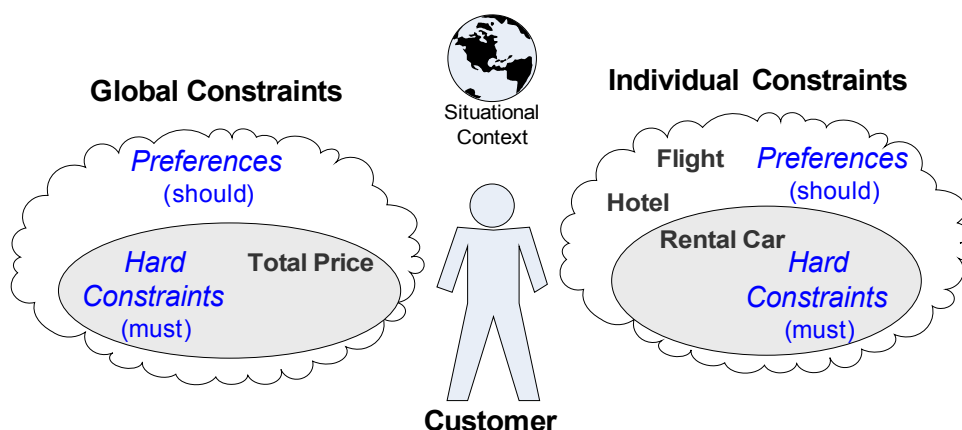


Figure 2.1. Modeling and satisfying customers' wishes in tourism: a non-trivial matter

## 2.2 An Overview of Existing Online Travel Portals

Both the non-trivial modeling process of customers' wishes and situations as well as the misery of current search engines which are often lacking any kind of personalization have led to insufficient, antiquated occurrences. For example, there is a new German traveling website called MyJack24 ([www.myjack24.de](http://www.myjack24.de) [Myj07]) which does not offer any search engine for customers at all. Instead, customers looking for a journey can formulate and enter their wishes on the portal. After a few days they will or won't get human-made offers from a registered travel agency. Obviously, this approach has several disadvantages. Customers have to be passive for a few days while waiting patiently at home. There is no warranty that they will get a good offer or even any offer at all. The portal is not really attractive for travel agencies either. There is no customer loyalty, but the effort to make a good offer is the same as for customers of the traditional travel agency. A customer may have gotten a huge deal of competing offers before the registered travel agency even notices his or her wishes. Hence, such an approach is not a promising alternative.

In the following, a sample of common online travel portals will be examined. The portals presented here are all offering a packaging option for flight, hotel, and rental car. Traveling sites solely offering standardized, ready-made journeys designed by tourist operators (such as cruises on the South Seas) have not been considered because they do not offer any personalization. However, customers' expectations have shifted from ready-made journeys for the uniform, standard human being to custom-tailored personalized trips, which meet their individual requirements ([CL07]).

Based on the problems mentioned above, the focus will be on the following issues:

- (A) Which constraints can be specified on the web form after choosing the option for a travel package?
- (B) Can the frustrating empty-result-effect occur?



- (C) Is it possible to specify global constraints, e.g., for the total price of the package?
- (D) Is there any support for the selection of an airport based on customers' constraints?

### 2.2.1 Expedia

Expedia Inc. is the parent company of the popular travel portal Expedia (www.expedia.com [Exp07], see Figure 2.2), representing the largest online travel agency in the world. In an article called 'What a Site: 36 Web Addresses You Should Know' of the Washington Post ([Was07]), Expedia was ranked as number 1 in the category of 'all-purpose booking' travel sites.

#### Expedia.com

#### Travelocity.com

Figure 2.2. Search forms of Expedia and Travelocity

In the following Expedia is analyzed with regard to issues (A – D) defined above.

- (A) When choosing the packaging option, customers are able to specify the following wishes: itinerary, dates and times, and the number of rooms and travelers. Customers may also choose an option to fill in two different destinations or to use the hotel only for one part of the trip.
- (B) It is unlikely that customers experience the empty-result-effect here because of the limited amount of wishes that can be specified on the search form. However, some-

times only pieces of the desired package are offered. Several times, only a package consisting of hotel and flight was offered, since there was no matching rental car (see Figure 2.3).

- (C) It is not possible for customers to specify global constraints, e.g., for a price limit.
- (D) There is no support for customers trying to find appropriate airports with respect to their preferences. But the search automatically considers the vicinity of major cities. For example, when choosing Munich as the departure airport, flights starting from the small airport of Augsburg will be added automatically. This may have a negative effect; if the results are not reviewed carefully, e.g., a customer might accidentally book a flight from Augsburg, although he or she was looking for a flight from Munich.

**Create your package from Munich to Madrid (and vicinity)**

Book your trip online or call our reservation agents toll-free at **1-800-509-1974**.

i **We were unable to find any rental cars that could be combined with your specific vacation package. Please choose a flight + hotel vacation package from the choices below, then use the car search tool to reserve a rental car.**

**Figure 2.3. Empty-result-effect: no rental car can be offered**

### 2.2.2 Travelocity

Travelocity (www.travelocity.com [Tra07a], see Figure 2.2) represents the second largest online travel agency of the world. It was ranked as number 3 in the category of 'all-purpose booking' travel sites by the Washington Post ([Was07]).

- (A) After choosing the packaging option, even less wishes can be specified than on Expedia. Customers are only able to specify itinerary, dates and times, and the number of rooms and travelers.
- (B) Since only a few wishes can be specified by customers, the probability of an empty result is low. If indeed there was no perfect match, customers would be asked to change the dates of the travel.
- (C) There is no possibility to specify global constraints.
- (D) Customers who are unsure about their optimal departure airport are not supported by the system.

### 2.2.3 TUI

The online travel portal TUI (www.tui.de [Tui07]) is provided by TUI interactive, which belongs to the mother company TUI AG. The TUI AG represents the leading traditional traveling company in Europe.

- (A) In contrast to the portals above, more wishes can be specified here by customers: itinerary, dates and times, and the number of travelers. Moreover, wishes regarding the flight are possible: category, airline, and the option to consider only non-stop flights. For the hotel, customers may specify a hotel chain, kind of meals, and the minimum category.
- (B) Unfortunately, the drawback for a bigger amount of wishes which can be expressed by customers is immediately apparent. Specifying a lot of search constraints can often lead to the empty-result-effect.
- (C) Global constraints regarding the entire travel package cannot be specified.
- (D) There is no support for customers trying to find a suitable airport.

#### 2.2.4 Traveltainment and '5vorFlug'

Note that the packaging option of the German online portal 5vorFlug ([www.5vorflug.de](http://www.5vorflug.de) [Fue07]) is restricted to flights and hotels. Rental cars cannot automatically be added to the package. The portal was chosen for this overview because it is based on the internet booking engine (IBE) of Traveltainment ([www.traveltainment.de](http://www.traveltainment.de) [Tra07b]), which is supposed to provide an advanced search. After having recognized the problem of unsatisfied customers frustrated by the empty-result-effect and a tedious search process, Traveltainment aims to deliver alternatives by using a fuzzy logic. Yet, as will be shown, this is no remedy either.

- (A) Choosing the option for a package consisting of hotel and flight, a customer is able to specify the following wishes: itinerary, earliest departure and latest return, duration of the trip, number of travelers, and a preferred tour operator. In addition, customers may also specify some wishes for a hotel: the category, the kind of room, meals, the hotel's name, and options about the location of the hotel or its leisure time facilities. Moreover, a total price limit can be specified by the customer.
- (B) A search based on fuzzy logic is no remedy for the search problem. The empty-result-effect may frequently occur. In particular, reasonable alternatives a customer could expect to get from a good human appointee are not delivered. For example, if the customer specifies a price limit of \$400 for his or her desired trip, no result will be delivered even if there is a reasonable alternative for only \$410.
- (C) It is possible to specify a limit for the total price of the trip.
- (D) Instead of filling in one departure airport, it is also possible to specify an area. For example, a customer may choose the area of 'Southern Germany' for departure. But this will often overwhelm the customer with a lot of results. Besides, even when the customer has been shown different departure airports, he or she may not be able to choose a suitable one.

### 2.2.5 Summary

Popular online travel portals have been examined above via a compact analysis. On some travel portals like TUI, a huge number of wishes can be expressed by customers. While it enables a detailed expression of wishes, it also raises the probability of the annoying empty-result-effect (Figure 2.4). For this, the rigid and unnatural treatment of customers' wishes as hard constraints turns out to be the main reason. As demonstrated above, searches based on fuzzy logic are no remedy either, since natural reasonable alternatives cannot be delivered. Other portals such as Expedia try to avoid this problem by allowing only a very limited number of wishes. This way, often several matches are delivered to customers, who are then forced to manually browse the descriptions of hotels, flights, and rental cars in order to get the best match (see also Figure 2.4). This so-called *flooding effect* can be tedious and frustrating too.

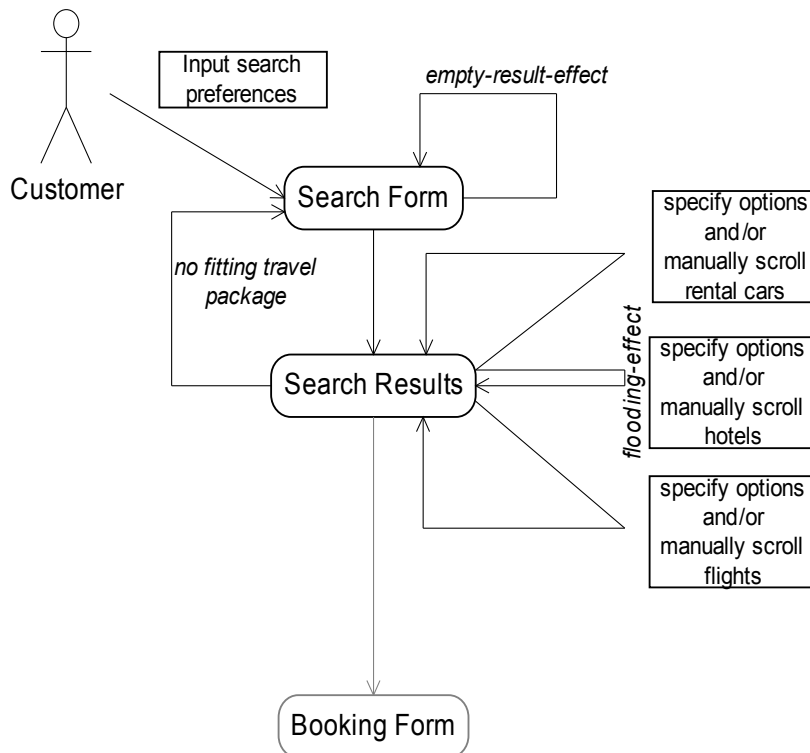


Figure 2.4. Search process on current travel portals

However, at first customers have to find an optimal departure airport on account of the time and costs to get there. One airport might be better reached by car, while another airport might be cheaper to reach by train. None of the portals provide customers with the appropriate support. Insufficient solutions might make it even worse. Since Expedia automatically includes airports of close cities in its offer generation, customers have to carefully check all the offers in order to avoid a negative surprise. An area based search as offered by 5vorFlug often returns a flood of results.

Only the portal 5vorFlug offers a possibility to specify a global constraint, that is, the total price of the package. However, this will often cause the empty-result-effect as long as no better modeling of the customer's wishes is used.

## 2.3 Objectives of this Thesis

Advanced and customer-friendly applications of the travel and tourism industry require a high level of personalization and situation awareness in order to provide individual recommendations and custom-tailored travel packages. A foundation for personalized applications was laid out by the fundamental work about preferences in databases introduced by Kießling ([Kie02, Kie05]). To offer tourists and travelers better search results and to make the search process more comfortable and less tedious in general, this thesis will be based on Kießling's semantically rich preference model. In these foundations, preferences are modeled as strict partial orders in 'A is better than B' semantics. Moreover, by usage of a preference based search engine, best alternatives can be delivered automatically in case of an empty result. In the following, novel research aspects and engineering contributions which are dealing with problems and challenges as described above are presented:

### 1. Modeling the search process in e-commerce

At first, a holistic approach for the personalization of the entire search process in e-commerce applications is presented. It is based on a cyclic model of the search including a preference analysis and modeling process, the construction of the search interface, the processing of database queries, and a presentation of search results. A situation model which influences each stage of a personalized search process is placed at the center.

### 2. Personalization of the search process in tourism

In the main part of this thesis, this new model for search processes is applied to the tourism domain as an important part of electronic commerce. That is, each step of the search has to be examined in the context of tourism. First, a situation model dealing with the peculiarities of tourism is presented. Built on this, several components for the domain specific search in tourism are modeled: namely a new preference constructor dealing with typical price-quality tradeoffs, a smart preference elicitation process supporting customers who have to find an optimal departure airport, the composition and evaluation of database queries supporting individual and global preferences, and an adaptation of the search interface and product presentation to tourism.

### 3. Engineering personalized search components for travel portals

The research has led to the development of several essential middleware components as well as to the implementation of a deeply personalized and situated prototype

called COSIMA<sup>T</sup>. Thereby, skills can be automated that so far could only have been executed by a human appointee in a travel agency.

#### 4. Evaluation by means of use case scenarios

Advantages of a situated and personalized search process in tourism will be demonstrated on the basis of several typical use cases. It will be shown how the interplay of personalized search components might lead to a customer experience similar to one with a human employee in a travel agency.

A personalized and situated approach has to take a huge amount of information into account. This means that a lot of knowledge is necessary in order to apply the results of this thesis. In the tourism domain a lot of work has been done in the field of customer-choice-behavior and psychology which has been trying to identify the most important situational variables influencing customers. Cognitive distances have been examined as well as personality or lifestyle. Yet, since the detection and specification of such knowledge belongs to other research fields, it is not part of this thesis. Nevertheless, there will be some remarks on how to deal with this separate problem. The scope of this work is illustrated in Figure 2.5.

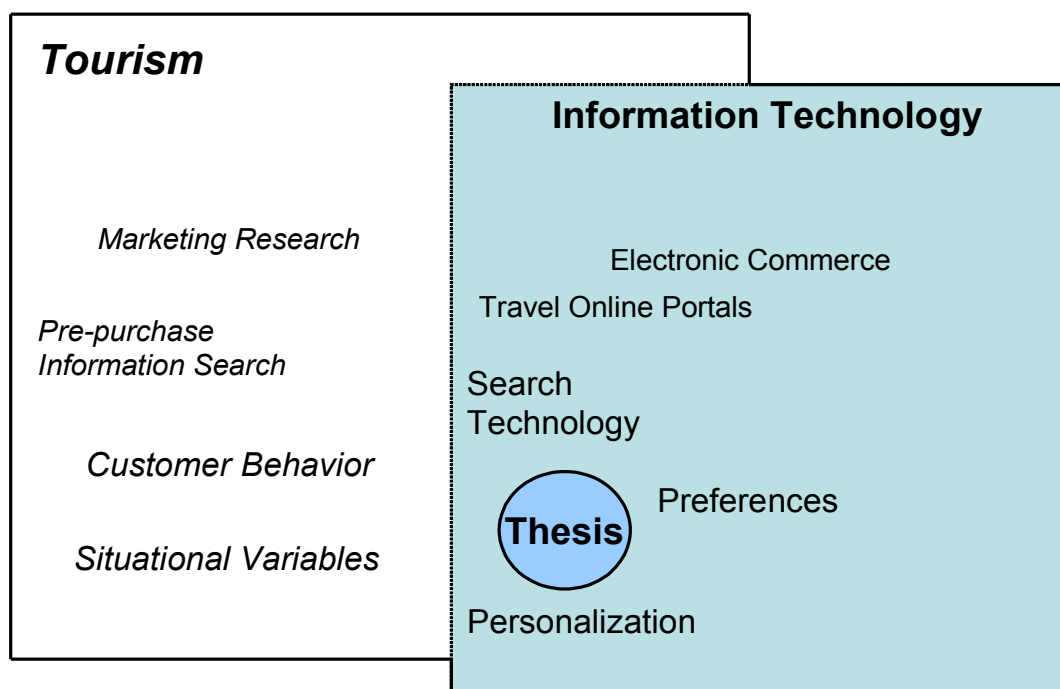


Figure 2.5. Positioning of this thesis

The contribution of this work covers the modeling of the search process with respect to personalization in e-commerce and in tourism in particular, so that the foundation for a deeply personalized and situated search process on online travel portals can be provided. It aims to provide a better search for those customers who have at least some ideas about their trip. This work does not cover the pre-purchase information search in tourism, which is, e.g., responsi-

---

ble for the first identification of the customers' preferences as described in [PM00]. In the pre-purchase stage, the customer is looking for information in order to identify his or her own preferences and needs such as: "Do I want to relax at the beach, or do I want an adventure in the mountains?" However, this work covers the subsequent phase, which supports customers who already know the characteristics of their trip. The general information search of travelers on the internet as described in [PF01, PF03] or a keyword based search for potential destinations ([Mit05, GW03]) are also not part of this work.

Thus, the essential issues of this thesis are:

- How can the search process of electronic commerce applications be modeled in order to maximize the level of personalization and situation awareness?
- Online travel portals represent an important part of electronic commerce. What kind of adaptation is necessary to apply the new model to this domain?
- Are there components which suit the model and deal with the described problems of current online travel portals?
- How about the implementation and engineering of such advanced components?
- Is a computer system able to offer services which so far only human agents in travel agencies could offer?

This work is organized as follows. At first a preference model is considered as an underlying basis that can be used to deal with preferences of customers. Since this model represents the foundation of the thesis, an overview is given in the next chapter. After that, a holistic model for a deep personalization of the entire search process in electronic commerce is specified. This model will then be applied to the tourism domain. Therefore, a situation model adapted to tourism is presented. Based on this, several advanced components for a customer-friendly search process are modeled and engineered. Finally, by means of numerous use case scenarios, the advantages and benefits of personalized search components will be demonstrated.





## 3 Foundations of Preferences Revisited

As stated before, advanced e-applications and information systems require a high level of personalization and situation awareness in order to provide individual recommendations, personalized advice, or custom-tailored products. Taking the preferences of customers into account is a promising approach for personalization. For this purpose a well-founded preference model is necessary. Several models have been presented and discussed in recent years (see [AW00, Cho02, Cho03, FPZ05, KI05, Kie02, Kie05]).

Within the research program ‘It’s a Preference World’ at the University of Augsburg many preference based technologies for e-commerce have already been developed. They are based on the foundations of Kießling ([Kie02, Kie05]), which provide a sophisticated, semantically rich model for preferences as well as a variety of intuitive preference constructors for numerical and categorical data. In the following, a brief description of this preference model is given. It is the basis for a deep personalization of advanced e-applications in tourism.

### 3.1 Modeling Preferences

Every child learns to formulate wishes in terms like “I like A better than B”. This kind of preference modeling is universally applied and intuitively understood by everybody. Moreover, in mathematics one can easily map ‘better-than’ preferences to strict partial orders. In [Kie02, Kie05] wishes are formulated as strict partial orders and can be engineered to complex multidimensional preference constructs.

#### Definition 3.1 Preferences

Let  $A = \{A_1, A_2, \dots, A_k\}$  be a set of attributes  $A_j$  with corresponding domains  $\text{dom}(A_j)$ . The domain of  $A$  is defined as  $\text{dom}(A) := \text{dom}(A_1) \times \dots \times \text{dom}(A_k)$ .

- A **preference P** on a set of attributes  $A$  is defined as  $\mathbf{P} = (A, <_{\mathbf{P}})$ , where  $<_{\mathbf{P}} \subseteq \text{dom}(A) \times \text{dom}(A)$  is a *strict partial order* (i.e., irreflexive and transitive).  $x <_{\mathbf{P}} y$  with  $x, y \in \text{dom}(A)$  is interpreted as “I like  $y$  better than  $x$ ”.

- The **indifference** relation  $\parallel_P \subseteq \text{dom}(A) \times \text{dom}(A)$  is defined as:  
 $x \parallel_P y$  iff  $\neg(x <_P y) \wedge \neg(y <_P x)$
- A preference P is a **chain** (synonym: **total order**) if for all  
 $x, y \in \text{dom}(A), x \neq y: x <_P y \vee y <_P x$
- A preference P is an **anti-chain** iff  $<_P = \emptyset$ . The anti-chain on A is denoted as  $A^{\leftrightarrow}$ .
- A preference P is a **weak order**, iff negative transitivity holds, i.e., for all  
 $x, y, z \in \text{dom}(A): \neg(x <_P y) \wedge \neg(y <_P z)$  implies  $\neg(x <_P z)$
- The **maximal** values of  $P = (A, <_P)$  are defined as:  
 $\text{max}(P) := \{v \in \text{dom}(A) \mid \neg \exists w \in \text{dom}(A): v <_P w\}$  D

Generally,  $\parallel_P$  is reflexive and symmetric, but not transitive. If P is a weak order, then  $\parallel_P$  is transitive.

For ease of use, a constructor-based approach for the specification of preferences  $P=(A, <_P)$  is applied. There are base preference constructors and complex preference constructors. The following notation will be used in this work.

### Definition 3.2 Preference notation

- Definition of a **base preference constructor**:  
*base bname(A, paramlist) {definition of  $<_P$ }*
- Defining a base preference P:  
 $P := \text{bname}(\text{actual\_}A, \text{actual\_}params);$
- Definition of a **complex preference constructor**:  
*complex Pref<sub>1</sub> cname Pref<sub>2</sub> {definition of  $<_{\text{Pref}_1 \text{ cname Pref}_2}$ }*
- Defining a complex preference P:  
 $P := \text{actual\_}Pref_1 \text{ cname actual\_}Pref_2;$  D

The syntactic terms 'base' and 'complex' mark the beginning of a preference constructor's definition. For this purpose relevant attributes, parameters, and a definition of the strict partial order have to be specified. Afterwards, preferences can be defined/engineered just by using the corresponding constructors with the actual attributes and parameters. In Definition 3.4, for example, the SCORE<sub>d</sub> preference constructor will be defined.

A good visual representation of preferences is possible by so-called better-than graphs.

### Definition 3.3 Better-than graph

In finite domains, a preference P can be drawn as a directed acyclic graph G, called the 'better-than' graph (BTG) of P. Note that the BTG is also known as Hasse diagram ([DP90]). Given G for P the following quality notions between values x, y in G are defined:

- $x <_P y$ , if  $y$  is predecessor of  $x$  in  $G$ .
- Values in  $G$  without a predecessor are maximal elements of  $P$ , being at level 1.
- $x$  is at level  $j$  if the longest path from  $x$  to a maximal value has  $j-1$  edges.
- If there is no directed path between  $x$  and  $y$  in  $G$ , then  $x$  and  $y$  are indifferent.  $\mathcal{D}$

For an intuitive and comfortable engineering of preferences the given preference constructors are described, starting with numerical base preferences.

### 3.1.1 Base Preferences

When dealing with numerical scores, it is often helpful to group ranges of scores together, e.g., for “payment due within two weeks”. In order to deal with such real-world scenarios, a so-called  $d$ -parameter was introduced ([Kie05]).

#### Definition 3.4 SCORE<sub>d</sub>

Given a utility function  $f: \text{dom}(A) \rightarrow \mathbb{R}$  and some  $d \in \mathbb{R}_0^+$ , one defines for all  $v \in \text{dom}(A)$ :

- $f_d: \text{dom}(A) \rightarrow \{\text{if } d=0 \text{ then } \mathbb{R} \text{ else } \mathbb{Z}\}$ , where  $f_d(v) := \{\text{if } d=0 \text{ then } f(v) \text{ else } \lceil f(v) / d \rceil\}$
- base SCORE<sub>d</sub>( $A, f$ )  $\{x <_P y \text{ iff } f_d(x) < f_d(y)\}$ ;  $\mathcal{D}$

SCORE<sub>d</sub> constructs a weak order ([Kie05]). Note that values with identical  $f_d$ -values become indifferent:  $x \parallel_P y$  iff  $f_d(x) = f_d(y)$ . As demonstrated below, certain indifferent values can be interpreted as ‘substitutable’ or ‘equally good’. Now several sub-constructors of SCORE<sub>d</sub> are presented, focusing on preferences  $P = (A, <_P)$ , where  $A$  is a single attribute with a numerical domain, i.e.,  $\text{dom}(A) \subset \mathbb{R}$ .

#### Definition 3.5 Preference sub-constructor

$C_2$  is a preference sub-constructor of  $C_1$ , if the definition of  $<_{C_2\text{-new}}$  can be gained from  $<_{C_1\text{-new}}$  by some specializing constraints.  $\mathcal{D}$

For instance, the following preference constructor BETWEEN<sub>d</sub> is a sub-constructor of SCORE<sub>d</sub>.

#### Definition 3.6 BETWEEN<sub>d</sub>

Given  $v, \text{low}, \text{up} \in \text{dom}(A)$  and  $\text{low} \leq \text{up}$ , the distance of  $v$  from the closed interval  $[\text{low}, \text{up}]$  is defined as follows:

- $\text{dist}[\text{low}, \text{up}]: \text{dom}(A) \rightarrow \mathbb{R}_0^+$

- $\text{dist}[\text{low}, \text{up}](v) := \begin{cases} \text{if } v \in [\text{low}, \text{up}] \text{ then } 0 \text{ else} \\ \text{if } v < \text{low} \text{ then } (\text{low} - v) \text{ else } (v - \text{up}) \end{cases}$

Given  $d \in \mathbb{R}_0^+$  distances are grouped together as follows:

- $\text{dist}_d[\text{low}, \text{up}]: \text{dom}(A) \rightarrow \{\text{if } d = 0 \text{ then } \mathbb{R}_0^+ \text{ else } \mathbb{N}_0\}$
- $\text{dist}_d[\text{low}, \text{up}](v) := \{\text{if } d = 0 \text{ then } \text{dist}[\text{low}, \text{up}](v) \text{ else } \lceil \text{dist}[\text{low}, \text{up}](v) / d \rceil\}$

Then the preference constructor  $\text{BETWEEN}_d$  is defined as follows:

- base  $\text{BETWEEN}_d(A, [\text{low}, \text{up}]) \{x <_p y \text{ iff } \text{dist}_d[\text{low}, \text{up}](y) < \text{dist}_d[\text{low}, \text{up}](x)\};$

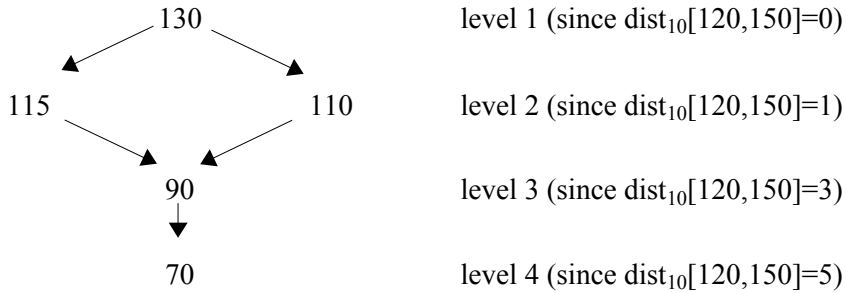
Note the reversal of  $x$  and  $y$  on the right hand side of the equivalence. For preferences constructed by this kind of distance function it holds that values with less distance are preferred. For  $d > 0$  a  $\text{BETWEEN}_d$  preference can be envisaged as a one-dimensional dart board: Perfect matches hit the interval  $[\text{low}, \text{up}]$  at  $\text{dist}_d$  being 0, second bests are those with  $\text{dist}_d$  being 1, and so on. Values with identical  $\text{dist}_d$ -values become indifferent.  $\triangleright$

### Example 3.1 $\text{BETWEEN}_d$

Meredith prefers rental cars with engines providing between 120 HP (horsepower) and 150 HP because they offer a good ratio between performance and fuel consumption. Differences of up to 10 HP can be grouped together. This preference can be defined as:

$P_{31} := \text{BETWEEN}_{10}(\text{POWER}, [120, 150])$ .

Given  $\text{dom}(\text{POWER}) = \{130, 115, 110, 90, 70\}$ , the 'better-than' graph as described above is:



$\diamond$

Special cases of the  $\text{BETWEEN}_d$  constructor are obtained by identifying  $\text{low} = \text{up} = z$  (setting  $\text{dist}_d[z] := \text{dist}_d[z, z]$ ) and by choosing  $z$  as the finite infimum or supremum of  $\text{dom}(A)$ . These sub-constructors are defined as follows.

### Definition 3.7 $\text{AROUND}_d$ , $\text{LOWEST}_d$ , and $\text{HIGHEST}_d$

- base  $\text{AROUND}_d(A, z) \{x <_p y \text{ iff } \text{dist}_d[z](y) < \text{dist}_d[z](x)\};$

- base  $\text{LOWEST}_d(A) \{x <_P y \text{ iff } \text{dist}_d[\text{inf}_A](y) < \text{dist}_d[\text{inf}_A](x)\};$
- base  $\text{HIGHEST}_d(A) \{x <_P y \text{ iff } \text{dist}_d[\text{sup}_A](y) < \text{dist}_d[\text{sup}_A](x)\};$  D

### Example 3.2 $\text{LOWEST}_d$

Christina would like to visit her friend Isobel. Since her budget is limited, she prefers the cheapest flight. Price differences of up to \$20 do not matter. This preference can easily be expressed with  $P_{32} := \text{LOWEST}_{20}(\text{PRICE})$ . ◇

People often like or dislike values of a categorical attribute, e.g., George likes IBIS and ARCOR hotels, but absolutely avoids HILTON hotels because of his limited budget. Now preference constructors for categorical data are presented which do not require any numerical operation to define a preference order.

### Definition 3.8 $\text{LAYERED}_m$

Let  $L = (L_1, \dots, L_{m+1})$ ,  $m \geq 0$ , be an ordered list of sets with the following properties:

- $L$  is a partition of  $\text{dom}(A)$ .
- Exactly  $m$  out of these  $m+1$  sets are given as finite enumerations of values from  $\text{dom}(A)$ .
- The remaining set is specified as ‘other values’.

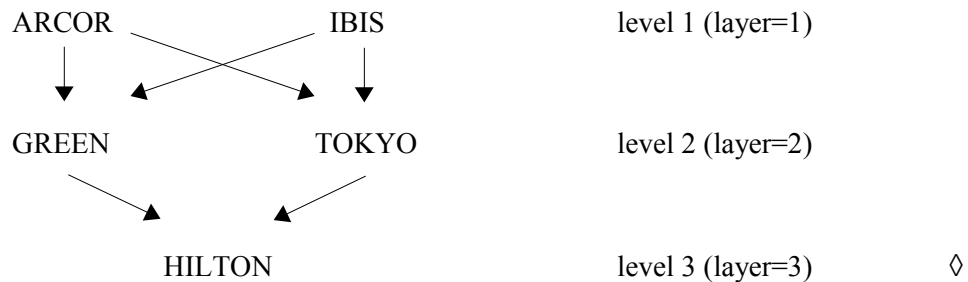
The function  $\text{layer}: \text{dom}(A) \rightarrow \mathbb{N}$  is defined as follows:

- for  $i \in \{1, \dots, m+1\}$ , for all  $v \in L_i$ :  $\text{layer}(v) := i$ .
- base  $\text{LAYERED}_m(A, L) \{x <_P y \text{ iff } \text{layer}(y) < \text{layer}(x)\};$  D

$\text{LAYERED}_m$  is a sub-constructor of  $\text{SCORE}_d$ , specializing  $d = 0$  and  $f(v) = -\text{layer}(v)$ .

### Example 3.3 $\text{LAYERED}_m$

Let George's preference for hotels be  $P_{33} := \text{LAYERED}_3(\text{HOTEL}, (\{\text{'IBIS'}, \text{'ARCOR'}\}, \text{'other values'}, \{\text{'HILTON'}\}))$ . Given  $\text{dom}(\text{HOTEL}) = \{\text{'ARCOR'}, \text{'IBIS'}, \text{'GREEN'}, \text{'HILTON'}, \text{'TOKYO'}\}$ , the 'better-than' graph is:

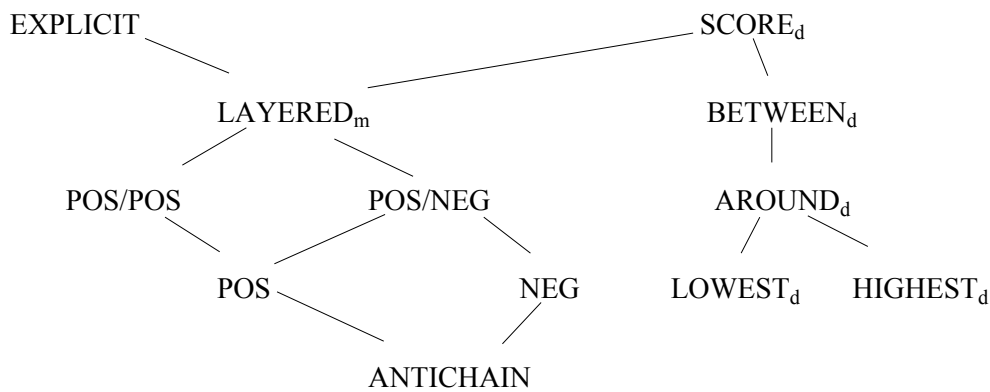


**Definition 3.9 EXPLICIT**

Given an E-graph =  $\{(val_1, val_2), \dots\}$  representing a finite acyclic ‘better-than’ graph as described in Definition 3.3,  $V$  be the set of all  $val_i \in \text{dom}(A)$  occurring in E-graph. A strict partial order  $E = (V, <_E)$  is induced as follows:

- $(val_i, val_j) \in \text{E-graph}$  implies  $val_i <_E val_j$
- $val_i <_E val_j \wedge val_j <_E val_k$  imply  $val_i <_E val_k$
- $\text{base EXPLICIT}(A, \text{E-graph}) \{ x <_P y \text{ iff } x <_E y \vee (x \notin \text{range}(<_E) \wedge y \in \text{range}(<_E)) \}$  D

Note that the EXPLICIT preference constructor is no sub-constructor of  $\text{SCORE}_d$ . However, there are several other constructors like POS/POS, POS/NEG, POS, and NEG as introduced in [Kie02], which can be defined as sub-constructors of  $\text{LAYERED}_m$  (see [Kie05]). The sub-constructor hierarchy of base preferences is shown in Figure 3.1.



**Figure 3.1. Base preference sub-constructor hierarchy ([Kie05])**

**3.1.2 Complex Preferences**

In everyday life preferences are often more complex than described above. In [Kie02] three possibilities for the accumulation of preferences are defined. There are the pareto preference, prioritized preference, and numerical complex preference.

In order to combine equally important preferences, the pareto preference can be used. The pareto-optimality principle ([Sam83]) has been studied for multi-attribute decision problems in the social and economic sciences for years. Contrarily, the prioritized preference can be applied if one preference is more important than another one. Note that pareto and prioritized preferences will be defined for  $n = 2$  preferences in the following (generalizing it to  $n > 2$  is obvious).

**Definition 3.10 Pareto preference**

In order for  $y = (y_1, y_2)$  to be better than  $x = (x_1, x_2)$ , it is not tolerable that  $y$  is worse than  $x$  in any  $y_i$ . Given the preferences  $P_1 = (A_1, <_{P_1})$  and  $P_2 = (A_2, <_{P_2})$ , for  $x, y \in \text{dom}(A_1) \times \text{dom}(A_2)$ , the pareto preference is defined as follows:

- complex  $P_1 \otimes P_2$   $\{(x_1, x_2) <_{P_1 \otimes P_2} (y_1, y_2) \text{ iff } (x_1 <_{P_1} y_1 \wedge (x_2 <_{P_2} y_2 \vee x_2 = y_2)) \vee (x_2 <_{P_2} y_2 \wedge (x_1 <_{P_1} y_1 \vee x_1 = y_1))\}$ ;  $\text{D}$

The maximal values of  $P$  form the pareto-optimal set.

**Definition 3.11 Prioritized preference**

$P_1$  is considered more important than  $P_2$ .  $P_2$  is respected only where  $P_1$  does not mind: Given  $P_1 = (A_1, <_{P_1})$  and  $P_2 = (A_2, <_{P_2})$ , for  $x, y \in \text{dom}(A_1) \times \text{dom}(A_2)$ , one defines:

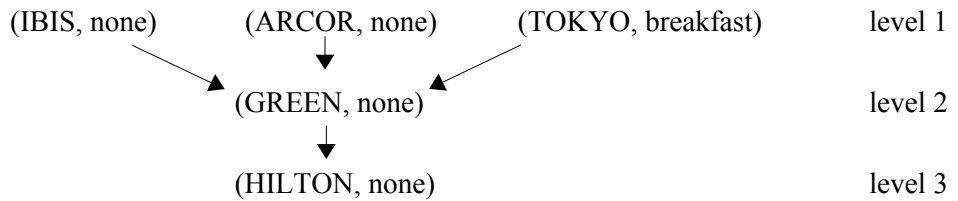
- complex  $P_1 \& P_2$   $\{(x_1, x_2) <_{P_1 \& P_2} (y_1, y_2) \text{ iff } x_1 <_{P_1} y_1 \vee (x_1 = y_1 \wedge x_2 <_{P_2} y_2)\}$ ;  $\text{D}$

**Example 3.4 Pareto and prioritized preference**

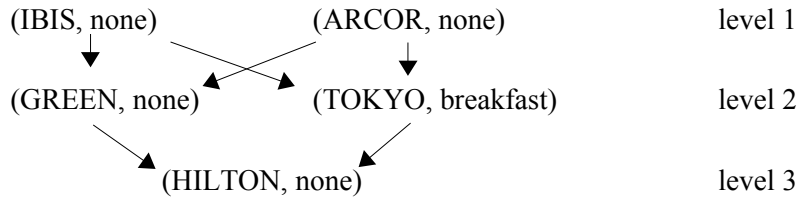
George prefers ARCOR and IBIS hotels, while avoiding HILTON hotels ( $P_{33}$  in Example 3.3). Furthermore, he would like to have breakfast included in his hotel. These two preferences are equally important to him and can be expressed as:

$P_{33} \otimes P_{34}$  where  $P_{33} := \text{LAYERED}_3(\text{HOTEL}, ( \{ 'IBIS', 'ARCOR' \}, 'other values', \{ 'HILTON' \} ))$  and  $P_{34} := \text{LAYERED}_2(\text{MEALS}, ( \{ 'breakfast' \}, 'other values' ))$

Given  $R(\text{HOTEL}, \text{MEALS}) = \{ (\text{ARCOR}, \text{none}), (\text{IBIS}, \text{none}), (\text{GREEN}, \text{none}), (\text{HILTON}, \text{none}), (\text{TOKYO}, \text{breakfast}) \}$ , the corresponding 'better-than' graph is:



Let us assume getting the right hotel (e.g., IBIS) is more important to him. This leads to  $P_{33} \& P_{34}$  while all else being equal. The 'better-than' graph is:



Note that GREEN and TOKYO hotel are on the same level, since they are indifferent in  $P_{33}$ . An extension of the preference model has already been specified introducing the definition of equally good values. It will be presented in the next subsection.  $\diamond$

The preference model also offers the possibility to accumulate numerical preferences built on SCORE preferences by applying a multi-attribute combining function  $F$  (see [Kie02]).

### Definition 3.12 Numerical preference

Given  $P_1 = \text{SCORE}(A_1, f_1)$ ,  $P_2 = \text{SCORE}(A_2, f_2)$ , and a combining function  $F: \mathbb{R} \times \mathbb{R} \rightarrow \mathbb{R}$ , for  $x, y \in \text{dom}(A_1) \times \text{dom}(A_2)$ , it is defined:

- complex  $P_1 \text{ rank}_F P_2 \{(x_1, x_2) <_{P_1 \text{ rank}_F P_2} (y_1, y_2) \text{ iff}$   
 $F(f_1(x_1), f_2(x_2)) < F(f_1(y_1), f_2(y_2))\}$ ;  $\rhd$

Note that in contrast to  $\otimes$  or  $\&$ ,  $\text{rank}_F$  can only be applied to SCORE preferences. However, numerical preferences as well as all other base preferences can be used as input for pareto and prioritized complex preference constructors.

### 3.1.3 SV-Semantics

A distinctive feature of strict partial orders is that indifferent values may exist. People often have an opinion about better-than relationships for a selected choice of options, but without being complete. For some values they do not mind or some values may be equally good for them in a particular scenario. Therefore, an extension of the preference model has been specified enabling the definition of substitutable (synonym: equally good) values ([Kie05]).

### Definition 3.13 SV-relation

Given  $P = (A, <_P)$ ,  $\cong_P$  is called substitutable values relation (SV-relation for short) iff for all  $x, y \in \text{dom}(A)$ :

- $x \cong_P y$  implies  $x \parallel_P y$
- $x \cong_P y \wedge \exists z : z <_P x$  implies  $z <_P y$
- $x \cong_P y \wedge \exists z : x <_P z$  implies  $y <_P z$
- $\cong_P$  is reflexive, symmetric, and transitive.  $\rhd$

Obviously, '=' is an SV-relation for each preference  $P$  (called trivial SV-relation). Indifferent values that are not substitutable are called alternative values, which cannot be substituted for each other. However, for  $\text{SCORE}_d$  it turns out that the full indifference relation is a valid SV-relation.



**Definition 3.14 Regular SV-relation for SCORE<sub>d</sub>**

Given a SCORE<sub>d</sub> preference  $P$ , let us define for all  $x, y \in \text{dom}(A)$ :  $x \cong_P y$  iff  $x \parallel_P y$

- $\cong_P$  is an SV-relation (called regular SV-relation).
- If  $P$  is not a chain, then  $\cong_P$  may be non-trivial. D

All values with equal  $f_d$ -values can be treated as ‘substitutable’ or ‘equally good’. But this behavior does not hold for other constructors such as EXPLICIT, which are no weak orders. In order to integrate the semantics of SV-relations into the preference model, the definition of preferences has been extended.

**Definition 3.15 Preferences with SV-semantics**

Enriching Definition 3.1 the following notation is used:

- A preference  $P$  with an SV-relation  $\cong_P$  is denoted as:  $P = (A, <_P, \cong_P)$
- Each base constructor receives one additional parameter for the SV-relation. D

Consider a base preference  $P_i = (A_i, <_{P_i}, \cong_{P_i})$ . Then  $\cong_{P_i}$  does not affect  $<_{P_i}$  itself, but expresses that it is admissible to substitute  $\cong_{P_i}$ -values for each other. Thus, a complex constructor  $C$ , using  $P_i$  recursively in its definition for  $<_P$ , can make use of  $\cong_{P_i}$ . In the following the complex preference constructors of pareto and prioritized preference are accordingly enriched.

**Definition 3.16 Pareto and prioritized constructors**

Assume  $P_1 = (A_1, <_{P_1}, \cong_{P_1})$  and  $P_2 = (A_2, <_{P_2}, \cong_{P_2})$ .

a) Pareto constructor ‘ $\otimes$ ’

- $A_1$  and  $A_2$  do not overlap: complex  $P_1 \otimes P_2 \{(x_1, x_2) <_{P_1 \otimes P_2} (y_1, y_2)$  iff  
 $(x_1 <_{P_1} y_1 \wedge (x_2 <_{P_2} y_2 \vee x_2 \cong_{P_2} y_2)) \vee (x_2 <_{P_2} y_2 \wedge (x_1 <_{P_1} y_1 \vee x_1 \cong_{P_1} y_1))$ ;  
 $(x_1, x_2) \cong_{P_1 \otimes P_2} (y_1, y_2)$  iff  $x_1 \cong_{P_1} y_1 \wedge x_2 \cong_{P_2} y_2$ };
- Otherwise: Identify overlapping attributes above.

b) Prioritized constructor ‘ $\&$ ’

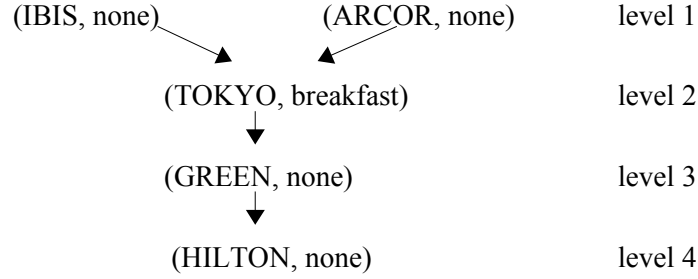
- $A_1$  and  $A_2$  do not overlap: complex  $P_1 \& P_2 \{(x_1, x_2) <_{P_1 \& P_2} (y_1, y_2)$  iff  
 $x_1 <_{P_1} y_1 \vee (x_1 \cong_{P_1} y_1 \wedge x_2 <_{P_2} y_2)$ ;  
 $(x_1, x_2) \cong_{P_1 \& P_2} (y_1, y_2)$  iff  $x_1 \cong_{P_1} y_1 \wedge x_2 \cong_{P_2} y_2$ };
- Otherwise: Identify overlapping attributes above. D

Note that  $\cong_{P_1 \otimes P_2}$  and  $\cong_{P_1 \& P_2}$  are recursively defined using  $\cong_{P_1}$  and  $\cong_{P_2}$ .

### Example 3.5 Prioritization with SV-semantics

Let us revisit Example 3.4. Getting the right hotel (e.g., IBIS) is still more important to George than the breakfast included. Using SV-semantics his preferences can be defined as:  $P_{33^*}$  &  $P_{34}$

with  $P_{33^*} := \text{LAYERED}_3(\text{HOTEL}, (\{'IBIS', 'ARCOR'\}, \text{'other values'}, \{'HILTON'\}), \cong_p)$  where  $\cong_p$  is regular. Thereby, the 'better-than' graph of Example 3.4 changes to:



By using the regular SV-relation, TOKYO hotel is preferred to GREEN hotel since it offers breakfast. In this scenario, the preference modeling seems to be more natural. However, whether to use or not to use the SV-semantics depends on user and situation.  $\diamond$

For a complete survey of the preference model and for a description of the *Preference Algebra* refer to [Kie02, Kie05].

## 3.2 The Preference Framework

The preference model introduced above has been used to develop advanced query languages and related technologies. These flexible components form the *Preference Framework* presented in this section.

Personalized constraints may be hard conditions or preferences, i.e., soft conditions. Whether preferences can be satisfied depends on the current database contents, capturing the status of the real world. Thus a match-making between wishes and reality has to be accomplished. For this purpose the BMO query model (“Best Matches Only”) has been introduced.

### Definition 3.17 Preference selection, BMO-size

For a given relation  $R(A_1: \text{dom}(A_1), \dots, A_m: \text{dom}(A_m))$  a preference  $P = (A, <_P, \cong_P)$  is considered, where  $A \subseteq \{A_1, \dots, A_m\}$ . Let  $P^R$  denote the subset preference obtained by restricting  $P$  from  $\text{dom}(A)$  to  $\pi_A(R)$ , i.e., to  $A$ -values occurring in  $R$ .

- **Preference selection**  $\sigma[P](R)$  is defined as:  $\sigma[P](R) = \{t \in R \mid t[A] \in \max(P^R)\}$
- $t \in \sigma[P](R)$  is a **perfect match** iff  $t[A] \in \max(P)$ .

- $\text{card}(\sigma[P](R))$  is called the **BMO-size** of  $\sigma[P](R)$ . D

$\sigma[P](R)$  retrieves all maximal values from  $R$ . They are not necessarily perfect matches of  $P$ . Any non-maximal values of  $P^R$  are excluded; hence can be considered as discarded on the fly. Thereby, only best matching tuples are retrieved. Note that SV-semantics and  $d$ -parameter as described above present an important means to influence the BMO-size and to combat the flooding effect. In [DEP05] a heuristic, statistics based approach has been presented to affect the result set's size.

### 3.2.1 Preference Search

#### Preference SQL

The search engine Preference SQL (PSQL) extends the well-known database query language SQL with preferences under a strict partial order semantics ([KK02]). In addition to standard hard conditions specified in the WHERE clause, PSQL enables soft conditions by usage of the keyword PREFERRING. Preference SQL includes a variety of base preferences as well as pareto accumulation ('AND'-operator) and prioritization ('PRIOR TO'-operator).

#### Example 3.6 Preference SQL query

Using Preference SQL syntax the hotel scenario of Example 3.5 can be expressed as:

```
SELECT * FROM hoteltable PREFERRING
hotel LAYERED(('IBIS', 'ARCOR'), OTHERS, ('HILTON')) REGULAR
PRIOR TO meals LAYERED(('breakfast'), OTHERS); ◇
```

#### Preference XPath

XML ([Bra06]) has become a very important standard for the exchange, storage, and presentation of data in e-commerce. The XML Path Language (XPath, [CD99]) provides comfortable possibilities to access fragments from an XML data structure. However, only hard conditions are supported. Preference XPath extends this standard query language by the capability to formulate preferences as soft selection conditions. The syntax extension is fully compatible with the XPath standard, enabling both hard and soft conditions ([KHF+01]). Note that hard conditions are syntactically framed by '[...]', whereas preferences are scoped by '#[...]#'.

#### Example 3.7 Preference XPath query

Formulating Example 3.5 in Preference XPath leads to the following expression:

```
/HOTELS #[
hotel LAYERED(('IBIS', 'ARCOR'), OTHERS, ('HILTON')) REGULAR
PRIOR TO meals LAYERED(('breakfast'), OTHERS)]# ◇
```

### 3.2.2 Situated Preference Model and Preference Repository

In real life a customer's preferences typically vary due to different situations. For instance, a customer may have various preferences for news depending on the underlying temporal situation: on Monday he or she likes to be informed about the sport results of the weekend, on Friday he or she is interested in the weather forecast, and on other days his or her preferred news are local politics. Preferences not only depend on different situations but also on the various roles a customer may have. For example, a customer may have the preference to get breakfast included when traveling for business, while preferring no meals in hotels during a private journey.

In [Hol04, HK04] a meta model of situation-oriented entities and relationships is presented. Thereby, the *Situation* is the most general entity type of situation models. *Timestamp* denotes the date and time of situations and the entity type *Location* can describe the current position. Attributes for timestamp can be SQL data types like date, time, time zone, etc. Attributes for the location are, for example, city, zip-code, or global positioning system coordinates (GPS). *Influences* describe other aspects affecting a situation. *Personal Influences* denote human factors of a situation such as physical state or current emotion. *Surrounding Influences* describe outer influences such as weather conditions or other people the current user is together with. Each situation can consist of one timestamp and of one location, but it can have one or more influences (e.g., a personal and a surrounding influence). A timestamp, location, or influence can be part of more than one situation.

In order to administer situated preferences, an XML based storage structure called *Preference Repository* was designed ([HK04]). An XML-based Preference Repository can be easily accessed either via XPath/Preference XPath or from object-oriented programming languages such as Java or C++ by using the document object model (DOM). Preference Repositories based on XML can be interchanged between various personalized applications. Moreover, relevant meta information about the situational context as described above can be managed ([Hol04]).

#### Example 3.8 Preference Repository

Alex has the preference to get breakfast included, when traveling for business. An excerpt of the appropriate storage structure of the Preference Repository is shown below.

```
<PreferenceRepository>
  <UserIdentifier>
    <Name xml:lang="en">Alex</Name>
  </UserIdentifier>
  <PreferenceData name="meal_business">
    <Situation>
      <Conditionkey="role" value="business"/>
    </Situation>
  </PreferenceData>
</PreferenceRepository>
```

```

<Preference>
  <POS att="Meals">
    <Value val="breakfast"/>
  </POS>
</Preference>
</PreferenceData>
</PreferenceRepository>

```

◇

### 3.2.3 Personalized Presentation of Query Results

A decisive factor for a successful deal is to argue about the quality of the presented products with respect to the search preferences of the customer ([HS69]). Based on a human-comprehensible linguistic model of five quality categories, an intuitive framework for valuations is defined in [Fis04]. This model represents the maximum of different categories a human normally recognizes. Therefore, it is appropriate for the valuation of search results.

#### Definition 3.18 Linguistic model for the quality of a BMO search result

- The quality of a BMO search result is described by the linguistic variable:  
**PREF\_QUAL**
- Domain of PREF\_QUAL: ('perfect', 'very good', 'good', 'acceptable', 'sufficient')

The domain for these quality valuations is defined as a descending ordered list. D

A valuation of 'perfect' denotes results matching the customer's preferences perfectly or nearly perfectly, while the valuation 'sufficient' can be used for results far away from the preference of the customer.

A situated quality function  $QUAL_{P,s}$  is used in order to assign quality valuations to search results of the BMO-set. Let  $V := (v_1, \dots, v_5)$  be the descending ordered list of linguistic quality terms of PREF\_QUAL and  $C(s) := \{C_1(s), \dots, C_5(s)\}$  be a partition of  $\text{dom}(A)$  into 5 parts depending on the situation  $s$ . Then the quality function  $QUAL_{P,s} : \text{BMO} \rightarrow V$  for a search result tuple  $t \in \text{BMO}$  with  $t[A] \in \text{dom}(A)$  regarding a preference  $P := (A, <_P)$  and a situation  $s$  is defined as follows:

#### Definition 3.19 Situated quality function $QUAL_{P,s}$

$$QUAL_{P,s}(t) := \begin{cases} 'perfect' & , t[a] \in C_1(s) \\ 'very good' & , t[a] \in C_2(s) \\ 'good' & , t[a] \in C_3(s) \\ 'acceptable' & , t[a] \in C_4(s) \\ 'sufficient' & , t[a] \in C_5(s) \end{cases}$$

D

Constructing a human-understandable instance of this quality function can be a delicate task. For example, it is intuitively not comprehensible that a less preferred element of an ordered pair belongs to a higher quality category than the more preferred element. Because of this, all instances of the quality function have to fulfill the following postulate:

**Definition 3.20 Quality postulate**

For a given preference  $P := (A, <_P)$  in a situation  $s$  the  $QUAL_{P,s}$ -function of Definition 3.19 must satisfy that for all elements  $t, t'$  with  $t[A], t'[A] \in \text{dom}(A)$ :

- $t <_P t'$  implies  $QUAL_{P,s}(t) \leq QUAL_{P,s}(t')$  D

Instantiated quality functions for all introduced base preferences and complex preferences can be found in [Fis04]. Therefore, tuples of the BMO-set can be valued with respect to the customer's preferences providing human-comprehensible presentation arguments. Based on this, a so-called presentation preference determines which results are predestined to be pointed out to the customer. This framework, which is called the *Preference Presenter*, enables a search engine to proactively present search results by respecting an underlying strategy, e.g., a special sales strategy ([KFD04, FKP06]).

**Example 3.9 Personalized presentation of query results**

Let us assume there is no perfect match for the search query of Example 3.7. By using the quality information delivered by the Preference Presenter, the following presentation of an alternative hotel could occur:

"Overall, the TOKYO hotel fits your preferences *very good*. It *perfectly* hits your preference for breakfast. Furthermore, matching your wishes, the HILTON hotel is avoided."

◇

Based on preferences modeled as strict partial orders, the Preference Framework enables the implementation of deeply personalized and situated e-commerce applications. Nevertheless, a domain-specific search tailored to travel and tourism is necessary in order to provide customers with personalized and situated travel packages. In the following chapter, a novel preference based approach for the adaptation of the search process in tourism will be presented.

## 4 Tailoring a Personalized Search for Tourism

Due to the emerging concept of *Travel 2.0*, a phrase coined by *PhoCusWright Inc*<sup>6</sup>, customers are no longer content just to find the lowest price, they are looking to take control and **identify the perfect trip** ([Gro07]). The concept of *dynamic packaging* was introduced to provide customers with a flexible opportunity to arrange their journeys. While it aims to provide travel packages exactly tailored to the customer, it suffers even more from insufficient search technologies. This could be one reason why only 23% of the airlines provide a combination with hotel and car reservations on their website [LL00].

In the last chapter, the advantages of preference based search technologies were outlined: a *Preference Search* avoiding the empty-result-effect, a *Preference Presenter* implementing a presentation of search results based on sales psychology, and a *Preference Repository* providing the management of long-term preferences. These components provide a good starting point for the personalization of search processes. Yet good search technologies have to be adjusted to the corresponding domain ([SBA01]). What pieces or stages are necessary for a deeply personalized search in tourism or in e-commerce in general? Of what does it consist? How about the integration of the situational context? These questions will be addressed in the following sections.

### 4.1 Personalization of the Search Process in E-Commerce

Taking the preferences of customers into consideration is a promising approach for personalization ([Cho03, KI05]). But as stated above, there might be different kinds of preferences in electronic commerce, which have to be treated carefully in order to deliver custom-tailored products. In the following, therefore, some typical preferences of electronic commerce will be introduced.

#### 4.1.1 Common Preferences in Electronic Commerce

Ideally, customers should be able to adequately specify all of their preferences in the search form. But the reality is different. Sometimes customers are not able to state all preferences at the beginning of the search process ([PF00]). They become aware of some preferences only

when solutions are proposed that violate them. For example, a customer is looking for a flight with Lufthansa, but after being offered a corresponding flight with 4 intermediate stops, he or she might recognize an additional preference for a non-stop connection. Such preferences will be defined as *hidden preferences*.

#### Definition 4.1 Hidden preferences in e-commerce

Given a schema  $R(A_1: \text{dom}(A_1), \dots, A_m: \text{dom}(A_m))$  and  $A \subseteq \{A_1, \dots, A_m\}$ . Assuming  $R^i$  is an instance of  $R$ , which represents a result relation for a customer's search request, one defines:

- Preference  $P = (A, <_P, \cong_P)$  is a **hidden preference**, if  $\exists t \in R^i: t[A] \notin \max(P)$ , so that the customer becomes aware of preference  $P$ . D

In addition, sometimes the possibilities of a search form do not concern the customer's true objective ([FPZ05]). For instance, consider a customer who simply prefers a cheap and fast transport to the airport via railway. Since most travel portals support the specification of a departure airport only, the customer is forced to choose an adequate airport corresponding to his or her own knowledge. However, sometimes customers are not aware of all the airports matching their preferences. In general, customers are often forced to formulate means objectives on insufficient search forms instead of their underlying preferences. Therefore, the term *form-driven preferences* is defined in an informal manner as follows.

#### Definition 4.2 Form-driven preferences in e-commerce

A preference  $P$  that can be specified on a travel portal's search form is defined as **form-driven** preference if it does not represent a customer's actual preference and its specification and, therefore, implicitly requires a matchmaking by the customer. D

This means that, instead of providing the specification of their actual preferences, search forms sometimes force customers to formulate provisional preferences (i.e., form-driven preferences) in order to meet their goals.

#### Example 4.1 Sample form-driven preference

Mark is looking for a flight to San Diego. He would like to be at his destination at 20:00, but the search form only allows him to specify his departure time. That is, instead of specifying his desired arrival time - his actual preference - he is forced to formulate a corresponding form-driven preference for the departure time.

Mark might believe that the flight necessarily involves a change of plane and takes about 12 hours. Based on this, he specifies the corresponding departure time 08:00. Yet it is possible that there is a non-stop flight starting at 11:00. Mark might not get this solution offered, only because he was forced to input the departure time instead of his preferred time for the arrival at his destination. ◇



In tourism, a travel and vacation package typically combines hotels with flights and/or rental cars. Of course, a customer might have individual preferences for parts of his or her travel, e.g., for a certain flight.

**Definition 4.3 Individual preferences in tourism-related electronic commerce**

Let  $T := \{T_1, \dots, T_z\}$  be a travel package consisting of  $z$  parts. Given a schema  $R_i(A_{1_i}: \text{dom}(A_{1_i}), \dots, A_{m_i}: \text{dom}(A_{m_i}))$  for each part of the travel  $T_i$ :

- Preference  $P = (A, <_p, \cong_p)$  is an **individual preference** for a travel part  $T_i$  if  $A \subseteq \{A_{1_i}, \dots, A_{m_i}\}$ . D

Moreover, he or she might have global preferences for the complete travel package. In electronic commerce, global preferences are mostly used on summed up attribute values, for instance, on the total price that is the sum of individual prices for parts of the travel. Thus, for the scope of this work the term global preference is defined as follows.

**Definition 4.4 Global preferences in tourism-related electronic commerce**

Assume a travel package  $T := \{T_1, \dots, T_z\}$  consisting of  $z$  parts and a database relation  $R_i$  for each part  $T_i$  and a numerical attribute  $A_{a_i}$  contained in each relation  $R_i$ . A customer's wish is defined as **global preference**  $P$  for travel package  $T$ , if  $P$  is defined on the aggregation of  $A_{a_i}$  attribute values of a combination of tuples  $(t_1, \dots, t_z)$ . D

**Example 4.2 Sample individual and global preferences**

Let us consider a travel package  $T$  which consists of a flight and a hotel. There is a schema for each part of the travel, e.g., FLIGHT(Airline, Class, Price) and HOTEL(Name, Category, Price\_per\_Night). Given a preference  $P = (\text{Class}, <_p, \cong_p) := \text{POS}(\text{Class}, \text{'Business'})$ ,  $P$  is an individual preference for the travel part flight, since its only attribute 'Class' is contained in the schema FLIGHT.

In addition, given a preference  $P = (\text{Total\_Price}, <_p, \cong_p) := \text{LOWEST}_{50}(\text{Total\_Price})$  for the complete travel package.  $P$  can be called a global preference, since the attribute 'Total\_Price' represents the summed up prices for combinations of flight and hotel. D

While hidden and form-driven preferences may denote problematic cases within electronic commerce which have to be dealt with, individual and global preferences can be used to take the character of the dynamic packaging concept into account. Therefore, typical preferences of e-commerce as described above should be taken into consideration for advanced search processes.

### 4.1.2 Design Principles for a Personalized Search Process

A deeply personalized search process should lead to custom-tailored products with respect to the individual customer's preferences and situation. In particular, the problems and challenges described above should be dealt with. In order to construct a deeply personalized search process, the following design principles are defined.

**1. Sophisticated and semantically rich preference model:**

In order to deliver custom-tailored products, it is absolutely necessary to take the preferences of customers into consideration. Thus, a sophisticated and semantically rich preference model is essential for a personalized search process.

**2. Smart preference elicitation:**

By using a novel kind of preference elicitation based on the integration of heterogeneous data sources, relevant preferences of customers can be gained. Customers should be able to specify their underlying preferences. This means that form-driven preferences have to be avoided. Furthermore, customers should be able to include new discovered preferences, i.e., hidden preferences, in an intuitive and comfortable way.

**3. Intuitive interfaces:**

The interface to customers should be designed carefully and in a personalized way in order to enable an intuitively comprehensible handling of the system.

**4. Search model and processing:**

After gaining and modeling customers' preferences, it is still necessary to perform a match-making between them (wishes) and the database content (reality). A query model is necessary which avoids the frustrating empty-result-effect and can deliver best alternatives if there is no perfect match. Moreover, individual and global preferences should be processed accordingly.

**5. Personalized presentation:**

The quality of search results represents an important factor in sales dialogs. But any discussion about the quality has to take into account the customer's preferences; even if two different persons get the same search result, one of them might prefer a better quality, while the other might prefer a lower price. Moreover, domain dependent quality terms should be regarded, e.g., stars for labeling the quality of hotels.

**6. Situation awareness:**

Preferences of customers strongly depend on their situation. For example, a family father may be less flexible in terms of time, since the family trip has to take place during the school holidays. Because of this, the situation of customers should be taken into consideration for the search process.

In the following subsection, a novel search model for electronic commerce is presented.

### 4.1.3 Search Model

There are several models for the search process in electronic commerce (e.g., [MHD00, ÖFA01, SBA01], see related work below). However, none of these is sufficient in terms of the stated design principles. Therefore, a novel four stage model is proposed in this work in order to enable a deep personalization of the entire search process.

#### Definition 4.5 Search process model for electronic commerce

The search in e-commerce is modeled in a cyclic process consisting of four stages:

- Preference Analysis & Modeling
- Search Interface
- Query Processing
- Presentation

A situation model represents the pivotal element, influencing each stage of the process. The model is illustrated in Figure 4.1. D

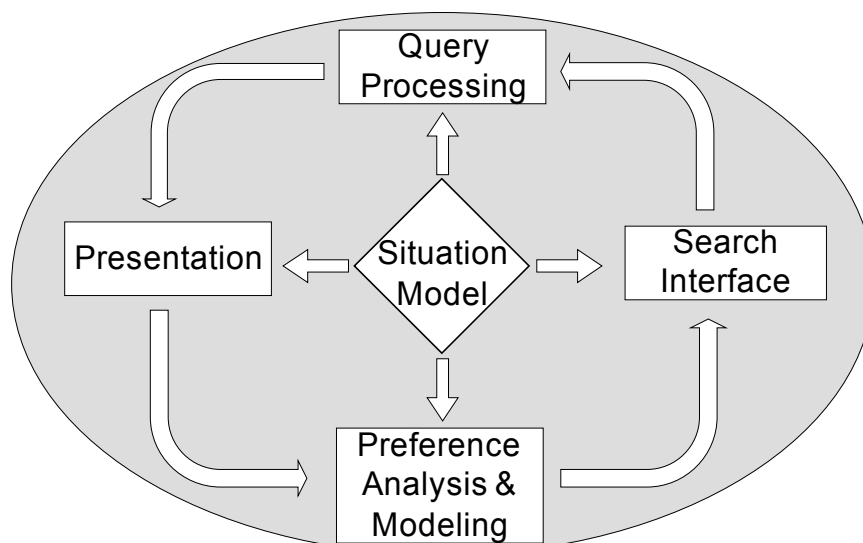


Figure 4.1. Personalized and situated search model for e-commerce

All pieces of the model are described in the following:

#### Preference Analysis & Modeling

During the first stage, relevant constraints have to be identified with respect to domain, customer, and situation. Thereafter, the constraints have to be modeled explicitly. This seemingly easy task includes rather complex activities. For example, even if a customer has a clear

price constraint related to his or her situation as a business traveler (as described above), it has to be modeled whether this represents a hard constraint or a preference, i.e., a soft constraint.

### **Search Interface**

In common systems, preference modeling is reduced to implicit hard constraints which are gained from the customer's specifications on the search interface. However, in order to provide a deep personalization, it is necessary to differentiate between preference modeling and search interface. Consider the following scenario: Traveler Mark registers to his preferred online travel portal in order to arrange a trip. The system knows by now that Mark is a business traveler with a strong price preference. Note that preferences can be mined automatically from log data as described in [HEK03]. Yet on the search form it is necessary to give Mark the opportunity to express his preference accordingly. There might be a single field for an upper price limit, or two fields for a preferred price interval, or a slider for enthusiastic mouse users, who are too lazy to fill in a number using the keyboard. Besides, Mark could be color-blind, requiring the avoidance of some combinations of colors on the form. Thus, the search interface has to be considered as a single and important piece of the search process.

### **Query Processing**

After the preferences of Mark have been identified and filled into a search form as described above, the search has to take place. Search queries have to be composed and executed accordingly. This process also has to be tailored to the individual being and situation. All preferences have to be mapped to the query. Since Mark has a strong price preference, this could be represented by a hard constraint in the query. But it could also be represented by a soft constraint which is prioritized to other wishes. This would prevent an empty result set if the price constraints cannot be fulfilled for any possible travel setup. Furthermore, an advanced query processing should deal with individual as well as global preferences.

### **Presentation**

A crucial element in customer electronic commerce is a search process that not only finds the product that best matches the customer's needs, but also convinces him that he has made the best choice ([PK04]). To argue about the results' quality is a promising approach in sales dialogs. But any discussion about quality has to take the customer's preferences into account. These preferences, therefore, have to be considered during the product presentation stage. Besides, the situation might be taken into account too; business travelers may have a precise list of results, while a young student may prefer a colorful presentation. The visualization of the quality of search results may even be adjusted to the tourism domain using stars well known from the hotel sector.

Sometimes customers are not able to state all preferences at the beginning of the search process. A smart presentation delivering arguments about the search results' quality may help customers to discover hidden preferences ([THS+03]). These have to be elicited from the cus-

tomers in order to deliver perfect custom-tailored travel products. Therefore, the personalized search model is represented by a circular process. Note that the circle does not mean that the search has to start all over again. On the contrary, new preferences are seamlessly included in order to refine the search result with respect to the customer's wishes.

The described search process represents a fine-granular model of buying behavior in electronic commerce. Preferences and situations are used in each stage of the process to support the customer as an individual being in an individual situation. However, a search adjusted to the corresponding domain is one of the central foundations for good e-commerce shopping sites on the internet ([SBA01]). In the following, therefore, the search process model will be applied to the travel and tourism industry. At first, a situation model for tourism will be outlined.

## 4.2 Situation Modeling for the Search Process in Tourism

More than once the importance of the *Situation Model* as a pivotal element of deeply personalized search processes has been emphasized. Belk ([Bel75]) has already identified the relevance of the *situational context* to buying behavior in 1975. A theoretical foundation as well as a framework for the integration of situational knowledge to the search process will be presented in the following.

Up to now, the term 'situation' has been used rather intuitively, e.g., for describing roles like *business traveler* or *family father*. Holland and Kießling ([HK04]) describe a meta model of situation-oriented entities and relationships. It comprises spatial-temporal aspects and influences. Due to the 'meta' character of the model, tourism-related aspects were not covered. Contrarily, in a different study Ricci ([Ric02a]) divides 'influencing factors' for tourism into two groups: the first group contains socioeconomic factors such as age, income etc., while the second group comprises travel features such as the travel-party's size. In another study, six 'decision styles' of customers are actually differentiated ([GZ02]). Moreover, there are studies about influences of social psychological processes ([ACB96]) and the nature of travel ([HGF02]). It is obvious that such factors may belong to situations influencing the behavior of travelers. Thus, a suitable situation model has to be adjusted to travel and tourism.

A situation model for a tourism-related search process is described by:

### 1. Temporal-spatial aspects:

Temporal-spatial aspects have to be considered, since the customers' preferences may change over time or may depend on the location. For example, a customer may have completely different preferences for a trip to the Alps in winter than for a trip to Hawaii in the summer time.

**2. Personal influences:**

This includes, for instance, the current role or the decision style of customers. As shown above, aspects such as the age, income or other social psychological factors can decisively influence a customer. Therefore, this kind of influence has to be regarded.

**3. Surrounding influences:**

Human beings usually interact with their environment. Because of this, it is important to know what surrounding influences are relevant for customers. Such influences can be people the customer is talking to or current weather conditions, e.g., a customer who arranges a trip during a cold winter night might prefer a destination with a hot and dry climate.

**4. Travel influences:**

Obviously, aspects about the travel itself are very important for a traveler. For example, the nature of the travel as well as the composition of the travel group can influence a customer's preferences very much.

**5. Search process stage:**

Since the situation model is meant to enable a deeply personalized search process in tourism, the stages of the process itself have to be considered as a situational aspect. A customer might, for example, like a plain *search interface*, while preferring a colorful *presentation* of the search results.

Obviously, travel influences and search process aspects can be modeled as special tourism-related aspects of surrounding influences.

**4.2.1 Situated Entity-Relationship-Model for Tourism**

In Figure 4.2 a novel model of situation-oriented entities and relationships for the tourism and travel domain is presented. Temporal-spatial aspects are included with entities for date/time and location, respectively. The time can be described by a timestamp denoting the validity period of a situation, e.g., if a person acts as a business traveler from Monday to Friday only. Entities for the location can be represented by a zip-code, city name or even by global positioning coordinates (GPS). For instance, a customer might want to include a special insurance if he or she rents a car in countries without regulated third party liability insurance.

The most important entities are for the different kinds of influences. Personal influences comprise all human factors. This includes roles, decision styles, or the physical state of the customer, for instance, color distinction deficiencies. Outer factors such as the weather or the size of the customer's screen are denoted by surrounding influences. Aspects of the trip itself are labeled as travel influences that are represented by a sub-entity of surrounding influences.

Common attributes are the travel party's size or the nature of the travel, e.g., private or business. Search stages are also included as sub-entity of surrounding influences.

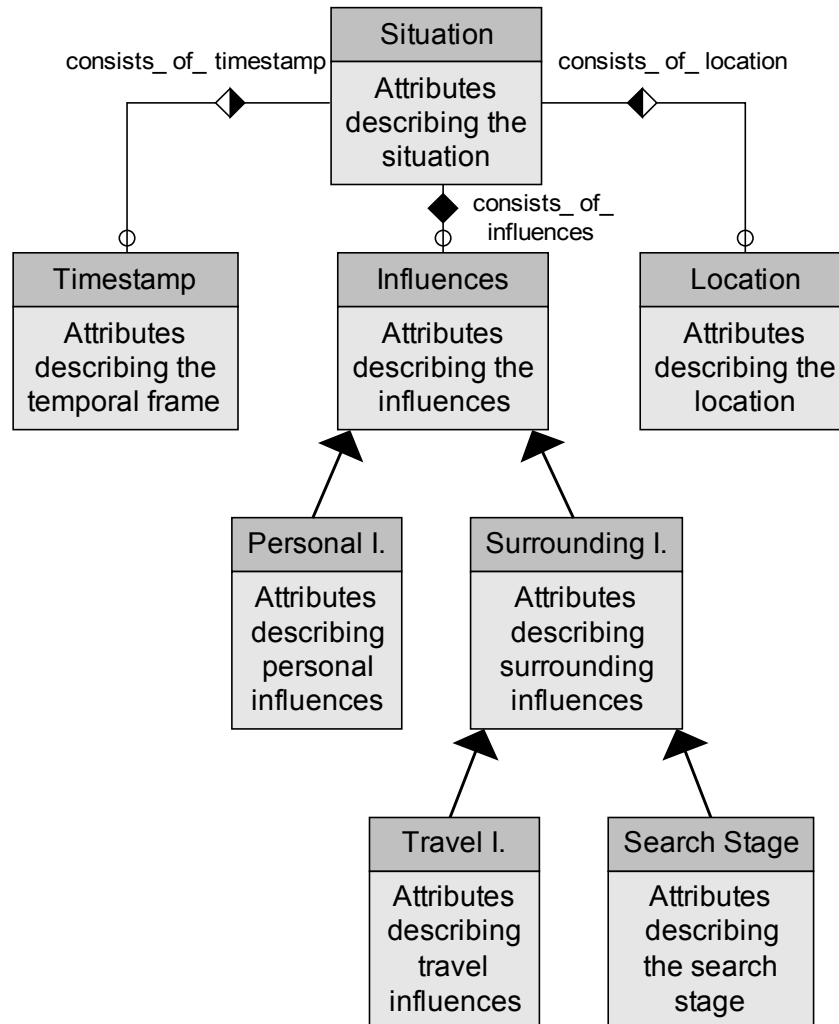


Figure 4.2. Model for situations in tourism

Thereby, entities describing the customer's situation can be seamlessly integrated into existing ER-models of tourism applications.

#### Example 4.3 ER-model for sample tourism portal

To illustrate the point, imagine the construction of a concrete situation-aware travel portal. As stated before, the preferences and wishes of a customer can depend on personal influences such as the decision style described in [GZ02], color distinction deficiencies, and the role of the customer. A personalized search form or a personalized presentation of search results, for instance, should take color deficiencies of customers into account. In addition, the composition of the travel group can also influence the customer's preferences. When traveling together with children, for example, a hotel denoted as 'child-friendly' might be preferred. These sit-

uational aspects can easily be defined using entity-relationship modeling techniques. Figure 4.3 represents a corresponding instance of the presented situation model for tourism. ◇

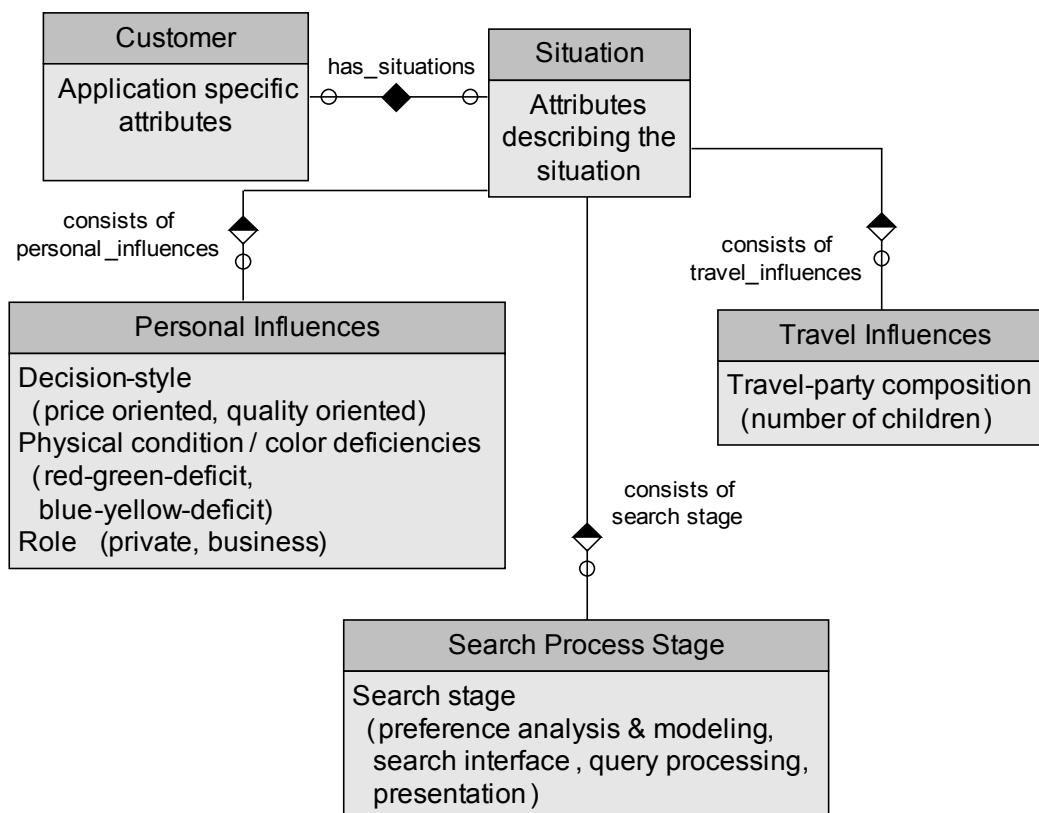


Figure 4.3. ER-model for tourism portal

Now tourism-related situations can be modeled and considered by means of entity-relationship modeling. But how can these situations, which are influencing the customer's wishes, be stored or integrated into the actual search process?

#### 4.2.2 Tourism-Related Preference Repository

The *Preference Repository* ([KFD04, Hol04]) described above allows the management of situation-aware long term preferences. It provides a well-founded framework for the storage of travelers' situated preferences ([DP08]).

##### Example 4.4 Preference Repository for sample scenario

Again consider traveler Mark. When traveling together with his wife and the two children, he prefers a hotel declared as 'child-friendly'. This situated preference is naturally being used during the query processing stage of the search process. Such a situational preference can be stored in the repository by a corresponding XML-structure:



```
<PreferenceData name="Group-Composition">
  <Situation>
    <Condition key="Children_in_group" value="yes"/>
    <Condition key="Search_Stage" value="query-processing"/>
  </Situation>
  <Preference>
    <POS att="child-friendly">
      <Value val="yes"/>
    </POS>
  </Preference>
</PreferenceData>
...
```

If Mark specifies on the search form that he is going to travel together with his children, a corresponding search preference for 'child-friendly' hotels can be added automatically to the search query. Thereby, the *Query Processing* stage of the search process can be influenced in terms of his preferences.  $\diamond$

In conclusion, the presented situation model incorporates specific aspects and influences of travel and tourism. Related situations can be modeled and integrated into existing ER-models by means of entity-relationship modeling. Furthermore, situated preferences can be stored and integrated into the *Preference Repository*. Since *Preference Repository* as well as *Preference Search* are based on the preference model of Kießling ([Kie02]), situated preferences can be integrated seamlessly into each stage of the advanced search model.

### 4.3 Tradeoff Preference Constructor

In electronic commerce in general, but in particular in the tourism domain, there are a number of attributes related to quality. For example, there is:

- The category of hotels,
- the class of flight and railway connections,
- and the category of rental cars.

In everyday life, often implicit decisions are made which represent a compromise, i.e., a tradeoff. Most humans would rather stay overnight in a 5-star hotel or would like to fly first class. However, everyone has to decide for himself, how much higher quality is worth to him. If one is looking for a 3-star hotel, it generally means that the quality of three stars represents a kind of personal lower limit. Four stars or more would be taken gladly, but probably do not match the personal price expectations according to experience.

Quality usually correlates with the price, i.e., the price increases with rising quality. Yet this is not always the case. For example, the price of a hotel also depends on its location and reputation. A qualitatively better hotel labeled with more stars could also match the price preference of a customer, e.g., if it was just opened/reopened or if it is not located in the center of the city. This means that sometimes qualitatively better hotels could additionally be offered to a customer, as long as they do not exceed his or her price limit. These exceptions are going to be addressed by the new preference constructor. Since it is meant to deal with the tradeoff between quality and price, it is called the TRADEOFF constructor.

First of all, the semantics of the TRADEOFF preference constructor will be outlined by means of reasonable scenarios and examples. For this purpose, consider a customer looking for a hotel which is labeled with three stars. Note that in order to be predictable and comprehensible, all hotels matching his or her quality preference have to be offered to him or her, i.e., all 3-star hotels. A filter effect inserted by us could inadmissibly cut the result. For instance, the customer could be looking for a 3-star hotel he or she already knows and likes. If he or she does not care about the price, it is not admissible to cut this hotel, even if there are cheaper hotels labeled with more stars. Yet qualitatively better results should additionally be presented to the customer if they fall into the price range of the desired 3-star hotel, i.e., if they are cheaper than the most expensive hotel with three stars. In conclusion, all hotels labeled with 3 stars as well as all hotels with higher quality which fall into the price range of 3-star hotels should be offered to the customer (see Figure 4.4). A detailed description of the figure will follow below.

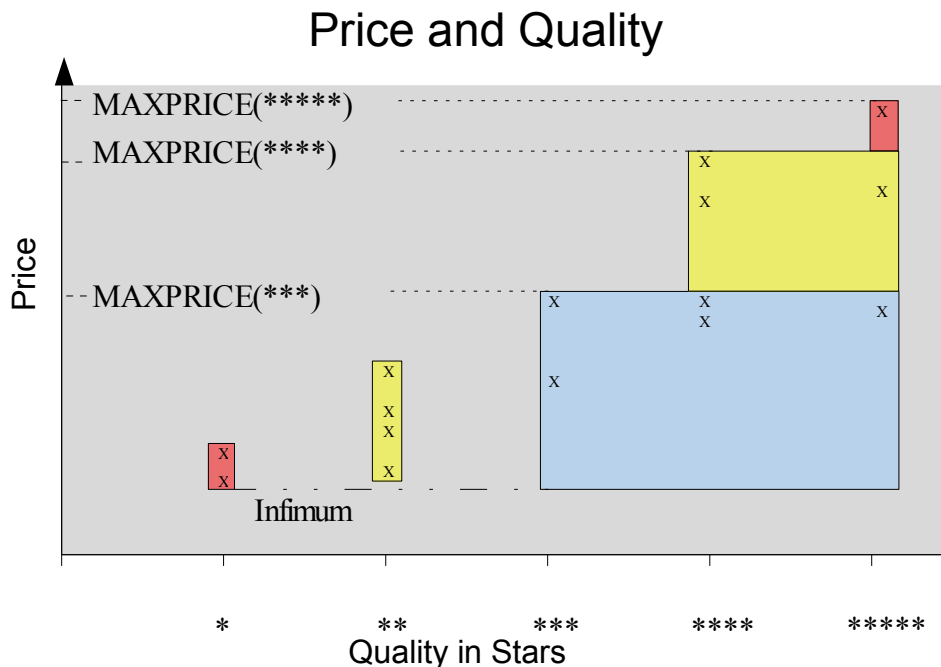


Figure 4.4. Price and quality tradeoff

If the problem is examined in a more general way, then it must be differentiated whether an object provides a quality below (A) or above (B) the desired level:

- A) Regarding objects with a lower quality than desired by the customer, an object is better the closer it is to the desired quality. For example, a hotel labeled with two stars will be considered as better than a hotel with only one star by a customer who is looking for a 3-star hotel. The price is insignificant, since the customer was already willing to pay for a qualitatively better and probably more expensive hotel. Considering Figure 4.4, there is one hotel with a quality of two stars which is more expensive than the cheapest 3-star hotel. Therefore, it may appear worse than cheaper hotels with 2 stars at the first sight. Why take a 2-star hotel if there is a cheaper one with 3 stars? However, it must not be forgotten that there might be significant reasons for this, for example, because the 2-star hotel offers an unmatched location in the heart of the city. The choice between different 2-star hotels has to be made by a further preference or explicitly by the customer.
- B) For objects that match the customer's wishes on quality or even surpass it, the following approach is applied: the usually very limited number of quality levels (denoted by an attribute  $Q$ ) is mapped to price ranges, which will be called price levels in the following. Price level 0 is limited by the infimum, the lowest price at all, and the highest price (MAXPRICE) for an object of the desired quality. Considering a customer who is looking for a 3-star hotel, price level 0 is illustrated by a blue rectangle in Figure 4.4. It includes 5 hotels of equal or better quality: two 3-star hotels, two 4-star hotels, and even one 5-star hotel. Thereafter, price level 1 immediately starts. It is limited by the highest price of the next quality level, e.g., by the maximum price of 4-star hotels called MAXPRICE(\*\*\*\*) in Figure 4.4. It includes three hotels labeled with 4 or 5 stars. An object is regarded as better the lower the corresponding price level is.

Please examine Figure 4.4 again. The customer expressed a wish for a 3-star hotel. Perfect hits are framed by the blue rectangle. These are all 3-star hotels, as well as 4- and 5-star hotels, which are included in price level 0, i.e., these hotels are cheaper than the most expensive hotel labeled with three stars. Best alternatives are framed by yellow rectangles and include hotels with two stars as well as hotels of price level 1. Finally, two 1-star hotels and one 5-star hotel of price level 2 represent the worst alternatives. They are framed by red rectangles.

In the following, this approach is demonstrated by two scenarios.

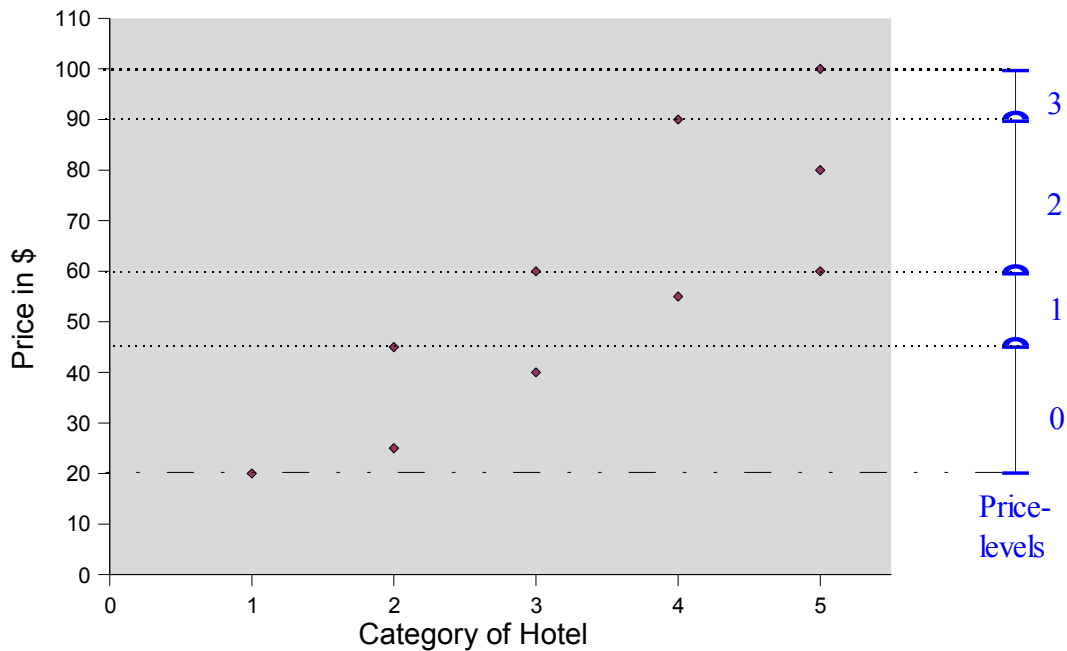
### **Demonstration Scenario 1:**

The quality of a hotel is expressed by the category. In general, a higher category corresponds to a higher quality. A sample database for hotels is represented by Table 4.1. Tuples of the table are illustrated in Figure 4.5. The highest price of each category, i.e., the MAXPRICE value, is marked by a dotted, horizontal line.

| ID               | Category | Price (in \$) |
|------------------|----------|---------------|
| t <sub>H1</sub>  | 1        | 20            |
| t <sub>H2</sub>  | 2        | 25            |
| t <sub>H3</sub>  | 2        | 45            |
| t <sub>H4</sub>  | 3        | 40            |
| t <sub>H5</sub>  | 3        | 60            |
| t <sub>H6</sub>  | 4        | 55            |
| t <sub>H7</sub>  | 4        | 90            |
| t <sub>H8</sub>  | 5        | 60            |
| t <sub>H9</sub>  | 5        | 80            |
| t <sub>H10</sub> | 5        | 100           |

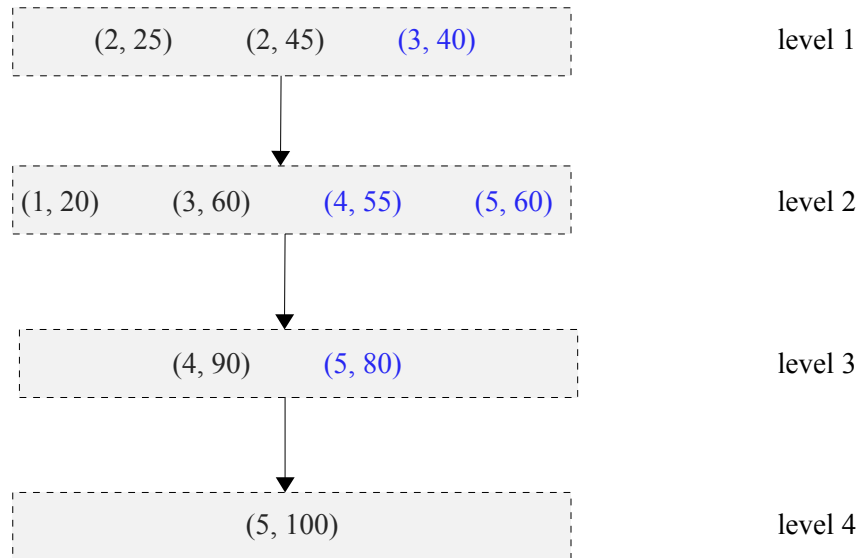
**Table 4.1. Hotel database**

A customer is looking for a hotel in **category 2**, i.e., which has two stars. The resulting price levels are shown in Figure 4.5 and are marked in blue color. Price level 0 starts with the infimum (\$20) and ends with MAXPRICE for the desired category of 2 (\$45). Thus it contains the prices of interval [20, 45]. The next price level immediately starts after MAXPRICE of category 2 and ends with MAXPRICE for category 3. Therefore, the interval [45, 60] is contained. This continues analogously.



**Figure 4.5. Price levels for customers looking for a hotel with 2 stars**

Considering the described approach when dealing with tradeoffs, the following BTG is formed:

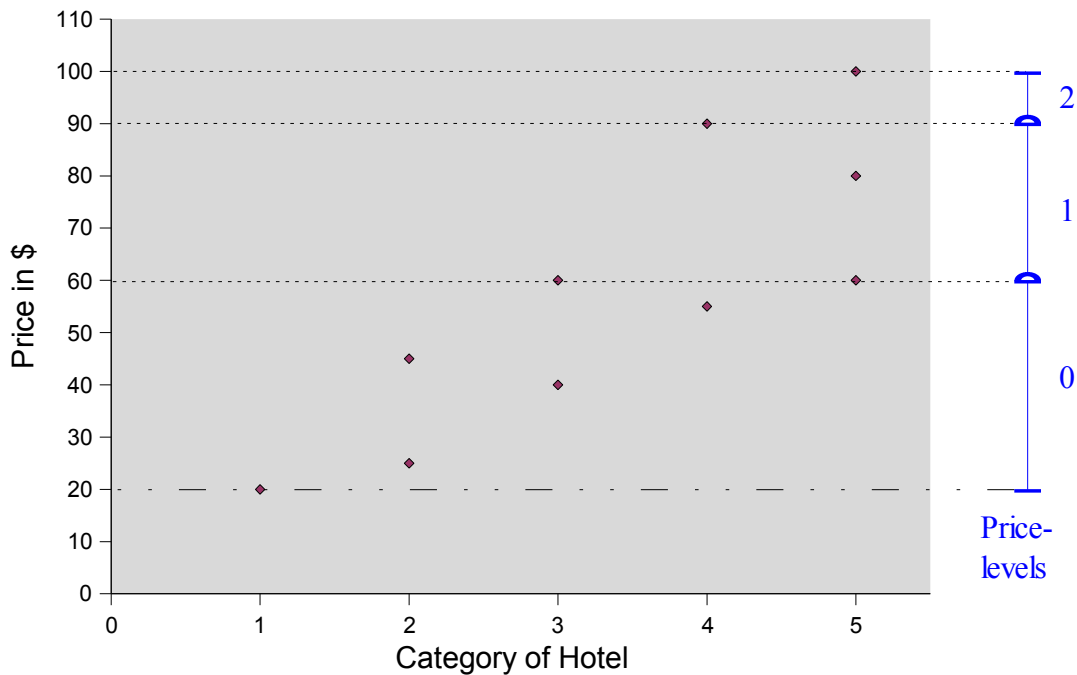


Note that the BTG was reduced for a better illustration: All elements of one level are indifferent to each other, while being better than any element of the successive level.

To a certain degree, the graph could represent an  $\text{AROUND}(\text{category}, 2)$  preference. But in contrast to this, tuples with a higher quality than specified by the customer, i.e., higher than category 2 in this example, can ascend in the BTG. Such tuples are marked in blue color. For example, since it is in the same price level 1 ( $[45, 60]$ ), tuple (5, 60) is now on the same level of the BTG as tuple (3, 60).

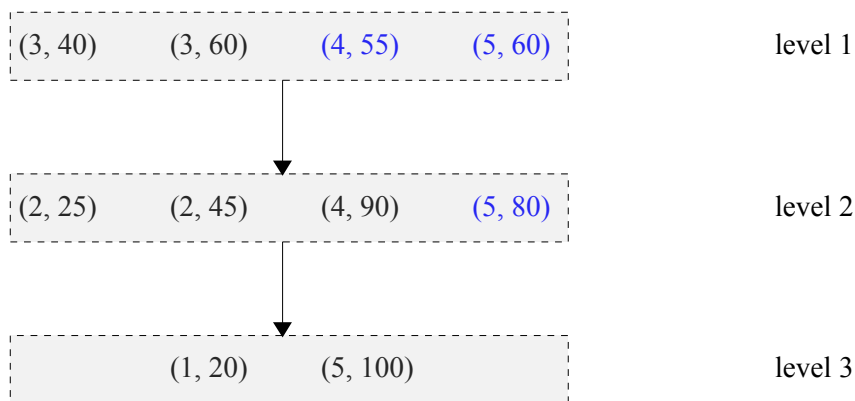
### **Demonstration Scenario 2:**

In this scenario, the customer is looking for a hotel of category 3. The resulting price levels are illustrated in Figure 4.6. One has to bear in mind that price level 0 is limited by  $\text{MAX-PRICE}$  of the desired quality, which is category 3 in this scenario. Therefore, price level 0 contains the interval  $[20, 60]$ . Price level 1 ( $[60, 90]$ ) and price level 2 ( $[90, 100]$ ) are determined analogously.



**Figure 4.6. Price levels for customers looking for a hotel with 3 stars**

This would lead to the following reduced BTG:



Note that only objects that are qualitatively better than desired can ascend in the BTG. Thus, the number of such objects decreases as the desired quality increases. For instance, if a customer prefers the highest quality, i.e., a hotel of category 5, then there are no qualitatively better tuples which can improve their position in the BTG. This behavior would exactly comply with an  $\text{AROUND}(\text{category}, 5)$  preference.

In conclusion, it is reasonable to provide additional offers to customers looking for a certain quality of a product, e.g., for a flight in the economy class. Such additional offers should provide an even better quality while meeting the desired price range. Therefore, the following approach can support customers with respect to the typical tradeoff between quality and price. The modeling of an appropriate preference constructor is shown below.

### 4.3.1 Definition of the Preference Constructor

There exist different possibilities denoting a quality property in tourism:

- **numerical valuations**, e.g., for the category of hotels usually described by **1** to **5** stars or like **1.** and **2.** class describing the category of railway tickets
- **categorical valuations**, for example for the categorization of rental cars into **economy**, **compact**, **standard**, **premium**, and **luxury** cars

In order to take the quality into account, it is necessary to provide a flexible preference constructor dealing with a variety of possible quality domains. In particular, it is essential to incorporate semantic knowledge describing the quality domain. In the following, the quality domain is integrated by means of an ordered quality set.

#### Definition 4.6 Domain dependent quality representation: QUAL\_SET

Let  $A_Q$  be an attribute describing a quality property with a corresponding domain  $\text{dom}(A_Q)$ .

- **QUAL\_SET** =  $\{q_0, \dots, q_m\}$ ,  $m \geq 0$ , with  $q_i \in \text{dom}(A_Q)$  represents a totally ordered, finite set of values.

The first element  $q_0$  of **QUAL\_SET** denotes the bottom quality of a domain, while  $q_m$  represents the best quality. D

#### Example 4.5 Sample quality representation of rental cars

The following set represents domain knowledge about the quality of rental cars:  $\text{QUAL\_SET}_{\text{cars}} = \{\text{'economy'}, \text{'compact'}, \text{'standard'}, \text{'premium'}, \text{'luxury'}\}$

There exist five different valuations for the quality of rental cars. A valuation of 'economy' denotes the bottom quality, while 'luxury' describes the best available quality for a rental car. ◇

In the proposed approach, customers should be provided with additional offers - which have a better quality than specified by the customer - as long as such offers belong to the price level desired by the customer. Therefore, firstly, it is necessary to determine the highest price for each quality level, e.g., the highest price for 'premium' rental cars.

#### Definition 4.7 MAXPRICE- utility function

Let  $A = \{A_Q, A_P\}$  be a set of attributes where  $A_Q$  represents a quality property and  $A_P$  represents a price with corresponding domains  $\text{dom}(A_Q)$  and  $\text{dom}(A_P)$ , respectively. The domain of  $A$  is defined as  $\text{dom}(A) := \text{dom}(A_Q) \times \text{dom}(A_P)$ . Given the utility function **MAXPRICE**:  $\text{dom}(A_Q) \rightarrow \text{dom}(A_P)$  one defines for all  $q_i \in \text{dom}(A_Q)$ :

- $\text{MAXPRICE}(q_i) := \{p \in \text{dom}(A_P) \mid \neg \exists (q_i, p^*) \in \text{dom}(A): p^* > p\}$   $\text{D}$

#### Example 4.6 Demonstration of the MAXPRICE-function

Consider the following small relation:  $\text{Cars}(\text{Category}, \text{Price}) = \{('economy', 40), ('compact', 50), ('compact', 80), ('standard', 70), ('standard', 80), ('standard', 95)\}$

The following prices can be determined by usage of MAXPRICE that in each case denotes the highest price for the corresponding quality level:

$\text{MAXPRICE}('economy') = 40;$   
 $\text{MAXPRICE}('compact') = 80;$   
 $\text{MAXPRICE}('standard') = 95;$   $\text{D}$

Now, by means of the MAXPRICE-function, all price levels can be calculated.

#### Definition 4.8 PRICELEVEL-utility function

Let  $\text{QUAL\_SET}$  be a quality representation of  $\text{dom}(A_Q)$ . Assume  $q_z \in \text{QUAL\_SET}$  representing the quality desired by a customer and  $I = (I_0, \dots, I_n)$ ,  $n \geq 0$  an ordered set of disjoint intervals in  $\text{dom}(A_P)$  with the following properties:

- $I_0 = [\text{INFIMUM}_{A_P}, \text{MAXPRICE}(q_z)]$
- for  $i > 0$ :  $I_i = ](\max(\text{MAXPRICE}(q_z), \dots, \text{MAXPRICE}(q_{(z+i-1)})), \text{MAXPRICE}(q_{z+i})]$

The well-known function  $\max(\text{argument}_1, \dots, \text{argument}_f)$  returns the maximal argument. This is necessary to cover cases, where  $\text{MAXPRICE}(q_{z+i})$  is less than  $\text{MAXPRICE}(q_{(z+i-1)})$ , i.e., the maximal price of products with a worse quality is higher.

The function  $\text{PRICELEVEL}: \text{dom}(A_P) \rightarrow \mathbb{N}_0$  is now defined as follows:

- for  $i \in \{0, \dots, n\}$ , for all  $v \in I_i$  it holds that  $\text{PRICELEVEL}(v) = i$ ;  $\text{D}$

#### Example 4.7 Determining price intervals

Mark would like to have a 'compact' rental car. A quality representation for the corresponding domain is given by  $\text{QUAL\_SET}_{\text{cars}}$  of Example 4.5. Using relation  $\text{Cars}$  and the determined MAXPRICE-values of Example 4.6, the following intervals can easily be calculated:

$I_0 = [40, 80]$   
 $I_1 = ]80, 95]$

Interval  $I_0$  includes all prices from the infimum to the maximal price of the desired 'compact' car, whereas  $I_1$  comprises higher prices up to the maximal price for 'standard' cars.  $\text{D}$



The process for determining the price levels with corresponding intervals is outlined in Figure 4.7 for illustration.

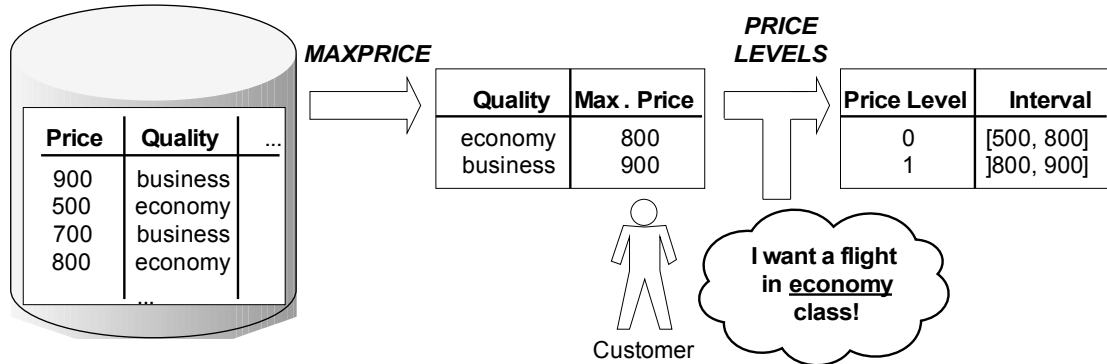


Figure 4.7. Determining maximal prices and price levels

In order to deal with typical tradeoff-scenarios in electronic commerce and in particular in tourism as described above, a preference constructor called TRADEOFF will be presented.

#### Definition 4.9 TRADEOFF preference constructor

Let  $A = \{A_Q, A_P\}$  be a set of attributes where  $A_Q$  represents the quality and  $A_P$  represents the price with corresponding domains  $\text{dom}(A_Q)$ ,  $\text{dom}(A_P)$ , and  $\text{dom}(A) := \text{dom}(A_Q) \times \text{dom}(A_P)$ . Let  $\text{QUAL\_SET}$  be a quality representation of  $\text{dom}(A_Q)$  with  $q_z \in \text{QUAL\_SET}$  representing the quality desired by a customer. In order to achieve a behavior as described in the example scenarios from the beginning of this subsection, the distance function for the TRADEOFF constructor  $\text{DISTT}[\text{QUAL\_SET}, q_z]: \text{dom}(A) \rightarrow \mathbb{N}_0$  is defined for all  $(q, p) \in \text{dom}(A)$  and  $q_i \in \text{QUAL\_SET}$  as follows:

$$\text{DISTT}[\text{QUAL\_SET}, q_z](q_i, p) := \begin{cases} z - i & \text{if } (i < z) \\ \text{PRICELEVEL}(p) & \text{else} \end{cases} \quad \begin{matrix} \text{(A)} \\ \text{(B)} \end{matrix}$$

If the index of  $q_i = q$  is less than that of the desired quality  $q_z$ , i.e.,  $q$  represents a quality lower than specified by a customer, the simple distance is delivered by  $z - i$  (A). Otherwise the distance will be determined by the PRICELEVEL-function (B).

Now on basis of the distance function DISTT, the TRADEOFF preference constructor is defined as follows:

- base TRADEOFF  $(A, \text{QUAL\_SET}, q_z) \{(q_1, p_1) <_P (q_2, p_2)\}$  iff  $\text{DISTT}[\text{QUAL\_SET}, q_z](q_2, p_2) < \text{DISTT}[\text{QUAL\_SET}, q_z](q_1, p_1)$

The new preference constructor TRADEOFF is a sub-constructor of SCORE<sub>d</sub> specializing d=0. For all  $(q, p) \in \text{dom}(A)$  the utility function of SCORE<sub>d</sub> can be defined as:  
 $f(q, p) := - \text{DISTT}[\text{QUAL\_SET}, q_z](q, p)$        $\gg$

#### Example 4.8 TRADEOFF preference constructor for renting a car

Consider the following database for rental cars:

| ID              | Provider | Category | Price (\$) |
|-----------------|----------|----------|------------|
| t <sub>C1</sub> | eSixt    | standard | 70         |
| t <sub>C2</sub> | eSixt    | luxury   | 90         |
| t <sub>C3</sub> | Europcar | economy  | 50         |
| t <sub>C4</sub> | Europcar | compact  | 60         |
| t <sub>C5</sub> | Hertz    | economy  | 30         |
| t <sub>C6</sub> | Hertz    | compact  | 45         |
| t <sub>C7</sub> | Hertz    | standard | 60         |

Table 4.2. Database for rental cars

Knowledge about the domain of category - that stands for the quality - is represented by

$$\text{QUAL\_SET}_{\text{rental\_car}} = \{\text{'economy'}, \text{'compact'}, \text{standard}, \text{'luxury'}\}$$

Consider customer John, who is looking for a 'compact' car. The following instantiation of the TRADEOFF constructor would occur:

$$\begin{aligned} P_{\text{rental\_car}} &= (\{\text{Category}, \text{Price}\}, <_{\text{rental\_car}}, \cong_{\text{rental\_car}}) \\ &:= \text{TRADEOFF}(\{\text{Category}, \text{Price}\}, \text{QUAL\_SET}_{\text{rental\_car}}, \text{'compact'}) \end{aligned}$$

Utilizing the MAXPRICE-function for the database of rental cars leads to:

$$\begin{aligned} \text{MAXPRICE}(\text{'economy'}) &= 50, \\ \text{MAXPRICE}(\text{'compact'}) &= 60, \\ \text{MAXPRICE}(\text{'standard'}) &= 70, \\ \text{MAXPRICE}(\text{'luxury'}) &= 90 \end{aligned}$$

For John this would result in the following price levels and intervals, respectively:

$$\begin{aligned} I_0 &= [30, 60], \\ I_1 &= [60, 70], \\ I_2 &= [70, 90]. \end{aligned}$$

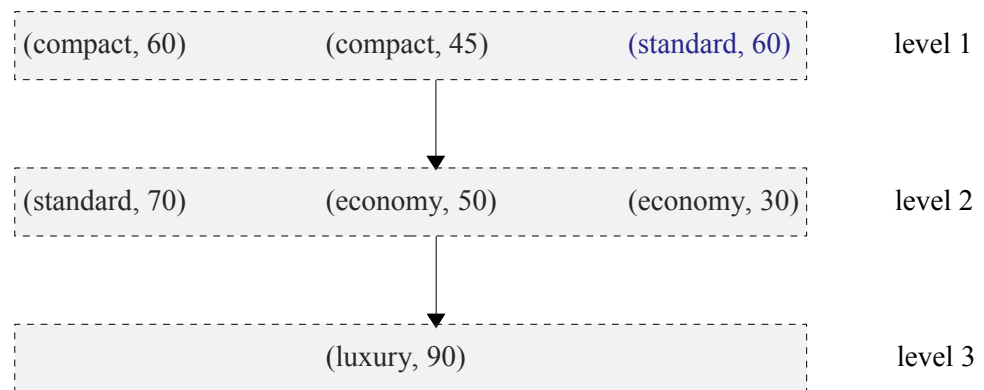
The distances determined by the DISTT-function are shown in Table 4.3.

| ID       | Distance | Provider | Category | Price (\$) |
|----------|----------|----------|----------|------------|
| $t_{c1}$ | 1        | eSixt    | standard | 70         |
| $t_{c2}$ | 2        | eSixt    | luxury   | 90         |
| $t_{c3}$ | 1        | Europcar | economy  | 50         |
| $t_{c4}$ | 0        | Europcar | compact  | 60         |
| $t_{c5}$ | 1        | Hertz    | economy  | 30         |
| $t_{c6}$ | 0        | Hertz    | compact  | 45         |
| $t_{c7}$ | 0        | Hertz    | standard | 60         |

**Table 4.3. Distances determined by DISTT**

Tuples  $t_{c3}$  and  $t_{c5}$  representing 'economy' class cars are qualitatively worse – with respect to the quality set QUAL\_SET - than the desired 'compact' car. For this reason the simple distance of indexes in QUAL\_SET is computed by DISTT (case A of Definition 4.9). Since all other tuples represent cars that are equal or better than John's wish, the PRICELEVEL-function is used to compute the corresponding distances.

This leads to the following reduced BT-graph:



In addition to the desired 'compact' cars, a qualitatively better 'standard' car is on the maximal level of the BTG. It is on the same price level, i.e., cheaper or equally costly than at least one 'compact' car. Note that for a comprehensible handling of indifferent values, the regular SV-Relation  $\cong_{\text{rental\_car}}$  is used. The next example will examine this aspect in more detail.  $\diamond$

#### **Example 4.9 TRADEOFF preference constructor for renting a car**

By using the regular SV-relation, all tuples with the same distance will be treated as substitutable. While this does not influence the base preference itself, it leads to a more comprehensible result within a complex preference. Firstly, the influence will be demonstrated by means of a prioritized preference and afterwards by a pareto complex preference.

## A) Prioritized preference

Again consider  $P_{\text{rental\_car}}$  of Example 4.8. Now, it is part of a complex preference:

$P_{\text{rental\_car\_C1}} := P_{\text{rental\_car}} \& P_{\text{rental\_car}^*}$  with:

$P_{\text{rental\_car}^*} = (\text{Provider}, \langle_{\text{rental\_car}^*}, \cong_{\text{rental\_car}^*}) := \text{POS}(\text{Provider}, \{\text{'Hertz'}\})$

The customer of this example would like to have a 'compact' car too. He or she prefers rental cars from the provider Hertz. However, the quality is more important to him or her than the provider.

Using Table 4.2 and the regular SV-relation for  $P_{\text{rental\_car}}$ , the maximal level of the BT-graph of  $P_{\text{rental\_car\_C1}}$  would only contain tuples  $t_{C6}$  and  $t_{C7}$ . Both match the customer's wish for provider Hertz and both offer the quality of a 'compact' or even better car while meeting the price level of 'compact' cars.

In contrast to this, by usage of the trivial SV-relation for  $P_{\text{rental\_car}}$ , tuple  $t_{C4}$  would also be on the maximal level of the BTG of  $P_{\text{rental\_car\_C1}}$ . That is because tuple  $t_{C4}$  is indifferent to the others. However, treating  $t_{C4}$  from provider Europcar as 'equally good' to the perfectly matching tuples  $t_{C6}$  and  $t_{C7}$  would contradict the understanding of most customers.

## B) Pareto preference

Now consider the following pareto preference:

$P_{\text{rental\_car\_C2}} := P_{\text{rental\_car}} \otimes P_{\text{rental\_car}^{**}}$  with:

$P_{\text{rental\_car}^{**}} = (\text{Provider}, \langle_{\text{rental\_car}^{**}}, \cong_{\text{rental\_car}^{**}}) := \text{POS}(\text{Provider}, \{\text{'Europcar'}\})$

The customer again prefers a 'compact' car. But in opposition to the example above, he or she would like to rent a car from Europcar. Both preferences are equally important to him or her.

Using the regular SV-relation for  $P_{\text{rental\_car}}$ , the maximal level of the BT-graph would only contain tuple  $t_{C4}$  as it is the only perfect match for both base preferences. However, since  $t_{C4}$  is indifferent to  $t_{C6}$  and  $t_{C7}$  with respect to  $P_{\text{rental\_car}}$ , both would also be maximal elements of the BTG by using the trivial SV-relation. This would contradict the intuitive understanding of a great deal of customers.  $\diamond$

### 4.3.2 Complexity and Performance Considerations

Performance issues are always a factor in the human-computer-interaction. It will be shown that the complexity of the rather intricate TRADEOFF constructor is linear when deployed in common e-commerce scenarios. Furthermore, first insights will be given for further performance considerations.

#### Lemma 4.1

Let  $n$  be the number of tuples in a relation  $R$  and  $k = |\text{QUAL\_SET}|$  the number of elements in a quality set. Then the complexity for computation of the BMO set of a TRADEOFF preference is in worst case  $O(k \cdot n)$ .  $\text{D}$

#### Proof 4.1

Lemma 4.1 holds due to the construction of the TRADEOFF preference (definitions 4.6- 4.9):

1. For the computation of  $k$  occurring MAXPRICE-values, a linear scan of relation  $R$  is necessary. For each tuple  $t$  it has to be examined whether the current price in  $t$  represents a new maximum price for the corresponding quality. This can be done in  $O(k \cdot n)$ . Note that the complexity can be reduced using an appropriated *hash table*, e.g., built up on a *minimal perfect hash function* as shown in [Cic80, FHC+92], for a fast lookup of a quality's present MAXPRICE-value. Assuming the lookup of the MAXPRICE-values is possible with  $O(1)$ , the complexity can be reduced to  $O(n)$ .
2. Determining the price levels with corresponding intervals is easily possible, as the computed MAXPRICE-values of all quality levels are used as limits of the intervals. Note that the MAXPRICE-values can be obtained from the hash table of the previous step. The number of price levels is obviously limited by  $k$ . Thus, it results in a complexity  $O(k)$ .
3. By using a linear scan, the index of the corresponding quality in QUAL\_SET has to be determined for each tuple of the database. Afterwards the distance can be determined just by subtraction (case A in Definition 4.9) or by using the price level function (case B in Definition 4.9). Since the amount of price levels depends on the size of QUAL\_SET, the computation of the distance of each tuple can be done in  $O(k \cdot n)$ .
4. Immediately after computing the distance of a tuple, it has to be examined whether it is better or equally good compared to the other tuples currently included in the BMO result set. Note that TRADEOFF represents a sub-constructor of SCORE<sub>d</sub> according to Definition 4.9, i.e., a weak order preference. Therefore, this can be done in  $O(1)$ .

Thereby, the total worst case complexity is  $O(k \cdot n)$ .  $\square$

Usually, humans do not differentiate more than 10 states ([Fis04]). With regard to the tourism domain, even less states are generally used, e.g., one differentiates between only 5 qualities

(stars) for hotels and only 3 qualities for flights (economy, business, first). It appears reasonable to regard this order of magnitude for a realistic view of complexity.

#### **Corollary 4.1**

Assuming  $k = |\text{QUAL\_SET}| \leq 10$  in a practical deployment of the TRADEOFF preference constructor. Then  $k$  is usually much less than  $n$  that represents the number of tuples in a relation  $R$ . Therefore, the complexity for computation of the BMO result set of a TRADEOFF preference on relation  $R$  can be regarded as linear with  $O(n)$ .  $\text{D}$

#### **Performance considerations**

Since an important part of the total complexity depends on the determination of price levels, this should be further examined. In practical deployment it is absolutely necessary to analyze whether the price levels must be determined separately for each query. Obviously, this depends on database, application, and the rate of change. Therefore, it should be differentiated between three scenarios:

1. Tuples of the database are rarely inserted or updated. For example, the German railway company 'Deutsche Bahn' rarely changes prices or connections within a calendar year. In this case *materialized views* ([GM99]), keeping the price levels ready, could be the method of choice.
2. Also for databases with a middle rate of changes, materialized views should be examined. However, this has to be done separately for each application.
3. However, if the database of an application is frequently updated or changed, the price levels have to be determined for each query.

Of course, these sample scenarios and performance issues are not meant to be complete. However, they offer a first insight and show that there is potential for further reduction of the complexity of this rather intricate constructor.

Note that the computed distances of the TRADEOFF constructor can be used directly as inputs for novel and efficient algorithms such as the *Hexagon* algorithm ([PK07]) for computation of the pareto optimal set. Thus, an efficient computation of pareto preference queries, which include a TRADEOFF preference, is possible.

#### **4.3.3 Quality Valuation**

As shown in [Fis04] arguing about the quality of search results is an important factor in selling goods. Obviously, the same applies to the tourism domain. Delivering quality valuations for travel packages or for parts of the travel could provide customers with important information. Thereby, the buying behavior of customers can be influenced. Furthermore, quality valuations with respect to the customers' preferences play an important role for the advanced

query processing approach that will be presented in section 4.5. In order to seamlessly integrate the TRADEOFF preference constructor into the preference framework of [Kie02, Kie05], an appropriate quality valuation will be provided in the following.

For the preference  $P := \text{TRADEOFF}(A, \text{QUAL\_SET}, q_z)$ , the quality for a search result can be determined by means of the corresponding distance. Obviously, if  $\text{DISTT} = 0$  for a tuple  $t[A] \in \text{BMO-set}$ , the quality can be declared as 'perfect'. For quality classifications except 'perfect', a knowledge engineer has to declare suitable ranges with respect to the situation  $s$ . An appropriate quality function  $\text{QUAL}_{P,s}$  is instantiated for a BMO result set as specified in Definition 3.19.

**Definition 4.10**  $\text{QUAL}_{P,s}$ -function of a TRADEOFF preference

For a given preference  $P := \text{TRADEOFF}(A, \text{QUAL\_SET}, q_z)$  in a situation  $s$  for a result tuple  $t \in \text{BMO}$ ,  $\text{QUAL}_{P,s}$  is of the following form, where  $0 \leq b_1(s) \leq b_2(s) \leq b_3(s)$  with  $b_i(s) \in \mathbb{N}_0$ :

$$\text{QUAL}_{P,s}(t) := \begin{cases} \text{'perfect' } & , \quad \text{DISTT}(t[A]) = 0 \\ \text{'very good' } & , \quad 0 < \text{DISTT}(t[A]) \leq b_1(s) \\ \text{'good' } & , \quad b_1(s) < \text{DISTT}(t[A]) \leq b_2(s) \\ \text{'acceptable' } & , \quad b_2(s) < \text{DISTT}(t[A]) \leq b_3(s) \\ \text{'sufficient' } & , \quad b_3(s) < \text{DISTT}(t[A]) \end{cases} \quad \triangleright$$

**Example 4.10** Sample  $\text{QUAL}_{P,s}$ -function of a TRADEOFF preference

By using a new online travel portal, student Derek is looking for a flight in the economy class. He would really like to take a flight in business or first class, as long as its price does not exceed his price limit. That means the price should be in a range usually only economy-flights can offer. His preference can be expressed:

$P_{\text{Flight}} := \text{TRADEOFF}(\{\text{Category}, \text{Price}\}, \text{QUAL\_SET}_{\text{flight}}, \text{'economy'})$   
with  $\text{QUAL\_SET}_{\text{flight}} = \{\text{'economy'}, \text{'business'}, \text{'first'}\}$ .

Due to the importance of the price to Derek, the shop owner may decide to use valuations of 'perfect', 'acceptable', and 'sufficient' to denote the quality of a result  $t$ . He chooses  $b_1(s) = b_2(s) = 0$ ,  $b_3(s) = 1$ . The corresponding  $\text{QUAL}_{P,s}$ -function is the following:

$$\text{QUAL}_{P,s}(t) := \begin{cases} \text{'perfect' } & , \quad \text{DISTT}(t[A]) = 0 \\ \text{'very good' } & , \quad \emptyset \\ \text{'good' } & , \quad \emptyset \\ \text{'acceptable' } & , \quad 0 < \text{DISTT}(t[A]) \leq 1 \\ \text{'sufficient' } & , \quad 1 < \text{DISTT}(t[A]) \end{cases} \quad \diamond$$

According to [Fis04] for an intuitive and believable valuation,  $QUAL_{P,s}$  must satisfy the quality postulate of Definition 3.20. That is, if a tuple  $t$  is better than tuple  $t'$  with respect to preference  $P$  in a situation  $s$ , then the quality of tuple  $t'$  must not be better than the quality of tuple  $t$ . As will be shown, the presented quality function satisfies this postulate:

**Lemma 4.2**

For the  $QUAL_{P,s}$ -function of the TRADEOFF preference it holds that for all elements  $t, t'$  with  $t[A], t'[A] \in \text{dom}(A)$ :  $t <_P t'$  implies  $QUAL_{P,s}(t) \leq QUAL_{P,s}(t')$ .  $\text{D}$

**Proof 4.2**

The proof is obvious because of the definition of the  $QUAL_{P,s}$ -function.

- 1)  $t <_P t'$
- 2)  $t <_P t'$  iff  $DISTT(t[A]) > DISTT(t'[A])$
- 3) Assumption:  $\exists t, t': t <_P t'$  and  $QUAL_{P,s}(t) > QUAL_{P,s}(t')$

Then according to the definition of  $QUAL_{P,s}$  either a) or b) hold:

- a)  $\exists v: DISTT(t[A]) < b_v(s) \leq DISTT(t'[A])$  which is a contradiction to 2)
- b)  $DISTT(t[A]) = 0 \wedge DISTT(t'[A]) > 0$  which is also a contradiction to 2)  $\square$

In conclusion, a novel preference constructor dealing with the typical price-quality tradeoff was presented. Customers usually do not want to get a flood of results by specifying a constraint such as “I would like to have a hotel with 3 stars or more”. The TRADEOFF constructor takes the customers' price constraints, which are mostly related to the quality, into account. Thereby, customers can get suitable products with respect to their underlying price constraint. While the deployment of the TRADEOFF constructor may not be beneficial for all types of customers and situations, it provides one more instrument for a deep personalization of the first stage of the search process presented above. In doing this, it represents one more component that assists customers in achieving their desires.

## 4.4 Smart Preference Elicitation

As demonstrated above, travelers usually have a wide variety of preferences. The work of Kiebling ([Kie02, Kie05]) offers a lot of intuitive all-purpose preference constructors for the modeling of typical preferences. Furthermore, the TRADEOFF constructor defined above can be applied for the typical price-quality problem in tourism-related electronic commerce. In other words, there are a lot of suitable preference constructors for an appropriate modeling of the customers' preferences during the first stage of the search process.



Yet before preferences can be constructed by such means, they have to be elicited from the customer. Travel online portals often try to elicit preferences from customers they are not able to specify. Because of this, customers are often forced to formulate means objectives (see form-driven preferences of Definition 4.2). For example, people often have preferences for their journey to an airport. Usually, the trip should be cheap, comfortable, and/or by a specific means of transportation. But instead of specifying such preferences, customers have to choose a corresponding departure airport by themselves. This may lead to unsatisfied customers and may even prevent the customer from finding and booking a suitable offer. Hence, both customer and travel portal provider should be interested in avoiding the problem. In the following, it will be shown how to deal with the problem.

The incomplete knowledge of customers represents one important part of the problem. Usually, customers do not know all the relevant facts, e.g., the prices for railway tickets to all the airports close to their location. But such domain knowledge is essential for specifying a suitable airport that matches their preferences. The need to specify a form-driven preference usually leads to the following customer behavior:

- Customers specify a means objective based on estimation. They might, for instance, guess that the airport in Munich is the best choice with respect to their underlying preference for a cheap arrival via railway (case A in Figure 4.8).
- Customers try to obtain relevant information to specify their objective. For example, they might ask friends or possibly might try to use online route finders or internet websites of railway companies to find out which airport would be the best choice with respect to their preferences (case B1 and B2 in Figure 4.8).

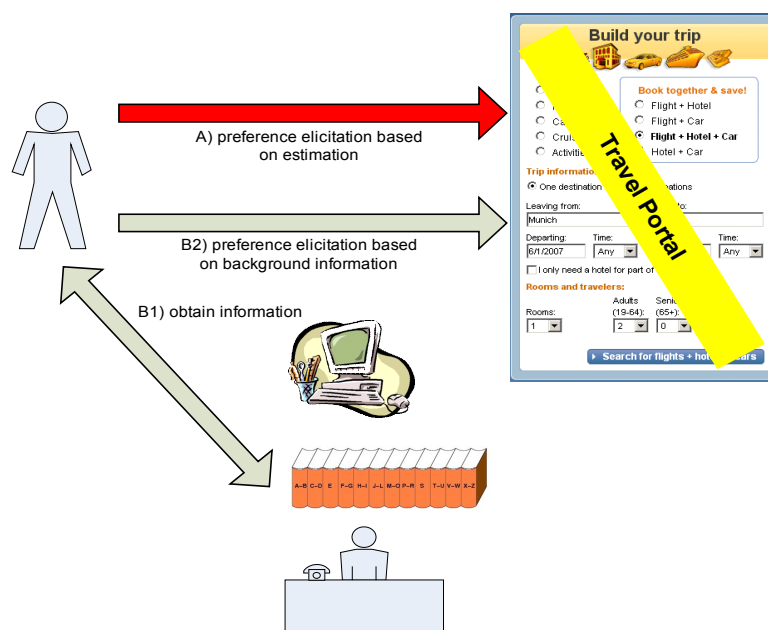


Figure 4.8. Specifying a form-driven preference

Obviously, a personalized and customer-friendly portal should avoid form-driven preferences by allowing customers to directly specify their underlying preferences. However, this can often require more information than a travel portal might have stored because it is not feasible to store every kind of possibly relevant information. Travel portals usually do not have, for example, data about railway connections. In the following, a preference elicitation based on the integration of heterogeneous information sources will be presented.

#### 4.4.1 Preference Elicitation Based on Information Integration

As described above, specifying a parameter for a form-driven preference requires implicit matchmaking by the customer. He or she needs information in order to find suitable data which correspond to the underlying preferences. How can a travel portal elicit the desired preference from a customer? Is it enough to give him or her a software component to find such data, e.g., to find an airport with respect to the underlying preferences?

First, an elicitation process based on the integration of data is presented in general. Afterwards, different possibilities for the connection between travel portal and integrated data will be examined in detail.

Definition 4.11 describes the major steps of the entire elicitation process leading to a preference of the customer in terms of the attribute or property the travel portal is interested in.

##### **Definition 4.11 Workflow for preference elicitation based on information integration**

The process is defined in three stages:

1. Identification of problematic attributes
2. Discovery of relevant data sources
3. Integration of data sources and determination of preferences D

##### **1. Identification of problematic attributes**

In a first step, relevant attributes that may cause form-driven preferences have to be identified. For this purpose, several case studies and customer questionnaires can provide valuable information and feedback.

##### **2. Discovery of relevant data sources**

All relevant data sources have to be discovered on the internet. Since the information space of the *World Wide Web* (WWW) is huge, this might require some effort as described in [Gri07]. Considering the airport example, online route finders and websites of railway companies can provide data, e.g., costs and times, about the journey to an airport.

### **3. Integration of data sources and determination of preferences**

A three-tier architecture with *Data Source Adapter Layer*, *Mediation Layer*, and *Client Layer* provides a good foundation for the integration process ([Bar00, Hal00]). Heterogeneous data are transformed into a structured format by so-called *wrappers* of the adapter layer. For technical aspects of the integration we refer to ([Bar00, Gri07, Hal00]), since the entire process and a corresponding architecture are described in detail there. Therefore, in the following it is assumed that the required data can be integrated into a database relation by a corresponding framework. But how can this relation be used to identify and gain the customer's preferences for a required attribute?

First, the database relation for the integrated information will be examined. It is obvious that such a database relation has to include at least problematic attributes identified in stage 1 as well as all attributes which are relevant for the customer's underlying preferences. Taking the airport scenario into consideration, it is absolutely necessary to integrate the airport names/codes and an attribute for the costs of the journey. Therefore, an integrated database relation is defined as follows:

#### **Definition 4.12 Integrated database relation**

Let  $P = (A, <_p, \cong_p)$  represents a customer's underlying preference and  $B$  be the attribute the travel portal requires a preference for. Then for an integrated database relation  $R(A_1: \text{dom}(A_1), \dots, A_m: \text{dom}(A_m))$ , it must hold that

- $\{A, B\} \subseteq \{A_1, \dots, A_m\}$ . D

#### **Example 4.11 Sample integrated database relation**

Student Meredith would like to book a flight to New York. As usual, she first has to define a departure airport. Since her budget is limited, she prefers to get to the airport as cheaply as possible. Price differences of up to \$10 do not matter. This preference can be expressed with  $P := \text{LOWEST}_{10}(\text{PRICE})$ . Yet she does not know which airports match her preferences (form-driven preference). Usually, she would have to try to gain the related information (see Figure 4.8). However, this time her travel portal aims to provide all necessary information in order to elicit a preference for airports from her.

By using a wrapper-based architecture ([Hal00]), relevant information can be gained from online route finders or railway companies. An appropriate database relation is represented by Table 4.4. It contains the names and identification codes for airports as well as information about the costs of the arrival.

| Airport-ID | Name            | Price in \$ | Time in min |
|------------|-----------------|-------------|-------------|
| MUC        | Munich          | 10.--       | 70          |
| NUE        | Nuremberg       | 35.--       | 70          |
| STR        | Stuttgart       | 20.--       | 90          |
| FDH        | Friedrichshafen | 25.--       | 90          |

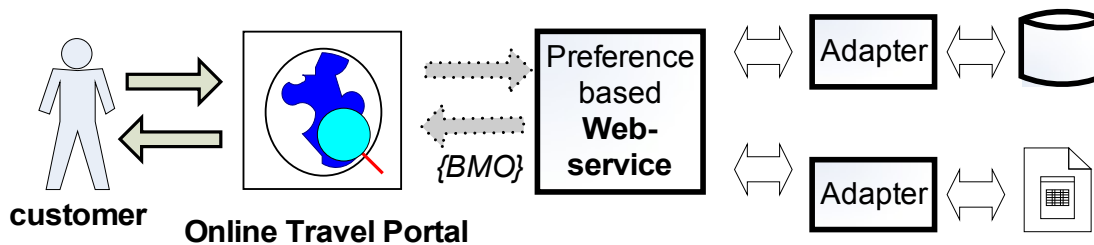
**Table 4.4. Sample database for airports**

◇

As described above, relevant data sources can be integrated into a database relation. However, there are several scenarios possible from which relevant preferences can be gained. How shall the preference be gained and how deep shall the gaining process itself be integrated into the online travel portal? In the scope of this work, three kinds of scenarios are presented, of which each is based on a different amount of effort for the travel portal.

Scenario A: delivering a BMO result set:

A customer who is required to specify a form-driven preference can be supported by a separate software component. This, e.g., a *web service* ([Web07]), can integrate relevant data sources from the internet and deliver a BMO result set with respect to the underlying preference of the customer (Figure 4.9). These kinds of components can be delivered by a third party, e.g., a solution provider, and can be coupled easily into existing travel online portals. Unfortunately, while this offers some kind of support for customers, due to the loose coupling to the travel portal only a basic level of personalization can be provided.



**Figure 4.9. Preference elicitation: low integration effort**

For illustration, consider a web service that delivers an appropriate airport with respect to a customer's preferences for the means of transportation and the necessary costs. For this purpose, it integrates information of railway companies and route finders. The web service can be coupled easily to the travel portal by a well-specified interface, i.e., the integration effort is rather low. After specifying the underlying preferences for the transport, only the matching airports are delivered to the online travel portal. Afterwards these airports can be taken into consideration for further processing. They can be integrated into the travel portal, for example, by a corresponding hard constraint or a POS preference.

While customers indeed are supported in finding appropriate airports, there could be other preferences or global preferences (see also subsection 4.4) which cannot be considered accordingly. There might be an airport, for example, that is not delivered by the web service because it requires a slightly more expensive arrival. However, a flight from this airport could exactly match the customer's preference for a certain airline.

#### Example 4.12 Sample scenario

Again, consider student Meredith of Example 4.11. In addition to her preference for the journey to the airport  $P := \text{LOWEST}_{10}(\text{PRICE})$ , she might have the preference to take a flight with 'Lufthansa'. The latter one can easily be modeled with  $P_{\text{airline}} := \text{POS}(\text{AIRLINE}, \text{'Lufthansa'})$ .

A web service as described above would only deliver the BMO result set, i.e., 'MUC', to the travel portal with respect to Meredith's preference  $P$  and the integrated data of Table 4.4. Considering Table 4.5 and treating the preference for departure airport 'MUC' as hard constraint, only flight 879 would be delivered. Yet this flight does not match Meredith's preference for the airline 'Lufthansa'. In opposition to that, a corresponding preference for the departure airport  $P_{\text{airport}} := \text{POS}(\text{DepartureAirport-ID}, \text{'MUC'})$  and a complex preference  $P_{\text{flights}} := P_{\text{airline}} \otimes P_{\text{airport}}$  could be modeled. Thereby, all three flights of Table 4.5 are delivered.

| Departure Airport-ID | Destination Airport-ID | Flight Nr. | Airline   |
|----------------------|------------------------|------------|-----------|
| MUC                  | JFK                    | 879        | Air Snow  |
| STR                  | JFK                    | 123        | Lufthansa |
| NUE                  | JFK                    | 540        | Lufthansa |

**Table 4.5. Sample database for airports**

◇

While the airports 'STR' and 'NUE' both offer a flight to New York with 'Lufthansa', the journey to 'STR' is cheaper (Table 4.4) and should be preferred by Meredith with respect to preference  $P$ .

In conclusion, the integrated data can be used to get a BMO result set with respect to the underlying preference of the customer. Thereby, a preference on an attribute required by the travel portal, e.g., for departure airports, can be constructed. This approach supports the customer, avoids form-driven preferences, and can easily be integrated into existing travel portals. However, as shown above, the BMO result set might not always be sufficient for the construction of a preference.

Scenario B: delivering a preference:

Again, consider a loosely coupled web service as described above (Figure 4.9). But instead of only delivering a BMO result set, an advanced approach could deliver an appropriate preference on the attribute required by the online travel portal. First, induced database preferences are defined:

**Definition 4.13 Induced database preferences:  $P^I$** 

Assume an instance of schema  $R(A_1: \text{dom}(A_1), \dots, A_m: \text{dom}(A_m))$ , a preferences  $P = (B, <_P, \cong_P)$ , and  $P^I = (A, <_{P^I}, \cong_{P^I})$ , where  $A, B \subseteq \{A_1, \dots, A_m\}$ . Let attribute  $A$  represent a candidate key of  $R$  and given  $x, y \in \text{dom}(B)$  and  $a, b \in \text{dom}(A)$ , one defines:

- Preference  $P^I = (A, <_{P^I}, \cong_{P^I})$  is an **induced database preference** of  $P$  on  $R$ , if for  $\forall a, b \in \text{dom}(A), \exists (a, x), (b, y) \in \pi_{A,B}(R): a <_{P^I} b \text{ iff } x <_P y.$   $\text{D}$

Of course, preference  $P$  may also represent a complex preference.  $B$  is defined on a set of attributes  $B_j$  according do the definition of preferences (Definition 3.1). In this case, each  $x$  and  $y$  is represented by a number of  $x_j$  and  $y_j$ , respectively.

**Example 4.13 Sample for induced database preferences**

Consider Meredith's price preference  $P$  and the integrated relation  $R$  of Example 4.11. A preference  $P^I = (\text{Airport-ID}, <_{P^I}, \cong_{P^I})$  with the following BTG can be induced from  $P$  on  $R$ :



It is reasonable to assume that Meredith would prefer 'MUC' (Munich) to 'STR' (Stuttgart), since the costs of the journey to Munich are not as high.

Now considering Example 4.12 again, Meredith's induced preference for an airport  $P^I$  and the preference for a flight with 'Lufthansa' can be combined to:  $P_{\text{Flights}} := P_{\text{Airline}} \otimes P^I$

Thereby, only the flights from 'MUC' and 'STR' are delivered, since the flight from 'NUE' (Nuremberg) is dominated with respect to the induced preference  $P^I$ . Hence, a more comprehensible, better personalized result can be delivered.  $\text{D}$

In the following, it will be shown that  $P^I$  represents a strict partial order preference.

**Lemma 4.3**

An induced database preference  $P^I = (A, <_{P^I}, \cong_{P^I})$  of preference  $P = (B, <_P, \cong_P)$  on relation  $R$  as defined in Definition 4.13 represents a strict partial order.  $\mathcal{D}$

**Proof 4.3**

1) Given  $x, y \in \text{dom}(B)$ . Attribute  $A$  represents a candidate key of  $R$  according to Definition 4.13, i.e., it holds for all  $\forall a, b \in \text{dom}(A)$  and  $(a, x), (b, y) \in \pi_{A,B}(R)$ :  $a = b$  implies  $x = y$ .

Given  $a, b, c \in \text{dom}(A)$  and  $x, y, z, w \in \text{dom}(B)$ :

- Irreflexivity:

Assumption:  $a <_{P^I} a$

a)  $\exists (a, x), (a, y) \in \pi_{A,B}(R)$ :  $a <_{P^I} a$  iff  $x <_P y$

b) due to 1):  $a = a$  implies  $x = y$

c) due to b):  $\exists (a, x) \in \pi_{A,B}(R)$ :  $a <_{P^I} a$  iff  $x <_P x$

which is a contradiction due to the definition of  $<_P$  (Definition 3.1)

- Transitivity:

$\exists (a, x), (b, y), (b, z), (c, w) \in \pi_{A,B}(R)$ :  $a <_{P^I} b \wedge b <_{P^I} c$  iff  $x <_P y \wedge z <_P w$

a) due to 1):  $b = b$  implies  $y = z$

b) due to a):  $\exists (a, x), (b, y), (c, w) \in \pi_{A,B}(R)$ :  $a <_{P^I} b \wedge b <_{P^I} c$  iff  $x <_P y \wedge y <_P w$   
implies  $x <_P w$  (due to Definition 3.1)

iff  $a <_{P^I} c$

□

The following algorithm determines an induced database preference  $P^I = (A, <_{P^I}, \cong_{P^I})$  of a preference  $P = (B, <_P, \cong_P)$  on relation  $R$ . If  $P$  represents a complex pareto and prioritized preference, then  $P^I$  will be determined on the basis of the underlying base preferences. For each base preference an accordingly induced LAYERED-preference on  $A$  will be determined. Note that only SCORE<sub>d</sub> base preferences can be considered, i.e., the algorithm does not work for EXPLICIT. While this preference offers a rather flexible constructor, the construction of suitable and comprehensible interfaces for the practical deployment is complicated. Furthermore, advanced algorithms for the computation of BMO result sets of pareto preferences such as Hexagon ([PK07]) do not work for EXPLICIT preferences. Therefore, this restriction is reasonable.

If  $n$  is the number of tuples in relation  $R$ , the algorithm for determination of an induced preference  $P^I$  is defined as follows:

**INPUT:** relation  $R$ , preferences  $P = (B, <_P, \cong_P)$ , attribute  $A$

```

1: baseP[] = determineAllBasePreferences(P);
2: for (i = 1, i <= n) do // each tuple of R
3:   for (j = 1, j <= baseP.length) do // each base preference
4:     scored = tuple(i).computeScoreD(baseP[j]);
5:     baseInducedP[j].buildNewLayers(scored, tuple(i).getA());
6:   end for
7: end for
8: accumulateComplexInducedPreference(baseInducedP[], P);

```

**OUTPUT:** quality induced database preference  $P^I = (A, <_{P^I}, \cong_{P^I})$

In the first line, the algorithm determines all base preferences of  $P$  and puts them into the array  $baseP[]$ . If  $P$  itself is a base preference, then  $P$  represents the first and only element of  $baseP[]$ . By using loops, thereafter, for all tuples of  $R$  (line 2) and all base preferences of  $P$  (line 3), the  $SCORE_d$ -values for these base preferences on the corresponding attribute values of the current tuple are computed (line 4). In line five, the induced  $LAYERED_m$ -preferences on attribute  $A$  are modified. The higher the computed score for a base preference (line 4), the better is a tuple with respect to this base preference and the lower is the layer of this tuple's attribute value of  $A$  in the induced  $LAYERED_m$ -preference. Remember that layer 1 represents the maximal elements of the  $LAYERED_m$ -preference and elements of a level with lower number are better (Definition 3.8). After line 5, for each base preference of  $P$  ( $baseP[]$ ) an induced preference ( $baseInducedP[]$ ) is determined. Each induced preference is represented by an appropriate  $LAYERED_m$ -preference on attribute  $A$ . For a base preference  $P$  holds, if two tuples have an equal  $score_d$ -value with respect to  $P$ , then the corresponding values of attribute  $A$  of these two tuples are in the same layer in the induced  $LAYERED_m$ -preference. Finally, in line 8 all induced base preferences of  $baseInducedP[]$  are accumulated exactly like the base preferences of  $P$ . For instance, if two base preferences were accumulated by the pareto complex preference constructor, the induced base preferences ( $LAYERED_m$ -preferences on attribute  $A$ ) will also be accumulated by pareto.

The functionality of the algorithm will be illustrated in the following example.

#### Example 4.14 Sample quality induced database preference

Consider Table 4.4 of Example 4.11. Assume Meredith has the following preference:  
 $P := P_1 \otimes P_2$  with  $P_1 := LOWEST_{10}(PRICE)$  and  $P_2 := LOWEST_{10}(TIME)$



The algorithm computes the  $SCORE_d$ -values for each base preference. For the  $LOWEST_d$  preference, the  $SCORE_d$ -value depends on the distance (Definition 3.7):  $score_d := -dist_d$ . This leads to the values illustrated in the table below.

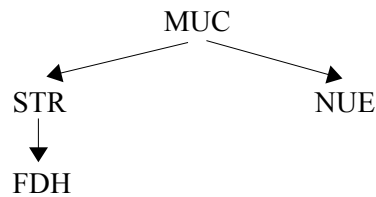
| Airport-ID | Name            | Price in \$ | $SCORE_d$<br>$P_1$ | Time in min | $SCORE_d$<br>$P_2$ |
|------------|-----------------|-------------|--------------------|-------------|--------------------|
| MUC        | Munich          | 10.--       | 0                  | 70          | 0                  |
| NUE        | Nuremberg       | 35.--       | -3                 | 70          | 0                  |
| STR        | Stuttgart       | 20.--       | -1                 | 90          | -2                 |
| FDH        | Friedrichshafen | 25.--       | -2                 | 90          | -2                 |

**Table 4.6. Sample database for induced preferences**

A preference on the required attribute, i.e., the Airport-ID, is induced by means of the  $SCORE_d$ -values. The higher the value is, the lower is the layer of the required attribute. Thereby the following two  $LAYERED_m$  preferences are induced by  $P_1$  and  $P_2$ :  $P_1^I := LAYERED_5(\text{Airport-ID}, (\{\text{'MUC'}\}, \{\text{'STR'}\}, \{\text{'FDH'}\}, \{\text{'NUE'}\}, \text{other values}))$  and  $P_2^I := LAYERED_3(\text{Airport-ID}, (\{\text{'MUC'}, \text{'NUE'}\}, \{\text{'STR'}, \text{'FDH'}\}, \text{other values}))$

For example, base preference  $P_1 := LOWEST_{10}(\text{PRICE})$  induced  $P_1^I$ . Thereby, the airport of 'MUC' is better, i.e., has a lower layer, than the airport of 'STR'. This is reasonable with respect to preference  $P_1$ , since the journey to 'MUC' is cheaper.

In the last step of the algorithm, the induced complex preference  $P^I := P_1^I \otimes P_2^I$  is constructed by pareto accumulation analogously to the original preference ( $P := P_1 \otimes P_2$ ). This leads to the following BTG:



Based on the data in  $R$ , it is reasonable that Meredith would prefer Munich (MUC), since the journey to Munich is faster and cheaper than to the others.  $\diamond$

#### Lemma 4.4

Let  $n$  be the number of tuples in a relation  $R$  and  $m$  be the number of base preferences  $P_i$ . For simplification assume  $k$  represents the maximum number of levels for base preferences on the corresponding attributes of  $R$ . Then the complexity for computation of the induced database preference  $P^I$  is in worst case  $O(n \cdot m \cdot \log k)$ .  $\text{D}$

**Proof 4.4**

Lemma 4.4 holds due to the construction of the algorithm:

1. One linear scan of the database is sufficient to compute the score-values for all  $m$  base preferences  $P_i$ . This can be done in  $O(n \cdot m)$ .
2. An induced base preferences  $P_i^l$  can be adjusted immediately after the computation of the corresponding score-value of a tuple. The complexity to sort the new value into the existing ordered list of layers of  $P_i^l$  depends on the number of layers that, obviously, corresponds with the maximum number of levels  $k$  of base preference  $P_i$ . This leads to  $O(\log k)$  per base preference and tuple.

Thereby, the total worst case complexity is  $O(n \cdot m \cdot \log k)$ . □

By using the presented algorithm, heterogeneous data sources can be integrated in a component that delivers an induced database preference of the customer to the travel portal. Thus, form-driven preferences can be avoided. Furthermore, this approach offers a better personalization than the first one that only delivers a corresponding BMO set with respect to the customers underlying preferences.

However, assume Meredith is interested in getting the cheapest combination of flight and journey to an airport. Scenario A and B as presented above cannot deal with such global preferences. Only the BMO set (scenario A) or an induced database preference (scenario B) are delivered with respect to the customers underlying preferences. In order to deal with complex personalized scenarios, more integration effort is necessary.

**Scenario C: integrating the entire relation:**

The best solution in terms of personalization is to integrate all relevant data sources into the travel portal. In [Hal00], for example, all relevant data sources are integrated into the mediation layer that is used to “simplify, abstract, reduce, merge, and explain data”. The architecture is illustrated in Figure 4.10.

It is obvious that this kind of integration requires much more effort and costs than the other ones. The corresponding relation schemes have to be modeled and integrated first. Later instances of these schemes have to be integrated into the travel portal on the fly. Consider the airport scenario again. The information of route finders etc. are directly integrated into the travel portal now. But what happens if the server of the online route finder is not available? The travel portal has to consider and implement fallback solutions. Thus, the system of the travel portal becomes significantly more complex. Furthermore, travel portal providers are often reluctant to intervene in their running systems in such a significant manner.

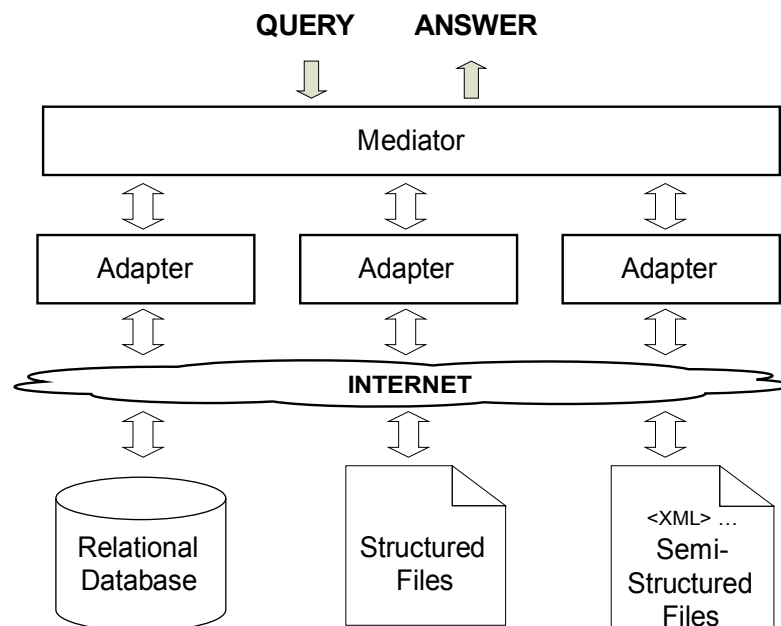


Figure 4.10. Architecture for the integration of data sources (excerpt from [Hal00])

Yet this kind of integration offers the best level of personalization. The integrated data can directly be used for personalized search queries. Not only can the required preferences of customers be gained, but also the integrated data itself can be used for further processing. For example, the prices for the arrival to an airport could be included in the computation of the total price of the entire packages.

#### 4.4.2 Smart Preference Elicitation in Conclusion

For a higher level of personalization usually more integration effort is required (see Figure 4.11). In scenario A), the integrated data are used to determine the best solutions (BMO result set) with respect to the customer's preferences, e.g., to deliver an airport that matches the corresponding preferences. Thereby, form-driven preferences can be avoided and the customer can be supported. This functionality can be loosely coupled to the travel portal, hence requires only little effort and costs. Yet this can often be insufficient in combination with other preferences. Therefore, in scenario B) the integrated data are used to determine an induced preference on the required attribute. More information can be offered to the travel portal, while only little more integration effort than in A) is necessary. However, the best solution in terms of personalization is the integration of all relevant data into the travel portal as described in scenario C). Unfortunately, this requires the most expensive and most complex integration effort.

As described at the beginning of the chapter, there are a number of suitable preference constructors for an appropriate modeling of the customers' preferences during the first stage of the search process. Unfortunately, before preferences can be constructed by such means, they

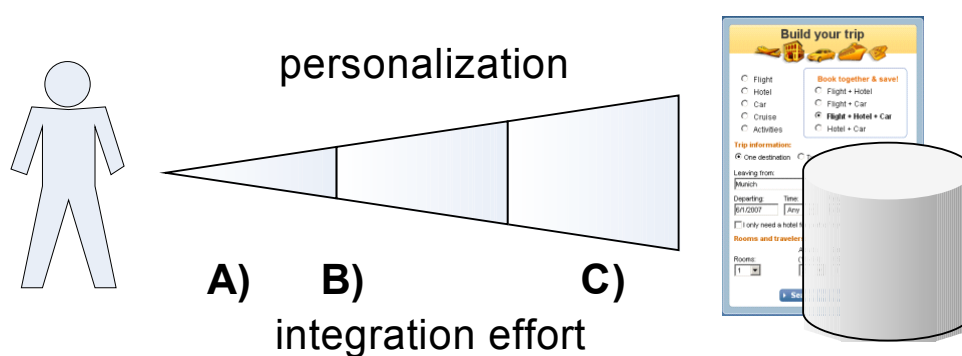


Figure 4.11. Level of personalization and required effort

have to be elicited from the customer. Sometimes customers are not able to specify a preference that is required by the portal, for instance, a desired airport. They do not have enough information to choose an appropriate one with respect to their underlying preferences.

In this section, a novel approach for a smart preference elicitation was presented. The integrated data can be used to get certain preferences elicited from the customer. Thereby, the annoying form-driven preferences can be avoided. Furthermore, three different approaches were presented for gaining the preferences by means of integrated data sources. Each approach requires a different amount of effort for the integration, but also offers a different kind of personalization. It depends on situation and application to decide which approach suits a specific travel portal's requirements best. But no matter which approach is chosen, the customers experience more support as before. Taking the underlying preferences into account can increase the *customers' satisfaction* ([HLH07]) and by means of this may lead to more business volume of the travel portal.

## 4.5 Advanced Preference Query Processing

As described above, even in simple travel scenarios there might be preferences for individual aspects of the journey and there are preferences about global constraints, e.g., for the total price of the travel package (see also Figure 2.1 / Definition 4.4). In this section, a novel query processing approach which can deal with global preferences will be presented.

For illustration purposes, first consider traveler Derek. He would like to travel to Barcelona for one week. He is able to express his preferences as follows:

1. The flight's airline **should** be Lufthansa and the category **should** be business class.
2. His hotel **should** have 3 stars and **should** be located downtown.
3. A rental car will also be needed, since he has to be flexible. The brand **should** be Renault.

4. The total package **should not** cost more than \$900.

A small sample database exists for each part of the travel. Table 4.7 represents the data for flights (F), Table 4.8 for rental cars (C), and Table 4.9 for hotels (H).

| ID              | Airline    | Category | Price (in \$) |
|-----------------|------------|----------|---------------|
| t <sub>F1</sub> | Lufthansa  | economy  | 400.--        |
| t <sub>F2</sub> | Air Spain  | economy  | 300.--        |
| t <sub>F3</sub> | Ryan Air   | economy  | 200.--        |
| t <sub>F4</sub> | Air France | business | 700.--        |
| t <sub>F5</sub> | Lufthansa  | first    | 900.--        |

Table 4.7. Database for flights (F)

| ID              | Brand   | Category | Price (for a week in \$) |
|-----------------|---------|----------|--------------------------|
| t <sub>C1</sub> | Opel    | medium   | 510.--                   |
| t <sub>C2</sub> | Renault | compact  | 290.--                   |
| t <sub>C3</sub> | BMW     | luxury   | 560.--                   |
| t <sub>C4</sub> | Ford    | economy  | 175.--                   |
| t <sub>C5</sub> | Skoda   | medium   | 470.--                   |

Table 4.8. Database for rental cars (C)

| ID              | Name     | Category | Location | Price (for a week in \$) |
|-----------------|----------|----------|----------|--------------------------|
| t <sub>H1</sub> | IBIS     | 3        | downtown | 595.--                   |
| t <sub>H2</sub> | ARCOR    | 3        | city     | 490.--                   |
| t <sub>H3</sub> | GREEN    | 3        | vicinity | 525.--                   |
| t <sub>H4</sub> | SUNSHINE | 3        | city     | 560.--                   |
| t <sub>H5</sub> | HILTON   | 5        | city     | 700.--                   |
| t <sub>H6</sub> | KING     | 4        | downtown | 560.--                   |
| t <sub>H7</sub> | TOKIO    | 2        | city     | 420.--                   |

Table 4.9. Database for hotels (H)

### Demonstration Scenario 1 – Hard Constraints:

Common travel portals would treat Derek's preferences as hard constraints and therefore compose a database query very similar to the following one:

$$\sigma_{IC \wedge GC} (F \times H \times C)$$

where

$$IC := F.Airline='Lufthansa' \wedge F.Category='business' \wedge H.Location='downtown' \wedge H.Category=3 \wedge C.Brand='Renault'$$

$$GC := (F.Price + H.Price + C.Price) \leq 900$$

Treating Derek's preferences as hard constraints, however, would lead to an empty result set. First, there is no flight matching Derek's preferences for Lufthansa and business class in the

database. Secondly, summing up the prices for the fitting hotel  $t_{H1}$  and the rental car  $t_{C2}$  already yields to \$885. No matter which flight of the database is picked, Derek's price constraint would be inevitably violated and, therefore, would lead to an empty result set.

### **Demonstration Scenario 2 – Soft Constraints:**

In order to be able to deliver alternatives, now Derek's wishes for the individual parts of the travel will be treated as preferences. This is a reasonable scenario, since people are often prepared to accept alternatives in the 'real world'.

Note that Derek would regard an 'economy' flight as alternative, if there is no suitable flight in the 'business' class. Moreover, if there is no matching hotel located in 'downtown', he would consider a hotel in the 'city' area as being better as a hotel in the 'vicinity'. He would not mind getting a qualitatively better hotel as long as the price does not exceed the price range of hotels with a category of 3 stars. Derek's preferences can, therefore, be modeled with the following preference constructors:

- $P_F := \text{POS/POS}(\text{Category}, \{\text{'business'}\}, \{\text{'economy'}\}) \otimes \text{POS}(\text{Airline}, \{\text{'Lufthansa'}\})$
- $P_H := \text{POS/POS}(\text{Location}, \{\text{'downtown'}\}, \{\text{'city'}\}) \otimes \text{TRADEOFF}(\{\text{Category}, \text{Price}\}, \{\text{'1'}, \text{'2'}, \text{'3'}, \text{'4'}, \text{'5'}\}, \text{'3'})$
- $P_C := \text{POS}(\text{Brand}, \{\text{'Renault'}\})$

The corresponding preference queries are:

- FLIGHT:  $\sigma[P_F](F)$
- HOTEL:  $\sigma[P_H](H)$
- RENTAL CAR:  $\sigma[P_C](C)$

Since there is no flight with Lufthansa in business class, tuples  $t_{F1}$  and  $t_{F4}$  would be offered as best alternatives. While tuple  $t_{F1}$  matches his wish for the airline Lufthansa, tuple  $t_{F4}$  offers a flight in business class. Tuples  $t_{C2}$  and  $t_{H1}$ ,  $t_{H6}$  perfectly matches his preferences for the rental car or hotel, respectively.

Unfortunately, there is no combination matching his price constraint of \$900 (see Table 4.10).

| Combination | Overall Price<br>(in \$) | Flight          | Rental Car      | Hotel           |
|-------------|--------------------------|-----------------|-----------------|-----------------|
| 1           | 1295.--                  | t <sub>F1</sub> | t <sub>H1</sub> | t <sub>C2</sub> |
| 2           | 1250.--                  | t <sub>F1</sub> | t <sub>H6</sub> | t <sub>C2</sub> |
| 3           | 1585.--                  | t <sub>F4</sub> | t <sub>H1</sub> | t <sub>C2</sub> |
| 4           | 1550.--                  | t <sub>F4</sub> | t <sub>H6</sub> | t <sub>C2</sub> |

**Table 4.10. Overall price and quality for travel product combinations**

Combination 2 could be offered to Derek because it matches his preferences for the individual parts of the travel (almost) perfectly. Moreover, it is closer to his price constraint than the other combinations. Yet the described approach implicitly prioritizes preferences for individual parts of the journey, since at first the results for flights, cars, and hotels matching Derek's preferences are retrieved. Only afterwards his global preference for the total price is regarded. How can these cases be treated when the preference for the total price is more or equally important to the customer than the individual preferences?

In the following, an advanced approach for preference query processing is introduced which allows customers to specify both individual and global preferences. Thereby, best alternatives can be delivered with respect to both of them.

#### 4.5.1 Preference Query Expansion Approach

As demonstrated above, global preferences are often defined on aggregated numerical attribute values. Especially in tourism-related electronic commerce, they are specified on summed up values (see also Definition 4.4). For a deeply personalized offer generation in e-commerce, two typical scenarios have to be considered:

- The summed up attribute value should be maximized or there is a lower limit for it, i.e., higher values are preferred. For example, there are travelers who want to maximize the amount of frequent flier miles for the complete travel package.
- There is an upper limit or the summed attribute value should be minimized. For example, the total price of a travel package should be lower than a certain value or the total price should be minimized in general.

The following approach is meant to deal with global preferences, i.e., soft constraints, as well as global hard constraints. Therefore, the term 'constraint' is used to denote both soft and hard constraints. Corresponding to the scenarios described above, a 'maximum' and a 'minimum sum constraint' are defined.

**Definition 4.14 Maximum and minimum sum constraint**

Given a travel package  $T := \{T_1, \dots, T_z\}$  consisting of  $z$  parts and a database relation  $R_i$  for each part  $T_i$ . Furthermore, there is a numerical attribute  $A_{a_i}$  contained in each relation  $R_i$ . A customer's wish is defined as:

- **maximum sum constraint**, if the sum of  $A_{a_i}$  attribute values of a combination of tuples  $(t_1, \dots, t_z)$  should/must be
  - a) equal or more than a given limit or
  - b) as high as possible.
- **minimum sum constraint**, if the sum of  $A_{a_i}$  attribute values of a combination of tuples  $(t_1, \dots, t_z)$  should/must be
  - a) equal or less than a given limit or
  - b) as low as possible. D

People often tend to underestimate or overestimate aggregated values. This might be one reason why they often use unrealistic values, e.g., for the total price constraint. Instead of delivering an empty result, a human travel agent would suggest an alternative, if there was no journey available matching the customer's maximum or minimum sum constraint. The presented approach will do the same. Sum constraints will be dealt with by using a novel query expansion process. In this, preference search queries for individual parts of travel, e.g., for hotels or flights, are adequately expanded and rewritten in order to address the sum constraint.

Definition 4.15 describes the major steps of the entire query processing approach dealing with individual preferences as well as maximum and minimum sum constraints.

**Definition 4.15 Workflow of query processing**

In order to deal with maximum and minimum sum constraints the query processing is defined in the following six steps:

1. Query composition
2. Query expansion
3. Preference search
4. Quality valuation
5. Composition and valuation of travel packages
6. Offer selection D

The workflow of Definition 4.15 is illustrated in Figure 4.12.



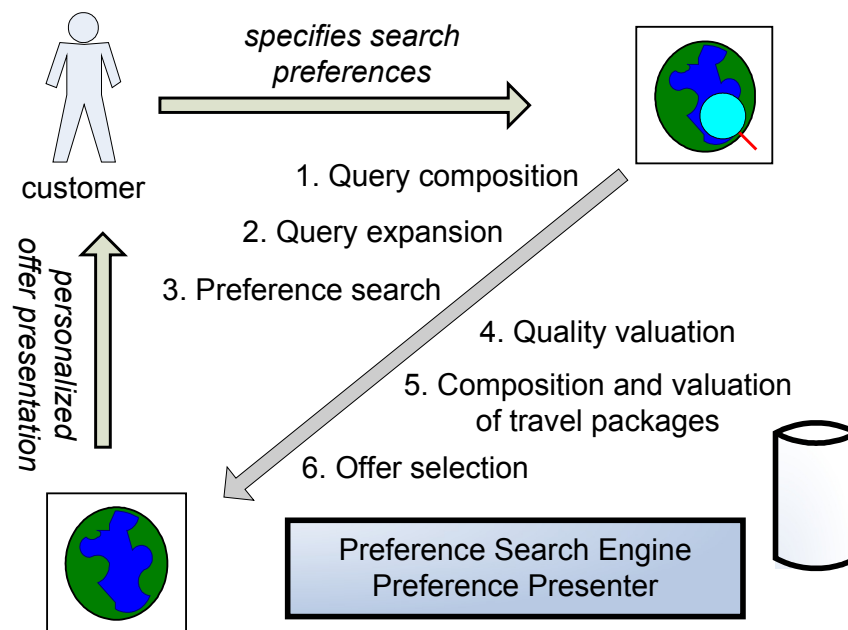


Figure 4.12. Query processing dealing with maximum/minimum sum constraints

### **1. Query composition**

Getting preferences elicited from customers can be a delicate task (see section 4.4). In particular, it requires an intuitively comprehensible and personalized search interface (section 4.1.2). After the preferences of a customer have been identified and filled into a search form, the corresponding preference search queries  $Q_i$  for each part of the travel  $T_i$  can be composed.

### **2. Query expansion**

In order to be able to provide customers with best alternatives in terms of their sum constraint, all preference queries  $Q_i$  for the individual parts of the travel are expanded. If a customer defines a maximum sum constraint, the preference constructor  $HIGHEST_d$  is applied to the corresponding attribute. In case of a minimum sum constraint, the  $LOWEST_d$  constructor is used. It will be added to the original part of the preference by the pareto preference constructor. Using pareto for the accumulation of preferences means that both parts of the query – the original preference(s) and the  $HIGHEST_d$  /  $LOWEST_d$  preference – will be treated as equally important. This way, best matches for the individual parts of the travel will be delivered with respect to the customers' original preference(s) as well as the  $HIGHEST_d$  /  $LOWEST_d$  preference.

**Definition 4.16 Preference query expansion for a maximum sum constraint**

The travel package  $T := \{T_1, \dots, T_z\}$  consists of  $z$  parts. Let  $Q_i := \sigma[P_i](R_i)$  be a preference query for a database relations  $R_i$  and the corresponding travel part  $T_i$ , respectively. For a numerical attribute  $A_{ai}$  of each database relation  $R_i$ , an expanded query  $Q_{\text{exp}i}$  for a maximum sum constraint is defined as follows:

$$\bullet Q_{\text{exp}i} := Q_i \cup Q_{i^*} \quad \text{where} \quad Q_{i^*} := \sigma[P_i \otimes \text{HIGHEST}_d(A_{ai})](R_i) \quad \text{D}$$

The expansion of preference queries for a minimum sum constraint can be done analogously. However, for the minimum sum constraint, the preference constructor  $\text{LOWEST}_d$  is used for the expansion.

**Definition 4.17 Preference query expansion for a minimum sum constraint**

Given a preference query  $Q_i := \sigma[P_i](R_i)$  for a database relation  $R_i$  and an attribute  $A_{ai}$  contained in each database relation  $R_i$ , an expanded query  $Q_{\text{min}i}$  for a minimum sum constraint is defined as follows:

$$\bullet Q_{\text{exp}i} := Q_i \cup Q_{i^*} \quad \text{where} \quad Q_{i^*} := \sigma[P_i \otimes \text{LOWEST}_d(A_{ai})](R_i) \quad \text{D}$$

Note that by just using the simple expansion  $Q_{i^*} := \sigma[P_i \otimes \text{LOWEST}_d(A_{ai})](R_i)$ , the result set could be ineligibly reduced. That is, because only the best matches with respect to **both** the original preference(s) **and** the new  $\text{LOWEST}_d(A_{ai})$  preference are delivered. Therefore, the results of expanded queries  $Q_{i^*}$  are fused with the results of original queries  $Q_i$  by using the union operator ' $\cup$ '. Thereby, **all best matches with respect to the customer's original preference(s)** will be delivered as well as alternatives with regard to the maximum or minimum sum constraint. Of course, the same holds for the maximum sum constraint.

**Example 4.15 Sample query expansion for a minimum sum constraint**

Again, consider student Derek from above. He expressed, for instance, a preference for a category 3 hotel in the downtown area. Since he does not mind getting a hotel in a higher category as long as the price falls in the range of hotels in category 3, it can be modeled with the  $\text{TRADEOFF}$  preference constructor introduced above:

$$P_H := \text{POS/POS}(\text{Location}, \{\text{'downtown'}\}, \{\text{'city'}\}) \otimes \text{TRADEOFF}(\{\text{Category}, \text{Price}\}, \{\text{'1'}, \text{'2'}, \text{'3'}, \text{'4'}, \text{'5'}\}, \text{'3'})$$

Derek also expressed a soft constraint of \$900 for the total price of the package. Since the summed up attribute - the total price - should be below a certain limit, this represents a minimum sum constraint (Definition 4.14). Therefore, the preference query  $Q_H := \sigma[P_H](H)$  can be expanded as follows (Definition 4.17):

$$Q_{\text{expH}} := Q_H \cup Q_{H^*} \quad \text{where } Q_{H^*} := \sigma[P_H \otimes \text{LOWEST}_0(\text{Price})] (H)$$

A preference search with expanded query  $Q_{\text{expH}}$  on the database of Table 4.9 delivers the BMO result tuples  $t_{H1}$ ,  $t_{H6}$ ,  **$t_{H2}$** , and  **$t_{H7}$** .

While tuples  $t_{H1}$  and  $t_{H6}$  perfectly match Derek's preference for hotels, the bold tuples  $t_{H2}$  and  $t_{H7}$  represent cheaper alternatives with regard to the minimum sum constraint. For example, although tuple  $t_{H2}$  is of category 3, it is not located downtown. However, it is a cheaper alternative to tuples  $t_{H1}$  and  $t_{H6}$ .

Just using preference query  $Q_{H^*} := \sigma[P_H \otimes \text{LOWEST}_0(\text{Price})] (H)$  as described above would not deliver tuple  $t_{H1}$ . But this might be incomprehensible for customers and might even cause problems, since tuple  $t_{H1}$  represents a perfect match.  $\diamond$

### **3. Preference Search**

The result sets  $BMO_i$  are computed by a preference search engine, e.g., the preference XPath search engine, for all expanded queries  $Q_{\text{exp}i}$ . In Example 4.15 the expansion of a preference query  $Q_H$  and the corresponding search result set  $BMO_H$  delivered by a preference search engine were already shown. In order to compute the entire travel package described above, the results of accordingly expanded queries for rental cars ( $BMO_C$ ) and flights ( $BMO_F$ ) would also be needed.

### **4. Quality valuation**

As already mentioned, the quality of search results represents an important factor in the proposed query processing approach. But any discussion about the quality has to take the customer's preferences into account. In this approach, result tuples are only valuated in terms of the customer's original preferences. An expansion as described in Definition 4.16 and Definition 4.17, i.e., a  $\text{LOWEST}_d$  or  $\text{HIGHEST}_d$  preference, will not be included in the computation of the quality. An enhanced BMO relation that includes quality information is therefore defined as follows.

#### **Definition 4.18 Enhanced BMO relations including quality information: $BMO_i^*$**

Assume a travel package  $T := \{T_1, \dots, T_z\}$  with  $z$  parts and a database relation  $R_i$ , a customer's preference  $P_i$ , and an accordingly expanded preference query  $Q_{\text{exp}i}$  (as defined in Definition 4.16 and Definition 4.17) for each  $T_i$ . Assuming a preference search with query  $Q_{\text{exp}i}$  on database relation  $R_i$  leads to result relation  $BMO_i(A_{1i}, \dots, A_{mi})$ , one defines:

- $BMO_i^* := (A_{1i}, \dots, A_{mi}, O_i)$  with  $\text{dom}(O_i) := \{\text{'perfect'}, \text{'very good'}, \text{'good'}, \text{'acceptable'}, \text{'sufficient'}\}$  where  $o \in O_i$  describes the computed quality information for a tuple with respect to preference  $P_i$ .  $\text{D}$

Thereby, the quality of each tuple  $t_{k_i}$  of a result set  $BMO_i$  is computed and expressed in linguistic terms from 'perfect' to 'sufficient' by the *Preference Presenter* component (see also [Fis04, KFD04]).

Note that in contrast to  $BMO^+$  of ([Fis04], Definition 3.4) not all preferences are used to compute the overall quality valuation of result tuples, since this would have included  $HIGHEST_d$  or  $LOWEST_d$  of the expansion. Only the original preferences of the customer, i.e.,  $P_i$  in Definition 4.18, are considered for the valuation. This approach makes sure tuples will be valued in terms of the customer's preferences only.

#### Example 4.16 Sample quality valuation

Consider the result tuples  $t_{H1}$ ,  $t_{H6}$ ,  $t_{H2}$ , and  $t_{H7}$  of Example 4.15 for student Derek. The quality valuation takes only Derek's original preference for hotels into account:

$P_H := POS/POS(\text{Location}, \{\text{'downtown'}\}, \{\text{'city'}\}) \otimes \text{TRADEOFF}(\{\text{Category}, \text{Price}\}, \{\text{'1'}, \text{'2'}, \text{'3'}, \text{'4'}, \text{'5'}\}, \text{'3'})$

This leads to the following  $BMO_H^*$  result relation:

| ID       | Name  | Category | Location | Price<br>(for a week in \$) | O          |
|----------|-------|----------|----------|-----------------------------|------------|
| $t_{H1}$ | IBIS  | 3        | downtown | 595.--                      | perfect    |
| $t_{H2}$ | ARCOR | 3        | city     | 490.--                      | good       |
| $t_{H6}$ | KING  | 4        | downtown | 560.--                      | perfect    |
| $t_{H7}$ | TOKIO | 2        | city     | 420.--                      | sufficient |

**Table 4.11.  $BMO_H^*$  result relation for hotels**

The quality of tuples  $t_{H1}$  and  $t_{H6}$  can be valued 'perfect', since these tuples match Derek's preference perfectly. Note that tuple  $t_{H6}$  with a category of 4 is also valued with 'perfect' due to the definition of the TRADEOFF preference constructor and its quality (see Definitions 4.9 and 4.10). Furthermore, while  $t_{H2}$  can still be valued with 'good' - it is at least of the desired category 3 -  $t_{H7}$  can only be valued with 'sufficient' in terms of Derek's preferences.

◇

### 5. Composition and valuation of travel packages

Now all combinations of tuples of the  $BMO_i^*$  relations are built with the cartesian product, e.g., all combinations of the result tuples for flight, hotel, and rental car. Furthermore, the overall quality of all combinations is determined by the equidistant linguistic average (see [Fis04]) that is intuitively comprehensible by customers. Values of the important  $A_{a_i}$  attribute, which is used for the maximum / minimum sum constraint, are accordingly summed up. Thus, a combined result relation  $BMO^T$  is defined as follows.

**Definition 4.19 Combined result relation:  $\text{BMO}^T$** 

Given a travel package  $T := \{T_1, \dots, T_z\}$  and  $z$  enhanced relations  $\text{BMO}_i^* := (A_{1i}, \dots, A_{mi}, O_i)$ , one defines:

- $\text{BMO}^C := \text{BMO}_1^* \times \dots \times \text{BMO}_z^* = (A_{11}, \dots, A_{m1}, O_1, \dots, A_{1z}, \dots, A_{mz}, O_z)$  is a temporary relation which represents the cartesian product of the  $\text{BMO}_i^*$  relations for  $i = \{1, \dots, z\}$ .
- $\text{BMO}^T := (A_{11}, \dots, A_{m1}, O_1, \dots, A_{1z}, \dots, A_{mz}, O_z, S^\#, O^\#)$  enhances  $\text{BMO}^C$  by the aggregated attributes  $O^\#$  and  $S^\#$  where  $O^\#$  represents the overall quality of a combination with individual qualities  $O_i$  and  $i = \{1, \dots, z\}$  determined by the equidistant linguistic average (see Definition 3.25 in [Fis04]) and  $S^\#$  represents the sum of all  $A_{ai}$  with  $i = \{1, \dots, z\}$  for a maximum / minimum sum constraint.  $\text{D}$

Note that the overall quality is determined by the equidistant linguistic average as defined in [Fis04], since it can be applied to most situations and customers. However, if a customer or situation requires another strategy such as the *optimistic* or *pessimistic valuation* ([Fis04]), it could be integrated into the query processing approach at this stage as well. Moreover, if the domain of the aggregated attribute  $S^\#$  is known, the aggregation can be adapted accordingly. For instance, assume that  $S^\#$  should represent the total price of a travel package. In that case, it would be reasonable to assume  $S^\#$  represents a personalized price offer including discounts provided by an appropriate component (see [Fis04]) instead of a plain sum of individual prices. Such domain and background knowledge can be integrated seamlessly at this stage of the query processing.

**Example 4.17 Sample  $\text{BMO}^T$  relation**

Imagine a travel package consisting of flight (F), hotel (H), and car (C). Further assume the following result tuples of the  $\text{BMO}_i^*$  relations are already determined for a traveler Mark:

$$\text{BMO}_F^* := \{(t_{F1}, 200, \dots, \text{perfect}), (t_{F2}, 100, \dots, \text{acceptable})\}$$

$$\text{BMO}_H^* := \{(t_{H3}, 200, \dots, \text{very good})\}$$

$$\text{BMO}_C^* := \{(t_{C1}, 200, \dots, \text{good}), (t_{C2}, 50, \dots, \text{sufficient})\}$$

Note that attribute values relevant for the example are shown for better illustration only. A projection on relevant attributes of the  $\text{BMO}^T$  relation is shown in Table 4.12.

| ID <sub>F</sub> | ID <sub>H</sub> | ID <sub>C</sub> | S <sup>#</sup> | O <sup>#</sup> |
|-----------------|-----------------|-----------------|----------------|----------------|
| t <sub>F1</sub> | t <sub>H3</sub> | t <sub>C1</sub> | 600            | very good      |
| t <sub>F2</sub> | t <sub>H3</sub> | t <sub>C1</sub> | 500            | good           |
| t <sub>F1</sub> | t <sub>H3</sub> | t <sub>C2</sub> | 450            | good           |
| t <sub>F2</sub> | t <sub>H3</sub> | t <sub>C2</sub> | 350            | acceptable     |

**Table 4.12. Sample BMO<sup>T</sup> result relation for travel package**

The summed up numerical attribute S<sup>#</sup> could, for example, represent the total price of the package, while O<sup>#</sup> represents the overall quality of a combination.  $\diamond$

## 6. Offer selection

In order to select the best travel packages with respect to the customer's constraints, both the overall quality and the summed up attribute value of combinations in BMO<sub>T</sub><sup>\*</sup> have to be considered. First and irrespective of the sum constraint, it is reasonable to assume that a customer is interested in getting a package of the best overall quality. Therefore, the following preference, which considers the overall quality of a combination in BMO<sup>T</sup>, is defined.

### Definition 4.20 Quality preference for offer selection: P<sub>q</sub>

Given a BMO<sup>T</sup> relation with an aggregated attribute O<sup>#</sup> for the overall quality as introduced in Definition 4.19:

- $P_q := (O^\#, \langle_{P_q}, \cong_{P_q}) = \text{LAYERED}_5(O^\#, (\{\text{'perfect'}\}, \{\text{'very good'}\}, \{\text{'good'}\}, \{\text{'acceptable'}\}, \{\text{'sufficient'}\}))$   $\rhd$

Thereby, a combination of travel parts with an overall quality of 'perfect' will be preferred over a combination denoted 'very good', which will still be preferred over a combination with 'good' quality and so on.

Up to now, the maximum / minimum sum constraints, i.e., the global preferences, were only considered in terms of the query expansion for individual parts of the query (Definition 4.16, Definition 4.17). However, the selection of the best matching travel packages requires deeper knowledge about the customer's sum constraint. With respect to a customer's wishes, the maximum / minimum sum constraint can be modeled either as a soft or as a hard constraint. Treating it as a hard constraint, the following query would select the best travel package(s) with respect to the customer's preference for a high overall quality P<sub>q</sub> and his or her hard maximum / minimum sum constraint.

### Definition 4.21 Offer selection query for hard sum constraint: Q<sub>sh</sub>

Assume the BMO<sup>T</sup> relation contains all relevant combinations of travel parts and the aggregated attributes O<sup>#</sup> and S<sup>#</sup> (Definition 4.19). Given a preference  $P_q := (O^\#, \langle_{P_q}, \cong_{P_q})$  for the

best overall quality as defined in Definition 4.20 and the customers hard constraint, one defines:

- $Q_{Sh} := \sigma[P_q](\sigma_{S^\#} \Theta_K(BMO^T))$  where  $\Theta$  represents a typical operator for numerical attributes such as '<', '>', '≤', or '≥' and  $K$  represents the customer's upper or lower limit for the aggregated attribute  $S^\#$ .  $\mathcal{D}$

First, the customer's hard constraint is applied on the summed up attribute  $S^\#$  of relation  $BMO^T$ . Afterwards, by using the quality preference  $P_q$  those combinations with the best quality are selected from the remaining relation.

However, treating the customer's maximum / minimum constraint as a soft constraint, i.e., as a preference, requires an appropriate preference modeling. Customers often have lower or upper limits, e.g., the price should not exceed a certain limit. Such preferences can be modeled with the  $LESS\_THAN_d$  and  $MORE\_THAN_d$  preference constructors. They represent special cases of the  $BETWEEN_d$  constructor (Definition 3.6), which can be obtained by setting *low* or *up* of distance function  $dist_d[low, up]$  to the *infimum* or *supremum* of  $dom(A)$ .

**Definition 4.22  $MORE\_THAN_d$  and  $LESS\_THAN_d$**

Given  $x, y, z \in dom(A)$  and  $inf_A$  the infimum of  $dom(A)$ :

- base  $LESS\_THAN_d(A, z) \{x <_p y \text{ iff } dist_d[inf_A, z](y) < dist_d[inf_A, z](x)\}$

Moreover, given  $z, x, y \in dom(A)$  and  $sup_A$  the supremum of  $dom(A)$ :

- base  $MORE\_THAN_d(A, z) \{x <_p y \text{ iff } dist_d[z, sup_A](y) < dist_d[z, sup_A](x)\}$   $\mathcal{D}$

While the preference constructors  $HIGHEST_d$  and  $MORE\_THAN_d$  can be used for the modeling of typical maximum sum constraints, e.g., for a customer who would like to get the biggest or at least a certain amount of frequent flier miles,  $LOWEST_d$  and  $LESS\_THAN_d$  can be applied for minimum sum constraints. The relation between the preference for quality and the preference for the maximum / minimum sum constraint can be modeled with complex preference constructors like pareto or prioritized.

**Example 4.18 Sample preference modeling for offer selection**

Assume a customer would like to have as much frequent flier miles as possible for the entire travel package. All individual queries were expanded and treated as described above. Miles are represented by the aggregated attribute  $S^\#$  within the  $BMO^T$  relation. His or her preference can, therefore, be modeled with  $P_a = (S^\#, <_{P_a}, \cong_{P_a}) := HIGHEST_d(S^\#)$ . Moreover, he or she is interested in getting the best results with respect to his or her individual preferences for travel parts, e.g., for a certain hotel or airline. This is modeled with  $P_q = (O^\#, <_{P_q}, \cong_{P_q})$  on the

aggregated attribute for quality  $O^\#$  (Definition 4.20). Since both preferences are equally important to him or her, a preference for offer selection can be modeled as:  $P_S := P_a \otimes P_q$ .  $\diamond$

**Definition 4.23 Offer selection query for soft sum constraint:  $Q_{S_s}$**

Given a  $BMO^T$  relation containing all relevant combinations of travel parts and the aggregated attributes  $O^\#$  and  $S^\#$  (Definition 4.19). Furthermore, given a preference  $P_S$  on  $S^\#$  and  $O^\#$ , one defines:

$$\bullet Q_{S_s} := \sigma[P_S](BMO^T) \quad \triangleright$$

**Example 4.19 Sample queries for offer selection**

Assume Table 4.12 represents the  $BMO^T$  relation, i.e., it contains all relevant combinations of travel parts with respect to the customer's individual preferences and his or her maximum / minimum sum constraint as described before. Furthermore,  $S^\#$  represents the summed up prices of individual travel parts and  $O^\#$  represents the overall quality of a package. The total price of a package *should/must not exceed \$500* in terms of the customer's wishes. Of course, the customer is interested in travel packages which have the highest overall quality, i.e., which match his or her preferences for the individual travel parts best. This is modeled as preference  $P_q$  (Definition 4.20).

First, his or her minimum sum constraint for the total price is considered as hard constraint. This leads to query:  $Q_{S_h} := \sigma[P_q](\sigma_{S^\# \leq 500}(BMO^T))$

Thereby, the two combinations with a quality valuation of 'good' and a price below \$500 can be presented to the customer.

Secondly, the minimum sum constraint for the total price is considered as a soft constraint. Assume both preferences for the total price and the best quality are equally important to the customer. This leads to preference  $P_S := P_a \otimes P_q$  with  $P_a = (S^\#, <_{P_a}, \cong_{P_a}) := LESS\_THAN_o(500)$  and query:  $Q_{S_s} := \sigma[P_S](BMO^T)$

Now in addition to the combinations with 'good' quality, also the combination with 'very good' overall quality can be presented. This combination does not match the customer's preference for the total price, but offers a better overall quality with respect to the individual preferences.  $\diamond$

Thereby, typical maximum and minimum sum constraints in tourism-related electronic commerce scenarios can be dealt with. They can be flexibly modeled and combined with individual preferences for travel parts. In other words, travel packages exactly tailored to the customer can be offered easily and quickly, using the introduced preference query processing approach. In the following, the entire approach will be illustrated by means of a complete sample scenario.



**Example 4.20 Complete query processing approach for a typical minimum sum constraint in electronic commerce: the total price**

Remember Derek's preferences from the beginning of section 4.5:

1. The flight's airline **should** be Lufthansa and the seat **should** be in business class.
2. His hotel **should** have 3 stars and **should** be located downtown.
3. A rental car will also be needed, since he has to be flexible. The brand **should** be Renault.
4. The total package **should not** cost more than \$900.

Note that Derek would regard an 'economy' flight as an alternative, if there is no suitable flight in the 'business' class. Moreover, if there is no matching hotel located in 'downtown', he would consider a hotel in the 'city' area as being better as a hotel in the 'vicinity'. He would not mind getting a qualitatively better hotel as long as the price does not exceed the price range of hotels with a category of 3 stars. After the appropriate specification of Derek's preferences on the search form, the query processing approach proceeds as follows.

The databases are represented by Table 4.7 for flights (F), Table 4.8 for rental cars (C), and Table 4.9 for hotels (H) on page 77.

1. Query composition:

Derek's individual preferences can again be modeled with the following preference constructors:

- $P_F := \text{POS/POS}(\text{Category}, \{\text{'business'}\}, \{\text{'economy'}\}) \otimes \text{POS}(\text{Airline}, \{\text{'Lufthansa'}\})$
- $P_H := \text{POS/POS}(\text{Location}, \{\text{'downtown'}\}, \{\text{'city'}\}) \otimes \text{TRADEOFF}(\{\text{Category}, \text{Price}\}, \{\text{'1'}, \text{'2'}, \text{'3'}, \text{'4'}, \text{'5'}\}, \text{'3'})$
- $P_C := \text{POS}(\text{Brand}, \{\text{'Renault'}\})$

The corresponding preference queries are:

- FLIGHT:  $Q_F := \sigma[P_F](F)$
- HOTEL:  $Q_H := \sigma[P_H](H)$
- RENTAL CAR:  $Q_C := \sigma[P_C](C)$

2. Query expansion

The price limit of Derek represents a minimum sum constraint. Therefore, the queries are expanded as follows (Definition 4.18):

- FLIGHT:  $Q_{\text{expF}} := \sigma[P_F](F) \cup \sigma[P_F \otimes \text{LOWEST}_0(\text{Price})](F)$

- HOTEL:  $Q_{\text{expH}} := \sigma[\mathbf{P_H}] (\mathbf{H}) \cup \sigma[\mathbf{P_H} \otimes \text{LOWEST}_0(\text{Price})] (\mathbf{H})$
- RENTAL CAR:  $Q_{\text{expC}} := \sigma[\mathbf{P_C}] (\mathbf{C}) \cup \sigma[\mathbf{P_C} \otimes \text{LOWEST}_0(\text{Price})] (\mathbf{C})$

### 3. Preference search

The following tuples are delivered by the preference search engine:

- FLIGHT:  $\{t_{F1}, t_{F4}, \mathbf{t}_{F3}\}$
- HOTEL:  $\{t_{H1}, t_{H6}, \mathbf{t}_{H2}, \mathbf{t}_{H7}\}$
- RENTAL CAR:  $\{t_{C2}, \mathbf{t}_{C4}\}$

Note that the bold tuples  $t_{F3}$ ,  $t_{H2}$ ,  $t_{H7}$ , and  $t_{C4}$  are only included in the corresponding result sets because of the search queries' expansion.

### 4. Quality valuation

Now the quality of each result tuple is computed and expressed in linguistic terms from *perfect* to *sufficient* by the *Preference Presenter* component. The expanded  $\text{LOWEST}_d(\text{Price})$  preference will not be included into the computation. This approach makes sure tuples will be valued in terms of the customer's preferences only.

- **perfect:**  $t_{H1}$ ,  $t_{H6}$ , and  $t_{C2}$
- **good:**  $t_{H2}$ ,  $t_{F1}$ , and  $t_{F4}$
- **sufficient:**  $t_{H7}$ ,  $t_{F3}$ , and  $t_{C4}$

The quality of tuples  $t_{H7}$ ,  $t_{F3}$ , and  $t_{C4}$  is only sufficient with respect to Derek's preferences. Tuple  $t_{F3}$ , for instance, does not match Derek's preferences for Lufthansa or business class. Still, including them is quite reasonable due to the fact that they represent alternative products with a lower price. Tuples  $t_{H1}$ ,  $t_{H6}$ , and  $t_{C2}$  match Derek's preferences perfectly and are valued accordingly. Some tuples are valued with 'good' because they match the preferences to a certain degree, e.g.,  $t_{F1}$  matches the preferred airline only.

The appropriate  $\text{BMO}_H^*$  result relation for hotels is shown in Table 4.11. The BMO relations for flights and rental cars can be extended by a quality valuation  $O$  analogously.

### 5. Composition and valuation of travel packages

All combinations are built with the cartesian product, i.e., all combinations of the result tuples for flight, hotel, and rental car. In addition, the aggregated attributes for the overall quality and the total price are added. For a better illustration, only a projection on relevant attributes of the  $\text{BMO}^T$  relation is shown in Table 4.13.

| Combination | O#         | S#      | Flight          | Rental Car      | Hotel           |
|-------------|------------|---------|-----------------|-----------------|-----------------|
| 1           | perfect    | 1295.-- | t <sub>F1</sub> | t <sub>H1</sub> | t <sub>C2</sub> |
| 2           | perfect    | 1250.-- | t <sub>F1</sub> | t <sub>H6</sub> | t <sub>C2</sub> |
| 3           | very good  | 1585.-- | t <sub>F4</sub> | t <sub>H1</sub> | t <sub>C2</sub> |
| 4           | very good  | 1550.-- | t <sub>F4</sub> | t <sub>H6</sub> | t <sub>C2</sub> |
| 5           | very good  | 1480.-- | t <sub>F4</sub> | t <sub>H2</sub> | t <sub>C2</sub> |
| 6           | very good  | 1410.-- | t <sub>F4</sub> | t <sub>H7</sub> | t <sub>C2</sub> |
| 7           | very good  | 1180.-- | t <sub>F1</sub> | t <sub>H2</sub> | t <sub>C2</sub> |
| 8           | very good  | 1110.-- | t <sub>F1</sub> | t <sub>H7</sub> | t <sub>C2</sub> |
| 9           | very good  | 1085.-- | t <sub>F3</sub> | t <sub>H1</sub> | t <sub>C2</sub> |
| 10          | very good  | 1050.-- | t <sub>F3</sub> | t <sub>H6</sub> | t <sub>C2</sub> |
| 11          | very good  | 980.--  | t <sub>F3</sub> | t <sub>H2</sub> | t <sub>C2</sub> |
| 12          | good       | 1470.-- | t <sub>F4</sub> | t <sub>H1</sub> | t <sub>C4</sub> |
| 13          | good       | 1435.-- | t <sub>F4</sub> | t <sub>H6</sub> | t <sub>C4</sub> |
| 14          | good       | 1365.-- | t <sub>F4</sub> | t <sub>H2</sub> | t <sub>C4</sub> |
| 15          | good       | 1170.-- | t <sub>F1</sub> | t <sub>H1</sub> | t <sub>C4</sub> |
| 16          | good       | 1135.-- | t <sub>F1</sub> | t <sub>H6</sub> | t <sub>C4</sub> |
| 17          | good       | 1065.-- | t <sub>F1</sub> | t <sub>H2</sub> | t <sub>C4</sub> |
| 18          | good       | 995.--  | t <sub>F1</sub> | t <sub>H7</sub> | t <sub>C4</sub> |
| 19          | good       | 970.--  | t <sub>F3</sub> | t <sub>H1</sub> | t <sub>C4</sub> |
| 20          | good       | 935.--  | t <sub>F3</sub> | t <sub>H6</sub> | t <sub>C4</sub> |
| 21          | good       | 910.--  | t <sub>F3</sub> | t <sub>H7</sub> | t <sub>C2</sub> |
| 22          | acceptable | 1295.-- | t <sub>F4</sub> | t <sub>H7</sub> | t <sub>C4</sub> |
| 23          | acceptable | 865.--  | t <sub>F3</sub> | t <sub>H2</sub> | t <sub>C4</sub> |
| 24          | acceptable | 795.--  | t <sub>F3</sub> | t <sub>H7</sub> | t <sub>C4</sub> |

Table 4.13. BMO<sup>T</sup> result relation for example

### 6. Offer selection

Since Derek's constraint for the total price was expressed with the term 'should', it represents a soft constraint. The preference for the total price and the preferences for the individual parts of the travel are equally important to him. Therefore, his preference can be modeled as:

$P_S := P_a \otimes P_q$  with

$P_a = (S^\#, <_{P_a}, \cong_{P_a}) := \text{LESS\_THAN}_0(900)$  and  $P_q = (O^\#, <_{P_q}, \cong_{P_q}) := \text{LAYERED}_5(O^\#, (\{\text{'perfect'}\}, \{\text{'very good'}\}, \{\text{'good'}\}, \{\text{'acceptable'}\}, \{\text{'sufficient'}\}))$

This leads to the following query:  $Q_{S_S} := \sigma[P_S] (\text{BMO}^T)$

Thereby, combinations 2, 11, 21, 23, and 24 are delivered. While combinations 23 and 24 match the preference for the total price, combination 2 perfectly matches the preferences for the individual parts of the travel. Combinations 11 and 21 represent compromises somewhere in between.  $\diamond$

Note that in contrast to 'Demonstration Scenario 2 - Soft Constraints' on page 79, also **alternative travel packages with respect to Derek's global price constraint** can be delivered. Thereby, Derek is able to choose between best matching packages with respect to both his individual preferences for parts of the travel and his global preference for the total price. Thus, a better support for customers of tourism-related electronic commerce can be provided.

#### 4.5.2 Algorithm and Complexity

The following compact algorithm represents the computation sequence of the workflow described in the subsection before. Best matching travel packages can be determined for a customer with respect to his or her individual preferences for travel parts as well as his or her maximum / minimum sum constraint for the entire travel package. Afterwards the complexity of the algorithm will be examined.

Note that the index  $i$  corresponds with a travel part  $T_i$  of travel package  $T := \{T_1, \dots, T_z\}$ , i.e.,  $Q_i$  is a query on relation  $R_i$  for travel part  $T_i$ . This can, for instance, be a preference query for flights on the corresponding database relation.

**INPUT:** preferences  $P_i$  and corresponding queries  $Q_i$ , relations  $R_i$ , numerical attribute  $A_i$  contained in each relation  $R_i$

```

1: for (i = 1, i <= z) do
2:   if (maximum_sum_constraint) then
3:      $Q_{exp_i} = \text{expandQueryMaximum}(Q_i, A_i)$ ; // Definition 4.17
4:   else if (minimum_sum_constraint) then
5:      $Q_{exp_i} = \text{expandQueryMinimum}(Q_i, A_i)$ ; // Definition 4.18
6:    $BMO_i = \text{preferenceSearch}(Q_{exp_i}, R_i)$ ;
7:    $BMO_i^* = \text{denoteQuality}(BMO_i, P_i)$ ; // Definition 4.19
8:   end for
9:   for (i = 1, i <= z) do
10:     $BMO^C = BMO^C.\text{buildCartesianProduct}(BMO_i^*)$ ; // Definition 4.20
11:   end for
12:    $BMO^T = \text{addAggregatedAttributes}(BMO^C)$ ; // Definition 4.20
13:    $Q_s = \text{composeSelectionQuery}()$ ; // Definition 4.21 or 4.23
14:    $BMO = \text{preferenceSearch}(Q_s, BMO^T)$ ; //

```

**OUTPUT:** BMO result

In a loop on all travel parts (line 1), it will first be checked whether the customer's constraint represents a maximum or minimum sum constraint (line 2 and 4). Then the queries for all travel parts will be expanded accordingly (line 3 or 5) and used to determine the search results (line 6). Finally, in line 7 the search result relation of each travel part is expanded by a quality valuation for each tuple. In the following, all relevant combinations of travel parts will be built by the cartesian product on the result relations. Average quality and the aggregated attribute are computed thereafter. Finally, a corresponding search query determines the combinations of travel parts which should be presented to the customer with respect to his or her individual and global preferences.

The workflow for the preference query processing was already illustrated in Example 4.20. In the following, the complexity of this approach will be examined.

#### Lemma 4.5

Let  $z$  be the number of travel parts  $T_i$  with corresponding database relations  $R_i$ . For the sake of simplification, given  $n$ ,  $k$ , and  $m$  where:

- $n$  is the maximum number of tuples a relations  $R_i$  contains, i.e.,  $n = \max(|R_1|, \dots, |R_z|)$ ,
- $k$  is the maximum number of preferences which are accumulated in a preference  $P_i$  for travel part  $T_i$ ,
- $m$  is the maximum number of tuples an intermediate result relation  $BMO_i$  contains, i.e.,  $n = \max(|BMO_1|, \dots, |BMO_z|)$ .

Then the complexity for computation of the BMO result relation is in the worst case:

$$O(z \cdot (m^z + m \cdot k + n)). \quad \text{D}$$

#### Proof 4.5

Lemma 4.5 holds due to the construction of the algorithm (Definitions 4.17-4.23):

1. For the computation of the expanded queries  $Q_{\text{exp}_i}$  only one pass through all preference queries  $Q_i$  is necessary. This can be done in  $O(z)$ .
2. Determining the BMO result relation  $BMO_i$  for an expanded preference query  $Q_{\text{exp}_i}$  on relation  $R_i$  can be done with linear complexity  $O(|R_i|)$  for typical e-commerce scenarios using novel algorithms such as defined in ([PK07, MPJ07]).

The overall complexity is  $O(z \cdot n)$ , since  $BMO_i$  result relations for all  $z$  travel parts  $T_i$  have to be computed and  $|R_i|$  is considered as  $n$  for simplification as described above.

3. The computation of one tuple's quality depends only on the number of involved preferences and can be done in  $O(k)$  (see [Fis04]). Thus, the computation of all tuples' qualities ( $BMO_i^*$ ) results in a complexity  $O(z \cdot m \cdot k)$ .

4. Building the cartesian product of  $BMO_i^*$  relations leads to a complexity  $O(m^z)$ . Adding the aggregated attributes  $S^\#$  and  $O^\#$  can be done simultaneously. Some more basic operations, i.e., additions and one division, are necessary for each tuple. The amount of operations depends on  $z$ . Thus, the overall complexity is  $O(z \cdot m^z)$ .
5. For an offer selection, a last preference search is necessary on the cartesian product represented by  $BMO^T$ . This can again be done by a novel algorithm such as Hexagon ([PK07]). Since  $BMO^T$  has a size of  $m^z$ , this leads to a complexity  $O(m^z)$ .

Thereby, the total worst case complexity is  $O(z \cdot (m^z + m \cdot k + n))$ . □

Note that the computation of the intermediate  $BMO_i$  result relations and their quality can be done simultaneously, which can significantly reduce the computation time. Usually, there are no more than 3 parts in tourism-related electronic commerce, i.e., flight, hotel, and rental car, which results in a complexity of  $O(m^3 + m \cdot k + n)$ . Obviously, the size  $m$  of the  $BMO_i$  result relations represents an important factor for the complexity. Although  $BMO_i$  result relations of typical preference queries are usually much smaller than the corresponding relation  $R$ , they should deserve highest attention. These relations should be reduced as much as possible using an appropriate preference modeling with advanced instruments such as SV-relations and d-parameter ([Kie05]). Furthermore, in [DEP05] a heuristic approach for the adaptation of numerical base preferences was presented, which can be used to influence the result set size.

## 4.6 The Concepts in Retrospect

In this chapter, novel concepts for a deeply personalized search process in tourism were introduced. At first, a novel search model was presented which takes the preferences and situations of customers during all stages of the search process into account. A situation model for tourism as well as a framework for the integration of such situational knowledge into the search process were presented afterwards. For the first stage of the search model, a new intricate preference constructor was presented which takes the typical price-quality-tradeoff into account. A smart preference elicitation based on the integration of heterogeneous information sources may support the customer in specifying suitable preferences. Finally, an advanced preference query processing approach which considers individual as well as global preferences of customers was presented. Hence, by combination of all these concepts and models, an advanced and deeply personalized search process is possible that takes the complex preferences and situations of travelers into consideration. Thus, customers receive exactly those travel packages which match their complex preferences and situations.

## 5 Preference Based Components for Tourism

In this chapter, the interplay of existing preference technologies with new software components - based on the novel concepts and technologies introduced above - will be described. First, the history of the project COSIMA and related preference components will be outlined briefly. Secondly, novel components for a personalized search in tourism will be described. Then the interplay of all those pieces will be illustrated. Finally, the aspects and advantages of the components' interplay will be demonstrated by means of several use case scenarios.

### 5.1 History of COSIMA

Project *COSIMA*<sup>7</sup> was initiated at the Chair for Databases and Information Systems at the University of Augsburg in 2000. It focuses on the development of more human-like, personalized and intuitive sales applications. Already in its first year, a prototype for comparison shopping ([FHK+01]) incorporating a female human-like avatar and natural speech synthesis was presented to a huge audience at the SYSTEMS<sup>8</sup> computer fair. In 2002 a component for multi-objective bargaining processes ([FKH+02]) was presented at CEBIT<sup>9</sup>.

Later that year foundations for preferences in database systems were laid by Kießling ([Kie02, KK02]). This intricate work influenced further research within the COSIMA project. Moreover, in 2002 COSIMA also became part of the *Bavarian Research Cooperation for Situated, Individualized, and Personalized Human-Computer Interaction* (FORSIP, [For07]) funded by the Bavarian State Ministry for Science, Research and Art in Germany from 2002 to 2007.

In the scope of FORSIP, COSIMA<sup>B2B</sup> a fully automated sales agent for e-procurement was developed ([KFD04]). It represents a typical B2B use case scenario and was modeled in cooperation with our industrial partners SSI Schäfer<sup>10</sup> and MAN Roland Druckmaschinen AG<sup>11</sup>

---

7 [www.mycosima.com](http://www.mycosima.com)

8 [www.systems-world.com](http://www.systems-world.com)

9 [www.cebit.com](http://www.cebit.com)

10 [www.ssi-schaefer.de](http://www.ssi-schaefer.de)

11 [www.man-roland.de/en](http://www.man-roland.de/en)

(see Figure 5.1). In 2004 COSIMA<sup>B2B</sup> was exhibited at e\_procure<sup>12</sup>, a fair for electronic procurement and supplier management. As a special use case, COSIMA<sup>B2B</sup> was furthermore equipped with natural language input via keyboard or microphone, emotion recognition via mimic, and an embodied character agent communicating via natural speech ([FDW+04]). This particular system was exhibited at the SYSTEMS 2003 computer fair in Munich.

The screenshot displays the COSIMA B2B web interface. At the top, there is a blue header with the text "COSIMA B2B". Below this is a navigation menu with links: Welcome, Search, Results, Basket, History, Open Offers, My Offer, Master Data, and Logout. The main content area shows search results in a table format:

| Article | Details  | Quantity | List Price |
|---------|--|----------|------------|
| 91954   | LF 221 ZW PE, 230/200X140X121MM BLUE<br>STORAGE CONTAINER MADE OF POLYETHYLENE<br>• Material: Polyethylene • Length: 230.0 mm • Height: 121.0 mm • Volume: 2.7 l<br>• Color: blue • Width: 140.0 mm • Weight: 0.3 kg | 1        | 3,16 €     |
| 92044   | LF 321 ZW PE, 350/300X200X145MM BLUE<br>STORAGE CONTAINER MADE OF POLYETHYLENE<br>• Material: Polyethylene • Length: 350.0 mm • Height: 145.0 mm • Volume: 7.5 l<br>• Color: blue • Width: 200.0 mm • Weight: 0.5 kg | 1        | 5,85 €     |

Below the table are buttons for "Back to Search" and "Add to Basket". A chat window at the bottom contains the message: "COSIMA: I found exactly the product you searched for. I was able to fulfill your wishes regarding color and material. This is just the perfect product for you." To the right of the chat window is a virtual agent character. At the bottom of the page, there are logos for "SCHAFFER", "MAN ROLAND", "WALCH", "It's a Preference World", and "Forsip".

Figure 5.1. COSIMA B2B – the fully automated sales agent ([KFD04])

COSIMA<sup>B2B</sup> included a personalized Preference Search Engine, the Preference Presenter implementing a sales psychology based presentation of search results, the Preference Repository responsible for the management of situated long-term preferences, the flexible Price Generation framework, the multi-objective Preference Bargainer, and the Personalization Manager providing an intuitive interface to adapt various personalization parameters driving the whole sales process (see Figure 5.2).

In two parallel projects called P-News ([WBK+04, HHK05]) and Preference Mining ([HEK03]), some of these components were also deployed. By using the personalized news dissemination system P-News, users can be adequately informed about relevant and new information in their specific area of interest. Preference Mining means the detection of preferences in user log data ([HEK03]). Once identified, preferences can also be used for enhanced product recommendations ([SEK06]).

The most important preference based components are illustrated in Figure 5.2 below.

<sup>12</sup> www.e-procure.de



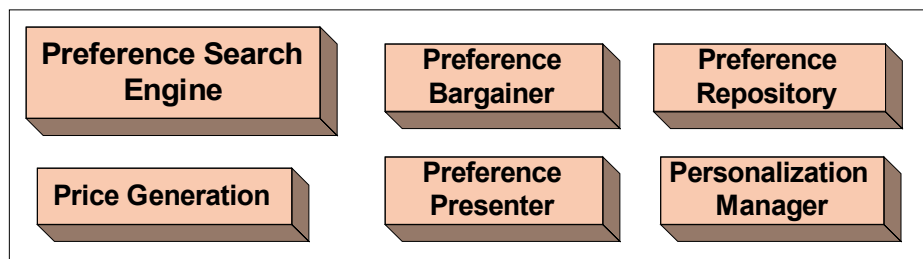


Figure 5.2. Preference based components in overview

## 5.2 Novel Components for a Deeply Personalized Search in Tourism

Related to this thesis, several novel components for personalized search processes in tourism have come into existence. These components will be presented in the following.

### 5.2.1 Airport Finder and Travel Recommender

As already stated above, customers sometimes are not able to specify preferences required by travel portals, for instance, a desired airport. In order to support customers avoiding such form-driven preferences, the concept of a smart preference elicitation based on the integration of heterogeneous data sources is presented in section 4.4. In the following, two appropriate components for typical use case scenarios are described.

#### Airport Finder:

As demonstrated by means of numerous examples, finding the right airport with respect to the underlying preferences, e.g., time and costs, is not as easy as it should be. There are almost countless information sources that can be taken into consideration for this purpose: route finders, websites of railway companies, airports' websites etc. All of these data sources can be relevant for customers to find the airport the travel portal asks for. Even if a customer knows all the relevant websites, it is a tedious and lengthy process to manually collect and evaluate these data.

Because of this, a component was developed that automatically integrates these data sources from the internet ([Gri07]). Thereby, customers are supported in finding exactly those airports matching their underlying preferences for their particular kind of journey (see Figure 5.3). The component is based on the concept described in scenario A (see page 68). That is, a BMO result set will be delivered by a corresponding *webservice* with respect to the customer's preferences for the journey. Thereby, only little integration effort is necessary to couple this service to a travel portal. It is possible to specify preferences for the costs, the length of time, the kind of transport etc. Furthermore, base preferences can be accumulated by a pareto or prioritized preference.

For example, a customer might specify that he or she would like to fly to Tokyo. Furthermore, assume the customer prefers an airport that he or she can reach as quickly as possible. By using the described component, all relevant airports can be determined and integrated into the further search process by the travel portal. Thereby, a form-driven preference can be avoided, the customer can be supported, and the travel portal may possibly sell one more travel.

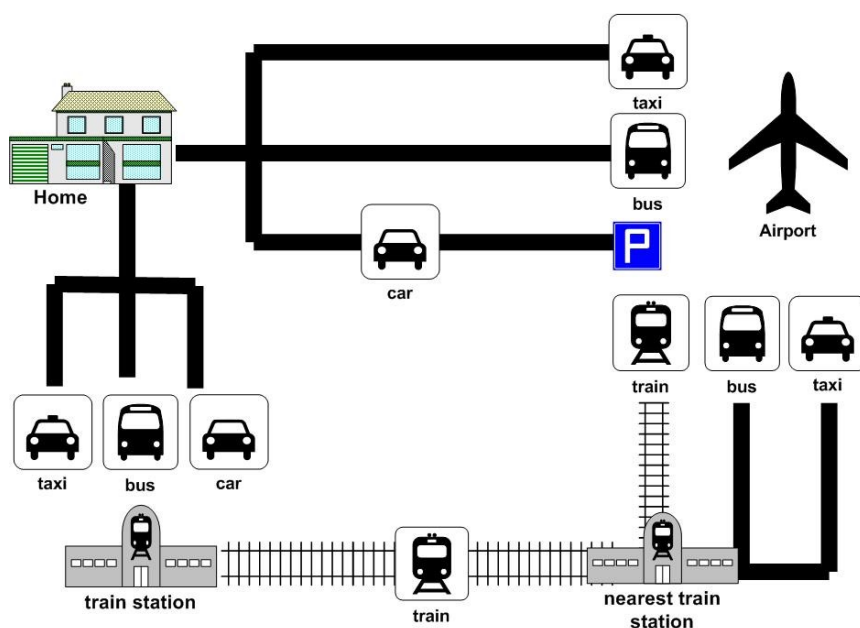


Figure 5.3. Smart preference elicitation to find a suitable airport ([Gri07])

The great number of potential preferences for a journey to the airport is illustrated in Figure A.1. and A.2. (appendix on page 135). Note that these screenshots are only meant to illustrate the amount of preferences, i.e., the actual interface to a customer has to be constructed carefully and personalized by a component as described in subsection 5.2.2.

#### Travel Recommender:

Sometimes customers are unsure about the destination of their travel ([PM00]). They might have some ideas about the travel itself. For instance, they know that they would like to make a language trip to an English-speaking country, yet sometimes they do not know which destination will best suit them with respect to their preferences.

By the novel travel recommender component of [Zie07], some relevant information sources can be integrated. Wikipedia<sup>13</sup> can provide information about a country such as the currency, the capitol or its language; the German Department for Foreign Affairs<sup>14</sup> can supply up-to-

13 [www.wikipedia.org](http://www.wikipedia.org)

14 [www.auswaertiges-amt.de](http://www.auswaertiges-amt.de)

date information about the security situation in those countries; an online weather service<sup>15</sup> can provide corresponding weather data. This component is also based on the concept described in scenario A (see page 68). That is, a customer may specify some preferences about the desired country, e.g., the language. Thereafter, a list of best-matching destinations can be delivered. Furthermore, this component can be used to integrate destination-specific information into the last stage of the search process, i.e., the product presentation. If the customer obtains travel packages for his or her vacation in Afghanistan, for example, some hints on his or her security can be given. In conclusion, customers can be supported in their decision for a destination and valuable information can be integrated into the presentation stage of the search process.

### 5.2.2 Personalization of the Search Interface: Visualization Component

According to the search model of Definition 4.5, the search interface has to be considered as a single and important piece of the search process. Therefore, a component for the visualization of preferences on personalized search interfaces was constructed ([Haa07]). The component differentiates between visual representations and the target domain itself, i.e., the preferences (see Figure 5.4).

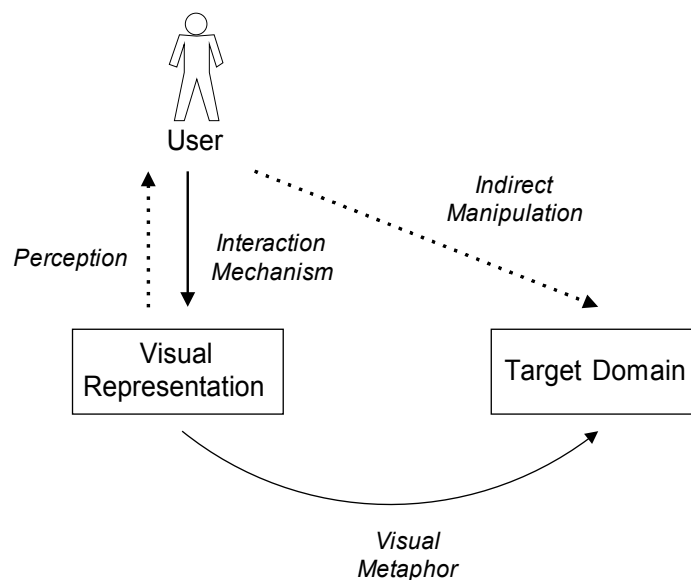


Figure 5.4. Visualization based on metaphors ([Haa07])

For a maximum of flexibility a *Visualization Factory* is used, i.e., different visualizations are possible for one preference (n:1 mapping). Hence, the visual representation of a preference can be adjusted to the customer's preferences and situations, offering a deeply personalized search interface. While the big screenshot of Figure A.3. (appendix on page 137) represents a visualization for the tourism domain, Figure 5.5 represents a generic visualization that can be applied in general. This way, the visualization component can be adjusted to the domain as

15 [www.weather.com](http://www.weather.com)

well as to the customer, thereby offering a novel kind of personalization and situation awareness for this stage of the search process.

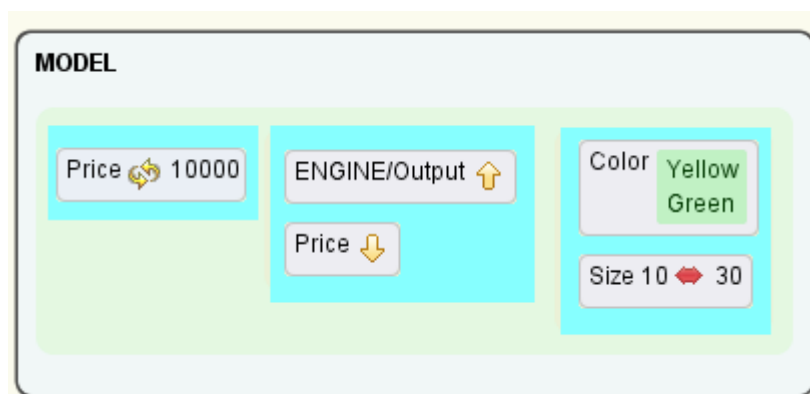


Figure 5.5. Generic visualization approach ([Haa07])

### 5.2.3 Advanced Preference Query Processing: COSIMA<sup>T</sup>

Even in simple travel scenarios, there are preferences for individual aspects of the journey and there are preferences concerning global constraints, e.g., for the total price of the travel package (see also Figure 2.1 / Definition 4.4). With COSIMA<sup>T</sup> a prototype has come into existence ([Dör06, DPE08]) which incorporates a component for the advanced query processing described in section 4.5. Thereby, the customers' individual preferences as well as a global preference for the total price of a travel package can be taken into consideration.

If the customer presses the search button, the advanced query processing approach is performed and will deliver deeply personalized results. Both best results with respect to preferences for individual parts of the travel, e.g., for a certain airline, and best results matching the total price preference of the customer can be delivered and highlighted. However, if there is no travel package available which matches the price constraint, an alternative offer can be presented and exemplified (see Figure A.4. on page 138). For instance, the following text can be generated:

*We cannot make you an offer which perfectly matches your preferences. If you are willing to **pay 2% more** for your journey, we are able to make you the following **acceptable** offer.*

Furthermore, for the prototypical implementation, the quality valuation is adjusted to the tourism domain using the well known 'star symbol' (see Figure 5.6).

In summary, several novel components for electronic commerce have been implemented on the basis of this thesis. Yet in order to realize a deeply personalized search process as described in chapter 4, the combination of existing preference components with the new ones is required.

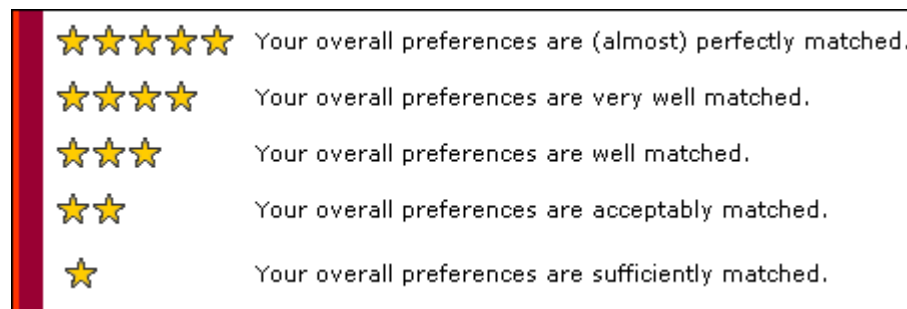


Figure 5.6. Quality valuation adjusted to the tourism domain ([Dör06])

### 5.3 Putting the Pieces Together: Interplay of Components

For the interplay and interoperability of all components, a common basis is required. This foundation is provided by the work of Kießling ([Kie02, Kie05]), where preferences are modeled as strict partial orders. The intricate interplay of preference based components, e.g., of the Preference Presenter and the Personalized Price Offer, has already been demonstrated in [KFD04]. Thus, existing components and technologies, for instance, the Preference Search Engine, as well as novel components such as the Visualization Component are based on this preference model.

In order to provide customers with deeply personalized and situated search results, the search process is divided into the four stages of Preference Analysis & Modeling, Search Interface, Query Processing, and Presentation (see the search model of Definition 4.6). In the following, it will be shown how existing and novel components fit into the model.

- **Preference Analysis & Modeling:**

In this stage of the search process, relevant preferences have to be identified and modeled accordingly. The preference constructors of [Kie05] as well as the novel TRADEOFF constructor represent a more than sufficient basis for the modeling of preferences in tourism-related electronic commerce.

- **Search Interface:**

An intuitively comprehensible access to the search process should be provided by a personalized search interface. The outlined visualization component offers a search interface for the customer's preferences. These can be defined and ordered easily and playfully per graphical drag-and-drop elements.

Furthermore, a smart preference elicitation component such as the Airport Finder can be utilized to get preferences elicited from the customer while avoiding form-driven preferences.

- **Query Processing:**

After gaining and modeling customers' preferences, it is still necessary to perform a match-making between those (wishes) and the database content (reality). By using a Preference Search Engine such as Preference XPath ([KHF+01]) or Preference SQL ([KK02]) both preferences and hard constraints can be evaluated. Thereby, deeply personalized products can be found for different travel parts. The introduced component for an advanced preference query processing, on the other hand, also takes global preferences for the entire travel package into account. Thus, preferences for individual and global aspects of a journey can be considered for the matchmaking, which leads to travel products tailored exactly to the customer's wishes.

- **Presentation:**

The quality of search results represents an important factor in sales dialogs. But any discussion about the quality has to take the customer's preferences into account. The Preference Presenter component ([Fis04, FKP06]) determines the search results' quality with respect to the customer's preferences. Furthermore, by applying the described adaptation to the tourism domain, quality valuations can be delivered which are even more comprehensible.

- **Situation Model:**

Preferences can strongly depend on situations. These should, therefore, be taken into consideration for the stages of the search process. The Preference Repository ([Hol04]) allows the management of long term preferences which are sensitive to particular situations. It provides a semantic and well-founded framework for the storage of travelers situated preferences.

The described interplay of components is illustrated in Figure 5.7.

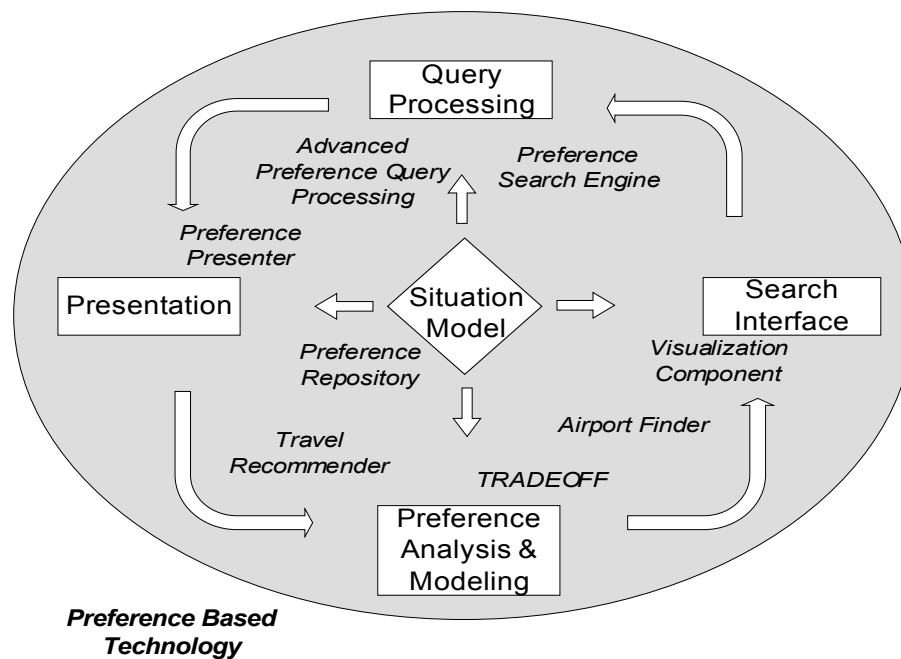


Figure 5.7. Preference based components for the search process in tourism

## 5.4 Typical Use Case Scenarios

In this section, the beneficial functionalities of a deeply personalized search process which is made up of the components described, will be demonstrated by means of several use case scenarios. These scenarios include a business traveler, a student, and a family.

### Scenario A:

Remember business traveler Mark from the beginning of this thesis. He has to travel to London again. His preferences can still be expressed as follows:

1. In order to represent his company in a positive manner, a premium class car **must** be rented due to company policy.
2. Accommodation **should** be in the northern part of London because of the proximity to a great deal of partners.
3. On account of good experiences in the past, the airline **should** be British Airways. This represents a long-term preference.
4. The limit of \$1.500 **must not** be exceeded due to company's policy.
5. Due to Mark's tight schedule, the journey to the departure airport **should** be as short as possible.

The company's policies (point 1 and 4) and Mark's long-term preference (point 3) can be stored within a Preference Repository as shown below. Thereby, these situated preferences can be seamlessly included into the search process.

```

<PreferenceRepository>
  <UserIdentifier>
    <Name xml:lang="en">Mark</Name>
  </UserIdentifier>
  <PreferenceData name="Company_Policies">
    <Situation>
      <Condition key="Role" value="business"/>
      <Condition key="Search_Stage" value="query-processing"/>
    </Situation>
    <HardCondition>
      <Condition key="RentalCarCategory" value="premium"/>
      <Condition key="MaximumSumConstraint" value="1.500"/>
    </HardCondition>
  </PreferenceData>
  <PreferenceData name="Airline">
    <Situation>
      <Condition key="Role" value="business"/>
      <Condition key="Search_Stage" value="query-processing"/>
    </Situation>
    <Preference>
      <POS att="Airline">
        <POSSet>
          <Value val="British Airways"/>
        </POSSet>
      </POS>
    </Preference>
  </PreferenceData>
</PreferenceRepository>

```

Mark registers onto the travel portal and subsequently fills out the web form only with his dates, his destination airport London, and the preferred location of the hotel. Instead of a departure airport, he is able to specify his starting point in Augsburg. Finally, he specifies his preference for a fast arrival at the airport.

As soon as Mark presses the 'Search' button, the travel portal can proceed as follows:

- First, the travel portal determines airports offering flights to London. Along with Mark's preference, these airports are then automatically submitted to the webservice of the Airport Finder (subsection 5.2.1). The component delivers Munich Airport together with a corresponding schedule (see also Figure A.2. in the appendix).



- By using the airport delivered in step 1, the travel portal can now start the search itself. First, the individual queries for flight, hotel, and rental car are composed. The company's policy for a rental car of premium quality as well as Mark's long-term preference for flights with British Airways can be integrated from the Preference Repository (subsection 4.2.1).
- These queries are expanded and evaluated by the novel component for an advanced preference query processing (subsection 5.2.3). This way, the best combinations of flight, hotel, and rental car can be determined with respect to the customer's individual and global preferences.

Finally, the best combinations are presented to Mark with respect to his preferences. The well known 'stars' symbol is used for the valuation of the combinations' quality (subsection 5.2.3).

In conclusion, in opposition to current travel portals, Mark is not forced to specify his departure airport. Instead, the Airport Finder determines the best match. Furthermore, it is possible to differentiate between the customer's hard and soft constraints. Long-term preferences and situations can easily be integrated from the Preference Repository. By using the component for the advanced preference query processing, moreover, global preferences and constraints can also be taken into consideration.

#### Scenario B:

Student George would like to make a language trip in preparation for his English exam. George is registered at the Friedrich-Alexander University of Erlangen-Nuremberg and lives in Nuremberg. He is looking for an appropriate weekend trip including flight and hotel. His needs and preferences are as follows:

1. George would like to improve his spoken English. Therefore, the official language at the destination **must** be English.
2. The total trip **should** be as cheap as possible due to budget limitations.
3. The journey to the airport **has to** be by public transportation because George has no car.
4. Since he has a red-green color-deficit, red color **should** be avoided on the search interface.

Points 3 and 4 denote situations which correlate with George's physical state and his travel opportunities, respectively. Since they do not change frequently, they can be stored in the Preference Repository.

```
<PreferenceRepository>
  <UserIdentifier>
    <Name xml:lang="en">George</Name>
  </UserIdentifier>
  <PreferenceData name="Means_of_Transportation">
    <Situation>
      <Condition key="Search_Stage" value="preference-analysis"/>
    </Situation>
    <HardCondition>
      <Condition key="Transportation" value="public"/>
    </HardCondition>
  </PreferenceData>
  <PreferenceData name="Physical_Deficit">
    <Situation>
      <Condition key="Search_Stage" value="search-interface"/>
    </Situation>
    <Preference>
      <NEG att="Color">
        <NEGSet>
          <Value val="red"/>
        </NEGSet>
      </Preference>
    </PreferenceData>
  </PreferenceRepository>
```

First, George registers onto the online travel portal. By using the Preference Repository, the system is aware of his physical deficit, i.e., a color distinction deficiency. Thus, the search interface can be constructed accordingly by the Visualization Component (subsection 5.2.2).

Thereafter, George can specify his current location and his preferences for an English-speaking country and the total price. After pressing the 'Search' button, the portal can proceed as follows:

- The Travel Recommender component determines English-speaking countries and corresponding destinations (subsection 5.2.1). Suitable departure airports will be identified by the travel portal.
- Afterwards, the Airport Finder determines the costs for journeys to these airports as well as the corresponding schedules (subsection 5.2.1). The hard condition for the means of public transportation is integrated from the Preference Repository.
- The travel portal is now able to determine the cheapest combination of flight, hotel, and journey to the airport by means of the advanced preferences query processing component (subsection 5.2.3).

Finally, the cheapest trip to a destination in an English-speaking country can be presented to George.

In contrast to current travel portals, George is not forced to specify a destination or departure airport. The best ones are automatically determined with respect to his preferences. Moreover, the search interface can take his color-deficit into account. Such influencing situations can easily be integrated by means of the Preference Repository.

### Scenario C:

Alexandra would like to find a suitable trip for the family's vacation in Istanbul. The family lives in Augsburg and consists of 7 persons including 5 children. Her preferences and needs are expressed as follows:

1. This time the journey to the airport **has to** be by public transportation. The journey **should** include as few changes as possible.
2. The hotel **should** have 3 stars. However, a better quality would be preferred as long as the price does not exceed the price range of a typical 3 star hotel. As always, child-friendly hotels are preferred by the family (long-term preference).
3. Due to the amount of persons, the rental car **should** be a 'van'.

Alexandra's long-term preference for child-friendly hotels (point 2) can be stored in the Preference Repository.

```
<PreferenceRepository>
  <UserIdentifier>
    <Name xml:lang="en">Alexandra</Name>
  </UserIdentifier>
  <PreferenceData name="Travel-Group-Composition">
    <Situation>
      <Condition key="Children" value="yes"/>
      <Condition key="Search_Stage" value="query-processing"/>
    </Situation>
    <Preference>
      <POS att="Hotel_Features">
        <POSSet>
          <Value val="child-friendly"/>
        </POSSet>
      </POS>
    </Preference>
  </PreferenceData>
</PreferenceRepository>
```

First, Alexandra has to register onto the online travel portal. Then she can specify the desired destination and her preferences for the quality of the hotel, the rental car, and the journey to the airport.

The search starts with pressing the 'Search' button:

- First, the online travel portal determines all airports offering flights to the desired destination. In a second step, these airports are then automatically submitted to the Airport Finder component along with Alexandra's preferences for the journey (subsection 5.2.1). The airports of Munich, Stuttgart, and Nuremberg are delivered together with corresponding schedules. All of them can be reached by public transportation and one change only.
- By using the airports from the first step, the travel portal can now start the search itself. First, individual queries for flight, hotel, and rental car are composed. Alexandra's long-term preference for child-friendly hotels can be gained and integrated automatically from the Preference Repository. Her preference for the quality of hotels is modeled by the TRADEOFF constructor (definition 4.9).

Finally, the best combinations are presented to Alexandra with respect to her preferences. The 'stars' symbol is used for the valuation of the combinations' quality (subsection 5.2.3). The product presentation can include the schedules for the journey to the airport as well as the flight schedules, since Alexandra prefers connections without many changes.

As opposed to common travel portals, Alexandra is not asked to specify a departure airport. The best choices are determined by the Airport Finder. By using the preference search, it is possible to differentiate between hard and soft constraints. In addition, the novel TRADEOFF constructor can be used to model quality-related preferences in an intuitive manner. Moreover, long-term preferences and situations can easily be integrated from the Preference Repository.

In conclusion it can be said that several essential components for a personalized search process in tourism have come into existence based on this thesis. Together with existing preference based components and technologies, a deeply personalized search process as described in the model of chapter 4 can be provided. Preferences and situation knowledge can be used in each stage of the process in order to support the customer as an individual being in an individual situation. This means that sales skills can be automated which so far could have only been executed by a human appointee in a travel agency. As demonstrated by means of several use cases above, that way the search results are tailored exactly to the customer's wishes. Furthermore, the search process itself is not only personalized and situated, but comfortable and fast. Taking the customer's situations and preferences into account can increase the customers' *decision satisfaction* ([HLH07]) significantly. This, in turn, may lead to more turnovers for the travel portal.

## 6 Achievements and Related Work

A deep personalization of the search process in tourism comprises different fields of research. It ranges from the modeling of search processes in general to the integration of heterogeneous data sources for an advanced preference elicitation. The contributions of this thesis are compared to existing approaches. Existing models for search processes in electronic commerce are compared in the first section. Thereafter, recent works from the field of situational influences and their modeling are outlined. In the third section, related approaches concerning the integration of heterogeneous data sources are shown. Finally, the improvements of this work are exhibited for all the different stages of the search process in tourism.

### 6.1 Search Process Model

A sequential three-stage model for the search process in electronic commerce is presented in [MHD00]. Here, search preferences of the customer have to be identified and managed during the first stage. Then the search for the product takes place. Note that the authors have not specified the nature of their search. It may be processed via search engines, browsing on websites or any other navigational technology. Finally, the products are compared by the customer in order to make a choice. Recognizing the important role of personalization in electronic commerce, this model is supplemented by three new components ([SBA01]); one component for the storage of customer preferences and psycho-demographic profiles, one personalization component, and also the opportunity to speak to a human contact person for customer help (see Figure 6.1).

Obviously, a human contact partner is a nice feature. But it is also a significant cost factor, which can be reduced by the integration of preference components. This has been demonstrated by means of a joint study ([DKP+06]) regarding the search process on the internal platform of the *MAN Group*<sup>16</sup>. Furthermore, the term 'preferences' is defined rather broadly in that specific model. It includes past buying habits, psycho-demographics, and tendencies. There is no model or concept for the representation and management of preferences. In addition, situational variables which may very much influence the customers' preferences and requirements are not explicitly modeled or included in the model.

---

16 [www.man2b.com](http://www.man2b.com)

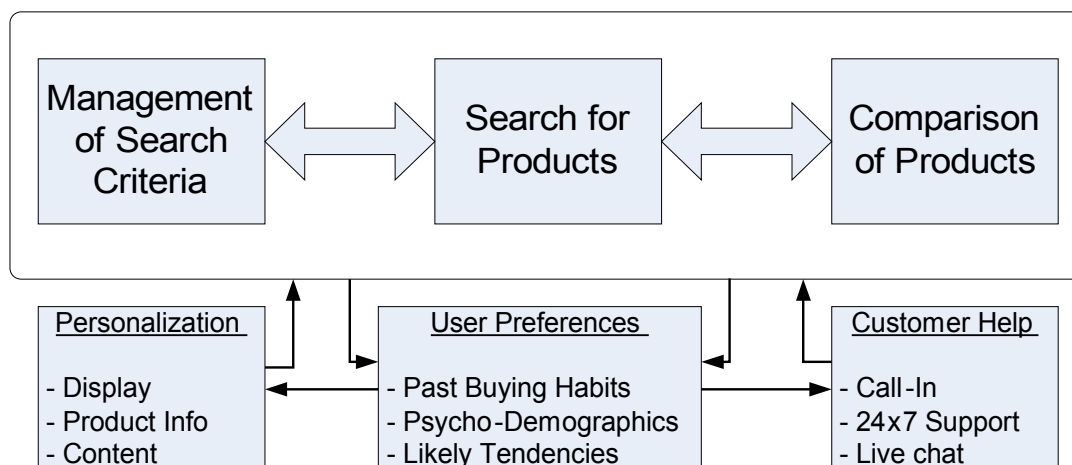


Figure 6.1. Model for the search process in e-commerce (SBA01)

In [ÖFA01] an e-procurement process for the customer is roughly described by four steps: searching through catalogs for desired products, pricing and ordering, delivery, and payment and controlling. Another work aims to automate the offer composition in e-procurement ([KFD04]). It comprises the search process and an offer generation. Firstly, the shopping cart is filled with the desired products and quantities. This includes the product search as well as the subsequent presentation of search results. Secondly, the price of the whole shopping cart is determined, possibly using a bargaining sequence. Individual steps are not bound to a linear sequence of actions. It is possible and reasonable to reiterate some steps. This work provides a first foundation for the personalization of the search process. However, because it covers the entire offer generation, some aspects of the search process itself are not examined in greater detail. The pivotal influence of situations to all the separate stages of the search process, for example, are not covered at all.

In conclusion, the model presented in this thesis provides a more fine-granular partitioning of the search process than existing models. For instance, as has been demonstrated, a search interface has to be treated as a separate and important stage of the search process for personalization. As opposed to other works, the search is modeled in a circular manner in order to support the exploration of hidden preferences. Furthermore, a situation model represents the central part of the search process, since it possibly influences each part of the search. Moreover, instead of just adding it as single component, *personalization* represents the underlying concept of the entire search process. Only then, a deeper personalization of the entire search process can be established.

## 6.2 Situational Influences

The importance of the situational context for buying behavior has already been identified by Belk ([Bel75]) in 1975. He distinguishes five types of variables: physical surroundings, social surroundings, temporal perspective, task definition, and antecedent states. The relevance of his work to the tourism and travel industry is shown in [PM00]. In this, instances for each type of variable are identified and demonstrated by means of examples. The geographical location of a particular destination in tourism may represent, for example, one part of the physical surroundings influencing a certain customer. There has been further research in tourism aiming at identifying such situational aspects since. In an observational study, six different decision styles of customers have been identified ([GZ02]). The authors distinguish between highly pre-defined, accommodation-oriented, recommendation-oriented, geography-oriented, price-oriented, and individual travelers. For instance, while highly pre-defined customers are quite sure about their destination, the recommendation-oriented customer does not have any fixed features of the trip in mind. Another work divides influencing factors into two groups: the first group contains socio-economic factors such as age, income etc., while the second group comprises travel features such as the travel-party's size ([Ric02a]). The influence of customer characteristics on planned and realized behaviors has been examined in [MW05]. This study especially confirms the importance of income and travel-party composition. In the work of [PF01], many records of caller dialogs were analyzed. This indicates that customers' interests prove to be very important to search processes.

More recently, research has been conducted which tries to determine situational factors in a user-friendly manner. There is an approach which aims at determining tourist types by using representative photos ([Ber07]). Instead of the traditional registration and profile generation process by means of long and tedious questionnaires, the authors let customers choose from a set of photos reflecting their tourism habits. In a second step, the corresponding tourist type is then inferred from the selected photos.

Holland and Kießling ([HK04]) describe an abstract meta model of situation-oriented entities and relationships. It comprises *spatial-temporal* aspects and *influences*. Influences are divided into *personal* and *surrounding* influences. Situational influences can be seamlessly integrated by means of entity-relationship-modeling. However, due to the meta character of the model, tourism-related aspects are not covered.

To conclude, a huge amount of research has been undertaken which aims at describing influential situational variables for the tourism domain. In addition, an abstract meta model for situations and, more recently, a way to identify situational aspects like the tourist type in a more customer-friendly manner have been developed. In this thesis, however, a novel situation model for the travel and tourism domain has been presented. In addition, the integration of tourism-related situational influences on the search process has been demonstrated by using advanced components such as the Preference Repository ([Hol04, KFD04]).

### 6.3 Integration of Heterogeneous Information Sources

The core of the *Harmonise* project ([HAR07, Mis03]) is a shared, conceptual reference schema, the so-called *Interoperable Minimum Harmonise Ontology* (IMHO), representing some relevant concepts for main parts of the tourism industry, e.g., accommodation and travel. Stakeholders in the tourism industry do not have to change their own data format, since mediators can transform the local data format into a representation based on IMHO and vice versa. However, a lot of possibly relevant information sources for travelers such as route finders or railway companies etc. are not covered. Moreover, the project is still in the process of being validated and tested for market release. Its success is directly connected to its acceptance on the market, since its stakeholders have to cooperate, i.e., to implement the mediators transforming their own data format into a representation based on IMHO.

The internet offers a huge amount of data for travelers. It is, for instance, possible to find information about the weather, the currency, the language, possible leisure activities, and so on. Within the ESPRIT project *MIRO-Web* ([Hal00]), a set of middleware components has been developed in order to provide transparent access from standard web browsers to multiple heterogeneous data sources. *MIRO-Web* is based on a three-tier architecture with a Data Source Adapter Layer, a Mediation Layer, and a Client Layer. Heterogeneous data are transformed into a structured format by wrappers. Then they are integrated and combined by mediators before they can be shown to users. In *TheaterLoc* ([Bar00]) an entire virtual application based on wrappers and mediators has been implemented which allows users to get information about theaters and restaurants for a number of cities in the USA. In another work ([Ash01]) mediator components based on the Resource Description Framework RDF ([Bri04]) have been proposed. Again, based on RDF, a mediation facility responsible for the integration of heterogeneous data from hotel suppliers has been presented ([KW00]).

In summary, there are several technical approaches for the integration of heterogeneous data from the internet, i.e., the above proposed architecture based on wrappers and mediators provides a good starting point. On the one hand, existing work is being used to connect stakeholders (e.g., hotel groups or airlines) of the tourism industry by a common data format ([KW00, Har07]). On the other hand, specific information on the internet, e.g., about theaters or restaurants, can be provided to customers ([Bar00]). While customers do no longer have to manually visit related websites, they now have even more information to consider. Due to the lack of preference modeling, such information may well worsen the flooding effect. Only when applying a sophisticated preference model and search engine, e.g., ([Kie02, KFD04]), the integrated data can be used as a means to elicit customers' preferences. The proposed data integration of this thesis, therefore, provides a basis for a smart preference elicitation of the first stage of the holistic search model which can avoid *form-driven* preferences.



## 6.4 Search Process in Tourism

Since the empty-result-effect is known as a major reason in causing frustrated customers, several attempts have been made to supply customers with alternative results if there is no perfect match. The *Vague Query System* (VQS) uses multidimensional concepts and so-called Numeric-Coordinate-Representation-Tables to carry out similarity searches ([Pal02, PDK00]). Using a computed total distance, VQS always tries to deliver best matches. Thus, the annoying empty-result-effect can be avoided. Unfortunately, results can only be presented in a ranked list together with a corresponding value for the total distance (see Table 6.1). However, such a numerical presentation for the results' quality is mostly not intuitively comprehensible to human beings ([KFD04]).

| ID | Hotel     | Total Distance |
|----|-----------|----------------|
| 6  | Harriot   | 0.043          |
| 1  | Royal     | 0.074          |
| 3  | Imperial  | 0.157          |
| 4  | Rose      | 0.249          |
| 2  | Kingston  | 0.333          |
| 5  | City View | 0.667          |

**Table 6.1. Result set of exemplary query ([PDK00])**

Another study proposes an interactive query management which can deliver results by the relaxation of query constraints in case of an empty result ([Ric02b]). It is inserted into the *Intelligent Travel Recommender* (ITR) system, which aims to support the customers in information filtering and product bundling ([RAM+02]). Yet only one constraint can be changed each time. In a kind of dialog, the traveler can decide which search attribute he or she is willing to relax or skip. Unfortunately, the relaxation of a wish regarding categorical data, which the authors call a *symbolic feature constraint*, means that the whole constraint is discarded. By using a sophisticated preference model like [Kie02] for tourism, best alternatives can be delivered even for categorical data. Furthermore, arguments for the quality of search results can be delivered ([Fis04]).

In more recent work, new search approaches have been presented in order to acquire the customers' preferences. In [Ric02a] the need for a 'bottom-up' way is identified. Initial queries are updated each time the customer has chosen a specific travel product. Thereby, the customer can try to explore the option space just by selecting products. Moreover, the work of Fesenmaier et al. ([FRS+03]) also tries to support 'inspiration-seeking' customers by offering an icon-based dialog. Yet these interaction dialogs can also be lengthy. In addition, the problem here is that a human customer may not be able to comprehend the system's answers or decisions.

### **Preference modeling**

In most existing travel systems, constraints and preferences are not modeled explicitly, but remain implicit in selections made by the customer. However, taking the preferences of customers into account is a promising approach for personalization in the tourism domain ([FPZ05, KI05]). With *reality*, for instance, a travel planner is presented that uses a conversational model instead of a sequential one ([THS+03]). In a flexible interaction process, the customer can react to suggested solutions by adding, modifying, or removing some preferences. After each modification, the corresponding new best solutions are automatically computed and displayed. Customers' criteria and preferences are explicitly modeled as constraint satisfaction (see also [PF00, TFP02]). However, a weighted sum model for the combination of individual soft constraints is used, which has been demonstrated to be hardly comprehensible to human beings ([KFD04]). There is another study dealing with preferences in the tourism domain. In INTRIGUE (INteractive TouRist Information GUide) homogeneous subgroups of people are modeled in order to recommend sightseeing destinations and itineraries ([Ard03, AGP+05]). Each model also contains several kinds of preferences. Those preferences for certain tourist attractions are simply specified by numerical values between 0 to 1.0, where 0 denotes no interest at all. A reasonable modeling of preferences is essential for the personalization of applications. It is difficult to accordingly adjust the weighted sum model to the customer's preferences because people mostly do not think along the lines of "My preference A should be weighted with 10, while preference B should be weighted with 13." An intuitively comprehensible and sophisticated preference model like [Kie02] can represent the customer's wishes far better.

### **Tradeoffs**

It is unlikely that all constraints and wishes of customers can be perfectly satisfied in electronic commerce and tourism. Thus, it is very common that tradeoffs and compromises between criteria have to be made. In [PF02, PKF03] a visualization of tradeoffs by a so-called *tradeoff map* is proposed (see Figure 6.2).

It aims to provide customers with information so that they are able to make a decision afterwards. Since it is visualization only, an involvement on the part of the customer is required to deal with the described tradeoffs. In the model of [FPZ05], tradeoffs can be dealt with by adding a preference or modifying the weight of some preferences. Yet before the preference model can be modified, customers have to express such preference tradeoffs or preference modifications in an additional interaction dialog. As described above it may be difficult for customers to adjust weightings according to their preferences.

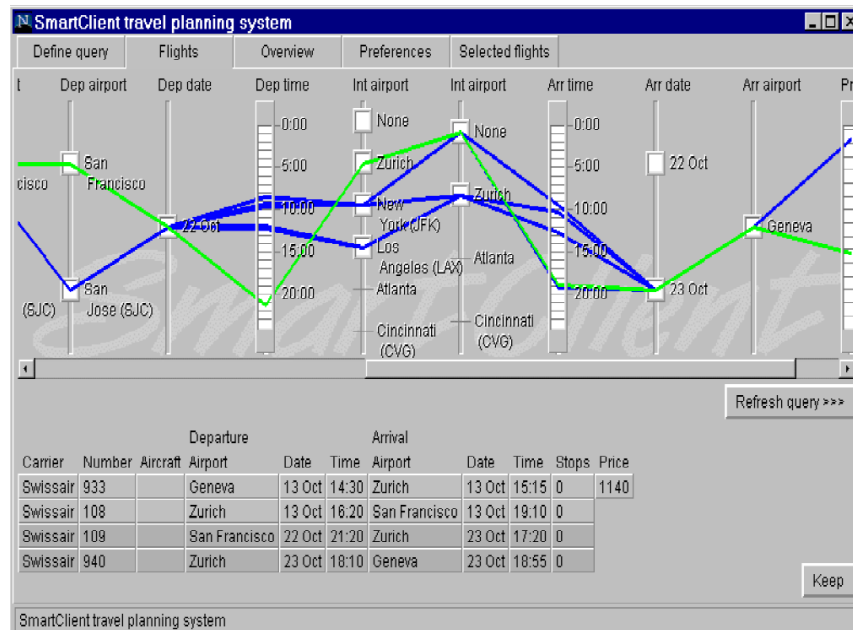


Figure 6.2. Visualization of tradeoffs ([PKF03])

The study of [BKO04], on the other hand, aims to automate the adjustment of soft constraints in case of a tradeoff. A framework is presented that implements the so-called 'entailment' operator in order to generate new constraints representing tradeoffs. For example, if there are no or only very few matches in the result set, 'filters' are used to decide, which preferences are involved in the tradeoff and which ones should be modified to deal with it. Obviously, the definition of these filters is the critical element of the approach. A modification that does not represent the customer's desires may lead to wrong results. The authors do not address this important point in detail. Instead, they suggest that the system learns the filters based on previous sessions with the user.

An intuitively comprehensible preference model such as Kießling's ([Kie02]) can appropriately represent customers' preferences without any weightings. Furthermore, the preference constructor TRADEOFF can be used to deal with typical tradeoffs between price and quality without requiring any additional involvement on the part of customers. It takes the customers' price constraints, which are mostly related to quality, into account. Thereby, customers get suitable products of a desired or even higher quality with respect to their underlying price constraint.

### Location based information

Recently, it has been recognized that information about tourism-related locations may represent an important factor for the decision process of travelers ([Pal02]). In [PRS02] a geographic support for the search is presented. Maps with an integrated view of geographic and tourist information are shown to customers. Points of interest, e.g., hotels or restaurants, can be highlighted in those maps. The calculation of distance to certain locations is based on the

Euclidean distance, which means that distance is the beeline between two geographic locations. In DESTOUR ([TZP04]) tourist information, for instance, accommodations and tourist services are integrated into geographic maps. While maps offer a good visualization of geographical context, they are only a passive means to support customers. Considering customers' preferences for the kind of journey, e.g., by car or train, or the costs or time of the journey, by the integration of heterogeneous data as described in this thesis is a novel and sophisticated approach to actively support the customers' search and decision process in tourism.

In conclusion, there exist several approaches to avoid the tedious empty-result-effect. Often a similarity search is offered. Yet this can not deliver any intuitively comprehensible argument about the quality of search results. There are also approaches offering a kind of dialog for the search process, but this can be a lengthy process. Only in a few cases, an explicit preference model is used within the tourism domain. However, these models are often based on hardly understandable numerical representations and weightings. In this thesis, the semantically rich model of Kießling ([Kie02, Kie05]) has been used as a basis. Thereby, best alternatives can be delivered automatically if there is no perfect match. Furthermore, the TRADEOFF constructor can deal with common tradeoffs and compromises in the tourism domain without further involvement on the part of customers.

## 7 Summary and Outlook

In this final chapter, the above is first summarized by especially pointing out its essential accomplishments. Then an outlook and suggestions for future work complete this thesis.

### 7.1 Summary of this Thesis

The search engine is the crucial link between a customer and the online travel portal and, hence, deserves highest attention. Benckendorff ([Ben06]) stated that searching belongs to the basic factors, which are critical because customers attach a high level of importance to them. Unfortunately, the average customer is often over-strained when arranging a vacation on the internet due to technical problems, incomprehensible interfaces, and insufficient search engines ([ÖK03]). While the concept of *dynamic packaging* aims to provide travel packages tailored exactly to the customer's needs, it suffers greatly from insufficient search technologies. This could be one reason why only 23% of the airlines provide hotel and car reservations on their websites [LL00].

At the beginning of this thesis, the drawbacks of today's search engines in tourism were analyzed. The impact of the tedious and frustrating empty-result-effect as well as the lack of suitable preference models was demonstrated by means of a compact market analysis. In doing so, also the complex nature of travelers' preferences was discussed. Even in simple scenarios there are preferences regarding individual aspects of the journey, for example the flight, and there are preferences about global constraints, e.g., for the total price of an entire package. Besides, there are different kinds of preferences, e.g., hidden or form-driven preferences, which have to be treated accordingly in order to enable a deep personalization of the search process. Following the analysis, a review of the semantically rich preference model of [Kie02, Kie05] and the achievements around this model and its features has been given.

In the main part of this thesis, a novel holistic approach for the personalization of the entire search process in electronic commerce applications was presented. The cycle model comprises four stages for the preference analysis and modeling process, the construction of the search interface, the processing of database queries, and the presentation of search results. Using this fine-granular model of the search process, each stage can be adequately adjusted to customers and situations. A situation model accordingly adjusted to the tourism domain is placed at the

center, thereby influencing each step of a personalized search. Travel influences such as the travel party's size and composition or personal influences, e.g., the customer's decision style, can be integrated seamlessly by using ER modeling techniques.

Afterwards, the search process in tourism has been closely examined with respect to the search model, i.e., each stage has been studied in the context of tourism. Firstly, it has been shown that the Preference Repository ([Hol04]) can also be used to store and integrate situated tourism-related preferences. For the *preference analysis & modeling* stage of the search model, a new preference constructor dealing with typical price-quality tradeoffs was presented. Furthermore, the integration of heterogeneous information sources from the internet was introduced as an instrument to avoid form-driven preferences. That way, customers can be enabled to express their underlying preferences for the trip. For the specification of preferences by the customer, an intuitive and personalized *search interface* plays an important role. For example, there might be a single field for an upper price limit of a customer, or two fields for a preferred price interval, or a slider for enthusiastic mouse users. Since the search interface is treated as a separate and important piece of the search process, a personalized adjustment of the search interface with respect to customers' situations and preferences could be demonstrated. An advanced *processing* of database *queries* dealing with the interplay of individual and global preferences was also presented. Thereby, situated and custom-tailored travel products with respect to both individual and global preferences can be delivered to customers. A product *presentation* adapted to the tourism domain can indicate the quality of travel packages, thus convincing the customer and increasing the customer's satisfaction.

The research of this thesis has led to the development of novel tourism-related components as well as to the implementation of a deeply personalized and situated prototype called COSI-MA<sup>T</sup>. The interplay of novel and existing preference based components automates skills that so far could only be executed by a human appointee in a travel agency. The advantages of a situated and personalized search process in tourism were demonstrated on the basis of several common use cases and scenarios. Finally, it has been demonstrated how the interplay of personalized search components might lead to a customer experience similar to the one made with a human employee in a travel agency.

In conclusion, a novel model for the search process in electronic commerce, a situation model adjusted to tourism as well as numerous sophisticated preference based technologies for the development of deeply personalized travel portals were provided.

## 7.2 Future Work

A next step would be to analyze the impact of the introduced model and technologies applied to existing online travel portals. All novel technologies are interoperable and the single usage of a component is possible, thereby providing maximal flexibility for deployment.

According to [Pre06], the influence of the so-called *Travel 2.0* will increase. Travel 2.0 is a term that represents the extension and customization of the concept of Web 2.0 into a form that applies to the travel and tourism industry. In particular, it comprises peer review websites, social communities, and blogs. It is about 'empowering' users, encouraging travelers to create online content to be shared with other readers. Customers are no longer content just to find the lowest price, they are looking to take control and **identify the perfect trip** ([Gro07]). Applying the knowledge and technologies provided in this thesis might represent one step in this direction, since it is supposed to deliver travel products tailored exactly to an individual customer in an individual situation. Furthermore, growing social communities on the internet might offer an opportunity to gain tourism-related preferences and situations. By a combination of information extraction technologies with sophisticated preference mining tools such as the Preference Miner ([Hol04]), information about interests and opinions of customer groups could be gained.

If the fast-growing databases of tourism portals demand a faster processing of preference queries and an even further improved response time, more sophisticated optimization techniques could be reasonable. Current research focuses on improved processing algorithms for preference queries such as BNL++ or Hexagon ([PKE06, PK07]). Both algorithms can be applied to the TRADEOFF-constructor, further reducing processing time. While the presented approach for the extensions of preference queries - regarding individual as well as global preferences - delivers travel products tailored exactly to customers and their situation, there could be performance issues due to the complex combinatorial nature of the deep personalization process. Current work aims to reduce the processing time by faster algorithms, while still delivering deeply personalized results.

Finally, more visionary advancements are imaginable. In [KFD04, FDW+04] the deployment of advanced technologies via FIPA-agents<sup>17</sup> for the integration of situational context is described, e.g., for emotion recognition via mimic and natural speech recognition. Obviously, customers of online travel portals would benefit from natural speech processing. Wishes could be expressed easily by anyone: "I am looking for a trip to Barcelona. I would like to have a hotel with 3 stars."

---

17 [www.fipa.org](http://www.fipa.org)





## Bibliography

- [ACB96] P. K. Ankomah, J. L. Crompton, and D. Baker, *Influence of Cognitive Distance in Vacation Choice*, in: *Annals of Tourism Research*, Vol. 23 (1), pp. 138-150, Elsevier, 1996.
- [AGP+05] L. Ardissono, A. Goy, G. Petrone, and M. Segnan, *A Multi-Agent Infrastructure for Developing Personalized Web-Based Systems*, in: *ACM Transactions on Internet Technology*, Vol. 5, Issue 1, pp. 47-69, ACM Press, 2005.
- [Ard03] L. Ardissono et al., *INTRIGUE: Personalized Recommendation of Tourist Attractions for Desktop and Handset Devices*, in: *Applied Artificial Intelligence*, Vol. 17 (8-9), pp. 687-714, 2003.
- [Ash01] A. Ashari et al., *A Framework for Integrating Heterogeneous Tourism Information Sources*, in: *Proceedings of the 8th International Conference on Information and Communication Technologies in Tourism*, Montreal, Canada, pp. 102-111, 2001.
- [AW00] R. Agrawal and E. Wimmers, *A Framework for Expressing and Combining Preferences*, in: *Proceedings of the 2000 ACM SIGMOD International Conference on Management of Data*, Dallas, USA, pp. 297-306, 2000.
- [Bar00] G. Barish et al., *The TheaterLoc Virtual Application*, in: *Proceedings of the 7th National Conference on Artificial Intelligence and 12th Conference on Innovative Applications of Artificial Intelligence*, Austin, Texas, USA, pp. 980-987, 2000.
- [Bel75] R. W. Belk, *Situational Variables and Consumer Behavior*, in: *Journal of Consumer Research*, Vol. 2, Issue 3, pp. 157-164, 1975.
- [Ben06] P. Benckendorff, *An Exploratory Analysis of Traveler Preferences for Airline Website Content*, in: *Information Technology & Tourism*, Vol. 8, Nr. 3/4, pp. 149-159, 2006.
- [Ber07] H. Berger et al., *Quo Vadis Homo Turisticus? Towards a Picture-based Tourist Profiler*, in: *Proceedings of the 14th International Conference on Information and Communication Technologies in Tourism*, Ljubljana, Slovenia, pp. 87-96, 2007.

- [BKO04] S. Bistarelli, J. Kelleher, and B. O'Sullivan, *Tradeoff Generation Using Soft Constraints*, in: LNCS Proceedings, Recent Advances in Constraints, Vol. 3010/2004, Springer, 2004.
- [Bra06] T. Bray et al., *Extensible Markup Language (XML) 1.0 (Fourth Edition)*, <http://www.w3.org/TR/REC-xml>, 2006.
- [Bri04] D. Brickley et al., *The W3C Resource Description Framework (RDF)*, <http://www.w3.org/RDF>, 2004.
- [CD99] J. Clark and S. DeRose, *The XML Path Language (XPath) Version 1.0*, <http://www.w3.org/TR/xpath>, 1999.
- [Cho02] J. Chomicki, *Querying with Intrinsic Preferences*, in: Proceedings of the International Conference of Advances in Database Technology, Prague, Czech Republic, pp. 34-51, 2002.
- [Cho03] J. Chomicki, *Preference Formulas in Relational Queries*, in: ACM Transactions on Database Systems, Vol. 43 (4), pp. 427-466, 2003.
- [Cic80] R. J. Cichelli, *Minimal Perfect Hash Functions Made Simple*, in: Communications of the ACM, Vol. 23 (1), pp. 17-19, 1980.
- [CL07] J. Cardoso and C. Lange, *A Framework for Assessing Strategies and Technologies for Dynamic Packaging Applications*, in: Journal of Information Technology & Tourism, Vol. 9 (1), pp. 27-44, 2007.
- [DEP05] S. Döring, A. Eberle, and T. Preisinger, *Delivering a Personalized Result Set by the Adaptation of Preference Queries*, in: LNI Proceedings, Informatik 2005, Vol. P-67(1), pp. 241-245, Springer, 2005.
- [DKP+06] S. Döring, W. Kießling, T. Preisinger, and S. Fischer, *Evaluation and Optimization of the Catalog Search Process of E-Procurement Platforms*, in: Electronic Commerce Research and Applications, Vol. 5, Issue 1, pp. 44-56, Elsevier, 2006.
- [Dör06] S. Döring, *COSIMA T – Preference Based Search Technology in Tourism*, in: Technical Report 2006-24, University of Augsburg, Augsburg, Germany, 2006.
- [DP90] B. Davey and H. Priestly, *Introduction to Lattices and Order*. Cambridge Mathematical Textbooks, Cambridge University Press, 1990.

- 
- [DP08] S. Döring and T. Preisinger, *Personalisation and Situation Awareness of the Search Process in Tourism*, in: Proceedings of the 15th International Conference on Information and Communication Technologies in Tourism, Innsbruck, Austria, pp. 497-508, 2008.
- [DPE08] S. Döring, T. Preisinger, and M. Endres, *Advanced Preference Query Processing for E-Commerce*, in: Proceedings of the 23rd Annual ACM Symposium on Applied Computing, Fortaleza, Brazil, 2008. (to appear)
- [Exp07] Expedia Travel, <http://www.expedia.com>, retrieved 21.03.2007.
- [FDW+04] S. Fischer, S. Döring, M. Wimmer, and A. Krummheuer, *Experiences with an Emotional Sales Agent*, in: Proceedings of the Workshop on Affective Dialogue Systems, Kloster Irsee, Germany, pp. 309-312, 2004.
- [FHC+92] E. A. Fox, L. S. Heath, Q. F. Chen, and A. M. Daoud, *Practical Minimal Perfect Hash Functions for Large Databases*, in: Communications of the ACM, Vol 35(1), pp. 105-121, 2004.
- [FHK+01] S. Fischer, S. Holland, W. Kießling, and T. Ehm, *COSIMA – Your Smart, Speaking E-Salesperson*, in: Proceedings of the 20th ACM SIGACT-SIGMOD-SIGART Symposium on Principles of Database Systems, Santa Barbara, USA, pp. 600, 2001.
- [Fis04] S. Fischer, *Personalized Query Result Presentation and Offer Composition for E-Procurement Applications*, Dissertation, University of Augsburg, Augsburg, Germany, 2004.
- [FKH+02] S. Fischer, W. Kießling, S. Holland, and M. Fleder, *The COSIMA Prototype for Multi-Objective Bargaining*, in: Proceedings of the 1st International Joint Conference on Autonomous Agents & Multiagent Systems, Bologna, Italy, pp. 1364-1371, 2002.
- [FKP06] S. Fischer, W. Kießling, and T. Preisinger, *Preference based Quality Assessment and Presentation of Query Results*, in: Flexible Databases Supporting Imprecision and Uncertainty, G. Bordogna, G. Paila (eds.), Springer, 2006.
- [For07] FORSIP, <http://www.forsip.de>, retrieved 07.02.2007.
- [FPZ05] B. Faltings, P. Pu, and J. Zhang, *Agile Preference Models based on Soft Constraints*, in: Challenges to Decision Support in a Changing World, AAAI Spring Symposium, Stanford, USA, 2005.

- [FRS+03] D. R. Fesenmaier, F. Ricci, E. Schaumlechner, and K. W. Wöber, *DIETORECS: Travel Advisory for Multiple Decision Styles*, in: Proceedings of the 10th International Conference on Information and Communication Technologies in Tourism, Helsinki, Finland, pp. 233-241, 2003.
- [Fue07] Fuenf vor Flug, <http://www.5vorflug.de>, retrieved 29.03.2007. (German)
- [GM99] A. Gupta and I.S. Mumick, *Materialized Views: Techniques, Implementations, and Applications*, ISBN 0262571226, MIT Press, Cambridge, USA, 1999.
- [Gri07] M. Gritsch, *Entwicklung von Präferenz basierten Komponenten für den Suchprozess im Tourismus*. Diploma Thesis, University of Augsburg, Augsburg, Germany, 2007. (German)
- [Gro07] D. Grossman, *Travel 2.0: Social networking takes a useful turn*, <http://www.usatoday.com/travel/columnist/grossman>, retrieved 01.09.2007.
- [GW03] U. Gretzel and K. Wöber, *Intelligent Search Support: Building Search Term Associations for Tourism-Specific Search Engines*, in: Proceedings of the 11th International Conference on Information and Communication Technologies in Tourism, Cairo, Egypt, pp. 239-248, 2003.
- [GZ02] K. Grabler and A. H. Zins, *Vacation Trip Decision Styles as Basis for an Automated Recommendation System: Lessons from Observational Studies*, in: Proceedings of the 9th International Conference on Information and Communication Technologies in Tourism, Innsbruck, Austria, pp. 458-469, 2002.
- [Haa07] M. Haarnagell, *Modellierungswerkzeuge für Präferenzanfragen mit Anwendung in der Touristik*, Diploma Thesis, University of Augsburg, Augsburg, Germany, 2007. (German)
- [Hal00] M. Haller et al., *Integrating Heterogeneous Tourism Information in TIScover – The MIRO-Web Approach*, in: Proceedings of the 7th International Conference on Information and Communication Technologies in Tourism, Barcelona, Spain, pp. 71-80, 2000.
- [HAR07] HARMONISE, <http://www.harmo-ten.info>, retrieved 22.02.2007.
- [HEK03] S. Holland, M. Ester, and W. Kießling, *Preference Mining: A Novel Approach on Mining User Preferences for Personalized Applications*, in: Proceedings of the 7th European Conference on Principles and Practice of Knowledge Discovery in Databases, Dubrovnik, Croatia, pp. 204-216, 2003.

- 
- [HGF02] Y. Hwang, U. Gretzel, and D. R. Fesenmaier, *Behavioural Foundations for Human-Centric Travel Decision-Aid Systems*, in: Proceedings of the 9th International Conference on Information and Communication Technologies in Tourism, Innsbruck, Austria, pp. 356-365, 2002.
- [HHK05] A. Huhn, P. Höfner, and W. Kießling, *Towards Evaluating the Impact of Ontologies on the Quality of a Digital Library Alerting System*, in: Proceedings of the 9th European Conference on Research and Advanced Technology for Digital Libraries, Vienna, Austria, pp. 498-499, 2005.
- [HK04] S. Holland and W. Kießling, *Situated Preferences and Preference Repositories for Personalized Database Applications*, in: Proceedings of the 23th International Conference on Conceptual Modeling, Shanghai, China, pp. 511-523, 2004.
- [HK05] B. Hafenrichter and W. Kießling, *Optimization of Relational Preference Queries*, in: Proceedings of the 16th Australasian Database Conference, Newcastle, Australia, pp. 175-184, 2005.
- [HLH07] M. Heitmann, D.R. Lehmann, and A. Herrmann, *Choice Goal Attainment and Decision and Consumption Satisfaction*, in: Journal of Marketing Research, XLIV, pp. 234-250, 2007.
- [Hol04] S. Holland, *Preference Mining and Preference Repositories: Design, Algorithms and Personalized Applications*, ISBN 3898213528, ibidem-Verlag, 2004.
- [HS69] J. A. Howard and J. N. Sheth, *The Theory of Buyer Behaviour*, John Wiley and Sons Inc., USA, 1969.
- [Hsi05] T.-C. Hsieh et al., *The Effects of Emerging Technologies on the Travel, Tourism and Hospitality Marketplace*, in: a New York University and PhoCusWright Inc. Joint Research (www.phocuswright.com), New York, USA, 2005.
- [KFD04] W. Kießling, S. Fischer, and S. Döring, *COSIMA B2B – Sales Automation for E-Procurement*, in: Proceedings of the 6th IEEE Conference on E-Commerce Technology, San Diego, USA, pp. 59-68, 2004.
- [KHF+01] W. Kießling, B. Hafenrichter, S. Fischer, and S. Holland, *Preference XPATH: A Query Language for E-Commerce*, in: Information Age Economy, Physika-Verlag, Augsburg, Germany, pp. 427-440, 2001.
- [KI05] G. Koutrika and Y. Ioannidis, *Personalized Queries under a Generalized Preference Model*, in: Proceedings of the 21st International Conference on Data Engineering, Tokyo, Japan, pp. 841-852, 2005.

- [Kie02] W. Kießling, *Foundations of Preferences in Database Systems*, in: Proceedings of the 28th International Conference on Very Large Data Bases, Hong Kong, China, pp. 311-322, 2002.
- [Kie05] W. Kießling, *Preference Queries with SV-Semantics*, in: Proceedings of the 11th International Conference on Management of Data, Goa, India, pp. 15-26, 2005.
- [KK02] W. Kießling and G. Köstler, *Preference SQL – Design, Implementation, Experiences*, in: Proceedings of the 28th International Conference on Very Large Data Bases, Hong Kong, China, pp. 990-1001, 2002.
- [KW00] M. Kaukal and H. Werthner, *Integration of Heterogeneous Information Sources*, in: Proceedings of the 7th International Conference on Information and Communication Technologies in Tourism, Barcelona, Spain, pp. 81-92, 2000.
- [LL00] R. Law and R. Leung, *A Study of Airlines' Online Reservation Services on the Internet*, in: Journal of Travel Research, Vol. 39 (2), pp. 202-211, 2000.
- [MHD00] G. E. Miles, A. Howes, and A. Davies, *A Framework for Understanding Human Factors in Web-based Electronic Commerce*, in: International Journal of Human-Computer Studies, Vol. 52, Issue 1, pp. 131-163, Elsevier, 2000.
- [Mis03] M. Missikoff et al., *Harmonise – Towards Interoperability in the Tourism Domain*, in: Proceedings of the 10th International Conference on Information and Communication Technologies in Tourism, Helsinki, Finland, pp. 58-66, 2003.
- [Mit05] N. Mitsche, *Understanding the Information Search Process within a Tourism Domain-specific Search Engine*, in: Proceedings of the 12th International Conference on Information and Communication Technologies in Tourism, Innsbruck, Austria, pp. 183-193, 2005.
- [MPJ07] M. Morse, J.M. Patel, and H.V. Jagadish, *Efficient Skyline Computation over Low Cardinality Domains*, in: Proceedings of the 33th International Conference on Very Large Databases, Vienna, Austria, pp. 267-278, 2005.
- [MW05] R. March and A. G. Woodside, *Tourism Behaviour – Travellers' Decisions and Actions*, ISBN 0851990215, CABI Publishing, 2005.
- [Myj07] MyJack24, <http://www.myjack24.de>, retrieved 29.03.2007. (German)
- [NMT06] M. Niemann, M. Mochol, and R. Tolksdorf, *Improving Online Hotel Search – What Do We Need Semantics For?*, in: International Conference on Semantics (Application Paper), Vienna, Austria, 2006.

- 
- [ÖFA01] H. Österle, E. Fleisch, and R. Alt, *Business Networking – Shaping Collaborations between Enterprises*, Springer Verlag, Berlin, 2001.
- [ÖK03] A. Öörni and S. Klein, *Electronic Travel Markets: Elusive Effects on Consumer Behavior*, in: Proceedings of the 10th International Conference on Information and Communication Technologies in Tourism, Helsinki, Finland, pp. 29-38, 2003.
- [Pal02] J. Palkoska et al., *Advanced Query Mechanisms in Tourism Information Systems*, in: Proceedings of the 9th International Conference on Information and Communication Technologies in Tourism, Innsbruck, Austria, pp. 438-447, 2002.
- [PDK00] J. Palkoska, A. Dunzendorfer, and J. Küng, *Vague Queries in Tourism Information Systems*, in: Proceedings of the 7th International Conference on Information and Communication Technologies in Tourism, Barcelona, Spain, pp. 61-70, 2000.
- [PF00] P. Pu and B. Faltings, *Enriching buyers' experiences: the SmartClient approach*, in: Proceedings of the SIGCHI conference on Human factors in computing systems, Hague, Netherlands, pp. 289-296, 2000.
- [PF01] B. Pan and D. R. Fesenmaier, *Exploring Caller Dialogue: Analyzing Directed Information Search for Tourism Products*, in: Proceedings of the 8th International Conference on Information and Communication Technologies in Tourism, Montreal, Canada, pp. 244-250, 2001.
- [PF02] P. Pu and B. Faltings, *Effective Interaction Principles for User-Involved Constraint Problem Solving*, in: Workshop notes of the 2nd International Workshop on User-Interaction in Constraint Satisfaction, New York, USA, pp. 77-91, 2002.
- [PF03] B. Pan and D. R. Fesenmaier, *Travel Information Search on the Internet: A Preliminary Analysis*, in: Proceedings of the 10th International Conference on Information and Communication Technologies in Tourism, Helsinki, Finland, pp. 242-251, 2003.
- [PK04] P. Pu and P. Kumar, *Evaluating Example-based Search Tools*, in: Proceedings of the ACM Conference on Electronic Commerce, New York, USA, pp. 208-217, 2004.
- [PK07] T. Preisinger and W. Kießling, *The HEXAGON Algorithm for Evaluating Pareto Preferences*, in: Online-Proceedings of the 3rd Multidisciplinary Workshop on Advances in Preference Handling in conjunction with VLDB, Vienna, Austria, 2007.

- [PKE06] T. Preisinger, W. Kießling, and M. Endres, *The BNL<sup>++</sup> Algorithm for Evaluating Pareto Preference Queries*, in: Proceedings of 2nd Multidisciplinary Workshop on Advances in Preference Handling in conjunction with ECAI, Riva del Garda, Italy, pp. 114-121, 2006.
- [PKF03] P. Pu, P. Kumar, and B. Faltings, *User-Involved Tradeoff Analysis in Configuration Tasks*, in: Workshop notes of the 3rd International Workshop on User-Interaction in Constraint Satisfaction, 9th International Conference on Principles and Practice of Constraint Programming, Kinsale, Ireland, pp. 85-102, 2003.
- [PM00] A. Pizam and Y. Mansfeld, *Consumer Behavior in Travel and Tourism*, ISBN 0789006103, Haworth Hospitality Press, New York, 2000.
- [Pre06] M. Prestipino, *From Information Behaviour of Independent Travellers to Requirements for Information Systems*, in: Proceedings of the 13th International Conference on Information and Communication Technologies in Tourism, Lausanne, Switzerland, pp. 262, 2006.
- [PRS02] F. Pühretmair, H. Rumetshofer, and E. Schaumlechner, *Extended Decision Making in Tourism Information Systems*, in: Proceedings of the 3th International Conference on E-Commerce and Web Technologies, Aix-en-Provence, France, pp. 57-66, 2002.
- [RAM+02] F. Ricci, B. Arslan, N. Mirzadeh, and A. Venturini, *ITR: A Case-Based Travel Advisory System*, in: Proceedings of the 6th European Conference on Advances in Case-Based Reasoning, Aberdeen, UK, pp. 613-627, 2002.
- [Ric02a] F. Ricci, *Travel Recommender Systems*, in: Journal of IEEE Intelligent Systems, Vol. 17 (6), pp. 55-57, 2002.
- [Ric02b] F. Ricci et al., *Intelligent Query Management for Travel Product Selection*, in: Proceedings of the 9th International Conference on Information and Communication Technologies in Tourism, Innsbruck, Austria, pp. 448-457, 2002.
- [Sam83] P. Samuelson, *Foundations of Economic Analysis, Enlarged Edition (Harvard Economic Studies)*, Harvard University Press, 1983.
- [SBA01] B. G. Silverman, M. Bachann, and K. Al-Akharas, *Implications of Buyer Decision Theory for Design of eCommerce Websites*, in: International Journal of Human Computer Studies, Vol. 55 (5), pp. 815-844, 2001.
- [SEK06] B. Satzger, M. Endres, and W. Kießling, *A Preference-Based Recommender System*, in: Proceedings of the 7th International Conference on Electronic Commerce and Web Technologies, Krakow, Poland, pp. 31-40, 2006.



- 
- [TFP02] M. Torrens, B. Faltings, and P. Pu, *SmartClients: Constraint satisfaction as a Paradigm for Scaleable Intelligent Information Systems*, in: Constraints, Vol. 7 (1), pp.49-69, Springer, 2002.
- [THS+03] M. Torrens, P. Hertzog, L. Samson, and B. Faltings, *reality: a Scalable Intelligent Travel Planner*, in: Proceedings of the 2003 ACM Symposium on Applied Computing, Melbourne, Florida, USA, pp. 623-630, 2003.
- [Tra07a] Travelocity, <http://www.travelocity.com>, retrieved 21.03.2007.
- [Tra07b] Traveltainment, <http://www.traveltainment.de>, retrieved 29.03.2007. (German)
- [Tui07] TUI, <http://www.tui.com/de>, retrieved 28.03.2007. (German)
- [TZP04] V. Tallinucci, A. Zehrer, and H. Pechlaner, *Using Interactive Maps as Tourism Information Source – The Case of DESTOUR.*, in: Proceedings of the 11th International Conference on Information and Communication Technologies in Tourism, Cairo, Egypt, pp. 197-211, 2004.
- [Was07] Washington Post, *What a Site: 36 Web Addresses You Should Know*, <http://www.washingtonpost.com>, retrieved 16.08.2007.
- [WBK+04] Q. Wang, W.-T. Balke, W. Kießling, and A. Huhn, *P-News: Deeply Personalized News Dissemination for MPEG-7 based Digital Libraries*, in: Proceedings of the 8th European Conference on Digital Libraries, Bath, UK, pp. 256-268, 2004.
- [Web07] Web Services – W3C recommendations, <http://www.w3.org/2002/ws/>, retrieved 01.09.2007.
- [Wer03] H. Werthner, *Intelligent Systems in Travel and Tourism*, in: Proceedings of the 18th International Joint Conference on Artificial Intelligence, Acapulco, Mexico, pp. 1620-1628, 2003.
- [WR04] H. Werthner and F. Ricci, *E-Commerce and Tourism*, in: Communications of the ACM, Vol. 47, Issue 12, pp. 101-105, 2004.
- [Zie07] B. Ziegler, *Integration von heterogenen Informationsquellen für einen Travel Destination Recommender*. Diploma Thesis, University of Augsburg, Augsburg, Germany, 2007. (German)



## List of Figures

|   |     |
|---|-----|
| Figure 1.1. Fly.de advertising slogan.....  | 10  |
| Figure 2.1. Modeling and satisfying customers' wishes in tourism: a non-trivial matter..... | 16  |
| Figure 2.2. Search forms of Expedia and Travelocity.....                                    | 17  |
| Figure 2.3. Empty-result-effect: no rental car can be offered .....                         | 18  |
| Figure 2.4. Search process on current travel portals.....                                   | 20  |
| Figure 2.5. Positioning of this thesis.....   | 22  |
| Figure 3.1. Base preference sub-constructor hierarchy ([Kie05]).....                        | 30  |
| Figure 4.1. Personalized and situated search model for e-commerce.....                      | 43  |
| Figure 4.2. Model for situations in tourism.....  | 47  |
| Figure 4.3. ER-model for tourism portal.....  | 48  |
| Figure 4.4. Price and quality tradeoff.....   | 50  |
| Figure 4.5. Price levels for customers looking for a hotel with 2 stars.....                | 52  |
| Figure 4.6. Price levels for customers looking for a hotel with 3 stars.....                | 54  |
| Figure 4.7. Determining maximal prices and price levels.....                                | 57  |
| Figure 4.8. Specifying a form-driven preference.....  | 65  |
| Figure 4.9. Preference elicitation: low integration effort.....                             | 68  |
| Figure 4.10. Architecture for the integration of data sources (excerpt from [Hal00]).....   | 75  |
| Figure 4.11. Level of personalization and required effort.....                              | 76  |
| Figure 4.12. Query processing dealing with maximum/minimum sum constraints.....             | 81  |
| Figure 5.1. COSIMA B2B – the fully automated sales agent ([KFD04]).....                     | 96  |
| Figure 5.2. Preference based components in overview.....                                    | 97  |
| Figure 5.3. Smart preference elicitation to find a suitable airport ([Gri07]).....          | 98  |
| Figure 5.4. Visualization based on metaphors ([Haa07]).....                                 | 99  |
| Figure 5.5. Generic visualization approach ([Haa07]).....                                   | 100 |
| Figure 5.6. Quality valuation adjusted to the tourism domain ([Dör06]).....                 | 101 |
| Figure 5.7. Preference based components for the search process in tourism.....              | 103 |
| Figure 6.1. Model for the search process in e-commerce ([SBA01]).....                       | 110 |
| Figure 6.2. Visualization of tradeoffs ([PKF03]).....                                       | 115 |



---

## List of Tables

|  |     |
|--|-----|
| Table 4.1. Hotel database.....   | 52  |
| Table 4.2. Database for rental cars.....                                   | 58  |
| Table 4.3. Distances determined by DISTT.....                              | 59  |
| Table 4.4. Sample database for airports .....                              | 68  |
| Table 4.5. Sample database for airports.....                               | 69  |
| Table 4.6. Sample database for induced preferences.....                    | 73  |
| Table 4.7. Database for flights (F).....                                   | 77  |
| Table 4.8. Database for rental cars (C).....                               | 77  |
| Table 4.9. Database for hotels (H).....                                    | 77  |
| Table 4.10. Overall price and quality for travel product combinations..... | 79  |
| Table 4.11. $BMO_H^*$ result relation for hotels.....                      | 84  |
| Table 4.12. Sample $BMO^T$ result relation for travel package.....         | 86  |
| Table 4.13. $BMO^T$ result relation for example.....                       | 91  |
| Table 6.1. Result set of exemplary query ([PDK00]).....                    | 113 |



# Appendix A

## Screenshot - Airport Finder Component:

**Plan your approach to airport**  
Approach search preferences

**Your travel data**

Departure:  Date:

choose Airports: preferred arrival time:

number of Airports:  special Airport:

**Your personal travel preferences**

| means of travel                             | Usage / Rejection                       | Importance                              |
|---|---|---|
| footpath                                    | <input type="text" value="don't mind"/> | <input type="text" value="don't mind"/> |
| interurban railway                          | <input type="text" value="don't mind"/> | <input type="text" value="don't mind"/> |
| subway                                      | <input type="text" value="don't mind"/> | <input type="text" value="don't mind"/> |
| streetcar                                   | <input type="text" value="don't mind"/> | <input type="text" value="don't mind"/> |
| bus   | <input type="text" value="reject it"/>  | <input type="text" value="low"/>        |
| train                                       | <input type="text" value="use it"/>     | <input type="text" value="high"/>       |
| taxi  | <input type="text" value="don't mind"/> | <input type="text" value="don't mind"/> |
| car   | <input type="text" value="reject it"/>  | <input type="text" value="high"/>       |
| transfer service                            | <input type="text" value="don't mind"/> | <input type="text" value="don't mind"/> |
| parking <input type="text" value="0"/> Days | <input type="text" value="don't mind"/> | <input type="text" value="don't mind"/> |

**Your personal price / travel time preferences**

Possessing Bahncard:  search for fastest approach:

Importance:

tolerated variation:  % search for cheapest offer:

Importance:

Figure A.1. Airport Finder component ([Gri07])

**Airport Finder – Determining the Fastest Approach:**

Plan your approach to airport  
 Approach search Offers

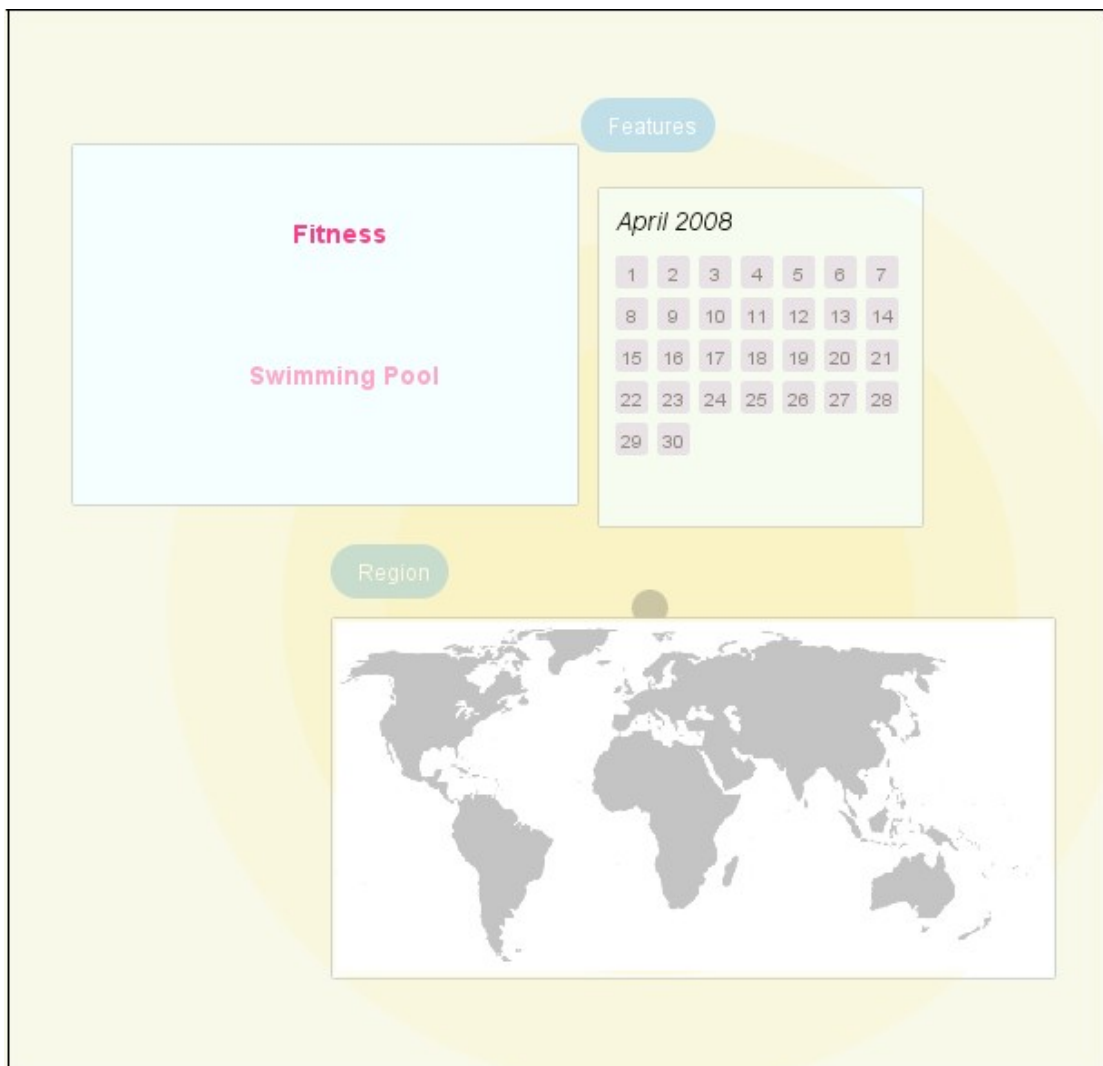
| Approach Offers |                |              |              |                          |  |         |                |   |              |
|-----------------|----------------|--------------|--------------|--------------------------|--|---------|----------------|---|--------------|
| Departure Date  | Departure Time | Arrival Date | Arrival Time | Departure                | Destination                                | Costs   | Time of Travel | Means of Transportation                     | Weekendliche |
| 09.10.07        | 15:13          | 09.10.07     | 16:58        | Zugsplitzstraße+Augsburg | Flughafen Terminalbereich AB, M&#252;nchen | 36.75 € | 105 Minutes    | footpath, interurban rail, bus, train, taxi | false        |
| 09.10.07        | 15:13          | 09.10.07     | 17:02        | Zugsplitzstraße+Augsburg | Flughafen Terminalbereich AB, M&#252;nchen | 43.09 € | 109 Minutes    | interurban rail, train, taxi                | false        |
| 09.10.07        | 11:41          | 09.10.07     | 13:37        | Zugsplitzstraße+Augsburg | Allg&#228;u Airport, Memmingenberg         | 69.04 € | 116 Minutes    | train, taxi                                 | false        |
| 09.10.07        | 13:50          | 09.10.07     | 15:48        | Zugsplitzstraße+Augsburg | N&#252;rnborg Flughafen                    | 30.5 €  | 118 Minutes    | footpath, subway, train, taxi               | false        |
| 09.10.07        | 14:50          | 09.10.07     | 16:48        | Zugsplitzstraße+Augsburg | N&#252;rnborg Flughafen                    | 28.1 €  | 119 Minutes    | footpath, subway, train, taxi               | false        |
| 09.10.07        | 14:20          | 09.10.07     | 16:18        | Zugsplitzstraße+Augsburg | Flughafen Terminal Bereich A, M&#252;nchen | 83.65 € | 119 Minutes    | train, taxi                                 | false        |
| 09.10.07        | 14:50          | 09.10.07     | 16:50        | Zugsplitzstraße+Augsburg | N&#252;rnborg Flughafen                    | 35.8 €  | 120 Minutes    | train, taxi                                 | false        |
| 09.10.07        | 13:50          | 09.10.07     | 15:51        | Zugsplitzstraße+Augsburg | N&#252;rnborg Flughafen                    | 38.2 €  | 121 Minutes    | train, taxi                                 | false        |

[select an offer and click here for a detailed view](#)

| Detailed View              |                |                |              |              |                         |                |
|----------------------------|----------------|----------------|--------------|--------------|-------------------------|----------------|
| Departure                  | Departure Date | Departure Time | Arrival Date | Arrival Time | Means of Transportation | Time of Travel |
| Zugsplitzstraße+Augsburg   | 09.10.07       | 15:13          | 09.10.07     | 15:15        | Taxi                    | 2 Minuten      |
| Augsburg-Hochzoll          | 09.10.07       | 15:25          | 09.10.07     | 15:56        | RE 4173                 | 31 Minuten     |
| München-Pasing             | 09.10.07       | 16:01          | 09.10.07     | 16:03        | S-Bahn                  | 2 Minuten      |
| München-Laim               | 09.10.07       | 16:11          | 09.10.07     | 16:46        | S-Bahn                  | 35 Minuten     |
| München Flughafen Termi... | 09.10.07       |                | 09.10.07     |              | Fussweg                 | 0 Minuten      |
| München Flughafen Termi... | 09.10.07       | 16:54          | 09.10.07     | 16:58        | Bus                     | 4 Minuten      |

Figure A.2. Airport Finder – journey descriptions ((Gri07))



**Screenshots - Personalized Search Interface:**

**Figure A.3. Sample visualization for tourism portal ([Haa07])**

## Screenshots – Result Presentation of COSIMA<sup>T</sup>:

**COSIMA<sup>T</sup> - DEMO** Basket Customer Service

0 Article

Home Plan your trip

**Search results**

**We cannot make you an offer, which perfectly matches your preferences.  
If you are willing to pay 2% more for your journey we are able to make you the following acceptable offer:**

**TOP OFFER** ★★

**Flight** ☹️

**Lufthansa | Economy**  
**06.01.07** Munich (MUC) - Barcelona (BCN)  
**13.01.07** Barcelona (BCN) - Munich (MUC)

**Hotel** ☹️

Green Hotel Location: **City** Arrival: see Flight  
 Meals: full Category: **\*\*\*** Departure: see Flight

**Rental car** ☹️

Ford Fiesta | **Small car** | Limousine  
 Pick-up: barcelona airport  
 Drop-off: barcelona airport

**Total price: 920,00 €**

[Back](#) [Modify this search >](#)

All available offers

### More offers

| Flight                        | Hotel   | Rental car   | Price in EUR | Departure | Overall quality |
|-------------------------------|---|--|--------------|-----------|-----------------|
| [LH1]<br>Lufthansa<br>Economy | [M606]<br>Motel 6 ***<br>Meals: half<br>Location: <b>Downtown</b> | [K1]<br>Ford Fiesta<br><b>Small car</b><br>Limousine       | 1.040,00 €   | 06.01.07  | ★★★             |
| [LH1]<br>Lufthansa<br>Economy | [M606]<br>Motel 6 ***<br>Meals: half<br>Location: <b>Downtown</b> | [K55]<br>Renault Megane<br><b>Compact car</b><br>Limousine | 1.302,00 €   | 06.01.07  | ★★★★            |

Figure A.4. Result presentation of COSIMA<sup>T</sup> (Dör06)

# Curriculum Vitae



## Personal Data

Name: Sven Döring  
 Date of birth: 13.01.1977  
 Place of birth: Leipzig, Germany  
 Nationality: German

## Education and Professional Experience

|                   |   |
|-------------------|---|
| 09/1983 – 08/1991 | Secondary school (Mittelschule), Belgern  |
| 09/1991 – 08/1996 | Grammar school (Gymnasium), Torgau with graduation  |
| 09/1996 – 06/1997 | Military service  |
| 10/1997 – 10/2002 | Study of computer science at the Technical University of Dresden with graduation to the degree Dipl.-Inf. |
| since 12/2002     | Research assistant at the chair for databases and information systems at the University of Augsburg       |

Augsburg, 04.02.2008

