Personalized Navigation of Heterogeneous Product Spaces using SmartClient

Pearl Pu
Ergonomics of Intelligent Systems and Design
Database Laboratory
Swiss Institute of Technology Lausanne
CH-1015 Ecublens EPFL, Switzerland
+41 21 693 6081
pearl.pu@epfl.ch

Boi Faltings
Artificial Intelligence Laboratory
Computer Science Department
Swiss Insitute of Technology Lausanne
CH-1015 Ecublens EPFL, Switzerland
+41 21 693 2738
boi.faltings@epfl.ch

Abstract

Personalization in e-commerce has so far been server-centric, requiring users to create a separate individual profile on each server that they like to access. As product information is increasingly coming from multiple and heterogeneous sources, the number of profiles becomes unmanageably large. We present SmartClient, a technology based on constraint programming where a thin but intelligent client provides personalized information access for its user. As the process can run on the user's side, it allows much stronger filtering and visualization support with a wider range of personalization options than existing tools. It also eliminates the need to personalize many sites individually with different parameters, and supports product configuration and integration of different information sources in the same framework. We illustrate the technology using an application in travel e-commerce, which is currently under commercial deployment.

Keywords

Personalization, data integration, electronic catalogs, intelligent user interfaces

INTRODUCTION

Assisting a shopper in finding the best product for his or her needs, called product brokering, is a key requirement in e-commerce. Products have multiple attributes, represented by a vector of predefined features. We distinguish simple products and configurable products. Simple products, e.g., restaurants, cameras, or books, do not contain parts that can be substituted by others in a customization process, whereas configurable products, such as travel, personal computers or insurance policies, include several constituent parts which must be composed according to users' needs before it is offered. Product brokering in both cases requires assisting users in finding information in a complex multidimensional space.

Several comparative shopping tools, based on technology such as Jango [2] or Junglee [13], have already been introduced. These tools work on servers connected to a central product database, or on an infomediary such as a portal. They generally assume that data can be accessed rapidly and reliably and that all users have the same criteria.

Since more merchants offer services and products via the Internet, product brokering requires gathering information services in a dynamic and distributed environment. Data come from multiple sources, are heterogeneous (not every product has the same features), and products are configurable rather than simple. For example, in the travel industry, a portal (such as Expedia [3] or Travelocity [9]) is much more likely to offer products of a variety of different companies and integrated products (e.g., packages including air, car rental, and event-related parts) for coordinated planning tasks. While there is often a database of constituent parts (such as segments of flights in flight databases), the number of all possible combinations of these parts (i.e., customized trips) makes deployment of a database difficult. Furthermore, the data comes from multiple sources and usually cannot be maintained in a central database. Information must often be transmitted through network connections that are transient, limited in capacity, brittle, and costly. In addition, travel e-commerce has to allow travelers to modify trips while away from their offices and interact with the travel site via mobile computing devices. These features are difficult to implement with an architecture that relies on a central server, but become easier in a distributed architecture.

Several new techniques have recently been developed for personalized information seeking in complex multidimensional spaces. Leading examples include the FindMe approach for assisted browsing in selecting restaurants among a range of other things [1], and Lieberman's work on finding apartments [4]. They present important advances in user experience and personalization.

We build on these advances to accommodate the requirements of travel e-commerce: many complex and often unsatisfiable criteria, configurable products, and distributed and heterogeneous data sources. To satisfy these requirements, both user experience and computation architecture have to be addressed simultaneously. Furthermore, this type of tools must also provide quality

services to mobile network users, thus further constraining the size of the tools. Here we describe **SmartClient**, a distributed agent-based architecture for gathering information. It implements navigational features that can be tailored to the exact needs of each user.

- It gathers the initial large quantity of "crude information" into a temporary data store,
- it uses constraint satisfaction problem solving techniques to model the data without the full deployment of a database,
- it guides users to browse in this complex data space by configuring solutions to their preferences and criteria,
- and it assists them to choose the best solutions that fit their profile and dynamic criteria.

SmartClient uses on the order of 200K bytes of Java code to deliver all the intelligence that a user requires to perform information seeking and searching. Each application of SmartClient for a particular domain requires slightly different implementation of the graphical user interface (GUI). Travel planning using SmartClient, *Isy-travel*, employs some of the most sophisticated GUIs called ***contextual navigation maps*** [7]. It requires additional 300K bytes for the GUI part, which includes a real-time zoomable world map. On the other hand, we succeeded in implementing a version of *Isy-travel* for the PalmPilot using no more than 250K bytes of code.

SMARTCLIENT: small but agile

The main idea of SmartClient is based on decentralizing the comparison processes to end users' computers by providing them with a configuration plug-in using constraint satisfaction technology. Rather than pre-classifying data at the server side, as would be done in common web services technology, data from different sources is sent to the client in compressed form and filtered according to the user's own combination of criteria at the client side.

Criteria are modeled as constraints. The main mechanism for obtaining them is by letting the user express critiques on solutions. The results are then presented with a variety of visual tools, including tables, Starfield displays, and graphical maps to allow users to navigate towards ideal solutions [6-7].

ACQUIRING USER CRITERIA

The main challenge in building a personalized filtering tool is to obtain an accurate model of the user's criteria and preferences. This is difficult for two reasons:

1) users cannot state all their criteria up-front. For example, we might hate to change planes in Heathrow, but we would not think state this requirement initially.

2) criteria are highly variable. For example, the same user may be planning a business trip in the morning, where time constraints are the most important, and a trip for his family in the evening, with price as the overriding criterion.

We have adopted a solution, similar to [5], based on the principle that people find it easy to critique proposed solutions - if there is nothing to critique, they have already found what they were looking for. We implement this using a mixed-initiative dialog: the system generates 30 solutions that are good according to the known criteria. The user states additional criteria in any order, generally expressing aspects that he dislikes about one or several of these solutions. The system again generates the 30 best solutions according to the revised criteria, and the cycle continues. We start the process with a user profile that contains invariant criteria such as preferred airlines, general preferences for short flights and few stops, etc. Criteria are added incrementally as planning proceeds. Please see [8] regarding details of the SmartClient architecture and the constraint satisfaction solving technology.

Currently, IsyTravel is being integrated into leading European business travel planning software and will be commercially deployed in early 2002.

ACKNOWLEDGEMENT

This work was sponsored by the Swiss National Science Foundation. We thank Marc Torrens, Sebastian Gerlach, and the team of Iconomic Systems for their implementation of the SmartClient architecture and graphical user interfaces. We also thank the reviewers for their valuable comments.

REFERENCES

1. Burke, D., Hammond, K.J., and Young, B.C. The FindMe Approach to Assisted Browsing. *IEEE Expert*, pp. 32-40, July-August 1997.
2. Excite Shopping. Available at: http://www.jango.com/. Accessed September 27, 2001.
3. Expedia. Available at: http://www.expedia.com/.
4. Lieberman, H. Autonomous Interface Agents. *In Proceedings of CHI'97*, March 1997.
5. Shearin, S. and Lieberman, H. Intelligent Profiling by Example, in *Proceedings of Conference on Intelligent User Interfaces*, ACM Press, 2001.
6. Pu, P. and Faltings, B. Enriching buyers' experiences: the SmartClient approach, in *Proceedings of ACM CHI'2000*, ACM Press, 2000.
7. Pu, P. and B. Faltings (2001). Contextual Maps for Personalized Navigation of Electronic Catalogs. Technical Report, Swiss Federal Institute of Technology (EPFL), Lausanne, Switzerland.
8. M. Torrens, B. Faltings, and P. Pu. (2001). "SmartClient: Constraint Satisfaction as a Paradigm for Scalable Intelligent Information Systems." *Constraint Journal* (Special Issue on Constraint Agents), in press
9. Travelocity. Available at: http://www.travelocity.com/