Constraint-Based Personalization Model: Multi-Channel Messaging

Kal Toth
Associate Professor
Dept. of Computer Science
Oregon State University, OR, USA
ktoth@cs.orst.edu

Sai Rajkiran Nagboth
Graduate Student
Dept. of Computer Science
Oregon State University, OR, USA
nagboth@cs.orst.edu

ABSTRACT

This paper presents a personalization framework for web and e-business applications that builds on the architectural model presented by Instone in [3] as well as foundation work presented by Toth in [1]. This personalization framework incorporates consumer-provider attributes; considers system/network variables; and models personalization as a constraint-based problem. This paper also describes an initial working prototype constraint engine for a prolific messaging application. Users collaborate with each other and with web applications using email, instant messaging and short text messaging services employing their PCs, Personal Digital Assistants (PDAs) and cell phones.

Keywords

Personalization, Preferences, Models, Constraints, Solvers.

1. INTRODUCTION

The growth in e-business and the need to accurately focus e-business marketing, sales and services, is motivating the search for better personalization models and techniques. Though personalization has attracted often-excessive hype, only a few authors have attempted to formulate practical models that lend themselves to analysis and design. The previous paper by Toth [1] identified a range of personalization problems and laid the foundation concepts for the current research. This paper formulates a practical model that provides a structure and solving technique that may be applied to solve such personalization problems.

The output of this research is a generalized personalization model and a constraint engine prototype. A prototype constraint engine has been developed in collaboration with The Wise Net Inc. (TWN) in support of TWN's multi-channel messaging architecture.

In this paper we start by defining personalization and presenting the limitations of existing models. The types of users ("actors") that expose their "personalities" in the e-business context are identified and characterized to further our understanding of how their behaviors can influence and become influenced by their surroundings. The symmetrical and asymmetrical behaviors "consumers" and "providers" are explored and related. The framework and impacts of both the wire-line and wireless information network is then described and their impacts on consumer-provider behaviors are explored.

Finally, we explain the applicability of constraint theory to personalization implemented by a software prototype. The prototype operates on personalization attributes representing the preferences and constraints of collaborating actors in a multi-channel messaging application.

2. WHAT IS PERSONALIZATION?

The primary focus of personalization is on improving a user's experience while engaged with the Web.

***Personalization** is defined as a process that facilitates interaction among consumers and providers such that individual consumers are enabled to more readily access the content and services of providers, and individual providers are enabled to more effectively and easily deliver their content and services to consumers. The "individuality" of a consumer and a provider is a key aspect of this definition. It implies that their individual (a.k.a. personal) attributes, including identity, preferences, constraints and disposition (e.g. location, presence) potentially impact on the personalization process.*

This definition permits an e-business service provider to be personalized much like any human user. At the implementation level this definition translates into software mechanisms that automate or optimize the interactions among consumers and providers in accordance with their personalization data. Such mechanisms can be achieved through appropriate user interface design, customization and tailoring of communication interfaces and channels, and dynamic adaptability of e-business applications and services.

Note also that personalization problems often need to address conflicting preferences and constraints among consumers and providers as well as the system itself.

3. PERSONALIZATION MODEL

Our model specifies personalization variables for human users, business applications and the system. The model also incorporates constraint-based methods and intelligent agents to automate the processing of personalization attributes to optimize or enhance the user experience and achieve the end-goals of personalization for both consumers and providers. The model recognizes that true optimization can often not be achieved but, that good decisions can be facilitated by the model to enhance any given entity's behavior and performance within the total system context.

3.1 Composite Model

Figure 1 illustrates the foundation components of our personalization model in a composite fashion. Providers deliver responses to requests, or asynchronous alerts driven by pre-set push criteria. This collaboration is modulated by provider and consumer content namely identification data, profile information and personal preferences. Preferences include the user's personalization variables, for example, to filter incoming messages, customize user interfaces and drive outgoing network searches.

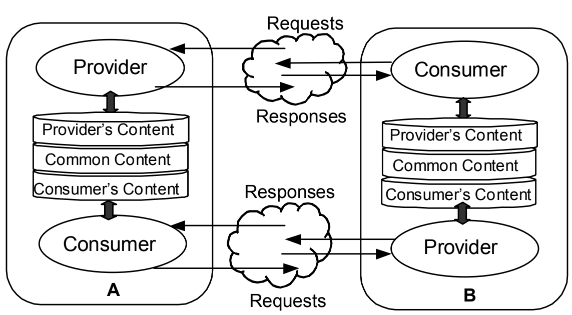

Fig. 1. Personalization Model: Composite Model

As discussed earlier, e-business and Internet collaborating entities are rarely pure "providers" or "consumers". The composite model per Figure 1 is a more complete representation of the sort of collaboration that might be going on between a given pair of interacting entities A and B.

3.2 Functional Layers

We have extended Instone's [3] model incorporating layers that operate on consumer and provider profiles and preferences. Figure 2 depicts this 2-layer model. The Application Layer is the primary focus of consumer-provider collaboration. It is assumed that applications will make use of the Profile & Personalization Layer for key data. Both providers and consumers will need to manage their profiles on an ongoing basis. The Personalization & profile Layer, meanwhile, contains a constraint engine that operates on consumer and provider preferences and profiles to personalize their interactions with these applications and with each other (via messaging and other collaboration tools in the Application Layer). Although Figure 2 has the appearance of a centralized single-server model, all layers could be distributed across multiple network processors. In particular, the profile and personalization data could be aggregated together with identification and authentication data at a single site or fragmented across many sites.

3.2.1 Application Layer

The application layer provides core business and collaboration services. This layer includes enterprise applications (e.g. real estate, recruitment, stock trading, health care) as well as generic services such as messaging, calendaring, task management, directory services, work group services, forms management and so forth. Note that this layer will depend on the access control and authentication services provided by the operating environment and will need to access user identification and authentication information (this is not show in the figure).

3.2.2 Profile and Personalization Layer

This layer contains profile information that is often private to the individual user or business. The data in this layer is highly dependent on strong authentication and security mechanisms provided by the operating environment.

The personalization aspect of this layer is designed to be independent as possible from components. Its main functions are to acquire data about user behavior and then use data to make recommendations to the application layer. In effect this layer is a decision support tool that helps optimize and streamline the applications being employed by the user. The personalization layer thereby uses the preferences of the provider, the consumer, and the system to optimize the user's experience. Here are a few examples that illustrate the processing carried out in the personalization layer:

- **Multi Channel Messaging:** This is the initial application area examined by the current research. In this context users have PCs, cell phones and PDAs (Palm Pilots and Pocket PCs). These client devices may employ email, instant messaging, or short messaging services (SMS) to transfer messages. Under various conditions users will prefer to receive messages on one or more of these channels. The Personalization Layer provides them with the tools to define these messaging channels and their preference rules and to make recommendations to the message router in the Application Layer whenever a message is received. An interesting feature of this strategy is that a user can use a single (universal) ID to receive messages on any of her channels. Possible personalization operations include:
    - **Automatic Forwarding of Messages:** incoming messages could be routed to a user's cell phone, PDA or a customer's e-mail under user specified

criteria (e.g. importance, message type, sender's ID, message contents and message capabilities).

  - **Filtering Incoming Messages**: messages containing specific keywords of interest could be highlighted or routed to a high priority channel.
  - **Message Blocking**: similarly SPAM, adult content and hate literature could be filtered and blocked automatically.

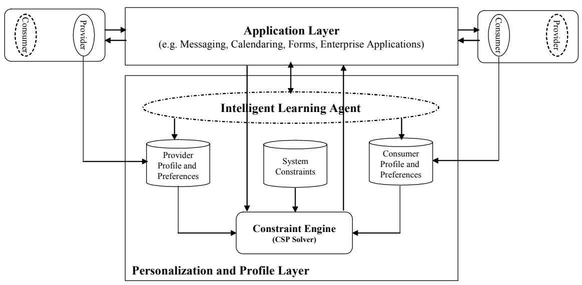

Fig. 2. Personalization Model: Functional Layers

- **Automatic scheduling of meetings and tasks:** Calendaring applications could also benefit from responding to user preferences. For example, preferred meeting times, dates and locations could be expressed as preferences and used to automatically schedule or re-schedule meetings.
- **Auto-completion of forms:** Based on prior entry of forms, fields could be automatically filled and most the most recently used values made available through pull-downs to facilitate data entry.
- **Mobile Content Personalization:** When receiving alerts ("pushed messages), preferences could be used to filter long messages and SPAM. When browsing web sites, display rendering could be executed according to user preferences and system constraints (e.g. display attributes).

3.2.3 Intelligent Learning Agent

This is an area for future study. This initial personalization models we are examining rely upon explicit definition of preferences by the users. Clearly this can be time-consuming many cases and requires the motivation of the user to perform this task diligently and on an on-going basis when preferences change.

The intelligent learning agent concept is built upon the idea that it is possible to observe user behavior and detect repetitive and recurring patterns to infer user preferences. Such an agent would be "plugged" into the Application Layer monitoring personalization variables to detect such patterns.

4. PERSONALIZATION THROUGH CONSTRAINT SATISFACTION

To facilitate mediation between a consumer and provider, some information about providers and consumers (users) is needed. This information could be a set of attributes (profiles) describing them, like name, address, social security number, employer, job title and so forth. This information could also include preferences and constraints (personalization data) describing their interests and desires, for example, what type of information they want to receive, where they want to receive it, how and when. Such profile data and preferences could be *explicitly* collected from the users by filling forms, giving ratings etc, or by observation (studying collaborative behavior). The general objective is to gather such data and provide users with the ability to manage this data.

A feasible approach for working with user profiles, preferences and constraints is to employ constraint-satisfaction techniques. These techniques use constraint variables and mathematical representations of user preferences, constraints and other personal attributes. The attributes are not always those of the user. Often they can be attributes of other collaborating users or of the networks and systems they are using.

For example in a B2C application, a consumer preference could be expressed as follows: "I prefer to receive messages on my cell phone about Nike products having more than 50% off". The consumer is specifying the preferences in terms of her attribute (cell phone) and the provider's content attributes (products with 50% off and company name).

Let us also consider the impact of "quality of service". Say some users are collaborating with each other using instant messaging on PCs and wireless PDAs and cell phones. A given user may prefer to receive important short messages via their cell phone rather than instant messaging or email. But what happens if the cellular network fails or the gateway between the cell network and the wire-line web fails? Clearly, the system should support the preference of the user to receive the message by instant messenger under such conditions.

Such expressions of user preferences and profile information can be solved using Constraint Satisfaction Problem (CSP) techniques.

Our primary objective is to make decisions on behalf of the user. To the extent possible, we would like the personalization system to automate decision-making. In many cases, however, the user will want to be presented with the "best options" or "recommendations" and make the final decision herself. This is much the way many Expert Systems work.

4.1 Constraint Based Representation

A Constraint Satisfaction Problem consists of a set of variables with associated domains and a set of constraints acting on the variables and restricting the values that the variables can simultaneously take. A solution to a constraint satisfaction problem is an assignment of a value to each variable from its domain such that all constraints are satisfied. More about CSP can be found in [6] and [11].

he attributes used to define the profile and preferences of users are CSP variables and preferences are CSP constraints. Each of the participating users (including the system) has their own sets of constraints. The constraints for each user might not depend only on their own variables. For example, a consumer may wish to receive messages according to the provider's device type. Thus the problem would have a set of variables:

$V \in \bigcup_{i=1...n} v_i$, constrains $C \in \bigcup_{i=1...n} c_i$ where v_i, c_i are the set of variables and constraints for each participating user. If d_i is the domain set for each variable vi for user i, then the total possible assignments for V would be the Cartesian product

$d_1 \times d_1 \times \ldots d_n$

A solution to the above problem would be assignments of the values to all decision variables such that the constraints of all of the participating users are satisfied. Potentially, the problem can be solved using any of the available algorithms [6][11] based on. In general, the consistency techniques (arc consistency, path consistency) are applicable in most of the cases and would help in pruning the search space through propagation.

5. MULTI-CHANNEL MESSAGING PROTOTYPE

The authors have been collaborating with The Wise Net Inc. (TWN) to develop a constraint engine for their middleware product. Their system, designed to support the real estate industry, includes a "universal messaging" capability that employs preferences and constraints to facilitate collaboration among users.

Real estate agents and their customers are assigned a single messaging identifier (wiseID) that they can use to contact each other on their PCs, cell phones and PDAs (Palm Pilots and Pocket PCs) through SMS, Instant Messaging (Yahoo!, ICQ, AIM, MSN) or e-mail. They are able to send and receive messages through any of these messaging channels with a single wiseID.

The primary goal of this system is to be able to use the single wiseID of the contact to send all messages (hence the term "universal messaging"). The sender need not know which devices are available to the recipient nor, indeed, which devices are currently available, online, or operational. In this context, all users define their preferences, including fall back scenarios, for receiving messages through their various channels – hence the use of the term "multi-channel messaging".

It turns out that multi-channel messaging is a good candidate for personalization and the application of constraint-based methods. Personalization involves automatically selecting the "best" or most appropriate channel for the user to receive messages over under various end user and system conditions.

As described earlier any problem to be solved in CSP way needs to have identified the attributes of the problem domain and the relation among them to represent desired preferences and constraints.

5.1 Formulating the Problem

In the multi-channel messaging context, the user defines personalization preferences expressed in terms of attributes (variables) that characterize incoming messages and the user's channels (email, instant messaging, SMS, etc and system attributes such as location, sender and receiver location, channel availability and presence information (whether the user is currently online or off-line). The following variables have been identified for our initial prototype:

- Variables characterizing incoming message: Sender's Identity, Sender's Channel, Importance, Message Length, Message Type, Attachment and Content (keywords)
- Variables characterizing system variables: Location, Channel Availability and Presence (on-line vs. off-line).

Consider the preference, "If the message is from Tom over ICQ and less than 150 characters then send it to my cell-phone". This can be re-expressed in terms of variables as follows: If SenderID = 'Tom' and SenderChannel = 'ICQ' and Message Length < 200 then MyChannel = 'SMS'.

Listed are some more examples of preferences are

- If the incoming message is a meeting reminder and I am online then send the message to my ICQ channel".
- If message is from Hary and through SMS then it is very important.
- If sender is Ron then message type is Real Estate.
- If a message is important send it to SMS1 if length < 150 characters, else send to my active IM channel.

5.2 Prototype Implementation

In consultation with the TWN development team, and consistent with the Personalization Model described earlier in this paper, a multi-channel selector or "recommender" was designed and prototyped. Figure 3 illustrates the prototype's design and how the prototype elements integrate with the TWN messaging architecture (only TWN's message routing agent is shown in this figure). The figure illustrates the separation between the Application Layer and the Personalization Layer.

Routing Agent (Operational TWN Component)

The roles of the principle components are as follows:

- Listens for incoming messages on various channels.
- Requests the channel recommendation from Channel Selector passing message and system parameters.
- Sends out messages to recommended channel(s)
- Channels include email, MSN, ICQ, AOL and SMS.

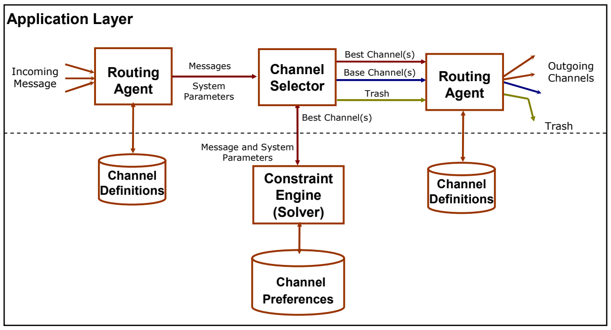

Fig.3. TWN Multi Channel Messaging Architecture

Channel selector

- Parses the incoming messages
- Calls Constraint Engine (Solver) passing message and system parameters
- Recommends "best" channel(s) to Router

Constraint Engine (CSP Solver)

This component is responsible for solving the preferences to determine the best receiving channel for the current message. The current prototype is tuned to fit directly with the multi-channel messaging application and is been implemented in Java. The prototype has been implemented in Java and has the following functionality

- Provides a template to capture message and system parameters
- Allows configuration for processing the variables and rules both conjunctively and disjunctively.
- Solves the preferences according to the steps below.
- Returns recommended "best" channel(s)
- Solver has been written to be flexible and adaptable to support easy addition of system rules and new variables.

The current prototype uses a custom algorithm based on reduction using consistency and propagation techniques. After parsing the message, the Channel Selector sends the required parameters to the constraint engine through the template provided by the constraint engine. The solver then performs the following:

- Reduces the solution space by removing the inconsistent values applying all the unary constraints imposed by the message parameters on each variable and propagating them to other variables. In this step up to 60% reduction is achieved. Reduction cannot be applied to domains of certain variables like "keywords";
- Performs a systematic search to find the values for the channel satisfying all the preferences in the reduced solution space;
- All the channels identified in the final solution set are returned.
- If more than one channel is present in the solution then the messages are sent to all the channels.

Most of reduction is achieved by running queries on the user preferences maintained in the database. This strategy lends itself to ready integration into an operational environment such as that provided by The Wise Net.

6. SUMMARY

This paper presents a layered personalization model suitable for modeling personalization for a wide range of Internet and e-business contexts. It has been generalized to support consumer, provider and system perspectives. The model makes use of Constraint Satisfaction Problem (CSP) techniques and an initial prototype has been developed. The authors have collaborated with The Wise Net Inc. to integrate this initial constraint engine to support multi-channel messaging, which is currently undergoing evaluation and enhancement.

7. REFERENCES

[1] K.C. Toth, Intelligent Agents for Business Applications Using Constraint-Based Personalization, Int'l Business & Economics Research Journal (IBER Journal), May 2002.

[2] Kal Toth and H.Todino, Instant Internet Intelligence for Wireless Business Applications, International Applied Business Research Conference, Cancun Mexico, Mar 2001.

[3] Instone.K., Information Architecture and Personalization, Argus Associates, December 2002.

[4] Markus Bylund, Conservation Policies for Personalization in Open Service Architectures, http://www.boeing.com/special/agents99/bylund.pdf, Seattle Washington, Autonomous Agents 1999.

[5] Junichi Suziki and Yoshikazu Yamamoto, Document Brokering with Agents: Persona Approach, JSSST WISS'98 (Workshop on Interactive Systems and Software).

[6] Bartak Roman, Guide to Constraint Programming 1998, http://ktiml.mff.cuni.cz/~bartak/constraints/intro.html

[7] Moshe Y.Vardi. Constraint Satisfaction and database Theory: a Tutorial, http://www.cs.rice.edu/~vardi/papers/

[8] Andreas Bartelt, Winfried Lamesdorf. A Multi-Criteria Business Models in Electronic Commerce. Proceedings of the IFIP/ACM International Conference on Distributed Systems Platforms (Middleware 2001), WS on Electronic Commerce. http://vsis-www.informatik.uni-hamburg.de/publications/viewpub.phtml/91

[9] The Personalization Consortium, http://www.personalization.org/personalization.html

[10] Ryszard Kowalczyk and Van Bui, On Constraint based Reasoning in e-Negotiation Agents, AMEC 2000.

[11] Roman Bartak, Constraint Programming – what is behind, Proceedings of the Workshop on Constraint Programming in Decision and Control, Poland, June 1999.