

Association for Information Systems AIS Electronic Library (AISeL)

AMCIS 2001 Proceedings

Americas Conference on Information Systems
(AMCIS)

December 2001

Trustworthy Personal Assistance: A Design Objective for Interactive Agents

Kurt Englmeier

German Institute for Economic Research

Josiane Mothe

Institut de Recherche en Informatique de Toulouse

Follow this and additional works at: <http://aisel.aisnet.org/amcis2001>

Recommended Citation

Englmeier, Kurt and Mothe, Josiane, "Trustworthy Personal Assistance: A Design Objective for Interactive Agents" (2001). *AMCIS 2001 Proceedings*. 223.

<http://aisel.aisnet.org/amcis2001/223>

This material is brought to you by the Americas Conference on Information Systems (AMCIS) at AIS Electronic Library (AISeL). It has been accepted for inclusion in AMCIS 2001 Proceedings by an authorized administrator of AIS Electronic Library (AISeL). For more information, please contact elibrary@aisnet.org.

TRUSTWORTHY PERSONAL ASSISTANCE: A DESIGN OBJECTIVE FOR INTERACTIVE AGENTS

Kurt Englmeier

German Institute for Economic Research
(DIW)
kurt@diwsysv.diw-berlin.de

Josiane Mothe

Institut de Recherche en Informatique de
Toulouse
mothe@irit.fr

Abstract

IRAlA is a research project to enrich available information with machine-processable semantics. The result is an approach for portal applications that provide a context-oriented navigation in heterogeneous and distributed data collections, enabling a software product to mediate between user needs and available information sources. In IRAlA, this characteristic of a semantic Web application is linked to personal assistance fostering personalized information delivery and experience sharing among users.

Tracking individual needs of users leads to personal digital assistants that appear in information retrieval systems as personal software agents. The approach presented here links the design of personal agents to portal applications.

Profiles of usage, reflecting the users' practices in searching and navigation data collections, are the building blocks for ontologies reflecting formal descriptions of personal assistance in our context.

Keywords: Agent design, semantic web, information retrieval, portal application, collaborative system.

A Model of Personal Agency in Information Retrieval

Agents are important in the area of personal assistance because they let software components interoperate within the necessary information retrieval (IR) processes of the respective application. Moreover, if appropriately designed they support a communication base of high confidence that is indispensable when it comes to transfer personal data to a remote system for analyze purposes. By the interaction with the user, in a step by step manner, a personalized agent may create "knowledge" on the user's retrieval practices. The collaboration of the different personal agents sharing "experiences" with different usage scenarios lead to a useful collection of best practices in retrieving data from complex information spaces. However, as experiences with systems capturing highly personal information tacitly show, users are getting more and more sensible to this kind of automatic profiling. It's therefore indispensable to think about ways to make the agents' behavior and its communication act more transparent and modifiable.

An agent system in this context is, first of all, a mean to provide higher level abstractions in distributed programming. These abstractions allow the designer more flexibility in determining behavior. For instance, instead of hardwiring a specific behavior into agents, system designers might have the agents negotiate with one another to determine the best practice profile for a given search situation.

All navigation in a context-oriented structure of our approach is described along concepts of a controlled vocabulary organized in a few thesaurus-like morphologies (concept hierarchies) that refer to certain global dimensions. Thus a query profile consists of concepts selected by the user from the corresponding hierarchies and descriptors of the actually regarded document. It can be assumed that a number of queries are submitted until retrieved results satisfy the information need. The profiles of all these queries constitute a scenario profile that reflects therefore a keyword description of an information need.

Concept generalization uses knowledge about the relationship between a concept and a super-concept to reformulate a query in terms of a more general concept. Concept specialization replaces a concept with a more subconcepts. In our context these two

operations are used to modify subtly and stepwise terms in a profile to detect further helpful similarities among profiles. Constraints determine to what extent replacing and interleaving may be applied, i.e. to what extent terms may be replaced or interleaved without running the risk to leave the sequence's originally inherent meaning.

The operations to match profiles are derived from the same probabilistic inference process in IR that compares document representations (descriptors) based on different forms of linguistic and/or statistical evidence on the one with representations of information needs (queries) based on similar evidence on the other side (Agosti et al. 1992).

Orchestrating Agent Activities

Agents acting in the context outlined so far need a slightly different design than those that act in a more general environment. The user's personal agent in our approach observes the user and decides which suggestions would suit best for actual as well as prospective retrieval steps. It also asks for the user's permission to communicate with remote sites to get further helpful information for its suggestions and to pass the user's record on browsing and navigation to remote agents in order to contribute to a useful stock of common retrieval "expertise", the best practice base.

According to research, a piece of software can be called an agent if it also complies to mental components such as beliefs, capabilities, choices, and commitments. These performatives (Cohen and, Levesque, 1997) have to be taken into account in the development of agents as a specialization of object-oriented programming. They define the permissible actions that agents may attempt in communicating with each other.

To act properly on ontologies, agents need a flexible transaction plan to fulfil their tasks (van Harmelen and Horrocks, 2000). It endows the agent with the capability of producing parallel execution plans, interleaving planning, execution, sensing, and re-planning requests that failed while executing other requests.

The capability of agents depends on their ability to process the requests that are directed to them. The agent has to develop an appropriate set of operations. This includes selecting the information sources for the data, the operations for processing the data, the non-agent processes the data should be delegated to, the sites where the operations will be performed, and the order in which to perform them. The transaction plan orchestrates the agents' operations triggered by the user's request.

With this results the personal agent may decide to proceed in its transaction plan to obtain better results or pass the best suggestion generated so far directly to the user, if the resulted value satisfies the agent's belief.

```

<pa.recommendation>
  <commit=user.querySequence, user.preferences ?scenarios.similarity > user.preferences.threshold>
    <choice=user.preferences.threshold, self.belief>
      <user.preferences.threshold>
        <ask=user, user.preferences.threshold>
          <subscribe=common.services.preferences.threshold>
            ...
          <choice=scenarios.similarity, self.belief, user.preferences.scenarioSet>
            <user.scenarios.longterm>
              <subscribe=local.services.scenarios.similarity, user.querySequence, user.scenarios.longterm>
                <subscribe.noCommitment -> next.choice>
                ...
              </choice>
            <common.bestPractice>
              <ask=user, permission>
              ...
            </pa.recommendation>

```

Figure 1. Fragment of the Personal Agent's Transaction Plan (In Pseudo Code) for Retrieving Usage Scenarios

Interaction with Users

The personal agent (PA) controls the communication between all agents on the user's site and the remote ones. It is the only one that may be authorized by the user to communicate with remote "colleagues" and exchange "knowledge" on the type of profiles

they are in charge with. Users can explicitly withhold their agents to pass any or certain kind of personal information to a remote instance. PA reacts only on the explicit request of the user.

The main tasks of the user's personal agent is to give recommendations for her or his next search strategy.

A suggestion may be derived exclusively from personal records ("In the past you've chosen ..."). In many cases however, it may be pretty helpful for the user if her personal agent also resorts to some common navigation expertise. This recommendations could be like: "The majority of users having navigated so far chose for the following steps the following concepts among the related coordinates".

Besides, the agent also informs the users on new documents recently published that may be of importance. Based on observing users during navigation, the system automatically adapts itself to the users' preferences and interests. This model of concept feedback is explained to the users in order to make the agents' actions transparent, controllable, and manageable.

Implementation

The approach outlined here draws on the concept of transaction agents. Its building blocks are ontologies reflecting the agents' environment and transaction plans mapping their tasks within the system architecture. The agents themselves are implemented in Java. They are partly realized as RMI objects. To make information exchange among them and with their environment also machine processable, the system architecture draws on recent data exchange and metadata standards such as XML and RDF.

Conclusion

This paper described important features of agent design in IR with a special emphasis on usage profiles representing scenarios. The outlined architecture inclines to Knoblock and Ambite's (1999) approach that includes modularity, extensibility, flexibility, efficiency, and adaptability.

In our approach we added responsibility on user's personal interests. As long as an agent is nothing more than a piece of software, responsiveness cannot be an inherent trait of an agent rather being a design objective for the interaction with the user. All the actions of an agent must be predictable and controllable to give the users the feeling they are dealing with their own agents. Most agents in current systems resemble more spies eavesdropping behind the curtain to siphon personal data to the good of business strategies of a company or the other. The quest is to develop simple communication conventions that makes an software agent fully controllable by the user while complying with the functional requirements of the remote system it is interacting with. Otherwise personal agency will be rejected like any other malformed metaphor for the design of user interaction.

Acknowledgment

Research outlined in this paper is part of the IRAIA project supported by the European Commission under the Fifth Framework Programme (IST-1999-10602). However views expressed herein are ours and do not necessarily correspond to the IRAIA consortium.

References

- Agosti, M.; Gradenigo, G.; Marchetti, P.G., "A hypertext environment for interacting with large databases." *Information Processing and Management*, 28, 1992, pp. 371-387.
- Cohen, P.R.; Levesque, H.J., "Communicative Actions for Artificial Agents", in *Software Agents*, J.M. Bradshaw (ed), AAAI Press, Menlo Park, 1997, pp. 420-421.
- Knoblock, C.A.; Ambite, J.L., "Agents for Information Gathering", in *Software Agents*, J.M. Bradshaw (ed), AAAI Press, Menlo Park, 1997, pp. 347-373.
- van Harmelen, F.; Horrocks, I., "FAQs on OIL: The Ontology Interference Layer". *IEEE Intelligent Systems & their applications*. (15:6), November/December 2000, pp. 69-72.