Requirements Eng (2007) 12: 23–40
DOI 10.1007/s00766-006-0038-5

ORIGINAL ARTICLE

Rosario Girardi · Leandro Balby Marinho

A domain model of Web recommender systems based on usage mining and collaborative filtering

Received: 9 February 2006 / Accepted: 17 July 2006 / Published online: 23 August 2006

Abstract Considering the increasing demand of multi-agent systems, the practice of software reuse is essential to the development of such systems. Multi-agent domain engineering is a process for the construction of domain-specific agent-based reusable software artifacts, like domain models, representing the requirements of a family of multi-agent systems in a domain, and frameworks, implementing reusable agent-based design solutions to those requirements. This article describes the domain modeling tasks of the MADEM methodology and a case study on the application of GRAMO, a MADEM technique, for the construction of the domain model of ONTOWUM, specifying the common and variable requirements of a family of Web recommender systems based on usage mining and collaborative filtering.

Keywords Multi-agent systems · Domain engineering · Domain models · Recommender systems · Usage mining · Collaborative filtering

1 Introduction

Recommender systems [1] are a particular type of information filtering applications. In a collaborative filtering approach [32], they provide a user with personalized recommendations based on the similarity between his/her profile and the ones of other users with similar interests. User profiles, representing the information needs and preferences of users, can be inferred from the ratings that users provide on information items, explicitly or implicitly, through their interactions with a system. A user model, a representation of this profile, can be obtained implicitly through the application of Web usage mining techniques [13, 18, 21, 47, 48].

The increasing demand of this kind of applications justifies the development of reusable software artifacts for their development.

MADEM (Multi-Agent Domain Engineering Methodology) is a software development methodology for multi-agent domain engineering defining the modeling concepts, tasks and products for the development of a family of multi-agent systems in a problem domain. A family of systems is defined as a set of existing software systems sharing some commonalities but also particular features [15].

Multi-agent domain engineering is a process for the construction of domain-specific agent-based reusable software artifacts, like domain models, representing the requirements of a family of multi-agent systems in a domain, and frameworks, implementing a reusable agent-based solution to those requirements.

The MADEM methodology integrates techniques for domain analysis, domain design and domain implementation: GRAMO (Generic Requirement Analysis Method based on Ontologies), DDEMAS (Domain Design technique of Multi-Agent Systems) and DIMAS (Domain Implementation technique of Multi-Agent Systems), respectively. Earlier work on GRAMO and DDEMAS has been already published [23–26].

The MADEM products are represented as instances of the ONTOMADEM ontology. ONTOMADEM (ONTOlogy-driven tool for the MADEM methodology) is a knowledge-based tool for guiding the modeling tasks, capturing and representing the products of MADEM. Ontologies [30, 31] are an appropriate abstraction mechanism for the specification of high-level reusable software artifacts like domain models, frameworks and software patterns. A complete description of MADEM and ONTOMADEM is out of the scope of this article.

R. Girardi (✉)
Federal University of Maranhão,
65085–580 Campus do Bacanga,
São Luís, MA, Brazil
E-mail: rgirardi@deinf.ufma.br

L. Balby Marinho
University of Freiburg,
Georges–Kökler-Allee 51,
D-79110 Freiburg, Germany
E-mail: marinho@informatik.uni-freiburg.de

This article describes a case study on the application of the GRAMO technique for the construction of the ONTOWUM domain model. ONTOWUM (ONTOlogy-driven model of recommender systems based on Web Usage Mining) specifies a family of multi-agent systems for Web personalized recommendations based on usage mining and collaborative filtering.

Two main purposes justified the development of this experience: first, the evaluation of the GRAMO technique in a problem domain characterized by its relative complexity but also relative maturity and second, the construction of a reusable model of the common and variable requirements of applications in this domain. Complexity is mainly related with the great number of available techniques leading to alternative computational solutions to the problem domain. As described in Sect. 3.3, these techniques have been evaluated through the development of several specific applications showing their respective advantages and drawbacks. However, most developments lack a methodological approach inhibiting their reuse. On the other hand, an integration of these development experiences in a framework is needed to attend the common and variable requirements of this application family.

The paper is organized as follows. Section 2 introduces the main modeling concepts of MADEM and briefly describes its tasks and products in the context of a multi-agent domain engineering process. Section 3 describes main knowledge of the ONTOWUM domain on recommender systems, information filtering and Web usage mining. A first description of the ONTOWUM family requirements is also introduced in this section. Section 4 details the modeling tasks of GRAMO and their application on the construction of the domain model of ONTOWUM. Section 5 analyzes related work on the applications of Web usage mining for Web personalization and Web usage mining for the semantic Web. Section 6 concludes the paper discussing the results, contributions and limitations of this case study and further work being conducted.

2 An overview of the MADEM methodology

Modeling concepts, tasks and products of MADEM are based on techniques for Domain Engineering [2, 15], development of multi-agent systems [8, 12, 14, 17, 44, 45, 55] and software patterns' specification and reuse [27, 28]. Figure 1 shows some semantic relationships between modeling concepts extracted from the ONTOMADEM ontology, a conceptualization of the MADEM methodology.

For the specification of the problem domain to be solved, MADEM focuses on modeling goals, roles, activities and interactions of entities of an organization.

Entities have knowledge and use it to exhibit autonomous behavior. An organization is composed of entities with general and specific goals that establish what the organization intends to reach. The achievement of specific goals allows reaching the general goal of the organization (*reached from* relationship of Fig. 1). For instance, an information system can have the general goal "satisfying the information needs of an organization" and the specific goals of "satisfying dynamic or long-term information needs". Specific goals are reached through the performance of responsibilities (*achieves* relationship of Fig. 1) that entities have by playing roles (*performed by*

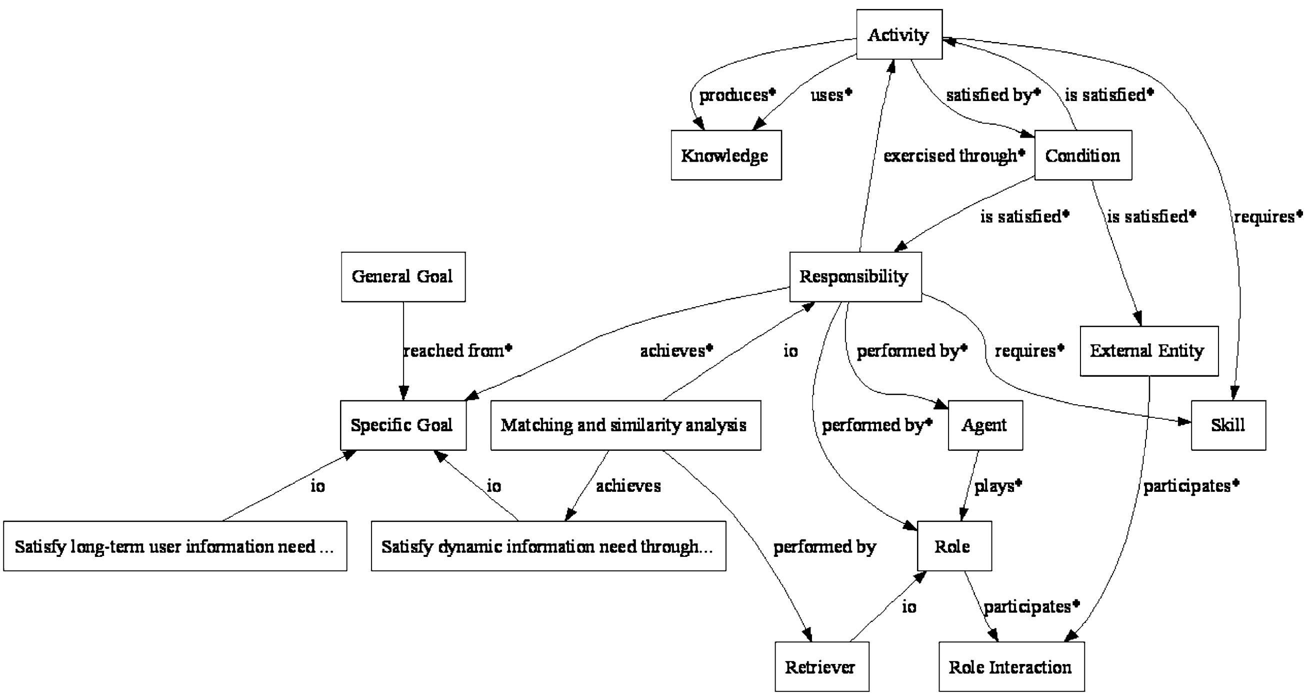

Fig. 1 Some relationships between modeling concepts and their instances in the ONTOMADEM ontology

relationship of Fig. 1) with a certain degree of autonomy. In the example of Fig. 1, the *retriever* role plays the *matching and similarity analysis* responsibility.

Responsibilities are exercised through the execution of activities (*exercised through* relationship of Fig. 1). The set of activities associated with a responsibility are a functional decomposition of it.

Roles have skills on one or a set of techniques that support the execution of responsibilities and activities in an effective way (*requires* relationship of Fig. 1). Preconditions and post-conditions may need to be satisfied for/after the execution of an activity (*is satisfied* and *satisfies* relationships of Fig. 1). Knowledge can be consumed and produced through the execution of an activity (*uses* and *produces* relationships of Fig. 1). For instance, an entity can play the role of "retriever" with the responsibility of executing activities to satisfy the dynamic information needs of an organization. Another entity can play the role of "filter" with the responsibility of executing activities to satisfy the long-term information needs of the organization. Skills can be, for instance, the rules of the organization that entities know to access and structure its information sources.

Sometimes, entities have to communicate with other internal or external entities to cooperate in the execution of an activity (*participates* relationship of Fig. 1). For instance, the entity playing the role of "filter" may need to interact with a user (external entity) to observe his/her behavior in order to infer his/her profile of information interests.

For the specification of a design solution, responsibilities of roles are assigned to agents (*plays* relationship of Fig. 1) structured and organized into a particular multi-agent architectural solution according to non-functional requirements.

Figure 2 illustrates the phases of the MADEM methodology in the context of a multi-agent domain engineering process, and Table 1 summarizes their modeling phases, respective tasks and modeling products. The framed concepts of Fig. 2 illustrate the phase and reusable products generated through the application of the GRAMO technique of MADEM used for the construction of the domain model described in this paper.

Domain analysis supported by the GRAMO technique approaches the construction of a domain model which specifies the current and future requirements of a family of applications in a domain by considering domain knowledge and development experiences extracted from domain specialists and applications already developed in the domain. Existing analysis patterns can also be reused in this modeling task.

Domain analysis is performed through the following modeling tasks: *modeling of domain concepts*, *goal modeling*, *role modeling*, *variability modeling* and *role interaction modeling* (Table 1). The product of this phase, a *domain model*, is obtained through the composition of the products constructed through these tasks: a *concept model*, a *goal model, a role model* and a set of *role interaction models*.

The *modeling of domain concepts* task aims at performing a brainstorming of concepts of the domain and their relationships representing them in a *concept model*. These concepts are refined in the subsequent modeling tasks.

The purpose of the *goal modeling* task is to identify the goals of the family of systems, the external entities with which it cooperates and the responsibilities needed to achieve them. Its product is a *goal model*, specifying the general and specific goals of the system family along with the external entities and responsibilities.

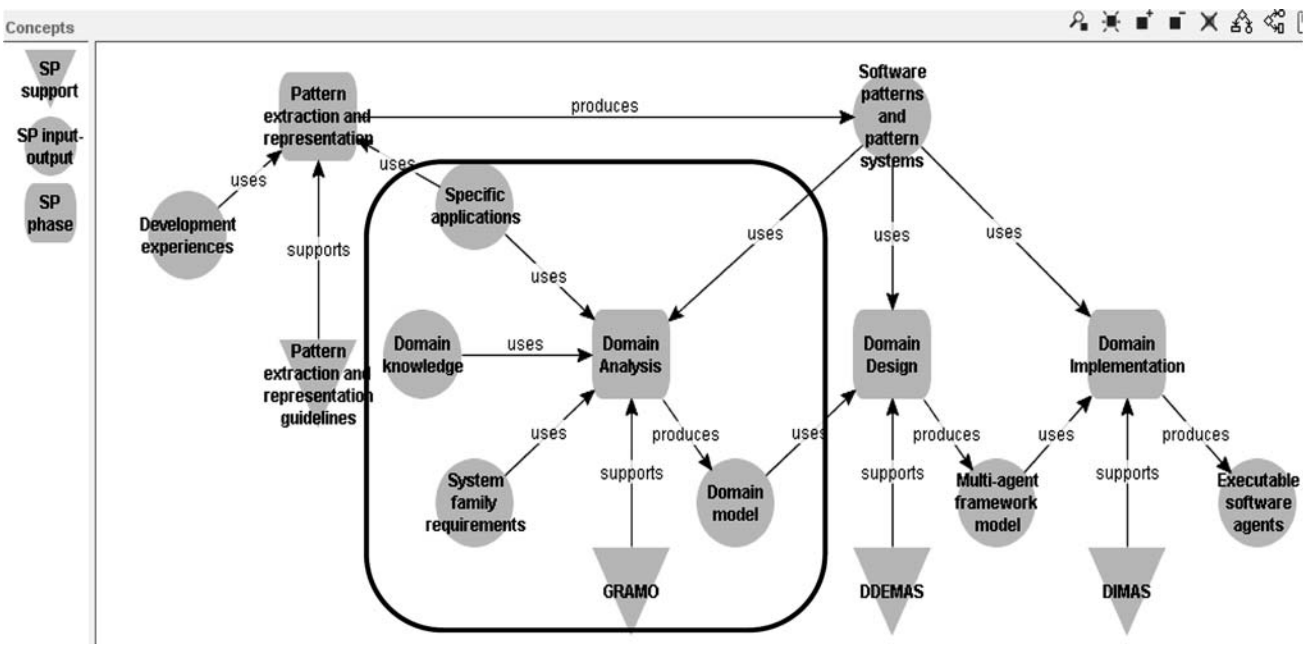

Fig. 2 The MADEM methodology in the context of the multi-agent domain engineering process

Table 1 Modeling phases, tasks and products of the MADEM methodology

<table>
<tr><th>Phases</th><th colspan="2">Tasks</th><th colspan="3">Products</th></tr>
<tr><td rowspan="4">Domain Analysis</td><td colspan="2">Modeling of domain concepts</td><td>Concept Model</td><td colspan="2" rowspan="4">Domain Model</td></tr>
<tr><td rowspan="2">Variability Modeling</td><td>Goal Modeling</td><td>Goal Model</td></tr>
<tr><td>Role Modeling</td><td>Role Model</td></tr>
<tr><td colspan="2">Modeling of Role Interactions</td><td>Role Interaction Models</td></tr>
<tr><td rowspan="6">Domain Design</td><td rowspan="3">Architectural Modeling</td><td>Agent Society Modeling</td><td>Multi-agent Society Model</td><td rowspan="3">Architectural Model</td><td rowspan="6">Model of the multi-agent framework</td></tr>
<tr><td>Agent Interaction Modeling</td><td>Agent Interaction Model</td></tr>
<tr><td>Cooperation and Coordination Modeling</td><td>Coordination and Cooperation Model</td></tr>
<tr><td colspan="2">Modeling the Knowledge of the Multi-agent Society</td><td colspan="2">Model of the Multi-agent Society Knowledge</td></tr>
<tr><td colspan="2" rowspan="2">Agent Modeling</td><td>Agent Knowledge and Activity Models</td><td rowspan="2">Agent Models</td></tr>
<tr><td>Agent State Models</td></tr>
<tr><td rowspan="2">Domain Implementation</td><td colspan="2">Mapping from design to implementation agents and behaviors</td><td colspan="2">Model of agents and behaviors</td><td rowspan="2">Implementation Model of the Multi-agent Society</td></tr>
<tr><td colspan="2">Mapping from agent interactions to communication acts</td><td colspan="2">Model of communication acts</td></tr>
<tr><td colspan="3">Pattern extraction and representation</td><td colspan="3">Software Patterns and Pattern Systems</td></tr>
</table>

The *role modeling* task associates the responsibilities identified in the *goal modeling* task to the roles that will be in charge of them. Responsibilities are decomposed into activities. The skills required for exercising a responsibility or activity and the pre- and post-conditions that must be satisfied before and after the execution of an activity or responsibility are also identified. Finally, the knowledge required from other entities (roles or external entities) for the execution of activity or responsibility and the knowledge produced from their execution are recognized. This task produces a *role model*, specifying roles, responsibilities; activities, skills, pre- and post-conditions, knowledge and relationships between these concepts.

The *variability modeling* task is performed simultaneously with the *goal and role modeling* ones. The purpose of this task is to classify goals, roles, responsibilities, activities and skills in *goal* and *role* models as common or variable features.

The *role interaction modeling* task aims at identifying how external and internal entities should cooperate to achieve a specific goal. For that, responsibilities and activities of roles are analyzed along with their required and produced knowledge specified in the *role model*. A set of *role interaction models* specifying the interactions between roles and external entities needed to achieve a specific goal is constructed as a product of this task.

The *goal* and *role models* provide a static view of the organization; the set of *interactions models*, a dynamic one.

Domain design supported by the DDEMAS technique approaches the architectural and detailed design of multi-agent frameworks providing a solution to the requirements of a family of multi-agent software systems specified in a domain model.

Domain implementation supported by the DIMAS technique approaches the mapping of design models to agents, behaviors and communication acts, concepts involved in the JADE framework [4], which is the adopted implementation platform. An *implementation model of the multi-agent society* is constructed as a product of this phase of MADEM, composed of a *model of agents and behaviors* and a *model of communication acts*.

3 The ONTOWUM problem domain

3.1 Information filtering and recommender systems

Information access can be characterized by the nature of both user information needs and sources of information. Particularly, information filtering techniques look for satisfying long-term information needs of a user or group of users [5] through the access to dynamic sources of information that can be non-structured, as the traditional Web, or semantically structured as the semantic Web.

One of the main applications of information filtering techniques is on the construction of recommender systems [1, 11, 18, 40]. These systems look for assisting the needs and behavior of heterogeneous users through the capture, representation and updating of their profiles, providing them with information items (e.g., Web pages hyperlinks) that satisfy those profiles, in the form of recommendations.

Information filtering techniques can be classified into three main groups: content-based (CBF), collaborative (CF) and hybrid (HF) [1].

In CBF, an item is recommended to a user if that item is similar to other items that he/she evaluated as relevant in the past. In these techniques comparisons between them are made and similarity is computed between the surrogates of information items and user's profiles. The user's profiles can be updated according to the evaluations made by the user on information items in the past. The hypothesis in this approach is that, if a user manifested interest for an information item in the past, he will prefer items with similar content in the future.

This approach has its roots in the area of information retrieval [3, 5]. In the traditional Web, CBF uses techniques developed in this area, as the vector space model, where profiles and information items are represented by vectors, whose elements are attributed weights of a particular interest or representative keyword of an information item. In the case of the semantic Web, the systematization of techniques for CBF [6, 16, 18] is object of current research.

In the approach of CF [32], the one currently considered in the ONTOWUM problem domain, the systems recommend information items to a user according to the similarity between his/her profile and the profiles of other users with similar interests that have manifested preference for those items in the past. The hypothesis in this approach is that, if a group of users manifested similar interests in the past, those users will agree in their preferences in the future. Any analysis of the content of information items is necessary to produce the recommendations.

Profiles representing user information needs and preferences can be acquired explicitly, for instance, through the completion of forms or explicit user evaluations of information items; or inferred implicitly, through the users' interactions with the system, for instance, using techniques of usage mining on access logs [13, 21, 39, 40, 47, 48].

The limitations of the techniques of CBF are overcome by the techniques of CF and vice-versa. For instance, the techniques of CBF do not have the ability to avoid the recommendation of items of "bad quality" while the techniques of CF can prevent those recommendations according to the criteria of quality of a certain group of users with similar interests. On the other hand, a requirement for the application of CF techniques is the existence of a great community of users and these techniques are not able to satisfy the interest of users of type "black sheep", i.e., those with dissimilar interests to the existent communities. These limitations do not exist in the CBF approaches. Therefore, the generation of effective recommendations requires the combination of both types of techniques, in the approaches known as hybrid (HF) [11].

The state of art of the techniques used in recommendation systems has been moving forward in the last years, improving considerably the effectiveness of these systems. The first generation of this type of systems used traditional techniques of the area of information retrieval for the representation of both user profiles and information items and the computation of the similarities between them [3, 5]. The new generation of recommender systems are based on the techniques of machine learning [42, 43, 51, 54] for pattern discovery used in the processes of Web usage mining, Web content mining and Web structure mining [35, 38, 41].

A new research area, known as semantic Web Mining [6, 52], has appeared in the context of the semantic Web research with two main goals:

- to improve the effectiveness of the results of the mining process by taking advantage of the semantic structure of the Web, and
- to develop mining techniques for automating at least partially the construction of the Semantic Web from the traditional Web.

3.2 Web usage mining

During the last years, researchers have proposed a new unifying area for all methods that apply data mining to Web data, named Web mining [35]. Web mining aims at extracting knowledge from the Web, rather than just retrieving information. Web mining is traditionally classified into three main categories: *Web content mining*, *Web usage mining* and *Web structure mining*, according to the part of the Web to be mined. Web content mining approaches the extraction of useful knowledge from the content of Web pages. This category is usually associated with text mining. Web structure mining approaches the extraction of useful knowledge from the implicit structure between pages following their hyperlinks. Finally, Web usage mining [13, 18, 21, 47, 48] aims at discovering interesting patterns of use by analyzing Web usage data. Usage data, such as those that can be collected when a user browses a specific Web site, represent the interactions between the user and that particular Web site.

Web usage mining provides an approach to the collection and pre-processing of those data, and the construction of models representing the behavior and the interests of users. These models can provide insights of how users behave when browsing a particular site. In this way, site administrators can optimize not only the site structure but also the offered services. These models can be automatically incorporated into personalization components, without the intervention of any human expert. Figure 3 shows the Web usage mining overall process (adapted from [47]). The phases of the process are briefly described below. A detailed description of these phases and applied techniques is provided in [21, 47].

- *Data collection*: In this phase, usage data from various sources are gathered and their content and structure are identified. Data are collected from Web servers, from clients that connect to a server, or from intermediary sources such as proxy servers and packet sniffers.
- *Data pre-processing*: In this phase, data are cleaned from noise, their inconsistencies are resolved, and they are integrated so as to be used as input to the pattern discovery stage. This involves primarily data cleaning, user identification and user session identification.
- *Pattern discovery*: In this phase, knowledge is discovered through the application of machine learning and statistical techniques, such as clustering, classification, association discovery, and sequential pattern discovery [54]. The patterns required for Web personalization correspond to the behavior and interests of users. This is the stage where the learning methods are applied in order to automate the construction and automatic maintenance of user models.
- *Knowledge post-processing*: In this last stage, the extracted knowledge is analyzed aiming to support the decision-making process of human experts. For that, the usage mining system can provide reports containing results of statistical analysis of the discovered patterns [56]; visualization tools that present navigational patterns as graphs [53] or query mechanisms, which are used for the extraction of rules from navigational patterns [13]. Sometimes, these decisions may lead to the personalization of Web services, where the extracted knowledge is incorporated in a personalization module to facilitate the personalization functions, like the extraction of recommendations. The following section discusses the relationship between Web usage mining and Web personalization.

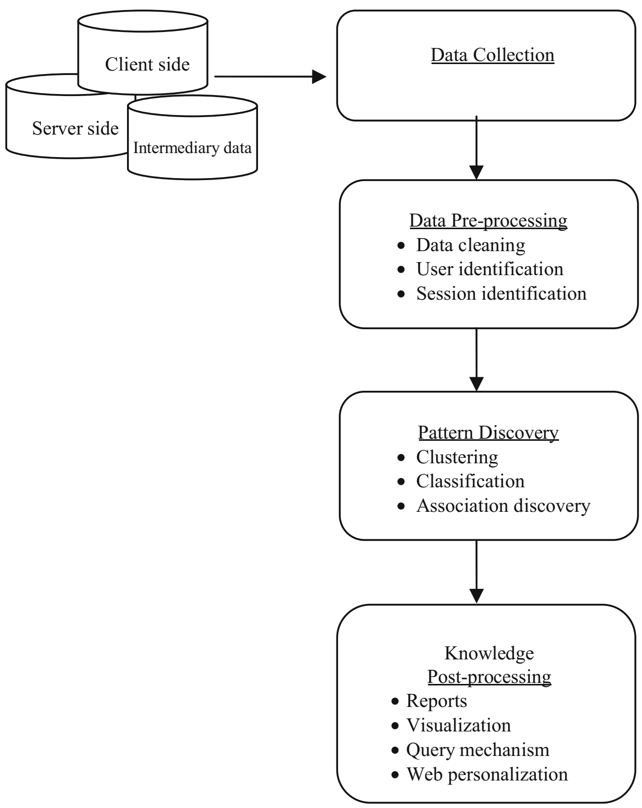

Fig. 3 A Web usage mining process (adapted from [47])

3.3 Web personalization and Web usage mining

Originally, the aim of Web usage mining has been to support the human decision making process. Thus, the outcome of the process is typically a set of data models that reveal implicit knowledge about usage patterns of users. These models are evaluated and exploited by human experts, such as a market analyst who seeks business intelligence, or a site administrator who wants to optimize the structure of a site and enhance the browsing experience of visitors [47].

Although most of the work in Web usage mining is not concerned with personalization, its relationship to personalization issues has brought promising results. Web personalization is a sub-area of adaptive hypermedia that aims at providing actions that tailors the Web experience to a particular user, or a set of users [9]. This process is composed of two components [40]: an off-line component, responsible for the data preparation and specific usage mining tasks and an on-line component, responsible for the user data collection and the generation of personalized services to users, based both on the discovered patterns and their current browsing activities. The knowledge discovered through the usage mining process serves as operational knowledge to personalization systems.

A Web personalization system can offer a great diversity of functions, ranging from simple salutation to more complex ones such as the adaptation of user interfaces [9]. Pierrakos et al. [47] propose a generic classification scheme for Web personalization functions. The proposed scheme takes into account what is currently offered by commercial systems and research prototypes. Four basic classes of personalization functions were identified in this scheme: memorization, guidance, customization and task performance support. These classes are briefly described below. A detailed description of this topic is provided in [20, 40].

- *Memorization*: This is the simplest form of a personalization function, where the system records and stores information about a user, such as name and browsing history. The system could recognize the returning user and display its name together with a welcome message, for example [47].
- *Guidance*: Guidance refers to a personalization function responsible for assisting the user in getting quickly the information he or she is seeking in a site, as well as to provide the user with alternative browsing options. This function alleviates to a great extent the information overload problem that the users of a large Web site may face. For example, the system could recommend a set of hyperlinks that are related to the interests and preferences of the user.
- *Customization*: Customization refers to a personalization function responsible for the modification of a Web page in terms of content, structure and layout, in order to take into account the user's knowledge, preferences and interests. This function is usually related to the research area known as adaptive user interfaces [9].
- *Task performance support*: The task performance support function refers to the execution of actions on behalf of the user. This is the most advanced personalization function, inherited from a category of adaptive systems known as personal assistants [9], which can be considered as client-side personalization systems.

3.4 A first description of the functional and non-functional requirements of the ONTOWUM family

According to the concepts introduced in Sects. 3.1, 3.2 and 3.3, a first description of the functional and non-functional requirements of the ONTOWUM family can be done.

Non-functional requirements relate to implementation issues. An environment supporting the implementation and execution of multi-agent systems should be provided.

On the other hand, recommender systems based on usage mining and collaborative filtering should provide functionality for:

1. Construction and update of user models of individual users which involves:
   - The capture of individual user profiles requiring the monitoring and extraction of the browsing session of the current user from the interactions of a user with a Web page. This functionality corresponds to the *data collection* phase of the usage mining process of Fig. 3.
   - The representation and updating of individual user profiles requiring the creation and updating of a model of the current user, a formal representation of the browsing session of the current user. This functionality corresponds to the *data pre-processing* phase of the usage mining process of Fig. 3.
2. Construction and update of models of groups of users with similar interests which involves:
   - The construction and update of a repository of usage data including the navigational behavior of users. This functionality corresponds to the *data pre-processing* phase of the usage mining process of Fig. 3.
   - The mining of pre-processed usage data to discover, and represent in a model, patterns revealing the past browsing behavior of users with similar interests. This functionality corresponds to the *Pattern Discovery* phase of the usage mining process of Fig. 3.
3. *Construction of personalized recommendations*, functionality corresponding to the *knowledge post-processing* phase of the usage mining process of Fig. 3, which involves:
   - The classification of the model of an individual user into an appropriate model of groups of users with similar interests;
   - The creation and update of an *adaptation model* specifying the adaptation rules to be applied on the interface of the current user according to the groups of users to which the current user belongs.
   - The customization of the user interface with the recommendations according to the rules of the adaptation model.

4 A case study on domain analysis of Web recommender systems

This section describes a case study on the application of the GRAMO technique of the MADEM methodology for the construction of the ONTOWUM domain model which specifies the common and variable requirements of a family of recommender systems using techniques of collaborative filtering and usage mining.

Next sections detail the tasks and generated products of the domain analysis phase of MADEM listed in Table 1 performed for the construction of the domain model.

4.1 Modeling of domain concepts

The purpose of this task is to develop a *concept model* representing the concepts of the domain and the relationships between them. A semantic network is developed where nodes represent concepts and links show the relationships between concepts.

Concepts in a *concept model* suggest main instances of modeling concepts to be captured in the subsequent modeling tasks of *domain analysis* and specified in *goal, role* and *role interaction models*. Moreover, *modeling of domain concepts* aims mainly at representing a brainstorming of concepts and relationships between them that are refined in the subsequent modeling tasks.

Figure 4 shows the *concept model* of the ONTOWUM *domain model*, summarizing main concepts involved in a recommender system based on usage mining and collaborative filtering techniques.

The navigational behavior of a user is captured by monitoring the user access to Web pages. A user model representing a user profile is dynamically constructed and updated based on his/her navigational behavior. A repository of usage data is constructed and updated including the navigational behavior of users. This repository is mined to discover usage patterns and to construct models of groups of users with similar interests. By classifying the current user in the appropriate group, an adaptation model can be constructed which is then used to construct personalized recommendations.

4.2 Goal modeling

The purpose of this task is to identify the goals of the family of systems and the external entities with which it cooperates to achieve them. By considering the problem that the system intends to solve, the general goal of the system is identified. Specific goals are obtained, first, through a decomposition of the general goal, considering the sub-problems into which the problem can be decomposed. Then, specific goals can be refined in a hierarchy of specific goals by introducing different problem solving approaches. The relationships between external entities and goals in the lowest level of refinement are then established. The responsibilities that need to be exercised to reach a specific goal in the lowest level of refinement are identified and related to the corresponding goal. A *goal model* consists of a general goal and a set of specific goals, the external entities which cooperate with the system and the responsibilities of the family of systems. The *goal model* is represented graphically in an n-level organizational chart. The general goal and the responsibilities are represented in the first and last level, respectively. The hierarchy of specific goals is disposed in the other levels along with the external entities in the lowest level of goal refinement. Non-functional requirements can also be associated with general and specific goals.

Figure 5 shows the *goal model* of the ONTOWUM *domain model*. By considering the general problem approached, *Provide collaborative recommendations through Usage Mining*, the general goal is identified. This main problem can be decomposed into two main sub-problems: *model users through usage mining* and *model adaptation through usage mining* specific goals. The *model users through usage mining* specific goal can be achieved through the performance of the following responsibilities:

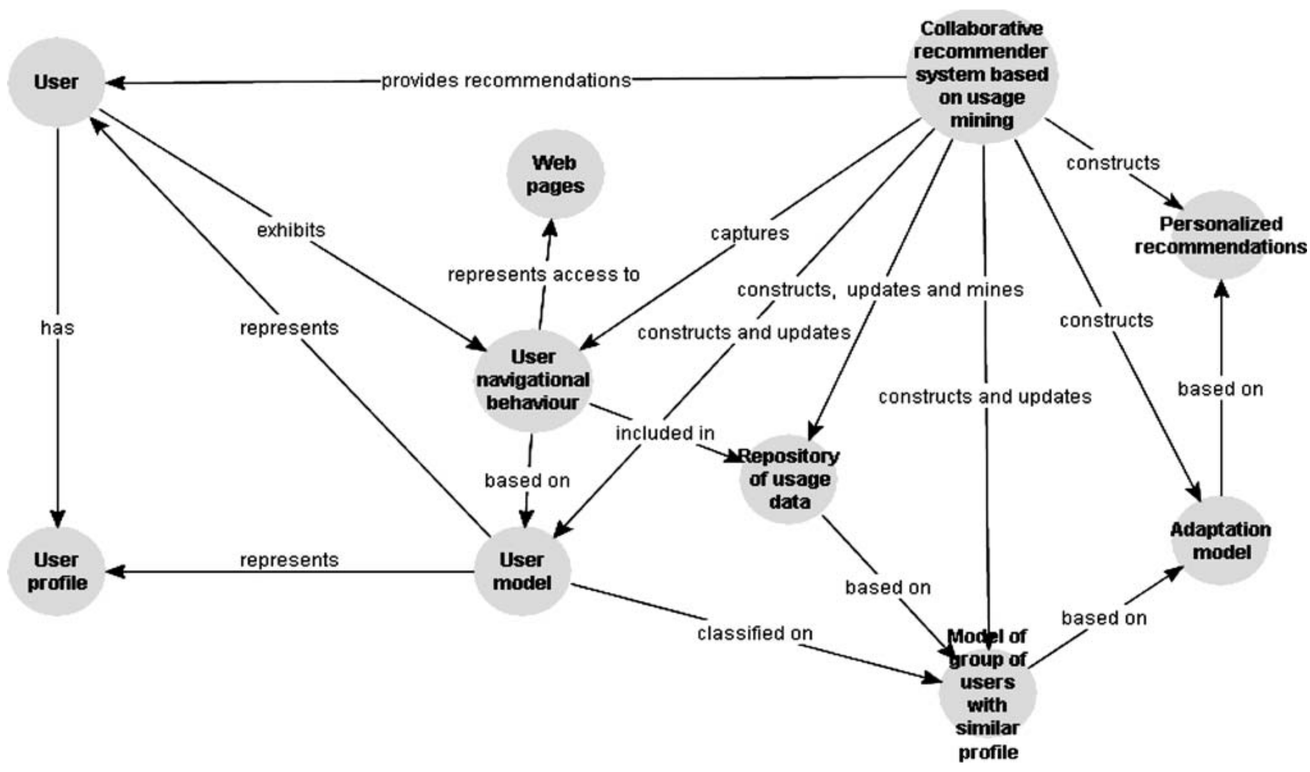

Fig. 4 The concept model of the ONTOWUM domain model

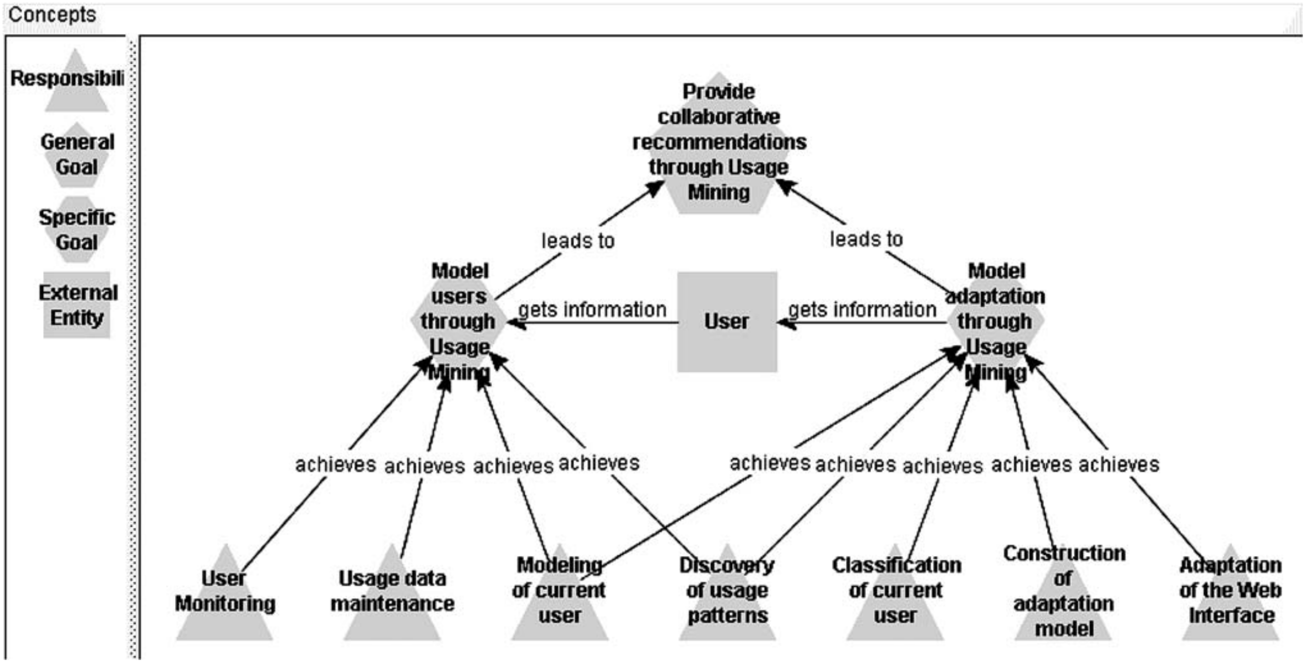

Fig. 5 Goal model of the ONTOWUM domain model

- User monitoring,
- Usage data maintenance,
- Modeling of current user, and
- Discovery of usage patterns.

The *Model adaptation through usage mining* specific goal can be achieved through the following ones:

- Modeling of current user,
- Classification of current user,
- Discovery of usage patterns,
- Construction of adaptation model, and
- Adaptation of the Web Interface.

The system requires information from a *user* to achieve the *model users through usage mining* specific goal; and a *user* gets information through the achievement of the *model adaptation through usage mining* specific goal.

The *user monitoring* responsibility performs the on-line monitoring of the browsing activities of a current user to capture the information needed for the execution of the *modeling of current user* responsibility to create a user model representing the browsing session of a current user. The execution of this last responsibility is required to achieve both the specific goals *model users through usage mining* and *model adaptation through usage mining*, considering that a model of the current user is needed to create an adaptation model.

The *user monitoring* responsibility corresponds to the *data collection* phase of the usage mining process of Fig. 3.

Through the execution of the *usage data maintenance* responsibility, a repository of usage data is updated with the model of the current user.

The *usage data maintenance* and the *modeling of current user* responsibilities correspond to the *data pre-processing* phase of the usage mining process of Fig. 3.

The discovery of usage patterns responsibility is in charge of performing the mining of the pre-processed usage data to discover, and represent in a model, patterns revealing the past browsing behavior of users. The execution of this responsibility is required to achieve both the specific goals *model users through usage mining* and *model adaptation through usage mining*, considering that a model of the past browsing behavior of users is needed to create an adaptation model. Through the execution of the *classification of current user* responsibility, the model of a current user is classified in one or more groups of users with similar interests. This is a collaborative approach frequently used in recommender systems [32]. Thus, recommendations for a user can be done according to the pages browsed by members belonging to the group where the current user was classified.

The *construction of adaptation model* responsibility constructs a model with the personalization rules to be applied in the user interface by the *adaptation of the Web interface* responsibility.

The *classification of current user*, *construction of adaptation model*, and *adaptation of the Web interface* responsibilities correspond to the knowledge post-processing phase of the usage mining process of Fig. 3.

4.3 Role modeling

The purpose of this task is to represent the behavior of all roles of the system along with the knowledge provided by other entities (roles and external entities); skills and pre-conditions required to exhibit it; and post-conditions satisfied through this behavior. The behavior of a role is represented through responsibilities and activities. Each responsibility identified through the *goal modeling* task is associated with a role. The responsibility fulfilled by a role can be decomposed into a set of activities. Then, the knowledge, skills, pre-conditions required for the execution of each responsibility or activity and the satisfied post-conditions are identified. The selection of a particular technique to be used as a skill for the execution of a responsibility is a design decision specified and detailed in the domain design phase of the multi-agent domain engineering process. In the domain analysis phase, known skills are listed and briefly commented. In the case that there is no need to decompose a responsibility into activities, then knowledge, skills, and pre-conditions are associated with the responsibility itself. A *role model* is represented graphically in a three level organizational chart. *Roles*, *external entities* and *skills* are represented in the first and third level, respectively. *Responsibilities*, *knowledge*, *pre- and post-conditions* are represented in the second level.

Figures 6 and 7 show the *role model* of the ONTOWUM *domain model*. From the responsibilities identified in the goal modeling task, the roles of the system (*user monitor*, *user modeler*, *acquirer*, *miner*, *classifier*, *customizer* and *interface*) are defined.

4.3.1 The user monitor role

The *user monitor* role plays the *user monitoring* responsibility which involves the extraction of the *browsing session of the current user* (produced knowledge), containing information like the URL of a visited page and the time spent on that page from *the interactions of a user with a Web page* (knowledge provided by a *User*). The pre-condition for the execution of this responsibility is *a new user connected* (condition satisfied by a user). Appropriate techniques (skills) for capturing the *information about the user browsing activities* should be used to perform this responsibility. Shahabi et al. [49], for example, proposes an approach for client side data acquisition through Java applet remote-agents.

4.3.2 The user modeler role

The *user modeler* role plays the *modeling of current user* responsibility which involves the creation and updating of a *model of the current user* (produced knowledge), a formal representation of the *browsing session of the current user* (knowledge of the *user monitor* role). Appropriate techniques (skills) for constructing and updating the *model of the current user* should be used to perform this responsibility, for instance, techniques based on weighted vectors of the vector space model [40], where the visited URLs are features of a vector quantified by some browsing feature, for example, like the time a user spend viewing a Web page. In [39] the

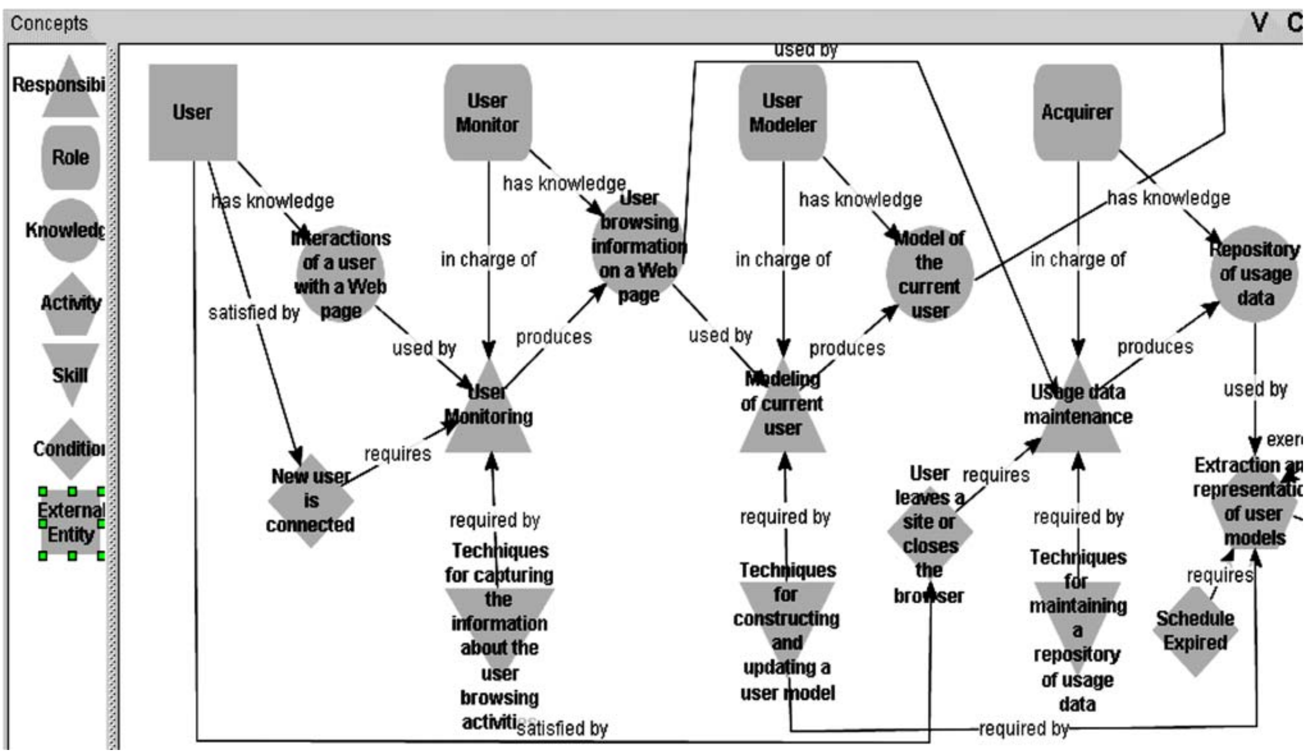

Fig. 6 Part of role model of the ONTOWUM domain model showing the *user monitor, user modeler* and *acquirer* roles

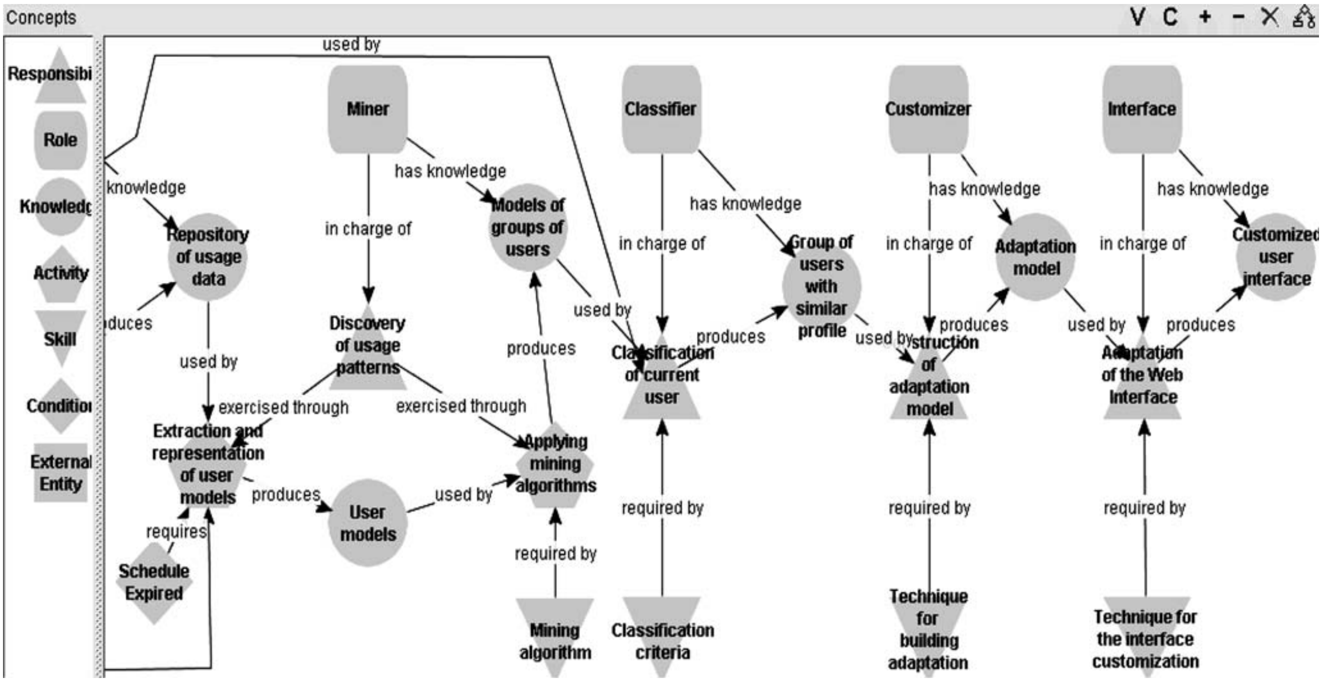

Fig. 7 Part of role model of the ONTOWUM domain model showing the *miner, classifier, customizer* and *interface* roles

Feature Matrices model [48] was used as the technique to formally represent the user models of the system. In this model, features like visited URLs and time of page view are indicators of the information embedded in sessions. Sessions can be seen as the path traversed by a user while navigating a site.

4.3.3 The acquirer role

The *acquirer* role plays the *usage data maintenance* responsibility which involves the creation and update of a *repository of usage date* (produced knowledge), from the *model of the current user* (knowledge of the *User Modeler* role). Appropriate techniques (skills) for maintaining a *repository of usage date* should be used to perform this activity, like remote components responsible for acquiring and handling user models that can arrive from many users. In [39], an *acquirer* agent is responsible for such task. When a user model containing the closed browsing session of a user arrives, the *acquirer* agent formats and stores this model in RDF (*resource description framework*) [36] format. In this approach, the user model, which in this case is based on usage data, is mapped to RDF semantic graphs.

The pre-condition *user leaves a site or closes the browser* must be satisfied for the execution of the *usage data maintenance* responsibility.

4.3.4 The miner role

The *miner* role plays the *discovery of usage patterns* responsibility which involves the identification and representation of *models of groups of users* with similar browsing behavior (produced knowledge). This responsibility is decomposed into two activities: *extraction and representation of user models*, and *applying mining algorithms*. The *representing user session* activity identifies the user sessions from the *repository of usage data* (knowledge of the *Acquirer* role) and represents them in a set of *user models* by using similar skills to the ones introduced for the *user modeler* role. The *applying mining algorithms* activity mines the *user models* and produces *models of groups of users* by using an appropriate *miner technique*, like clustering, association rules discovery, classification and sequential patterns [47, 54]. In [39], for instance, the *KMeans* clustering algorithm [51] is used for the clustering of users with similar profiles. This algorithm is responsible for the formation of clusters in numerical domains, and for partitioning instances (in this case, user models) into disjoint clusters.

The *miner* role should work periodically according to a pre-defined schedule.

A pre-condition for the execution of the extraction and representation of user models responsibility is a schedule expired.

4.3.5 The classifier role

The *classifier* role plays the *classification of current user* responsibility which involves the classification of the *model of current user* (knowledge of the *user modeler* role) in the *models of groups of users* discovered by the *miner* role, thus determining the *groups of users to which the current user belongs* by using an appropriate *classification criteria* (skill). These criteria can be, for example, the distance or degree of similarity between the model of the current user and the groups discovered by the *miner role*. In [39] the use of a *dynamic clustering* algorithm for the classification of the current user model in one of the groups discovered by the *miner role* is specified.

4.3.6 The customizer role

The *customizer* role plays the *construction of adaptation model* responsibility which involves the creation and update of an *adaptation model* (produced knowledge) specifying the adaptation rules to be applied on the interface of a current user according to the *groups of users to which the current user belongs*. A user can be classified several times into different groups of users during his/her browsing session. Therefore, his/her adaptation model needs to be updated to reflect his/her behavioral changes. An appropriate technique (skill) for building adaptation models must be used. These techniques can be based, for example, in manual decision rule systems, content-based filtering systems, and social or collaborative filtering [32].

In [39] the use of a collaborative approach for the construction of the adaptation model is specified. In this approach, the browsing behavior of the current user is matched with groups of other users with similar profiles. Thus, the adaptation is done according to the profiles of other similar users. As the profiles are derived from the browsing behavior of the users, it will be represented, basically, through sequences of visited URLs. So, when the current user is matched with other users with similar profiles, a set of links is recommended based on pages that other users have visited.

4.3.7 The interface role

The *interface* role plays the *adaptation of the Web interface* responsibility which involves the creation and update of a *customized user interface* according to the *adaptation model*. An appropriate technique for interface customization must be used. The customization can be based on a set of recommended links. The presentation of the recommended links is usually done either in a separate frame of the Web page or in a pop-up window. In [39] the use of an invisible HTML layer in each page of the site is specified , and as soon as there are new recommendations, the layer becomes visible showing the current recommendations to the user.

4.4 Variability modeling

The purpose of this task is to classify the instances of the modeling concepts (goals, roles, responsibilities, activities and skills) as common or variable features. Common features will be present in all subsystems of a family of systems in a knowledge or problem-solving area. Variable features will be present only in some systems of the family.

Variability modeling is performed simultaneously in the construction of goal and role models and is based on traditional feature modeling [15, 33]. A feature is a prominent or distinctive user-visible aspect, quality, or characteristic of a software product like functional features of individual products in the system family. In MADEM, the feature model is embedded in the goal and role models and considered features are the modeling concepts *specific goals*, *responsibilities*, *activities* and *skills* which can be *mandatory*, *alternative* or *optional*. Following we describe some examples of variability.

4.4.1 Specific goal variability

A particular system in the family does not need to satisfy, necessary, all the specific goals. That is, the general goal can be satisfied by alternative or optional specific goals which can represent alternative or optional problem-solving approaches to the general problem of the family of systems. For instance, a family of applications for information retrieval and filtering can have the general goal of “satisfy user information needs”. The two specific goals “satisfy dynamic user information needs” and “satisfy long-term user information needs” lead to the achievement of the general goal. Both goals will be represented as *mandatory* if all systems of the family provide information retrieval and filtering capabilities. However, they can be considered as alternative specific goals if some systems of the family support only one of the two types of information access capabilities.

Moreover, the specific goal “satisfy long-term user information needs”, for instance, can be considered as an optional feature and “satisfy dynamic user information needs” as a mandatory one. Thus, all systems in the family would provide information retrieval capabilities and some of them can, at demand, support information filtering ones.

In the example of Fig. 5, all goals are represented as mandatory features because the achievement of all of them is required to achieve the general goal of the organization.

4.4.2 Responsibility variability

Responsibilities are alternative if the exercise of anyone allows the achievement of a specific goal. For instance, the responsibility “capturing implicitly a user profile” is alternative to the responsibility “capturing explicitly a user profile”. All of them would contribute to the achievement of a specific goal like “model user profiles”. A responsibility can be also optional to a mandatory or alternative one if they provide an optional capability that is not necessarily required to achieve the specific goal. In the example of Fig. 5, all responsibilities are represented as *mandatory* because the exercise of all of them is required to achieve the corresponding specific goals.

4.4.3 *Activity variability*

Since activities are different forms of decomposing a responsibility, a group of activities performed to exercise a responsibility can be alternative to another group derived from other decomposition. Also, a group of one or more activities can be optional to a group of one or more mandatory ones. For instance, the activity ''process payment with credit card'' performed in the exercise of the responsibility ''process order'' is optional to the one of ''process payment''. In the example of Fig. 7, all activities are represented as mandatory features because only one decomposition has been found for the responsibility of *discovery of usage patterns* and none to the other responsibilities.

4.4.4 *Skill variability*

For performing responsibilities and activities, roles have a set of skills that can be mandatory, optional or alternative to other ones. Figure 8 shows an example of an alternative skill, illustrating the different mining algorithms that the *miner* role can use for exercising the responsibility of discovery of usage patterns: clustering, association rules, classification, and sequential pattern discovery.

4.5 Modeling of role interactions

The purpose of this task is to identify the interactions between roles and between roles and external entities through an analysis of their respective responsibilities and activities along with their required and produced knowledge specified in the *role model*. Considered roles are those whose responsibilities lead to the achievement of a specific goal. One *role interaction model* for each specific goal is constructed as a product of this modeling task.

The graphical representation of a *role interaction model* is similar to the collaboration diagram of UML [10]. Figures 9 and 10 show the *role interaction models* of the ONTOWUM *domain model* describing the interactions needed in the multi-agent society to achieve its specific goals. For achieving the *model users through usage mining*

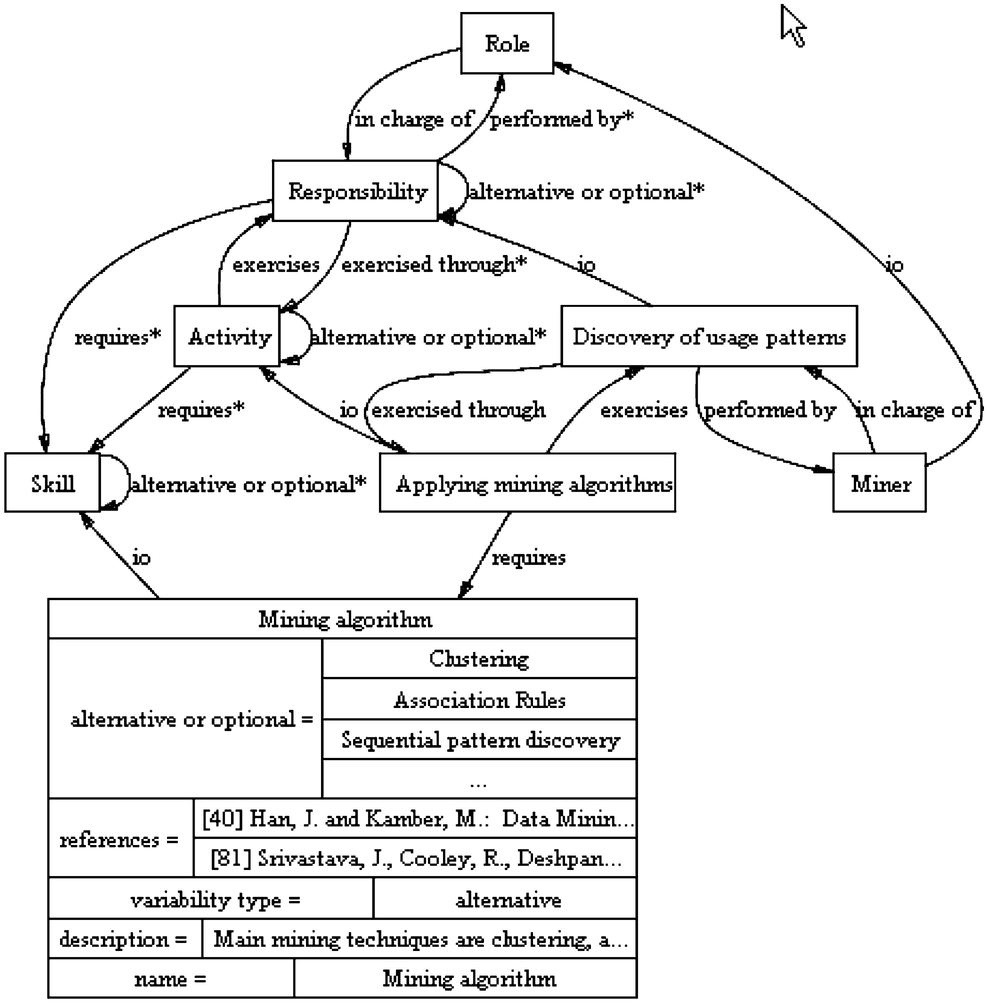

Fig. 8 An example of representation of an alternative skill in ONTOMADEM

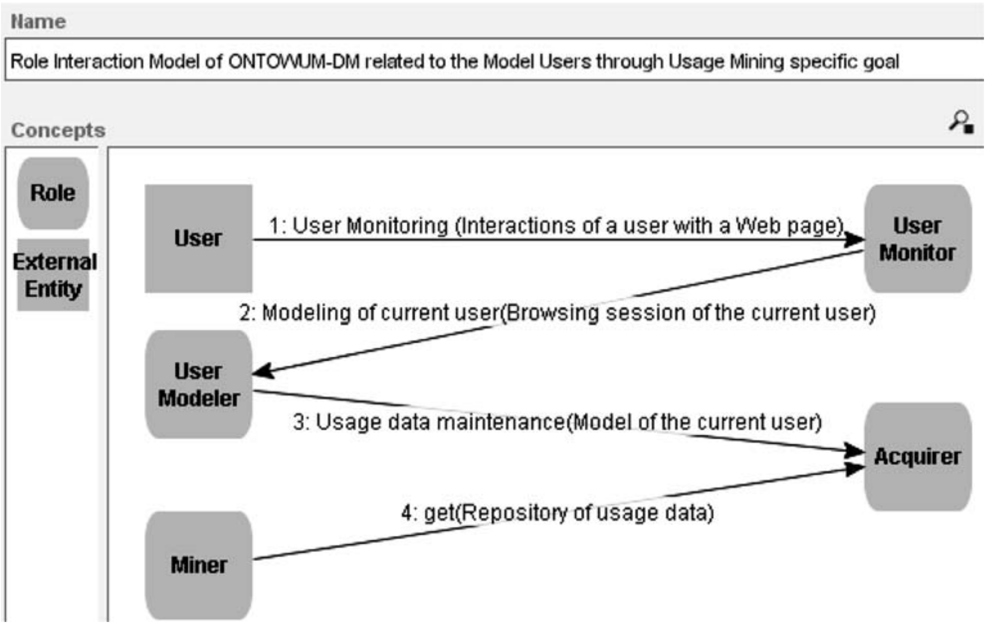

Fig. 9 Role interaction model of ONTOWUM domain model related to the model users through usage mining specific goal

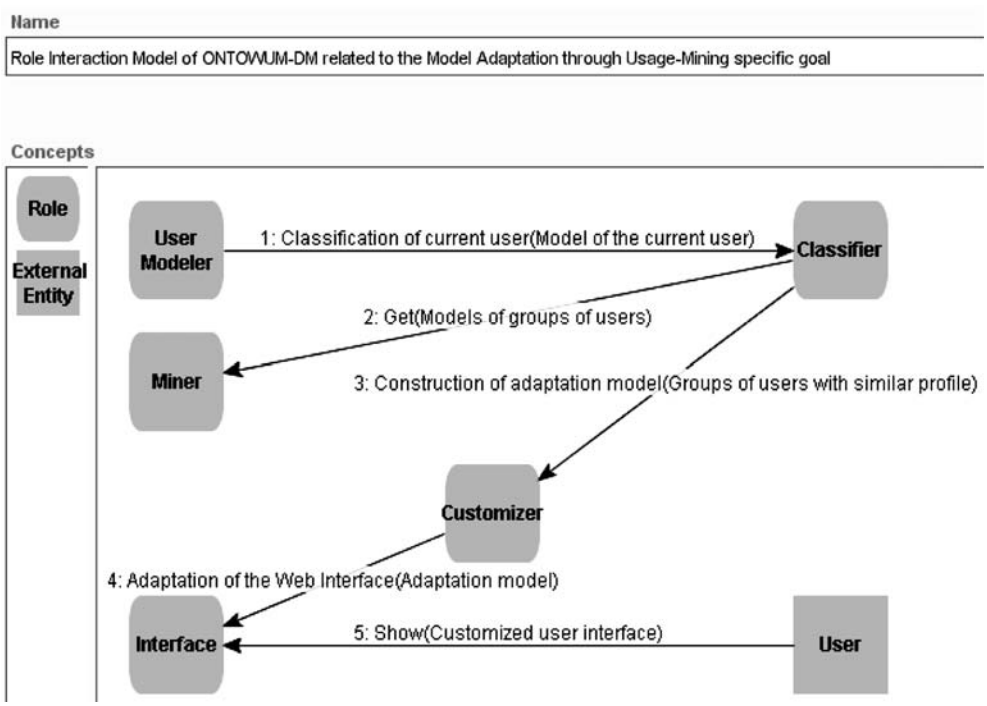

Fig. 10 Role interaction model of ONTOWUM domain model related to the model adaptation through usage-mining specific goal

specific goal, the *user monitor* role should capture the browsing behavior of a user through his/her interactions with a Web page and provide the *user modeler* role a representation of the browsing session of the current user for the construction of a model of the current user. This model is used by the *acquirer* role to update a repository of usage data which is periodically consulted by the *miner* role for the identification and representation of groups of users with similar browsing behavior (Fig. 9).

For achieving the *model adaptation through usage mining* specific goal, the *user modeler* role provides the *classifier* role with a model of the current user. The *classifier* role will ask the *miner role* for the models of groups of users and identify to which group of users the current one belongs. Using this group, the *customizer* role constructs an adaptation model and asks the *interface* role to build and display to the user a customized interface (Fig. 10).

5 Related work

5.1 Applications of Web usage mining for Web personalization

Many applications of usage mining on the development of Web personalization systems have been published. Some of them are following briefly described. WebPersonalizer [40] is a research prototype that uses clustering techniques for the automatic discovery of aggregated user profiles. The system generates recommendations based on the proximity between the current user profile and the previously discovered profiles. In [43], an approach for Web personalization based on a dynamic clustering technique is also proposed. WUM [50] is a system that offers multi-user customization functionality by modifying the hyperlinks of a particular Web page to include links to pages that have been visited by customers and not by non-customers. The modified Web pages are presented once in a session, to non-customers, with the ultimate goal of turning them into customers. ROSA (remote open site agents) [34] is a multi-agent architecture that facilitates integration of different mining methods and permits the discovered knowledge to be verified and updated automatically.

Besides usage data, other complementary sources or channels can be exploited to improve the quality of the patterns discovered by Web usage mining, such as Web content, Web structure, and even special domain ontologies [38]. Mobasher et al. [40] introduces a general framework for the automatic personalization based on Web usage and content mining. In this work, the user preferences are automatically learned from Web usage data and integrated with domain knowledge and the site content. This has the potential of eliminating subjectivity from profile data as well as keeping it up-to-date. Furthermore, it was experimentally showed that the integration of usage and content mining increases the usefulness and accuracy of the resulting recommendations. Li and Zaiane [38] propose an approach for building Web recommender systems where three information channels are exploited: web access logs, the structure of a visited web site, and the content of visited web pages. The terms within visited web pages are used to partition visit sessions into overlapping sub-sessions, called missions. Their preliminary experiments show improvements on the quality of recommendations by considering content and connectivity of Web pages in addition to usage history.

Usage patterns discovered through Web usage mining are effective in capturing item-to-item and user-to-user relationships and similarities at the level of user sessions. Without the benefit of deeper domain knowledge, such patterns provide little insight into the underlying reasons for which such items or users are grouped together. In Dai et al. [16] a general framework for using domain ontologies to automatically characterize user profiles containing a set of structured Web objects is introduced. In this work, a Web site is considered as a collection of objects belonging to certain classes. Given a collection of similar user sessions, obtained, for instance, through clustering, each containing a set of objects, it has been shown how to create an aggregate representation of the whole collection based on the attributes of each object as defined in the domain ontology. This aggregate representation is a set of pseudo objects characterizing objects of different types commonly occurring across the user sessions.

5.2 Web usage mining and the semantic Web

A new class of Web usage mining methods and techniques has appeared in the context of the semantic Web, an approach for enriching the Web with machine-processable information with meaning organized on different abstraction levels [37]. This new area of research is being named Semantic Web Mining [6, 52]. The idea of Semantic Web Mining is to improve the results of web mining by exploiting the semantic structures of the Web, as well as to use web mining to help building the semantic Web.

6 Concluding remarks

6.1 Analysis of results and main contributions

This work contributes with an ontology-driven domain model for usage mining in the context of agent-based Web recommender systems. This model represents the common and variable requirements of ONTOWUM, a family of multi-agent systems to be developed in this problem-solving area.

The case study described in this article has also contributed to the evaluation of GRAMO. The ONTOWUM domain model was specified with GRAMO [24], a technique of the MADEM methodology for the construction of ontology-based domain models, through the instantiation of ONTOMADEM, an ontology-driven tool which guides the elicitation and specification of the concepts and products in a knowledge or problem-solving area.

This domain model has been used by DDEMAS [26], a technique of the MADEM methodology for domain design through which a functional multi-agent framework of ONTOWUM has been developed. This framework comprises two layers where four agents are distributed according to their responsibilities: an *interface* agent, responsible for both capturing the user browsing information and executing the adaptation effects; a *user modeling* agent, responsible for creating and updating both a user model and an adaptation model of the current user; an *acquirer* agent, responsible for creating and updating a repository of usage data containing models of users that have been accessed in the system previously; and a *miner* agent, responsible for both discovering group of users with similar browsing behavior and classifying the current user in these groups. The *interface* and *user modeling* agents are deployed on a *user information-processing layer*, and the *acquirer* and *miner* agents in the *pattern-discovering* one.

The GRAMO technique, and ONTOMADEM, the tool supporting it, have shown their usefulness for capturing and specifying the requirements of multi-agent applications through appropriate guidelines, representation and decomposition mechanisms able to cope with the complexity of a domain, as the one of usage mining and collaborative filtering for Web personalization approached in this article.

The ONTOWUM reusable artifacts are part of the ONTOMADEM knowledge base where concepts are semantically related and where inferences can be made thus facilitating the understanding and reuse of the common and variable requirements of a family of multi-agent applications.

6.2 The ONTOWUM domain model, the GRAMO technique, the MADEM methodology and related approaches

As described in Sect. 3.3, many applications of Web usage mining for Web personalization already exist. However, there is a lack of high-level software abstractions of these developments, including requirement models, formal or even informally specified, able to be reused on the construction of new applications in this area. Therefore, the ONTOWUM domain model constitutes an original approach to attend the increasing demand of recommender systems development.

For the GRAMO technique and the MADEM methodology, several techniques and methodologies for development of multi-agent systems [8, 12, 14, 17, 44, 45, 55] have influenced in different aspects the conception of them. Two main features distinguish MADEM from other existing approaches. First, it provides support for the construction of reusable agent-based software artifacts, and second, it is a knowledge-based technique where software artifacts are included in the ONTOMADEM knowledge-based repository.

On the other hand, some prototypes of knowledge-based tools and environments, like ODYSSEY [7] and ODE [22], have been already developed to increase the productivity of the software development process, the reusability of generated products, and the effectiveness of project management. One main characteristic distinguishing ONTOMADEM from these approaches is its reuse support for agent-oriented software development.

6.3 Current limitations of the ONTOWUM domain model and further work

The variability part of the ONTOWUM domain model is currently limited. Several extensions are being approached. First, this model is being generalized and refined to include responsibilities and roles for complementing Web usage mining with other Web mining categories, such as Web structure mining and Web content mining [38]. Also, we are working on the extension of the domain model to include support for user modeling using different machine-learning techniques [42, 54].

The domain model and framework of ONTOWUM have been reused in the development of a recommender system for the traditional Web in the tourism domain [46] and a recommender system for the semantic Web in the legal domain [18]. On the basis of this last experience and on the research we are conducting on the semantic Web, we plan to generalize these results and use MADEM in the construction of a domain model and framework for the development of semantic Web recommender systems.

ONTOWUM is also being used by TOD-DSL [29], a technique we have developed for constructing ontology-based domain specific languages in a generative approach to multi-agent domain engineering. With TOD-DSL, a domain specific language for the specification and automatic generation of usage mining based Web recommender systems is being developed.

Acknowledgments This work has been supported by CNPq, an institution of the Brazilian Government for scientific and technologic development.

References

1. Adomavicius G, Tuzhilin A (2005) Toward the next generation of recommender systems: a survey of the state-of-the-art and possible extensions. IEEE Trans Knowl Data Eng 17(6):734–749
2. Arango G (1988) Domain engineering for software reuse. Ph.D. Thesis. Department of Information and Computer Science, University of California
3. Baeza-Yates R, Ribeiro-Neto B (1998) Modern information retrieval. Addison-Wesley, Reading
4. Bellifemine F, Caire G, Poggi A, Rimassa G (2003) JADE a white paper. Exp v. 3 n. 3, Sept 2003. http://www.jade.tilab.com/
5. Belkin NJ, Croft WB (1992) Information retrieval and filtering: two sides of the same coin? Commun ACM 35(12):29–38
6. Berendt B, Hotho A, Stumme G (2002) Towards semantic web mining. In: Proceedings of the 1st international semantic web conference (ISWC-02). Springer, Berlin Heidelberg New York, pp 264–278
7. Braga R, Werner C, Mattoso M (1999) Odyssey: a reuse environment based on domain models. In: IEEE symposium on application-specific systems and software engineering technology (ASSET'99), Texas, pp 50–57
8. Bresciani P, Giorgini P, Giunchiglia F, Mylopoulos J, Perini A (2004) TROPOS: An agent-oriented software development methodology. J Auton Agent Multi Agent Syst 8(3):203–236
9. Brusilovsky P, Maybury M (2002) From adaptive hypermedia to the Adaptive Web. Commun ACM 45(5):31–33
10. Booch G, Rumbaugh J, Jacobson I (1999) Unified modeling language user guide. Addison Wesley, Reading
11. Burke R (2002) Hybrid recommender systems: survey and experiments. User Model User-adapt Interact 12(4):331–370
12. Caire G et al (2001) Agent-oriented analysis using MESSAGE/UML. In: Second international workshop on agent-oriented software engineering, AOSE 2001, pp 101–108

13. Cooley RW (2000) Web usage mining: discovery and application of interesting patterns from Web data. PhD Thesis, Department of Computer Science, University of Minnesota
14. Cossentino M, Sabatucci L, Sorace S, Chella A (2003) Patterns reuse in the PASSI methodology. In: Proceedings of the fourth international workshop engineering societies in the agents world (ESAW'03), Imperial College London, UK, pp 29–31
15. Czarnecki K, Eisenecker UW (2000) Generative programming: methods, tools, and applications. ACM Press/Addison-Wesley Publishing Co., New York
16. Dai H, Mobasher B (2002) Using ontologies to discover domain-level Web usage profiles. In: Proceedings of the second semantic Web mining workshop at PKDD 2002, Helsinki
17. Dileo J, Jacobs T, Deloach S (2002) Integrating ontologies into multi-agent systems engineering. In: Proceedings of 4th international bi-conference workshop on agent oriented information systems (AOIS 2002), Bologna, pp 15–16
18. Drumond L, Girardi R, Lindoso A, Balby Marinho L (2006) A semantic Web based recommender system for the legal domain. In: Proceedings of the European conference on artificial intelligence (ECAI 2006) workshop on recommender systems, August 28th to 29th. Riva del Garda, IOS Press (to appear)
19. Eirinaki M (2004) Web mining: a roadmap, Technical Report, DB-NET 2004, available at http://www.db-net.aueb.gr
20. Eirinaki M, Vazirgiannis M (2003) Web mining for web personalization. ACM Trans Internet Technol 3(1):1–27
21. Facca FM, Lanzi PL (2005) Mining interesting knowledge from weblogs: a survey. Data Knowl Eng 53(3):225–241
22. Falbo RA, Guizzardi G, Duarte KC (2002) An ontological approach to domain engineering. In: Proceedings of the XIV international conference on software engineering and knowledge engineering (SEKE 2002) ACM Press, Italy pp 351–358
23. Girardi R, Faria CA (2003) Generic ontology for the specification of domain models. In: Overhage S and Turowski K (ed) Proceedings of 1st international workshop on component engineering methodology (WCEM'03) at second international conference on generative programming and component engineering, Germany, pp 41–50
24. Girardi R, Faria C (2004) An ontology-based technique for the specification of domain and user models in multi-agent domain engineering. CLEI Electron J 7(1)
25. Girardi R, Faria C, Marinho L (2004) Ontology-based domain modeling of multi-agent systems. In: Gonzalez-Perez C (ed) Proceedings of the third international workshop on agent-oriented methodologies at international conference on object-oriented programming, systems, languages and applications (OOPSLA 2004), Canada pp 51–62
26. Girardi R, Lindoso A (2005) DDEMAS: a domain design technique for multi-agent domain engineering. In: The seventh international bi-conference workshop on agent-oriented information systems (AOIS-2005) at The 24th international conference on conceptual modeling (ER 2005), 2005, Klagenfurt. Proceedings of ER Workshops, LNCS, vol 3770. Springer, Berlin Heidelberg New York, pp 141–150
27. Girardi R, Balby L, Oliveira I (2005) A system of agent-based patterns for user modeling based on usage mining. Interact Comput 17(5):567–591
28. Girardi R, Oliveira I, Bezerra G (2003) Towards a system of patterns for the design of agent-based systems. In: Proceedings of the second Nordic conference on pattern languages of programs 19 a 21 de setembro de 2003 (VikingPLoP 2003), Norway
29. Girardi R, Serra I (2004) Using ontologies for the specification of domain-specific languages in multi-agent domain engineering. In: Grundspenkis J, Kirikova M (eds) Proceedings of the sixth international bi-conference workshop on agent-oriented information systems (AOIS-2004) at the 16th international conference on advanced information systems engineering (CAISE'04). Riga, Latvia, pp 295–308
30. Gruber TR (1995) Toward principles for the design of ontologies used for knowledge sharing. Int J Hum Comput Stud N43:907–928
31. Guarino N (1998) Formal ontology in information systems. In: Proceedings of the 1st international conference on formal ontologies in information systems FOIS 1998, IOS Press, Italy, pp 3–15
32. Jin R, Chai JY, Si L (2004) Content-based filtering & collaborative filtering: an automatic weighting scheme for collaborative filtering. In: Proceedings of the 27th annual international conference on research and development in information retrieval, pp 337–344
33. Kang K, Cohen S, Hess J, Nowak W, Peterson S (1990) Feature-oriented domain analysis (FODA) feasibility study. Technical Report, CMU/SEI-90-TR-21, Software Engineering Institute, Carnegie Mellon University, Pittsburgh
34. Kazienko P, Kiewra M (2003) ROSA—Multi-agent system for Web services personalization. In: Proceedings of the first international Atlantic Web intelligence conference, AWIC 2003, Madrid, pp 297–306
35. Kosala R, Blockeel H (2000) Web mining research: a survey. SIG KDD Explor 2:1–15
36. Lassila O, Swick R (1999) Resource description framework (RDF) model and syntax specification. W3C recommendation, World Wide Web Consortium
37. Lee TB, Hendler J, Lassila O (2001) The semantic Web. Scientific American
38. Li J, Zaiane O (2004) Using distinctive information channels for a mission-based Web recommender system. In: Proceedings of WebKDD 2004 Seattle, pp 35–46
39. Marinho Leandro B (2005) A multi-agent framework for usage mining and user modeling-based Web personalization (in Portuguese). Master dissertation, Federal University of Maranhão—UFMA-CPGEE 2005
40. Mobasher B, Cooley R, Srivastava J (1999) Automatic personalization based on Web Usage Mining. Technical Report TR99010, Department of Computer Science, DePaul University
41. Mobasher B, Dai H, Luo T et al (2000) Integrating Web usage and content mining for more effective personalization. In: Proceedings of the international conference on E-Commerce and Web Technologies (ECWeb), LNCS—Lecture Notes in Computer Science, vol 1875. Springer, Berlin Heidelberg New York, pp 165–176
42. Nasraoui O, Pavuluri M (2004) Complete this puzzle: a connectionist approach to accurate Web recommendations based on a committee of predictors. In: Proceedings of WebKDD 2004, Seattle
43. Nasraoui O, Cardona C, Rojas C, Gonzalez F (2003) Mining evolving user profiles in noisy Web clickstream data with a scalable immune system clustering algorithm, In: Proceedings of WebKDD 2003—KDD workshop on Web mining as a premise to effective and intelligent Web applications, Washington, pp 71–81
44. Odell J, Parunak HVD, Bauer B (2000) Extending UML for agents. In: Proceedings of the agent-oriented information systems workshop at the 17th national conference on artificial intelligence, AOIS Workshop at AAAI, pp 3–17
45. Omicini A (2001) SODA Societies and infrastructures in the analysis and design of agent-based systems. In: Proceedings of the first international workshop, AOSE 2000 on agent-oriented software engineering, Ireland, pp 185–193
46. Osvaldo R (2006) Modeling of a multi-agent system for recommendations of touristic packages through Web usage mining and collaborative filtering, Final degree work, UFMA-CGCC (in Portuguese)
47. Pierrakos D, Paliouras G, Papatheodorou C, Spyropoulos CD (2003) Web usage mining as a tool for personalization: a survey. User Model User Adapt Interact 13:311–372
48. Shahabi C, Banaei-Kashani F (2003) Efficient and anonymous Web usage mining for Web personalization. INFORMS J Comput Special Issue Data Mining, 15(2) Spring
49. Shahabi C, Banaei-Kashani F, Faruque J (2001) A reliable, efficient, and scalable system for Web Usage data acquisition. In: WebKDD'01Workshop in conjunction with the ACM SIGKDD 2001, San Francisco

50. Spiliopoulou Myra, Faulstich Lukas C (1998) WUM: a web utilization miner. In: International workshop on the web and databases, Valencia
51. Steinbach M, Karypis G, Kumar V (2000) A comparison of document clustering techniques. In: KDD workshop on text mining
52. Stumme G, Hotho A, Berendt B (2002) Usage mining for and on the semantic Web. Next generation data mining. In: Proceedings of the NSF Workshop. Baltimore, pp 77–86
53. Wexelblat A, Maes P (1997) Footprints: history-rich web browsing, In: Proceedings conference computer-assisted information retrieval (RIAO), pp 75–84
54. Witten IH, Frank E (2000) Data mining: practical machine learning tools and techniques with Java implementations. Morgan Kaufmann Publishers, San Francisco
55. Wooldridge M, Jennings N, Kinny D (2000) The Gaia methodology for agent-oriented analysis and design. Int J Auton Agent Multi-agent Syst 3(3):285–312
56. Wu K, Yu PS, Ballman A (1998) Speedtracer: A Web usage mining and analysis tool. BM Syst J 37(1):89–105