Personalized Faceted Navigation in Semantically Enriched Information Spaces

Michal Tvarožek, Mária Bieliková
Institute of Informatics and Software Engineering,
Faculty of Informatics and Information Technologies,
Slovak University of Technology
Ilkovičova 3, 842 16 Bratislava, Slovakia
{tvarozek,bielik}@fiit.stuba.sk

Abstract

Existing information retrieval systems provide users with limited support for efficient navigation in large semantically enriched information spaces. Several possible solutions were proposed, such as using faceted metadata search or semantic clusters of search results. We explore the possibilities of using enhanced faceted navigation with support for personalization, collaboration and Semantic Web technologies for (semantic) information retrieval. Furthermore, we propose the extension of faceted browsers with support for dynamic facet generation based on an automatically acquired user model, and evaluate the proposed ideas in multiple domains – scientific publications, digital images and job offers.

1 Introduction

The present Web along with many web-based resources comprise a unique ubiquitous source of information and an environment for collaboration and interaction of many users and businesses. While the amount of available information and the quality and capabilities of information search and processing tools are growing at an incredible rate, so do the size and diversity of the Web's user base and the expectations and requirements of individual users.

Although existing information retrieval (IR) methods are continuously improving, they still fail to address the increasing requirements and expectations of many users with specific needs. For example, most existing search engines such as Google or MSN Live Search employ keyword-based search, while sharing systems such as Flickr or YouTube might extend this with tag-based search. The infamous "advanced search" interfaces allow users to specify even more complex (keyword-based) queries, optionally with some additional domain specific attributes (e.g., size, filetype for images). Video search sites such as IMDb and MovieLens take complexity to another level by offering (multistep) interfaces with many text fields, drop-down menus and multichoice listboxes.

However, several studies have repeatedly indicated that typical search queries are short (up to four words; depending on the domain) [12] and that advanced search is impractical to use for many users [21]. While existing systems are generally good when searching for very specific items, they do not support browsing and exploratory tasks sufficiently [28]. A field study of journalists and newspaper editors selecting photos for newspaper articles conducted by Markulla and Sormunen reported that "professional users" needed to search on multiple categories [16], yet found an elaborate advanced search interface with about 40 input forms unusable.

The Web is a dynamic open information space as many "information artefacts" – documents, articles, images, videos, music files etc. are continuously added, modified, removed, rated or tagged. Thus, user diversity and the evolution of information and user characteristics over time play a crucial role in effective user-centred IR system design. For example, people who grew up with the Web and the Internet, i.e. the "Net Generation", have a natural understanding of this new ubiquitous environment quite unlike their predecessors [18]. Consequently, they have (radically) new requirements, expectations and modes of operation compared to the previous generation of web users.

Accordingly, current changes include a shift from traditional lookup tasks (e.g., fact retrieval) towards more advanced and open ended learning and investigation tasks (e.g., knowledge acquisition, comparison, aggregation, analysis or planning) collectively described as *exploratory search* [15]. Furthermore, the trend towards more interaction and active (social) participation encourages the combination and cross-fertilization of approaches from human-computer interaction, information retrieval, the Adaptive Web and the Semantic Web.

In this paper we build upon several existing approaches and describe an enhanced faceted browser, which is built around the view-based search paradigm using faceted navigation [11] as suitable means for exploratory search. We take advantage of Semantic Web technologies (ontologies in particular) [20] and adaptation based on an automatically acquired user model to improve usability and reduce information overload via personalization [4, 5], ultimately improving overall user experience.

Section 2 describes related work in exploratory search and faceted browsing. Section 3 outlines our design goals and provides a high-level overview of our approach, while section 4 describes the relevance model used to drive our personalization engine and the corresponding user modeling back-end. Next, in sections 5 and 6, we describe the details of our personalization approach for facets and restrictions and for search results respectively. Lastly, we present our evaluation of the proposed approach in multiple application domains in section 7 and draw conclusions in section 8.

2 Related Work

Exploratory search encompasses a broad range of research fields and search and navigation approaches – keyword-based, content-based and view-based search.

2.1 Keyword-based Search

Keyword-based search is currently successfully used, e.g., in all major web search engines (e.g., Google, Live Search, Yahoo) thanks to its simplicity and ease of use, while its disadvantages include ambiguity, low expressiveness and the lack of guidance and interaction. Typical search queries are short (one to three words) though their length varies between domains [13], while advanced search forms are too complex to be practical [21]. Moreover, "guessing" the right keywords is difficult for many users.

The keyword-based IGroup image search engine presents search results in semantic clusters thus alleviating some problems with short, general or ambiguous search queries [25]. IGroup clusters the original result set into several clusters and provides users with an overview of the result set by means of representative cluster thumbnails and names, which users can choose for further navigation. Thus, IGroup improves usability and makes users' search query formulation easier by providing both query suggestion and browsing by textual category labels.

2.2 Content-based Search

Interactive content-based approaches, such as query-by-example (QBE) have been used in multimedia domains where textual descriptions of instances are sparse, unavailable or inconsistent with user expectations. The current state of the art in content-based IR and its broader implications, are surveyed in [14]. Unlike keyword-based search, content-based IR allows users to search interactively – a query is a set of positive (or negative) examples of instances similar to the users' information need.

TagSphere is an approach to visual presentation of search results obtained by QBE information retrieval using collaborative tagging, originally developed for the digital image domain [3]. It stresses usability and user interaction in the search process by providing different tools for tag visualization, selection, query construction and recommendation.

In [8], the authors describe mental matching – a QBE based approach that facilitates exploratory search by bridging the gap between low-level representation of information in databases (i.e., what metadata are available) and high-level semantic descriptions meaningful to end users (i.e., how they understand and use them). The approach employs a Bayesian relevance feedback model and allows users to interactively choose the most similar images out of a set of sample images – a "visual query", which the system than matches against other images in the collection.

2.3 View-based Search

Similarly, view-based search interactively guides users by showing them with successive views of the respective information space and showing them the available options for further query refinement. In practice, view-based search is most commonly realized in faceted browsers often used, for example in online shops for product selection. Faceted browsers allow users to formulate queries via navigation by successively selecting metadata terms in a set of available facets, and to interactively browse the corresponding search results. Authors in [27] compare three major faceted browsers developed in course of research projects aimed at discovering new possibilities of view-based search – Flamenco, mSpace and RelationBrowser.

mSpace is a domain specific browser of RDF data, which provides users with a projection of high dimensional information spaces into a set of columns (filters) shown in the GUI, which can be manually added, rearranged or removed by users [26]. The ordering of individual columns in the GUI is important as the contents of the next column are dynamically determined based on the selection in the previous column. If, in the music domain, columns *TimePeriod*, *Composer* and *MusicPiece* are available, then selecting a time period updates the composer column to only display composers from that period. Similarly, selecting a composer populates the *MusicPiece* column with his works.

Flamenco [28] stresses interface design and guides users through the information seeking process. Users first see a high level overview of the available metadata ("opening"), then refine their query and preview results ("middle game") and lastly explore individual results via horizontal navigation ("endgame"). While in Flamenco the facets are static and predefined, users can manually adapt columns in mSpace to match their needs. Both Flamenco and mSpace support keyword-based search over the entire information space, however only mSpace supports keyword-based filtering in individual facets. Moreover, neither Flamenco nor mSpace provide personalization nor user adaptation.

The overall user response to these approaches was positive – nearly all users preferred them over a baseline approach/interface. Nevertheless, several of the approaches suffer from scalability and information overload issues. E.g., the faceted browser in [28] had an average response time of 3.7s vs. 0.3s for the baseline approach. Furthermore, neither of these solutions provide personalized features based on individual users' characteristics. However, even though some of the aforementioned solutions work with RDF data, they do not take advantage of semantic markup for user interface generation and/or personalization in open information spaces.

The BrowseRDF faceted browser [19] supports automatic facet generation from arbitrary RDF data and extends the expressiveness of faceted browsing by extending typical faceted queries with RDF semantics, e.g. with existential selection, inverse selection, non-existential selection. It identifies facets in source data based on several statistical measures – predicate balance, object cardinality and predicate frequency, yet does not directly address issues of information overload or interface usability and adaptivity.

The faceted browser called /facet [10] is intended for heterogeneous information spaces consisting of distributed semantic repositories represented in RDFS. It takes advantage of the *rdfs:subClassOf* and *rdfs:subPropertyOf* properties in order to process facet restriction hierarchies. Furthermore, /facet supports multi-type queries and runtime facet specification thus greatly increasing flexibility and support for heterogeneous repositories. The multi-type capability effectively translates into an additional facet, which is used to specify the target data type. Based on the selection in the type facet, other facets are made available.

Moreover, /facet supports keyword-based search, which allows users to perform keyword-based search on both data (instances) and metadata (facets and restrictions). Lastly, /facet supports the grouping of search results based on individual properties and timeline visualization of dates. However, it does not support personalization nor advanced link generation and recommendation techniques.

Even though the described approaches present progress in improving search mechanisms, there is still much space left in the sense of combining different approaches together and adapting the resulting approach to individual users' needs ultimately changing the way we search for information in the new [*social adaptive semantic*] Web.

3 Personalized Faceted Navigation Overview

We propose a method for personalized faceted navigation using an enhanced faceted browser, which takes advantage of Semantic Web techniques for ontological knowledge representation, and Adaptive Web techniques for personalized facet and search results recommendation.

Our primary design goals were:

- *Information overload prevention* by recommending relevant content while hiding less relevant content (e.g., facets, restrictions, result attributes).
- *Guidance support* via navigational shortcuts, which streamline navigation in deep/complex faceted hierarchies (e.g., restriction recommendation).
- *Orientation support* by showing additional information/cues simplifying user decisions about further navigation (e.g., tooltips showing future facet contents).
- *Improved response times* due to selective processing of facets and restrictions, since advanced (semantic) approaches proved to be "time consuming".
- *Universality and flexibility* – suitability to different/changing application domains facilitated by (semi)automatic user interface generation.

In order to achieve the aforementioned goals, we take advantage of ontological data representation in OWL:

- The *domain ontology* describes domain concepts, the relations between them and their attributes. It contains metadata that describe the structure of the domain model (i.e., classes and properties) as well as actual domain data (i.e., instances). For example, in the scientific publications domain, it describes authors, publications and venues.
- The *user ontology* describes the characteristics and preferences of users as well as their broader context – the time, location and properties of the device and network they use. Since we address generic browsing in large information spaces, we focus on individual user characteristics and omit the issues of acquiring and using a broader user context, which would be required, e.g., for mobile applications.
- The *event ontology* describes the events that occur in the faceted browser and its states during user interaction so that they can be used for the subsequent automated user characteristics acquisition.

The enhanced faceted semantic browser extends the typical request handling of faceted browsers with additional steps that perform specific tasks (see Figure 1).

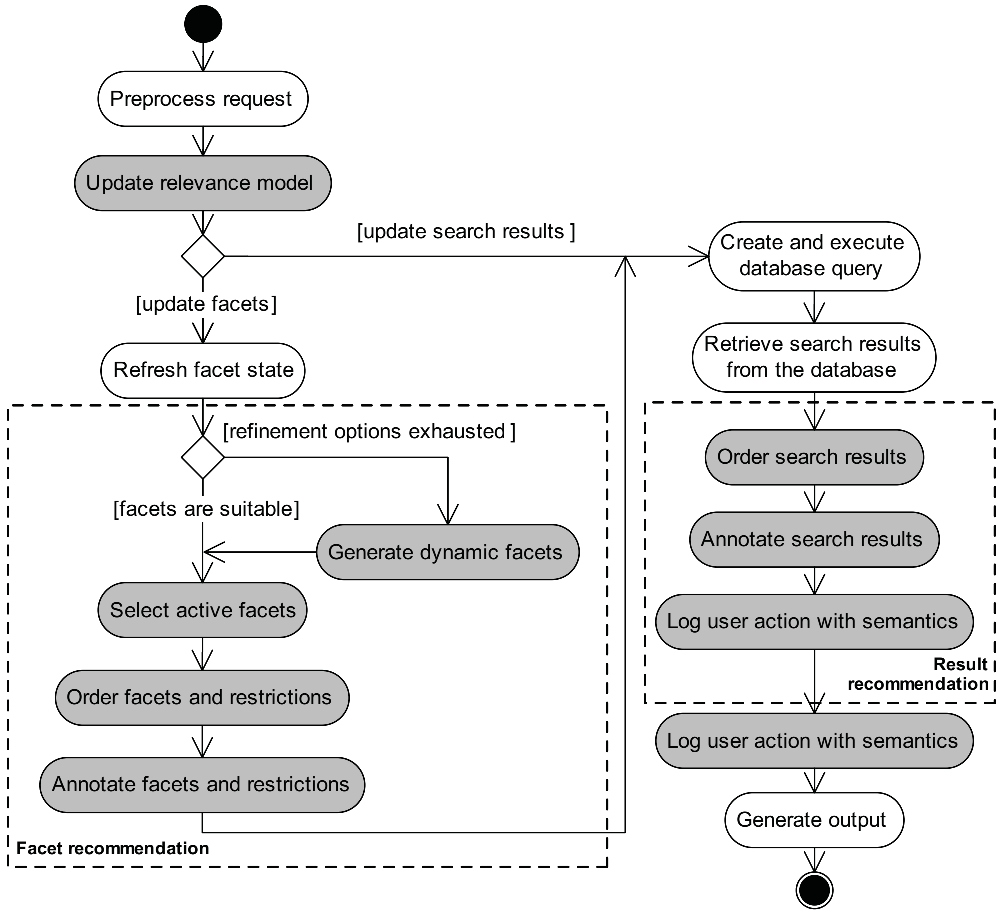

Figure 1. Request handling of the enhanced faceted browser, extensions shown in gray.

Facet processing is extended with *facet recommendation* – active facet selection, facet and restriction ordering and annotation, which improve orientation and guidance support, reduce information overload and alleviate some disadvantages of faceted classification (Figure 1, bottom left). If the set of available facets is insufficient (e.g., the refinement options were exhausted), we use dynamic facet generation to add new facets at run-time on a per user basis thus allowing the user to further refine the search query.

Search result recommendation extends the processing of search results with support for personalized result ordering, annotation and view adaptation (Figure 1, right). We employ external tools that evaluate the relevance of individual search results, e.g., by means of concept comparison with the user model [2] or via the evaluation of (explicit) user feedback [9]. Subsequently, we reorder the search results or annotate them with additional information. We also generate adaptive views, which show only selected search result attributes to prevent information overload.

To facilitate automatic user model acquisition, which is crucial for our personalization approach, we take advantage of the personalized presentation layer described in [22]. We log events that occurred as results of user interaction with the browser and the current state of the browser via a specialized external logging service which preserves the semantics of events [1] (Figure 1, bottom right). The acquired events are processed by the user modeling back-end and and in turn retrieved as an updated relevance model, which drives our personalization engine (Figure 1, top left).

4 Model for Relevance Evaluation

Figure 2 shows our user modeling and personalization loop. Our personalization engine logs user actions and their semantics explicitly as opposed to traditional web server logs, which store them only implicitly in request URLs (Figure 2, top). Each logged event uses our event ontology to specify the semantics of the respective user action and also references the domain and user ontologies as required.

Since the detailed description of the event ontology and logging approach are beyond the scope of this paper, we give only a simplified example. If a user selects New York

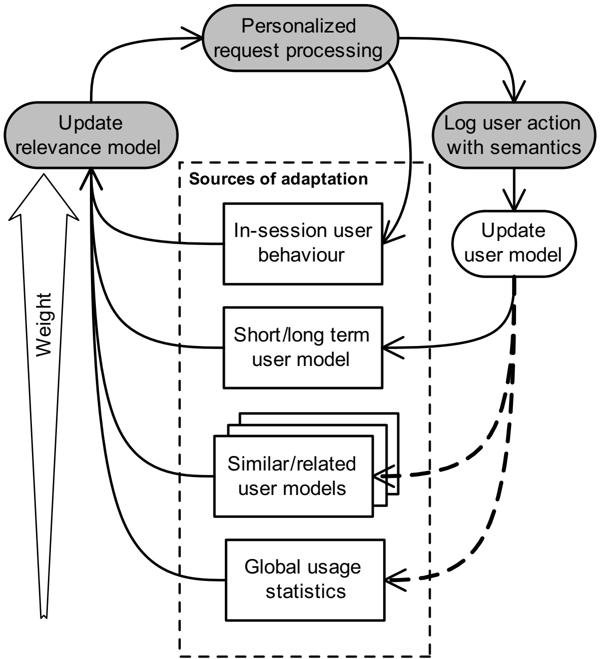

Figure 2. Overview of our user modeling-personalization loop (in gray) and the used sources of adaptation (in-session behaviour, short/long term user preferences and global usage statistics).

in a location facet, we log the event *sl:SelectRestriction*, whose attributes are *sl:Facet* and *sl:Restriction* describing the respective URIs of the used facet and restriction – *r:Facet_location* and *r:NewYork*.

The user modeling back-end provides us with several sources of adaptation, which we employ with different weights depending on how closely related they are to the current user task (Figure 2, bottom):

- *In-session user behavior* – user navigation, facet and restriction selection during the current user session (i.e., user clicks). Frequent use of specific items indicates higher relevance to the current task and/or user interest in the corresponding domain concepts. For example, if *ConferencePaper* is selected as the publication type, showing user interest, additional facets associated with the domain concept *Conference* are likely to be generated in order to allow the user to further refine her query.

- *Short/long term user model* – user characteristics acquired during multiple sessions described by their *relevance* to the user and the *confidence* in their estimation in the range $\langle 0, 1\rangle$. High relevance in the user model denotes good choices for facet generation and restriction recommendation, while high confidence results in high weights when considering the user's needs.

- *Similar/related user models* are assumed to belong to users with similar needs and are thus used for relevance evaluation if user specific data is unavailable or has low confidence. Social user context can be exploited by assigning custom weights to specific relations between users resulting in social recommendation. Moreover, if usage data about other users are "publicly" available, users might directly browse the trails of their peers (e.g., see what images their friends viewed or what papers their colleagues downloaded).

- *Global usage statistics* computed from the overall relevance and usage of individual domain concepts (e.g., facets, restrictions, target objects – be it images, publications or job offers) from all user models. The overall "popularity" of facets and restrictions increases the likelihood of their recommendation for a specific user, especially if his or her specific preferences are unknown or have low confidence.

Let $L_U(X) = relevance_U(X)$ be the local relevance of concept X from the domain ontology for user U. For example, X might be a facet, a restriction, a search result or a property. We define $C_U(X)$ as the cross relevance of X determined as the average local relevance for all users V weighted by their similarity $sim(U,V)$ to user U (1), and $G(X)$ as the global relevance of X defined as its mean local relevance for all users (2).

$$C_U(X) = \frac{\sum\limits_{V \in users} \left(sim(U,V) * L_V(X)\right)}{1 + \sum\limits_{V \in users} sim(U,V)}, U \neq V \quad (1)$$

$$G(X) = \frac{\sum\limits_{V \in users} L_V(X)}{|users|} \quad (2)$$

To evaluate the user similarity $sim(U,V) \in \langle 0, 1\rangle$ we employ external concept comparison tools [2]. Alternatively, similarity can be evaluated via the sum of square differences in concept relevance between users (3).

$$sim(U,V) = 1 - \frac{\sum\limits_{X \in concepts} \left(L_U(X) - L_V(X)\right)^2}{|concepts|} \quad (3)$$

We define $T_U(X)$ as the temporary in-session relevance of concept X determined as the percentage of user clicks on concept X from the total number of clicks on that concept type – e.g., a facet or a restriction (4).

Static relevance $S_U(X)$ defines the relevance of concept X based on the user model and the respective *confidence* in the relevance estimation (5). Dynamic relevance $D_U(X)$ defines the total relevance of concept X based on the user model and the current in-session user behavior (6).

$$T_U(X) = \frac{Clicks(X)}{1 + TotalClicks} \tag{4}$$

$$\begin{gathered} S_U(X) = L_U(X) * confidence_U(X) + \\ (C_U(X) + G_U(X)) * (1 - confidence_U(X)) \end{gathered} \tag{5}$$

$$D_U(X) = S_U(X) + T_U(X) \tag{6}$$

As an alternative and/or addition to cross relevance, we use weighted social relevance $\hat{C}_U(X)$ if social network data for a specific relation $rel(U, V)$ are available (7).

$$\hat{C}_U(X) = \frac{\sum\limits_{rel(U,V) \in relations} (w(rel) * L_V(X))}{|rel(U, V)|} \tag{7}$$

5 Facet Recommendation

Facet recommendation distinguishes three types of facets adapted at run-time to the specific needs of individual users:

- *Active facets* are fully accessible facets (also known as primary facets), which can be used for faceted query construction, and whose content (i.e., restrictions) is visible and entirely processed (e.g., annotated).
- *Inactive facets* are partially accessible facets (also known as secondary facets), which are used in faceted queries if they have active selections. While their content is not directly visible and thus left unprocessed, they can be activated automatically or per user request.
- *Disabled facets* are partially accessible facets, which are only available after all active/inactive facets were exhausted or on specific user demand. They are not used in queries and their content is not visible.

5.1 Facet and Restriction Personalization

The adaptation process first determines the relevance of individual facets and restrictions in our relevance model (see section 4) and then uses it in these steps:

1. *Active facet selection* – the total number of active facets is reduced to a relatively low number, e.g. 2 or 3 facets, since many facets are potentially available in complex information spaces. Active facets are selected based on relevance and on recency and number of accesses – the most relevant facets or recently/often accessed facets are likely to be active. The rest of the facets is made inactive or left in disabled state.
2. *Facet and restriction ordering* – all facets are ordered in three groups (i.e., active, inactive, disabled) in descending order based on their relevance with the last used facet always being at the top. Restrictions are ordered alphabetically, since alternative orderings based on relevance or the number of matching search results were not well accepted by users as they made it difficult to search for specific items.
3. *Facet and restriction annotation* – active facet restrictions are annotated with the number of matching instances, the relative number of matching instances by means of font size/type, or directly recommended (e.g., with background color or the “traffic lights” metaphor) effectively providing shortcuts to deeply nested restrictions. Additional tooltips can describe individual facet/restriction meanings (e.g., the rdfs:comment annotation in ontologies), annotated child restrictions with relevance, or (personalized) annotations generated by external tools [17].

5.2 Dynamic Facet Generation

Normally, facet generation is only triggered when the set of available facets is exhausted, i.e. when no or very few active/inactive facets are available.

During facet generation we examine the attributes of target instances as defined in the domain ontology. For example for images, we examine attributes of the domain concept *Image* and its associate concepts (via properties), e.g., *Location* denoting the place where the image was taken.

We search for eligible candidate properties of individual instance types, which can be used for facet construction based on low-level metadata facet templates used for automated facet construction from the domain ontology (we manually used these templates to create the initial user interface). For example in the publication domain, a class hierarchy facet for the property *rdf:type* is constructed from the *rdfs:subClassOf* class hierarchy rooted at *pub:Publication*.

Since it is not desirable to generate all possible facets due to the their large number, we evaluate the aggregate suitability of individual attributes based on the aforementioned relevance model (see section 4). Lastly, we determine a suitable presentation method for each new facet and forward the resulting set of new facets to the following facet personalization stage. Figure 3 illustrates the proposed facet presentation methods:

- *Simple facets* – top-level facets based on direct or indirect attributes of target instances, e.g. directly for images – the object, keywords or location, or indirectly – the resolution of the camera used to take the photo.
- *Nested facets* – facets that in addition to (or instead of) a set of individual restrictions contain a set of *child* facets, e.g., a facet that contains facets for the type of place, popularity and climate of the location where a photo was taken.

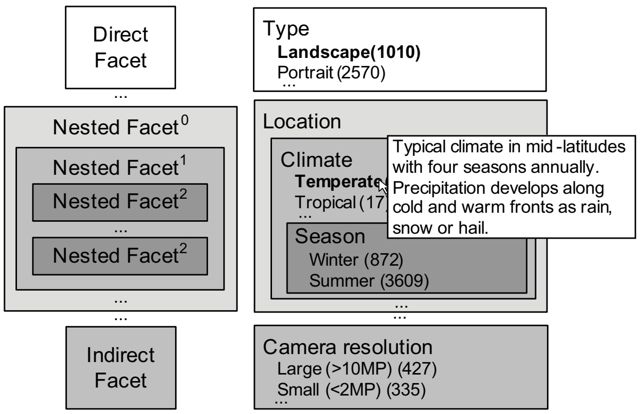

Figure 3. Facet presentation methods (left) and adaptation examples (right). Bold text is used for recommendation, tooltips and instance counts for annotation.

Direct attributes of target instances are presented via simple (direct) facets. If only one indirect attribute of an associated instance type is presented a simple (indirect) facet is used. If multiple indirect attributes of the same type are presented a nested facet can be used so that each nesting level corresponds to one level of attribute indirection.

6 Search Result Recommendation

Based on the computed relevance and the results of external tools, we perform these recommendation steps:

1. *Search result ordering* – we support simple results ordering – unordered results or ordered based on a single attribute (e.g., date). Additionally, we employ external ordering (relevance evaluation) tools, which either evaluate relevance based on common global preferences, or on personalized ratings constructed from explicit user feedback (i.e., rating of instances) [9]. Furthermore, we employ external similarity evaluation tools, which enable users to search for instances similar to a given search result [17].
2. *Search result annotation* – individual search result attributes are annotated similarly to facets and restrictions. Tooltips show their meanings (*rdfs:comment*) or their properties from the domain ontology. Alternatively, external annotation tools are used to provide custom (personalized) annotations generated from the domain and user ontologies [17]. For example, in the movie domain, we can display the suitability of a movie, based on its estimated relevance to the user's preferences, as background color or via emoticons.
3. *View adaptation* – we support several adaptive views – simple overview, extended overview, thumbnail matrix or detailed view, which display increasingly more detailed information about individual search results (ontology instances). The attributes of the displayed instances are adaptively chosen and ordered based on their estimated relevance derived from the user model. Moreover, the faceted browser can show instances of different types so that the user can seamlessly switch from browsing/searching for e.g., images to videos, then to actors and back to images.

7 Evaluation

7.1 Architecture and Implementation

For evaluation, we developed *Factic* – a prototype of our enhanced faceted browser [24], which implements selected parts of the proposed navigation method based on the faceted browser processing pipeline described in section 3. The overall architecture of our solution is based on the integration and cooperation of several loosely coupled components – software tools, as defined by the personalized presentation layer architecture [22]. We used Apache Cocoon (`cocoon.apache.org`) as the underlying portal framework, which is based on the pipelines architectural pattern, and thus allowed us to construct different XML based pipelines to handle our request processing and XML/XSL transformations.

Factic is divided into two relatively independent parts each facilitating the presentation of information and adaptation of the GUI respectively (Figure 4, top left). The adaptation part of Factic performs faceted queries and relevance model updates with the successive adaptation of facets and views, while the presentation part transforms its XML output via XSLT into the final XHTML rendered on the client web browser.

Since Factic relies heavily on user characteristics stored in the user model, it forwards events with semantics occurring during user interaction to the user modelling backend consisting of components for server-side and client-side user behaviour evidence acquisition and user characteristics evaluation (Figure 4, centre). In our solution, these correspond to tools the SemanticLog, Click and LogAnalyzer respectively [1]. In order to further enhance the functionality offered to end users, Factic also takes advantage of several external information retrieval (CriteriaSearch), relevance evaluation (UpreA/TopK), annotation (Pannda) and concept comparison (ConCom) agents from the application layer of our solution (Figure 4, lower centre) [9, 2, 17].

Lastly, the aforementioned components all work over common knowledge repositories comprised of the domain ontology, user ontology and event ontology corresponding to the domain model, user model and event logs respectively (Figure 4, bottom). We store the populated domain and user ontologies in the Sesame ontological repository (`openrdf.org`) for easy access via ontological query languages, and the event logs in a relational database for quick incremental stream processing of incoming events. During evaluation, we identified several scalability issues with the ontological repository, which forced us to perform additional optimizations (e.g., caching, query tuning) though satisfactory response times were still difficult to achieve.

7.2 Examples and Domains

We applied our approach to three different application domains – online job offers (project NAZOU [17], `nazou.fiit.stuba.sk`), scientific publications (project MAPEKUS, `mapekus.fiit.stuba.sk`) and digital images.

For each domain, we have constructed both a domain and a user ontology describing the main domain concepts and their properties. The job offer ontology had the most complex schema consisting of some 740 classes with hierarchical classifications up to 6 levels deep. The publication ontology was of medium complexity with only one hierarchical classification (the ACM classification), while the digital image ontology had a relatively simple flat schema.

We populated the ontologies with instance data of different sizes acquired from publicly available web resources (e.g., `careerbuilder.com`, `eurojobs.com`, `profesia.sk`, DBLP, Springer and ACM DL). We worked with manually/semi-automatically created "toy-size" datasets having 100s-1000s of instances to automatically acquired, large integrated datasets in excess of 100,000s of instances and several times that many triples.

To demonstrate the flexibility and relative domain independence of our approach, we configured Factic for use in individual application domains (i.e., for their domain and user ontologies). We build upon existing successful faceted browser interface concepts and adaptive hypermedia interfaces. Figure 5 shows the sample GUI of our adaptive faceted browser in the digital image domain employing the general faceted browser layout (facets on the left, query at the top, search results in the centre, optional manual search result customization, e.g. sorting, above search results). Our enhanced faceted browser offers a combined searching and browsing interface, and is suited for effective viewing of and navigating in large open information spaces represented by an OWL ontology. It can also be used as an information retrieval tool where the search query is visually created via navigation – the selection of restrictions in the set of available facets, which are dynamically adapted to users' needs. We also provide users with advanced browsing, searching and visualization features as described below.

Information overload prevention. We adaptively reduce the number of accessible items so that users can efficiently focus on the most relevant facets and restrictions without having to scroll several screens down. If users seek images, only facets for the creation date, object and tags would be displayed while others concerning image size and acquisition data would be available on demand (Figure 5, left).

Orientation and guidance support. We provide visual cues recommending further navigation – the number of instances matching restrictions, and textual descriptions of their meanings. Background color indicates restriction recommendation for navigational shortcuts, while "traffic lights" denote their relation to the users' fields of interest (Figure 5, left). Individual search results show additional attributes along with average user ratings (Figure 5, centre).

Query refinement. If the available set of facets becomes exhausted, additional facets created via dynamic facet generation allow users to refine their queries beyond what would have been possible with statically defined facets.

Social navigation and collaboration. Collaboration and social networks are considered via *Global relevance,* which describes the overall "popularity" of concepts (i.e., what others think is good) while *cross relevance* also considers similarity and/or relations between users. We can also define additional facets based on social network data (e.g., relation types) allowing users to browse their peers' "trails" directly. Hence, users might access facets, which select only content, e.g., created, viewed, tagged or rated by their peers.

7.3 Experiments and Discussion

In total, we performed several different sets of experiments to validate our approach. We present some of the experimental results in the job offers domain, where our approach proved to be particularly suitable, since it is a very complex information spaces with several deep hierarchical classifications (e.g., regions or positions) and intricate concept relations. We experimented with different adaptation,

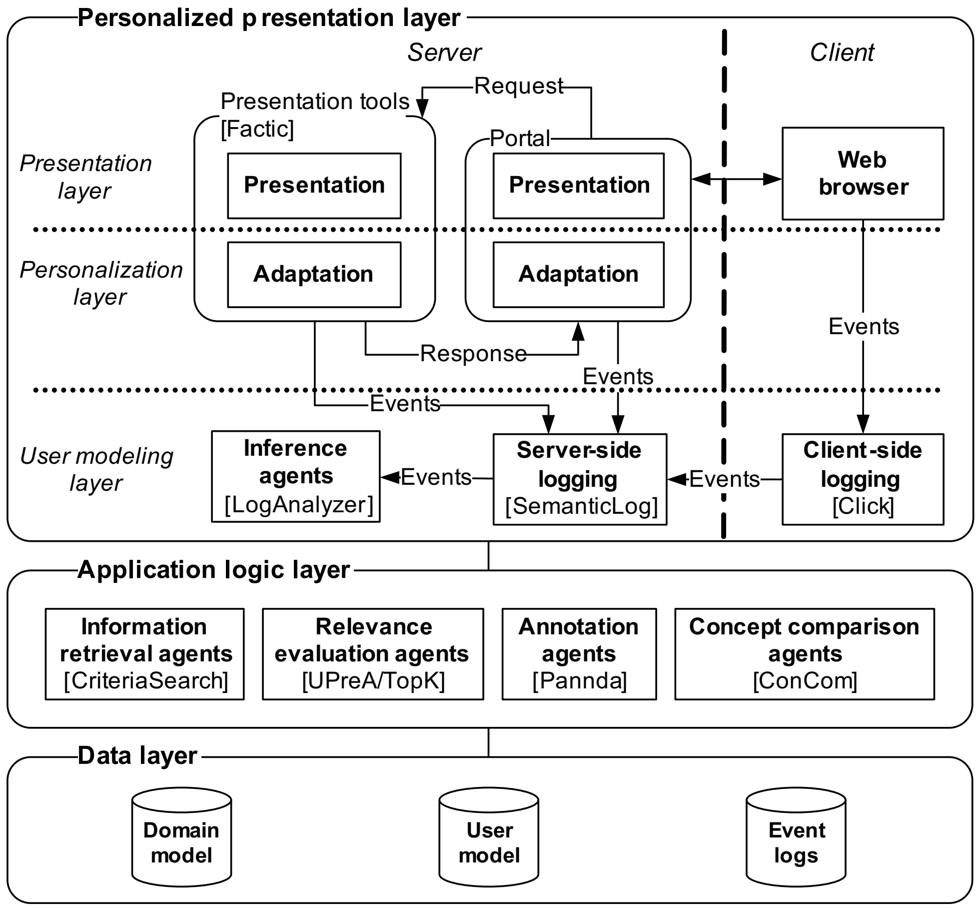

Figure 4. The architecture of our personalized presentation layer.

annotation and recommendation modes. Figure 6 illustrates the time and number of user clicks, which represent the total user effort that was necessary to complete a given scenario, i.e. to find a set of job offer instances.

Our evaluation showed that adaptive selection of active facets can significantly reduce total processing time which depends roughly linearly on the number of displayed facets (assuming an average branching). However, the number of clicks increased since the right facets were not always active and thus had to be manually enabled. This results in shorter refresh times and consequently shorter total task times.

Recommendation of suitable ontological concepts based on the user model further improved total task time and also decreased the number of necessary clicks due to the effective creation of navigational shortcuts that allowed users to skip several clicks by directly recommending suitable restrictions within a restriction hierarchy. As before, the number of clicks increased as the number of active facets decreased as more facets had to be manually activated.

We encountered one significant bottleneck that seri-

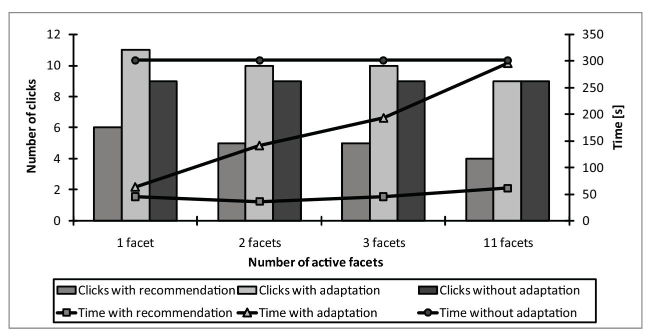

Figure 6. Experimental results for different adaptation modes – non-adaptive, with adaptation, with recommendation, for different numbers of simultaneously active facets.

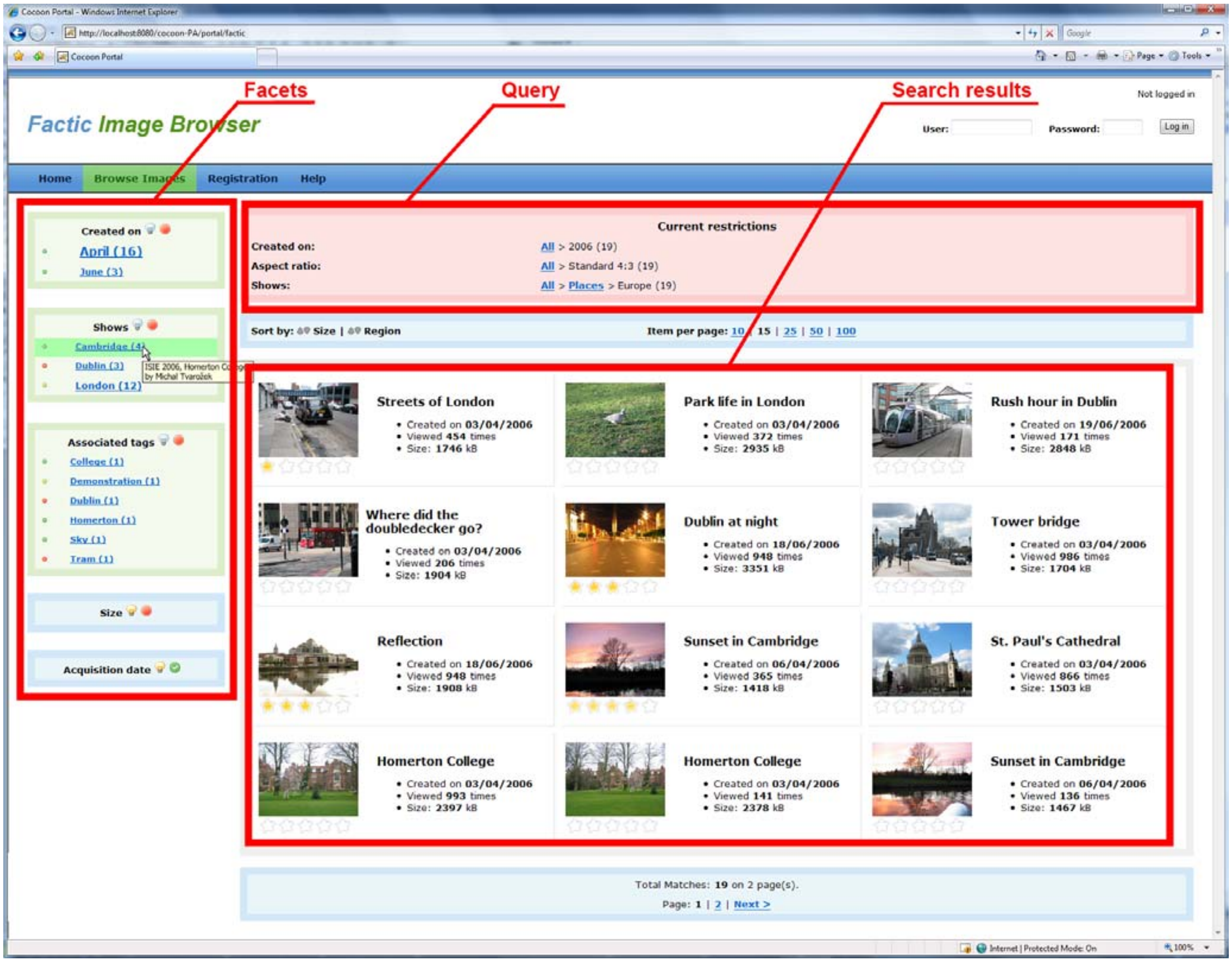

Figure 5. Example GUI of our enhanced faceted browser Factic in the digital image domain.

ously limits widespread deployment of Semantic Web applications – the immaturity of ontological repositories (we used Sesame) in terms of their query processing speed, and query language deficiencies that had to be emulated (missing aggregation and ordering operators in SeRQL). While SPARQL addresses some problems, the one most crucial aggregate operator – `COUNT()` is still unavailable (or `MINCOUNT()` due to the open world assumption).

Furthermore, effective evaluation of Semantic Web approaches is still somewhat difficult since few "good" – rich, complex and large enough ontological datasets are available, while the bad scaling of ontological repositories puts strong bias on every real-world usability study. Our larger datasets yielded only limited results due to their "quality" – their effective use would require extensive preprocessing, which can be only partially achieved by automated means [7].

8 Conclusions

We presented a novel method of personalized faceted navigation in semantically enriched information spaces using dynamic facet generation with successive facet recommendation as an enhancement for generic faceted browsers. Our approach is suitable for open information spaces as it not very susceptible to changes which are a distinguishing characteristic of open information spaces.

The main advantages of our approach are:

- the *visual construction of semantic queries* via navigation aided by *personalized recommendation of browsing* in a faceted browser,
- the *improved user experience* due to *decreased information overload* and *navigation guidance and orientation support* in large information spaces,

- the flexible *(semi)automatic interface generation* and *dynamic facet generation* based on semantic metadata from the domain and user ontologies.

We already see several promising direction of future research, which are likely to further improve overall user experience. Visual presentation methods for facets, search result overviews and details are likely to improve the understandability of the domain and the available data. Visual navigation in clusters might provide users with the necessary "global" overview of the respective information subspace selected in a faceted browser, while incremental horizontal navigation might be used for details browsing [23]. Likewise, the integration of novel social and collaborative approaches as well as the inclusion of mobile application considerations has potential to improve navigation efficiency and ubiquitous deployment. Lastly, the design of optimized graphical user interfaces from the HCI perspective with the corresponding usability studies would be of great interest for practical applications.

Acknowledgment. This work was partially supported by the Slovak Research and Development Agency under the contract No. APVT-20-007104 and the Scientific Grant Agency of Slovak Republic, grant No. VG1/3102/06.

References

[1] A. Andrejko, M. Barla, M. Bieliková, and M. Tvarožek. Software tools for acquiring, organising and presenting information and knowledge on the Web. In P. Vojtáš and T. Skopal, editors, *Datakon'06*, pages 139–148, 2006.

[2] A. Andrejko and M. Bieliková. Estimating similarity of the ontological concepts instances for the adaptive applications based on semantic web. In V. Snášel, editor, *Znalosti 2008: Proceedings of the 7th annual conference*, pages 30–41, 2008. [in Slovak].

[3] M. Aurnhammer, P. Hanappe, and L. Steels. Augmenting navigation for collaborative tagging with emergent semantics. In Cruz et al. [6], pages 58–71.

[4] P. Brusilovsky. Adaptive Hypermedia. In A. Kobsa, editor, *User Modeling and User-Adapted Interaction, Ten Year Anniversary Issue*, pages 87–110, 2001.

[5] P. Brusilovsky, A. Kobsa, and W. Nejdl, editors. *The Adaptive Web, Methods and Strategies of Web Personalization*, LNCS 4321. Springer, 2007.

[6] I. F. Cruz, S. Decker, D. Allemang, C. Preist, D. Schwabe, P. Mika, M. Uschold, and L. Aroyo, editors. *The Semantic Web - ISWC 2006, 5th International Semantic Web Conference, ISWC 2006, Athens, GA, USA, November 5-9, 2006, Proceedings*, volume 4273 of *Lecture Notes in Computer Science*. Springer, 2006.

[7] W. Dakka, P. G. Ipeirotis, and K. R. Wood. Automatic construction of multifaceted browsing interfaces. In *CIKM '05: Proc. of the 14th Int. Conf. on Information and knowledge management*, pages 768–775, NY, USA, 2005. ACM Press.

[8] M. Ferecatu and D. Geman. Interactive search for image categories by mental matching. In *Proc. of the IEEE 11th Int. Conf. on Computer Vision, ICCV 2007*, pages 1–8, Rio de Janeiro, Brazil, 2007. IEEE.

[9] P. Gurský, T. Horváth, R. Novotný, V. Vaneková, and P. Vojtáš. Upre: User preference based search system. In *Proc. of the Int. Conf. on Web Intelligence 2006*, pages 841–844, Washington, DC, USA, 2006. IEEE.

[10] M. Hildebrand, J. van Ossenbruggen, and L. Hardman. /facet: A browser for heterogeneous semantic web repositories. In Cruz et al. [6], pages 272–285.

[11] K. Instone. How user interfaces represent and benefit from a faceted classification system. In *SOASIST*, 2004.

[12] B. J. Jansen, A. Spink, and J. Pedersen. An analysis of multimedia searching on altavista. In *MIR '03: Proc. of the 5th ACM SIGMM int. workshop on Multimedia information retrieval*, pages 186–192. ACM Press, 2003.

[13] B. J. Jansen, A. Spink, and J. Pedersen. An analysis of multimedia searching on altavista. In *MIR '03: Proc. of the 5th Int. Workshop on Multimedia Information Retrieval*, pages 186–192, NY, USA, 2003. ACM.

[14] M. S. Lew, N. Sebe, C. Djeraba, and R. Jain. Content-based multimedia information retrieval: State of the art and challenges. *ACM Trans. Multimedia Comput. Commun. Appl.*, 2(1):1–19, 2006.

[15] G. Marchionini. Exploratory search: from finding to understanding. *Comm. of the ACM*, 49(4):41–46, 2006.

[16] M. Markkula and E. Sormunen. End-user searching challenges indexing practices in the digital newspaper photo archive. *Information Retieval*, 1(4):259–285, 2000.

[17] Návrat et al., editors. *Proceedings of Tools for Acquisition, Organization and Presenting of Information and Knowledge, Research Project Workshop*, Poľana, Slovakia, 2007.

[18] D. Oblinger and J. Oblinger, editors. *Educating the net generation*. Educase, 2005. www.educause.edu/educatingthenetgen.

[19] E. Oren, R. Delbru, and S. Decker. Extending faceted navigation for rdf data. In Cruz I. et al., editor, *LNCS: The Semantic Web – ISWC 2006*, volume 4273, pages 559–572. Springer-Verlag Berlin heidelberg, 2006.

[20] N. Shadbolt, T. Berners-Lee, and W. Hall. The semantic web revisited. *IEEE Intelligent Systems*, 21(3):96–101, May/June 2006.

[21] TASI: A Review of Image Search Engines. May 2006, http://www.tasi.ac.uk/resources/searchengines.html.

[22] M. Tvarožek, M. Barla, and M. Bieliková. Personalized Presentation in Web-Based Information Systems. In J. van Leeuwen et al., editor, *Proc. of SOFSEM 2007*, LNCS 4362, pages 796–807. Springer, 2007.

[23] M. Tvarožek, M. Barla, G. Frivolt, M. Tomša, and M. Bieliková. Improving semantic search via integrated personalized faceted and visual graph navigation. In V. Geffert, J. Karhumäki, A. Bertoni, B. Preneel, P. Návrat, and M. Bieliková, editors, *SOFSEM*, volume 4910 of *Lecture Notes in Computer Science*, pages 778–789. Springer, 2008.

[24] M. Tvarožek and M. Bieliková. Personalized Faceted Navigation in the Semantic Web. In L. Baresi, P. Fraternali, and G. Houben, editors, *ICWE 2007*, LNCS 4607, pages 511–515. Springer, 2007.

[25] S. Wang, F. Jing, J. He, Q. Du, and L. Zhang. IGroup: presenting web image search results in semantic clusters. In *CHI '07: Proc. of the SIGCHI conf. on Human factors in computing systems*, pages 587–596. ACM Press, 2007.

[26] M. L. Wilson and m.c. schraefel. mspace: What do numbers and totals mean in a flexible semantic browser. In *The 3rd Int. Semantic Web User Interaction Workshop at ISWC2006*, 2006.

[27] M. L. Wilson, m.c. schraefel, and R. W. White. Evaluating advanced search interfaces using established information-seeking models. *J. of the American Society for Inf. Sci. and Tech., to appear*.

[28] K.-P. Yee, K. Swearingen, K. Li, and M. Hearst. Faceted metadata for image search and browsing. In *CHI '03: Proc. of the SIGCHI conf. on Human factors in computing systems*, pages 401–408, New York, USA, 2003. ACM Press.