Grant Agreement Number: 257528

KHRESMOI

www.khresmoi.eu

Report on User Input for Updating Resources

Deliverable number	*D3.4*
Dissemination level	*Public*
Delivery date	*31 August 2013*
Status	*Final*
Author(s)	*Lorraine Goeuriot, Liadh Kelly, Sascha Kriewel, Gareth Jones, Ivan Martinez Rodriguez, Miguel Angel Tinte Garcia, Pavel Pecina, Ales Tamchyna, René Donner*

This project is supported by the European Commission under the Information and Communication Technologies (ICT) Theme of the 7th Framework Programme for Research and Technological Development.

Abstract

This deliverable describes the collaborative functionalities developed within Work Package 3 of the Khresmoi project. They include "social" components on the interface, allowing users obtain a personal profile and interact with each other. The deliverable also includes details on functionalities to allow users contribute to Khresmoi knowledge. As automatic annotation of documents can sometimes contain errors, or can lack precision, experienced users could contribute to the correction of these annotation errors. In the same way, the translations made available through the Khresmoi Machine Translation system [3] could contain erroneous translations that could be corrected by the users. Some of these components have already been evaluated during the first round of user-centred evaluation, and the others will be included in the second round, in year 4 of the Khresmoi project.

Table of abbreviations

DoW	Deliverable of Work
CM	Correction Manager
REST	Representational State Transfer
API	Application Program Interface
UUID	Universally Unique Identifier
GUI	Graphical User Interface
MT	Machine Translation

Table of contents

1. Introduction

This deliverable describes the collaborative functionalities developed within the Khresmoi project. They include "social" components such as the creation of user profiles, the ability to search among users, create groups, as well as capability to share links and documents between users or groups of users. Users are also able to store information related to their searches, such as a search history and a hotlist of Khresmoi documents.

The deliverable also presents an exploration of functionalities to allow users contribute knowledge directly to the Khresmoi knowledge base. A knowledge base [1] has been created for Khresmoi and used to annotate the document collection [2]. The knowledge added to Khresmoi documents can be used by the various technologies which process documents, but it can also be made available to users, in the form of annotations in the documents. As automatic annotation of documents can sometimes contain errors, or can lack precision, experienced users could help correct these annotation errors. In the same way, the translations made available through the Khresmoi Machine Translation system [3] could contain erroneous translations that could be corrected by the users. We published a first investigation on these user contributions in [4].

The deliverable is structured as follows: in Section 2, we detail the requirements for collaborative functionalities within the Khresmoi project; in Section 3, we present the social components that have been developed in the Khresmoi prototypes; and Section 4 describes research conducted to allow users to contribute to Khresmoi knowledge and components developed for this purpose.

2. Collaborative Functionalities Requirements

The collaborative elements identified in the Khresmoi project's DoW are the functionalities allowing users to update resources:

"*T3.4 User input for updating resources (DCU, ATOS)*

Users will be able to update and correct errors in resources while using the system. This can take several forms: direct correction of errors or omissions in annotation or translation, manual contribution of new knowledge, e.g. translations, or verification or clarification of automatically extracted suggested updates for resources. In addition to supporting users of KHRESMOI in updating resources in operation, we will also explore methods such as collaborative editing [...]" (p. 28)

The functionality mentioned here are mainly to let the user correct and add annotations and translations in the system.

Three deliverables have been published on user requirements: general public [5], health professionals [6] and radiologists [7]. Among the list of tools that could enhance a medical search engine, some collaborative functionalities were mentioned.

From the general public:

- Result rating and rating display ("*54% think it would be useful to directly rate search results and view other users' ratings.*", p. 27)

From health professionals:

- Information quality rating and ratings display (*"When asked about which tools they preferred an interesting result was that the most important tool was 'being able to quality rate information/websites and perceiving ratings of other physicians'*, p. 34)
- Knowledge sharing (*"specialised, secured physician communities where we can exchange knowledge about patient cases with other physicians"*, p. 36)

From radiologists:

- *"Confidence score in the diagnosis"* (p. 18)
- Knowledge sharing (*"[...] the communication among colleagues is used to share this knowledge not only during training, but also in clinical practice. Past cases store experience of other colleagues and could make this experience available in a more systematic way."*, p. 20)

From all our user surveys, we can see that search history is also an important feature. This functionality is reported in T3.2 [8]. A personalized history, or hotlist, could also be made available for registered users, where they could save and share interesting results they found using the public search tool.

To allow users register with the system and interact with each other, new features need to be added to the Khresmoi system. Specifically, user profiles, user search features and user group features.

Table 1 provides an overview of the components that have been developed within Khresmoi for the physician prototype.

Component	**Status**	**Description**
Search history	Done	Storage for users of their search history
User profile	Done	Storage for users of their profile
Groups of users	Done	Creation of groups of users
File sharing	Done	Document sharing between users/groups
Personal library	Done	Ability for users to save documents in a hotlist
User search	Done	Ability to search for users by their names
User annotations	Done	Comments from users on documents
User updating translations	Under development	Ability for users to update wrong translations

Table 1: Overview of Khresmoi developed collaborative components.

These components were made available on the physician's prototype as a starting point for several reasons: (1) Medical professionals would more likely benefit from collaborative functionalities; (2) As the components would firstly need to be evaluated before being definitively included in the prototype, it is easier to focus on a single prototype.

These components will be evaluated during the second phase of user-centred evaluations, in year 4 of the Khresmoi project. Depending on the results of these evaluations, these components may be

modified, and may be considered for implementation in the radiologist and/or in the general public prototype.

These components are only available on the physician's prototype to registered users. We describe the components in detail in the next section.

3 Users Collaborating on the Khresmoi System

As described in the previous section, several new interface components were developed to support collaboration of users and user driven updates of content. They were integrated into the two versions of the Khresmoi Professional prototype (for physicians) and are described in this section.

3.1 Search History

Users' queries are stored along with the associated unique document identifiers from the result list (see Fig. 1). Storing past queries is already part of the WP3 framework: only during one session for guest users; beyond the session for registered users. Storing search actions is part of the user model. Past queries are also taken into account when providing tactical search suggestions. Document contents are not stored.

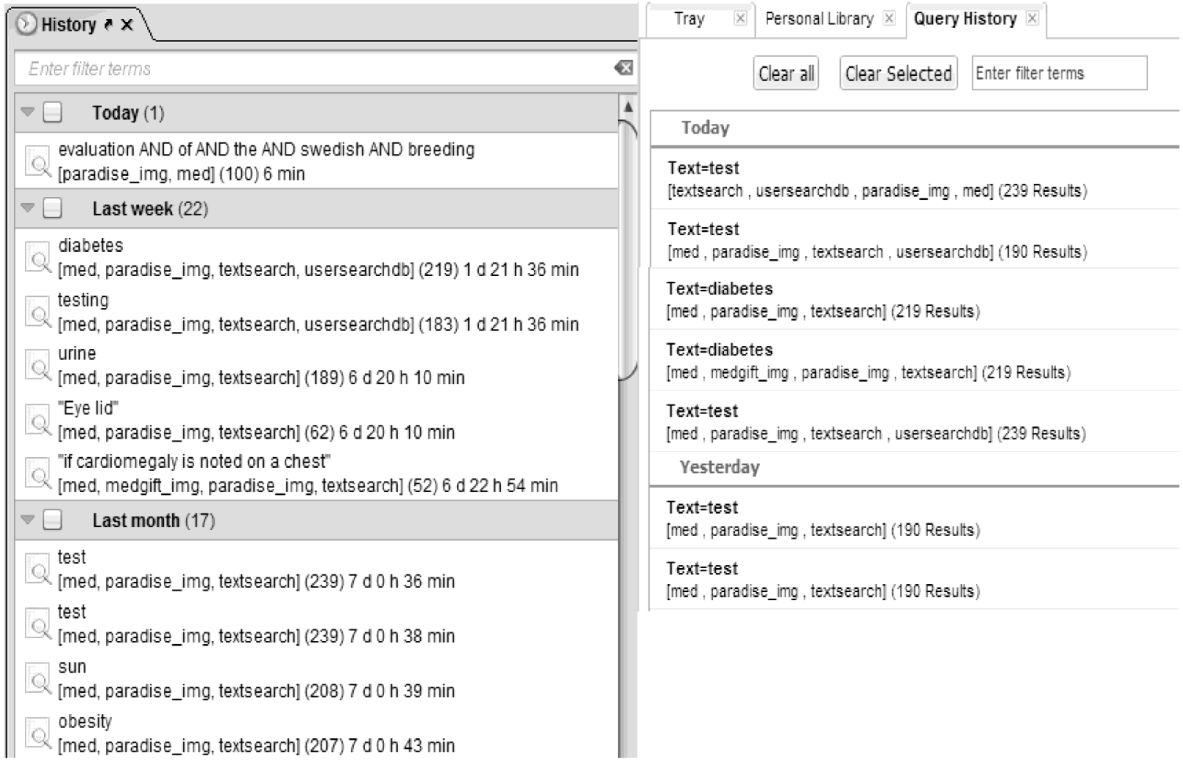

Figure 1: Query history component.

3.2 Personal User Profile

Users can maintain a public profile which contains information about the user stored in the Khresmoi system, e.g. name, country of residence, preferred language of communication, organization, picture or a personal description (describing their medical field, specialization or research interests). This profile can be filled by the user using the configuration dialogue of the user interface and can be viewed by other users (see Fig. 2).

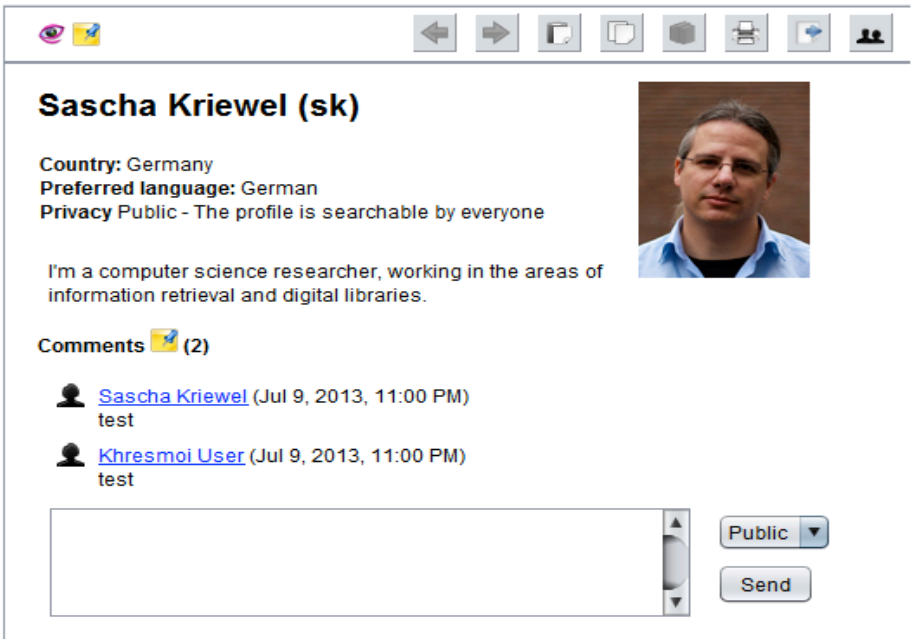

Figure 2: User personal profile.

3.3 Groups of Users

Users can create groups of colleagues sharing the same interest, or working in the same hospital for discussion(s) or sharing of documents. File sharing and discussion/annotation threads can be carried out with selected individuals or within a selected group (see Fig. 3).

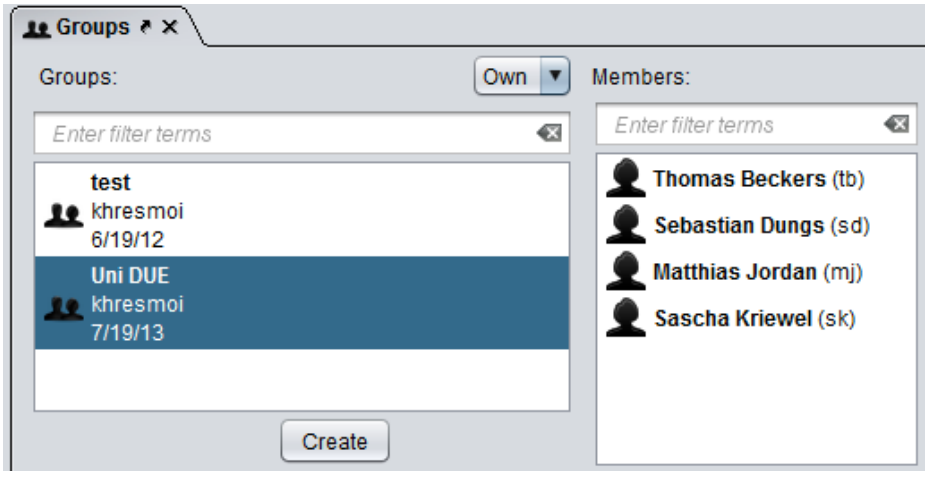

Figure 3: Group management component.

3.4 Searching for Users

Users can search for colleagues, friends or potential cooperation and collaboration partners based on their name, medical field or affiliation (if those have been set in the user's profile description). This is done using the search tools, results are linked to the public view of the users' profile (see Fig. 4).

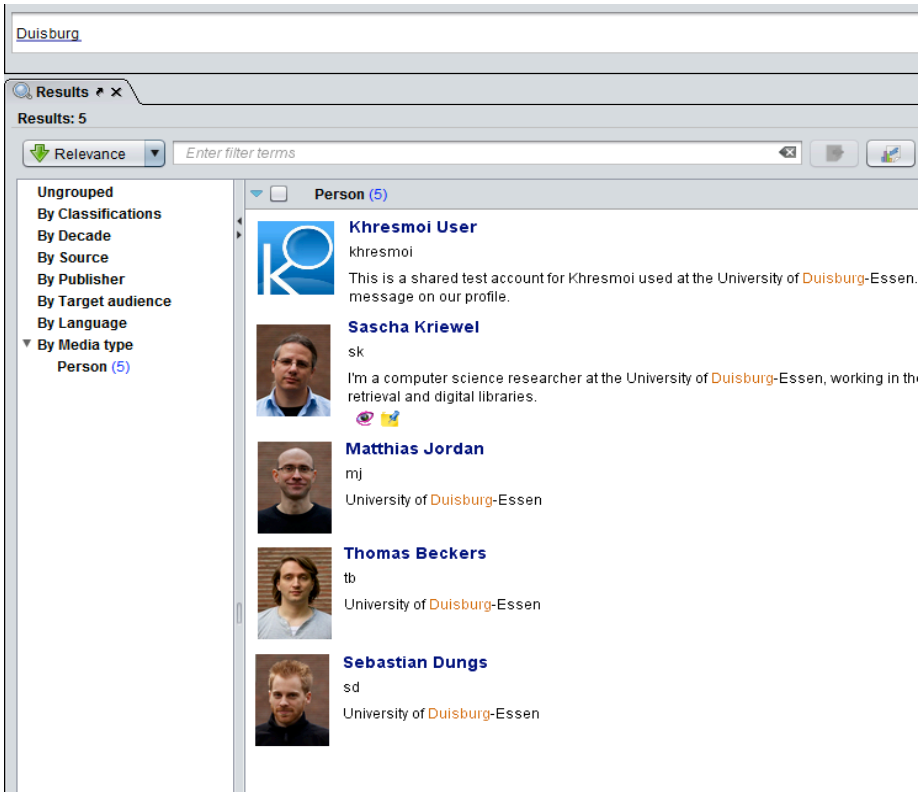

Figure 4: User search component.

3.5 Sharing Documents

Users will be able to upload their own files (new articles, images, medical reports, etc.), share them with colleagues and create discussion or annotation threads linked to these files (see Fig. 5). In order to deal with the potential need for anonymization in the files, users will have to tick a checkbox to acknowledge that he/she has to upload anonymized data.

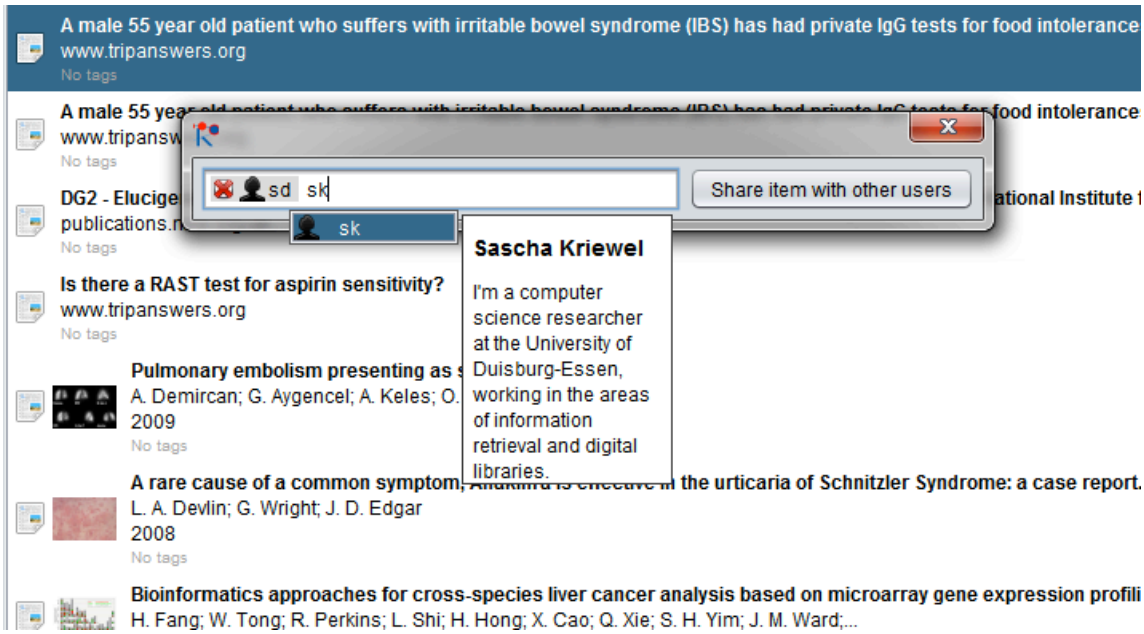

Figure 5: File sharing functionality.

3.6 Personal Library from the Khresmoi Database

Users can create and maintain a personal library of interesting items (see Fig. 6). This is similar to a 'favourites list' or a personal folder in which the user can store items. The folder is stored on the Khresmoi servers and can therefore be accessed from any device or using any of the Khresmoi Professional clients. Similar to bookmark lists used on the web, the facility to create categories in the form of tags is also provided. Future extensions might allow nested or hierarchical tags.

Items in the personal library can be shared with selected users (friends, groups). They can also be tracked by the Khresmoi system and used as an additional source of feedback in ranked retrieval or for providing implicit feedback for tactical suggestions (if a result that was retrieved by following a system suggestion is stored in the personal library, this is taken as a positive feedback for this suggestion - thereby improving suggestions for future users).

Items in the personal library might also be considered during iterative Khresmoi database updates, if such updates include deleting of items from the database or propagating up or down items in the database.

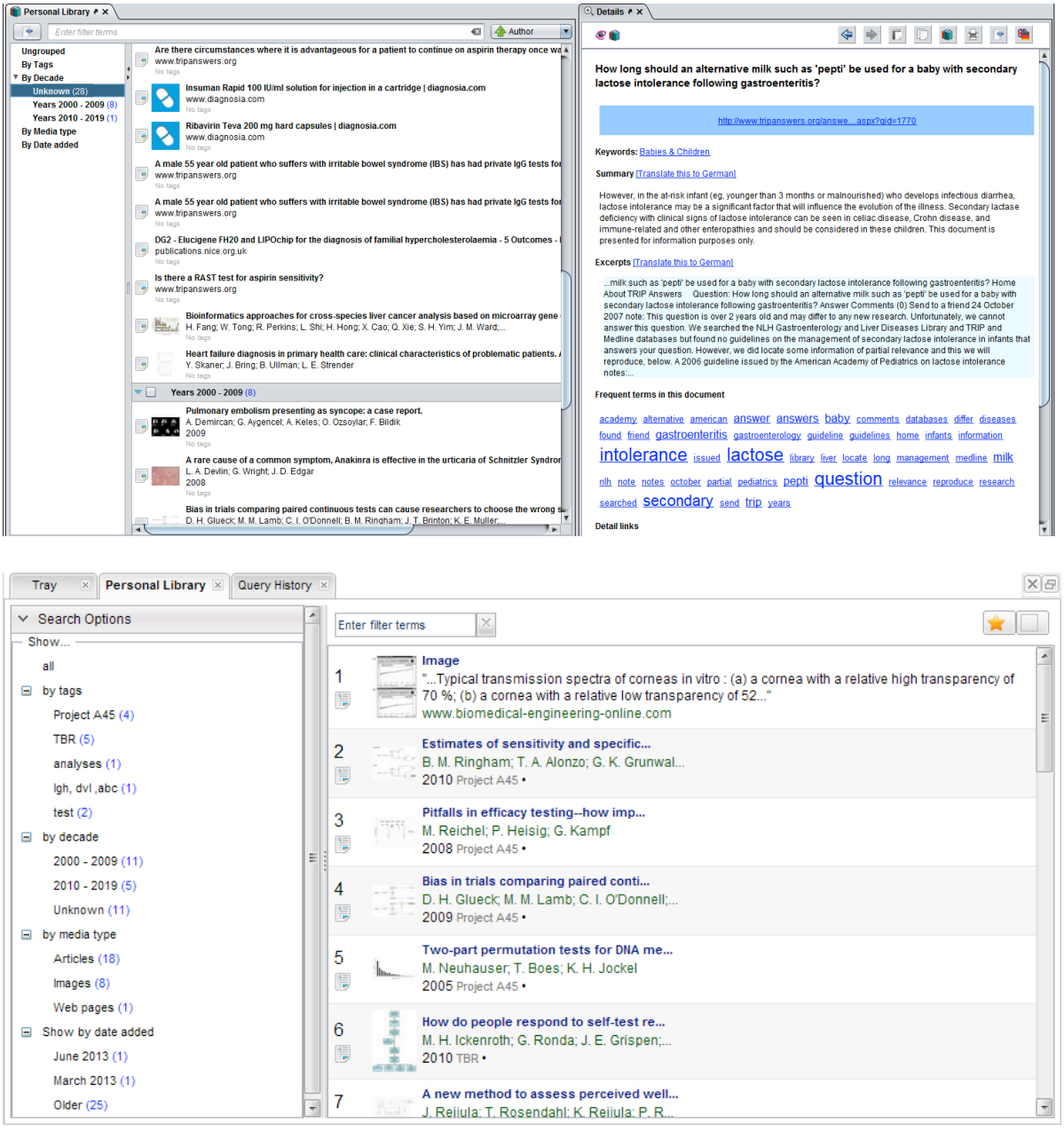

Figure 6: View of the personal library

3.7 Components Evaluation

Most of these components have been evaluated during our user-centred tests. Results can be found in [9] and [10].

4 Users Contributing to Khresmoi Knowledge

In Section 3 we described the social components developed in the Khresmoi prototype, personalising the Khresmoi experience for users and allowing users to interact with each other. In this section, we present development and research carried out to allow users contribute to Khresmoi knowledge. Firstly, the backend components both storing and processing users' potential contribution to Khresmoi knowledge are described. We then present the Khresmoi prototype features which allow users to add comments to documents, as well as a system allowing users to collaboratively annotate medical images. Finally, we give an overview of our investigation into users contributing to translation.

4.1 Backend Components: Correction Manager

The Correction Manager Component is responsible for managing user annotations on the documents provided in the Khresmoi prototype. The main goal of the Correction Manager Component is to support the prototype's social functionalities, allowing users to add and to edit information related to medical documents. The Correction Manager (CM) is exposed as a Web Service so it provides different methods via a REST (Representational State Transfer) API (Application Program Interface)[1].

The CM manages the creation and modification of annotations stored in a Solr[2] index (separately from system annotations added by WP1). Users' annotations are stored along with additional fields:

- **id:** UUID (Universally Unique Identifier) value of the annotation.
- **annotationIdRoot:** reference to another UUID if the annotation is related to another annotation.
- **comment:** annotation text itself (free text in the user's original language).
- **comment_de:** pre-calculated translation by Hadoop job to German language.
- **comment_fr:** pre-calculated translation by Hadoop job to French language.
- **comment_cs:** pre-calculated translation by Hadoop job to Czech language.
- **comments:** it stores possible annotations created over this annotation.
- **group_id:** identifier of the user group. There is no range of values and it is free text input by the user. If the *type* of the annotation is not *group*, group_id value is -1.
- **lang:** source language. The range of possible values are: {en,de,fr,cs}
- **rate:** annotation rating (from other users), ranges from 1 to 5.
- **status:** current status of the annotation. The range of values are: {pending, validated, refused}
- **tr_status:** status of the annotation translation (translated, modified, etc.).
- **timestamp:** annotation creation time.
- **type:** annotation type (public, private, group)

1 http://khresmoicloud3.es.atos.net:8080/CorrectionManager/

2 http://lucene.apache.org/solr/

In addition to the information associated with annotations, the Correction Manager also provides support for the multilingual translation of the annotations into supported pairs of languages: **en-de, en-fr, en-cs**. Currently, it is possible to translate automatically on demand or return the pre-calculated translation for any annotation into any of the supported languages.

An example annotation follows:

```
<annotation>
 <id>a1964f6b-8b4e-4979-b225-b9023bb6e49e</id>
 <annotationIdRoot>3cc4cfe4-f87c-4ff4-8e4b-986d5ef156aa</annotationIdRoot>
 <comment>test to add an annotation over an annotation</comment>
 <comments/>
 <group_id>-1</group_id>
 <lang>en</lang>
 <rate>0</rate>
 <status>pending</status>
 <timestamp>2013-06-10 13:15:46.646</timestamp>
 <type>private</type>
</annotation>
```

The Correction Manager is aligned with other components such as the Khresmoi user interface. User annotations and ratings are both stored in the same Solr index data store. Multilinguality is provided either on demand using the Khresmoi Multilingual Translator Component dynamically; or by accessing the annotation translated fields calculated in the background.
Fig. 7 describes the complete Correction Manager workflow and the functionalities and components involved in it.
It is composed of two main components:

- Batch processing: responsible for the translation of annotations into the languages supported by Khresmoi and their storage. The primary component here is the Apache Hadoop framework[3] that allows for the performance of some analysis over data input. In our case, Hadoop has been used in order to extract the comment, its original language and translate it into the different languages supported in the Khresmoi project. Hence, the implemented periodic Hadoop job should be able to access the Multilingual Translator Service in order to calculate the translations and store them offline.
- Stream processing (real time): the real time processing is focused on the communication process between the Solr framework and the Correction Manager in order to update the incoming annotations from Khresmoi end users and store them in the Lucene indexes. This part therefore represents the presentation layer where the results are shown.

3 http://hadoop.apache.org/

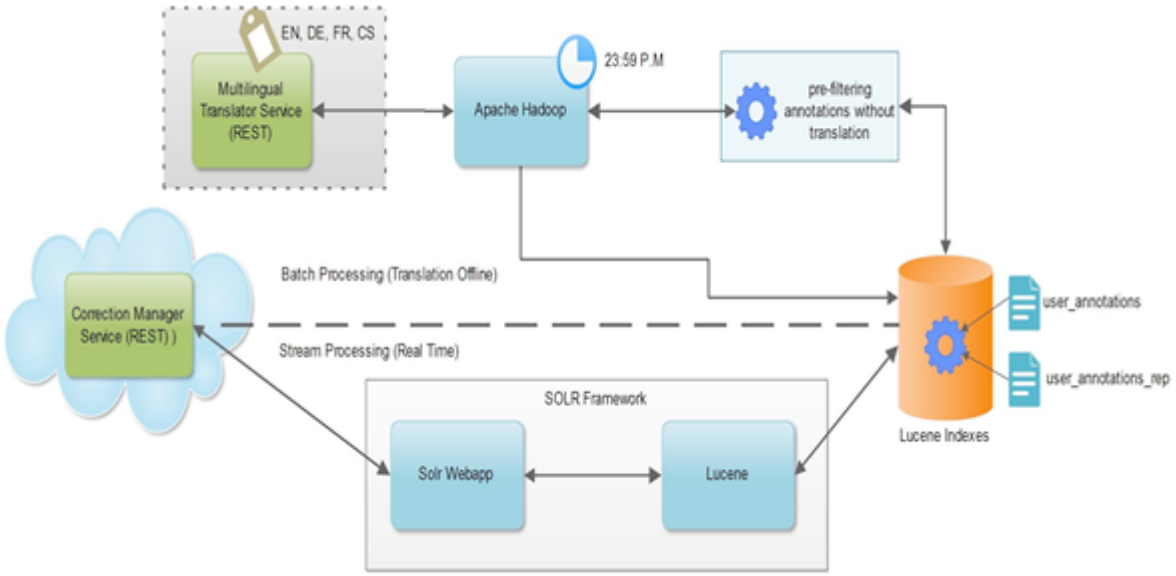

Figure 7: Correction Manager Architecture.

Correction Manager Functionalities

The main functionalities provided by the Correction Manager are the creation, the update and the deletion of annotations. Concretely, the methods exposed by the API are:

- Add annotation: add new user annotations on a concrete document. The annotation data is included in an xml request. The POST method URL is: http://khresmoicloud3.es.atos.net:8080/CorrectionManager/rest/annotations/<doc_id>/<user_id>.
  Xmlrequest:

```
<annotation>
        <comment>this is my annotation</comment>
        <comments />
        <rate>0</rate>
        <status>pending</status>
        <type>private</type>
        <lang>en</lang>
</annotation>
```

- Update annotation: update some existing user annotation that references a concrete document. The annotation data is included in an xml request. The PUT method URL is: http://khresmoicloud3.es.atos.net:8080/CorrectionManager/rest/annotations/<doc_id>/<user_id>/<annotation_id>
  Xmlrequest:

```
<annotation>
        <comment>updating the annotation</comment>
        <comments />
        <rate>2</rate>
        <status>validated</status>
        <type>public</type>
        <lang>en</lang>
</annotation>
```

- Add new child annotation: add a new user annotation that references a previous one. The annotation data is included in an xml request. The POST method URL is: http://khresmoicloud3.es.atos.net:8080/CorrectionManager/rest/annotations/<doc_id>/<user_id>/<ann_id>/<user_id>
  Xmlrequest:

```
<annotation>
        <comment>this is a new annotation on a previous annotation  </comment>
        <comments />
        <rate>2</rate>
        <status>pending</status>
        <type>private</type>
        <lang>en</lang>
</annotation>
```

- Get user annotations: retrieves all user annotations created by a given user, even those created as responses to others. The GET method URL is: http://khresmoicloud3.es.atos.net:8080/CorrectionManager/rest/annotations/user/<user_id>
- Get doc annotations: retrieves all annotations done over a given document. The GET method URL is: http://khresmoicloud3.es.atos.net:8080/CorrectionManager/rest/annotations/<doc_id>
- Get translated annotations: retrieves all annotations done by a user over a document with the comment field translated into a given language. The parameter targetLang is used to specify in the query the language which annotations are translated into. The GET method URL is: http://khresmoicloud3.es.atos.net:8080/CorrectionManager/rest/annotations/<doc_id>/<user_id>?targetLang=<lang_id>
- Delete annotation: to delete an annotation. The DELETE method URL is: http://khresmoicloud3.es.atos.net:8080/CorrectionManager/rest/annotations/<doc_id>/<user_id>/<ann_id>

4.2 User Updating Annotations

In this section we describe two ways for users to contribute to the system knowledge by adding annotations to documents: on textual documents through comments with the prototype and on images with a collaborative annotation tool.

4.2.1 Annotation Creation/Modification on Textual Data

Annotation creation/modification on textual data is part of the existing work in T3.4 of the Khresmoi DoW (DCU and ATOS).

Users can add annotations to documents and images. The Correction Manager manages the creation and the modification (correction) of annotations. Two phases are distinguished:

- Annotation phase: in the private interface, the user adds user annotations or modifies system annotations that are stored by the CM.
- Validation phase: users with a validator role validate the corrections of system annotations (authors (users) receive points and become validators when they reach a predefined points level).

The CM provides a ranked list of users by decreasing number of user points.

The Khresmoi prototype allows users to add annotations to documents, as well as to view annotations in several languages, as shown in Fig. 8.

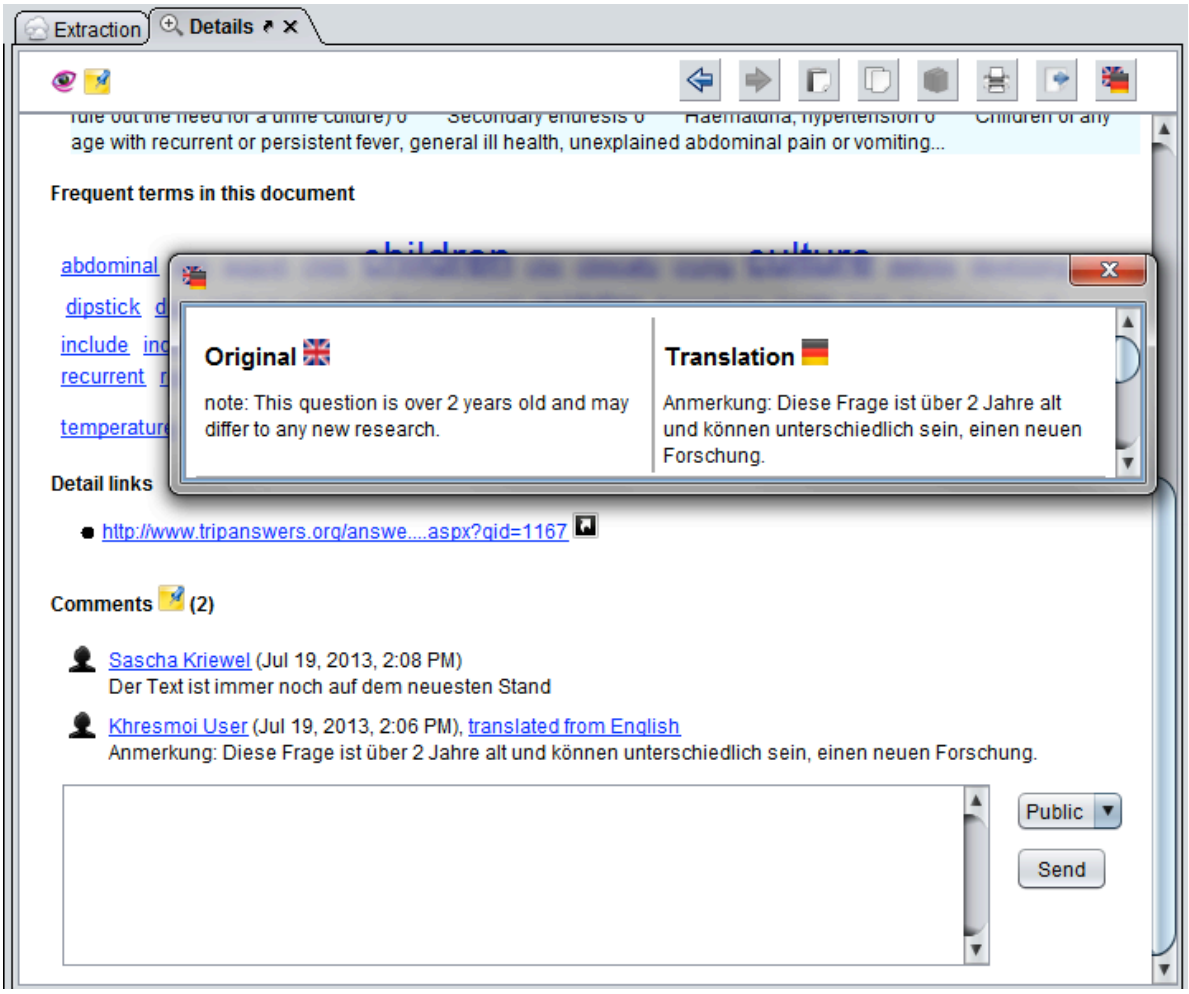

Figure 8: View of users annotations.

As it is possible to reference an existing annotation when creating a new user annotation, users can use comments or annotations to create discussion threads among colleagues or existing groups based around specific documents, so that they can for example discuss a case. These are linked directly to a file on which they can add annotations. When users are discussing a particular case (from their files), the system could suggest similar cases in the Khresmoi database, based on annotations for example.

4.2.2 Collaborative Annotation Tool for Anatomical Structures

As described in detail in Deliverable 2.1 [11], we developed a collaborative annotation tool which enables the rapid and accurate annotation of centres of view for large amounts of medical image data. The goal is to identify the location of the center of volumes (e.g. CT or MR) in the medical imaging data.

The information about the actual location of the center of a DICOM in relation to an atlas (the template depicted in Fig. 9) is important for several tasks:

1. Complement inaccurate and incomplete metadata: The meta information regarding anatomical location present in medical imaging data is coarse, and typically only available for part of the data (see Sec.5 for analysis).
2. Selection of sub-sets: The position allows selection of a subset of the images that depict a similar body region as input data for subsequent algorithms.
3. Location information can serve as partial annotation for semi-supervised learning.
4. Standard of reference for validation: The position annotation serves as a basis for the automatic localization and anatomical structure identification tasks.

The main requirements for the annotator are thus as follows:

- The position corresponding to the center of a volume has to be specified in relation to a fixed atlas.
- The annotation task has to be as easy and time-efficient as possible for the user.
- It should allow users to annotate concurrently, thus allowing for a crowd-sourcing of the annotation tasks.

The end-user GUI

To be able to easily deploy the application and to make it simple to have many users annotate at the same time (crowd-sourcing) a web-based application was developed.

For data security and privacy considerations (established during the implementation of data transfer) the annotator is only available within the firewall of the Medical University Vienna.

The user interface displays only the required information to a user. The interface is shown in Fig. 5. On the left-hand-side the DICOM is displayed through a series of 3 images, each of which show a slice through the volume, either coronal, sagittal or transversal. A small point is overlaid in the center of each slice to facilitate relating this position to the atlas.

Annotation workflow

The annotation workflow is as follows:

1. The user clicks onto the atlas to position a marker (or move the marker to reposition it) in both images until the marker corresponds to the desired position.
2. The user can then optionally select one of the two check boxes: “Needs anonymization”, and “Left/right unclear”. The first indicates that there is information present in the image which could potentially relate to the patient depicted.
3. The “Left/right unclear” indicator means that the volume shown to the user could not be related to a single position due to the symmetry of the human body.
4. The user submits the information by selecting “Submit” or "Skip", if the volume could not be positioned on the atlas.
5. A high-score table shows how many volumes were placed by each user. This incorporates social aspects into the annotation tools which might motivate the users to compete with each other.

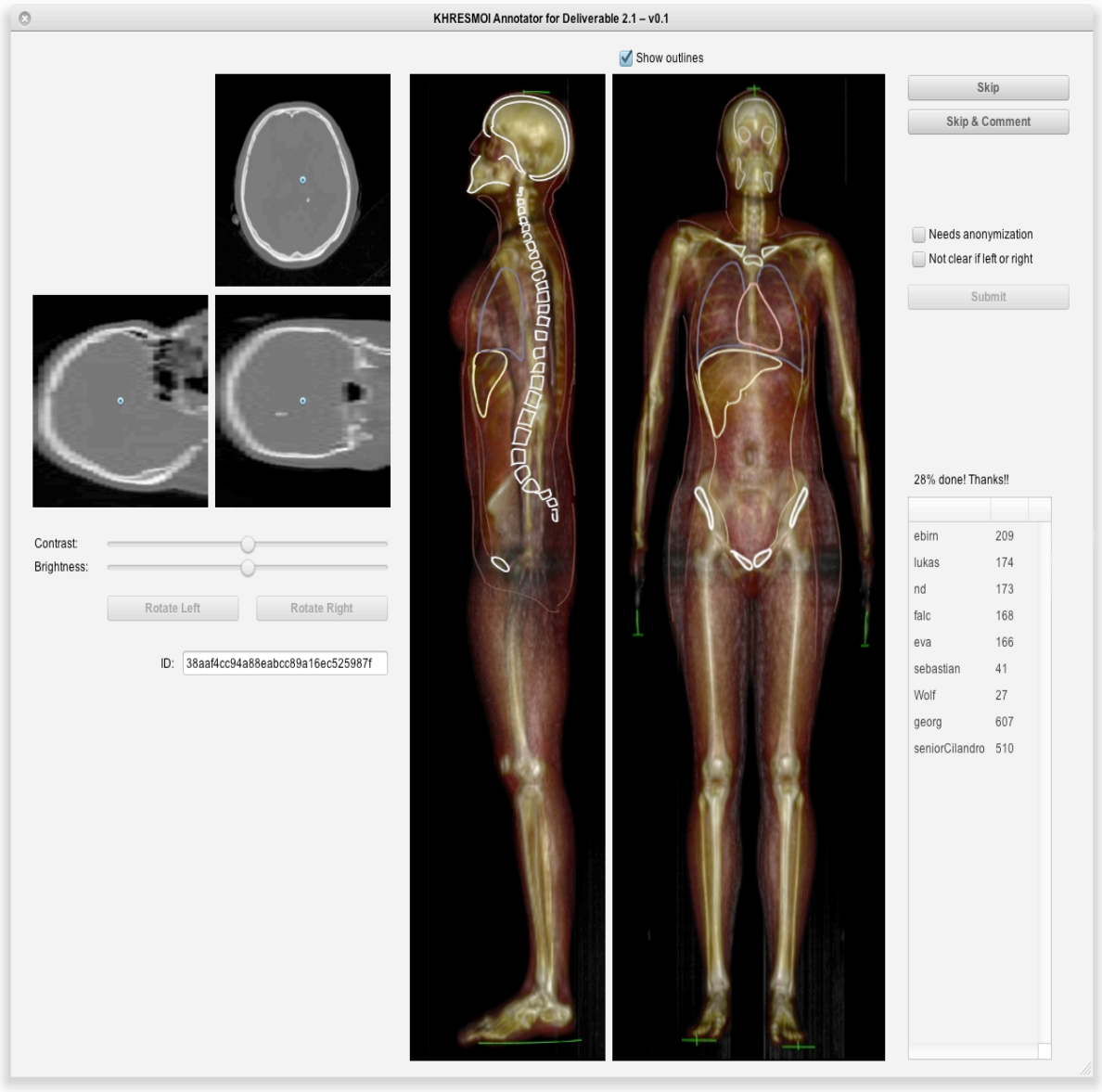

Figure 9: Collaborative online annotation tool for reference annotation of large data sets. The annotation time for one volume is approximately 3-6 seconds including loading time.

4.2.3 User Updating Translations

The physician prototype interfaces allow the users to update translations, as shown in Fig. 10. In a similar way to how translated annotations from users are displayed, original and translated text can be shown and the user can update the translation if they believe it to be erroneous. This input from the user is currently stored in the correction manager.

Research on incremental machine translation is on-going. This research is investigating how such user updates could be included in a Machine Translation (MT) system.

Existing Methods

The simplest approach to adapting MT models towards user edits is to simply include them as new parallel sentences in the training data and re-train the whole system periodically. This is naturally a very costly process and it does not react to users' corrections instantly, which can have a discouraging effect on users.

Another very simple technique is to maintain a custom dictionary (possibly one for each user). When the MT system mistranslates a word or phrase, the user can mark it in the source sentence and provide a correct translation. This suggestion is stored in the dictionary and future occurrences of this word/phrase will always be translated the way that user specified. However, this approach does not fit into the current user interface for providing corrections.

Adaptive variants of the main components of MT systems were proposed in [12], i.e. the translation model and the language model. User feedback is taken into account by updating these *cache* models on the fly. Users can thus benefit from their post-edits instantly. Data for updating the models (i.e. phrases, n-grams) can be automatically extracted from the triplet *(source, MT output, post-edit)*, requiring less effort from the user and fitting nicely into our user interface. The original method of [12] was extended and evaluated in a phrase-based scenario using a state-of-the-art MT system by [13].

Alternative techniques for *online learning* of the MT system were described in [14]. An additional feature in the phrase table is used to penalize phrases which were considered in search but were not confirmed by the user post-edit and vice versa. Additionally, an online algorithm in employed for adapting model weights (tuning) over time.

Implementation within Khresmoi

From the described techniques, we will implement the baseline method (re-training with user post-edits as additional parallel data). We will deploy the cache-based model [13] and the additional phrase feature [14] and compare them in the setting of the Khresmoi project.

In order to carry out the experiments, we will create a data set of user post-edits. A baseline MT system will be used to translate a set of user queries (around 1000 segments). The translations along with the input will be given to annotators for post-editing. The annotation will be carried out offline (outside the Khresmoi interface). The resulting data set will contain input segments (queries), baseline translations and their post-edits.

We will use a part of the created set as development data to tune the weights of the evaluated adaptation techniques. The remainder will be used for evaluation.

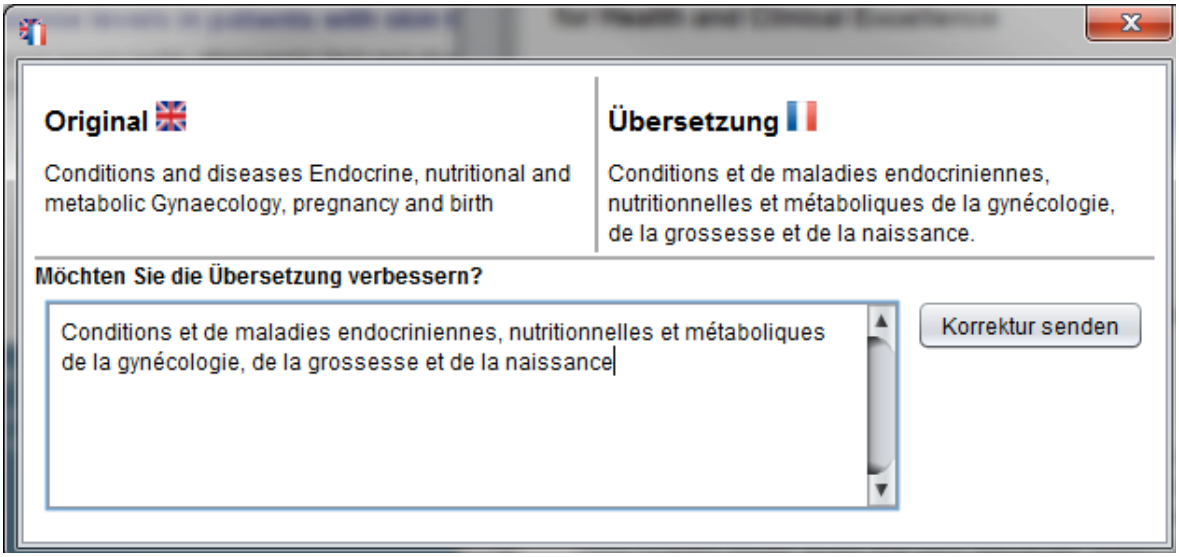

Figure 10: Translation update functionality.

5. Conclusion

This deliverable presents a set of components allowing user collaboration within the Khresmoi prototypes for physicians and radiologists. First, a set of social components have been developed. They allow users to interact on the system: create their own profile, share it, search for other users, create groups, share files and links with users or groups, and manage their personal library.

Moreover, we investigated ways to enable users to contribute to the Khresmoi knowledge. To do this, we created functionalities allowing the users to add annotations to the documents (text or images) as well as correct potential erroneous translations. A backend component called the Correction Manager has been developed in order to store users' information and annotations/corrections. Some of the components have already been evaluated during the first phase of user-centred evaluation. It is planned to include evaluation of the remaining components in the second phase.

References

[1] K. Petchev (2012). *D5.2: Large Scale Biomedical Knowledge Server*, Khresmoi project deliverable.

[2] M. Greenwood, A. Roberts, N. Aswani and P. Gooch (2012). *D1.2 Initial prototype for semantic annotation of the KHRESMOI literature*, Khresmoi project deliverable.

[3] J. Hajič, P. Pecina, J. Bystron, Z. Uresova, J. Hlavacova (2012). *D4.3 Report on results of the WP4 first evaluation phase*, Khresmoi project deliverable.

[4] L. Goeuriot, A. Hanbury, G. J. F. Jones, L. Kelly, S. Kriewel, I. Martinez Rodriguez, H. Müller, M A. Tinte (2012). Supporting Collaborative Improvement of Resources in the Khresmoi Health Information System. In Proceedings of the LREC workshop on Collaborative Resource Development and Delivery.

[5] N. Pletneva, A. Vargas, C. Boyer (2011). *D8.1.1 Requirements document for the general public health search*, Khresmoi project deliverable.

[6] M. Kritz, M. Gschwandtner, C. Boyer (2011). *D8.1.2 Requirements document for health professional search*, Khresmoi project deliverable.

[7] H. Müller, M. Holzer, G. Langs, A. Burner, R. Donner (2011). *D9.1 Report on image use behavior and requirements*, Khresmoi project deliverable.

[8] S. Dungs, S. Kriewel, M. Jordan, T. Beckers, I. Martinez, M. Angel Tinte (2013). *D3.3 Report on query specification, result presentation and personalization*, Khresmoi project deliverable.

[9] T. Beckers, S. Dungs, N. Fuhr, L. Goeuriot, J. Ignalski, M. Jordan, L. Kelly, S. Kriewel (2012). *D3.2 Report on results of the WP3 first evaluation phase*, Khresmoi project deliverable.

[10] C. Boyer, M. Gschwandtner, L. Goeuriot, J. Hajic, A. Hanbury, M. Kritz, J. Palotti, N. Pletneva, A. Sachs, M. Samwald, P. Schneller, V. Stefanov, Z. Uresova, R. Ruiz de Castaneda, F. Baroz (2012). *D10.1 Report on user tests with initial search system*, Khresmoi project deliverable.

[11] G. Langs, R. Donner, A. Burner, M. Holzer and H. Müller (2010). *D2.1: Report on medical data with ground-truth*, Khresmoi project deliverable.

[12] L. Nepveu, P. Langlais, G. Lapalme, and G. Foster (2004). *Adaptive language and translation models for interactive machine translation.* EMNLP 2004.

[13] N. Bertoldi, M. Cettolo and M. Federico (2013). *Cache-based Online Adaptation for Machine Translation Enhanced Computer Assisted Translation*. Machine Translation Summit 2013. To appear.

[14] P. Mathur, M. Cettolo and M. Federico (2013). *Online Learning Approaches in Computer Assisted Translation*. WMT 2013.