BULGARIAN ACADEMY OF SCIENCES

CYBERNETICS AND INFORMATION TECHNOLOGIES • Volume **9**, No 3

Sofia • 2009

Using Domain Knowledge to Speed up the Annotation of Digital Content with Conceptual Graphs[1]

Steffen Lalande, Kamenka Staykova**, Michel Chein***, Alain Gutierrez***, Valia Saraydarova**, Danail Dochev***

**National Audiovisual Institute, France*
***Institute of Information Technologies, 1113 Sofia*
**** Laboratory of Computer Science, Robotics, and Microelectronics, Montpellier, France*
E-mails: slalande@ina.fr {staykova, saraydarova, dochev}@iinf.bas.bg
{chein, alain.gutierrez}@lirmm.fr

Abstract: *The paper presents an investigation how the knowledge technologies may facilitate the manual annotation process by enabling and recommending the use of available annotation templates. The use of templates spares annotator's effort by giving access to already codified domain knowledge for previously annotated complex objects. The investigation continues the research work, done in compliance with 6FP IST project LOGOS and is based on knowledge formalization with the Conceptual Graphs formalism and usage of project's tools (Ontology Manager, Content Description Tool). Some results of the experiments with spatial pattern/layout recommendation mechanism were discussed on the example of "Bulgarian Iconography" content.*

Keywords: *Knowledge representation, conceptual graphs, ontologies, semantic similarity.*

1. Introduction

The continuous growth of multiple digital repositories increases the significance of the problem to open enormous existing digital resources to be easily available. Digital archives are expected to support the reuse of resources for various user

[1] This work was partially supported by IIT–BAS Project No 010079 "Methods and Tools for Processing Semantic Information".

needs. This involves re-purposing – integrating and relating existing resources into a new context. The need to specify and separate the digital resources from the information about the context of their usage led to the creation of various kinds of metadata schemas.

Different metadata formats and strategies are currently used to annotate media, including free text, weighted keywords, and conceptual structures based on domain thesaurus or domain ontologies. Domain ontology is used to conceptualize a domain by providing a formal way of describing precisely entities (usually things and events) related to this domain. Thus, when used to describe content, stored in a repository, ontologies help to reduce noise and silence during the querying process in a more efficient way than the other formats.

Annotations can be added to a document for various reasons, e.g. comments, information concerning the context of the document, summary, certification, indexation, etc. The idea of semantic annotation is to provide digital materials with computer understandable semantic metadata, i.e. information about the content and context of usage of the digital materials, represented in a declarative and semantically precise way.

Nowadays annotations have to be built manually in many cases:

– when the documents are not in electronic form or not directly available through the system;

– for documents like images, sounds, videos, etc., when there is no automatic indexing mechanism to extract high-level concepts from documents or when detailed comments about documents are essential and automatic annotations are not precise enough for the targeted goals;

– when knowledge, represented by a semantic annotation is not necessarily within the document, and cannot be deduced by general knowledge (ontology) about the document domain.

There are three main critiques concerning manually built annotations. Firstly, this is a labour-intensive and time-consuming task (even for users fluent with the model used); actually such a method can not be used for annotating large bases of documents. Secondly, manually built annotations are too author-dependent. This can be important when annotations are simply indexations, but in many cases several annotations built by different persons can be built for a sole document. Each annotation then represents a specific viewpoint, it is simply “A” way to access the document. Thirdly, for many domains there is no methodology, no strict template structure.

The present paper describes an investigation how the knowledge technologies may facilitate the manual annotation process by enabling and recommending use of available annotation templates. The use of templates spares annotator's effort by giving access to already codified domain knowledge for previously annotated complex objects. This task and the process of testing different decisions to solve it were experienced by the authors during the FP6 European project LOGOS “Knowledge-on-Demand for Ubiquitous Learning” **http://logosproject.com**. The project deals with accessing existing large-scale repositories of digitized information and facilitating their transformation into learning content, considering also possibilities for its cross-media delivery. The overall aim of the project was to contribute to the adequate enhancing and facilitating the human knowledge building during e-Learning processes.

It is common now that the knowledge, which represents “human” semantics and is used “by machines” is organized in ontologies within the knowledge based applications. The term *ontology* is borrowed from Philosophy and is defined in the field of modern Information Technologies as “formal specifications of a shared conceptualization of a certain domain” [4]. At the level of knowledge formalization the Conceptual Graphs (CG) formalism was chosen for the purposes of LOGOS project. According to the CG terminology [1], each ontology consists of a vocabulary (*Concepts*, *Relations*, *Individuals*, *Nested Types*) and Simple Graphs such as *Facts*, *Individual Graphs*, *Rules*, etc. While *Conceptual Graphs* represent formalized knowledge, *Projection* is the core operation of a knowledge processing system, which enables the enforcement of knowledge presented in the form of *Rules*, *Constraints*, etc.

The paper has the following structure: Section 2 presents briefly the domain ontology of Bulgarian Iconographical Objects (shortened as OBIO), used to provide knowledge structures for semantic annotation. Section 3 discusses means and tools used by annotators within the LOGOS Authoring Studio. Section 4 goes on with description of the knowledge-supported process of digital pictures annotation. Section 5 discusses the knowledge-based mechanisms applied for Supported Annotation Processes. Section 6 contains brief considerations about presenting similarity between annotations.

2. Knowledge structures for the use-case semantic annotation

In the work here presented, OBIO ontology is used to provide the semantic-annotation structures for the described annotation process. By itself the ontology of Bulgarian Iconographical Objects is created as one of LOGOS project domain ontologies and contains the main concepts in the subject field “Bulgarian Iconography” [8]. Specification of OBIO vocabulary and knowledge elicitation was an iterative process of interviews and brainstorming sessions with domain experts and developers of the multimedia library “Virtual Encyclopaedia of the Bulgarian Iconography” **http://mdl.cc.bas.bg** from the Institute of Mathematics and Informatics, Bulgarian Academy of Sciences. OBIO is also a specialization of the Conceptual Reference Model of the International Committee for Documentation (CIDOC), which was chosen as a top level ontological model. The CIDOC's Conceptual Reference Model is developed by the Documentation Standards Working Group of International Council of Museums [6] and is a standard for description of art collections and cultural heritage artifacts. OBIO provides specializations to domain specific concepts like *Iconographical Image* (sub-class of *Image*), Iconographic School (sub-class of *Group*), etc., but also to the more general concepts, concerning time, space, spatial relations.

OBIO is implemented in Conceptual Graphs with the Ontology Management Tool of LOGOS Authoring Studio, which is based on the instrumental system

CoGUI **http://www.lirmm.fr/cogui/,** developed at Montpellier Laboratory of Computer Science, Robotics, and Microelectronics (LIRMM). CoGUI facilitates the work of ontology engineers by all standard functions: ontology structures visualization, easy knowledge insertion and deletion, ontology verification, etc.

2.1. OBIO Vocabulary

Within an ontology represented in Conceptual Graphs, the set of *Concepts* (or Classes) is equipped with *A-KIND-OF* relation, which must not be mixed up with the *IS-A* relation; *A-KIND-OF* relation builds not necessarily a tree over the *Concepts* set. The OBIO hierarchy of *Concepts* includes 135 notions like *Iconographer*, *Iconographical Object*, *Icon*, *Manuscript Book*, etc., see a screen-shot of CoGUI presenting the OBIO *Concept's* hierarchy on Fig. 1.

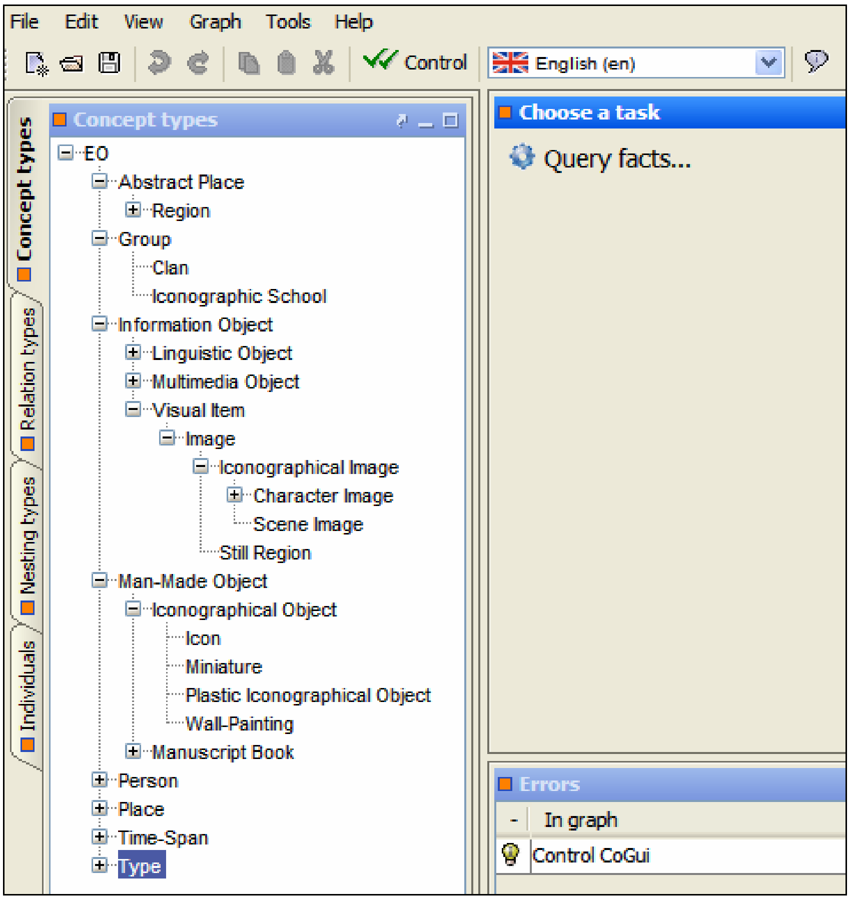

Fig. 1. OBIO hierarchy of concepts (upper level)

Relations of a Conceptual Graphs' ontology can be of any arity ≥ 1, the arity of each *Relation* is simply the number of related *Concepts*. The OBIO hierarchy of *Relations* represents the set of 42 binary relations which can be used between OBIO

Concepts (e.g. *has title, is created by, is composed of*). The set of *Relations* is also structured by a *A-KIND-OF* relation.

Individuals mark a precise entity of the discourse universe and have *Privileged Type*. During the ontology development for the LOGOS project purposes, 217 *Individuals* were introduced within OBIO; they are instances of different *Concepts* like *Iconographer*, *Iconographical School*, *Iconographical Image*, etc.

The set of ***Nesting Types*** is hierarchically structured by *A-KIND-OF* relation in the same way as the set of *Concepts* and *Relations*. *Nesting types* are used within OBIO to facilitate the construction of annotations. The root of OBIO *Nesting Types* hierarchy is called *Annotation Types* and the defined sub-classes are *Iconographical Object General Info* and *Pictorial Info.* They are meant to build up the semantic annotation of an annotated object (digital picture in our Use-Case) and to connect it to the corresponding OBIO knowledge constructions presented by *Pattern Graphs*.

2.2. OBIO knowledge constructions (contextual knowledge)

Contextual ontological knowledge is presented by Simple Graphs within the ontology. Each Simple Graph is labeled bipartite graph. One class of Simple Graph nodes (the Concept nodes) is used to represent entities of the discourse universe. A Concept node is labeled by a *Concept* type and, possibly, by an *Individual*. A Concept node which is labeled by a *Concept* without an *Individual* is called a Generic Concept node. Such a node represents an unidentified element of the concept type. The second class of Simple Graph nodes represents the relationships between the Concept nodes. Labeling the edges with 1, 2, 3, etc. is used to represent different Concepts' roles within the *Relation* (e.g. to distinguish the subject from the complement). When two different Concept nodes are linked by a Coreference Link, they represent the same entity.

Facts are Simple Graphs, which contain Individuals and express factual statements about them. The OBIO first version contains 27 Facts.

Rules represent implicit (common sense) knowledge and are posed in the ontology to spare the insertion of factual statements; then the needed conclusions could be derived from some already existing knowledge constructions. OBIO contains 5 Rules.

Prototypical Graphs present knowledge focused on a particular Concept or Relation of the ontology and define their usual contexts. The Concept or Relation on focus are called Head of the Prototypical graph. A Concept or Relation can have several prototypes. OBIO contains definitions of *Prototypical Graphs* for notions of *Iconographer*, *Iconographic School*, *Iconographical Object* and *Manuscript Book.*

Individual Graphs are Simple Graphs having a special concept node (Head), which is an *Individual* node. *Individual Graphs* represent exact knowledge about the Head-Individual. Through their *Privileged Type* the newly introduced *Individuals* could be connected to the already defined specific knowledge prototypes, and this connection makes the process of data insertion easier. *Individual Graphs* in OBIO concern mainly the knowledge about *Iconographical Images*. Fig. 5 below shows an example of an *Individual Graph*.

Pattern Graphs are Simple Graphs associated with a Nesting Type. In an annotation context of use of the ontology (like with OBIO), a Nesting Type is an annotation type and the Pattern Graph is a template for a kind of annotation. Several Pattern Graphs can be associated with the same Nesting Type. *Prototypical Graphs* and *Pattern Graphs* could be obtained by inheritance. *Pattern Graphs* for *Iconographical Image*, *Icon*, *Miniature, Plastic, Wall-Painting* are defined in OBIO ontology as far as these concepts represent objects of annotation interest.

3. The annotation process with LOGOS content description tool

3.1. The media annotation platform

The Content Description Tool (CDT) provides advanced functionalities to annotate images and audiovisual content as well as subparts of them (e.g. audiovisual segments or rectangular regions of images) with *Conceptual Graphs* (CG) based on ontologies. *Conceptual Graphs* annotations can be built by annotators through the use of a specific version of CoGUI restrained to the graph edition task which is embedded into the tool. Similar work using CG to annotate video content can be found in [5].

Some specialized graphical editors allow the temporal or spatial decomposition of a media in a user-friendly way. For instance, annotators can use an interactive timeline linked to a video player for editing temporal segment boundaries. In a similar way, rectangular sub-regions of a picture can be defined by using few mouse actions.

The CDT also embeds a specific component to deal with *Individuals* creation and search. In an "Open World" scenario, this component allows to create new *Individuals* and *Individual Graphs* dynamically during the annotation process. It is also used to search for *Individuals* stored into a shared database in order to insert them into the annotations under construction.

The Digital Object Repository provides distant services (Web services) to store and retrieve the media descriptions created by CDT as well as *Individuals*. A specific service dedicated to semantic search handles the retrieval of media segment by using CG query.

XML native databases are used to store different metadata including the conceptual graph annotations which are serialized by using the COGXML format. A HTTP Server is used to provide a shared access to the medias (image/video) and video streaming can be used to deliver fast access to the video segments during the content retrieval process.

CoGITaNT [2, 3] is a library of C^{++} classes enabling to easily build applications based on the *Conceptual Graph*s model. *CoGITaNT* provides classes for each object of the CG model (support, graph, rule, constraint, etc.) and the main operations of the model (projection, application of rules, etc.). *CoGITaNT* is used in the server platform to embed the knowledge bases and respond to graph query. A separate knowledge base is associated with each domain ontology; Fig. 2 presents a screenshot of CDT interface during an image annotation process.

3.2. Structure of the annotation

Hybrid structure

Main characteristics of the CDT include the possibility to use different description languages like Conceptual Graphs or RDF to annotate digital content. A proprietary object-oriented description language similar to Mpeg7 and based on Eclipse EMF is used internally and acts as a backbone for all the specialized CDT components. This language is used to handle textual and numeric information not effectively manageable by CG as it's in CDT version allowing Conceptual Graphs annotations. This metadata includes numeric boundaries of segment, annotation title and comment in natural language. The semantic part of the annotation is handled by Conceptual Graphs.

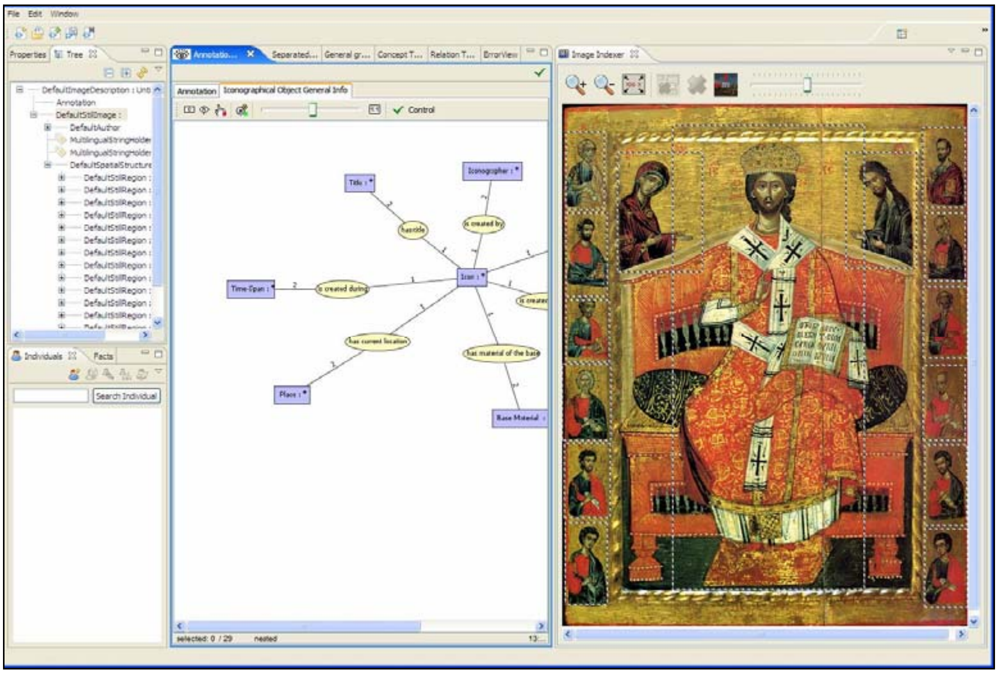

Fig. 2. Annotation of a digital image: Screenshot of CDT interface

The CG annotation structure related to the presented Use-Case

Annotations of digital pictures of *Iconographical Images* or their sub-parts are nested into specific concept nodes of the OBIO class named *Still Region*. These nodes reify the notion of annotated segments within the CG *Annotation Space*. Two different nesting types defined within OBIO are used to separate the annotation concerning general information about the icon (*Author*, *Location*, *Time-Span*, etc.) on one hand, and, on the other hand, the information related to the pictorial composition of the *Iconographical Image* or its part (ref. to Section 2 above).

A simple annotation of an *Iconographical Image* (without decomposition) leads to an isolated *Still Region* node within the *Annotation Space*, whereas the creation and annotation of image sub-parts turns to a two-level graph with some composition relations linking sub-parts and the main *Still Region*. Fig. 3 displays a fully expanded *Still-Region* node with the two types of nesting as well as connected sub-regions with their own annotations. Nesting and the composition relation used together allows to factorize knowledge about general information at a single place, e.g. within the “main” *Still Region* without loss of knowledge.

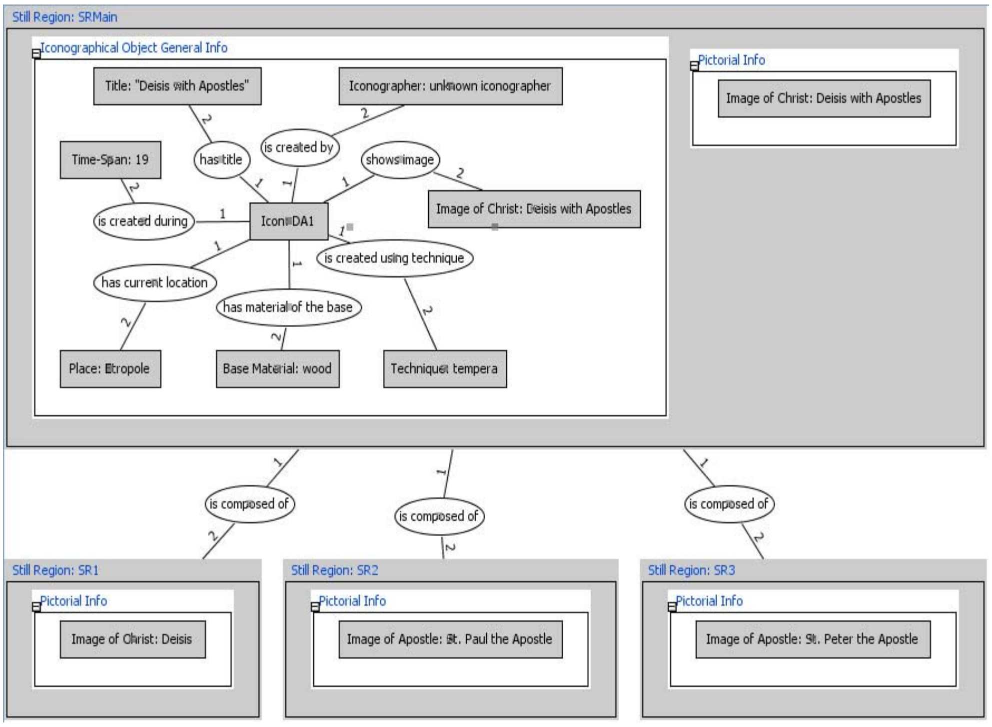

Fig. 3. Structure of complex annotation

All the annotation graphs, as well as possible *Facts* are merged together server-side into one unique graph using a normalized sum of graphs. *Individual Graphs* are also dynamically inserted within annotations for each occurrence of the related *Individuals*, enhancing in this way the query process. The resulting two-level graph constitutes the knowledge space including the annotation space where segment/annotation nodes holding their nested knowledge are potentially linked together by relations. Rules belonging to ontologies are applied on each general graph in order to create new knowledge, improving the efficiency of query by reducing silence in a controlled way.

3.3. The standard annotation process

From user’s point of view the following steps are required to produce the structure shown on Fig. 3:

– the particular media which depicts a compound *Iconographical Image* representing “Deisis with the Apostles” is loaded into the CDT;

– the annotator adds a new CG annotation for the whole picture by creating a new nesting of type *Iconographical Object General Info* by using the CoGUI component (see the screenshot on Fig. 2). Then the *Pattern Graphs* supported by this particular *Nesting Type* can be displayed using the contextual menu linked to the nesting.

– The annotator chooses the pattern related to a compound *Iconographical Image* description and a copy of this description is inserted into the nesting; rich knowledge structure that has to be specialized is provided. The annotator has the possibility to specialize some concept nodes or relations and to insert *Individuals* within the concept nodes. As *Individuals* have a *Privileged Type*, they are recommended by the system according to the concept type of the currently selected node (e.g. *Wood* is one of the proposed *Individuals* for the node typed as *Base Material* in the pattern).

– Once this individualisation/specialization task is done for all (or part of) the pattern, the annotator can fill-in data related to the *Pictorial Info* nesting by using the same process.

After this step the structure corresponding to the main *Still Region* of Fig. 3 is complete.

The user can create new *Still Regions* by using the "Image Annotator" component. The rectangular extent of the regions can be defined with a few mouse actions. The creation of both new *Still Regions'* nodes and composition relations linking these *Regions* to the main one is done behind the scene by the system. In the case of the "Deisis with the Apostles" *Iconographical Image*, the annotator can define all the 13 regions corresponding to the images of Christ and each of the Apostles, and can annotate each one in a similar way as the main region. The user can concentrate on the *Pictorial Info* annotation as the information stored in the *Iconographical Object General Info* nesting of the main region is inherited by the other *Still Regions*.

3.4. Retrieving content through graph query

Queries on the knowledge base are made by using graphs. An efficient graph operation called Projection (a Graphs Homomorphism) is used to find possible matching between the query graph and part of the targeted graphs (here the graph of all the annotations). In order to successfully project on another graph, a graph has to be a generalization of this other graph with respect to the nesting/concept/relation type inheritance hierarchies and with generic nodes being more general than individualized nodes.

Projection (Graphs Homomorphism)

Let G and H be two conceptual graphs defined over the same ontology. A Projection p from G to H is a mapping from CG, the concept node set of G, to CH, the concept node set of H, and from RG, the relation node set of G, to RH, the relation nodes of H, which preserves edges and may decrease concept and relation labels, that is: for any edge (r, c) labelled i in G, $(p(r), p(c))$ is an edge in H labelled i, furthermore for any concept or relation node x in G, the label of x is greater than or equal to the label of $p(x)$. Answering a query is generally defined through a subsumption relation, the answers to a query being all facts which are subsumed by the query. Note that the projection notion leads to the definition of a subsumption relation: a graph *G* subsumes a graph *H* if and only if there is a projection from *G* to *H*. Thus, using projection for answering query complies with the generally accepted notion of the answer to a query.

Pattern Graphs can also help to build the queries in the same way as annotations, what is in fact a good strategy to maximize chance to find a potential matching between an accurate query and the set of annotations.

In order to deal smoothly with the complex query structure required to handle the inherited structures described in the previous part, a simple query modification mechanism has been designed. The mechanism splits a single query node, which contains different nestings to a set of more complex multi-node queries. Nestings are distributed among the nodes connected by the composition relation.

Queries can additionally include a non-CG part in order to deal with textual search or with numerical properties. A joint between the graph query results and the non-CG part result is realized before returning the final result. A software component which can be embedded into third party clients has been realized to build these hybrid queries as well as to display digital objects returned as results.

3.5. Limitation of the standard approach

Despite the different improvements proposed to deal with the scalability issue, pure manual annotation of audiovisual content or complex pictures will remain tedious and time consuming task, especially when the aim is creation of fine-grained, directly reusable digital objects and, at the same time, exploitation of ontologies in order to realize accurate descriptions of the content.

One partial solution is to lighten the indexing effort by splitting it during different steps. The knowledge inheritance mechanism enables the realization of a coarse-grained description at the first step, which can be improved later by cutting of the media in fine-grained reusable subparts and by providing additional knowledge. This strategy has the drawback to require that annotation and content exploitation stages are tightly coupled, and thus content authors have to be fully involved in the annotation process.

Manual annotation can also be coupled with some automatic content analysis tools which can help to extract some knowledge coming from several modalities (automatic speech transcription, OCR, object/event visual recognition, multimodal fusion). But this approach is not always possible depending on the type of content and on the semantic level of description required for the targeted usages: it’s the well-known “Semantic Gap” problem. Another major issue is to deal often with the unavoidable noise added by the stage of automatic analysis during the manual process. In the worst case, correction of mistakes can be as costly as to annotate manually the content from scratch.

When some accurate knowledge concerning the structure of the described content is available, it can be used to guide and speed up the annotation process by proposing temporal canvas or spatial layouts. The next part describes the on-going experiment concerning the implementation of such pattern/layout recommendation mechanisms in the context of the “Bulgarian Iconography” content.

4. How the annotation process could be supported by recommendation of knowledge structures

Implemented ontologies now contain particular domain vocabulary and present knowledge constructions agreed by the experts of the domain. For the purposes of our Show-Case, *Layouts* of compound *Iconographical Images* are presented here as they are defined within the OBIO ontology.

Layouts comprise the domain experts' knowledge that a particular *Compound Iconographical Image* consists of particular images, conceptually well described. The sub-regions within the whole picture (icon), where each sub-image is presented are required by the religious cannons. For example, Fig. 4 below shows different layout schemes.

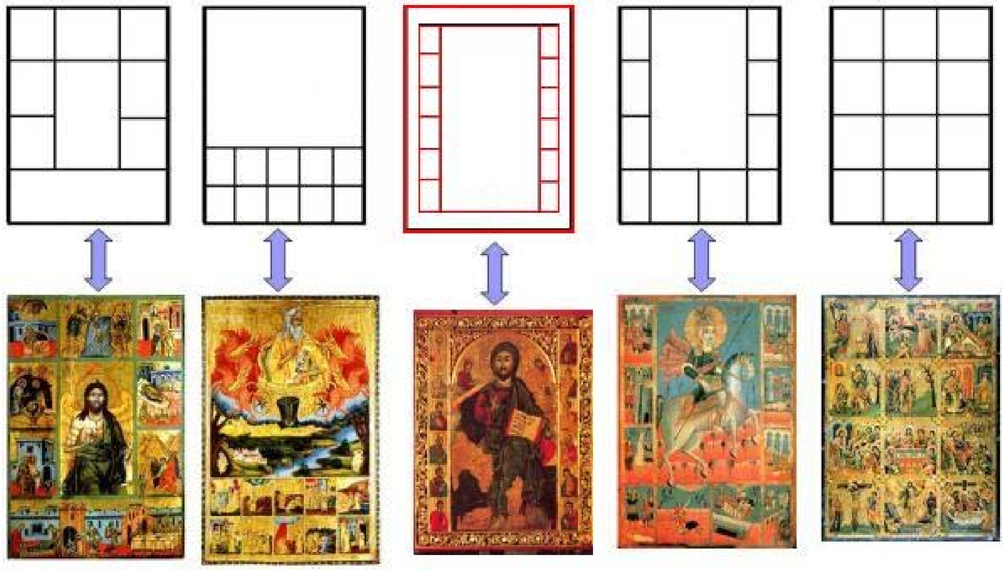

Fig. 4. Layout schemes

OBIO ontology contains 5 *Layouts*. They are connected to the *Individuals* of *Iconographical Images* which require the particular arrangement of sub-images. The *Layouts* are presented in the *Individual Graph* of the *Iconographical Image Individual*. For example, the composition of the 13 sub-images of "Deisis with the Apostles" *Iconographical Image* are presented as a *Layout* and the *Individual Graph* is shown on the Fig. 5 on the next page.

4.1. Supported annotation process: Variant 1

The objective of this variant is to support the annotation process by providing formalized domain knowledge (concerning *Compound Iconographical Images* in Use-Case), in order to improve querying.

The annotating procedure starts when the domain ontology is loaded and the digital picture, which has to be annotated is chosen. Let us have an example with a digital picture presenting a shot of the icon "Deisis with the Apostles".

To insert the needed information within *Pictorial Info*, the annotator chooses among the *Individuals* of *Iconographical Images*, and when the ontological individual “Deisis with the Apostles” is fixed, the corresponding *Layout* could be used. It means that the scheme of sub-region positions is visualized and the knowledge about the content of each sub-region is available. The annotator adjusts the size of each sub-region she/he wants to annotate and confirms the information about the content. The annotation cycle finishes with saving the annotation.

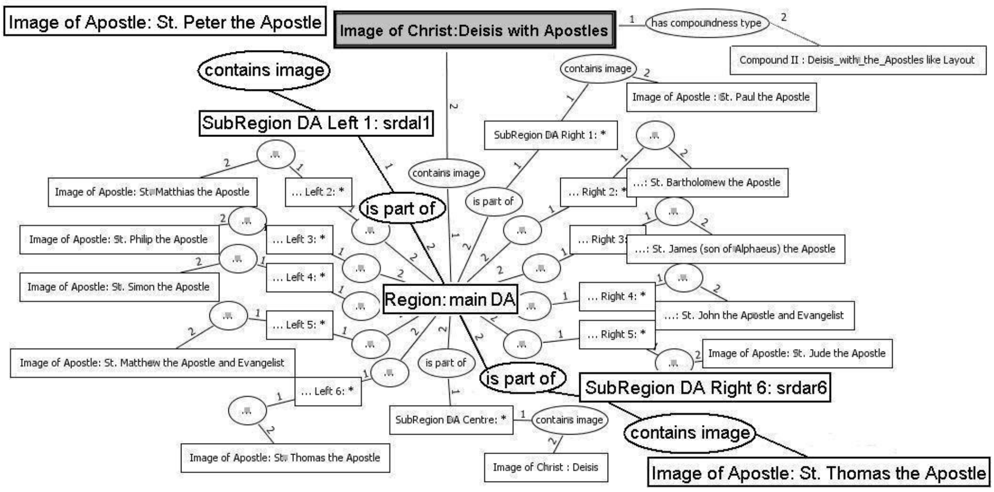

Fig. 5. The individual graph of the iconographical image “Deisis with the Apostles”

Concerning the structure of the created annotation, once the *Layout* is chosen and saved within the annotation, the ontological knowledge about the picture content is presented within the *Pictorial Info* nesting. Even if no annotation of sub-region is introduced and the annotation of the picture has the shape of “simple annotation”, the knowledge within the *Pictorial Info* is accessible and could “answer” some queries. Let us have, for example, an *Annotation Space* containing 3 simple annotations. Two of the annotations comprise information about digital pictures of icons with Apostles portrays: Apostle Paul and Apostle Peter. The third annotation concerns a digital picture of icon “Deisis with the Apostles” and is a “simple annotation” without annotated sub-regions, but within the *Pictorial Info* the *Layout* for the “Deisis with the Apostles” *Compound Iconographical Image* is saved. The knowledge that the annotated image “Deisis with the apostles” contains portrays of all the 12 Apostles (among others Apostle Paul and Apostle Peter) is formally presented there and accessible. Now, if a query for portrays of Apostle Paul is posed against the *Annotation Space*, the annotated digital picture “Deisis with the Apostles” will be retrieved in addition to the annotated self-dependent Apostle Paul’s portray, because it is known that Apostle Paul’s portray is included in “Deisis with the Apostles” *Compound Iconographical Image*.

If we take as an example the *Annotation Space*, where the portray of Apostle Peter has annotation within the “complex annotation” of “Deisis with the Apostles” digital picture, a query about Apostle Peter’s portrays will be answered by 3 results: the self-dependent Apostle Peter’s portray, the picture of “Deisis with the Apostles” icon – *Compound Iconographical Image*, and the particular annotated sub-region of the *Compound Iconographical Image* with Apostle Peter's portray.

4.2. Supported annotation process: Variant 2

The task now is to support the annotation process by recommendation (filtering) of domain knowledge provided by the system, in order to help the annotator to select the appropriate knowledge model for the current annotation among several possibilities.

In the case of *Compound Iconographical Images* the annotator provides *Pictorial Info* by attempting first to fix a sub-region on the loaded digital picture under annotation by using the spatial selection widget. It means that some spatial information about the sub-region position and size (its coordinates, in fact) is available for the system firstly. Comparison between inserted coordinates and numeric models coupled to conceptual sub-regions of the *Layouts* within the ontology gives potentially fitting *Layouts*. The system displays them by using the spatial widget, as well as associated content knowledge of each recommended *Layout*, in order to let the annotator choose.

The annotator could choose a *Layout* or could delay the recommendation by drawing other regions of interest. In the first case when the *Layout* is selected it is inserted into the *Pictorial Info* and all the other sub-region annotations are created according to it, as in Variant 1. In the second case, with spatial information about more sub-regions, the set of suggested *Layouts* could get smaller. The annotator could choose a *Layout* at any moment or could continue the annotation of sub-regions without the proposed support, and then all the information needed for sub-regions annotation has to be manually introduced. When the sub-regions of interest are annotated, the annotation process is completed by saving the annotation.

4.3. Supported annotation process: Variant 3

The objective of this annotation process variation is to investigate the possibility to retrieve and re-use knowledge already saved within the numerous created annotations instead of using the ontological knowledge structures, so quite rich *Annotation Space* is required.

From the annotator's point of view the procedure of recommending possible knowledge structures by the side of the system looks the same as the procedure, described in Variant 2. The difference is in the fact that the current annotation under construction is now compared with the set of existing annotations. In case of our example with *Compound Iconographical Images*, the system displays graphically both the spatial extent of the recommended annotations and the domain knowledge saved for each sub-region of the annotated digital pictures.

If one of the recommendations is fitting and carries useful information, the annotator could choose it and re-use the available knowledge structures. If not, the information needed for the current annotation has to be introduced manually.

5. Knowledge-management mechanisms behind the supported annotation processes

5.1. Supported annotation process: Variant 1

Decisions concerning knowledge design within OBIO ontology are important for the mechanisms of further access to the knowledge structures. The key role in managing spatial knowledge is played by the concepts under *Abstract Place* conceptual type of the ontology (see Fig. 1). This ontological class comprises the notion *Region* and some of its sub-classes are connected to the different *Layouts*' sub-regions. A number of spatial relations are defined among Regions. The concept Region carries the sense defined as "a rectangular region of a plane understood like in Euclid's Geometry". On the other hand, the notion Still Region is a key concept, which meaning (and the name) come from Mpeg 7 standard and within OBIO it represents the notion of a "digital picture, which could be visualized on a plane rectangular surface". For the purposes of annotation we use the concept *Still Region* as a basic node of the created *Annotation Space*, and it presents there "an annotation of a digital picture".

An important task is the creation of annotation (*Still Region*) from *Layout* (abstract *Region* and *Sub-Regions*). The process of annotation or *Still Region* generation must create *composed-of* relations between the main *Still Region* and the new ones. *Pictorial Info* nesting of each new *Still Region* must contain knowledge about the depicted character. Spatial coordinates of the different sub-regions are provided by a CG-external model coupled to the *Layout.* This generation process is currently handled by some external routines dealing with Conceptual Graph interpretation, generation, and with the non-CG part of the knowledge related to *Layouts*. Handling this generation process by pure CG *Rules* is foreseen when the integrality of the *Layout* knowledge including numeric component is modelled by Conceptual Graphs.

5.2. Querying this model of annotation

As knowledge about spatial composition of *Iconographical Images* is inserted into the annotation through the use of *Individual Graphs* and spread by the generation process, different content retrieval scenarios dealing with spatiality are possible, e.g.:

– Select all digital objects containing a portray of a given character: the query will consist of a single node with the character *Individual* inside a *Pictorial Info* nesting. Both the *Compound Iconographical Images* and the included *Still Regions* will respond.

– Select all digital objects containing a image of a given character, which is located in a certain region of a compound *Iconographical Image*: in this case generalization of region types can be used. *General Information* about the *Compound Iconographical Image* is factorized to the corresponding nestings of the main *Still Region*.

The mechanism of knowledge inheritance used during the content retrieval process allows simplifying the query. The queries previously described can be enhanced by some general information filtering provided in the *General Info* nesting of the query.

More complex query with constraints between regions and content, spatial relations between *Regions* can be used.

5.3. Supported annotation process: Variant 2

Comparison between coordinates provided and numeric models coupled to *Regions* of the ontology is used to generate potentially fitting regions node (one at a time) into the annotation under construction (within the *Pictorial info* nesting) of the main still region. The graphs obtained by this operation are compared (Projection) to the layouts available in the ontology (*Individual Graphs*) and Layouts which match are recommended.

The result of the described annotation process of *Compound Iconographical Images* is a number of saved annotations of the *Annotation Space*, which could be simple or compound. In respect to the *Pictorial Info* inserted in *Still Regions*, the annotations could contain concepts of *Layouts*' abstract regions and sub-regions if the *Layouts* are followed during the annotation process. If not, the concepts connected with the spatial position and size of the sub-regions are instances of the ontological concept type *Region*.

5.4. Supported annotation process: Variant 3

As in the previous case, knowledge related to spatiality is computed from coordinates and inserted into the annotation under construction. In this scenario, *Layout's* specific *Regions* are not available and more generic types of regions have to be used. The annotation is compared with the Annotation Space applying exact Projections. Results are filtered in order to deal with the un-avoidable multiple recommendations due to very strong similarity between *Iconographical Images* (and annotations) belonging to the same “cluster”.

In a more general case, connection between content and spatiality can be more “fuzzy”, and the similarity measure between the annotation under construction and the other annotations can't be computed by an “exact” Projection because this later produces a simple Boolean result which can not produce recommendation at all. A more scalable approach is to adopt a kind of approximate projection with a similarity measure connected to the number and/or type of query modifications necessary to match.

6. Projection and similarity

Probabilistic or numerical approaches are commonly used to deal with approximate searches. Nevertheless, we do not use a probabilistic approach but rather a combinatorial one (with a handful of integer parameters, easily understandable and adjustable by the user) because we think that, due to the vagueness of the issue, the user should have the greatest possible control over the search process.

In order to build a system in which the user and the search process communicate (i.e. to take into account the computer-centred view and the human-centred view), the way the system is functioning must be understandable to the user. This is not really the case for probabilistic or numerical approaches, but it is one of the main motivations for using conceptual graphs: graph transformations and computations are easily displayable and understandable by non-specialist users. More precisely, our approximate search strategy can be considered as an implementation of the second form of van Rijsbergen's Uncertainty Principle proposed by Nie: "Let d and q be two propositions, a measure of the uncertainty of $d \rightarrow q$ relatively to a knowledge base is determined by the minimal transformation from d to d' such that $d' \rightarrow q$ holds." [7]. The transformations from an annotation d to an annotation d' are based on graph operations, and these operations can also be used to transform the queries. Furthermore, it can be shown that $d' \rightarrow q$ holds if and only if $d \rightarrow q'$ holds, where q' is obtained from q by a dual transformation of a transformation from d to d'. Thus, in some ways, modifying d or q are equivalent, and our approach can also be considered as a dual technique of those classically used in query reformulation (partial query, term masking, expansion, etc.), because the content descriptions are transformed instead of transforming query statements. One advantage of modifying annotations rather than queries is that the annotation base can be enriched in different ways by using the system. This can be seen as a technique for improving annotations using a relevance feedback given by users (we have not yet developed this aspect). Remark that, if the annotator can see the modifications that the system makes to his/her annotations when answering questions of the user, this could be an interesting form of indirect communication between users and annotators. The approximate answers are ordered by a ranking procedure, based on the total pre-order relation defined over the transformation set, and this order represents a relevance score (see [1, Chapter 13 and Chapter 7], for more details about approximate search).

7. Conclusion

The paper presents investigation of a possible way to facilitate the manual annotation process by enabling and recommending the use of available annotation templates. Some results of an on-going experiment with spatial pattern/layout recommendation mechanism were discussed on the example of "Bulgarian Iconography" content. As in all attempts to support real non-trivial information processes the functionality of the proposed solutions has to be checked with more experiments. Nevertheless it is clear, that to be usable, an annotation environment has to be supplied with well thought out easy-to-use graphical user interfaces, hiding from the user the details of intricate navigation on semantic structures. From another viewpoint, any help from (semi-)automatic extraction of features from digital documents should lessen the burden on annotators, especially when annotating large volumes of documents.

References

1. C h e i n, M., M u g n i e r, M.-L. Graph-Based Knowledge Representation. London, Springer-Verlag, 2009.
2. G e n e s t, D., E. S a l v a t. A Platform Allowing Typed Nested Graphs: How CoGITo Became CoGITaNT. – In: Proc. of the 6th Int. Conference on Conceptual Structures, Montpellier, France, August 98. Lecture Notes in AI, Vol. **1453**, Springer 1998, 154-161.
3. G e n e s t, D., M. C h e i n. A Content-Search Information Retrieval Process Based on Conceptual Graphs. – Knowledge and Information Systems, Vol. **8**, 2005, 292-309.
4. G r u b e r, T. R. Toward Principles for the Design of Ontologies Used for Knowledge Sharing. Presented at the Padua Workshop on Formal Ontology, March 1993. – International Journal of Human-Computer Studies, Vol. **43**, Issues 4-5, November 1995, 907-928 (Later Published).
5. N a n a r d, M., J. N a n a r d. Cumulating and Sharing End Users Knowledge to Improve Video Indexing in a Video Digital Library. – In: Proc. of the First ACM/IEEE-CS Joint Conference on Digital Libraries, Roanoke, VA: ACM Press, 2001, 282-289.
6. N. Crofts, M. Doerr, T. Gill, S. Stead, M. Stiff (Eds.). Definition of the CIDOC Conceptual Reference Model, March 2008.
   http://cidoc.ics.forth.gr/
7. N i e, J., F. L e p a g e. Toward a Broader Logical Model for Information Retrieval, Information Retrieval: Uncertainty and Logics, Advanced Models for the Representation and Retrieval of Information. Dordrecht, Kluwer Academic Publishers, 1998, 17-38.
8. S t a y k o v a, K., D. D o c h e v. Ontology “Bulgarian Iconographical Objects” – Creation and Experimental Use. – Cybernetics and Information Technologies, Vol. **9**, 2009, No 1, 25-36.