Documentation, Cataloging, and Query by Navigation: A Practical and Sound Approach

F.J.M. Bosman[1], P.D. Bruza[2], Th.P. van der Weide[3], and L.V.M. Weusten[4]

[1] Department of Information Systems, University of Nijmegen, Toernooiveld, NL-6525 ED Nijmegen, The Netherlands robb@cs.kun.nl
[2] bruza@icis.qut.edu.au
[3] tvdw@cs.kun.nl
[4] Department of Letteren, University of Nijmegen NL-6525 ED Nijmegen, The Netherlands

Abstract. In this paper we discuss the construction of an automated information system for a collection of visual reproductions of art objects. Special attention is payed to the economical aspects of such a system, which appears to be mainly a problem of data entry. An approach is discussed to make this feasible, which also strongly provokes consistency between descriptions. Another main target of such a system is the capability for effective disclosure. This requires a disclosure mechanism on descriptions which is easy to handle by non technical users. We show the usefulness of query by navigation for this purpose. It allows the searcher to stepwise build a query in terms of (semi-)natural language. At each step, the searcher is presented with context sensitive information.
The resulting system is described and we discuss an experiment of its use.

1 Introduction

It is typical for a department of History of Art to administer a large collection of reproductions of art objects. For example, the Department of History of Art of the University at Nijmegen, The Netherlands has a collection of about 60,000 slides depicting art objects, covering all areas of art, and originating worldwide. The purpose for having such large collections is to allow both students and teachers to quickly and efficiently find slides they need as a support during teaching activities. Traditionally, card indexes are used to provide access into the slide collection through a number of keys. Usually, an index on author name, and an index on artwork topic are available.

Such a manual disclosure mechanism has several drawbacks. First, the procedures for cataloging are rather laborious and time consuming for the documentalist. Second, iconographic indexing is a subjective activity. As a result, the classification of the documentalist might be quite different from that of a searching person. For example, it may occur that a searcher will search in vain for a slide depicting merchants in the category *trade* , while these images are stored in the category *professions*.

C. Nikolaou, C. Stephanidis (Eds.): ECDL'98, LNCS 1513, pp. 459–477, 1998.

The need for automation has been largely recognized these days. This has led to a number of approaches, which all seem to have their own intrinsic limitations. For example, in 1976 Bildarchiv Foto Marburg started with the disclosure of its large arthistorical photocollection. This gave them a restraining lead: the available software at that time did not focus as much on retrieval as it did on data-input, resulting in a well filled database with inferior query-possibilities. Other institutions spent a lot of time developing a very detailed description-format, demanding a too time devouring research for each description, resulting in a too low processing rate.

The Odilon project started as a cooperation between the Department of History of Art and the Department of Computing Science, both at Nijmegen. It was continued in the Odilon for Windows project, a cooperation with Fratelli Alinari in Florence, Italy for disclosing a part of their enormous collection. This has resulted in a new, Windows based, implementation of Odilon.

First a thorough analysis of the situation in the slide library was performed, not only from a technical point of view, but also from a economical, and, last but not least, a cognitive point of view. A first conclusion was that a hypertext-like approach could meet the objectives of the different areas of competence, provided an improved mechanism for disclosure would be used. This mechanism should be easy to use for all categories of searchers, and help them to find their way through a vast amount of information. Secondly, if being user friendly is considered to be a critical success factor, being *input friendly* directly addresses "le raison d'être" of the system. Filling the system with input data is the main cost factor, in which face all other costs vanish to nothing.

The program resulting from the Odilon project has been tested extensively by the Department of Art History, since its first running prototype in 1991. At the moment, more than 80% of the slide descriptions have been entered into the system. Experiences of documentalists are promising. Furthermore, students as well as teachers and researchers of History of Art are more and more basing their haunt for pictorial information on Odilon.

The structure of the paper is as follows. In section 2, the architecture of Odilon is presented. Section 3 introduces two disclosure mechanisms for subject classification; ICONCLASS and *HyperIndex*, the indexing technique based on so-called index expressions. Both are part of Odilon. In section 4, we discuss the HyperIndex in Odilon and the experiences so far, from a practical point of view. Section 5 contains a number of conclusions, and gives guidelines for further research.

2 The Architecture of Odilon

The intention of the slides library is to provide images and information of art objects. Note that several slides can be available for the same art object, to show its different aspects. The slides are used both by scholars and professors. They can be used for research and for preparing lectures. The library is also available for non-educational purposes such as advertisements and public relations.

2.1 System Approach

In the manual system, slides were described by indexcards. In order to make the enormous amount of information contained in the existing slide catalog easier and better accessible, it was decided to build an automated information system. Globally, this information system is aimed at two distinct targets:

1. to relieve the documentalist from laborious, time consuming cataloging and maintenance tasks.
2. to provide an effective disclosure mechanism.

These requirements had to be satisfied, taking into account the system as used before the introduction of the automated system. That is, positive elements of the former catalog had to be preserved, while the negative elements were to be eliminated. This resulted in some additional requirements regarding the automated information system:

- The system should let the documentalist record the slide descriptions from their index cards. This suggests maintaining the fields slide number, date, subject, technique, material, size, artist and place, which were recorded on the indexcards.
- Information should be entered using a system dictionairy to avoid synonyms and spelling errors. This will result in a standard language with a predetermined terminology to yield homogeneous and consistent information. Menu options should be used where possible to provide the documentalist and searchers with easier and faster access to the data.
- When users are consulting the information system, they should be able to refine and join previous selections.
- The not standardized classification on topics should be replaced by the ICONCLASS system ([Waa85]). This system is described in section 3.1.
- To prevent tampering with important information, there has to be a priority distinction between the documentalist and other users. Only the documentalist will be allowed to add new slide descriptions or to change incorrect descriptions. Obviously, all users must be able to consult the information system.

3 Classification

When cataloguing artobjects or visual reproductions of artobjects, it is desirable to be able to classify the subject descriptions. It is as least as likely that a searcher would want to find a painting of *sunflowers* (subject) as a painting with *oil on canvas* (technique). Traditionally, keywords were used to accomplish this. In this section we discuss two different ways to classify subject discriptions.

3.1 ICONCLASS

ICONCLASS is a standardized, well documented system for classifying representations. The term ICONCLASS is derived from **Icon**ographic **class**ification.

ICONCLASS has been designed as a framework for characterizing and searching art objects independent from any language. Subjects, themes, and motifs in the art of the Western world can be classified. In ICONCLASS nine *Main Divisions* form the basic classification. Each division has a number of subclasses; subclasses have subclasses of their own etc.

Basic Principles of Classification

ICONCLASS provides a collosal potential for classifying art objects. For example a representation of *children playing in a park* may be classified by:

- Main division: Society, civilization, culture.
- Primary subdivision: Recreation, amusement.
- Secondary subdivision: Recreation.
- Tertiary subdivision: Enjoying nature.

In order to make this classification manageable, a coding scheme for classes is introduced (see Figure 1). The position of each class within its superclass is coded by an according single letter or digit. The code which identifies a specific class is obtained by concatenating the codes from the classification path. For the example described above, this would lead to 43B1.

The combination of digits and letters, indicating together an iconographic item is called a *notation*. This results in such notations as 11Q7612 (=christmas), 25H213 (=river), or 34B11 (=dog).

Note that the secondary subdivisions are not indicated by digits but by letters in their alphabetical order. This has three advantages:

- more subdivisions: 25 secondary subdivisions can be distinguished instead of 9. (The letter J is not used)
- greater legibility of notations.

Problems with ICONCLASS

Describing representations, from its very nature, depends on the interpretation of the person involved. As with each characterization mechanism, a description shows only some aspects of the object. Which aspects are recognized depends on the point of view of this person at that very moment. The main challenge of Information Retrieval is to try to overcome the problem of vague descriptions and vague queries as good as possible. For ICONCLASS this problem manifests itself as follows:

1. Imagine a slide depicting a painting of a man walking through a field with his dog and a flock of sheep. The following classes may be used to characterize this slide: (1) 25F (=animals), (2) 25F2 (=mammals), (3) 25H (=landscapes), (4) 34 (=man and animal), (5) 34A (=taming and training

Main divisions

1	Religion and magic
2	Nature
3	Human being, Man in general
4	Society, civilization, culture
5	Abstract ideas and concepts
6	History
7	Bible
8	Literature
9	Classical mythology, history

Primary subdivisions

41	Material aspects of life
42	Family, descendance
43	Recreation, amusement
etc.	

Secondary subdivisions

43A	Festivities
43B	Recreation
43C	Sports, games + performances
etc.	

Tertiary subdivisions

43B1	Enjoying nature
43B2	Open-air recreation
43B3	Indoor recreation
etc.	

Fig. 1. The hierarchical structure of ICONCLASS

of animals), (6) 34B11 (=dog), (7) 34C (=protection of animals), (8) 43B1 (=enjoying nature), (9) 46A14 (=farmers), (10) 46A8 (=unusual manners of living), (11) 47I221 (=herding), or (12) 35 (= pastorals, Arcadian scenes). It is clear that for both the documentalist and the searcher a choice of classification is usually hard to make.

2. The hierarchical structure of ICONCLASS makes a hierarchical use of the class structure most appropriate. This way of working may lead to problems, as it is not always clear a priori in what subclass a term should be located. This problem has been recognized, and led to the production of an (automated) index system for the use of ICONCLASS. This index provides a translation of terms to classes in some specified context. This, however, does not solve the problem metioned above. For example, when trying to find all slides of an artwork depicting a *baby*, the user is confronted with the following dilemma: what is the relevant context? In other words, is baby a

subclassification of main division 3 (= Human being, man in general), or main division 4 (= Society, civilization, culture)?
3. ICONCLASS is also restricted in its expressive power. For example, it cannot be used to classify architecture.

3.2 HyperIndices

A useful aid when searching for information is an index such as found in a book. The advantage of an index is that the searcher is not faced with the problem of having to express their information need in the form of a query. Also, for those searchers who have no clear notion of their information need, an index is often helpful to clarify this need.

A HyperIndex is an index organized in the form of a hypertext. This form of index is formally introduced in [Bru90] where it is presented in the framework of a two level architecture [BW90b]. (See figure 2). The HyperIndex in Odilon is the first implementation of this disclosure mechanism.

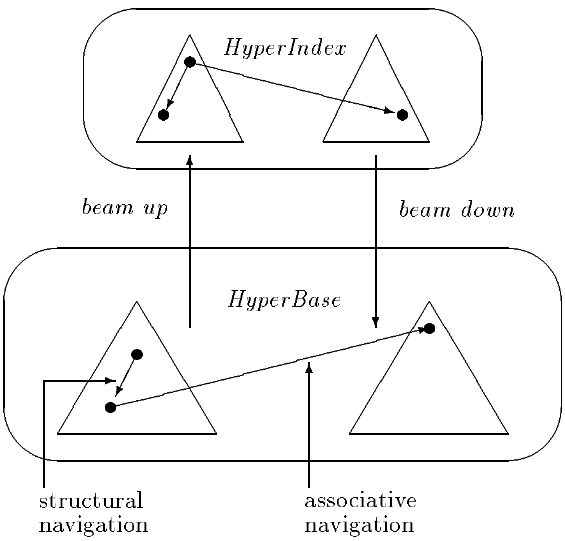

Fig. 2. The Architecture of Two Level Hypermedia

The HyperBase can be understood as hypermedia as is typical in current systems [Con87,SK89], but which is fragment based with a strong emphasis on the notion of a *view*.

The HyperIndex forms the top level of the architecture. It is a hypertext of index terms which index the underlying HyperBase. The feature of the HyperIndex is its structure. (See figure 3). The structure provides the opportunity to browse through the index terms in an organized fashion. This facet not only facilitates information retrieval but can guide a possibly distracted or lost searcher. The basis of the organized search capability is the fact that any focus (the current index term being scanned by the searcher), can be *refined* (context contraction) or *enlarged* (context extension). Figure 4 shows two examples of refining. In this figure we see that a node consists of a number of entries where all entries represent indexing information. The first entry represents the current focus of the user in the HyperIndex, and the other entries are *buttons* which can be activated to refine (△) or enlarge (▽) the current focus. By enlargement or refinement, the button activated becomes the new focus.

When an index term is found which describes the information need, the objects from the underlying HyperBase which are characterized by this index term can be retrieved and examined. This operation is referred to as a *beam down* because the searcher is transferred from the HyperIndex to a view at the HyperBase level which is constructed from the relevant objects. Navigating through the HyperIndex and retrieving information in the above way has been coined *Query By Navigation* [BW90b].

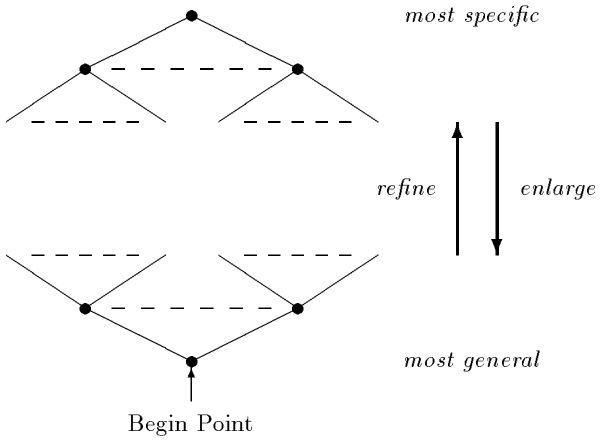

Fig. 3. Conceptual View of a HyperIndex

The index terms in the HyperIndex have the form of so called *index expressions* [Bru90,BW90a]. In contrast to keywords or term phrases index expressions have a structure (see figure 5). This figure also shows that the relationships between terms are also modeled. These relationships are termed *connectors*, or *operators*, which are basically restricted to prepositions. Figure 6 shows some of the allowable connectors and the relationship types that denote.

From the structure of an index expression the so called power index expression can be derived. This is a lattice-like structure which supports Query by

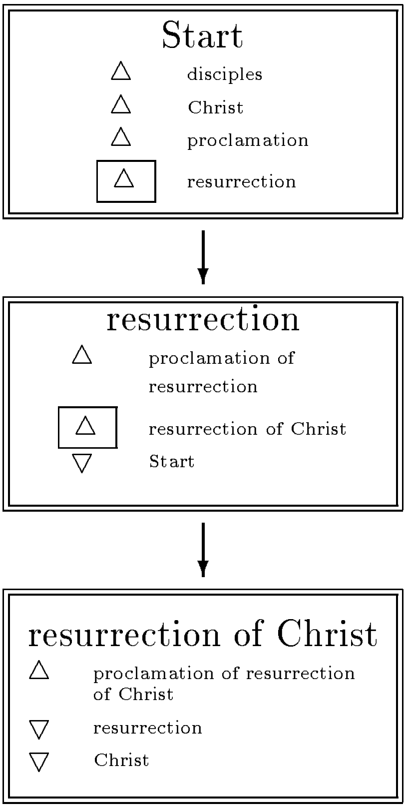

Fig. 4. Example of Refining in the HyperIndex

Navigation. For example, the refinement operation seen from the user interface perspective in figure 4 can be understood in terms of the power index expression depicted in figure 7. The HyperIndex is a union of power index expressions.

3.3 The Derivation of Index Expressions from Title Descriptions

An important process in constructing the HyperIndex is the derivation of index expressions from the slide titles. As there are 60,000 slides in the slide library it is not practically feasible to do this manually. A transducer was therefore implemented that translates a title description into an index expression. From the resultant index expressions the power index expressions are then generated resulting in the HyperIndex.

It has been observed in [BW91,B98] that titles of documents, section subsections and figures often have a form very similar to that of index expressions. This similarity becomes even more evident when articles such as the and a are

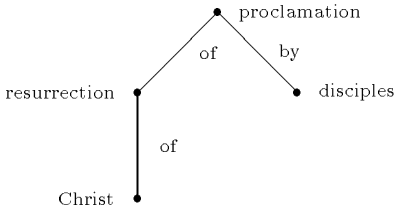

Fig. 5. Example Index Expression Representation

Connector	*Rel. Type*	*Examples*
of	possession action-object	castle of queen pollination of crops
by	action-agent	voting by students
in, on, *etc.*	position	trees in garden
to, on, for, in	directed assoc-iation	attitudes to courses research on voting
with, ·, and	association	Napoleon with army fruit · trees
as	equivalence	humans as searchers

Fig. 6. Connectors and their associated Relationship Types

removed. The first phase of the transducer was to remove such articles. For example,

On the rejection of the mitre and of the crosier by Saint Bruno

results in

On rejection of mitre and of crosier by Saint Bruno

While removing articles the transducer also checks for so called connector irregularities. This occurs when there are two or more connectors between terms. As there may only be one such connector, the transducer chooses the first. On the other hand, if there are no connectors between two successive terms, a null connector is inserted. Furthermore, connectors at the beginning of the descriptor

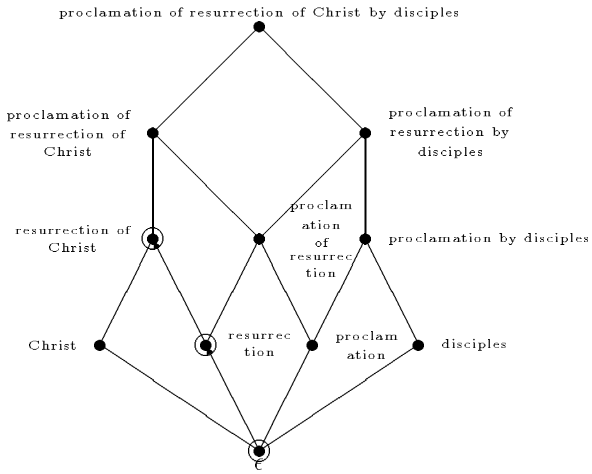

Fig. 7. Refinement within a lattice structure

are removed. The running example after connector irregularities have been resolved looks like the following:

rejection of mitre and crosier by Saint · Bruno

The problem remains to detect the underlying structure. This is achieved by employing a two level priority scheme in relation to the connectors. This priority scheme was a result of the observation that some connectors bind terms more strongly. The connectors deemed to bind most strongly are ·, and, with and of. These connectors have therefore priority 0. In terms of figure 6 the first three of these connectors form the *association* term relationship type. On the case of the connector of, we found it particularly binding in the context of a *possession* relationship type. The remaining connectors all receive priority 1 because we observed no consistent behavior which could form the basis of criteria which would lead to more priorities.

The connector priorities are used to derive an underlying tree structure in the descriptor. This structure is built up as the descriptor is scanned left to right. The heuristic used is that the tree is *deepened* if a high priority connector is detected, otherwise it is *broadened* at the root.

Figure 8 shows the successive build up of the structure derived from the running example. Up to the point of parsing by the structure developed thus far would be two levels deep as both and and of have priority 0. When by is parsed the tree is broadened at the root. Thereafter no broadening occurs.

The transducer turned out to be surprisingly good. In a test of more than a thousand descriptions parsed, less than ten percent needed manual intervention. These were mostly descriptions that weren't in the passive form, for example,

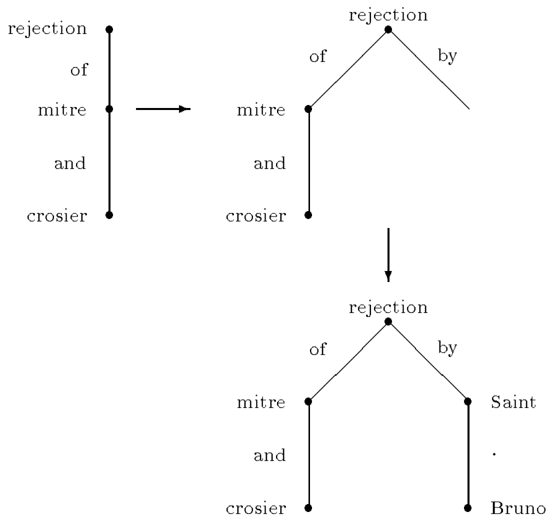

Fig. 8. Build up of index expression structure

Saint Bruno rejects mitre and crosier instead of Rejection of mitre and crosier by Saint Bruno.

We were fortunate that the languages used in title descriptions (Dutch, German, English, French and Italian) all have connectors embedded in them in the form of prepositions and basically for this reason the transducer worked equally effectively for all languages. Note, however, that automatic derivation of index expressions from titles in languages such as Finnish would not have been possible using the above scheme, because such languages do not contain prepositions.

Finally, the two level priority scheme generally produced good structures, where "good" means that a large percentage of the index (sub)expressions generated from these structures for the HyperIndex were meaningful.

4 The Odilon HyperIndex System

4.1 Effectiveness of HyperIndex

This section reports a study carried out with Odilon to investigate the effectiveness of the HyperIndex. One aim was to gather information which could be used to improve the HyperIndex implementation. Query by navigation seems to be a useful concept from a technical point of view, but how effective is it in practice? Therefor the effectiveness of the HyperIndex was compared with ICONCLASS. This was done by comparing the results of searchers using either ICONCLASS or HyperIndex to answer a number of questions. In order to do this formal criteria are needed. For more information, see [BBB91].

Effectiveness Criteria

Given that n is the set of found slidenumbers, and N is the set of relevant slidenumbers, then

Recall : The recall of an answer set is the fraction of relevant slidenumbers retrieved.

Precision : The precision of an answer set is the fraction of retrieved slidenumbers that are relevant.

Two other criteria can be used to measure the result of both ICONCLASS and HyperIndex: The number of **Logical decisions** (see section 4.1) by a searcher and the number of times the searcher wants to see the slides belonging to the current selection, referred to as **Show** in the following text.

The User Interface

An important facet of indexes is how they are displayed to the user, because this is a factor to their effectiveness. When the users enter the HyperIndex they are presented with a screen like that depicted in figure 9.

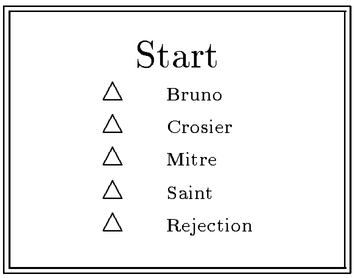

Fig. 9. Example start node of the HyperIndex

The first screen is the gateway into the HyperIndex. All terms in the HyperIndex will be directly accessible from this screen. To facilitate easy access to descriptions the *Finder* was implemented using the word wheel technique. Users are presented with a window in the screen where they can type in a keyword. After typing the first letter a pop up window appears showing all keywords beginning with that letter arranged in alphabetical order. As the searchers continue to type, the entries in the pop up window are adjusted accordingly. At any time the searchers can choose a keyword in the pop up window thus avoiding further typing. This choice becomes the begin point for Query by Navigation [BW90b]. From any focus in the HyperIndex the user can navigate to more specific indexing information (refining), or to more general information (enlarging).

Database Content

The total number of slides in the database for the experiment described below was 437. 153 of these were architecture slides, having no ICONCLASS. Leaving 284 to select from by using ICONCLASS. The 437 slide subject descriptions were used to build the HyperIndex. The resultant HyperIndex contained 2434 entries of which 650 were keywords. This means that the starting screen (see figure 9) contained 650 entries.

The Users

Three groups of users were identified in this experiment. The first group consisted of art historians who used the ICONCLASS system. It was decided to use only art historians for this part of the experiment, because as stated before ICONCLASS users need at least some knowledge of both art and ICONCLASS to use it effectively. The second group also consisted of art historians. This group used the HyperIndex system. In the third group computer scientists with little or no knowledge of art history, used HyperIndex. The motivation to let both art historians and computer scientists test HyperIndex was twofold. First it would test the effectiveness of HyperIndex in regard to naive (in the art history sense) searchers. Secondly, as the computer scientists have knowledge of mathematical structures, it would be tested if they use this knowledge and thus navigate more effectively over the HyperIndex structure than the art historians.

Experimental Procedure

The objective of the experiment was to test how effective searchers could satisfy predefined information needs using either ICONCLASS plus other available avenues, or HyperIndex.

Experimental Environment

The log file for each HyperIndex user consisted of instances of each of the following events:

- Starting a new session by selecting a keyword in the Finder.
- Each new selected focus.
- Each view of the current selection (**Show**).

For ICONCLASS users the recorded events were:

- Each class selected.
- Each view of the current selection.

Each instance in the log file is called a **Logical decision**.

Interpretation of Experimental Data

First, the sequence of actions by a searcher cannot be classified as correct or incorrect; in fact most of the searcher's actions are not known to the system.

Considerable time may elapse between two recorded actions: the searchers, may be reading information on the screen, may be confused, or may be consolidating their knowledge. This will influence their next action, but is not recorded in the log file. As a consequence it is not sure if the users know what they are doing, or at least think they do, or if they are just guessing their next action.

What can be measured easily is how the set of slidenumbers found by the searcher compares to the ideal set of slidenumbers for a particular question, recall and precision, as well as the number of logical decisions and shows.

Performance of Art Historians

Question 1: Find slides depicting the Annunciation.
This question was expected to be easy with both ICONCLASS and HyperIndex. For HyperIndex this turned out to be true. All searchers selected *Annunciation* and found all slides. Some refined their choice to see if more slides would turn up. This points to a lack of mathematical knowledge, as this obviously cannot lead to more hits. ICONCLASS worked not as well as thought. Annunciation is indeed a class, but a large number of users did not find the path to it.

Question 2: Find three slides depicting flowers.
This question was a little harder for both. All HyperIndex users selected *flowers*, but only one slide could be found this way. A path had to be found to *Sunflowers* to find the other slides. Not all users managed to do this. ICONCLASS proved to be not suited for this type of question at all. All slides in the relevant set were paintings with ICONCLASS *Still-live*, while the searchers selected classes within *Nature* etc.

Question 3: Find slides with French city-views by Van Gogh.
This question, again, was not an easy one. Most slides could be found by starting HyperIndex with *Paris*, and using the lattice structure to arrive at *Arles*. ICONCLASS users had to find ICONCLASS *City-view*. Both methods let to almost equal Recall results.

Question 4: Find a slide depicting an equestrian statue of Aurelius.
An easy one for HyperIndex. All subjects chose either *Equestrian statue*, *Aurelius* or, showing their knowledge of the matter, *Marcus* to find the requested slide. None of the ICONCLASS searchers succeeded. Due to the obscure classification, *Traffic on land*, they searched confused until forced to give up (The average searcher gave up after 15 Logical decisions).

Question 5: Find slides depicting harvest scenes.
Two slides of the relevant set were easily found with both methods. ICONCLASS has a class *Harvest*, which takes a while to find though, and with HyperIndex a searcher can select *Harvest*. The third slide gave trouble in both methods. A *Mower* in a cornfield has been given ICONCLASS *Landscapes*. None of the ICONCLASS users found this slide. HyperIndex users could find it by browsing through the index and selecting *Harvester*.

Question 6: Find slides depicting Greek Gods.
This question and the next proved to be real trouble makers for HyperIndex searchers, while they are both easy enough for ICONCLASS. ICONCLASS *Classical*

Mythology reveals almost all slides of the relevant set. HyperIndex can only be used to select one God after another, because no links consist between individual Gods.

Question 7: Find slides depicting the passion of Christ
Again, an easy problem to solve with ICONCLASS by selecting *Passion of Christ*. HyperIndex users had to select every single element of the passion, which can only be done if the searchers have knowledge to help them chose.

Question 8: Find slides depicting Mary.
A hard question for both methods. Who would find a slide depicting Mary on *flight into Egypt*, without being an expert on this matter. ICONCLASS users were able to find most requested slides, but a number of them did not chose *Madonna* as a second class after finding ICONCLASS *Mary* first.

Table 1 in Appendix A summarizes the effectiveness of art historians using ICONCLASS.

Performance of the Computer Scientists

Table 3 in Appendix A summarizes the effectiveness of computer scientists searching in the HyperIndex. It does not significantly differ from the art historians (see table 2 in Appendix A). The supposition that the computer scientists would make better use of the underlying structure is partially confirmed. In question 3, all computer scientists began to search with the term *Paris* because they lacked the knowledge to begin with city names such as *Arles*. From the focus *Paris* it was possible to navigate to *Arles* by refining *Paris* to *view of Paris*, enlarging this to *view* and refining again to *view of Arles*. This last focus could be enlarged to *Arles* which characterized quite a few of the relevant slides. The art historians tended to use their knowledge of painting and began searches with the names of places where Van Gogh painted whilst in France. They therefore navigated less.

Overall Performance

ICONCLASS has an average Recall of 0.64, for this 9.8 Logical decisions were needed. Precision is 0.88. Show is used 3.2 per question. Problems occurred with questions 2 and 8. These questions very much show the limitations of ICONCLASS. If it is not perfectly clear to which class a piece of art belongs or more classes are appropriate, and no alternative avenue is at hand, it is indeed very hard to find the wanted information. The users of ICONCLASS showed they had the following problems:

- For religious subjects it is not clear whether to look in main division 1: *Religion and Magic*, or in division 7: *The bible*.
- Users do not know what subclasses they can expect within a division or class.
- Users sometimes have no idea what ICONCLASS is appropriate for a subject. For example: An equestrian statue of Aurelius turns out to be of ICONCLASS *Traffic on land*.

- Painted objects are not classified as objects as such, but as the paintings ICONCLASS. For example: A Painting of Van Gogh with Sunflowers has ICONCLASS: *Still-live*.

HyperIndex has an average Recall of 0.68. To achieve this, 6.0 Logical decisions are made. Precision is 0.91 and Show is used 3.0 times. The users of HyperIndex showed they had the following problems:

- Some users do not see the fact that by only refining they will not find more answers, resulting in more Logical decisions and Shows.
- Users often do not use the refinement and enlargement possibilities to go from one topic to another. They see a certain term within their current focus, leave the index and start with this term from scratch.
- Users forget which points have already been visited.
- Users arrive at a certain point which does not give them the desired result, and they don't know where to go from there.

First consider the problem of *disorientation*. Since the HyperIndex system does not have a linear structure but lets users move up and down the lattice, it can be difficult to ascertain the layout of the structure. Users become disoriented in that they do not know where they are in the information space and do not know how to get at some place they believe exists.
It appears that some of the difficulties that searchers have in maintaining their orientation while using the HyperIndex system, could easily be overcome by implementing a *history* function. This addition allows the user at any given focus in the HyperIndex to see the used path to that focus. Users will thus be able to recall the items which detracted them from the main search path, they can decide to select any old focus from the history and make this the current focus.

From the results of the experiment it is perfectly clear that one type of question is particularly hard to answer with HyperIndex. Questions which can only be answered by browsing through the itemindex to find items which are applicable yield a low to very low Recall. This problem too can be easily overcome. The searcher should not have to browse through the entire itemindex, but have all applicable items grouped together to chose from. Grouping can be done by adding a cross-reference to the HyperIndex. The searchers can now use associative navigation to provide answers to their information need.

The Second Experiment

How would the changes in the system effect the results of searchers using these changes to find answers to the questions of the first experiment? A second test was performed using the same slidedatabase and experimental procedure as before. Two Computer scientists were selected to participate. The searchers used the same HyperIndex system as in the first experiment. Added were the history function and the cross-references needed to answer the questions.

Interpretation of New Experimental Data

Since this experiment was done only to test the additions to the system, only three questions were of particular interest; questions 6, 7 and 8, which in the first experiment were hard to answer using HyperIndex. The other questions were needed to compare results to the first experiment. The results for these questions should not differ from the first experiment.

As predicted, the searchers had much better results for questions 6, 7 and 8. (see table 4 in Appendix A). Selecting all cross-references, however, drove up the number of Logical decisions and Shows. The history function was seldom used. This is probably due to the relative short search paths which had to be used and therefore the ease with which searchers can find each focus again by selecting the same term.

4.2 Later Additions

To provide searchers with direct access to the visual reproduction of the art-objects in their selection an imagemodule was added to Odilon. A thumbnail image is now displayed with the object data and the searcher can also switch to a fullscreen image.

The first version of Odilon used a staightforward implementation of the HyperIndex. A disadvantage was that the HyperIndex may grow alarmingly, because of the possibly huge number of subexpressions. A recent Master's thesis project resulted in an implementation for dynamic generation of the HyperIndex. This resulted in a marginal overhead during query by navigation, while saving almost 75% diskspace [H97].

The high number of Logical decisions and Shows for HyperIndex, resulting from the addition of cross-references can be brought down by splitting them into two relationship types: A term cross-reference relation, which still allows *Mary* and *Madonna* to be related. The second is an ISA relation. For example, *Hercules* ISA *God*. This allows the HyperIndex searcher to refine the focus *God* to *Hercules*, but focus *God* will also show all *Hercules* related slides in the selection.

5 Conclusions

The fact that computer scientists results did not differ significantly from those of Art historians demonstrates one of the advantages of HyperIndex over ICONCLASS, where domain knowledge is needed in order to use it as an effective disclosure system. The experiments showed that ICONCLASS is especially suited searching classes of objects, while HyperIndex performs very well on searches for objects which can be in different classes. To state this differently, ICONCLASS is a semantic, HyperIndex a syntactic disclosure system. Some disadvantages of HyperIndex stem from the fact that it is derived from an automatic syntactic process. The index expression *Announcement of birth of Christ* will never be related to the expression *Mary*, even though the two are highly related. This is because the index expression transducer is a syntactic analyzer which cannot make the connection to *Mary*. The documentalist, however, would most certainly make this connection. This is the strength of semantic classification, although it is a manual process. HyperIndex doesn't have to stay far behind though in answering semantic queries. Some problems can be solved by manually adding new links. Only once has the relation between *Announcement of birth of Christ* and *Mary* to be added to link all current and future occurrences of both expressions. Adding too many links can however turn against the searchers in the long run; Just as large software programs with many jumps can turn into "spaghetti" code, so a HyperIndex system can turn into a swamp of meaningless, obscure connections and references. The HyperIndex system, therefore, must allow editing and deleting manually added links easily.

References

B98. F. Berger. Navigational Query Construction in a Hypertext Environment. Phd Thesis, Department of Computer Science, University of Nijmegen 1998.

BB91. R. Bosman and R. Bouwman. The Automation and Disclosure of a Slides Library. Master's thesis, University of Nijmegen, Nijmegen, The Netherlands, April 1991.

BBB91. R. Bosman, R. Bouwman, and P.D. Bruza. The Effectiveness of Navigable Information Disclosure Systems. In G.A.M. Kempen, editor, *Proceedings of the Informatiewetenschap 1991 conference*, Nijmegen, The Netherlands, 1991.

Bru90. P.D. Bruza. Hyperindices: A Novel Aid for Searching in Hypermedia. In A. Rizk, N. Streitz, and J. Andre, editors, *Proceedings of the European Conference on Hypertext - ECHT 90*, pages 109–122, Cambridge, United Kingdom, 1990. Cambridge University Press.

BW90a. P.D. Bruza and Th.P. van der Weide. Assessing the Quality of Hypertext Views. *ACM SIGIR FORUM (Refereed Section)*, 24(3):6–25, 1990.

BW90b. P.D. Bruza and Th.P. van der Weide. Two Level Hypermedia - An Improved Architecture for Hypertext. In A.M. Tjoa and R. Wagner, editors, *Proceedings of the Data Base and Expert System Applications Conference (DEXA 90)*, pages 76–83, Vienna, Austria, 1990. Springer-Verlag.

BW91. P.D. Bruza and Th.P. van der Weide. The Modelling and Retrieval of Documents using Index Expressions. *ACM SIGIR FORUM (Refereed Section)*, 25(2), 1991.

Con87. J. Conklin. Hypertext: An Introduction and Survey. *IEEE Computer*, 20(9):17–41, September 1987.

H97. E. Hebing Implementation of Hyperindices. Master's thesis, University of Nijmegen, Nijmegen, The Netherlands, June 1997 (in Dutch).

SK89. B. Schneiderman and G. Kearsley. *Hypertext Hands-On!* Addison-Wesley, Reading, Massachusetts, 1989.

Waa85. H. van der Waal. *ICONCLASS an iconographic Classification System.* North-Holland, 1985. Completed and edited by L.D.Couprie, E.Tolen and G.Vellenkoop.

A Experiment Results

Question	*Recall*	*Precision*	*Logical decisions*	*Show*
1	0.56	1.00	11.6	2.4
2	0.30	0.70	12.4	2.8
3	0.69	0.40	10.4	4.3
4	1.00	1.00	∞	∞
5	0.55	1.00	11.0	2.8
6	0.77	0.98	4.2	4.0
7	1.00	1.00	5.4	1.4
8	0.26	0.98	8.4	2.8
Average	0.64	0.88	9.8	3.2

Table 1. Experiment results of art historians using ICONCLASS

Question	*Recall*	*Precision*	*Logical decisions*	*Show*
1	1.00	1.00	1.8	1.6
2	0.65	1.00	13.0	2.8
3	0.71	0.61	6.2	3.1
4	1.00	1.00	2.6	1.4
5	0.80	1.00	3.4	2.2
6	0.43	0.92	7.6	5.2
7	0.30	0.94	9.0	5.4
8	0.54	0.84	4.4	2.4
Average	0.68	0.91	6.0	3.0

Table 2. Experiment results of art historians using HyperIndex

Question	*Recall*	*Precision*	*Logical decisions*	*Show*
1	1.00	1.00	1.3	1.3
2	0.75	1.00	11.2	2.5
3	0.82	0.56	5.1	4.2
4	1.00	1.00	3.0	1.3
5	0.88	1.00	3.0	3.5
6	0.44	0.69	7.5	4.8
7	0.16	1.00	8.0	4.5
8	0.35	0.93	3.5	2.4
Average	0.68	0.90	5.3	3.1

Table 3. Experiment results of computer scientists using HyperIndex

Question	*Recall*	*Precision*	*Logical decisions*	*Show*
1	1.00	1.00	1.5	1.0
2	0.75	1.00	8.5	3.0
3	0.73	0.58	6.3	3.6
4	1.00	1.00	2.5	1.5
5	0.88	1.00	2.5	2.0
6	0.92	0.93	24.0	20.0
7	0.57	1.00	18.5	14.0
8	0.61	0.97	2.0	2.0
Average	0.81	0.94	8.2	5.8

Table 4. Second experiment results of computer scientists using HyperIndex