InfoCrystal:

A visual tool for information retrieval

Anselm Spoerri

Center for Educational Computing Initiatives
Massachusetts Institute of Technology
Building E40-370, 1 Amherst Street, Cambridge, MA 02139
aspoerri@athena.mit.edu

Abstract

This paper introduces a novel representation, called the ***InfoCrystal™****, that can be used as a* ***visualization tool*** *as well as a* ***visual query language*** *to help users search for information. The InfoCrystal visualizes all the possible relationships among N concepts. Users can assign relevance weights to the concepts and use thresholding to select relationships of interest. The InfoCrystal allows users to specify* ***Boolean*** *as well as* ***vector-space*** *queries graphically. Arbitrarily complex queries can be created by using the InfoCrystals as building blocks and organizing them in a hierarchical structure. The InfoCrystal enables users to explore and filter information in a flexible, dynamic and interactive way.*

Keywords: Information visualization, visual query language, information retrieval, graphical user interface, human factors.

1.0 Introduction

Information is becoming available in ever growing quantities as the access possibilities to it proliferate. However, better methods are needed to filter the potentially unlimited influx of information. Researchers at Xerox PARC, for example, believe that managing large quantities of information will be the key to effective computer use in the 1990s and that visual interfaces that recode the information in progressively more abstract and simpler representations will play a central role [Card 91].

Recent work in scientific visualization shows how large data sets can be visualized in such a way that humans can detect patterns that reveal the underlying structure in the data more readily than a direct analysis of the numbers would. Similarly, information visualization seeks to display structural relationships between documents and their context that would be more difficult to detect by individual retrieval requests [Card 91].

Most of the visualization problems that are currently being investigated involve continuous, multi-variate fields that vary over space and time. Hence, the transformation problem is simplified, because the data has an explicit spatial structure that can be exploited. This paper, however, will address the problem of how to visualize abstract information, such as a document space, that does not have explicit spatial properties.

1.1 Problem Statement

This paper addresses the problem of how to enhance the ability of users to access information by developing better ways for both visualizing abstract information and formulating queries graphically. As the amount of available information keeps growing at an ever increasing rate, it will become critical to provide users with *high-level visual retrieval tools* that enable them to explore, manipulate, and relate large information spaces to their interests in an interactive way. We use the term "high-level" because these tools are designed to give users flexibility with both how to retrieve and how to explore information. These tools provide users with a visual framework that enables them to integrate and manipulate information that has been retrieved by different methods or from different sources.

The InfoCrystal is an example of such a high-level retrieval tool and it has the following functionality: 1) Users can *explore* an information space along several dimensions simultaneously without having to abandon their sense of overview. 2) Users can *manipulate* the information by *creating useful abstractions*. 3) Similar to a spreadsheet,

1070-2385/93 $3.00 © 1993 IEEE

users can ask *"what-if"* questions and observe the effects without having to change the framework of a query. 4) Users receive *support* in the search process because they receive *dynamic visual feedback* on how to proceed. They can selectively emphasize the *qualitative* or the *quantitative* information provided by the feedback to help them decide how to proceed. 5) Users can formulate queries *graphically*, and they have *flexibility* in terms of the particular methods used to retrieve the information. For example, users can seamlessly move between a *Boolean* and a *vector-space* retrieval approach, or they can easily switch from a keyword-based to a full-text retrieval approach.

This paper is organized as follows: 1) We will consider a concrete retrieval example to set the stage. 2) We will introduce the *InfoCrystal*. 3) We will review and compare relevant previous work with the developed tool. 4) We will provide a brief summary and talk about the research currently underway.

1.2 Concrete Example

It is best to consider a concrete example to describe some the problems a user currently faces when searching for information. For example, if we are interested in documents that talk about "visual query languages for retrieving information and that consider human factors issues" then the following concepts could capture our interest: *(Graphical OR Visual)*, *Information Retrieval*, *Query language*, *Human Factors*. Most of the existing on-line retrieval systems use Boolean operators to combine the identified concepts to form a query. On the one hand, the most exclusive query would join the concepts by using the AND operator. We performed such a query, using a CD-ROM version of the INSPEC Database for the years 1991-92. Only one document was retrieved that contained all the four concepts. On the other hand, the most inclusive query would join the concepts by using the OR operator; it retrieved 19,691 documents. Hence, we are presented either with too few documents or too many documents. How should we proceed and modify the exclusive query or narrow the inclusive query to retrieve more relevant documents? We will revisit this example after we have introduced the InfoCrystal and we will show how it could help users to modify the query successfully.

2.0 The InfoCrystal

In this section we describe how we propose to help users search more effectively for information. We will first address the question of how to visualize all the possible relationships among N concepts. Towards that end we will develop the *InfoCrystal*, whose elements can be selectively visualized to emphasize the *qualitative* or the *quantitative* information associated with them. Second, we will demonstrate how the InfoCrystal can be used to formulate *Boolean queries* graphically. We will also show how the InfoCrystals can be used as building blocks and integrated in a hierarchical structure to formulate arbitrarily complex queries. Third, we will show how users can assign *relevance weights* to the concepts and use *thresholding* to select relationships of interest. We will describe the *rank layout* and the *bull's-eye layout* principle that visualize an InfoCrystal so that the relationship with the highest rank or the one with the largest relevance score, respectively, will lie in its center. Fourth, we will show how the InfoCrystal can be generalized so that *vector-space* queries can be specified graphically. Finally, it is worth mentioning that the InfoCrystal can be integrated with a query outlining and a navigation tool, as described in [Spoerri 93a], to enable users to create and maintain complex search queries.

2.1 Visualizing Relationships

How can all the possible combinations or relationships among several search criteria be visualized in a two-dimensional display ?

A common approach is to use Venn diagrams to visualize set relationships by intersecting geometric shapes that represent each set. However, it is difficult to represent all the possible relationships among more than three sets in a visually compact and simple way. We will now demonstrate how we can move beyond the Venn diagram approach so that all the possible relationships among N variables can be represented at same time in an elegant way. Figure 1 shows how a Venn diagram of three intersecting circles can be transformed into an iconic display. We start out by exploding the Venn diagram into its disjoint subsets. Next, we represent the subsets by icons whose shapes reflect the number of criteria satisfied by their contents, also called the *rank* of a subset. Finally, we surround the subset icons by a border area that contains icons, also called *criterion icons*, that represent the original sets.

The goal is to arrive at a visual representation that lets users use their visual reasoning skills to establish how the interior icons are related to the criterion icons, and the following visual coding principles are used in a redundant way:

- **Shape Coding**: is used to indicate the number of criteria that the contents associated with an interior icon satisfy (i.e., one -> circle, two -> rectangle, three -> triangle, four -> square, and so on).
- **Proximity Coding**: The closer an interior icon is located to a criterion icon, the more likely it is that the icon's contents are related to it.
- **Rank Coding**: Icons with the same shape are grouped in "invisible" concentric circles, where the rank of an icon is equal to the number of criteria satisfied and it increases as we move towards the center of an InfoCrystal.
- **Color or Texture Coding**: is used to indicate which particular criteria are satisfied by the icon's contents.
- **Orientation Coding**: The icons are positioned so that their sides face the criteria they satisfy.
- **Size or Brightness & Saturation Coding**: is used to visualize quantitative information, i.e. the number of elements represented by an icon.

Figure 1 shows the InfoCrystal that involves three concepts. Figure 2 shows an InfoCrystal for four search criteria. Figure 3 contains a schematic representation of an InfoCrystal for five criteria (for a detailed rendering, see [Spoerri 93a]). The number of possible combinations or relationships among N different criteria grows exponentially and it is equal to $2^N - 1$ (excluding the case where documents are not related to any of the criteria). We have developed a layout procedure that enables us to generate InfoCrystals with N inputs (for a detailed discussion and examples with N > 5, see [Spoerri, 1993a]). The objective of this algorithm is to create a layout of the interior icons that ensures that none of their locations coincide. We call it the *rank layout* principle, because it strictly enforces the rank coding principle. However, it also attempts to resolve the conflict between the rank and the proximity coding principle for the icons with rank two as follows: We will represent icons that involve relationships between two non-adjacent criterion icons twice and we will place them in such a way that they are close to their related criterion icons as well as at the correct distance from the center (see Figure 2).

The user can selectively render the interior icons to emphasize the *qualitative* or the *quantitative* information associated with them: If the user is interested in how the interior icons are related to the inputs then they are displayed as shown in Figure 1. If, however, the user wants to visualize the number of documents associated with the interior icons then the icons are represented as circular pie chart icons whose size and brightness reflect the numerical information (see Figure 3). The pie chart icons are similarly oriented as the polygon icons and the colors or textures of their slices indicate which criteria are satisfied.

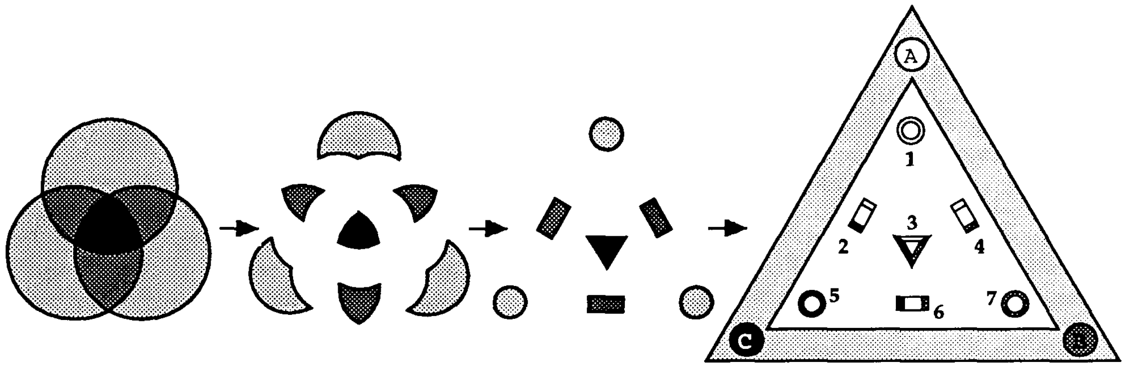

Figure 1: shows how to transform a Venn diagram into an iconic display, called the *InfoCrystal*, which represents all the possible Boolean queries involving its inputs in a normal form (see section 2.2). The interior icons have the following Boolean meanings: 1 = (A and (not (B or C), 2 = (A and C and (not B), 3 = (A and B and C), 4 = (A and B and (not C), 5 = (C and (not (A or B)), 6 = (B and C and (not A), 7 = (B and (not (A or C)).

2.1.1 Example Revisited

In this subsection we show how the InfoCrystal could help users modify the query example introduced earlier so that they do not retrieve either too few or too many documents. Figure 2 displays how the contents of the INSPEC Database (1991-92) relate to our four stated interests.

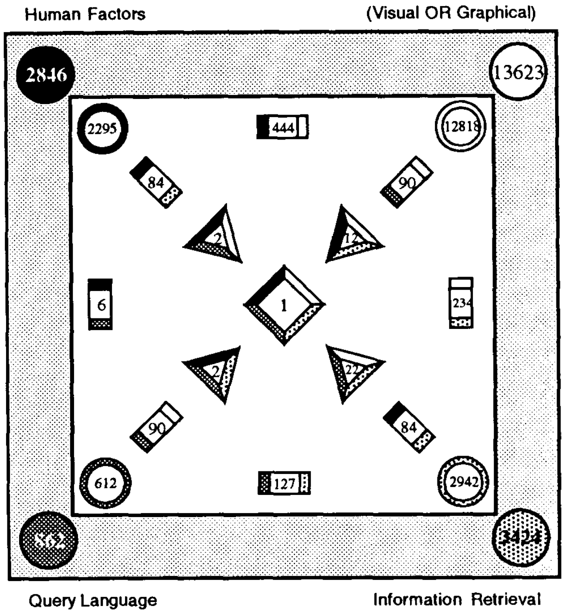

Figure 2: The number associated with an icon indicates how many of the retrieved documents satisfy the conditions represented by it.

The center icon of the InfoCrystal represents the documents that satisfy all the four criteria. In our example there is just one document. We can easily broaden our focus of interest by examining the icons that surround the center icon and satisfy three of the four concepts. For example, there are 22 documents that are indexed under and are related to the *(Graphical OR Visual)*, *Information Retrieval*, and *Query Language* concept but not to the *Human Factors* concept. If we wanted to move further away from our initial interest then we could explore the 6 documents that have been indexed under the *Query Language* and *Human Factors* concept but not under the *(Graphical OR Visual)* or *Information Retrieval* concept. Hence, the InfoCrystal enables us to explore an information space in a flexible and fluid way. The organization of the InfoCrystal ensures that we can easily infer how the retrieved documents relate to our stated interests.

2.2. Visual Query Language

The InfoCrystal has the desirable property that each of its interior icons represents a distinct Boolean relationship among the input criteria (see Figure 1). Hence, the InfoCrystal can be used by users to specify Boolean queries by interacting with a direct manipulation visual interface. Users do not have to use logical operators and parentheses explicitly. Instead they need to recognize the relationships of interest and select them. Users just have to click on an interior icon to select or deselect the Boolean expression associated with it.

In an InfoCrystal we partition the space defined by its *N* inputs into $2^N - 1$ disjoint subsets or *constituents* in such a way that no information is lost. It can be easily shown that any Boolean query that involves the inputs of an InfoCrystal and that applies the Boolean operations of union, intersection or negation can be represented by the union of a certain number of the constituents (i.e., all the possible queries are represented in *normal form* by the InfoCrystal). Hence, users can specify graphically any Boolean query that involves the inputs by selecting the appropriate interior icons, because each of the constituents is represented by an interior icon. We establish the intuitive convention that the elements associated with the selected interior icons are combined to form the output of an InfoCrystal. We do not have to worry that certain elements appear more than once because we are merging disjoint subsets.

It is worth stressing that users can select a subset of interior icons in multiple ways: 1) They can select specific relationships by clicking on the appropriate interior icons. 2) Users can select subsets of interior icons by clicking on the criterion icons, thereby performing complex Boolean operations with only a few mouse clicks. 3) They can activate the appropriate interior icons by interacting with a threshold slider and/or the weighting sliders for the inputs (see Section 2.3 for explanation).

Existing visual query languages allow users to formulate specific queries, but the proposed visual query language enables users to formulate a whole range of related queries by creating a single InfoCrystal. For N inputs there are 2 to the power of 2^N - 1 possible queries and each of them can be specified by just selecting the appropriate interior icons. Hence, in the case of five inputs there are over 2 billion possible queries and they are all represented compactly by an InfoCrystal !

2.2.1 Creating Complex Queries

The InfoCrystals can be used as building blocks and organized in a hierarchical structure to create complex Boolean queries. First, an InfoCrystal can be thought of as having several inputs, represented by the criterion icons, and as having an output that is defined by the selected interior icons. Second, the output of one InfoCrystal will be one of the inputs to an InfoCrystal one level up in the query hierarchy.

The hierarchical query structure differs from a simple tree structure as follows: The parent nodes do not just inherit the data elements associated with their children's nodes. Instead there is an intermediary step where the relationships among the children's nodes are represented by an InfoCrystal. Users have to select the relationships that should be included in the InfoCrystal's output that is passed on to the parent.

Figure 3 shows how the InfoCrystals can be "chained together" to form a hierarchical query structure. Similar to a spreadsheet, users can ask "what-if" questions by changing which interior icons are selected in one InfoCrystal and observe how the contents of the dependent icons higher up in the hierarchy change dynamically. Further, users can build a library of queries in an incremental fashion, where they can create complex queries by integrating simpler ones.

2.2.2 Interfacing with the Retrieval Engines

The atom or "leaf" nodes of the query structure represent the criteria that the user has decided not to break down any further (see Figure 3, where the atoms are represented by circular InfoCrystals). The atoms specify the query statement that a retrieval engine will use to search for information in the selected database(s). Users can also specify at the atom level whether to search the author, title, keywords or abstract field, or whether to use proximity or stemming as a search strategy.

A key feature of the InfoCrystal is that it works with any data type, provided its corresponding retrieval method returns unique data identifiers, which are then used to initialize the query structure. Hence, at any point in the search process users could switch from a keyword-based to a full-text retrieval approach by replacing an input criterion with a particular document that better captures a specific interest.

2.3 Relevance Weights & Thresholds

There will be situations where the search criteria are not of equal importance to a user. Further, users can find it initially easier to retrieve information by using a vector-space approach, where they can assign relevance weights to their search interests [Belkin 92]. We will now show how the InfoCrystal can be generalized so that users can seamlessly move between a Boolean and a vector-space retrieval approach. As a first step, we show how users can assign relevance weights to the inputs of an InfoCrystal to reflect the degree of importance they assign to them (see Figure 3). By interacting with a slider, they can choose values between -1 and 1, where negative weights indicate that users are more interested in documents that do *not* contain the concept represented by the input (i.e., the weight -1 is equivalent to the logical NOT). The assigned weights can be used to compute a relevance score for each interior icon by taking the dot product between the vector of the input weights and a vector, whose values are equal to 1 or -1 depending on whether the corresponding criteria are satisfied or not by the icon. By interacting with the threshold slider, users can select only the interior icons whose relevance score is above the threshold.

2.3.1 Bull's-Eye Layout

The key design principle used in the layout of the interior icons of an InfoCrystal is to ensure that users will find towards its center the relationships that they consider as more important. So far we have considered the *rank layout* that enforces that the number of criteria satisfied by an icon increases as we move towards the center of an InfoCrystal. We will now describe how users can display the interior icons to reflect the current setting of the relevance weights. This mapping, called the *bull's-eye layout,* causes the relationships with a higher relevance score to be placed closer to the center (see Figure 3.b). We use a novel polar representation to determine the placement of the interior icons. The radius value is determined by the relevance score. The angle, however, is not affected by the weights. It is a function of the line that passes through the InfoCrystal's center and the center of mass of the criterion icons that is computed as follows: a vector pointing towards a criterion icon that is satisfied by an interior icon receives a positive mass of 1, whereas the vector pointing towards a criterion that is not satisfied receives a negative mass of -1. Thus, an interior icon is closer to those criterion icons that it satisfies than to those it does not.

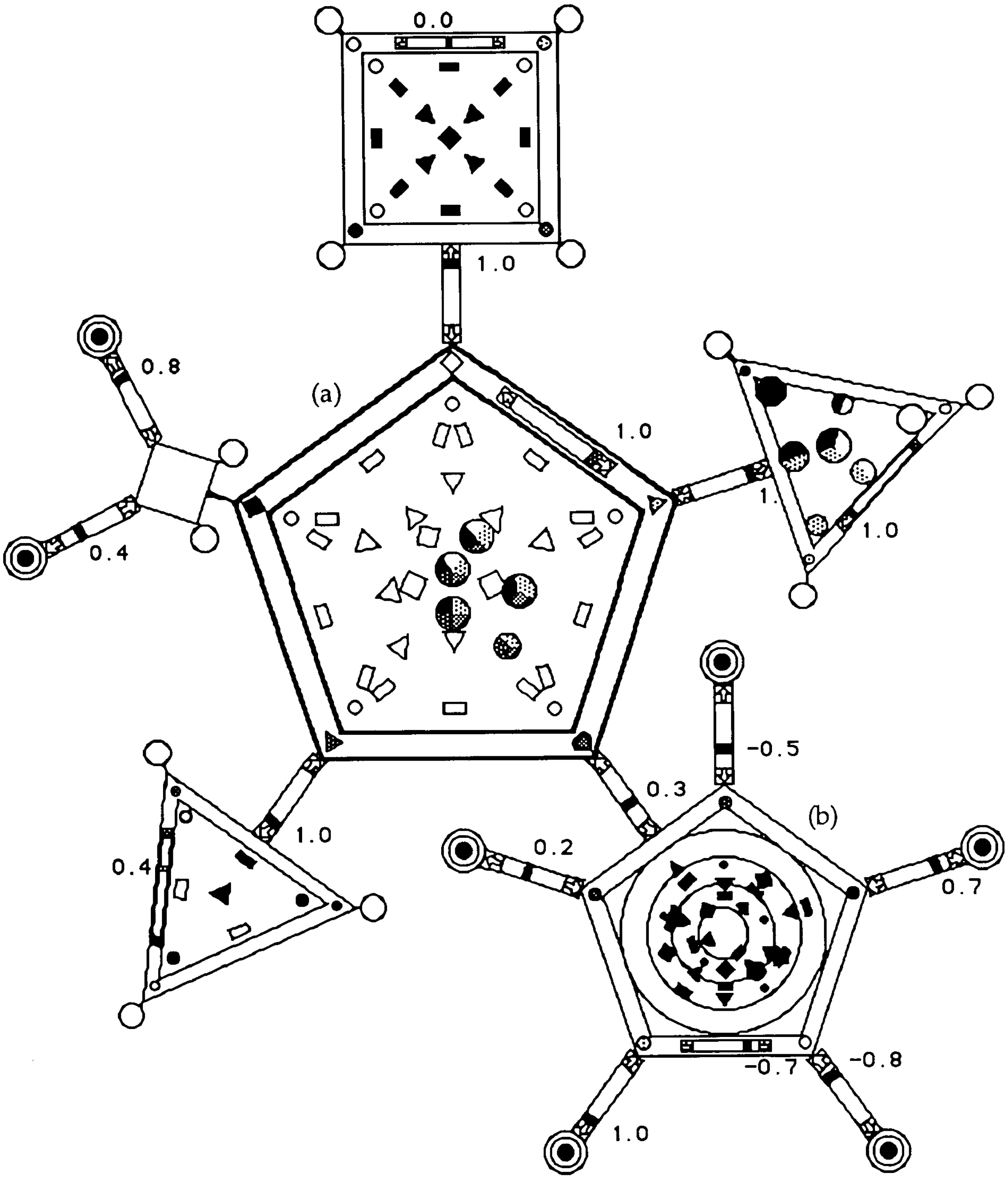

Figure 3: shows how the InfoCrystals can be organized in a hierarchical structure. Users can interactively change the way the InfoCrystals filter their inputs and they can dynamically observe how the information coming in through the circular InfoCrystals is propagated through the query structure. The interior icons that are shown in solid black indicate that they have been selected by the user to define the output of an InfoCrystal. Some of the InfoCrystals are displayed only as an outline, but the user can just click on them to view them in full detail. (a) Shows the top-level InfoCrystal, using the *rank layout* principle, where the selected icons are rendered, using a pie chart representation, to emphasize the quantitative information associated with them. (b) Shows the *bull's-eye layout* of the interior icons when the input weights are set to -0.5, 0.7, 0.8, 1.0 and 0.2; and it shows which icons will be selected if the threshold is set to -0.7 (in this case all of them).

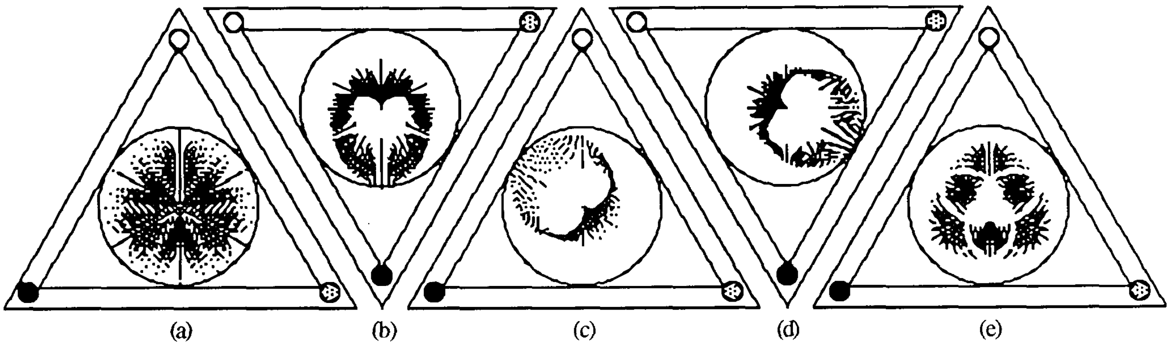

Figure 4: shows the distribution pattern of the relevance scores of all the document vectors that lie in the cube {[-1, 1]; [-1, 1], [-1, 1]}, using the bull's-eye layout principle that takes into account the particular values of the input weights. (a) - (d) Show the distribution patterns for input weights (1,1,1), (1,1,-1), (-0.75,1,-0.25), and (0.85,-0.6,-0.4), respectively. (e) Shows the clustering of the relevance scores of the documents that satisfy one or two of the input criteria, respectively, where the input weights are (1,1,1).

2.4 Visualizing Vector-Space Queries

So far we have considered the discrete case where a document either satisfies a criterion or not at all. We can generalize the InfoCrystal so that we can consider the continuous case where documents have a value between -1 and 1 to reflect the degree to which a criterion is satisfied or not. This allows us to specify vector-space queries graphically, since the vector-space approach computes the relevance of a document by taking the dot-product of the vectors of the index terms that represent the query and the document respectively [Belkin 92].

We can apply the bull's-eye layout principle to visualize how the retrieved documents satisfy the input criteria to varying degrees, where the vector pointing towards a criterion icon is now scaled by the degree to which the criterion is (not) satisfied by a document (see Figure 4). When mapping an N-dimensional space into a two-dimensional space, we are faced with the challenge of how to compress the information in such a way that it still captures what we are interested in. The polar transform used to map the documents has the attractive feature that it not only visualizes the ranking, but it also provides users with a qualitative sense of how the ranked documents are related to the input criteria (for a more detailed discussion and further examples, see [Spoerri 93b]. In Figure 4.b the input weights are equal to (1,1, -1), and as expected the documents with the lowest score are displayed close to the third & black criterion icon. Figure 4.e shows that the documents, which satisfy the same criteria and are therefore represented by the same interior icon in the discrete mode, will cluster in an orderly fashion when shown in the continuous mode.

3.0 Comparison with Relevant Previous Work

In this section we will briefly review and compare the developed tool with relevant previous work (for a more detailed discussion, see [Spoerri 93a]). We will focus on the shortcomings of existing proposals and indicate how the InfoCrystal addresses them. We hope this type of exposition will better motivate the approach taken in this paper.

Current on-line retrieval systems require users to use Boolean operators to formulate queries, but such queries can be difficult to generate [Belkin 92]. Further, either too few or too many documents are retrieved, and a pure Boolean approach does not rank the retrieved documents [Salton 88].

The InfoCrystal enables users to specify Boolean queries graphically. Users select relationships of interest instead of having to use logical operators and parentheses. The InfoCrystal displays the retrieved documents in a ranked order that enables users to control the output.

Best-Match models use statistical techniques and vector-space models to compute the similarity between documents and they rank the retrieved documents based on some relevance measure [Belkin 92]. However, a ranked linear list provides users with a limited view of the information space and it does not directly suggest how a query could be modified.

The InfoCrystal does not lock users into just one way of viewing the data. It helps users decide how to proceed in the search process because the quantitative information associated with an interior icon tells them how much additional items they can expect if they select it. Further, the bull's-eye layout shows users in a qualitative way how the ranked items are related to the input criteria.

Several researchers have developed overview maps that attempt to visualize the similarity between documents [Lin 91, Chalmers 92, Korfhage 91]. However, these maps can not visualize multiple relationships between documents and they become hard to interpret as the size of the document space increases. Further, these maps can not be used to formulate queries graphically.

The InfoCrystal is designed to visualize the similarity and the possible multiple relationships between the contents of an information space and N search criteria. It is both a visualization tool and a visual query language, and it scales well, because its organization is size independent.

Existing visual query languages suffer generally from the limitation that the organization of a query needs to be modified to generate a different query [Michard 82, Anick 90, Young 92].

The InfoCrystal represents all the possible Boolean queries involving its inputs in normal form. In the case of vector-space queries, users can interactively assign *relevance weights* to the concepts and use *thresholding* to select documents.

4.0 Conclusion & Future Work

This paper has presented a novel representation, called the *InfoCrystal*™, that can be used both as a *visualization tool* and a *visual query language*. The InfoCrystal can be used to visualize all the possible discrete as well as continuous relationships among N concepts. In the discrete case, the InfoCrystal uses proximity, rank, shape, color and size coding to enable users to see in a single view how a large information space relates to several of their interests. In the continuous case, a novel polar representation has been presented that visualizes the relevance scores of the retrieved documents and provides users with a qualitative sense of how the ranked documents are related to the input criteria. Further, the InfoCrystal allows users to specify *Boolean* as well as *vector-space* queries graphically. Complex queries can be created by using the InfoCrystals as building blocks and organizing them in a hierarchical structure.

The InfoCrystal is an example of a high-level visual retrieval tool that is designed to give users flexibility with both how to retrieve and how to explore information. It provides users with a visual framework that enables them to integrate and manipulate information that has been retrieved by different methods or from different sources in a flexible, dynamic and interactive way.

We are currently completing the implementation of the InfoCrystal on the Macintosh. We are also in the process of integrating some of its functionality with the CONIT expert retrieval assistant system [Marcus 91]. Further, we will conduct user studies to test the effectiveness of the InfoCrystal.

Acknowledgments: Support for the research described in this paper has been generously provided by the Union Bank of Switzerland (UBS). We would like to thank especially: Dr. Marty and Dr. Frei at UBS, the members of CECI at MIT; Prof. Lerman, Prof. Resnick and Mr. Marcus; the MIT Patent & Licensing Office; and Pamela Robertson-Pearce.

5.0 References

Anick, P.; Brennan, J.; Flynn, R.; Hanssen, D.; Alvey, B. & Robbins, J. (1990) "A Direct Manipulation Interface for Boolean Information Retrieval via Natural Language Query," Proc. ACM SIGIR '90.

Belkin, N. & Croft, B. (1992) "Information Filtering and Information Retrieval: Two Sides of the Same Coin" Comm. of the ACM, Dec., 1992.

Chalmers, M. & Chitson, P. (1992) "BEAD: Exploration in Information Visualization," Proc. ACM SIGIR '92.

Card, S.; Robertson, G. & Mackinlay, J. (1991) "The Information Visualizer, an information workspace," Proc. CHI'91 Human Factors in Comp. Systems, 1991.

Korfhage, R. & Olson, K. (1991) "Information display: Control of visual representations," Proc. IEEE Workshop on Visual Languages, Oct., 1991.

Lin, X.; Soergel, D. & Marchionini, G. (1991) "A Self-organizing Semantic Map for Information Retrieval," Proc. ACM SIGIR '91.

Marcus, R. (1991) "Computer and Human Understanding in Intelligent Retrieval Assistance," American Society for Information Science, 28, 1991.

Michard, A. (1982) "Graphical presentation of Boolean expressions in a database query language: design notes and an ergonomic evaluation," Behaviour and Information Technology, 1:3, 1982.

Salton, G. (1988) "A simple blueprint for automatic boolean query processing," Information Processing & Management, 24:3, 1988.

Spoerri, A. (1993a) "Visual Tools for Information Retrieval," Proc. IEEE Workshop on Visual Languages, 1993, and MIT-CECI-TR 93-2.

Spoerri, A. (1993b) "InfoCrystal: a visual tool for information retrieval," MIT-CECI -TR 93-3.

Young, D. & Shneiderman, B. (1992) "A Graphical Filter/Flow Representation of Boolean Queries: A Prototype Implementation and Evaluation, " Uni. of Maryland Report.