On the Construction of Selection Systems

Michael K. Buckland
Christian Plaunt

School of Library and Information Studies
University of California
Berkeley, CA 94720-4600
LP01

In: Library Hi Tech, 12:4:15–28, 1994

Abstract

An examination of the structure and components of information storage and retrieval systems and information filtering systems. Analysis of the tasks performed in such selection systems leads to the identification of thirteen components. Of these components, eight are necessarily present in all such systems, mechanized or not; the others may, but need not be, present. We argue that all selection systems can be represented in terms of combinations of these components. The components are of only two types: representations of data objects and functions that operate on them. Further, the functional components, or *rules*, reduce to two basic types: (i) *Transformation*, making or modifying the members of a set of representations and (ii) Sorting or *partitioning*. The representational *transformations* may be in the form of copies, excerpts, descriptions, abstractions, or mere identifying references. By *partitioning*, we mean dividing a set of objects by using matching, sorting, ranking, selecting, and other logically equivalent operations. The typical multiplicity of knowledge sources and of system vocabularies are noted. Some of the implications for the study, use and design of information storage and retrieval systems are discussed.

1 Introduction

Information storage and retrieval systems (catalogs, commercial online services, bibliographic utilities, CD-ROM products, etc.) have developed rather empirically as demand and enabling technology have evolved. Filtering systems (selective dissemination of information (SDI) systems, e-mail filters, etc.) have also developed pragmatically.

Hitherto, these two types of selection systems were usually developed and built in relative isolation: a local retrieval or filtering system designed to work with a specific dataset in a particular domain for specific users and interests, and employing or assuming various kinds of external knowledge (thesauri, etc.). Perhaps the practical consideration that such systems have to be complete in order to work at all has encouraged an emphasis on the construction and comparison of *complete* systems rather than on the individual *components* (or subsystems) of such systems.

1.1 Purpose

Three considerations now encourage detailed analysis of the *components* of selection systems:

1. Academic curiosity: Can all information storage and retrieval systems (or, better, all selection systems) be viewed as composed of a common set of components? If so, what are they and how many are there? Which are necessary and which are sufficient?

2. The recent Text REtrieval Conferences (TREC) [Harman 93] have provided a welcome revival of interest in the comparative evaluation of retrieval and filtering systems. We suggest, however, that there are significant limits to the benefits that can be derived from comparing whole, *complete* systems. Sooner or later, the advanced design and evaluation of selection system performance also requires the systematic comparative evaluation of alternatives at the level of individual components within complete systems.

3. In the emerging network environment selection systems have moved away from the traditional "unitary" model of one retrieval (or filtering) engine operating on one dataset. We now have a situation which we have called "extended retrieval" [Buckland n.d.]. It is easy to think of multiple retrieval engines connected to each other and to multiple databases over networks. But so simple a view begins to break down as soon as one begins to examine how extended retrieval might work: Where are the indexes, for example? Are they part of the their respective databases on the server or part of the client retrieval engine? In the NISO Z39.50 Search and Retrieval protocol (cf. ISO 10162 & 10163) an EXPLAIN function is being developed to enable the client to ascertain the available options and constraints of the server. What, in principle, *could* the server explain about itself that might be useful to the client?

In brief, a general conceptual framework and vocabulary for the *components* of selection systems is needed. This paper seeks to analyze the "anatomy" of selection systems. Such analysis should advance the theory of selection systems: What are the components of retrieval and filtering systems? Which are the necessary and sufficient components and which are optional? What different types of components are there? Which functionally similar techniques might be substitutable within any of the components? Which might be substitutable across different, but similar components? In what different ways can the components be combined to design more sophisticated systems? Our hope is that a functional analysis of components will stimulate the design of improved selection systems.

We will first propose a basic functional model of information storage and retrieval systems and discuss these components in some detail. Next, we reduce the non-data components of the system model to two functional types, *transformers* and *partitioners*. This is followed by a generalization of the model to other similar selection tasks. Finally, we comment on some of the implications for the study, use and design of selection systems. We approach this in the context of bibliographic and text systems, but believe the approach to be of general applicability.

1.2 Terminology

Throughout this paper we will be using several words and phrases in specific technical ways.

System boundaries define what is considered the "system" rather than the "environment". Inputs flow into the system, are processed, and eventually emerge as output. If the scope of the system is expanded, i.e. additional processes become incorporated into the system, then the system boundaries are moved to include more of what was previously part of the environment.

In examining the decomposition of selection systems into their functional components, a series of processes is found: objects are processed into modified objects, which are, in turn, affected by other processes to become further modified objects. The granularity of the analysis is somewhat arbitrary: processes can typically be broken down into finer and finer subprocesses. Hence the level of analysis (the extent to which subsystems are defined) can reasonably depend on the purpose of the analysis.

A *transforming* operation in this context is the mapping of some procedure across each of the members of one set in order to derive a new *transformed* set of objects. It is necessarily a one-to-one mapping from the original set to the new set, where each member of the new set is a (possibly) modified copy of its corresponding member of the original set. A simple example of such an operation is copying. Each member of the original set is copied into a new derived set.

At some level of generality all information selection systems processes can be thought of transformations from one state to another, but, for the present purposes, the distinction between two types of transformation appears useful:

- Representation Making. Using rules to derive a representation (a copy or a version) of a datum into a corresponding, modified datum. Data are changed or at least copied.

- Partitioning (sorting, selecting) a subset of data objects according to some criterion expressed as a query for a matching process or as an ordering rule. Data are reorganized rather than changed.

The term "retrieval" tends to subsume three meanings: selecting (identifying); locating (lookup); and fetching (delivery). The first meaning – selecting (identifying) – is what interests us here. We follow [Belkin & Croft 87] who provide a useful classification of retrieval techniques and characterize the process as a matter of comparing and matching, either exactly or partially. The variety of retrieval techniques – the form and degree of acceptable comparability – is very large: exact match; partial match; match using truncation; fuzzy, positional, and other relationships; Boolean matches; etc. Multiple techniques can be combined and there are limitless degrees of progressively weaker matching. We follow [Belkin & Croft 87] in regarding the retrieval process itself as a comparing or matching process. However, the purpose or *function* (as distinguished from the procedures) of this matching is to *partition* the stored representations into a set of subsets.

In information selection systems, Representations are partitioned into the two subsets: retrieved and not-retrieved, as in basic Boolean systems. But there can be degrees of matching and each different degree of matching can be used to create another partition. The limit is reached in document-ranking systems in which, at least in principle, each representation is partitioned into a separate subset with one member. We can, therefore, while accepting that the process is a matching procedure, emphasize that it is functionally a partitioning activity. With this in mind, we can regard the formal query as being a partitioning instruction. It may sound odd to refer to information retrieval as "partitioning with respect to relevance" but that is an accurate statement of the intent.

In brief, while the process may be one of matching, the function is one of partitioning and we can conclude that this is a different kind of operation from, say, copying. Sorting is logically the same as partitioning: To sort into categories is to partition into categories.

2 A Basic Model of Information Retrieval Systems

Models of information retrieval systems are commonly found in information retrieval texts and papers (e.g. [Lancaster 79, p. 8]; [Meadow 92, p. 5]; [Soergel 85, p. 58]; [Vickery & Vickery 87, p. 11]; [van Rijsbergen 79, p. 7]). Such models are generally in the form shown in Figure 1, with varying amounts of additional descriptive detail depending of the purpose of the description.

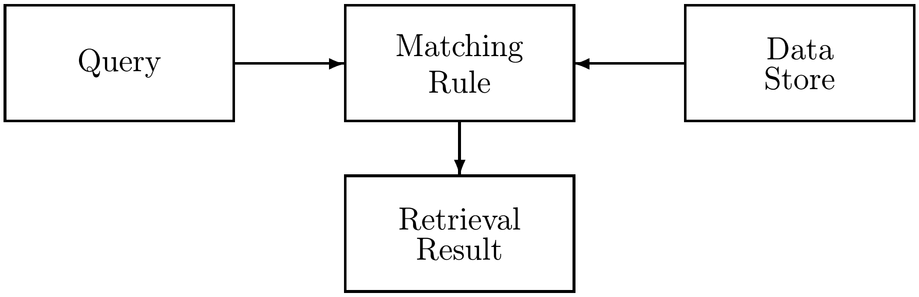

Figure 1: General model of information retrieval systems.

[Tague *et al* 91] published a "complete formal model for information retrieval systems" using production grammars and hypergraphs to represent text structure, indexing, and access. However this is really a procedural model of text retrieval techniques. Descriptions of the operation of individual retrieval systems are likely to have detailed flow diagrams of that particular system's components. Here, however, we are interested in developing a complete, generalized functional analysis of information selection systems.

To develop a complete and general model of the functional components of bibliographic information storage and retrieval systems we proceed by outlining a descriptive model of information storage and retrieval procedures. This illustrative model is intended to be minimally complete in that it includes all the *different types* of functional components found in all retrieval systems. The hope is that the components identified in this basic, illustrative model could be used to construct a functional representation of any information storage and retrieval systems of any complexity, including extended retrieval architectures. As a check on the adequacy of the analysis three examples of information storage and retrieval systems will be examined later.

2.1 Input

System boundaries are arbitrary. Where they are drawn determines which flows of data are regarded as flowing in to and out of the system in whatever form. The inputs, queries and records, may be retained or discarded (perhaps by being relegated to other storage). There can also be feedback concerning any process.

- One kind of input supplies the stored and potentially retrievable data: documents, bibliographic records, images, etc., and/or representations of them.

- Queries constitute another kind of input in one of several forms: free-text, boolean keywords, formal query language statements, etc.

- External knowledge may also be drawn upon in the form of controlled vocabularies, syndetic structures, subject headings, descriptions, etc.

There can be multiple outputs. The most obvious output is the expression of the retrieval results, in whatever form. More generally there can be feedback reporting the effects of any procedure.

With this in mind, we now outline the functional components that appear to be necessary and sufficient to represent information storage and retrieval systems. Not all components are present in all systems. Some components could be present more than once. Components may be implemented in more than one way, i.e. using different techniques. Note also that, as is usually the case in systems analysis, the granularity of the components is somewhat arbitrary. We propose that the following components, displayed in summary in Figure 2, are necessary and sufficient, between them, to represent the functionality of all operational information storage and retrieval systems. The intention is that the analysis will be technologically independent, one that would be as valid for paper-based as for computer-based retrieval systems.

2.2 Input Streams

User Query

One form of input from the environment is the User's query, an expression of the user's information need, more or less compromised by the user's expectation of and experience with the information retrieval system. The "user" is ordinarily thought of as a human being, but the query could well be generated by a machine and only indirectly by a human being, such as in the case of relevance feedback or multi-stage retrieval.

Source Objects

A set of Source Objects of interest: documents, records, artifacts, images, signals, etc. These arise in the environment outside of the information selection system. A general theory of information storage and retrieval should be able to include bibliographic systems (searching records representing documents), "full-text" searching, copies or representations of museum artifacts, and, indeed, of any kind of definable phenomena, including imaginary ones. The set of source objects may well be a carefully selected set, as in a library or museum collection.

These objects and/or copies and/or transformations of them become "resource input" to the retrieval system. Through a variety of possible processes, they become the Representations in the system.

External Knowledge Sources

External Knowledge Sources are used in information selection Representation Making, Index Making and Query Development. In Representation Making, the external knowledge may be in the form of what people know or has been recorded concerning the Source Objects, their contexts (e.g. domain knowledge), their possible representations (e.g. linguistic knowledge, thesauri, etc.), or their internal structures and interrelationships, are another resource which can be drawn on as a supplement to or as a substitute for the source objects.

Such knowledge can also be used in Query Development and Index Making in the form of thesauri, controlled vocabularies, subject heading lists, classification schemes, syndetic structures (use, see also, etc.), dictionaries, search intermediaries, etc. One of the major research areas in this field is to see how far this external knowledge can be formalized and moved *inside* the system and used in this way.

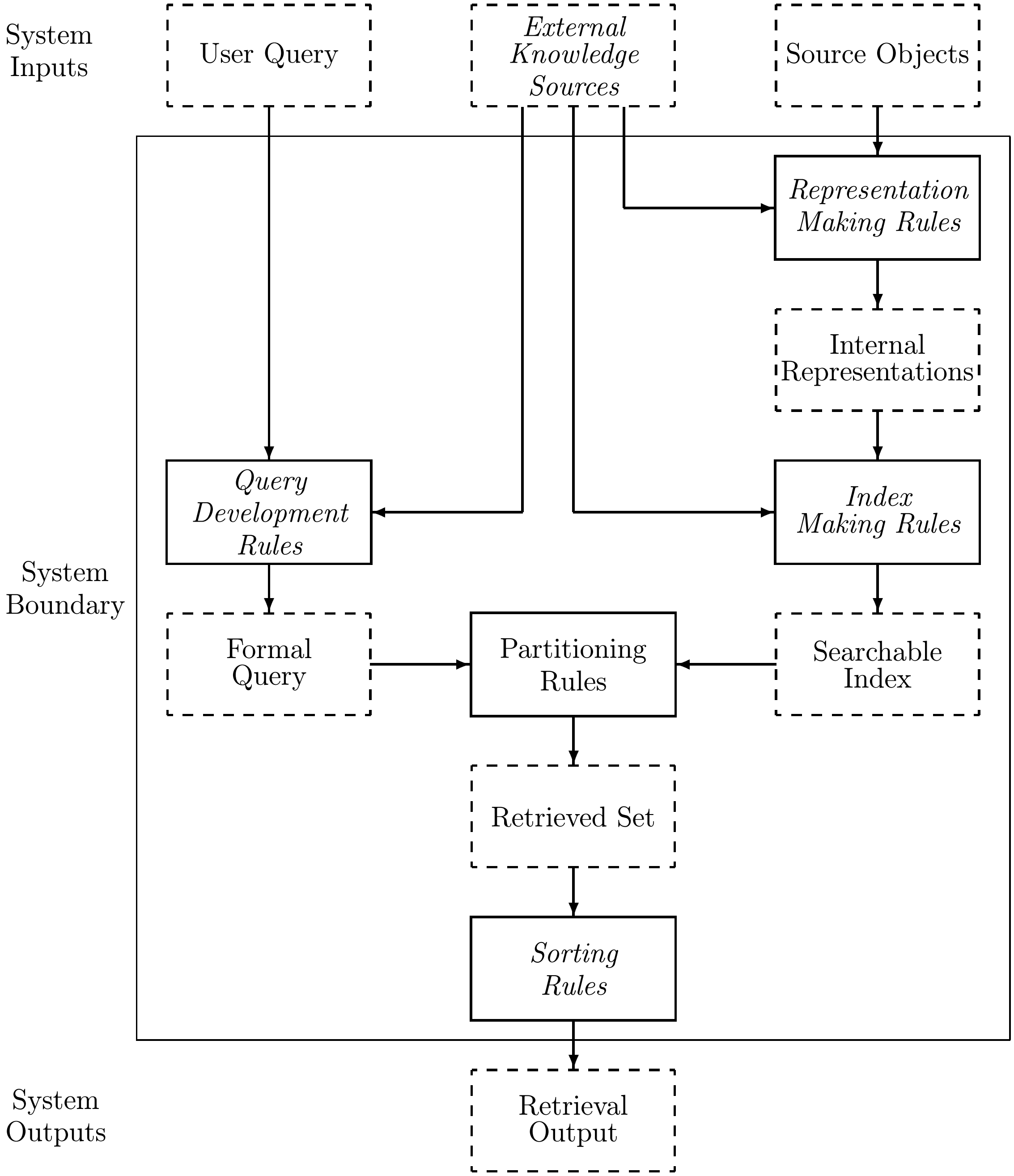

Figure 2: A minimally complete model of bibliographic oriented information retrieval (selection) systems. Solid boxes indicate processes ("transformers" or "partitioners"), dashed boxes data objects. *Italics* indicate optional components. Arrows show flows (or streams) of information objects. Note that the only required "process" is the central partitioning rule, and that subcomponents are formed by patterns of Objects ⇒ Process ⇒ Objects.

In an ideal world these three processes (Representation Making, Index Making, and Query Development) would all draw on the same, identical knowledge sources, but this is unlikely in practice. With the rise of client/server architectures, we can expect separation of the Query Knowledge Sources from the Representation Knowledge Sources and the Searchable Index Knowledge Sources. Knowledge Sources need to be continuously revised and updated and there is no assurance that the updating will be identical and synchronized. Further, the use of an External Knowledge Source in creating Representations is chronologically prior to the use of the External Knowledge Source for Query Development and may be several years prior, creating possible vocabulary problems even if the same External Knowledge Sources were used.

2.3 Internal Components

Representations

Representations of the source objects are composed from the resource inputs in some combination of a copy or transformation according to the Representation Making Rules of part (or all) of the Source Objects and/or any (external) representations or descriptions (from External Knowledge Sources) of those resource objects. Representations can be derived from:

1. Source Objects
    - Part or all of the object itself, possibly copied and/or transformed. For textual objects these could include the text, title, original abstract, etc. For images, these could be scanned copies. (These are the "brute facts" of [Blair 90].)
    - Descriptive features implicitly in or algorithmically derivable from the object: e.g. word occurrence, frequency, and co-occurrence; automatic abstractions from (or patterns recognized in) images; etc.
2. External Knowledge Sources
    - Features derivable from other objects inside (e.g. relative word frequency in relation to a corpus) or outside (e.g. synonyms of topical terms) the retrieval system that are related to this object.
    - Description or documentation of the object: description of the physical object and/or statements about the origins of the object and/or what the object signifies, e.g. subject headings, subject classification.

Depending on the nature and extent of the Representation Making Rules, the Representations, then, might be a more or less transformed copies of the Source Objects: in a collection of unedited full-texts, each text (or copy of it) would constitute its own Representation. It might be a more of less transformed description of the object: in museum registration the representation might include an image of the object, but none of the original object itself (unless, presumably, it is a museum of electronic objects). In other cases the representation could be derived in part from the source object and in part from a description: in bibliographic systems, such as a library catalog, fragments derived from the object (e.g. title, publisher's name) would be combined with pieces of description (e.g. subject headings).

Searchable Index

Since the Representation is what is stored, the Representation is also that which could, in principle, be searched and, following selection, produced as output for display or other purposes. But this is not necessarily supported in practice. Current online library catalogs, for example, typically restrict searching to a few fields (notably author, title, and subject headings) within Representations that contain several other fields in which searching is not supported. This is sufficient reason why it is necessary to make a distinction between the Representation and the *Searchable Index*. The Searchable Index, in this technical sense, is the searchable part of the Representation. We use "Searchable Index Rules" to denote whatever determines what is to be searchable. Retrieval systems commonly have in addition, a syndetic structure for mapping permissible searches (see, see also, stop words, etc.), which we also treat as a second component of the Searchable Index. Again, in the case of unedited full-text, the Searchable Index will be co-extensive with the Representation and, therefore, with the Source Object (the original text). But, as noted, in other cases, such as library catalogs, the Index Making Rules can restrict which parts of the Representation are available in the Searchable Index. The Searchable Index (like the Representation and the Source Object) might be partitioned into separate (sub)indexes, to allow more precise, targeted searching.

Procedurally, the Index Making process can be implemented in different ways: the Searchable Index might be derived by literally making parts of the Representations available for searching; it might be derived by copying parts of Representations; it may even be that part or all of the Representations exist physically only as fragments distributed via the Index Making Rules to the Searchable Index to be reassembled if and when needed. But we regard these alternatives as functionally equivalent and are not interested here in the technical details of implementation (storage costs, search effort, delay, etc.) that will make one technique preferable to another.

Query Development Rules and Formal Queries

Query development is a function that mediates between the User Query and the Formal Query. It transforms the user's query in order to harmonize it with the system's vocabulary of retrieval commands, index specification, and index vocabulary prior to retrieval. This role has traditionally been seen as an important function for skilled human intermediaries.

Computer-based query development that can match queries with the vocabulary in (or expected to be in) the system's Searchable Index is commonly called an "entry vocabulary" module. Examples include CITE [Doszkocs 83], PaperChase [Horowitz *et al* 83], and Grateful Med [Cahan 89]. Automation of this function is promising and offers scope for expert and probabilistic techniques. "Entry vocabulary" modules parallel the syndetic structure, thesaurus and controlled vocabulary aspects of External Knowledge Sources used to create the Representation. It might ideally draw on the *same* thesaurus or other knowledge representation scheme, but it cannot be assumed that the same external sources will be used for these different components.

A query development system may be absent, present, or multiply present in any given retrieval system. (We will discuss query development in more detail below).

The Formal Query is the query as it is seen by the Matching Rule, after it has been transformed by the Query Development Rules. Examples of such formal transformations include truncation, weighting, substitution, normalization, vectorization, etc., many of which are conversions of "external" representations to "internal" representations. Such transformations apply both to computer and human based retrieval systems.

Retrieved Sets

A Retrieved set is logically a subset of the Representations as partitioned off by the outcome of the Matching Rule applied to the Formal Query and the Searchable Index. When displayed (or delivered as output) the retrieved set may be complete copies or very incomplete, transformed versions of members of the set of Representations. Note that this is not necessarily a simple binary outcome: Retrieve and Not Retrieved.

Sorting Rules

Commonly, but not necessarily, there is a separate process of sorting the retrieved set. Online library catalogs typically reorder retrieved sets alphabetically by author (strictly, by "main entry") prior to display. In card catalogs the order of the retrieved set is predetermined by the order in which the cards were filed. With retrieval systems that generate a strict rank-ordering, the ranked order preempts any postretrieval reordering. For a more detailed discussion see [Buckland *et al* 93].

2.4 Output Streams

Retrieval output, traditionally in the form of a display, but increasingly in the form of a stream of objects to be used elsewhere or for some other purpose, completes the basic retrieval cycle. Such streams can be directed to visualization tools, storage for later processing, or use as Input Streams to other selection systems, or as feedback within the system itself.

2.5 Feedback

There is, in principle, an opportunity for feedback from the output of any process. For example, the output from any process can constitute feedback to other processes and to the searcher. Feedback can provide the basis for expert judgment or expert systems at any stage. Since feedback is a property of all systems and is not specific to retrieval systems we note it here, but do not explore it further in this paper.

3 Implications

This illustrative description of a complete retrieval system provides a basis for further analysis. We do not claim that all retrieval systems are as depicted in Figure 2. Some are less complete, others more complex, but we do suggest that all retrieval systems could be depicted using these functional components. Not all components are always present. Some components may be present multiple times. The assumption with which we proceed is that in all cases the User Query, Formal Query, Source Objects, Representations, Searchable Index, Partitioning Rule, a Retrieved Set (possibly empty) and Retrieval Output (also possibly empty) are *always* present in any functional information selection system. The Representations, being derived from some combination of Source Objects and/or External Knowledge Sources, require that at least one or other of the latter must be present as a direct source input. The other components may be absent or present. We now take a closer look at the nature of these components.

3.1 Types of Components

The minimally complete model in Figure 2 represents a series of data objects, transformations and partitionings. The Representations are derived from Source Objects and/or External Knowledge Sources by way of Representation Making Rules. The Searchable Index is derived from the Representation and the Index Making Rules, which itself partitions the Representation into index terms. The Formal Query is a transformation, by way of the Query Development Rules, of the User Query. The Matching Rule partitions the Representations, through the Searchable Index and the Formal Query, into the Retrieved Set. The Retrieved Set may be subjected to a Sorting Rule before becoming Retrieval Output. These transformations of datasets can be seen as being of two types.

All of the operations ("methods" in Object Oriented terminology) carried out in existing information selection systems are either *transformations*, i.e. *representation making* operations, or *partitionings*, i.e. rearranging operations. User Queries are transformed into Formal Queries. Source Objects are transformed into Representations. Representations are transformed into Searchable Indexes. Representations are partitioned into retrieved and non-retrieved sets. A sorted Retrieved Set is transformed (i.e. copied) to Retrieval Output.

3.2 Implicit Sorting

A rather overlooked feature of retrieval systems is the practice, common in boolean bibliographic retrieval systems, of re-ordering (applying a *Sorting Rule to*) a retrieved subset (usually alphabetically by author) prior to display or other form of output. Conceptually, a "sort" instruction is brought to bear on any set of retrieved records – sort by author, for example – which is specified either ad hoc or by default. This component (a Sorting Rule applied to the Retrieved Set) of the retrieval process is functionally equivalent to the retrieval (Matching) rule. The only significant difference is that the Sorting Rule operates only on a subset of records, i.e. those previously partitioned off by the retrieval mechanism.

3.3 Query Development

The Query Development Rules component serves to transform a user's information need, or query, into a representational form suitable for submission to the Matching Rule. This role may be performed by a human intermediary and involve complex interaction and negotiation with the user (the "reference interview" [Jennerich 87]). Where query development is performed algorithmically it most often appears to have two parts:

- Query normalization transforms the query into a form expected to fit the matching process, e.g. the forms "Jane Doe" and "Doe, Jane" need to be processed into a form of surname and forename that the matching process can handle. This, then, is a matter of using rules to the create and a version (a representation) of the user's query that will be acceptable at the next stage.

- A different kind of query development occurs in the process, sometimes referred to as "entry vocabulary", by which terms used in the user's query are replaced by terms used in (or expected to be in) the system's Searchable Index. It is, especially, a matter of substituting an "authorized" index term for a user's term that is not used in the index (e.g. converting "jail" and "gaol" to "prison") or with the one or more terms that appear to be the closest approximation. As use of the term "substitution" indicates, this is conceptually a one-to-one mapping or “transformation” process. However, it can also be seen as a “matching” process, that is, functionally, another, separate retrieval (i.e. sorting and ordering) process: Find and separate out the term(s) that most closely (or most likely) match the term(s) in the user’s query. Like a “known item” search for a book it may often be thought of as a one-to-one correspondence, but it is not necessarily so. It may be a one-to-many or a probabilistic process. This, then, is another case of using a partitioning procedure to select some subset from a predetermined set.

“Entry vocabulary” procedures in the form described above are not yet commonly provided though we expect them to become so. However, two other forms are common:

- Where the user’s query is expressed by selection from menus, the menu options are the system’s “terms”

- Where the browsing of index terms is supported, a user’s query generates a selection (partitioning) of related system terms, retrieving only the Searchable Index or from a separately stored thesaurus. Selecting from this partitioned subset can then generate a new query.

3.4 Representation Making

As with Query Development, matching (partitioning) processes can occur in Representation Making. The process of deriving a Representation from a Source Object, as noted above, can be seen as having two components. In some cases the representation will be a copy of the object, possibly modified by technical constraints in the system (e.g. accents lost in representing a foreign text or loss of color or resolution in a copy of an image). In other cases, the representation may be a description of the source object (e.g. in a catalog composed of textual descriptions of non-textual objects.) The representation could be a combination of copy and of description (as in library catalog records which include fragments such as title copied from the original and added description such as subject headings).

In the terminology that we have adopted the creation of Representations is primarily and obviously Representation Making. This is clearly the case with any portion of the representation that is a copy derived from the Source Object. An algorithmic representation (such as automatically generated word occurrence vectors) is derived in the same way. It may no longer be a recognizable copy, but it is a descriptive representation of the source object. The Representation may also derived from a copy of a description from some other source such as an existing database descriptions (as in the bibliographic utilities heavily used in creating library catalogs) or be an expression of what some human indexer’s perception.

Nevertheless, an ordering process may occur within Representation Making if and when some part of the Representation is constrained by a controlled vocabulary. It is desirable to select only terms acceptable to the system, whether done by machine or by human. This activity is functionally very similar to entry vocabulary control.

A significant problem arises in retrieval systems with large, long term stores of Representations when changes are made to the Representation Knowledge Source or to the Representation Making Rules. Should these changes be applied retroactively to previously processed representations? Is it feasible? Notoriously, library card catalogs were often not updated retroactively, although in some cases, such as changed names or terms, the consequences could be mitigated by changes to the syndetic structure in the Searchable Index (e.g. by adding ”see also” links from new to old forms).

3.5 Multi-Stage Retrieval

There are no grounds to limit retrieval to these two stages: partitioning, then (possibly) ordering. It is a trivial matter to imagine more than two stages. Simple boolean retrieval systems lend themselves to a series of progressively modified searches by allowing prior queries to be successively modified (e.g. with the addition of further limits in the form of boolean ANDs or NOTs). One might start by selecting records containing, say, the subject keyword NAPOLEON; the retrieved set, especially if large, might then be partitioned (or filtered or sorted) by language, to bring forward items in English; each subset record might be subsorted by date and/or by title and/or by author. Such a two-stage strategy is one answer to the difficulty of achieving high precision as well as high recall [Buckland *et al* 93]. A search intended to achieve high recall can be followed by a different kind of search of the initially retrieved set in order to produce a second retrieved set with better precision [Buckland & Gey 93]. More generally, information retrieval is ordinarily an iterative process in that the result of one matching is often the basis for a secondary or modified search with another query that modifies or replaces an earlier one.

3.6 A More Abstract Formulation

Our examination of the basic illustrative model depicted in Figure 2 has led to the conclusion that each of different components can be viewed as being composed of either data objects, or one of two different processes which we have called "transforming" (representation making) or "partitioning" (selecting or ordering). If accepted, this view has interesting consequences:

- At a theoretical level it indicates that information storage and retrieval systems are composed functionally of sequences of just two kinds of activity: Transformations (the deriving of new versions) of data and Partitioning (sorting, ordering, selecting) of subsets from larger sets of data objects.

- We established above that the partitioning function (alias matching, ordering, selecting, sorting) does not occur only once, but, except in very simple information storage and retrieval systems, can be expected to occur repeatedly.

There are several different implementable procedures (techniques) for partitioning and transforming. At a technical level, this more abstract formulation creates, in effect, an invitation to experiment by substituting alternative partitioning and/or transforming techniques at each point at which any ordering occurs, and a framework within which to evaluate such work.

Figure 3 offers a more abstract restatement of Figure 2 in terms of these two functional types.

4 Three Examples

As a check on the validity of this approach we describe three different selection systems in the terms we have been using.

Boolean Library Catalog

The illustrative model in Figure 2 was, itself, based on the typical form of current first and second generation online library catalogs. The Representations (catalog records) are derived, via Representation Making Rules, from the Source Objects (books, periodicals, microforms, etc. in the

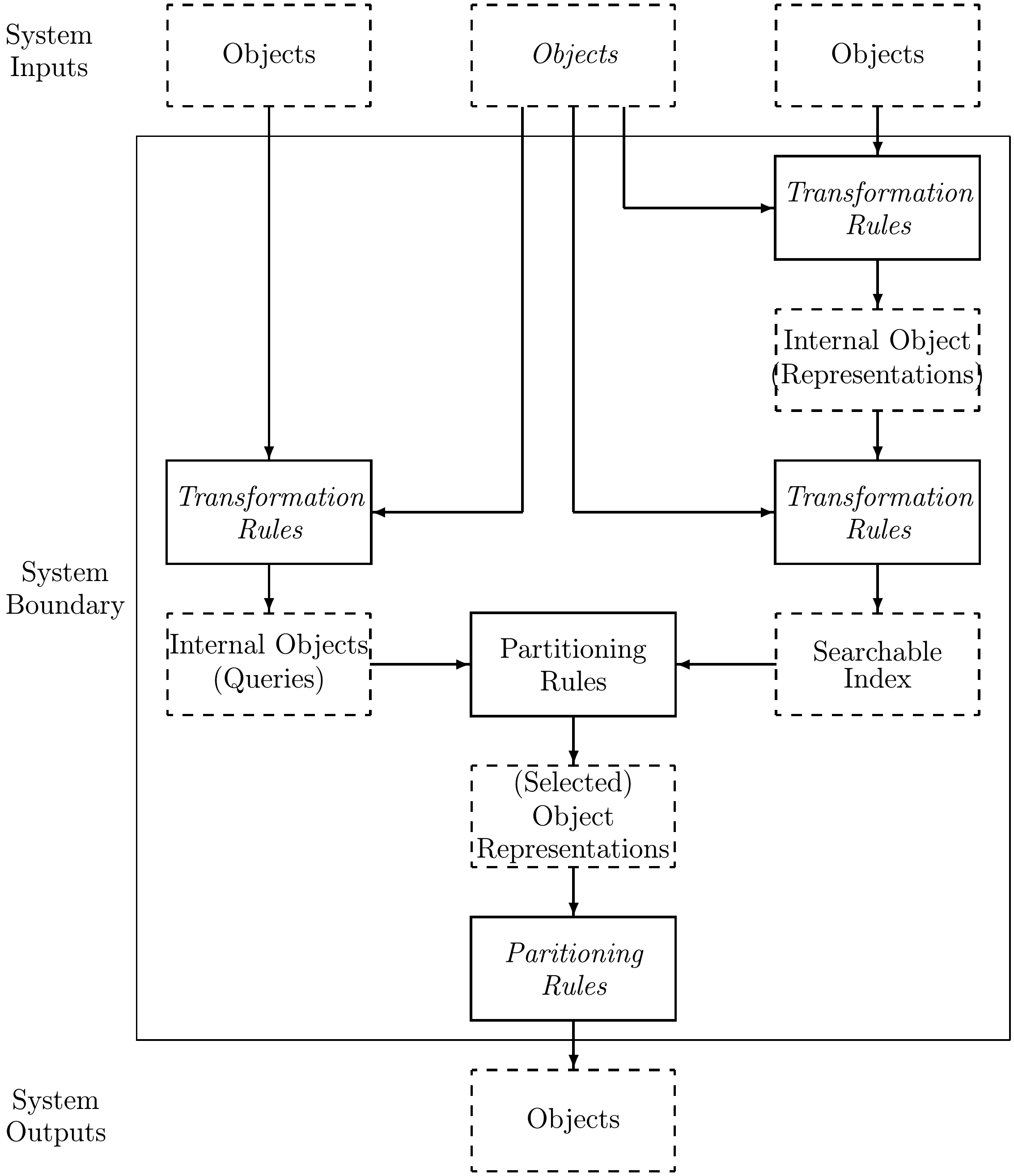

Figure 3: A minimally complete abstract model of information retrieval (selection) systems. Solid boxes indicate processes ("transformers" or "partitioners"), dashed boxes data objects. *Italics* indicate optional components. Arrows show flows (or streams) of information objects. Note that the only required "process" is the central partitioning rule, and that subcomponents are formed by patterns of Objects ⇒ Process ⇒ Objects.

collection). Some of this is the direct copying of fragments (e.g. titles, call numbers); some is a more complex intellectual representation derived from External Knowledge Sources (e.g. assignment of subject headings and classification numbers). In practice the representations are largely indirect copies, being derived directly from external sources, especially in the form of other cataloger's previously created catalog records (copy cataloging) rather that directly from the source object. The Searchable Index is limited in practice to a small number of the fields of the catalog records (Representations).

User Queries are accepted into the system from users either as well-formed and normalized Formal Queries, in the case where the searcher is experienced and uses the "command line interface", or in some less well-formed format that must be go through a Query Development process before searching.

The Query Development process commonly has an option (or requirement) that it be a two stage process. Two of the primary examples of such transformations in library catalogs are index browsing and transformations which rely on External Knowledge Sources. In the former, the User Query is used to select a subset of the terms from the Searchable Index which the user *scans*, and from which, the user then selects one or more terms which are used to retrieve the Representations associated with these terms. Such selection by "browsing" is, in effect, a two-stage retrieval process.

Queries may also be transformed with various sorts of External Knowledge, such as thesauri, dictionaries, controlled vocabulary lists, subject headings, etc. In online catalogs, these sources nearly always have some sort of syndetic structure ("broader term", "narrower term", "use for", "see also", etc.) which can be used, either algorithmically or by hand, to harmonize the User Query with the system vocabulary (Searchable Index) for better results. For example, a syndetic structure may be in place so that one form of a query term will be represented as another, e.g. a search for Mark Twain will retrieve Samuel Clemens and vice-versa.

Retrieved sets are normally re-ordered by main entry before being output as a display or as a stream of records. This re-ordering of retrieved sets is, in effect, an automatic, "hardwired" partitioning instruction. Future online catalogs will probably allow the user to choose what re-ordering or aggregating (by date, by availability, etc.) to invoke.

Full-Text Retrieval

In a simple case of retrieval from full-text, electronic texts would be stored (copied into the system) to become the Representation, and all of the texts would be searchable for the occurrence of specified phrases, words or word fragments. In such a case, if all of the text can be searched, the Searchable Index is actually (or logically) co-extensive with the Representation. The Retrieved Set could consist of either partial or complete copies of the those texts which satisfy the Matching Rule. The syndetic structure component of the Searchable Index is absent.

One degree more complex would be to represent the relative location of pairs of terms or to impose some vocabulary control in the form of stop words so that the significance of a term could be represented more reliably. More sophisticated still would be information storage systems which use or include algorithmically generated representations (e.g. weighted vectors of terms) of the texts.

Message Filtering

Systems for filtering electronic messages (or other objects) constitute an example in which objects are represented, filtered (searched) and then discarded or relegated to other storage. In this case the User Query, once developed, remains indefinitely in place as a stored instruction (Matching

Rule) which is used to select messages (Representations) as soon as they have been copied into the system. A stored, “standing” query resembles the default alphabetizing of retrieved sets in online catalogs: Both are, in effect, latent matching instructions, instrumental in partitioning whatever data may come their way. In this sense, filters with stored queries and transient data objects are symmetrical with retrieval systems with trier transient queries and stored data objects.

Primitive retrieval systems based of a serial scan of searchable records, such as the mid-twentieth century “rapid selector” machines for scanning long spools of microfilm, can be seen as an intermediate design between typical modern filters and typical modern retrieval systems.

5 Discussion

5.1 Vocabulary Issues

Much of the difficulty in non-trivial information retrieval arises from discrepancies between vocabularies, especially between the two vocabularies that are always present: the vocabulary of the User Query and the vocabulary of the Searchable Index. In practice, several more or less different vocabularies may be present:

- The vocabularies as used in the source documents themselves. These can be expected to vary among the documents, especially when written by different authors
- The vocabulary of the Searchable Indexes derived from the Representations, and moderated by any syndetic structure (External Knowledge Sources)
- The vocabulary of the user as expressed in the User Query
- The vocabulary of the Formal Query resulting from query development through the influence of any human or algorithmic intermediary
- The vocabularies of the External Knowledge Sources used in Query Development, Representation Making and Index Making.

A vocabulary problem can arise in the transition from any one of these vocabularies to any other. Vocabulary control (standardization) can occur in document creation, in the creation of document representations, in syndetic structure, and/or in query development. A remedy could, in general, be provided at any of these stages, not necessarily the stage at which the mismatch or ambiguity arises.

5.2 Functional Decomposition

As described above, we conceive of all the functional components in a retrieval system as being either “objects”, “transformers” (representation makers) or “partitioners”. Further, as Figure 3 suggests, the components follow a regular alternating pattern of Object Streams ⇒ Transformer or Partitioner ⇒ Object Streams as in Figure 4.

The uniformity of this compositionalily of system components suggests that this is a useful level at which to describe the *subsystems* of information retrieval systems. Further, we argue that if these subsystems contain at least one Partitioner, they can be characterized as *complete* selection systems. Like the models in Figures 2 and 3, these subcomponents have Input Streams, Transformers or Partitioners (or possibly both) and Output Streams. For example, an “entry

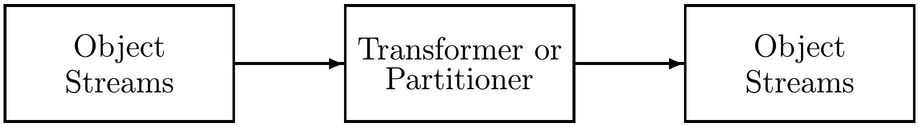

Figure 4: The common pattern of relationships between all subcomponents of information selection systems.

vocabulary" module in a Query Development subsystem may "select" (or "retrieve") the correct authorized terms from a system vocabulary. Its inputs would be the User Query, the Source Objects would be the authorized system vocabulary terms, and the might even be External Knowledge in the form of syndetic references. The output of such a subsystem, consisting of the authorized terms that best match the User Query, can be feed into its associated retrieval systems, as illustrated in Figure 5.

If accepted, this argument leads to a basic definition of information selection systems: information selection systems can be viewed as one or more interconnected *processors* (i.e. a *transformer* or *partitioner subsystems*), at least one of which is a *partitioner*, operating on one or more sets of input objects, and producing one or more sets of output objects.

A definition of this nature in turn suggests that one could build very flexible "workbench" for information retrieval experimentation out of a number of relatively simple subsystems, themselves built from the components described above, which could be mixed and matched. Given such a basis, several projects immediately suggest themselves, such as comparing alternative functional components on representative test collections with some hope of making reasonable evaluations or comparing of the consequences of trying to remedy vocabulary problems at different stages in the selection process.

5.3 Functional Diagnostics

A functional analysis of the type presented above provides an framework for the systematic exploration of the range of possibilities in the design of information storage and retrieval systems. It may be that because information storage and retrieval systems have to be complete in order to function at all, there has been an understandable tendency to concentrate on the construction of complete systems and, when comparison has been attempted, to compare complete but complexly different systems or to compare minor variations within a single system. Comparative investigations of retrieval performance, examining the effects of systematic changes in individual components of the system, can be undertaken only with an analytical, modular approach.

Some simple examples of problems and of possible remedies can indicate the variety of options [Buckland n.d.]:

Retrieval problem 1: Insufficient choice of documents. Solution: Add more Source Objects (documents).

Retrieval problem 2: Insufficient basis for appraisal (relevance judgment). Solution: Expand the Representations by copying more of the document and/or adding more description (External Knowledge) from what is known about the document.

Retrieval problem 3: Insufficient indexing for adequate searching. Solution: Expand the Searchable Index by improving the Index Making Rules.

Retrieval problem 4: Retrieval is unreliable because the Searchable Index vocabulary is inconsistent. Solution: Extend the syndetic structure by adding more cross-references expressing relationships between index terms (External Knowledge).

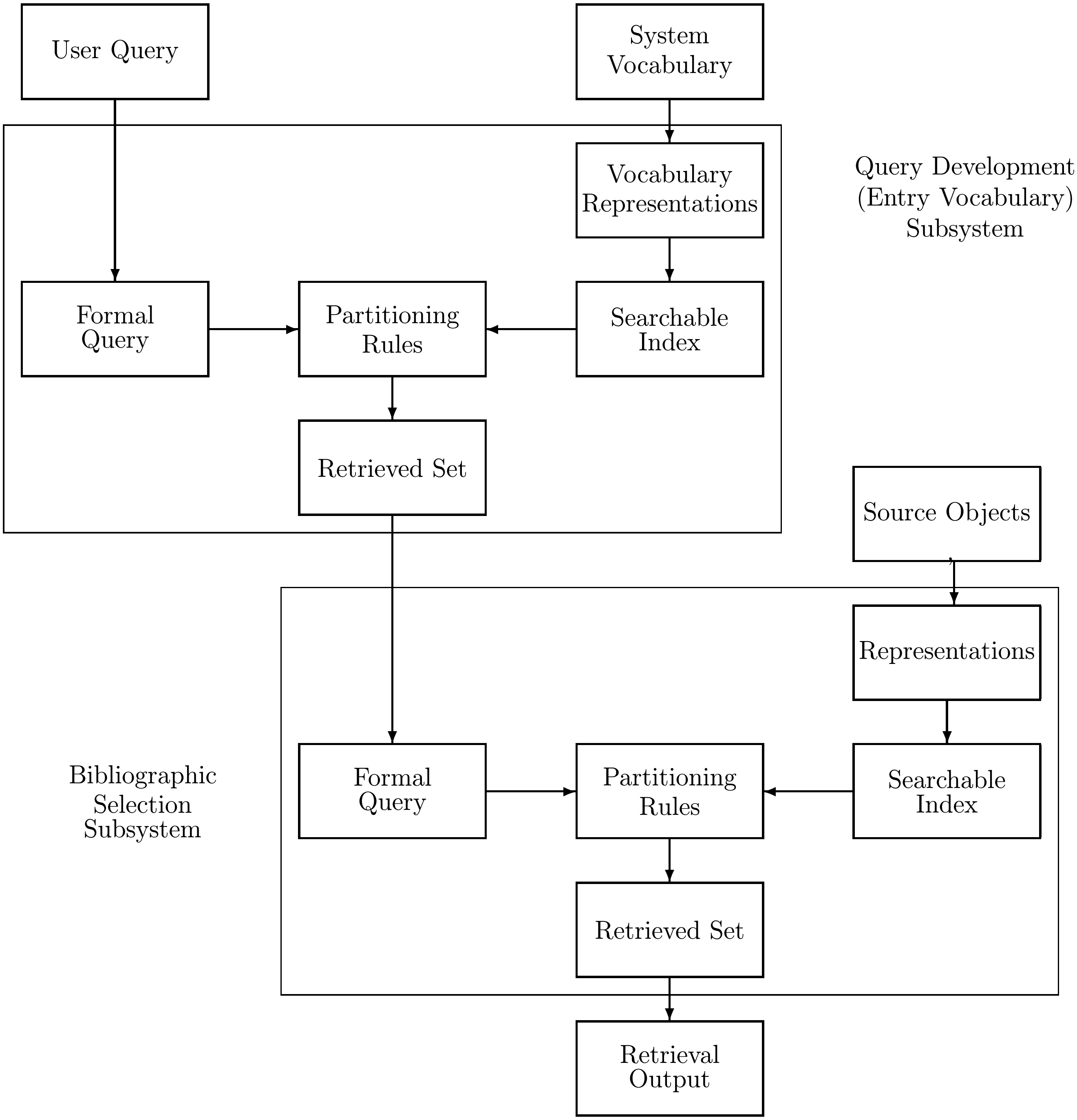

Figure 5: An expanded model which illustrates one way in which retrieval subsystems can interact within a selection system. The "Query Development" component of a bibliographic selection system has been expanded into complete, separate "Entry Vocabulary Subsystem". Scanning (or browsing) authorized subject headings lists, whether manually or in an automated environment, fits this model. The system components shown here are only those which are required.

Retrieval problem 5: Clumsy retrieval: Significant information is in the Searchable Index but is not readily usable (e.g. due to output overload). Solution: Extend retrieval capability, possibly by more refined two-stage retrieval: a broad search for high recall followed by a secondary searching (or filtering) of downloaded supersets to improve precision.

Retrieval problem 6: The index vocabulary is unfamiliar to the user. Solution: Provide an entry vocabulary module to the query development module in order to modify the vocabulary of the Formal Query.

In each case a different module provides the location for a plausible solution. For a systematic examination of these issues a detailed functional analysis of information selection systems is needed. Gross comparisons of two substantially different entities cannot provide the analytical power of controlled comparisons based on the systematic variation of individual components.

5.4 Library Science and Information Science: A Unified View

This functional analysis provides a path away from the arguments that used to characterize information retrieval in the post-World War II period. Any relatively complete functional analysis of information storage and retrieval should provide a basis for the mapping of research and development activities. We suggest that the following components that have been and remain of most interest in Library Science:

- The use of human intermediaries in query development
- An emphasis on incorporating external knowledge (expert descriptive cataloging, classification, and assignment of subject headings) into the representation (catalog record)
- Vocabulary control (alias authority control) in creating representations, in syndetic structure, and in query development
- In online catalogs, minimally a two-stage approach (a Boolean operation to partition the Representations, followed by the alphabetization of the Retrieved Set) and commonly, a three stage approach (the two-stage approach preceded by a search of the Searchable Index only, for feedback)

The activities generally referred to information retrieval research have historically tended to emphasize:

- In storage, the use of algorithmic alternatives to human expertise in creating representations and indexes. Good examples are automatic keyword indexes (e.g. KWIC) and the generation of vector space representations of documents' terms
- In retrieval, the use of highly elaborate partitioning (retrieval) and transforming algorithms leading to the strict ranking of a set of retrieved documents

Others might prefer to nominate other techniques as being characteristic of these streams of research and development, but any realistic mapping on to a general framework of information storage and retrieval theory is likely to reveal how *complementary* rather than *contradictory* these interests are. There is a difference in emphasis: the former tending to emphasize quality of data, consistency, and expert human intervention; the latter, exploring efficient algorithmic approaches to large volumes of data. Neither approach alone can provide a complete approach to selection systems in theory or in practice.

Acknowledgments

This work was partially supported by the United States Department of Education Higher Education Act Title IID grant #R197D00017, Prototype for an Adaptive Library Catalog. We are grateful for the comments of Clifford A. Lynch and Patrick Wilson.

References

[Belkin & Croft 87] Nicholas J. Belkin and W. Bruce Croft. Retrieval Techniques. *Annual Review of Science and Technology*, 22:109–145, 1987.

[Blair 90] David C. Blair. Language and Representation in Information Retrieval. Amsterdam: Elsevier, 1990.

[Buckland n.d.] Michael K. Buckland. The Potential of Extended Retrieval. *United Nations University Second International Symposium on the Frontiers of Science and Technology: Expanding Access to Science and Technology – The Role of Information Technologies, Kyoto, 12-14 May 1992.*, Proceedings. Tokyo: United Nations University Press, (forthcoming).

[Buckland *et al* 93] Michael K. Buckland, Barbara A. Norgard and Christian Plaunt. Filing, Filtering, and the First Few Found. *Information Technology and Libraries*, 12:3:311-319, 1993.

[Buckland *et al* 93] Michael K. Buckland, Mark H. Butler, Barbara A. Norgard and Christian Plaunt. OASIS: A Front-End for Prototyping Catalog Enhancements *Library Hi Tech*, 4:10:7–22, 1993.

[Buckland & Gey 93] Michael K. Buckland and Fredric Gey. The Relationship between Recall and Precision. *Journal of the American Society for Information Science*, 45:1:12–19, 1994.

[Cahan 89] M. A. Cahan. GRATEFUL MED: A Tool for Studying Searching Behavior. *Medical Reference Services Quarterly*, 8:4:61–75, 1989.

[Doszkocs 83] T. E. Doszkocs. CITE NLM: natural-language searching in an online catalog. *Information Technology and Libraries*, 2:4:364–80, 1983.

[Harman 93] Donna K. Harman, ed. Text REtrieval Conference (TREC-1) (NIST-SP-500-207). Washington DC: NIST, 1993.

[Horowitz *et al* 83] G. L. Horowitz, J. D. Jackson, and H. L. Bleich. PaperChase: Computerized Bibliographic Retrieval to Answer Clinical Questions. *Methods of Information in Medicine*, 22:4:183–88, 1983.

[Jennerich 87] Elaine Z. Jennerich. The Reference Interview as a Creative Art. Littleton, CO: Libraries Unlimited, 1987.

[Lancaster 79] F. W. Lancaster. Information Retrieval Systems, 2nd ed. New York: Wiley, 1979.

[Larson 92] Ray R. Larson. Evaluation of Advanced Retrieval Techniques in an Experimental Online Catalog. *Journal of the American Society for Information Science*, 43:1:34–53, 1992.

[Meadow 92] Charles T. Meadow. Text Information Retrieval Systems. New York: Academic Press, 1992.

[Soergel 85] Dagobert Soergel. Organizing Information: Principles of Data Base and Retrieval Systems. Orlando, FL: Academic Press, 1985.

[Tague *et al* 91] J. Tague, A. Salminen & C. McClellan. Complete Formal Model for Information Retrieval Systems. *Proceedings of the Fourteenth Annual International ACM/SIGIR Conference on Research and Development in Information Retrieval, Chicago, 1991*, 14–20, ACM Press, 1991.

[van Rijsbergen 79] C. J. van Rijsbergen. Information Retrieval, 2nd ed. London: Butterworths, 1979.

[Vickery & Vickery 87] B. Vickery & A. Vickery. Information Science in Theory and Practice. London: Butterworths, 1987.