

by the lexical component of the natural language. Moreover, he believed that not all words of a document or a request were necessarily important in transmitting meaning. Of course, this did not mean that all words could not be considered important. This only meant that not all *had* to be considered important. Taube presumed that those words that were important in the opinion of the IR system creators for the transmittal of the meanings of texts must be included in the vocabulary of the IRL system. He suggested that such words be called *units*. The name is a portmanteau created from combining the words "unit" and "term," and it is intended to emphasize the system's use of single terms. A little later, other terms, specifically *keywords* or *descriptors*, began to be used instead. The majority of researchers uses these later terms.

Now let us discuss why the words of the texts in natural language must comprise the lexical component of the IRL. It would seem that in cases in which the meaning of the texts is transmitted by all words of such texts written in natural language, that is, when all the words of a natural language are meaningful words, the need to create an artificial IRL does not exist. However, this is not so. In reality, Taube's suggestion means that for purposes of retrieval, one may ignore both the grammatical and the semantical components of the texts in natural language when transmitting the meaning. Obviously, the use of the lexical component of a natural language does not mean the use of the natural language itself. The same lexical components may exist in the natural as well as the artificial languages. Their principal differences are their different grammatical and semantical rules. It is precisely these rules for the artificial IRL that comprise the basis of Taube's idea. "If we presume," he reported, "that the meaning of a communication is transmitted only by a set of words contained in that communication, then the documents which contain a set of words contained in some request also contain the meaning possessed by the request" (Taube et al.). This leads to the following rules for a grammatical component and the rules for selection and output based on the semantical component. We start with grammatical rules.

1. The contents of each document in the IRL must be represented by a set of keywords (uniterms), not in any particular order, encountered in the text of this document.
2. The contents of each request in the IRL must be represented by a set of keywords (uniterms), not in any particular order, encountered in the formulation of this request.

The rules of comparison and selection are formulated in the following way:

1. The set of keywords of a request must be compared with the set of keywords of each document.
2. When the set of keywords from a document contains all the keywords from a set of a request, the document is considered found.

The IRL proposed by Taube was an important step on the way to creating the modern IR systems, because the implementation of these simple and formal rules on the computer did not pose difficulty. It should be noted that even now this approach forms the basis of the overwhelming majority of the IRLs in use.

During the initial stages of the creation of IR systems (the early 1950s), no practical attempt was made to use the entire lexical component of a natural language as the lexical component of the IRL. This approach was only considered theoretically. This was, first of all, explained by the fact that the speed and memory of the first generation of computers were literally in the infant stages. In addition, technical methods, other than typing, for entering information into a computer were practically nonexistent. Partly because of that, the indexing of texts was often looked at not only—not even primarily—in terms of transferring texts into an IRL. It was generally accepted that the indexing process must significantly reduce the volume of the entered text, and a number of researchers considered this property to be the main intent of indexing. This belief led to a substantial effort on the part of IRL developers to select a small number of lexical units that would be able to represent as great a portion of the meaning of the text being indexed as possible.

But what is a part of the meaning? This question was addressed by C. Mooers who viewed meaning as something single, homogeneous, and consisting of a certain set of independent elements. In September of 1948, C. Mooers patented an automatic document retrieval system which he called a system of "zato-coding" (Mooers, 1951). In 1950, he introduced the term "descriptor" to define the lexical units of an IRL utilized in his retrieval system. The author described the essence of the method of text indexing, which was utilized in the IR system he created in the following manner:

The subject of each document is characterized or described by means of a certain set of descriptors borrowed from the formal vocabulary of relevant terms. As a rough approximation of what I mean, I recall the subject heading list. . . . Each document is characterized by a certain set of descriptors from the descriptor dictionary. Each descriptor from this set is applicable or, in a certain sense, correct with respect to the information content of the information element (element of meaning). In this case descriptors function independently from each other. The fact that this set consists of several descriptors can imply that they form some interrelated composition in the source document. This also can imply that they correspond to different independent ideas spread throughout the document. (Mooers, 1951)

According to Mooers, a descriptor is a word or a word combination designating a certain concept with more or less broad meaning. The meaning of the descriptor is not necessarily required to coincide with the meaning normally corresponding to the word or word combination used as a descriptor. Each descriptor is supplied with an explanation mark disclosing its specific meaning in the given IRL. Normally, descriptors have broader meaning than corresponding words and word combinations. Next we give an example of building an alpha-