Although the information used for adaptation is given by the user, the user represents it in the way that was required by the developers of the system. Already in developing the very first functioning IR systems, especially for the selective dissemination of information (SDI) in the early 1960s, the researchers were quite aware that during the first few iterations the system had to adapt to the user's POIN (see, e.g., Curtis & Rosenberg, 1965; and Meadow, 1967). The adaptation was done on the basis of previous outputs. This was typical adaptive feedback, although it was quite primitive in its implementation. At that time it was not called adaptation—since there were no on-line systems—and the user interacted with the system through an intermediary searcher. In the process of such feedback, the user's evaluation (or reaction) to the output after obtaining the pertinent documents was used as a signal to start adaptation, and the evaluated documents were used (analyzed) by intermediary searchers in the process of correcting query formulations.

The user's evaluation of the output was not always directly used for correcting query formulations. Often the developers of the system assumed that the user would reformulate the search request, and the developers therefore planned to use this new formulation, containing additional information, for correcting the old query formulation. But, as is clear from properties of POIN, it is possible that the user's new formulation may be worse than the original formulation (Frants & Brush, 1988). Also, constructing a new search request is often a problem for both the user and the developers of the system (for example, the user may be reluctant to change a previous request that he or she considers the best possible). These reasons probably led the developers to conclude that it is more expedient to obtain additional information about POIN directly from the user's evaluation of the previous output, that is, from marked (pertinent) documents. This approach clearly minimized the user's efforts in presenting additional information about his or her POIN, since the user evaluated the output regardless of the presence of feedback in the system.

It seems clear that the first (manual) attempts to realize feedback in IR systems were directed not toward developing a mechanism of adaptation but toward developing methods for collecting additional information about POIN. This gave some positive results; for example, researchers concluded that the marked documents provided the best way to represent the additional information about POIN as well as the easiest way for users to formulate search requests. After the first successful implementation of automatic feedback by Salton in the system SMART (which does not use the Boolean search criterion), the use of marked documents to provide additional information about POIN became predominant (Salton, 1968). Hence, that additional information about POIN entering the system through the line of feedback is now the user's reaction to the output. In the majority of systems, the user interacts with the system directly (on-line) and the user's reaction can be obtained very easily. It is sufficient to indicate for each document evaluated by the user if it is pertinent (for example, by pressing key "y") or not (by pressing "n").

Hence, we know now that the state of the system in changed (for the purpose of adapting to the user's POIN) by correcting query formulations on the basis of information contained in the user's reaction to the output. It seems that we are ready to proceed with the discussion of existing approaches to automate feedback process in IR systems. However, first it is necessary to look at the way the collection of documents is used during search because the mechanism of feedback depends on how we use this collection during search.

Two different cases affect the character of a system's adaptation to the user. The first occurs when the collection of documents does not change during each iteration (a *static collection*) and the control is performed, for example, during one session in the on-line system or in the case of a retrospective search. The second case occurs when the collection changes in different iterations (a *dynamic collection*), such as in the case of SDI.

In realizing feedback for a static collection of documents, the task of control consists of an attempt to find pertinent documents that were not found during the previous search without exceeding an acceptable noise level (for a given system) in the output. The feedback algorithm in this situation will construct a new query formulation, which will be used in an attempt to find new (presumably relevant) documents.

In realizing feedback for a dynamic collection of documents, the task of control is different from that in a static collection. In a static collection, we try to obtain a new query formulation (at every new iteration) that does not intersect with any of the previous query formulations. But in a dynamic collection, we only try to correct (improve) the previous query formulation. In other words, we try to remove all subrequests that led to the unsuccessful search and to add new (original) subrequests that will presumably give us pertinent documents. Hence, in the case of a dynamic collection our task is not to find relevant documents that were not found during a previous search but to have a more successful search in a new collection of documents; that is, we want to construct a new query formulation (by correcting the previous one) so that if we performed two searches in the same collection using the previous query formulation and the new one, the latter search would give us better results.

But what is the basic difference between two types of collections, that is, what kind of searches are performed in a static collection versus a dynamic collection? In both Chapter 2 and Chapter 4 we stressed that the design of an IR system depends on the properties of POIN, and in creating an IR system we are trying to take into account all of POIN's properties (and if a system contains some process or mechanism, it exists only because it is required by POIN). We know, for example, that SDI is based on attribute 3 of POIN and one time search (or search in one particular collection) is only a stage in a process of providing continuous service to the user. From the analysis of attributes of POIN, it is clear that the process of adaptive feedback in a static collection is directed primarily toward satisfying attribute 2 of POIN, whereas in a dynamic collection this process of feedback is connected, first of all, to attribute 4 of POIN.