Chapter 4
THE FERRET SYSTEM

I'll fer him, and firk him, and ferret him.
—William Shakespeare
Henry V, Act IV, Scene 4

This chapter describes the architecture and components of the FERRET retrieval system, except for the script learning component, which is discussed in Chapter 5.

4.1. Architectural Overview

The FERRET information retrieval architecture shown in Figure 4-1 consists of three major components:

- The text parser
- The query parser
- The case frame matcher

The query parser is not currently implemented, but is instead simulated by the text parser and a small post-processing function. For more discussion of the requirements of a real query parser, see Section 7.3.1. In addition to these, there is a very simple user interface that allows the user to interrogate the FERRET database and retrieve texts. The interface is really just the visible part of a huge iceberg.

The text parser is like the rest of the iceberg: it’s big and slow and mostly invisible. Its job is to read incoming stories from NETNEWS and convert them into symbolic representations called “abstracts” that are stored in an “abstract database.” Two kinds of objects are produced that are called “abstracts”: the first is the instantiated sketchy script, and the second is the actual CD graph that matches the script. The core of the parser is called MCFRUMP, because it is a reimplementation of the FRUMP parser. Some of
the complexity of FRUMP is gone, and three new components have been added, a lexical scanner that preprocesses the text, an interface to *Webster's Seventh Dictionary* and a script learning component. The MCFRUMP parser is described in Section 4.3, the scanner in Section 4.2 and the dictionary interface in Section 4.4. The script learning component has its own chapter, Chapter 5.

The query parser performs a similar task for queries to the database, producing "abstract patterns" that match the abstracts from the text parser. This is the part that does not exist; but is reserved for future work. Currently, all queries are entered directly as partial CD graphs or partial instantiated sketchy scripts built by the simulated parser. See Section 7.3 for more discussion.

The case frame matcher brings the two parsers together, determining whether a given abstract pattern matches a specific abstract. Since both the instantiated sketchy script and the CD graph are case frames, the matcher has a relatively simple task to match each frame slot-for-slot and each filler by checking the type hierarchy. The matcher is described in Section 4.5.