4 CONCRETE IMPLEMENTATION OF AN AUDIO DESKTOP

4.1 INTRODUCTION

In chapter 3 we introduced the concept of an audio desktop and outlined the basic building blocks that go to make up a fluent auditory interface. The focus was on designing an audio desktop independent of any specific implementation. An audio desktop was characterized in terms of the basic user-level functionality such an environment needs to enable and the tools and techniques that can be used in achieving these goals.

These general design principles have been applied in implementing Emacspeak—a concrete instance of the ideal audio desktop envisioned in the previous chapter. Emacspeak provides a complete speech interface to a wide range of user applications. We examine the user interaction on this platform in different application settings with a view to illustrating the various design principles outlined in the previous chapter. Individual sections in this chapter focus on specific user tasks and illustrate the different search and browse strategies outlined in Sec. 3.4 with concrete examples.
Outline

The Emacspeak core facilitates a fully speech-enabled audio desktop by providing a set of basic speech and audio services that are used to speech-enable the rest of the environment. We describe these basic speech services in Sec. 4.2 and leave the implementation details for Sec. 4.11—the final section of this chapter. The intervening sections describe the speech-enabled interface in different user settings.

The basic building blocks that make up the Emacspeak desktop and the user’s mental model for navigating around this environment are described in Sec. 4.3. The editing tools available on the speech-enabled desktop are described in Sec. 4.4 with special emphasis on the differences between visual and aural interaction. Editing and navigating structured information is described in Sec. 4.5—the examples in this section emphasize the relevance of information structure and template based authoring in enabling efficient editing and navigation of complex information. Browsing special structures such as tables and spreadsheets is described in Sec. 4.6.

Today’s desktop is incomplete without a full suite of messaging applications. Emacsspeak’s rich collection of speech-enabled networking and messaging applications are described in Sec. 4.8. The focus is on designing an interface that allows the user to quickly and efficiently process large amounts of information, e.g., the flood of electronic mail and Usenet news that arrives every day on the average desktop. Along with messaging, managing personal information such as an appointment book or telephone directory makes up some of the most common day-to-day activities on the desktop. We describe speech-enabled information management tools in Sec. 4.7.

The Emacspeak desktop also provides a rich software development environment. Program source code forms a specialized class of structured documents, and this structure can be exploited when speech-enabling the editing and maintenance of large software systems—see Sec. 4.9. A productive software development environment needs more than editing tools; Emacsspeak provides fluent spoken output to a development environment that includes tools for interactively compiling, debugging and browsing large bodies of software—see Sec. 4.10.

Finally, no desktop of the 1990’s is complete without access to the World Wide Web (WWW). Emacsspeak’s speech interface to the WWW will be described in a separate chapter—see Sec. 5.3.

4.2 BASIC SERVICES FOR SPEECH-ENABLING THE DESKTOP

The Emacspeak desktop is composed of a collection of modules that can be classified as:

- Low-level interface to speech and audio devices.
- Basic services for producing audio formatted speech and non-speech auditory icons that enable high-quality auditory display.