Emacspeak

Direct Speech Access

T. V. Raman

Senior Computer Scientist

Adobe Systems

Outline

- Overview of speech applications.
- Emacspeak –Architecture.
- Emacspeak –The user experience.

Screen Access

User explores visual display to:

- **Construct a mental model of the interface.**
- **And interpret intent of the UI.**

Aural output –Derived from visual display.

Screen Access Design

- Encapsulate display in an off-screen model.
- Present this model aurally.
- Enable navigation of this model.

Aural feedback is application independent.

Available Screen Access

Character-based interfaces . . . GUI.

Richer layout of GUI

Harder to build OSM

But richer OSM

Enables possibly richer feedback

Screen Access Innovations

- Speak specific areas of the screen.
- Attach meaningful labels to icons.
- Navigate window hierarchy.

Access separate from application.

What Is UI?

1. Obtain user input

2. Compute on the information

3. Display the results

UI = Input + Output

Speech Enabling Applications

- Treat speech as a first class medium.
- Application produces its own feedback.
- Exploit features of the spoken medium.

Audio output independent of visual display.

Contrasting Approaches

Read Screen	Speech Enable
Speak display	Speak information
No context	Full context
Independent	Integrated
Global scope	Local context
Uses display	Uses environment

Advantages

- **Produce intuitive feedback.**
- **Provide a simpler user model.**
- **Reduces users cognitive load.**

User works with *one* –not two– applications.

Example

March 1996

Sun	Mon	Tue	Wed	Thu	Fri	Sat
					1	2
3	4	5	6	7	8	9
10	11	12	13	14	15	16
17	18	19	20	21	22	23
24	25	26	27	28	29	30
31						

Easy to see relevant information.

EMACS

A customizable, extensible user environment:

- Editing and browsing electronic information.
- Software development.

Unified interface→Cooperating applications

Emacspeak

- Device independent speech interface.
- Core modules for basic spoken feedback.
- Extensions for various applications.

Does not *modify* Emacs code-base.

Advice Overview

BEFORE Called *before* advised function.

AFTER Called *after* advised function.

AROUND Called *instead of* advised function.

Advice enables code re-use.

Advice

Let $x \rightarrow$ **f** $\rightarrow y$.

BEFORE **AROUND** **AFTER**

Result in function f' that computes y'.

The User Experience

- **Succinct contextual speech feedback.**
- **Auditory icons augment interaction.**
- **User action flushes prior speech *immediately*.**

User focuses on task at hand.

Demonstration –Editing

- **Simple editing, search and replace.**
- **Completion and spell checking.**
- **Syntax coloring using *voice-lock* mode.**

Intuitive interface enables fluent interaction.

Demonstration –Browsing Information

- **Browsing the file system.**
- **Reading and responding to email and news.**
- **Browsing the WWW.**

Window to digital information.

WWW –Speech Style Sheets

- **Voice properties,**
- **Auditory icons,**
- **Sound cues for document elements.**

Generate richly formatted *audio* documents.

Demonstration –System Tasks

- Running a shell.
- Running terminal based applications.

Behaves like a traditional screen-reader.

Device Independent Speech

- Dectalk Express and MultiVoice
- Software Dectalk –DEC-ALPHA
- DECFACE –A Talking Face

Supporting other devices is straightforward

Core Modules

Speech enables

- **Navigation, editing, and prompting.**

Sufficient to implement a working system.

Application Specific Extensions

The icing on the cake!

- WWW browser, Mail and News reader
- Tools: Calendar, Spell checker, . . .

Code-size is less than 1% of original.

Advice Statistics

Package	Code	Ext	%	Advice
Core	217,295	3,152	1.45%	188
W3	17,384	515	2.96%	12
Mail	17,943	270	1.50%	14
GNUS	35,528	501	1.41%	37

Emacspeak –A model of code re-use.

Conclusion

- Treat speech I/O as a first-class medium.
- Separates UI and computation.
- Enables rich, well-structured interfaces.