

Speech Dispatcher and Emacs

Milan Zamazal

Brailcom, o.p.s.

Speech Dispatcher — power speech interface

Speech Dispatcher:

- Complete support of speech synthesis
- Well defined simple interface, communication through a TCP socket
- The interface can't be programmed
- Application or common library support necessary

Speech Dispatcher — power speech interface

Application:

- No need to implement speech synthesis functions
- Needs just to define what to say

Speech Dispatcher — power speech interface

User:

- Text editor
- Mail
- Web
- Shell

Speech Dispatcher — power speech interface

So, how to connect speech synthesis to a user?

Emacs — power user tool

- Text editor
- Mail
- Web (with an additional package)
- Shell
- Programming
- ... and more

Moreover, Emacs is very extensible.

Emacs — power user tool

If we can interface Speech Dispatcher to Emacs, the basic user requirements are satisfied.

Speech Dispatcher and Emacs

- Emacs provides basic support for socket communication and full programming support, so direct connection to Speech Dispatcher is possible.
- Thanks to Emacs extensibility, many user commands can start speaking without any change of the Emacs code.
- Emacspeak can already talk to several speech synthesizers.

Speech Dispatcher and Emacspeak

- Emacspeak provides quite complete speech support for Emacs.
- Talks to synthesizers through external servers.
- One-way communication only — impossible to read Speech Dispatcher answers.
- Due to implementation constraints hard to customize, e. g. to use the priority system efficiently.
- Much of the speech synthesis support provided by Emacspeak is handled by Speech Dispatcher, in a different way — necessary to disable in Emacspeak.

Speech Dispatcher and Emacspeak

- Using Emacspeak with Speech Dispatcher not straightforward and quite difficult.
- Easy and cheap alternative right now: To write a small Speech Dispatcher client (**speechd-speak**).
- Third party support of Speech Dispatcher in Emacspeak possible.

speechd-speak versus Emacspeak

Emacspeak and speechd-speak don't compete to each other, they're alternatives instead. They serve different needs of users.

speechd-speak versus Emacspeak

User's point of view:

Emacspeak

speechd-speak

Complete desktop

Basic speech support

Everything customized

Customizations up to a user

Hard to change customizations

Easy to customize/extend

Hard to use by a non-blind user

Easy to use by any user

Older Emacs versions supported

Only latest Emacs release supported

Many years of development

Quickly written experimental package

speechd-speak versus Emacspeak

Programmer's point of view:

Emacspeak

Dtk support only

External Dtk emulators

Built-in TTS support

> 30000 LOC

Special support of many functions

Hard to hack

speechd-speak

Speech Dispatcher support only

Speech Dispatcher plug-ins

TTS support by Speech Dispatcher

< 3000 LOC

Generic support preferred

Easy to hack

speechd-speak versus Emacspeak

The basic difference:

Emacspeak	speechd-speak
-----------	---------------

Complex	Simple
---------	--------

Speech Dispatcher and Emacs (II)

On the contrary to the popular believe, Emacs is not an operating system.

Problems:

- Operating system interaction is not perfect.
- Extensibility is limited.
- Missing features.

Some speech features can't be implemented reliably without improving Emacs itself.

Speech Dispatcher and Emacs (II)

Socket communication in Emacs is not perfect:

- The `accept-process-output` function apparently suffers certain performance problems.
- Moreover, it can't be restricted to a single given process. Thus recursive invocations of `speechd-speak` functions are possible and the whole `speechd-speak` code has to be reentrant.

These problems are not fatal, but make some things harder than necessary.

Speech Dispatcher and Emacs (II)

Emacs is not extensible enough:

- Advising built-in functions doesn't always work. For instance, `self-insert-command`, `forward-char` and `backward-char` are usually invoked directly, without calling their advice; this is even an inconsistent behavior.
- Some kinds of messages and prompts are displayed without any possibility to detect them (`interactive "c"`).
- After change functions are not invoked in the `*Message*` buffer.
- It's impossible to extract text from mode line.
- Even Elisp functions can be written in a non-extensible way: `widget-choose`.

Speech Dispatcher and Emacs (II)

Missing features:

- Language translation support.

The future

- Speech Dispatcher together with speechd-speak provides the user immediate access to many facilities, without vast amount of programming.
- We believe Speech Dispatcher together with Emacs can help visually impaired users a lot.
- Past development of speechd-speak is promising, we believe it can be extended to a full featured Emacs speech support.
- speechd-speak is not the only way of interfacing Speech Dispatcher to Emacs. For instance, Emacspeak could be changed to work with Speech Dispatcher as well.
- Certainly, without appropriate resources, no work can be done.

The future

How you can help:

- Testing Speech Dispatcher and speechd-speak, reporting bugs.
- Help with development of Speech Dispatcher and/or speechd-speak.
- Improving Emacs.
- Learning, and spreading the received knowledge and experience to other users and developers.
- Any kind of support, provided with a true intention to help the impaired people and ideas of free software is welcome.

References

Speech Dispatcher

<http://www.freebsoft.org/speechd>

Emacs

<http://www.gnu.org/software/emacs>

Emacspeak

<http://emacspeak.sf.net>

speechd-speak

<http://cvs.freebsoft.org/repository/speechd-el/speechd-el.tar.gz?tarball=1>

Thank you!