

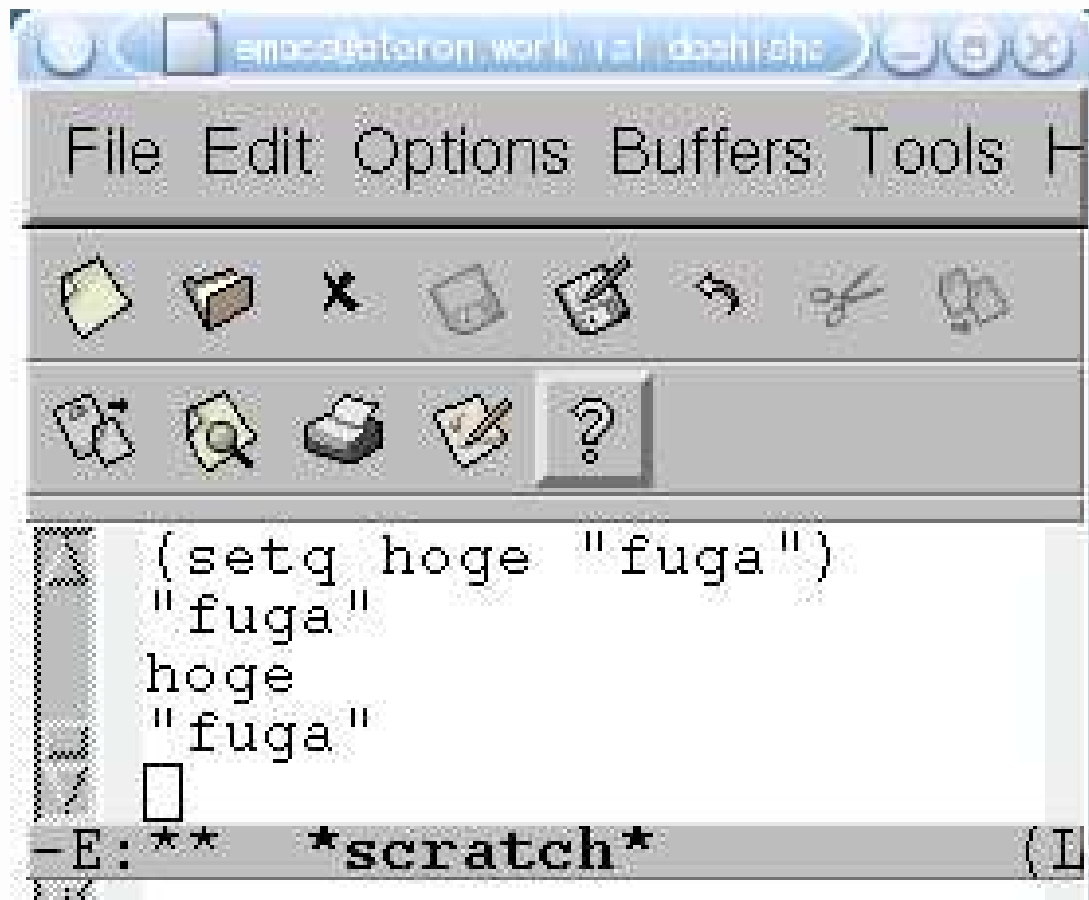
emacs を使ったウェブサービスへの展開

LILO Monthly Seminar
2003年1月

上川純一@LILO

emacs とは

- 文書編集を核とした作業環境
- elisp と呼ばれる lisp の実装



Web serviceって

- Web 経由でいろいろなサービス
 - ⇒ http を利用
 - ⇒ 動的に HTML を生成
 - ⇒ 個々のケースに対応

elserv

- Yuuichi Teranishi さんの製作した web server
- emacs lisp (+ ruby) で動く http を解析し, ページを elisp で処理して返す
- `apt-get install elserv`
- `elserv-0.4.0.tar.gz` を取得, `tar xzf` で展開, `make install` する

elserv 使用例

- Web ブラウザを利用した emacs アプリケーション
 - ⇒ mhc
 - ⇒ devscripts-el
 - ⇒ wysihtml
 - ⇒ emacs-wiki

mhc

- メールから予定を抽出するシステムを核としたスケジューラ
- メールの文面から日付とかを抽出してくれる



The screenshot shows the mhc interface with a toolbar at the top containing icons for Read, Next, Prev, Enter, Sync, Delete, Mark, New, Reply, + Cite, Forward, and Exit. Below the toolbar is a calendar for December 2002, with the 1st highlighted in red. The main display area shows a summary of an email with the following text:

```
-E:%% Wanderlust: +schedule/2003/01 (0 new/0 unread) (Summary mhc [ONLINE])
```

01/01 Wed
01/02 Thu
01/03 Fri 14:00-16:00 2003新年会

来年も1月3日(金)にJR京都駅前のルネッサンスビルを予約しました。

日時 2003年 1月3日(金) 14:00~16:00
場所 JR京都駅前 ルネッサンスビル1F カフェ・ルネッサンス
<http://www.renaissance.co.jp/kyoto/1/main.html>
京都市下京区東洞院通塩小路下ル東塩小路町849

mhc with elserv

- elserv を利用して mhc の情報をブラウザから利用できるように



mhc with elserv 使用方法

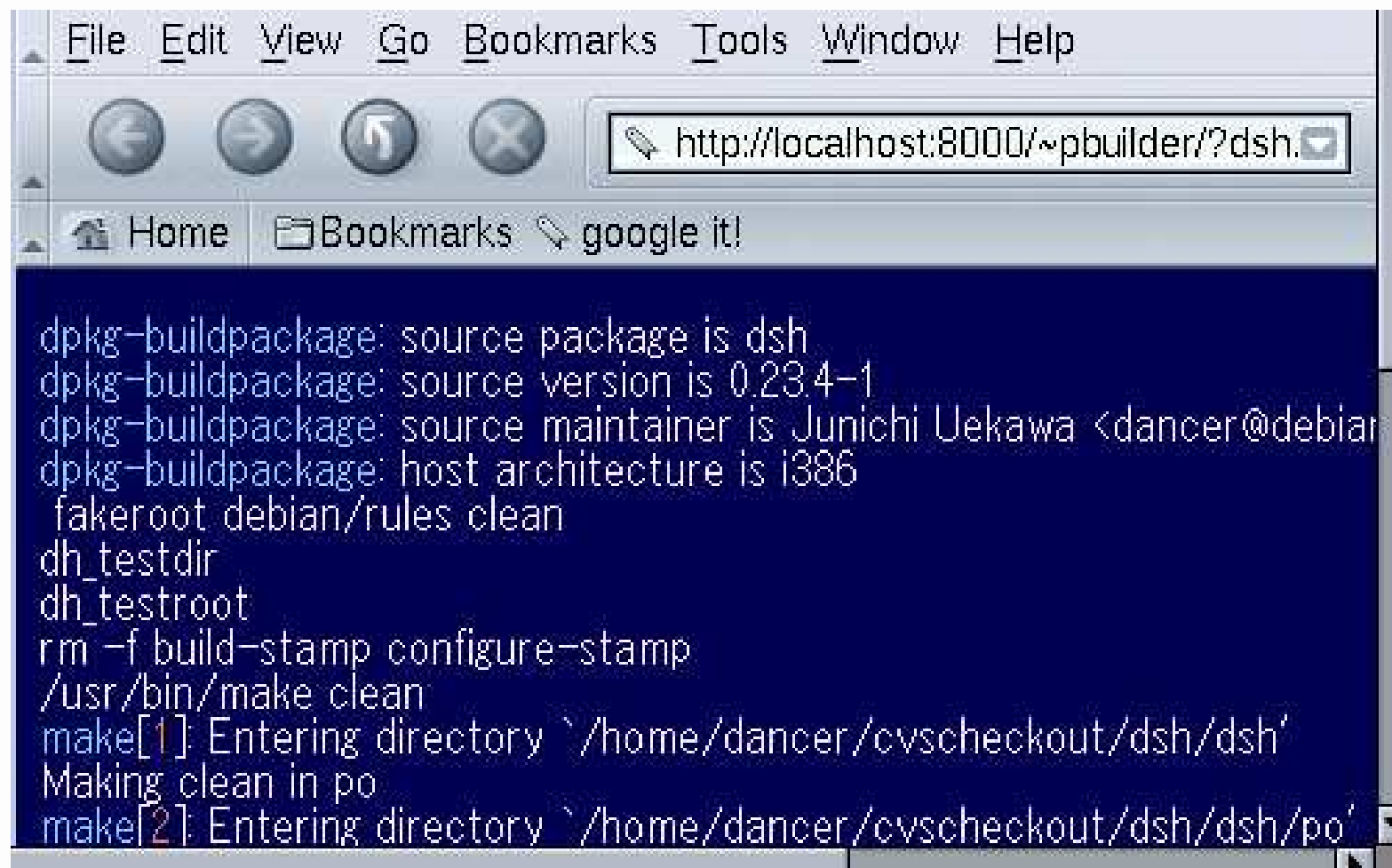
- M-x load-file es-mhc.el
- M-x elserv-mhc で
`http://localhost:10000/`
でアクセス可能

devscripts-el

- Debian パッケージコンパイル環境
- コンパイルログをブラウザ経由で閲覧



devscripts-el ログ表示



The screenshot shows a web browser window with the address bar containing `http://localhost:8000/~pbuilder/?dsh.`. The browser's menu bar includes File, Edit, View, Go, Bookmarks, Tools, Window, and Help. Below the address bar, there are navigation buttons (back, forward, home, stop) and a search bar with the text "google it!". The main content area of the browser displays a log output in white text on a dark blue background. The log output is as follows:

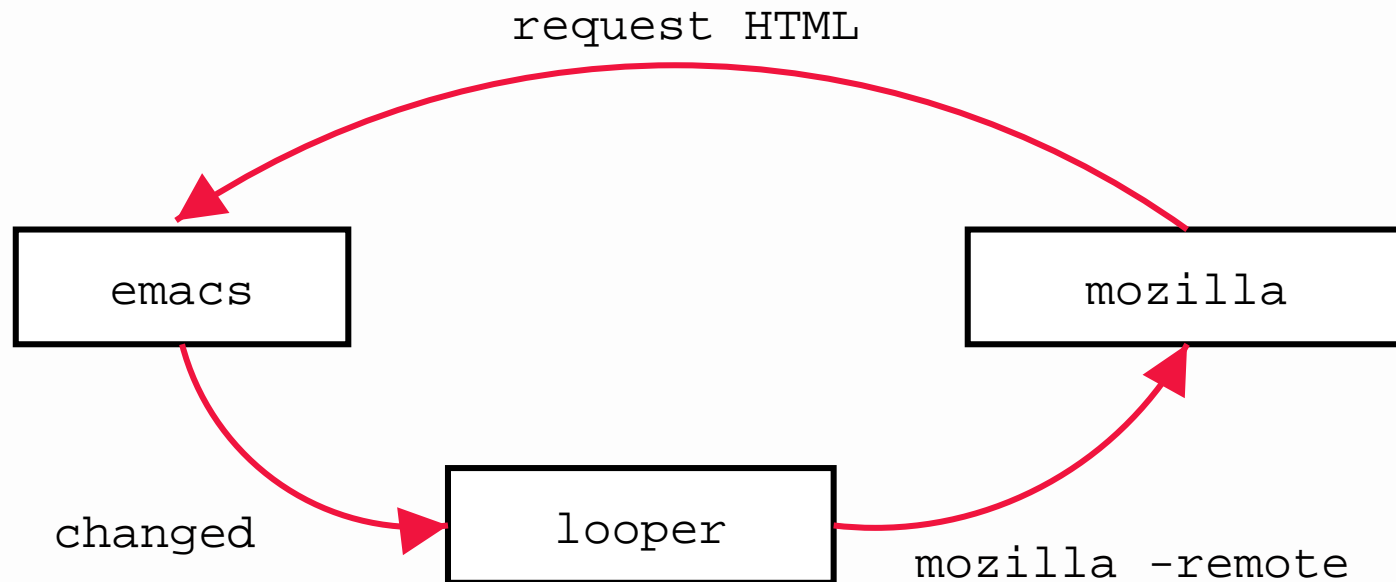
```
dpkg-buildpackage: source package is dsh
dpkg-buildpackage: source version is 0.23.4-1
dpkg-buildpackage: source maintainer is Junichi Uekawa <dancer@debian.org>
dpkg-buildpackage: host architecture is i386
fakeroot debian/rules clean
dh_testdir
dh_testroot
rm -f build-stamp configure-stamp
/usr/bin/make clean
make[1]: Entering directory `/home/dancer/cvscheckout/dsh/dsh'
Making clean in po
make[2]: Entering directory `/home/dancer/cvscheckout/dsh/dsh/po'
```

devscripts-el

- `http://www.netfort.gr.jp/~dancer/software/downloads/` からソース取得
- `devscripts-el-xx.tar.gz` を展開
- `M-x load-file` で読み込む
- `debuild` とか
- `M-x pbuilder-log-view-elserv` で起動

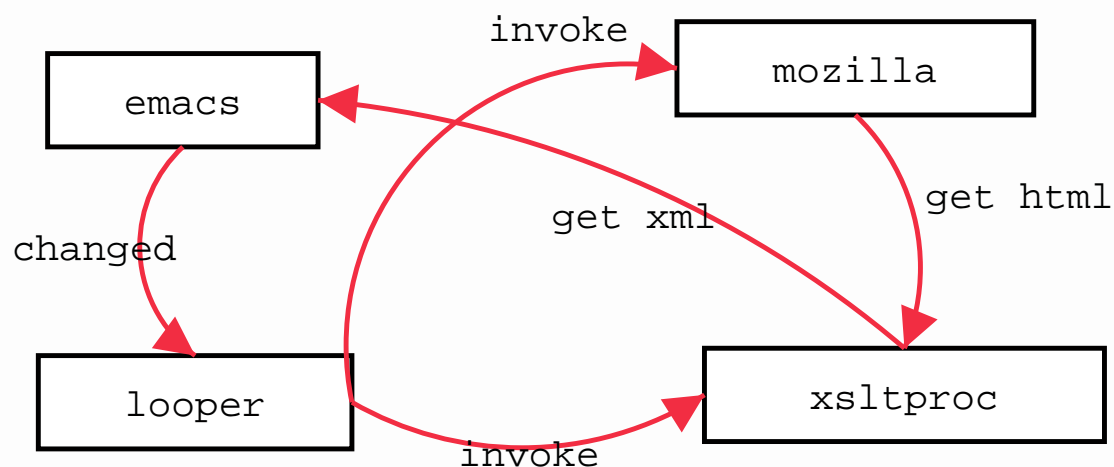
wysihtml

- emacs で編集する HTML ファイルを mozilla でプレビューしたい
elserv があれば, わざわざセーブする必要もない



wysidocbookxml

- wysihtml の考えを応用して docbook に適用してみる
- emacs 編集集中の XML の一部を抽出，elserv 経由で提供し，xsltproc で処理して mozilla で表示



wysidocbookxml 画面写真

```
File Edit Options Buffers Tools SGML Modify Move
and set the environment variables required for
<command>pentium-builder</command> to function.
</para>
</sect1>
</chapter>
<chapter>
<title>Frequently asked questions</title>
<para>
Here, known problems and frequently asked questions are
documented. This portion was initially available in README.Debian
file, but moved into here.
</para>
<sect1>
<title>pbuilder create fails</title>
<para>
It often happens that pbuilder cannot create latest chroot.
Try upgrading pbuilder and de
```

File Edit View Go Bookmarks Tools Window Help

Search

Home Bookmarks とどけこの思いK... Google

Table of Contents

- [Frequently asked questions](#)
 - [pbuilder create fails](#)
 - [Notes on usage of \\$TMPDIR](#)
 - [Specialized requirement on apt sources list](#)

Chapter 1. Frequently asked questions

Table of Contents

- [pbuilder create fails](#)
- [Notes on usage of \\$TMPDIR](#)
- [Specialized requirement on apt sources list](#)

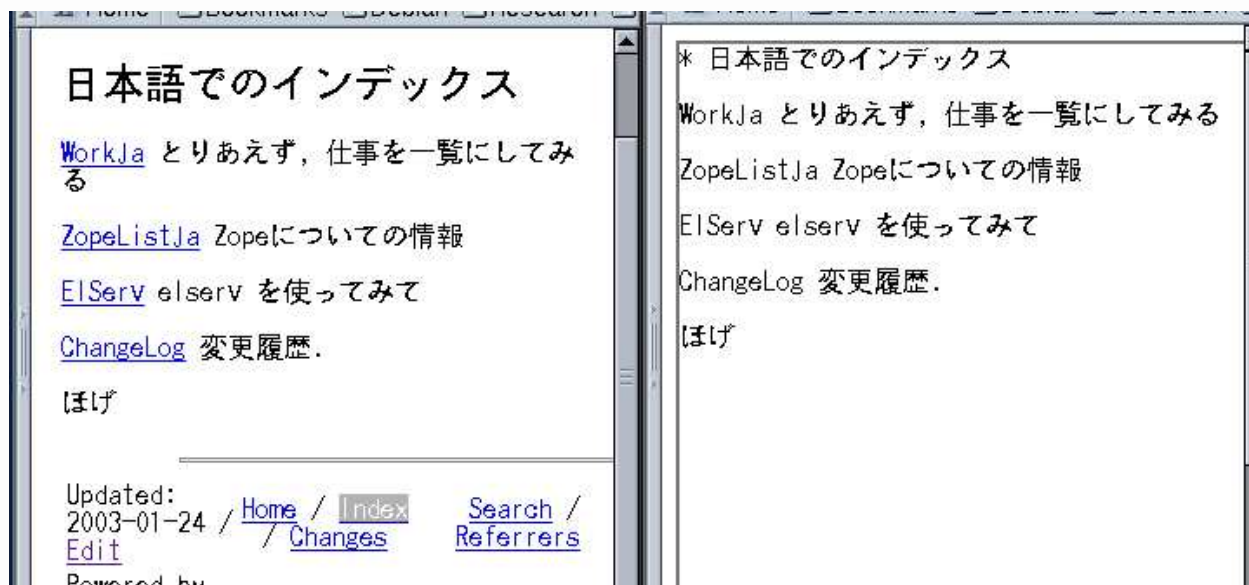
Here, known problems and frequently asked questions are

wysihtml

- `http://www.netfort.gr.jp/~dancer/software/wysihtml.html.ja`
- `wysihtml-xx.tar.gz` を展開。`./configure && make install` でインストール
- `M-x load-file`
`/usr/local/share/wysihtml/wysihtml.el`
- `M-x wysihtml-mode` で **wysihtml-mode** になる

emacs-wiki

- emacs を利用した wiki
- WikiName というように入力したらリンクになる
- ページを生成, 共同作業



emacs-wiki

- emacs-wiki をインストール
- M-x `elserv-wiki-start`
- Wiki ファイルが `a+w` の場合 `editable` になる
- emacs-wiki 単体だったら emacs から Wiki 的編集ができる

elserv のアプリケーション

- elisp が書ければ比較的簡単に作成できる
- emacs から外のアプリケーションのインタフェース
- elisp のアプリケーションと一般ユーザとの架け橋？
 - ⇒ elserv 経由のメールソフト
 - ⇒ elserv 経由のシステム管理ツール

elserv の起動

- **elserv** を指定したポート番号で起動する

```
(setq elserv-id  
(elserv-start port-number))
```

- 返り値は **elserv** のプロセス
(デフォルトの値は
(elserv-find-process)
で見付かる)

elserv でウェブページを発行

- 固定文字列のページを発行

```
(elserv-publish (elserv-find-process)
  "/hoge/fuga.html"
  :string "Hello World"
  :content-type "text/plain"
  :description "これは説明文")
```

elserv で動的ページを

- elisp の関数を呼ぶようにする

```
(elserv-publish (elserv-find-process)
  "/hoge/fuga.html"
  :function 'some-handler-function
  :description "これは説明文")
```

```
(defun some-handler-function
  (result path ppath request)
  ... body )
```

elserv で動的ページを...続

```
(defun some-handler-function
  (result path ppath request)
  (elserv-set-result-header result
    '(content-type
      "text/html; charset=euc-jp"))
  (elserv-set-result-body
    result
    (encode-coding-string
      "<html><head><title></title></head>
<body><p>テスト</p></body></html>" 'euc-jp)))
```

http://localhost:8000/

File Edit View Go Bookmarks Tools Window Help

http://localhost Search

Home Bookmarks とどけこの思いK... Google

2003/01 LIL... Elserv Elserv アブ... Elserv Monit...

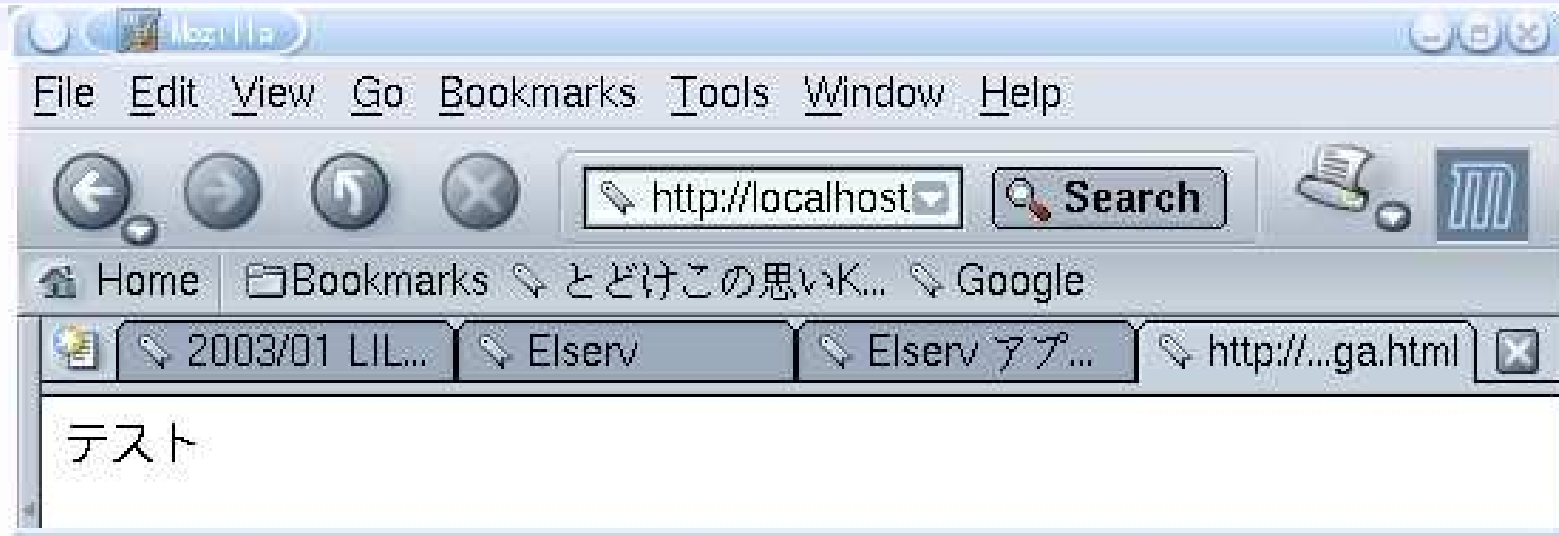
Elserv Monitor

Path	Content Description	Type
/	Elserv Monitor (Function: elserv-monitor-function)	-
/hoge/fuga.html	これは説明文 (Function: some-handler-function)	-
/icons	(Directory)	-
/monitor	Elserv Monitor (Function: elserv-monitor-function)	-

Powered by Elserv/0.4.0 (Never Surrender)

Done

http://.../hoge/fuga.html



指定した出力が出る

elserv ハンドラ関数の引数

- **result**: `elserv-set-result-*` でクライアントへ返す値を設定するのに利用
- **path**: 公開パスからの相対的値
- **ppath**: 公開パス
- **request**: HTTP リクエストに関する情報

path と ppath

- `(elserv-publish (elserv-find-process)
 "/published"
 :function 'myhandler
 :description "これは説明文")`
- `http://localhost/
published/fugafuga` へのアクセス
 - ⇒ `ppath=http://localhost/published`
 - ⇒ `path=/fugafuga`
- URL に何かをエンコードするのに使える

path と ppath

- path を利用して情報を URL に埋め込む
- elisp のプログラムが URL の値によっていろいろと違う反応を返せる
セッション管理等が可能になる

elserv で HTML をそのまま返す

```
(require 'mcharset)
(setq charset
  (detect-mime-charset-region (point-min) (point-max)))
(elserv-set-result-header
  result
  (list 'content-type (concat "text/html; charset="
    (symbol-name
      charset))))
(elserv-set-result-body result
  (encode-mime-charset-string
    (buffer-string)
    charset))
```

htmlize.el を使えば emacs の font-lock をそのままブラウザ表示で使える

まとめ

- **emacs** で動く **elserv** というウェブサーバがある
- **elserv** を使って **emacs** を外部のアプリケーションとのインタフェースを作成できる