

cweb Source Code for T_EX

Why, How, and What

Martin Ruckert, August 2021



85. Here the author of T_EX apologizes for making use of the numerical relation between 'Q', 'R', 'S', and the desired interaction settings *batch_mode*, *nonstop_mode*, *scroll_mode*.

```
< Change the interaction level and return 85 > ≡  
{ error_count = 0;  
  interaction = batch_mode + c - 'Q';  
  print("OK,  $\square$ entering $\square$ ");  
  switch (c) {  
  case 'Q':  
    { print_esc("batchmode");  
      decr(selector);  
    } break;  
  case 'R': print_esc("nonstopmode"); break;  
  case 'S': print_esc("scrollmode");  
  } /* there are no other cases */  
  print("...");  
  print_ln();  
  update_terminal;  
  return;  
}
```

This code is used in section 83.

Why?

Why do we want

cweb Source Code of $T_E X$?

- Toolchain
- Modification
- Debugging
- Education
- Experiments

Example: `ktex.w`

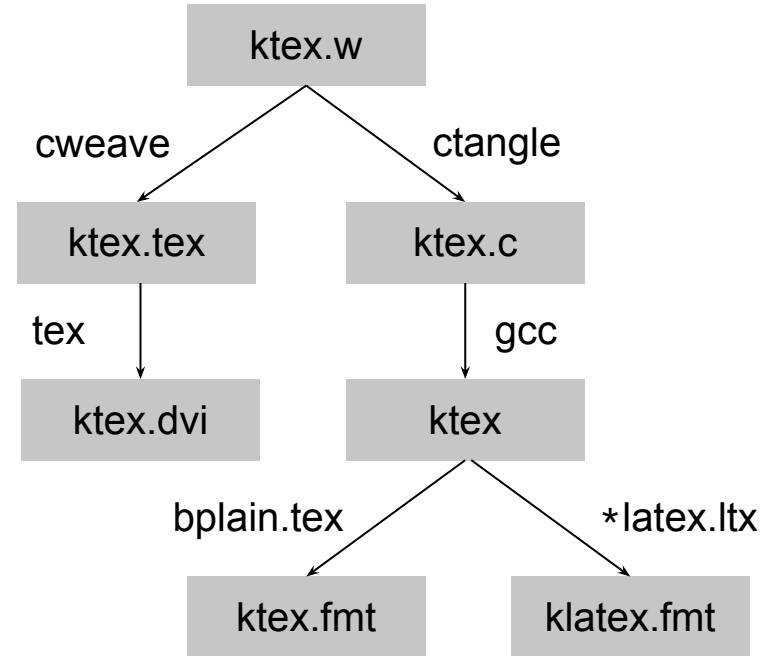
- `tex.web`
 - + `etex.ch`
 - + 32bit pointer type
 - + `kpathsea` library
 - = `ktex.w`
- It is not the purest implementation of $T_E X$.
- It is a useful basis.
- It is still close to the original $T_E X^2$.

Why?

Why do we want

cweb Source Code of T_EX ?

- **Toolchain**
- Modification
- Debugging
- Education
- Experiments

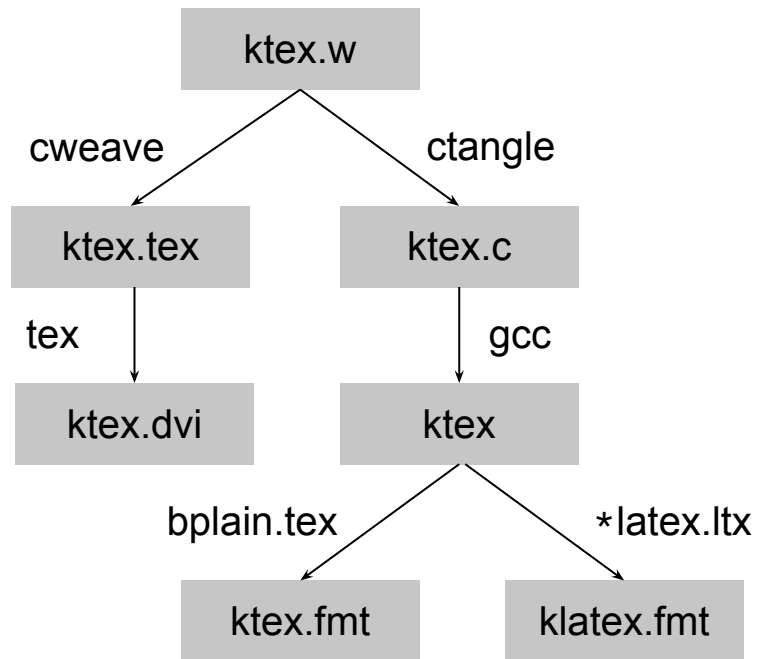


Why?

Why do we want

cweb Source Code of T_EX ?

- Toolchain
- **Modification**
- Debugging
- Education
- Experiments



Why?

Why do we want

cweb Source Code of T_EX ?

- Toolchain
- Modification
- **Debugging**
- Education
- Experiments

Example: HintView, the HINT viewer.

- Backend:
 - literate cweb programs
- User Interface:
 - **C (Windows)**
 - Java (Android)
- Rendering:
 - OpenGL

Why?

Why do we want

cweb Source Code of T_EX ?

- Toolchain
- Modification
- **Debugging**
- Education
- Experiments

```
disc_node
replace_count = 1
pre_break = f-
post_break = fi
```

```
lig_node:
ffi
f f i
```

abababcdef **difficult** dif

```
ignore_node:
info = 1
list =
```

```
lig_node:
ffi
f f i
```

abcdef abababcdef **dif-**
ficult dif diffusion file

```
ignore_node:
info = 0
list = null
```

Why?

Why do we want

cweb Source Code of T_EX ?

- Toolchain
- Modification
- Debugging
- **Education**
- **Experiments**

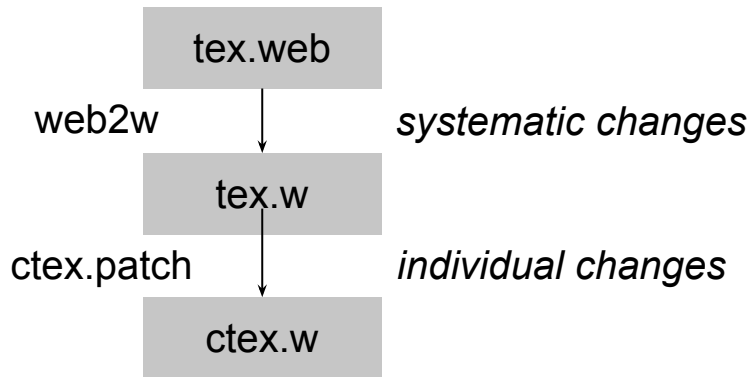
How ?

- Start with tex.web
- Apply web change files, e.g. ϵ -T_EX
- Apply patch files
- Convert web file to cweb file
 - run web2w
 - apply patch file
- Apply cweb change files, e.g. for kT_EX
- For T_EX related projects
 - Extract code and Link

How ?

- Start with tex.web
- Apply web change files, e.g. ϵ -T_EX
- Apply patch files
- Convert web file to cweb file
 - run web2w
 - apply patch file
- Apply cweb change files, e.g. for kT_EX
- For T_EX related projects
 - Extract code and Link

The basis: ctex.w

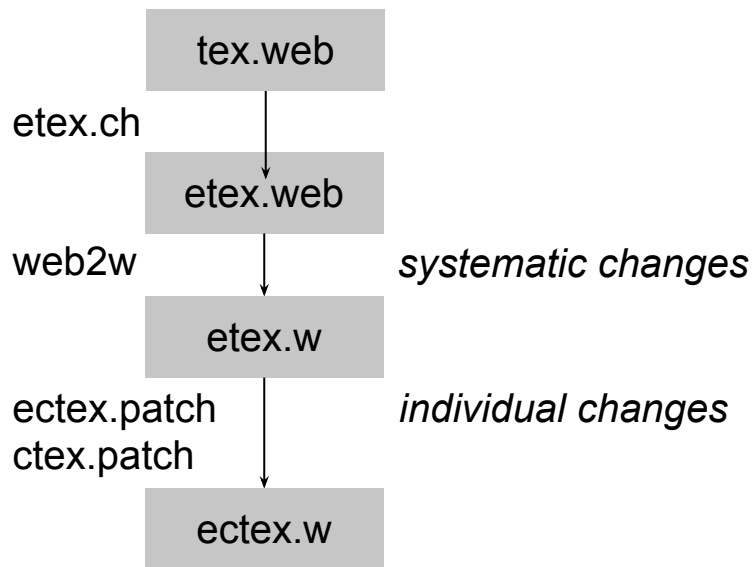


- Close to Don Knuth's original

How ?

- Start with tex.web
- Apply web change files, e.g. ϵ -T_EX
- Apply patch files
- Convert web file to cweb file
 - run web2w
 - apply patch files
- Apply cweb change files, e.g. for kT_EX
- For T_EX related projects
 - Extract code and Link

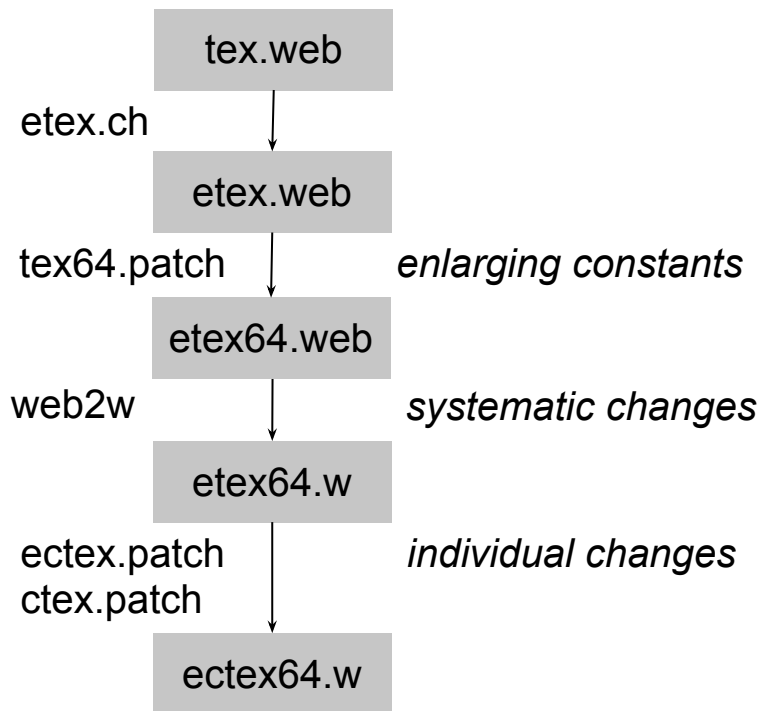
The extended basis: ectex.w



- Extended features of ϵ -T_EX

How ?

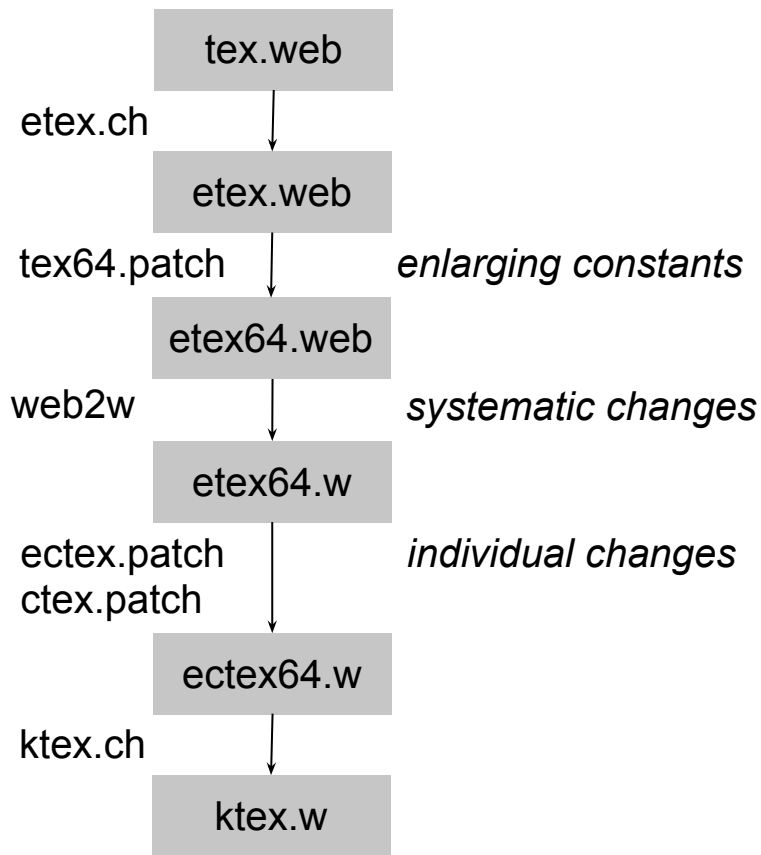
- Start with tex.web
- Apply web change files, e.g. ϵ -T_EX
- Apply patch files
- Convert web file to cweb file
 - run web2w
 - apply patch files
- Apply cweb change files, e.g. for kT_EX
- For T_EX related projects
 - Extract code and Link



- Large constants, e.g. 64 bit memory words
- Extended features of ϵ -T_EX

How ?

- Start with tex.web
- Apply web change files, e.g. ϵ -T_EX
- Apply patch files
- Convert web file to cweb file
 - run web2w
 - apply patch files
- Apply cweb change files, e.g. for kT_EX
- For T_EX related projects
 - Extract code and Link



- T_EX Live compatible T_EX engine

What?

systematic changes: Macros

- web2w translation of WEB macros to cweb macros

WEB:

```
@d define(#)==if global then geq_define(#)+else eq_define(#)
```

...

```
@<Assignments@>=
```

```
set_font: define(cur_font_loc, data, cur_chr);
```

cweb:

```
@d define(A, B, C) if (global) geq_define(A, B, C);+else eq_define(A, B, C)
```

What?

systematic changes: Macros

- New in web2w version 1.0

WEB:

```
@d char_info_end(##) == #].qqqq
```

```
@d char_info(##) == font_info[char_base[#]+char_info_end
```

```
...
```

```
    cur_h := cur_h + char_width(f)(char_info(f)(c));
```

cweb:

```
@d char_info(A, B) font_info[char_base[A]+B].qqqq
```

```
...
```

```
    cur_h = cur_h + char_width(f, char_info(f, c));
```

What?

individual changes: nonlocal goto

```
@<Error hand...@>=  
static void jump_out(void)  
- {@+goto end_of_TEX;  
+ {@+ close_files_and_terminate(); exit(0);  
}
```

What?

individual changes: cwebmac.tex has no \L

```
@<Types...@>=
```

```
typedef int scaled; /*this type is used for scaled integers*/
```

```
- typedef int32_t nonnegative_integer; /*$0\L x<2^{31}$*/
```

```
+ typedef int32_t nonnegative_integer; /*$0\le x<2^{31}$*/
```

```
typedef int8_t small_number; /*this type is self-explanatory*/
```


What?

individual changes: single character strings

@<Put each...@>=

- primitive('-', discretionary, 1);

+ primitive("-", discretionary, 1);

@!@:Single-character primitives -}{\quad\.{\\-}@>

What?

individual changes: use / as directory separator

```
- @d TEX_area "TeXinputs:"  
+ @d TEX_area "TeXinputs/"  
  @.TeXinputs@>  
- @d TEX_font_area "TeXfonts:"  
+ @d TEX_font_area "TeXfonts/"  
  @.TeXfonts@>
```