# H2L CT3370 - A HTML to LaTeX translator

Eddy L O Jansson <eddy@klopper.net>

March 23, 2006

## 1 Problem description

A program was to be written for translating a subset of HTML into the source language of the type-setting system LaTeX.

## 2 Assumptions

The software assumes that the input is reasonably *well formed*, specifically with regard to opening tags being properly matched with closing tags. The software does not handle advanced structures, specifically not nested tables. The subset of HTML supported most closely matches HTML4 strict.

## 3 Design and Use

I have elected to implement my software in `perl`, using a simple hand-written table-based *parser* for the main document structure, a separate *"tag-parser"* for parsing tag attributes, comments and DTDs, both of which use a *lexing* module which in turn builds on a very simple *scanner*.

All these modules come together in an utterly simplistic driver, `h2l.pl` which expects a source document on `stdin`, and will output the translated document onto `stdout`.

## 4 Implementation

The main code of interest is the `TranslatorTD` module, which makes up the bulk of the `SLOC`. It is an object which when built takes as input an instance of `Lexer`. Using the lexer as a source, the translator then uses a rather large parse table to match tags and execute parsing routines that generate outout.

The parse table is a hash table, consisting of a set of states, which each feature a set of tags tha are allowed in that state, and for each tag, a reference to the output subroutine to call, and an optional state to move into. The states also features a variety of flags used to guide the parser. For instance, only states flagged with `ALLOW_BAREWORDS` will pass the content *between* tags to the output, so it is set for the tags `P` and `PRE` for instance, but not for `HTML`, `HEAD` or even `BODY`.

### 4.1 Features supported

All the required document tags are supported. In addition, my software supports hyperlinks, images and (simple) tables, and it will pick up author name(s) from meta-tags and apply to the output document.

| **TAG/FEATURE** | **Support** |
|---|---|
| Comments | Passed on to output document. Supports tags in comments. |
| DTD | Passed on to output as a comment. |
| A | Supported (for non-local links). |
| B | Supported. |
| BODY | Supported (is a container, does not support bare words!) |
| BR | Supported. |
| CENTER | Supported. |
| EM | Supported. |
| HEAD | Parsed for `TITLE` and `META`-tags. |
| HTML | Supported. |
| I | Supported. |
| UL | Supported (with nesting). |
| LI | Supported. |
| P | Supported. |
| PRE | Supported. |
| SUP | Supported. |
| SUB | Supported. |
| TITLE | Supported. |
| IMG | Supported, though no rewriting. |
| TABLE | Supported. |
| TR | Supported (no attributes used). |
| TD | Supported (no attributes used). |
| TT | Supported. |

## 5 Conclusion

My implementation is slow and very rough in parts, but it works reasonably well and implements all the requested *and* optional features of the original specification. This document was translated from HTML using my software.