

一体型アウトラインプロセッサ"WaferView"

大谷 紀子 研究室
0031069 神谷 徹

指導教員 承認印

1. 研究の背景と目的

パソコンで文章を執筆・編集する利点は、内容を自在に変更できることである。パソコンにおける文章編集といえば、ワードプロセッサ・ソフトウェア、いわゆる「ワープロソフト」を使うのが一般的であろう。「モニタ上での見た目そのままに印刷できる」ことがワープロソフトの利点である。しかしこれは同時に弱点でもある。紙に印刷した文章の視覚イメージに捕らわれてしまうからだ。

印刷される文章のイメージから離れて、パソコンで文章を執筆・編集することの利便性に主眼をおいたソフトウェアに「テキストエディタ」がある。テキストエディタには、ワープロソフトが持つ視覚的効果を設定する機能、例えば文字サイズ・色の指定、が備わってない。好きなところで行を折り返せたり、改行記号やタブ記号を表示させたりできる。紙を再現することとは別の、パソコンでの文章編集に特有の機能のみを備えている。

本研究では、パソコンでの文章編集に効果的な機能の探求を目的とし、文章を階層表示しながら編集できるソフトウェア"WaferView"を構築する。

WaferView は、ワープロやテキストエディタのように文章が最初から最後まで全て表示するのではなく、文章を任意に区切り、自在に表示・非表示できる。区切られた文章を、パソコンがディレクトリをツリー表示するように、階層表示しながら編集できる。

2. アウトラインプロセッサとは

文章を階層表示しながら編集できるソフトウェアに「アウトラインプロセッサ」がある。

アウトラインプロセッサは、「2 ペイン型」と「一体型」に大別される。この呼称は統一されておらず、ソフトウェアコミュニティごとに異なるものである。

2 ペイン型とは、Windows のファイルエクスプローラの右側のペイン、フォルダの中身が表示される枠がエディタになっているようなものである。左側のペインに文章の概要（アウトライン）がツリー表示され、右側の枠に本文が表示される。

一体型とは、ツリー構造と本文が一体になったものである。

アウトラインプロセッサ一般についての解説は、「アウトラインプロセッサで書け！」(http://petraining.s21.xrea.com/knowledge_tools/outline-processor.html) に詳しい。

文章を任意に区切り表示・非表示するという、本研究の目的に合致しているのは一体型である。

3. WaferView

WaferView は、Java アプリケーションである。SDK に付属しているサンプルプログラム"Notepad"を核にした。開発は、WindowsME 上の Java 2 SDK, Standard Edition, Version 1.4.2 で行い、Linux(VineLinux, SuseLinux)と Macintosh(OSX)での動作を確認した。

WaferView は以下のサイトからダウンロードできる。

<http://www.yc.musashi-tech.ac.jp/~g0031069/waferview/>

3.1. WaferView の階層構造の独自性

フリーで公開されているアウトラインプロセッサを以下に示す。

"NS アウトライナー" <http://www.nsd.co.jp/share/nsout/>

Windows 対応。

"ActaClassic" <http://www.a-sharp.com/acta/>

Macintosh 対応。

日本語化パッチが Iimori's Home Page(<http://www.bekkoame.ne.jp/~iimori/index.html>)で入手可能。

"Sol" <http://homepage1.nifty.com/knowledge-osmo/>

Windows 対応。

3つのアウトラインプロセッサでは、いずれも図 3-1 で示すように階層が表現されている。この構造では、階層の先頭に文章があり、階層はその中に階層を含むことができる。図 3-2 に示すように、フォルダのツリー表示と類似している

しかしこの方法では、文章を任意に区切ることができない。このため WaferView では、図 3-3 で示すように、階層の中に文章と階層を交互に並べることができるようにした。これで階層構造をより柔軟に表現できる。

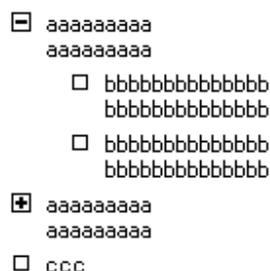


図 3-1 NS アウトライナーの階層構造

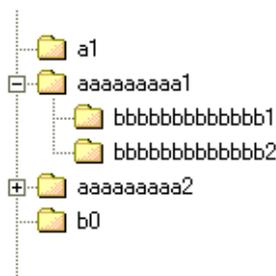


図 3-2 フォルダの階層構造



図 3-3 WaferView の階層構造

4. 評価実験

WaferView の有用性を確認するため、以下の評価実験を実施した。

実験 1.

被験者 研究室の三年生 3 人

内容 WaferView を使った作業とアンケートへの回答。

結果 被験者にアウトラインプロセッサを知っている者がいなかったこと、また、Java の API が組み合わせられることにより、各々のコンポーネントの正常な動作が思いもよらない不具合を生み出し、被験者に混乱を招いた。結果、基本的な機能を理解できるかどうかという論議に終始した。

実験 2.

被験者 4 年生 1 人

内容 被験者が WaferView を使用し、筆者が隣で作業を与えたり質問に答えたりする。

結果 テストは約 52 分間続いた。基本的な機能に関する質問をされてそれに回答するに終始した。

5. 結論

実験 1 のアンケート結果において、次も WaferView を使いたいと答えた学生はいなかった。

実験 2 の追跡調査でも、その後 WaferView はほとんど使われていなかった。

WaferView 独自の機能に関する評価以前に、既存の高機能を持つテキストエディタ・ワープロソフトなら持っている機能、例えば矩形選択・Undo・Redo・カラーカスタマイズ・キーカスタマイズ、が存在しなかったり貧弱であったりするために、使われなかったものとする。

現在、継続的に WaferView を使用しているユーザーは私だけである。この概要も WaferView によって執筆した。本研究が目的とした機能が、本当に優れたものかどうかの結論は出せなかった。

参考文献

- [1] 結城浩, 『Java 言語で学ぶデザインパターン入門』, ソフトバンクパブリッシング, 2001.
- [2] ルディ・ラッカー, 『思考の工具箱』, 工作舎, 1993.