

MOSDENVの実装とその評価
(保守を考慮したソフトウェア開発環境)

5J-3

おさだかずひさ たなかひでひこ もとおかとおる
長田 和久, 田中 英彦, 元岡 達
(東京大学工学部)

1. はじめに

ソフトウェアの生産性を上げるために相対的にも絶対的にも安くなってきたハードウェア資源をふんだんに使うことが考えられている。しかしハードウェアも設計やそれ以前を考慮にいれればソフトウェアとみることもできよう。ソフトウェアコストの当面の問題は、「保守」コストである。なかでもドキュメンテーションは、やっかいで取り扱にくい問題である。(最近のワードプロセッサの普及で少し改善のきざしがみられるようである。)

従来、ソースコードの中にコメントとして種々の情報を記述しておく方法がとられてきた。ソースプログラムが唯一信用できるドキュメントであるというのがその理由である。一方、ワークステーションやマルチウィンドウエディタにみられる開発環境の整備により、ソースプログラム以外のドキュメント群も利用できるようになりつつある。これらをいかした MOSDENV (Maintenance Oriented Software Development ENVironment) [1] は、予防保守をしつつ効率的なソフトウェア開発を支援する環境である。

今回、実験システムとしてトップダウン設計・モジュール分割による(木構造を基本とする)ドキュメンテーション方式とマルチウィンドウエディタを対応させることをねらったプロトタイプを作成しその評価を行ったので報告する。

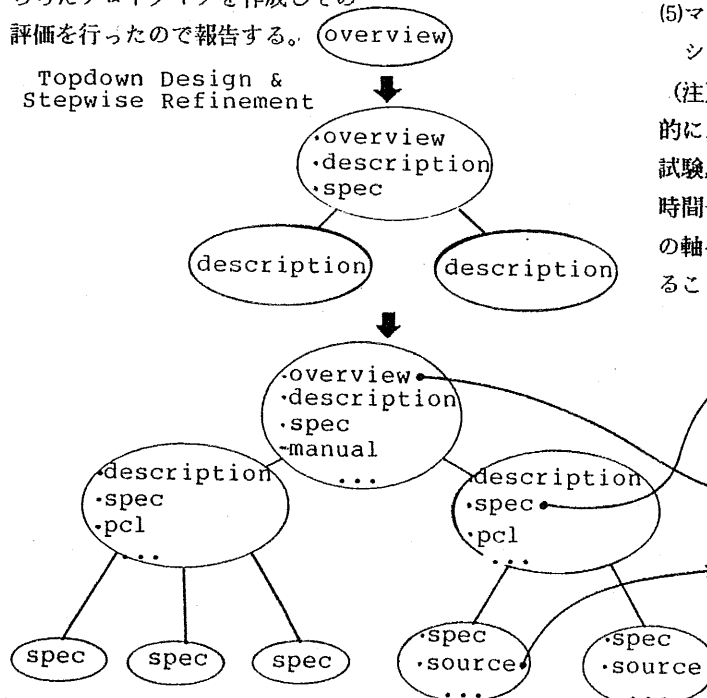


Fig. 1 Document & Multiwindow

2. MOSDENVの概要と実装方針

ソフトウェア開発過程を「螺旋型モデル」(注)で表すと図5のようになる。MOSDENV が対象とするソフトウェアは、そのプロトタイプが数多く作られたり、改版が何度も引き続いて繰り返されたりするようなものである。

図3に示すようにMOSDENV は、「ソフトウェア開発方法論」、「保守用データベース」、「ユーザーフレンドリな環境」という3つの側面から「環境」をとらえたものを基本方針としている。これらより、その中味は、

- ・ドキュメンテーション方式
- ・プロンプティングシステム
- ・改版支援ツール

である。

MOSDENVの特徴(実装方針)は、以下の通り。

- (1)初期開発時には、プロンプティングシステムと対話的に操作することにより階層設計・段階的詳細化を支援する。
- (2)トップダウン設計・モジュール分割に自然に対応したドキュメンテーション方式。
- (3)各々のモジュールのドキュメント間に関係を決めておくことにより改版支援を行う。
- (4)既存のツール群の積極的活用。

Unix, Emacs, etc.

- (5)マルチウィンドウとドキュメンテーションの対応によるシステム。

(注)螺旋型モデル表現は、ソフトウェア開発過程を視覚的に表す方法である。「要求分析, 仕様化, 設計, 実装, 試験, 改版, ...」と繰り返されるソフトウェア開発の時間-コストのグラフにおいて時間軸を中心としてコストの軸を回転させたもの。また、各版の各フェイズをそろえることにより相互関係を示しやすい [2]。

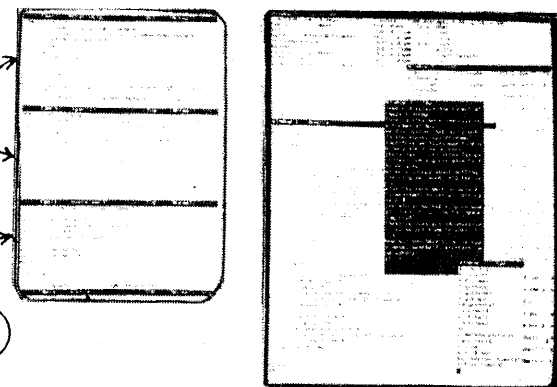


Fig. 2 Photo of Multi-window Terminal

3. 実験システムとその評価

今回の実験システムでは、Unix TM とマルチウィンドウエディタ (Emacs と当研究室のウィンドウシステム) をもとにハードウェア構成を図4のようにした。

ドキュメンテーション方式については、作成されたドキュメント群を管理し、適当な側面に取り出すツール (フィルタ) を用意した。ドキュメンテーション方式は、プログラムのコンパイル単位を1モジュールとして、“overview”，“description”，“spec”，“source”と段階的詳細化の手法をとっている (図1)。ドキュメントの記述量はソースプログラムのみと比べて約2倍であるが、その分プログラムの readability, understandability は向上した。

ドキュメント間に図6のような相互関係を設定し、それを保ちつつ編集ができるという点で広い意味での構文エディタといえよう。これは、プログラムの初期開発時においてはプロンプティングシステム、改版時においては改版支援ツールとなる。

個々のドキュメントを各々のウィンドウに割り当て、図6の関係をもつドキュメント群の作成支援を行うのがプロンプティングシステムである。ウィンドウ間の移動はマウスまたはウィンドウ番号を用いて指示する。マルチウィンドウシステムは、ウィンドウが重なり合うものと重なり合わないものを用意して現在比較を行っている。(図4)

改版支援ツールは、図6の関係を保つように半強制的に作成の順序関係を規定する。そのために現在のままでは使用上やや煩わしい点が残る (人間が一貫性の確認をするため)。しかし、改版時においては、必ずそれに関連するドキュメントが up-to-date されることになる。

今後さらにツール群の整備が進めば、階層化されたドキュメント、木構造ファイル、そしてマルチウィンドウエディタとの組合せでソフトウェア開発時におけるユーザーインターフェースの一層の向上が期待できる。

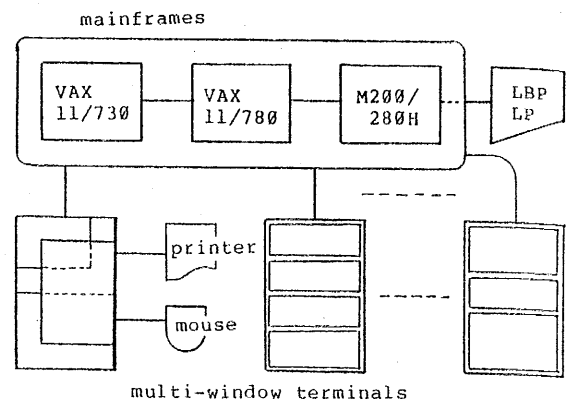


Fig. 4 Machines Configuration of MOSDENV

4. おわりに

ドキュメントのオンライン化やデータベース化にともないソフトウェア開発の方法論も環境も変わりつつある。

MOSDENVのねらいは、開発時に保守を強く意識してソフトウェア開発を行う環境である。今後、ドキュメンテーション以外の問題についても検討を行う予定である。

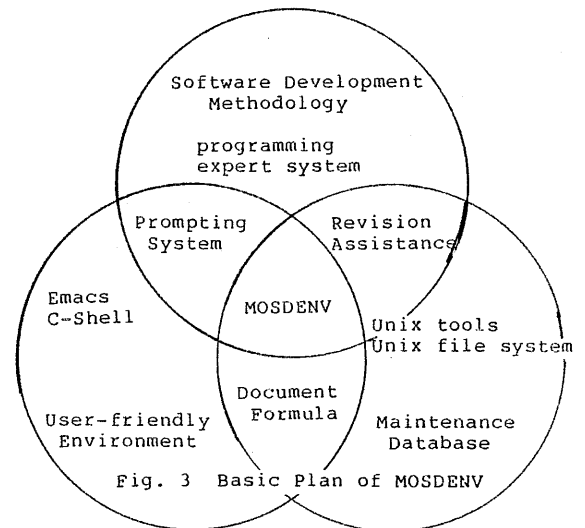


Fig. 3 Basic Plan of MOSDENV

Fig. 6 Dependency Graph

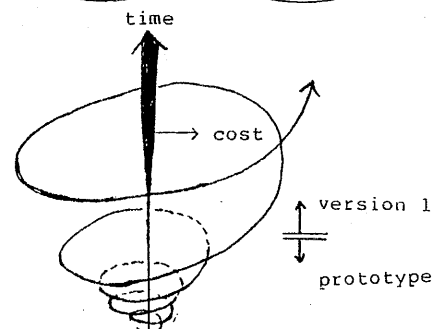
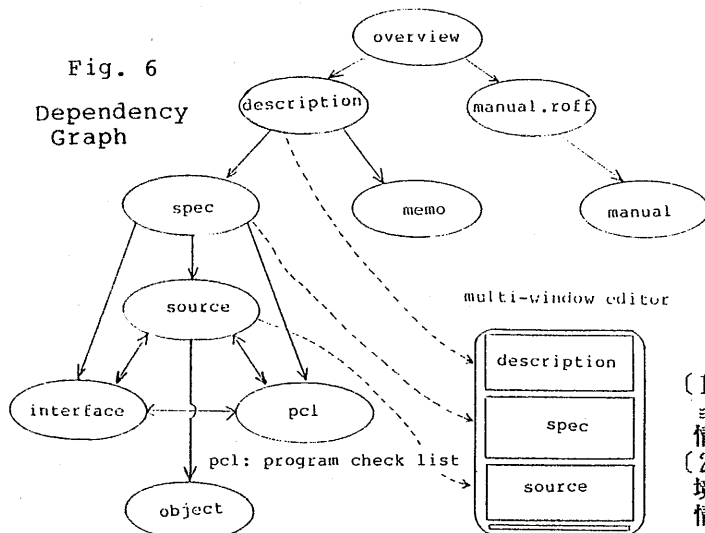


Fig. 5 Spiral Model Representation

参考文献

- (1) 長田 他: 「MOSDENV におけるドキュメンテーション方式」 情報処理学会第27回全国大会, 1983, pp.619-620
- (2) 長田 他: 「保守を考慮したソフトウェア開発環境: MOSDENV」 情報処理学会ソフトウェア工学研究会 34-4, 1984