

仕様書エディタ：SPEC

堀川博史，古川栄司，高野 彰，春原 猛
(三菱電機(株) 情報電子研究所)

1. はじめに

ソフトウェア生産においては、プログラムコードだけでなく、外部仕様書、内部仕様書、検査仕様書などの多くの仕様書が作成されている。これらの仕様書は大規模なソフトウェアを複数人で作成する時に特に有効になる。企業におけるソフトウェア製造(リリースまで)に要する費用の半分近くは設計とその生産物である仕様書の作成に費やしている。しかるに、プログラムコード作成を支援するエディタは存在するが、仕様書の作成を支援するエディタは通常のワードプロセッサが存在する程度である。ここでは、仕様書作成作業を軽減する目的で、あらかじめ外から与えられた仕様書の記述規則に従って、自動記述、ガイドンス、自動複写、矛盾検出等を行う仕様書エディタ：SPEC (SPECifications editor の略) [1][2][3] について報告する。

2. 対象としている仕様書

ソフトウェアの開発工程を計画段階、設計段階、製造段階、試験段階、運用・保守段階の5段階に分けて捉える。設計段階を更に、システム仕様設計、外部仕様設計、内部仕様設計の3段階に区分する。それぞれの設計段階ではシステム仕様書、外部仕様書、内部仕様書の文書が作成される。システム設計では実現するシステムの仕様を明確にし、機能ブロックに分割し、ハードウェア、ソフトウェアの分担を明確にし、機能ブロック構造を決定する。外部設計では各機能ブロックの仕様を定義し、それぞれのプログラム構造を決定する。内部設計ではプログラムの仕様を定義し、それぞれのモジュール構造を決め、各モジュールの仕様を定義する。各仕様書には記述対象の目的、位置付け、機能、入出力情報、使用法、操作員インタフェース、エラー処理等の仕様を記述しなければならない。

仕様書は通常のオフィス文書と比較して次のちがいを持つ。

- (1) 記述量が多い。
- (2) 記述規則が多い。
- (3) 他の仕様書(特に前段階で作成される仕様書)との間の関係付けが行われる。
- (4) 一冊の仕様書の中での記述順序が想定できない(特に、懸案事項を残して先に進むことが多い)。その結果、(懸案事項の)修正が多く発生する。
- (5) 文章の代りに、図・表で表現することが多い。
- (6) 定形用紙への記入が多いが、作成されるソフトウェアの適用分野の違いなどにより、それを完全に統一できない。

3. SPECの機能

仕様書と通常のオフィス文書の上述したような違いを考えると、通常のワードプロセッサだけでは十分な支援を行うことができない。仕様書作成のために次の機能を実現することにした。

- (a) 仕様書の標準化と記述量の軽減のために定形的な部分を自動記述する機能。

- (b) 個人差の解消と作成時間の短縮のためのガイドンス機能。
- (c) 他の仕様書との関係付けと記述量の軽減のための自動複写機能。
- (d) 同一記述の記述ミス、修正誤りを防ぐための矛盾検出機能。
- (e) 記述もれを防ぐための未完成部管理機能。
- (f) 図・表の編集機能。

これらの機能の具体的仕様がツールの中に作り付けにすると、作成対象となる仕様書の形式が完全に統一されてしまう。しかし、作成されるソフトウェアの適用分野の違いなどにより、それを完全には統一できない。そこで(a)~(d)の機能は実行のタイミング、実行時のパラメータ指定を記述規則として外部から与える形で実現した。その結果、ツールが扱える仕様書の形式は自由度を持つ。

(1) 自動記述機能

仕様書中の記述項目は通常決まっている。すなわち、仕様書の章題、節題は決まっている。そこで、章立てに関する情報をあらかじめ与えておくことにより、仕様書を新規に作成した時に、SPECが章題だけを持った仕様書を自動的に生成するという方式を用いた。その結果、SPECを用いて作成された仕様書は章題を持つことが保証される。例えば、システム仕様書の場合は「1章 概要、2章 適用範囲、... 8章 共通環境」という章立てとなる。

また、次のものの記述は使用者の負担を軽減するため、SPECが自動記述する。

(a) 仕様書の中の定形的な文章

例えば、エラー処理では「システムは検出できるエラーをすべて処理し、エラーメッセージを表示する。...」という文章から始まる。これらの定形的な文章は自動記述される。

(b) 定形用紙

モジュール記述シート、エラーメッセージ一覧シートなどの枠が自動記述される。

(2) ガイドンス機能

仕様書に何を記述するかは、SPECがガイドする(例:「システム名を入力して下さい。」。ガイドメッセージはあらかじめ与えておくことができる。

(3) 自動複写機能

仕様書を記述する時に同じ記述を繰り返さなければならないことがある。例えば、「機能ブロック構成」の中の「機能ブロック構成図」にも「各機能ブロックの説明」にも同一の機能ブロック名が記述されなければならない。同じ記述を繰り返すことは使用者の負担になるので、SPECはそれらを自動複写する(図3.1)。

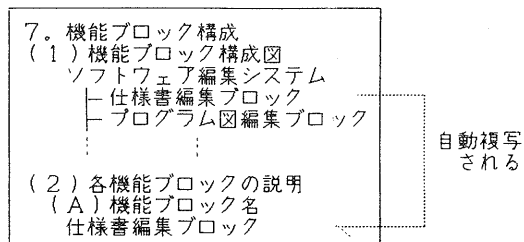


図3.1 自動複写機能

1参照)。

システム仕様書の中には、そのシステムで使用される共通ソフトウェア群の記述があり、それらは下位の仕様書への自動複写の対象となる。しかし、システム中のあるプログラムがすべての共通ソフトウェアを使用するとは限らない。そのようなプログラムに対する仕様書の作成状況を想定し、共通ソフトウェア等の自動複写時に、個々の対象を複写するか否かを複写時に使用者が指定することもできる。

(4) 矛盾検出機能

自動複写が行われた場合は、複写先と複写元の間で記述内容が同一でなければならぬという表明が自動的に設定される。同一でない時にはSPECが表明違反メッセージを表示する。これにより、例えば、「機能ブロック図」の修正忘れや修正ミスを防ぐことができる。

また、記述が制限に従っているかどうかの検査が行われる。違反時にはメッセージが表示される。記述の制限には次のものがある。

(a) 文字列長

例えば、モジュール名は6文字以下でなければならないという制限。

(b) 頭文字

例えば、モジュール名の頭文字は「Z」以外の文字(「Z」は共通モジュールを示す)でなければならないという制限。

(c) 文字列

例えば、アクセスモードは「IN」、「OUT」、「IN OUT」のどれかの文字列でなければならないという制限。

(5) 未完成部管理機能

記述内容を持たない部分と、懸案事項であることを指示するコマンドにより懸案事項と指示された部分は未完成部表示コマンドを用いて表示することができる。

(6) その他の機能

その他、文章、図形を編集するための機能として次のものを提供している：各種ページ表示機能。ページの生成・削除機能。日本語文書の生成・削除・複写・移動・挿入・置換・検索機能。直線・点線・矢線・円弧・四角・円の生成・削除・複写・拡大・縮小機能。他仕様書の参照機能。印刷機能。

4. SPECの文書アーキテクチャ

文書は文書内容とそれを特徴付ける特定論理構造と特定レイアウト構造および論理・レイアウト関係から構成される(ここで用いる用語は参考文献[4]に基づく。ただし、SPECは文献[4]に準拠しているわけではない)。特定論理構造と特定レイアウト構造は木構造をしており、木のノードをオブジェクトと呼ぶ。木構造の枝はオブジェクトの一つ下のオブジェクトへの分割を示す。特定レイアウト構造と特定論理構造を図4.1に示す。なおブロックとは特定レイアウト構造の基本オブジェクトであり、その側辺がページの側辺と平行である長方形の領域である。ブロックは文書内容のコンテナである。

SPECに仕様書の記述規則を与えるには汎用論理定義を用いる。汎用論理定義には文書間で共通の論理構造・文書内容を定義しておく。汎用論理定義は属性付きBNF風言語で定義する。図4.3に汎用論理定義の構文を示す。SPEC

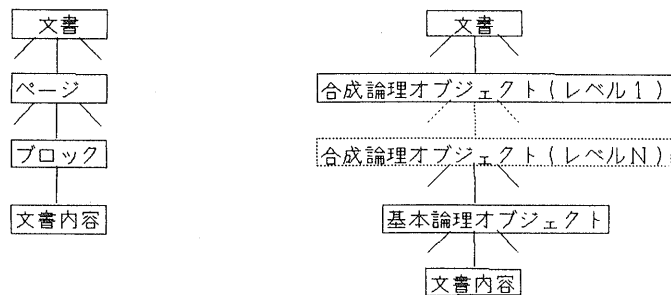
汎用論理定義を解析し、汎用論理定義のBNF部で定義されている構造と等価な木構造を持つ特定論理構造を生成する。その際、特定論理構造の各基本論理オブジェクトには汎用論理定義の属性部の情報が書き込まれる。SPECは特定論理構造の各基本論理オブジェクトに書かれている属性部の情報に基づいて、自動複写や入力促進の表示等を行う。そして、得られた文書内容に対応する基本論理オブジェクトに關係付ける。汎用論理定義の例を図4.4に、変換された特定論理構造を図4.5に示す。

汎用論理定義が繰り返しを持つ時、対応する特定論理構造の初期状態は繰り返し要素の合成論理オブジェクトの情報を持った（子供を持たない）繰り返しオブジェクト（図4.5の「*」）である。図4.5においてモジュール仕様が入力（複写）されると、その都度、「*」の下にモジュール仕様が1単位生成される。

BNF部には構造情報の他に定形用紙を生成する「&FORMAT」、基本論理オブジェクト間の相対位置關係（改行）や、基本論理オブジェクトの絶対位置（行とカラムの位置指定「&MP」等）の指定を行うレイアウト関数が記述できる。

属性部に直接文字列を書いた場合、この文字列はそのままだ文書内容となる（図4.4の「@章題」参照）。

ガイドンスを行う属性関数（\$PROMPT等）は引数として与えられた文字列をガイドメッセージとする。この属性関数の実パラメータには、文字列だけでなく、文字列と同時に、ある基本論理オブジェクトの指定を書くことができる。この場合、指定された基本論理オブジェクトに關係付けられている文書内容と与えられた文字列とでガイドメッセージを合成する。図4.4の@機能には「@2



(A) 特定レイアウト構造

(B) 特定論理構造

図4.1 特定レイアウト構造と特定論理構造

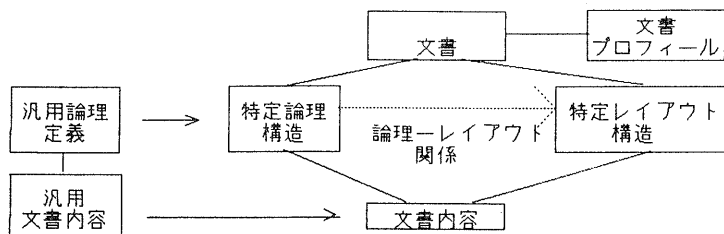


図4.2 文書アーキテクチャモデル

= \$PROMPT (モジュール仕様。... の機能)」が書かれており、基本論理オブジェクトとして「@モジュール名」が指定されている。「@モジュール名」に文書内容「MAEN」が関連していると、「@機能」の文書内容を得るためのガイドメッセージは「MAENの機能を入力して下さい。」となる。これにより、先に入力された文書内容に応じた柔軟なガイダンスを実現している。

複写を行う属性関数のうち「\$COPY」はそこで指定されたパラメータ(パス名)から一意に決定できる基本論理オブジェクトに関連する文書内容を複写元とする。「繰り返し」により、指定されたパラメータに対応する基本論理オブジ

```

形式定義 ::= ( 生成規則 ) + ;
生成規則 ::= BNF部 [ 属性部 ] * ;
BNF部 ::= 合成論理オブジェクト * ;
属性部 ::= / * 属性規則 * / ;
式 ::= ( 合成論理オブジェクト | 繰り返し |
      省略可能 | レイアウト関数 ) * 基本論理オブジェクト * ;
合成論理オブジェクト ::= 文字列 ;
基本論理オブジェクト ::= @ 文字列 ;
繰り返し ::= ( 合成論理オブジェクト * ) * ;
省略可能 ::= [ 基本論理オブジェクト ] * ;
属性規則 ::= @ 数字 = 属性 | 属性 * ;
属性 ::= 文字列 | 属性関数 ;

```

図4.3 汎用論理定義の構文

```

内部仕様書 = 概要 外部インタフェース 処理方式
              モジュール構成 共通環境 オーバレイ構造 ;
              |
              |
              |
モジュール構成 = @ 章題 モジュール一覧 (モジュール仕様) * ;
                  / * @ 1 = 4. モジュール構成 * /
              |
              |
              |
モジュール仕様 = &FORMAT (モジュール記述シート)
                  &MP (3, 1) @モジュール名
                  &MP (5, 1) @機能
              |
              |
              |
              / * @ 1 = $COPYO (モジュール構成, モジュール一覧,
                  @モジュール名) ;
                  @ 2 = $PROMPT (モジュール仕様, @モジュール名
                  の機能) ;
              |
              |
              |

```

図4.4 汎用論理定義の例

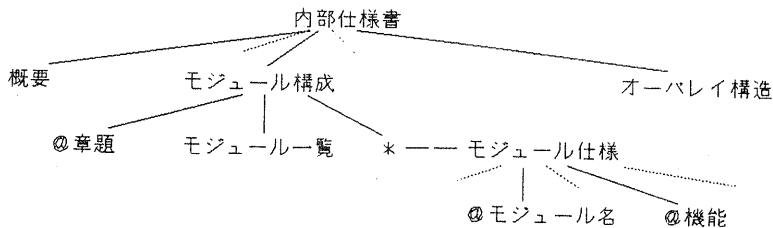


図4.5 特定論理構造の例

エクトが複数個存在するものの複写には「\$COPY0」や「\$COPY5」などを用いる。これらは「繰り返し」の中で用い、他の「繰り返し」内の同一名の基本論理オブジェクトの記述をあるだけ順番に複写したり（\$COPY0）、選択的に複写したりする（\$COPY5）。また、他の仕様書からの複写を行うものとして\$ECOPY等がある。

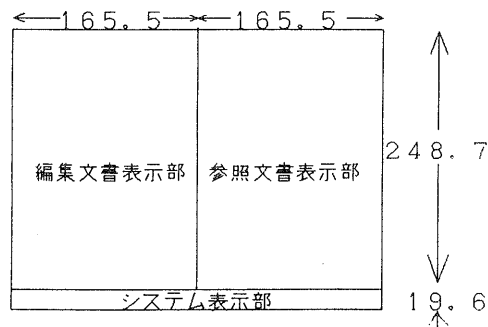
5. SPECの実現

(1) ハードウェア

SPECはMELCOM COSMO 900Ⅱ上でグラフィック端末（20インチの画面を持ったラスタースキャン方式）を用いて動作するエディタである。作成された仕様書はハードコピー装置およびレーザービームプリンタで印刷することができる。

(2) 画面

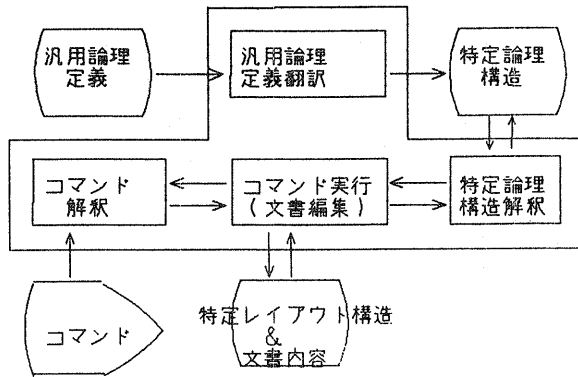
SPECの画面は編集文書を表示する編集文書表示部と、参照文書を表示する参照文書表示部と操作上のガイダンスや入力された文字を表示するシステム表示部の3個の領域から構成されている。編集文書表示部と参照文書表示部は文書を表示するための領域で、34文字×51行の文面：A4サイズ1ページ分（上下左右2cmずつ除いた大きさ）を一度に表示する。システム表示部は4行からなり、ページ表示、システムメッセージとガイドメッセージの表示、かな（ローマ字）漢字変換の表示、ユーザ入力のエコーバックに使用される（図5.1参照）。



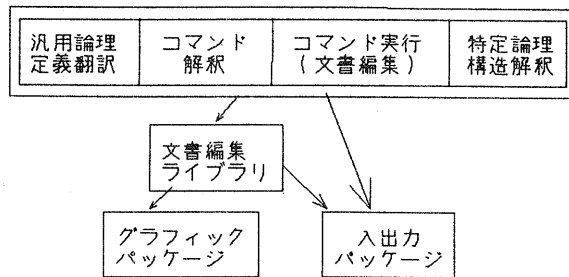
(単位ミリ)
図5.1 画面レイアウト

(3) ソフトウェア

SPECのソフトウェア構成を図5.2に示す。(A)はユーザから見たSPECのソフトウェア構成図であり、(B)は内部構成図である。汎用論理定義翻訳は汎用論理定義から特定論理構造を生成する。コマンド解釈は使用者のコマンドを解釈し、コマンド実行部は仕様書の編集を行う。特定論理構造解釈は論理・レイアウト関係を管理すると共に、特定論理構造中の（自動複写等の）属性の解釈実行を行う。文書編集ライブラリは特定レイアウト構造と文書内容の扱いを容易にする。具体的には、ページの生成・削除・表示、フロッグの生成・削除・移動などの機能と共にかな（ローマ字）漢字変換機能も提供している。可搬性を考慮し、OSまわりの入出力機能は入出力パッケージが提供する。



(A) 正面図



(B) 平面図

図5.2 SPECのソフトウェア構成

SPECは(論理)構造を扱うエディタであるが、極力、構造をユーザに見せないように実現しようとした。しかし、完全には構造を隠すことはできなかった。次の2点においてユーザは構造を意識せざるを得ない。

(a) 仕様書論理構造の考え方

仕様書で何を書かなければならないかは「仕様書作成マニュアル」で決められている。例えば、システム仕様書の「第1章 概要」には「システム名」、「目的」、「概略機能」、「特長」を書かなければならない。自動複写等を行うためには、例えば、「概略機能」が仕様書中のどの記述であるかを管理しておく必要がある。図5.3の場合、「仕様書とプログラム図と…の編集を行う」が「概略機能」

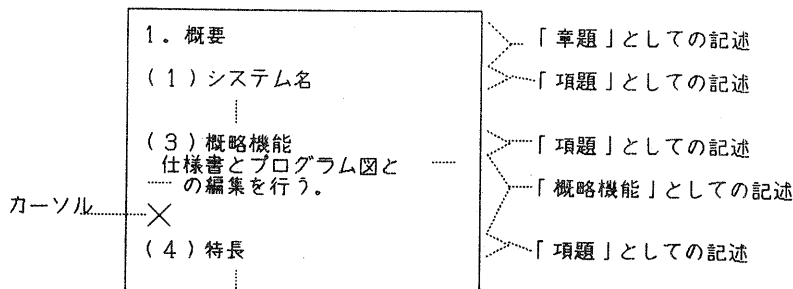


図5.3 仕様書論理構造の考え方

としての記述である。記述が変更された時には、変更後の結果は変更前の論理オブジェクトに属するものとして扱われる。記述が追加された時には、カーソル位置から追加される記述がどの基本論理オブジェクトに属するかが決定される。図5.3において、カーソル位置に追加する記述は「概略機能」として扱われる。

初期状態の仕様書には章題しかない。この場合、何を追加しても、すべて「章題」として扱われる。次の基本論理オブジェクトに記述したい時には、「記述内容完成コマンド」または「懸案事項であることを指示するコマンド」を入力し、「章題」に対する記述を明示的に終了しなければならない。

(b) 繰り返し構造の考え方

モジュール名のように、いくつ入力されるかわからないものに対しては、汎用論理定義翻訳時に対応する基本論理オブジェクトの個数を把握することは困難である。そのため、SPECは「繰り返し構造」を持つ。仕様書はプログラミング言語のように繰り返しの終わりを示す予約語がないため、繰り返しの終了検知には次の方法を採用している：SPECは繰り返し要素に与えられる文章や図形（ユーザが入力するもの、またはSPECが自動複写するもの）がない時に繰り返しの要素の生成を終了する（図5.4参照）。

繰り返し要素を削除する時は、通常の文章や図形の削除を用いる。また、繰り返し要素を追加する時は、カーソルを繰り返し部分に合わせ、「繰り返し要素追加コマンド」を入力すると、指定された位置の繰り返し要素の次に繰り返し要素を一単位作る。

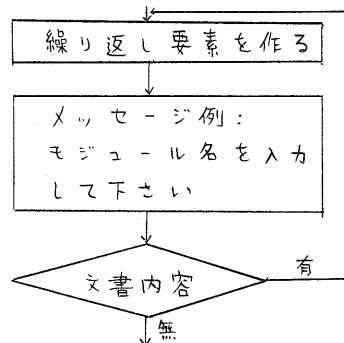


図5.4 繰り返し構造の処理

6. おわりに

SPECは現在、試使用評価中である〔5〕。汎用論理定義を外付けにしたことによる柔軟性、自動複写等によるユーザ入力作業の軽減、矛盾検出による信頼性の向上などが有効な機能とみなせるものの、応答速度の遅さ、定形用紙等の編集機能の弱さなどが問題となっている。また、SPECの持つ仕様書中の情報を他のソフトウェア生産支援ツールと共有化していく必要がある。

参考文献

- [1] 高野, 大川, 春原. 「仕様書作成支援ツール」, 27回情処全大, 3B-1.
- [2] 古川, 堀川, 高野, 春原. 「仕様書エディタの機能について」, 28回情処全大, 7T-6.
- [3] 古川, 堀川, 高野, 春原. 「仕様書エディタ-仕様書の形式定義」, 30回情処全大, 4T-1.
- [4] ISO. 「Information Processing Text Preparation and Interchange - Text Structure Part 2: Office Document Architecture」, DP8613/2.
- [5] 堀川, 古川, 高野, 春原, 山本. 「仕様書エディタの適用と評価」, 30回情処全大, 4T-2.