

マニュアル自動変換ツール「ドキュメントシステム」の紹介

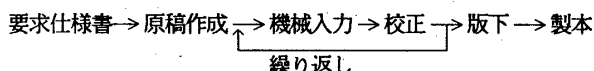
4S-9

小林広美¹、島雅人²、弘中康文²、植木京子²

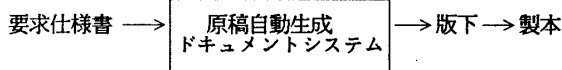
日本電気(株)、日本電気テレコムシステム(株)

1. はじめに

我々は量産商品であるデジタルキーテレホンのソフトウェアの開発及び商品に添付するマニュアル作成の支援を行っている。我々が作成するマニュアルは主に工事段階で専門家が使用するテクニカルマニュアルである。マニュアルを作成する工程は以下のとおりである。



原稿作成から版下までの工程の間に印刷業者が介入し、仕様の変更、原稿のミス等による機械入力、校正回数が多くなる。そのため版下完成までに多くの日数と経費がかかっていた。そこで、マニュアルの原稿を要求仕様書からデータ抜粋してできないか、また、自動生成した原稿をそのまま版下にできないかという事に着目した。そしてマニュアル原稿を自動生成するツール(以下、ドキュメントシステムという)開発・導入した。その結果、マニュアル作成の工数を大幅に削減することが出来たので報告する。



2. ドキュメントシステム概要

ドキュメントシステムはソフトウェア要求仕様書ファイルを入力し、ユーザーズマニュアル原稿を自動生成するシステムである。PC9801上のワードプロセッサで作成された要求仕様書からマニュアルに必要なデータを取りだし、そのデータにA4紙面上に配置するための座標データ及びマニュアルで使用するグラフィックデータを付加しその結果をファイルに出力する。最後に出力されたファイルを印刷するためにワークステーション上のDTP (Desk Top Publishing) アプリケーションを使用しマニュアル原稿を印刷する。DTPは文字の級数、字体、図形、などを多数サポートし印刷精度が高いためマニュアル原稿印刷用として使用する。図1にハードウェア構成図を示す。

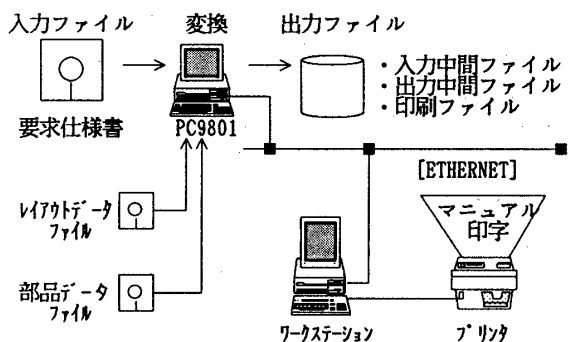


図1. ハードウェア構成

3. ソフトウェア構成

ドキュメントシステムは3つのタスク構成になっている。そのタスク構成図を図2に示す。

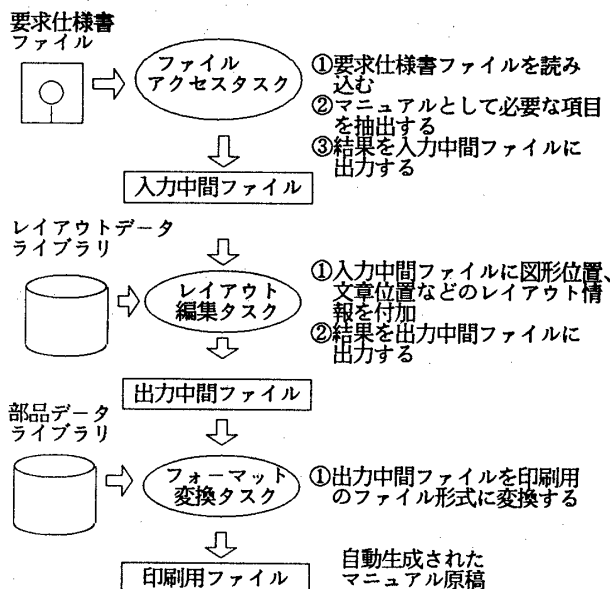


図2. タスク構成図

各タスクは独立しているため、要求仕様書のフォーマット変更、マニュアル構成の変更、印刷用のDTPソフト変更があった場合他のタスクに影響することなく変更することが可能である。要求仕様書フォーマット及び、各タスクの詳細について以下に記述する。

3. 1 要求仕様書フォーマット

要求仕様書フォーマットはあらかじめデータを取り出すためのキーワードを付加しておく。フォーマット例を図3に示す。キーワードは「[]」に囲まれている文字列で(以下、「識別子」という)その直後の「{ }」内にマニュアルに必要なデータを記述する。マニュアルに必要でないデータは「{ }」外に記述する。

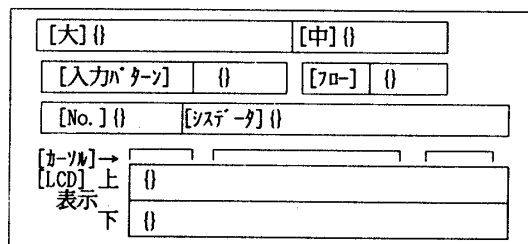


図3. 要求仕様書フォーマット

要求仕様書に実際に記述した例を図4に示す。

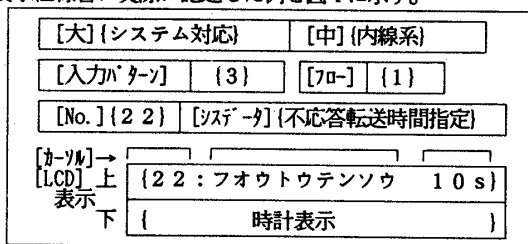


図4. 要求仕様書

Introduction of the Automatic Translator System for Users Manual "Document System"

Hiroki Kobayashi, Masato Shima, Yasufumi Hironaka, Kyoko Ueki
1, NEC Corporation 2, NEC Telecom System, Ltd.

3.2 ファイルアクセスタスク

ファイルアクセスタスクは「[識別子]」毎に「{}」で囲まれたデータを抜き出し入力中間ファイルに出力する。1つの「[識別子]」の後に複数「{}」があれば全てその「[識別子]」のデータとして取り出す。図5に入力中間ファイルを示す。中間出力ファイルはレイアウト編集タスクとのインタフェースとなっている。

```
begin_system(
  [大] {システム対応} ← [ ]で囲まれているのが識別子
  [中] {内線系}          ← [ ]で囲まれているのがマニュアルに記述するデータ
  [入力パターン] {3}
  [row] {1}
  [No.] {2 2}
  :
  :
  )
```

図5. 中間出力ファイル

3.3 レイアウト編集タスク

入力中間ファイルからデータを読み込み、それに対応するレイアウトデータライブラリ(*1)から取り出し2つのデータを合成した結果を中間出力タスクに出力する。レイアウト編集タスクの機能には以下の処理がある。

- ① 中間出力ファイルから「[識別子]」とデータを取り出す。
- ② レイアウトデータライブラリから「[識別子]」に対応するレイアウトデータを検索する。
- ③ レイアウトデータに指定してある範囲(x幅、y幅)からその範囲内に入る文字の大きさ(0.1mm単位)を計算し、最適な文字の大きさでデータを配置する。また、データをキーワードにして当てはまる部品IDを配置する。
- ④ 編集したデータを中間出力ファイルに出力する。

図6に出力中間ファイルを示す。このファイルはフォーマット変換タスクとのインタフェースになっている。
 (*1) レイアウトデータライブラリとは「[識別子]」毎に得たデータ及びマニュアルを構成するための部品ID番号で管理をA4紙面上に配置するための座標データ(x位置、y位置、x幅、y幅等)を設定したファイルである。マニュアルのフォーマットを変更する場合はこのレイアウトデータライブラリの変更のみで対応することが出来る。

図5にレイアウトデータライブラリのファイル構造を示す。

```
syd_name (GRPH(1, (36, 0, 110, 28, 0, 0)),
  [識別子] 部品ID x,y座標, x,y幅, x,y範囲
  TEXT((80, 0, 3, 39, 6, 104, 15, 0, 0), (CHAR1))
  ↑
  文字サイズ x,y座標 要求仕様書から
  行間         x,y幅   取り出したデータ
  行揃え      x,y範囲
```

図5. レイアウトデータライブラリ構造

```
beginpage(
  GRPH(1, (36, 0, 110, 28), 5, (8, 3, 10, 10), 3, (100, 150, 20, 20))
  ↑
  部品ID x座標, y座標, x幅, y幅
  TEXT((50, 0, 3, 30, 50, 100, 50), (システム対応), (50, 0, 3, 30)
  :
  :
  ↑
  文字サイズ 要求仕様書から
  行間         x,y座標, 取り出したデータ
  行揃え      x,y幅
  )endpage
```

図6. 出力中間ファイル

3.4 フォーマット変換タスク

出力中間ファイルと部品データライブラリから印刷用DTPソフトに対応したファイルフォーマットに変換する。フォーマット変換タスクの機能には以下の処理がある。

- ① テキストデータ(要求仕様書から抽出したデータ)の指定する文字の大きさ(0.1mm単位)から対応する文字の級数を求め、配置する。
- ② 部品データライブラリを検索し対応する部品データを取り出す。部品データ(*2)を出力中間ファイルに指定してある範囲(x座標、y座標、x幅、y幅)のグラフィックデータに変換し配置する。
- (*2) 部品データは可変長で出力中間ファイルに指定されているx,y座標位置及びx,y幅にあわせてA4紙面上に図形を描くことができる。
- ③ 変換したデータを印刷用ファイルに出力する。図6に部品データライブラリ構造、図7に印刷用ファイルを示す。

```
部品ID
{
  minx, 0; ← 部品を描ける最小x、y幅
  miny, 3;
  3(a1=5, a2=20, a3=0); ← 変数
  4:rect(x1+a1, y1, x2, y2) (517, 257, 2)
  :
  :
  ↑
  X、y座標、X、y幅
  x1, y1, x2, y2は出力中間ファイルから与えられる値
}
```

図7. 部品データライブラリ構造

```
beginPage
object, 014490, (10000, 14140), {
p(7536, 1392, 1632, 432, text, (7536, 1392, 1632, 432),
  attr(1, 1, h, MM, (648, 30, ¥Ng¥As¥Q14¥S0¥C0¥10
  ¥Fn¥Kn¥Pn 2 2)));
p(2352, 768, 4992, 720, text, (2352, 768, 4992, 720),
  attr(1, 1, h, MM, (984, 528, ¥Ng¥As¥Q28¥S0¥C0¥1
  0¥Fn¥kn¥Pn 不応答転送時間指定)));
p(5328, 5328, 3072, 192, text, (5328, 5328, 3072, 192), attr
  (1, 1, h, UL, (0, 84, ¥Ng¥As¥Q7¥S0¥C0¥10¥Fn¥Kn
  ¥Pn 2 2 : フォウトウテンソウ
  1 0 s)));
}
endpage
```

図8. 印刷用ファイル

4. 自動生成したマニュアル

以上の操作により出力した印刷用ファイルをワークステーション上に転送し、DTPソフトで印刷したマニュアルを図9に示す。

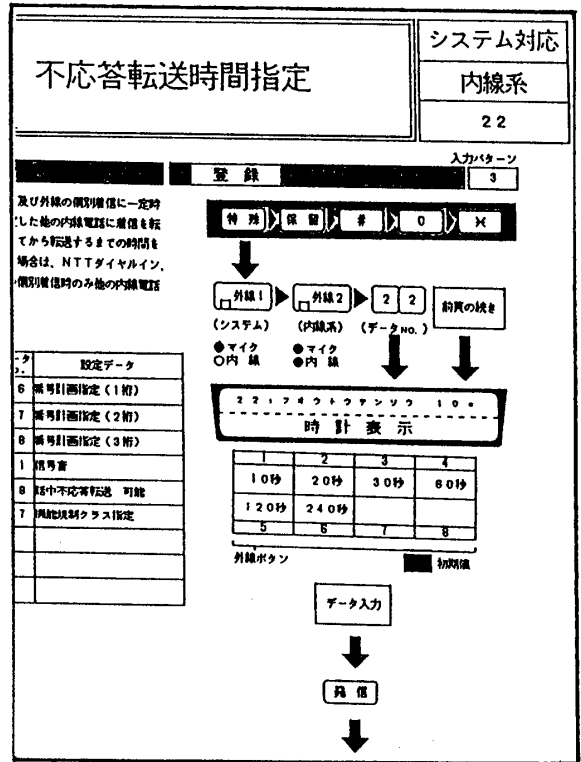


図9. 自動生成したマニュアル

5. おわりに

今回紹介ドキュメントシステムはプロトタイプとして開発したものであるため、現時点では半自動である、汎用性に欠けるなどの問題が残されている。また、ドキュメントシステムだけでなくS/W設計者が作成する要求仕様書はどうあるべきか(何を書くべきか)等の問題点もある。これらの残された問題の解決を図るためこれからの開発課題として、ドキュメントシステムにデータベース、AI、PostScript等を導入すると同時に、要求仕様書の再検討を行っていく予定である。