

フォームエディタの実装とその考察

3C-4

谷川 康平†, 田中 朗†, 阿倍 博信†, 土田 泰治†, 坂下 善彦†

三菱電機エンジニアリング (株) †
三菱電機 (株) 情報電子研究所†

1 はじめに

「フォームエディタ」は業務上頻繁に使用される様式の決まった「定型文書」の作成を支援するものである。

一般に「定型文書」は、罫線で複数の領域に分割され、各領域には記述すべき内容が決まっている。そして、このような文書には全体を見渡せる一覧性、必要最低限の情報を記述する要約性、1~数枚で内容全てを記述する簡潔性が要求される。

「フォームエディタ」では、このような文書の様式を特に「フォーム」と定義[1][2]し、この「フォーム」の複雑な領域構造を簡単に編集できる機能を実現した。

本稿では「フォームエディタ」の実装の概要と、考察及び今後の拡張性について述べる。

2 フォームエディタの実装

2.1 フォームエディタの目標

「フォームエディタ」を実現する上で、以下の目標を挙げた。

- WYSIWYG、直接操作
フォームを作成する時には、常にそのレイアウトが見えていること。そのレイアウトに対して直接「削除」、「挿入」、「変更」などの操作が出来ること。
- レイアウトと内部構造の一貫性
同一のフォームを作成する過程はいくつも考えられるが、最終のレイアウトが同じものは、その内部構造も同じでなければならない。
- 様式データと文書データの区別
定型文書は様式と内容から構成される。様式データは共用され、文書データは個別に作成される。このため、様式編集と文書編集をモードにより区別し、そのデータファイルも拡張子で区別することにより別々に管理する。
- フレーム内の内容データの独立
フォームは複数の矩形の領域（フレーム）で構成され、フレームは4つ以上の罫線（図1の太線）で構成

される。フレームはブロック（図1の細線内部の領域）と呼ばれる表示領域を内部に持ち、フレーム内の内容はこのブロック上に表示される。この内容データは他のフレーム内の内容に依存せず独立である。

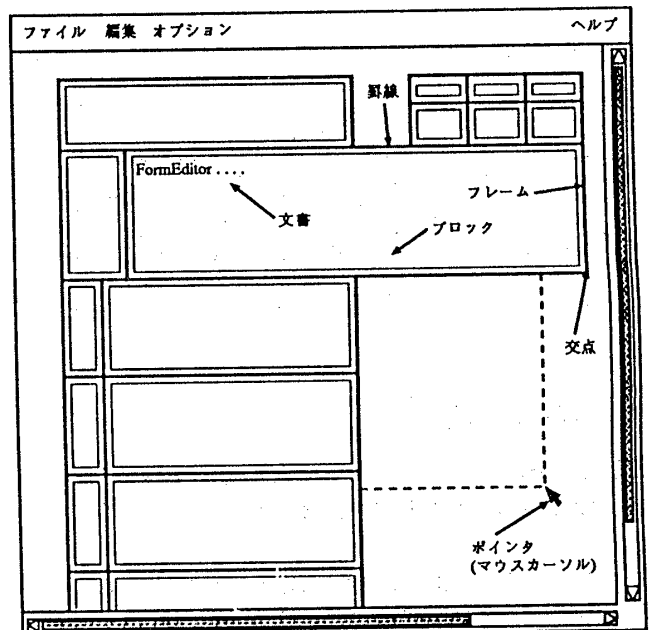


図1: フォームエディタの様式表現

2.2 「フォームエディタ」の機能

「フォームエディタ」の操作は大きく分けて3つの機能がある。

2.2.1 様式編集

共用データである定型文書の様式を定義（作成）する。様式編集の編集対象となるオブジェクトは「罫線、フレーム、ブロック、交点」などで、ポインタにより各オブジェクトを選択し、ドラッキングによる「生成、削除、移動、複写、拡張」の直接操作が行なえる。

2.2.2 文書編集

様式編集で作成された様式の各ブロックに対して、個別データである文書（内容）を書き込む。

内容を記述したいブロックをポインタでクリックすることによりブロック内は文書の編集が可能となる。文書作成の入力手段にはキーボードからの入力の他にコピー、ペーストが文字、行単位で行なえる。更に、書式変更がブロック単位で可能である。

2.2.3 印刷

作成された定型文書を印刷する。

画面のイメージをそのまま出力することが可能である。

2.3 実装環境

「フォームエディタ」は以下の環境で開発及び実装を行った。

表1. 開発及び実装環境

使用機種	MITSUBISHI ME 250
OS	ME-UX R3V4 B40R9009
Window 環境	X Window System ¹ Ver.11R4 JMotif1.1 ²

3 考察

3.1 実装の考察

様式の作成をポインタを使用して視覚的に行なえることは、様式を作成する手間を軽減し、効率を大幅に向上させた。

しかし、「定型文書」を含む文書を作成するには「フォームエディタ」だけで行なうことは難しい。また、「フォームエディタ」の特徴を有効に活用し、アプリケーションの設計ツールとして利用することを考慮すると、いくつかの拡張案が考えられる。

以降に、「フォームエディタ」の拡張について述べる。

3.2 「フォームエディタ」の拡張

3.2.1 ユーザインタフェースの拡張

現在の「フォームエディタ」の入力に関して、以下の課題が残っている。

- 英数字、文字などの入力されるデータの型の制限。
- あるブロックの入力後、次のブロックを入力可能な状態にするようなナビゲーションの機能。
- 定型文書には、慣例的な文がしばしば使用される。これらの文例をポップアップメニューなどで提供。
- ポインタの座標や編集モードなどのステータス表示を行なう。

- モードレスでの様式作成。

今後、エンドユーザに対する操作性の向上と、操作が直観的に理解できるようなユーザーインタフェースを提供していく必要がある。

3.2.2 アプリケーションシステムのための拡張

「フォームエディタ」の特徴を有効に活用するため、以下のような機能の追加が必要である。

- 他のアプリケーションにユーザインタフェース (UI) の提供

アプリケーションの UI 設計の手間を軽減するために、「フォームエディタ」の UI 部を他のアプリケーションに提供する。すなわち、アプリケーションはユーザインタフェースを「フォームエディタ」に任せて独自の処理を行ない、「フォームエディタ」はアプリケーションとインタラクティブに通信を行ないながらデータの入力、表示を行なう。

UI のカスタマイズの役割も果たせるので、画面設計にも利用できる。

- CDF³のような共通フォーマットの採用

各種アプリケーションのデータ入力システムとして「フォームエディタ」を利用する。このために、フォームエディタが現在利用している独自のフォーマットではなく、共通フォーマットにし、アプリケーションに利用しやすいデータ形式にする。

- ファイル管理の強化

文書には「機密文書」もあるので、こうしたファイルの保守性の強化を図らなければならない。

4 終りに

今回「フォームエディタ」の実装を行なうことにより、定型文書の視覚的操作による操作性の向上を認めることが出来た。他のシステムとの融合という観点から「フォームエディタ」の今後の可能性を確認し、実用に向けてさらに改善・拡張を行なっていく予定である。

参考文献

- [1] 田中他: フォームエディタ, 第41回情報処理全国大会, 7J-1 (1990)
- [2] 林他: フォームエディタにおける情報分割に関する一考察, 第41回情報処理全国大会, 7J-2 (1990)
1. X Window System は、MIT の登録商標です。
2. OSF/Motif は、Open Software Foundation の商標です。
3. CDF (Common Document File Format) は、SONY (株) の商標です。