

Gyaim

匿名で査読を行うため著者名なし*

概要. 誰もが毎日のようにテキスト入力と編集を行っている。ユーザの用途に応じた、多くのエディタソフトが存在するが、それらが持つ編集操作の方法と、編集機能は必ずしも同じではない。そこで、日本語などの入力を可能にする IME(Input Method Editor) を、編集操作にも活用することにより、様々なエディタや OS でのテキスト編集作業を共通化することが可能になる。本論文では、エディタ間の操作や機能の差異を IME により解決する例を示す。

1 はじめに

1.1 エディタとテキスト編集における問題

計算機上でテキストを編集するために様々なテキストエディタが利用されている。メールを書くためにはメールクライアントを利用し、コンソールでのプログラム開発には vi や Emacs を利用し、IDE を利用した開発では付属のエディタを利用し、Web 上でデータを書き込むにはブラウザ上のテキストフォームを利用するといったように、場合によって異なるエディタを利用することが日常的になっている。

これらのエディタソフトには、それぞれ、異なる機能と異なる操作体系が実装されている。たとえば、ブラウザ上のテキストフォームには秘密文字列を隠蔽する機能があり、特定のテキストエリアをフォーカスすることで動作する。文書編集ソフトやプログラム開発用のエディタはそれぞれ、スペルチェックや文法チェック、単語補完の機能を備えている。Emacs や vi といったエディタではユーザが拡張スクリプトを導入することで、操作と機能を拡張していくことができる。

これらの機能を利用するために必要な操作がエディタごとに異なっているのは不便である。ブラウザ以外の場所でも秘密文字列を入力したいという要望や、Emacs と同じキーバインドでブラウザのテキストフォームを移動したい、といった要望に答えるためには、それぞれのエディタ開発者が随時アップデートを行うほかなかった。理想的には、どのような状況でも同じ操作でテキスト編集を行なえるべきであろう。

個別に実装されたエディタの操作を統一することは不可能かもしれないが、様々なエディタで共通に利用できるソフトウェア層をユーザとアプリケーションの間に置くことにより、異なるエディタの編集操作をある程度共通化できる可能性がある。

1.2 問題の解決手法

現在のパソコンには IME(Input Method Editor) と呼ばれる文字入力機構が用意されており、様々な言語のテキスト入力に利用されている。たとえば日本語入力用の IME を利用すれば、Emacs でもブラウザでも IDE でも同じ操作で日本語を入力できる。IME はテキストエディタのようなアプリケーションとは独立した実装であり、ユーザとアプリケーションとの間にあるソフトウェアである。すべてのキー入力をハンドルし、各国語に変換した結果をアプリケーションに送出している。この IME が、テキストの挿入/移動/削除といった編集操作を受け持つことにより、入力の場合と同様に、どのエディタでも同じ操作で編集が可能になる。

2 実装例

MacRuby で記述された IME である「Gyaim」に様々な編集機能を追加したものを利用した編集作業の例を示す。一般に IME はアルファベット以外の文字を入力するときだけ有効にするのが普通であるが、ここでは Gyaim を 常に有効にしておくことによりあらゆるキー入力を Gyaim が取得している。

2.1 ブロック移動

2.2 文字列暗号化

2.3 Dynamic Macro

以上の機能に加え、文字のスワップやキャレット移動といった基本的な編集操作を Gyaim は実装しており、システムやアプリケーションによらないテキスト編集が可能である。

3 実装

Gyaim は MacRuby で記述された Mac 用の IME であり、ソースが 500 行程度とコンパクトであるにもかかわらず、他の IME に見られない機能を実装しており、本論文のような実験も容易である。

Copyright is held by the author(s).

* 匿名で査読を行うため所属名なし

3.1 MacRuby

MacRuby は、Mac 用のアプリケーションを開発するために拡張された Ruby 実行環境であり、Mac の Objective-C ライブラリを Ruby で扱うことが出来る。Gyaim では、日本語などの入力を補助するフレームワークである InputMethodKit Framework を MacRuby から呼び出すことによって基本的な IME の機能を実装している。

しかし、ブロック移動やインデントの処理をあらゆるテキストエリアで行うためには、テキストフィールドに 入力されている全文を IME が取得する必要がある、InputMethodKit はこの機能を備えていない。Gyaim では、AppleScript などを併用することによりこの問題を解決したが、実装には課題が残っている。

謝辞

謝辞は、ブラインドレビューのため、投稿時には削除すること。カメラレディ時に、必要があれば追加すること。

4 参考文献

未来ビジョン

未来ビジョン