

続・理想のマップエディタを求めて

文責：古川 史幸

1 前置き

昨年、『理想のマップエディタを求めて』¹というタイトルで会誌の記事を書かさせていただきました、古川です。今回、『続』という文字がタイトルに含まれるということから、現在の状態を比較的容易に想像できることと思われます。

...そうです、まだ完成してません。
しかし、着実に進歩はしています。

そんな、なかなか完成しないマップエディタについて、いろいろと書き連ねてみようと思います。

2 昨年からの経過

記憶を頼りに以下の表を書いています。思い違いは多々あるかもしれませんが。

[2004年11月] 第54回調布祭。マップエディタは結局完成せず。

[12月~2005年2月] 調布祭を過ぎたことによるモチベーションの低下、学科実験による忙しさなどからあまり開発に手を出さず。

[3月] 春休みになり、開発を再開するも、大きな問題が発生。この点については後述する。

[4月~5月] マップ描画ルーチンの書き直し。旧バージョン以上の機能と使いやすさを実現。

[6月] マップ描画ルーチンの完成により気の緩み。ほぼ1ヶ月開発には手をつけず。

[7月] エディタ本体のプログラミングに手をつけ始める。このとき、プラグイン構造などは未決定。

[8月] プラグイン構造をある程度決定し、新規作成機能の実装。マップ描画ルーチンのわずかな手直し。

[9月] エディタに用いる他の方が制作したライブラリに対し、手を加える。サークルの合宿中にアプリケーションアイコンを描いてみる。

[10月] 標準のリストボックスが気に入らないので自作。現在に至る。

3 旧バージョンでの問題点

開発を再開しようと思ったとき、そこにはこれらの問題がありました。

マップの編集方法

旧バージョンでは、既にマップを表示するところまでは進んでいました。しかし、残念ながらその次の段階である、「編集」まで進むことが出来ませんでした。

理由は明確です。プログラムがマップの編集を考えた構造をしていなかったのです。

例えば、マップ上でマウスをドラッグすると、対応するマップチップがどんどん置かれていくとしましょう。しかし、

マップ描画UIにマップチップが変化したことをどうやって伝えよう？

ドラッグ中に右クリックすると編集はキャンセルされるが、どうやって元に戻そう？

Undo/Redoの仕組みをどう提供しよう？

範囲選択やクリップボードの扱いはどうしよう？

...などと、考えることが山ほどでてきました。そして旧バージョンのプログラムでは、これらを解決する方法を考慮していなかったのです。

無理やりこれらの機能を実現しようとする、様々なところでプログラムが破綻するのは目に見えていました。したがって、全体を一度見直し、新たに作り直すことが重要であると考えたのです。

マップ描画ユーザインターフェース

問題はもうひとつありました。旧バージョンのマップ描画UI(ユーザインターフェース)には、レイヤーが2枚までしか重ね合わせ出来ないという欠点がありました。最初にこれを書いていた当時は、描画速度の向上を目論んで2枚までに制限したものだと思われます。

しかし、よく考えればレイヤーの枚数を増やせ

¹昨年度の当サークルの会誌をお持ちの方、あるいは今年度の68CDをお求めになった方は、まずは昨年度の記事をお読みになってから、本記事を読むことをお勧めします。(私の書いた昨年度の記事は、今年度の68CDに収録されています。)

ばそれだけ描画速度が低下するのはあたりまえの話で、描画速度の向上を目指すのにレイヤーの枚数を制限するのは、最適化の方向性が違っていたといえます。それに、レイヤーを多層にすることで多彩な表現の可能性があることを見出しました。例えば、グリッドの表示です。レイヤーとしてマップチップの一つ一つに枠を描画すれば、それはグリッドになります。

このようなことから、マップ描画 UI は多層レイヤー対応にすることに決め、一から新しく書き起こすことにしました。

4 昨年の記述から変わった点

プラグイン

マップファイルフォーマットを自由に決めるために必要なプラグインですが、旧バージョンではデータ構造自身を記述するという方針でした。今回は方向性を変えて、データ構造は単純に、編集機能を抽象化する方針をとることにしました。したがって、データ構造を定義するものと、編集機能を定義するものの2種類のプラグインを用意しました。

データ構造については昨年話したので、ここでは編集機能について説明します。編集機能とは、グラフィックエディタなどにある「ペン」、「矩形描画」、「塗りつぶし」などの機能、いわゆるツールボックスを指します。単純なデータ構造に対しては、プログラム本体側で用意される編集機能を用いればプラグイン開発の手間を少なくできます。逆に特殊な編集機能が必要なデータ構造に対しては、その編集機能をプラグインとして用意することによって、対象のデータを編集できるようになります。

多層レイヤー

前述の通り、多層レイヤーの表示が可能になりました。ここで、表示および編集するレイヤーの選択をリストコントロールなどの標準コントロールで行おうと思っていましたが、要求を満たす標準コントロールが見つからなかったため、ついカットとなって自作の UI を作ってしまいました。後悔はしていません。

5 メモ書きのススメ

ここで、現在のプログラムの規模が気になったので、調べてみました。その結果が以下の表です。

表: 現在のエディタのプログラム規模

モジュール	ファイル数	行数	文字数
プログラム本体	10	1,928	56,259
プラグインベース	7	615	11,369
マップ描画 UI	21	4,235	100,127
リスト表示 UI	1	857	18,201
サンプルプラグイン	5	1,173	40,487
合計	44	8,808	226,443

コメント・空行なども含みます。
他の方が作ったライブラリは含みません。
開発環境による自動生成の部分も含まれますので、実際に私が書いた部分はこれより少ないです。

表を見ると、かなり大規模になってしまったということがわかります。これだけ大規模なプログラムになると、それぞれのプログラムの部品を上手いこと協調動作させ、プログラムとして正しく動作させるのは至難の業です。

そのため、実際にコーディングを行う前に、自分の作りたいものをはっきりとさせておく必要があるでしょう。ただ、個人ベースで制作するものですので、ちゃんとした仕様書は必要ないでしょうが、これから作るものの簡単なメモを書いておくといいかもしれません。これは実際に紙に書くということが重要だと思われます。ちなみに、チラシの裏など、ばらばらになってしまうのではなく可能性が大きいので NG です。書くならノートやらくがき帳などがいいでしょう。紙にいろいろと書いているうちに自分の頭の整理もできますので、プログラマの方はぜひメモ書きをやってみることをオススメします。

サークルでゲームなどを一度も完成させたことのない自分が偉そうに書くのもなんですが、マップエディタについては時間がかかっても着実に完成に近づくように努力していこうと思います²。最後までお読みいただきありがとうございました。

(2005/10/03 古川 史幸)

²時間を気にしなくていい点が、趣味でプログラムを書いている人の特権ですね。ただ、やはりある程度の完成目標(例として、『調布祭に出展できるようにする』など)がないとプログラムは完成しないでしょう。