B-011

既開発部分の解析によるレイアウト調整済み GUI の生成
Generation of Layout-Arranged GUI Using Existing Programs

白井 盛太郎† 白銀 純子‡ 岩田 一* 深澤 良彰†
Seitaro Shirai Junko Shirogane Hajime Iwata Yoshiaki Fukazawa

1. はじめに

現在、様々なソフトウェアが GUI(Graphical User Interface)を備えている。GUI はユーザが直接操作を行う、ソフトウェアとの接点であるため、ユーザビリティ[1][2](使いやすさ)が重要視されている。

ユーザビリティの中の重要な要素として、一貫性[3]がある。一貫性とは、GUI におけるウインドウ内のウィジェットの使用方法や配置方法を、そのアプリケーション内やプラットフォーム内で統一することにより、アプリケーションの操作感を統一することである。

しかし、一貫性を保ったウインドウを 1 つ 1 つ構築していくのは、開発者にとって大きな負担である。ウインドウが増えれば増えるほど、一貫性を保ちづらく、変更容易性も低下する。

そこで本研究では、アプリケーション内での GUI の一貫性を容易に保つことを目指す。具体的には、開発者の GUI 設計における負担を軽減するために、既に開発したソフトウェアにおける GUI のソースコードを解析し、一貫性を保ったレイアウト調整済みの GUI を作成する手法について提案する。

2. 本手法の特徴

GUI のレイアウトには、OS を提供する企業や、組織内でガイドラインが用意されており、それに沿ってレイアウトされることも多い。しかし、大規模なソフトウェアではウインドウが多く存在し、それらすべてにガイドラインを厳密に適用することは非常に困難であり、コーディングにも時間がかかる。

そこで本研究では、開発済みの GUI ソースコードを解析することにより、複数のウインドウ間で共通して用いられているレイアウト方法からルールを発見する。そして、そのルールを適用した GUI を自動的に生成する。

これにより、開発済みの GUI から生成したレイアウトルールが、未開発部分のウインドウに適用されるため、ソフトウェア内での GUI の一貫性が容易に保つことができる。また、一度ルールを生成すれば、新しく開発する別のソフトウェアの GUI にも適用させることが可能であり、組織内で GUI レイアウトに一貫性を持たせる際にも活用できる。

さらに、自動的にウインドウのソースコードを生成する仕組みを設けることにより、開発者の GUI コーディングの手間を軽減させることができる。

3. レイアウト調整方針

現状で本研究は、ウインドウの最下部に位置するボタンのレイアウトルールを発見し、自動レイアウトする部分を確立する。

最下部に位置するボタンは、一般的にソフトウェアに処理の決定を命令する(ウインドウを終了または遷移させる)役割をもったものが多い。実際に代表的な 28 種類のアプリケーションにおける 209 ウインドウに存在するボタンのラベルを調査した。表 1 に、最下部にある決定を命令するボタンを集計した結果を示す。

表 1 ウインドウ中の決定ボタンの出現回数と出現率

ラベル名	出現回数	出現率
キャンセル	178	85%
OK	124	59%
ヘルプ	37	18%
次へ	21	10%
閉じる	17	8%
戻る	10	5%
完了	9	4%
前へ	7	3%

表 1 より、頻出するラベルはある程度決まっており、ラベル名を基準とした解析でルールを発見できる可能性が高く、発見したルールを適用する機会も多く存在すると考えられる。そこで、ラベル名を基にルールの解析・適用を行うこととする。

4. 本手法の手順

本手法のシステム構成を図 1 に示す。本手法は、現段階では Java 言語の GUI を構築するためのツールキットである Swing を使って作成された GUI を対象としている。

4.1 ソースコードの構文解析と AspectJ ファイルの生成

本手法では、まず JavaCC[4]を用いた Java の構文解析器(Java Parser)により、Java ソースコード中のウィジェットオブジェクトを抽出し、情報を出力する AspectJ[5]ファイルを自動生成する。

ソースコードを解析することで、ウィジェットの変数名の取得や ID 付けが容易になる。また、ウィジェットの X 座標、Y 座標、幅、高さの 4 種類の情報をプログラム実行時に動的に取得できる。

† 早稲田大学 Waseda University
‡ 東京女子大学 Tokyo Woman’s Christian Universiry
＊神奈川工科大学 Kanagawa Institute of Technology

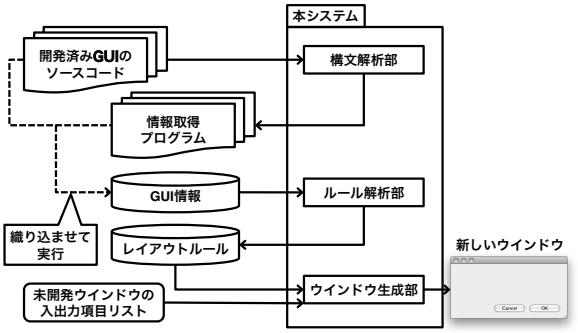

図 1 本手法のシステム構成

4.2 レイアウトルールの解析

4.1 節で取得した情報を利用し、ウインドウ右端からウィジェット右端の距離(ピクセル)dfr、ウインドウ中央からウィジェット中央の距離(ピクセル)dfc、ウインドウ中央からウィジェット中央の距離(パーセンテージ)dfcPercent、ウインドウ下端からウィジェット下端の距離(ピクセル)dfb の 4 種のレイアウト情報を求め、レイアウトルールを解析する。図 2 に、ウインドウ中の“OK”ボタンに対応する 4 種のレイアウト情報の対応を示す。

レイアウトルールの解析は、まず、ウインドウ下端からの距離を表す dfb の最小値を記録し、その dfb 値を持つウィジェットをウインドウ最下部に存在するウィジェットとして判定する。

そして、最下部のウィジェットを型とラベルの組み合わせごとにグループ分けをする。X 軸方向に関するレイアウト情報(x, dfr, dfc, dfcPercent)ごとに値の出現回数を記録し、複数のウインドウを解析した結果、複数回出現した中で最も多く出現したレイアウト情報と値の組み合わせをルールとする。

解析したルールは、型、ウィジェットのラベル名、揃えの基準(left, right, center)、X 軸方向に関するレイアウト情報の値(x, dfr 等)、Y 軸方向に関するレイアウト情報の値(x, dfb)、ウィジェットのサイズの 6 種の情報をまとめたものである。

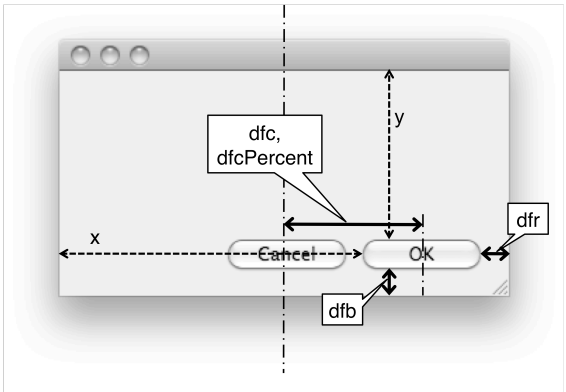

図 2 レイアウトに関する 4 種の情報

ルールは XML で出力される。図 3 に出力される XML ノードの例を示す。ルールに関係の無い要素は“-1”として出力している。

```
<OK align="right" dfb="6" dfc="-1"
    dfcPercent="-1" dfr="6" height="-1"
    type="JButton" width="100" x="-1" y="-1"/>
```

図 3 出力されるレイアウトルールの XML ノード

4.3 GUI の自動レイアウト

4.2 節で求めたルールを読み込み、ウィジェットの型とラベル名のマップ、ラベル名とルールのマップを作成する。

開発者がウインドウのタイトル、クラス名、配置するウィジェットの型とラベルを新しく開発するウインドウの情報として入力すると、各ウィジェットに対するレイアウトルールが存在するか、読み込んだルールと比較し、ルールが存在するならば、そのレイアウトルールに則したウィジェットがレイアウトされたソースコードを出力する。

例えば、JButton 型の OK というラベルを持つウィジェットを入力した場合、まずシステムは型とラベル名のマップに照らし合わし、入力された組み合わせに対応するマッピングが存在するか調べる。マッピングが存在する場合、システムはそのラベル名をキーに、ラベル名とルールのマップからルールを取得する。

こうして開発者が入力したウィジェットにそれぞれに対応するルール取得し、ソースコードに変換し出力する。

5. 今後の課題

本研究ではウインドウ最下部ボタンのルール発見と適用手法について提案した。今後の課題は以下の通りである。

- ボタン以外のウィジェットへのシステムの適用
- グループ化されたウィジェットへのシステムの適用
- 最下部以外のボタンへのシステムの適用
- ルール判定基準の見直し
- 変数名によるウィジェット間の関連の判別

参考文献

[1]社団法人 人間生活工学研究センター, “ワークショップ人間生活工学 第 3 巻 − インタラクティブシステムのユーザビリティ− ”, 丸善株式会社 (2005).

[2]黒須 正明, “ユーザビリティテスティング ユーザ中心のものづくりに向けて”, 共立出版株式会社 (2003).

[3]Alan Cooper and Robert Reimann and David Cronin, “About Face 3 The Essentials of Interaction Design”, Wiley Publishing, Inc. (2007) (長尾 高弘 訳 『About Face 3 インタラクションデザインの極意』, 株式会社アスキー・メディアワークス (2008).)

[4]五月女 健治, “JavaCC コンパイラ・コンパイラ for Java”, 株式会社テクノプレス (2003).

[5]瀬嘉秀, 天野まさひろ, 鷲崎弘宜, 立堀道昭, “AspectJ によるアスペクト指向プログラミング入門”, ソフトバンクパブリッシング株式会社 (2008).