

# 端末資源情報に基づくコンポーネント選択手法

谷沢智史<sup>†</sup> 中川晋吾<sup>\*</sup> 金指文明<sup>†</sup> 富樫 敦<sup>‡</sup>

静岡大学情報学部<sup>†</sup> 静岡大学大学院理工学研究科<sup>\*</sup> 有限会社カラビナシステムズ<sup>‡</sup>

## 1.はじめに

現在、PCをはじめ、携帯電話、PDAなどの情報端末の多様化が進んでいる。これらの端末でビジネスアプリケーションを利用したいという要求は強いが、マルチプラットフォームアプリケーションを開発することは非常にコストがかかる。そこで、我々は資源適合型アプリケーション統合開発環境“GISEN”を提案している<sup>[1]</sup>。

資源適合は実行環境に合わせてコンポーネントを切り替えることで実現するため、コンポーネントの検索手法が必要となる。本論文では、コンポーネントに記述された仕様情報と実行環境などによる要求情報に基づき、最適なコンポーネントセットを選択する手法を提案する。

このような手法に関する既存研究には次のようなものがある。

### ● 手動記述方式

コンポーネントの仕様情報をプログラムコードとは別に用意し検索する方式。利用局面を自然言語で記述し、検索する研究<sup>[2]</sup>などがある。

### ● 自動取得方式

プログラムから自動的に取得できる情報をもとに検索する方式。メソッドの実行結果やクラス仕様によって検索する研究<sup>[3]</sup>などがある。

いずれの方式もコンポーネント固有の機能に着目しているが、複数の役割を持つコンポーネントが協調する際、それぞれの役割を担う候補が複数ある場合の具体的な選択方式については深く言及されていない。

## 2.コンポーネントと要求の定義

本論文ではコンポーネントの選択を実現する際、コンポーネントと要求という2つの要素を扱う。

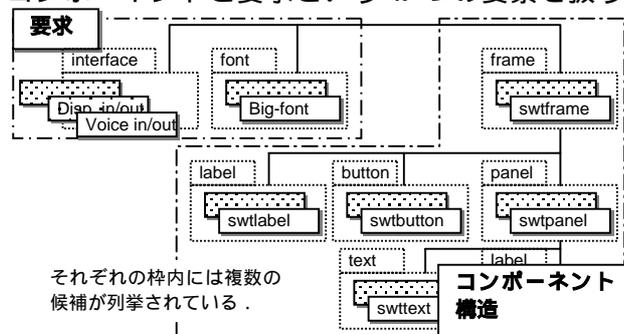


図1 コンポーネントと要求

Component Selection Method based on Terminal Description

<sup>†</sup>Faculty of Information, Shizuoka University

<sup>\*</sup>Graduate school of Science and Engineering, Shizuoka University

<sup>‡</sup>Carabiner Systems, Inc.

これらをまとめたモデルを図1に示す。

まず、コンポーネントの構造情報とは、アプリケーションでのコンポーネントの階層的な配置情報である。これらは抽象的なコンポーネントであり、それぞれに複数の実装コンポーネントの候補が与えられる。

次に、要求とはディスプレイの解像度や表示サイズなど、端末あるいはユーザによって定義される情報である。要求は、インタフェース形態要求、フォントサイズ要求など、いくつかのカテゴリに分けて定義されるものとする。各カテゴリに対応する要求は唯一ではなく、優先度(0.0~1.0)をつけて複数の異なる要求を記述可能とする。よって、最高優先度ではない要求で妥協することが可能となる。

コンポーネントセットの選択は、図1のモデル上で要素全体の相性が最大になるようなセットを選び出すことで実現する。相性は、コンポーネントと要求のそれぞれに設定された表1のような仕様セットを比較することによって評価する。なお、仕様セットは、コンポーネントならばその開発者、要求ならば端末ベンダあるいはユーザによって記述される。

表1 コンポーネントの仕様セット例

	プロパティ名	型	値
属性	WidgetType	文字列	Swing
	DisplaySize	2次元サイズ	(100×100)
	UseMemory	1次元サイズ	100000
文脈	Output	文字列	Display
	Input	文字列	Mouse
制約	DisplaySize	最大2次元サイズ	(100×100)
	Palette	最大1次元サイズ	256

基本的に、要素間の相性計算は同じプロパティ名をもつ仕様同士を比較することで行なう。左の“属性”、“制約”、“文脈”とは仕様の有効範囲を表す。“属性”は兄弟(同一階層の要素)の間で有効な仕様、“制約”はその要素の直接の子に対して有効な仕様、“文脈”はその要素のすべての子孫に対して有効な仕様である。

仕様同士の比較は、ある仕様から一方の仕様へ“投票”することで行なう。例えば、文字列型の場合、“SWT”から“AWT”ならば一致しないので0票、“SWT”から“SWT”ならば1票となる。

また、型には制限を表現するため、“拒否”を示すものがある。例えば最大2次元サイズ型(100×100)は(120×120)には“拒否”を示すものとする。

### 3.コンポーネント選択手法

(a) 同一階層の組み合わせ(兄弟セット)を評価  
はじめに、図2のように階層ごとに兄弟について組み合わせをつくり、“拒否”関係が発生しない全ての兄弟セットについて評価していく。

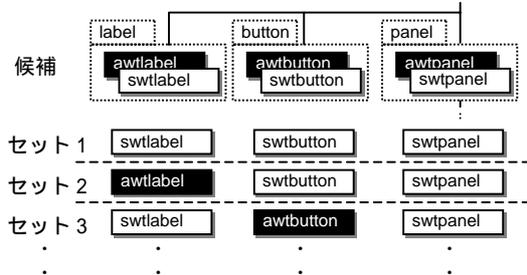


図2 兄弟セット

あるセットの評価値の計算は、同一セット中の要素から要素への相性計算からはじめる。例えば swtlabel から swtbutton に対する相性を計算する場合、双方が共通に持つ“属性”仕様のそれぞれについて swtlabel の仕様から swtbutton の仕様へと投票させ、その平均値を求める。そしてその値に swtlabel の優先度をかける。同様に自分以外の全要素からの投票値を計算すると、兄弟の要素数を  $m$ (この場合は 3)とすれば、総和は  $0 \sim m-1$  の値となる。そこで、 $m-1$  で除算し  $0.0 \sim 1.0$  の得点となるよう補正する。これを全要素について計算し、総和をセットの評価値とする。

(b) 兄弟セットのグループ化処理

次に、それぞれの階層について評価した兄弟セットを親子間で組み合わせていく。ただし、兄弟セットをさらに親子間で組み合わせると組み合わせ数が膨大なものとなるため、似ている兄弟セットをグループ化する。この規則は以下のように設定した。

- 等価な制約および文脈を持つ
- 構成要素のうち、1/2 以上が同じもの
- 評価値の差が  $0.5 \times$  (兄弟の要素数)以下

なお、得点の低い兄弟セットほど大きなグループへとグループ化を行なう。

(c) 親子の組み合わせを評価

グループ化したものをすべて組み合わせ、“拒否”関係が発生しない親子の組み合わせについて評価値を算出する。親の“制約”と“文脈”仕様から、兄弟セットにある要素の“属性”仕様へと投票させ総数を計算する。これを全ての兄弟セットについて行い、最高得票数の兄弟セットの投票総数が  $1.0$  になるように、各兄弟セットの投票総数を  $0.0 \sim 1.0$  の間に正規化する。これを兄弟セットの評価値にかける。

(d) コンポーネント一貫性の評価

最後に、ある役割を持つコンポーネントについ

て一貫性を評価に加える。例えば、label という名前の抽象コンポーネントが複数利用されている場合、すべて swtlabel で占められているセットの方が awtlabel が 1 つだけ含まれるようなセットよりも望ましいとする重みづけを行なう。

以上の方法で兄弟セットを親子間で組み合わせたものの合計点を計算し、全兄弟セットの合計点が最も高い親子の組み合わせを結果とする。

### 4.評価とまとめ

本研究では、複数の要求に優先度をつけ、要求とコンポーネント間の相性を考慮しながら最適なコンポーネントセットを選択する手法を提案した。

この手法をシミュレートするプログラムを作成し、グループ化を行なう場合と行なわない場合の計算時間を比較した。この結果を図3に示す。

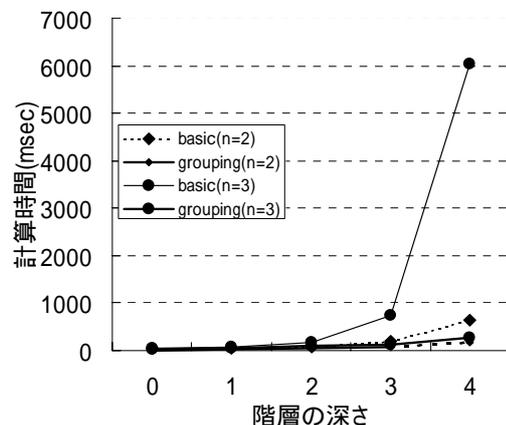


図3 コンポーネント階層と計算時間

グラフ中の  $n$  は 1 階層あたりの要素数である。このグラフから、グループ化は 1 階層あたりの要素数が増えるほど計算時間の削減効果が高いことが確認できた。

今後の展望としては、コンポーネント利用情報を収集し検索に利用する研究<sup>[4]</sup>のように、すでに“正しく”実行されたコンポーネントの組み合わせ情報を保存し、評価値計算に反映することが考えられる。

### 参考文献

- [1] 中川晋吾, 谷沢智史, 金指文明, 富樫敦, “資源適型型アプリケーション統合開発環境の提案”, FOSE2002
- [2] J.Grundy, “Storage and retrieval of Software Components using Aspects”, Australian Computer Science Conference, 2000
- [3] 鷲崎弘宜, 深澤良彰, “有向置換性距離に基づくコンポーネント検索システム”, SPA2001
- [4] 古賀健太郎, 飯田元, 松本健一, 井上克郎, “ソフトウェアコンポーネント利用情報の収集と共有”, 電子情報通信学会技術研究報告, ソフトウェアサイエンス研究会, Vol. 100, No. 472, SS2000-27, pp.1-8