

## 研究課題別研究評価

1. 研究課題名：Visibility Programming の研究

2. 研究者名：原田 康德

3. 研究のねらい

使用者の立場から、プログラムを柔軟に拡張できるようにするために、ソフトウェアの構築法を探り、それを実証する。具体的には、従来のプログラムはデータとそれを操作する小さなプログラムの組をひとまとまりのオブジェクトとして扱い、それを沢山集めてひとつのアプリケーションとしていた。これはデータごとの拡張には優れているが、あるデータを操作する新しい機能を拡張することは難しい。それを本研究では、すべてのデータを使用者とコンピュータとで共通にアクセスできるようにし、それらを複数の機能別のプログラムによって操作することでアプリケーションを作成する。新しい機能は新しいプログラムを作成するだけで追加できる。

4. 研究結果及び自己評価

### 研究結果

1) Visibility Programming

システムのすべての状態を表現する共有データ空間と、それをアクセスする複数のプロセスからなる構造によるプログラミング (Visibility Programming (VP)) が実現されると、小さな機能をユーザレベルで拡張することが容易なシステムが実現できる。従来はアプリケーション、アルゴリズムが決まってから、それに必要なデータ構造が決まっていた。しかし、VP のためにはどのような用途にでもデータを使えるように設計しておく必要がある。そこで、共有構造として全ての参照が逆参照を持つデータ構造を採用した。

2) CCC：ユーザ定義可能なオブジェクト構造

ユーザによってオブジェクトの構造を自由に定義できる言語 CCC を開発した。C++オブジェクトが共有メモリに格納できなかった理由はメソッドディスパッチのためのテーブルが原因であった。CCCは従来のオブジェクト指向言語が言語の組み込み機能として提供してきたメソッドディスパッチをユーザが自由に定義できるようにするものである。CCCの極めて重要な特徴は同じデータに対しても、条件式の本を代えることで、自由にそのデータの解釈を変更することができるという点である。この性質はVPが目指していた、「データが最初にあり、後からそのデータを自由に解釈する」そのものである。

3) パターンマッチによるメソッドディスパッチ

任意のパターンによりクラスを定義し、そのなかにメソッドを定義する。ある対象に対してメッセージを送ると、それに対応するメソッドを定義しているパターンに対してパターンマッチを試み、成功したらそのメソッドを呼び出すようにする。これによってすでに存在している任意のデータ構造を任意の粒度でオブジェクトとして扱うことが可能となった。

4) ダイナミックデータ抽象

従来のデータ抽象が静的に行われていたのに対して、VPでは動的にデータ抽象を行う。すなわち、条件式やパターンマッチに成功した時点で、対象にはある制約がついたことになるが、その部分構造に動的に名前をつけて、プログラムから参照できるようにする。名前によって抽象化さ

れた構造を用いることで、プログラム再利用性が向上する。

#### 自己評価

- 1) 当初、予定していた計画の半分も実現できなかった。それは、向かおうとしていた問題が良く見えていなかったからである。まず、計画では研究を開始してすぐに、デモ用のシステムを作るようになっていた。しかし、実際に研究をすすめてみると、提案している方式ではオブジェクト指向で使われている近代的なプログラミング技術がほとんど使えず、プログラミングは難航した。このことから、研究はより具体的に、いかにして近代的なプログラミング技法を VP という新しいプログラミングの枠組みで実現するか、という方向に進むことになる。
- 2) 研究は最初は直感で進み、後からその裏付けの理由を考える、として進めてきた。もちろん、そのヒントは様々な検討の末に生まれたものであるのだが、最初の方向を決めた後は、過去の経験や美しいシステムに対する信仰などに従って進めてきた。たとえば、ダイナミックデータ抽象という考えは、当初からアイデアとしてはぼんやりとあったものであるが、このような名前をつけたことによりずいぶんとはっきりと見えてきた。逆ポインタを持つシステムは直感的には正しいと信じるが、まだ説得力をもつ説明は見つかっていない。
- 3) まだ、具体的な応用につなげるまでには越えにくい壁があるように感じる。これは、単にツールが整備されていないからなのか、それともまだ、根本的な問題が残っているからなのか、よく分からない。しかし、研究をスタートした3年前からみると、格段に見通しがよくなった。
- 4) これから研究が面白くなるところで、本制度の期限がきてしまった。しかし、この先はこれまでのような、成果の出しにくい状況ではなく、順調に成果を積み重ねることができそうであり、特に本制度の直接的な支援がなくとも進められる。逆に言えば、本制度の保護があったからこそ、このような冒険的な研究を進めることができたといえる。

#### 5 . 領域総括の見解

様々なユーザが多種多様な用途で計算機を使用している今日、プログラム開発者はユーザの要求を満足させるようなアプリケーションを実現することは不可能である。この問題の解決には、ユーザ自ら、あるいはユーザに近い開発者がアプリケーションの拡張を行えるようにする必要がある。原田康徳は、ユーザの立場から、プログラムを柔軟に拡張できるようにするためのソフトウェアを開発し、検証したことは評価できる。

#### 6 . 主な論文等

- 1) \*Yasunori Harada , Kenichi Yamazaki , Richard Potter: CCC: User-Defined Object Structure in C . ECOOP 2001: 118-129
- 2) 原田康徳 , 山崎憲一: CCC --データの内部表現に依存しないオブジェクト指向 , 情報処理学会論文誌:プログラミング , Vol . 42 , No . SIG2(PRO 9) , pp . 48-60 (2001) .
- 3) Kenji Miyamoto , Yasunori Harada , Richard Potter: KVispach : A Visual Language for Animation that Rewrites Kinematic Objects , Advanced Visual Interfaces 2000 , 255-260 .
- 4) 宮本健司 , 原田康徳 : 分散書換えビジュアル言語によるネットワークアプリケーション構築法 , コンピュータソフトウェア , Vol . 16 , No . 5 (1999) .
- 5) 原田康徳 : 逆ポインタを持つ CONS , プログラミングシンポジウム ( 2000 ) .
- 6) 原田康徳 : Visibility Programming - 宣言的状態記述によるシステム記述 , SPA2001 .
- 7) 原田康徳 : データ抽象のないプログラミング , 夏のプログラミングシンポジウム ( 2001 ) .