様式 C-19

科学研究費補助金研究成果報告書

平成　22 年　5 月　10 日現在

機関番号：21602
研究種目：若手研究(ｽﾀｰﾄｱｯﾌﾟ)
研究期間：2008～2009
課題番号：20800041
研究課題名（和文）　自己説明型ソフトウェアコンポーネントデータベース

研究課題名（英文）　Self-explanatory software component database

研究代表者
渡部　有隆（WATANOBE　YUTAKA）
会津大学・コンピュータ理工学部・准教授
研究者番号：30510408

研究成果の概要（和文）：　ビジュアルプログラミング言語で記述されたソフトウェアコンポーネントをデータベース化し、プログラミングによる問題解決・シミュレーションの簡略化及びアルゴリズムの学習効率を改善するためのプログラミング環境の基盤を開発した。グラフアルゴリズム、計算幾何学、粒子シミュレーションのためのコンポーネントを新たに開発し、これまで開発されたコンポーネント部品をデータベースに格納し、さらなる研究活動へ繋げるためのサーバ・ＡＰＩを開発した。

研究成果の概要（英文）：A basis of the programming environment for problem solving, simulation as well as for algorithm education has been developed. The environment is based on visual programming languages and a special software component database. Components related to graph algorithms, particle-in-cell algorithms and algorithms for computational geometry have been developed and stored into the database. In addition, a special server and its API (Application Program Interface) have been developed for further research activities.

交付決定額

（金額単位：円）

	直接経費	間接経費	合　計
2008 年度	1,310,000	393,000	1,703,000
2009 年度	1,040,000	312,000	1,352,000
年度			
年度			
年度			
総　計	2,350,000	705,000	3,055,000

研究分野：工学
科研費の分科・細目：ソフトウェア
キーワード：　ビジュアル言語、プログラミング、データベース、サイバーフィルム

１．研究開始当初の背景

(1)コンピュータ理工学における背景：CPU やLSI チップの性能向上、インターネットの普及は驚嘆そのものであり、これらハードウェアや通信技術の発展は、次世代の情報社会

ではより多くの複雑なソフトウェアが必要とされることを示していた。しかし、未だに「ソフトウェア危機」に関する記事が目立つように、当初（現在）のソフトウェア技術はハードウェア技術の発展とは比較にならないほど遅れをとっていた。また、次世代の情報社会では、高齢者、障害者、コンピュータの非専門家も含めたより多くの人々がコンピュータのパワーを公平に有効活用し、情報資源の生成に携わることが可能となることが理想的であるが、当初のプログラミング環境がこれらの人々をサポートすることは期待できなかった。さらに、地球シミュレータ等のスーパーコンピュータの登場に続き、Cell BE を搭載したソニーPS3 に代表されるいわゆる’パーソナルスーパーコンピュータ’の登場により、並列プログラミングがさらに身近なものとなりつつあった。物理シミュレーションやナノテクノロジー等の科学技術における最先端の発見・発明のカギとなる重要な高性能演算及び大規模演算においては、プログラミングの専門家ではない科学者らが、いかに簡単に並列プログラムを実装できるかが大きな課題となっていた。しかし、並列プログラムを実装することはさらに難しいタスクであることは明白であり、これを解決するための革新的なプログラミング環境が急速に求められていた。

(2)問題を解決する関連技術に関する背景：プログラミングを簡易化・効率化する技術として挙げられるのが、コンポーネント化技術やビジュアル言語であった。コンポーネント化技術では、目的に応じた機能をライブラリ化し、開発の高速化、実績のあるものを再利用することによる信頼性の向上、問題発生時の原因解決の簡素化、将来にわたる拡張性の確保等、ソフトウェアの開発、運用、保守の様々な段階で技術者の生産性を向上することができた。しかし、当初のコンポーネント化技術は、コンポーネントの利用方法や接続方法の定義に重点が置かれていたため、内部を理解し用途に応じた変更を加えることは専門家であっても非常に困難であった。また、既存のライブラリはソフトウェア開発に必要な部品だけでなく、幾何、グラフ構造、数値計算等の様々な領域に特化したあらゆるアルゴリズム群を含み有効利用されているが、上記したいわゆる’Black Box’として提供されるコンポーネントの理解の難しさに加え、膨大な量のコンポーネント群から必要なものを指定し取得するための技術が欠如している問題があった。ソースコード内のキーワードを基にプログラムを検索する技術も存在したが、専門家向けであった。必要なコンポーネントを「いかに簡単に指定する」ということは、今後ソフトウェア開発や分散コンピューティングにおいて主流となるSOA（Service Oriented Architecture）やSOC(Service Oriented Computing)においても重要となると考えられた。一方ビジュアル言語とは、プログラミングの作業を文字ベースではなく、絵、図形、色、アニメーション等、人間にとってより直観的に分かりやすいメディアを用いたプログラミング及びモデリング言語である。この分野では、ビジュアル言語を中心とした様々な研究成果が発表されていたが、ソフトウェア危機の現状を見ると、これまでに主流とされてきたデータフローまたはコントロールフローをベースとしているビジュアルプログラミングにおいても、この分野における飛躍的な貢献は見られなかった。

(3)サイバーフィルム技術による問題の解決：先行の研究としてサイバーフィルム（自己説明型コンポーネント）をプログラミング言語としたプログラミング環境に関する研究開発を行ってきた。サイバーフィルムは、アルゴリズム（プログラムやソフトウェアを含む）をより直観的・透過的・簡潔に記述することができ、さらに多くの分野のアルゴリズムにおいてその表現力が実証されているプログラミング言語である。この技術の哲学は、Open Set なユーザ指向ソフトウェアコンポーネント群から簡単に対象を取得し、目的のプログラムを直接的に作成するフレームワークを確立することであった。そのためには以下の項目の研究課題が挙げられた：

①新たなサイバーフィルムコンポーネント（アルゴリズム）の追加

②コンポーネントを蓄積するためのデータベースの構築

③プログラミング環境におけるデータベース要素を取得するための機能の実装

２．研究の目的

本研究の目的は、ソフトウェアコンポーネントのデータベース構築及び検索システムの開発であり、ソフトウェア開発及びコンピュータによる各種問題解決のプロセスを単純な検索操作に限りなく近づけ、プログラミングの非専門家を含む多くの人々が、計算処理や情報資源の生成に携わることを可能とする次世代のプログラミング環境を構築することであった。本研究は、ビジュアルモデリング、ビジュアルプログラミング言語で記述された新しいタイプのソフトウェアコンポーネントであるサイバーフィルムを用いたプログラミング環境の開発を目指している。

まず本プログラミング環境を構成する３

つの基礎概念である(1)サイバーフィルム（自己説明型ソフトウェアコンポーネント）、(2)サイバーフィルム言語、(3)開発環境、を説明する。(1)サイバーフィルムとは、ソフトウェアの効率的な理解・作成・管理を可能とする、既存のコンポーネント技術には見られない自己説明性を徹底的に追及したソフトウェアコンポーネントである。サイバーフィルムは、図形、色、動画、文字などのメディアから構成されるフレームの集合であり、それらはアルゴリズム及びソフトウェアの観点から、いくつかの「特徴」ごとに分類され管理されている。この「特徴」には主に、「計算の空間と処理の流れ（スキーム）」、「空間に定義された変数」、「スキーム上の計算式」、「入力」、「出力」、「統合ビュー」、が含まれる。これらの各「特徴」をそれに最も適した方法で表現・記述するものが(2)サイバーフィルム言語である（つまりサイバーフィルム言語はビジュアル言語の集合である）。(3)開発環境は各「特徴」専用のブラウザとエディタを提供し、状況に応じてコンポーネントの特徴を的確に組み合わせたビューを提供する。各「特徴」に応じたビジュアル言語を使用するため、アルゴリズムを抽象的な文字にたよらず直観的に表現しかつ直接的に記述することができる（ビジュアルシンボルの直接的な操作がプログラミングとなる）。またサイバーフィルムには C/C++/MPI などの既存のプログラミング言語で実装されたテンプレートが付随し、開発環境は上記「特徴」の記述・変更に応じて（ユーザがテンプレートに触れることなく）実行可能コードを自動生成する。

　これまでにサイバーフィルムのコンポーネント構造、サイバーフィルム言語の言語仕様、コード生成器を含んだ試験的な開発環境を確立した。さらに近年、サイバーフィルムによって多岐に渡るアルゴリズムをライブラリ化することによりサイバーフィルム言語の表現力を証明し、かつ大規模なユーザビリティテストでその理解・編集の容易さも実証した。しかし、本ライブラリのコンポーネントは本格的なデータベース化に至っておらず、コンポーネントやそれらの部品を効率的に取得する仕組みも本プログラミング環境へ実装されていなかった。さらには、本サイバーフィルム言語によるプログラムの編集作業においても、本データベースからのコンポーネントや適切なシンボルの取得は必須機能であった。サイバーフィルム言語の要素は数を限定しない(Open Set な)ビジュアルシンボルの集合であるため、データベースにコンポーネントを随時追加していくことが可能であり、本検索システムにより様々なユーザをサポートすることが目的であった。
研究目的の詳細は以下の通りである：

①　新たなアルゴリズムの分野に関するコンポーネントを追加する。コンピュータサイエンスに関するもの全般が対象であるが、研究期間では主にグラフアルゴリズム、物理シミュレーションや計算幾何学のアルゴリズムに焦点を置いた。
②　コンポーネントデータベースのスキーマ設計と専用データベースサーバの構築、及びＡＰＩの開発・提供。
③　コンポーネント及びそれらの部品を適時適切に取得するためのインタフェース及び関連言語の開発。
④　データベースのプログラミング環境への統合とプログラミングインタフェースの開発。

コンポーネントのデータベース化とそれらを取得する機能を備えた本プログラミング環境は、冒頭の研究目的を達成できる有望なプログラミングパラダイムを確立するものと期待された。さらに、検索システムを備えたデータベースがプログラミング環境のエディタでの操作をさらに簡易化(マウスによるアイコンの操作のみでシンボルが適時適切に選択され、プログラミングが可能)させることを目的とし、さらに将来的には SOC や分散コンピューティングの分野においても応用する目的があった。また、検索手法においては、コンポーネントの「What」における拡張された言語によって、コンポーネントのインタフェース（許可する入力と提供する出力）を厳格に定義し、よりセキュアで透過的なソフトウェアコンポーネントを提供することが目的であった。また、本ソフトウェアコンポーネントデータベースはソフトウェア開発や各種シミュレーション等をサポートするだけでなく、コンピュータ科学におけるアルゴリズム、数値計算用の教育資料やE-ラーニングシステムの基盤として応用する目的があった。

３．研究の方法

コンポーネント作成においては、書籍等を参考にし既存の言語（C, C++, あるいは FORTRAN）で書かれたプログラムを解析し、それらをサイバーフィルムの記述として変換しさらに対応したテンプレートコードを C++言語で作成した。コンポーネント開発は２年間を通して行われた。新たな分野におけ

るアルゴリズムをサイバーフィルムで記述するためには、本ビジュアル言語の拡張が必要となってくる。言語拡張においては、本言語の先駆者である Nikolay Mirenkov 特別栄誉教授の助言を得ることができた。新しい言語開発あるいは言語拡張において必須となるものがその形式的な言語仕様の提供であるが、ビジュアル言語に関してはテキストによるものと比べその形式的な言語仕様の提供は極めて難しいものとなる。本研究に関しても、サイバーフィルム言語仕様の形式的な記述に奮闘し、計画通りに研究を遂行できなかった経緯がある。しかし、この問題に関してはローマ大学の Paolo Bottoni 教授（ビジュアル言語の形式的記述を専門の１つとしているビジュアル言語の分野では著名な研究者である）との共同研究により、サイバーフィルム言語を形式的に記述するフレームワークを確立しているため、その形式的な言語仕様を拡張することができた。具体的には、UML ベースのメタモデルと OCL によって定義した。
作成されたコンポーネントの要素は、データベースの対応したレコードとして格納された。データベースの開発においては、サイバーフィルムが持つ属性を抽出し、エンティティや必要なリレーションを定義し、サイバーフィルムデータベースを設計した。サイバーフィルム本体及びそれらの部品は XML で定義し格納した。試験的なデータベースを構築するために、RDBMS(Relational Database Management System)として MySQL を採用した。設計が完了した後、これまでに開発されたライブラリ内のサイバーフィルムおよびそれらの部品をデータベースに格納するためのサブシステムを開発し、随時コンポーネントを追加した。本サブシステムでのサイバーフィルムの追加登録は単純なもので、かつライブラリは多岐に渡る膨大な数のアルゴリズムを含むため、ルーチンワークであるデータベースへの登録作業やテンプレートの作成は学生が行った。
本データベースをビジュアルプログラミング環境へと統合した。統合するにあたり、データベースの要素をクライアントシステムが取得するためのＡＰＩを開発し公開した。本 API は、サーバーサイドの Java アプリケーションとして実装した。クライアントシステムにおけるＧＵＩでは、まずデータベース内のコンポーネントを検索・取得するインタフェースのプロトタイプを設計・実装した。また、本ＧＵＩでは、上記ビジュアルプログラミング言語（サイバーフィルム言語）の構文に従い、適時その時必要とされるコンポーネントの部品が自動的に取得できるインタフェースを開発した。また、コンポーネントのみではなく付随するテンプレートの取得さらには一部のコンポーネントに対するプログラム自動生成及び実行を可能とした。クライアントサイドのインタフェースは、WEB で動作する Adobe Flash アプリケーションとしてそのプロトタイプを実装した。
データベース内のコンポーネントを検索するための新たなビジュアル言語の設計を行った。ソフトウェアコンポーネント（サイバーフィルム）が「どのような問題を解けるか（What)」を定義した「特徴」を記述するビジュアル言語（サイバーフィルム言語のサブ言語である）の言語仕様を確定した。What を記述する言語の具体的な構造を説明する。この言語はコンポーネントがどのような問題を解けるか（どのような機能を果たすか）をコンポーネントの入力におけるインタフェースと出力におけるインタフェースによって視覚的に表現する。これらのインタフェースは、２次元パネル上に配置されたローカルビュー（LV）、リージョナルビュー(RV)、グローバルビュー(GV)によって定義される（全てが必須ではない)。入力側の LV においてコンポーネントが許可するデータユニットのセットを定義し、出力側の LV においてコンポーネントが提供するデータユニットを定義する。各データユニットはデータの形、流れ、型、物理的・数学的な制約等を表すパラメタ付きのアイコンの集合で定義される。入力（出力）側の RV において入力（出力）側の LV で定義されたデータユニット間の関係(relation)を定義する。この'関係'には、データユニットが到着する（提供される）時間的な関係及び物理的・数学的な関係を含み、アイコンの集合やタイミング図によって定義される。GV では入力におけるデータユニットと出力におけるデータユニット間の関係を RV と同じ手法で定義する。言語仕様はビジュアル言語を定義するためのメタモデルと OCL(Object Constraint Language)及び自然言語によって構文・意味論を定義し、本言語の形式的な仕様書としてまとめた。さらに What に関するビジュアルシンボルやアイコンを作成した。しかし、本言語を用いた検索システムのプログラミング環境への統合には至らなかった。一方で、当初の予定にはなかったサイバーフィルムによるプログラムの編集作業において、データベースコンポーネントを適時自動的に取り出す仕組みを確立し、プログラム編集時におけるコンポーネント検索の一部を実装することができた。

４．研究成果

アルゴリズムの記述、シミュレーション、問題解決、ソフトウェア開発におけるプログラ

ミングをより直観的・透過的に行うためのサイバーフィルム及び対応するビジュアルプログラミング言語の普及を図るため、様々なアルゴリズムを本言語で記述し、言語要素をデータベースとして格納することが本研究の目的であった。研究期間では、グラフ・木構造におけるアルゴリズム、平面幾何アルゴリズム、物理シミュレーションにおける粒子運動に関するサイバーフィルムを開発し言語を拡張した。追加コンポーネントとユーザビリティテスト結果の分析により、本言語の表現力をさらに実証することができた。データベースの実装においては、スキーマ設計を基にデータベースサーバを構築した。サイバーフィルム、構造、計算の流れ、関数、式等のスキーマを定義し、これまで開発されたサイバーフィルムを表現するための部品を格納した。さらに本サーバをアプリケーションサーバに拡張し、外部システムがデーターベースの部品を取得するためのＡＰＩを開発し公開した。本ＡＰＩを利用し、データベース内のコンポーネントを取得するためのインタフェース、プログラムの部品であるビジュアルシンボルを適時取得し送信するためのサーバ及びそれらを言語構造に従って表示するためのクライアントインタフェース（エディタとなるもの）のプロトタイプを開発した。本インタフェースは WEB ブラウザ上で稼働するサービスとして実装され、サイバーフィルムの検索、観覧及び一部のアルゴリズムについては実行可能コードの取得が可能となった。これらデータベース、アプリケーションサーバ、及び各種インタフェースにより、サイバーフィルムへのアクセスがより容易となり、エンドユーザが WEB 上で最小限のタイピングでプログラミングを遂行することができる環境の基盤が得られた。さらに、本年度の成果は、サイバーフィルムのエディタ開発、データ可視化言語開発、並列アルゴリズムの分散処理サーバの構築等に関する応用研究へと繋げることができる。

５．主な発表論文等

（研究代表者、研究分担者及び連携研究者には下線）

〔雑誌論文〕（計２件）

Yutaka Watanobe, Nikolay N. Mirenkov, and Rentaro Yoshioka, Algorithm Library based on Algorithmic CyberFilms, *Journal on Knowledge-Based Systems*, 22, 195-208, 2009, Elsevier Publisher.

Yutaka Watanobe, Victor Malyshkin, Rentaro Yoshioka and Nikolay Mirenkov, Filmification of Methods: Representation of Particle-In-Cell Algorithms, V. Malyshkin (Ed.): 10th International Conference on Parallel Computing Technologies (PaCT-2009), LNCS 5698, pp. 360-376, 2009.

〔学会発表〕（計２件）

Yutaka Watanobe, Rentaro Yoshioka, Shuichiro Tsukumoto and Nikolay Mirenkov, Filmification of Methods: Convex Hull algorithms, In: Proceedings of IEEE TENCON 2009, pp. 1-6.

Rrentaro Yoshioka, Yutaka Watanobe, Nikolay Mirenkov, Incorporating Security into Software Development Process, In: Proceedings of the 7th International Conference on New Software Methodologies, Tools, and Techniques, pp.99-108, 2008.

６．研究組織

(1)研究代表者
渡部　有隆（Yutaka Watanobe）
会津大学 准教授
研究者番号：20800041