

科学研究費助成事業 研究成果報告書

平成 26 年 6 月 24 日現在

機関番号：23901

研究種目：基盤研究(B)

研究期間：2010～2013

課題番号：22300011

研究課題名(和文) ユーザ・デザイナー・エンジニア三位一体の自律進化型Webサイトの開発・実行環境

研究課題名(英文) Web development environment for evolutionary Web sites

研究代表者

山本 晋一郎 (Yamamoto, Shinichirou)

愛知県立大学・情報科学部・教授

研究者番号：40240098

交付決定額(研究期間全体)：(直接経費) 6,800,000円、(間接経費) 2,040,000円

研究成果の概要(和文)：Webサイトのデザインに関して、"レンダリング結果の抽象化に基づくWebサイトデザインの共通点抽出法"に関する研究を行い、複数のWebサイト群から共通するレンダリング構造を抽出することが可能となった。クライアントサイドにおけるソフトウェアに関して、"言語変換系を用いたソフトウェア開発におけるソースレベル開発支援"に関する研究を行い、サーバーサイドで動作するJavaScriptを高い抽象度でデバッグすることが可能となった。システムの主体となるサーバーサイドにおけるソフトウェアに関して、"ソフトウェアの概要把握のための動的解析結果の静的解析を用いた一元化"に関する研究を行った。

研究成果の概要(英文)：Web application mainly consists of three parts, server side programs, client side programs, and user interface facilities in Web page. And this research also divided into three parts as well. First, we have developed a tool to elicit the common HTML design skeleton from several Web pages. Second, in relation to software on client side, we developed source level debugging facility on Web browser for JavaScript program generated by Fay from Haskell subset. The last part of this research is about server side program. We have conducted research on the method to summarize dynamic analysis results using static analysis.

研究分野：総合領域

科研費の分科・細目：情報学・ソフトウェア

キーワード：Webアプリケーション Web開発環境

1 研究開始当初の背景

Web の発展と拡張に伴い、あらゆる人間活動に Web が用いられるため、従来からの機能性や信頼性に加えて、使用性が重要になりつつある。すなわち、ソフトウェアがコンテンツとしての側面を強く持つため、メニューの配置やページ内のレイアウトなどの情報の論理的な配置や見栄えなどの重要性が増している。

Amazon などの巨大ショッピングサイトは多様な機能を提供しているが、動的に生成されるコンテンツとしての側面も合わせ持っている。すなわち、機能だけがソフトウェアの価値を決めるのではなく、機能を提供する狭義のソフトウェアとそれが管理するコンテンツとの総和がソフトウェアの価値を表す時代が到来しつつある。そのため、ソフトウェアをエンジニアだけで開発することはできなくなり、デザイナーとの協調作業が重要増加して重要になる。また、コンテンツとしての側面が強くなるほど、開発時や運用時にユーザを取り込む必要性も増す。

しかし、従来のソフトウェア工学ではエンジニアとそのマネジメントに焦点が当てられていたため、デザイナーとエンジニアの協業や、運用段階におけるユーザとの連携に関する研究は進んでいない。

2 研究の目的

本研究の目的は、ユーザ、デザイナー、エンジニアの3者が連携することによって始めて実現できる、柔軟に自律進化可能な Web サイトの構築手法を確立することにある。

具体的には、(1) 開発時に行なわれるデザイナーとエンジニアの協調作業を支援するツールに関する研究、(2) 運用時にユーザがクライアントサイドで安全かつ簡単にカスタマイズするために必要となる機能の研究、(3) サーバサイ

ドにおけるソフトウェアの概要を把握して保守を支援する手法の研究を行なう。

3 研究の方法

Web サイトを構築する要素技術は、サーバサイドで動作するソフトウェア、クライアントサイドにおいてブラウザ上で動作するソフトウェア、利用者への直接的なインタフェースの3つに分けることができるため、本研究でもそれらに関する研究を行った。

デザイナーが主に関わる Web サイトのデザインに関して、“レンダリング結果の抽象化に基づく Web サイトデザインの共通点抽出法”に関する研究を行った。ユーザに密接に関係するクライアントサイドにおけるソフトウェアに関して、“言語変換系を用いたソフトウェア開発におけるソースレベル開発支援”に関する研究を行った。システムの主体となるサーバサイドにおけるソフトウェアに関して、“ソフトウェアの概要把握のための動的解析結果の静的解析を用いた一元化”に関する研究、および Web システムを平行システムとして捉えた場合の表現手法に関する研究を行った。また、開発の全ての工程において重要な役割を果たす文書化に関連して、Android を対象としたプラットフォームの学習支援に関する研究を行った。

4 研究成果

本研究の主な成果を以下に示す。

4.1 レンダリング結果の抽象化に基づく Web サイトデザインの共通点抽出法

開発対象 Web サイトと類似したデザインを備えた同業他社の Web サイトのデザインを分析し、その中に多く含まれるデザインの共通点を抽出する手法を提案した。抽出した共通点から、Web サイトの開発者が主観ではなく体系的な分析結果に基づいたデザインを提供するこ

とが可能となり，クライアントの要望に客観的な根拠を持って応える事が可能となる．

具体的には，既存の Web サイト群から，共通するレイアウトを抽出するツールを作成した．これにより，新たな Web サイトを制作する際に必要となるレイアウトの規範を得ることができる．このツールは，Web ブラウザのレンダリング結果を，各 DOM 要素のタグ名，id 名，Class 名，ページ上の座標と大きさからなるレンダリング情報として出力する．得られたレンダリング情報を，要素の包含関係と表示の位置関係を表すレンダリングツリーに変換して，異なる CMS を用いて作成された Web サイトのレイアウトにおける共通点と相違点を表すことが可能となった．

Web サイトを作成する際にユーザの使いやすさは重要であるが，既に多くの Web サイトが存在する中で，主観的に新たなデザインのサイトを作成しても，既存 Web サイトを使い慣れた利用者にとって必ずしも利用しやすいとは限らない．

本研究で提案する手法を用いることにより，複数の Web サイトの中に存在するデザインの共通点を抽出することができ，既存 Web サイト群に存在するデザインに関する暗黙の規則から逸脱することがない，いわゆる業界標準の Web サイトを作成することが可能となる．また，意図的に個性的な Web サイトを開発する場合でも，業界の共通点を分析し，そこからの逸脱の度合いを考えることが可能となる．

4.2 言語変換系を用いたソフトウェア開発におけるソースレベル開発支援

4.2.1 背景

プログラミング言語処理系の形態には，インタプリタとコンパイラの他に，他の高級言語へと変換するトランスレータ（以下，言語変換系とする）がある．特に最近では，Web ブラウザ

におけるクライアントサイドのプログラミング言語が JavaScript のみであるため，JavaScript の代替を目的とした言語変換系が多く開発されている．以下では，言語変換系を用いたプログラム開発に関して，プログラマが記述する S（ソース）プログラムと，言語変換系が出力する T（ターゲット）プログラムを考える．

プログラムの開発は，デバッグやプロファイリングなどプログラムの実行時の挙動を観察しつつ進められる．このときプログラマは S プログラムの意味論に基づくモデルを考えている．これに対して計算機上の実行は言語変換系が出力した T プログラムで行われる．T プログラムに対する実行時情報は S プログラムの意味モデルとは距離があり，それを用いた直接のデバッグは困難である点が言語変換系共通の問題である．

この問題に対応するため，JavaScript においては SourceMaps[1] が提案されている．これは S プログラムと T プログラムである JavaScript の行の対応を示すことでソースレベルデバッグを支援する．しかし SourceMaps は，JavaScript と意味モデルが近い種類の言語のみを対象としており，関数型のように意味モデルを異にする言語への適用は考えていない．

4.2.2 提案ツール

本研究では，遅延評価関数型言語 Haskell を S プログラムとし，手続き型言語 JavaScript を T プログラムとする言語変換系 Fay[2] を対象として，T プログラムの実行時情報から S プログラムの実行時情報を擬似的に生成する手法を提案し，その手法に基づくソースレベルのデバッグ支援環境を実現した．支援環境は Firefox 上の JavaScript デバッガである Firebug のプラグインとして実装した．

Fay が出力する JavaScript において，Haskell 中の変数名は同じ名前で現れ，関数

名は“モジュール名. 関数名”という形で現れるため、Haskell と JavaScript で対応したシンボルが現れる行番号の組を保持するテーブルを作成した。ここで双方向の対応を保持しておくことで、JavaScript の実行時情報を Haskell に対応付けることが可能となる。

このテーブルを用いて、実行中の JavaScript の行番号から対応する Haskell の行番号を求め、その行における JavaScript が持つ変数の値を取得する。しかし、取得した変数に関する情報は、Haskell では引数であるシンボルが JavaScript 中では局所変数として出現することや、Fay が内部的に使用する変数を含むことから、Haskell に直接対応していない。そのため、この情報を Haskell の対応する変数の値として表現することで Haskell の擬似的な実行時情報を生成した。

参考文献

- [1] Source Maps, <http://www.html5rocks.com/en/tutorials/developertools/sourcemaps>
- [2] Fay, <https://github.com/faylang/fay/wiki>

4.3 ソフトウェアの概要把握のための動的解析結果の静的解析を用いた一元化

ソフトウェアの開発・保守においてソフトウェア理解は重要な役割を果たしていることは言うまでもない。開発プロセスにおいて、開発者は使用するライブラリやソフトウェア部品を理解し、その挙動が目的に適しているのかを確認する必要がある。また、他のサブシステムを理解して、その整合性に気を配らなくてはならない。保守プロセスでは、プログラムの品質改善や機能拡張に先立って、保守対象のプログラムを理解することによりどのような機能を持ち、どのような振る舞いをするのかを把握しなければならない。

本研究では、ソフトウェア理解の初期段階においてソフトウェアの概要把握を行う開発者に対して、その手掛かりとして構造体型 (レコード型) を特徴付ける手法を提案した。

本研究の貢献は次の 2 点である。(1) 動的解析による複数の実行履歴を統一的に扱うために、静的解析を用いて時点軸という概念を提案した。それにより、複数の動的解析の結果を大局的に把握することが可能となった。(2) 2 つのソフトウェアを対象として、利用頻度に基づいて構造体型を特徴付けることにより、開発者はソフトウェアの機能を分類し、概要を把握することが可能であることを確認した。

5 主な発表論文等

〔雑誌論文〕(計 2 件)

1. 藤浦祥雅, 大久保弘崇, 粕谷英人, 山本晋一郎: プラットフォーム学習支援のためのサンプルアプリケーションを用いたドキュメントの関連部分抽出, 日本ソフトウェア科学会コンピュータソフトウェア, Vol.28, No.4, pp.358-370, 2011
2. 竹内亮太郎, 粕谷英人, 大久保弘崇, 山本晋一郎: モデル検査における妥当性確認を目的とした並行システムの表現手法 日本ソフトウェア科学会コンピュータソフトウェア, Vol.28, No.1, pp.293-299, 2011

〔学会発表〕(計 8 件)

1. Yoshihiko Hotta, Hiroataka Ohkubo, Hideto Kasuya, Shinichiro Yamamoto and Kunihiko Saito: Synergistic Use of both Static and Dynamic Information to Generate Reversed Uml Diagrams for Better Software Understanding, The 22nd International Symposium on

- Software Reliability Engineering
(ISSRE 2011), 2011
2. 竹治勲, 大久保弘崇, 粕谷英人,
山本晋一郎, 齋藤邦彦: 大規模ソフトウェアの概要把握支援のための動的解析結果の静的解析を用いた一元化, ソフトウェアエンジニアリングシンポジウム 2013 論文集, Vol.2013, pp.1-8, 2013
 3. 西村将広, 大久保弘崇, 粕谷英人,
山本晋一郎: 言語変換系を用いたソフトウェア開発におけるソースレベルデバッグ支援, 電子情報通信学会東海支部平成 25 年度卒業研究発表会論文集, p.28, 2014
 4. 丹羽一平, 大久保弘崇, 粕谷英人,
山本晋一郎: API の使用状況に基づく機能のグルーピングとそれを用いたソフトウェア自動分類, 研究報告ソフトウェア工学 (SE), Vol.2013-SE-179, No.3, pp.1-8, 2013
 5. 竹治勲, 大久保弘崇, 粕谷英人,
山本晋一郎, 齋藤 邦彦: 大規模ソフトウェアの概要把握の支援を目的とした構造体型の特徴付け, 研究報告ソフトウェア工学 (SE), Vol.2013-SE-179, No.2, pp.1-8, 2013
 6. 小川順平, 粕谷英人, 大久保弘崇,
山本晋一郎: テキスト値出現位置に着目した XML 文書のスキーマ変換器学習手法, 研究報告ソフトウェア工学 (PRO), Vol.2011-4-(5), pp.1-9, 2012
 7. 東祐太郎, 粕谷英人, 大久保弘崇,
山本晋一郎: Web サイト群の構造分析による典型的構造の抽出法, 第 10 回情報科学技術フォーラム (FIT2011) 講演論文集, Vol.1 B-024, pp.295-301, 2011
 8. 藤浦祥雅, 大久保弘崇, 粕谷英人,
山本晋一郎: サンプルプログラム自動生成

のためのリソースを考慮した依存グラフ,
研究報告ソフトウェア工学 (SE),
Vol.2011-SE-171, No.11, 2011

6 研究組織

(1) 研究代表者

山本 晋一郎 (YAMAMOTO Shinichirou)
愛知県立大学・情報科学部・教授
研究者番号: 40240098

(2) 研究分担者

粕谷 英人 (KASUYA Hideto)
愛知県立大学・情報科学部・講師
研究者番号: 10295579

大久保 弘崇 (OHKUBO Hiroataka)
愛知県立大学・情報科学部・講師
研究者番号: 40295580