

ファセットによるカテゴリを用いたソフトウェア検索システムの開発

田村 真吾 上田 賀一

茨城大学 工学部 情報工学科

近年、ソフトウェア開発の大規模化に伴い、ソフトウェア開発の効率向上のための再利用技術が注目されている。ソフトウェアの再利用のために、ソフトウェアリポジトリに格納されたソースコードや設計書といったソフトウェア成果物を検索するシステムが必要である。しかし、リポジトリには大量のソフトウェア成果物が格納されるため、成果物を体系的に整理しなければ、効率的な検索は難しいといえる。本研究では、機能や用途といった側面でファセット分類したソフトウェアに対して、Web インタフェースによるカテゴリ検索を行い、再利用を支援するソフトウェア検索システムを開発した。本システムを用いることでソフトウェアの管理・検索が容易になりソフトウェアの再利用を促進できると考える。

Software Retrieval System Using Categories of Facet Classification

Shingo Tamura and Yoshikazu Ueda

Ibaraki University

Recently, because software development becomes large-scale, the reuse technology for the efficiency improvement of software development is being watched. The system which retrieves the source code and the software products like design specifications stored in software repository is necessary to reuse software. But, an efficient retrieval is difficult if products aren't put in order systematically because large amount of software products are stored in repository. This research developed the software retrieval system to support reuse. The system does category retrieval through the web interface to the software that was classified by the facet of function and purpose etc. Software management and software retrieval become easy by using this system, and the reuse of software will be promoted.

1 はじめに

近年、ソフトウェア開発の大規模化に伴い、ソフトウェア開発の効率向上が求められおり、再利用性、拡張性に優れたオブジェクト指向開発が用いられている。オブジェクト指向の普及により、複数のオブジェクトから構成されるコンポーネントや設計における問題の解決法であるデザインパターン [8] などの再利用技術が注目されている。過去のソフトウェア資産を再利用することで、新規開発のコストを削減し、ソフトウェア開発の期間短縮、信頼性の向上に繋げることができる。

ソフトウェアの再利用を促進するツールとして、リポジトリがある。リポジトリにはソースコード

や設計書といったソフトウェア開発工程の成果物が格納・保存され、必要に応じて再利用される。ソフトウェア成果物の管理にリポジトリを有効活用することで、それらの再利用を促進することができる。しかし、リポジトリには大量のソフトウェア成果物が格納されるため、それらを格納するためのデータベースと、そのデータベースからソフトウェアを検索するシステムが必要になる。このデータベース内の成果物が体系的に整理されていなければ、目的のソフトウェアを効率良く検索することは困難になる。

例えば、一般の図書館では利用者が容易に検索できるよう、あらゆる図書が歴史、コンピュータといったカテゴリでジャンル分けされ、分類・整理

されている。ソフトウェアも同様に、対象世界（ドメイン）に関する情報や提供する機能などの付加的な情報を用いて分類を行える。しかし、図書館で用いられている分類法が複数の観点での分類を想定していないのに対して、ソフトウェアの分類の観点は機能や用途、適応分野など様々なものが考えられる。そのため、ソフトウェアが持つ側面を全て扱えるような分類体系を考える必要がある。

本研究では、ソフトウェア開発支援ツール（CASE）の成果物のうち、オブジェクト指向言語 Java のソースコードと、設計・分析のためのモデリング言語である UML[9] による設計仕様を対象とし、ソフトウェアの持つ機能や用途といった側面をファセット分類 [7] することで、ソフトウェア成果物を体系的に整理した。そして、Web インタフェースによるカテゴリ検索を行うことで、成果物の再利用を支援するソフトウェア検索システムを開発した。本報告では、ソフトウェアの検索・分類について紹介した後、提案するソフトウェア分類のためのファセットとカテゴリを用いた検索システムについて述べる。

2 ソフトウェアリポジトリ

ソフトウェアの再利用を促進するため、ソフトウェアリポジトリには要求を満たすソフトウェアを検索するシステムが必要になる。検索システムは、利用者の検索要求に対してリポジトリからその要求を満たすソフトウェアを探し出し検索結果として返す仕組みをもつ。本節では、ソフトウェアの検索手法とソフトウェアの分類体系について述べる。

2.1 ソフトウェアの検索手法

一般に自然言語文書の検索手法として、ユーザが要求する質問に適合する文書を結果として返すキーワードによる検索と、用意されたカテゴリを順に辿っていくことで検索対象の文書を絞り込むカテゴリによる検索がある。キーワードによる検索は適切な質問を与えると要求に合致する文書の検索を行える反面、検索者が適切な質問キーワードを与えないと要求に合致しない不要な文書を検索してしまう。一方カテゴリによる検索は質問キー

ワードを考慮することなく、カテゴリを選択することで段階的に文書の検索が行える。しかし、カテゴリの構成や文書のカテゴリへの分類が適切でないと検索性が低下するという欠点がある。

ソフトウェア文書では、ソフトウェアのメトリクスや構造情報から求められるソフトウェアの特徴からの検索も考えられる。このような検索はソフトウェアの特徴そのものが要求として与えられるため、確実に要求を満たすソフトウェア文書を検索することが可能であるが、専門知識を持っていることが前提となっており検索手法としては汎用性に欠ける。本研究では、ソフトウェアに対する深い専門知識を持っていない利用者も対象として考えているので、より直感的な検索が可能であるカテゴリ主導による検索システムを考える。

2.2 ソフトウェアの分類体系

多数のソフトウェアが格納されたりポジトリから要求を満たすソフトウェアを効率良く検索するためにはソフトウェアを体系的に分類し整理する必要がある。

SourceForge[11] などのソフトウェアリポジトリでは、ソフトウェアはあらかじめ定義されたカテゴリによって分類・整理されている。カテゴリによって、専門用語を知らなくてもソフトウェアを検索することができ、ソフトウェアが属するカテゴリの名称で検索しているソフトウェアの特徴を掴むことができる。

ソフトウェアの検索サービスには、Sourceforge、窓の杜 [12]、Vector[13] など様々なものがある。一般にこれらのサービスではソフトウェアはディレクトリによる階層構造で分類されている。しかし、ディレクトリはカテゴリへの分類を 1 つに限定する排他的な分類であるため、機能や用途、適応分野といった様々な側面を持つソフトウェアの適切なカテゴリへの分類が困難であり、結果として検索性を低下させてしまう。また、単一階層のディレクトリではソフトウェアの持つ側面を網羅することは難しく、ソフトウェアの分類に適していないと考えられる。

本研究では、情報をもつ属性や性質に対して様々な観点から多面的な分類を行うファセット分類の考えに基づいてソフトウェアの分類体系を考える。ファセット分類は多面的な分類であるため、複数

の階層を用いて表現される。ファセットによるカテゴリの分類・検索として次の点に考慮する必要がある。

- ソフトウェアの持つ側面を分類の観点であるファセットとして扱う。
- 各ファセットはカテゴリから構成され、互いに独立したカテゴリの階層構造を所有する。
- ソフトウェアの複数のカテゴリへの分類を許可する。
- 各観点からの検索を支援するインタフェースを提供する。

ソフトウェアがもつ様々な側面をファセット分類することで、ソフトウェアを体系的に整理し、カテゴリ検索を支援する Web インタフェースを提供することでソフトウェアの検索支援を行う。

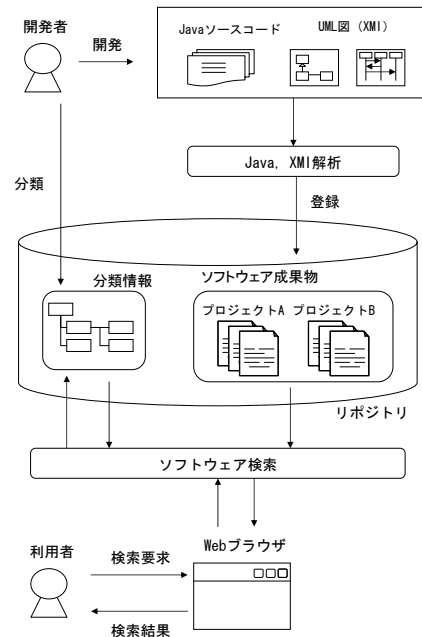


図 1: システムの概要図

3 提案するソフトウェアの検索システム

3.1 概要

本研究では、Java のソースコードと UML による設計仕様を対象として、これらのファイル内のクラスなどの情報をデータベースに格納し、ファセット分類を用いて整理されたソフトウェアを、カテゴリにより検索するシステムを提案する。

UML の情報入力には XMI (XML Metadata Interchange) [10] を用いた。本システムではソフトウェアの検索を支援するためカテゴリ検索のための Web インタフェースを提供する。システムの概要を図 1 に表す。

3.2 機能

システムの機能を以下に示す。

1. ソフトウェアの登録

Java, UML の入力形式としてそれぞれ Java ソースコード, XMI ファイルを対象とする。Java ソースコード, XMI ファイルを解析して得られるクラスなどの情報をリポジトリに格納する。

2. ソフトウェアの分類

登録されたソフトウェア成果物はプロジェクト単位で管理する。プロジェクトに対して Web ブラウザを用いて各ファセットごとにカテゴリへの分類を行う。プロジェクトがどのカテゴリに分類されているかといった情報もリポジトリに格納する。

3. ソフトウェアの検索

Web ブラウザを利用したファセットによるカテゴリ検索, キーワード検索を併用して利用者の要求に合うソフトウェアを絞り込み、登録されたソフトウェアの検索を行う。

3.3 管理するソフトウェア成果物

オブジェクト指向においてクラスは基本となる要素であり、再利用の対象として扱うことができるが、クラス単体で実現できる機能は限られており、一般に関係のあるクラス同士を組み合わせさせて再利用される。

しかし、関係のあるクラスやソースコード群を検索しても、クラスの情報やソースコードだけではソフトウェアの設計情報が不十分であるため、それらが要求を本当に満たすかどうか利用者が判断

することは難しい。ソフトウェアが大規模であるほど、その判断は困難になる。

よって、リポジトリに格納する設計成果物には下記の情報が必要になる。

- クラスの構造や、クラス間の継承関係や依存関係などソフトウェア全体の構造
- クラス間のメソッド呼び出し関係や、メソッドが呼び出される順番などソフトウェア全体の処理の流れ

前述の情報は UML のクラス図から、後述の情報はシーケンス図から取得することが可能である。Java ソースコードに加えて、設計成果物として UML を格納することで、ソフトウェアの理解を容易にし、ソフトウェアの再利用の判断材料にすることができる。

3.4 リポジトリ

リポジトリには Java ソースコードや XMI ファイルだけでなく、それらの内部のクラスを始めとした様々な情報を格納する。リポジトリには次の情報が格納される。

- 要素情報
Java ソースコードや UML 図に含まれるクラス、メソッドなどが持つ情報をファイルの要素として格納する。
- ファイル情報
登録された要素を含む Java ソースコード、XMI ファイルの名称やファイルパスの情報を格納する。
- 関係情報
クラス間の継承関係といった、要素間の利用関係や依存関係を格納する。
- 分類情報
登録したソフトウェアの側面であるファセットとそのカテゴリが格納される。この情報はソフトウェアの整理やカテゴリ検索に利用される。

3.5 ソフトウェアのファセット分類

ファセット分類と、ソフトウェアの分類に用いたファセットについて述べる。

3.5.1 ファセット分類

ファセット分類は情報をもつ属性や性質などの側面（ファセット）に注目し、様々な観点から情報を多面的に分類する手法である。具体的には情報が持つ様々な側面を取り出し、各側面に対して適切な語を当てはめることでカテゴリを作成し分類を行う。ファセット分類は柔軟性があり、様々な側面を持つソフトウェアの分類に適しているといえる。ファセット分類は様々なサイト [14][15] で対象となる情報の分類に使用されている。

3.5.2 提案したファセット

ソフトウェアの複数の観点からの分類のために、ソフトウェアの分類に用いるファセットを考える。各ファセットは相互に独立していて、ソフトウェアの持つ側面とその側面を表現するカテゴリを扱う。このカテゴリを用いて各観点からソフトウェアを分類することで、ソフトウェアの適切な分類を行える。しかし、扱うファセットの個数が多すぎると分類の観点が不明確になり、検索性が低下する可能性がある。そのため扱うファセットの個数は 8 から 10 個程が好ましいと考える。

ソフトウェアの分類のために提案したファセットを次に示す。

1. Purpose（目的）

このファセットではソフトウェアの利用目的を扱う。一般にソフトウェア検索サービスの利用者は文書作成、表計算、ネットワークといった目的に応じたソフトウェアを検索するケースが多いと思われるので、このファセットのカテゴリは重要であると考ええる。

2. Type（種類）

ソフトウェアには GUI アプリケーションだけでなく、Web アプリケーションやライブラリ、プラグインなど様々な種類がある。このファセットではソフトウェアの種類によるカテゴリを扱う。

3. Technology (利用技術)

ソフトウェアは様々なプログラミング言語やライブラリを用いて実装される。このファセットではソフトウェアの実装に用いたプログラミング言語、ライブラリ、フレームワークなどのカテゴリを扱う。ソフトウェアの利用技術はパフォーマンスや環境に影響を与えるので、検索の観点として重要である。

4. Hierarchical Classification (階層分類)

ソフトウェアは階層構造で表現できる。階層による分類では基本ソフトウェア、ミドルウェア、応用ソフトウェアといったカテゴリがある。

5. OS (オペレーティングシステム)

ソフトウェアの実行環境である OS を扱うファセット。

6. Data Type (データの種別)

ソフトウェアが主として扱うデータの種別に対するファセット。例として、数値データ、文字データ、画像、動画といったカテゴリが挙げられる。

7. Domain (対象分野)

このファセットはソフトウェアが利用される分野を扱う。分野には販売管理、在庫管理といった各業務、製造、IT 通信などの各業種、科学・工学といったカテゴリが考えられる。

8. User (ソフトウェア利用者)

ソフトウェアが利用対象とするユーザによる分類を扱うファセット。ソフトウェア開発者、エンドユーザといったカテゴリがある。

これらのファセットを用いたソフトウェアの分類例を図 2 に示す。このようにソフトウェアをファセットで分類することで、成果物の管理が容易になり、ソフトウェアの検索時には、用意されたファセットからカテゴリを選択していくことで、ソフトウェアの検索を行える。

4 システムの開発

本節ではシステム内で扱う要素のクラスや、ソフトウェアの登録・分類・検索の流れを示す。

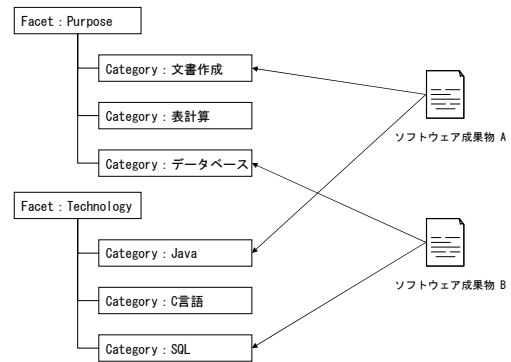


図 2: ソフトウェアの分類例

4.1 クラス構成

データベースに格納される要素のクラスを図 3 に示す。各ファセットのカテゴリは階層的に表すことができ、登録したプロジェクトに対して複数のカテゴリへの分類を行える。クラス間の継承や型の参照といった要素間の関係情報、カテゴリに属するプロジェクト、プロジェクト内のファイルといった要素間の所有関係の情報は図 4 に示すように Relationship クラス、Ownership クラスを用いて管理する。

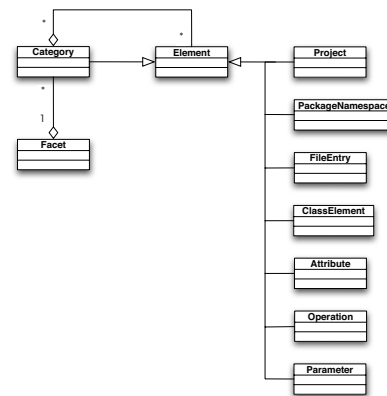


図 3: データベースに格納される要素のクラス構成

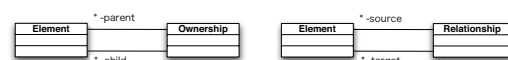


図 4: 関係情報を扱うクラス

4.2 登録・分類

以下にプロジェクトとその成果物の登録・分類の流れを示す。

1. プロジェクトの登録

ソフトウェア成果物を管理するためのプロジェクトを登録する。プロジェクト名や管理者などの情報の入力を行う。

2. プロジェクトの分類

プロジェクトに対して、ファセットとそのカテゴリ群からカテゴリを選択し分類を行う。カテゴリについては前節で示したものを利用する。利用者は Web ブラウザから用意されたカテゴリを選択することで、分類を行える。

3. ファイルの登録

プロジェクトの成果物となる Java ソースコード、XMI ファイルを登録する。Web ブラウザを介してアップロードされたファイルを解析しデータベースに格納する。

4.3 検索

検索にはカテゴリ検索とキーワード検索がある。

● キーワード検索

キーワード検索では、ソフトウェアの登録時に解析したクラスなどの要素名を索引語とし、ユーザが入力した質問文（キーワード）にマッチするプロジェクト、ファイルを抽出して検索結果をリスト表示する。

● カテゴリ検索

各ファセットに用意されたカテゴリを選択することで、カテゴリに属するソフトウェアを抽出して検索結果をリスト表示する。カテゴリを組み合わせて選択することで検索対象の絞込みを行い、条件に合ったソフトウェアの検索ができる。

カテゴリ検索とキーワード検索を併用することでソフトウェアの絞込みを行える。以下に検索の流れを示す。

1. 検索の開始

検索の開始時に利用者はファセットのカテゴリを選択することで、ソフトウェアの検索を開始する。検索開始画面を図 5 に示す。

2. ソフトウェア検索

検索画面ではカテゴリ、およびキーワードを入力することでソフトウェアプロジェクトの検索を行う。“Web アプリケーション” カテゴリに属するソフトウェアの検索結果を図 6 に示す。次に、画面左のカテゴリのリストから“在庫管理”カテゴリを選択し、“Web アプリケーション”、“在庫管理”の2つのカテゴリに属するソフトウェアを検索した結果を図 7 に示す。このように利用者がカテゴリを選択することで、検索カテゴリのフィルタが追加され、検索対象を絞り込むことができる。

3. パッケージブラウザ

検索したプロジェクトの画面から、プロジェクトに登録されているクラスの一覧を取得することができる。

4. ファイルの取得

利用者は任意のクラスを選択してファイルをダウンロードすることができる。



図 5: 検索開始画面

4.4 実装技術

システムの実装は Java 言語をベースに行った。リポジトリ内のデータベースには PostgreSQL を、ソフトウェアの分類部、検索部には JSP と Java



図 6: “Web アプリケーション” カテゴリに属するソフトウェアの検索結果



図 7: “Web アプリケーション”, “在庫管理” カテゴリに属するソフトウェアの検索結果

Servlet を用いた。Java ソースコードの解析には JavaCC を、XMI ファイルの解析には DOM を利用した。

5 関連研究

この節では関連研究としてファセット分類、ソフトウェア検索に関する研究を紹介する。

5.1 ファセット分類を用いた研究

ファセット分類をソフトウェア工学に応用している研究を紹介する。

久保田らはアプリケーション、コンポーネントの分類のためのファセットを提案し、ツールによる分類支援を行うことで、リポジトリの構築支援

を行っている [2]。塚本らは要求記述から生成される要求モデルを始めとしたドメインモデルをファセットによって分類することで、各ドメインの特徴付けを行い、要求分析・要求獲得段階での要求モデルの再利用性を高めている [1]。しかし、これらの研究では分類し、整理したソフトウェアの具体的な検索方法については述べられていない。それに対し本研究では検索手法としてカテゴリ検索を扱い、検索のためのインタフェースを提供することで、ソフトウェアの管理・検索を容易にし、再利用支援を行っている。

5.2 ソフトウェア検索に関する研究

本研究では、ソフトウェアが持つ特性や側面からソフトウェアの分類体系を考え、カテゴリによる検索を支援するシステムを開発した。ここではソフトウェア検索に関する研究について述べる。

Java ソースコードを対象としたソフトウェアの検索システムとして SPARS[3]がある。SPARSではソースコードを解析して得られる利用関係、依存関係や、ユーザーが入力したキーワードとソースコードの索引語との適合度から、検索結果の順位付けを行っている。また鷲崎らは、入力とした Java ソースコードの依存関係からコンポーネントを抽出し、キーワード検索を行うことで、コンポーネントの再利用支援を行っている [6]。一方亀井らは、WWW を対象にソフトウェア資源の収集を行い、収集したソフトウェアから測定したメトリクスを用いた検索システムを提供している [5]。

これらの研究では、ソースコードから取得できる情報を基に検索を行っており、本研究とは検索のためのアプローチが異なる。今後はファセットによるカテゴリ検索だけでなく、ソースコードや UML 図の情報を活用することで他の検索手法との併用を検討する。

また、検索システムではないが、ソフトウェアの分類に関する研究として MUDABlue[4]がある。MUDABlueでは、ソースコード内の関数名や変数名から識別子を取得し、類似した識別子を持つ文書の集合を作成することでソフトウェア文書の自動分類を実現している。しかし、解析対象がソースコードだけであるため、ソフトウェアの持つ情報を十分に取得できない、利用者にとって解釈しづらいカテゴリ名を持つ集合が作られる可能性が

あるという問題もある。

6 おわりに

本研究では、ソフトウェアの再利用促進を目的に、ソフトウェア検索システムの開発を行った。

Java ソースコードと UML の設計仕様を管理するデータベースの設計を行い、管理しているソフトウェアの検索支援のために、ソフトウェアをファセット分類し、ファセットに基づいたカテゴリによる検索システムを開発した。ファセットを基に設計した Web インタフェースを用いてカテゴリを選択していくことで、ソフトウェアが持つ側面からの検索が容易に行えることを確認できた。

今後の課題として、検索インタフェースの改善によるソフトウェアの検索性の向上や、UML を対象にメトリクスを測定することで UML 図の評価を行い、検索したソフトウェアが再利用可能かどうかの情報を利用者に提供することで、ソフトウェアの再利用支援を行うことが挙げられる。また、システム内の分類体系の編集機能を追加することで、様々なドメインへの対応が可能になり、システムの有効性を向上できると考える。

参考文献

- [1] 塚本 明, 小飼 敬, 上田 賀一: ドメイン情報を利用した要求モデルの構築支援, 情報処理学会, 研究報告 (SE), Vol.2003, No.22, pp.23-30 (2003 年 3 月)
- [2] 上田 賀一, 久保田 深雪: ソフトウェアリポジトリ構築のためのファセット分類アプローチ, 茨城大学工学部研究集報, Vol.49, No.1, pp.11-19 (2002)
- [3] 横森 励士, 梅森 文彰, 西 秀雄, 山本 哲男, 松下 誠, 楠本 真二, 井上 克郎: Java ソフトウェア部品検索システム SPARS-J, 電子情報通信学会論文誌, Vol.J87-D1, No.12, pp.1060-1068 (2004 年 12 月)
- [4] 川口 真司, パンカジ ガーグ, 松下 誠, 井上 克郎: MUDABlue: ソフトウェアリポジトリ自動分類システム, 電子情報通信学会論文誌,

Vol.J88-D1, No.8, pp.1217-1225 (2005 年 8 月)

- [5] 亀井 俊之, 門田 暁人, 松本 健一: WWW を対象としたソフトウェア検索エンジンの構築, 電子情報通信学会技術報告, ソフトウェアサイエンス研究会, SS2002-47, pp.59-64 (2003 年 1 月)
- [6] 鷺崎, 深澤: オブジェクト指向プログラムのためのコンポーネント抽出型検索システム, オブジェクト指向 2003 シンポジウム (OO2003) (2003) 仕様書
- [7] Louis Rosenfeld, Peter Morville: Web 情報アーキテクチャ, O'REILLY (2005)
- [8] 結城 浩: Java 言語で学ぶデザインパターン入門 (2004)
- [9] UML: <http://www.uml.org/>
- [10] XMI: <http://www.omg.org/technology/documents/formal/xmi.htm>
- [11] SourceForge.jp: <http://sourceforge.jp/>
- [12] 窓の杜: <http://www.forest.impress.co.jp/>
- [13] Vector: <http://www.vector.co.jp/>
- [14] Wine.com: <http://www.wine.com/>
- [15] Epicurious: <http://www.epicurious.com/>