

C++言語向け母体理解支援ツールの開発（1）

7N-8

- ツール概要 -

坂本俊行* 梶原茂喜* 佃軍治** 江良倫夫*** 木戸口普方***

* (株)日立製作所ソフトウェア開発本部 ** (株)日立製作所システム開発研究所 *** 日立東北ソフトウェア(株)

1. はじめに

近年オブジェクト指向言語で記述されたソフトウェアを母体とする開発が増加している。筆者らは、この開発に必要な母体ソフトウェアの理解を支援する母体理解支援ツールを開発した。本ツールはC++のソースコードを静的に解析し、その情報を整理して解析情報リポジトリに保存する。その結果をもとに、母体理解作業に対応したドキュメントを生成する（図1）。ドキュメントは、クラス/メンバ関数に着目したプログラムの初期的な理解作業や、クラス/メンバ関数のプログラム内でのソースコードの変更による影響範囲の探索など、母体理解時の作業内容に応じて段階的に利用でき、これにより作業負荷が軽減できる。

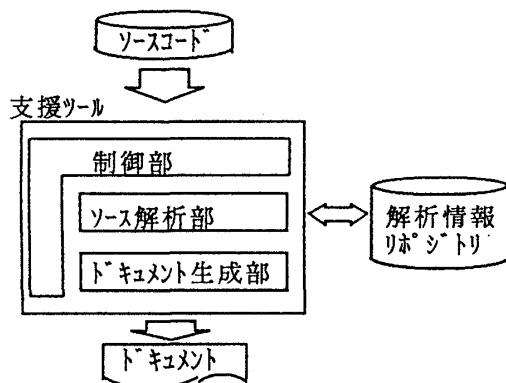


図1. システム構成概略

2. ツールの概要

ツールは以下の2つの部分から構成する。

(1) ソース解析部

C++ソースコードを解析し、各クラスの定義情報を抽出して整理し、各クラス毎に解析情報リポ

Development of Software Basic Structure Analysis Tool for C++ Language(1)

- Tool Overview -

Toshiyuki SAKAMOTO, Shigeki KAJIWARA, Norio Era, Hironori KIDOGUCHI, Gunji TSUKUDA

* Software Development Center, Hitachi, Ltd.

** Systems Development Laboratory, Hitachi, Ltd.

***Hitachi Tohoku Software, Ltd.

ジトリへ格納する。解析情報リポジトリは、各クラスの名称、記述されているファイル名、データメンバの名称/型、メンバ関数の名称/型、クラス継承情報、メッセージ情報などを保持している。

(2) ドキュメント出力部

生成された解析情報リポジトリから作業内容に合致したドキュメントを表示する。指定により部分的な解析/表示も可能とする。

3. 生成ドキュメント

母体理解作業で段階的に利用されることを想定し、全体構造から詳細構造までの理解を支援する次のドキュメントを生成することにした。

(1) 全体構成の理解

第一段階として、まずシステム全体の構成を理解するために、以下のドキュメントを生成する。

- クラス一覧表/クラス・メンバ関数一覧表
プログラム内のクラス一覧、メンバ関数一覧を表示する。
- クラス継承図
クラス継承関係を木構造で表した図を表示する。

(2) クラス構造の理解

次に、クラス一つ一つの詳細を理解するために、以下のドキュメントを生成する。

- クラス仕様表
個々のクラスの概要、基本クラス情報、データメンバ情報、メンバ関数情報など、定義情報を記述した仕様表を表示する。
- メンバ関数仕様表
個々のメンバ関数の概要、所属クラス名、引数情報など、定義情報を記述した仕様表を表示する。
- クラスインタフェース図
個々のクラスのメンバの種別、アクセス属性、引数関係、基本クラスなど、構造やインタフェースを表した図を表示する（図2）。クラス仕様表と同等の情報を視覚的に表現した図である。

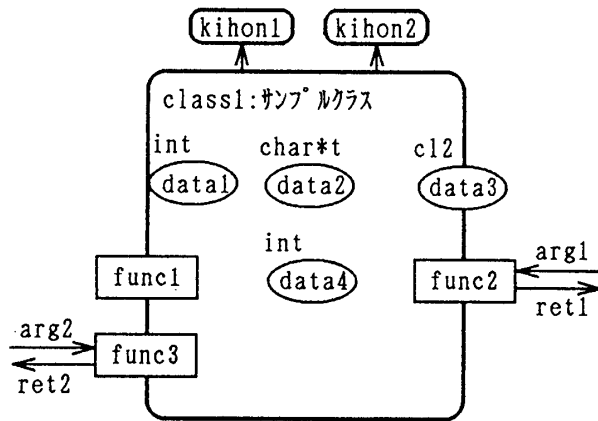


図2. クラスインタフェース図

(3) 変更時の影響範囲の把握

さらに、ソースコード変更時の影響範囲を把握するために、オブジェクト指向設計に特有の継承、多態性、メッセージパッシングなどを考慮した以下のドキュメントを生成する。

■ クラス構成図

クラスのソースコードをフォローしていく場合、継承している上位クラスの情報ソース上に分散しているため、利用できるメンバなどの情報が把握し難い。これを解決するために、継承している上位クラスも含めて、対象クラスの中から使用できるデータメンバ、メンバ関数に着目して、クラスの構成を表した図を表示する(図3)。クラス自身の修正やクラス使用部分の修正時、その影響範囲の把握に利用する。

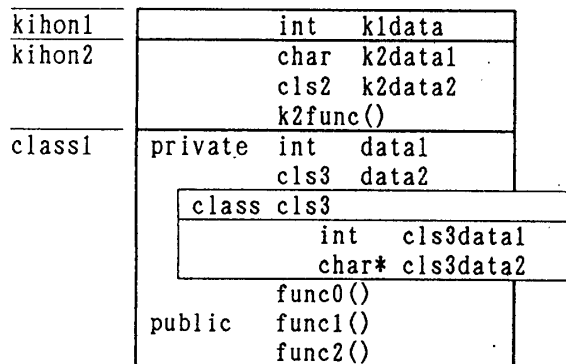


図3. クラス構成図

■ 仮想関数継承図

仮想関数の動作をフォローしていく場合、複数個存在する定義の中からどの定義を選択するかを見極めるのは、手間のかかる作業である。この見極めを迅速に行うために、仮想関数宣言クラスと定義クラスの間を木構造で表した継承図を表示する(図4)。これにより、仮想関

数実行時に使用される「仮想関数の定義部」の探索を行う。

var_func(arg1)

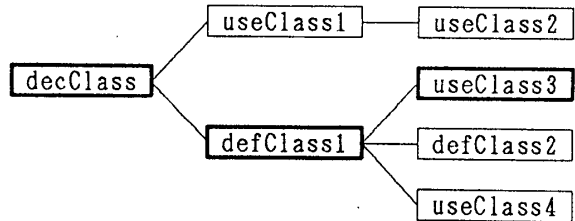


図4. 仮想関数継承図

■ メッセージフロー図^[1]

メッセージとメッセージ送信クラス、受信クラスにより、クラス間のメッセージパッシング関連を表した図を表示する。メンバ関数に着目した全体構造の把握、および各メッセージが影響を与える範囲の把握に利用する。

■ ソース情報付加リスト

C++ソースコード上のクラスインスタンス宣言など各種定義使用部分に、ファイル名、定義開始行などその定義の参照情報を付加したリストを表示する。ソースコードの詳細な理解や、影響範囲の把握に利用する。

4. まとめ

以上、ソースコードから母体理解の作業内容に応じた各種ドキュメントを生成するツールを開発した。詳細仕様設計以降の工程でのリエンジニアリングを行うために、クラス仕様書など比較的上流工程で利用されるドキュメントから、ソースコードフォローのナビゲーションを行うものまでを支援した。

今後、本ツールによる生産性向上に対する評価を行って行く予定である。また、今回はソースコードの静的解析を行ったが、多態性を考慮した動的な解析も取り入れ、影響範囲の把握を十分に支援していく必要がある。

参考文献

[1] 佃 他, C++言語向け開発母体理解支援ツールの開発(2) -メッセージフロー図逆生成ツールの開発と評価-, 情報処理学会第49回全国大会論文集, 1994