

## 8Q-05 Web ブラウザを用いたソースプログラムを読む支援システムの設計と実現

河津俊之、並木美太郎  
(東京農工大学)

### 1. はじめに

近年、情報技術の発展がめざましくソフトウェアの大規模化、複雑化により、従来のソフトウェア開発では情報技術革新のスピードに対応できなくなってきた。スピード重視の時代の中で注目されはじめたのが、オープンソースである。一般的に、オープンソースを再利用してシステムを作るとリリースを頻繁に行える。オープンソースを再利用するためには、そのソースプログラムを読み、再利用できるかまたは再利用するのに適したプログラムであるかを調べなくてはならない。

そこで、本研究ではソースプログラムを読むことに着目して、ユーザをできるだけ制限せずに簡単にソースプログラムが読めるように、プログラムを読む支援システムの設計と実現（以下、本システム）を行う。

### 2. ソースプログラムを読むことの問題点

#### (1) プログラムの流れが分かりにくい

関数の構造を理解するにはソースプログラムのままでは分かりにくい。

#### (2) 開発者が読むプログラムの移動量が多い

分からない識別子を調べることなどプログラムを読む際に視点の移動量が多くなる。

#### (3) コメント支援がない

プログラムにコメントを書くと他人のコメントと区別がつかない。

#### (4) プラットフォームに依存する

プログラムを読む際に、開発者は環境ごとにプログラムを読むツールを変えて使用しなければならない。

### 3. 目的

本システムのプログラムの読み方として次の二つを支援する。

#### (1) プログラムの流れを読む

#### (2) プログラムを一行一行に読む

問題点とこの二つの読み方から本システムでは次の

#### (a) ~ (d) の支援を行う。

#### (a) 関数ツリー機能

#### (b) 識別子の呼出し部から定義部への相互参照機能

#### (c) コメント挿入機能

#### (d) Web ブラウザを用いてプログラムを表示することにより、プラットフォーム独立とする

### 4. 対象ユーザ

本システムの対象言語は言語 C とする。また、対象とする規模は研究室などの LAN 内で 10 人程度で使うことを想定している。

### 5. 機能設計

本システムはブラウザによりソースプログラムを表示する。実行画面は次の図 1 のようになる。



図 1 本システムの実行画面

次に、本システムの主な機能について説明する。

### (1) 相互参照機能

図 1 の左側にある四つのフレームに表示されている識別子を指定すると、その指定された識別子の詳しい情報が図 1 の右のフレームに表示される。サーベで調査した GLOBAL というソース読解支援ツールでは、関数だけ相互参照のためのハイパーリンクを張っている。しかし、本システムでは、関数に加えて変数、構造体、enum、define、typedef についても支援を行う。

プログラム部の識別子からリンクが張られている場所を指定するとその識別子の定義部が右フレームに表示される。これは実際のプログラム実行と同じ動作であり、開発者に自然な流れでプログラムを読むことを支援する。

### (2) 関数ツリー機能

図 2 のような関数のツリー構造を表示してプログラムの流れを分かりやすく開発者に提供する。

```
cmm.c(16)      :main()
cmm.c(20)      :- printf()
cmm.c(21)      :- exit()
cmm.c(23)      :- strcmp()
cmm.c(27)      :- fopen()
cmm.c(28)      :- fopen()
cmm.c(29)      :- printf()
cmm.c(30)      :- exit()
cmm.c(32)      :-
lex.c(148)     :- init token table()
cmm.c(34)      :-
lex.c(61)      :- getsym()
lex.c(67)      :-
```

図 2 関数ツリー表示

### (3) コメント挿入機能

コメントは、各ファイルに表示されているコロンマークを指定することにより、任意の位置に挿入できるようにする (図 3 参照)。コメントは簡潔にプログラムに書くことが一般的であるが、簡潔なコメントから詳細なコメントへリンクを張ることができるようにする。

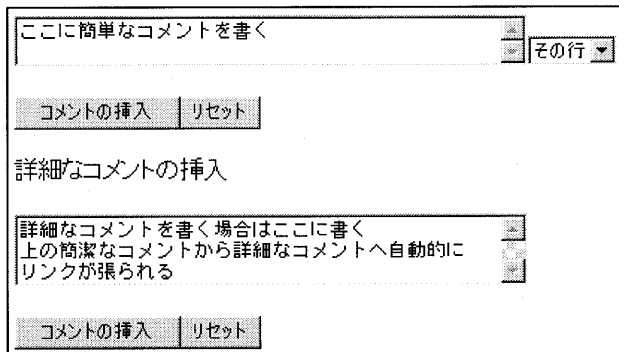


図 3 コメント挿入画面

## 6. システム構成

本システムは、サーバにソースプログラムや HTML ファイルを保存し、そのサーバにあるファイルをクライアント側のブラウザを用いて読む。言語 C のプログラムは次の図 4 のような過程で HTML ファイルを生成する。

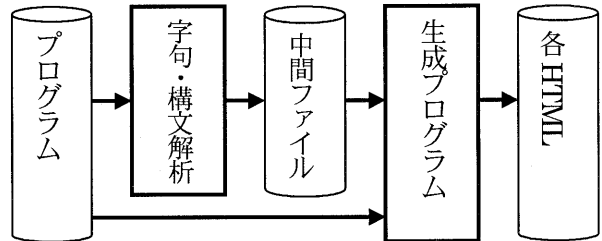


図 4 各 HTML ファイル生成の過程

## 7. ハイパーリンクの設計

本システムのハイパーリンクは次の図 5 のように張られる。

まず、図 1 の左側にある四つのフレームを意味する各インデックスファイルからは詳細な情報を調べるための各情報ファイルへのリンクが張られている。情報ファイルには詳細な情報が書かれているので、多くのリンクが張られている。

そして、識別子の呼出し部からは、定義部へのリンクがある。この呼出し部から定義部へのリンクがプログラムの実行順序に読むためのリンクである。

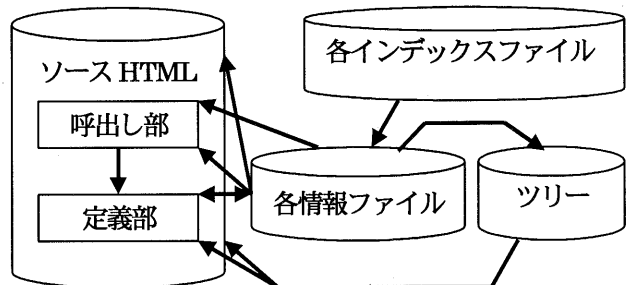


図 5 ハイパーリンクについて

## 8. 実現

字句解析、構文解析部は yacc/lex を使い、ANSI 規格の言語 C を解析できるようにした。本システムのサーバ部分は Linux で実現した。

## 9. 終わりに

本システムによりプラットフォーム独立な読む支援が行える。そして、言語 C のプログラムの流れや意味不明な識別子を調べることが簡単になる。