

## 5M-9

## プログラム構成図をベースとしたプログラム開発環境

上田 裕一, 笠原 道治, 渡邊 豊  
( 富士通株式会社 )

## 1. はじめに

プログラム開発環境は、各種ツールから構成される。最近では便利なツールが数多く提供されるようになったが、プログラマがそれらツールを使いこなすことは容易でない。我々は、プログラマが、各ツールを意識することなく、プログラム開発の一連の作業を自然な操作ビューで進めることができる環境を構築したので紹介する。

## 2. 概要

プログラム開発作業には各種作業があるが、図1に示すように大きくは4つの作業があり、それらはパターン化している。従来、プログラマは、これらの作業に合わせて各ツールを使い分ける必要があり、ツールを意識せざるを得なかった。近年マルチウィンドウ・システム上では、各ツールに別々のウィンドウを割当て、一つのディスプレイ上で操作できるようにはなってきたが、ツールを意識していることには変わりはない。

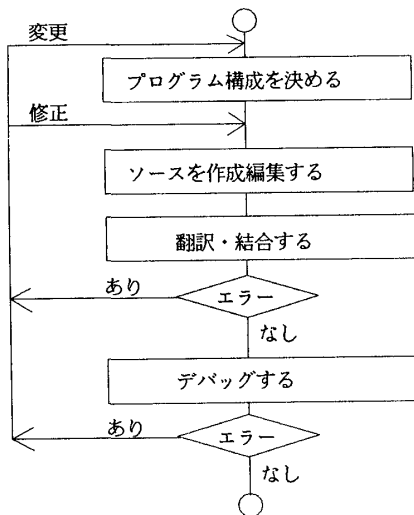


図1 プログラム開発作業の流れ

我々が今回構築した環境は、プログラム構成図から、構成定義～編集～翻訳結合～デバッグの作業を、ツールを意識せずに、自然な操作ビューでプログラム開発ができるCプログラマ向けの環境——「Cプログラマブラウザ環境」と呼ぶ——である。

## Cプログラマブラウザ環境の特徴：

## 1) プログラム構成定義

まずプログラム構成を定義する。構成定義がないと、Cプログラマブラウザがプログラマに対して構成定義作業を促す。以降プログラマは、このプログラム構成図をベースに開発作業を進めることになる。構成の変更も容易にできる。

## 2) 翻訳結合情報の設定

翻訳や結合のための情報は、翻訳時や結合時に設定するのではなく、プログラムの情報として事前に設定できる。

翻訳情報は、モジュールに対して設定する。但し、モジュール間で共通な情報は、各モジュールに設定する必要はない。結合情報は、プログラムに対して設定する。情報の変更は、容易にできる。

## 3) ソース作成編集支援

翻訳エラーが発生した場合は、エラーが発生した行を明示し、診断メッセージと連携して容易にソース修正ができる。

使用するエディタを選択できる。

## 4) 自動翻訳結合支援

Cプログラマブラウザでは、自動翻訳結合処理が標準である。これは、ソースの修正は勿論、プログラム構成の変更、翻訳結合情報の変更も自動処理の対象である。

翻訳結合の結果は、プログラム構成図のモジュール単位に、「○×表示」でわかり易くしている。

## 5) デバッグ支援

ソーストレース機能に加え、モジュール単位及び関数単位のトレースができる。モジュール/関数トレースの画面では、通常のプログラム構成図にモジュール毎に関数名が付加されて表示される。

これ以外に、テストカバレッジ、ライブラリ作成編集、プログラム管理などの豊富な機能も持っている。

また、Y P S (YAC II Programming System) もサポートしている。

3. 効果

1) 操作性の大幅な向上

- ・プログラム構成図から一連の作業ができる
- ・操作方法が自然でツールを全く意識しなくて済む

2) ミスの大幅な削減

- ・構成変更の同期が自動的にとれる
- ・修正忘れや翻訳忘れ等のミスが防止できる

3) 良い開発手順の自然な体得

例えば、プログラム構成をきちんと定義しないと他の作業ができないため、先ずプログラム構成を決めるようになる等、経験が浅いプログラマは、基本的なプログラム開発作業の仕方（ノウハウ）が自然に体得できる。ベテランのプログラマも、プログラムそのものの作成により専念できる。

4. おわりに

Cプログラマブラウザ環境は、現在、社内で実際の開発や、新人教育に適用している。全体的には概ね好評で、基本の操作ビューはかなり良い評価を得ており、利用者が増加している。プログラム開発のベースとなる操作ビューの有効性は確認できたと考える。

今回の取組みに当っては、開発環境全体の操作ビュー（使い勝手）を徹底重視した。プログラマの作業モデルを設定し、“より自然な操作でプログラム開発に専念できる”環境の構築を目指した。今後は、サポート範囲を広げ、ソフトウェア・エンジニア向けに、“より自然でより強力な”支援環境の構築を目指して、課題に取り組んでいきたいと考えている。

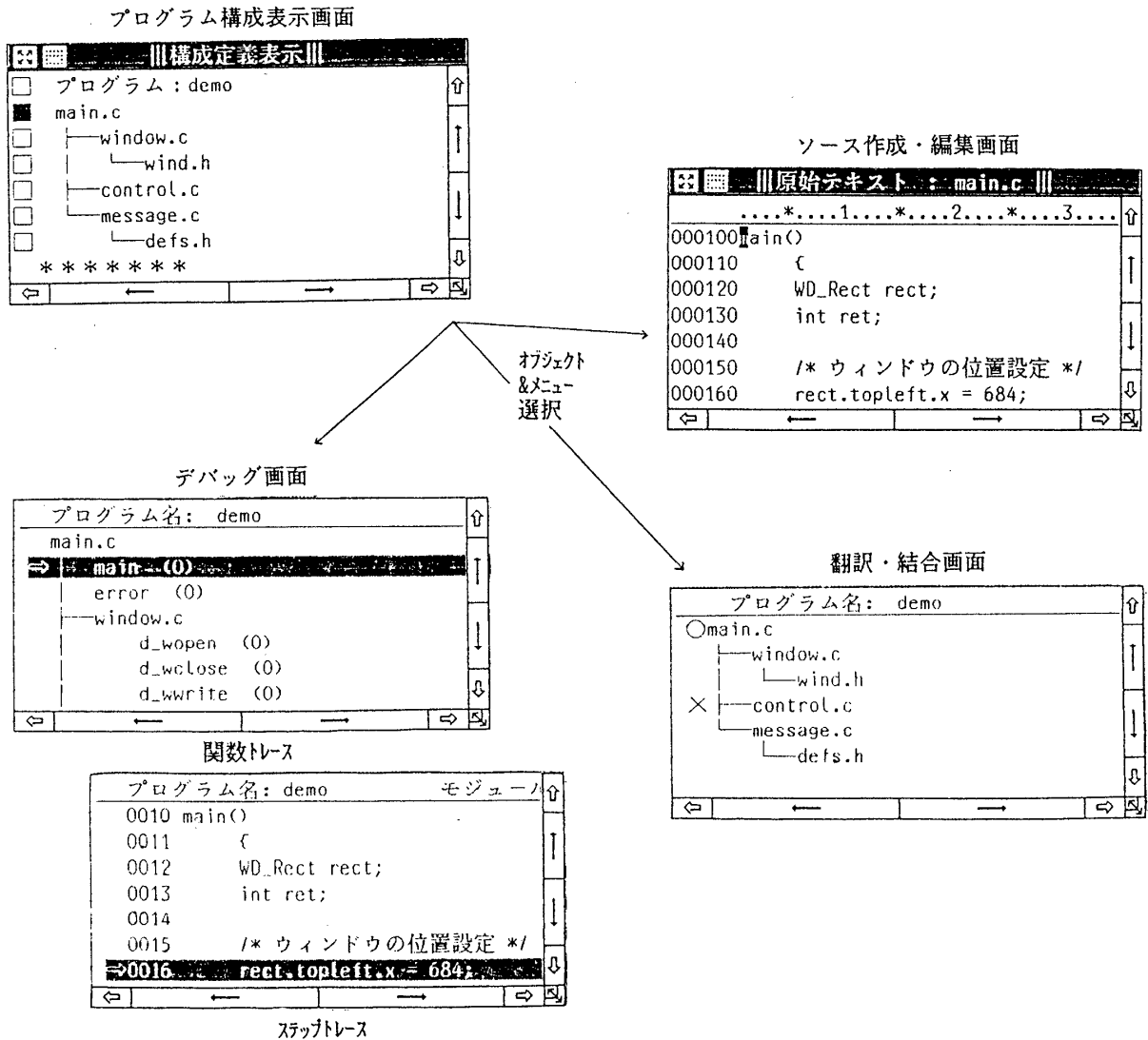


図2 操作画面の概要