

言語適応型プログラミング環境の 設計思想

4B-7

中所 武司

渡辺 坦

林 利弘

(株)日立製作所 システム開発研究所

(同 大みか工場)

1. はじめに

ソフトウェアの生産性と信頼性向上のために、1970年代には、構造化プログラミングをはじめとする新しいプログラミング方法論が提案され、その支援言語も数多く開発されてきた。しかしながら、ソフトウェア開発費用全体に比してこれらの新しい言語の寄与部分には限度があることから、最近、統合プログラミング環境が重視されてきた。ここでは、その一手法として、特定言語を対象に、構造エディタをフロントエンドプロセサ、プログラム解析系をバックエンドプロセサとする言語適応型プログラミング環境について述べる。

2. 基本方針

従来のプログラミング環境においては、エディター、コンパイラ、デバッガなどの基本ツールの他に、ライブラリ管理やドキュメンテーションのためのツールがあるが、ユーザから見て次のような問題があった。

- (1) エディタやデバッガなど、多くのツールは、プログラミング言語に依存しないようにして汎用性を持たせているため、機能的には低水準である。
 - (2) 各ツールは個別に開発され、相互独立性が強いため、お互いの入出力仕様が多岐にわたるため、機能の重複、実際などの不都合が生じ、組合せ使用が難しい。
- そこで、我々は、この問題を解決のために次の方針を有効であると考へた。

(方針1) 各ツールの言語適応化による高機能化

(方針2) 各ツールの有機的結合による統合化

本研究の対象言語は、段階的詳細化とデータ抽象化を支援する構造化プログラミング言語 SPL^{*}であるが、その基本ツールとして、

- (1) 構造エディタ PARSE^{**3)}
- (2) 構造化プログラミング言語コンパイラ⁴⁾
- (3) 構造テスト支援ツール SCORE^{***5)}

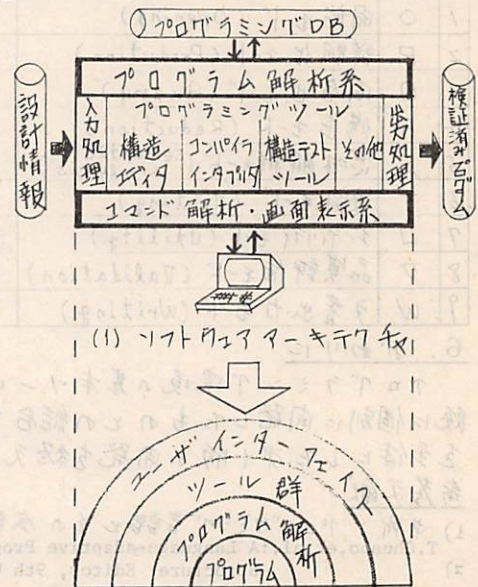
を根拠に方針1に沿って開発して来た。

3. 統合化の条件

方針2の有機的結合は次の4条件により達成する。

- (1) プログラミング方法論の共有

プログラミングの主要な作業である設計、作成、検証の間に一貫性を持たせるために、各部分の支援ツールに同一の方法論を反映させる。



(2) 段階的データ抽象構造

図1. 言語適応型プログラミング環境の設計構成

* SPL : Software Production Language

** PARSE : Production And Reduction Screen Editor

*** SCORE : Source-level Coverage Rate Evaluator

(2) プログラム解析技法の共有

将既言語向ツールはプログラム解析処理を伴うが、ツール間で共通な部分も多い。例えば、構文解析はコンパイラの他にも、構造エディタやリスト網を率計利用コード埋め込みツールで必要である。そこで、構文解析やフロー解析などの処理を共有化することにより、各ツールの高機能化が可能になるほか、対象とする言語の仕様がツール間で異なることも防止できる。

(3) データの共有

プログラムや各ツールの入出力データなどをデータベース化し、それらの間の関係を保持することにより、ツール間インターフェイスの不一致を防止できるほか、プログラム変更に伴う再コンパイルや再リストの自動化、あるいは関連データの統合などの機能を付与できる。

(4) ユーザインターフェイスの共有

ツール毎にフランド言語や端末操作法が異なっているのはシステムとして使いづらい。ユーザインターフェイスの統一化は統合化の必須条件である。

4. システム構成

以上の分析に基づく言語適応型プログラミング環境のシステム構成を図1に示す。図の(1)はソフトウェア構成で、図(2)の条件に従い、プログラム解析系をバックエンドプロセサとする。一方、フロントエンドプロセサが与えるユーザインターフェイスは、事實上、会話処理の多い構造エディタによって決まる。図の(2)は、ソフトウェア構成を階層的データ抽象構造で表現したものである。

5. 基本機能

言語適応型プログラミング環境の基本機能は、以下の9グループに分けられ、ソフトウェア開発工程に従って概ねこの順序で使用される。

No.	記号	モード名称	機能
1	O	開始モード (Opening)	システム使用時の開始処理
2	P	詳細化モード (Production)	文法誘導方式によるプログラムの段階的詳細化
3	Q	問合せモード (Query)	コードレビュー用プログラム解析情報の表示
4	R	修正モード (Reduction)	プログラムの変更
5	S	意味解析モード (Semantic analysis)	プログラムの意味解析
6	T	検証モード (Testing)	プログラムの検証
7	U	ユーティリティモード (Utility)	既存ツールの利用など
8	V	品質評価モード (Validation)	品質の総合的評価
9	W	文書出力モード (Writing)	各種ドキュメントの出力

6. おわりに

プログラミング環境の基本ツールのうち、コンパイラと構造リストツールは、既に個別に開発したものとの統合を検討している。構造エディタは、P, Rモードを主体とした第1版の開発を終え、現在、拡張版を開発中である。

参考文献

- 1) 中折: プログラミング言語とその会話型多機環境, 情報処理, 24, 6, 715-721 (1983)
T. Chusho, et al.: A Language-adaptive Programming Environment Based on a Program Analyzer and a Structure Editor, 9th World Computer Congress IFIP'83 (to appear).
- 2) 田中他: 言語適応型プログラミング環境における構造エディタ PARSE の機能と実装方式, 本大会, 4B-8 (1983)
- 3) 中折他: 段階的詳細化, データ抽象化を伴う言語 SPL のシミュレーション, 情報処理, 21, 3, 223-229 (1980)
- 4) 田中他: 構造化プログラム用ソフトウェア性評価多機ツールの開発, 情報処理, 25回全国大会, 493-494 (1982)