

5H-5 設計情報リポジトリを用いた 改造支援方法

岡 敦子、山本 修一郎
NTT ソフトウェア研究所

1. はじめに

既存のソフトウェアを改造する場合、各種の設計情報を参照するが、設計情報が十分に整理されていないために効率的に進められていない。本稿ではソフトウェアの設計情報を「設計情報リポジトリ」を用いて管理し、各種の設計情報を有効利用することにより、改造作業を支援する方法を提案する。

2. 改造作業の問題点

実際の改造作業を分析すると、以下の問題があった。[1]

- (1) ドキュメントを調査/検討/修正する作業工数が大きい。
- (2) 改造で発生するバグには影響分析ミスが多い。

以上の問題点の原因は、設計情報とソースコードを対応づけていないために必要な設計情報を効率よく検索できないことにある。

3. 設計情報モデル

2. で述べた問題点を解決するためには、改造作業に必要な設計情報のモデル化を行い、このモデルに基づいて設計情報を有効利用できる支援機能を実現する必要がある。

3.1 設計情報のモデル化

まず、設計情報として、ソフトウェアの各工程ごとの生産物(設計ドキュメント、ソースコード等)と、その構成要素を抽出した。次に、これらの設計情報を約40種の設計情報(表1)と、3分類約80種の設計情報間の関係(表2)としてER図(Entity Relationship Diagram)を用いてモデル化した。

3.2 設計情報モデルの実現法

3.1で明らかにした設計情報と設計情報間の関係を関係データベースを用いて具体化することにより、設計情報リポジトリとしてモデルを実現した。このとき、各設計情報を一意に識別するための識別子を付与することにより、設計情報の一貫性管理の容易化を図った。

ER図と関係テーブルとの関係を図1に示す。

4. 改造支援システム

4.1 システム構成

改造作業支援システムの構成を図2に示す。

4.2 改造支援機能

①設計情報管理機能

設計情報リポジトリへの設計情報の追加、修正、削除処理に対して、設計情報を一元管理する。

②設計情報検索機能

改造に必要な情報を、設計情報間の関係をたどることによって提示する。帳票形式の画面(フォーム)を用いた、わかりやすいユーザインタフェースを提供する。

③チェック機能

ER図を用いて抽出した関係を基に、設計情報の関係の矛盾をチェックする。

④修正箇所提示機能

改造に伴うソースコード上の修正箇所、修正候補箇所を提示する。

5. 具体例

(1) 検索作業の効率化

設計情報間の関係をたどることによって、意味情報、依存箇所などを効率よく検索できる(表3)。この結果、改造時の影響波及範囲の調査作業などを効率化できる。

(2) 修正箇所/修正候補箇所の特定作業の効率化

改造にともなう修正箇所を、過去の改造知識を再利用する次の2種類の方法によって特定することができる。

①改造履歴利用法

改造項目を分析した結果、類似する改造項目が多いことが判明した。この事実に基づき、過去の改造履歴情報を表示することにより、修正の候補となる箇所を抽出する。

②実現関係追跡法

改造項目から修正対象となる機能を推測する。その機能仕様から実現関係をたどることによって、修正箇所をトップダウンに抽出する(図3)。

6. おわりに

本研究では、改造作業に必要な設計情報のモデル化と設計情報リポジトリとしての実現、さらに設計情報モデルに基づいた改造支援システムの試作を行った。その結果、改造作業のボトルネックであった修正箇所の検索作業を効率化できる見通しを得た。

今後は、各種設計支援ツールとの統合化を行い、設計情報モデルの妥当性と、設計情報の利用方法を検討する。その結果に基づいて、本改造支援システムを利用した場合の自動化率を評価する。

[謝辞]

日頃ご指導頂いているソフトウェア基礎技術研究部磯田部長、国立主幹研究員に感謝いたします。

[参考文献]

[1]岡、斎、磯田「設計情報ネットワークを用いた改造支援方法」情処第39回全国大会(1989)

表1 抽出した設計情報

| 分類 | 設計情報種別 | 設計情報の具体例 |
|-------|--------|--------------------------------------|
| 工程生産物 | 基本設計 | データフロー図 機能仕様 |
| | 詳細設計 | モジュール構成図 データ構造図 メッセージ テスト項目 |
| | 製造 | プログラム |
| | デバッグ | 改造要求票 |
| 共通 | ファイル | ドキュメントファイル、プログラムファイル |

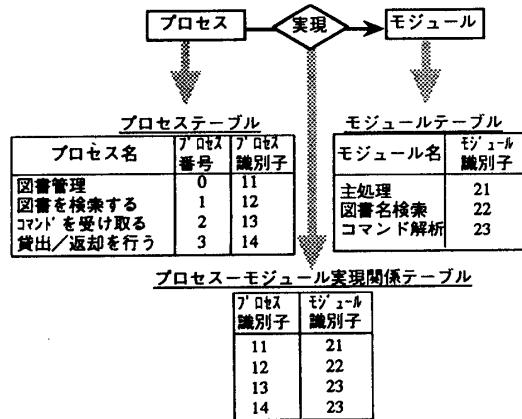


図1 ER図と関係テーブルの対応

表2 抽出した設計情報間の関係

| 関係の分類 | 設計情報間の関係の具体例 |
|-------|---|
| 1:1 | モジュール構成図 ↔ ロードモジュール プロセス ↔ プロセス仕様 データフロー図 ↔ プログラム |
| 1:n | モジュール構成図 ↔ モジュール データフロー図 ↔ データストア データ構造図 ↔ ファイル |
| n:m | パラメータ ↔ 関数 改善項目 ↔ 関数 |

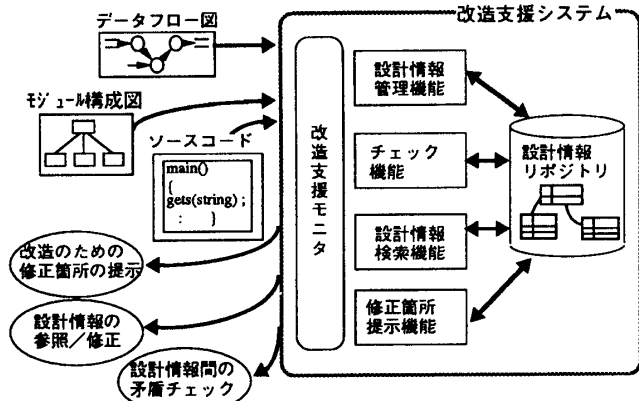
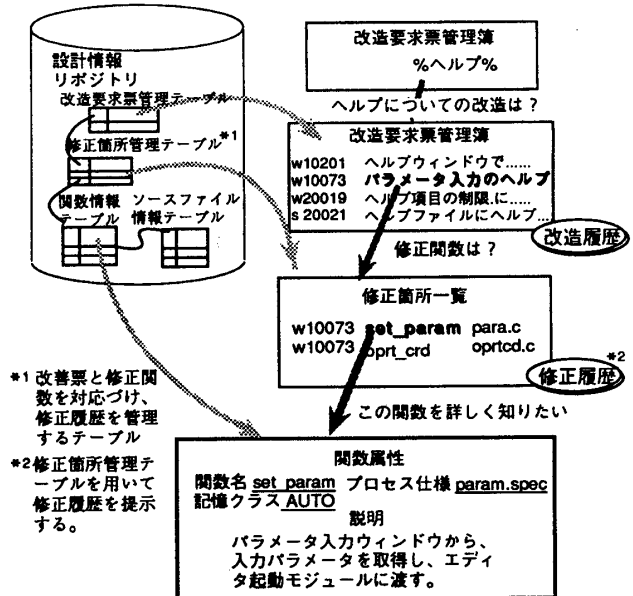


図2 改造支援システム構成

表3 改造時の検索作業の中で自動化した作業

| 検索情報種別 | 自動化した検索作業 |
|--------|--|
| 意味情報 | 関数一覧表から指定した関数の概要を調べる モジュール一覧表から指定したモジュールの概要を調べる プロセス一覧表から指定したプロセスの概要を調べる |
| 定義位置 | 関数が所属するファイル調べる プロセスが所属するデータフロー図を調べる モジュールの所属するモジュール構成図を調べる |
| 依存箇所 | 関数が実現する機能を抽出する 関数が実現する改善項目を抽出する |
| 参照箇所 | 参照し合う改善項目を改善票一覧から抽出する 関数をコールするコール元関数を抽出する 関数がコールするコール先関数を検索する |



【仮定】新たな改善：ヘルプ項目の表示域を拡張するについての修正箇所を改造履歴利用法を用いて検索する

図3 改造履歴利用法に従った修正箇所の検索