

ハイパテキストを用いた設計プロセス支援ツールの試作

田村 直樹、 中島 毅、 藤岡 卓、 上原 憲二、 高野 彰
三菱電機(株) 情報電子研究所

我々は、ソフトウェアの設計作業の過程(設計プロセス)を記述するモデルとして、PPKモデルを提案している。PPKモデルは、設計プロセスを問題解決作業の連鎖として捉える。我々はこのモデルを基に、設計プロセスを記録し、様々な視点で記録を提示することにより設計作業の進行を支援する、設計プロセス支援ツールを試作した。PPKモデルの性質から、ツールの実現にはハイパテキストを用いた。本ツールの試使用の結果、設計プロセスで生じる問題を明確にしながら作業を進めることができた。また記録された設計プロセスの分析から、設計プロセス支援ツールの機能として、設計プロセスの記録の整理や構造の把握のための機能が必要なことが明らかになった。

A Prototyping of Hypertext Tool Which Supports Design Processes

Naoki TAMURA, Tsuyoshi NAKAJIMA, Taku FUJIOKA,
Kenji UEHARA, Akira TAKANO

Information Systems and Electronics Development Laboratory
Mitsubishi Electric Corporation
5-1-1 Ofuna, Kamakura, Kanagawa, 247 Japan

We have proposed a process model called Problem-Product-Knowledge model (PPK-model) which treats a design process as a chain of problem solving activities. Based on this model, we developed a design process supporting tool which records design processes and provides various viewpoints to the record. We implemented this tool on a hypertext framework. The tool allows designers to grasp their design problems easily. In this paper, we describe an overview of this tool. Through analyzing two recorded processes, we found that it is necessary to provide facilities which allow designers to reorganize and browse their design processes.

1. はじめに

近年、設計作業の過程（設計プロセス）をプログラムとして記述することによって、新しい設計支援の方向を見つけだそうとするプロセスプログラミングの議論が盛んである [1] [2]。こうした中で、設計作業の支援に有効なプロセス記述を得るためには、まず実際の設計プロセスの記録から始める必要があるという認識が広まりつつある。

我々は、実際の設計プロセスを記録する手法として、PPK法を提案してきた [3] [4]。このPPK法では小さな問題を解決する作業の連鎖として設計プロセスを捉える。これらの問題解決作業の各々は、PPKモデルと呼ぶプロセスの記述モデルに基づいて記録される。

我々はさらに、このPPKモデルに基づいてソフトウェアの設計作業を支援する方法を検討している。設計プロセスで生じる問題とその問題を解いた結果である生産物をPPKモデルに基づいて記録し、整理することで、作業の流れや未解決の問題を明確にして設計を進めることができるというのが、基本的な考えである。我々はこの考えに基づいて、設計プロセス支援ツールをパーソナルコンピュータ上のハイパテキストを用いて試作した。また、本ツールを簡単な設計の問題で試使用し、設計プロセス支援ツールとして必要な機能を検討した。

本稿では、試作したツールの概要と、試使用を通して検討した設計プロセス支援のための機能について報告する。以下、2章ではPPKモデルに基づく設計プロセスの支援の方法を検討する。3章で試作したツールの概要を述べる。4章でこのツールの試使用の結果を示し、5章で設計プロセス支援ツールに必要な機能の検討を示す。

2. 設計プロセス支援の方法

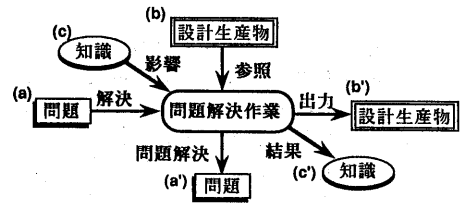
2.1 PPK法

我々は、ソフトウェア設計プロセスの記録・分析の手法としてPPK法を提案している [4]。このPPK法の基本となるプロセスの記述モデルがPPKモデル（Problem-Product-Knowledge model）である [3]。このモデルでは、設計作業で扱う情報を問題、設計生産物、知識の3つに分類し、これらの情報は問題を解決する作業によって関係付けられるとする。PPKモデルは設計プロセスを記述し、その意味を説明する良い枠組を与える（図1）。

PPKモデルでは、ひとつの設計プロセスを小さ

な問題解決作業の連鎖により形成されるネットワークとして表現する。図2にPPKモデルを用いて記述した実際の設計プロセスの一部を示す。

PPKモデルでは、設計プロセスを問題解決過程と捉えている。すなわち、設計者は、新しい問題を次々と発見し、その問題を解決するという作業を繰り返しながら、設計作業を進めていく。従って、PPK法では、まず解決すべき問題を書き下し、この問題を解決するという作業を繰り返しながら設計プロセスを記録する。このため結果としての記録は、



入力

- (a) **問題** 解決すべき問題や動機
- (b) **設計生産物** 以前の設計の出力で、この作業で参照が起きたもの
- (c) **知識** この作業に影響を与えた頭の中の考え
例：設計手法や以前の設計経験など

出力

- (a) **問題** 作業中に生じた問題点や次の作業への動機
- (b) **設計生産物** 作業で生成された生産物
- (c) **知識** 作業で得た生産物にならない考え
例：システムのイメージや達成感など

図1 PPKモデルの概念図

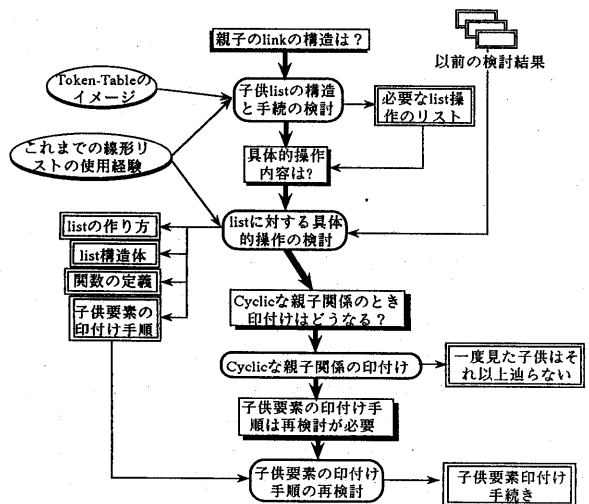


図2 PPKモデルに基づく設計プロセス記述例（線形リスト操作関数群の作成）

設計プロセスの軌跡を表わす。図2中の太い矢印は、この問題解決過程を表わす。

2.2 設計プロセス支援に対する基本的アイデア

このPPK法は、設計作業の支援という立場で捉えた場合、次の特徴を持つ。

①問題を明確にできる

設計プロセスの記録ではまず問題を書き下すため、問題が漠然としている場合もその内容を明らかにできる。また設計作業の途中で思い付いた問題も記録するため、後で未解決の問題リストとして利用して、設計の抜けを減少することができる。

②作業内容を把握できる

生産物や知識を問題解決作業と関係付けて記録するため、作業の流れや未解決の問題を捉えやすい。

我々は、こうした特徴を持つPPK法に基づいて設計プロセスを支援する方法を検討している。基本的なアイデアは、設計プロセスの記録を支援しながら、その記録を積極的に利用する手段を提供することにより、設計作業を効率良く進めることができるようにすることである。ここでは、こうした支援を設計プロセス支援と呼ぶ。設計プロセス支援の具体的な内容は次の3点である。

①記録の支援

設計プロセスで現われる設計情報をPPK法によって記録することを支援する。これにより、問題、設計生産物、知識の各要素の記録と合わせて、それら相互の関係付けを記録することを容易にする。

②モニタリングの支援

記録された設計プロセスを、設計者に提示する。要素間関係付けを辿って関係する情報を参照することで、作業の流れや未解決の問題を把握しやすくする。この機能により、今後の設計作業の進め方(戦略)の検討が容易になる。また、残された問題の数と難易度を認識することで、設計の進行状況の把握にも役立つ。

③再利用の支援

記録された過去の設計プロセスや他人の設計プロセスを参照する機能を提供すれば、作業の進め方についてのノウハウを再利用できる。効果的な設計の作業手順に従うことで、設計の途中で発生する試行錯誤を削減する。

2.3 ハイパertextの利用

近年新しいデータベースとして注目されているハ

イパertext(Hypertext) [5, 6, 7]は、情報を小さな素片に分割し、この素片を相互に関係付けて記録するものである。この情報の素片はノードと呼ばれ、基本的な情報単位として扱われる。またノード相互の関係付けはリンクと呼ばれる。

計算機による設計プロセス支援ツールには、MCCで開発された議論支援ツールgIBISがある[8]。gIBISの目標は、議論進行の枠組をツールが提供することにより理路整然とした議論を実現することにある。議論でなされる発言には、他の発言への参照、同意、反論などの相互関係が多く含まれる。発言の増加に伴ってこの関係付けは増加するため、議論の流れが捉えにくくなる。このためgIBISでは議論支援の環境としてハイパertextを利用している。

PPKモデルでは、設計プロセスを部分問題を解決していく過程と捉え、小さな問題解決作業の連鎖として扱う。そしてこの問題解決作業はそれぞれ問題・生産物・知識によって相互に関係付けられている。設計の進行に伴って、問題・生産物・知識の記録は増加し、また各々の関係付けも増加する。従って、こうした複雑な情報を記録し、また容易に参照するために、ハイパertextの利用が適している。

3. 設計プロセス支援ツールの概要

この設計プロセス支援ツールのプロトタイプをMacintoshのハイパertext HyperCard上に作成した。設計プロセス支援のうち、ここでは設計プロセスの記録と、記録の参照に機能を絞って実現している。

3.1 プロトタイプニングの目的

このプロトタイプニングの目的は、次の2点の検討にある。

①PPKモデルの記述力の検証

PPKモデルに基づいた設計プロセス支援ツールをソフトウェアの設計で利用することで、PPKモデルの持つ設計プロセスの記述力を検証する。

②設計プロセス支援ツールの機能の検討

設計プロセス支援ツールの機能を、実際のソフトウェア設計作業での利用を通して検討する。特に設計プロセス支援ツールとして、ハイパertextの機能に加えて何が必要かを検討する。

このうち①は[4]で示した。本稿では②について示す。

3.2 ノードの型とリンクの型

PPKモデルをハイパテキストへ実装するにあたり、まずノードの型として問題ノード、生産物ノード、知識ノードを用意した。図1にある問題解決作業には具体的な記述内容がないため、問題解決作業の結果である生産物ノードに含めることとした。さらに、問題、知識、生産物の各ノード間のリンクを、図3のように決定した。

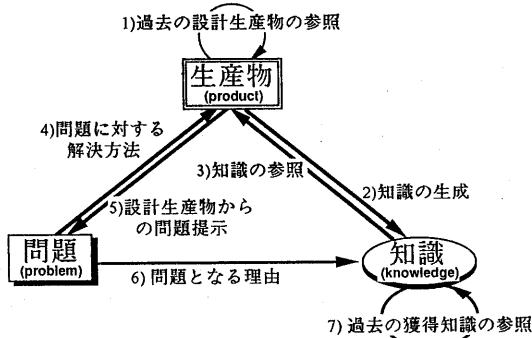


図3 ノード間に用意したリンクの種類

まず、生産物ノードには問題解決作業の結果を設計生産物として記録する。生産物ノードからは、この作業の動機となった問題ノードや、作業を進めるにあたって設計者が参照した過去の生産物ノード、過去に得たノウハウを記した知識ノードが参照できる。特に、動機となった問題ノードの参照(図3 5)は、設計作業終了後に行なわれるレビューを考慮して用意したリンクである。また、ここから新しく発見された問題を問題ノードに記し、この生産物を

得る作業のなかで獲得した知識や設計全体の見通しなどが知識ノードに記録される。

問題ノードには、問題の詳細な記述を残す。この問題は設計者が問題解決作業中に発見するものである。本ツールでは、設計作業後のレビューを考慮し、問題ノードから生産物ノードへの参照リンク(図3 4)を用意した。また、問題ノードに記録したものをなぜ問題として捉えたかの理由付けを、それまでの設計者のノウハウと関係付けて記述する目的で、知識ノードへのリンク(図3 6)も用意した。

知識ノードには、設計の見通し、達成感、あるいは設計時の考察過程で学習した内容など、生産物としても問題としても扱えない内容のメモを残す。知識は、問題解決作業の中で生産物と共に生成される。従って、その内容も問題解決作業に大きく依存する。このため、後で設計プロセスを参照する場合「どういう作業と共にこの知識が得られたか」を把握する必要がある。そこで、知識を生んだ生産物ノードへのリンク(図3 3)を用意した。またこれらの知識は、それ以前に獲得していた知識(ノウハウなど)が基になっている場合が多い。そこで、これに対応する知識ノード間の参照リンク(図3 7)を用意した。

3.3 ユーザインタフェース

図3の各ノード間の関係付けを基に、HyperCard上のユーザインタフェースを決定した。図4に試作ツールの実行画面例を示す。これは、生産物ノードの画面である。

図4の生産物ノードを例に機能を説明する。問題

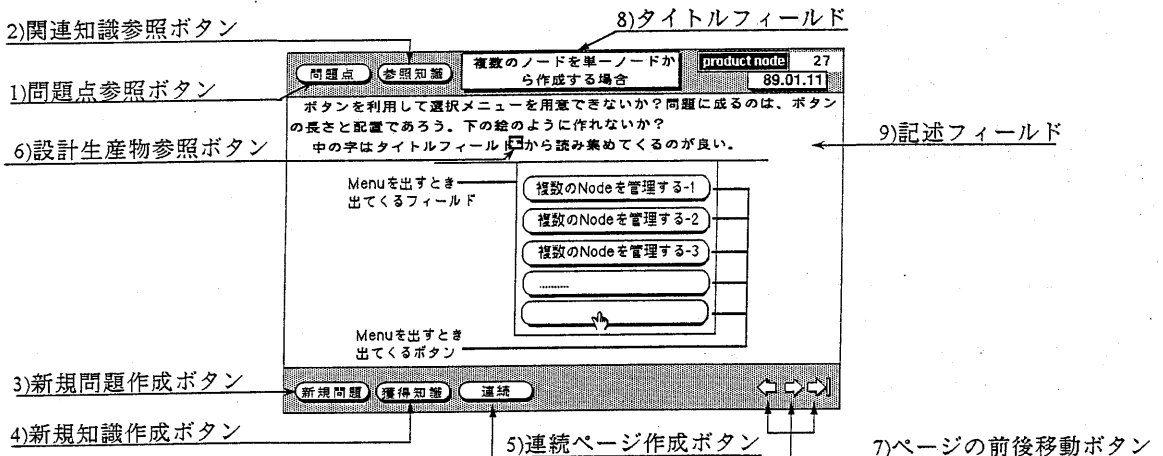


図4 設計プロセス支援ツール実行画面例 (設計生産物ノード)

点参照ボタン(図4 1))をマウスでクリックすると、このノードでの議論の前提となる問題ノードを参照できる。関連知識参照ボタン(図4 2))は、これまでに得た知識を参照するためのボタンである。新規問題ノード作成ボタン(図4 3))及び新規知識作成ボタン(図4 4))によって、問題ノード・知識ノードの新規作成を行なうことができる。また、右下部の矢印(図4 7))もボタンである。それぞれ前のノードへ、次のノードへ、あるいは最後のノードへとジャンプする。この他、文書中に適宜設計生産物参照ボタン(図4 6))を埋め込み、利用することも可能である。連続ページ作成ボタン(図4 5))は、1ノードの記録内容が多く1つの画面に記述が収まらない場合に利用する。タイトルフィールド(図4 8))にはノードのタイトルを記録し、記述フィールド(図4 9))には具体的に設計作業中に考察したことを思い付くままに記録する。

このユーザインタフェースと図1のPPKモデルの対応付けは図5のようになる。

ところで、ひとつの問題解決作業の中で複数の問題点を思い付くことがある。同様にひとつの問題ノードに対し、複数の解決方法を列挙することもある。こうした設計プロセスの分岐に対しては、メニューを用いた分岐状況の提示、選択を行なう機能を用意している。

4. 試使用と評価

この設計プロセス支援ツールを用いて2つの設計作業を行なった。ここでは、この試使用を通して得た本ツールの利点と今後の課題について述べる。

4.1 試作ツールの試使用

以下に、2つの設計の問題と設計者の経験について簡単に示す。

問題1「本ツールの試作」

開発環境：HyperCard環境

規模：HyperTalkのソースコード約400行

設計者：HyperCardでのプログラム経験なし

問題2「与えられた語の依存関係の解析」

開発環境：設計はHyperCard上、コーディング、デバッグはEWSのUNIX環境

規模：C言語のソースコード約700行

設計者：プログラム歴4年のCプログラマ

この設計作業の結果得た記録から、表1に示す数のノードと表2に示す数のリンクを得た。まず表1より知識ノードの利用頻度が低いことがわかる。本来、知識ノードに記録されるべき作業の達成感や実現イメージなどの記述は、生産物ノードや問題ノードに記録されていた。特に問題2では、知識ノードの利用が0であった。また問題1で記録された5個の知識ノードの内容は、設計作業を進める際に参考

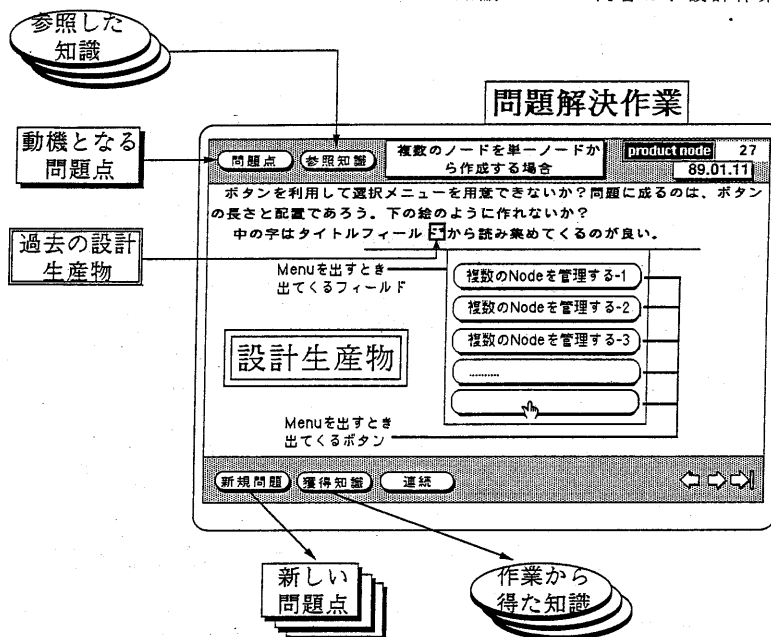


図5 生産物ノードの画面とPPKモデルの対応

とした本の内容を整理したものであった。問題2で知識ノードが利用されなかったのは、設計作業を進める上で新たに知識の整理を行なう必要がなかったからである。

また表2では、問題→生産物リンクの数が多いが、これは1つの問題に対して複数の解を記述したためである。なお、問題1に対する記録は試作を進めながらのものであり、リンク機能が準備されていない時点での記録も含むため、表2にはこの記録は記していない。

表1 ノード種別ごとのノード数

ノード種別	問題1	問題2
問題ノード	30	34
生産物ノード	52	84
知識ノード	5	0
合計	87	118

表2 問題2におけるリンク種別ごとのリンク数

リンク種別	リンク数
生産物→問題リンク	35
問題→生産物リンク	79
過去の設計生産物参照リンク	21

4.2 試使用の感想

本ツールの試使用を通じて、次の2点が利点としてあがった。

- (1) 問題を提示し、それを解決するという問題解決作業の枠組がツールによって提供されており、ひとつの問題を掘り下げて検討する作業を効率良く行なえた。特に問題を書き下すことにより、検討すべき内容を明確にして作業を進めることができ

た。

- (2) 設計作業中の考察を問題・生産物・知識に分離し、小さな断片に分けて記録することに異和感はなかった。本ツールで扱った設計の記録は仕様書と異なり、むしろ設計メモを相互に関連づけて記録する状況に近い。このため、設計作業途中で検討した代替案なども全て記録でき、設計プロセスを詳細に記録できた。

4.3 設計プロセス支援ツールの問題点

試使用の結果、現在のプロトタイプの問題として以下の3点が挙げられた。

- (1) 情報の分割の不十分さ

表1に示したように、知識ノードの利用頻度が低い。知識に相当する記述は、生産物や問題から分離されずに記録されていた。

また、1ノード中に複数の作業の記録が残されていることがあった。図6にこうした作業記録の例とPPKモデルによる設計プロセスの記述との対応を示す。ここでは、ひとつのノードでふたつの作業が行なわれ、さらに次の設計作業に繋がる問題提起とその検討結果が記録されている。

このため、1ノードを情報の単位として設計プロセスを設計者に示しても、本来の設計プロセスの姿が反映されなくなり、プロセスのモニタリングや再利用の支援が困難になる。

- (2) 検討対象の移り変わりに伴う混乱の発生

記録された設計プロセスを整理した結果、問題の

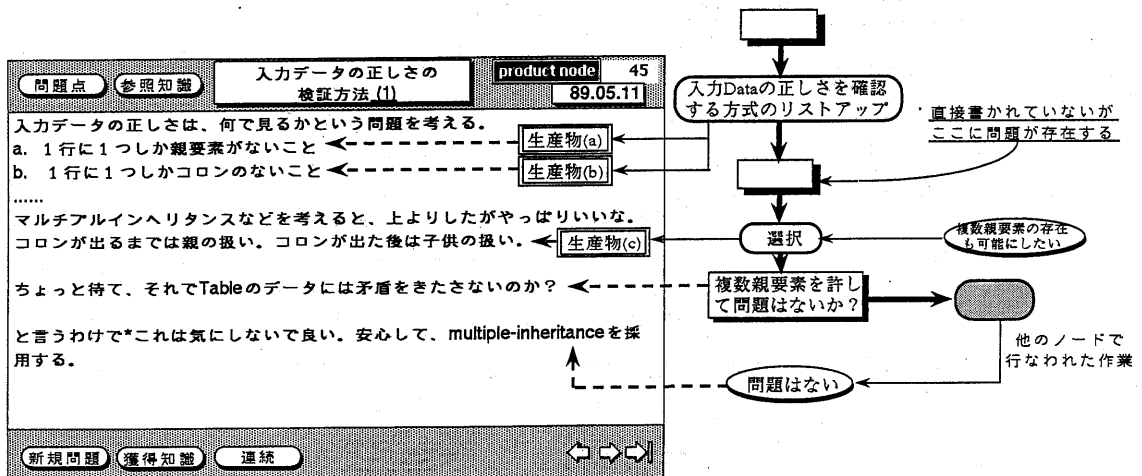


図6 設計の記録とプロセス記述の関係

発生の仕方が設計プロセスの進行に大きな影響を持つことが明らかになった。特に、設計者は新しい問題を思い付くと、それまでの問題を未解決のまま、新しい問題の検討を始めることが多い。問題の思い付きが繰り返されると、設計プロセスは複雑に分岐し、設計者は設計作業の流れを把握できなくなることがある。図7にこうしたプロセスの例を示す。図7では、作業(a)の結果得られた問題(b)を検討する時に、問題(c)を思い付いている。さらに問題の思い付きが重なり、これ以降の設計作業では「親要素の見分け方」と「エラー処理の検討」が交互に行なわれた。設計者はこの一連の作業の最後で、それまでの設計作業に混乱を感じている。

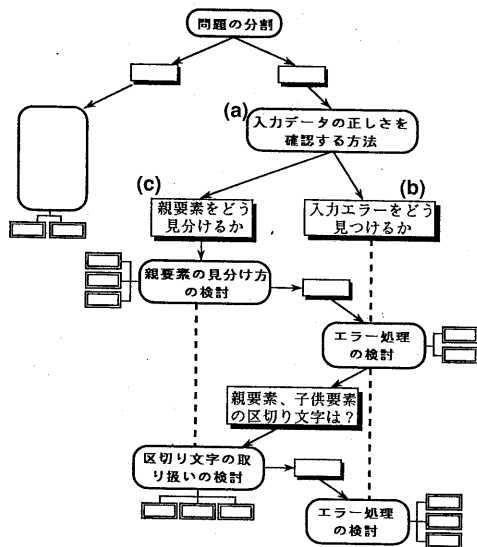


図7 混乱を引き起こす設計プロセス

(3) 記録されない設計プロセスの存在

試使用の結果得た記録では、生産物ノードにソースコードが書かれることも多かった。一方でコーディング・デバッグ中に気が付いた設計の誤りや修正も多かったが、これらは設計プロセスの記録として残されなかった。

5. 設計プロセス支援ツールに必要な機能

4.3で示した問題は設計プロセスをどのように支援していけば良いかを考える上で重要である。本章では、前章の評価結果に基づいて、設計プロセス支援ツールとして必要な機能についての要求をまとめる。

(1) 記録を整理する機能

情報の分割が不十分のまま記録される原因は、設計作業を進めている最中に情報を分割して記録することが難しいことにある。

例えば、1ノードに複数の問題解決作業が記録されることがある。これは、新しい問題が新しいノードに記録されないために起きる。問題を思い付いたときに、あるいは問題をツールに記録しながら解決方法を思い浮かべてしまうため、設計者が新しい問題を明確に認識しないためである。

また、P P Kモデルの知識の記述が生産物や問題から分離されないことも多い。これは、設計者が、進行中の問題解決作業に対する達成感や作業の背景となる知識を、生産物や問題から独立して認識していないためである。設計者の思考プロセスを中断せずに、こうした知識を意識的に分離することは難しい。

こうした問題は、ハイパテキストを利用する際の一般的な問題であり「認識のオーバーヘッド」と呼ばれる[7]。多くのハイパテキストでは、この問題を解決するためにユーザインタフェースの改良を行ない、思い付いたアイデアを即座に記録する方法を検討している。設計プロセス支援ツールでも、思い付いたことを即座に記録するための機能は必要である。

しかし、ユーザインタフェースの改良のみでこの問題に対処することは難しい。これは、問題・生産物・知識に相当する記述が各々比較的小さく、また設計者がこれらの情報を明確に区別しながら設計作業を進めていないためである。従って、設計作業中にはなるべく自由に記録を残し、記録された情報を後で整理し、まとめ直す機能が重要である。一度設計作業での考察をひとつのノードに書き出した後に、問題・生産物・知識を分割することで、設計プロセスを正しく記述することができ、後のプロセスの再利用でも役立つ記録となる。

(2) 設計プロセスのネットワークを把握する機能

設計作業の途中で混乱を生む原因に、本ツールでは設計プロセスの全体構造を把握できず、作業の進行状況を見て今後の作業の戦略を検討できなかったことがある。記録の過程で起きる混乱は、多くのハイパテキストでも問題になっている[7]。この問題を軽減するものとしてブラウザがある。ブラウザは記録の全体像を把握しやすいように、ハイパテキス

トネットワークの構造をユーザに表示するものである。しかし、本ツールではHyperCardの機能の制約からブラウザを用意していない。

設計プロセスでは、ひとつの問題解決作業から複数の問題ノード、生産物ノードを得ることが多い。このため、PPKモデルに基づく設計プロセスの記述はネットワーク構造を形成する。設計の進捗はこのネットワークの成長の程度で把握できると考えている。また、このネットワークにより、現在扱っている問題が設計プロセス全体の中でどういう位置付けにあるかを把握することも可能となる。このため、設計プロセス支援ツールでは、ブラウザを用いた設計プロセスの構造の表示は重要な機能となる。

ブラウザを用いた設計プロセスの提示方法には、生産物とプロセスの2つの視点がある(図8)。プロセスという視点(図8(a))では、問題解決作業の流れを表示する。生産物による視点(図8(b))は、ソフトウェアの構成に基づいた視点であり、生産物の意味的な関係を中心に扱う。設計プロセスの記録をとると、この2つが混在した形で情報が記録される。例えば、問題解決作業を進めていく過程では、現在未解決の問題やそれまでの作業の流れが把握できたほうが、次に何を検討すべきかといった作業戦略を立てやすい。一方、保守やデバッグの過程で設計を変更する場合は、生産物に沿ったネットワークの表示の方が、元の設計と比較した記述を行ないやすい。これら2つの視点は、設計作業を円滑に進める上でともに必要な視点である。このためブラウザの実現では、このような異なる視点から情報を表示する方法、そのために必要な情報を記録する方法も検討しなければならない。

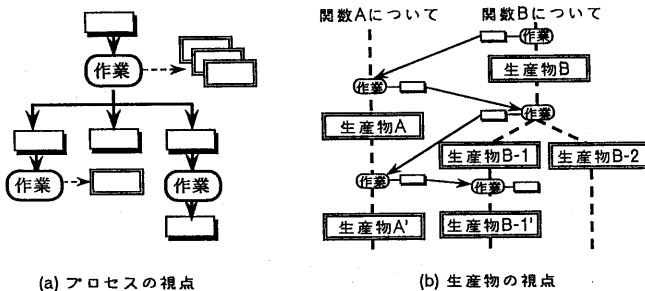


図8 プロセスに対する2つの視点

(3) プログラミング環境としての機能
設計プロセスの再利用を考えた場合、正しいプロ

グラムに至るまでの、デバッグなども含めた全開発プロセスの記録が必要となる。従って設計プロセス支援ツールでは、ソフトウェアの開発環境も取り込むことで、コーディングやデバッグの過程を含めた全開発プロセスを取り扱えるようにするべきである。

6. まとめ

我々の提唱するPPKモデルに基づいた設計プロセス支援ツールをHyperCard上に試作した。本ツールの試使用の結果、設計メモを基に設計プロセスと設計生産物を詳細に記録できた。また、ツールが考察の枠組を与えるため、問題を掘り下げて検討することができた。さらに、設計プロセスの記録を分析した結果、設計プロセス支援ツールには、設計プロセスの記録の整理や構造の把握のための機能が必要なことが明らかになった。今後、これらの機能の実現方法を検討していく予定である。

参考文献

- [1] Osterweil, L., "Software Processes Are Software Too.", Proc. of ICSE 9 (pp.14-16), IEEE(1987).
- [2] Lehman, M. M., "Process Models, Process Programs, Programming Support," Proc. of ICSE 9 (pp.2-13), IEEE(1987).
- [3] 中島 他, "ソフトウェア設計における実プロセスの記述法", ソフトウェアシンポジウム (1989-6).
- [4] 中島 他, "PPK法: ソフトウェア設計プロセスの記録と分析の手法", 情処学会 ソフトウェア工学研究会 67-2 (1989-7).
- [5] Halasz, F.G., "NOTE CARDS: Seven Issues for the Next Generation of Hypermedia Systems," Comm. of ACM vol.31 No.7(1988-7).
- [6] Delisle, N. and Schwartz, M., "Neptune: A Hypertext System for CAD Applications," Proc. ACM SIGMOD '86, ACM(1986-5).
- [7] Conklin, J., "Hypertext: An Introduction and Survey," IEEE Computer, Vol.18, No.9 (pp.17-41), IEEE(1987).
- [8] Conklin, J. and Begeman, M.L., "gIBIS: A Hypertext Tool for Exploratory Policy Discussion," ACM Trans. on Office Information Systems, Vol.6, No.4 (pp.303-331), ACM(1988-10).