ソフトウェア工学 77－4
（1991. 2. 7）

ソフトウェアの仕様化過程における協調作業の分析

海谷治彦　　佐伯元司
東京工業大学

複数の作業者がソフトウェアの仕様を作成する過程の履歴をとり，その分析を行った結果を報告する．複数の作業者によるソフトウェアの作成過程では，他の作業者に影響を及ぼす作業者の言語行為が大きな役割をはたしている．このような考え方に立って，２つのソフトウェアの仕様化の実験を行ない，その仕様化作業の様子をビデオテープに記録した．その記録を基に，複数の作業者による仕様化過程を，発話によってつくり出される話題の状態遷移としてモデル化した．

Analysis of Cooperative Work in Software Development Process

Haruhiko Kaiya　　Motoshi Saeki
Tokyo Institute of Technology

This paper presents a model of cooperative work in software development process. Speech acts of a worker which have an effect to other workers, play important role in cooperative software design process. In order to clarify the model of cooperative work, we have recorded several pre-designed meeting for designing software using videotape, and established analysis method of videotaped records. We construct a model of cooperative software process based topic transition caused by speech acts.

1 はじめに

作成対象となるソフトウェアが複雑になるにつれ，複数の人間が仕様化／設計作業に携わるようになった現状では，協調作業としての人間の活動を明らかにする必要がある．[1]では，従来の協調作業を支援する電子会議室 [3] などがグループ作業の効率を改善することができなかったことを紹介し，その原因は作業者が効果的に振舞えるような支援をしなかったことにあると主張している．

ソフトウェアの仕様化過程における協調作業を効果的に支援するためには，仕様化過程における協調作業のモデル化を行なう必要がある．モデル化のための複数の作業者による仕様化過程の分析はいくつか行なわれてるが [1]，分析をするための基準や手法が確立されていない．

本研究では，複数の作業者からなるグループが仕様化／設計作業を進めていく環境下での人間の作業，特に協調のための作業を明らかにするために，ビデオカメラを用い，作業者グループが議論しながらプロダクトを作成していく活動の記録をとった．それらをもとに分析手法の検討を行ない，協調作業分析のためのモデルを提案する．

2 実験目的と方法

2.1 目的

図 1に研究目的の階層を示す．

第1階層の目的を果たすためには，実際の作業の分析を行なうことが有効だと考えられる．しかし，その分析のための基準が明らかでないため，分析のモデルの構築が必要である．

我々は，話題という行為の列を整理分類することで，分析モデルの構築の足掛かりとする．以下に示す実験の記録を観察することから，この考え方を採用することにした．行為とは仕様化過程において作業者が行なう発話や行動の内で仕様化過程において意味のある行為のことを指すことにする．

本研究の目的は 図 1 における第3，4階層である．

2.2 方法

図 2に実験方法の手順図を示す．［グループ1］にたいし予備実験として数回の仕様化作業を行なってもらい，その実験結果からある程度の［基準］を抽出する．その［基準］を基に［グループ2］に対して同様の実験を行ない，分析の［モデル］を構築する．その［モデル］を使って再び

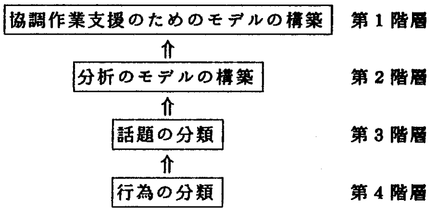

図 1: 研究目的の階層

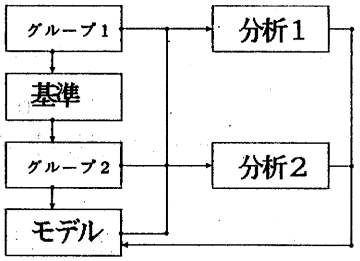

図 2: 実験方法の手順図

［グループ1］,［グループ2］に対する分析を行ない，その結果を［モデル］に対して反映させることにする．実験の過程はビデオ，メモ等を用いて記録した．

それぞれのグループに対して共通の条件は，

- 作業形式 — 会議室を使って討論形式で作業を行なう．メモ，黒板の利用が可能である．
- 作業者 — 本学，情報工学科4年生の学生5人
- 問題の与え方 — ソフトウェアを発注する顧客にあたる作業者を1人参加させている．他の作業者にたいして問題に対する事前の知識は与えない．
- 問題の性質 — 対話型のソフトウェア．既存のソフトウェアが存在し，ワークステーション（WS），パーソナルコンピュータ（PC）上で稼働している．顧客はそれに対しなんらかの不満を抱いている．

2つのグループのメンバーは互いに重複している．そのために，それぞれのグループに対して違った問題を与えている．

3 グループ 1

3.1 実験過程

作業に対する入力:

問題 グラフィックエディタ
　論文などの図を書くツール

制約 5人のプログラマに作業を割り当てるための，作業分割を行なうように指示．

作業過程:

会議は合計3回行ない，それぞれの日時と時間は以下の通りである．

第1回	5月15日	2時間12分
第2回	6月6日	1時間15分
第3回	6月12日	1時間16分

作業後の出力:

8ページ程度の自然言語+図で書かれた仕様書．

モジュール	関連モジュールにやって欲しいことを指示
関連モジュール	要求を受理もしくは拒否
各モジュール	分担境界の決定

モジュール1	利用者界面の決定
⋮	⋮
モジュールn	利用者界面の決定
	利用者界面に当たるモジュールの追加

利用者	機能を提案
設計者	設計した場合の制約を述べる
調整者	妥協の決定を促す.

設計者	利用者界面の選択を提示
調整者	顧客にどちらを選ぶかを質問
顧客	選択について質問
設計者	選択事項の説明
顧客	その利用者界面に対する要求を明確化
顧客	選択

表 1: 行為のパターン

3.2 分析

作業はモジュール分割の段階（第1回）と，モジュール間の調整の段階（第2，3回）に別れた．

実際に分割されたモジュールは，グラフィクエディタにおける，

1. 描画 .... 特定の図形を描く．
2. 編集 .... 図形の編集をする．
3. 文字 .... 文字入力をする．
4. 画面 .... メニューレイアウトや画面表示をする．
5. 入出力 ...... ファイルやプリンタなどとのやりとりの管理をする．

のようなモジュールであり，作業者1人に対して1つのモジュールが割り当てられた．

> 作業過程は，話題の列であり，それぞれの話題の決定は作業者の行為の列として捉えることができる．

この仮定を基に，行為のパターンを抽出することで話題を特定する．

表 1に，話題を特定するために行為の列のパターンを抽出した例を示す．これは何らかの提案がされてから結論がでるまでの行為の列を，それぞれの行為に適当な分類をして並べたものである．以上のパターンの中に現れる行為は，

- 対象とするソフトウェアを利用，製作，保守する立場
- 作業を円滑に進めていく立場
- 特定のモジュールを支援する立場

話題変化	説明
A → C	ツールの外見を決めてからその機能に移る
B → A	利用者のやりたい作業に合わせて外見の議論に移る
C → B	機能を決めてから利用方法に移る
C⋯C → D	機能に共通な内部構造について取り決める

表 2: 話題の変遷パターン

の様な性質に分類できる．

次に行為の列として捉えた話題の決定事項の種類を A.〜D.のように特定した．

A. 利用者界面について

B. 利用者の作業について

C. ツールの機能について

D. ツールの内部構造について

決定事項と対応付けた話題の列を仕様化過程として我々はとらえる．仕様化過程における話題の変化のパターンを表 2に示す．これは上記の A.〜D.のような種類に分類された話題の変化のパターンを示し，それに対しての説明を加えている表である．

4 グループ2

［グループ1］の分析の結果をふまえ，事前に作業者に対して役割を与えることによって実験を行なった．

具体的には，作業者の行為が対象とするソフトウェアに対する利用者や製作者の立場や，作業を速やかに進めようとする立場に立ったものが作業過程に影響を与えていたので，明示的に顧客，利用者，製作者，調整者，書記という役割を作業者に割り当てた．

話題は作ろうとするツールのイメージを明らかにするものと，要求を明らかにするものと，実際のプログラムに関するものに分類できたので，それらをプロダクトとして明示的に要求した．

4.1 実験過程

作業に対する入力:

問題: ハイパーテキスト型のツール

ソフトウェアを仕様化する人間の思考過程の履歴をとるためのツール [7]

制約: 以下に示すような役割を与える．

顧客 (Cus) .. 要求を提案する立場．
利用者 (Use) ソフトウェアを利用する立場．
製作者 (Des) 技術的な助言をする立場．
書記 (Sec) .. 記録をとり，仕様書を書く立場．
調整者 (Coo) 作業進行を調整する立場．

プロダクトとして,

1. 基本調査書
このソフトウェアがどんなものかを明確にする.
2. 要求仕様書
ソフトウェアが何をすれば良いのかを明らかにする.
3. 設計仕様書
ソフトウェアをどのように実現すれば良いのかを明らかにする.

の作成を文書で指示.

作業過程:

会議は合計4回行ない, それぞれの日時と時間は以下の通りである.

第1回	7月 10日	2時間 45分	基本調査
第2回	7月 11日	2時間 27分	要求仕様化
第3回	7月 17日	2時間 54分	設計仕様化
第4回	7月 18日	3時間 30分	設計仕様化

作業後の出力:

1. 要求調査書: 自然言語+図 2ページ
2. 要求仕様書: 自然言語+図 2ページ
3. 設計仕様書: 自然言語+図 12ページ

4.2 分析

[グループ1] と同様に話題の遷移と行為の遷移を整理してみる.

行為の列を 表 3 に示し, それぞれの列を話題に対応させる.

話題は, 表 3 より,

Cus:要求 ⇒ Sec記述

の様に, 顧客の要求を意味する行為で始まり, 書記の記述に対応する行為で完結するものが多いことがわかった.

次に, 話題の種類を [グループ1] 同様に分類してみる.

A. 利用者の作業について.

B. 追加要求のについて.

C. 現ソフトウェアについて.

D. 決定事項について.

話題の変遷パターンとその因果関係を 表 4 に示す.

5 分析のモデル

[グループ1, 2] の分析に用いた話題と行為を使った仕様化過程の形式化は, 以下のような分析モデルとして整理することができる.

このモデルは仕様化過程を2階層の状態遷移系としてとらえる. 第1階層は話題の状態遷移であり, 第2階層は立場に裏付けられた作業者の行為の状態遷移である.

Coo	実行例を要求.
Cus	実行を言う.
Des	実行中の計算機界面を特定.
Cus	実行を言う.
Use	利用者がやるかソフトウェアがやるかを質問.
Sec	記述

Coo	他に要求がないか確認
Cus	要求を出す.
Cus	実行を言う.
Des	ソフトウェアがどのように変化するか質問.
Des	具体的なモジュールに対応付ける.
Sec	記述

Cus	要求を出す.
Des	要求に満たすため事前に満たされなければならない要求を提案

Cus	要求を言う.
Des	技術な観点での問題点を言う.
Des	妥協案を出す.
Sec	記述

表 3: 行為のパターン

話題変化	理由
A→B	シュミレーションによる新しい要求の発生
C→A	現ソフトウェアの機能を確認 現ソフトウェアの欠点の調査
D→A	決定事項の機能を確認 決定事項の欠点の調査
B→D	追加事項と決定事項の無矛盾性の検証

表 4: 話題の変遷パターン

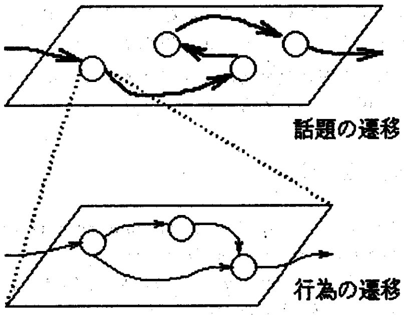

図 3: モデルの概要図

5.1 分析モデルの形式化

複数の作業者によるソフトウェアの仕様化過程は作業の状態が作業者の行為によって推移することにより仕様を作成する過程であり，

仕様化過程 : 行為 × 状態 → 状態

のように表現できる．行為は [4], [5] における言語行為に近い概念である．

ここでの状態とは仕様化過程において言及された話題に対応するプロダクトの集まりを表している．プロダクトとは仕様を構成している概念を表している．

同一のプロダクトに対して言及している行為の状態遷移を第 2 階層の状態遷移とする．

第 1 階層は注目しているプロダクトの遷移に対応している．

このモデルを図示すると 図 3 のようになる．

具体的な第 1 階層の状態に対応する話題は，[グループ 1，2] の実験から，

- 利用者要求の確定
- 利用者界面の確定
- 利用者作業の確定
- ソフトウェア機能の確定
- 決定事項の確認

などがあることがわかった．プロダクトの中には話題となったが最終的な仕様には残らない物も存在する．

これらの話題を決定するための第 2 階層の状態は，

提案 ･･･ 新しいプロダクトに話題が移った状態

議論 ･･･ プロダクトに対しての決定をしている状態

決定 ･･･ プロダクトに対して何らかの決定が終った状態

の 3 つの状態が考えられ，状態の遷移を促す作業者の行為の裏付けとなる立場は，

1. 顧客 .... 最終的にソフトウェアを受けとることを前提とした問題領域に依存する立場．
2. 利用者 ...... ソフトウェアを実際に利用することを前提とした利用領域に依存する立場．
3. 製作者 ...... 実際にプログラムを作ることを前提とした技術領域に依存する立場．
4. 書記 .... 仕様化過程の記録をとり，実際に仕様書を書くことを前提とした領域に依存する立場．
5. 調整者 ...... 作業進行を調整し，速やかなる決定を促すことを前提とした領域に依存する立場．

が考えられる．

5.2 分析モデルを用いた評価

§5.1 のようなモデルを基に，仕様化過程における作業の分析を行なう．分析は作業を記録したビデオテープの記録と作業者のとったメモを見ながら行ない，不明な点があれば作業者に記録を見せ問い合わせる．

実際に観測されるのは行為であることより，

1. プロダクトが提案され，なんらかの結論が得られた行為を調べる．
2. あるプロダクトに言及している行為の列を集める．
3. それぞれの行為について，どの作業者が行なっているのかを調べる．
4. それぞれの作業者が，どんな立場で行為しているかを調べる．

以上の結果より，

- 行為の列のパターンを抽出する．
- 行為の分類をする．
- 行為の列としての話題の特定を行なう．
- 話題の分類をする．

§3.2,4.2 の分析はこのような方法に従って行なった．

6 議論

- 話題の開始と終了について：

  話題の終了については書記が記述するか，もしくは何らかの立場から却下されたことで，捉えることができる．

  話題の開始は注目しているプロダクトを変化させた行為で捉えることができる．

  行為の背景となる立場を§5.1で示したように 5 種類用意したので，その範囲内での未使用の立場を抽出可能である．

- **話題の重複の検査:**

  プロダクトは既出の話題と対応するので，それによって話題の重複を検査できる．既出の話題についての作業者の行為の変遷を調べることで，その話題の重複かどうか確かめることができる．

7 おわりに

仕様化過程における協調作業を明らかにするための，基礎実験を行ない，分析のためのモデルを提案した．提案したモデルは実際の仕様化過程の分析記録から考案したものである．

問題点および今後の研究課題として，以下のような項目が考えられる．

1. 実験方法について

   - 作業者に対して立場（顧客，調整者，製作者など）にあった役割を果たしてもらう．今回の実験では，役割に合わない行為が多く見られた．今後，実験の分析を形式的に進めていくためには，それぞれの役割の指標を与えなければならない．
   - 今回の実験では特定の仕様化技法，仕様記述言語を用いないで実験を行なった．今後，構造化分析法，ジャクソン開発法といった仕様化/設計の方法論に従って協調作業を進める場合についても，同様な実験，分析を行い，本研究の手法を評価する．

2. 対象とするソフトウェアについて

   - 今回の実験では，event driven型のソフトウェアに対して実験を行なった．今後，標準問題 [2] などを取り上げる．

3. 分析方法について

   今回の実験では，分析はビデオカメラで記録した会議の様子を，分析者が見ることで行なった．実際の分析には大変な時間が掛かり，その分析基準は恣意的である面が多いので，適切な分析基準と体系的な分析方法を提案する必要がある．

   例えば，今回の実験結果を基に協調作業のプリミティブを規定することで，それ以外の行為を許さない方法で仕様化作業を行なってもらうなどの方法が考えられる．

4. 個人の仕様化作業との比較

   仕様化過程において個人で行う方が良い作業とグループで行った方が良い作業がある．

   例えば，ソフトウェアに対する一貫したイメージを保つのは個人での作業の方が有利であるが，多面的な視野から仕様を評価検討するには，グループ作業の方が有利である．

   [7], [6] では個人の仕様化過程についての分析を行っており，これらの研究成果との比較検討をする必要がある．

5. 協調作業支援ツールについて

   複数人で仕様化作業を行なう際の協調作業を支援するツール，例えばどの項目について同意が得られたか，どの項目が未解決となっているか等を整理，管理し，提示することによって，仕様作成をガイドするツールをワークステーション上に実現する．

   また，協調作業の利点を利用できるような，個人作業の支援ツールも考えられる．

参考文献

[1] Bill Curtis. Implication from empirical studies of the software design process. In *Proceedings of an International Conferrence organized by the IPSJ to communicate the 30th Anniversary*, 1990.

[2] R. A. Kemmerer, S. White, A. Mili, and N. Davis. Problem set for fourth international workshop on software specification and design. In *4th International Workshop on Software Specification and Design*, pages 9–10, Apr 1987.

[3] Mark Stefik, Gregg Foster, Daiel G. Bobraw, Kenneth kahn, Stan Lanning, and Lucy Suchman. BEYOND THE CHALKBOAD: computer supported for collaboration and problem solving in meetings. *Communications of the ACM*, 30(1):32–47, Jan 1987.

[4] Terry Winograd. Where the action is. *BYTE*, 13(13), Dec 1988.

[5] Terry Winograd and Fernand Flores. *Understanding Computers and Cognition*. Ablex Publishing Corporation, Norwood, N.J., 1986.

[6] 池，海谷，佐伯，本間．ソフトウェア仕様記述過程分析のための基礎実験．，情報処理学会ソフトウェア工学研究会，68–5ページ，9　1989.

[7] 渡辺智弘，佐伯元司．思考過程モデルに基づいたツールを用いた仕様化作業の分析．，電子情報通信学会全国大会，6:43ページ，Oct　1990.