

ソフトウェア・プロセスによる
システム設計・教育・訓練の試行および評価

望月純夫 山内 顯 片山卓也
三菱スペース・ソフトウェア(株) 東京工業大学

あらまし 近年、情報処理産業の発展に伴い若手ソフトウェア技術者の教育が重要な課題となっている。我々は、実際のプロジェクトを対象に、その基本設計段階におけるソフトウェア・プロセスの分析と評価を行い、それに基づいてSE (System Engineer) 教育用ツールの開発を行ってきた。その結果、最終的にかなり現実に即したソフトウェア・プロセスが得られ、それを用いたSE教育用ツールの作成と試験的なSE教育の施行を試みた。本稿では、これまで我々が行ってきた検討の経緯と成果をまとめると共に、今後の見通しと課題について述べる。

THE TRIAL AND EVALUATION FOR THE EDUCATION AND TRAINING IN SYSTEM DESIGN
USING SOFTWARE PROCESS

Sumio MOCHIZUKI Akira YAMAUCHI
Mitsubishi Space Software
MSS Nagasima Bldg. 524, Kamimachiya,
Kamakura-shi 247, Japan

Takuya KATAYAMA
Tokyo Institute of Technology
1 Ookayama, Meguroku, Tokyo 152, Japan.

Abstract The training of inexperienced software engineers has recently become an important theme, along with the growth of information processing industries. We have analyzed and evaluated software processes of the actual projects at their system design phase. And we have accordingly developed a experimental instruction tool to educate system engineers. As a result, we have finally obtained the fairly realistic software process concept; and in which we have tried to produce the effective instruction tool to conduct an experimental education of the inexperienced system engineers. This paper describes our study so far and summarizes the obtained results, and it also depicts both the future prospect and the present problems.

1. はじめに

近年の情報処理産業の成長は目ざましいものがあるが、市場の拡大に比べて技術者数の伸びが追いつかず慢性的な技術者不足の傾向は今後益々顕著になるものと予測される。

一般に、若手技術者にソフトウェア設計技術を習得させる手段としてOJT (On the Job Training) が最も効果的だと考えられている。しかし、OJTには多くの時間と、教育される技術者数に見合った適当な熟練技術者が必要であり、現状においてはそのような恵まれた環境を維持することは非常に困難であると言わざるを得ない。

そのような状況に対応して、我々はシステム・エンジニア (以下SEと略す。) を教育するOJTに代わる手段として、実務を対象としたソフトウェア・プロセスの分析と、それに基づくSE教育用ツールの開発を試みてきた。[3~8] 本稿では、これまでに我々がやってきた検討の経緯と成果をまとめると共に、ソフトウェア・プロセスの教育への応用という立場から、今後の課題と見通しについて述べる。

2. ソフトウェア・プロセスの分析

2. 1 経緯

ソフトウェア・プロセスは、設計現場における技術者の動き、情報の流れを観察、分析し、それをできる限り忠実に記述することにより、設計過程を究明しようとするソフトウェア工学の取り組みから生まれたものである。[1]

設計過程が明かになれば、これを応用することによって、

- ・ソフトウェア設計作業の標準化
- ・ソフトウェア設計作業の可視化及び管理の容易化
- ・ソフトウェア・プロセス再利用による生産性向上

等の効果が期待できる。

これ等の項目は、何れも電算機システム開発部門、応用ソフトウェア開発部門にとっては、極めて重要なものであるが、我々の最終的な目的は、このソフトウェア・プロセスの分析を通じて、標準的な (理想的な) ソフトウェア・プロ

セスの存在を探り、最終的には、若年層の育成・教育ならびにシステム設計支援のための教育ツールを確立することにある。

本研究において我々は、以下に示す様な順序で検討を進めることとした。

(1) 過去に何度も設計経験があり、その結果設計手順も十分にリファインされているシステムを選び、それらのシステム開発に参加した熟練SEの設計プロセスを記述する。

(2) (1)で得られたソフトウェア・プロセスを別の技術者チームが使用し、別の設計課題を解くことにより、このソフトウェア・プロセスの評価と改善を行う。

(3) 改善されたプロセスをもとにSE教育用ツールを作成し、設計教育の試行を行う。

2. 2 ソフトウェア・プロセスの記述方法

本研究では、ソフトウェア・プロセスの記述方法として、プロセス・モデルH F S P (Hierarchical Functional Software Process)を用いることとした。[2]

以下に我々がプロセス記述を行うにあたって基準としたH F S Pの基本的な考え方を示す。

(1) ソフトウェア・プロセスをその入出力関係から捉え、入力から出力へ変換する小さなプロセスの集合として設計手順を表現する。各設計プロセスだけでなく、その入力オブジェクト、出力オブジェクトも明確に定義する。

(2) プロセスの機能が複雑である場合は、そのプロセスを更に細かいサブ・プロセスに分解する。

(3) この作業を各々のプロセスが十分単純化されるまで続ける。

(4) 各オブジェクトがプロセスの途中で消滅したり、急に現れたりしないように注意し、オブジェクトの一貫性を保つようにする。

(5) 各々のオブジェクト間の関係 (どのオブジ

エクトを資源としてどのオブジェクトがつくられるか)を明かにする。

H F S Pでは、プロセス間で受け渡されるオブジェクトが重要な役割を担うが、実際のソフトウェア設計において、設計途中に現れるオブジェクト類の数は非常に多く、総てを取り上げることが困難である。そこで我々は、次のような基準を設定しオブジェクトを記述した。

- ・設計の成果として最終的にドキュメント(設計仕様書等)として残るものを正式オブジェクト(または、単にオブジェクト)とする。
- ・オブジェクトとオブジェクトの間で作成される(作業の補助となるような)オブジェクトを、中間オブジェクトとする。
- ・繰り返し作業中に何度も作り直されるようなオブジェクトも中間オブジェクトとする。

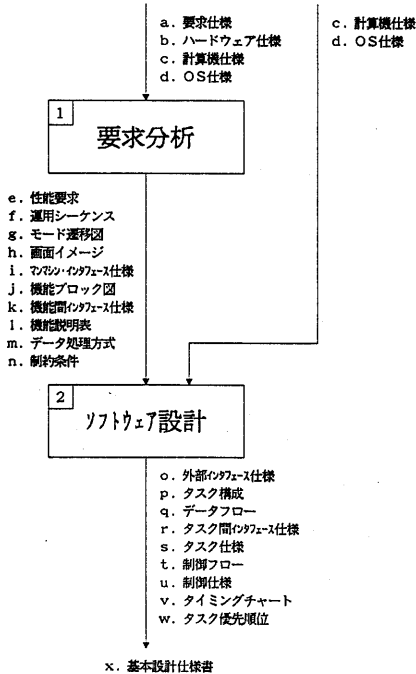


図1 実時間処理システム基本設計の第1階層ソフトウェア・プロセス

2. 3 ソフトウェア・プロセスの分析結果

我々はこれまでに人工衛星のチェックアウトシステムの設計を対象として検討を行ってきた。これらの検討結果については既に報告しているので([3]、[4])、ここでは現在最終的に得られているソフトウェア・プロセスを示しその分析結果について述べる。

このソフトウェア・プロセスは、次のような3層の階層構造となっている。(図1、図2、図3)

(1) 第1階層のソフトウェア・プロセス(図1)

基本設計段階における設計内容は、a. ユーザ要求分析と b. ソフトウェア設計の2つのフェーズに分けて考えることが出来る。

ユーザ要求分析段階では、ユーザのシステムに対する要求機能、性能に対する事項を十分に把握し、ソフトウェア設計段階では、それをソフトウェア設計に反映させるのが主な作業である。図1にはまた、各々の段階の作業で処理の

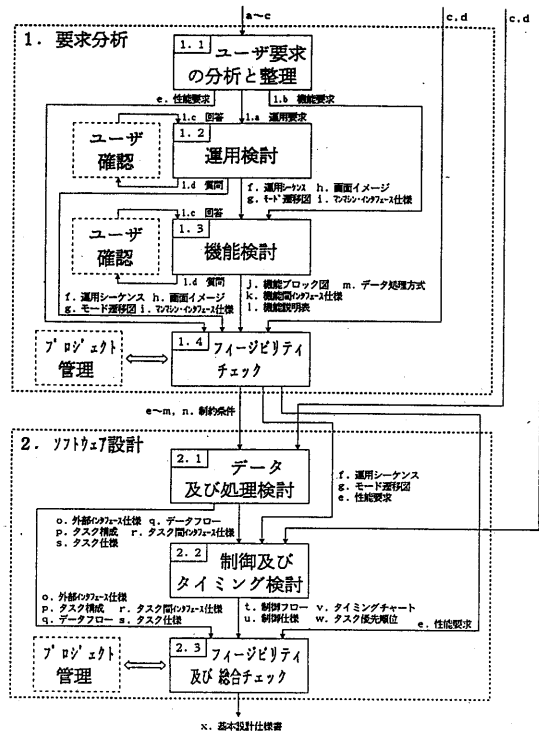


図2 実時間処理システム基本設計の第2階層ソフトウェア・プロセス

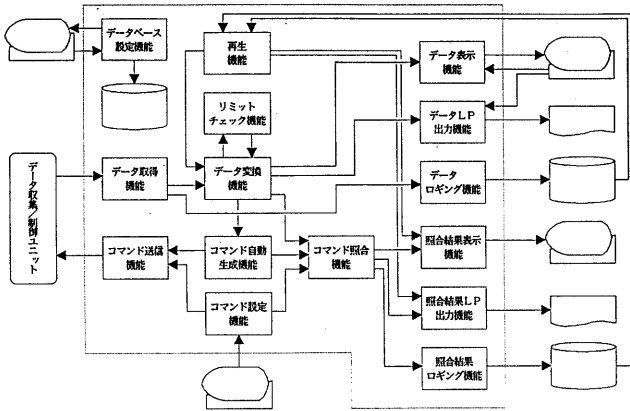


図4 オブジェクト例 j. 機能ブロック図

期間で設計作業全体が把握できるような教育教材と教育方法を開発しようというものである。

そのために、我々は2. 1に示したような流れでソフトウェア・プロセスの分析を行いその結果の再利用を検討してきた。

上記の様な試みの中で明らかになってきたことの一つは、優れたSEは基本設計段階において特に以下の2点に優れた特性を示すということである。

- ・ フィージビリティ・スタディ
- ・ 設計作業の先読み

すなわちこの2点がSE教育のポイントといえるものと考えられるが、これらについて以下に説明する。

3. 2 基本設計におけるフィージビリティ・スタディ

ソフトウェアの応答性に関するフィージビリティ・スタディは、要求された時間性能を設計上満足できるかという検討であり、この検討が、ソフトウェア仕様の詳細化された後の設計段階において行われるのは、当然のことであるが、実際には、初期のソフトウェア仕様検討段階においても同様に行われている。

但し、設計の上流過程（初期設計）になるほどこのフィージビリティ・スタディの内容は簡易化されたものとなる。

すなわち、初期設計段階におけるフィージビ

リティ・スタディはおよそ次のように表現できよう。

(1) 最も初期段階におけるフィージビリティ・スタディ

要求仕様が未だ極めてマクロ的にしか表現できない時期にも、一部の重要な機能の応答時間に対する要求が有り得る。

このような場合、通常はそのSEの過去に経験したシステムの処理内容（処理量及びその複雑性）と今回のシステムの処理内容とを感覚的に比較して、その結果から今回のシステムの応答性を類推する

のである。従って、この初期段階におけるフィージビリティ・スタディは、不確実性が強い。

(2) 要求仕様がある程度まとまった段階におけるフィージビリティ・スタディ

システムに関する要求仕様がある程度まとまった段階においては、不確実な情報の基に応答時間を算定し予め設定された目標性能を満足するか否か検討する必要がある。

このような時には、ソフトウェア構造を正確に設計しないままトータルな応答性能を見積ることになる。

このように、詳細化されつつあるシステム仕様を基に行う設計上のフィージビリティ・スタディは、最初は、きわめてマクロ的に行い、詳細な情報が揃うに連れて次第に本格的検討に移ると云うように進めている。

ここで述べているフィージビリティ・スタディは、システムの処理を実行するときの応答性能を対象としている。

一般にプロジェクト活動を進めるに当たっては、この応答性能以外に工程、コスト、品質などいくつかの制約条件があり、それを守りつつ設計を行うことになる。プロジェクト管理（本論のシステム設計とは、別階層のソフトウェア・プロセスとして記述することにしてはいる。）には、このような制約条件を対象としたフィージビリティ・スタディが重要な位置を占めており、前述のフィージビリティ・スタディで検討されたデータを基にして行われる。即ち、ソフトウェア設計の結果からそのソフトウェアの規

構及びその複雑さを見積り、これを開発するための工程、コスト、体制などを含むプロジェクト計画を立案することになる。

3. 3 基本設計における設計作業の先読み

これまでに述べてきた基本設計作業の各要素は、実際には、必ずしもこのソフトウェア・プロセスの示す順に（すなわちWaterfall Model的に）、一つの作業が完了した後に次の作業が開始されるわけではなく、いくつかの作業が同時並行的に行われている。

即ち、一般に設計作業の各項目はその一つ一つを独立に実行して設計を進めていくと、いずれは前後の設計項目間のインターフェイスに不具合を生じ、その調整のために大きな手戻りを生じる結果となり、従って、一つの時点に注目すると対象となる設計項目を実施しながら、その設計項目とインターフェイスを持つ後工程の設計項目とのインターフェイスを意識して設計を進めている。

このような設計作業の先読みの状況は、設計対象であるシステム規模の大小により異なってくる。小規模システム及び大規模システムの場合

の並行動作の状況を比較すると表1のようになる。この表は、設計作業のフェーズの進捗と共にいくつかの作業が同時に進められる様子を現しており、表の中の円の大きさは、そのフェーズに於けるその項目の作業量を現している。

すなわちこの表は、小規模システムに比較して大規模システムを設計する場合には、相当に先の作業まで並行して進めている様子を現している。これらの関係は、図5のように、設計項目及び設計フェーズの2次元平面上に表現することもできる。

ソフトウェア・プロセス中のこれらの設計項目の実行順序及び並行作業は、システム毎に異なり、そのメカニズムを記述することは困難である。この関係は、設計技術者が独自に把握すべきものであり、我々は適当な設計課題を与え、このようなことを考える訓練を行うことによりSE教育をしようとしているのである。

4. SE教育の試行と評価

4. 1 SE教育の試行

今回実施したSE教育のやり方は、概略以下

表1 システム設計フェーズにおける検討項目の比較
(表中の円の大きさは作業量を示す。)

作業項目 作業フェーズ	作業 開始	要求分析			ソフトウェア設計			
		ユーザ要求の 分析と整理	運用検討	構成検討	フェーズ別 設計	データ及び 処理検討	制御及び バリエーション 検討	フェーズ別 設計 及び 結合検討
1		○	○	○				
2			○	○	○			
3				○	○			
4					○	○	○	
5						○	○	○
6							○	○
7								○

作業項目 作業フェーズ	作業 開始	要求分析			ソフトウェア設計			
		ユーザ要求の 分析と整理	運用検討	構成検討	フェーズ別 設計	データ及び 処理検討	制御及び バリエーション 検討	フェーズ別 設計 及び 結合検討
1		○	○	○	○	○	○	
2		○	○	○	○	○	○	
3			○	○	○	○	○	
4			○	○	○	○	○	○
5					○	○	○	○
6						○	○	○
7							○	○

小規模システム

大規模システム

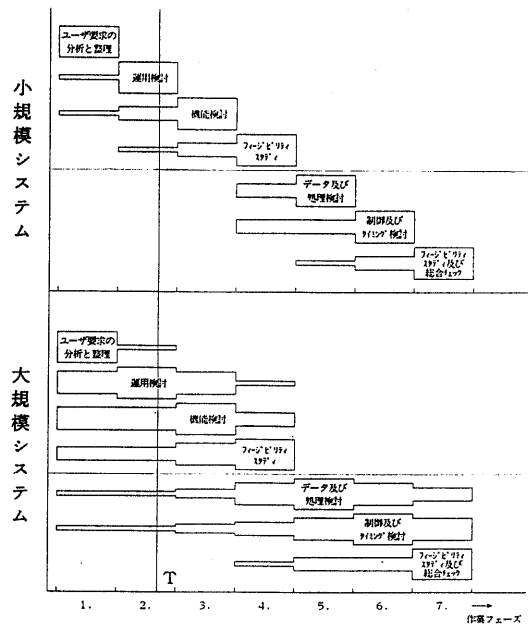


図5 ソフトウェア・プロセスと時間経過の関係 (T時点の各作業の断面の大きさは作業量を示す。)

の通りである。

まず受講者にこれから行う授業の主旨、内容、実施要領について簡単な説明を行う。さらに、対象となるシステムの概要と設計概論（大雑把に全体が把握できる程度）を説明し、ここまですを第一段階とする。

次に、例題として既に熟練SEが設計を行ったソフトウェア・プロセスを基に、その内容をオブジェクトを示しながら説明する。これを第二段階とする。

つづいて、練習問題を用いたシステム設計の演習に入る。この時第二段階で示したソフトウェア・プロセスとオブジェクトをその都度参考にしながら演習をすすめ、講師はその中で設計のポイント、ノウハウを伝達するよう心がける。これを第三段階とする。

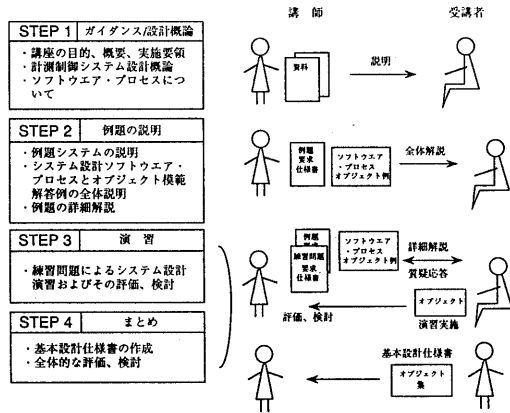


図6 SE教育実施要領

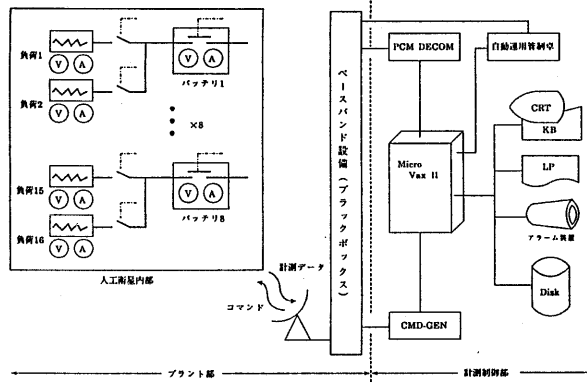


図7 練習問題用モデルシステムの概念図

最後に第四段階として、それまでに演習で作成したオブジェクト類を基本設計仕様書にまとめ、全体の評価、検討を行う。

図6に上記のSE教育試行の実施要領を示す。また、受講者の練習問題として用意したシステムの概念図を図7に示す。この仮想的なモデルシステムは、比較の実用システムに基づいた応用範囲の広い計測制御システムを想定しながら、しかも教育用としてなるべく簡略化してSE教育としての本質を外さないという事に留意して設定した。

今回の試行においては、SE教育を施す対象である受講者（1人）として、以下の要素技術を有している者を選択した。これは、今回の試行が教育システムそのものの評価、検討という要素が強いため、教育のし易さという点を特に考慮したためである。

- ・コンピュータ標準機器及びOSに関する知識
- ・リアルタイム（マルチタスク）システムプログラミング技術（経験1年半）
- ・基本設計以降の業務経験
- ・SA/SD設計手法に関する基礎知識

4.2 SE教育用ツール

SE教育用ツールとして今回作成したものは、実際のシステムのひな型としての教育用モデルシステムの要求仕様書と、それに基づいて実際に熟練技術者が設計を行った時に作成したソフトウェア・プロセス例及びオブジェクト集、さらに練習問題として与える別のシステムのシステム要求書である。これを図8に示す。

4.3 SE教育の評価

教育終了時点で受講者が作成した基本設計仕様書は、技術的に修正すべき箇所がいくつもあるものの、ほぼ実用可能な（基本設計以降の開発作業に十分移行できる）仕様書としてまとまったものであった。

また、今回の試行の結果、ソフトウェア・プロセス及び教育システムに対する受講者の感想が得られたので主なものを以下に示す。

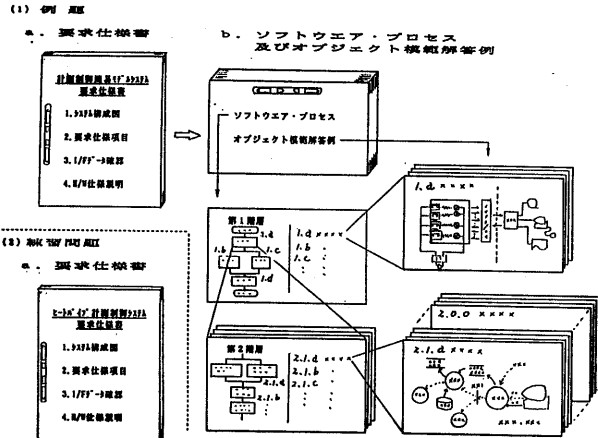


図8 SE教育用ツールの構成

- ・オブジェクト及びソフトウェア・プロセスで誘導されると設計作業を進め易い。
- ・要求分析から始まる基本設計の流れを少なからず身につけることができた。
- ・今回の試行の例題及び練習問題のレベルは短期間のシステム設計教育訓練に適している。
- ・現行のプロセスでは、どのフェーズで実現性検討やユーザへの確認を行うべきかがあいまいである。(改善を要す。)
- ・各オブジェクトに詳細な補足説明があると設計しやすい。

最後に今回のSE教育試行の結果得た教育用ツールへの評価をまとめると次の様になる。

- (1) 今回のソフトウェア・プロセス及びオブジェクトを中心とした教育は、基本設計技術習得のガイドラインとして有効なものと考えられる。
- (2) 受講者の設計技術力向上の定量的な評価は困難であるが、ほぼ実用的な基本設計仕様書ができた事により、基本的には有効なものであると判断できる。
- (3) 実現性検討やユーザへの確認の作業をソフトウェア・プロセスに適宜追加し、教育用ツールを実業務により近づいたものにする必要がある。

(4) その他、現行の教育用ツールには内容的な改善点がいくつかあげられる。特にソフトウェア・プロセスにおいては設計上のポイントを併記してノウハウをも理解しやすく補強することにより、手引書としてもより有効なものとする必要がある。

5. まとめと今後の課題

実際のプロジェクトを対象に、その基本設計段階のソフトウェア・プロセスを分析し評価、改善を行った結果、最終的にかなり現実に近いソフトウェア・プロセスが得られたと考えている。

また、この成果をSE教育に利用すべく検討を行い、

- ・教育用ソフトウェア・プロセスの設定
- ・教育シナリオの構築
- ・教育用ツールの基本的な整備
- ・SE教育の試行

を行った。これらの試みによって、ソフトウェア・プロセスの利用によるSE教育の有効性が確認できた。また、今後内容をより充実し教育システムとして実用化できる見通しがついたものと考えている。

今後、SE教育システム実用化に向けての課題は、以下の通りである。

- ・教育用ソフトウェア・プロセスの見直し
- ・ノウハウを盛り込んだSE教育用手引書の作成検討
- ・実習時間等を考慮した複数の受講者に対する講座形態の検討
- ・SE教育試行の積み重ねと教育システムの充実化
- ・プロジェクト管理教育への発展

参考文献

- [1] L. Osterweil, "Software Processes are Software too," Proceeding of the Ninth International Conference on Software Engineering, Monterey, California, pp. 2

- [2] T. Katayama, "A Hierarchical and Functional Software Process Description and its Enaction," Proceeding of the 11th International Conference on Software Engineering, pp. 343-352, 1989.
- [3] 望月、山内、片山他：ソフトウェア・プロセス－実時間処理システムにおけるケース・スタディ－：情報処理学会研究報告、90-SE-71、P P. 139-148 (1990)
- [4] 望月、山内、片山他：ソフトウェア・プロセスの設計教育用ツールへの適用及び評価：情報処理学会研究報告、90-SE-73、P P. 83-90 (1990)
- [5] 望月、山内、片山他：ソフトウェア・プロセスの分析及び評価：第41回情報処理学会全国大会論文集 2G-8、(1990.09)
- [6] 望月、山内、片山他：ソフトウェア・プロセスを利用した教育用ツールの開発：第41回情報処理学会全国大会論文集 2G-9、(1990.09)
- [7] S. Mochizuki, A. Yamauchi, T. katayama et. al., "Applying the Software Process to the Instruction Tool in System Design," Proceedings of 6th International Software Workshop, 1990.
- [8] S. Mochizuki, "Design Process of RealTime Systems," Proceedings of International Software Process Symposium, 1990.