

アプリケーションプログラムの流通商品化

荒川淳三

札幌大学経営学部

コンピュータの発展普及にともない、産業界ではシステムエンジニアの不足が深刻な問題になっているが、最大の問題は、アプリケーションプログラム(AP)の生産技術である。AP生産の技術進歩が遅い根本原因は、個々のユーザ企業が自社の必要とするAPを自社で開発するという前近代的な生産体制にある。問題の抜本的解決はAPの流通商品化を進めることによって始めて可能で、これを行うことがソフトウェア工学の焦眉の課題であろう。この問題の体系的分析と、APの流通商品化のためにソフトウェア工学に期待される具体的課題の抽出を試みた。問題をより正確に把握説明するためには、ソフトウェア生産技術の現状の幅広い調査等が必要である。

COMMERCIALIZATION OF APPLICATION PROGRAMS

Junzo Arakawa

Faculty of Business Administration

Sapporo University

3-7-3-1 Nishioka, Toyohiraku, Sapporo, Japan

In industry today, computers are getting more and more popular and are used in almost every field. Deficiency of software engineers is now a serious problem. Large part of the problem is due to the poor productivity in application programs(AP), which is caused by the fact that each of APs are basically developed by its user company. Production of APs has to be modernized, has to be implemented by companies specialized in software production. This reorganization in AP production is the biggest problem which is expected to be solved by software engineering researchers.

1. はじめに

ソフトウェア危機という言葉が広く使われるようになってから久しい。この言葉の最初の意味はプログラム生産における分業の困難さに関するものであった。プログラム生産における分業の難しさはそれに参加するプログラマの数とともに加速的に増大するので、ある規模をこえるプログラム（システム）の生産は不可能であるというのが、その意味するところであった。

この問題は今日も解決しているわけではない。しかし情報化の急速な進展でコンピュータシステムが社会の重要な下部構造となり、また新規プログラムおよび既存システムのメンテナンスに対する需要が巨大化したことにより、今日のソフトウェア危機の意味するところは、ソフトウェア供給能力の量的、質的不足に変わってきている。

ソフトウェアは、コンピュータシステムにおける位置づけによって、オペレーティングシステム（OS）、ミドルソフト、アプリケーションプログラム（以下にAPと略記）に大別できるが、この中で危機の様相が深刻化しているのはAPである。今日のAPの生産は多くは一品生産的に行われ、生産のための費用は高額であり、工期は極めて長期におよぶ。また生産過程で生じる全てのバグを、単一のAPが負わねばならない（表1）。

本論文では、まず、今日のAP生産における問題点を段階的に整理・説明し、その根本原因が今日のAP生産において大勢を占める、前近代的な自製体制にあることを明らかにする。

APの生産体制の近代化はAPを完全な流通商品にすることで達成されるが、これを実現するためには何が必要かを次に考える。具体的には流通商品としてのAPのライフサイクルを丹念に追いかけて、個々のフェーズでどのような問題が生じるかを明らかにする。

最後に以上の結果からAP流通商品化のソフトウェア工学的課題を体系的に示し、この問題に対するソフトウェア工学者の取り組み方について提案する。

種類	汎用性	安定性	納期
OS	機種に依存する汎用的商品	高品質、バグを多数ユーザに分散	既製品、ほぼ即納
ミドルソフト	OSに依存する汎用的商品	高品質、バグを多数ユーザに分散	既製品、即納
AP	多くは非汎用、個別対応製品	バグの危険性を単一ユーザが被る	数年にもおよぶ開発工期が必要

表1. プログラムの種類と利用上の特性

2. AP生産上の問題点

一部のAPは今日でも汎用パッケージとして流通商品化されているが、大手のユーザ企業における大勢は、自社の必要とするAPは自社の責任で開発するという自製体制である。ここではこの体制でのAP生産の問題点を整理し、その原因を探る。

2.1 製品としての問題点

2.1.1 コストに関する問題

第1の問題点は、APのコストが非常に大きいことであり、一つの試算では、大手ユーザにおけるAPの費用は、全費用の50%を上回る¹⁾。

第2の問題点は、個々のAPに関する費用が、正確に把握されていないことである。プロジェクトとしてのAPの開発は、当然予算化して進められるが、予算化の範囲は通常粗設計からインストールまでの範囲であり、システム分析以前とメンテナンス段階を包含したライフサイクルとしての把握はほとんどなされていない。

また自社社員の労務費の把握が不完全である企業が多く、予算自体の算出精度がよくないのに加

えて、表面的な予算金額を節減するため、プロジェクトの一部がメンテナンスの名目で行われることもある。

同一のAPについての初期開発の費用とメンテナンス費との分離も、データ精度に関する大きな問題である。APの設計製作が的確に行われればメンテナンスの費用が激減することは、多くの調査研究結果が示すところであるが²⁾、現時点ではAP稼働の状態で行われるプログラムの修正は、全てがメンテナンスとして扱われる。

精度の高いデータは科学的分析の基礎であり、AP生産の合理化を進めるためには、データ精度の向上から取り掛かる必要がある。

2.1.2 品質に関する問題

品質に関する最大の問題は、具体的な品質特性についての定量的保証が行われていないことである。これはAPが近代的な財としての市民権を得ていないことを意味する。

適正な機能が作りこまれているかどうかの機能品質については、不要な機能がどの程度存在しているかが不明である。Boehmらはプロトタイピングの実施により、APの規模を40%縮小できると報告しているが、これは今日のAPの抱える不要な機能の比重を示唆するものである²⁾。

2.1.3 工期に関する問題

工期に関する第1の問題は、それが非常に長いことである。コボル言語で10万論理行を越える中規模以上のプロジェクトは、大抵は2年以上の開発工期を必要とするが、これは技術進歩の速い今日では、非常識と言える値である。

第2の問題は、工期の定義があいまいなことである。プロジェクトの予算化に先行する基礎調査あるいはシステム分析の期間、プロジェクトが不完全な部分を残したまま立ち上がったあとの手直しの期間を加えると、開発工期の実態は、さらに長くなる筈である。

2.2 見積り精度の問題

AP開発に必要なコスト、工期の見積り精度は極めて悪い。図1A、図1Bは、他の文献から引用したものであるが³⁾、ほとんどのプロジェクトにおけるプログラムサイズの実績は、初期の見積り値のほぼ2倍に達している。当然のことながら、プログラムの規模の増大は、必要な開発コストを増加させる。

開発工期の実績値も全てのプロジェクトにおいて計画値を上回っているが、いずれの場合にも、工期の超過がほぼ2カ月であるのは興味深い。

コストや工期が計画値を常に上回るのは、それらが高価に過ぎ、かつ長期に過ぎるからであることは明白であるが、なぜこれほどまでに大きな誤差が生じるのかは極めて興味深い問題である。

2.3 ソフトウェア工学の適用状況

ソフトウェア工学の成果のAP生産への適用に関しては、2つの点が際立って明確である。

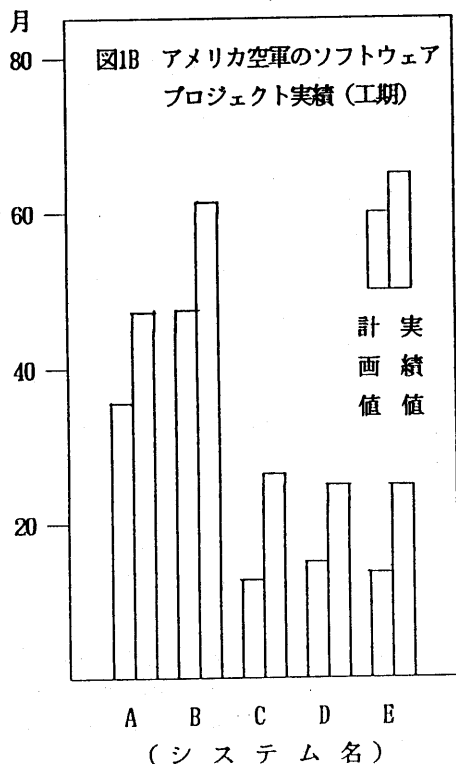
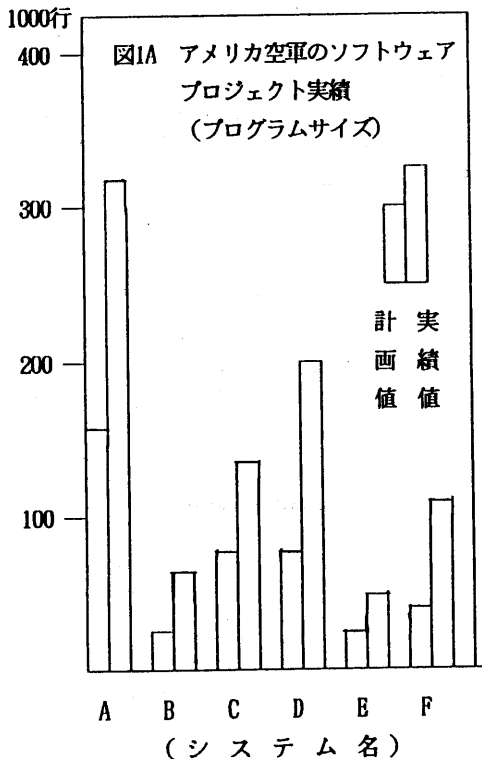
1つは、評価の確定した方法論を実務に適用すると、多くの場合素晴らしい成果が得られるということである。

BSP、データフロー図を用いた構造化分析、プロトタイピング、部品化・パターン化による再利用技術、第4世代言語などがこの代表例である。

もう1つは輝かしい成功例にもかかわらず、これらの技術が、少なくともユーザ企業のシステム部門においては、なかなか普及定着しないことである。今日、ユーザ企業の多くはコボル、PL/Iなどの第3世代言語に移行し、いわゆるウォーターフォール型のプロジェクトライフサイクルを標準化し、コーディング以降の開発支援環境を持ち、相当の標準化を行っているが、これを越えてのソフトウェア工学的武装は貧弱である。

2.4 ソフトウェア工学成果不適用の理由

APの生産が多くの問題を抱え非常に深刻な状態にありながら、ソフトウェア工学の成果が極めて不十分にしか活かされていない根本的理由は、AP



の生産の大勢がそれを必要とする企業によって行われるという、前近代的な自製体制にある。

自製体制がAP開発技術の阻害要因になる理由は次の通りである。

(1) 自製体制においては健全な競争環境が損なわれる。ユーザ企業内のシステム部門はありあまる業務量を独占的に抱えた存在で、個々の構成要員は非常な努力をしていますが、その本質は売手市場における独占企業的である。

(2) ユーザ企業内のAP生産は、基本的には多品種一品生産である。ソフトウェア工学の原理は一度作られたソフトウェアを多数回利用することにあるが、多品種一品生産の環境下では、ソフトウェアの再利用は困難である。

(3) ソフトウェア工学の所産としての方法論の多くは技術的に高度であり、十分な教育訓練を受けた技術者でなければ使いこなせない。再利用可能なソフトウェアを蓄積するためには、周到な計画と高度な設計製作技法とが必要である。

ユーザ企業内の開発集団はこれに耐えるだけの専門性に欠け、ソフトウェア工学の成果を大きな体系として取り込むには、力量が十分ではない。

3. AP生産体制の近代化

APの近代的生産体制は、APを必要とするユーザがこれを専門生産企業から購入する体制である。この体制に関して特記すべき事項は次の通りである。

(1) APの生産技術がユーザ企業から離れ、専門企業に移る。

(2) 個々の専門企業がユーザ企業全体を営業対象とするため、AP再利用の可能性が大きくなる。

(3) 熾烈な健全な企業間競争の醸成が期待できる。

(4) 新体制はユーザ企業の経営にとって有利に作用しなければならない。従来に比してより高品質のAPが、より安価に、かつ必要なときに入手できなければならない。

このような体制は、前述のOS、ミドルソフトにおいては、すでに確立されていることに注意する

必要がある。ここではメンテナンス作業がソフトウェアの供給者によって行われているのに注意すべきである。

OS、ミドルソフトとAPとの間には、ユーザごとの特殊性のレベルにおいて大きな差があり、ユーザの特殊性に対応するためのカスタマイズ作業を如何に効率的に行うかが、新しい体制の成否の鍵である。このためにはカスタマイズ作業そのものが十分に小さいこと、カスタマイズ作業を行う専門企業側がユーザの作業を熟知していることが必要になる。すなわち専門企業は

- (1) 特定のAP領域に特化すること
- (2) この領域をユーザ以上に理解した技術者を育成すること

が必要になる。

また専門企業の提供するシステムは、最新の技術を十分に活用し、ユーザにとって有利なものでなければならない。すなわち専門企業は、自己の特化した領域における最新技術を反映したモデルシステムを開発・維持し、このシステムのカスタマイズによってユーザの要求に対応する体制を確立しなければならない。

このような体制は、OSやミドルソフトを含む近代的商品においては、既に確立されているものである。

4. 新体制確立における問題点

APを専門企業から購入する体制を拡大していくときに何か問題なのかを明らかにするために、流通商品されたAPのライフサイクルを想定し、ここにおける作業の流れを追跡する。想定されるライフサイクルは図2の通りである。

図のフィージビリティスタディは、新しいシステムの導入が経営上有利かどうかの基礎調査であり、次の詳細計画に入るか否かが決定される。

詳細計画ではフィージビリティスタディの結果を受けて、現状の問題を解決するためのいくつかの有力な案について詳細な検討が行われ、費用効果等から最良の案が選ばれ、予算が確定する。

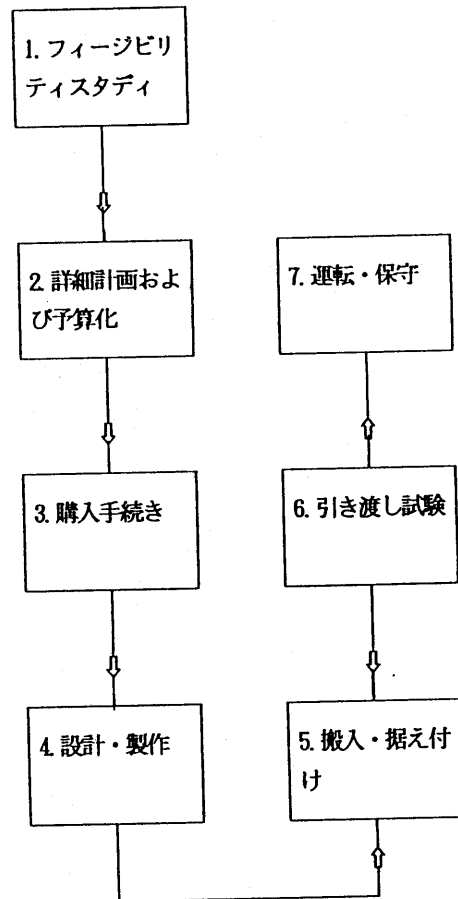


図2 流通商品としてのAPのライフサイクル

購入手続きは購入仕様書による見積り照会、見積り書提出、発注先と発注条件の決定等である。

納入者は契約に基づきシステムの設計・製作、据え付けを行い、引き渡し試験で契約内容が確実に履行されていることが確認される。

運転・保守の段階でのAPの特徴は、機械や電気設備に見られる劣化が全くないことである。

4.1 購入か自製かの選択

あるAPが流通商品としてのライフサイクルに入る前に、ユーザ企業はこれを外部から購入することが経営上有利であることを確信できなければならない。この確信は、一般には自製と購入におけ

るAPの品質、価格、納期および保守運転費を比較して行われるが、APの購入が常識化するまでの過渡期においては、次の努力が必要である。

(1) 品質、価格、納期の定義を明確にすること

前述の通り、今日の自製体制においては、予算確定前の企画段階の費用、システム立ち上がり後の不都合箇所の手直しの費用が多くの場合欠落しており、また自社社員の労務費の把握方法にあいまいさがある。自製の場合の費用の把握精度を改善しないと、導入費用の適正な比較はできない。

APの品質については、定量的な指標が確立されておらず、APの開発プロジェクトにおいても、具体的な品質の保証は行われていない。品質は流通商品の基本特性であり、その定義と保証方法の確立が急がなければならない。

(2) 品質、価格、工期の見積り精度は極めて不十分である。自製と購入の比較は、見積り結果の比較であり、精度の良くない推定値の比較では、微妙な差は判定できない。二つの推定値精度の向上が必要である。

(3) 当然のことであるが、購入の方が圧倒的に有利である状態を築かねばならない。このためにはモデルシステムをカスタマイズして納入APを生産する技術の確立が急がなければならない。

(4) 保守体制の確立

新しい体制における保守作業は、システムに異常が生じたときの復旧作業である。具体的にはシステムをリセットして立ち上げる作業と、システム異常によって生じた外的な不具合の後始末とであって、APの機能追加や改善は、原則として外部の専門企業によって行われなければならない。

ユーザがAPを安心して購入できるためには、このような保守体制が整備されなければならない。

4.2 フィージビリティスタディの問題点

フィージビリティスタディの目的は、次の事項を短期間で安価に行うことである。

(1) 現行システムの機能、処理能力、信頼性などについての問題点を明らかにする

(2) 新システムの備えるべき仕様を明らかにする

(3) 新システムの制約条件を明らかにする

(4) 新システムのいくつかの有力案を作成する

この業務は基本的にはユーザ企業によってなされなければならないが、APの生産技術が外部にある状態では、ユーザ企業は複数の専門企業の意見を聞き、仮見積りをとるなどして、これを行うことになる。ユーザからの問い合わせに対して最新の技術を取り込んだ案を提示し、十分な精度の見積りを返せるだけの技術の蓄積が必要である。

4.3 詳細計画および予算化の問題点

自製体制においては、このフェーズでは以下の作業が行われる。

(1) 新システムと外部環境との相互作用をモデル化する

(2) 新システムの動作をモデル化する

(3) 新システムにおける機械と操業者の境界を決める

(4) 新システムの設置、品質、性能等についての制約条件を定める

(5) フィージビリティスタディで示された各案についての費用効果分析を行い、最適案を選ぶ。

(6) システム仕様書を含む予算書を作成し、予算申請、予算審議をへて、予算を確定する。

APの生産が外部の専門企業によって行われるときは、このフェーズは全く変わったものになる。新システムの機能と外界とのインターフェースが確定し、これを購入するための費用が定まれば十分なので、これまで最大の負荷を必要とした(2)項は、ユーザから見た機能の記述だけでよい。費用の算定は専門企業の仮見積りなどを参考に決めれば足りる。従来膨大な文書であったシステム仕様書も、大いに簡略化される筈である。

新システムについての納入者責任を明確にするためには、外部環境との結合は、できる限り疎でなければならない。新システムは専用のハードウェアにインストールされ、APの設計製作標準は納入者に一任される。また周辺システムとのデータ

の共用形態も、可及的に疎にすべきである。

ハードウェア的にもソフトウェア的にも疎な相互結合を持つサブシステム群で有機的に機能する系を構築する技術が追求されねばならない。

4.4 購入手続きの問題点

ユーザ企業から購入仕様書に基づく見積り照会がなされ、外部の専門企業が見積り仕様書を提出する。ユーザ企業は見積り仕様書の内容が自社の要求仕様に合致することを確認する。

APの仕様を文面で確認することは不可能に近い。ため、この過程ではプロトタイプングが必須である。専門企業は自社のモデルシステムを用いてユーザ企業に対するプロトタイプングを行い、ユーザの要求を確認する。またこの要求に応じるためのカスタマイズ作業を考慮して見積り金額を算定する。

すなわち、専門企業には、ユーザの要求するシステムに十分に近く、カスタマイズ容易なモデルシステムが無くてはならない。

4.5 設計製作の問題点

前述の通り、ユーザの要求するAPを一から作ると、現在の技術水準では非常に高価になり、非常識なほどに長い工期を必要とし、品質の保証は難しい。すなわちAPの商品化のためには、モデルシステムをカスタマイズして納入するという体制が不可欠である。

モデルシステムは最先端の技術を十分に活かした合理的なシステムでなければならず、個々の専門企業は、自社の特化領域におけるモデルシステムを開発保有し、技術の進歩に合わせてこれをアップデートしておかねばならない。このようなシステムの開発維持には大きな資金が必要であり、これが成り立つには、一度開発されたモデルシステムには十分に多くの適用の機会が必要である。

ユーザ要求に対するカスタマイズ作業量が十分に小さくて適用の機会が十分に多いモデルシステ

ムが可能かどうかは大きな問題であるが、この存在は新しい体制にとって必須である。これを可能にするための技術の開発は、今後のソフトウェア工学の大きな課題であろう。

カスタマイズ作業の品質、工期、コストが改善されれば、モデルシステムの開発はそれだけ容易になる。これもまたソフトウェア工学の重要な課題である。

4.6 搬入・据え付けの問題点

設計製作と十分な出荷試験を終えたAPは、ユーザ企業に搬入、インストールされる。このときの品質性能が確実に保証されるためには、この新システムと周辺の既存システムとの結合は、可及的に疎でなければならない。疎の結合が保証される限り、ここでは大きな問題は生じない。

4.7 引き渡し試験の問題点

引き渡し試験での大きな問題は、試験ルールの確立である。新システムあるいはAPの設計製作は見積り仕様書に基づいてなされるのであるから、引き渡し試験は見積り仕様書に基づいてなされなければならない。設計製作の過程でユーザの要求や設計上の都合などで生じた変更は、見積り仕様書に確実に反映されていなければならない。

4.8 運転保守の問題点

ここにおける最大の問題は、保守作業の範囲の規定である。劣化の生じないAPでは、保守作業は原則として異常発生時のシステムの復旧と、異常時に生じた後遺症に対する処置に限定される。APの追加改善が保守作業に含まれると、ユーザ企業はAPの設計製作技術を維持しなければならず、APの流通商品化は成り立たなくなる。

ユーザが安心して任せることのできるAPの追加改善体制の確立が要求される。

5. ソフトウェア工学的課題

以上の課題は次のように体系化できる。

(1) AP生産の専門体制への移行促進

第1は現実の、および可能性としての専門体制の優位性を明らかにすることが重要である。

このためには先ず、APの品質、コスト、工期の定義を確立し、データ収集を容易にし、正確な比較を可能にしなければならない。

次に現在の大勢である自製システムと、専門企業により開発納入され保守されている一部のシステムとを調査比較し、品質、コスト、納期についての差異を明らかにすることが必要である。

この結果から、専門体制確立の社会的意義が理解できるに違いない。

(2) ユーザ支援技術の確立

フィジビリティスタディを効率的に推進する技術、特にユーザと専門企業の協力を円滑に進めるための技術、十分に簡単な購入仕様書によって必要なAPを購入することを可能にする技術、プロトタイピングを主体とした仕様確認技術などの確立が必要である。

(3) AP専門生産技術の確立

適用対象が広くてカスタマイズ作業量の少ないモデルシステムの開発技術、効率的なカスタマイズ作業を可能にする技術の開発が焦点となる。

また納入されたAPの追加改善を効率的に実施することを可能にする技術も重要である。

(4) 新システム納入を容易にする技術の確立

ハードウェア的にもソフトウェア的にも可及的に疎結合の状態にあるサブシステムで有機的な機能を実現する技術の確立が必要である。

(5) 商取引ルールの確立

契約条項を引き渡し試験で確認することを骨子とした商取引ルールの確立により、APの売買が構成に行われる体制を確立することが必要である。

6. むすび

産業界で使用されるAPの生産は、洋の東西を問

わず、大勢としては自製されている。

またこの事実に起因すると思われるが、APの仕様は、その最終的な利用者であるエンドユーザが詳細まで決めるという傾向が強い。

この結果AP生産現場の技術は遅々として進歩せず、社会の要請からはほど遠いほどの高コスト、低品質、および非常識とも言える長い工期が定着している感がある。

近代社会における財の生産技術は生産体制の専門化によってなし遂げられたが、APの生産でも生産体制の専門化が必要なのである。今日のソフトウェア工学に最も期待されるところは、AP生産の専門化を促進し、これによってAP生産技術の革命的な進歩を可能にすることであると信じる。

AP生産の専門化は、裏返せばAPの流通商品化であるが、ある規模を越えたAPの開発では工期も品質も保証できない今日の技術では、APを流通商品化する唯一の方法は、専門企業が特定の対象についてのモデルシステムを開発維持し、このカスタマイズによって製品を納入するしかない。

本論文では、このために必要な技術的課題を指摘し体系的に示した。内容的に満足すべきものではないが、AP生産の専門化を促す一つのきっかけになることを願ってこの論文を作成した。

筆者個人としては、ソフトウェア企業の生産技術に関する実態調査から入り、ここに指摘した課題の解明に取り組む計画を進めている。

参 考 文 献

- 1) 荒川: アプリケーションソフトウェアの品質、コスト、工期, 日本CHIDE/SHARE, AUTUMN, 1987
- 2) Charette, R. N.: Software Engineering Risk Analysis and Management, McGraw Hill, 1989
- 3) Boehm B. W.: Prototyping Versus Specifying: A Multiproject Experiment, IEEE Transactions on Software Engineering, May, 1984