

わが国の計算科学技術ソフトウェア開発のありかた

小池 秀耀*

Problem of Software Development of Computational Science in Japan

Hideaki Koike*

1. なぜソフトウェアの開発でわが国は立ち後れているのか

今日、コンピュータ・シミュレーションは科学技術や産業技術の共通基盤技術として、イノベーションや産業競争力の鍵を握っている。コンピュータの目覚ましい進歩によりコンピュータの性能を引き出せるアプリケーション・ソフトウェアが最も重要となっている。わが国は、残念ながら、産業界で使用されているソフトウェアに関しては欧米に大きく後れをとっている。表1に示すように、産業界で実際に使用されているソフトウェアのほとんどが欧米製である。

2. 後れている原因は何か

日本は欧米になぜ負けているのか？2004年に開かれた第3回「戦略的基盤ソフトウェアの開発」シンポジウムで三菱重工業株式会社 技術本部 高砂研究所 技監・主幹研究員（当時）川田 裕氏は三菱重工業時代の経験から次のように述べている。

◆欧米の成功の要因

- ・欧米では国家的援助により生み出された成果をベンチャー企業が商業化
- ・高学歴研究者を集中的に開発へ投入して、先進機能開発の迅速化と充実を実現
- ・大学における開発の成果を商業化できる強い仕組みがあり、また大学も産業界での実用化に熱心
- ・欧米には航空・宇宙を始めとする、シミュレーションを必要とする広い市場があり、先進的な新しいツールを受け入れた。

*アドバンスソフト株式会社 代表取締役社長
Representative director president , AdvanceSoft Corporation

川田氏の指摘はもっともであると思う。ただ、わが国の科学技術分野の実用ソフトウェアが欧米に後れている原因は必ずしも簡単ではないであろう。例えば、ソフトウェアを開発し事業化できる企業がほとんどない、わが国の大学や研究機関は実用的なソフトウェアを開発する力が弱い、優れたソフトウェア技術者が育成されていない、等々の原因が考えられるだろう。この現状を打破するには、やはり国の政策が重要となる。国が優れたソフトウェア開発や開発・保守体制を確立を先導する必要がある。従来、わが国の実用ソフトウェア開発戦略は不十分ではあったと思われる。長年、欧米の計算科学技術の動向をウォッチしていた先端情報技術研究所内田俊一氏も同様の意見である。

欧米各国はシミュレーション・ソフトウェアは産業競争力や科学技術の発展にとって基盤となる技術であると認識して国をあげて実用的ソフトウェアの開発に取り組んできた。しかも、開発は極めて戦略的である。図1は米国大統領のシミュレーション技術に関するアドバイザリー委員会がクリントン元大統領に提出した資料をアレンジしたものである。重要なことは応用ソフトウェアの開発は段階を踏んで行っており、各フェイズ毎に適切な国の資金の投入、担当研究機関を定めて取り組んでいることである。基礎研究の段階は米国科学財団が担当し、実証ソフトウェアの開発はエネルギー省や国防省が担当し、国費を投入する。そして実用化のフェイズではベンチャー企業等が商用化して、開発・保守資金は市場から調達するという戦略である。しかも、商用化した後も「戦略的調達」と称して、ソフトウェアの普及、利用を推進している。このような、戦略は米国の長年にわたる経験か

ら生み出されたもので、この戦略がうまく機能しているので、実用ソフトウェアに関しては米国が優位性を確立しているといえよう。このような戦略はソフトウェアの性格上極めて適切であることを、後で述べる。更に、米国はこのような開発戦略にリンクして人材、ベンチャー企業の育成を積極的に行っている。このため、米国のベンチャー企業が応用ソフトウェアの分野で次々と生まれ、それが米国の活力ともなっている。

有力な科学技術計算用大規模ソフトウェアの大半は国家プロジェクトや大学での実証レベルの研究開発をベースに、ベンチャー企業により事業化されている。

一方、わが国においては、科学技術分野に関して、実用ソフトウェアを開発する国の明確な戦略は存在しなかったと思われる。大学を中心に研究資金を配布して「基礎研究」を行っているのが大部分であろう。実用的ソフトウェアの開発は例外的にしか行われてこなかったし、実用ソフトウェアを開発する企業の育成も行われてこなかった。40年前の3000ステップ程度の時代とは違って、現在は、実用ソフトウェアの規模は100万ステップになろうかという時代である。実証ソフトウェアの開発プロジェクトなくして実用ソフトウェアは生まれない。では、実用ソフトウェアを民間企業が開発すればいいという意見もあるが、現時点では人材、資金の面で非現実的である。欧米も実証ソフトウェアの開発までは国が資金を投入している。科学技術には国境がないので、このような国の支援を受けた欧米のソフトウェアと民間企業が単独で競争するのは難しい。わが国では実用ソフトウェアを開発する機会が少ないので、実用ソフトウェアを開発できる企業が育たない。そうすると大学でいくら勉強しても、それを活かす場所が存在しないことになり、結局実用ソフトウェアを開発できる人材もいなくなるという「負の連鎖」を引き起こす。実際、わが国の実用ソフトウェアの開発技術は風前の灯火であり、経験者がいるうちに後継者を育成する必要がある。

これが「戦略的基盤ソフトウェア開発プロジェクト」を実施した主な理由である。

ただ、この10年くらいの間に、わが国においても実用的ソフトウェア開発、開発・保守体制を目指す国の事業も開始されつつある。とくに、戦略的基盤ソフトウェア開発プロジェクトと革新的シミュレーション・ソフトウェア開発プロジェクトは、世界レベルのソフトウェアの開発と同時にわが国のソフトウェア開発・保守体制の確立を目指したものであり注目すべきプロジェクトである。このプロジェクトは優れたソフトウェアを開発するとともに、産業界へのソフトウェアの普及体制も確立しつつある。また、このプロジェクトを通じて科学技術用実用ソフトウェア開発を専門とするベンチャー企業アドバンスソフト株式会社が生まれ、社員約70名（技術者のほとんどが博士）の人材を擁し開発したソフトウェアの販売、改良、保守を行っている。わが国の先端的ソフトウェアの開発・保守体制の確立に貢献している。わが国にはソフトウェア開発の基礎研究については国際的にも高いレベルにある。戦略的基盤ソフトウェア開発プロジェクトや革新的シミュレーション・ソフトウェア開発プロジェクトのような国の事業を継続的に強化していくならば、欧米に追いつくことも不可能ではないであろう。このようなソフトウェア開発事業を継続的に発展強化していくことが喫緊の課題である。

表 1 産業界でよく使用されているソフトウェアの例

ソフトウェア名	開発機関	ソフトウェアの内容
NASTRAN	NASA	汎用構造解析ソフトウェア
ABAQUS	ABAQUS.Inc.	汎用構造解析ソフトウェア
DYNA	ANSYS.Inc.	衝撃解析ソフトウェア
GAUSSIAN	カーネギーメロン大学	量子化学計算のソフトウェア
SPICE	カリフォルニア大学	回路シミュレータ
TCAD	スタンフォード大学	LSI 設計シミュレータ
Phoenics	インペリアルカレッジ」	汎用流体解析ソフトウェア
FLUENT	Fluent.Inc.	汎用流体解析ソフトウェア
STAR-CD	CD-adapco.Inc.	汎用流体解析ソフトウェア
CHEMKIN	サンディア米国立研究所	汎用流体解析ソフトウェア



図 1 米国のソフトウェア開発戦略

3. 実証ソフトウェアとは何か

ここで、先ほどから出てくる実証（＝実用）ソフトウェアとは何かについて説明する。ひと言でいえば実用で使えることを検証したソフトウェアである。基礎研究のソフトウェアは自分だけ使用でき、論文が書ければいいという

程度のものであるから、不特定の第 3 者の使用は想定していない。このようなソフトウェアは実用で使えるとはいえない。実用的なソフトウェアの要件をまとめると、ユーザーが誰でも安心して使用できるソフトウェアである。具体的にいうと、ちゃんと答えが出る（ロバスト

性)、答えが信頼できる、困ったときに相談できるサポート体制が完備している、長期的に改良が行われ時代遅れにならない、説明書が完備している、操作性がよくすぐ使えるようになる、というところであろう。また、ソフトウェアの性能以外にサポートが重要であり、サポートのないソフトウェアは実用的とはいえない。これらの条件を満足しなければ産業界で実用的に使用してもらえない。

ついでにフリーソフトウェアと商用版の差異について触れると、ほとんどの場合、フリーソフトウェアはメンテナンスや、継続的開発は不明瞭である。このため、自分でソフトウェアの改良・保守を行わなければならない、産業界では特別のユーザーしか使えない。一方、商用版はメンテナンス、技術サービスが行われており、ソフトウェアの専門家でなくとも利用可能である。また、将来的にも継続的に改良をが実施され必要に応じたカスタマイズもサポートしている。従って実務でソフトを使うユーザーは商用版を使用することになる。

4. なぜ基礎研究の枠では実用的なソフトウェアが開発できないか

大学等の基礎研究で実証ソフトウェアも開発できると思われる人もあるが、それは誤解である。(大学等が参加した実証ソフトウェア開発プロジェクトは除く)理由は簡単である。研究者は高性能なソフトウェアの開発に興味はあるが実用的なソフトウェア開発は、自分の本業ではないから無理である。従って、大学等で作成したソフトウェアは次のようになる。マニュアルのないソフトウェア、条件を変えるとすぐとまってしまうソフトウェア、GUIがない操作性の悪いソフトウェア、計算結果の正しさはユーザーが確認(保証はない)、サポートは一切ない、ほとんど必要なテストが行われていない。そもそも、大学等で開発するにはソフトウェアが大規模・複雑になりすぎた。すなわち、実用ソフトウェア・システムは10万ステップを超えるものが大半であり研究の片手間に実用的ソ

フトウェアを開発するのは困難である。大規模ソフトウェアは最も複雑な工業製品であることを認識する必要がある。つまり、実用ソフトウェアの開発は非常に難しく、図1に示すように基礎研究をベースとしつつも別の枠で開発を行う必要がある。それでも大規模ソフトウェアの開発は困難で、開発プロジェクトの半分は中止になるという米国の統計もある。

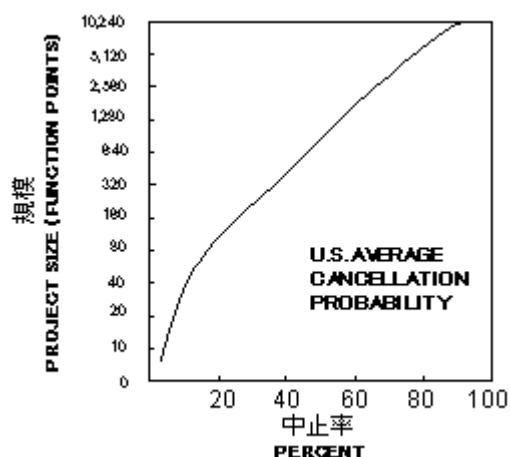


図2 ソフトウェア開発の中止率

出典：W.W. Gibbs: Scientific American, Sept. (1994) p86.

参考のために、米国のソフトウェア開発コンサルタントのヨードン氏がまとめたソフトウェア開発の難易度を表2に示す。

5. 実証ソフトウェアの開発には多くの費用と人材が必要

共通基盤技術のような実証ソフトウェアの開発には多くの優秀な人材と費用が必要となる。実これらのソフトウェアの規模は百万行に近くなっている。ヨードン氏の表から分かるように、優秀なプログラマーが100人近い規模で5年間の張り付くプロジェクトとなる。このような規模になるのはテストが格段に難しく手間がかかるためである。大ざっぱに言って大規模ソフトウェアのテストの手間はステップ数に幾何級数的に比例して増大する。実証ソフトウェア開発(商用ソフトウェアではない)にかかる費用を情報処理振興協会の先端情報技術研究所の内田

俊一氏米国の例から調査した報告書がある。国の開発のターゲットになる新規分野を開拓するような実証ソフトウェアの開発には5年間のプ

ロジェクトで年間2億円から4億円必要とされると指摘している。

基礎研究のソフト	実証ソフト	商用ソフト
◆革新的原理(モデル, アルゴリズム)の確認が目的 ◆実用問題への適用は莫大な労力と付随するソフト開発が大変なので基本的には実施しない。 ◆科学的に興味がある問題を解いて論文にして終わり ◆第3者が使用する場合のマニュアルも整備されていない。 ◆産業の日常業務では使用できない	◆基礎研究の成果を基に, 実用問題に適用できるようにソフトを改良整備し, かつ, 実際に実用問題を解析し, 実用問題が解けることを実証済みのソフトウェア ◆マニュアルの整備, バージョン管理など第3者が使用できるように整備されている。 ◆計算効率, 使い勝手等の完成度は, 必ずしも高くはない。 ◆保守, 改良等のサービスはない	◆各社の事業戦略に基づき計算速度の向上, 使い勝手の向上, 他のソフトウェアとの統合などの改良が行われる。 ◆保守, 利用上の各種サービスが用意される。 ◆絶え間ない改良が行われる
1960年代 ボーイング社などでコンピュータを使った構造解析(有限要素法)の研究実施	1970年, 汎用構造解析プログラム NASTRANをNASAが開発しフリーソフトとして公開(COSMIC版 NASTRANと呼ばれる) 従来の物づくりを一新	1971年, NASTRANの開発を担当したベンチャー企業 MSC社が商用版を発売。事業化デファクトスタンダードとなる。(MSC/NASTRANと呼ばれる)

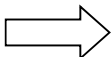
◆実証ソフトウェアとは何か  実用で使えることを検証したソフトウェア

図 3 基礎研究、実証ソフトウェア、商用ソフトウェアの比較

表 2 ソフトウェア開発の難易度

ソフトウェアの規模	難易度	開発人数	開発期間
1 千行以下	非常に簡単	1	数週間
1 万行まで	簡単	2,3	半年
10 万行まで	やや難しい	6~10	2,3 年
1 百万行まで	難しい	50~100	3~5 年
1 千万行まで	不可能に近い	経験者はほとんどいない	
1 千万行以上	非常識	Star Wars 計画	

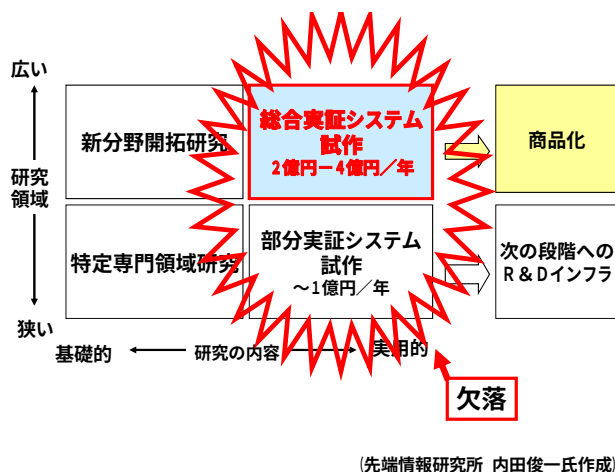


図4 実証システムの開発費用

人材の育成ももちろん重要であり、大規模ソフトウェアを開発できる人材が必要不可欠である。このような人材を確保するためにはソフトウェア技術者の社会的地位の向上が不可欠である。

6. ソフトウェアは成長する

ソフトウェアの開発戦略を考える上で、重要なことはソフトウェアは成長するということである。これは重要なことであり、ハードウェアの開発と根本的に違うところである。

実際、大規模ソフトウェアが成熟するまでには10年が必要である。例えば

- ・NASTRAN 1960年代に開発
- ・GAUSSIN 1960年代に開発
- ・STAR-CD 1980年代に開発

このことは、開発するだけではだめで継続的に保守改良していく必要があることを意味している。

すなわち、ソフトウェア成長をサポートすることが重要である。このためには、国の資金を永久に投入することは不可能であるから、ベンチャー企業等が事業化し自己資金で継続的に開発、改良、サポートを続けていける仕組みを作ることが必要である。

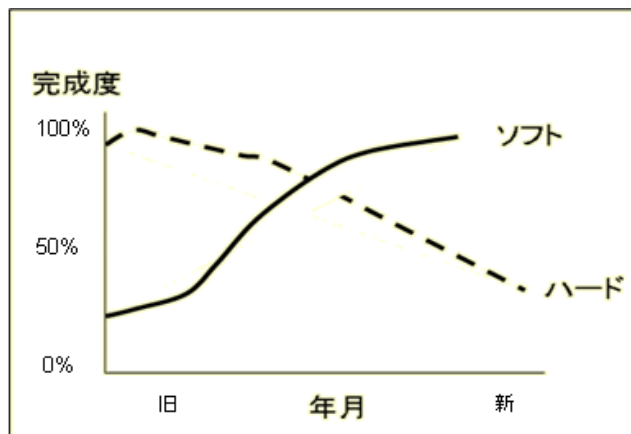


図5 ソフトウェア・ハードウェアの成長/劣化

7. 知的所有権に関する問題

ソフトウェアの開発に関して、国家プロジェクトで開発したソフトウェアの知的所有権をどのように処理すべきかは、ソフトウェアの普及と成長にとって重要な問題である。

前述したように、実用ソフトウェアは、使われながら改良・修正が行われて5-10年かけて成熟していくものである。したがって、国家プロジェクトで作成したソフトウェアのままでは実用的なソフトウェアにならず普及もしない。このため、ソフトウェア開発のモデルとしては、図6に示すように国家プロジェクトで開発したソフトウェアを民間企業が商用化し、市場から資金をあつめ、改良・修正を継続的に行っていく体制を構築する必要がある。

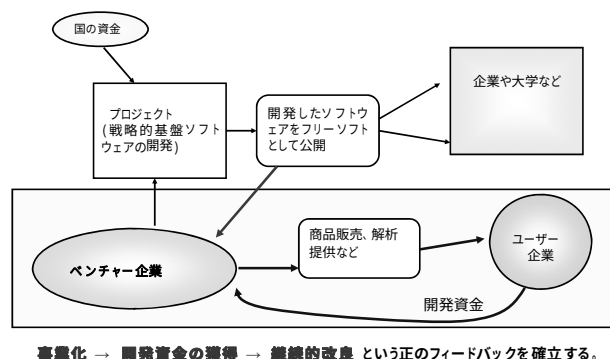


図6 ソフトウェアの普及体制

すなわち、国家プロジェクトで開発したソフトウェアは、適切な資格のある会社なら自由に商用化できる必要がある。ここでソフトウェアの

知的所有権が問題となる。

かつては、国家プロジェクトで開発したソフトウェアの知的所有権は、国に属していたために、事実上商用化できず、作りっぱなしで、結局は実用にならなかった。このため、最近では日本版バイドール法に基づき開発者に知的所有権が属する例が多くなっている。この考えは開発者が事業化するだろうというものであるが、ソフトウェアに関しては、この考えは通用しない。その理由は、まず、第1に大規模な実用ソフトウェアの開発者は一人ではないし、複数の国の制度の資金が投入されている場合が大多数である。そうすると、すべての権利者の合意を得なければソフトウェアの商用化はできないことになり、事実上不可能である。

従って、実用ソフトウェアを普及するための現実的方法は、全開発者、関係者がソフトウェアの財産著作権を放棄し、適切な資格のある会社自由に商用化する権利をあたえることである。NASTRAN等の多くの欧米の実用ソフトウェアは、このような方法で商用化され、世界標準になってきた。