

第120号

平成11年9月

E-mail: © 1999

shimz@mb.infoweb.ne.jp

LDG04167@nifty.ne.jp

## SCだより

編集発行人

清水吉男

(株)システムクリエイツ

横浜市緑区中山町 869-9

電話 045-933-0379

FAX 045-931-9202

## ソフトウェア開発の原則

「ソフトウェア開発 201の鉄則」から 38 かい

効果的なソフトウェア構成管理 (SCM) は、単に誰が、コードまたは文書類の上で、何を、何時変更したかを記録するツールを備えるだけではない。それはまた、ソフトウェアの変更に関わるすべての関係者が変更を確実に実施するために、名称づけルール、方針、及び手続きを注意深く作成することでもある。これは、各プロジェクトに合わせて調整しなければならない。これが実施されていることは、次のことを意味する。

- ・ソフトウェアの問題をどのように報告するかを知っている。
- ・新たな要求項目をどのように要請するかを知っている。
- ・すべての関係者は、提案された変更が知らされ、これらに対する見解が求められる。
- ・構成管理委員会は、変更要求を優先順位づけし日程を決める。
- ・すべての基準品目化された中間、または最終製品が制御下にある (すなわち、該当する手続きに従わないでこれらを変更することは許されない)。

これらのすべてを記録する最も適切な場所は、通常ソフトウェア構成管理計画 (SCMP) と呼ばれる文書である。この文書は、一般にソフトウェア要求仕様書が承認されるのと同様時期という、プロジェクトのかなり早い時点で書かれるべきである。

(201の鉄則: 原理174 <製品保証の原理 = ソフトウェア構成管理手順を早期に確立せよ >)

## 解説

最近、「構成管理」が一つの流行のようになっています。「構成管理」に関する本も最近になって出版されています。

構成管理は、もともと軍需製品とくに航空機の部品の構成を管理する所から始まったものです。民生品と比べて製品寿命が長く、途中で部品の交換や設計の変更なども行われるため、ある部品を使っている製品を追跡する必要があったのです。それが民生品にも用いられ、さらに保守開発が頻繁に行われるソフトウェアの開発にも用いられるようになったのです。

ソフトウェアの構成管理の役割は、簡単に言えば、

- 1) ソフトウェアを構成する「部品」を管理し、
- 2) 要求の追跡を行い
- 3) そのために変更を制御することにあります。

ソフトウェアの機能が複雑化し、開発のサイクルが短くなっている今日では、適切なツールとの協同作業でなければ実現が困難な方向にあります。保守開発 (派生モデル開発) で、ベースを共有する複数のプロジェクトが同時に走る状況では、一ヶ所で問題が発見された場合、その「部品」を使っているモデル (プロジェクト) を追跡し、訂正しなければならない。関連プロジェクトが現在進行形の時は、もはや適切なツール無しでは実現しないといってもいいでしょう。

## 要求の追跡を支援する

構成管理ツールと言えば、部品 (モジュール) の「バージョン管理」をするもの、という認識では不足です。要求、特に変更要求がどのように実現されたかを追跡する方法を提供することが求められています。開発の途中で発生する変更要求も、バグを訂正するための変更要求も、それがどのように実現されようとしているのか、そしてどのように訂正されたのかを知る手掛かりが必要です。

ちょうど「流通システム」のように、「要求」という荷物が、今の工程にいて、どのように対応されているのか、そしてどの評価セッ

ト (梱包) で実現したのかを追跡できる機能が求められているのです。

直っているはずの変更要求に関連するバグが新たに発見されたとき、その変更が何時、誰によって行われたのか、そして誰が承認したかが分かることも必要です。もちろん、これはその人を単に責めるための情報ではありません。その人の周りに適切な「プロセス」を構築したり、再教育を実施したりして、再び問題を起こさないようにするための貴重な情報なのです。

## 変更を制御する

構成管理を実現するには、変更作業が「制御」されなければなりません。変更制御については前月号で詳しく触れましたので、ここでは補足する程度に留めますが、当事者が、資料も残さずに勝手に変更 (修正?) してしまうようでは構成管理は実現しませんし、今日では、そのことによってプロジェクトを危機に陥れかねないのです。その意味でも、変更作業は「管理」ではなく「制御」されなければならないのです。

そこで必要なのは、『みんながうまく行くように進んで制御される』という姿勢です。ソフトウェアの開発作業そのものが「協同作業」である以上、必要な制御は受け入れられなければならない。進んで制御を受けるという姿勢が

なければ、構成管理を自分たちの手で改良していくことも出来ないのです。

この国は「和」の国と言われてきました。でもそれは波風を立てないと言う意味での「和」であって、意見 (異見) をぶつけながらも協調して取り組むという「和」ではありません。はっきりしていることは、「孤人主義」では構成管理は実現しないということです。

## 何を「調整」するか

ここで重要になってくるのは、構成管理委員会の役割です。構成管理がうまく行かないもう一つの理由は、プロジェクトの実態と、そこで行われようとしている構成管理の手続きが合っていないことです。

いつも同じようなソフトを作っているようでも、1回の開発サイクルで十分な場合と、変更規模が大きくて「インクリメンタル」を採用するときとは、プロセスは異なってきます。当然、構成管理の手順も異なってきます。派生モデルの開発と、新規モデルの開発とでも、構成管理の手順や成果物が異なってきます。さらにソフトウェアの分野によっても、構成管理の手順やベースラインの設定の細かさなどで違ってきます。

開発期間が厳しいときは、構成管理の手順もぎりぎりの所まで削る必要があるかもしれません。もちろん、人選もそれに合っていないかもしれません。このように適当な調整機能を持たない構成管理は、最初から継続されないことを保証しているようなものです。

## 品質保証への橋渡し

こうした構成管理がうまく進むためには、必要なドキュメントが必要な時に書かれていることも重要になってきます。そのような「文書」が何もなく、1ヶ月経って突然ソースが出てくるという状況では、構成管理は機能しません。確かに、ツールがあれば、でき上がってきたソースのバージョン管理は可能かもしれませんが、それは単に「結果の管理」に過ぎません。要求の実現方法が間違っていれば、1ヶ月のリワークとなる可能性があり、その結果、構成管理が続けられなくなるのです。

構成管理は、品質保証に繋がってこそ価値があるのであり、継続させることが出来るのです。

(次号に続く)

## 円高を国益にできない日本

ここに来ての急激な円高に日銀・大蔵の対応が割れている。アメリカの圧力を受けている大蔵省に対して、それだけに独自性を失うまいとする日銀の姿勢がぶつかっているような印象を受ける。いや、バブルを恐れているだけのようにも見える。

今年に入って日本の幾つかの経済指標が上向きを示していると発表した途端に、海外からの円買いが入り、株式相場が上がり始めた。株は先行指標なので、為替もそれに連動して動いた訳だが、構造改革が進んでいない状況では、折角の円高も国益にできない。

その国の経済が発展すると資金需要が発生し、為替のレートは上がっていくのは当然の話です。ただ、問題は、そのときの円高に耐えられるだけの生産性の裏付けがあるかどうか。すでにアメリカの製造業との間では2倍の差がついているのではないかと。

かつて日本の経営者はアメリカの経営者に対して、株主の力におされて「目先の利益ばかり追いかけている」と言って非難したことがある。今、その言葉は、そのまま日本の経営者の前に立っている。「あなた達こそ、本当に日本経済の5年後10年後を考えていますか」と。



# 暁鐘の音 103

## 10年という時間

「SCだより」が、今月で丸一〇年が過ぎたこととなります。これも読者の皆さんがいたから実現したことで、皆さんにお礼を申し上げます。私事ですが、一〇年前といえ、下の娘が小学校の一年生でした。まさにバブルの絶頂期で、その数年後にバブルはじけ今日の低迷が始まったわけですね。

十二年前、私はプログラムを書くのを止めて、それよりもプログラムをうまく書く方法を教える側に回ると、方向転換を図ったわけですが、それが実際問題としてどのくらいの仕事になるかは見えていませんでした。ただ、時代の流れから、二

当時、どこまで書き続けられるかわかりませんでした。とにかく書くことがある間は続けてみようと思つたのです。中学生の時に、校内新聞の発行に携わっていたのですが、まさか三〇年後に役に立つとは思いませんでした。人生って不思議な感じがします。どれも途切れていないのです。あの時の経験が今生きていけると感じることもあ

ります。当時は苦しい経験でも、それがあつたから今の自分があると感じられるのです。五年ほど前に、「その人の人生にとって必要としないことには遭遇しない」という言葉に釘付けになった自分を、今でもくつきりと思ひ出せます。

### 変わったこと

一〇年前、IEEE Softwareで「プロセス成熟度」という言葉を見つけ、私自身が進んでいく方向と同じであることに心強く思つたものでした。アメリカでも、「ここに手を付けよう」として、それが「CM M」となつて出てきたときには、少しばかりショックを受けました。熟練のステップが、自分が経験してきたものと似ていたにも関わらず、その時の私のカリキュラムは、まさに世間一般のものと同じでした。自分の経験を活かすつとして始めたはずなのに、当時の常識に囚われていたのです。急いでCM Mを研究し、その中で不足している部分を、自分の経験を元に、顧客の協力を得ながらいろいろ実験し、私なりの方法を研究してきました。その後、

同じ個所を、ハンフリー氏の方は「PSP」として補ってききましたが、IEEE Softwareを見てみると、この一〇年、アメリカではCM Mに限らず、ソフトウェアの生産性を上げ、品質を改善すべく取り組みが、研究者とそれらの現場を持つ実践者が連携して行われているのがよく分かります。

インターネットの発達も、目を見張るものがあります。一〇年前には、軍用のもは見えていませんでしたが、ここまで発達するとは想像もしていませんでした。電子メールは、私の仕事の形を大きく変えることになりました。

インターネットは、「共有」の概念を大きく変化させました。世界中で、会つたこともない人達の間で、情報やノウハウを共有します。ニュースも紙面や時間の制限を受けることなく流すことができます。一方では、無報酬による情報の提供という新しい姿も定着しつつあります。それは、人や社会の役に立つ、という人の生きる本質を思い起こしてくれます。

経営面でも「CIO」という執行役員の位置付けがこの数年で重要性を増しています。そこから、「ナレッジ・マネージメント」へと広げること、社内にあるキラリと光るノウハウを全社的に共有して生産性を上げたり、そのノウハウ自体を収益源に変えてまいります。

世界の医療も、「治療」から「予防」あるいは「再生」へと大きく変化し始めています。日本では相変わらず「治療」が中心なのが気掛かりです。

そして二一世紀を前にして、「ユーロ」が実現し、あらたな経済の枠組みが出来ました。私たちが学校で習ってこなかったことが、これから起きようとしています。

### 変わっていないこと

翻つて、この国の現状を見るに、この一〇年間、一体何が変わったでしょうか。ソフトウェアの世界では、確かにオブジェクト指向が広がっています。根本が忘れられているため、状況は心細いばかりです。この後、派生モデルの展開の段階で、今まで以上に混乱する可能性が高いと見えています。ソフトウェアの世界に建築基準法のようなものがあれば、殆どが違法建築で、ちよつとした地震で簡単に倒壊してしまはずです。

オブジェクト指向や、新しいツールが出てくるのは、時代が変わっているからです。それまでの方法やツールでは、次の時代の要求に匹敵しなくなる可能性があるからです。もちろん、すべての分野で一気になるわけではありませんが、何らかの形で波及していきます。

時代の何が変わっているのか、あるいは変わるうとしていのかを知らないと、新しい手法やツールを活かしきれず、結局は変化に失敗することになり、「変化の先取り」も出来ません。

企画からの対応の仕方、仕様変更の際の手續き、バグ検出後の作業の流れなどが、三年前と何も変わっていないとしたら、それはまさに「赤信号」です。そこまで「固定」しているこ

今から動き出しても、意識が変わって結果に現れるまで、早くても二、三年はかかってしまいます。

企業内の組織の形や制度も変化が遅すぎます。経営者の意志と責任で変化させているのではなく、状況の変化に迫られて踏み出している状態ですから、中途半端で弊害の方が表面に出てくるのです。

先日のテレビ番組で、ルービン前財務長官は、日本の問題は、この一〇年何も変わっていないことだと言っていました。ベンチャー企業を育てると言いますが、そのための有効な仕組みは未だに整備されていません。ちなみに、今、韓国では設立後7年は法人税が「0」ということです。

先頃、国立大学を「独立行政法人化」とする方針がようやく出されましたが、まだ成立したわけではありません。これによって大学間で「差」が付くことを理由に、現場が拒んでいるのです。この国の「結果平等」を誘導してきた人たちが、変化を拒んでいるのです。「教育」こそ、時代に先駆けて変化しなければならぬのに、自ら吐いた「普遍性」という言葉に縛られているのです。

今こそ変化しないと、この国は「奇跡を起こした国」として歴史の中に葬られてしまつ。そんな危機感を改めて感じてまいります。

「今月の一言」  
はお休みします