

第65号

平成7年2月

©1995

E-mail :

LDG04167@niftyserve.or.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



CMM v1.1



6 4 レベル5

<レベル5においてのキープロセス領域は、絶え間なく、そして測定可能なプロセス改善を実現するために、組織及びプロジェクトが取り組まなければならない問題に焦点を絞る。>
いよいよレベル5へのステップアップまでやってきました。ただし、前回にも述べましたが、実際にレベル4から5にステップアップしようという開発組織は、ほとんど存在しないと思います。アメリカにおける最近のS E Iの調査でも、数百社評価したうちで、レベル5は1社しか認定されていません。ちなみに、そこはスペースシャトル関連のソフトを開発している企業です(さすが!)。
しかしながら、理想は理想として、このようなステップが考えられる背景にまで思いを巡らすことで、それぞれのレベルに応じて、有用なヒントが得られるはずですし、また、そのような読み方をしなければ、現実の組織を変えて行くことはできないでしょう。

< "Defect Prevention (欠陥予防)" とは、欠陥の原因を識別して、そしてそれらを分析し、定義されたプロセスを変更することによって欠陥の再発を予防することを意味する >

ここではっきりしていることは、欠陥の原因はプロセスに存在するのであって、製品に存在するのではないということです。製品はプロセスの欠陥を反映したに過ぎないのです。にもかかわらず、多くの開発現場では、製品に入り込んだ欠陥を見つけて、取り除こうとする。もちろん簡単な欠陥はそれで済むでしょうが、プロセスのレベルが低いと、製品に侵入した欠陥の性質は、そんなものでは済みません。結局、内部に侵入した欠陥を除去するために、製品を傷つけることになるのです。欠陥の内部侵入を許したプロセスが、そのままの体制で侵入した欠陥の除去に取り組むのですから、結果は予想がつくでしょう。

欠陥の原因は、製品にあるのではなく、その製造工程にあるのです。もっと厳密に言えば、そこにいるソフトウェア・エンジニアにあるのではなく、その「仕組み」にあるのです。もちろんこれは極論です。或いは、この位のレベルのプロセスにあって初めて言えることかもしれませんが、レベルの低いプロセスでは、個々のエンジニアのスキルの問題は、どうしても否定できません。言い替えば、たとえレベル5の「仕組み」が用意されても、その中に、「レベル1的なエンジニア」が混じってしまえば、やはり、多くの欠陥は入り込んでしまうでしょう。つまり、いい仕組みがあっても、その仕組みに応えられるエンジニアのスキル、あるいは、その仕組みを支えられるエンジニアのスキルが必要なことは否定できません。
たとえば欠陥の原因が「仕様の食い違い」と判断されたとき、それでは「要求段階で仕様の内容をもっと詳しく詰めることにする」という対策を考えたとしても、仕様を詰められなかった原因が、要求仕様書に対するインスペクションの不足なら、これでいいでしょうが、関係者が要求仕様書そのものをまともにも読んでいないとすれば、こんな対策は何の意味もないことにな

ります。これがプロセスレベルの違いでもあるのです。4から5に上がろうというレベルでは、単にインスペクションを実施するタイミングや、仕様書の書き方、或いは、要求を拾い集める行為をどこに入れるか、といった対応で結果を変えることができるでしょう。しかしながら、レベルの1や2のプロセスでは、そういう仕組み以前に、仕様書そのものがまともにならなかつたり、人の書いたものを読むことができなかつたり、第一、そのような「時間」すら、自分でコントロールできないことが多いのです。
「クリーンルーム手法」は、まさにここを突いたものでもあります。開発者は一切テストをしない。そしてテスト中に一定個数以上の欠陥(実際には1 K N C S S (Non Comment Source Statement)あたりに数個の欠陥)が発見された段階で、そのモジュールは廃棄されて、もう一度作り直します。もちろん、そのような欠陥が入った原因を追究し、その対応を講じた上でこのことですが。

< "Technology Change Management (技術変更管理)" とは、例えばツールや手法やプロセスなど、自分たちに有益な新しい技術を探し求めて、それらを規律正しい方法で組織に導入することを意味する。目指すところは絶えず変化の世界における効果的な革新にある。 >

ソフトウェアの世界は変化の世界です。今日では、メインフレームもパソコンもMPUを共有しており、その意味で、MPUの性能の改善が、そのままソフトウェア界全体に大きな影響を与えることとなります。MPUの性能が、1年で数倍、さらに3~5年で、新しいアーキテクチャなどによって飛躍的な性能アップが図られています。メモリも後2、3年で256メガの段階に入ります。ハードディスクもその頃には、1ギガが1万円台になっていると予想されます。このほか、通信手段も高速化し一般化するでしょう。ますます、大規模なシステム、或いは、中規模システムの有機的な結合形態が求められるでしょう。このように、MPUの性能が上がり、メモリの規模が大きくなれば、これまで対象外だった世界も、ソフトウェアの対象となってきます。こうしてソフトウェアの世界は、非常に短い期間で求められるものが変わってきます。性能的な面だけでなく、開発期間やコストに対する要求も変化します。

このような変化が、いくつか組み合わせざったとき、それ以前の開発体制や開発方法では、耐えられなくなります。それを素早くキャッチし、新しい、かつ時代の要請に対して効果的な開発方法や適当な開発技術を導入しなければなりません。現実の開発を進行させながら、このような技術導入を図るには、おそらく専門のスタッフが必要になります。彼等が十分に検討し、導入しやすいように準備したうえで、現場に取り入れていく必要があるでしょう。そして、これがレベル5を維持する上で、欠くことのできない施策でもあります。

< Process Change Management (プロセス変更管理) は、品質を改善し、生産性を向上し、開発期間を短縮する目的で、組織のプロセスを絶えず改善することを目指す。 >

こちらは、前項の「プロセス版」です。時代の要請は新しい開発技術を生みだします。時に、新しい開発技術は新しいプロセスの形態を求めることがあります。もちろん、この位のレベルに達したプロセスでは、それ程大きな変化は求められないかも知れませんが、それでも1970年にウォーターフォールのライフサイクルモデルが提案されたとき、いわゆるプロセス(当時はこのような呼び方はなかった)は、大きく揺らぎました。

そして、88年にスパイラル・モデルが提案されたことで、プロトタイプングのはずみがつきました。この時ウォーターフォール型の開発組織の形態は、このモデルにはうまく適合しないことが明らかになっています。ドキュメントの全体構成も、今までと同じでは対応できません。インスペクションの内容や深さ、あるいはチェックポイントも変わってきます。つまり、新しい開発モデルを導入することで、それに適した開発技術の導入が必要になり、一方、開発組織もそれに合わせて変更しなければなりません。

近い将来、遠隔地間での開発形態に対する新しい提案がなされることでしょう。ネットワークが普及し、実際にソフトウェアの開発の現場に普及している現状を考えると、これを想定したソフトウェア開発の在り方が当然提案されるはずですが、

要するに、この世界は、一時もじっとしてはいないのです。この現実を受け入れるか、拒否するかは、各人の判断です。しかしながら、エンジニアがどちらを選ぼうと、それには関係なく「一時もじっとしてない」のです。

(次号に続く)

下がり初めた株価 -株価PKOの行き詰まり

昨年末から株価が下がり初めている。さらに年を明けてずるずると下がりだして、最近では1万8千円を挟む形で推移している。その上、阪神大震災によって景気の回復の遅れが懸念され、復興景気も吹き飛んでしまい、却って売り圧力を強めている。

皮肉なことに、去年の秋に経済企画庁が「景気回復宣言」を出した頃から、外人株の売りが目立ってきて、最近では大幅な売り越し状態である。あまりの売り圧力に、さしもの公的資金も枯渇したのか支えきれない。ちなみにニューヨークでは史上最高値を付けているというのに。

一説によると、既にPKOで8兆円以上の年金や簡易保険などの公的資金が注ぎ込まれたという。これも株価を維持し、銀行の不良債券処理をやりやすくするためと言われているが、実際には不良債券の始末はそれ程進んでいない。第一、銀行がリストラをやったとは聞かない。

PKOが行き詰まることで市場本来の機能を回復する可能性もあるが、その前に、PKOに注ぎ込んだ公的資金が、1兆円単位の巨額の評価損を抱えてしまう(すでに大幅な評価損を抱えているはず)。この額は資本主義の授業料にしては高く付き過ぎる。大震災の復興資金の調達にも支障を来しそうである。PKOという妙な小細工が、ここに来て破綻した。

