

第 5 2 号

平成 6 年 1 月

© 1994

(株) システムクリエイツ

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

プロセスレベルの改善 25

定義されたプロセス
(9)

ソフトウェア構成管理

「ソフトウェアシステムのエントロピーは、それを管理する手段を講じない限り、たえず増大する」(Lehman & Belady)。これは「ソフトウェア進化の第二の法則」というのだが、うまく言ったものである。

つまり、システムをあるべき状態に維持しようという努力がなされない限り、システムの秩序が低下し構造は次第に悪化する。しかもそれらはすべてシステムが最初に完成した後の変更作業によってもたらされる。それが最初に作られた時の論理を十分に理解されないまま変更され、機能が拡張されるのである。

ソフトウェアシステムは、たとえ最初に入念に検討され、望ましい設計書が書かれたとしても、明確な作業ルールがなければ、保守の段階でそれらのドキュメントが正しく参照されることはない。しかも保守段階で与えられる時間は短く、関係者はさっさとソースプログラムを直して片付けてしまおうとする。たとえそのドキュメントが自分で書いたものであっても、ソースプログラムだけを直してしまうものである。

したがって、ソフトウェアシステムの寿命を長く保つためには、保守作業を個人の「善意」に委ねるのではなく、作業のルールとして、関係する設計書類を併せて変更することが義務付けられなければならない。そうでなければ如何なるソフトウェアシステムも足の踏み場もない廃虚となってしまふ。

ベースライン設定



その様なルールを確立できない一つの理由は、あるプログラム・モジュールに関わるドキュメントが一体どれなのか、誰にも分からないことです。もっとひどい状況では、一体このソフトウェアシステムに関してどのようなドキュメントが書かれたのかすら分からない。管理の仕組みも制御の手だてもない状況では、いつでもこの状態に陥るのです。当然の帰結として、その後の保守作業がコントロールされることはありえません。

このような状況に陥らないために、「ベースライン」を設定する方法があります。勿論、これで全てが解決する訳ではありませんが、手だての「材料」は手に入ります。

ベースライン設定は、ある時点で当該作業フェーズで作られた成果物を『公式』の成果物にすることです。当然その作業フェーズで各人がどのような成果物を生成するかは、予め明らかにされていることが前提です。それすら明らかにされないまま、成果物を「公式化」しても何の意味もありません。したがって、少なくともこの時点で、どのようなドキュメントが存在するのか良く分からない、という先の問題は解消します。以下にハンフリーの提案するベースラインの中で、現実の開発組織で使えるようなものを紹介します。

(1) 要求ベースライン

このフェーズの役割は、顧客から出された要求を満たすシステムを設計する前に、要求内容を設計者が十分に理解していることを確認することにあります。「要求仕様書」はそれを確認する手段で、インスペクションによって要求を双方が確認したことでベースラインが設定され、仕様書が確定します。顧客の機能に対する要求には制限がなく、どこかで区切らなければプロジェクトが前に進みません。勿論、ここでベースラインを設定したからといって、それ以降の顧客の要望をすべて排除することはできないでしょう。それらは要求仕様変更のための「正式な手続き」に基づいて処理され、その都度ベースラインが確定することになります。このベースラインは最上流のダムに相当し、ここがコントロールされなければ、下流のダムもうまく機能しくなくなります。

(2) 仕様ベースライン

ここで言う「仕様」とは、いわゆる外部仕様のことです。「要求」を受けて個々の要求の中に含まれる機能を明らかにしたり、実現するための必要な環境なども明確にします。また「分析」作業が行われているときは、分析仕様書がここでの対象となります。このベースライン設定で重要なことは、要求とのつながりを明らかにすることです。つまり、この部分の仕様は、どの要求を満たすためのものか参照できるようにすることです。そしてこの逆方向の参照を受けて、要求仕様からの順方向の参照表が作られます。

(3) 設計ベースライン

このフェーズでの作業は構造図からモジュールの詳細設計まで含まれます。設計が最初に完成し、インスペクションが終わったことでベースライン設定されます。ここでも、設計と仕様との間で相互参照ができるように考慮される必要があります。こうすることで、あるプログラム・モジュールとそれに繋がるドキュメントが明らかになるし、逆にある機能(要求)を変更することで、設計仕様書のどの部分と、どのプログラム・モジュールが影響を受けるかもわかるようになります。

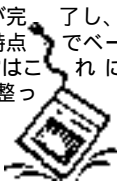
(4) 単体ベースライン

各モジュールの単体テストが終了し、適当なレベルでの結合テストが終了し、そのテスト結果を評価するためのインスペクションを経て単体ベースラインが設定されます。

ベースラインが設定されることで、その成果物はすべて『公式』のものとなり、許可なく変更することは許されません。したがって、単体ベースラインが設定されるたプログラムは、それ以降は、たとえ制作者と謂えども許可なくプログラムを変更することが許されなくなります。

(5) 統合ベースライン

これは統合環境のなかでテストが完了し、インスペクションで承認された時点でベースライン設定がなされます。通常はこれによって外部に出荷できる準備が整ったことを意味します。



ところで、このようなベースライン設定を実施することの難しさは、現実の場面がここでいう姿とあまりにもかけ離れていることです。現実のソフトウェア開発現場での混乱の多くは、たとえば十分な単体テストが行われないまま統合環境に提供されることから起きており、その状況からは、とてもベースライン設定などできそうもないように思われます。

また、現実問題として、各担当者の作業スピードの違いから、一斉にベースラインを引くことが躊躇されることも考えられます。1週間後にベースライン設定されても問題ない人もいれば、予定の半分しか成果物が用意されていない人もいられるかもしれません。もし遅い人に合わせると、進んだ人にとって作業のムダが生じます。

これらの議論は、実際に多くの開発現場で目にする光景で、至極もっともなように聞こえる議論ですが、これらの議論に共通していることは、「現状」をそのままにして、そこにベースラインを設定しようと考えていることです。ベースライン設定という考え方を取り入れるために、現状の作業のどこをどのように変えるべきか考え、併せてそこから手をつけるべきなのです。

(次号に続く)

危険される『理系』離れ



▲先日、共通一次試験が実施された。今年は景気の影響もあってか、親の負担や入学後のアルバイト先の不足が懸念されるようで、国公立を受験する人が多くなるという。私立大学も手をこまねていては、優秀な学生を集めることが出来ない。

▲それにしても気になるのは、理系科目の受験者が非常に少ないことである。試験会場によっては受験者は数えるほどしかない。特に女子の受験者は少ないようである。数年前から、理系の学部でも入試科目から理系科目を除外し始めていることが大きく影響しているようだ。その様な状況で理系の学部に入學できても、彼等に対してどのような授業が出来るというのか。

▲ある調査によると、小学校低学年までは理科(今は生活科)は好きな科目に挙げられているようだが、中学生になるとはっきりと「嫌い」という生徒の方が多くなるという。さらに高校では理科の実験の評判が悪く、予定の3割程度しか消化していないという。その上、受験科目でないために見向きもされない始末である。その様な人がしばらくして小学校や中学校の理科の先生として回ってくる。これでは理科を楽しく教えることなど出来る道理がない。

▲一体、この国は何で立って行くつもりなのだろう。資源に乏しい国が、科学を捨てて何があるというのか。確かにそのような学生では、たとえ理系を卒業したといっても金融関係の企業に就職するしかないかも知れないが、それにしても、どうして製造業をはじめ、理系の人材を必要とする企業は、大学に対して何も要求しないのだろうか。今から着手しても10年かかる。あゝあ

かね

暁鐘の音

35

長所と短所

終身雇用や年功賃金は高度成長期以降の日本経済を支えてき

た。間違いなくこれらは日本経済の強さの背景であり長所であった。だが、環境によって意味が変わる。社会の仕組みも例外ではない。

普段「おっとりしている」という短所も、場面が変われば「慎重な行動」となる。「活発な」という長所も、重大な決断が迫られた時に判断を誤るかもしれない。「根回し」の上手な人も、組織がフラットになって情報技術が発達すると活躍の場を失うかもしれない。

エンジニアで文章を書くのが苦手と言う人も、もしかすると頭のなかで概念を組み立てるのは得意かもしれない。その様な人にはむしろアルゴリズムを考えるような作業が向いているだろうが、問題はいつまでもその様な「場」を与えられるかどうかである。

勿論、自分の長所が活きる環境を自分の努力で維持すればよいのだが、必ずしも自分の思いだけでそのような環境を維持できるとは限らない。一般には年齢とともに自分に求められる役割が変化することがあるし、

組織や機構の変更によって求められる役割が突然に変化することがある。その時、自分の長所が短所になって自分の前に立ちはたかることになる。しかもその「短所」は、これまで「長所」として自分を「そこまで」運んできた源泉だけに厄介なことになる。「長所」も「短所」も、その環境や状況によって「長所」であり「短所」であるという拘束から逃れることはできない。

気付かない背景

しかもその「長所」のおかげで好調なときには、背後に迫り来る問題に気付くことは難しい。長所が長所としてあり続けるための環境が変化しつつあっても、「長所」が存在し続けている間はそのことに気付かない。尾根を越えるまで尾根を越えたことに気付かない。尾根に近づいていることなどは気付かないし、気付こうとしない。

終身雇用も年功賃金も退職金制度も、高度経済成長という旗印の下に、労使が協調して作り上げた制度である。もちろんこれに近い考え方は江戸時代中期から芽生えていたことは確かだが、今日のように神話化されてからは五〇年も経っていない。その時にはこれらの制度が必要であつたから制度として成立したのである。幸運にもこの時は痛みを伴わない形で制度として成立したが、今日、この制度を一部手放す際には、既得権も一緒に手放すことになるために大きな抵抗となっている。

長所が短所になる

年間一四〇億ドルもの貿易黒字を出しながら、国民として何も豊かにならない原因が制度的な問題に起因すること、一部の人は気付いているし、多くの人もいずれ気付かされることになる。組織や機構に柔軟性を欠くことがどういう結果をもたらすか、もうすぐ明らかになる。

非競争経済から競争経済に変わっていく中で、まず障害になるのは規模の問題であろう。IBMが四五万人から二万人に減らしても、なお、その重さに耐え兼ねて分裂しようしている。一部の日本の産業も、自らの規模に苦しめられるときが来るかもしれない。終身雇用や今日の退職金制度は、この「規模」を支えるために考え出された制度である。まさに、「長所」のために身動きがとれなくなっている。「短所」ならこたわりなく捨てるのであるが、「今」を作り上げてきた長所であるだけに簡単に捨てられない。だが、いずれ嫌でもかつての長所であつたものを一部捨てて、新しいものに置き換えて行くことになる。もちろん全部が置き換わるとは思えない。制度が変わるとしても、まったく異質なものは簡単には根付かないが、元来、日本には伝統的に「職人」と

いう職制があり、必ずしも終身雇用にとどまらないところがある。この「職人」という概念を拡張した職制なら受け入れられる可能性がある。しかも、現実問題として本当に終身雇用されている労働者は、実際には一%未満であり、「制度」としての実態はなく、単なる了解事項に過ぎない。そのことに気付けば意外と転換は早いかもしれない。

「今は相撲ではなくて相撲取りを見に来るお客さんが多い。こんな人気は長続きするわけない。こういう時はしっかり考え、相撲そのものを充実させて行かなければ」

二八代立行司・木村庄之助

二八代木村庄之助は先の九州場所を最後に定年を迎え引退した。九歳の時に子供行司でデビューして以来、実に五六年の行司生活である。しかも立行司は八四年（当初は式守伊之助として）から勤めているので一〇年間勤めたことになる。立行司は結びの一番しか仕切らないが、その緊張は前日の晩から始まるという。わず



か一分足らずの勝負に全神経を集中させるため、前日から両者の動きを想定し、いかなる動きにも対応できる準備をして土俵に上がる。単に年功序列で立行司に成れるわけではない。ちなみに初場所は立行司はいない。

その木村氏は「相撲取りファン」と

といつてもそんなことが二年や三年でやられては叶わないが、五年、一〇年という単位では明らかに変わって行くことになるだろう。エンジニアといえどもその方向を見定めて今から備えることが必要になる。「長所」は、そのままではいつまでも「長所」であり続けることはできない。

「リーグも、サッカー人気に煽られてサッカー選手を見に行く客が多いように思われる。関係者はこの人気の間にチームプレーとしてのサッカーのレベルを上げて、本当の意味でサッカーファンを育てる努力をしなければ、気付いたときに梯子を外される危険がある。」