

第 47 号

平成 5 年 8 月

© 1993

(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 クリエイト

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

20



定義されたプロセス

(4)

計 測 - ソフトウェア・メトリクス

今回はソフトウェア・メトリクスをとりあげます。ハンフリーによると、計測はレベル4のプロセスで実施するようになっていきます。つまり、ソフトウェアに関する各種の計測を実施するには、「管理されたプロセスレベル」以上でなければ難しいということです。確かに各種の生産性データを集めようとするれば、収集するデータの意味を正確に定義しなければならないし、収集したデータを開発作業に反映する仕組みを持っていなければならない。例えば、「プログラムの規模」に関するデータだけでも、ソース・プログラムの行数にするのか、オブジェクトのサイズにするのか、さらに、ソース・プログラムの行数とすれば、今度は、コメント行はどうするのか、「INCLUDE」等で組み入れられたソースの行数は数えるのか、といった問題に突き当たります。一方、膨大な開発量をこなすには、少しでも再利用できるものは再利用する、という「再利用」の考え方にしなくてはならない。実際に彼が書くプログラムの行数は減ってしまい、集計の仕方によれば、彼の生産性は低下することになります。

このように、「プログラムの規模」に関するデータだけでも、これだけの難しさがあります。その他「プログラムの複雑度」や、「ファンクション・ポイント」「エラーに関するデータ」「テストの効率」「再利用率」等まで考えると、やはりレベル3以下の開発組織では難しいでしょう。

しかしながら、トム・デマルコも言っているように『計測されないものは制御できない』ことは言うまでもありません。プロセスレベルを改善するためには、改善の進行が見えるものを並行して計測しなければなりません。ということは、レベル3以下のプロセスにおいても、何らかの計測の習慣を持っていることが必要になります。本格的な計測はレベル4以上で実施するとしても、そのレベルに達するために「計測」が必要なのです。そこで、初期のプロセスでもできる計測として、「プログラム規模」と「欠陥」の計測が考えられます。ただし、初期のプロセスであることを考慮に入れてこれらの項目の内容を、少し簡単にする必要があります。

規模の計測

プログラムの行数を数えることはそれ程難しいことではありません。要は定義すればよいのです。今、対象となっている開発組織のプロセスレベルが「初期のレベル」であることを考えると、組織として、再利用を推進していることはないと考えられるので、まずは、単純な行数で定義しておけ

ばよいでしょう。計測のノウハウが手に入って、計測方法も確立したところで、段階的に計測データの定義を広げていきます。計測方法も未熟な段階で、大上段に構えたところで旨いくはずはないので、最初のステップとしては、次の様な定義で足りるでしょう。

1) モジュール単位で計測すること

ソース・プログラム全体の行数を数えても役に立ちません。尚、モジュールの種類(数)は構造図が書かれることで把握することができます。

2) 「INCLUDE」行を1行とカウントする

「INCLUDE」される行が何行あると、コーディングは1行しか書いていないのと、「INCLUDE」されるのは、殆どが「定義行」であることを考えれば、大きな支障はないものと思われます。

3) モジュール内の総行数に対する、コメントの行数。

初期のプロセスレベルにおける「プログラム規模」の計測はこの程度で始めることを勧めます。

規模の予測

この程度の規模の計測なら、大きな負担にはならないはず。そこで、プログラムを書くまえに規模の予測をすることを勧めます。どの程度の規模になるか書いて見なければわからないというのでは、自分の作業すら見積もれないということになります。

このモジュールならどれ位の規模になるか予測しようとするによって、予測に必要な表現や、ポイントをつかむノウハウが身について行くのです。またモジュール単位で予測するためには、どうしても構造図やモジュール仕様書を書かなければならず、結果的に有効なドキュメントが整ってきます。

規模を予測するときに、ついでに開発作業にかかる時間も予測するとよいでしょう。このモジュールなら2日で仕上げられるだろう、という予測をするのです。そのためには自分のコーディング能

力を把握していなければなりません。自分が1時間に何行書けるのかを知らないで、如何なる予測もできません。こうしてほんの僅かな工夫で、自分の技術を向上させることが出来るのです。

欠陥の計測

これはいわゆる「バグ」の計測です。ここでは欠陥のどのような項目を計測するかということ、開発作業のどの時点から計測を開始するかということが問題になります。

プロセスレベルが低い組織においては、早い段階で計測を開始すると、あふれる欠陥のために収集がつかなくなります。

このレベルにおける欠陥の計測は、将来のための準備であることを考慮して、リーダー又はプロジェクトの管理者が「計測の開始」を宣言するかたちで実施すればよいでしょう。但し、宣言が遅れると収集されたデータは余り意味がなくなります。

さて、計測する欠陥の項目としては、まずは欠陥の種類と影響度から始めればよいでしょう。

1) 欠陥の種類

一般には「コーディング・エラー」「設計エラー」「分析エラー」に分類されます。これによって、単にコーディング時の不注意であったのか、設計時の検討が足りなかったのか、といったことを知るようになります。しかしながら、現実には、この他に修正に伴う欠陥があります。そこでもう一つ、「変更管理エラー」の分類を追加します。

この時、それぞれの分類に含まれる具体的なケースを記述しておかなければ、統一した判断が行われなくなります。

2) 影響度

初期のプロセスレベルでは、せいぜい「致命的」「影響大」「軽微」の3段階程度の分類でよいでしょう。この時も各影響度には具体的な状況を記述しておく必要があります。

初期のプロセスレベルでも、このように影響度を判断することで、単に書式が用意されているから書くという姿勢ではなく、視野を広げ、コスト感覚を身につけることができます。こうして僅かの機会を捉えて、ソフトウェア開発に対する『感性』を磨くことが重要です。

(次号に続く)

1 ドル 100 円 時代 ?

▲ 8月17日の為替市場で1ドル100円台に突入した、この後しばらくは100円~105円あたりでもみあう時期が続くものと思われるが、当面、ヨーロッパの通貨の安定が望めないのと、基本的に「円」の独歩高であることから、欧米の中央銀行による協調介入は望めないとの観測から、この基調が暫く続くものと思われる。

▲ 今回の円買いを中心とした為替市場には、1兆ドルという膨大な資金が、行き場を失った形で動いていると言われている。日銀が数億ドル程度の買い支えをしたところで、支え切れるものではない。結果は価値の低下したドルが溜まるだけだ。

▲ 言うまでもなく、「円高」は原因ではなく結果である。多くの人はその原因に薄気付いているが、「建て前」に阻まれて、手をつけるのが遅すぎた。今、下手にやると牛を殺しかねない。どのような施策がとられるかわからないが、自由競争を柱にコストを切り下げる方向に向かうしかない。

▲ そうなれば、これまでの様な、皆が同じように果実を手に入れるという「結果平等」的な社会通念は少しづつ崩れていくことになる。そして日本にも「自然失業率」という言葉が定着し、選別される時代が来る。逆に言えば、各自の適性を考えた進路の選択が必要になる。

変化

暁鐘の音

30

「変化」をスローガンにしたクリントンが勝った。もし五年前にクリントンのような人が現われて、同じ様に「変化」を旗印に大統領選挙を戦っても、同じように勝てたかどうか。

日本でも先の衆議院選挙で、結党してわずかに一年余の日本新党を中心に「変化」の兆しが見えている。これが本格的な変化につながるかどうかは、変化の主体である国民の意識にかかっている。

一般に世間は「変化を誘導する人」に対して残酷である。クリントン大統領にして、細川首相にして、変化が大勢となると国民は考えつく限りの「変化」を一気に要求する。

貿易黒字の問題、雇用確保の問題、縦割り行政や役人の天下りの問題、談合体質の問題、偏差値の問題、大学改革の問題、医療等の福祉の問題など、挙げたらきりが無いほど未解決の問題が転がっている。これから国民はそれぞれの立場でこれらの問題の解決を要求するであろう。だが、これらの問題の多くは単独に存在しているのではない。それぞれの問題の領域から、周

囲の領域に向かつて枝を伸ばし、根を広げている。しかもそれらは、巧みに補完しあい依存しあっている。言い替えれば、先に挙げた問題は、そのように所にかれた種が根を広げたものである。したがって、一つづつ解決しようとする、大きな混乱が生じることは目に見えている。たとえば貿易黒字を解決しようとするれば、とんでもない雇用問題を引き起こすか、故意にインフレを誘発する方法が考えられるが、現在の円レートではインフレ誘導はできない。大学改革の問題は直接的には偏差値の問題につながっており、そのことが更に、企業の活力の問題につながる。今日問題になっている終身雇用と年功序列の原形が大学にあり、そこを卒業してきた企業人の多くは既にその洗礼を受けているのである。その上、そこは偏差値の牙城であり、それを崩されると立ち行かない人がいる。このようにどこが問題の始まりなのか良く分からないのである。

そして、はっきりしていることは、「現状の変化」は全て雇用に影響することである。現状を「維持」している人達がそこにいる限り、「変化」はその人達に対する変化の要求であり、時には彼等を必要としないことを意味する。全員とは言わないまでも、果たして変化に対応できる人が半数もいないのではない。元々人は、知らず知らずの变化には無抵抗だが、明らかに变化を好まないところがある。そのため、変化の初期には既得権益を守るべく「守旧派」と呼ばれる人達が現われ、変化に伴う混乱を指摘して、変化が不当であることを主張する。その結果、たとえ時代の流れから判断して「変化」すべきことであっても、そのときの状況によつては変化を押し戻そうとする力の方が強いことがある。実際にこうして何度か押し戻されながらも、次第に「変化」すべき力のほうが強くなり、ある限界点を超えたところで、後戻りできない「変化」となる。もしかすると今回の「変化」は後戻りできない「変化」かも知れない。一般には、革命でも起こさない限り「変化」は何度か後戻りする。

ソフトウェア開発組織の改善、すなわち「プロセスレベルの改善」も、管理者も含めて当事者の意識を変化させ、行動を変化させることを狙っている。人の意識が変わらなければ行動は変わらないし、行動が変わらなければ結果は変わらない。残念ながら現実問題として、この種の「変化」の仕掛けは、今のところなかなか実を結ぶところまで至っていない。その原因の一つは、「変化」がその組織において「大勢」になつていないからである。現状は無抵抗に現在の作り方を受け入れてきた人達が「大勢」なのである。彼等にも、確かに将来に対する不安はある。だがそれ以上に今日までやってきたという実績、或いは自負がある。さらにその自負にすぎるところとする意識が、明日の不安から目を背けさせる。月曜日に会社したときに、方向転換の業務命令が出されること局はこの自負にすぎるとは、結局はその組織において彼等が「大勢」であるかぎり、新しい「変化」は表面には現われてこない。

だが実際には、裏側で変化が起こり始めている。それは、組織において我が身の寄り所となるほどの実績を持ち合わせていない人達が、次第に大勢となりつつあることである。先輩たちが次第に実際の作業から遠ざかる

につれて、「先輩の威光」が力をなくしつつある。それ程の実績もない若い人達にとつて、五年後の自分が存在し得るかどうかが、何の保証もないのである。このような人達が「大勢」となつたとき、「変化」は避けられないものとなる。但し、その時で間に合うかどうか問題ではある。しかしながら、そのときに「変化を誘導する人」がいなければ、この「変化」はただの混乱となり、求めるものはここから生まれてこない。

したがって、今最も大切なことは、焦つて今すぐ「変化」することではなく、変化を推進しながら、「変化を誘導できる人」を育てることです。

「人は自分の心を養うためよりも、何千倍も多く、富を得るために心を使っている」

ショーペンハウエル

富はあくまでも結果である。その富を持つにふさわしくな

をかえて金を追いかけている。うちは、金のほうが逃げてしまつて絶対に金はできない。(奥村)ともいう。つまり、金と女性を追えば逃げるのである。

自分とは別だと思つている。あるいは自分

うだけでは富は長続きしないし、地位と実力を錯覚して墓穴を掘かねない。

今月一言

人は身体の空腹に対しては満たすことを忘れないが、心の空腹を感じることは少ないようだ。だが、この状態を長く放置すれば、心に「鬆」が入ってしまう。

野村証券元会長の奥村雄氏は「仕事をすれば金は結果として、後から必ずついてくる」という。それも求められる仕事をすればよいのであつて、そのことに苦心すればよい。だから、金、金、金と目の色