

第96号

平成9年9月

E-mail: © 1997

shimz@mb.inforeweb.ne.jp

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

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 14 かい

カップリング（結合度）とコヒージョン（凝集度）は、1970年代に Larry Constantine と Edward Yourdon によって定義されたものだが、これらは今なお、あるソフトウェア・システムに固有の保守性と適応性を測定する最善の方法である。要するに、カップリングは、2つのソフトウェア・コンポーネントがどのように相互に關係するかを測定するものであり、コヒージョンは、あるソフトウェア・コンポーネントによって実行される機能にどのような關係があるかを測定するものである。我々は、低いカップリングと高いコヒージョンに向けて努力する必要がある。なぜなら、高いカップリングは、あるコンポーネントを変更すると、他のコンポーネントも変更される可能性が高いことを意味し、低いコヒージョンは、エラーの原因や、新たな要求項目に適應させるための修正箇所を特定するのが困難であることを意味するからである。Constantine と Yourdon は、この2つの概念を測定する簡単な方法も提供している。1979年以降に書かれたソフトウェアの設計に関する本のほとんどは、これらの測定について述べている。これらを学び、設計上の決定の手引きとして利用せよ。

(201の鉄則：原理73 <設計の原理 = カップリングとコヒージョンを使え>)

解説

凝集度とか結合度というのは、構造化手法では入り口のところにあって、本来はここを通過して構造化設計や分析に入るようになっていました。もっとも、構造化プログラミングとは、3つの部品で書くことであるとか、構造図を書くことが構造化設計であると考えた人たちは、このステップを飛ばしたかもしれません。そのため、「C言語」で書かれたプログラムでも、関数の呼び出しの際にはパラメータは空っぽであったり、皆で仲良くグローバル・データを更新しあったり、構造図も概念的な階層を表すだけで、そこからは必要な情報は何も得ることができません。その結果、幾つかの関数は緊密な關係に陥り、こちらを修正すれば、あちらも修正しなければならないような状況が、派生モデルのための修正を重ねる毎にひどくなっていきます。Yourdonは、その著書の中で「構造図を書くことだけが構造化設計ではない」と嘆いています。構造図は結合度や凝集度の判断を助ける「ツール」であって、構造化設計の主題は、モジュールをこれらの尺度に従って判定し、保守性や適応性を高めることにあるのです。

設計行為が前提

設計という明確な行為なしでは、結合度や凝集度の尺度を織り込むことは出来ません。多くの人は、まともな設計書も書かずに、昨日のプログラミングの続きをするために、CRTの前に座ってエディターを起動し、作り掛けのソースを開きます。そうして昨日の作業を思いだし、新しいコンポーネントを頭に浮かべて、やおらキーを叩き始めるのです。もちろん、その途中で何度も昨日までに打ち込んだソースを見るために、画面を忙しくスクロールさせます。こうしてインプリメントされたソースに、結合度や凝集度という尺度の入る余地はありません。コンポーネントを分割するときの動機は、単に長すぎて自分でも分かりにくくなったときや、考えるのが複雑になってしまうので、取りあえずそこに新しいコンポーネントの呼び出しの手続きをいれるときぐらいです。“乗っているときは”コンポーネントの大きさや、相互の關係など“關係ない”のです。関数の呼び出しの形も統一されているわけではなく、同じデータでも、あるときは要素で渡したり、あるときは構造体で渡したり、グローバル・データを直接参照しにいたりします。

保守性の確保が主目的

プログラムは、保守という行為がなければ、極端に言えば、少々いい加減なモジュール構造でも構わない。もちろん、テストやデバッグの作業で混乱するかも知れませんが、それでも、設計してからそれほど時間が経っていないためなんとかなるものです。問題は、6ヶ月後に始まる派生モデルの開発です。そこでは、操作性や性能の向上が求められ、新しい機能が追加されます。当然、新規開発のときと違って開発期間が短いため、時間をかけて思い出したり、影響範囲を入念にチェックしたり、試行錯誤を繰り返すことが出来ません。それだからこそ、修正するたびに新しいバグを織り込んでしまうのです。結合度や凝集度という尺度は、保守のために考えだされたものです。もちろん、この他にも「複雑度」という尺度も保守性を高めるのに有効な尺度です。最初からこのような尺度に沿ってコンポーネントが設計されていることで、保守開発をやり易くしたり、新規バグの侵入を防ぐのです。また、これらの尺度の詳細な定義を知っていることで、目の前の修正されようとし

ているコンポーネントに対して、どのような注意が必要かもわかります。こうしたコンスタンチンとヨードンの素晴らしい研究成果があるにも関わらず、そのことを知ることなくプログラムの生産活動に就いていることは、驚きとしか言いようがありません。

オブジェクトと指向でも必要

そして、この問題はオブジェクト指向の時代に入って、さらに悪くなるように思えます。オブジェクト指向の文献には、この結合度とか凝集度という用語は、全くといってよいほど出てきません。オブジェクト指向にはクラス間の結合やクラス自体の凝集という考え方も必要です。もちろん、結合度とか凝集度という言葉は使いませんが、クラス間の接続を複雑にしてしまうと、「is-a」や「has-a」の關係が説明できなくなってしまい、再利用に支障を来します。当然、メソッドに於ては、凝集度や結合度という尺度はそのまま持ち込むことが出来ます。オブジェクト指向の言語であろうと、構造化手法の言語であろうと、複雑な凝集度をもつコンポーネントは、間違いなく保守の敵です。オブジェクト指向だからといって、派生モデルの開発がなくなるわけではないですから。機能的に膨らむ傾向のあるクラスや、ひとつのメソッドでの処理が大きくなってしまいうような場合、クラスを上手く分割することでメソッドを分割することもできます。その際、結合度や凝集度の尺度が判断の参考になります。しかしながら、そのような行為は、結合度や凝集度の考えを明確に持っているから判断されることであって、そのような考えを持ちあわせていなければ、分割のための工夫や知恵が働く余地はないのです。

わが国では、ソフトウェアの開発作業に就いているエンジニアでも、分析・設計手法について基礎的な勉強を履修している人は少ないので、今後、オブジェクト指向が広がるにつれて、この結合度や凝集度という考え方は希薄になっていく危険が高く、ソフトウェアの保守能力は確実に低下することが予想されます。

(次号へ続く)



高齢運転者マーク



高齢運転者による事故の比率が高くなっているのを受けて、さきごろ「高齢運転者マーク」なるものが公表された。運転者が高齢者であることをマークによって他のドライバーに知らせ、注意を呼び掛けると同時に、高齢者の運転する自動車に対して幅寄せなどの危険な行為を禁止しようというのである。今年の10月ごろにも実施されるという。

発想は「初心者マーク」と全く同じであるが、初心者マークの方は、しばらくすれば外れるに對して、高齢者マークは一度使用すれば最後まで外されることはない。それだけに心理的に「自分は高齢者なんだ」という意識を与える危険があるのと、逆に、このマークを貼ることで「守られるべきなんだ」という居直りの意識に陥る危険がある。特に時間が経って、このマークを使用する高齢者の人数が増えてくると居直りそうな気がする。

高齢者にとって、自動車は欠くことの出来ない移動手段である。病院への通院や買い物など、日常生活を営むうえで若い人よりもむしろ必要性が高い。老夫婦がカーナビのハイテク装置を使って、見知らぬ土地に旅をするなんて想像しただけでわくわくする。このような移動手段がなければ、足が弱っただけで家の中に閉じこもってしまうことになるし、移動には常に誰かが付いていなければならない、経済的にも大きな負担になってしまう。

だから高齢者も「高齢者マーク」に頼るのではなく、日頃から運転技術を落さないように、動体視力や反射神経、反応速度などに注意し整えておくぐらいの努力をして欲しいし、その上で、運転を続けるかどうかの判断を自分で責任をもってやって欲しい。そうでなければ「高齢者マーク」は逆に嫌がらせや攻撃の標的になってしまう。

か ね 暁 鐘 の 音

79



アジアの追い上げの意味

二一世紀はアジアの時代だとか、日本もアジアの国々に追い上げられる、という文字を時々目にします。実際に、ソフトウェアの世界では、インドの発展は目覚ましいし、周辺のアジアの国々も、インドに倣ってソフトウェア産業に重心を移しているようです。

ところで、日本の企業の関係者たちは、一体「アジアの台頭」をどのように見ているのでしょうか。アジアの国々も二一世紀に入って経済力をつけてくるだろう、という程度にしか考えていないのではないのでしょうか。そこでは、日本という国は「今のまま」存在していて、その前提でアジアの国々の台頭をイメージしているのではないのでしょうか。

一九五〇年、六〇年代に、日本から民生用電子機器や工作機械、あるいは自動車などが大量にアメリカの市場に輸出されました。その結果、アメリカでは自動車産業は残ったが、他の産業は大きなダメージを受けました。当時のアメリカの生産性は、世界的に見ても決して悪くはなかったのですが、それ以上に日本の生産コスト、特に労働コストが安かった

のです。

アメリカは、その状態から規制の撤廃に踏みきり、競争原理を全面的に取り入れたのです。確かに、レーガン政権の段階では、その痛みだけが目立ったため失業率が上昇し社会的に混乱したことがありましたが、今日になって、当時のレーガン大統領の政策は評価されているようです。アメリカの再生は、そこから始まっているのです。

さて、視線を日本とアジアに向けてみると、そこには四〇年前の姿が見えます。経済大国という呼称に酔って、自らを改革することを忘れていた国と、月収が四〇〇〇円の労働力を大量に（二〇億人以上）抱え、環境が整いだすのを今か今かと待っている国々が見えます。彼らが、今では民生用電子機器を生産し、工場に使うための工作機械を生産し、豊かになつていく人々の衣食を満たすために繊維製品を生産して輸出を開始するのです。GEのウエルチ会長が、ほんの一部を残して、一〇〇年も続いた「白物家電」から撤退した理由は、まさにここにあるのです。

今、製造業であれソフトウェア産業であれ、五年前と変わり映えのしないやり方で仕事をしている人たちは、間もなく開くこの幕の向こう側が見え

ていないのではないのでしょうか。二一世紀の日本は、過去の延長線上には存在しないということが見えていないのではないのでしょうか。シエルの最近の長期予測では「改革しないと生き残れない改革競争の時代」に入ったと警告しています。

一九八九年に発表され、その後のアメリカ再生のシナリオを提供した「Made in America」には、日本に敗れた原因として真っ先に批判しているのは「時代遅れの経営戦略」です。「経営者と労働者が、昔ながらの事業経営の慣習をあまりにも忠実に



に守っているために、新しい経済環境を理解できなくなっている」と指摘しています。そして、あのアメリカであつても、経営者の視野が国内に向いていて、世界を見ていなかったことを指摘しています。その結果、アメリカは日本やヨーロッパ勢に負けたといつのです。まるで、今の日本の状況そのままであり、このあと起きることを予感させます。

「読書と同じように単なる経験もあまり思索の補いにはなりえない。単なる経験と思索の関係は、食べることと消化し同化することの関係に等しい」

シーペンハウエル

今月の一言

さて、二一回連続してシーペンハウエルを引用します。

経験も、そこから思索を引き出さない限り、ただの経験で終わってしまう。なぜこつなつたのか、他に方策はなかったのか、今度起きたときはどのように対処しようか、などと考え抜かなければならない。

人は、何らかの活動をしている限り、いろんな経験をjする。そしてその時々jに思索を取り入れる余地があるのに、多くの人はそのまま放置してしまふ。それは、折角口に入つたものを、消化も吸収もし

ないでそのまま排出するようなものである。私は、プログラムのバグや、スケ

ジュールを狂わせる事態に遭遇したら感謝することを勧めている。多くの人は首をかき上げるが苦笑いしているだけで、「言葉の遊び」とでも思っているようである。思索ができなくなつている人には、それが限界かも知れない。

まりました。一時的に大量の失業者が溢れましたが、彼らは中小企業に吸収されると同時に、新しい産業を作り出した。アメリカには、社会の風土も含めて、そのような方向転換の余地があつたとも言えます。

そしてアメリカの産業界は、史上最高の収益を上げながらも、なおも生産性の手綱を弛めようとしません。私には、来たるべきアジアの台頭の意味を認識しているように思えてなりません。中国を含めたアジアの国々が台頭してくることは時間の問題です。世界戦争でも起きないかぎり、この流れを変えることは出来な

いでしょう。そこまで明白な以上、今のうちに走つてしまおうといつのでしよう。GEの会長であるジャック・ウエルチの言葉に、「八〇年代にはぶらぶら散歩していたようなものだと思えるほど、九〇年代は競争環境が厳しくなる」というのがあります。この言葉は八〇年代のもですが、これを二一世紀になつて日本の経営者は経験することでしょう。

最近の日本の企業は、何となく惰性で動いているように見えます。そこでは以前ほどには「生産性」という言葉は耳にしません。耳に入つてきたとしても、ほとんどは念仏のように口にしていて、生産性にながらう具体的なイメージも為されていないのでしょうか。果して、このような状況で二一世紀に直面するアジアの台頭にどのように対応するといつのでしょうか。アジアの台頭の意味が、本当に分かつていないのでしょうか。それとも、それ以上に「今日のこと」の方が大きな問題なのではないでしょうか。