

## Lyee Internet Information

### 対談

「21世紀の新しいソフトウェア・メソドロジーについて」(後編)

カナダ オタワ大学 グレゴール・V. ブックマン教授

ソフトウェア生産技術研究所株式会社 根来文生社長



司会 : アイサム・A・ハミド教授(岩手県立大学)

#### 【グレゴール・V. ブックマン(Gregor V. Bochmann)教授 略歴】

- 1941年 西ドイツのニーブルに生まれる
- 1971年 カナダのモントリオール、マギール大学より理論物理学博士号取得。
- 1972年 モントリオール大学において準教授に就任。
- 1983年 モントリオール大学において教授に就任。
- 現在 オタワ大学情報技術・工学部教授。

< 主な研究分野 >

コミュニケーション・プロトコル、ソフトウェア・エンジニアリング、  
高速ネットワーク、分散アプリケーション、システム管理、  
UML ( Unified Modeling Language : 統一モデリング言語 )

< 主な著書 >

・ Concepts for Distributed Systems Design  
( 分散型システム設計コンセプト ) 他

## 【 方法論におけるメジャメントとは？ 】

ハミド教授

ここで1つ申し上げたい事は、新しい世紀における新しいソフトウェアというものをどう考えるのかという事です。結局21世紀における理想のソフトウェアというのは柔軟性に富んでいて、インテリジェンスであり、それからいろいろな環境の変化にすばやく対応できるといった事が、言葉で言うならば考えられると思うのですが、ソフトウェアの歴史を見まして、確かにオブジェクト指向の技術というのは取り上げられてはおりますけれども、これだけでこういった問題を解決できる程の十分なパワーは持っていません。こうした背景の中で、Lye が新世紀における新世代のソフトウェア開発方法論としてどう位置付けられるかという事についてお話をうかがいたいと思います。

ブックマン教授

その質問に答えるには、私はあまりに Lye の事がわかっていないので、考えを述べるのはふさわしくないと思います。それよりむしろ皆様の方が、考えを持っているのではないかと思います。ですが、今おっしゃられた事は、みな興味を持っている問題で、どうやってそれを解決していくかという事なのですが、具体的に考えた時に、いろいろな方法があります。しかしある方法を選んだ時に、その方法の生産性・効率性を計測していかなければなりません。もっとも簡単な方法は、1つのプロジェクトに対して、複数の方法論で開発し、その結果を比較することでしょうが、予算面での制限もありまして、単純な比較はしにくいです。りんごとオレンジは違うものなのに、違うままで比較しなければならないという様な状況になるのではないかと思います。

根来社長                   しかしその問題を克服していかなければなりませんね。その為には理論の重要性を認識する必要があります。相対比較をする為に理論があるというのが非常に重要だと思います。理論にそれ以外の役割はないのです。

ブックマン教授           それでは計測はどの様にするのですか？

根来社長                   ですから理論をメジャメントとすべきなのです。  
例えば UML 関連のソフトウェア開発でしたら、UML は自分の理論を明確にして、初めて他との比較の対象となりうるのです。

ブックマン教授           あくまでも共通項の特性を持っているメジャメントがないと比較はできないと思います。計測の為にパラメーターというのが当然ある訳で、ある複雑なシステムがあったとして、それに対する開発の時間ですとか、エラー率がどれくらいあるかとか、プログラムの変更をどのくらい簡単にできるかとか、そういったものがパラメーターであり、それを基に計測していきます。

根来社長                   私の概念ではそういうものは狭義の部分でのメジャメントだと思います。それからもう1つ申し上げたい事は、メジャメントというものは理論毎に固有のものではありますが、開かれた、公共性のあるものでなければなりません。特に共通のメジャーがないソフトウェアの場合には、方法を比較するのに理論が大切になります。  
例えば UML にはクラスという概念があります。これが UML のメジャメントにあたりますが、これは個人に依存した恣意的な実現であり、理由づけなので、本当の意味で共通のメジャメントにはなりません。

ブックマン教授           その時のメジャメントとはどういうものでしょうか？

                                  例えばいくつかクラスがあるかなどという事でしょうか？

根来社長                   UML はクラスについて明確に定義したり、記述しているかどうかという点がメジャメントたりうるかどうかということです。  
例えば Lyee の場合でしたら述語構造がメジャメントです。このメジャメントに共通項を求めようとしたら、クラスを C++ 言語で定義して下さい。それから述語構造で定義して下さい。その時に初めて一定の比較ができます。クラスはソースコードでは書けません。そういう意味で私が先程申し上げました様に、UML のノーションもオブジェクト指向も理論に裏打ちされたメジャメントを持ち得ないのです。先程先生がおっしゃられたプロダクティビティやメンテナンスピリティは正確にはメジャメントたり得ないと思います。

ブックマン教授           では何なのですか？

根来社長                   例えば Lyee は、従来法よりも5倍の生産性を上げられるという実績が

出ておりますが、しかし世界中が Lyee で開発する様になった場合には、もはやそれはメジャメントとはなりません。そういう意味でプロダクティビティやメンテナンスビリティは厳密なメジャメントではないのです。理論とは何かと言いますと、メジャメントを定義しているというのが理論の意味付けなのです。私が申し上げている立場に立つと、ほとんどのソフトウェア方法論というのは、理論ではなく、政策に過ぎないのです。

ブックマン教授

理論の話ななさっていましたけれど、それを聞いている内に、ここ数日でやったセッションの中で聞いた話を思い出しました。

Lyee は7つのルールでシステム開発ができるという事でした。

従来法ではもっとたくさんのルールがあるという形で説明され、Lyee の簡潔性について語っていらっしゃいましたが、それを聞いていて、システムが簡単にできるという印象を受けました。言語の問題もそうですが、それは常に複雑なもの、数が多いものよりも、数が少ないとか単純な構造を持っている方が優れているという風を感じられます。そういう考え方が1つのメジャメントになっていると思います。

例えば、過去に PL/I という言語が出ましたが、これは非常に複雑でしたので、後に消えてしまいました。それから Pascal という言語が出まして、これはあまり多くのものをカバーできないのですが、簡潔なため、今も使われております。C 言語も比較的単純な構造を持っているために使われています。これが1つの基準になると思います。

いろいろな人が関わってきた場合には、どうしても一定の断面を切る様な基準がある事によって、理解していくことができます。根来さんはメジャメントがないとおっしゃりながら、Lyee のプログラムは単純な規則でできるということを1つのメジャメントとして話をされていると私は理解しました。Lyee はシステム開発を単純な規則で行えるという利点があると言えるでしょう。

根来社長

私の考えでは、理論があつてこそ、他の関連事項との比較が許されると思います。これはソフトウェアの世界に限らず、あらゆる学問の世界の理論に共通に当てはまると思います。今度は理論を同一比較する場合について触れたいと思います。Lyee は概念を持っています。オブジェクト指向も UML も概念を持っています。そして概念をソースコードで表現するという事を考えます。そうすればソースコードで同一比較をする事ができます。その時 Lyee のコンセプトは全てソースコードに置き換えられます。

しかしオブジェクト指向の概念はソースコードに置き換える事はでき

ません。そういう問題を私は指摘しているのです。  
こういう観点から、オブジェクト指向はメジャメントを持っていない  
という印象を受けています。  
それからプログラム言語についてですが、世界にあるソフトウェアの  
約7割が COBOL で書かれています。しかし COBOL は約3000  
種類ある訳です。方言性を非常に取り入れていった為に、COBOL は  
3000種類も生まれ、世界の70%のシェアを占めています。  
それから PL/I についてですが、非常に大きな日本を代表するある企業  
が、現在でも頑なに PL/I を使っています。要はプログラム言語という  
ものは評価が分かれる性質を持っています。その会社の人達は PL/I  
ほど優れた言語はないという事で導入したのだと思います。  
それから日本では、Pascal はもうほとんど使われておりません。  
PL/I 同様大変優れた言語だと思いますけれども、私はメインフレー  
マーがこれを推進しなかったというのが普及しなかった理由の1つ  
だと思っています。  
また同じ理由で ALGOL という優れた言語も使われなくなりました。  
ですから、いろいろな理由によって、プログラム言語の置かれている  
立場は大変不安定な立場にあります。技術論では解決できない問題が  
含まれています。ですが、Lycée はプログラム言語に依存しないという  
構造を持っています。Lycée のプログラムを C 言語で作ってみました。  
COBOL でも作りました。アセンブラでも、RPG でも作りました。  
もちろん PL/I でも作りました。いろんな言語で作りましたが、多分  
機械語で作るのが一番よいだろうというのが私達の結論です。  
それからもちろん Java でも作りましたが、プログラム上の問題の原因  
となるような、冗長性がありました。  
そんなわけで、Lycée の世界で言語を評価すると世間一般の評価とは  
違ったデータがでてきます。

ブックマン教授

コードを自動的に生成するので、コードそのものをあまり読む必要が  
ないからですね。

ハミド教授

今回のディスカッションの本論に戻りたいと思いますが、根本的な  
問題は21世紀における新しいソフトウェア開発をどう考えていくか、  
それに関連する問題をどう解決していくかという事なのですが、それ  
は取りも直さず、そのための最良の方法をどうやって達成するかと  
いう問題に置き換わると思います。

ブックマン教授

そうですね、みんなそのように考えていますが、それを実際どうやる

かというのが問題なのです。結局問題なのは、どのシステムも目的を中心にソフトウェア開発が行われているという事です。ある時は質を問われ、ある時は単純にコストだけが問われ、ある時はとにかく締め切りだけを守る事が優先されて、システム開発が行われているというのが現状です。

こういった中で、どれが一番優れているかという事を限定する事は大変難しいです。文脈によりますから。ですから先程もいいましたけれど、可能でしたら同じプロジェクトを違う方法で開発するというのが、最もはっきりと優れたものができる方法だと思います。それも違う会社で違う方法を用いてやってみるという事ができれば、最もいいでしょう。先程根来さんの話にありましたが、あるシステム開発がこれくらい早くできましたと聞いて、なるほどそうなのだろうとは思っても、これが実際に行った開発と比較する事ができれば、最も簡単に有効性がわかると思います。

根来社長

何かその様なテーマを先生の方から頂けませんでしょうか？

あまりにも大きいと問題はありますが、ある程度のものでしたらやってみたいと思います。1000人月くらいのもなら適当です。

ブックマン教授

どこかの会社を引き込まないとそうした規模のプロジェクトは、私自身は今持っておりません。

ハミド教授

根来さんが今言われたのは、時間やコストの面で Lyee が非常に生産性が高いという事を信じて頂きたい、話だけではないというのを認めさせたいというのがあると思います。Lyee は普遍的な構造を持っています。普遍的という事はどんなソフトウェアに対しても適応できて、それが1番だという事を主張していることです。こういった考え方を持っていますので、他のものを組み合わせるといった発想ではなく、むしろ他のものを排斥してしまいます。私たちには、このような普遍性というものを必要と思うのでしょうか？またこういった発想についてはどう思われますか？

根来社長

私は人間の深層心理は公理系でできていると信じておりますので、普遍的なメソッドを人間は将来必ず求めるようになると思います。

ブックマン教授

しかしながら、人は物事のさまざまな面について考えるために、違った方法や表現を必要とします。例えば異なるノテーションが様々なことをするのに必要で、そのために UML は異なるノテーションを提供しています。

根来社長

ただそれは現象の話であって、そういう問題を解決するには普遍性と

いう概念を無視してはいけないと思います。私達はオペレーティング・システム（OS）というものの存在を当然という風に捉えていて、OSの質についてもいろいろと議論されております。もちろんアプリケーション・システムの質についても議論をされておりますが、私達が今考えているのは、OS というものを軽くしてしまえば、ソフトの問題をより深く議論する事ができるということです。そういう意味では私達自身の発想を変えるといった事を持ち込まないといけません。ソフトウェアの世界は誕生してから約50年ですけれども、もうすでに破綻をきたしているという認識を持つ事が必要ではないでしょうか。そういう意味でより強く質が求められる時代に入っていると思います。ソフトウェア・サイエンスの世界も変わらなければならないのではないのでしょうか。

## 【 Lyee の課題 】

ハミド教授

Lyee の国際ネットワーク・プロジェクトは既に1年間の準備期間を置きまして、世界各国の先生方と対面し準備してきました。その大きな目的は、Lyee という方法論がアカデミックな世界にどうやって関わっていくかという事で、Lyee は産業界から生まれたもの、実際のプロジェクト開発から生み出された新しいスタイルです。プロジェクトは、Lyee を学問的な観点からどのように評価していくか、どうやって研究していくかという形で立ち上げました。この件に関してどの様に思われますか？

ブックマン教授

今回Lyeeの説明を聞いて、科学者としてまず理解したいと思いました。その為にはこれまで自分が知っているものと比較することになります。もちろん人によって違った観点で見ていくという事もありますし、Lyee を違った文脈の中に置いた時にどう見えるかとか、違った技術と比べてどうだとか、いろいろあり、それぞれ見える側面が違ってくると思います。それは先程から問題になっておりますが、結局どうやってLyee を評価し計測するか、そのためにはどんな基準やパラメーターを使うかという問題になると思います。

ハミド教授

ソフトウェア・サイエンスが解決するのが困難な問題に対して、Lyee が解決を提案しようという事についてはどう思われますか？

例えばフォーマル・モデル、フォーマル・メソッドについてですが、これについては何とか解決策を模索していますけれども、それを効率的にやる事ができていないと思うのですが。

ブックマン教授

フォーマリゼーションというのは結局ツールが動く為に必要なものですので、フォーマリゼーションというのは意味があると思います。ただソフトウェアのプロセス全体をフォーマライズするという事はまだできていないと思います。しかし言語の問題についてですが、ある意味で言語についてはユニバーサリティが各言語ごとにあると思います。

ですが、ユニバーサリティの問題よりも、Lyee に対して興味があるのは、適応範囲はどのくらいあるのかということです。

先程話した事に1つ付け加えたいのですが、結局ノーテーションしても意味の定義まではできていないという意味で、フォーマリゼーションができていないという事は確かにその通りだと思います。UML にしてもノーテーションの定義はできても、意味については曖昧さを残してしまう訳で、そこで当然違う解釈が生まれて、それが違ったパターンを取ってしまいます。いかなるノーテーションもフォーマルな意味を定義できるものとはなっていません。そういう意味でフォーマリゼーションが必要であるというのは全くその通りだと思います。私はコミュニケーション・プロトコルに関していろいろと開発をしてきましたが、その中でいろいろな言語を扱いました。SDL とか LOTOS などはかなりシンプルな言語で、高いレベルでのフォーマリゼーションに成功していたと思いますけれども、もちろん完璧ではありません。そういう意味でハミド先生がおっしゃられたフォーマリゼーションの問題については同感です。

Lyee に関して1つ質問があるのですが、最初にオタワ大学にいらっしゃって説明をして頂いた時には、全く Lyee について理解する事ができませんでした。この数日でその頃に比べるとはるかにどういふものであるか理解が進みました。それはもちろん具体例を見せて頂いたり、あるペーパーの中にわかる説明があったからですが、それでもなお、まだ曖昧でぼやけたところが非常に多く残っています。それは結局、自分の使っている言語と違っているからでもあるので、なんとか自分の使っている言語で、Lyee では、ある内容をどう表現しているのかという事を言って頂けると理解が進むと思います。



例えばインプット・アウトプット・メッセージはSDLでもUMLでもある訳で、おそらく共通のものを指していると思います。しかし基本構造について説明がありましたけれど、この中で私が言ったダイナミック・ビヘイビアに相当する部分はどうなっているのか、そういうものがわかりますと、Lyeeがもっと明確に見えてくると思います。

根来社長 先生がおっしゃられたダイナミック・ビヘイビアですけれども、実はLyeeにはダイナミック・ビヘイビアを定義するという概念はないのです。それはなぜかと申しますと、Lyeeの場合には述語構造という固有の構造が背景に控えているからです。これが従来の方法と全く異なる特徴です。

ブックマン教授 ではデフォルト・ビヘイビアに相当する部分はあると思うのですがいかがですか？

根来社長 それが処理経路図です。それから述語構造もそういう意味ではデフォルト・ビヘイビアのカテゴリに入ります。

ブックマン教授 こういうものは、私としては、インプット、アウトプットという形で定義されて、その次にどういったダイナミック・ビヘイビアがあるのかという形に展開されるべきだと思うのです。

根来社長 インプット・オペレーション、アウトプット・オペレーションというのは、ダイナミック・ビヘイビアではないという風に捉えます。Lyeeの世界ではそれは非同期構造という区分になります。アクション・ベクターの構造の中に封じ込めてしまうのです。

ブックマン教授 私から見れば非常に特異な形の言語、いろいろなタームを使って説明されている訳ですが、ところがLyeeが他の言語や方法とともに使われていった場合、共通のメッセージを交換したり共通項がないと、それが上手くできなくなります。ですから、その為の共通のフォーマリゼーションが必要になってくると思います。

それから定理などの公理に関する詳細な説明は後でやられるか、今後出てくるのかも知れませんが、私はまだ全く理解しておりません。それでも根来さんが考えを表現する上で、他の公理系で使っている用語であるとか、意味上他と共有できるようなものがあつた方が今後、他の人々が理解していく上でも、何かシステム上の環境を作っていく上でも、よいと思います。

もちろん、まず根来さんが考えた構造の意味を理解するべきですし、それが重要な事ではありますが、その考え方の背景にあるフォーマリティについても、当然理解されるべきだと思います。

根来社長 Lyeeの概念を説明するのに、まだ私の個性の部分に留まっている

ところがあります。今回先生とお話して、私自身も大变得るところがありました。そういう意味でも、先生と一緒にやると、我々にとっても大変有意義だと思いますし、広い意味では大変社会に貢献できるのだと確信しています。

以上

\* 当内容の無断転載を禁じます。

Copyright (c)2001 CATENA CORPORATION