

Lyee Internet Information

東京工科大学 穂鷹 良介教授 インタビュー（後編）



日付 : 10月25日(水)

場所 : 東京工科大学 メディア学部

穂鷹 良介氏（筑波大学名誉教授 東京工科大学教授） 略歴

- 1937年 3月20日小樽生まれ。 現在63歳。
- 1959年 小樽商科大学卒業。
- 1959年 北海道大学大学院へ進学。
この時デジタルコンピューターを初めて使用。コンピューターの仕組みに興味を持ちのめり込む。
- 1964年 小樽商科大学講師、その後助教授として計算機科学を教える。
- 1966年 経済学博士を取得。
- 1969年 どうしてもコンピューターの世界を極めたく上京。
日本ソフトウェア株式会社に入社。
- 1978年 筑波大学 社会工学系に教授として任官。
データベース、データモデルを研究。
- 2000年4月 筑波大学を退官し、東京工科大学教授として着任。
時を同じくして、筑波大学名誉教授を授与される。

主な著作

- 「データベース入門」(オーム社刊)
「データベースシステムとデータモデル」(オーム社刊)
「データベースの論理設計」(情報処理学会刊) など

<インタビュー内容>

- Q : Lyee とデータベースに関して、メモリ千畳敷理論についてはどのようにお感じになりましたか？

- A : 良いと思いますね。全然驚きません。データベースでも全く同じように考えているからです。千畳敷のメモリが一枚だけあると仮定すると、全員がそれにアクセスするのですから、それでデータベースがシェアできて、何ら問題はありませぬ。その時排他制御は自然と出てくる訳です。

だから現在のデータベースは“千畳敷”みたいなものを見せかけているわけです。理論的には、狙っているところは、メモリ千畳敷理論も DBMS も似ているのではないのでしょうか？

- Q : Lyee とデータベースに関しては、物理的にどの様にブレイクダウンするのがわかりにくいとの声を聞きますが？

- A : ええ、それはやはり本来データベースの問題が難しいからだと思います。例えばデータベースは排他制御の仕組みを使って、皆が共通にみているということを実現します。共通に見ているということと少し矛盾するように聞こえるかもしれませんが、実は更新中のデータを他人が見ることは誤解を招くことがあるので禁止した方が良い場合があります。例えば、Aさんが自分の銀行の口座から1万円引き下ろすということをBクレジット会社が見ていて、自分もAさんの口座から5万円引き落とすということをしようとする場合、どちらも同じ残高を見るということはない訳です。この場合にはAさんの取引が終わるまでBクレジット会社はAさんの口座残高を見られないように制御してやる必要があります。

これとは別に単に参考の為だけ口座残高を見るだけならば他人が更新しているのがいまいが構わず見せてしまいたい場合もあります。同じデータベースに対してもこれらの差は利用者に意識してもらわなくてはなりませんから、利用者にその差を述べてもらう必要があり一見難しいように見えるでしょう。

- Q : LyeeALLの使い勝手は如何ですか。

- A : 正直言ってあまり良くないですねえ。(笑)
作った人には申し訳ないのですが。(笑)

- Q : どういう部分でしょうか？

- A : あれは自分が何を操作しているのか？という論理関係が自然に掴めないですからねえ。例えば“変更画面”というところで情報を見に行ったり、そのボタン

の意味と実際の自分がやってる事の意味がわからなくなったりしました。もう少し構造を上手く使ったら、いいインターフェースになると思います。私は今 Lyee に入っているデータを XML に書き直してみているのですが、XML を使うと、このカテゴリのどの部分のどの辺りをいじっているのか、位置が決まります。

つまり今自分がどこにいるのか良くわかります。LyeeALL だと、少しわかりづらいですね、平坦に見えて。

それと照会をしているのか、更新をしているのか、追加をしているのか、その辺のボタンの使い方が少しチグハグな印象を受けます。まあ、LyeeALL は Lyee の本質ではありませんが、実務的には重要な部分ですから、もう少し工夫すると、生産性に寄与するのでは? と思いますが。

- Q : Lyee に基づく XML の記述についてもう少し詳しく教えてください
- A : Lyee と XML の相性が良いとわかっているんです。Lyee での仕様の記述がツリー状であるようですし、XML はツリーを表現するのに便利な表現方法ですから。自分が巨大なツリーの中のどこにいるのか解ると、情報の投入が見やすくなるでしょう。
XML で書かれた記述のファイルを、インターネットエクスプローラの中にドラッグ&ドロップで落としてあげると、ツリー構造が素早く見られます。ですから学生に Lyee の構造を見せるには良いかなと思って。
LyeeALL が、入力方法として XML を採用すると、Lyee 構造を理解するには良いかもしれません。
そうすると選択肢も広がることになります。
- Q : Lyee では従来法と違った言葉遣いをしていますが如何ですか?
- A : もう少しわかりやすい表現にしてもらえると技術者にはありがたいですね。少し抵抗があります。
- Q : 単語毎に独立させたプログラム、フローチャートはいかがですか?
- A : いやー、あれはびっくりしましたね。あれが自動生成を可能にしているメカニズムなんだなあと。
- Q : もう少しその辺を詳しくお願い致します。
- A : 普通は、ユーザが概念設計から始めて、いろいろな工程・手順を経てプログラミングしますが、Lyee はスペックを書いたところでプログラミングが自動的にできてしまう。そのメカニズムは私自身未だはっきり理解している訳ではありません。

ませんが、2つあって、1つは意味を表現する為に使う全ての素材を予め用意してしまうんですね、Lye は、使われる可能性のある様なもの全て用意してあって、それを使ってストーリーを語らせる。W02 パレット、W03 パレット、W04 パレットなどですね。そのストーリーを語らせる場合に非常に簡単な要素、同期構造ですか？これでしか語らせない。全部素材が用意されているから開発ごとに新規の素材を改めて考える必要は無い。素材は対象に無関係に非常に簡単な構造だけが許されている。不思議なことに多くの対象の表現がこの簡単な素材の集まりだけで能力を持っているらしい。

素材の網羅性と秩序によって自動生成が可能となっているメカニズムだなあ！と。で問題は、それがうまく動くのかという点が勝負なのですが、どうも上手く動いているらしい。そこにどうして気づいたのだろうか？と、その点非常に不思議であり、驚いています。それから、用意ができていない時にはプログラムは何もしないで Exit してしまうという、あの仕組み。あの仕組み、あれがいいですね。

- Q : あのフローチャートは Lye の “ ミソ中のミソ ” の部分ですね。

- A : そうですね。あれはミソですね。あれは本当に良く考えられています。考えて、考えてあんなったのでしょうか、どうしてあれに気づいたのか、そこが知りたいですね。気づく必然性があったと思うのですよ。

私を知るデータベースのアルゴリズムの中で、エレガントだと評価できるものの一つで Lye に似た思想を持つものがあります。“ リニアハッシュ ” というんですが、データベースの中にランダムなキーのデータを入れていくのですが、たくさん入れていく内に要したバケット(容器)が満杯になっていきますよね。で、普通は満杯になったところで、セルを2倍にするとかして、ジタバタするのですが、リニアハッシュは満杯になってからジタバタするということはしないのですよ。

バケットの使用率は全体で何%か？というのを見ていて、それが一定の閾(しきい)値を越すとバケットの容量を増やすというようなことをやる。

丁度自分が困難に遭遇したとき、自分ではジタバタ解決しないんですよ。

世の中が解決してくれるのをじっと待つという態度を取るんです。そうすると個々のものに影響されないでしょう。この辺り Lye に似ているでしょう。“ 何もしないで ” 時がきて誰かが自然に解決してくれるのを待つ。その誰かはきちんと解決してくれるんですよ。

そういう摂理を持っている。好きですね。この逆転の発想というか、この思想が。そうすると個々のものは関係ないんですよ。考える必要はない訳です。全体の事だけ考えるメカニズムがあれば良い。Lye のロジックは、全体を記述す

るものあって、全体を制御するものがあれば、個々のところはやらなくても動くというメカニズムになっている。これは本当に面白いアルゴリズムですね。言い方が悪いですが、言ってみれば“他力本願”って奴ですかねえ。(笑)単にソフトウェアに限らず、独自で全て問題を解決できるものなんてありませんからその辺りが本当に世の中の基本原則に基づいている。Lye はね。

- Q : 単語毎に独立しているという事でメンテナンスビリティが向上すると思われませんか？

- A : それは Lye が個々にこだわってなくて全体がうまく動けば個々は、非常に良くなるという理論ですね。ですから個々について、いちいちコンパイルだの特別なコードを出してあげる必要がない訳です。皆一様に扱ってあげれば個々は良くなるそんな発想です。ですからメンテナンスビリティは向上するのではないのでしょうか。

- Q : Lye と DOA と比較するといかがですか？

- A : 実は私は DOA 信奉者ですからね。(笑)情報処理って、私自身プログラムをたくさん作りましたが、それで何が一番大事かという事を悟ったかという、入力と出力のスペックがきちんとしている事が一番大切で、それを決めてしまえば、ロジックは自然と決まると思っています。後は明々白々でしょ？常識のある人だったらコーディングできる。そういう理論が SDDM だった訳です。データ中心の考え方なんです。

これは穂鷹流 DOA ですが、DOA とは概ねこういった形ですから、Lye と似てはいますね。

データフローと ER ダイアグラムを以って DOA という I 社の DOA は、本家本元の言うものとは異なっています。本家本元がいうんですから。単に DOA という 3 文字略語が流行ったので、自社の持っているツールを組み合わせただけを以って DOA と言っただけです。いわゆる DOA っていうのは、データを定める事がプログラムの情報処理の基本だと思うんです。

DOA、つまりデータ中心って事は Lye でいうところの“単語”中心ということなんです。ただ大きな違いは、Lye はその単語と単語、言い換えればデータとデータを結びつけるところをスペックとしてきちんと規定してある。しかも同期構造という簡単なもので実現した。そこが凄いですね。そこは DOA と大きく違います。

- Q : Lye で開発を行う場合、どの様な分野に適しているとお考えになりますか？

- A : 業務アプリケーションには向いているでしょうね。事務計算プログラムはこの

まま行けば失業しちゃうでしょうね。

- Q : 例えば401kみたいなシステムですね。
- A : あれはLyeeに非常に都合のいいというか、適しているアプリケーションだったと思います。つまり自己生成ロジックがそれ程複雑じゃないと思うんですよ。画面遷移は難しいと思いますが。しかも新規業務の新規開発ですからね。Lyeeは全部自分で作っちゃうという色彩が強いですね。理論も含めて。時にはネットワーク上にサービスされているプログラムや部品を使わざるを得なかったり、そこに有るデータベースを使わなきゃいけなかったりと、Lyeeではない他人の製品やプログラムを合わせて開発していく事が多いと思うんですよ。その時にうまくアダプト出来るのか？出来る事はもちろん出来るのですが、どう実務的にパフォーマンス・効率を落とす事なく対応していくという課題があるんじゃないかなあとと思います。
一から十まで新しい物を作るプロジェクトではなくて、制約のある枠をはめられた環境でLyeeを使ってもいけるという事例が数多くあるといいですね。
- Q : 逆に、適していないとお考えになる分野はありますか？
- A : 自己生成の部分に偏微分方程式とか積分なんかの、サイエンティフィックなものが入ってくる用途とか、無限ループを使わなければならないものだったり、アニメーションとかエンジニアリングなど、そういった用途に向いているかどうかは、今の時点ではわかりません。
- Q : Lyeeのドキュメントである処理経路図はいかがですか？
- A : 非常に精密でよいと思います。スペックをきちんとしようと思ったら、あの様な形で書かざるを得ないでしょうね。
- Q : 何かLyeeの問題点はありますか？
- A : わかりづらい事なんだなあ。概念がね。あとこれは技術的な事だけど、コンパイルエラーが出た時に原因を探すの大変なんだよね。距離があるからね。リソースが膨大ですからね。
- Q : Lyeeの将来性はどう考えますか？
- A : 死ぬか世界制覇か、どっちかしかないと思いますよ。サクセス=世界制覇でしょう。もっと解りやすくすると爆発的に普及すると思いますね。そこに私も及ばずながら協力したいと思います。

Lyee が普及した暁には、現在の技術しか持っていない事務計算プログラムは討ち死にかもかもしれません。(笑)いや冗談抜きで。中途半端は許されないですよ。5割が Lyee なんて話しは無いでしょう。事務計算プログラムは大変だと思いますよ。

- Q : 総括致しますが、Lyee の最大の特徴は？

- A : 前にも述べましたが、根本原理を掴まえているらしい事と、あとハードウェア的発想を持っている事ですね。

ハードウェアって最初から全部用意してあるんですね。配線でね。バスだって全部のデバイスに信号を送るでしょう。で、個々が動くし、そうすると個々は忘れていい訳です。Lyee も最初に全部の素材を用意して、個々のロジックは全部同じにしておく。ロジックが難しくなるとハードウェアはコスト高くなるでしょう？だからシンプルな基本構造にしておく。Lyee もその部分、ソフトウェアとして具現化している。そこが見事ですよ。そこが工業製品と表現しても良い様な製品として成り立っている。そこが素晴らしいですね。

- Q : 弊社多摩センターにはソフトウェア工場も設置してあります。

- A : そうですね。あれこそハードウェア的発想がないと実現しないものでしょう。

- Q : 先ほども少し出ましたが、学生さんに Lyee を教える計画もあるそうですね。

- A : ええ。来年度 200 名から 300 名教える事になると思いますが、出来れば Lyee についても教えたいと思っています。ただ、今は理論が難しすぎてね。

Lyee の原理を 30 分で説明出来るように、教科書を書こうと思って、今必死に勉強してるんですよ。

Lyee の何から何まで知り尽くしたいですね。

- Q : その時点でまたインタビューさせて頂ければと存じます。今日はありがとうございました。

以上

* 当内容の無断転載を禁じます。

* Copyright (c)2000 CATENA CORPORATION