

## Lyce Internet Information

デンマーク デンマーク工科大学

### ヨルゲン・フィッシャー・ニルソン教授インタビュー



日時 : 4月12日(木)

#### 【ヨルゲン・フィッシャー・ニルソン(Jorgen Fischer Nilsson)教授 略歴】

- 1948年 コペンハーゲン生まれ 現在53歳。
- 1974年 デンマーク工科大学(DTU) コンピュータサイエンス学部で  
修士号取得。
- 1977年 同大学同学部で博士号を取得。  
IBM 研究室にて博士過程の研究を行う。
- 1979年 同大学同学部の助教授に就任。
- 1980年 同大学同学部の準教授に就任。
- 1987年 同大学情報技術学部の教授に就任。現在に至る。

<主な著作>

- ・ Conceptual Graphs as Algebras  
(代数学としての概念グラフ)
- ・ Inductive Logic programming with Well  
(帰納的論理プログラミング) など

## 【 Lyee の位置付け 】

- 今回来日されて Lyee に関する詳しい説明を聞かれた感想についてお聞かせ下さい。

**ニルソン教授** Lyee はもしかすると新しいコンピューティング・パラダイムかもしれませんが。現在コンピュータの世界には大きく分けて2つのパラダイムがあります。最もよく知られているパラダイムはシンボリック・パラダイムと呼ばれるもので、プログラム言語を通して表現されるものです。もう1つはニューラルネット・パラダイムで、コネクショニズムもしくはPDPとも呼ばれています。アーティフィシャル・ニューラルネットという言い方もします。コンピューテーションが分散されているという考え方です。Lyeeはそのどちらかに属するというのではなく、その中間に属すパラダイムなのかも知れません。

- コンピュータの世界には大きく分けて2つのパラダイムがあるとの事ですが、オブジェクト指向はどちらに属するのでしょうか？

**ニルソン教授** シンボリック・パラダイムです。

- ニューラルネット・パラダイムとは具体的にはどのようなパラダイムなのですか？

**ニルソン教授** アルゴリズム的なプログラムを書かないでシステムを構築する理論です。

- Prolog の様な第5代言語の様なものでしょうか？

**ニルソン教授** Prolog もシンボリック・パラダイムです。ニューラルネット・パラダイムでは、プログラム言語は使いません。マシンにノードのネットがあって、そのネットをトレーニングするのです。文字等を認識する様なタスクができる様に、ネットをトレーニングしていきます。ノードのネットをトレーニングした後に、分散された論理構造を得る事ができます。おそらく Lyee はこのニューラルネット・パラダイムとシンボリック・

パラダイムの中間に位置するものだと思います。

- 何故、Lyee はシンボリック・パラダイムとは違うとお考えなのでしょう？

ニルソン教授     シンボリック・パラダイムにも属していると思うのですが、ロジックがもっと分散された形になっていますので、プログラムの一部を移動させたりしても、その機能を維持する事ができます。私はニューラルネット・パラダイムの専門家ではありませんし、これは私の Lyee に対する現時点における感想です。Lyee を評価するにあつたては、更なる分析が絶対必要だと思います。

- 先生から見て、Lyee のアプローチで最も気に入った点はどこですか？

ニルソン教授     好奇心を掻き立てられたのは、Lyee の哲学的な考え方の部分です。3次元空間モデル、シナリオ関数等も興味深いのですが、これらの考え方をまだ完全に理解した訳ではありません。と言いますのは、まだ数学的な説明を聞いていないからです。こういった Lyee の基本的な考え方については、更に明確にされるべきだと思います。

## 【 Lyee の特徴に関する評価 】

- それでは Lyee がメリットとして表明しているものについて1つずつお伺いしていきたいと思います。まずはプログラムを開発する際に処理順序を考えなくてもよいというところについて意見を聞かせて下さい。

ニルソン教授     その部分が私にニューラルネット・パラダイムに少し関連すると思わせるポイントです。それはこの部分が非アルゴリズム的という感じがしたからです。しかしその部分について、自分は完全に理解できた訳ではありません。

- データ項目毎に独立したプログラムを作っていくというアプローチについてはいかがでしょうか？

ニルソン教授     オブジェクト指向に非常に関連性があると思います。ですけれどもだからと言って Lyee がオブジェクト指向と同じであると考えべきではないと思います。オブジェクト指向には最も重要な考え方が2つありまして、1つはインヘリタンスで、もう1つはカプセル化です。この部分が Lyee では何に相当するのかというのは、まだよくわかりま

せん。もし必要であればオブジェクト指向のプログラミングの専門家と話し合っ、見出していく事はできると思います。Lye e で使われている用語は従来のコンピュータ・サイエンスで使われている用語とは全く違います。ですから比較するのが非常に難しいのですが、マッピングや変換を行う前には、完全に理解をする必要があります。

- 先生が研究されている人工知能や知識ベースシステムと Lye e との関連性や共通点等  
はございますでしょうか？

**ニルソン教授** Lye e はソフトウェア工学に関連が強いと思うので構造化プログラムや UML の様なモデリング言語等と比較した方がいいと思います。今現在の私の理解では知識ベース等との関連はあまりないと思います。今日 Lye e で開発したアプリケーションを見せてもらったのですが、それは知識ベースシステムというよりはどちらかという、データベース・アクセス等に関連したものだと思うので、ビジネス系の Web システム等、ビジネスアプリケーションに向いていると思いました。

- どうしてビジネスアプリケーションに向いていると思われたのですか？

**ニルソン教授** 先程も申し上げました通り、プログラマーに対しての訓練を体系的に行え、生産性を一様にする事ができ、生産性が上げられるからです。ソフトウェア工場を見たり、頂いた情報の中で言える事は、従来法と比べて、生産性はかなり向上していると思います。

私はソフトウェアの生産に携わっている専門家ではありません。現在は科学者であり、大学で教える講師であり、ソフトウェア業界にいたのは、20年以上も前になりますし、実際の現場にいる訳ではありませんので、はっきりと申し上げられませんが、技術や技能の方が科学よりも重要になる場合があります。

例えば数百年前に船の正確な位置を図る為に、2つの解決策がありました。1つは科学的な解決策で、天文学的な計算に基づいたものであったのです。そしてもう1つは技術的な解決策がありました。技術的な解決策は正確な測定器を用いるというものでした。それにより数学的な知識のない乗組員でも、経度を図る事ができる様になったのです。

科学的な改善策は最善の改善策ではありませんでした。何故かと言いますと、船の上に数学者がいる訳ではありません。しかし、科学的な解決策を用いる為には数学者が必要になります。結局技術的な解決策の方が科学的な解決策よりも実践的であったという意味で優れていたのです。

- Lyee にもそれが言えるという訳ですね。

**ニルソン教授** それはみなさんのご想像にお任せしますけれども、これは歴史的な技術の 1 例でして、要するに我々は能力や技能と科学を混ぜ合わせて、判断を行わなければならないという事です。

- 先程、生産性が向上するとおっしゃられましたが、それはプログラマーが自分でロジックを組む事がなく、フローチャートの 7 つのボックスが全て決まっています。誰が作っても同一になるという事に起因すると思いますか？

**ニルソン教授** 私が聞いた説明の範囲で判断しているのですが、そういう意味で生産性は上がると思います。そして今聞かれた事がキーポイントだと思います。前もって解決策に課された構造という事ができると思います。Lyee はプログラマーに対して、前もって枠組みを提供しています。そして、訓練された様に Lyee が提案した手順を追っていかねばなりません。つまり全ての人全てが同じ用語を使って、同じものを同じ構造で作る事になります。従来法ではプログラマーが自分の想像力を使って人とは違ったものを作り上げてしまうといったリスクが常に伴う訳です。それによって管理やテストやメンテナンスが難しくなっていました。これは Lyee のアプローチの非常に良い点だという事は明らかなのですが、このアプローチの適応範囲というものが、非常に疑問です。プログラマーの生産性を上げる為に枠組みを提供しますが、その枠組みの範囲に問題があると思います。例えば顧客というのは、Lyee の提供する枠組みでは適応できないような機能を持ったシステムをほしいというかもしれません。

- 具体的にはどのようなものが考えられますか？

**ニルソン教授** 具体的な例を申し上げるのは非常に難しいです。と言いますのはそれを説明する為には、全体的なアプリケーションのフレームを作って説明しなければなりません。先程も申し上げましたが、私も実際に顧客と仕事をした事がありますけれども、顧客と一緒にプログラムを開発した事のある人なら必ず経験していると思うのですが、まずデザインをして、実行して、テストをして、そして顧客に見せます。それで顧客が満足してくれれば、それでいいのですけれども、もっと優れたもっと違ったこういった機能がほしいと言ったりするかも知れません。しかしそういった機能というのが、自分たちが既に作ってしまったデザインに当てはまらない様なものかも知れません。

また聞いた話なのですが、ログを印刷するシステムがありまして、印刷した紙が出てきた時に、顧客が出力されたログの一部を消すといった機能を追加したいと言ったそうです。でも現在のソフトウェア技術では、そういった機能を提供するのは難しいと思います。これは逸話的な例ですが。

全ての開発者は経験していると思うのですが、顧客は実現するのが難しいこちらが予測できないような機能を望む事があります。

- データ項目を追加すると言ったデモンストレーションをご覧になったと思うのですが、Lye は従来法に比べて、はるかにカスタマイズするのが容易であるというのをご覧になったと思うのですが、いかがでしょうか？

**ニルソン教授** もちろん生産性は上がると思いますし、顧客も生産性上がるシステムを作る事が可能だと思います。でもそれは当面の事であって、ユーザーが今後こういった要求をしてくるかというのは、予想しきれない部分があります。これはソフトウェアの開発・生産・メンテナンスの分野において、常に難題として位置付けられています。それは、どんな要求がされるかわからないからです。ですが、Lye の枠組みに適合した範囲については、広い範囲のいろいろなシステムは作れるというのは間違いないと確信しています。

- 先生が研究されている分野で、顧客が予測できない様な要求を出してくる事に対しての対策はございますか？

**ニルソン教授** この問題に対しての解決策はあるとは思っておりません。ソフトウェア工学においても同じだと思います。ある意味データベースというのはファイルにアクセスするのを制限している様にも取れます。要するにデータベース・プログラマーというのは、自分のやりたい方法で、ファイルにアクセスするのではなく、決められた制限された方法でしか、アクセスしたりという事ができないという意味なのですけれども、Lye の枠組みについても言える事は、プログラマーのスクリーンや画像の使用している様な状態になっていると思います。プログラミングに関して統一し訓練していくことは特に経験の少ないプログラマーにとっては、非常に良いと思います。しかし創造性というのも必要です。複雑なシステムをデザインしたり、開発する前に長い間分析をしている人達には、システム開発の上流レベルで創造性というのは非常に重要です。Lye の枠組みのメリットは、別のプログラマーが引き継ぐのが、簡単に

行えるという点です。例えばあるプログラマーが会社を辞めた時に、彼が作ったプログラムを誰も理解する事ができないといった問題が現状に起こっている訳です。そういった問題に Lyee の枠組みは非常にメリットを発揮すると思います。

- 今後、Lyee についてどのような取り組みをされる予定ですか？

ニルソン教授 これまでに説明して頂いた Lyee の考え方を、数学的に表現していく事が非常に有効であると考えています。そして Lyee の概念や用語をソフトウェア工学とコンピューティング・サイエンスにおけるそれらに用語に対応付けていくという事が重要だと思えます。

以上

\* 当内容の無断転載を禁じます。

\* Copyright (c)2001 CATENA CORPORATION