

## Lye Internet Information

### 対談

カナダ ラバール大学

モハメッド・メジリ教授

ソフトウェア生産技術研究所株式会社 根来 文生社長



場所 : ソフトウェア生産技術研究所株式会社 本社  
司会者 : 岩手県立大学 藤田八ミド教授

#### 【モハメッド・メジリ(Mohamed Mejiri)教授 略歴】

- 1968年 チュニジアのナブール生まれ。現在34歳。
- 2000年 カナダ ラバール大学でコンピュータ・サイエンスの博士号を取得。
- 2001年 ラバール大学コンピュータ・サイエンス学部の教授に就任。現在に至る。

#### <主な研究分野>

- ・プログラミング言語、コンピュータ・セキュリティ、フォーマル・メソッド、ソフトウェア・エンジニアリング

#### <主な著書>

- ・ Security by Typing
- ・ A Logic for Specifying Security Properties of Electronic Commerce Protocols

ハミド教授

まずこれまで研究されてきた事について、簡単に自己紹介を兼ねてお話下さい。

メジリ教授

私は1995年にチュニジアでコンピュータ・サイエンスに関するエンジニアリングの分野で学位を6年間かけて取りました。その後カナダから奨学金を得て、1996年から1998年にかけて修士号を取り、それからすぐ1998年から2000年にかけて博士号を取りました。その後ラバール大学からポストを頂き、教授になりました。現在に至ります。

修士課程から博士号を取得するまで私が研究してきた事はコンピュータのセキュリティについてです。特に電子商取引のプロトコルを分析したり、検証したりするといった研究を行ってきました。この研究はコンピュータの様々な分野と非常に密接な関係があります。セキュリティの問題を解決しようとする、プログラム言語の問題やフォーマル・メソッドの問題などに関係してくるのです。

現在、私は大学でLSFMというグループに所属しています。LSFMとはランゲージ、セマンティック、フォーマル・メソッドの3つを表している言葉なのですが、この名称が意味する様に、言語とフォーマル・メソッドについて特に興味を持っているのですが、あくまでもセキュリティとの関係の上で興味を持っています。

現在私のもとには修士課程の学生がおりますが、その学生達もセキュリティについて学んでいます。

またこれまでも他の大学やグループとの共同研究も行ってきました。例えばスタンフォード大学のジョン・ミッチェル氏と共同研究を行いましたし、ケベックの国防省との共同プロジェクトではセキュリティ問題に関する研究を行いました。ここでプログラムの静的・動的分析を行ない興味深い結果を得ております。

ハミド教授

では次にLyeをどういう経緯で知ったのか、また今回来日された経緯について教えて下さい。

メジリ教授

そもそも先程申しましたLSFMにLyeを学ばれた経験のあるモラド・デバビ先生（ラバール大学教授、LSFMディレクター）とジーン・ベルジェロン先生も参加されておりました。

最初にLyeの話聞いたのはデバビ先生からでして、その時の感触としましては、とても面白いものだなという感じだったのですが、哲学的な側面も含め、多くの疑問も持ちました。その後、ベルジェロン先生からもLyeの話聞きまして、同時に資料も頂きました。

2人から頂いた情報を基に、まだ疑問はありましたけれど、徐々にイメージが湧いてきました。2人からは技術的な話、特にレガシー変換の話をして頂きました。アセンブラから COBOL に変換するといった話をして頂いたのですが、その際に私の持っている情報と理解を合わせて、例え Lyee の哲学的側面がわからなくても、高度な自動変換ができるものなのだという事はわかりました。

最初の何となく面白そうだという感触から始まりまして、これは面白い背景があるといった実感が深まった訳です。今後まだ気づいていない興味深いアイデアなどが出てくるのではないかと思います。また Lyee の背景にある理論についても探究していきたいと思います。

## 【 Lyee の評価について 】

Lyee ではメンテナンスが非常に簡単になるという事は容易に理解できます。これは非常に大きな貢献だと思います。

ハミド教授

メジリ先生は国際学術共同研究プロジェクトの参加メンバーの内で最後に来日された訳ですが、ある意味で一番知識を持たれた状態でいらっしやいました。それはベルジェロン先生やモラド先生といった、他の先生から Lyee についてかなりの情報を得て、よい感触を持たれた上で来日して頂いているからです。

他のメンバーの先生方は既に来日され、研究を進められています、ある意味でいい環境の中でいらしたと言えます。

私達はメジリ先生がどのような研究をやられるのかという事だけでなく、ラバール大学のユニットとしてどういった事をやられるのかについても、とても興味があります。

それでは次の質問ですが、現時点の Lyee の評価について教えてください。それからどういった研究を行いたいのかといったビジョンについても教えてください。

メジリ教授

Lyee の最も重要なポイントは、哲学的な側面に支えられていながらも、それがわからなくても立派に機能するという事です。その要因の1つとしてプログラムの自動生成があげられます。これは非常に重要な事で、プログラマーの負担を軽減する事ができます。

ある意味でコンピュータが出来た事で、プログラマーの仕事が生み出された訳ですが、今度は逆にその仕事をコンピュータにやらせて

しまおうという事なります。この部分が非常に優れていると思います。自動的にプログラムを作る事によって人間的なミスが減らす事が出来ます。またプログラムを作る作業が簡潔で単純なものになり、ソフトウェアを作っていく上でコンピュータ・サイエンスの知識や学位は必要なくなります。この事について今朝多摩ソフトウェア工場を見学して確認できました。

次に検証や確認、いわゆるテストの問題ですが、Lyeではテストをしなくても良いという点も大きなポイントです。ただし現在どれくらいテストがいらぬのか、100%なのか、10%なのか、正直に言ってまだ正確にはわかっていません。この点についてはもう少し研究をしていきたいと思っています。

3つ目のポイントとしてメンテナンスについてですが、Lyeではメンテナンスが非常に簡単になるという事は容易に理解できます。

どうして可能なのかと言いますと、単にリクワイヤメントの定義と条件を与えるだけでプログラムを作れるからで、その部分に論理的な側面が入っていないからです。

メンテナンスの点でもう1つ付け加えますと、通常ソフトウェアを開発する際に、メンテナンスが最も面倒な作業となります。アプリケーション次第ですけど、現状では70%以上がメンテナンスに費やされています。これを単純化する事ができれば、当然時間とコストが大幅に削減できます。これは非常に大きなLyeの貢献だと思います。

以上が現時点で私が理解したLyeの特徴です。まだよくわかっていないのは、Lyeの哲学的な側面です。この点は非常に重要だと思いますし、この点が理解できる様になりますと、他の問題についても答えられる様になるのではないかと思います。つまり哲学的な側面が理解できるようになれば、Lyeの他の特徴も容易に見出せる様になるのではないかと思います。また哲学的・理論的な背景がわかると、テストがいらぬという事がどういう事を意味しているのかといった問題に対して、明解な答えが出せると思います。

根来さんはこの方法論を考えるにあたりまして、多大な時間を費やされて、非常に細かい点まで考えられてきた訳ですから、短い時間で説明を聞いただけの私が理解できないのも、ある意味当然だろうと思いますが、理論面について話し合う機会があれば、はっきりしない部分や確信が持てない部分についても、答えが出てくると思います。はっきりしない点についてもう1つあげますと、Lyeがどういった

アプリケーションに適しているかという事です。全てのアプリケーションに適応できるという説明を受けましたが、実感としてはまだわかっていません。ただビジネス・アプリケーションに向いているという事はわかります。しかしながら数学的なアプリケーションに向いているかについては、私自身確認していきたいと思います。

ハミド教授

最後にメジリ先生がおっしゃられた、Lye がどの分野に適しているかという点について焦点を絞って話を進めたいと思います。

どの分野に適しているかという点はいかなる方法論についても言える事で、これはアプリケーションとの関係の相対的な問題だと思います。根来さんは理論や哲学を基にこの方法論を開発されましたが、その過程の中で実際にビジネス・アプリケーションの開発もビジネスとしてやられた訳ですから、それと合わせて方法論を整理していったと思います。そういう事情からビジネス・アプリケーションに向いている様に見えるということもあろうかと思います。

しかしこの方法をビジネス・アプリケーション以外の分野に適用する際にどういった調整が必要なのかといった事が次に問題になると思います。どんな調整の仕方がありうるのか、その為には何が必要なのかといった事です。

ビジネス・アプリケーションに関しては、既にある種の標準型ができており、この分野についてはある意味で専門家になられたのですから、標準形をいろいろな形で適応する事で、開発ができる様になっていると思います。

しかしながら、例えば数学モデルなど経験していない分野に関しては、専門知識やノウハウをまだ構築しておりません。例えばリアルタイム・システムなどが当てはまると思うのですが、そういう分野でも一旦モデルができましたら、それを他に適応していけば自動化が可能になるだろうと思います。

この問題は非常に興味深い問題ですので、国際学術共同研究プロジェクト全体で考えて行きたい大きなテーマだと思っております。

根来社長

メジリ先生のお話を興味深く聞かせて頂きました。Lye の特徴について3点あげておられましたが、もう1点開発作業が並列化できるという事も Lye の特徴にあげられると思います。

また Lye の特徴の1つとしてセマンティックス (semantics : 意味論) への配慮があります。Lye はセマンティック・テストをある種のシンタックス (syntax : 文章論)・チェックに置き換えているのです。この

関係についての説明はまだ不十分だし、あるいはその中には整理を要する部分も残っていると思います。

少しコメントを加えますと、従来法にセマンティクスは存在していません。セマンティクスの定義を従来法ではまだ誰も行っていません。

最後にどの分野に適しているかという点ですけれども、この問題は先程ハミド先生も指摘されました様に、方法論の場合に必ず出てくる問題です。その時、大事な事はコンピュータの使い方の問題とソフトウェアの問題を混在してはならないという事です。Lyeというソフトウェアの構造と現在のコンピュータ・アーキテクチャーは、実はまだ完全にフレンドリーではないのです。そういう意味ではLyeのプログラム構造に適したものはありませんので、それは工学的問題として将来解決される事を希望しております。

最後にコメントを聞いて嬉しかった事を1つ申し上げたいと思います。Lyeの背景にある考え方を理解すると、Lyeの可能性がわかるとおっしゃられましたが、全くその通りだと思います。

メジリ教授

根来さんが指摘された並行に作業が行えるという特徴については、私も理解していたのですが漏れておりました。

それからコンピュータのアーキテクチャーを変えなくてはならないという点については私も賛同しております。現行のアーキテクチャーが作られたのはかなり前の事なので、21世紀の私達のニーズに適応していないと思います。それなのにこれまでに誰も新しいアーキテクチャーを考えようとしていないというのは驚くべき事だと思います。それからもう一点、Lyeがあらゆるアプリケーションに適応できる事の事ですが、おそらく本当なのだろうと思います。ただ現時点で私はそれが具体的な形で見えていないのです。何故見えてないのかと言いますと、結局問題を解決する際に、私にはある特定の考え方ができてしまっています。そこには当然ロジックが入っている訳です。私には自分なりの考え方が染み付いていますので、それに対して1日とか1週間で全く新しい考え方を付け加えて考えるという事は人間でするのでできません。ですので、ステップ・バイ・ステップで追っていく事により理解していこうと思います。

今朝多摩ソフトウェア工場を見学しまして、なまじコンピュータ・サイエンス等の知識があるよりも、プログラムとは何かとかアルゴリズムとはどういうものかとか、そういう知識がない方がわかり易いという事がよくわかりました。

それからどの分野に適応するかという点ですが、例えば Prolog は人工知能分野に向いていますが、これを使ってビジネス・アプリケーションを作ろうとすると大変難しくなってしまいます。ですからそれぞれの方法論が適した分野を持っているという事はあるのではないかと思います。しかしながら Prolog でもビジネス・アプリケーションを作ろうと思えば作れるのです。ですから、こういった適性を持っているかという点について、Lye をみていきたいと考えています。

根来社長 私はコンパイラは分野別のものがあるのがあって当然だと思います。しかしメソドロジーは自然言語に変わるものですから、普遍性が求められるべきです。分野毎のメソドロジーという言葉は正しいメソドロジーの定義ではありません。

メジリ教授 それは面白い考えだと思います。今根来さんのおっしゃられた事はその通りだと思います。

## 【 Lye のカテゴリーについて 】

ある意味 Lye はそれ自体 1 つのカテゴリーだと思うので、あるカテゴリーで説明をする事はできないと思います。

ハミド教授 それでは次の質問にいききたいと思います。メジリ先生は Lye を学ばれて間もないので、まだはっきりとしないとは思いますが、コンピュータ・サイエンスの専門家でいらっしゃると思いますので、Lye はどういったカテゴリーに分類されると思うかご意見を聞かせて頂きたいと思ます。

メジリ教授 現時点 Lye を他の方法と比較して、どの分野・分類に入るという事を 100% 言い切る事はできませんが、あえて近いと言えば、宣言型言語に近いのではないかと思います。

Lye を見てみますと、ある部分は確かに宣言型言語に似ているとか、またある部分は並列処理に似ているとか、あるいは定義の仕方がオブジェクト指向に似ているとか、そういう事は部分部分では言えますけれども、全体を通してこれというのはありません。ですからその為に全く新しい分類を必要とするものだと考えています。

確かに人によっては単語の定義の仕方であるとか、ある種の分類があるところから、オブジェクト指向にかなり近いといった見方をする人もいるかもしれませんが、よく見れば全く似ても似つかない

ところがあります。また宣言型言語に関しても同じ事が言えます。確かに宣言型言語にも繰り返しの機能はありますが、Lyeの場合には述語構造の7つのボックスのルールがイタレーションを決定しています。他の言語にはこういった決定の仕方ではできません。Lyeは普遍的な構造が組み込まれています。ですから、これも似ても似つかないものになると思います。ある意味Lyeはそれ自体1つのカテゴリーだと思うので、あるカテゴリーで説明をする事はできないと思います。

今まで皆、Lyeを他の方法と比較しながら見ていこうとしてきた訳ですが、私がプロジェクトでやってみたいのは、逆にLyeを使って他の方法を見てみるという事です。

## 【 Lyeの研究テーマについて 】

並列プログラムの生成や生成するプログラムの最適化について  
研究する事を考えています。

ハミド教授

では次に国際学術共同研究プロジェクトに関する質問ですが、こういった形でプロジェクトに参加できるかについてお聞かせ下さい。

メジリ教授

こちらに伺ってから数日の間に様々なツールを見せて頂きまして、いろいろなアイデアが浮かびました。

特にレガシー変換に興味を持っているのですが、今可能なのは非常に限られた言語のあるクラスから非常に限られた言語のあるクラスへの変換ですが、それを非常に大きなクラス、いかなる言語間でも扱える様にするという事ができると思います。またあるオペレーティング・システムから違うオペレーティング・システムへの変換なども可能だと思います。

またひょっとするとサブプロジェクトに成り得るかもしれないのですが、Lye構造のプログラムから並列プログラムを生成するという事も考えています。結局アイデアとしてはLyeALLがやっている事と同じ事なのですが、しかし目標とするところが並列プログラムであるというところが違います。この並列プログラムはいろいろなネットワークに結び付けられたところで実行できます。

あと考えているのはLyeALLで生成するプログラムの最適化についてです。最適化をどのレベルで行うのかというと、メモリと実行時間に



ついてです。この部分は LyeeALL と結び付ける事ができると思います。そして最適なプログラムを作るのです。LyeeALL と結び付けて最適なプログラムを作る事も可能ですし、LyeeALL とは独立して別に最適化の為のプログラムを作る事も可能だと思います。独立した形にする場合、LyeeALL で生成したプログラムをインプットとして、最適化プログラムをアウトプットします。これはたぶん重要なポイントだと思います。最適化プログラムを作る事でメモリの使用量の削減と実行時間の短縮が可能になります。

また LyeeALL で生成されたプログラムの検証についてですが、LyeeALL で生成されたプログラムにはいろいろなポリシーがありますが、それがリクワイヤメントに沿ったポリシーになっているかどうかを検証するツールを作ることが考えられます。言い換えますと、顧客はリクワイヤメントを通じてある種の方針を表現したい訳ですから、LyeeALL で生成されたプログラムが、例えばセキュリティであれビジネスであれ、自分達の方針をちゃんと尊重したものになっているかどうか、それを検証するという事です。例えば銀行の業務で言いますと出金と入金のパランスが絶対を取れていないとならないとか、そういう事をチェックするのです。

それからレガシー変換についてですが、顧客はレガシー変換を行ったプログラムが、自分達が持っていた元々のプログラムと同じ事ができるかどうかを知りたいと思いますので、変換したプログラムが同じ成果を発揮する事ができるという事を証明するツールを作るという事を考えています。ツールで自動的に証明する事ができれば、顧客が心配して自分達で調べる必要がなくなります。

今朝多摩ソフトウェア工場を見学した際にそこで働いている人達に、レガシー変換を終えた後、顧客はどうするのかと聞いたところ、顧客が工場に来て変換したプログラムが同じかどうか確認するか、顧客が持ち帰ってチェックするという答えが返ってきました。そこでもし自動的に確認を行えるツールがあったら便利だと思わないかと聞いたところ、そう思うと皆さんおっしゃっていました。

更にもう1点ありまして、現在は顧客から単語の属性をリクワイヤメントとして提供してもらっている訳ですけれども、この属性というのはコンピュータと関係があるのであって、思考と関係があるのではありません。ですから、この部分を簡素化できるのではないかと、ツールを作って単純化できるのではないかと、思います。更に進化させていけば、この部分を完全にツール化して排除してしまう事も可能では

ないかと思うのですが、完全自動化はできないまでも、可能な限り簡素化してしまう事はできると思います。

以上が現時点で考えている事ですが、LyeeALL を実際使ってみると更に面白い発想が出てくると思います。

ハミド教授

今日はどうも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) 2002 CATENA CORPORATION  
& The Institute of Computer Based Software Methodology and Technology