

システムズエンジニアリングの最前線

— 技術・人・社会を含めて考えるシステム開発と運用 —

The Aerospace Corporation
Principal Engineer

ジェイムズ・マーティン



SEC 所長

松本 隆明

大規模で複雑なシステムを、ソフトウェアやハードウェアといった従来の視点ではなく、システムの利用者や社会構造などの全体から捉え、高い信頼性をもったシステムを開発・運用するというアプローチがシステムズエンジニアリングの考え方である。その第一人者として活躍するジェイムズ・マーティン氏にお話を伺った。

INCOSE とはどのような組織か

松本：今日は、システムズエンジニアリングに関する国際組織である INCOSE の中心メンバ、ジェイムズ・マーティンさんにおいでいただき、システムズエンジニアリングについてお話を伺いたいと思います。IPA/SEC は、ソフトウェア・リライアビリティ・エンハンスメント・センターの英語名称の通り、基本はソフトウェアを中心にシステムの安全性を考えていくということに取り組んでいます。ところが最近では、システムがどんどん複雑化・大規模化し、更に色々なシステムが複雑に連携する形になっており、これはソフトウェア、これはハードウェアという形で単純に区別をしてシステムの安全性を考えることはできなくなってきています。

これからは、よりシステム的な見方や考え方、あるいはシステム的なアプローチに注目し、ソフトウェアも含めて、システム全体で捉えていくシステムズエンジニアリングの重要性が増してくると思っています。

さっそくですが、まず INCOSE という組織についてお伺いしたいと思います。

ジェイムズ・マーティン（以下、マーティン）：INCOSE はインターナショナル・カウンシル・オン・システムズエンジニアリングを正式名称とする組織で、1990年に設立されました。システムズエンジニアリングを行っている様々な企業が、複雑なシステムの設計を単独で行っていくことは難しいということから、協力してそれを解決していくために設立された組織です。従って、当初は、とくに複雑なシステムを扱う可能性の高い航空、宇宙、防衛に関連する企業が参加しました。

松本：会員は一般企業だけでしょうか。それとも、政府あるいは大学の機関・関係者も入っているのですか。

マーティン：二種類のメンバが参加しています。まず個人メンバ。それからコーポレートメンバです。コーポレートメンバには、企業、大学、政府官公庁といった様々な組織が含まれます。しかし、もともと INCOSE は個人メンバのために設立されたものであり、システムズエンジニアリング学会のような、システムズエンジニアリングのプロが集うという趣旨でスタートしています。



ジェイムズ・マーティン

エアロスペース社所属のエンタープライズ・アーキテクト、システム・エンジニア。INCOSE (The International Council on Systems Engineering) のフェローであり、標準技術委員会のリーダーとしても活動している。SEBoK (Systems Engineering Body of Knowledge) の編集においてはBKCASE プロジェクトの主要執筆者として参加し、エンタープライズ・システムズエンジニアリングについての見識を広めた。また、システム工学のプロセスを定義した米国の標準 ANSI/EIA 632 の制定を担うワーキンググループを率いる。過去には AT&T のベル研究所にて、無線通信製品や海底の光通信ケーブル製品に従事した経験を持つ。著作に「Systems Engineering Guidebook」(CRC Press) などがある。

INCOSE の活動内容

松本：具体的にはどのような活動をしているのですか。

マーティン：主な活動の1つは国際シンポジウムです。これは個人または企業の方から、新しいコンセプト、新しい方法・手法、新しいツール、または事例などを発表していただくものです。それ以外にも、国際的なワークショップを行っています。

また、INCOSE の中には30ほどのワーキンググループがあり、それぞれのワーキンググループで、独自の問題に取り組んでいます。例えば、要求、アーキテクチャ、リスク、デザインなどです。

松本：ワーキンググループでは、方法論であるとか、そうした開発を通して標準化につなげていくという活動をされているのでしょうか。

マーティン：現行のやり方を文書化する活動は行っていますが、国際的な標準化団体のように標準作りをするという事はしていません。現行のプロセス、手法、ツールなどを記述し、ガイドブックやワークペーパーに編さんしていくという作業を行っています。また、そういうツールがどう実装されているのか、それを文書化し、ワークショップを行う時に、それぞれの領域で行われた作業結果を比較しています。

松本：そのガイドブックは、会員企業以外の人でも見たり使ったりすることはできるのでしょうか。

マーティン：「ハンドブック」と呼んでいます。有償で購入していただくことはできます。

松本：「ハンドブック」を作って広げていくということには、システムズエンジニアリングの人材育成という意味もあるのですか。

マーティン：「ハンドブック」は、一つの目的のためだけに作られているものではありませんが、INCOSEが行っているシステムエンジニアの認定を受けるための試験は、このハンドブックに基づいています。また、この「ハンドブック」は国際標準 ISO/IEC 15288 に基づいたものです。もちろん、これを個人が仕事のために使うこともできます。

松本：その認定は、INCOSE が認める認定の仕組みですか。

マーティン：システムエンジニアの認定は、INCOSE として行っています。

松本：つまり、ISO の認定ということではなく、それをガイドブックの形で少しわかりやすくしたものの認定

を INCOSE が独自に行っていると理解して良いのでしょうか。

マーティン：ISO ではシステムエンジニアが何をやるのか、という記述がありますが「ハンドブック」は、それをより膨らませ、そのプロセスをどのように行っていくのかというところまで含めた内容になっています。先ほど申し上げた認定試験というのは「ハンドブック」に記述されている内容の知識を測ることによって認定していくものです。ただし、認定は試験に受ければ良いだけではありません。知識レベルは試験で調べますが、加えて経験、更に他の人からの推薦が必要です。他の人というのは、ほかの認定されているシステムエンジニアやその他の経験豊富な人からの推薦という意味です。従って知識の部分は、経験と推薦を加えた3つの要件のうちの1つということになります。

システムズエンジニアリングの拡大

松本：INCOSE における現在の一番のトピックは何ですか。

マーティン：トピックとなっているのは、モデルを使ったシステムズエンジニアリングです。更にもう一つ、よく議論になる重要なトピックは、従来の領域を超えたところに考え方を広げていくということでしょう。例えば、商用のプロダクトエンジニアリングや医療のエンジニアリング、自動車または交通といった領域です。もともとシステムズエンジニアリングが始まったのは、防衛や軍事の世界ですが、そこから更に他のドメインに広げていくということがホットトピックになっています。

松本：モデルベース・システムズエンジニアリング (MBSE) については、SysML などのツールもあり、やり方としてはある程度ポピュラーになりつつあるのではないかと思います。MBSE についてはどうい



松本 隆明 (まつもと たかあき)

1978年東京工業大学大学院修士課程修了。同年日本電信電話公社(現NTT)に入社。オペレーティング・システムの研究開発、大規模公共システムへの導入SE、キャリア共通調達仕様の開発・標準化、情報セキュリティ技術の研究開発に従事。2002年に株式会社NTTデータに移り、2003年より技術開発本部部長。2007年NTTデータ先端技術株式会社常務取締役。2012年7月より独立行政法人情報処理推進機構(IPA)技術本部ソフトウェア高信頼化センター(SEC)所長。博士(工学)。

議論がされていますか。

マーティン：INCOSE の中で話をしているのは、例えば、色々なモデリングの手法と統合していくというようなことです。SysML というのは、ソフトウェアの世界における UML の拡張版と言われています。ソフトウェアの世界からシステムの世界へ広げていくということで生まれてきたのが SysML ですが、それを様々なツール、様々なモデルと統合していくという話をしているのです。というのも、既存のシステム、または従来型のシステムが様々なあり、分析のモデル、パフォーマンスモデル、更にはシステムシミュレーションなどといったものを、SysML と統合していくということです。

システムというのは、単なるハードウェアやソフトウェアというより、もっと大きなものです。システムの中には人も含みます。その人の部分をどうモデリングするのか、ということについては、SysML にはできない。その人の部分のモデリングのために、どういうモデリング言語を使うのか、あるいは分析ツールを使うのか、というところも大きな問題になってきます。システムの部分、またはセキュリティというところにも人はかかわってくる。その人の部分をどうモデリングするのか、というのが大きな問題なのです。

松本：それは重要なポイントだと思います。ヒューマン・ファクター（人的要因）というのは、システム全体を考えた時に、非常に大きな比重を占めるものですね。

私たちも今、色々なシステムの障害がなぜ起きているのか、ということ調べているのですが、やはり人の要因によるところがかなりあります。人を具体的にどのようにモデルの中に取り込んでいくのかについて、INCOSE では何か具体的なアイデアを出されているのでしょうか。

マーティン：残念ながら INCOSE でその部分を考えていくのは困難であると思っています。というのは、INCOSE のメンバは、そのバックグラウンドとして、ハードウェアまたはソフトウェアの世界の人間です。従って、人の側面、またはソーシャルエンジニアリングや社会学の側面を一番の専門としているメンバではないのです。人の部分に対して、最も適切に包括的に考え方を作り出していく適切なメンバではないのです。ただ、人の部分の本当に一部ではありますが、そういった部分を扱うワーキンググループはあります。ヒューマンセーフティのワーキンググループやエンタープライズ・アーキテクチャのワーキンググループです。

システムエンジニアは何をするのか

松本：INCOSE の会員の数は、どんどん増えているのでしょうか。

マーティン：現在のメンバ数は 8,000 程ですが、毎年少しずつ増えています。

松本：日本では 3 年ほど前に JCOSE という支部のようなものができていますが、まだ日本の国内では、システムズエンジニアリングの技術者はそれほど多くないというのが実状ではないかと思います。日本とアメリカあるいは海外との違いということで、何か感じられるところがありますか。

マーティン：今 8,000 と申し上げた INCOSE のメンバのうち、半分の 4,000 は、米国以外のメンバです。もともと、米国のみで始めたものが、かなり米国以外で増えているという状況です。

現在は、先ほどもお話したように、防衛だけでなく商用の領域にも広がっていき、としているわけですが、その際の難しい点が、商用の領域になった時に、システムエンジニアがシステムエンジニアとは呼ばれていないというところ。例えば、プロダクトエンジニアまたはマニファクチャリングエンジニアと呼ばれているので、システムエンジニアであると認識されていない。それが一つの障壁になり、そういう人達にアクセスしにくいという状況が生まれています。

松本：そもそもどういうことをやる人がシステムエンジニアなのか、そこが明確に理解されていないということがありますね。

マーティン：おっしゃる通りです。そもそもエンジニアが何をやるのか、ということがわかりにくい。エンジニアが行うのは、抽象的なものであったり数学的なものであったりします。それがシステムエンジニアになると、更に抽象的、更に数学的で、具体的なものではない。結果が目に見える有形のものではないということによって、システムエンジニアが何をやる存在なのかという理解が難しくなっているのだと思います。

松本：私が伺ったマーティンさんのプレゼンテーションの中で、PICARD 理論のお話がありました。システムというのは、人によって見方が全然違ってくる。システムというのは、単にパーツの組み合わせだけではなくて、色々な見方によって変わってくるものだと。それをエンジニアリングしていくというのが、システムズエンジニアリングの仕事になってくるということでしょうか。

マーティン：エンジニアリングはパーツにフォーカスしたものだと思いますが、システムズエンジニアリングと言う時には、パーツ間の関係、これをエンジニアリングするものだと思います。従ってパーツよりも、そのパーツ間の関係や、パーツ間の相互作用、つまりインターアクションの重要性が高くなります。

実は私の PICARD 理論ですが、これは日本で作った理論なのです。ある電機メーカーに、システムのコンセプトを教える時、自分の母国ではない国で、違う言語で、大変複雑なコンセプトを教えなければならない。そのため理解していただきやすい方法が必要であるということから策定したのが PICARD 理論でした。

松本：同じパーツを組み合わせても、使われるコンテキスト、状況に応じてシステムは変わってくるし、何を目的にパーツを組み合わせ、システムとして提供するかによっても、見方は違って来る。つまりシステムには色々な属性がある、と考えればいいのでしょうか。

マーティン：システムズエンジニアリングは、そうした様々な属性を調べて、システムがきちんと正しい方法で振る舞い、正しい成果を出せるようにしていくということを行っていきます。その際は、正しいコンテキストで正しい成果を出せるように考えていくと共に、未知の環境、想定されていなかったような状況の中でも堅牢性が持てるようにしていかなければなりません。システムが使われる環境というのがすべて知り尽くされているとは限りません。従って、未知のコンテキストに対しても、きちんとした振る舞いができるようなシステムにしていくということも、考慮に入れる必要があります。

松本：未知の環境というのは、未知なので結局はわからないわけですね。例えば、どのように世の中が変化していくか、ビジネスが変わっていくか。技術が進歩していくか。それを予測するのは、非常に難しいと思うのですが、その点はどう考えていらっしゃるんですか。

マーティン：一つの方法は、システムを学習型にしていくということです。システム自体が新しい環境を理解し、それに対して適応ができるようにしていくということ。そうすれば、複雑なシステム、適応型のシステムになり、設計自体は難しくはなりますが、未知の環境に対して対応ができるようになると共に、システムを未知の脅威、未知の条件に対して対応できる堅牢なものとすることもできます。また、様々な多数のシナリオを使ったテストを行っていくことも考えられます。更に、進化型のプログラムにいく、ということも考えられます。システ

ム自体が学習し進化していく。将来が一旦飛びに予測できなくても、暫時出てくる条件に合わせて進化していくという考え方です。

「ユーザ要件」だけでは不十分

松本：先ほどのお話にあった、システムを色々な属性で見えていくという時に、誰がそのシステムを見るかによっても変わってくるのではないかと思います。つまり、ビジネスパーソンがシステムを見る時の捉え方と、技術者や開発者がシステムを捉える見方と、今度は運用する人が、オペレーターも含めて、システムとして捉える見方というのは、それぞれ変わってくると思うのですが、どういう視点でシステムを捉えるかということについて、一つの考え方はあるのでしょうか。

マーティン：以前は、システムズエンジニアリングということを行っていく際、ユーザ要件だけが考えられていました。しかし、それでは十分ではないということで、今ではオペレーター、マネージャ、オーナー、構成管理をしていくビルダー、検証をするテスター、更に政府、スポンサー、資金調達をする銀行、こういったところすべてを含めて、いわゆる“ステークホルダ”、利害関係者と呼ぶ人を考慮に入れる必要があります。一番最初から、フロントエンドでシステムのエンジニアリングをする時から、ユーザ要件だけを考えるのではなく、こういう様々なステークホルダの要件を考えていく、というように考え方を膨らませているのです。

松本：そうすると、システムズエンジニアに求められるスキルが、ビジネスのこともわからなければいけない、ユーザビリティもわからなければいけない、開発者として、実際の開発をどうするか、ということもわからなければいけない、というように本当に幅広い知識が必要になってくると思うのですが、そういう人を育てるのは、かなり難しいのではないかと思います。

マーティン：おっしゃる通りです。一つのシステムエンジニアのチームに、すべてのそうした部分のエキスパートをそろえるということはできません。システムエンジニアが、そのチームの中で、いわゆるファシリテーター（進行役）の役割を担い、様々なステークホルダを関与させていく、そういったことが必要だと思います。外部の人たちにアクセスし、そこから情報を吸い上げ、それを理解し、問題に関しての全体的な理解が取れるようにしていくということが必要になると思います。

標準フレームワークとしての「7SAMURAI」

松本: マーティンさんが提案されているものに「7SAMURAI」があります。システム構築の方法を、7つの視点から考えていくという意味で“七人の侍”という比喻が使われているのだと思いますが、それについてご説明いただけますか。

マーティン: 基本的には、まずエンジニアリングで考えていきます。ソリューションというのは、システムの中の一つを考えていくわけですが、それと共に、システムが存在する全体的なコンテキスト、このコンテキストたるシステムというものも考えて行かなければなりません。そして、開発されたソリューションですが、例えば、それを開発する開発組織というものも別の一つのシステムとして考えていく必要があります。更にまた、開発されたソリューションが実際に効果を発揮できるかどうかということにかかわる実現のためのシステム、これが、どれぐらいのシステムなのかによって制約を受けることになるので、その実現システムに合わせたソリューションにしていくという調整も必要になります。

松本: 一つの問題をどう解決するか、システムによってどう解決するか、ということを考えるのが最初のターゲットだと思うのですが、それをどんどん広げていくことをしていくと、検討の範囲が広がり、実際に開発すべきシステム範囲が、なかなか確定できなくなるのではないかという心配があるのですが、そのあたりは、どのように絞っていけばいいとお考えですか。

マーティン: この「7SAMURAI」というのは、標準のフレームワークを確立するものだと考えています。問題に対して、フレームワーク、枠組みを調整していくことができる。良い枠組みがあれば、様々な情報が入って来ても、その情報を整理して考える手助けになっていきます。情報同士がどうインターアクションをしていくのか、ということも考えていかなければなりません。7つのシステムというのは、標準のやり方でインターアクションをしていく形になります。そのフレームワークを持つことによって、色々な情報、考え方を整理することができますし、何かを見過ごしてしまう、忘れてしまうということを回避することにもつながります。

例えば、何らかのソリューションにフォーカスをしてしまうと、問題の解決だけではできただけでも、それが持続できない、持続するためのシステムに制約があるために、

その解決が長続きしないということが起こります。そういった見過ごされてしまうものを、無くしていくということにも使えるフレームワークなのです。

松本: 実際にシステムの設計で使われた例は、あるのでしょうか。

マーティン: まだありません。リサーチの早い段階にあるものなので、実践的な方法論として確立されるころまではまだ至っていません。

松本: 実際に、先ほどの会員企業などで、こうした考え方に共鳴されているという方もあるのでしょうか。

マーティン: 企業ではまだありません。使われているのは学術の世界です。システムズエンジニアリングを教える際に使われています。システムの思考ということで、カリキュラムに入れている大学が幾つかあります。

松本: 実際に産業界に広げていくために、こういったところが重要になってくるのでしょうか。たとえ研究的なものでも、実際に産業界で使われないと効果が出てこないということがあると思います。どうやって実際の現場に適応していくか、お考えがありますか。

マーティン: まずは学術界でこの考え方に沿った方法論を策定する必要があると思います。または SysML を、この枠組みに合わせて広げていくということを行っていくことが、実際に産業界で採用されていくためには必要でしょう。SysML を「7SAMURAI」に適応させていくためにどうすればいいのか。こういったところには、リサーチグループや研究のプログラムが必要になると思います。従って、まず方法論やツールを策定していく。現在私が考えている方法論は、一般的に使うためのものではありません。企業なら企業が、自社で使われているシステムなどに適合させていく、ということが求められると思います。

松本: 「7SAMURAI」の考え方は、非常にすぐれたものだという印象を受けています。7つという視点が適切かどうかは、私にはよくわからないのですが、それ以外にも、既存のシステムとの協業も考えていかなければいけないと思います。いずれにしても、単に一つのソリューションを考えるのではなくて、色々な視点を持ってシステムを考えていかないと、きちんとしたシステムができないというのは、非常に良く整備されたアイデアだと思います。

一方で産業界に実際に広めていくためには、ツールや仕掛けがないと、なかなか難しいのかも知れないですね。単にコンセプトだけで、これでやりなさい、と言っても

産業界ではすぐに対応できない。具体的に SysML で書けるとか、色々なツールがそろってくると、徐々にそういう考え方が浸透していくのかも知れないと思います。

マーティン：おっしゃる通り、ある意味では科学的理論のようなものだと思います。エンジニアは、科学的理論を直接適用するということはしません。それをいったん、方法論や手法、またはツールという形に変換し、変換した上で採用する、というものになると思います。今の段階は、システムサイエンスという概念であり、それをシステムズエンジニアリングのツール、またはメソッドに変換して、初めて採用という形になっていくのだと思います。

松本：ところで一つお聞きしたかったのは、なぜ「七人の侍」という名前を付けられたのか。その動機はどの辺にあるのでしょうか。

マーティン：最初に、そのシステム単体だけを考えていては足りない、ということを感じました。他のシステムを全体的に見ていかなければならない、と考えを進めていったところ、7つのシステムを考えなければならぬという考えに至り、7という数字が出てきました。私の好きな映画監督が黒澤明だったので映画の「七人の侍」が浮かんで来ました。そのストーリーが、ちょうど合うと思ったわけです。

ストーリーはご存知だと思いますが、個人個人の侍は、もちろん強いわけですが、それだけではなく、彼らが協力し力を合わせることで、更に力を増していく、というものです。システム同士のシナジーという考え方、またシステムの全体というのは、システムを構成するパーツの総和よりも大きなものになるのだ、という考え方にちょうど合う映画だと思いましたので、映画タイトルを使いました。一人一人では問題が解決できない。しかしながら、それが手を携え、協力することによって、全体として問題が解決できる、コラボレーションをする、インターアクションをする。それによって、適切な振る舞いになっていくのだという、全体的なシステムの考え方に合うということです。

エンタープライズ・システムズエンジニアリングとは

松本：マーティンさんが主張されているもうひとつの新しい考え方で、エンタープライズ・システムズエンジニアリングという概念があります。ここで言われているエンタープライズというのは、基本的にはビジネスだと思

えばいいのでしょうか。

マーティン：エンタープライズと言う時には、大きな、複雑なアクティビティを行うもの、という考え方をしています。そのためビジネスをイメージされることが多いと思いますが、ではすべてのビジネスがエンタープライズかといえば、そうではない。例えば、長きにわたって事業を行っているような企業、ルーティンで業務をこなし、構造はきちんとできている、色々なものがすべて予測できるという場合には、エンタープライズという考えではないですね。エンタープライズと言う場合には、色々な挑戦的で難しいことがあり、不確定要素や困難が多々あるということです。従って、例えば複数の企業が、一社ではできない困難なことを行うために協力をするという時に、その複数の全体を指してエンタープライズという言い方ができるかも知れません。

松本：そうすると、エンタープライズ・システムズエンジニアリングという考え方は、エンタープライズをいかにシステムとして構築するかと捉えればいいのでしょうか。例えば、最近で言うと、スマートコミュニティとかスマートシティのような幅広いコンセプトや概念を、いかにシステムとして構築するかということが、エンタープライズ・システムズエンジニアリングの一つだというように。

マーティン：2つあると思います。1つは大きなエンタープライズのアクティビティの中で使われる大きな技術的なシステムやソーシャルの部分の中にある技術的な部分。そして、もう1つは大きな複雑なエンタープライズが、技術とどういったインターアクションを持つのかということです。そういったところをエンジニアリングはすべて一緒に考えて行かなければなりません。

松本：そういうソーシャルな部分とかテクニカルな部分とか、色々な局面の視点から、システムを幅広く捉える、というのが、エンタープライズ・システムズエンジニアリングだと思えばいいわけですね。

マーティン：そうですね。社会の動き、ソーシャルのダイナミクスであるとか、人の動き、人間の力学であるとかダイナミクス、こういったものも全体の大きなアクティビティの一環となっていきます。個人個人が、どうシステムの一環として動いていくのか、というところだけではなく、システム間の連携など、大きな目標のためにコーディネーションを取っていかなければならないのだ、という考え方です。

松本：そうすると、やはりヒューマンの問題というのが、

一番大きなファクターになってくるような気がするのですが、先ほど人をどうやってモデル化するのかというのが非常に難しいというお話がありました。エンタープライズ・システムズエンジニアリングの中でも、人の要素をどのように位置付けていくかという点が、同じように難しい問題になるような気がします。

マーティン：そうです。更には、人を個人としてモデリングするだけでも十分ではない。人をグループとして見ていく。そのグループとの相互作用、グループ対グループという場合には、例えば協力をしたり、または競い合ったり、というような動き、力学が発生するという場合もあるので、そういうモデリングも必要になってきます。人の側面を考えるとという場合には、ソーシャルサイエンスもそうですし、ソーシャルなダイナミクス、それから行動学、それからビジネス。こういったすべてのモデルが必要になります。

松本：そういう意味で言うと、もう一つのキーワードはダイナミズムということになるのでしょうか。時々刻々変化していくものを、どう捉えていくか。つまり、今までのシステムズエンジニアリングは、どちらかと言うと静的なモデリングのようなイメージが強い気がするのです。そこにもう少しダイナミックなものを入れているところ、もう一つのポイントと言えるのでしょうか。

マーティン：人は動的なものです。予測しづらい。そして、人はコントロールされることを嫌います。ソフトウェアやハードウェアを考える際には、すべてコントロールをするということを考えます。ソフトウェアにしてもハードウェアにしても、その挙動、振る舞いに対してきちんとタイトにコントロールしていきますけれども、人はなかなかそれができるものではないですね。コントロールされることを好む反面、自由を欲しがりますし、個人の選択ができるということを求めます。そこで重要になるのが、経済学であり心理学であり、または社会生理学であるということだと思います。電子や複合化合物や、エネルギー変換といったことだけを考えればいいのではなく、ソーシャルダイナミクスまで考えて行かなければならない、ということです。

ITソリューションを部分として含むもの

松本：似たような考え方として、だいぶ前に、ザックマン・フレームワークをベースにしたエンタープライズ・アーキテクチャの考え方というのが出てきてたのですが、あれは、どちらかと言うと、エンタープライズと言っ

ても、もう少し企業に閉じたような狭い範囲で、マーティンさんが言われているエンタープライズ・システムズエンジニアリングのように幅広くはないと思いますが、考え方としては、それに近いでしょうか。

マーティン：似ていると思います。ただ、ザックマン・フレームワークに関しては、コンピューティングシステムを考えているだけのものではあったと言えらると思います。企業全体のITソリューションのアーキテクチャであるということですね。

しかし私が言っているのは、ITにとどまらず、より高いレベル、技術だけではなくソーシャルまたは組織、または人、こういったところを含めた全体のもので、もちろん、その中でエンタープライズレベルのITを使うこともありますけれども、それらの全体を指しているということです。

松本：そういう意味では、これをITにカスタマイズすることもできると考えられるのでしょうか。

マーティン：このエンタープライズ・システムズエンジニアリングとエンタープライズ・アーキテクチャを、一緒にITのために使っていくということは可能だと思います。ただし、このエンタープライズ・システムズエンジニアリングは、ITだけではない、というところは注意しておかなければならないと思います。

エンタープライズ・システムズエンジニアリングという言葉を知ると、どうしてもITだけに使うものと考えがちになってしまうんです。

松本：この考え方は、具体的にどこかに適用しているという話があるのでしょうか。

マーティン：INCOSEの中に、交通運輸を扱うトランスポート・ワーキンググループというものがあります。インテリジェント・トランスポートシステムという部分に、幅広いエンタープライズ・システムズエンジニアリングの考え方を適用しているという考え方があります。

松本：それは例えば、最近注目されている自動運転というようなものも含まれるのでしょうか。ああいうものは人間的な要素も非常に重要な部分を占めるので、社会活動全体で捉えていかないと、単に自動でブレーキをかけたり、運転したりという形だけでは非常に狭い範囲のソリューションにしかならないと指摘されています。そういうものも視野に入っているということでしょうか。

マーティン：インテリジェント・トランスポートシステムの中に、自動車は一部として含まれますが、そ

れだけではなく、自動車が全体的な交通システムと、どう一緒に機能するのか、全体的な社会インフラとどう連動するのか。例えば、都市における金融システムとどう関係するのかということまで含めて、統一したやり方で全体的な大きな目標を実現していくという考え方。これがインテリジェント・トランスポテーションシステムです。

目標の一つとしては、例えば、事故を減らす、全体的な交通システムの安全性を上げていく。またはスループットを上げていく。遅延を減らしていくということ、グローバルな都市全体として考えていくというものになります。

システムズエンジニアリングにおける安全性

松本：安全性のお話が出てきましたけれども、私たちもシステムの安全性をどうやって確保していくかということについて力を入れようとしています。エンタープライズ・システムズエンジニアリングで考える時に、全体の安全性というのは、どのように捉えればいいのか。

マーティン：エンタープライズにおける目標というものを、考えていかなければならないと思います。もちろん、エンタープライズにとって安全というのは、一つの関心事であるわけですが、例えば、成功の確率とバランスを取っていくということも必要になります。宇宙プログラムで、例えば宇宙船を火星に飛ばすというプロジェクトがあった場合、様々な安全の問題、課題というのが出てきます。ただ、それを本当に完璧にすべて、ということになると、その目的自体の実現可能性が無くなってしまふ。それでは目標が果たせないということになるので、そのバランスを取っていくということも必要になります。宇宙プログラムであった場合には、それは有人なのか無人なのか、ということによっても安全の捉え方というのは、変わってくると思います。また、原子力などに関しても、安全というものを考えていかなければなりません。

松本：安全性、そしてコスト面も当然重要な問題だと思いますし、利用者の立場から見ると、ユーザビリティや、色々な視点があると思います。そういった視点も、先ほどのエンタープライズ・システムズエンジニアリングの中に入ってくるのでしょうか。

マーティン：先ほどステークホルダの分析という話をしました。ユーザがステークホルダだと考えれば、ユーザ

は安全を求める。しかし、別のステークホルダであるオーナーは、利益追求をしたいということからコスト効率を高めることを求める。安全をきちんと実現することと相反してしまうかも知れない、そうしたこととの間でバランスを取っていくことが必要になります。

幅広いステークホルダがいれば、安全、ユーザの使い勝手、経済性、セキュリティ、という様々な目標があります。これらのバランスを取って、エンタープライズの中で両立させていくことが求められるということだと思います。

松本：そのバランスをどう取ればいいのか。色々なステークホルダがいる時に、誰を重要視するかというのは、ケースバイケースで違ってきますね。

マーティン：そのベストのバランスを見極めていくのは、アーキテクチャを決めていくところで行っていきます。アーキテクチャ作りの作業の一環の中で、誰がステークホルダなのかを理解し、どういう関心事、どういうプライオリティがステークホルダにあるのか、ということを理解していくわけです。そして、それに基づいたアーキテクチャを作っていくわけですが、そこでは様々な理論を借用して使っていくことができます。オペレーションの分析、ミッションの分析、またはビジネスの分析。こういったものが活用できると思います。

松本：なるほど。私たちも、安全性を考える上で、もっとシステム的に見ていかなければいけないと考えてきましたが、これまで私たちが考えていたシステムというのは、いわゆる IT システムといった少し狭い範囲であったかも知れません。マーティンさんがおっしゃるような、もっと幅広い社会システム的なものまで含めて捉えていかないと、全体が見えてこないということをおぼろげに感じました。

本日は、幅広いお話をありがとうございました。

