

“IT基盤人材”育成の重要性

—野村総合研究所の取り組み事例より—

システム開発において重要な人材としてITアーキテクトの役割が注目されている。特にシステム基盤に目配りできる人材の重要性が認識されはじめた。野村総合研究所（以下、NRI）でも、ITアーキテクチャの全体を理解できる人材の育成に力を入れている。本稿では、1つの事例としてNRIの情報技術本部における人材育成の取り組みを紹介する。

“IT基盤人材”育成に向けた課題

NRIは、人材育成の1つの目標として“ITアーキテクト”の育成を掲げている。ITアーキテクトとは、端末、ネットワーク、サーバーといった個別技術に精通し、さらに業務アプリケーションやビジネスとの関連を含めたITアーキテクチャの全体を理解できるIT人材のことである。建築で言えば、規格や部材や工法に通じ、また施主の希望を具体化すべく建物全体をデザインし図面に落とししていく建築士にあたる。

ITアーキテクトの重要性は、いまではIT業界で広く認識されるようになったが、一般企業においてもこのような高度な専門性をもった人材の重要性と、同時にそういう人材を育成する難しさが認識されはじめたのではないだろうか。ITアーキテクトの育成に必要なのは本人の資質だけではない。そのため、NRIは情報技術本部を中心に組織的に人材育成の改善・強化に取り組んでいる。

ITアーキテクトの守備範囲のなかでも、特にサーバーやネットワークなどの基盤領域は重要である。基盤領域は、アプリケーションなどの領域と比べて全体像の把握が難しい部

分である。

NRIでは、基盤領域に目配りができる“IT基盤人材”は、サーバーやネットワーク関連技術の研究開発とシステム適用のための“基礎体力”を高める上で重要と認識しており、人材育成の上でも特に力を入れている。

全体理解を難しくする“縦割り”業務

今日のシステム開発は、オープン化やWeb化の進展・成熟に伴って“縦割り”の業務分担で行われることが多くなっている。このやり方は、システムを構成する機器やソフトウェアが多くなってきたことに対応して、仕事を分担して効率化を図るという目的には合っている。ここ数年、システム開発量が増加し、同時にスピードが求められるようになっていなかでは、これは必要なことでもあった。

しかし、ITアーキテクトの育成を考えると、縦割りの業務分担はマイナスである。たとえば、Webシステムの担当者が、縦割りの業務分担のために一部の機器の動作環境維持しか担当していない場合、密接に関連しあう他のシステム構成部位について理解できないといったことが起こる。

ITアーキテクチャ全体を理解するためには



いわゆる“横ぐし”で物事を見ることが必要だが、そうしたことは縦割りの業務分担のために難しくなっているのが実情ではないだろうか。

“ブラックボックス”化の弊害

オープン化の初期には、クライアントサーバー型のシステムに用いるPCやサーバーなどのハードウェア、それらに搭載されたOS（基本ソフト）およびソフトウェアは、機能的に十分に成熟しているとは言えなかった。そのため、不足する機能を自分で作らなければならないケースも多かった。

自分で作る過程では、知らないことは勉強し、理解したことを試しながら前に進むことを余儀なくされる。だからこそ、システム全体の処理の流れや、個々の構成部位における処理構造を自然に体得することになる。

しかし、インターネット技術が発達し、Web型と呼ばれるシステム構成を採用することが多くなってきた現在では、ソフトウェア製品の組み合わせによってシステムに必要な機能が満たせるようになった。システム開発者は、それぞれの分野で日進月歩で改良されていく製品を使用することで最新のシステムを作ることができる。各構成部位の処理構造や、それに必要な動作環境を詳細に理解する必要は特になくなったのである。

しかしながら、このことはITアーキテクチャ全体を理解するという観点から見れば、ど

のように動作しているのか正確には理解できない“ブラックボックス”がシステムにいくつも存在することにほかならない。

次のようなシステム障害のニュースは、システムのブラックボックス化がどんな事態を引き起こすかを示すわかりやすい例と言えるだろう。

ある企業が、一般消費者向けに自宅のPCから利用できるサービスをインターネットを通じて提供していた。そのシステムが、突然、停止してしまった。さらに悪いことに、原因の究明に手間取って復旧に時間がかかってしまった。後になって、システムが停止した原因はプログラミングのミスではなく、システム負荷が開発時の想定を超えて増加したためであることがわかった。

このトラブルの原因と、復旧に時間がかかった背景として以下の点を考える必要がある。Webシステム上では、いくつかのネットワーク機器を通じて端末からの要求があり、それをWebアプリケーションサーバーと呼ばれるプログラムが実行する。多くの場合、その処理構造を意識しなくともWebシステムの開発は行うことができる。このトラブルでは、その処理構造がブラックボックス化しており、システムの全体を把握している技術者がいなかったため、端末からサーバーへの要求処理と結果処理がどこでどのように行われているのかを理解できなかった。そのため、システムの負荷が設計時の想定を超え、障害が発生

したときに迅速に対処できなかったということである。

身をもって考え体得する研修

技術に関しては、e-ラーニングなどで個人が学習することはできる。また、ベンダー企業が個別の製品や技術に関して有益な研修を提供している例も多い。しかしながら、個々の製品や技術をシステム設計の全体的な枠組みの中で“横ぐし”で把握する能力は、単なる技術知識の習得や、製品や技術の研修から得ることは困難である。

その上、1つの土俵の上で仕事が完結していたかつての大型汎用コンピュータの時代と違って、オープン化、Web化が浸透した現在のシステムは、より多くの製品によって構成されるようになったため、“横ぐし”で見なければならない製品や技術の対象ははるかに多くなっている。

このような状況に対応するため、NRIの情報技術本部では、サーバーやOSといった基盤要素からシステム環境を自分の手で構築することを前提とした研修環境と研修プログラムを開発した。

この研修プログラムは次のように5つのステップから構成され、研修を受ける者はこれを1人で行うようになっている。

①必要な機器を用意する

Webブラウザーが稼働する端末、およびプレゼンテーションロジック、ビジネスロジッ

ク、データベースを実装したサーバーを用意する。

②アプリケーションのための環境設定を行う

それぞれの機器へOS、ソフトウェアをインストールし必要な設定を行う。

③Webアプリケーションを開発する

Webアプリケーションを自分で実際に作ってみる。

④データベースを設定する

RDBMS（リレーショナルデータベース管理システム）製品をインストールし、アプリケーションで必要なテーブルの論理設計、物理設計、実データの生成などを行う。

⑤テスト項目を策定して実行する

いわゆる総合テストにあたる内容について、テスト項目設定、テスト準備、テスト実施、問題発生時の原因分析、対策検討を行う。

以上の研修の大きな特徴は2つある。

1つは、事前学習を必須にしている点である。たとえばLinux OSのセットアップ（インストールと設定）でも、手順どおりに行えば一応は完了するが、手順の意味や設定値の意味を理解しないまま作業を進めるとつまづくように仕掛けがしてある。事前学習をしていないと全日程の消化が難しくなるので、研修の修了を目指す者は必然的に事前学習をすることになる。

もう1つは、基本的に1人で研修メニューをこなせるように組み立てられている点である。研修を受講するまでは実際の開発現場で

特定の機器しか扱ったことがなくても、研修の過程では自分が経験していない分野の作業をいやでも1人で行わなければならない。システムを構成する個々の要素について、ハードウェアのセットアップからソフトウェア製品のセットアップまですべて1人で行ってみることによって、実際のアプリケーション処理を実現するための、隠されたさまざまな構造や環境が理解できるはずである。

テストの観点を重視

システム構築におけるテストには、プログラム単体レベルから、プログラム連結レベル、システム間の連結レベル、システム全体でのシステム運用を確認するレベル、システム全体性能を確認するレベルなど、多くの「テスト」と名の付くものがある。

ITアーキテクチャ全体を見渡すプロフェッショナルとしては、アプリケーション処理やシステム運用処理、それらの障害時の処理、システム性能の確保などについて、“横ぐし”の目線でどうあるべきか、そのためにはどういうテスト項目や実施方法、確認方法があるのかを考え実行する経験が不可欠である。

そのため本研修は、講師からシステムの動作テストを指導されるまでの間に、端末およびサーバーのハードウェア、OS、ソフトウェア製品、アプリケーションプログラム、データベースが線としてつながっていることを理解できるようにしてある。

研修の効果

研修後、受講者はアンケートに答えることになっている。それによると、座学や聴講ではなく、自分自身が頭と体を使って取り組む研修スタイルがよかったという声が多い。また、物事を“横ぐし”で見る力を養うことと、従来はブラックボックスになっていた構造を理解することについては、9割以上の受講者が有益と評価している。

テストを体験してみたことについては、隣の部位や関連部位のことを考えたシステム運用設計を怠ると、実運用に入ってからそのツケが回ってくることが体感できてよかったという感想が聞かれた。関連する部位のことを考えたシステム運用設計は、システム基盤技術者が関係者とシステム要件の擦り合わせを行う局面で、網羅性のある的確な提案を行うために重要な能力である。

これらの結果から、研修の目標は達成できていると考えられる。すなわち、特定の技術領域の専門家の立場からではなく、システム全体のデザインができるITアーキテクトとして、必要な着眼点や思考が身に付けられるということである。

今後は、受講者が後輩社員の講師として活動する機会を設け、教える側の立場からITアーキテクトとしての思考・行動の質をさらに高めていけるよう、研修内容をさらに拡充していく予定である。 ■