

9 これから世界を視野に入れて働くエンジニアへ



原田隆宏 | Advanced Micro Devices, Inc.

世界で通用するエンジニアになるまで

私は現在アメリカ合衆国のカリフォルニア州に住むエンジニアで、世界に散らばるエンジニアと日々プロジェクトを進めている。私が就職を機に日本を出てから15年ほどが経ち、その間に超優秀なエンジニアからもう一緒に仕事をしたくないようなエンジニアまで多くの方々と、さまざまなプロジェクトにかかわってきた（記憶に強く残っているのは高卒だけど数学等にめっぽう強いエンジニアで、彼によると数学は独学だとのこと）。実はその中でも日本人のエンジニアと一緒にした仕事はほとんどなく、ともに仕事をしてきた大半のエンジニアは私とは育った境遇がまったく違う世界各地出身のエンジニアたちである。そのような私の経験から今後世界で通用するエンジニアに必要な要素は何かということ、私なりに考察してみたいと思う。

軌跡

私は大学院で流体力学とそのコンピュータシミュレーションを学び研究していた。きちんとしたプログラミングの講義を受講したことのない私はそのころは独学で必要なことを学び、研究室で開発していたコードを元に研究をしていた。2000年半ば、グラフィクス向けのプロセッサである Graphics Processing Unit (GPU) を無理やり数値計算に使うという研究(?)に日々明け暮れていた。

その後、物理計算のミドルウェアを開発している Havok 社で働くチャンスを得て、本社のあるア

イルランドに飛んだ。そこで私を待ち構えていたのは当初から予測していた言葉の壁だけではなく、私に馴染みのない洗練された商用コード、多様なアルゴリズムについての議論であった。今までの私のプログラミングの世界は一研究者のものであり、ここで出会った商用コードの複雑性は想像以上であった。大学での研究では大体その研究に定番なアルゴリズムとデータ構造が確立されており、ほとんどはそれらを使い回すことになるが、ここで議論に出てくるアルゴリズムは幅広いものであった。

まったく新しい環境で、色々な意味での挑戦であったが、知識については時間の許す限り、書籍からインターネット、論文まで手に入る文献は片っ端から読み漁り、実装して知識の定着をさせていった（アイルランドは緯度が高く雨が多いため勉強には最適であった）。追いつくというところから走り出し、新しいものを生み出せるところまで成長することができ、徐々により先進的な研究開発に携われるようになった。また対象とするプロセッサも汎用 CPU だけでなくセルプロセッサや Intel 社の開発した Larrabee のプロトタイプでの開発にも携わることができ、非常に貴重な経験をさせてもらったと思っている。このときにプロセッサアーキテクチャを強く意識したソフトウェアを書いていたため、ハードウェアの低レベルな知識が身につき今でも最適化などを考えるときに重宝している。開発言語としては私がメインで使っている C++ での開発であったが、私はもちろんこれらのプロセッサで開発した経験がなかったので、それらの特性等を一から学び、最適化されたコードを書くというところまで

なるべく早く持っていかなければならなかった。学習に数カ月かけている暇はなかった。オフィスはアイルランドにあったが、このときにともに働いていた同僚はアイルランド人だけでなく、ヨーロッパ各国から来ているエンジニアたちであった（なので皆英語の訛りが強くて苦労した）。

2010年の夏頃、私は Advanced Micro Devices, Inc. に転職をすると同時にアメリカ合衆国に渡った。加入したチームは製品リリースに向けてデモを作るチームであり、ここで私は再度 GPU を仕事で使うことになった。しかし開発に使うグラフィックス Application Programming Interface (API) は DirectX11 と、OpenCL であり、新しいプログラミング言語を学び GPU を使った物理計算の研究開発をした後、リアルタイムレンダリングの研究開発に携わった。リアルタイムレンダリングは昔から趣味の程度でコードを書いていたが、趣味のレベルからプロのレベルまで知識を引き上げる必要があった。リアルタイムレンダリングの分野で新規手法を開発した後、さらにリアルなレンダリングを求めて、ほとんど経験と知識のなかったレイトレーシングを用いた大域照明の実験をし始め、数年かけて、今では Radeon ProRender¹⁾ というエンジンとしてチームで開発しているプロジェクトにまで成長させた (図-1)。

今のプロジェクトにかかわっているエンジニアは、アメリカ合衆国内のさまざまな都市だけでなく、カ



■ 図-1 Maxon Cinema4D に搭載されている AMD Radeon ProRender を使って作成された写実的な画像例

ナダ、イギリス、ドイツ、フランス、ロシア、ウクライナ、インド、韓国、日本等にいるので日々彼らと連絡を取り合い、開発を進めている (図-2)。実際、期日までに成果さえ出せばいいので物理的にどこにいるかはあまり問題ではなく、正直なところどこで仕事をしているかよく知らないエンジニアも多い。

プログラミングについて

私はこのように日本の大学での研究者としてキャリアをスタートさせたが、すぐに産業界に舵を切りヨーロッパの企業で数年研究開発を行い、今はアメリカの企業でエンジニアとして研究開発を行っている。エンジニアとして働いていくためにプログラミングの技術とプログラミング的思考が重要なのは当然であるが、プログラミングを体系的に学ぶ機会がなくても落胆することはない。私はそういう機会に恵まれなかったが、独学でどうにかここまで来られている（もちろん私は凄腕プログラマではまったくないので、そうなりたい方には私の方法はお勧めしない）。

また時と場合によってプログラミングに求められるものは異なる。研究においてはプロトタイプをいくつも作り試行錯誤することが多いので私はチームで開発、メンテナンスしやすいコードのデザインよりも完成までの速度を重視する。研究でも、将来的にその研究から発展させて大規模プロジェクトとな



■ 図-2 普段離れて仕事をしている同僚たちが集まったとても希な機会であった、Siggraph 2017 での写真。以前から一緒に仕事をしていてもここで初めて会ったエンジニアもいた。左から3番目が私

る場合は、ライブラリとして再利用できるパーツをきちんと作っておいた方が高いスループットを出せることも多い。また商用コードのようにチームで開発するときは、ある程度将来の拡張も考慮に入れたインタフェースと仕様を定義し、実装はそれぞれのエンジニアもしくはチームに任せるといようにするのが一般的であろう。実際に我々も、実装を取り替えたり拡張しながら、私が2011年に書き始めたコードを今でも使用している。

現在でも私は開発速度重視の試験的プログラミングとデザイン重視のライフスパンの長いコードへのプログラミングをプロジェクトによって使い分けている。

世界で通用するエンジニアに必要な要素

単にプログラミングができる人ならば世界中にいくらでもおり、単純なプログラムを書く作業は安い人件費の国に徐々に移ってきている。実際、私もプログラミングは自分のアイデアを形にする手段だとはか思っていない。

それではそのような労働力としてのプログラマとは一味違う、「エンジニア」として世界で通用する人材になるために必要な要素は何かを、多くのエンジニアと仕事をしてきた私の経験を通して考えてみた結果、それらは以下の3点だと思う。

変化を受け入れ楽しみながら最短ルートで習得する

前述したように私の仕事の内容は過去20年ほどの間に大きく変わってきた。大学院での流体シミュレーションから Havok 社でのさまざまな物理計算、そして今取り組んでいるレイトレーシング。このような経験から新しい課題に出くわしても怖じけずチャレンジしていく精神が大切だと思う。もちろん私も転機が訪れたときに挑戦を避け、今までやって

きたことを続けるという選択をすることも可能だった。もちろん人生は一度しかないのでもうしていたらどうなっていたかは分からないが、色々なことにチャレンジすることでより多くの素晴らしいエンジニアの方々に出会い刺激を受け、自分をより高められたと思う。また新しいチャレンジが新しい機会を創出する。しかし新しい課題や分野に取り組んでもそれを習得するのに時間がかかりすぎるとは価値がなくなってしまうので、なるべく速く飲み込み自分のものにできるようにしたい。つまり新しい課題に果敢にチャレンジできるメンタリティと最速で課題を解決する方法を見出せる思考力と実行力を兼ね備えているのが望ましいと思う。

ではどうしたら速く新しい分野や課題を習得できるであろうか。私はその新しい課題を楽しみどっぷり浸かってしまうことではないかと考える。さらに自分で危機感を作るのもよいと思う。私の場合は勤めていたのが日本企業ではなかったため、成果が出なければすぐゲームオーバー、日本に強制送還されてしまうという危機感がいつもあった。

軸を持つ

今述べたように私は色々なプロジェクトにかかわってきて、自分の大学での専門とは一見まったく違うようなことにも携わってきた。それらは表面的にはまったく違うように見えるが、実は根本のところでは多少共通点がある。今私が取り組んでいるレイトレーシングも根本的には私が大学院のときに専攻した物理シミュレーションの一種と言ってもよく、流体の物理が光学物理に変わっただけであると言えるかもしれない。ほかに携わったプロジェクトでも基本的に数学、アルゴリズム、データ構造を中心に体系化されていることが大半なので、これらへの理解を一度深めておけばある程度は新しいチャレンジにも対応していけるように思える。つまり、世界で渡り合えるエンジニアになるには、ほかの人には負けない専門分野もしくは技術を1つしっかり究める

ことを勧める。

また英語の議論もまともにできないまま日本を出て思ったのは、自分の意見、アイデアがあれば相手はきちんと耳を貸してくれるということである。もちろん私が心の広い方々の中で仕事をさせてもらったのかもしれないが、この経験から自分ならではの考えを主張できる軸を一本持つとよいと思った次第である。私の場合は物理シミュレーションが私の軸になり、言葉で表現しきれないことはコードを書きながら説得させることもできた。

国境を超える

世界で通用するエンジニアになるには、まずは情報收拾をするにあたり日本語という壁を乗り越えられるかどうか重要なポイントであると思う。統計によると日本語のインターネットの情報量は3.4%にすぎないが、英語の情報量は2019年で54%である²⁾。世界中のエンジニアが英語で情報收拾している中で、日本語で情報收拾してはまったく勝ち目がない。先ほども述べたが、速く習得する必要があるので、情報はあればあるほど良い。また専門書籍の多くは英語で出版されるので英語の書籍からでも情報を收拾できるようになっておきたい。

私が一緒に仕事をしてきた同僚のいる場所は徐々に拡大していった。大学院のときは大学の中、Havok社のときはヨーロッパ各地から来たエンジニアと同じオフィスで働いていた。そして今の私のプロジェクトのチームメンバの国籍の数もさらに増え、働いている場所も世界各国色々な国にいる。私のチームが柔軟なワークスタイルに特に寛容なのかもしれないが、このようなエンジニアのグローバル化の傾向が起こっているのは確かだと思う。このように世界各国からのさまざまな母国語を持つ人々が英語でコミュニケーションをとって日々仕事をしている。その輪に入るにはもちろん自分も英語でコミュニケーションをとらなければいけない。日本人の英語のコミュニケーション力が平均して低いのは

確かだが、不完全な英語だからといって日本から出ていかなければ何も始まらない。実際私のチームにいるエンジニアもほとんどが英語を母国語としないエンジニアであり、はっきりいって正確な英語でなければならぬということとはまったくなく、自分の主張を自分なりの英語でズケズケと言えるくらいがよいと思う。

また今の私のチームのように優秀なエンジニアであればどこにいても採用しようという企業が増えているのは確かである。プログラミングができ、英語で自分の意見をきちんと言え、自分の強みとなる軸を持っていれば、今でも世界でやっていけるエンジニアになれるチャンスが多くあるが、今後はさらに増えていくと思う。

ともに仕事をしたいと思われるエンジニアとは

最後に今自分はどのようなエンジニアと仕事をしたいかということを考えてみた。もちろんここで述べた3点が備わっており、さらに言うと何事でも精力的に取り組め、課題をやり抜く力があり、解のない問題にも最善を尽くし自分なりの解を導き出すことのできるエンジニアと私は仕事がしたい。プログラミングの技術だけでなく、仕事に向き合う姿勢がチームの成功を大きく左右するからだ。

参考文献

- 1) AMD Radeon ProRender, <https://www.amd.com/en/technologies/radeon-prorender>
- 2) Historical Trends in the Usage of Content Languages for Websites, https://w3techs.com/technologies/history_overview/content_language

(2019年2月11日受付)

■原田隆宏 takahiroharada@gmail.com

神奈川県生まれ。東京大学工学部卒業。東京大学大学院で修士号、博士号取得。東京大学大学院で助教として研究を続けたあと、Havok入社と同時にアイルランドに渡り、物理シミュレーションの研究開発に携わる。その後AMDのデモチームに加わり2010年に渡米する。現在はAMDにてPrincipal Member of Technical StaffとしてRadeon ProRenderという大域照明計算ソフトウェアの開発チームリードを務めている。趣味はスキー、自転車、旅、工作。