



小特集

# プログラミング・ エクスペリエンスの 新潮流

—言語設計から産業応用まで—

## 編集にあたって

+++++ 加藤 淳 (産業技術総合研究所)  
+++++ 増原英彦 (東京工業大学)

今、プログラミングが面白い。スマートフォンなどはもちろんのこと、車、家電製品、家、ロボット、私たちの生活を取り巻くあらゆるものにコンピュータが内蔵され、プログラムで駆動される時代だ。プログラミングできれば、これらを思い通りに動かせる。たった数十年前はキーボードを叩いてディスプレイに文字や絵が出るだけで心が踊っていたのを考えると、可能性

の広がりは無敵大にすら思える。その一方でプログラムを書くやり方はこの数十年、大きく変わることはなかったように思える。最近になって、ようやくエディタやデバッガなども含めたプログラミング環境の研究開発が盛り上がってきている。特徴として、何か画期的な技術が生まれたのではなく、色々な分野で同時多発的に、しかし確実に、プログラマ目線で使い勝手の改

善を試みる新潮流が生まれていることが挙げられる。

本特集では、これをユーザ体験（ユーザエクスペリエンス，UX）のプログラマ版として「プログラミング体験（プログラミングエクスペリエンス，PX）」と呼び、各界の第一人者の寄稿記事を通して状況を俯瞰することを狙う。具体的には、(1) PX が流行る端緒の1つとなった「ライブプログラミング」の第一人者、(2) スタートアップ企業の専門家、(3) ヒューマンコンピュータインタラクションの研究者、(4) IoT 時代の日曜大工、メイカー文化の発信者、(5) プログラミング言語の研究者、(6) ソフトウェア工学の研究者、(7) ソフトウェア開発におけるヒューマンファクタの研究者の方々に各章を担当いただいている。以降は、各特集記事を紹介した上で、補遺として「なぜ今 PX なのか?」「国内外の研究動向」を紹介する。

まず、「1. ライブプログラミングによる滑らかなプログラミング体験」では、プログラミング言語の研究分野で近年注目を集めている「ライブプログラミング」という技術を紹介いただく。プログラミング言語の研究はコンピュータ科学全体では基礎分野に位置すると思われるだろう。しかしプログラミング言語の設計では「プログラマにとって書きやすい、分かりやすい」という応用面から見た性質が重要視されている。本稿ではそれが色濃くあらわれていて、どうすれば滑らかなプログラミング体験を実現できるか?という問いに対する答えが、分かりやすい例えも交えながら明快に論じられている。

次に、「2. スタートアップに見るプログラミング体験の新潮流」では、プログラミングを効率化し、プログラマが扱える範囲を拡大するような新技術を研究開発するスタートアップ企業を数多く紹介いただく。コンピュータは、言うまでもないことだが、企業によって作られている。自然科学の大部分は人間がいなくても起きている自然現象を対象とする学問だが、コンピュータ科学は、そもそもの始まりから人が人のために作った道具「コンピュータ」を扱う学問である。したがって産業界とアカデミアは共生関係にあり、産業界の発展が先じて学問が置いていかれることも多々ある。情報技術系のスタートアップ界隈は特に流れが速く、またその中心にいる人々がプログラマであるた

めに、プログラミング体験の新潮流を知るためのサンプルソースとしてかなり適していると言えよう。プログラマの読者の方なら、日々の業務で使えるツールが見つかるかもしれない。

「3. ヒューマンコンピュータインタラクションとしてのプログラミング」では、同研究分野において、プログラミングをもっと多くの人ができるように、より分かりやすく取り組めるように、と研究されてきた内容を紹介いただく。かつてのコンピュータは、そもそもプログラミングできない人には使えないものだった。その後、マウスが発明され、ウィンドウ・アイコン・メニュー・ポインタというメタファが確立し、グラフィカルユーザインタフェースができあがると、プログラミングしなくても直接操作でコンピュータのさまざまな機能呼び出せるようになった。そこで初めて、プログラマでないコンピュータのユーザ、すなわち「エンドユーザ」が登場した。ヒューマンコンピュータインタラクション (HCI) はそれに呼応して 1980 年代に生まれた研究分野であり、このとき分野形成に大きな役割を果たした人々には、プログラミング言語やソフトウェア工学などの研究者がいた。HCI ではその後も脈々とプログラミング体験を改善する研究が続けられており、本稿では将来像までかなり具体的に予見している。HCI 研究者はもちろん、それ以外の人々にもぜひ一読していただきたい。

続く「4. 家電とおうちのプログラミング」では、今日からすぐに始められる、家電製品や家での生活に関するプログラミング事例を紹介いただく。日曜大工といえばどんな家庭でも多少なりともできるもので、ドライバーやトンカチといった工具がない家のほうが珍しい。一方で、情報技術を活用した日曜大工はいまだ普及しているとは言えず、プログラマからすれば「どうしてこんなこと手動でやらないといけないんだ? 自動化したい」と思うことも多々あるものである。本稿では、Internet of Things (IoT) 技術の発展に伴ってそうした状況が変わりつつあることを感じられるだろう。IoT の普及期には、どうやったら簡単に生活を便利にできるのかというノウハウを草の根で共有することが必要であり、本稿の中で紹介されている「おうちハック同好会」のようなコミュニティ、より広くは「メイカー文化」の

担い手たちが重要な役割を果たすに違いない。

さて、たいていのプログラミング言語は本質的には汎用に使い得る、というのは、コンピュータ科学でチューリング完全の概念を学んだ者なら誰もが知っていることだが、一方で実践的には、言語設計により得手不得手があるのが当たり前である。「5. ドメイン専用言語と言語指向プログラミング」では、特定の目的のために設計されたプログラミング言語（ドメイン専用言語、DSL）が持つ効用について紹介いただく。ここまでならプログラミングするときに目的に応じて使う言語を選ぶ（Web ページを作るなら HTML、データベース操作には SQL など）という至極当たり前の話だが、門外漢にとって面白いのは、目的に応じてプログラミング言語を作ってしまう開発スタイル（言語指向プログラミング）が存在し、そうしたスタイルを支援する「プログラミング言語の開発環境」まで用意されているということである。より良いプログラミング体験を実現するためには、言語だけでなく周辺ツールやデータの可視化手法なども目的に応じて動的に組み上げられる「プログラミング環境の開発環境」が必要となるだろう。そのような未来を予見させる一稿である。

「6. 初心者にやさしいコンパイラ警告メッセージを目指して」では、言語そのものでなく、言語を受け取って機械語に翻訳するコンパイラを改善するソフトウェア工学の研究事例を紹介いただく。より良いプログラミング体験を実現するためには、コンパイラやエディタ、デバッガなどを綿密に連携させる必要があることがお分かりいただけるだろう。最後に「7. 生体情報を用いたプログラミング活動の測定」では、同じソフトウェア工学の中でも、個々のプログラマの状態を生体センサで計測する手法を紹介いただく。プログラミング体験の研究においては、それが本当に良い体験かどうか評価する実験が必要となる。プログラミングの成果物であるソースコードを分析するだけ（たとえばソースコードの行数が短くなった、など）では不十分な場合、被験者の情報を何らかの方法で取得しなければならない。そこで、インタビューのように主観が混じらない生体情報を活用する試みは興味深い。なお、ソフトウェア工学の研究全般に興味のある読者

は、会誌 Vol.58, No.8 (2017 年 8 月号) の特集「ソフトウェア工学の最前線」も参照されたい。

## なぜ今、PX か？

プログラミング体験 (PX) に関する研究開発がさまざまな人の手で行われていることが、ここまでの章立て紹介でお分かりいただけたと思う。それではなぜ今、PX が盛り上がっているのだろうか？ 背景には、技術の進歩と各研究分野の事情が考えられる。

### ❖ 情報技術の進歩

冒頭でも述べたように、あらゆるものにコンピュータが内蔵されるようになったことと PX の間には深い関係がある。かつてのプログラミングでは、抽象的で効率的なロジックを綺麗に間違いなく組み上げることが重要であった。そのために必要な情報はたいてい紙の上やコンピュータの画面内に収まるもので、プログラマは机に向かって黙々と作業を続けていればよかった。

一方で、今のコンピュータはさまざまなセンサ・アクチュエータが接続されたり、インターネットに接続されたりして、大量のデータが連続的にプログラムへ流れ込むのが当たり前である。データ処理用のライブラリは、たとえば画像処理用のライブラリ OpenCV のように肥大化し、使い方をすべて把握するのは容易ではない。また、機械学習を用いたプログラムは、正しく動作させるためにまず具体的なデータを収集して前処理する必要がある。プログラムに与えられる入力データは複雑かつ一過性で再現できないことが多く、デバッグが難しい。機器の設置場所によって動作が著しく変わることもあって、プログラマが事前にすべてのパラメータチューニングを行えるとも限らない。

このように、プログラミングのあらゆるフェーズにおいて従来の開発手法が通用しないことが増えてきている。一方でプログラミングに関する要素技術の研究開発も相応に進んできた。たとえば、統合開発環境のオープンソース化などによって、新しい PX を提供するための実装コストが著しく下がった。問題が複雑化しているが、解決に使えるツールも増えているの

が現状なのである。こうした文脈を踏まえたプログラミング環境の研究については、加藤らの『多量のデータを利用するユーザインタフェース開発における例示を伴うプログラミング手法』<sup>1)</sup>が詳しい。

### ❖ 各研究分野の事情

先述したように、かつてコンピュータのユーザは全員プログラマであり、1980年代半ばまでHCIという研究分野は存在しなかった。グラフィカルユーザインタフェースの登場とともに、プログラミングに関する研究は、プログラムを作る立場（プログラミング言語やソフトウェア工学）と、利用する立場（HCI）の研究へと分化していった。その後各々の観点で研究が進展し、研究対象が広がりを見せるにつれ、改めてオーバーラップが大きくなってきたのが今だと考えられる。

たとえばプログラミング言語分野では、「ライブプログラミング」のように、言語設計だけでなくデバッグなどの改善も含めた広いスコープの研究が盛んになってきた。また、HCI分野では、「ツールキット」の研究（特定のプログラムを作りやすくするドメイン専用言語の一種）をしていた人々が、ツールキットだけでなくプログラム開発のワークフロー全体を支援するための統合開発環境（IDE）の研究に取り組むようになった。ソフトウェア工学では、初学者向けのプログラミング教育への関心の高まりなどと相まって、プログラマの探索的な開発フローを支援する研究が増えている。なお、プログラミング教育もPXが重要なトピックではあるが、これについては会誌の過去の特集Vol.57, No.4（2016年4月号）「プログラミング入門をどうするか」やVol.58, No.10（2017年10月号）「情報教育とワークショップ」に譲り本特集では扱わない。

プログラミングをより良いものにしていこうという大きな課題に対して、別々の分野で独立して取り組んでいるのは非効率（知見が共有されない、など）で、学術的にも不誠実（引用されるべき文献が引用されない、など）なのかもしれない。

## 国内外の研究動向

プログラミング体験（PX）に関する研究は、さまざまな研究分野で散発的に行われてきた。たとえば各分野のトップ国際会議（HCIならACM CHI, UIST, プログラミング言語ならOOPSLA, ECOOP, PLDI, ソフトウェア工学ならICSE）で、関連論文が発表されている。また、ビジュアルプログラミング言語に関する国際シンポジウムだったものが、HCI寄りのプログラミング研究を取り込んでできた国際会議VL/HCCや、プログラミングとソフトウェア工学周辺で生じる心理学関連のトピックを扱う歴史の古いワークショップPPIGなどでも、PX関連の発表がある。新興国際会議だと、即興でプログラミングして音楽や動画を生成するメディア芸術の表現手法「ライブコーディング」に関する国際会議ICLCや、本稿でも紹介した「ライブプログラミング」に関するワークショップLIVE、その名の通りプログラミング体験全般に関するワークショップPXなどがある。また会誌Vol.58, No.9（2017年9月号）でも報告された国際会議<Programming>にも関連論文が見られる。

こうした国際動向を全部追うのは大変であるため、日本国内では、本特集のエディタが発起人となっているSIGPXという勉強会が半年に一度程度のペースで開催されている。基本的には関東圏での開催だが、Skype参加なども可能である。また、SIGPXは、PXに関するツールや統合開発環境などの開発に携わる研究者やソフトウェアエンジニア、PXに関する認知心理学的な知見や興味を持つ人々が集まる稀有な機会ともなっている。SIGPXのWebサイト上には、本特集に関する特設ページ<sup>2)</sup>が用意されており、分野をまたいで収集したPX関連の研究を行っている人や研究室の情報などにアクセスできる。本特集と併せて、読者のみなさまにプログラミング・エクスペリエンスの新潮流について知っていただければ幸いである。

### 参考文献

- 1) Kato, J. Igarashi, T. and Goto, M. : Programming with Examples to Develop Data-Intensive User Interfaces, IEEE Computer 49 (7), pp.34-42 (July 2016). <https://junkato.jp/programming-with-examples/>
- 2) SIGPX 情報処理学会誌特集「プログラミング・エクスペリエンスの新潮流」特設ページ, <https://sigpx.org/ipsj2017> (2017年9月7日)