

2010年9月25日～10月2日
ボルチモア(メリーランド州, USA)

稲葉一浩 国立情報学研究所

ICFP とは

ICFP (ACM SIGPLAN International Conference on Functional Programming) は、関数型プログラミング言語に関する国際会議である。1996年に前身であるLFP (LISP and Functional Programming) とFPCA (Functional Programming Languages and Computer Architecture) の2つの会議が統合されて以来毎年開催され、今回で第15回目となる。

この会議はこれまで偶数年は北米、奇数年は欧州を交互に開催地としており、2010年の今回は、米国メリーランド州のボルチモア市が舞台となった。実は恥ずかしながら、私には今回訪れるまで馴染みのない街であったが、不滅のメジャーリーガー“Babe” Ruthの生誕の地であり、また、合衆国国旗および国歌である『星条旗』もこの地で生まれたという、歴史ある都市だそうだ。

会議名になっている関数型プログラミングとは何か、と厳密に語り始めると百人百様の答えが得られそうな議論の尽きない概念ではある。おおまかには、「関数を引数として受け取る、関数を生成して返すなど、関数を実行時の値として積極的に扱う」「可能な限り、変数や外部状態の書き換えを行わない、純粋に数学的な意味での“関数”を中心に記述」といったプログラミングスタイルのことを指す。ICFPのトピックとしては、当初から関数型を指向して設計されたプログラミング言語であるMLやHaskell、あるいはその基礎理論から発展した研究発表が主となっている。しかし現在では、JavaScriptやPythonなどのスクリプト言語はもちろんのこと、



写真1 ボルチモア港の夕暮れ

C++ や Java といった言語でも、式中で新しく関数を定義する無名関数式の言語仕様への導入が検討されるなど関数型スタイルのプログラミングは、非常に一般的なものとなっている。これら主流言語の動向を占う上でも、重要な会議である。

併設ワークショップ

ICFPの本会議は、全日程の中心の3日間、9月27日から29日にかけてシングルトラックで開催された。その前後に、9つのワークショップが併設されている。うち4つがML, Haskell, Erlangの各関数型言語にテーマを絞ったワークショップおよび、Haskellの特に実装面を語らうHaskell Implementers Workshopである(LispやScheme関係のワークショップが併設される年もあるが、今年は開催されなかった)。ほかに、高水準の並列プログラミングに関するHLPP (High-Level Parallel Programming and Applications)、総称的プログラミングを議論する



写真2 発表会場の様子

WGP (Workshop on Generic Programming), より数学的なトピックの MSFP (Mathematically Structured Functional Programming), 関数型プログラミングの理論の発展により言語の性質の定理証明を自動化あるいは支援する研究に関する WMM (Workshop on Mechanizing Metatheory), そして関数型プログラミングの商業利用を議論する会議 CUPF (Commercial Users of Functional Programming) が開催された。

CUPF では Intel 社や Twitter 社などの内部での関数型プログラミング言語の使用ケースが紹介された。企業での利用に関する報告は ICFP 本会議でも Google 社のサーバ仮想化システム Ganeti や, XenSource 社の同じくサーバ仮想化技術 Xen など著名な企業での使用例が発表されており, 勢いが感じられる。

招待講演

ICFP 本会議は, 関数型言語界の巨人であり, 2010 年 3 月に逝去した Robin Milner 博士の追悼講演で幕を開けた。博士は定理証明支援系 LCF や関数型言語 ML, さらにその型システムの理論, 加えて並行計算の数学的モデル「 π 計算」などの創始者として知られ, まさに, ICFP の関連する分野の礎を築いた研究者の 1 人だ。

講演では, Milner 博士の高校時代から始まり, ML 言語が誕生するまでの歴史を振り返りながら, なぜ

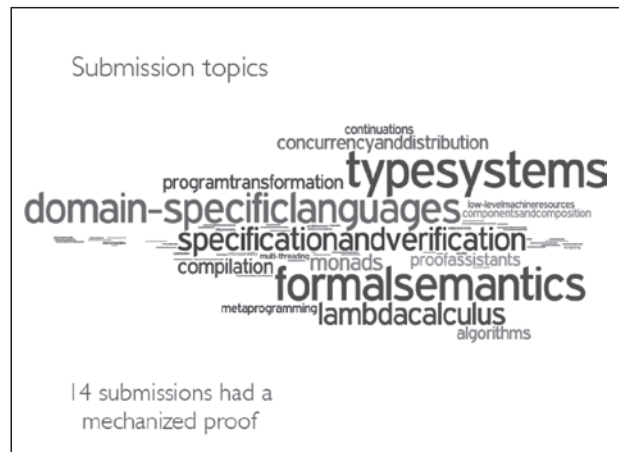


図-1 投稿論文のトピック (プログラムチェアの報告スライドより)

ML がいまあるような言語設計となったかを紐解いてゆくエキサイティングなものであった。特に, 証明支援系と呼ばれる, 厳密に公理系にのっとった形式的な証明を計算機プログラムとして書き下ろすことの支援・半自動化のためのシステムのスキプトの記述言語として始まった, という由来が隔々に影響していることが感じられた。

以下で述べるように, 証明支援系それ自体の研究や, 証明支援系を用いて完全に機械検証された証明つきで研究を進めるスタイルは今再び活発化してきているテーマでもある。温故知新を感じるころである。

中心的なトピック

会議全体の動向としては, 2 日目の最後にプログラムチェアからいわゆる「タグクラウド」の形にまとめた一目で分かりやすい紹介があった。許可を得てここに引用させていただく(図-1)。型システム (type system) や仕様と検証 (specification and verification), 形式的意味論 (formal semantics) といった, 「プログラムの正しさを保証する」ことを目的としたさまざまな研究テーマが目を引く。今回特に私の印象に残った発表は, Yale 大学の Antonis Stampoulis 氏, Zhong Shao 氏による共同研究 “VeriML: Typed Computation of Logical Terms inside a Language with Effects” であった。証明支援言語の研究の中でも, 証明言語そのものの記述力



写真3 コーヒーブレイクの様子

を強化していくよりも、その記述を証明プログラムが対話的に構築するための補助となるスクリプトの柔軟性の向上を考えるべき、との提言とともに発表された新しい関数型言語で、非常に今後の発展が気になる研究である。

10年前の最重要論文

多くの会議で同様の賞が設けられているが、ICFPでも、10年前のICFP 2000で発表された論文のうち、“The Most Influential Paper”，すなわち、以降10年間に後続の研究に最も影響のあったものが表彰される。

今回は、Koen Claessen氏とJohn Hughes氏の共著論文“QuickCheck: A Lightweight Tool for Random Testing of Haskell Programs”が受賞した。これはソフトウェアの正当性検証に関する「テスト」と「自動検証」の間のトワイライトゾーン、と銘打たれた研究だ。具体例を使って検査するため低コストだが網羅性に欠けるテストと、論理式などで形式的に表現した仕様を自動検証で保証する、網羅性は高いが実現コストの高い手法。この2つの間に立ち、ランダムなテストケースを多数実行することで形式仕様を検査、というハイブリッドな手法を確立した研究であった。

元々は関数型言語Haskellのテスト用ライブラリとして発表されたが、同じコンセプトに基づいた

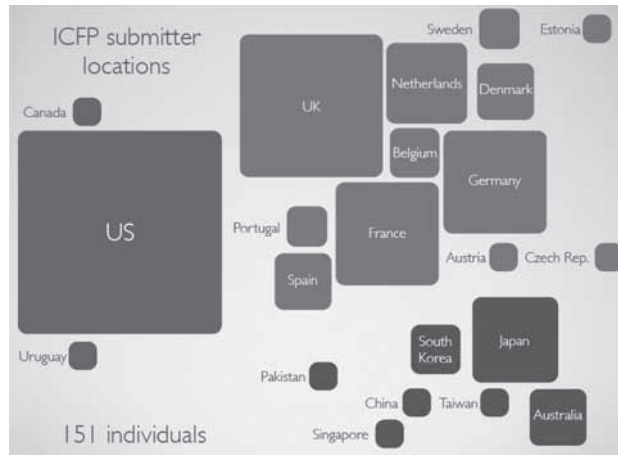


図-2 投稿論文の地域分布(プログラムチェアの報告スライドより)

Ruby, Python, Java, Scala などさまざまな言語で作られるようになってきている。この10年の影響力という面で、確かに納得の受賞であった。

ICFP 2011 in 東京!

ICFPでの日本のプレゼンスは比較的高い。2010年では、参加者17人/238人、採択論文数2件/33件、投稿数の詳細は不明だが、図-2に示すように、英米独仏に次ぐ規模となっている。また、ICFPの開催するプログラミングコンテストでは2年連続で日本からのチームが優勝を飾っている。

本稿の冒頭で、これまでICFPは欧州と北米で交互に開催されていると紹介したが、このような状況もあってか次回はこの慣習が破られることが発表された。ICFP 2011は、9月18日から24日にかけて、東京の一橋記念講堂にて開催される。詳細は<http://www.icfpconference.org/icfp2011/>を参照されたい。

ACMのプログラミング言語分科会SIGPLANの国際会議が日本で開催されるのはこれが初めてとのことで、読者の皆様も、この機会にぜひご参加を検討いただければと思う。

(平成22年11月6日受付)

稲葉一浩 | kinaba@nii.ac.jp
 東京大学情報理工学系研究科博士課程修了。国立情報学研究所・特任研究員。