Escaped From the Lab! Does ICSE Research Make a Difference?

Steven Fraser, Innoxec, Santa Clara, CA, USA, sdfraser@acm.org
Dennis Mancl, MSWX Software Experts, Bridgewater, NJ, USA, dmancl@acm.org

ABSTRACT

For 50 years ICSE has brought together researchers and practitioners in software engineering to share innovative research ideas. Steven Fraser (Innoxec) and Dennis Mancl (MSWX Software Experts) organized a panel to discuss how ICSE research has influenced software education, practices, and products. Panelists included Jonathan Aldrich (CMU), Robert Biddle (Carleton University), Margaret Burnett (Oregon State), Cierra Jaspan (Google), Tim Lister (Atlantic Systems Guild), and Gail Murphy (UBC). The panel observed that while ICSE is a great place for academics to publish and network, it has failed somewhat – because practitioners don't generally attend ICSE or read the conference proceedings. Suggested strategies to improve ICSE included – adding an "experience report" track at future ICSEs and developing other mechanisms to increase the participation of industry software practitioners. Panelists also suggested that academics and industry practitioners need to be better motivated to share research ideas and industry practices.

INTRODUCTION

To celebrate 50 years of ICSE, a panel "Does ICSE Research Make a Difference?" was organized by Steven Fraser (Innoxec) and Dennis Mancl (MSWX Software Experts). The context for the panel was given as "Escaped from the Lab" – research that originates from university, industry, or government labs. Panelists were drawn from the intersection of academia and industry, and included Jonathan Aldrich (Carnegie Mellon University), Robert Biddle (Carleton University), Margaret Burnett (Oregon State University), Ciera Jaspan (Google), Tim Lister (Atlantic Systems Guild), and Gail Murphy (University of British Columbia).

The discussion began with introductions based on bios and position statements published on the conference website [1]. Audience members were also polled. Most attendees identified themselves as researchers (56%) or researcher/practitioners (42%). Only 3% claimed to be "practitioners." An overwhelming majority of the audience (77%) believed that ICSE makes a difference through "personal connections" in contrast to those who believe that ICSE papers (10%) or presentations (6%) are influential.

Gail Murphy (UBC) is proud to have successfully converted research into products and influenced how software is developed by Fortune 500 companies. Her main area of interest has been in relieving pain points in software development. She made two points related to "Escaped from the Lab." First, academics frequently don't understand how industry builds software at scale; and second, academics should constantly demonstrate value and learn from industry.

Ciera Jaspan (Google) studies and works to improve Google developer productivity. Ciera offered a contrarian perspective: Although ICSE research could be influential, it hasn't been to the extent possible. Ciera noted that the impact of laboratory research is bidirectional. Innovations flow from academia to industry and from industry to academia. She observed that both communities ought to work to create incentives to bridge a communication gap she believes exists between academia and industry. In her view, practitioners struggle to identify the academic research that is relevant to their product work. Similarly, academics do not have incentives to share their research and write papers in a form that is easily understood by industry practitioners.

Jonathan Aldrich (CMU) introduced himself as Ciera's former doctoral advisor. He disagreed with Ciera's position. Jonathan gave examples of ICSE influence on industry: many research results presented at ICSE have found their way into advanced software architectures, refactoring tools, formal methods, and a variety of programming languages including Rust. However, academics are challenged to demonstrate and deploy their research results at scale.

Tim Lister (Atlantic Systems Guild) believes that industry seeks new ideas. Tim mentioned several ICSE-based ideas that have impacted industry development practices including stepwise refinement, modular design, information hiding, encapsulation, and design-by-contract.

Margaret Burnett (Oregon State) stated that research impacts the world, because it helps guide our thinking to get "outside the box." Margaret emphasized that we need to collaborate with people outside our field. She challenged the audience to have "big thoughts" and be receptive to extreme reviews – both positive and negative.

Robert Biddle (Carleton University, University of Auckland) related how things have changed since the 1980s. Today there are millions of programmers around the globe. Software is everywhere and greatly influences the world. That said, Robert believes that ICSE's influence faces a problem of scale. While ICSE brings together a growing academic community, the number of industry practitioners as authors or attendees is miniscule.

CONNECTIONS AND COLLABORATIONS

The panelists explored potential incentives for connections and collaborations between academic researchers and industry practitioners.

Robert started with his elaboration of Ciera's explanation of research incentives. He believes that the disconnect between research and practice (academics and industry practitioners) is due to the wrong incentives. Metrics are now everywhere, with many academic researchers reviewing their citation and H-index metrics daily. Robert believes that citation metrics overly focus on academics as individuals rather than as a collective community. In his opinion, citation-based metrics tend to increase and reinforce individual isolation.

Robert suggested approaches to foster academic-industry collaboration and communication. Industry software practitioners "in residence" at universities and academics working "in industry" (specifically in product development, not just industry research labs) would be a first step. An increase in academic focus on industry collaboration and the business of producing software might also help accelerate an "escape from the lab."

Margaret believes that the cost-reward structure for research is wrong. If research is put into practice, it generally doesn't bring any rewards that benefit the researcher. While research might produce a great innovation, it comes with a "cost" for industry to learn and adopt. As suggested by Goodhart's Law ("when a measure becomes a target, it ceases to be a good measure"), there are unintended consequences when publication metrics become a "target." The number of publications is a poor measure for assessing industry impact. Jonathan agreed. He suggested that research-practitioners should be evaluated on their industry impact, equally to citations or academic fame.

Robert gave an example from New Zealand in the 1990s before research grants, where researchers at his university delivered short courses to industry practitioners to generate funding for research needs including laptops and conference travel. As a side effect, these short courses provided researchers with the opportunity to learn first-hand about the problems and tools used by industry. Over time, industry attendees from past short courses collaborated more often with researchers, and they injected more industry-relevant material into regular university courses.

INFLUENCE THROUGH PUBLICATIONS

What incentives might help motivate industry practitioners to learn more through research papers? This seems to be a challenging task reflecting on a quote by Fred Brooks from a 2007 OOPSLA Panel [2]: "I know of no other field where people do less study of other people's work."

From the audience, Margaret-Anne (Peggy) Storey (UBC) suggested that academic papers are not easily understood by practitioners. Most papers lack sufficient evidence to demonstrate the applicability of emergent software practices. She noted that papers most relevant to practitioners are likely found on websites such as *Hacker News* or *LinkedIn*.

Ciera reported that at Google she frequently consults on practices related to technical debt, code quality, and programming languages. However, a small cadre of software coaches can't be everywhere. A discoverable medium of "best practices" would help industry staff explore and learn. The *What Works Clearinghouse* website has done this for education programs.

Gail observed that other technical fields have "continuing education" requirements. Lifelong learning for software professionals, both in industry and academia, is an idea that might be valuable to keep software engineers current with technological and social change. Gail also suggested that there is a role for people who "translate" research ideas into a more practical form. As an example, she indicated Greg Wilson's popular blog – *It Will Never Work in Theory*.

Transferring ideas to industry is hard. Margaret cited Barbara Ryder's 2005 study of the impact of software engineering research on programming languages [3], which explains that ideas don't simply transfer from academia to industry – there is no "button" to push that makes this happen. The panelists explained that the commercialization of new technologies and tools has been fostered through personal channels: connections, graduating students, open source, and collaboration. Everett Rogers proposed in his book *Diffusion of Innovations* [4] that the adoption of an innovation depends not only on interest (marketing) in the innovation, but also on adopter communication networks.

A SUCCESS STORY

How do publications and industry "activity" influence political and governmental interests? Margaret responded, "I think that everyone can do something to make the world more pleasant." She believes that "making things better" is more valued in software engineering and computer science. She notes ongoing R&D in the area of "human-centered AI."

The need to improve the world motivates and influences academia, industry, and governments. She claimed that academics write about new ideas which may then be evangelized by influencers. Margaret gave an example, describing how she launched her cross-disciplinary *GenderMag Project*, a project to make the design of software more inclusive. The project began with a graduate student who was excited about inclusive design. After reading papers from ICSE, TOSEM, and behavioral science research, the student designed experiments to explore how behavioral science and software could be connected. After the student graduated, Margaret kept the project active. She presented the work at a Microsoft workshop, which led to some of the workshop attendees contributing to the project. She has disseminated more ideas from the GenderMag project through publications, teaching, and collaborations.

EXPLAINING SOFTWARE ROLES

An audience member observed how little the public understands the role of software researchers and practitioners in producing value. Margaret said that she explains her job to others by giving a personalized response: she tells stories how something she builds in her "software world" helps specific users.

Jonathan shared a story he uses to explain his work as a software architecture researcher to non-technical individuals: Consider how the workers constructing a house need to follow an architectural plan. If they didn't follow the plan, there would be problems. This analogy worked for his research, but people working in other parts of software engineering might need a different story. It's good to have these easy-to-relate-to stories to connect to non-technical people.

Tim offered an explanation in the context of his software consultant role. He noted that there is no real "boundary" between software engineering and other kinds of work. Tim maintained that a software engineer's "bailiwick" includes more than understanding the technical issues such as project management and corporate politics. Tim asserted that his mission as an industry consultant is to help projects "survive and thrive." This is more than just guiding teams to follow software engineering best practices.

Robert shared a thought from Mary Shaw: "She talked about how much programming isn't to produce a product (and we know what it is), but to explore what a product like this might be." We aren't always building something we know, we are exploring what our eventual programming solution might be. Our work often looks like empirical science, but it is also sometimes more nebulous.

The ICSE 2025 General Co-chair Tim Lethbridge (University of Ottawa) interjected from the audience a personal admission into the discussion: "I have to admit that over the years, I haven't used papers from ICSE as a core source of knowledge." While Tim began his career in industry, he now is an academic who connects with industry through student capstone projects, industry co-ops, and open-source projects. Tim mused how people in industry increase their knowledge since it doesn't come directly from conferences like ICSE. Tim observed that practitioner sites like *StackOverflow* or the *Nielsen Norman Group* are great sources of information when he is working on a practical project.

Tim gave another story. As ICSE Co-Chair, he scanned all the accepted ICSE research papers, because he had to group them into categories to plan each conference session. Tim's assessment of papers: the abstracts were too short and most papers were too long. Who can read them all? There needs to be a way to systematically extract valuable information from research papers – in particular, information useful to industry.

SHARING PRACTICAL EXPERIENCE

How do we collect more practical experience? One audience member suggested that academic residencies for industry software engineers might parallel the learning of postgraduate medical doctors as they develop specialized clinical expertise in "residency." While residencies could work for software engineering, it is difficult to find sufficient case studies to teach students.

Written case studies help, if industry will write them. Rebecca Wirfs-Brock pioneered "Experience Reports" at ACM's OOPSLA and the XP series of conferences [5]. However, industry has little interest in documenting case studies because companies receive no direct benefit and may have proprietary concerns. Similarly, since academics receive no publication credit for experience reports, they aren't motivated to document practitioner case studies. Perhaps ICSE could help motivate academics with an experience report track? Jonathan discussed how CMU had established a requirement that students obtain industry experience and then share what they learned.

An audience question inspired a discussion on the diversity, inclusion, and human aspects of software engineering research. How can these topics be addressed when companies and governments downplay their importance? Margaret replied that her research is motivated by passion. Governments don't direct her passion. She knows how to navigate political trends and to write acceptable proposals. Margaret re-iterated that she has tried to make the field of software technology open and available to all. She believes the skills needed for software engineering research over the next decade include thinking skills – and collaboration with experts from other fields.

The panel returned to the question of incentives and how costs are borne by people in different roles when translating academic research into industry practice. Ciera shared her view of the value of offering one year industry fellowships for students or professors selected based on their research. In her experience, neither the academics nor Google would benefit. Ciera preferred to have researchers come to Google and work on any problem available – because with practical experience, the researcher learns and contributes to Google.

A member of the Uber platform team explained their tech transfer issues. He commented that their mission is simple: they try to build the best tools and platforms for Uber's developers to write better, cheaper, and faster software. Uber has a small team of new PhD grads, but they are struggling. Although they are often expected to use their research to create production-level tools, they are not trained to write production code. They are frustrated because they would prefer to do more research work, but they are also being asked to build high-quality tools for others. Margaret suggested that the best way forward is to get the new grads to pair with staff.

NETWORKING TO INSPIRE INNOVATION

An audience member (Tom DeMarco) said that if we expect ICSE to be a mechanism for injecting great software ideas into industry, it's a failure. Industry practitioners don't attend ICSE or read ICSE proceedings. A better way to view ICSE is as a networking event: an opportunity for people to make connections. Tom thought that academics and industry practitioners will have more impact if we work on strengthening the networks across the software engineering community.

Bertrand Meyer commented from the audience about the erroneous belief that there is a one-way flow of ideas, that academic ideas and research are translated into industry practice and products rather than vice versa. Bertrand explained that many software innovations originate in industry, often at smaller companies. For example, modern object-oriented programming, design patterns, web search engines, blockchain, and LLMs were revolutionized by companies – not academic research. Bertrand went on to advise academics to study software innovation at scale in industry. For example, in the field of building architecture, academics will continue to learn about design from the study ofhistorical buildings.

Bertrand challenged academics to learn from industrial practices. There are brilliant things that have emerged from industry – DevOps, web frameworks, and Agile development – these were new approaches to practical programming that have made an enormous impact. Some of these ideas receive surprisingly scant attention from academics. There are Agile software conferences (like XP) and the patterns community has conferences like those organized by the Hillside Group – but "industry" topics are not the main focus of prestigious conferences such as ICSE.

SUMMARY AND OBSERVATIONS

The panelists summarized their thoughts on "Escaped from the Lab" and whether ICSE research makes a difference. For some it was clear that students "escape from the [academic] lab" to influence the world. Others suggested that by helping humans thrive in a technological world we can inspire everything from cross disciplinary learning to new product design. Two additional observations from the panel:

1) Social connections and networking are necessary to help research "escape from the lab" and to influence software engineering. ICSE has become more academically focused and is less connected to industry practice than it might be – with few industry practitioners in attendance.
2) Some strategies were suggested for improving ICSE's influence on software engineering practice including: residencies (academics in industry, industry practitioners in academia), making it easier "to find" and "apply" research results, and increasing the focus on emergent industry practices.

The panel impresarios were somewhat surprised to learn that business technology transfer strategies were not more widely adopted given some earlier discussions at ACM/IEEE conferences with similar conclusions [6]. Researchers and practitioners, whether academics or industry practitioners, would likely benefit from improved communication and sharing results [7] – as would humanity in general.

REFERENCES

[1] ICSE 2025 conference. 2025. Panel: Escaped from the lab! Does ICSE research make a difference? - ICSE 2025 panel 2 position statements. Retrieved May 18, 2025 from https://conf.researchr.org/info/icse-2025/panel:-escaped-from-the-lab!-does-icse-research-make-a-difference%3F

[2] Steven Fraser and Dennis Mancl. 2008. No silver bullet: software engineering reloaded. IEEE Software 25, 1 (Jan./Feb. 2008), 91-94. https://doi.org/10.1109/MS.2008.14

[3] Barbara G. Ryder, Mary Lou Soffa, and Margaret Burnett. 2005. The impact of software engineering research on modern programming languages. ACM Trans. Softw. Eng. Methodol. 14, 4 (October 2005), 431–477. https://doi.org/10.1145/1101815.1101818

[4] Everett M. Rogers. 2002. Diffusion of Innovations (fifth ed.), Free Press, New York.

[5] Dennis Mancl and Steven D. Fraser. 2021. The stories we tell: experience, research, or patterns? In: Gregory, P., Kruchten, P. (eds) Agile Processes in Software Engineering and Extreme Programming – Workshops (XP 2021). Springer, 215-221. https://doi.org/10.1007/978-3-030-88583-0_22

[6] Steven Fraser, Ethan Hadar, Dennis Mancl, Bill Opdyke, David A. Owens, Dirk Riehle, and Linda Rising. 2008. Escaped from the lab: innovation practices in large organizations. In Companion to the 23rd ACM SIGPLAN conference on Object-oriented programming systems languages and applications (OOPSLA Companion '08). Association for Computing Machinery, New York, NY, USA, 787–790. https://doi.org/10.1145/1449814.1449862

[7] Steven Fraser, Sheri A. Brodeur, Randy H. Katz, Xue (Steve) Liu, Stefanie Molthagen-Schnöring, Sheng-Ying Aithne Pao. 2021. 2021 Strategies for "Socially Distant" University-Company Collaborations, 2021 IEEE/ACM 8th International Workshop on Software Engineering Research and Industrial Practice (SER&IP), Madrid, Spain, 2021, 26-27. https://doi.org/10.1109/SER IP52554.2021.00011

This report is based on the ICSE panel session held on May 2, 2025 in Ottawa, Canada. The panel report will appear in the July 2025 issue of SIGSOFT Software Engineering Notes.