Computational Fluency: Empowering Human Beings in the Digital Age

Gerhard Fischer

University of Colorado, Center for LifeLong Learning & Design (L3D), Boulder, CO, USA

Abstract

My contribution to the CoPDA workshop tries to enumerate different facets which can serve as starting points to create a meaningful shared understanding of the concept of "computational fluency". I hope that the contributions from other participants and the discussions at the workshop will result in the clarification of the relevance of computational fluency in the digital age.

Keywords

Computational fluency, barriers, design trade-offs, digital age

1. The Different Media Ages

Table 1 characterizes the different media ages since reading and writing transformed oral to literal societies almost 3000 years ago.

Table 1: The Different Media Ages

- age of orality
  - localized, personalized communication
  - importance of narratives and stories
- age of printed literacy
  - distributed cognition between knowledge in the head and knowledge in the world
  - externalization of knowledge (supporting sharing and critiquing)
  - learning efforts
  - emergence of scribes
- age of digital literacy
  - computers are capable of doing many tasks previously only done by humans
  - emergence of high-tech scribes
  - digital divide
  - change is greatly accelerated
- age of decentralization /
  - social production
  - mass collaboration (e.g.: Wikipedia)
  - social media
  - information overload problems
- age of COVID-19
  - changes are necessary based on an externally imposed problem
  - distant socializing needs to be supported
- age of digital fluency

IS-EUD 2023: 9th International Symposium on End-User Development, 6-8 June 2023, Cagliari, Italy
EMAIL: gerhard@colorado.edu (G. Fischer)

- engage and have control in personally meaningful activities
- independence of high-tech scribes:
- fundamental changes for thinking, working, learning, and collaborating
- technological changes are so fast that cultural transformations cannot keep up

2. Differentiating Computational Literacy and Computational Fluency

Literacy. Literacy with digital media [1] has often been characterized and practiced in educational institutions to teach learners to acquire skills and a degree of competency with some of today's computer applications (e.g., word processing, email, html, drawing programs, etc.). It has focused on teaching learners to generate syntactically correct expressions with the primary concern on form rather than content. The limitations of literacy understood in this way are that it lacks conceptual understanding, it is ill-suited to cope with change (e.g., no migration path to new skills are de- developed), and it prevents people from using digital media for personally meaningful problems.

Fluency. Fluency with IT [1] is defined as "the ability to reformulate knowledge, to express oneself creatively and appropriately, and to produce and generate information rather than simply to comprehend it" [2]. Fluency goes beyond traditional notions of computer literacy by requiring a deeper, more essential understanding and mastery of IT, and it is a prerequisite to creating a personal and deep relationship with media [3]. Fluency is characterized by different levels of sophistication. In addition, it is dynamic and changes over time, requiring an engagement in lifelong learning. Fluency should not be reduced to formalized knowledge about programming, especially if programming is understood as writing step-by-step "recipes", as it has been mostly conceptualized in the past. Fluency should include contemporary skills such as using existing applications, supporting self-expression, and being engaged in cultures of participation [4] by allowing users to tailor, customize, and evolve systems to their wants in personally meaningful tasks thereby making users independent of high-tech scribes [5].

3. Why is computational fluency important in the digital age

Computational fluency is important in the digital age for several reasons including:

Dealing actively with personal meaningful problems: with computational artifacts readily available, the primary concern (at least in many countries) has shifted from who has access to information technologies to who will have the knowledge that will position them to design, create, invent, and use the technologies to enhance their personal lives.
Fostering Social Ties and Civic Engagement: a key ingredient of a healthy democracy is a vibrant civil society. Achieving this requires an active citizenry with the values, skills, and knowledge to better their communities. Computational fluency allows individuals to participate more fully in democratic societies [6].
Workplace Skills: In today's economy, many jobs require the use of computers and digital tools. Having computational fluency is a crucial skill for the job market, and it can improve employability and job performance. The ability to use technology efficiently will improve personal productivity, organization, and communication.
Digital Citizenship: In the digital age, being a responsible citizen means being able to navigate and understand the digital world. Computational fluency allows individuals to participate more fully in society, whether through online communication, accessing information, or engaging in online communities.

4. Barriers for Becoming and Remaining Computational Fluid

Information, Participation, and Choice Overload: The space of computational artifacts available today is overwhelming (e.g.: there are approximately 9 million apps available worldwide, with 2 million available in the Apple App Store). Even the most sophisticated and dedicated individuals can only know a tiny fraction of them, Search tools and AI tools (e.g.: supporting user and task modeling and learning on demand) are a necessity to assist users in selecting the tools that may be relevant for the problems and tasks they are facing.

Lifelong Learning Requirements for Coping with Rapidly Evolving Technologies: Digital tools and technologies are constantly evolving, and it is an ongoing challenge to keep up with new developments and changes making it a demanding and time-consuming task to maintain computational fluency over time.

Having Control of deciding not to participate in specific parts of digital life. People should be able without suffering too many disadvantages to choose and exercise digital abstinence and information celibacy (at least for parts of their daily activities and their lives).

5. Support environments for Reducing the Efforts and Learning Demands to Acquire Computational Fluency

Human Problem Domain Interaction: Domain-orientated environments [7] will bring tasks to the forefront and support domain designers with *human problem-domain communication* with a layered architecture as indicated in the figure.

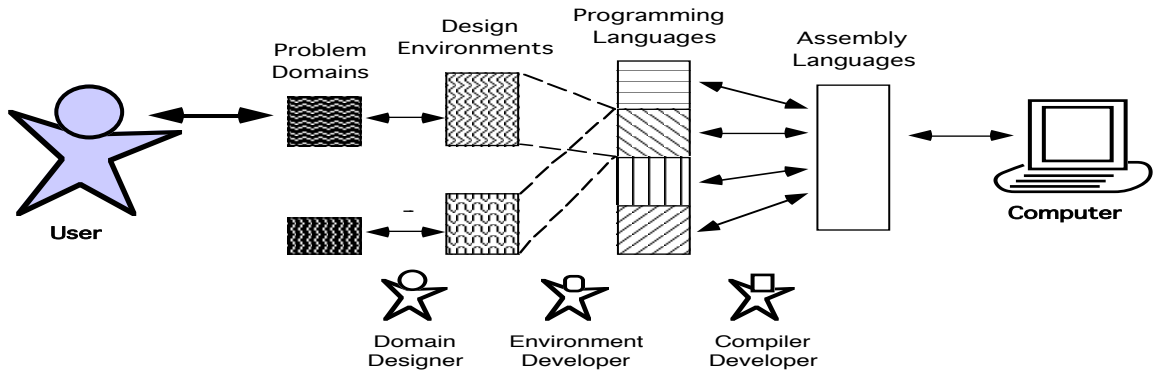

Figure 1: Layered Architecture for Supporting Human Problem Domain Interaction

Supporting Rich Ecologies for Computational Fluency: Computational Fluency is not a binary concept being either absent or mastered. There are many different levels for participation in systems such as Scratch [8], Wikipedia or Open Source Environments. Our empirical research [9] has identified the different roles of participation in Open Source Environments as indicated in the figure below.

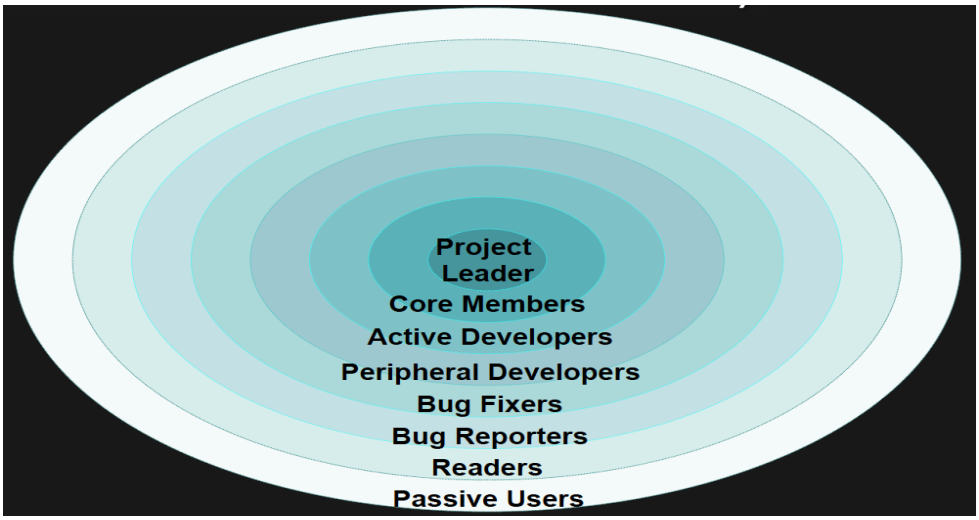

Figure 2: Different Roles of Participation in Open Source Environments

Finding the Right Mixture between Skills and Challenges: the challenge level of the task is an important factor in both computational fluency and flow. In order to enter a state of flow [10], the task must be challenging enough to require the individual's attention and skills, but not so challenging that it becomes overwhelming or frustrating. Similarly, in order to develop computational fluency, individuals must be challenged with tasks that are appropriate for their skill level, and gradually increase in difficulty over time. Systems that meet this demand will need a "Low Threshold and High Ceiling".

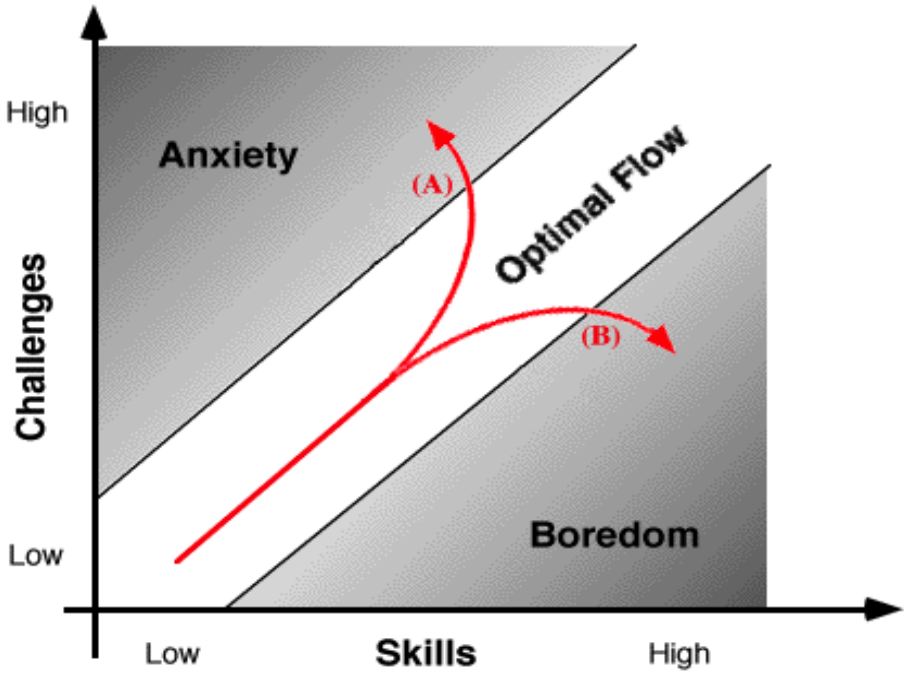

Figure 3: Flow — Identifying the Best Combination between Skills and Challenges

Transcending the Power of Printed Media. Innovative uses of digital media should transcend and exploit the unique properties of computational media that are absent in principle in printed media. Printed media *do not have interpretive power* — they can convey information, but they can- not analyze the work products created. For example, *simulation* is a process that can show us the implications of our assumptions and allow us to engage in "what-if" problem-solving, and *critiquing* is a process that *analyzes our work products* and increases the "back-talk" of an artifact [11] by presenting a reasoned opinion about it [12].

An Example for Putting Owners of Problems in Charge: An interview that we conducted some time ago with a geoscientist at CU Boulder highlighted the importance of supporting end-user development as an important characteristic of computation fluency. The geoscientist used a couple of domain-specific software systems to analyze his research data but none of the existing systems could provide complete solutions to his problems as his research unfolded and his understanding of the problem, data, and results proceeded. During the interview, he says:

Comments by the Geoscientist	**Underlying Rational and Justification**
"I spend in average an hour every day developing software for myself to analyze the data I collected because there is not any available software"	there is an infinite number of different problems in the world → there is a need for an infinite number of software systems
"Even if there is a software developer sitting next to me, it would not be of much help because my needs vary as my research progresses and I cannot clearly explain what I want to do at any moment."	ill-defined problems cannot be delegated; they require "unselfconscious cultures of design"
"Even if the software developer can manage to write a program for me, I will not know if he or she has done it right without looking at the code."	back-talk of the artifact under construction has to go back to the owner of the problem
"So I spent three months to gain enough programming knowledge to get by".	"to get by" is an important objective
"Software development has now become an essential task of my research, but I do not consider myself a software developer and I don't know many other things about software development."	The rationale for engaging in end-user development in domains that are personally relevant

Clearly, this geoscientist is not a professional software engineer, and he does not intend to become one. He is not a mere end-user either because he engages regularly in intensive software development that goes beyond what most end-user programming environments have tried to support. The activity of software development is not anymore an exclusive activity of professional software engineers. Many domain experts (such as the geoscientist) are engaging in software development as intensive and technically challenging as many professional software engineers, but for the problems of their own rather than the problems of others, and as an instrument for a larger context rather than an end artifact to be delivered.

6. Design Trade-Offs associated with Computational Fluency

Another opportunity to gain a deeper understanding of Computational Fluency is to identify relevant design trade-offs for the concept such as:

- Enriching and empowering (with Intelligence Augmentation (IA) approaches) versus automating human activities (with Artificial Intelligence (AI) approaches leading to Overreliance and deskilling;
- Democratizing decision-making processes versus overburdening citizens with personally irrelevant information and activities;
- Being in control (autonomy) versus being controlled (prescriptive guidance);
- Finding a balance between flexibility and stability in working life;
- Increasing versus decreasing the digital divide [13].

7. Topics and Objectives for the Workshop

A core objective of the workshop should be creating a (shared) understanding of computational fluency as a fundamental objective for the digital age. Topics for discussion at the workshop could be:

- Will AI hinder or promote Computational Fluency?
- Will ChatGPT have a substantial impact to rethink Computational Fluency?
- Will Meta-Design and End-User development hinder or promote Computational Fluency?
- How can the learning demands associated with Computational Fluency be reduced?
- Will Computational Fluency be an essential capability to contribute positively to quality-of-life issues (e.g.: actively engaging in personally meaningful activities, having control, assisting in sense-making, and supporting a desirable work/life balance in the digital age)?

8. The Past, the Present, and the Future of the CoPDA Workshops

I would like to repeat an argument that I have made previously:

The IS-EUD'2023 workshop is the 7th CoPDA workshop (see Figure below). An important challenge for the researchers getting together in the workshop this year will be to explore the foundational idea(s) that these workshops have pursued and how they are related to each other. A particular objective of all previous CoPDA workshops has been to *collectively identify important and interesting themes for future workshops* and my hope is that this will happen again this year.

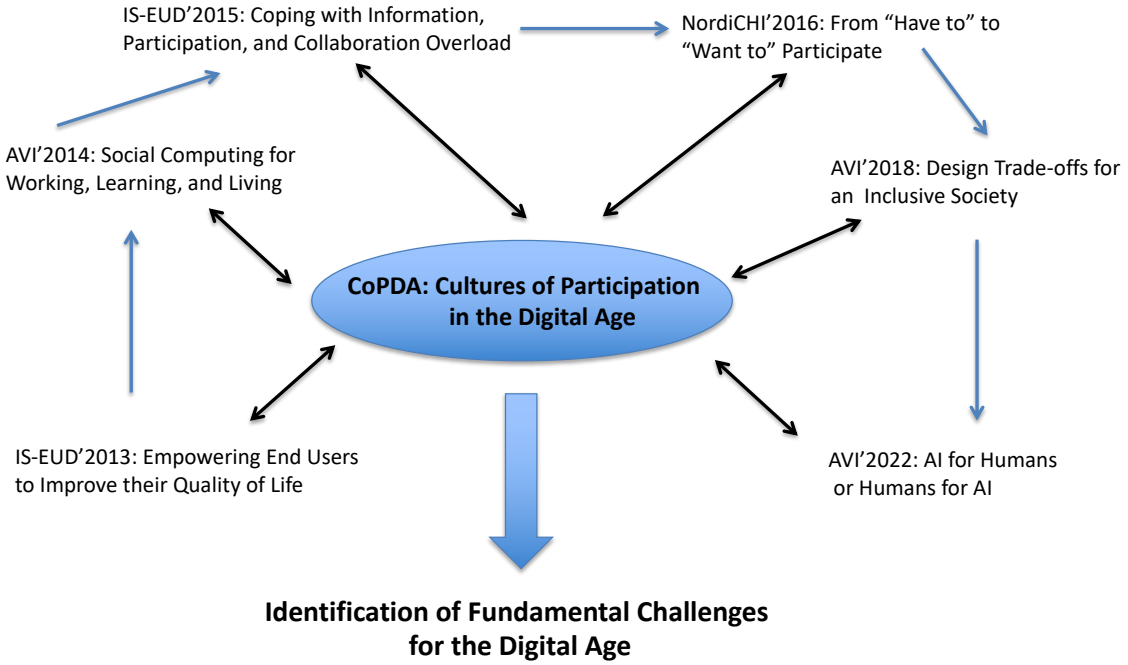

Figure 4: Overview of the CoPDA Workshops (2013-2022)

References

[1] National-Research-Council, Being Fluent with Information Technology. 1999, Washington, DC: National Academy Press.

[2] Shneiderman, B., et al., Creativity Support Tools: Report from a U.S. National Science Foundation Sponsored Workshop. International Journal of Human–Computer Interaction, 2006. 20(2): p. 61–77.

[3] Kay, A.C., Computer Software. Scientific American, 1984. 251(3): p. 52-59.

[4] Fischer, G., Understanding, Fostering, and Supporting Cultures of Participation. ACM Interactions 2011. XVIII.3: p. 42-53.
[5] Fischer, G., Computational Literacy and Fluency: Being Independent of High-Tech Scribes, in Strukturieren - Modellieren - Kommunizieren. Leitbild mathematischer und informatischer Aktivitäten, J. Engel, R. Vogel, and S. Wessolowski, Editors. 2005: Hildesheim. p. 217-230.
[6] Jenkins, H., Confronting the Challenges of Participatory Cultures: Media Education for the 21st Century. 2009, Cambridge, MA: MIT Press.
[7] Fischer, G., Domain-Oriented Design Environments. Automated Software Engineering, 1994. 1(2): p. 177-203.
[8] Resnick, M., et al., Scratch: Programming for All. Communications of the ACM, 2009. 52(11): p. 60-67.
[9] Fischer, G., K. Nakakoji, and Y. Ye, Meta-Design: Guidelines for Supporting Domain Experts in Software Development. IEEE Software 2009. September/October: p. 37-44
[10] Csikszentmihalyi, M., Creativity — Flow and the Psychology of Discovery and Invention. 1996, New York, NY: HarperCollins Publishers.
[11] Bruner, J., The Culture of Education. 1996, Cambridge, MA: Harvard University Press.
[12] Fischer, G., et al., The Role of Critiquing in Cooperative Problem Solving. ACM Transactions on Information Systems, 1991. 9(2): p. 123-151.
[13] Schön, D.A., B. Sanyal, and W.J. Mitchell, eds. High Technology and Low-Incoming Communities. 1999, MIT Press: Cambridge, MA.