Internalist and Externalist HCI

Harold Thimbleby
Swansea University
Wales
SA2 8PP
+44 1792295393

harold@thimbleby.net

Will Thimbleby
Swansea University
Wales
SA2 8PP
+44 1792295393

will@thimbleby.net

ABSTRACT

The history of technology, as a discipline, supports alternate points of view termed *internalist* and *externalist*, which terms highlight an approximately similar division in points of view within HCI. Conventional HCI is externalist, rightly concerned with human-centered issues; but externalism risks ignoring important internalist issues. A successful human-computer system is better if it is successful from *both* perspectives.

This discussion paper argues that the externalist view, while necessary and immensely useful, is not sufficient—and in the worst case, risks eclipsing innovation from internalist quarters.

1. INTRODUCTION

David Nye's review of the history of technology [14] uses the clear terms *internalist* and *externalist*, applying them to styles of historical analysis.

Why did the internal combustion engine triumph over the alternatives, horse, steam and electric? An internalist might emphasize the power-to-weight ratio of the internal combustion engine; an externalist might emphasize the lower cost of the Ford Model T and the dramatic impact cost had on a growing market. An internalist, then, considers the technology as such.

- **Externalism** is focused on the world external to the user interface: *human*-interaction and e.g., observation, evaluation, cognition, etc.

- **Internalism** is focused on the world internal to the user interface: *computer* interaction and e.g., logic, engineering, computation, etc.

An example illustrating human-computer interaction issues is Tracy Kidder's classic *The Soul of a New Machine* [10]. The book traces the development of a computer, the Data General Eclipse MV/8000, all the technical issues, right up to the point that the finished product is brought to market. Then the book ends, just when the external world of the computer and its possible use starts to get interesting. The book takes an internalist view.

Of course both views are needed in a balanced discussion, and indeed Nye provides a masterful analysis. We believe Nye's internalist/externalist terms from the history of technology have

value in distinguishing major styles in the way HCI is viewed, presented and undertaken.

Clayton Lewis proposed a similar, but, psychological distinction for HCI, that of *inner* and *outer HCI* [13]. Here, inner and outer refer to cognitive processes and human behavior respectively. Lewis emphasizes the potentially fruitful interplay of inner and outer HCI. Curiously, while his the terms "inner" and "outer" might at first seem to cover everything, Lewis *excludes* the computer (or other interactive system)—he simply does not mention it in his conception of HCI! It is as if the interactive system is a given, taken for granted, rather than a legitimate object of study in its own right.

Similarly in the "Kittle House Manifesto" [3] Carroll suggests that academic psychology has had no impact on interactive design practice, and that major innovations in practice (e.g., Sketchpad, an innovative graphics program) have made no explicit use of psychology. He bemoans the fact that HCI does not use science, or that if it does the relation is haphazard. Yet, curiously, he overlooks that computer science is science too, and in fact underlies the major contributions he describes as driving innovation. While it seems to us quite right to try to promote psychological science and explore why it is in some sense under-rated or used haphazardly, it seems counter-productive to the wider purpose of HCI to overlook computational science. Carroll's more recent collection [4] sees HCI as something computer scientists need to be taught, as something quite other than computer science, rather than something that can draw on computer science *as well as* human sciences.

This externalist emphasis of the HCI field is routinely found in the standard HCI textbooks, of which most take externalist points of view—indeed, [5] suggests that teaching HCI should cover the computer science which standard HCI textbooks omit.

Barnard, May, Duke and Duce remind us of "syndesis," binding together systems that contain interacting subsystems, such as people and computers. They introduce the terms "Type 1 theory" and "Type 2 theory," referring to approaches that go *deeper* or that go *across* interaction respectively. They warn that we are not very good at establishing Type 2 connections, and this weakness may lead to "the fragmentation and demise of HCI as a coherent science" [1].

It seems that HCI needs terminology to discuss these issues. Our internalist/externalist distinction is analogous to the Lewis inner/outer HCI distinctions, but from the point of view of the computer rather than the human. Without repeating Lewis's arguments here, we too see the great potential of fruitful interplay between internalist and externalist perspectives.

Just as a brain-computer interaction (BCI) researcher would certainly wish to go deeper into the "inner HCI" than Lewis does, so also our "internalist" perspective has a rich internal

structure—it isn't just "the computer" set against the wide range of standard HCI disciplines, anthropology, psychology, social science, economics, marketing, design; the internalist sees algorithms, complexity, information theory, proof, requirements, hardware, graphics, databases, and so forth … a rich science contributing to HCI.

1.1 The Authors' Perspective

Both authors of this paper have an internalist background, and it is unashamedly from this perspective that this paper has been written. The paper has a twofold purpose: to name and introduce a useful distinction for HCI, and to stimulate debate on the balance—or the lack of balance—in HCI as practiced, and hence stimulate thinking on strategies for doing better.

We believe the internalist/externalist distinction allows a constructive discussion about the methodologies of HCI, without diminishing either internalist or externalist points of view. By naming the distinction, we suggest that there are different *and valid* views about how HCI, and particularly HCI research, can and should be done. Nevertheless, we believe internalist HCI tends to be under-valued by the more dominant externalist point of view, and this paper therefore makes an enthusiastic case for internalism.

HCI could not exist without programming computers, which is an internalist perspective, and also HCI could not exist without the human context and study, which is an externalist perspective. Singly, internalist and externalist perspectives are monocular and lack depth and perspective. Both are needed.

2. HOW WE GOT HERE

The HCI community's traditional emphasis of externalist perspectives to some extent eclipses internalist perspectives. Historically, existing externalist methodologies were ready when they were needed: there was and still is a very substantial resource of experimental psychology that was applied and works to a high standard. In contrast, it might be said that most early internalists did not know what they were doing; see below when we comment on the Therac-25.

A second, crucial, reason for the current emphasis on externalist methods in HCI is that external experimental methods can be used independently of the specifics of internalist details. Every HCI system has very different internals, and requires investment in specific programming and design; in contrast, the externalist methods (e.g., cognitive walkthrough, think aloud, eye tracking) work on all systems. Experimental designs, statistical methods and so on, can be applied to a word processor or to a graphics package with little modification. In contrast, a new contribution to HCI by an internalist might take years of work that has no other application. It is noteworthy that most externalist studies of programming in HCI design use trivial programs, because programming real user interfaces is too slow. Inevitably, few internalists contribute to mainstream HCI.

Perhaps the HCI community has changed too. As fewer internalists contribute at the same rate as externalists, the peer community becomes dominated by externalist values. If an internalist submitted a result to a conference or journal now, most referees calling themselves members of the HCI community would be externalists.

ACM CHI, the major international HCI conference, is primarily externalist. In contrast one of the major internalist conferences, *DSVIS* (Design, Specification and Verification of Interactive Systems) has only a hundredth of the participants. This reflects a difference in the sizes of the communities. Thus, internalists face higher hurdles to participate in the development of the field. Then, as the externalists operate in a community dominated by externalists, it appears reasonable to *require* externalist criteria for contributing to that community: possibly even a hegemony—being defined as the emphasis of cultural beliefs, values, and practices to the dismissal and over-looking of others.

3. SAMPLE SYSTEMS

3.1 Therac-25

Horrific stories of bad HCI abound. The Therac-25 was a medical device that killed patients as a result of "operator" error (actually system design error). It is primarily an example of inadequate internalist HCI, an argument for better internalist HCI rather than fixing design problems with externalist HCI. Bad programming killed people.

Although the Therac-25 story is an extreme example, the case illustrates how important it is for user-centered design to react against sloppy programming practices—this paper is not arguing internalism is a panacea! Given that many programmers are not computer scientists, UCD *is necessary* to improve things.

One could argue that iterative design gained prominence to compensate for the difficulty of writing good software, particularly given the typical programmer skills available to industry.

3.2 Calculators

By considering logic programming, Runciman and HThimbleby introduced an analytic concept, *equal opportunity* [15]. HThimbleby used equal opportunity to constrain the design of a new user interface, choosing a calculator, as this is a well-researched artifact. Background research revealed how conventional calculators were badly designed, an internalist criticism of their poor technology [16]. Somehow this critical observation had escaped externalist research on calculator user interfaces.

We question the point of externalist research when it ignores the *intrinsic failure* of the technology; what point is iterative design or working with users when the conceptual problems of the user interface are so hard, complex and broken? HThimbleby made a technically improved calculator available to the community in 1986. However, it was not till 2004 that it had any externalist evaluation [2]. More recently, WThimbleby generalised the calculator, and made its user interface recognize handwriting [17,18,19]. This calculator has had a modest externalist evaluation [17].

The new calculator was developed entirely by internalist considerations. Specifically, it should do mathematics properly [19]. Few externalist considerations drove its design, yet it is very successful. The calculator was exhibited at Royal Society Summer Science Exhibition, 2005; at the exhibition, several thousand people used it. 90% of respondees said they really liked it or loved it. But despite the unusually large scale of the survey and feedback we gained no new ideas from users that would contribute to iterative design improvements.

Some feedback from users at the exhibition is listed below:

- "It visualizes the internal workings of abstract calculations, fun, as it is wonderful! Fun! Engaging and importantly visible!"—University Professor

- "Calculators seem clumsy and hard to use—the new method is genius!—when can I buy one in the shops (If I

had had one I would have done A level maths)"—A–Level Student

- "Engagement, excitement, interactivity, seamless, more visually appealing and easier to use!"—Teacher

- "I've never seen anything that's brought a smile to my face while doing addition, but this has. For that reason alone, I want one!"—Artist

The point we would like to make is that an internalist design program has produced a good user interface, recognized as such by users. Yet by conventional externalist HCI criteria, the work would not be acceptable for publication.

3.3 Graphics Programs

The calculator is an example of an internalist HCI research program, spanning twenty years before it resulted in a user interface that attracted attention. In contrast WThimbleby conceived, designed and built a vector graphics editor within two years, as a purely internalist project.

The resulting program, Lineform, was fully formed on its initial release. No early focus on users, no empirical design, no iterative design [7] informed its development—though of course computer science and HCI principles did inform and direct its development.

The quality of the design was recognized by the award to WThimbleby of the 2005 Apple Student Design Award. Arguably, this shows the user interface design was better than of thousands of others (i.e., the number of competitors)—which, had they been realistically entered into the review, should have been excellent programs in their own right.

Lineform is sold by Freeverse Software and has been commercially successful. The program has been reviewed in commercial magazines and web sites. Its reception has been uniformly favorable.

Below are some sample quotes from reviews. They are included to support the claim that the HCI in Lineform is successful, regardless of its lack of externalist methodology. Like the facts we presented about the calculator, the evidence supports our view that HCI contributions can be good despite the lack of externalist, practices.

- "Lineform from Freeverse Software claims to be the solution for modern drawing and illustration. It is. Winner of a 2006 Apple Design Award, Lineform is not only easy to use, but the interface design makes the application so intuitive, Mac users need no explanation to start illustrating."—*CreativeMac* (Feb 2007)

- "It's not often that you find a product you literally have to gush over … but Lineform, for me at least, is that product."—*AppleGazette* (Jan 2007)

- "Lineform has two other selling points. First, its speed: the program launches in a couple of seconds and shames Illustrator throughout in its responsiveness. Second, its ease of use. The simple interface alone makes it easier to find things."—*MacUser* (Issue 22 Volume 22)

An internalist design program produced a very good user interface, recognized as excellent by the market and critical reviewers. Yet by conventional externalist HCI criteria, the work would not be acceptable for publication.

3.4 Google

On any measure Google is an extremely successful user interface, with a value to users that exceeds most conventional user interfaces studied in HCI. Google is in fact just a text field with a substantial algorithm behind it [12]: its user interface is successful because it has a good internalist design. First, the internalist algorithm *then* the user interface. *Once* Google works it *then* makes sense to evaluate it and refine it from an externalist point of view: what services do users want given that Google works, and how can they be made better? However, the original, key HCI innovation was internalist.

Few of the services Google now offers would have made any sense to users or anyone else until after the basic algorithm worked, and had been demonstrated working well. Although externalism is now essential to Google, it was not how it started.

4. SAMPLE ISSUES

4.1 Anecdotes

If Jo is using a system, and this is reported in a research contribution, then an externalist wishes to know in what way Jo is typical of the population and to what extent, if at all, the particular interaction is typical. Jo may be idiosyncratic; the experimenter may have misdirected Jo. If we wish, ultimately, to design better interfaces for anybody other than Jo, we need reliable, generalizable knowledge. Statistics is a good way to characterize reliable generalization, and a one-off experiment with a unique individual would be hard-pressed to be reliable.

From an internalist perspective things look very different. Internal arguments are independent of the user. For example, computability could show that certain tasks are impossible. Not just for Jo, but for *anybody*—impossible for the whole human population, martians, dogs and bacteria. One hardly needs to recruit conventional experimental methods to make such claims reliable. This is not an anecdotal claim, but an analytic claim.

The confusion of these two methodologies undermines communication. It is our experience that internalist papers submitted to journals and conferences have been rejected because the referees have interpreted our analytic descriptions as "anecdotal."

The desire that contributions to HCI must include sufficient (and valid) externalist content before they are acceptable, increases the burden on the internalist researcher. Few researchers are able to span the internal/external bridge; different skills, different theory, different methods are required. Moreover, in the way of things, externalist work can only follow after internal work—or simulate it (e.g., with paper prototyping, which has no internalist content). Perhaps this is *the* gulf of HCI? An internalist has to do twice as much work?

4.2 Reproducibility

The systems mentioned in this paper are fully working systems and can be downloaded by interested researchers (www.freeverse.com/lineform for the graphics program, www.cs.swansea.ac.uk/calculators for the calculator, and labs.google.com for an API). From an internalist perspective, the research these systems embody is reproducible. That is, the claims we make about the quality and design can readily be checked by any interested researchers; because the claims are user independent.

From the perspective of the present paper, of emphasizing internalist HCI, it seems a great advantage that exactly what we have contributed—the underlying science, the programs, and so forth—are completely available to any researchers who wish to build on or critique our work. This level of reproducibility is very rarely the case with externalist HCI research.

4.3 Opposition or complimentarity?

At the BCS HCI 1995 conference, what we would now call an internalist/externalist debate was presented by an externalist in a keynote, metaphorically, as an actual *war*: "Which trench are you shooting from?" [6], illustrated with pictures of carnage. Another keynote at the same conference [8] suggested that "in a nutshell … what I see is a need to get away from the computer at centre stage, and a need for methods of description that make themselves useful …" If it's a war, consider [21], which starts off, "If you want to make software developers squirm…" and sets out to create the impression that developers don't know what they are doing. Some don't, no doubt, but most have a hard enough job getting systems to work at all, and they should not be blamed for problems that arise through poor management expectations and requirements that *nobody* understood until their systems were working.

Landauer's *The Trouble with Computers* [11] blames programmers for being "arrogant" (p173)—not designing for users, testing, evaluating, and so on. Programmers have "fantasies" he says. Yet he also mentions that Stu Card "a leading expert in HCI" was "confident" that a new word processor would be "vastly" better—but was proved wrong. Thus he makes rhetorical distinctions whose effects are to discredit the internalist perspective in HCI: internalists are "arrogant" whereas equally wrong externalists are "leading."

We surely need more balanced views, particular as both internalist and externalist share the same goals for the user. A first step in being more balanced is to name the imbalance.

Grudin, one-time editor of the *ACM Transactions on Computer-Human Interaction*, presented a mature view of the diversity of the HCI community [9], based on his experience as editor and final arbiter between conflicting referee and author points of view. A non-partisan view is [20], which argues how easy it is for differences to escalate to unconstructive conflict.

5. CONCLUSIONS

This paper has proposed a distinction between externalist and internalist approaches to HCI. The distinction helps clarify the nature of HCI research and practice, as well as preferred approaches within the HCI research community.

This paper described a selection of very different products of internalist HCI. None have been developed through or supported research that would have met conventional externalist HCI criteria, indeed none followed any recommended externalist HCI development cycles—yet all are successful. Of course the systems beg a wide range of externalist questions, but the fact that one can now do externalist work does not mean it was necessary to do it for the overall work to form a valid contribution to HCI.

Our purpose is not to dismiss externalist approaches, but to recognize that an internalist approach to HCI can be very effective and lead to good user interface design. Internalist design and research can be valid without any externalist evaluation.

Given that the computer science community argues that design should start with a mathematically rigorous specification, and then refine to implementation—almost the opposite of the externalist HCI view of design—there are new questions to be asked. Can internalist approaches lead to quality HCI, and if so, to what extent and under what assumptions? This paper has shown that internalist HCI can. We need to see more internally-driven HCI, and we need to explore when and why it is successful.

6. REFERENCES

[1] Barnard, P., May, J., Duke, D. & Duce, D., systems, Interactions, and Macrotheory, *ACM Transactions on Computer-Human Interaction*, **7**(2):222–262, 2000.

[2] Cairns, P., Thimbleby, H. & Wali, S., Evaluating a Novel Calculator Interface, *Proceedings BCS HCI Conference*, **2**:9–12, 2004.

[3] Carroll, J. M., Introduction: The Kittle House Manifesto, *Designing Interaction*, J. M. Carroll, ed., pp1–16, Cambridge University Press, 1991.

[4] Carroll, J. M., ed., *HCI Models Theories and Frameworks*, Morgan Kaufmann, 2003.

[5] Cockburn, A. & Bell, T., Extending HCI in the Computer Science Curriculum, ACM International Conference Proceeding Series, **3**, *Proceedings of the 3rd Australasian conference on Computer Science Education*, 113–120, 1998.

[6] Gasen, J. B., Support for HCI Educators: A View from the Trenches, Proceedings BCS HCI Conference, 21–36, 1995.

[7] Gould, J. D. & Lewis, C., Designing for usability: key principles and what designers think, *Communications of the ACM*, **28**(3):300–311, 1985.

[8] Green, T. R. G., Looking Through HCI, Proceedings BCS HCI Conference, 21–36, 1995.

[9] Grudin, J. "Crossing the Divide," *ACM Transactions on Computer-Human Interaction*, **11**(1):1-25, 2004.

[10] Kidder, T., *The Soul Of A New Machine*, Back Bay Books, 2000.

[11] Landauer, T., *The Trouble with Computers*, MIT Press, 1995.

[12] Langville, A. N., Meyer, C. D., *Google's PageRank and Beyond*, Princeton, 2006.

[13] Lewis, C., Inner and Outer Theory in HCI, in *Designing Interaction*, J. M. Carroll, ed., pp154–161, Cambridge University Press, 1991.

[14] Nye, D. E., *Technology Matters*, MIT Press, 2006.

[15] Thimbleby, H. & Runciman, C., Equal Opportunity Interactive Systems, *International Journal of Man-Machine Studies*, **25**(4):439–451, 1986.

[16] Thimbleby, H. Calculators are Needlessly Bad, *International Journal of Human-Computer Studies*, **52**(6):1031–1069, 2000.

[17] Thimbleby, W., A novel pen-based calculator and its evaluation, *Proceedings ACM Nordic Conference on Human-Computer interaction*, 445–448, 2004.

[18] Thimbleby, W. & Thimbleby, H., A Novel Gesture-Based Calculator and Its Design Principles, *Proceedings BCS HCI Conference*, **2**:27–32, 2005.

[19] Thimbleby, H. & Thimbleby, W., Mathematical Mathematical User Interfaces, *DSVIS 2007*, in press.

[20] Thimbleby, H., Supporting Diverse HCI Research, *Proceedings BCS HCI Conference*, **2**:125–128, 2004.

[21] Udell, J., Capturing user experience closes the feedback loop, *InfoWorld*, www.infoworld.com/article/04/06/04/23FEuser_1.html 2004.