

(re) configuring the web user in the digital economy

Ramine Tinati, Catherine Pope, Susan Halford, Leslie Carr
Web Science Doctoral Training Centre,
University of Southampton
lac@ecs.soton.ac.uk

ABSTRACT

The role of ‘the user’ is critical to the development of Web Science, a discipline that seeks to promote a multi-disciplinary understanding of the Web with regards to its evolution and its future. In this paper, we address the formulation of ‘the user’ has in computer science and social science. Our aim is to explore how we might bring these different perspectives closer together to enhance our understanding of users, and hence to improve our ability to innovate new kinds of Web environment and, ultimately, a Web-enhanced society. At one level we can see the Web as simply a computer system ‘writ large’ such that an improved understanding of the user would be beneficial to technologists and sociologists alike. However, we suggest that the scale, scope and impact of the Web mean that we need to consider a new approach to understanding its users and usage.

Categories and Subject Descriptors

D.2.7 [Software Engineering]: Distribution, Maintenance, and Enhancement; K.4.0 [Computers and Society]

General Terms

Design, Human Factors, Theory.

Keywords

Web Science, co-constitution, Socio-Technical perspectives

1. INTRODUCTION

In proposing a framework for Web Science [2], Berners-Lee *et al.* note the importance of studying the relationship between the social and technical aspects of the web. They hint, but do not explicitly say, that the relationship between society and the web is one of co-constitution – that society and the web shape each other [9].

This paper approaches the relationship between society and the web by looking at how the user is perceived from both computer science and social science. We suggest that within the computer science, the user is perceived as homogeneous and static, typically represented as a set of user requirements and/or specifications that are gathered during the software development lifecycle. Various models have been developed to capture ‘user’ input to computer design, and some of these are outlined in the paper.

The second section of this paper looks at how the user is viewed in social science, tracing how different approaches such as SCOT (Social Construction of Technology) [3] are used to understand user engagement with technologies like computer software. We examine how concepts like technological determinism, social determinism [5] and interpretive flexibility [3] have been used to offer new models of the relationship between technology and society. From this viewpoint the user is

not defined as a set of design requirements, but rather as an actor at the intersection of a complex set of inter-related phenomena including culture, gender, wealth and education, which influence interactions with a given technology and at the same time, shape how that technology evolves and develops.

The paper argues that there is an important gap between the perspectives of computer science and social science with regard to the user. It goes on to highlight the gap between these understandings of the user of early computer/digital technologies and ‘users’ of the web. The paper concludes by considering the implications of this discussion for the emerging discipline of Web Science.

2. THE COMPUTER SCIENCE ‘USER’

Within computer science, various software engineering design methodologies and formal processes are used to create and continuously improve software and systems. Since the early 1990’s, partly in response to the introduction of object oriented programming, more intricate processes have been developed, and there has been a shift to user-centric models [10] aimed at reducing costs and production time and increasing software reliability and integration. The waterfall model [1], first released in 1956, suggested that software should be completed in steps, starting from the analysis of the problem, gathering (user) requirements, designing and building it, and then finally releasing it. Modifications by Royce noted how crucial ‘customer’ involvement was in successful design [14]. Later the Spiral Model [4] based on the idea “start small, think big” allowed these steps to be revisited – principally by introducing a prototype per iteration cycle.

The extent to which the user should interact with technologies is never fully explored in these models, but it is significant that these approaches gave rise to the creation of early user manuals [4], suggesting that users needed to be trained (and that by implication a problem with using a technology might not be the software malfunctioning, but the user.)

Other models such as the V Model [7] and the collection of design models that fall under the Rapid Application Development (RAD) methodology all share some common characteristics, such as prototyping and interactive processes, and claim to focus on the interaction between user and designer. Agile Software Development [12] and Joint Application Development [15] are two such models, based on the RAD methodology, which both attempt to get at the users’ needs and requirements by using techniques such as workshops and acceptance testing.

Early computing systems were often designed with a clearly delineated (probably small, often technically-minded) set of users. By contrast web applications have potentially millions of members of a globally dispersed, heterogeneous user community. Nonetheless the historic computer science view of

users persists, and fails to address the nature of ‘a web user’, beyond a set of application design requirements.

3. THE SOCIAL SCIENCE ‘USER’

Social science provides a critique of the picture of the user as simply the passive holder of a set of requirements for a technology such as a software program (or the Web), and focuses attention on the relationship between technology and society. Woolgar argues [8] that technologies set parameters for the user, in essence ‘configuring the user’ to perform the tasks that it was designed to do. Although users appear free to do what they want with the technology, they can only do so within the ‘interpretive context’ – namely the set of social relations that are tied to the technology. The Social Construction of Technology (SCOT) model expounded by Pinch and Bijker [3] attempts to uncover how these social relations play a role in shaping technology. One of the main concepts introduced from this way of seeing the world is that of ‘interpretative flexibility’ – the idea that different social groups (for example men and women) can view a technology in very different ways and this leads to different use and adaptation of the technology. Lindsay [11] notes that computer scientists tend to design technology for users that are envisioned in their own image, but this does not capture the ‘future user’ who is not static and who changes once the technology is released. Cowen [6] and Pursell [13], both feminist sociologists also argue that users are not a homogeneous category [6].

4. DISCUSSION AND CONCLUSIONS

There is then, an important gap between the computer science and social science framings of the user. The aim of our work is not argue that one disciplinary approach is more valid than the other but rather to unpick the differences – the underlying epistemological and ontological assumptions that produce the gap – and to explore what we might gain from *closing* the gap in understanding.

Clearly, we need both Computer Science and Sociology. But how can they work together? Drawing upon the work of Robert Straus, who discusses the relationship between sociology and medicine [16], we can conceptualize the difference between ‘sociology of the web’ and ‘sociology in the web’, the later providing a radical new synthesis of techniques for the development of the Web. The structure of this new methodology will not resemble anything like a traditional software development model, drawing upon the epistemological and ontological understandings of both Computer Science and Sociology. Rather it will appreciate that society and technology are equally important within the development process. A co-constitutional approach that combines the technical knowledge of Computer Science, and the sociological knowledge of Sociology will emphasise the importance of a ‘pro- human’ Web and practically address the Web Foundation’s grand aims of “reducing poverty and conflict, improving healthcare and education”¹.

4. ACKNOWLEDGMENTS

This work is funded by the Web Science Doctoral Training Centre at the University of Southampton EP/G036926/1.

5. REFERENCES

1. Benington, H. D. (1987) Production of large computer programs. *ICSE '87: Proceedings of the 9th international conference on Software Engineering*. Monterey: IEEE Computer Society Press, pp. 299-310.
2. Berners-Lee, T., Hall, W., Hendler, J., O’Hara, K., Shadbolt, N. (2006) *A Framework for Web Science*. Foundations and Trends in Web Science, Vol. 1, pp. 1-130.
3. Bijker W.E., and Pinch., T. (1987) *The Social Construction of Technological Systems*. s.l. : MIT Press.
4. Boehm, Barry W. (1986) *A spiral model of software development and*. 4, 1986, A spiral model of software development and, Vol. II, pp. 22-42.
5. Bromley, H. (1994) *The Social Chicken and the Technological Egg: Computing and the technology/society divide*. [ed.] Wiley. 1st Edition.
6. Cowan R. (1987) The consumption junction: A proposal for research strategies in the sociology of technology. In W. Bijker et al. *The Social Construction of Technological Systems*. MIT Press, 1987.
7. Forsberg, K. and Mooz, H. (1991) *The relationship of system engineering to the project cycle*. NCOSE. Chattanooga
8. Grint, K. and Woolgar, S. (1997) *The Machine at Work: Technology, Work and Organisation*. 1st Edition. Polity Press.
9. Halford, S., Pope, C. and Carr, L. (2010) *A Manifesto for Web Science*. Second International Conference on Web Science, WebSci10. Raleigh, NC.
10. Hanning, Y., Tanni, H., and H., Jun. (2008) *Research on Agile Development Methodology of Service-Oriented Personalized Software*. International Conference on Computer Science and Software Engineering. 1075-1078.
11. Lindsay, C. (2003) From the shadows: Users as designers, producers, marketers, distributors, and technical support. In N. Oodshorn and T. Pinch. *How Users Matter: Th co-construction of users and Technology*. Cambridge: MIT Press, 2003.
12. Martin, R.C. (2003) *Agile Software Development: Principles, Patterns, and Practices*. New Jersey: Prentice Hall, 2003.
13. Pursell. C. (2001) Feminism and the rethinking of the history of technology. [book auth.] Schiebinger L. *Science, Medicine and Technology in the 20th Century*. s.l.: Princeton University Press.
14. Royce, W. W. (1987) Managing the development of large software systems: concepts and techniques. *ICSE '87: Proceedings of the 9th international conference on Software Engineering*. Monterey: IEEE Press, pp. 328-338.
15. Yatco, M.C. (1999) *Joint Application Design/Development*. University of Missouri.
16. Straus, R. (1957) The Nature and Status of Medical Sociology, *American Sociological Review*, 22(2), 200-204.

¹ <http://www.webfoundation.org/vision>