human relations
THE TAVISTOCK INSTITUTE

From female computers to male comput♂rs: Or why there are so few women writing algorithms and developing software

human relations
2021, Vol. 74(8) 1296–1326

DOI: 10.1177/0018726720914723
journals.sagepub.com/home/hum
SAGE

Rana Tassabehji
University of Bath, UK

Nancy Harding
University of Bath, UK

Hugh Lee
Kedge Business School, France

Carine Dominguez-Pery
University Grenoble Alpes, France

Abstract

Software development is one of the few professions in Europe and the USA from which women are disappearing. Current explanations range from unproven assumptions that women cannot write algorithms to insights into the misogynistic culture of this profession. This article argues these explanations are inadequate, and illuminates how forms of masculinity constituted within software development put women in the ambivalent position of being *either* female *or* a coder, but not both. Using a poststructural theoretical position to analyse materials from a qualitative, interview-based study, we identified three constitutive ontologies of the person circulating within the profession. The Comput♂r is presumptively male and can merge with the machine, although a subset, Geeks, cannot demerge from it. The Human, presumptively female, can communicate with people but not the machine. The Ideal developer claims the

Corresponding author:
Nancy Harding, School of Management, University of Bath, Quarry Road, Bath, BA2 7AY, UK.
Email: H.N.Harding@bath.ac.uk

best of both, that is, adept at writing algorithms and communicating with people. These ontologies are informed by a theory of the body circulating within software development whose norms are unattainable by women. Female bodies are envisaged as 'flesh', and male bodies as a futuristic merger of body and machine. This Janus-faced theory excludes female developers from practising their profession.

Keywords
Algorithms, embodiment, feminism, masculinities, software development, the body

In the 1970s, women held only 10% of jobs in the professions, a proportion that has since increased remarkably, as we outline below. In software development (SD), in contrast, women's representation has fallen from a post-war peak to no more than 4% today (WISE, 2015). Software developers write the programmes that operate computers and make new technologies possible (Madsen and Leiblein, 2015). They are fundamental to organizations in what is often called 'the fourth industrial revolution' (Schwab, 2017). The disappearance of women from SD is of concern not only because of an ethics of equality and diversity in the present, but also because technologies are intimately interconnected with future societies (Jackson et al., 2002). There is increasing awareness of how software programmes may have gender and other biases built into them, and concerns have been expressed that the gender(s) embedded in software may perpetuate inequalities in future e-societies (Perez, 2019; Wajcman, 2007). The cause, in part, lies in implicit assumptions about gender that software developers build into algorithms (Bano, 2018; BBC News, 2018; Wachter-Boettcher, 2017). It is therefore important both ethically and strategically to ensure women materially influence and contribute actively to writing software.

Research into causes of women's absence from SD offers two competing explanations. The first presumes that men have a 'natural' proclivity for writing algorithms that women lack. Similar arguments were made about medicine, law and other professions in the 1980s, but women's entry into these professions in large numbers disproves such essentialist arguments. The second is that the profession has a hostile, macho culture that deters women. Again, the professions in the 1970s and 1980s were hostile to women, treating them with disdain and disrespect, but women overcame such hostility (Atkinson et al., 2015). Numerous initiatives aiming to increase the representation of women in SD fail because women recruited into the profession tend not to stay (WISE, 2019). Given that women have successfully entered and remained within other professions that initially fought against their ingress, is there something peculiar to SD that inhibits women's participation in this particular occupation? The study we report on here aimed to answer this question. Its focus was upon exploring what makes it difficult for women who have started careers as developers to remain in those careers. We explored what it is like to work in the largely 'woman-free' SD workplace, aiming to understand what makes it difficult for women to stay in the profession. Research has shown that this workplace is misogynistic (Howcroft and Trauth, 2008; Lennon and Whitford, 1994), but drawing a cause/effect relation between misogyny and the absence of women, whilst important, is insufficient.

Rather, better understanding is required of how such a culture can be constituted and maintained, and how it works by, through and upon men as well as women. Most research in this area has focused on women, so little is known about male software developers, how masculinities are constituted within the profession, how those masculinities are powerful enough to exclude women and, indeed, how they may rely upon the exclusion of women for their constitution. Our study therefore asked: what forms of masculinity are constituted by, and maintained within, the SD workplace, and how do these forms of masculinity work to exclude women? We included both men and women, as each is central to processes governing masculinities (Connell and Messerschmidt, 2005).

Our objectives were to: understand what forms of masculinity are constituted in and through the absence of women from SD; develop understanding of how they deter female developers; and grasp how interplays of femininities and masculinities constitute an androcentric professional culture.

We used a 'strategic essentialist' approach (Spivak and Harasym, 2014) and carried out an exploratory, in-depth qualitative study that suggests that masculinities within SD are constituted within and through an exclusionary 'lay' or local theory of the body as a merger of flesh and machine. This body is Janus-like, incorporating within itself outdated and outmoded notions of the female that it imports into the technological future towards which it gazes, rendering the term 'female developer' an oxymoron. In exploring a profession little studied in organization studies, this article makes the following contributions. It demonstrates the importance of understanding how local, or lay, theories influence organizational interactions, offers new insights into the subtle ways in which women can be deterred from the technological professions, and provides a warning about feminism's need to be prepared to fight battles one might have thought were already won.

We next discuss women and the SD profession in more depth, before turning to the study itself.

Women in the professions but not in software development

Women held only 10% of professional jobs in the mid-1970s in the UK when the first Equal Opportunities legislation was passed. Today they comprise 45% of the medical profession (Medical Women's Federation, n.d.), 48% of academics (HESA, 2018), and 47% of the legal profession, but remain underrepresented in senior positions and on boards of directors (O'Neil et al., 2008). Similarly, in the US in 2010, women made up 46.7% of the US workforce and held 51.5% of management, professional and related positions but only 15% of Fortune 500 leadership positions (Catalyst, 2016). Women's success in entering the professions has been hard won, and they continue to experience considerable disadvantages, earning less than men for work of equal value, and struggling to conform with the demands of the archetypal male career path (Tomlinson et al., 2013).

The situation is quite different in the Science, Technology, Engineering and Mathematics (STEM) subjects (WEF, 2016). In 2014, women made up just 12.8% of the UK STEM workforce as a whole and 15% of the ICT sector specifically, showing a slow but rising trajectory over the previous 2 years, from 12.6% of the total in 2012.

The figures continued to rise to 2017 (WISE, 2018). With regard to ICT professions specifically, the numbers of women in this workforce increased by 3.5% between 2012 and 2014 (to 114,494), but male employment increased by 7%, to 722,681 (WISE, 2015). The 2017 statistics showed 'very positive' trends, with more women working in core STEM industries than ever before, but still they numbered merely 172,411, or 17% of the total (WISE, 2018). In 2019, WISE reported that the proportions of women working in ICT had fallen by 1% in the previous year, suggesting a 'leaky pipeline' in which initiatives to recruit women into technical IT careers fail because women do not stay after being recruited.

In SD, only around 4% of the profession are female (Ratcliffe, 2015). This is a vitally important sector because all digital technology in the 21st century is enabled and controlled by software designed and constructed by developers/programmers/engineers (Rapanotti and Hall, 2016). They design the structure and content of software applications, databases and web pages, and write computer code necessary to create and implement systems software.[1]

The 'leaky pipeline' of women in STEM subjects (WISE, 2019) has a long history in SD. In 1994, Lennon and Whitford reported that women's participation in the IT professions was shrinking, as did Howcroft and Trauth in 2008. In the USA, the numbers of women reading computer science rose sharply during the 1970s so that they comprised one-third (34%) of students in 1984, but then declined sharply (Quoctrung Bui/NPR). In France, women's representation is similarly much lower than men's and, as in the UK, continues to decrease (Tandon, 2012).

So, SD contrasts sharply with other, formerly male-dominated, professions. Women had worked in computer science and programming from the field's inception but their presence has been, literally, airbrushed out. Photos taken during the Second World War[2] show women working on the vast computers of the day, but male images were later superimposed upon them. Until 1945, the term 'computer' defined a human (usually female) who carried out calculations – who 'computed'. Hardware development was men's work; programming was women's. After 1945, 'computer' came to refer to a machine, and staff who 'computed', increasingly men, were labelled 'operators' (Stevens, 2009).

The IT professions recruit very few women, and few of those who *are* recruited stay for long (Scott et al., 2018). We classify previous research into this issue according to the theory of gender that informs the studies. One category regards both gender and technology as fixed, essentialized categories (Adam et al., 2004). It understands gender as a binary variable requiring little or no interpretation (Howcroft and Trauth, 2008; Trauth, 2013). For example, research into technology acceptance and use divides the world into rigidly differentiated 'men' and 'women' (Venkatesh et al., 2012). This presumes women are fundamentally incapable of understanding SD. Various media implicitly inform women of their deficiencies, reinforcing such beliefs (Ridley and Young, 2012).

Another category rejects essentializing theories that locate the reason for women's absence in their biology. Scholars 'consider the deeper, underlying reasons for women's absence from the technical sphere . . . which goes beyond the traditional commentary of "add more women"' (Adam et al., 2006: xxxviii). Studies located in this constructionist

perspective argue SD's overwhelmingly masculine culture grants men a power of genitalia-speak that excludes women's voices (Harvey, 1997). Misogynistic attitudes are deeply embedded in recruitment practices, organizational structures, language, images and symbols in dot-com start-ups (Tapia, 2006). France's 1990s computer industry was defeminized through subtle tactics to persuade women that 'true' women worked in 'communication' roles more 'suited' to them (Stevens, 2009). Such studies suggest IT's culture militates against women (Von Hellens et al., 2012) because it is insular, self-reinforcing, intimidating, alienating, discriminatory and misogynistic (Kiss, 2015; Tapia, 2006). Those comparatively few women who enter the field tend to quickly depart.

A limitation of previous research is its focus only upon women. For example, Stevens (2009) explored how women were *ex*cluded, without analysing how men were *in*cluded. Adya and Kaiser's (2005) research into girls' choice of careers in IT shows the important influence of families, especially fathers (Adya and Kaiser, 2005), but no comparison is made with boys. Men appear in these studies only indirectly, in references made by women research participants, such as the 'old school' boss whose 'feeling was that women should be home' (Trauth and Howcroft, 2006: 279), sexist pranks by male colleagues (Trauth and Howcroft, 2006: 284), or banter deliberately designed to appal female colleagues (Adam et al., 2006: 374). Males remain the largely unmarked, unexplored but agentive Other in studies of the gender imbalance in SD and IT. There is an implication that all men working in SD are misogynistic and thrive in the misogynistic culture they themselves constitute.

Is this so? Without further research, that question cannot be answered: the gendering of the SD profession in all its complexity needs to be understood. As Hearn (2019: 54–55) warns, too little is yet understood about 'the possible empirical, theoretical and political relations of the conceptual categories of (wo)men, (fe)male, and masculinities', and the 'social category of men' constitutes a 'social category of power' that requires deconstruction. In pursuit of such understanding, our study focused on the constitution of masculinities within SD's largely woman-free domain. We refer to masculinities in the plural because of the long-standing awareness that masculinity is plastic and takes multiple forms (Connell and Messerschmidt, 2005; Knights, 2019). To paraphrase Butler (1997), the terms 'man' and 'woman' are polysemous.

We locate our study within a theoretical context that regards gender not as biologically given but performatively constituted in interaction (Butler, 1990, 1993). Our focus was thus upon language, discourses and materialities (Butler, 2015). Through an interview-based study, we aimed to record the language through which software developers describe and understand the work they do, the people with whom they work and the gendered nature of the work, and identify the discourses that make possible the languages used in that talk, references to materialities within that talk, and forms of gendered selves that are thereby performatively constituted. The study does not explore the broader contextual issues that may influence women's entry into the profession, such as childhood, family and educational experiences, nor how industry-specific practices such as human resource management strategies may contribute. Our intention was to 'drill down' to local level, to identify forms of masculinity constituted in an organizational environment that repels female developers so much that they move on.

In summary, this study explores why the SD profession is so inimical to women. We focus upon the constitution of the previously-unexamined forms of masculinity constituted within that workplace, to understand their power to exclude female colleagues.

Methodology and methods

The study's overarching theoretical location is poststructuralism, in which persons and genders are understood to be fluid and performatively constituted within norms and through interaction (Butler, 1993). There are no fixed categories such as 'woman' and 'man' in this perspective, so we adopt a political location of strategic essentialism (Spivak and Harasym, 2014) that licenses the use of essentialist categories for political purposes in anti-essentialist feminist research. We use the terms 'male-' and 'female-' speaking subjects (Fotaki and Harding, 2013), or put 'female' and 'male' in quotation marks.

Poststructural perspectives oppose textbooks' assumptions that research interviews – the most widely-used form of data-gathering in qualitative research (Holstein and Gubrium, 1995) – are naturally occurring meetings between stable, coherent and autonomous individuals who can report the truths of their experiences (Brinkmann, 2016; Weedon, 1987). Poststructural theorists understand that 'interviewees' engage in processes of self-making at the site of the interview. They draw upon norms and discourses of 'their' organizations during these 'conversations with a purpose' (Robson, 2002), rendering interviews in some ways mimetic of the day-to-day/moment-to-moment encounters that participants 'live' (Harding, 2007). Both interviewers and interviewees are themselves constituted (come into being as individual(ized) female- or male-speaking subjects) during each on-going moment in which 'ontological politics' (that is, *who* can exist) (Mol, 1999) governs processes of individuation (the emergence of a self as seemingly an individual) (Clough, 2008). Participants are multiple, relational and continuously becoming; they do not pre-exist what they say but are constituted (come into being) through the acts of speaking, interpellation and recognition that 'make' 'the interview'. It follows that there are no such things as 'truths' that can be 'gathered' in interviews; interviewees are not repositories of knowledge but constructors of knowledge in collaboration with interviewers (Holstein and Gubrium, 1995).

From this perspective, interviews are important sites for understanding the constitution of subjects, subjectivities and subjectification (Mazzei and Jackson, 2012). They are sites within discursive practices in which 'things said' are points of departure for inquiry into the mechanisms, procedures and processes at work in the production of identities (Bonham and Bacchi, 2017). It follows that interview spaces are linked to, and interact with, many other sites within discursive practices, helping 'researchers' understand how 'subjects' (here those writing software and those researching them) emerge as male- and female-speaking 'I's through entanglements within multiple discourses and mechanisms. Asking, 'can you describe your work as a software developer?' calls participants to differentiate and scrutinize some elements of their lives and not others, and draw on the discursive practices that inform those particular elements of their lives and not others. Invitations to 'tell me about a typical working day as a software developer' interpellate (Butler, 1997) speakers as developers. Participants 'recognize' each other (you the software developer/me the academic – me the software developer/you the academic), and through such acts of recognition constitute their respective identities (software developer/researcher).

In sum, within interviews interviewer/ees co-constitute their identities *and* their respective workplaces (Harding, 2007) as they relate experiences, thoughts, feelings, languages, narratives and stories. These are all entangled within discourses that establish the limits of narrative possibility, and that occlude, or make impossible, other potentials.

The interview*ers* were female-speaking subjects who, in the on-going flux of the interview, performatively constituted identities as both academics and 'female'. This has implications for reflexivity. Our poststructuralist stance challenges the basis of reflexivity, both ontologically (there can be no self with or about or on which to reflect) and philosophically (who, then, is the self that is asked to stand back to reflect on the self?). However, *practically*, it must be acknowledged that the questions asked (see Appendix), even though open-ended, positioned participants in ways that limited possibilities for responding. That interviewers were 'female' may have invoked defences in 'male' participants against fears of allegations of collusion in sustaining SD as no place for women. Repeating this study using male-speaking interviewer-subjects could be useful. Finally, materialities of space and place may influence gendering, so gender performances in interview spaces may differ from those in the software laboratory.

We turn now to data analysis. Poststructuralism understands interview talk as located within discourses that make both speech and speaker possible (Bonham and Bacchi, 2017; St Pierre, 2011); thus, 'data analysis' consists of identifying constitutive discourses (Brinkmann, 2016) and what those discourses constitute. Data reduction's focus on words is problematic for poststructuralist researchers who understand that words *make* rather than *reflect* reality. This problem of theoretical incommensurability has been somewhat overcome by Fairclough's (1995) three-dimensional framework for critical discourse analysis that starts with a traditional textual analysis. Further support is found in Gioia et al.'s (2013) influential approach to data analysis that is built on 'the basic assumption that the organizational world is socially constructed' (Gioia et al., 2013: 17). Data reduction, from this perspective, requires systematically identifying the words and phrases participants use when making statements, before exploring what discourses make those statements sayable, and finally analysing what is made possible within and through those discourses. Data analysis was therefore guided by three questions arising from the study's aims and objectives: (i) What language do software developers use when responding to questions about SD?; (ii) What discourses of gender make possible the languages they use in that talk?; and (iii) What forms of gendered selves are thereby performatively constituted? We adapted Gioia et al.'s (2013) three-stage process of data analysis, deviating from its theoretical assumptions but retaining its structure. The process is illustrated in Figure 1.

Stage 1 identifies first-order concepts. This involves identification, collection and collation to impose homogeneity upon interviewees' heterogeneous utterances. We deviated from this classic data reduction approach by focusing on forms of language used by speakers, doing this by dividing the transcripts according to the (essentialized) biological sex of participants to explore how women talk about women, men and technologies, and how men talk about women, men and technologies.

We did this manually, distributing the scripts between us to organize the first-order concepts into six files of statements of 'men's' and women's' talk about themselves, the 'opposite sex', and technologies. Table 1 illustrates the language they used.

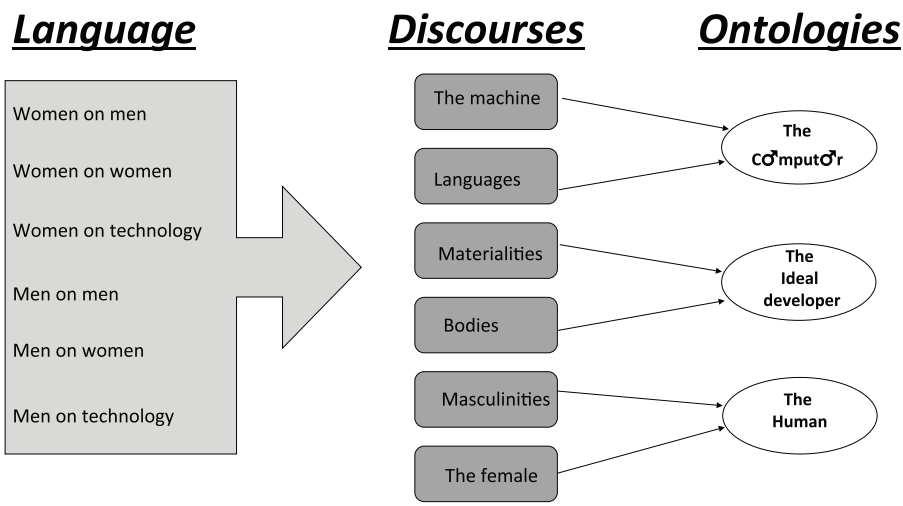

Figure 1. Mapping the process of data analysis.

Stage 2 traditionally involves deriving themes from concepts through reducing first-order concepts to second-order themes (Gioia et al., 2013). At this stage, the process moves from summarizing to analysis, or from 'informant-centric' to 'researcher-centric'. Academic insights come to the fore in 'surfac[ing] concepts and relationships that might escape the awareness of the informants' (Gioia et al., 2013: 17). Our study required exploration of discourses that made concepts speakable. We drew on Fairclough's (1993: 138) approach that links the study of spoken or written language to 'the totality of discursive practices of an institution, and relationships between them'. In Gioia et al.'s (2013) terms, this involved exploring spaces between boxes to allow 'leaps' from concepts to themes or (for us) discourses. This involves immersion in the 'theoretical realm' (Gioia et al., 2013: 20). What distinguishes discourse analysis from Gioia's perspective is analysis of statements to identify normative frameworks and the discourses within which they are located. For example, a 'female' software developer's statement that, 'I like nice clothes, I like make-up. I like wearing y'know and I like . . . so and I think it's quite good for girls to see a role model that is . . . a "girly girl" who's into technology', is coded in Stage 1 as 'woman talking about women' and in Stage 2 is related to discourses of gender and embodiment. A 'male' developer's statement that 'People can be quite nasty, defensive and nasty in IT, like that, yeah they can be . . . "why's he asking?" yeah and a lot of IT people go around saying "oh what he did was *crap* [emphasis]"', coded as 'men talking about men', in Stage 2, is related to discourses of masculinity. We occupied this 'space between' codes and discourses for much of Stage 2, immersing ourselves in the six files' numerous concepts and engaging in wide-ranging discussions. This seemingly logical progression through stages is bedevilled by ideas, metaphors and concepts that are present in the transcripts or emerge during early stages of analysis; these can stimulate ideas and intimations that send the analysis down various paths that must be confirmed, rejected, supplemented and complemented as the analysis unfolds

Table 1. Illustrative examples of words used by respondents talking about men, women and technology.

Women on men: 'As with pretty much all software development companies, it's probably 90% male'; 'Men/techies tend to have similar experiences and similar viewpoints on things and they all sort of agree with each other and then slap each other on the back and go down the pub'; 'I sat through so many presentations where they told us our target market for smartphones was men. It was men who were interested in technology, and . . . we spent months, years arguing that women like technology, what you've got to do is make it useful'; 'I think you can find in IT especially in development work, erm . . . guys that are erm . . . that are in a little bit of difficulty with social relations erm'.
Women on women: 'The problem with technology and software development is there aren't enough women'; 'My feeling is that a woman has more rigor in her work. She is more rigorous . . . I feel that women are more, maybe not perfectionist, but they like doing a good work'; 'Very few women, but the women that worked there on the engineering side had exactly the same perspective [as the men]'; 'A more deeper understanding of the processes that are around, a little bit more intuition maybe . . . and you have some social intelligence? . . . or emotional intelligence'.
Women on technology: 'We are here to create . . . build something, we are engineers to build something. We trust each other, as people talking the same language'; 'Sometimes we need to create a new item or to change one, in which case it's IT, not modelling. We would go inside the item and write codes . . . So coding. This time it's not C++, it's QML. It's IT logic. FOR loops, IF . . . really an IT language'; 'For the most part I go digital as much as I can. I have a lot of tech – I've got a mobile, I've got an iPad and a high spec PC and even my book is an ebook, y'know'; 'I'd like to sell a work technique that enables creation of better software, closer to human intelligence'.
Men on men: 'I can see . . . one of the downsides of really, really insular, normally male guys, are not necessarily very socially skilled, not very good at organizing and co-ordinating and motivating'; 'I think there is some autism going on . . . it's very much like sort of machismo sort of male stock broker type things'; 'Another guy, can't remember his name . . . what was great about him was he was technically minded person, but he had time for you, listened, supported you'; 'He's doing incredible things, sometimes, at the cost of being a nice human being, he's kind of blunt . . . He's just . . . he has such computational power in the brain, it is just mind blowing'.
Men on women: 'They said they were techies, but they weren't very good'; 'They are not very good coders'; 'You see 15 plus permanent people, in a room and you don't see a woman in sight . . . it becomes er quite feral, becomes quite feral, yeah'; 'They are basically . . . very determined ladies, y'know very determined, self-assured, yeah they do become a bit . . . a bit like lads at times, yeah'; 'They kept the group well organized . . . and they were much better at the project management side of it'; 'They weren't as good on the techie side . . . they would often rely on me or other good developers for the techie side'; 'To look feminine or to wear make-up, for example, is a kind of a flag signal that you are not a good developer, because if you were a good developer, you wouldn't care about how you look and this is so dumb'.
Men on technology: 'Technology is quite unforgiving if you leave it long enough, it's kind of hard to get back into it'; 'I think you have to be very structured and very logical in your thinking, I think you have to be very right brain, very analytical, very mathematical'; 'I have an engineer's brain really, always have done . . . that means I can understand how complex pieces of equipment work from end to end, I understand which drives which and cause and effect and how to translate this into a working piece of machinery'; '[what characteristics makes a good technologist?] Hah . . . not being so technical and being technical at the same time'.

Table 2. Illustrative examples of words mapped into six discourses.

The machine	[Techies] spend their whole life . . . with their headphones on. Full nerds . . . work at night . . . in the dark . . . alone. Perfectly happy to communicate on computers from the safety of their screen but will not talk. [Becoming] a genius in IT [involves] being disconnected from the world. You need to be a bit strange to do those kinds of jobs . . . to the exclusion of all else.
Languages	[Their work is] writing code for 8 hours and [after work] writing code is their hobby. The geek developer is . . . withdrawn, [without] a capacity for communication. The dialogue with the machine . . . the machine and the mechanics, is . . . masculine. A lot of engineers do not know how to write a correct sentence.
Materialities	[Job involves] sit[ting] there and code for hours and hours and hours on end [types on table]. It's pretty much people surrounded by about 3 or 4 computers [each]. [You can] hear a pin drop where they are all just very [types on the desk] focused on their coding. A . . . lot of people . . . just don't like paper, they don't see the point.
Bodies	You have this image of this geek eating pizza and drawing women. Advice to women going into technology: get used to wearing dowdy clothes. We're . . . after your brains. Just have to put my earphones on and just [typing on table].
Masculinities	The dialogue with the machine . . . the machine and the mechanics is . . . quite masculine. Code has to be written by a guy [because] guys are very much more logical than women. Geek was generally an unhealthy-looking person who . . . played a lot of games, didn't see much sunlight. We [university students] were true geeks – not much social skills . . . and very awkward towards women.
The female	Being able to understand, absorb, empathise . . . that skill base is more . . . feminine. The natural need to have an interaction with others is more important for woman . . . Women are really good communicators . . . have emotional intelligence. IT men are quite disorganized and are not very good communicators, and . . . that's where the women's strengths are. But they are not very good coders. [Women] weren't as good on the techie side . . . they would often rely on me or other good developers for the techie side.

(Gioia et al., 2013: 20). Numerous possible discourses were posited, explored and discussed before we concurred on six dominant discourses: the machine, languages, materialities, bodies, masculinities and the female. Table 2 shows ways in which words mapped onto discourses.

Where Gioia et al. (2013) take the resultant second-order concepts, refine and consolidate them into 'Aggregate (Theoretical) Dimensions' (Dacin et al., 2010), Stage 3

Table 3. Stages in the data analysis.

Stage	Focus on	Method	Product
1	Language	Logical, line-by-line coding of statements	Six thematic files of concepts
2	Discourses	Analysis of what discourses may inform each statement through immersion in the concepts, discussion and exploration	Six dominant discourses
3	Ontologies	Conceptual leap based on interpretation and insight	Three ontologies

for us involved exploring the constitutive work of those six discourses. We asked what ontological regimes – categories into which participants organize human actors (Hui, 2016) – are made possible within and through them. We apply the labels 'Comput♂r', 'Human' and 'Ideal developer' to the three ontologies we finally identified. The Comput♂r arises from accounts of how developers do their work. Comput♂rs can merge with the machine, but have a subgroup, the (presumptively male) 'Geek' that can neither demerge from the machine nor communicate with people. The Human is presumptively female, embodied flesh that cannot merge with the machine but interacts easily with people. The Ideal developer can merge with and demerge from the machine, feels emotions and can communicate with people. Geek and Human are the Ideal developer's necessary, constitutive others. Participants rarely, if ever, used these terms: they emerged through a 'conceptual leap' (Gioia et al., 2013: 22) of the type essential if interpretation is to go beyond descriptive, journalistic reporting to offer insightful interpretation of participants' voices.

In summary, analysis of interview materials had three stages, as Table 3 shows.

Fieldwork

This study is a joint project between researchers in the UK and France. Seeking depth rather than breadth of understanding, we deliberately limited the number of participants. Purposive snowball sampling was used to recruit equal numbers of French and British participants, and equal numbers of 'men' and 'women'. The majority were found through various networks, including visitors to the project's website, via an editorial discussing the project in a specialist magazine, and following the first author's invitation to speak at an industrial conference on women in IT. Two participants were recommended by interviewees. We recruited both 'men' and 'women', not only because masculinities cannot be known without their constitutive other but also because 'women' can occupy masculine subject positions (Halberstam, 2019). After interviewing six 'men' and six 'women' from both the UK and France (in 24 interviews), we were satisfied the 'data were saturated'. There was no aim to make cross-country comparisons. Interviews lasted between 46 and 120 minutes, with only three finishing in less than 1 hour. (See Appendix for interview schedule). All interviews were fully transcribed, resulting in transcripts exceeding 250,000 words. The first author carried out all UK and seven of the French interviews. The remaining five were carried out by French colleagues. Interviews conducted

in English were transcribed by the interviewer. French-language interviews were transcribed in French and then translated into English by a French-speaking translator. Fieldwork took place in 2015, extending over a period of 6 months because finding participants often proved difficult. Table 4 summarizes biographical details of the participants and the fieldwork.

Outcomes of the analysis: Presenting the three ontologies

In what follows, we weave statements made within interview spaces into narratives. These are both inside (what was spoken at the time of the interview) and outside (accounts of/from the workplace) the interview room; this doubled position illuminates the language that, working through discourses and norms, constitutes the three ontologies, or participants' performative descriptions of who exists within SD. We begin with the two 'masculine' identities of, firstly, the Comput♂r, using the male symbol that takes the place of an 'o' in this gendered neologism, and then the Ideal developer.

The Comput♂r

Asked to describe their work, developers gave accounts suggestive of merging with the machine through 'plugging themselves in' to write code all day. Coding 'is a head down, know your job, that's what I've been tasked to do, I've got to build this, and I'll just sit and build it' (Poppy). Note the personal pronoun in this 'female' developer's self-description: she identifies as a Comput♂r. Developers put on headphones and focus on computer and keyboard, shutting themselves off from the external world. As Tracy says, 'if I've got my earphones on it means I am concentrating on something that's really complicated'; the earphones signal 'don't talk to me . . . I'm doing something really important' (Richard). Helen, positioning herself within the 'we' of developers, says, it is:

> . . . almost cultural that we don't talk unless somebody . . . is obviously not looking at their computer screen. So . . . if you want to talk to somebody you type in chat [a web interface] . . . even if they're in the same office, even if they're sitting next to you.

Developers work in silence apart from 'the odd swear . . . "oh shit"' (Thomas), and 'some days . . . you hardly speak to anybody all day' (Tracy). Actually speaking starts to feel 'a bit strange' (Helen).

The accounts are rich with descriptions of the language of code: 'Writing code, reading code, developing code, testing code' is easy (Arnaud). Brains or minds become computers: 'constantly, all day long we have an amount of information that we keep in our brain uh . . . It's like a chain of neurones, we have the chain in our mind' (Bettine). Note again, how, Bettine, like Poppy, Helen and some other 'female' participants locate themselves within the 'we' of developers who merge with the machine (Comput♂rs).

Immersion in coding language can make the transition to day-to-day language difficult. Bettine, demonstrating a facility with writing algorithms that others deny 'women' can possess, describes it thus: writing in words is 'pretty but . . . it goes kind of disorganized . . . Whereas when it's translated into code, there is an architecture, there is a

Table 4. Participants.

Profile	Pseudonym	Age	Interview details	Interviewee background	Sample selection process	Interview language
Male, France	Arnaud	30	2hrs	Computer engineer & lecturer	French network	English
	Danton	32	2hrs	Developer now CTO	French network	French[b]
	Etienne	24	1hr 15 Skype	Back-end developer	UK network	English
	Rayce	56	1hr	CEO of IT services company	French network	French[b]
	Thierry	26	46 mins Skype[a]	Open source programmer	UK Network Snowball	English
	Theon	27	1hr 11 Skype	Software engineer/front-end developer	UK Network Snowball	English
Male, UK	Thomas	43	1hr 46	Programmer/consultant	UK Network	English
	Richard	43	1hr 15	Developer/consultant		English
	John	50	1hr 46	Engineer, developer		English
	Finlay	42	1hr 16	CEO & games developer		English
	Callum	45	47 mins	Software developer/project manager		English
	Dylan	45	1hr 40	Programmer		English
Female, France	Bettine	25	30 & 90 mins	Programmer	UK Twitter	French
	Chantell	35	1hr 13 Skype	Computer engineer	French network	French[b]
	Desarae	28	1hr 16	ERP IT systems consultant also runs French coding school	French network	English
	Majori	30	1hr 40 Skype[a]	Database developer	UK Twitter	English
	Trinetta	44	1hr 20 Skype[a]	IT project manager (originally British)	UK Twitter	English
	Valeraine	48	1hr 20	IT and database developer	French network	French[b]
Female, UK	Helen	Mid 30s	1hr	Engineer now software project manager	UK Network	English
	Tracy	Mid 20s	1hr 15	Web developer, team leader		English
	Angela	46	1hr 5	Gamer & technology writer		English
	Millie	Early 40s	54 mins	Digital project manager local government		English
	Poppy	Mid 40s	1hr 45	Programmer		English
	Daisy	65	1hr 30	Programmer		English

Developer, programmer and engineer are all professions that involve writing code. All names are pseudonyms; unless stated, all interviews are face-to-face and sampling purposive.
[a]Video; [b]translated into English.

structure'. Arnaud quoted his girlfriend: emails written by technologists are 'written in some kind of language that is not French that is not English that is some kind of rubbish language . . . that is [not] comprehensible by other human beings'. Even communicating in coding languages can be difficult 'because the guys were expert in one system but couldn't work in another' (Valeraine).

The constitutive language used by developers conjures a workplace peopled by actors who use headphones to screen out sounds, isolating themselves within their own worlds – communicating via keyboards and computer screens in the language of algorithms. Technology replaces human senses: keyboards (and thus fingers) replace tongues; computer screens (and eyes that read them) substitute for ears. Sentient actors become machines, hearing nothing save through headphones, saying nothing save through keyboards and seeing nothing save through monitors. Using language rich in references to bodies and other materialities, interviewees describe humans and equipment collapsing into each other, entanglements of body, dress, keyboard, monitor, headphones, desk and chair: developers are the interface with the hardware into which they are plugged. All the 'men' and at least half the 'women' interviewed describe themselves in language suggesting the machine is their appendage and they are the machine's appendage. Importantly, assimilation is not complete – they can be called back into the human world. As Bettine articulated, 'we have a job where we need someone to wake us up now and then to tell us, "Hey you have to talk now" and to remind us why we have to talk as well because we're so much in something else that we think "why are we talking?"'

Speakers use this ability to demerge from the machine to distinguish themselves from the 'extreme computer scientists' (Etienne) who enter so 'deeply into their own world' (Richard) they forget how to be human. We use the term 'Geek' to describe this subgroup. They are 'sweaty guys in t-shirts and pimples . . . men in dark rooms . . . who drink a lot of Coke and don't get out much' (John). They 'are very messy . . . around them is complete disorganisation and chaos' (Finlay); they are, indeed, 'the messiest people going' (Poppy). They come to life in participants' descriptions as socially inept *men* who live off pizzas, ignore appearance and personal hygiene, and are essentially 'brains' (Richard) whose bodies are encumbrances. 'Disconnected from the world' (Desarae) they are 'rather withdrawn' (Danton) and sometimes 'incredibly incredibly shy' (Helen). Geeks 'are perfectly happy to communicate on computers from the safety of their screen but will not talk' (Helen); they communicate via computer even when 'speaking' to the person on the next desk (Thomas). They cannot write in non-coding language 'without three or four mistakes per sentence' (Thomas). Their spoken language is little better: Helen describes how managers have to 'decipher' what Geeks say, translating it 'into legible English to communicate to customers'. Discussions are hindered by the absence of 'a type of language they understood' (Trinetta).

'Male' and 'female' participants talked about workplaces populated by Geeks who are so merged into the machine they have forgotten how to speak in anything other than algorithms. Interviewees emphatically denied they were Geeks, but the main interviewer reported feeling a strong sense that male participants were describing *themselves* rather than this other, this not-I, who supposedly did not come to the interview room. There being no supportive 'proof', that reported feeling raises the question of whether or not 'Geeks' have a material existence. Could the Geek be a negative ontology or

necessary fiction, as Rachel and Woolgar (1995) suggest of technology in technological settings – that is, a metaphysical object lacking a material existence but integral to identity work in SD?

What we *do* know is that participants made many direct references to the body, such as 'brains' and 'fingers', but indirect references too, as it is embodied persons who put on headphones, type on keyboards and somehow forget they are human. This ontology points towards a theory of the body circulating within SD that is predicated upon the intimate relationships of worker and machine, in which flesh and machine merge and demerge.

The Ideal developer

This ontology describes 'male' participants who draw on discourses that allow them to theorize that they can, in our terminology, both merge with the machine *and* (unlike the Geek) demerge from it, and communicate not only with the machine, as Comput♂rs do, but also with humans, as the Human does (see below). Highly rational, they share with the Human a capacity for emotions. That is, Ideal developers claim for themselves characteristics they admire in both Comput♂r and Human, while rejecting their shortcomings.

Ideal developers profess a 'special' ability to speak both machine and human languages; they can compute but also emote, and can move between machine and social worlds. They are 'Geek hybrids' who 'can communicate' embodied-person to embodied-person, and have 'a very strong technical knowledge' (Danton). Arnaud explains how they are in one person 'two kinds of people' that are 'not being so technical and being technical at the same time'. These 'techie people who are human' can 'verbalise change, ideas, collaborate, communicate; all those things are what make [us] special' (Richard). Their claim to be passionate about both technology and humanity is encapsulated by Theon – 'I read books, I live technology . . . I love it' – who emphasizes needing interaction with people: 'Every time there is someone talking about something, I have to listen to . . . I have to be part of it to know everything'. Finlay claims two sides to himself: he is 'creative . . . I pushed myself out of the box because I was always a creative person' but at the same time is 'logical'. Dylan similarly described in detail his immersion in computers but also his wide-ranging interests: 'I wanted to know everything about everything even though it wasn't technical'. In their own estimation, Ideal developers are rare. Arnaud exemplifies this, claiming membership of a minority who go 'to a lot of concerts', are cultured and who combine technical skills with pleasure in people's company.

That is, the Ideal developer valorizes the ability to become immersed in both machine and flesh. They claim bilingualism, as exemplified by Dylan, who positions himself as a combination of Human (a communicator) and someone who can merge with and understand the machine: 'so I would fix it, explain it in a non-mocking way, you know, whereas the rest of the [technologists] would huff and puff and shake their heads like they [non-technologists] were imbeciles'. Technical brilliance is transcendent for them – they admire and worship it – but only in developers who combine technical and social skills: 'Geeks' are dismissed as 'dicks' (Dylan).

Kelan (2008) shows that men's claiming of what are traditionally regarded as 'women's' skills elevates those skills from 'normal' (for women) to 'ideal' (when practised by men). We see this in Ideal developers who distinguish themselves from 'Geeks' through their ability to use the supposedly human, female skills of communication. Theon, for example, describes 'the most amazing guy I have met in IT'. Superb at writing software programmes, he is also 'running everywhere, he is talking to everyone, he knows everyone, he is curious, he's asking . . . questions all the time . . .'. Dylan similarly admires someone who could move between the sentient and non-sentient worlds. He would make jokes 'but once he was in the zone, that was it – zoom' until he 'would say "right I've had enough of this, come on, coffee, let's go" and he would round us up and go off to the canteen'. It is on such individuals that Ideal developers model themselves.

When talking about 'women', Ideal developers refer to other male developers' aggressive and sexist stance towards 'women' (see below), but distance themselves from such behaviours through expressing much admiration for women's abilities to communicate. This may be a function of meeting 'female' interviewers (again, the body intervenes as the primary mark for distinguishing 'male' and 'female'): politeness could militate against overtly sexist behaviours. Alternatively, it may be that blatant misogyny is practised by some, but not all, male developers. Ethnographic studies are required.

Ideal developers thus claim for themselves (constituting their identities through so doing) the best of both Comput♂r and Human and the failings of neither. Through such language, and its circulation of the norms contained within the discourses that govern the possibilities of such language, they constitute a masculinity that is fleshy, machinic, emotional and social. This is an embodied masculinity. Direct references to the body are unnecessary; bodies are inevitably present even though absent from thought (Leder, 1990). Ideal developers going to concerts or rounding up colleagues (embodied developer selves) are inevitably embodied. The ontology of the Human qualifies their claims.

The Human

The third ontology, the Human, describes an embodied, flesh-and-blood person working in the 'female dominated' 'fluffy stuff, the humanities' rather than the sharply-dichotomized 'male dominated' 'techie side' (Poppy). In these accounts, Humans appear firstly as the opposite of the Comput♂r: they cannot merge with the machine. Secondly, all Humans were 'women'. 'Men' who leave the industry, about whom nothing is known, may also be denigrated as 'Human', hence the label 'Human' rather than, say, 'Female'. Much of what follows refers to female-speaking subjects as 'the female Human'. We saw when exploring the ontology of the Comput♂r how female-speaking subjects, including Poppy, Helen and Bettine, used language indistinguishable from 'male' participants when describing their work. Here, all the male- and some female-speaking subjects articulated a belief that 'women' cannot code.

Asked why there are so few women in SD, participants defined 'the woman' in three different ways, with that composite definition the rationale for her absence. They, firstly, distinguish sharply between male/female – 'a man and a woman are not the same' (Rayce). Male technologists have brains that, conflated with computers, are 'built', whereas women have minds that are 'natural'. As Callum said, 'I think it's a mechanical

thing, I've always had a mechanical mind'. Tracy similarly thinks, 'Guys are very much more logical than women in general', because of 'the kind of way a man's mind is built'. The ideal technologist is a 'he' who 'has such computational power in the brain, it is just mind blowing' (Etienne), whereas women 'think a lot more with their souls and with their hearts' (Finlay). 'Women' 'think differently' from men (John). They are understood to have a 'natural need to have an interaction with others' (Danton), so their thinking is suffused with emotion:

> Women think a lot more with their souls and with their hearts . . . I'm not saying men don't have empathy, I'm a terribly emotional . . . erm . . . but there is a different level of empathy between men and women because we are quite different (Finlay).

Some female participants concurred in this essentialized discourse; like 'men', they regard brains as software determining people's actions. As Trinetta said, 'women are unable to code, maybe there is something inbuilt in their brain . . .'.

The female Human, secondly, is equated with her body. Echoing studies arguing that SD has an oppressive, sexualized culture that excludes women, Finlay observed that 'obviously there's the sex thing as well . . . it's quite hard for a woman to handle [the sexualized culture]. So, 'If you're a woman and you're pretty and tall and have all the model features you're going to be looked at as an object' (Callum). Women's presumed concern with appearance is described as career-inhibiting: 'to look feminine . . . is a kind of a flag signal that you are not a good developer, because if you were a good developer you wouldn't care about how you look' (Etienne). Thus, women wanting careers in SD are advised, 'Don't wear too short skirts' (John). This is 'a real stereotype. Girls who are developers aren't sexy' (Bettine). To survive, 'women' have to 'adapt your behaviour' and become 'one of the boys' (Daisy) through hiding 'our femininity because when we wear a skirt . . . boys . . . made us not feel comfortable being a woman' (Majori).

The female Human, thirdly, is understood to possess 'soft skills' for communicating with people. 'Most of the women I have worked with', said Helen, 'have shown more management and leadership and more communication [skills than men]'. That is, 'women are *brilliant* at management and project management . . . because women are better at being social than men' (Finlay). They are 'really good communicators . . . really good at talking to people' (Tracy) because, in Desarae's words, women are more 'intuitive' and possess 'social' and 'emotional intelligence'. Indeed, women's presumed need to talk excludes them from coding:

> I need to communicate with my team because I'm a sociable person . . . Coding type development teams need quiet and they don't talk to people . . . and perhaps that's another reason why we don't find many girls doing it (Trinetta).

Thus, 'men code [work with technology] and women decode [what clients have said]' (Arnaud), or, as Danton said: 'men tend to focus on technical, mechanical aspects . . . women can put things into perspective, and interpret the situation with their own sensibility'. A simple formulation is embedded in these accounts: all 'men' working in IT are capable of merging with the machine, but only a minority of 'women' have that capacity; ergo, 'women' are the residual category, the Human.

Geeks find human language difficult, but Humans are deemed to fail at coding languages. Tracy, for example, says: 'I think that code has to be written by a guy really because . . . guys are very much more logical than women in general'. For Thomas, 'women' are 'not very good coders', so were 'permanently just relying on you [the male speaker] to direct them how to do something'. Danton had 'found indeed that it was harder for women to follow the strict rules of coding. They didn't really understand their importance'. Tracy agrees: she finds coding 'monotonous', and knows her 'limits' so sometimes needs help. Females are described as lacking interest in IT: they are 'not attracted to the nuts and bolts of it' (Callum) or, as Theon says: 'there are no girls in IT . . . because they don't want to do it'. They construct an account of the impossibility of being both a technologist and embodied as female. As Bettine said, 'we're told that we're not real women' or, in Chantelle's description of a female project manager, she is 'a woman but she's kind of a man too . . . It's not a woman as we often see'.

In some ways, it is unsurprising that some 'women' talk about women in ways that echo 'men's' talk: they swim within the same discourses that make possible but limit what can be thought and said. Indeed, becoming female may require positioning themselves as 'male's' constitutive other who cannot code. However, female-speaking subjects sometimes draw on an alternative discourse governing 'women's' perceptions of achieving professional success (Atkinson et al., 2015): the need to be better than 'men'. Women have to fight 'so hard to be accepted in an industry that's mostly male dominated (Helen), and have 'to work twice as much if we are a woman' (Chantelle), to prove 'that women are brilliant . . . that we are as strong (Bettine).

The language used by all the 'men' and some of the 'women' in these interviews, even though not overtly articulating misogynistic attitudes, is redolent of outdated perceptions of the female. It is not mere description: the performative power of language turns it into a law (Butler, 1997) that creates the very thing it governs. If 'women' are understood to have 'minds' rather than computer-like brains, they cannot write software, and if they cannot write software, then anything they do produce is judged substandard, as the history of artificial intelligence (AI) illustrates. Early processing programmes given female-sounding pseudonyms such as 'ELIZA' were judged inferior to similar programmes given more masculine-sounding names such as 'Racter' (Phillips et al., 2016). Contemporary developers may be similarly biased (Phillips et al., 2016). Our study suggests that in SD becoming coherently constituted as female requires that 'women' have poor skills at, or lack interest in, coding. Not only deemed incapable of speaking machine language, they are so equated with their bodies that they cannot *dis*embody and merge with the machine. The normative 'woman' is an embodied, Human communicator.

This Human is materialized within and through very traditional norms regarding femininity (Brewis et al., 1997): she must be warm, caring, embodied and articulate, imminent to her body and thus to nature/not-machine (Benn and Gaus, 1983). But our study complicates such presumptions. Not all 'women' concur in this discourse (Stevens, 2009). Some, like Trinetta, do, but others did not negate their own skills. We therefore posit that the Human 'woman' may be, like the Geek, an organizing fantasy necessary to the constitution of masculinities within SD, but here its performative effect is very different. That is, some 'women' conform with the normative injunction that they become incapable of coding, whereas others seem impervious to such norms.

A closer focus on Tracy suggests something more complex. In describing herself as having 'focus, definitely being able to block everything out and just concentrate on the job in hand' but also 'able to communicate really well', she appears to identify as the archetypal Ideal developer. It is curious, therefore, that she also said, 'I think that code has to be written by a guy really because . . . guys are very much more logical than women in general', and that she found coding 'monotonous' and her abilities limited.

Tracy's job involves a mix of web design, SD and management (she is head of a team of four). Her father bought her her first computer when she was 11, 'and I just remember spending hours [types on table] typing all the codes in and I thought it was brilliant, y'know, to get the little man to do the little dance'. In school and university, she had been one of a small minority of women studying computing. She socialized with 'male' fellow students: 'Maybe that is the way I am I suppose, for the most part I do generally get on better with guys'. Her language suggests she is a Comput♂r: 'I was more into computers, I can't remember really exactly what sparked it, I guess I've just got that kind of brain. I am quite analytical, and I like rules'. Early in the interview she emphasized her pleasure in coding: 'I really like the coding and the hands-on stuff' and, later, 'I like doing HTML, I like doing CSS, I like doing JavaScript. I will do some cold fusion stuff . . . I need the variety [of working with different languages] to keep me interested'. Her enjoyment of coding means she is not ambitious for further promotion:

> I don't want to go any higher than I am now. Because if I go any higher . . . then I won't be able to do any coding anymore. Because the more senior you get, the less coding you do.

She describes herself as gregarious, extrovert and a good communicator. She appears to occupy the masculine subject position of 'Ideal developer'.

But there are contradictions throughout her account. Compare the above with her statement that:

> I think that code has to be written by a guy really because they are very fo . . . very dri . . . erm focused on the endpoint . . . I think guys are very much more logical than women in general . . . the kind of way a man's mind is built more than a woman erm . . . yeah.

There were ambiguities in how she described her work. Asked specifically if there were differences between her work and that of male colleagues, she said: 'Not really, no, because I have been taught by guys. Pretty much everything I've learned, I've learned from . . . the guys that I've worked with, so no'. But she also stated that, 'Well anything I do is always prettier than what the guys do [laughs]. I think girls have a better eye for colour and design', implying that she, like other 'girls', brings something different from men to their coding work. Her pride in her ability to code contrasted with, later, her statements that her strengths lie in:

> mak[ing] . . . the website pretty and functional and structured. [After] the guys [have] . . . done the [types on desk] codey bit, I will then go in and put the style and layout and that's what I prefer to do, to be honest.

She stated that, 'I am definitely one of them people that won't give up. I'll get there eventually – it might take me three years, but I will get there [laughs]. I will figure that piece of code out', and in almost the same breath observed, for the second time in the interview, that she knew:

> when it's time to say, 'I don't know what I am doing' and go and ask for help . . . I am quite happy to hold my hand up and go to my boss and say 'can you help me'.

These contradictions are insightful. Tracy's job requires occupation of two subject positions: when 'I am doing systems work, then I just have to put my earphones on and just [typing on table] [laughs]', that is, she merges with the machine. But she also works in an office where 'there's much more banter going on, it's much more friendly y'know', and she is immersed in human-to-human interactions. She moves between the subject positions of Comput♂r and Human. In one she can, like her 'male' colleagues, celebrate her skills at coding: she is constituted as masculine. In the other she performatively constitutes that coherent female identity without which life becomes unliveable (Butler, 1990). She is subjected to and subjectified by (Butler, 1997) local norms that require 'the female' to be someone who cannot communicate well with the machine (she must not write good code) but must communicate well with people. If so, then previous (essentialist) research that argues women cannot develop coding skills is correct, but only insofar as the norms governing SD prohibit them from doing so. Similarly, previous research blaming SD's misogynistic culture for women's absence is also correct, but only insofar as misogyny is expressed not 'merely' through overt behaviours: it circulates within discourses that make possible professional identities and require 'the woman' to be the Ideal developer's constitutive other, unable to write code.

We next explore this lay or local theory encapsulated in these ontologies, to help better understand Tracy's ambivalence and explain why women do not stay in SD.

Discussion: From lay to academic theory

This study aimed to understand what is peculiar to SD that renders it impervious to that ingress of women evident in other professions in the past 50 years. Because women enter the profession but tend not to stay in it, we sought insights in the workplace itself, through exploring the working lives of both men and (some of the few) women working in it. We specifically explored the constitution of masculinities in this largely woman-free environment. Our analysis suggests a lay or local theory of embodied masculinity circulates within SD that understands 'Ideal developers' as having bodies that can both meld with and de-merge from the machine. They can speak both human and computing languages, and possess not only machine-like brains but also social skills. It is a *masculine* body: female bodies cannot comply with its governing norms. Previous studies argued women's inabilities either to code or to resist a culture of toxic masculinity caused their absence from SD. This study illuminates how the SD workplace is governed by norms that *require* women to be poor coders, while the masculinity with which they cannot comply is not overt but contained within a very 'subtle' (Bevan and Gatrell, 2017) lay theory of embodied masculinity.

We next interrogate this lay theory to understand its constitutive power. We show how in its future focus it echoes academic thought, but how, Janus-like, it incorporates into itself historical concepts of 'the woman'. This shows that strategies for increasing the representation of women in SD will fail if they do no more than train women to code: women also need to understand this lay theory and how to resist its norms. We start with the future-facing aspects of the theory.

Future-facing bodies

We begin by outlining the theoretical grounds for approaching bodies' materialities via the immaterial signifiers that issue from bodies as language. The influence in western thought of Descartes' attempt to separate body from 'mind', reducing bodies to dumb matter, necessitates this discussion. We draw on Butler's (2015) challenge to Descartes' heritage. She writes (Butler, 2015: 15) that philosophy founders 'time and again on the question of the body, it tends to separate what is called thinking from what is called sensing, from desire, passion, sexuality, and relations of dependency'; it is an artificial distinction with wide-ranging implications. Descartes' text, she argues, cannot do anything but figure bodies (in order to attempt to separate mind and body he has to develop a theory of the body), but in so doing he cannot reduce bodies to their figuration: bodies are not that which is discussed in words. This difference between referent and reference means, '[i]f the body is not reducible to its figuration or, indeed, to its conceptualization, and it cannot be said to be a mere effect of discourse, then what finally is it?' (Butler, 2015: 34). Bodies' ontological status must be *posed* using grammatical terms but cannot be answered in grammatical terms; because their materiality can be questioned only from an embodied position, bodies are separated from themselves, always beyond what can be expressed in or reduced to language. In short, discussions of bodies must inevitably be undertaken within a language that refers to, but cannot capture, flesh. Given that words are organization theorists' main research tools, this could preclude us from studying bodies and embodiment.

However, Butler's arguments suggest otherwise, for bodies are agentive: body and language are not separate and distinct. Bodies 'talk' and signify (Butler, 2015: 15), their speaking enabled because of a temporal lag between language and meaning in which sensing finds language that facilitates comprehension of what is sensed. In other words, what is felt by the body can be translated into language, albeit always inadequately. In terms of qualitative interviews, as our participants, for example, mimed typing on keyboards, or perhaps gazed in their mind's eye at the monitor that absorbs much of their waking attention, their bodies perhaps occupied two spaces at once: the chair in the interview space and the chair in their workplace. Mention of the workplace may conjure up sensations of being in that place; the mind seeks words to describe that remembered experience. When it returns, its language may carry no more than hints about its embodiedness, but the experience it remembers will have been inherently embodied.

In this study, we noted speakers' references to minds, brains, bodies and the paraphernalia that enable their work – those references suggest some form of collapse of the time/space distinction between interview room and workplace. This is emphasized by Butler's

(2015) insistence that embodiment occurs not only within and through those discourses it exceeds, but also within and through numerous other materialities:

> I am affected not just by this one other or a set of others, but by a world in which humans, institutions, and organic and inorganic processes all impress themselves upon this me . . . who is . . . as it were, in the 'hands' of institutions, discourses, environments, including technologies and life processes, handled by an organic and inorganic object field that exceeds the human. In this sense, 'I' am nowhere and nothing without the nonhuman (Butler, 2015: 7).

This was illuminated in this study in collapsed boundaries between comput♂rs' bodies and keyboards, monitors, headphones, etc. That is, there are no developers without the artefacts to which speakers referred during the interviews, but attempts at indicating such things always founder upon the rocks of language's inadequacies. In summary, Butler's arguments justify our inevitable and unavoidable recourse to using words to theorize bodies and embodiment, even though bodies remain always out of the reach of language.

Turning now to the lay theory of the ideal software developing body. This requires poststructuralist insights (Braidotti, 2012) to interpret bodies as not just pieces of meat awaiting enlivenment by an agentive mind, but as constituted through performative acts carried out within norms that, if transgressed, can result in abjection and unliveable lives (Butler, 1990, 1993). Thus, it is now 'widely recognized' that bodies are 'never *merely* physical' (Thanem and Knights, 2019: 3), and are constituted within complex matrices of materialities, discourses and affects (Butler, 2015).

Calls for studies of embodiment in organizations (Hockey and Allen-Collinson, 2009) have resulted in studies about how bodies are: materialized through 'body work' (Swan and Flowers (2018); achieve conformity with certain (imagined) expectations (e.g. Rajan-Rankin, 2018); involved in sense-making (Cunliffe and Coupland, 2012); foundational to organizational ethics (Pullen and Rhodes, 2014); required to thrive in extreme conditions (Yates, Riach and Johansson, 2018); and central to qualitative research (Thanem and Knights, 2019). Fotaki and Harding (2017) explore embodiment as experiential resistance. There is much to resist: female bodies are always-already pregnant, their careers inhibited by assumptions of actual, potential or non-maternity (Gatrell, 2008). Unfortunately, none of these studies help understand a lay theory in which flesh can merge with and de-merge from the machine. We turn to other disciplines' focus on technologies for insight.

In 1984, before the internet and hand-held computers, Turkle (1984, 2005) observed people who already thought of themselves both as computers and something more, something ineffably human. 'Where we once were rational animals', she wrote (Turkle, 1984: 285), 'now we are feeling computers, emotional machines', who project aspects of the human onto the computer/object'. In 2005, observing that computers were no longer objects but subjects, she asked 'what kinds of people are we becoming as we develop more and more intimate relationships with machines?' (Turkle, 2005: 294). Ideal developers' answers would refer to collapsing boundaries between flesh and machine, a response echoing that of technogenesis theorists. They argue that, in this 'present age of technocultural mutation' (Berardi, 2015: 11), patterns of thinking, behaving and feeling are transformed by 'the shift from mechanical to digital technologies [that have] provoked a mutation in the texture of human experience, and in the fabric of the world itself'

(Berardi, 2015: 12). Or, as Hayles (2012: 81) writes, 'contemporary technogenesis' breeds 'coordinated transformations' of humans and technologies. That is, contemporary immersion in digital worlds meant technologies were no longer regarded as controllable outcomes of social action but as driving forces behind the social, putting the very concept of 'the human' into question.

Materialities are understood to be agentive and fluid; theories of a new unconscious are emerging, and 'the mind' is seen to transcend the body's limits and extend into the environment (Hayles, 2012). Selves are 'located neither inside nor outside the brain/body, but [are] instead constantly enacted in-between brains, bodies and things and [are] thus irreducible to any of these three elements taken in isolation' (Malafouris, 2008: 1997). Malafouris' aim to dissolve 'the ontological bounds of the res cogitans' (Malafouris, 2008: 1997) is extended in Hansen's (2004, 2012) haptic theory of 'sensorimotor power'. In 'today's electronic technosphere' (Hansen, 2004: 10), bodies are immersed in new media and technics in bodies through fluid crossings of virtual and physical realms (Hansen, 2012: 2). Echoing Hui's (2016) understanding of computing's digitalization of material objects and materialization of digital objects, Hansen argues that flesh becomes the primary sensing organ (Hansen, 2004) and affects, rather than sight, the dominant register. A 'shift from a dominant ocularcentrist aesthetic to a haptic aesthetic rooted in embodied affectivity' (Hansen, 2004: 11) means embodied, sensed experiences contrast with earlier cognitive sense-making. In short, Hansen (2004) argues against the thesis of a disembodying of humans in the service of computers (the ontology of the Geek, in this study), and enunciates a thesis of the embodiment of technics, where 'the virtual' is 'an embodied form of living' (Hansen, 2012: x), a mixed reality that 'foregrounds the constitutive or ontological role of the body in giving birth to the world' (Hansen, 2012: 5).

The software developers in this study articulate these complex and sophisticated theories of merger of flesh and machine, albeit in everyday language. This is perhaps unsurprising: developers work in a 'zone of ontological artifice' (Fuller and Goffey, 2017, location 660) creating models of things that otherwise do not exist. Perhaps they would concur with Hansen's emphasis on 'the truly remarkable capacity human beings have for generating (mental) images out of digital information' (Hansen, 2012: 70), notwithstanding that the lay theory perceives that this ability is the province of 'men'.

Thus, software developers (lay theorists) and philosophers of technology (academic theorists) alike now speak of flesh/machine mergers. Software developers, however, suggest something of which philosophers appear unaware: embedded within these performative theories are hierarchies of superior 'male' and inferior 'female' constitutive others.

The backward-looking theory of 'the woman'

To facilitate insight into lay theory's understanding that women are inferior because they cannot code, we begin this section by explaining coding languages, of which there are thousands, each of which instructs computers what to do. Coders use these languages to break actions into specific logical instructions to signify precise actions. Sentences are reduced to components of nouns and verbs, often abbreviated, and lack adjectives,

adverbs, conjunctives, etc. Words without a specific purpose are redundant. These languages, located within a 'tacit ontology of world as sequences of events' (Fuller and Goffey, 2017, location 625), constitute new kinds of materialities that 'take shape on a screen or hide in the back end of a computer program' (Hui, 2016: 1), where technical, social and economic systems converge and integrate.

This suggests coding languages are deracinated, their authors requiring only motor skills and the capacity to speak (in algorithms), think (in algorithms) and be(come) (through algorithms): the archetypal Geek. This assumption is wrong. Cox and McLean (2013: xiii) emphasize that although algorithm-writing subjects are active agents in writing code, they are immersed in cultural meanings generated 'not . . . from intentionality or source code as such, but from the complex interplay of forces involved in the encoding and decoding of texts and programs' (Cox and McLean, 2013: xiii), forces including the wider social and cultural environment. Indeed, software circulates as 'sites saturated in machinings of various kinds: interpretation which conjoins knowing and transforming, perceiving and doing; interaction with its fusing of experience and technique . . .' (Fuller, 2017, location 456). Butler's (1997) exploration of speech's agency thus applies to coding languages: subjects are constituted in language *and code* (Cox and McLean, 2013: 6), and coders incorporate 'the social' into the languages they develop and use. A major aspect of the social embedded in such languages is gender.

Mackenzie's (2005) exploration of the performativity of coding language first observed this. Using the origins of the widely used Linux operating system as his example, he shows how conceptualizations of 'the woman' became stabilized in the IT sector. In early 1991, Linus Torvalds posted calls on newsgroups for participants:

> Do you pine for the nice days of minix-1.1, when men were men and wrote their own device drivers? . . . Are you finding it frustrating when everything works on minix? No more all-nighters to get a nifty program working? Then this post might be just for you. (Torvalds, 1991, quoted in Mackenzie, 2005: 87).

This call, for volunteers who are 'all man' and enjoy 'all-nighters', 'deeply ties' the Linux kernel 'to a gendered corporeal set of practices of programming work' (Mackenzie, 2005: 87) that, as Linux is 'ported' onto different platforms, is carried into new systems. Fuller's (2017) analysis of Linus's more recent works shows them to be sites of production where 'some of the conditions of software culture, derived both from computational forms and from cultural tendencies that are manifest as conventions realized in software, drive forms of work, communication and value-creation' (location 442). As these include gender, gender is inherent in the IT sector's 'complex temporalities' in which the 'past is enfolded into the present through skeuomorphs, details that were previously functional but have lost their functionality in a new technical ensemble' (Hayles, 2012: 89). That is, contemporary technologies contain a 'past [theory of gender] nestling inside present, present carrying the embryo of the future' (Hayles, 2012: 89).

Of the many concepts of 'the woman', which was imported into coding languages? A dominant understanding of 'the woman' available to Linus and his collaborators in 1991 was one being interrogated by second-wave feminism. 'She' was required to be a passive object of an active male gaze. Note its echo in our interviews, where Ideal developers

warn women not to dress in ways that mean they will be 'looked at'. De Beauvoir (1953/2015) had argued females' inferior status was located in males' supposed ability to transcend bodies, while 'women' remained immanent to their flesh. This study shows this assumption informing contemporary developers' lay theory, which conceptualizes 'women's' constitutional embodiment as a barrier to merger with the machine.

We similarly see how SD, that forward-focused profession, resonates with the heritage of an Enlightenment project that ushered in theories of a binary divide between the sexes, drawing a dichotomous and mechanistic Man/Nature relation. Under Descartes' influence, males' bodies were equated with machines and females' with Nature (Nast and Kobayashi, 1996). Nature (for which read 'woman') was constructed as passive materiality controlled by the rationalizing gaze and hand of science (for which, read 'man'). That is, 'reason has been defined in opposition to the feminine, such that it requires the exclusion, transcendence and even the domination of the feminine . . . characterized as the site of the irreducibly irrational particular and corporeal' (Alcoff, 1996: 14). Alcoff's definition remains pertinent to this study, even though what she wrote otherwise appears outdated. SD's lay theory conceives of the 'woman' as 'soft', 'fluffy', and good at emoting and communicating, characteristics valorized in males but not females. 'Women', as nature rather than culture, are precluded from machine/flesh mergers and are incapable of writing good code.

The lay theory of gendered embodiment that governs SD is dependent upon outdated perceptions of women thought to have fallen into disrepute, in contrast with the future-facing theories regarding flesh/machine merger. 'Woman' remains equated with her body, its flesh understood as unable to merge with the machine, her mind as something that cannot speak/write good code. Women entering this profession face the irresistible normative power of this theory: she learns she is incapable of working with new technologies.

Conclusion

This study sought to illuminate why the small numbers of women who make it through numerous barriers to become software developers do not stay in the profession. We suggest that a local or lay theory of the body governs the constitution of the Ideal developer. The male body can merge with and de-merge from the machine, and 'men' are highly fluent in both machine and human languages. The 'woman', as his constitutive other, is required to be incapable of merging with the machine or speaking machine languages. This lay theory is Janus-faced: it conforms with emerging academic theories of futuristic bodies, but at the same time absorbs within itself outdated theories of 'the woman'. It incorporates norms that require 'women' to be incapable of writing software: the woman who is good at writing software cannot be a woman. The 'female' developer thus occupies an untenable subject position: she is an impossibility. Given this, it is not surprising that women do not remain in the profession.

This study demonstrates the importance of understanding how local, or lay, theories influence organizational interactions. They may erect norms and rules that preclude 'women' or other others from conforming with or obeying them. This shows how tentative are the successes of some of feminism's struggles. In the case of SD, for women to develop successful and satisfying careers they require strategies that train them not only

in how to code, but also how to challenge dominant norms or belief systems that are the oxygen that allows (some) 'male' colleagues to flourish.

Acknowledgements

We express our sincere thanks to the three anonymous reviewers who took great pains in guiding this article to its final state, and especially to the associate editor, Tim Kuhn, for his patience and support.

Funding

The authors disclosed receipt of the following financial support for the research, authorship and/or publication of this article. This study was funded by the British Academy/Leverhulme Small Research Grants Scheme.

ORCID iD

Nancy Harding https://orcid.org/0000-0002-5225-2870

Notes

1 We use the term 'software development' to include all programming activities.

2 See 'World's first all-electronic programmable computer (ENIAC)', *Popular Science*, April, 1946, http://blog.modernmechanix.com/the-first-all-electronic-programmable-computer-eniac/1/#mmGal. Also http://www.columbia.edu/cu/computinghistory/eniac.html, and http://blog.modernmechanix.com/how-much-is-%E2%88%9B258916/.

Appendix: Interview schedule

1. Context

   Life and career history section of the interview: what led the participants to take up a career in technology – what were they like at school, where and when did their interest in technology emerge, how did they get into a technological career, what do they do now?

2. Working environment

   a. Can you describe a typical working day/week.
   b. What does the typical workplace of a technologist look like?
   c. What does your work space look like? Equipment, facilities, etc. Who do you share it with?
   d. How would you describe the culture of where you work?

3. The technologist

   a. What characteristics make a person good at SD?
   b. Do you have those characteristics?
   c. Can you describe two male technologists you have worked with you think are very good.

d. Can you describe two female technologists you have worked with that you think are very good.

4. Bringing about change

a. There is a lot in the press about the lack of women technologists and the push from government to get more women into STEM – what is your view of this?
b. Do you consider there to be a gender bias in the technology sector? Explain why.
c. Do you think there is a need to do anything to encourage more women technologists to get involved in development and design of emerging technology. If so what? If not why not?
d. What would your advice be to women wanting a career in SD?

References

Adam A, Howcroft D and Richardson H (2004) A decade of neglect: Reflecting on gender and IS. *New Technology, Work and Employment* 19(3): 222–240.

Adam A, Griffiths M, Keogh C, et al. (2006) Being an 'it' in IT: Gendered identities in IT work. *European Journal of Information Systems* 15(4): 368–378.

Adya M and Kaiser K (2005) Early determinants of women in the IT workforce: A model of girls' career choices. *Information Technology & People* 18(3): 230–259.

Alcoff LM (1996) *Feminist Theory and Social Science. BodySpace: Destabilizing Geographies of Gender and Sexuality*. London: Routledge, 13–27.

Atkinson C, Ford J, Harding N, et al. (2015) The expectations and aspirations of a late-career professional woman. *Work, Employment and Society* 29(6): 1019–1028.

Bano M (2018) *Artificial intelligence is demonstrating gender bias*. Kings College London New Centre. Available at: https://www.kcl.ac.uk/news/artificial-intelligence-is-demonstrating-gender-bias-and-its-our-fault (accessed 23 July 2019).

BBC News (2018) *Amazon scrapped 'sexist AI' tool*, 10 October. Available at: https://www.bbc.co.uk/news/technology-45809919 (accessed 23 July 2019).

Benn SI and Gaus GF (1983) *Public and Private in Social Life*. London: Croom Helm.

Berardi F (2015) *And: Phenomenology of the End*. South Pasadena, CA: Semiotext(e).

Bevan V and Gatrell C (2017) *Knowing her Place: Positioning Women in Science*. Cheltenham: Edward Elgar.

Bonham J and Bacchi C (2017) Cycling 'subjects' in ongoing-formation: The politics of interviews and interview analysis. *Journal of Sociology* 53(3): 687–703.

Braidotti R (2012) Becoming-world. In: Braidotti R, Hanafin P and Blaagaard BB (eds) *After Cosmopolitanism*. Abingdon: Routledge, 18–37.

Brewis J, Hampton MP and Linstead S (1997) Unpacking Priscilla: Subjectivity and identity in the organization of gendered appearance. *Human Relations* 50(10): 1275–1304.

Brinkmann S (2016) Methodological breaching experiments: Steps toward theorizing the qualitative interview. *Culture & Psychology* 22(4): 520–533.

Butler J (1990) *Gender Trouble*. New York, NY: Routledge.

Butler J (1993) *Bodies That Matter*. New York, NY: Routledge.

Butler J (1997) *Excitable Speech*. New York, NY: Routledge.

Butler J (2015) *Senses of the Subject*. New York, NY: Fordham.

Catalyst (2016) Statistical overview of women in the workforce, 6 April 2016. Available at: www.catalyst.org/knowledge/statistical-overview-women-workforce (accessed 23 July 2019).
Clough P (2008) The affective turn: Political economy, biomedia and bodies. *Theory Culture & Society* 25(1): 1–22.
Connell RW and Messerschmidt JW (2005) Hegemonic masculinity: Rethinking the concept. *Gender & Society* 19(6): 829–859.
Cox G and McLean A (2013) *Speaking Code: Coding as Aesthetic and Political Expression*. Cambridge, MA: MIT Press.
Cunliffe A and Coupland C (2012) From hero to villain to hero: Making experience sensible through embodied narrative sensemaking. *Human Relations* 65(1): 63–88.
Dacin MT, Munir K and Tracey P (2010) Formal dining at Cambridge colleges: Linking ritual performance and institutional maintenance. *Academy of Management Journal* 53(6): 1393–1418.
De Beauvoir S (1953/2015) *The Second Sex*. London: Vintage Books.
Fairclough N (1993) Critical discourse analysis and the marketization of public discourse: The universities. *Discourse & Society* 4(2): 133–168.
Fairclough N (1995) *Critical Discourse Analysis*. London: Longman.
Fotaki M and Harding N (2013) Lacan and sexual difference in organization and management theory: Towards a hysterical academy? *Organization* 20(2): 153–172.
Fotaki M and Harding N (2017) *Gender and the Organization: Women at Work in the 21st Century*. London: Routledge.
Fuller M (2017) *How To Be a Geek: Essays on the Culture of Software* (Kindle edn). Cambridge: Polity Press.
Fuller M and Goffey A (2017) The obscure objects of object orientation. In: Fuller M (2017) *How To Be a Geek: Essays on the Culture of Software* (Kindle edn). Cambridge: Polity Press, Chapter 1.
Gatrell C (2008) *Embodying Women's Work*. Maidenhead, UK: OUP.
Gioia DA, Corley KG and Hamilton AL (2013) Seeking qualitative rigor in inductive research: Notes on the Gioia methodology. *Organizational Research Methods* 16(1): 15–31.
Halberstam J (2019) *Female Masculinity*. Durham, NC: Duke University Press.
Hansen MB (2004) *New Philosophy for New Media*. Harvard, MA: MIT Press.
Hansen MBN (2012) *Bodies in Code: Interfaces with Digital Media*. New York: Routledge.
Harding N (2007) On Lacan and the 'becoming-ness' of organizations/selves. *Organization Studies* 28(11): 1761–1773.
Harvey L (1997) A genealogical exploration of gendered genres in IT cultures. *Information Systems Journal* 7(2): 153–172.
Hayles NK (2012) *How We Think*. Chicago, IL: University of Chicago Press.
Hearn J (2019) So what has been, is, and might be going on in studying men and masculinities? Some continuities and discontinuities. *Men and Masculinities* 22(1): 53–63.
HESA (2018) *Higher education staff statistics: UK, 2016/17*. Available at: https://www.hesa.ac.uk/news/18-01-2018/sfr248-higher-education-staff-statistics (accessed 23 July 2019).
Hockey J and Allen-Collinson J (2009) The sensorium at work: The sensory phenomenology of the working body. *The Sociological Review* 57(2): 217–239.
Holstein JA and Gubrium JF (1995) *The Active Interview*. Qualitative Research Methods, Vol. 37. Thousand Oaks, CA: SAGE.
Howcroft D and Trauth EM (2008) The implications of a critical agenda in gender and IS research. *Information Systems* 18: 185–202.
Hui Y (2016) *On the Existence of Digital Objects*. Minneapolis, MN: University of Minnesota Press.

Jackson MH, Poole MS and Kuhn T (2002) The social construction of technology in studies of the workplace. In: Lievrouw LA and Livingstone S (eds) *Handbook of New Media: Social Shaping and Consequences of ICTs*. London: SAGE, 236–253.

Kelan EK (2008) Emotions in a rational profession: The gendering of skills in ICT work. *Gender, Work & Organization* 15(1): 49–71.

Kiss J (2015) Women of 2015: Where are the cracks in the glass ceiling? Technology. *The Guardian* [Online]. Available at: http://www.theguardian.com/world/2015/dec/30/women-feminism-2015-glass-ceiling (accessed 20 December 2015).

Knights D (2019) Gender still at work: Interrogating identity in discourses and practices of masculinity. *Gender, Work & Organization* 26: 18–30.

Leder D (1990) *The Absent Body*. Chicago, IL: University of Chicago Press.

Lennon K and Whitford M (1994) Introduction. In: Lennon K and Whitford M (eds) *Knowing the Difference: Feminist Perspectives in Epistemology*. London: Routledge, 1–14.

Mackenzie A (2005) The performativity of code: Software and cultures of circulation. *Theory, Culture & Society* 22(1): 71–92.

Madsen TL and Leiblein MJ (2015) What factors affect the persistence of an innovation advantage? *Journal of Management Studies* 52(8): 1097–1127.

Malafouris L (2008) Between brains, bodies and things: Tectonoetic awareness and the extended self. *Philosophical Transactions: Biological Sciences* 363(1499): 1993–2002.

Mazzei LA and Jackson AY (2012) Complicating voice in a refusal to 'let participants speak for themselves'. *Qualitative Inquiry* 18(9): 745–751.

Medical Women's Federation (n.d.) *Facts and figures*. Available at: http://www.medicalwomens-federation.org.uk/about-us/facts-figures (accessed 18 March 2020).

Mol A (1999) Ontological politics: A word and some questions. *The Sociological Review* 47(S1): 44–89.

Nast H and Kobayashi A (1996) Re-corporealizing vision. In: Duncan N (ed.) *Bodyspace: Destabilizing Geographies of Gender and Sexuality*. London: Routledge, 75–93.

O'Neil D, Hopkins M and Bilmoria D (2008) Women's careers at the start of the 21st century. *Journal of Business Ethics* 80(4): 727–743.

Perez C (2019) *Invisible Women: Exposing Data Bias in a World Designed for Men*. London: Penguin.

Phillips A, Smith G, Cook M, et al. (2016) Feminism and procedural content generation: Toward a collaborative politics of computational creativity. *Digital Creativity* 27(1): 82–97.

Pullen A and Rhodes C (2014) Corporeal ethics and the politics of resistance in organizations. *Organization* 21(6): 782–796.

Quoctrung Bui/NPR (2014) cited in Henn S (2014) *When women stopped coding*. Available at: http://www.npr.org/sections/money/2014/10/21/357629765/when-women-stopped-coding (accessed 18 March 2020).

Rachel J and Woolgar S (1995) The discursive structure of the social-technical divide: The example of information systems development. *Sociological Review* 43(2): 251–273.

Rajan-Rankin S (2018) Invisible bodies and disembodied voices? Identity work, the body and embodiment in transnational service work. *Gender, Work & Organization* 25(1): 9–23.

Rapanotti L and Hall JG (2016) *An introduction to software development*. Open Learn, Open University (ebook). Available at: http://www.open.edu/openlearn/science-maths-technology/introduction-software-development/content-section-0 (accessed 18 March 2020).

Ratcliffe R (2015) I thought the tech world would be full of new voices and people. It hasn't happened. The number of women in the sector is tiny. It's such a waste. *The Observer*. [Online] Available at: http://www.theguardian.com/technology/2015/oct/18/martha-lane-fox-technology-women-computing (accessed 29 October 2015).

Ridley G and Young J (2012) Theoretical approaches to gender and IT: Examining some Australian evidence. *Information Systems Journal* 22(5): 355–373.
Robson C (2002) *Real World Research* (2nd edn). Oxford: Blackwell.
Schwab K (2017) *The Fourth Industrial Revolution*. London: Penguin.
Scott A, Klein FK, McAlear F, et al. (2018) *The leaky tech pipeline*. Available at: https://www.leakytechpipeline.com/wp-content/themes/kapor/pdf/KC18001_report_v6.pdf (accessed 18 March 2020).
Spivak GC and Harasym S (2014) *The Post-colonial Critic: Interviews, Strategies, Dialogues*. London: Routledge.
St Pierre EA (2011) Post qualitative research: The critique and the coming after. In: Denzin NK and Lincoln YS (eds) *The SAGE Handbook of Qualitative Research* (4th edn). Los Angeles, CA: SAGE, 611–626.
Stevens H (2009) The professional fate of woman engineers in the computer sciences: Unexpected reversals. *Sociologie du Travail* 51S: e15–e33.
Swan E and Flowers R (2018) Lasting impressions: Ethnic food tour guides and body work in southwestern Sydney. *Gender, Work & Organization* 25(1): 24–41.
Tandon N (2012) *A Bright Future in ICT Opportunities for a New Generation of Women*. Geneva, Switzerland: ITU United Nations.
Tapia AH (2006) Hostile work environment.com: Increasing participation of underrepresented groups. *Database for Advances in Information Systems* 37(4): 79–98.
Thanem T and Knights D (2019) *Embodied Research Methods*. London: SAGE.
Tomlinson J, Muzio D, Sommerlad H, et al. (2013) Structure, agency and career strategies of white women and black and minority ethnic individuals in the legal profession. *Human Relations* 66(2): 245–269.
Trauth EM (2013) The role of theory in gender and information systems research. *Information and Organization* 23: 277–293.
Trauth EM and Howcroft D (2006) Critical empirical research in IS: An example of gender and the IT workforce. *Information Technology & People* 19(3): 272–292.
Turkle S (1984, 2005) *The Second Self: Computers and the Human Spirit*. Cambridge, MA: MIT Press.
Venkatesh V, Thong JY and Xu X (2012) Consumer acceptance and use of information technology: Extending the unified theory of acceptance and use of technology. *Management Information Systems* 36(1): 157–178.
von Hellens L, Trauth EM and Fisher J (2012) Increasing the representation of women in the information technology professions: Research on interventions. *Information Systems Journal* 22(5): 343–353.
Wachter-Boettcher S (2017) *Technically Wrong: Sexist Apps, Biased Algorithms, and Other Threats of Toxic Tech*. New York, NY: W. W. Norton.
Wajcman J (2007) From women and technology to gendered technoscience. *Information, Communication & Society* 10(3): 287–298.
Weedon C (1987) *Feminist Practice and Poststructuralist Theory*. New York, NY: Basil Blackwell.
WEF (2016) *Where are the women in tech?* World Economic Forum Report. Available at: https://www.weforum.org/agenda/2016/04/where-are-the-women-in-computing/ (accessed 18 March 2020).
WISE (2015) *Women in Science, Technology, Engineering and Mathematics: The Talent Pipeline from Classroom to Boardroom*. London: WISE.
WISE (2018) *Women in STEM workforce 2017*. Available at: https://www.wisecampaign.org.uk/statistics/women-in-stem-workforce-2017/ (accessed 18 March 2020).

WISE (2019) *Workforce statistics – one million women in STEM in the UK*. Available at: https://www.wisecampaign.org.uk/statistics/2019-workforce-statistics-one-million-women-in-stem-in-the-uk/ (accessed 18 March 2020).

Yates S, Riach K and Johansson M (2018) Stress at work, gendered dys-appearance and the broken body in policing. *Gender, Work & Organization* 25(1): 91–105.

Rana Tassabehji is Visiting Fellow at the University of Bath School of Management. Her research interests include holistic approaches to managing information security, internet technology and supply chain management, reverse e-auctions, e-government, internet technologies, and gender in organizations and technologies. She has published widely in international journals, including *Journal of Business Ethics, International Journal of Operations & Production Management, Information Technology & People* and *Government Information Quarterly*. Her books include *Applying E-Commerce in Business* (London: SAGE). [Email: rt@tassabehji.co.uk]

Nancy Harding is Professor of Human Resource Management at the School of Management, University of Bath. Her research interests include working lives, gender, critical management studies and ways of researching and writing 'differently'. Her articles have appeared in journals including *Human Relations, Organization Studies, Organization, Journal of Management Studies* and *Gender, Work & Organization*. Her two sole-authored books explore the construction of management and the manager, and ways of being at work. Her joint-authored books are on bringing feminism into 21st-century management studies (with Marianna Fotaki), a critical perspective on leadership (with Jackie Ford and Mark Learmonth), and the social construction of dementia (with Colin Palfrey). [Email: h.n.harding@bath.ac.uk]

Hugh Lee is Assistant Professor at Kedge Business School, Bordeaux. His research explores business ethics, social responsibility, organization behaviour, diversity and inclusivity. He has published in journals including *Human Relations*, *Organization Studies* and *Public Administration*. [Email: hugh.lee@kedgebs.com]

Carine Dominguez-Péry is Professor at the University of Grenoble Alpes (UGA) – Grenoble IAE School of Management and member of the CERAG laboratory. Her research interests include information systems as strategic levers to create value (in particular in inter-organizational and/or service innovation contexts), knowledge management and digital ecosystems, human interactions with information technologies and societal challenges. She has published two joint-authored books, one exploring how management tools create value within organizations, and the second presenting the main authors in the Information Systems field. She has published in several journals, including *International Journal of Production Economics*, *Journal of Strategic Information Systems*, *Production Planning and Control* and *Revue Française de Gestion* and *Systèmes d'Information et Management*. [Email: carine.dominguez-pery@univ-grenoble-alpes.fr]