Exploring Creative Learning for the Internet of Things era

Alan F. Blackwell, Samuel Aaron

Computer Laboratory
Cambridge University
[Alan.Blackwell,Sam.Aaron]@cl.cam.ac.uk

Rachel Drury

Cambridge
rachel.drury@gmail.com

Keywords: POP-I.A. arts collaboration, POP-II.A. end-user

Abstract

We describe a study of a group of artists commissioned to create a new artwork involving programming with the Raspberry Pi computer. From an initial sample of 10 professional artists who mainly work with conventional media, 5 were selected for an intensive series of workshops leading to public presentations of the project and their work. The artists learned to program using the Sonic Pi environment that had recently been created for use in schools. During the project, the Sonic Pi language was also enhanced with new features in response to the artists' creative objectives. Throughout the project, data was collected to record the experiences of the artists, including initial self-efficacy questionnaires, reflective diaries, workshop evaluations, and focus group discussion.

1. Introduction

As low-cost computing devices become increasingly ubiquitous, the opportunity to customise, configure and assemble them into systems increasingly resembles a craft pursuit. The growing popularity of the hacker and maker movements as cultural movements means that programming, as a democratically accessible skill, is acquiring the status that might in previous times have been associated with carpentry, gardening, cookery, home decorating or scale model construction. Popular craft movements are pleasurable and creative pursuits, while also being associated with some degree of utility or public display. In most cases, popular crafts stand in relation to a body of professional craft skills - whether small-scale artisanal or large-scale engineering occupations. The social, practical and creative potential of such skills, together with their professional applications, makes them a natural focus for school curricula. Creative and practical craft skills are an important aspect of citizenship and participation in material culture.

2. Learning to Code as Creative Craft

The fact that low-cost computing is becoming more ubiquitous means that programming is becoming a craft skill of this kind - a creative material competence rather than simply an intellectual or scientific pursuit. We believe that the role of computing in the schools curriculum is becoming modified in recognition of this change. Ten years ago, educational usage of computers was almost wholly oriented toward consuming packaged services (word processors, presentation software etc), with the implicit assumption that the creation of new applications would be relevant only to the technocratic elite rather than the general population. Use of educational programming languages such as Logo and Basic had largely disappeared from the schools curriculum. However, recent popular movements such as Code Clubs, together with the Computing At Schools consortium, have reemphasised programming as a universal practical skill. The rising recognition of hack days, make spaces, and other creative technology buzzwords means that amateur and hobbyist engagement with programming is once again a rising trend.

This new popular interest in programming recapitulates to some extent the UK fashions of the early 1980s, when BBC series The Mighty Micro drew attention to newly available low cost technology, and the BBC Microcomputer offered a standard educational platform for home and school use. In the

Internet of Things era, a group of the early users and developers of the BBC Micro have reconvened[1], in a project that led to the launch and popular success of the low-cost Raspberry Pi computer. The Raspberry Pi Foundation, the educational charity responsible for that initiative, has been a partner in the research described here, in association with the development of Sonic Pi – an environment for creating live-coded music at a level of complexity that is suited to first-language teaching (Aaron & Blackwell 2013).

The implications of these trends for psychology of programming are firstly that "craft-like" creative practices already recognised among highly skilled professional artist-engineers (Woolford et al 2010, Blackwell 2013) are becoming more widely distributed among end-user programmers, and secondly that end-user programming is taking place in the context of a broader range of technical and material practices. Both of these tendencies involve technologies that bridge professional artists and hobby/enthusiasts, for example under the banner of the "maker movement". The preferred computing platforms among these communities include the Arduino, Microsoft Gadgeteer, ARM mBed, Beagleboard and of course the Raspberry Pi.

As noted by one reviewer of an earlier version of this paper, these research concerns are closely linked with those of the "Critical Making" community. We are aware of this relationship, and are continuing to explore it, especially in the context of an experimental graduate course being taught in Cambridge this summer, under the title "Critical coding: An introduction to digital design for researchers and graduate students in the humanities and social sciences"[2].

3. Research Context

In this paper we describe an observational study that was designed to explore individual experiences during creative learning of a technological "craft" by a defined end-user population. We had recently been developing school curriculum materials, and carrying out classroom lesson observations, using the Raspberry Pi computer running Sonic Pi (Aaron & Blackwell 2013, Burnard et al in press). In this new study, we were interested in observing adult users, also using the Raspberry Pi, but in a creative professional context rather than an educational setting. This builds on previous research into programming tools for professional artists that has been reported at PPIG (Church et al 2012, Blackwell & Collins 2005), and also on an earlier investigation of the craft practices of professional software developers who work in professional arts contexts (Woolford et al 2010).

The Defining Pi project was hosted by Wysing Arts Centre near Cambridge, and supported by a grant from Arts Council England. Ten established artists were invited by Wysing Arts Centre to participate in an experimental programme exploring ways in which the Raspberry Pi and Sonic Pi software might adapted or extended for use in creative arts contexts. The ten artists attended a half-day workshop in Cambridge, at which they were given a hands-on introduction to the Raspberry Pi and Sonic Pi. The ten were then invited to write brief proposals for a short research and development project, describing an artistic question or artwork that would be developed using these tools. Five of these proposals were selected by a judging panel, and were funded as artistic commissions. The funded artists worked for two months, during which they attended three further workshops to share their work in progress and receive technical assistance. The final outcomes of Defining Pi, including the work generated and the participants' experiences, were presented to the public at a range of talks and workshops representing different perspectives on creative technology: from popular academic audiences (the Cambridge Festival of Ideas) to visual arts (Wysing Arts Centre) technology enthusiasts (Cambridge Makespace) and performing arts (Cambridge Junction). Selected press coverage of these events, and of the project as a whole, is recorded in the URLs of Appendix D.

The four pieces that were presented at the final workshop of the project, and in various combinations to public audiences, were as follows:

[1] http://www.moviesandquotes.com/the-blues-brothers-1980/were-putting-the-band-back-together/
[2] http://www.digitalhumanities.cam.ac.uk/Methods/Criticalcoding

- Slow Scan Raspberry Pi: Rob Smith's project uses the Raspberry Pi, Sonic Pi and Raspberry Pi camera module to capture an image, encode it into audio and transmit it to a second computer that decodes the sound into an image, which builds up line by line as it is received. He wanted to make the transmission of data tangible to the viewer – the conversion of image data to sound that can be heard travelling through space seemed like a good way to do this.
- Kate Owen's starting point was looking at the movement of fingers typing on a computer keyboard and the parallel to a pianist's fingers on a piano. The final outcome is a live performance which is both visual and aural. The performer types the code of a Sonic Pi program, which is also projected for viewing by the audience. The executing code triggers the playing of sound patterns and recorded samples, including recordings of Kate's fingers hitting the keys and 'tap dancing', and of her voice reading out the file and folder names, alongside more staccato tapping sounds. The code can also be read as a visual poem based on the names of the directories and sound files embedded in it.
- Shapes & Things: Richard Healey's code picks an image at random from a collection of cut-out images of tropical plants that are hosted on the Raspberry Pi's SD card. Sonic Pi then places it at random on the screen, repeating the process infinitely, filling the screen with foliage, in which layers of information appear as dense and exotic as the primeval jungles of Henri Rousseau.
- A transcription of worldwide and cosmic events: Chooc Ly Tan saw the potential for data to become a medium, through which to create accidental associations of text, sound and imageries – as part of a score. She used Sonic Pi to collect: data sourced from the Internet (specifically selected forums, etc); information that surrounds the environment and, the physical phenomena were keys to investigate notions of synchrony (of events) and complexity (of systems), present in the universe. This data was presented via a series of real and computer generated images and sounds.

4. Data collection

Research data was collected throughout the project, in the following forms:

Introductory workshop (10 artists + 4 facilitators)

- At the start of the introductory workshop the artists completed a short written questionnaire, describing their expectations of the project at the point they had been recruited. (Appendix A)
- At the end of this first workshop the artists answered the same questions again, so that we could assess the way that their expectations had changed in response to the workshop presentations and activities.
- All artists completed a self-efficacy questionnaire, designed to explore individual confidence and frequency of computer use along with a range of practical technical skills. (Appendix B)

Commission proposals (10 artists)

- Following the introductory workshop, eight out of the ten of the invited artists submitted short proposals, describing artistic objectives and potential outcomes for the commissioned work. Consideration of these proposals was carried out independently of the questionnaire data from the initial workshop - artists had been told that selection for commission would not be based on their responses to the questionnaire, and the research team did not look at the questionnaires until after commissioning was complete.

Reflective diaries (4 artists)

- Each artist was asked to keep a diary, reporting their experiences and thoughts about work in progress on every day that they worked on the project (Appendix C). Although potentially valuable, only two of the five kept detailed daily diaries. Two others completed partial diaries, and one did not return the diary.

Workshop assessment (5 artists)

- At each of the joint workshops held during the commissioning period, artists were asked to complete the same short questionnaire used at the start and end of the introductory workshop (Appendix A).

Focus group discussion (4 artists, 3 commissioning team, 1 facilitator)

- The final workshop closed with an hour-long facilitated discussion of artists' experiences during the project, which was recorded and transcribed for analysis.

5. Analysis

5.1. Self-efficacy questionnaire

All ten artists completed the initial questionnaire (Appendix B). This allows us to make some broad observations about the previous experiences found across our sample. However, we recognise that this is a very small sample, and that the analysis below can only be offered as tentative observations of trends that might be explored in future research, rather than statistically reliable.

Based on our previous experience as administrators and collaborators in the professional arts, we believe that this sample of ten artists is reasonably representative of demographics in the intended group. They are aged between 30-45, 50% male/female, and almost all educated to the level of a Masters in Fine Art. They are established as professional artists, but not primarily digital arts practitioners (this last was a recruitment criterion in the goals of the project). None had studied maths or science beyond secondary school level, none had professional programming experience (questions 33-40) and the majority had never programmed a computer, although two had some experience of Basic, and two had used AppleScript.

Most scored relatively highly in the computing self-efficacy questionnaire (questions 1-10), with an average of 4.06 on the 5-point scale. Two of the sample were widely separated from the main group on this measure, with average scores of 3.1 and 3.3 over the 10 questions. There is some potential for bias in the responses to this self-rating scale, given that all respondents were aware of the forthcoming selection process for award of commissions. Although the preamble to the questionnaire did emphasise that the research data would not be used for selection, it is possible that the artists may have been unsure of this, especially as this was the first time that they had met the research team. As it turned out, one of the two low-scoring individuals was selected for the commissioned group, as was one of the highest-scoring.

The questions exploring tinkering (questions 26-29) showed a clear distinction between individuals who dismantled devices and made small repairs, and others who did not. With regard to our research question exploring “craft” creativity in the software domain, this distinction was predictive of the degree of ambition in the resulting artworks. We therefore intend to use this kind of question again in future, as we continue to investigate this technical culture trend.

The questions regarding home maintenance (questions 19-23) showed quite widely distributed responses, with no clear correlation to measures of computer self-efficacy or programming. This suggests that people carry out home maintenance tasks according to their particular domestic arrangements, rather than because of natural inclination or aptitude. We did see some stereotypical distribution in responses according to conventional male/female gender roles. As observed by Blackwell (2006), this stereotypical pattern occurs more strongly in the tinkering questions 26-29 than in the utilitarian home maintenance questions.

A notable feature of the questionnaire results was that some behaviours that might be unusual, or evidence of enthusiasm for creative hobbies, in the general population are ubiquitous among professional artists. All respondents built layered Photoshop images on a daily basis, wrote HTML for their own websites, and regularly used hand and power tools. These behaviours were consistent across the sample, independent of the other tinkering and craft-related measures.

Finally, it seems that three specific computer-related configuration activities are fairly well correlated with the computing self-efficacy questionnaire. These are: regular adjustment of Facebook privacy settings (Q18), creation of Excel formulae (Q17), and use of a command line (Q13). The last of these may be particularly useful for future research, in that it is consistently associated with the highest total across all our craft and software measures, and was the central feature in an exploratory cluster analysis of the questionnaire scores. However, we note once more that the very small sample size means that we are unable to treat these suggestions as statistically reliable.

5.2. Workshop questionnaires

The responses to the workshop questionnaires were analysed from the perspective of narrative development through the course of the project. Each artist completed the same questionnaire five times. They did not have direct access to their earlier answers, but individual concerns clearly emerged and were developed across the sequence of responses. Our analysis therefore responded to this data by treating it longitudinally, considering the developing experience of each artist in terms of the "stories" formed by the sequences of five answers to each question.

The camera module – this was an initial point of interest for all, but only two of the artists' projects made use of the camera. This is partly because use of the camera was optional – the artists' projects were driven by creative goals rather than the specific equipment available. However, the technical challenges of providing code support for the new module may have been an additional consideration when project ideas were being developed, meaning it was only used those who tended to be more confident.

The Raspberry Pi platform – all the artists were enthusiastic about the idea of coding in such a "simple" context, often with rather poetic ideas of what this might entail. However lack of fluency by comparison to familiar computing tools, through system setup, speed of response and technical obstacles, caused significant reframing of objectives and adaptation to the Raspberry Pi. This was not what they expected, and made it harder to be spontaneous and creative.

Educational resources – the online resources provided for a technically literate hobbyist community seem to be inappropriate to this group, who found there were significant barriers to entry in terms of the level of technical knowledge required. Several tried Google in the hope of finding solutions to problems, rather than exploring specialist forums. Advice from Sam was essential, but at times may have reduced independence and self-efficacy. Initially, the main exchange of information was during the in-person workshops – both interaction with Sam, and exchanging advice and experiences among themselves. However, as the project progressed and hardware problems were overcome, the creation of a shared space on Github provided a wonderful forum for individual problem-specific threads of conversations with artists starting to help each other out during the final phase of the project. The project mailing list was also relatively successful serving to offer a means for more broader questions and discussions.

Development tools – a key element of the project was the intention to adapt the Sonic Pi language with new media capabilities to support the creative projects proposed by the artists. However as the project progressed, the artists only gradually became familiar with our intention that (by working with Sam) they could extend the Sonic Pi language, and that those extensions would become their tools. We discuss this further below.

Creative outcome – all artists struggled to identify and maintain a concept of what they were trying to achieve artistically and technically, having to adapt their working methods while feeling that they were "near the bottom of quite a steep hill". The relatively short period of the project and the steep learning curve meant that all of the artists had to reduce the ambition of their original proposals. In retrospect, it might have been better for us to have offered a more extended period at the start of the project to learn the technical aspects, with greater variety of phased experiments and alternative outcomes for each artist.

Audience – with uncertain outcomes, the artists found it difficult to answer our question about audience. Because the work was commissioned as research and development projects without a specific audience in mind, there was significant ambiguity with regard to who the audience for the commissioned work might be – researchers, programmers, Raspberry Pi users, schools, arts centre visitors or festival audiences. As a result, the artists felt a tension between the action of programming and the notion of audience. One artist eventually concluded that using the Raspberry Pi "feels like an isolated experience" rather than something for an audience, while others emphasised visceral experiences of otherwise invisible Internet technologies, or even opened up the experience of programming itself, in a way that communicated some degree of the challenge and obscurity that they had experienced themselves.

We had asked artists to comment on ways in which they felt the Raspberry Pi designers might better support work of the kind they do. All of them found that the practical challenges of assembly and configuration were an obstacle that should be addressed – we report on this further below.

5.3. Reflective diaries

Although all of the five artists set out to make daily reflective diary entries as requested, only one of them continued through the whole of the development period. It was unfortunate that more of this data was not available, because the result did provide real insight into the experience of working with the Raspberry Pi to develop an artistic concept. However, the one complete diary does provide a valuable opportunity for triangulation with respect to other sources of data collected during the project. As a result, we are able to be reasonably confident that the analysis of this diary is to some extent representative. Importantly, the partially completed diaries from the other artists do confirm this analysis.

As with the workshop questionnaires, the approach that we take to the diary data is to explore narrative themes as they develop over the period of the project.

Technical theme: The first few weeks of the project were mainly concerned with start-up – getting the right combination of peripherals and connectors assembled in a suitable working location. For most of the project, descriptions of the technology were concerned with obstacles, and reported failures and frustration.

Artistic theme: Technical frustrations generally presented an obstacle to satisfying artistic achievements. They did occasionally result in creative sidesteps – for example, when the bare Raspberry Pi was unable to produce audio or video output of any kind, the heat of the CPU was used to melt a bar of chocolate. In order to explore the apparent obscureness of interaction with code, the keyboard was painted brown (incidentally resembling a large chocolate bar) to see whether code could be entered by touch-typing – an experiment that was soon abandoned! Satisfaction did start to appear after three weeks or so, especially as aspects of the original concept started to emerge (in altered form), and the somewhat encouraging closing entry after nearly two months “feel like I have found a creative way to engage with RP … finally”.

Concept theme: The development of the artistic concept did proceed in direct response to experiences of using the Raspberry Pi and Sonic Pi. Serious attention was given to its strengths and weaknesses, although many of these might have seemed less of a barrier to an artist with more experience of programming, electronic hardware development or Linux tools. The result succeeds in expressing a personal creative vision, although it is accompanied by concerns over whether this personal experience will be appreciated by an audience who are accustomed to more polished outcomes.

5.4. Focus group discussion

We recorded and transcribed a facilitated discussion involving four artists at the final workshop, together with the commissioning and technical team. Thematic analysis of the transcript identified the following issues:

Collective community: The joint workshops were a key element of the project - these were points at which artists experienced each other's projects, were able to meet with Sam in person, and most importantly, developed confidence by seeing what their peers were achieving, and "went away renewed".

Platform development: The intention of the project to evolve the Sonic Pi language with support for a wider range of technical capabilities and genres was extremely ambitious, requiring artists to look ahead to what they might be able to achieve, but before they had been able to experiment with the facilities they would use. By the last workshop, the potential of defining a new language was starting to be understood.

Hardware configuration: The Raspberry Pi appeals to tinkerers in part because of its openness – a bare circuit board that can be packaged or applied by hobbyists in many different ways. However, despite the fact that this aspect initially appealed to the artists in this project, the need to assemble components and connections from materials at hand posed immediate obstacles to their work. Furthermore, although the Raspberry Pi is cheaper and less encapsulated than a laptop or tablet, that lack of encapsulation also makes it less convenient in a studio environment that benefits from portability - being constrained by network sockets or TV screen locations.

Experimental momentum: Both development and hardware issues meant that the artists found it difficult to establish and maintain momentum. When unable to make progress with Raspberry Pi, they would move on to another project. This experience reflects anecdotal reports that Raspberry Pi's bought by non-technical purchasers can often get put away in a drawer unused, if users experience initial difficulty getting to grips with setup and programming. If we had been able to establish a residency model (perhaps like immersion courses in language teaching), with local technical support, this would almost certainly have been less of a problem.

Educational motivation: The experience of typing and seeing something happen was perceived as motivational. Sam's teaching methods are effective, but the artists were impatient to achieve more at an early stage. Their memory was that at the introductory workshop they "just made a beep," although an audio recording of that workshop reveals considerable sonic experimentation, which, as one would expect, demonstrates a difference between the creative ambitions of professional artists and the school students that Sam had previously worked with. The teaching materials for using Sonic Pi in schools include a simple "cheat sheet" of functionality supported by the language, but for use with artists, an ideal way forward would support avenues for exploration, combined with opportunities to be playful or transgressive.

Creative outcome: The artists perceived their final results as scaled down and less ambitious than the original proposals. Nevertheless, they did create distinctive work that spoke directly to their experiences of the device. At the time of this focus group discussion, none of the work had yet been seen by the public. Subsequent public shows and performance, generally combining short talks about the project by Rachel and demonstrations by the artists with Sam's live coding performance, and sometimes hands-on experimentation with Sonic Pi, have been very well received.

6. Discussion

In this final section, we discuss some of the most salient findings that emerged across the analysis of the different data sources collected during this research. These offer reflection on the project as a whole, and also some indication of opportunities that we consider important for further research.

6.1. Engagement and Attrition

A key element of this project was the objective to work with established professional artists, who, in order to maintain a steady flow of paid work, often have a broad portfolio of activities, much of which is driven by schedules determined months or years in advance – including teaching responsibilities, exhibition openings, construction commissions, performances and so on. As a relatively small commission, taking place over a short timescale, their commitments to the Defining Pi project had to be integrated into these timetables.

The Defining Pi workshops, although a central aspect of our research process, also had to be scheduled to account for bookings at Wysing Arts Centre, Sam Aaron's international bookings for live coding performance, and the obligations of several other collaborators. Finding dates when facilities would be available and the five artists would also be able to attend was difficult, with all of the originally planned dates having to be rescheduled to accommodate the best possible attendance. Allowing for more time between the selecting the artists and the start of the programme would most likely have eased this issue. Subsequent analysis of interviews, workshop feedback and diary reports confirmed that these workshops were indeed an essential part of the development process, in the eyes of the artists themselves. However, some of the five were able to attend only a small proportion of these. Those who did missed workshops were less engaged, and were not able to complete work that they found satisfying, with one artist unable to complete his project at all.

This can be taken as advice for those planning similar research in future – but it also interacted with the elements of frustration with basic hardware and software capabilities discussed earlier.

6.2. Out-of-box experience of empowerment

The Raspberry Pi is intended to be an open platform, that encourages experimentation not only by providing a supportive community and an open-source operating system, but by revealing the circuit board, its components and its interfaces. A low-cost platform is also perceived as low-risk, in the sense that users can experiment with it without being overly concerned that they might damage an expensive piece of equipment. In our informal observation of computing education for users with low self-efficacy, they find it empowering to assemble their own computer – for example, by adding the USB keyboard and mouse, the display, power supply and memory card to make their Raspberry Pi operational.

In promoting the Raspberry Pi concept, we have sometimes drawn comparisons to the relatively closed consumer experience offered to users of devices such as the Apple iPad. The software, hardware and media retail "ecosystem" of that product is strictly controlled by Apple, in contrast to Free/Libre Open Source Software such as GNU/Linux. Customers are not encouraged to program their iPad (for a long time, the AppStore terms specifically prohibited the distribution of Apps that provided programming functionality), and they are certainly not encouraged to open it up and look at its circuit board. The iPad is sealed shut, providing as closely as possible a completely seamless external surface, with no visible screws or fasteners.

The result is of course a trade-off – Apple products provide a well-controlled "out-of-box" experience, in which the user's experience provides a journey of induction into a refined and predictable commercial world.

In contrast to the Apple experience, open-source platforms such as Raspberry Pi suggest a narrative of technical empowerment, in which all the components are visible, anything can be changed, and the device can be configured in limitless ways. In the Defining Pi project, we hoped that professional artists would experience this empowerment as a liberating starting point for creative experience.

Unfortunately, evidence from workshop reports and diaries suggests that this ambition was frustrated by the difficulties experienced when setting up the Raspberry Pi at home. We provided each of the artists with a Raspberry Pi in the most basic state – the stand-alone board – and demonstrated the process of how to connect it up in the first workshop, at which each of the artists successfully set up their own Pi. The unanticipated obstacles related to the range of behaviours that can be found even in the "standard" interfaces of the Raspberry Pi. Raspberry Pi uses HDMI video output for compatibility with domestic televisions. However, several of the artists did not own HD televisions – and even where they did, these were not necessarily located in their studio workspace. Without a television, it was necessary to obtain an HDMI to VGA converter for use with older computer monitors. However, details such as the difference between a "converter" and a "cable" were unnecessarily worrying. Several of the artists had Apple keyboards that had apparently compatible USB connectors, but included internal USB hubs that contended with the Raspberry Pi system. Others had older keyboards and mice with PS/2 rather than USB connectors.

As a result, the early experiences of configuring and starting the machine were rather "frustrating" – a word that was used consistently and repeatedly in the artists' reports. A consequence was that much of the early enthusiasm for the project dissipated, as cables, keyboards and monitors were bought, tested, swapped, reordered and so on. In retrospect, this was a poor decision on our part. In the same way as some Raspberry Pi users have given up shortly after getting started due to similar issues, busy artists are equally likely to respond to obstacles by shifting their effort to other projects in which they are making rapid progress, especially where they are must wait for external parties before an obstacle will be resolved. If we had provided a complete kit of parts (several are available for Raspberry Pi), and more detailed instructions on what to do, everything would have run more smoothly – though perhaps with different ultimate outcomes.

6.3. Domain specific language as an artistic tool

Sam's live coding research has been heavily influenced by his previous experience as a developer of domain-specific languages for business and commercial applications. Sonic Pi, in particular, has been developed as a set of domain-specific extensions to the Ruby language, supporting basic music synthesis functions (Aaron & Blackwell 2013).

As a result, it was natural that he should approach this project with the objective of supporting the technical needs of the artists by providing new domain-specific language functionality in Sonic Pi. This is consistent with previous recommendations for art-technology collaboration, suggesting that technologists should provide new tools that empower artists to use technology in new ways (Turner et al 2005). However, it is rather different to our previous observations of digital artist-engineers as craftspeople for whom the creation and maintenance of their own tools is a key element of their creative practice (Woolford et al 2010). Indeed, Sam himself (as with many other live coders) developed and continually refines his own tools – both Sonic Pi, and the more powerful Overtone language oriented toward expert users.

In the case of the artists commissioned to work on this project, we deliberately chose to work with professional artist "end-users" (Ko et al 2011) rather than professional software artist-engineers. We wished to work with this group in order to understand their technical ambitions through early experiments, and to extend the Sonic Pi language into a tool that would help realise those ambitions. However, this meta-level of collaboration was not initially understood by the artists, due to their inexperience with programming concepts and their set-up frustrations. Although it was made explicit from the outset, several of the artists commented that they did not really understand the implications until much later in the project.

In the early stages of the work, the delays in hardware configuration of the Raspberry Pi morphed into perceived delays as Sam waited to receive feedback from artists, and artists waited for technical assistance from Sam. However, as these delays were overcome and confidence grew, the relationship between Sam and the artists as a group developed with an increasingly level of shared communication via the Github forum and discussion group. It was one of the more technically experienced artists (who had some previous programming experience, and also rated high on computer self-efficacy and "tinkering" scales) who fully absorbed the meta-level nature of domain-specific language development in the following diary entry: "realised it's not just about Sam helping us - he wants a conversation about how to structure functions and things".

As this was discussed with other artists in Github exchanges, subsequent workshop meetings, and the final focus group, the implications of creating a new language for computing became more apparent to all members of the group. As an objective for our own future work, we look forward to finding more effective ways to communicate this concept of the conceptual "tool" in a digital era, and reconciling it with the creative and motivational experiences that people have through craft.

7. Conclusion

Overall, there is much we can learn from this short experiment, to inform future projects with artists who would like to work with technology but who don't have programming experience. Despite the "self-help" ethos of communities such as Raspberry Pi, we would advise longer project timeframes and increased support with setting up equipment at home, or use of one of the standard Raspberry Pi kits that are now available. Nevertheless, the work produced by these artists demonstrated interesting and experimental uses of embedded technology. The public talks and workshops were highly popular, attracting total bookings of nearly 300 (limited by room capacity, with almost all events sold out).

Collaboration with the artists has provided valuable input to the Sonic Pi project, including a significant number of new features. The artists themselves gained new insight and skills that they intend to apply in future, building on these projects. Finally, we have benefited greatly from the perspective of Raspberry Pi users who share the creative craft ambitions of the Raspberry Pi Foundation itself, but bring very different life experiences and working practices to their work.

8. Acknowledgements

A large number of people and organisations have contributed to and assisted with this project. We are grateful to Arts Council England, who supported the commissioning and management expenses of the Defining Pi project through the Grants for the Arts scheme. We are grateful to Donna Lynas and Gareth Bell-Jones of Wysing Arts Centre for hosting workshops and working with us to commission and facilitate the work by artists. We are grateful to the Raspberry Pi Foundation for donation of equipment used in this project, and for a donation that has funded Sam Aaron's appointment. We are grateful to the venues and individuals who have helped to present this work to the public: Nicola Buckley at the Cambridge Festival of Ideas, Daniel Pitt at Cambridge Junction, Saar Drimer at Cambridge Makespace, Jason Fitzpatrick at Centre for Computing History, and Tahira FitzWilliam-Hall of the Circuit project. Finally, we are very grateful indeed to the artists who have collaborated with us in this experiment, and most especially for their patience and good will during an unusually challenging process.

9. References

Aaron, S. and Blackwell, A.F. (2013). From Sonic Pi to Overtone: Creative musical experiences with domain-specific and functional languages. Proceedings of the first ACM SIGPLAN workshop on Functional art, music, modeling & design, pp. 35-46.

Blackwell, A.F. and Collins, N. (2005). The programming language as a musical instrument. In Proceedings of PPIG 2005, pp. 120-130.

Blackwell, A.F. (2006). Gender in domestic programming: From bricolage to séances d'essayage. Presentation at CHI Workshop on End User Software Engineering

Blackwell, A.F. (2013). The craft of design conversation. In A. Van Der Hoek and M. Petre, (Eds), Software Designers in Action: A Human-Centric Look at Design Work. Abingdon: Chapman and Hall/CRC, pp. 313-318.

Burnard, P., Aaron, S. & Blackwell, A.F. (in press). Using coding to connect new digital innovative learning communities: Developing Sonic Pi, a new open source software tool. Presentation at SEMPRE 2014 - Researching Music, Education, Technology: Critical Insights. Society for Education and Music Psychology Research.

Church, L., Rothwell, N., Downie, M., deLahunta, S. and Blackwell, A.F. (2012). Sketching by programming in the Choreographic Language Agent. In Proceedings of the Psychology of Programming Interest Group Annual Conference. (PPIG 2012), pp. 163-174.

Ko, A.J., Abraham, R., Beckwith, L., Blackwell, A.F., Burnett, M., Erwig, M., Lawrence, J., Lieberman, H., Myers, B., Rosson, M.-B., Rothermel, G., Scaffidi, C., Shaw, M., and Wiedenbeck, S. (2011). The State of the Art in End-User Software Engineering. ACM Computing Surveys 43(3), Article 21.

Thrift, N. (2006). Re-inventing invention: new tendencies in capitalist commodification. Economy and Society 35(2), 279-306.

Turner, G., Weakley, A. Zhang, Y. and Edmonds, E. (2005) Attuning: A Social and Technical Study of Artist-Programmer Collaborations, In Proceedings of the 17th Annual Workshop on the Psychology of Programming Interest Group (PPIG 2005), 106-119.

Woolford, K., Blackwell, A.F., Norman, S.J. & Chevalier, C. (2010). Crafting a critical technical practice. Leonardo 43(2), 202-203.

Appendix A: Workshop Feedback

Technical In what ways does the camera module seem to meet your needs or not?

Platform In what ways does the Raspberry Pi seem a good or bad platform for your work?

Resources In what ways do the educational resources for Raspberry Pi suit your needs or not?

Tools Which tools for Raspberry Pi seem to meet your needs or otherwise?

Outcome In what ways has your work with Raspberry Pi satisfied your ambitions or not?

Audience How do you think work with Raspberry Pi will be received by an audience?

Aspiration How could your process or outcomes raise aspirations for Raspberry Pi users?

Strategy Does the work and/or process suggest any modification or critique of policy for the Raspberry Pi foundation?

Other Anything else we should be aware of or thinking about?

Appendix B: Self-efficacy Questionnaire

Part 1: Computing Confidence

Imagine you were given a new software package for some aspect of your work. The following questions ask you to indicate whether you could use this unfamiliar software package under a variety of conditions. For each condition, please indicate whether you think you would be able to complete the job using the software package.

1. if there was no one around to tell me what to do as I go.
2. if I had never used a similar tool like it before.
3. if I only had the software manuals for reference.
4. If I had seen someone else using it before trying it myself.
5. if I could call someone for help if I got stuck.
6. if someone else had helped me get started.
7. if I had a lot of time to complete the job for which the software was provided.
8.if I had just the built-in facility for assistance.
9. if someone showed me how to do it first.
10. If I had used similar packages before this one to do the same job.

Part 2: Typical computer usage

11. What operating systems are you familiar with (MacOS, Windows, Linux …)?
12. What software packages do you use regularly (Word, Excel …)?
13. How often do you use a command line?
14. If you use Microsoft Word, how often do you use paragraph styles / create new paragraph styles?
15. If you use Photoshop (or equivalent, like Gimp), how often do you organise an image as layers?
16. If you maintain your own web site, how often do you write HTML code?
17. If you use Excel (or other spreadsheet), how often do you create a formula?
18. If you use Facebook, how often do you modify your privacy settings?

Part 3: Tinkering

19. How often do you personally change parts on a bicycle or car?
20. How often do you re-organise your household files?
21. How often do you change time settings on central heating controls?
22. How often do you service plumbing or electrical fittings?
23. How often do you use a sewing machine?
24. How often do you use carpentry or hand tools?
25. How often do you use power tools?
26. How often do you take something apart to see how it works?
27. How often do you keep old parts in case they are useful?
28. How often do you make an electrical circuit?
29. How often do you make mechanical repairs?

Part 4: Technical education and experience

30. What was your highest school/university qualification in mathematics?

31. What was your highest school/university qualification in a science subject?
32. Have you ever written computer program? (if not, skip to the final question in this part)
33. What programming language are you most familiar with?
34. Where did you learn this programming language?
35. How many years have you been using this language?
36. What is the largest program you have written in this language?
37. What other programming languages have you used?
38. Do you write software as part of your work? If so, how many weeks, months or years have you spent writing software for your work?
39. Have you been paid to write software for other people? If so, how many weeks, months or years have you spent writing software professionally?
40. Is there any other experience or training you have had that you think may be relevant to this research?

Part 5: Demographic data

41. Age
42. Gender
43. First language
44. Highest educational qualification

Appendix C: Reflective Diary

Technical	What technical breakthroughs did you make, or major obstacles encounter, today?
Artistic	What aspects of today's work have been satisfying or otherwise?
Concept	How did your thinking about the project develop today?

Appendix D: Selected Press Coverage of Defining Pi

http://www.cam.ac.uk/sites/www.cam.ac.uk/files/uni-newsletter-summer-2013.pdf

http://www.cambridge-news.co.uk/Business/Business-News/When-Raspberry-Pi-met-the-artists-20130813170131.htm

http://www.cambridge-news.co.uk/Business/Business-News/Raspberry-Pi-finds-new-application-in-the-arts-20130820093523.htm

http://www.cambridge-news.co.uk/Whats-on-leisure/Choice/Our-top-10-picks-for-the-Festival-of-Ideas-20131018060030.htm

http://www.wysingartscentre.org/archive/wysing_on_tour/raspberry_pi_cambridge_festival_of_ideas/2013

http://www.cam.ac.uk/festival-of-ideas/events-and-booking/defining-pi-artist-led-experiments-with-the-raspberry-pi

http://www.cam.ac.uk/festival-of-ideas/events-and-booking/junction-university-sonic-pi-with-dr-sam-aaron