

Restart: The Resurgence of Computer Science in UK Schools¹

NEIL C. C. BROWN, University of Kent
 SUE SENTANCE, Anglia Ruskin University
 TOM CRICK, Cardiff Metropolitan University
 SIMON HUMPHREYS, BCS, The Chartered Institute for IT

Computer science in UK schools is undergoing a remarkable transformation. While the changes are not consistent across each of the four devolved nations of the UK (England, Scotland, Wales and Northern Ireland), there are developments in each that are moving the subject to become mandatory for all pupils from age 5 onwards. In this paper, we detail how computer science declined in the UK, and the developments that led to its revitalisation: a mixture of industry and interest group lobbying, with a particular focus on the value of the subject to all school pupils, not just those who would study it at degree level. This rapid growth in the subject is not without issues, however: there remain significant forthcoming challenges with its delivery, especially surrounding the issue of training sufficient numbers of teachers. We describe a national network of teaching excellence which is being set up to combat this problem, and look at the other challenges that lie ahead.

Categories and Subject Descriptors: K.3.2 [Computers and Education]: Computer and Information Science Education—*Computer science education*; K.4.1 [Computers And Society]: Public Policy Issues

General Terms: Human Factors

Additional Key Words and Phrases: Computer Science Education; High School; Teachers

ACM Reference Format:

Neil C. C. Brown, Sue Sentance, Tom Crick and Simon Humphreys. 2013. Restart: The Resurgence of Computer Science in UK Schools. *ACM Trans. Comput. Educ.* 1, 1, Article 1 (January 2013), 22 pages.
 DOI: <http://dx.doi.org/10.1145/0000000.0000000>

1. INTRODUCTION

Computer science in UK schools has been on a roller-coaster journey. In the 1990s and early 2000s computer science and programming slowly vanished from UK schools, swallowed by the subject “Information and Communication Technology” (ICT), which too often focused on the use of technology and software rather than its creation and the underlying principles of computation. There was little, if any, awareness of the difference between *using* a computer and *programming* a computer, both among the general public and also among education policy-makers.

¹This paper covers similar topics to a previous publication [Brown et al. 2013], but with more of a focus on the education system and formal teacher training initiatives and less on the CAS organisation. This article is also greatly updated to include more recent developments, such as the new national curriculum.

Author’s addresses: N. C. C. Brown, School of Computing, University of Kent, UK; S. Sentance, Department of Education, Anglia Ruskin University, UK; T. Crick, Department of Computing, Cardiff Metropolitan University, UK; S. Humphreys, BCS, The Chartered Institute for IT, UK.

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies show this notice on the first page or initial screen of a display along with the full citation. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, to republish, to post on servers, to redistribute to lists, or to use any component of this work in other works requires prior specific permission and/or a fee. Permissions may be requested from Publications Dept., ACM, Inc., 2 Penn Plaza, Suite 701, New York, NY 10121-0701 USA, fax +1 (212) 869-0481, or permissions@acm.org.

© 2013 ACM 1946-6226/2013/01-ART1 \$15.00

DOI: <http://dx.doi.org/10.1145/0000000.0000000>

In the past five years, however, computer science has begun a dramatic resurgence that sees it poised to become mandatory for pupils in England from age five upwards. This turn-around has come through a mixture of pressure from industry and lobbying by interest groups, with the government now apparently convinced of the wider value of computing education for all pupils. The argument that has been presented is that computing develops useful transferable skills (often referred to as “computational thinking” [Wing 2006]) as well as valuable principles for a modern, digital world and thus is of benefit to all students.

This sudden growth in computer science education is not without its challenges: an increase in subject uptake requires a corresponding increase in available teachers. There are few existing teachers trained in computing, and thus a major challenge exists to retrain the current ICT teachers – many of whom switched to ICT from other subjects (for example, business studies, geography, physical education, etc) without extensive ICT training, let alone computer science training.

In this paper, we start by detailing how ICT came to eclipse computer science (section 2), how the case for computer science was (successfully) presented and the resulting developments in English education policy (section 3), before explaining the current state of computing in UK schools (sections 4 and 5) and the challenges that lie ahead for teaching computer science (section 6).

2. BACKGROUND: A BRIEF HISTORY OF COMPUTING IN UK SCHOOLS

In the 1980s, computer science was available in schools, under the name “Computer Studies”, which included hardware, logic, binary, programming and various other aspects of computers [Doyle 1988]. In the 1980s, home computers (such as the BBC Micro, popular in the UK) had a restricted range of end-user applications compared to present-day computers. Most came with a BASIC interpreter, and many loaded this interpreter by default, making programming an obvious facet of computer use.

The spread of home computers, especially IBM PCs and their clones, was accompanied by more end-user software, for tasks such as word processing and spreadsheets. The government recognised that it was important to instill digital literacy in the population via schools. Computing in schools slowly became focused on computer use, under the banner of Information and Communication Technology (ICT) popularised by the Stevenson report [The Independent ICT in Schools Commission 1997].

In the 1990s, programming and understanding the workings of computers became secondary to using computers and computer applications. While the ICT syllabus was predominantly focused on using computers, there was still mention of programming – but programming was often not covered by schools, due to the permitted flexibility in which parts of the curriculum were taught [Ofsted 2011; The Royal Society 2012, p34].

By the 2000s, ICT was prevalent in schools and focused on using computers (IT skills and digital literacy), while computer science was found primarily at A-Level (the age 16–18 qualification). However, ICT was suffering from a problem of a worsening reputation among pupils and other stakeholders for being dull and unchallenging [The Royal Society 2012] and being regarded as a low-value discipline, especially compared to other STEM subjects. With ICT now embedded across the curriculum at primary schools in the UK, pupils in secondary school increasingly found ICT unstimulating if they already had the skills that were being taught.

The Royal Society report suggested that the problems with ICT were interlinked in a vicious circle [The Royal Society 2012, section 7.9, p85], as follows. It was difficult to recruit specialist ICT teachers, due to the well-paid alternatives for people with those skills, along with the low reputation of ICT. This meant that schools were forced to use non-specialists for ICT teaching. In turn, a lack of specialist teachers meant that qualifications tended to be less demanding, which contributed to ICT’s low reputation.

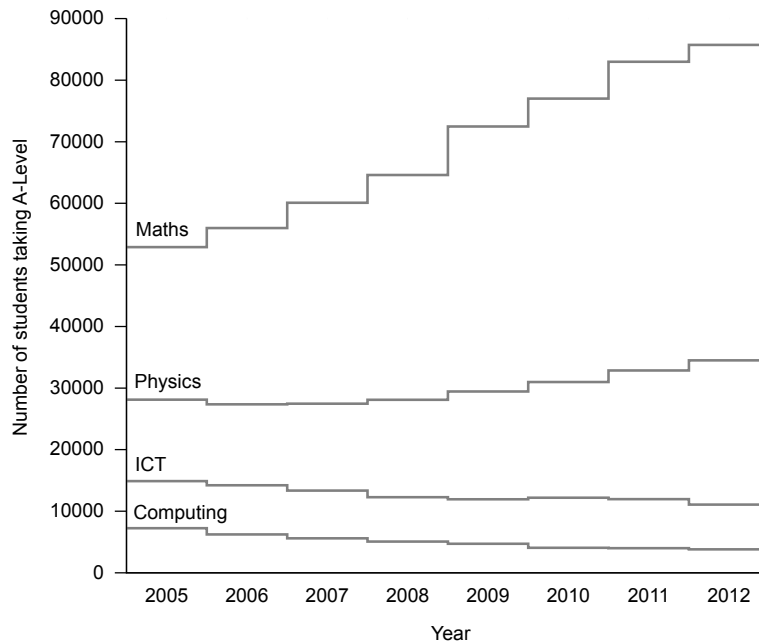


Fig. 1. Number of students taking A-Level Computing and ICT in the UK. For comparison with other STEM subjects, figures for A-Level Physics and Mathematics are also shown. Source: Joint Council for Qualifications, <http://www.jcq.org.uk/>.

One of the emergent effects of this vicious circle – the undemanding nature of many ICT qualifications – caused many schools to enter a lot of students for the qualifications [The Royal Society 2012, section 7.9, p85]. This provided easily-achieved high results which would boost a school's league table position. The Wolf report found that this was a problem across several disciplines, due to an unduly high league table weighting allocated to some vocational qualifications [Wolf 2011].

This focus on ICT in schools had meant that there were often no options for pupils who wished to study computer science, and who often mistakenly believed that ICT and computer science were one and the same subject. Not having a clear understanding of what exactly computer science was seriously hindered choices regarding its selection at a later stage. The majority of students left school actively disliking what they mistakenly believe to be computer science [The Royal Society 2012].

The one place in schools (outside Scotland) where computing still had a presence was A-Level, the age 16–18 qualification that precedes university. However, following a dot-com-based boom around the year 2000, A-Level numbers reduced and have continued to decline ever since. This is in contrast to a slower decline in ICT, a small increase in Physics and a large increase in Maths during the same period (see Figure 1).

3. RECENT DEVELOPMENTS: LOBBYING AND RE-INTRODUCTION OF COMPUTING

In 2008, computer science had thus almost disappeared from the curriculum. That was the year that an interest group, Computing At School (CAS)², was formed to help promote the cause of computing in schools. Although it was not the only initiative that

²Disclaimer: the authors are all members of CAS.

was pushing this agenda, it would come to play a central role, and would eventually be recognised as the official subject association for computer science in the UK.

At its inception in 2008, CAS seemed to fight a lonely battle against the odds [Brown et al. 2013]. While many individuals agreed that the state of computer science teaching in UK schools was problematic, few organisations seemed inclined to act to improve it. Government were initially unconcerned with the problem of computer science in schools. The subject of ICT was widely taught in schools, with learners obtaining qualifications in record numbers; as far as most government officials were concerned, ICT addressed all computing-related needs. This belief was only unseated by the publication of several reports, combined with interventions by industry³.

In 2011, however, the tide started to change [Crick and Sentance 2012]. Several organisations became actively involved in promoting improvements in UK computing education: e-skills UK [e-skills UK 2012] (the UK's Sector Skills Council for Business and Information Technology), the CBI [CBI 2012] (the UK's business lobbying organisation) and the UCU [Universities and Colleges Union 2012]. That year also saw as the publication of the Nesta *Next Gen* report [Livingstone and Hope 2011]. This high-profile report highlighted the skills and talent pipeline required to support the UK's digital/creative industries by explicitly arguing for computer science to be a core part of the school curriculum. These reports were supported by various public statements, including a widely-publicised speech by Google's executive chairman Eric Schmidt at his MacTaggart Lecture at the 2011 Edinburgh International Television Festival⁴:

"I was flabbergasted to learn that today computer science isn't even taught as standard in UK schools...Your IT curriculum focuses on teaching how to use software, but gives no insight into how it's made."

The year 2012 thus became a breakthrough year in the UK, when computer science as a school subject moved out of the shadows of education policy and into the mainstream media, the public discourse of politicians, and the review of the national curriculum. In January 2012, the Royal Society – the UK's Academy of Sciences – published a highly-noted report on the state of computing education in the UK, entitled *"Shutdown or restart? The way forward for computing in UK schools"* [The Royal Society 2012], which made a number of clear recommendations, including the reintroduction of computer science into schools. In the same week, Michael Gove, the UK's Secretary of State for Education, gave a major policy speech⁵ in which he "disapplied" (i.e. removed) the existing ICT programmes of study in England, whilst keeping the subject as a compulsory part of the curriculum until the age of 16, thus leaving schools free to teach whatever they like (and facilitating the adoption of a broader computing curriculum). The UK Department for Education thus declared the re-introduction of computer science teaching into English schools an official goal, with the ICT curriculum to be rewritten and supported by the development of "new, high-quality Computer Science [qualifications]". This was in light of a full National Curriculum review in England, but signalled the first time a senior government representative had declared the wider educational value of computer science:

"We're encouraging rigorous computer science courses. The new computer science courses will reflect what you all know: that Computer Science is a rigorous, fascinating and intellectually challenging subject. Computer Science requires a thorough grounding in logic and set theory, and is

³For example, see <http://academy.bcs.org/upload/pdf/cs-school-gove.pdf>

⁴<http://www.guardian.co.uk/media/interactive/2011/aug/26/eric-schmidt-mactaggart-lecture-full-text>

⁵<http://www.education.gov.uk/inthenews/speeches/a00201868/michael-gove-speech-at-the-bett-show-2012>

merging with other scientific fields into new hybrid research subjects like computational biology.

Although individual technologies change day by day, they are underpinned by foundational concepts and principles that have endured for decades. Long after todays pupils leave school and enter the workplace – long after the technologies they used at school are obsolete – the principles learnt in computer science will still hold true.”

This led to several further important developments. The first related to the English Baccalaureate (EBacc). The government were concerned by the propensity of schools to chase qualifications that allowed for easy high grades (and thus higher league table placings), as touched on by the Wolf report [Wolf 2011]. The EBacc was a meta-qualification intended as a new league table measure: it combines several GCSE (General Certificate of Secondary Education) results into a single measure. The included GCSEs are traditional core subjects: English, maths, sciences, modern foreign languages, geography and history. This was intended to arrest a decline in science and languages. It has already had a marked impact on the take-up of foreign languages: the proportion of students studying for a foreign language GCSE increased from 36% to 51% in a single year after the announcement of the EBacc [Tinsley and Han 2012].

The government had previously stated that the subjects included in the EBacc were fixed, but following prolonged lobbying, they announced that computer science GCSEs would be included in the EBacc, in the sciences category. This is an important signal from the government of the respect that it holds for computer science. At this stage, soon after the announcement, it is hard to determine the impact EBacc inclusion has had on computer science GCSE take-up, but it is nevertheless a positive development.

The other major development relates to the new National Curriculum in England. This was due for review in 2011–2012, and thus was fortuitously timed with respect to the resurgence in computer science, and the negative opinion of ICT. The recently published national curriculum [Department for Education 2013] broadly followed the recommendations of the Royal Society report [The Royal Society 2012], reforming ICT into a new subject named “Computing”, composed of three strands: digital literacy, information technology and computer science. We will discuss this important development more thoroughly in section 5.1.

These developments in government policy were in large part down to the lobbying conducted by CAS. The success of CAS at policy level has been boosted by several factors. One factor has been the support of industry and other bodies to whom government was attentive; another, the broader international focus on computing education and digital skills [European Commission 2012; Informatics Europe & ACM Europe Working Group on Informatics Education 2013]; another is the composition of the CAS member body: teachers. Being an organisation that represents thousands of school-teachers gave more weight than being purely a lobbying organisation. The other factor was that CAS gained the support of the BCS⁶ (the UK’s chartered professional society for computing, an analogue to the ACM), an established organisation, which helped to legitimise CAS as representing the wishes of industry as well as education.

It should be noted that the nature of the UK education system is such that it has been possible to advocate for change to be driven by central government; we recognise that this may not be the case in other countries [Snyder 2012].

Significant work at policy level has been on changing the perception of the discipline of computer science, including its wider utility across education. It has also been im-

⁶<http://www.bcs.org>

portant to differentiate between ICT and computer science – it was clear around the formation of CAS in 2008 that there was a profound misunderstanding of computer science. It was not recognised as a rigorous academic discipline, with strong theoretical foundations and a body of knowledge, distinct from developing technology competencies and digital literacy skills. This misperception was prevalent in government as well as the wider public.

The members of CAS believed that a simple labour shortage would be insufficient justification for changes to the curriculum – and that a focus on only those pupils who would become computing professionals was short-sighted. The value of history in school education is not justified by a shortage of historians; physics is not taught because of the figures related to how much engineering can boost the economy. The choice of subjects, especially compulsory subjects, is driven by their value to all pupils. Thus there was been a strong focus in the promotion of computer science on the transferable skills that it can yield: often termed “computational thinking”, this includes logical reasoning, problem-solving, debugging strategies, algorithmic thinking, and so on.

Many of these skills are covered by no other subject in the existing curriculum. Mathematics contains logical reasoning, but in schools does not typically provide the sort of open-ended problem solving that computing does [Ofsted 2012]. Several computing teachers have commented to the authors that computing is the one subject in schools where failure is an acceptable part of the subject, and that this in itself is a valuable experience for pupils.

4. SCHOOL SYSTEMS IN THE UK

As reported by Hubweiser et al., when establishing a model for viewing school CS education [Hubweiser et al. 2011, p21], it is apparent that there is much diversity between school education systems, and this can create an obstacle when trying to understand progress made in a different country. Here we describe the context of school education in the UK.

For historical reasons, the UK does not have a single nationwide education system. The UK is primarily composed of four devolved nations: England (population: 53.0 million), Scotland (5.3 million), Wales (3.0 million) and Northern Ireland (1.8 million) [UK Office for National Statistics 2011]. Each nation has its own education system, although they are broadly similar in England and Wales.

4.1. England and Wales

Figure 2 shows the system of five Key Stages (KS) used in England and Wales (although KS1 is called Foundation Phase in Wales) with compulsory schooling until age 16. All subjects are compulsory until the end of Key Stage 3 (KS3) and then students can choose approximately ten subjects to study for the next two years, which each lead to GCSE (General Certificate of Secondary Education) qualifications. However, while the National Curriculum in England and Wales are broadly similar, they are distinct and use different terminology.

There is state provision for education in the UK up to the age of 19, with mostly comprehensive, mixed ability schools across the UK. A few areas in England have retained a system of selective 11+ schools called grammar schools, which require students to sit an exam prior to entry, but these schools are in the minority. As well as state schools, 10% of schools in the UK are independent fee-paying schools. Overall, in England there are approximately 24,000 schools, including 16,800 primary schools, 3,400 secondary schools and 2,400 independent schools (primary and secondary). However, the primary and independent schools tend to be smaller: the state-funded schools had 4.2 million primary pupils and 3.2 million secondary pupils, with 0.6 million pupils in independent

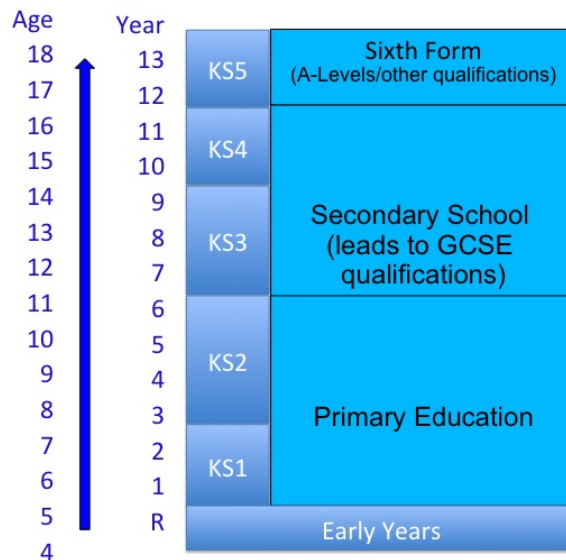


Fig. 2. Key Stages in the English education system

schools [Department for Education 2012b]. The ICT curriculum in Wales (2008)⁷, was perceived to be less prescriptive than the ICT curriculum in England, but exhibiting many of the same issues. It was recently reviewed by an independent steering group appointed by the Welsh Government [Arthur et al. 2013], making clear recommendations for reforming the ICT curriculum as part of a broader national curriculum review for September 2014.

4.2. Northern Ireland

Originally, Northern Ireland had the same National Curriculum as England and Wales but this was subsequently replaced by a revised Northern Irish version, which is less prescriptive. ICT is taught across the curriculum. There are GCSEs, as in England, but with a different awarding body; students can take a GCSE in ICT. Northern Ireland has made fewer steps than England towards formally offering Computing in the curriculum.

4.3. Scotland

Scotland has a different education system. Instead of GCSEs and A-Levels, pupils in Scotland take Standard Grades, Highers and Advanced Highers. They spend one year longer in primary school and go to secondary school at age 12. Computer Studies had been a subject within the Scottish Curriculum for many years. New teachers trained to teach Computer Studies: in this way Scotland was much further ahead than England in the introduction of computer science. However, there are smaller departments and not as many posts in schools as there are in England. The Scottish national curriculum (implemented in 2010), named the 'Curriculum for Excellence'⁸, defines what children

⁷<http://wales.gov.uk/topics/educationandskills/schoolshome/curriculuminwales/arevisedcurriculumforwales/nationalcurriculum/ictnc/?lang=en>

⁸<http://www.educationscotland.gov.uk/thecurriculum/index.asp>

should learn from age 3 to 18 and specifies a new subject called Computing Sciences. To build capacity for this new subject, a national CPD initiative has been funded by the Scottish Government (PLAN C (Professional Learning And Networking for Computing)⁹), along with teaching resources¹⁰ from the Royal Society of Edinburgh and the BCS.

5. THE NATIONAL CURRICULUM IN ENGLAND

A National Curriculum was established in 1988 with particular subjects compulsory to age 16, which came to include ICT, as described in section 2. All students studied the subject in some form, and ICT became an important part of the school curriculum. However, there have been some significant changes to education, particularly in England, over the last three or four years, initially introduced by a new government in a white paper [Department for Education 2010]. These included a complete overhaul of the National Curriculum, an English Baccalaureate to measure performance in academic subjects, and the introduction of Academies and Free Schools. Schools can elect to become Academies and then have much more financial autonomy than state schools, can employ teachers with no teaching qualification in any subject and can deviate from the national curriculum; they are publicly funded, but free from local authority and national government control. The government subsequently announced that they would “create a national network of Teaching Schools...which will take a leading responsibility for leading and quality assuring initial teacher education in their areas”) [Department for Education 2010]. The intention seems to be to reduce the number of new teachers being trained to teach in universities. Alongside these very fundamental and structural changes involving schools, a review of the primary and secondary National Curriculum in England was announced on 20 January 2011:

“It is the Government’s intention that the National Curriculum be slimmed down so that it properly reflects the body of essential knowledge which all children should learn and does not absorb the overwhelming majority of teaching time in schools. Individual schools should have greater freedom to construct their own programmes of study in subjects outside the National Curriculum and develop approaches to learning and study which complement it.” [Department for Education 2011]

These changes have been rapid and far-reaching across the education system and are not all popular with teachers and other stakeholders. While a refresh of the curriculum and programmes of study in a number of subjects had been widely encouraged, the top-down structural changes to the funding of schools in England, alongside the significant changes to recruitment and training of new teachers, has disrupted the focus on the development of a new National Curriculum.

5.1. The National Curriculum Review 2012-2014 and its impact on Computer Science in school

Within these far-reaching changes to education have been developments that have benefited computer science teaching in English schools (described in section 3):

- Computer Science has become part of the English Baccalaureate
- As part of the National Curriculum review, ICT has been rebranded as “Computing”
- The National Curriculum for Computing incorporates the teaching of Computer Science as compulsory from ages 5-16

⁹<http://www.planforcomputing.org.uk/>

¹⁰<http://www.royalsoced.org.uk/1034.ComputingScience.html>

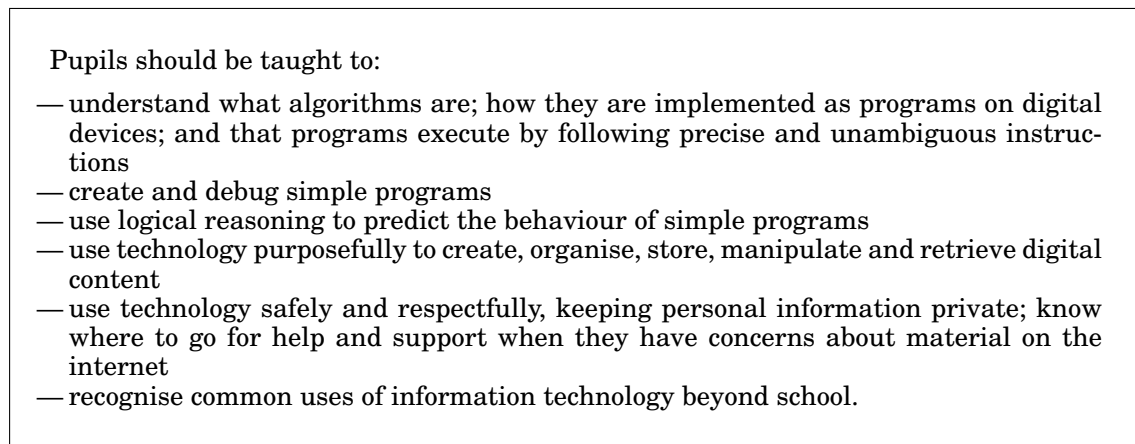


Fig. 3. Subject content for Key Stage 1 (age 5–7) [Department for Education 2013]

The new curriculum has significantly changed from the previous curriculum for ICT. Formerly the national curriculum for ICT had an emphasis not just on use of standard software but also on evaluation of the quality and suitability of available software resources; this was very much within a framework of developing informed consumers of technology. The new curriculum seeks to build on this by encouraging an increased understanding of the creative view of using technology as well as involving the development of computational thinking skills and the understanding of how technology works. The subject is to be taught throughout primary school and pupils are to be taught to think algorithmically and to learn to write simple programs before they reach 11 years old. Significantly, the subject will change its name from ICT to Computing.

Extracts of the new National Curriculum for computing [Department for Education 2013] are shown in Figures 3, 4 and 5. The curriculum for Key Stage 1 (see Figure 3) calls for pupils aged 5–7 to create and reason about programs. This may initially seem ambitious – but this is not about sitting five year olds in front of a C++ development environment, but about developing deeper computational thinking and problem solving skills [Bitto and Mirolo 2013]. It is deliberately not mandated that this learning should take place on an actual computer. Exercises such as CS Unplugged [Bell et al. 2009] and similar non-computer-based exercises can cover the requirements of the curriculum without programming a physical computer.

One example exercise is the Jam Sandwich Robot¹¹, where the teacher role-plays a robot, and asks the pupils to instruct him in making a jam sandwich. The pupils start with vague instructions (*“put butter on the bread”*) and the teacher deliberately misinterprets all vague or ambiguous instructions (e.g. by putting the tub of butter on top of the loaf of bread) until the pupils become precise enough to make the sandwich.

It is only later in the curriculum that programming languages are mentioned, but the exact choice of language is not mandated, only the modality – at age 11–14, pupils should “use two or more programming languages, at least one of which is textual” (see Figure 5). Given the current popularity of the Scratch programming system, this line is broadly interpretable as ‘Scratch plus a text-based programming language’; while the requirements before age 11 can be fully satisfied in Scratch, the 11–14 curriculum deliberately ensures that pupils move to full-text programming.

¹¹Made known to the authors by Phil Bagge, a computing primary specialist.

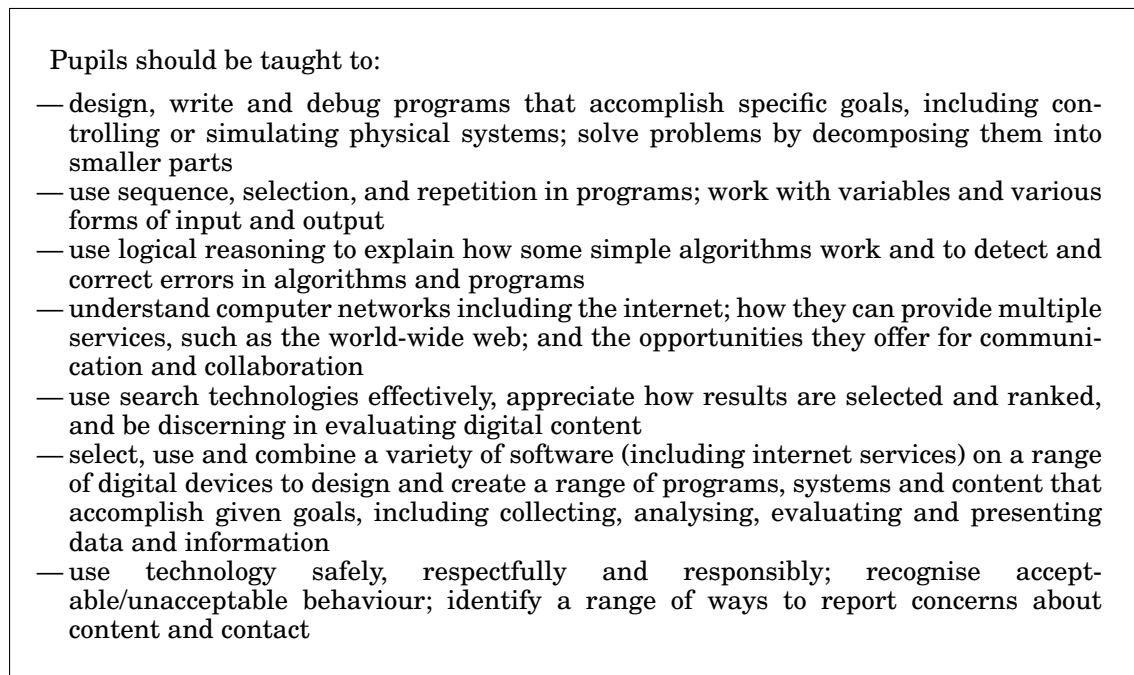


Fig. 4. Subject content for Key Stage 2 (age 7–11) [Department for Education 2013]

5.2. New school qualifications

Alongside the developments within the National Curriculum, and encouraged by CAS, the awarding bodies (who establish school qualifications across the country), have developed GCSE qualifications (taken at age 16) in Computer Science (also called Computing), to augment the existing suite of qualifications which include GCSEs in ICT and more vocational coursework-based qualifications in ICT [The Royal Academy of Engineering 2012]. To date, there are four GCSE programmes in Computing/Computer Science available to schools.

The OCR awarding body published the first of these modern computing GCSEs in 2010, and its syllabus [OCR 2011] is typical of the GCSEs from the other awarding bodies. It consists of three modules: a theory module and two practical project assessments, one of which is completely centred around programming a solution to a set task. An outline of the theory content is given in Figure 6.

5.3. Other Initiatives

Alongside the focus on government policy and curriculum/qualification reform over the past few years, a number of other independent groups with similar goals have also sprung up in the UK over the same period, invariably with little or no funding. A particular focus of many of these grassroots initiatives has been on the importance of “getting kids coding” (see [Douglas Rushkoff 2011]); prominent among them are:

- **Apps for Good**¹² This project helps young people at school create apps, focusing on entrepreneurial as well as technical skills.

¹²<http://appsforgood.org/>

Pupils should be taught to:

- design, use and evaluate computational abstractions that model the state and behaviour of real-world problems and physical systems
- understand several key algorithms that reflect computational thinking [for example, ones for sorting and searching]; use logical reasoning to compare the utility of alternative algorithms for the same problem
- use two or more programming languages, at least one of which is textual, to solve a variety of computational problems; make appropriate use of data structures [for example, lists, tables or arrays]; design and develop modular programs that use procedures or functions
- understand simple Boolean logic [for example, AND, OR and NOT] and some of its uses in circuits and programming; understand how numbers can be represented in binary, and be able to carry out simple operations on binary numbers [for example, binary addition, and conversion between binary and decimal]
- understand the hardware and software components that make up computer systems, and how they communicate with one another and with other systems
- understand how instructions are stored and executed within a computer system; understand how data of various types (including text, sounds and pictures) can be represented and manipulated digitally, in the form of binary digits
- undertake creative projects that involve selecting, using, and combining multiple applications, preferably across a range of devices, to achieve challenging goals, including collecting and analysing data and meeting the needs of known users
- create, re-use, revise and re-purpose digital artefacts for a given audience, with attention to trustworthiness, design and usability
- understand a range of ways to use technology safely, respectfully, responsibly and securely, including protecting their online identity and privacy; recognise inappropriate content, contact and conduct and know how to report concerns.

Fig. 5. Subject content for Key Stage 3 (age 11–14) [Department for Education 2013]

- **Fundamentals of computer systems**
- **Computing hardware:** The Central Processing Unit, Binary logic, Memory, Input and output devices, Secondary storage
- **Software**
- **Representation of data in computer systems:** Units, Number, Character, Images, Sound, Instructions
- **Databases:** The database concept, The DBMS, Relational databases
- **Computer communications and networking:** Networks, The Internet
- **Programming:** Algorithms, Programming Languages, Control flow in imperative languages, Handling data in algorithms, Testing

Fig. 6. Outline of *Unit A451: Computer systems and programming*, taken from the OCR GCSE Computing syllabus [OCR 2011].

- **Code Club**¹³ Launched by Clare Sutcliffe and Linda Sandvik in April 2012, supports schools in running after-school programming clubs in primary schools; by September 2013 they had more than 1000 clubs across the UK.
- **cs4fn**¹⁴ A project to enthuse and teach both students and others about interdisciplinary computer science research. It consists of a print magazine and supporting website about “computer science for fun”. For example, magic tricks each with a computational component, feature prominently.
- **Raspberry Pi**¹⁵ The low-cost credit-card-sized computer has been astonishingly successful. The Raspberry Pi Foundation had initially hoped to sell 10,000, but have now sold over over 2,000,000. See also *#RaspberryJam*¹⁶, the global community of events for enthusiasts, formed by Alan O’Donohoe, a schoolteacher.
- **Technocamps**¹⁷ A long-running initiative led by Swansea University, currently funded by the Welsh Government through the European Social Fund, to provide extra-curricula workshops for computing and raise awareness of computing and STEM careers.
- **Young Rewired State**¹⁸ Hack days for teenagers.
- **YouSrc**¹⁹ A website aimed at supporting programming tuition, built by Paul Clarke. It is specifically adapted for the workflow of the UK education system.

The above list of projects and initiatives is illustrative rather than exhaustive – there are a huge number of regional and national projects, ranging from hardware/programming through to broader socio-technical initiatives, that have aligned to support the new curriculum changes, as well as changing the wider public perception of computing as an academic discipline and showcasing the range of computing careers. For example, the Raspberry Pi has already had a significant impact on schools and teachers, partly due to the worldwide publicity associated with the project (more so in the UK, now it is manufactured in Wales rather than China) and how it can be used to teach computing and programming. The Raspberry Pi Foundation have partnered with OCR to develop resources and teaching materials to support the new curriculum and qualifications²⁰.

Many of these initiatives have drawn inspiration from similar groups in other countries, such as CS Unplugged²¹ and Code Avengers²² in New Zealand, CoderDojo²³ in Ireland, in the USA the AP CS Principles project²⁴, Codecademy²⁵, Bootstrap²⁶, Georgia Computes!²⁷ and Exploring Computer Science²⁸ in Los Angeles.

¹³<http://www.codeclub.org.uk/>

¹⁴<http://www.cs4fn.org/>

¹⁵<http://www.raspberrypi.org/>

¹⁶<http://raspberrypi.org.uk/>

¹⁷<http://www.technocamps.com/>

¹⁸<https://youngrewiredstate.org/>

¹⁹<http://www.yousrc.com/>

²⁰<http://www.ocr.org.uk/qualifications/by-subject/computing/raspberry-pi/>

²¹<http://csunplugged.org/>

²²<http://www.codeavengers.com/>

²³<http://coderdojo.com/>

²⁴<http://www.csprinciples.org/>

²⁵<http://www.codecademy.com/>

²⁶<http://www.bootstrapworld.org/>

²⁷<http://gacomputes.cc.gatech.edu/>

²⁸<http://www.exploringcs.org/>

5.4. Issues in Implementing Changes

Various issues need to be considered within schools: firstly, how schools react to the pace of change, secondly, how digital literacy can still be incorporated into the curriculum and thirdly, the provision of teachers to deliver the new curriculum.

5.4.1. Primary schools. The curriculum changes that have taken place have been rapid; secondary schools are gearing up for these by reviewing what is delivered in the compulsory stages (ages 11-14) and what courses that they can offer at Key Stage 4 (ages 14-16). In primary schools the implications of the changes to the national curriculum are more gradually becoming apparent and this will require widespread teacher development to implement the new curriculum across primary schools in a meaningful way.

Most primary school teachers are generalists, who teach all subjects to a particular class. Computing is not their main focus, but they will need to take on what is likely to be entirely new material. One computing specialist has told us of going into a primary school and being asked by a slightly panicked-looking teacher: “*Thank goodness – are you the one who’s going to tell me what an algorithm is?*” The material required in primary schools is not as extensive as that for secondary schools, but it will be new to many of the teachers required to teach it (primary teachers who taught in the 1980s would likely have taught Logo and similar, so older teachers will have seen much of the material before).

CAS is very much aware of the issue of teacher development and steps are being taken to address it (see Section 6). Another challenge faced by CAS is to ensure that head teachers and governing bodies are aware of the importance and significance of these curriculum changes.

5.4.2. Digital literacy. To have a curriculum throughout primary and secondary that encompasses computational thinking and programming is exciting and will facilitate a generation of young adults who have a foundation of computational skills to build on, whatever their career choice. However, we also have a responsibility to give all children sufficient skills in digital literacy that they can enter any workplace as a competent user of technology. This needs to be incorporated into the teaching in school, potentially across the curriculum. For digital literacy to be embedded in every subject there does need to be some planning and staff development and this has not been detailed at any level in the governments proposed changes. Thus, the issue of digital literacy for all is an issue which will have to be addressed in schools alongside the new areas of the national curriculum.

5.4.3. Professional development. The third issue relates to professional development for teachers and will be discussed in the next section.

6. CHALLENGES IN TEACHING THE NEW CURRICULUM

In this section we will discuss how teachers in the UK (primarily England) are being prepared to teach the new Computing curriculum.

6.1. Recruitment of new teachers

In the UK, there are a variety of different ways in which to train as a teacher. The most common is via a postgraduate certificate in either primary or secondary education which lasts for one year, has a large component of actually teaching in the classroom and requires an undergraduate degree (for secondary education, usually in the subject in which one wants to teach). Alternatively, some undergraduate degrees in education are available which also give students qualified teacher status; these are available

more for prospective primary teachers than secondary teachers. Another option for prospective teachers is in-school training which is salaried and “on-the-job”.

Previously it has not been easy to recruit sufficient teachers to teach ICT in schools, particularly those with a degree containing an element of Information Technology or Computer Science. Graduates with degrees in a variety of first degrees have often been recruited to teach ICT in school, providing that they had a good understanding of ICT or relevant work experience. In the last year the task to recruit teachers has changed. Secondary teachers will not be trained to teach ICT, but rather will be trained on courses with “Computer Science” in the title. Guidance on the subject knowledge needed is available to guide recruiters. Graduates with a computer science degree are now being encouraged to consider teaching as a career and the profile of entrants to teaching is changing. IT professionals are being sent promotional material to promote teaching as a career they may consider. It is perhaps too early to discover if it will be easier to recruit teachers on to postgraduate teacher training programmes in computer science than it was for ICT whilst in this period of transition.

This demonstrates that one key issue for the UK will be to encourage suitable entrants to the teacher profession, and this is a common issue found internationally [Schulte et al. 2012]. Schulte et al found that common areas of concern across the countries they surveyed included the following: a lack of qualified teachers, not enough teacher education for computer science teachers and a lack of support for computer science teacher education.

There are 438,000 teachers employed in England. Data from the School Workforce Census taken at the end of 2011 showed that there were 30 vacancies in England in ICT (this is before Computing was introduced, so teachers delivering Computing were still known as ICT teachers), representing 0.4% of the total workforce [Department for Education 2012a] – see table I. This is second only to mathematics, and compares to an average figure of 0.3%. However even this may not be an accurate figure as we are aware that many “ICT” teachers have been borrowed from other subject areas. The number of posts where there is not a suitably qualified ICT teacher may be higher. This is an issue which will need to be addressed as Computing comes into the National Curriculum in 2014.

6.2. Skills of existing teachers

Another significant issue in the UK is the upskilling of teachers to be able to deliver the new curriculum [Sentance et al. 2012]. Professional development for teachers that offers deep understanding of the subject and transformational change for the individual is essential.

Teachers in England, Wales and Northern Ireland have been trained for the last 20 years to teach ICT without computer science. A few teachers teach Computing to the 16+ age group so are able to more easily adapt to the recent changes in the curriculum. However there has been a decline in students taking A-Level Computing (post-16) and this has meant that teachers of this subject have had to be redeployed. Others may have not used their degree for many years, or may have a degree subject that did not include any computer science. With the rapid pace of change we have seen, there is an urgent need to provide professional development in a range of forms for these teachers.

A small survey of 86 UK teachers [Sentance et al. 2013] revealed that 71% felt that they needed “guidance on ways of teaching computing”, in other words the pedagogical content knowledge (PCK) [Shulman 1986]. Other topics that teachers wanted were introductory and advanced programming courses, to give them the confidence they needed for this aspect of the curriculum.

The rebranding of ICT to Computing implies that the responsibility for the changed curriculum will fall on those teachers in ICT departments. The phrase ‘qualified to

Table I. Teacher vacancies in England [Department for Education 2012a]

	VACANCIES AS A PERCENTAGE OF TEACHERS IN POST ²		NUMBER OF VACANCIES	
	2010	2011	2010	2011
ALL VACANCIES	0.4	0.3	630	520
MAIN TEACHING SUBJECT				
Mathematics	0.7	0.5	120	100
Information technology	0.5	0.4	40	30
All sciences	0.4	0.4	80	90
Languages	0.3	0.4	30	40
English	0.5	0.4	110	80
Drama	0.3	0.1	10	-
History	0.2	0.3	10	20
Social sciences	0.6	0.3	20	10
Geography	0.2	0.2	10	10
Religious education	0.3	0.2	10	10
Design and technology	0.4	0.2	40	20
Commercial/business studies	0.5	0.1	20	-
Art/craft/design	0.2	0.1	10	10
Music	0.2	0.2	10	10
Physical education/sport/dance	0.2	0.1	30	20
Careers	-	-	-	-
Other main and combined subjects	0.5	0.4	50	50
Unknown subjects	.	.	30	20

Source: School Workforce Census

Table II. Numbers of ICT teachers [The Royal Society 2012]

	Number
ICT Teachers in England	18,400
Hold relevant qualification	6,400 (35%)
Hold relevant 1st degree & teacher training qualification	4,600 (25%)

teach' also needs to be understood before determining the number of teachers who fit that descriptor and thus what the training challenges actually are. An easy benchmark is academic qualifications, specifically post-A-Level qualifications for the subject they are being required to teach.

The Royal Society report [The Royal Society 2012] records data published in November 2011 from the Department for Education, as shown in Table II. When compared with other core curriculum subjects (see Table III) these figures are low [The Royal Society 2012], with many more science teachers having a relevant qualification than ICT teachers.

However, there are a large number of teachers who have entered the profession from other employment whose work-based experience provides sufficient background and understanding of the subject. In addition those who are currently teaching ICT from a non-IT background have built up a wealth of experience and understanding of their subject on the job. The change to the ICT curriculum is a sufficient change of direction that it is fair to regard it as a new subject and the need to retrain (or 'upskill') the

Table III. Qualifications of Science teachers [The Royal Society 2012]

Subject	% with appropriate A-Level qualification
Mathematics	74%
Physics	69%
Chemistry	73%
Biology	88%
ICT	35%

existing workforce is a challenge that can not be underestimated. Ofsted reported that “[ICT] teachers’ subject knowledge was weakest in data logging, manipulating data and programming” [Ofsted 2009]

6.3. CAS Network of Excellence

The level of understanding and expertise with respect to computer science amongst existing teachers is insufficient for the national implementation of computer science as a school subject. There are number of interconnected issues:

- Teacher confidence and experience. Some teachers are scared of the subject, others overwhelmed; some concerned that their pupils know more than they do thus exacerbating their nervousness.
- Pedagogy. Teachers need to know how to approach the subject and better understand the particular requirements of the subject to enable learning.
- The lack of Continuing Professional Development (CPD)²⁹ opportunities across a range of topics and skills. Where opportunities do exist accessing them is difficult due to budget constraints and increasing teacher workload.
- Support. Teachers need to know where they can access guidance and expertise.

What is needed is a national school network for computer science teachers, which, supported by universities, employers, learned societies and professional bodies, can provide somewhere for the teacher to go for support and training. The CAS/BCS Network of Computer Science Teaching Excellence³⁰ was started to address this challenge and develop a network, building on the successful Computing At School regional network, that aims to support a thousand secondary schools in three years (a third of the total number of state-maintained secondary schools in England); with the appropriate regional coverage and support, the aim is for this network to becoming self-sustaining at the end of the three years.

With support from the Department for Education, BCS, The Chartered Institute for IT, Microsoft, Google, CPHC, OCR and AQA the Network was launched and in its first six months provided over 700 hours of CPD contact time through university partners or CAS Master Teachers. An increase in funding was announced March 2013 by the Department for Education which will see the work of the Network increase to create a cohort of 600 “Master” Computer Science Teachers and support the work of university partners in providing subject knowledge CPD. Within the Network of Excellence programme the government funding is used to release Master Teachers from school for half or one day per week to prepare and deliver training to other teachers. This is a key part of the model and recognizes that a major issue for teachers is actually time for professional development [Sentance et al. 2012; Thompson et al. 2013]. The structure of the Network is shown in figure 7.

At the heart of the Network of Teaching Excellence in Computer Science are a number of principles:

²⁹The term CPD is used to refer to training and other activities that support the professional development of teachers

³⁰<http://www.computingatschool.org.uk/index.php?id=noe>

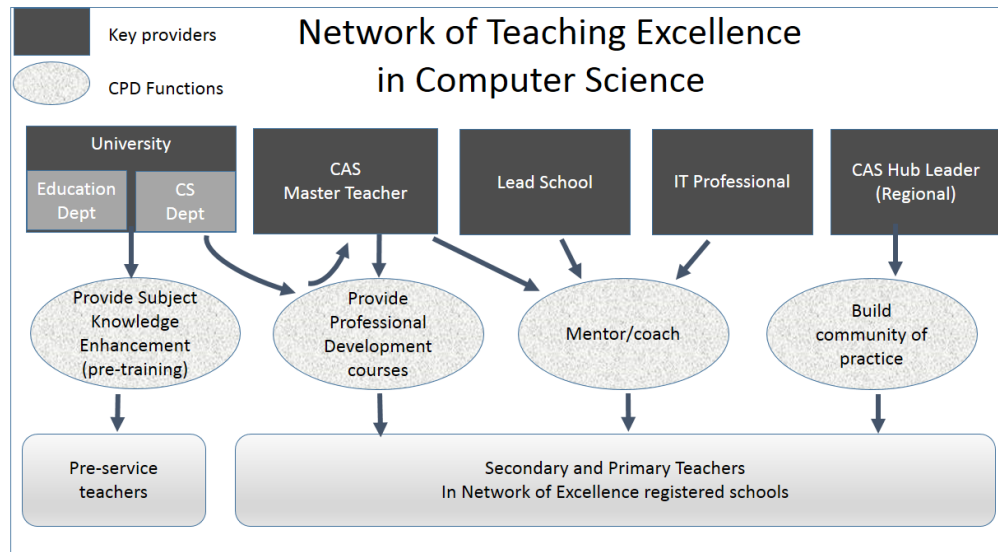


Fig. 7. The structure of the CAS Network of Excellence.

- Face to face delivery
- Local delivery
- Teachers teaching teachers

6.3.1. Face to face delivery. There are a plethora of online courses available but these are too often targeted at a non-teacher audience for whom the acquisition of a new skill or understanding of theory is only one part of their problem. Also, teachers spend their working lives dealing face to face with children and students. This is the world they understand and the interaction thus achieved is more effective than the online experience. Such online resources are not to be ignored but where possible should not take the place of face to face.

6.3.2. Local delivery. There should be no need for teachers to have to travel long distances to access CPD. Most work long hours as it is and obtaining permission from headteachers for whole days out of school to attend CPD is becoming ever more difficult. Leaving school early perhaps to attend a training session in their town or city is preferable to expecting them to travel to e.g. London. This also keeps costs down.

6.3.3. Teachers teaching teachers. Credibility gained for trainers with existing classroom practice cannot be underestimated with potential trainees. There are many teachers with enough expertise in the subject coupled with years of classroom practice who are best placed to share their knowledge with other teachers, as peers. There is an additional role for those in higher education (Computer Science and Education) departments to support the teachers with subject knowledge. Universities are a great resource for teachers seeking to increase their subject knowledge [Black et al. 2013] and those in the UK have shown a great willingness to support teachers in this way.

The Network of Excellence builds on the foundations created by the Computing At School regional Hubs, connecting teachers with other teachers and with university computer science and education departments. It will also connect schools to IT professionals in their locality; provide courses in both subject knowledge and pedagogy

in their local area; access the experience and expertise of teachers to support their colleagues in both primary and secondary schools and provide career development opportunities for teachers.

Between 2013 and 2018 the CAS Master Teachers, working in partnership with universities, employers, Teaching School alliances and professional bodies will offer not-for-profit computing CPD to schools in England so that by the end of this period:

- 600 Computer Science Master Teachers have been recruited from secondary and primary schools.
- Each Master Teacher supports 40 local schools by designing and running not-for-profit CPD activities for those schools, totaling 4,000 secondary schools and 20,000 primary schools within the Network
- The university computer science departments will lead on the training and development of the Master Teachers.
- The network partners will maintain and develop a comprehensive set of classroom ready resources covering all key stages.

6.4. Other support for teachers

Teachers need a professional identity [Ni and Guzdial 2012] and in this scenario they need to feel that the role of “Computer Science high school teacher” is one that exists and that they can identify with. To be able to provide this for teachers we need to give them confidence in their own abilities and a network in which they feel they belong. We also need to recognise expertise and excellence amongst teachers and the Master Teacher programme detailed above is the CAS vehicle for achieving this. It is clear how we accredit new teachers starting to train from 2013 but an issue exists of how we accredit existing teachers who are putting in a significant amount of time and effort to feel qualified to deliver the new Computing curriculum (see section 6.5)

What has been developing in the UK, through the enthusiasm and participation of keen and excited teachers, are regional networks of teachers where teachers meet and support each other and share resources. Mostly these run within the CAS system of local “Hubs”. Such meetings are held after school to discuss computer science teaching issues. Guest speakers are invited to share their own areas of expertise and teachers also share their experiences and resources. There are more than 70 hubs across the UK, and these provide a community of practice [Wenger 1998] for participating teachers where they can discuss issues relating to teaching computer science in school and find out about new developments and resources. Hubs are also setting up their own localised professional development sessions, for example, learning to use Scratch or Python.

In addition to regional activities, teachers are willing to share resources via networks and discussion groups. Again, CAS is providing a system for doing this with a website for teachers to upload and give feedback on useful resources [Brown and Kölling 2013]. This reflects the way in which teachers feel that they can make a positive contribution to developing computer science in their schools and the grass roots’ mentality underlying many of the evolving changes in the UK. Enthusiastic practitioners have ideas which they share with others. Experienced teachers support other teachers. Some formal training courses are set up, but these are often delivered by teachers based on their own experiences.

6.5. Residual issues

Although great progress is being made with respect to upskilling the Computing teaching workforce and bringing new entrants into the professions, there remain some issues that will face the UK over the coming years. In particular, schools and teachers

in England are facing these now, being a little further forward in the process of change than some of the other devolved nations.

- **Rural areas.** The plans around the Network of Teaching Excellence are to reach as many teachers as possible in their local area and on a face-to-face basis. This may not always be possible in more remote and rural areas. Online provision is being considered as a solution to dealing with this problem.
- **School networks.** We have a system in our schools where school networks may be managed regionally, with control over what can and cannot be installed or used held at a level above an individual school. This can cause problems where teachers find new tools or programming languages that they wish to incorporate into their teaching yet they are prevented from installing particular software. This can be frustrating for enthusiastic teachers and it will take a little time for solutions to these issues to be found, as many different parties are involved. This may sound like a minor issue but its impact should not be underestimated: for some schools it has been a blocker on offering computing in the school.
- **Accreditation.** The issue of how to accredit experienced teachers who have converted to teaching computer science in school is not an easy one. However it is obviously important not only to the teachers themselves, but to schools that either wish to employ them or be sure that their existing staff are qualified to deliver qualifications offered by the school. As the work on computer science education in schools in Israel showed [Hazzan et al. 2008], an appropriate CS teaching qualification is an essential part of establishing a successful program in schools. We have this in place for new teachers, but not yet for teachers converting from ICT.

These three issues are important to teacher development and to the further development of CS education in schools in the UK and will continue to be addressed.

7. CONCLUSIONS

Computer science in UK schools has gone from almost extinct to mandatory in the space of a five year period. This remarkable turn-around was achieved through several steps:

- Industry voices helped to draw the government's attention to the issue, and lend credence to the idea that there was a problem.
- Concerted lobbying first focused on educating government ministers/departments about the difference between IT and computer science, and then presented an argument that computer science was valuable for *all* students, not purely those who would study the subject to degree-level.
- This lobbying was accompanied by a grass roots movement among computing teachers (many of whom were teaching computing under the banner of ICT). This showed that the lobbying being performed had teacher support, but also put in place a peer-to-peer teacher support network that would later be needed to help upskill teachers.
- The first step in re-introducing computing was to introduce qualifications that the schools could study for – this was then followed by making the qualifications valuable by having the government endorse them and include them in school performance measures. This in turn laid the groundwork for reintroducing computer science throughout the curriculum.
- The changes to the curriculum were benefited by being able to mutate the existing ICT curriculum into a computing curriculum (with elements of digital literacy, IT, and computer science) rather than attempting to introduce an entire new subject onto the curriculum.

While computing is mandatory on the new curriculum for ages 5–14, there remains some work to be done at the 16–18 age range. The A-Level qualification offered there now overlaps substantially with the new GCSE qualification (which precedes it), and numbers taking the A-Level are still in decline. Eventually, the changes to the qualifications will also have an impact at university level.

Many challenges still remain. The curriculum may have been reformed, but the delivery of the subject is a massive challenge. There are insufficient numbers of trained teachers, and little or no resources (or time) to train up new and existing computing teachers. Many ICT teachers have no formal training in ICT, and are now being asked to transition to computing, with no officially allotted time or resources to help them. There remains a real danger that the gap between curriculum and actual delivery may scupper the resurgence of the computer science thread of the curriculum.

The CAS group have attempted to mitigate this problem by setting up a national training network, whereby “lead” schools with expertise in delivering computer science can assist those nearby schools without it. Several of these lead schools have “master teachers” who are seconded for roughly one day a week to deliver training and support for their local schools. This provides an interesting model for how to train up a large number of CS teachers (the same problem faced by the CS10K initiative in the USA) and it will be interesting to observe its progress.

Finally, we hope that the UK’s successes and challenges may provide useful information for those facing the same problems in other countries. The UK is certainly not alone in realising the deficiencies of computer science education in schools and trying to do something about it. The USA, for example, has long been conscious of areas for improvement where developing more CS in school is concerned [The Computer Science Teachers Association 2010; Wilson and Harsha 2009]. Many other countries may also face the same problem.

REFERENCES

- Stuart Arthur, Tom Crick, and Janet Hayward. 2013. *The ICT Steering Group’s Report to the Welsh Government*. <http://wales.gov.uk/about/cabinet/cabinetstatements/2013/ict/?lang=en>
- Tim Bell, Jason Alexander, Isaac Freeman, and Mick Grimley. 2009. Computer Science Unplugged: School Students Doing Real Computing Without Computers. *The New Zealand Journal of Applied Computing and Information Technology* 13, 1 (2009), 20–29.
- Diana Bitto and Claudio Mirolo. 2013. Archaeology of Information in the Primary School. In *Informatics in Schools. Sustainable Informatics Education for Pupils of all Ages*. Lecture Notes in Computer Science, Vol. 7780. Springer, 115–126.
- Jonathan Black, Jo Brodie, Paul Curzon, Chrystie Myketiak, Peter W. McOwan, and Laura R. Meagher. 2013. Making Computing Interesting to School Students: Teachers’ Perspectives. In *Proceedings of the 18th ACM Conference on Innovation and Technology in Computer Science Education (ITiCSE ’13)*. ACM, New York, NY, USA, 255–260. DOI: <http://dx.doi.org/10.1145/2462476.2466519>
- Neil C. C. Brown and Michael Kölling. 2013. A tale of three sites: resource and knowledge sharing amongst computer science educators. In *Proceedings of the 9th Annual International ACM Conference on International Computing Education Research (ICER ’13)*. ACM, New York, NY, USA, 27–34. DOI: <http://dx.doi.org/10.1145/2493394.2493398>
- Neil C. C. Brown, Michael Kölling, Tom Crick, Simon Peyton Jones, Simon Humphreys, and Sue Sentance. 2013. Bringing Computer Science Back Into Schools: Lessons from the UK. In *Proceedings of the 44th ACM Technical Symposium on Computer Science Education (SIGCSE 2013)*. ACM Press, 269–274.
- CBI. 2012. *First Steps: A new approach for our schools*. <http://www.cbi.org.uk/campaigns/education-campaign-ambition-for-all/>
- Tom Crick and Sue Sentance. 2012. Computing At School: Stimulating Computing Education in the UK. In *Proceedings of the 11th Koli Calling International Conference on Computing Education Research*. ACM Press.
- Department for Education. 2010. *The Importance of Teaching - The Schools White Paper*. <https://www.gov.uk/government/publications/school-workforce-in-england-november-2011>

- Department for Education. 2011. *National Curriculum Review Launched*. <https://www.gov.uk/government/news/national-curriculum-review-launched>
- Department for Education. 2012a. *School Workforce in England, November 2011, Statistical First Release, SFR 06/2012*. <https://www.gov.uk/government/publications/school-workforce-in-england-november-2011>
- Department for Education. 2012b. *Statistical First Release: Schools, pupils, and their characteristics, January 2012*. <https://www.gov.uk/government/publications/schools-pupils-and-their-characteristics-january-2012>
- Department for Education. 2013. *National Curriculum from September 2014*. <https://www.gov.uk/government/collections/national-curriculum>
- Douglas Rushkoff. 2011. *Program or Be Programmed: Ten Commands for a Digital Age*. OR Books.
- Stephen Doyle. 1988. *GCSE Computer Studies for You*. Hutchinson Education.
- e-skills UK. 2012. *Technology Insights 2012*. <http://www.e-skills.com/research/research-publications/insights-reports-and-videos/technology-insights-2012/>
- European Commission. 2012. Digital Agenda for Europe. <http://ec.europa.eu/digital-agenda/digital-agenda-europe>. (2012).
- Orit Hazzan, Judith Gal-Ezer, and Lenore Blum. 2008. A model for high school computer science education: the four key elements that make it!. In *Proceedings of the 39th ACM SIGCSE Technical Symposium on Computer Science Education*. ACM Press, 281–285.
- Peter Hubwieser, Michal Armoni, Torsten Brinda, Valentina Dagiene, Ira Diethelm, Michail N. Giannakos, Maria Knobelsdorf, Johannes Magenheimer, Roland Mittermeir, and Sigrid Schubert. 2011. Computer science/informatics in secondary education. In *Proceedings of the 16th Annual Conference Reports on Innovation and Technology in Computer Science Education – Working Group Reports (ITiCSE-WGR '11)*. ACM Press, 19–38.
- Informatics Europe & ACM Europe Working Group on Informatics Education. 2013. *Informatics education: Europe cannot afford to miss the boat*. <http://europe.acm.org/iereport/ACMandIEReport.pdf>
- Ian Livingstone and Alex Hope. 2011. Next Gen. Nesta. <http://www.nesta.org.uk/publications/next-gen>
- Lijun Ni and Mark Guzdial. 2012. Who AM I?: understanding high school computer science teachers' professional identity. In *Proceedings of the 43rd ACM Technical Symposium on Computer Science Education (SIGCSE '12)*. ACM Press, 499–504.
- OCR. 2011. *OCR GCSE in Computing J275 specification, version 2 September 2011*. <http://www.ocr.org.uk/Images/81949-specification.pdf>
- Ofsted. 2009. *The importance of ICT: information technology and communication technology in primary and secondary school 2005–2008*. <http://www.ofsted.gov.uk/resources/importance-of-ict-information-and-communication-technology-primary-and-secondary-schools-20052008>
- Ofsted. 2011. *ICT in schools 2008–11*. <http://www.ofsted.gov.uk/resources/ict-schools-2008-11>
- Ofsted. 2012. *Mathematics: Made to Measure*. <http://www.ofsted.gov.uk/resources/mathematics-made-measure>
- Carsten Schulte, Malte Hornung, Sue Sentance, Valentina Dagiene, Tatjana Dagiene, Neena Thota, Anna Eckerdal, and Anne-Kathrin Peters. 2012. Computer Science at School/CS Teacher Education: Koli Working-group report on CS at School. In *Proceedings of the 12th Koli Calling International Conference on Computing Education Research (Koli Calling 2012)*. ACM Press.
- Sue Sentance, Mark Dorling, and Adam McNicol. 2013. Computer Science in Secondary Schools in the UK: Ways to Empower Teachers. In *Informatics in Schools. Sustainable Informatics Education for Pupils of all Ages (Lecture Notes in Computer Science)*. 15–30.
- Sue Sentance, Mark Dorling, Adam McNicol, and Tom Crick. 2012. Grand challenges for the UK: upskilling teachers to teach computer science within the secondary curriculum. In *Proceedings of the 7th Workshop in Primary and Secondary Computing Education (WiPSCE '12)*. ACM Press, 82–85.
- Lee S Shulman. 1986. Those who understand: Knowledge growth in teaching. *Educational researcher* 15, 2 (1986), 4–14.
- Lawrence Snyder. 2012. Status Update: High School CS Internationally. *ACM Inroads* 3, 2 (June 2012), 82–85. DOI: <http://dx.doi.org/10.1145/2189835.2189861>
- The Computer Science Teachers Association. 2010. *Running On Empty: The Failure to Teach K-12 Computer Science in the Digital Age*.
- The Independent ICT in Schools Commission. 1997. *Information and Communications Technology in UK Schools: An Independent Inquiry*.

- The Royal Academy of Engineering. 2012. *Computing qualifications included in the 2014 Key Stage 4 Performance Tables: a guide for schools*. http://www.raeng.org.uk/news/publications/list/reports/computing_qualifications.pdf
- The Royal Society. 2012. Shutdown or restart? *The way forward for computing in UK schools*. <http://royalsociety.org/education/policy/computing-in-schools/report/>
- David Thompson, Tim Bell, Peter Andreae, and Anthony Robins. 2013. The Role of Teachers in Implementing Curriculum Changes. In *Proceeding of the 44th ACM Technical Symposium on Computer Science Education (SIGCSE '13)*. ACM, New York, NY, USA, 245–250. DOI: <http://dx.doi.org/10.1145/2445196.2445272>
- Teresa Tinsley and Youping Han. 2012. *Language learning in secondary schools in England*. CfBT Education Trust. <http://www.cfbt.com/en-GB/Research/Research-library/2012/r-language-learning-in-secondary-schools-in-england-language-trends-2012>
- UK Office for National Statistics. 2011. *UK Census 2011*.
- Universities and Colleges Union. 2012. Choice cuts: *How choice has declined in higher education*. http://www.ucu.org.uk/media/pdf/c/h/Choice_cuts_report_Feb12.pdf
- Etienne Wenger. 1998. *Communities of practice: Learning, meaning, and identity*. Cambridge University Press.
- Cameron Wilson and Peter Harsha. 2009. IT Policy: The Long Road to Computer Science Education Reform. *Communications of the ACM* 52, 9 (2009), 33–35.
- Jeannette M. Wing. 2006. Computational Thinking. *Communications of the ACM* 49, 3 (2006), 33–35.
- Alison Wolf. 2011. *Review of Vocational Education – The Wolf Report*. https://www.gov.uk/government/uploads/system/uploads/attachment_data/file/180504/DFE-00031-2011.pdf

Received July 2013; revised March 2014; accepted March 2014