Report

Computer Systems and Networking Within the Computing Curriculum

July 2021

Teach Computing

Raspberry Pi

Summary

This report explores two strands of the computing curriculum in England: computer systems and computer networks. It emphasises the foundational nature of these topics and the role they play in supporting learning across other strands of computing. The main goal of this report is to demonstrate learners' progression in this area of the curriculum and appropriate pedagogical strategies for these two strands. We have included many supporting resources from the National Centre for Computing Education (NCCE) which reflect the approach present in this report. The report consists of six sections:

- In Section 1, we discuss the role and importance of the computer systems and computer networks strands of the curriculum. We emphasise the value in learners developing a robust mental model (notional machine) of how these systems work. This understanding of the system and its limits is to benefit learners as they discover how to use, consume, and create with technology.
- Section 2 proposes a theme/tier model for this area of the curriculum, which has several big (and overlapping) themes that can be explored by learners in ever increasing detail. We also include examples of topics and concepts for each area of the model.
- We use this model in Section 3 to map relevant objectives from NCCE curriculum content for key stage 1 to 5. With this model, we can observe a general trend from the highest (most abstract) tier to the lowest (most detailed) tier, as learners progress through their education. We use this data to present an overall progression of 'big ideas' in computer systems and networks in the form of a learning graph.
- Given that the computer systems and networks strands are predominantly made up of concepts (rather than skills), Section 4 recommends five pedagogical principles and associated practices that computing educators can apply to support their practice. Each of these principles and practices come with associated further reading.
- Section 5 covers the many opportunities for teachers to develop their own skills, understanding, and practice through a range of professional development offered by the NCCE.
- To conclude, Section 6 provides an overview of the key findings of this report.

Contents

1. Introduction

The use and application of computing devices is present in many areas of our world, from the personal devices we use on a daily basis, to the systems that control and automate industrial processes. Through these devices we are almost always connected to a wider network in some form or another.

It is important that educators equip learners with the knowledge and skills to thrive in a world where they are surrounded by computer systems and networks. An understanding of how our networks, systems, and devices work is foundational knowledge for all students of computing. Empowered with this knowledge, learners can understand the advantages and limitations of computer systems, and discover how data is transmitted and the associated risks. The same understanding helps learners develop mental models of how computers operate, interpret, and execute instructions. Accurate mental models (notation machines[1]) can also support learners as they develop their programming skills, which in turn help them to avoid common misconceptions. An understanding of how systems and networks work, and the factors that affect their performance, enables learners to design and evaluate solutions to real world scenarios, as well as understand the impact of those solutions on our lives.

Once your learners have a better understanding of how computers and computer systems influence their lives, you can encourage them to take control of digital devices. They can begin to see devices such as computers and tablets as platforms on which they can express themselves creatively, or look for solutions to problems, instead of just consuming content produced by other people.

Computers are so central to our everyday tasks and routines that computers and networks are often hidden; in some circumstances, people may not even be aware that they are interacting with a computer. Your learners might not realise that a computer system controls automatic doors in a supermarket, or the timing of a pedestrian crossing. Therefore, the study of computer systems and networks begins with identifying their presence all around us, from the general purpose computers we use regularly, to embedded systems around the home and school, to the devices that help automate many of our activities.

Once learners identify the systems and networks around them, they can begin to look inside and discover the devices and components that work together to perform a task. Learners begin to think about the inputs to a system, the processes it carries out, as well as the outputs it produces. This input, process, output (IPO) model is then evident throughout the study of computer systems and networks.

[1] Fincher, S., Jeuring, J., Miller, C.S., Donaldson, P., du Boulay, B., Hauswirth, M., Hellas, A., Hermans, F., Lewis, C., Mühling, A. and Pearce, J.L. (2020) Notional Machines in Computing Education: The Education of Attention. In *Proceedings of the Working Group Reports on Innovation and Technology in Computer Science Education* (pp. 21-50).

Delving deeper still, learners can begin to explore how the different devices and components work together, and how the system is implemented. From this perspective, they explore the logical structures that underpin computation, the various methods developed to physically store data, and the layers and protocols that ensure rapid, reliable, and secure transmission of data.

The study of computer systems and networking can be characterised as:

- A closer look at the computer systems all around us and how they communicate
- The identification and demystification of computing devices and processes
- An exploration of a system from the outside in, beginning with its purpose or context, its inputs and outputs, and its constituent parts and how they work together

The National Centre for Computing Education (NCCE) was launched in 2018 to work with schools across England to support the teaching of computing. In the first two years, we have engaged with 29,500 teachers, of which 7,500 teachers have participated in professional development. The NCCE includes 34 regional Computing Hubs that take a leadership role in their localities and support schools to deliver a high quality computing education.

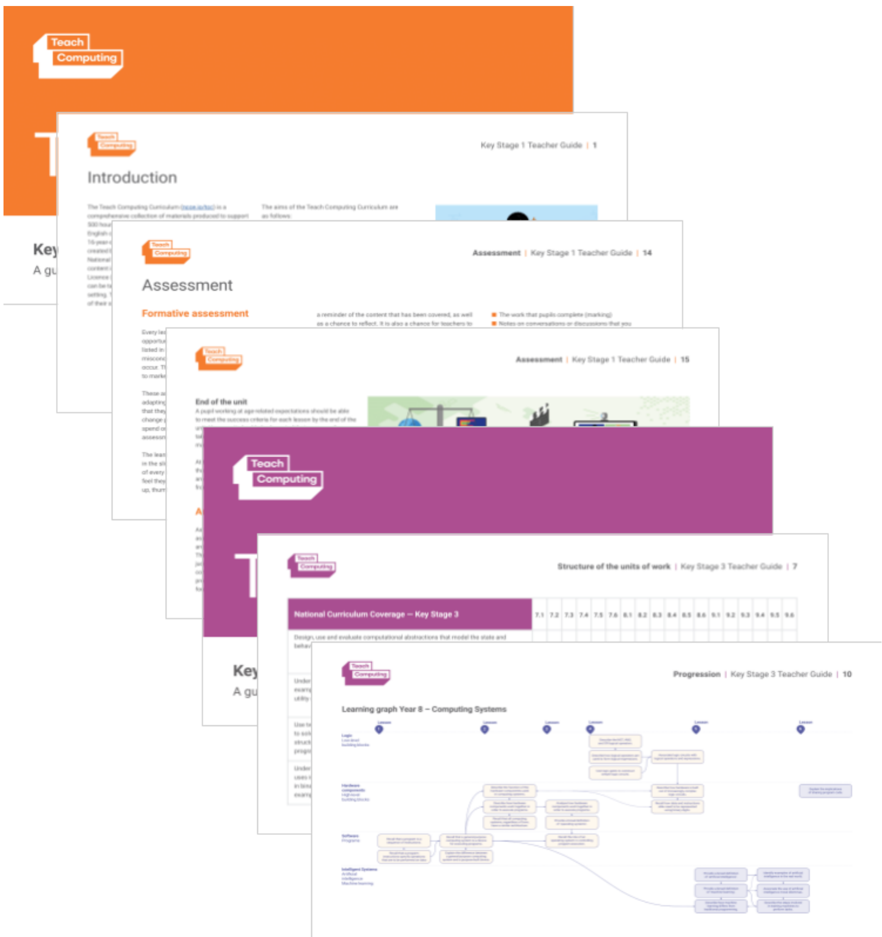

Figure 1: Teach Computing Curriculum teacher guides

The NCCE's role has been to support the **entire computing curriculum**. A central part of this role has been the development of the Teach Computing Curriculum[2], which offers teaching resources for each stage of the curriculum. This groundbreaking, and freely available, curriculum supports teachers and learners alike on a journey from key stage 1 to 4; it builds upon the latest research, as well as years of expert teaching experience.

You can read more about our approach to curriculum design in our teacher guides for all key stages. In each guide, we describe how units are structured, the progression within and between units, as well as emphasising appropriate pedagogical approaches.

To complement these curriculum resources is the Isaac Computer Science[3] website, which provides direct support to learners studying A level (and soon GCSE) computer science.The combination of computing content and high quality questions makes the resource ideally suited to the classroom, self study, and revision.

Together, the Teach Computing Curriculum and Isaac Computer Science cover the teaching of computing and computer science from key stage 1 to 5 (5- to 18-year-olds). Both have been expertly designed with progression in mind and exemplify our approach to sequencing concepts and skills.

This report is part of a series of NCCE reports; each explores teaching and learning within a different aspect of the curriculum. The purpose of this report is to outline the ways in which the NCCE can support you with all aspects of the teaching and learning of computing systems and networks. It has been written in relation to the curriculum in England, although you may also find it interesting if you're reading this from another context. The intended audience is all serving teachers, prospective teachers, and educators involved in teaching computing, as well as those leading on remote education for their school.

[2] Teach Computing. (n.d.) *Teach Computing Curriculum*. [online] Available at: https://teachcomputing.org/curriculum [Accessed 21 Jun. 2021].
[3] Isaac Computer Science. (n.d.) *Isaac Computer Science*. [online] Available at: https://isaaccomputerscience.org/ [Accessed 21 Jun. 2021].

2. Computing systems and networks in the national curriculum

An understanding of computing systems and networks is essential to access the national curriculum for England. Computing systems and networks feature throughout the curriculum, from learners' awareness of what computer systems are at key stage 1, to a detailed understanding of how computer systems and networks function by the time learners reach key stage 5. By understanding these core principles, learners can gain an insight into what happens behind the screen and develop a curiosity to find out more. This awareness prepares learners to access other areas of the computing curriculum, such as programming and creating media, where a knowledge of how information technology functions can broaden learners' appreciation of the potential of technology.

At its highest level, the national curriculum for computing in England[4] states that learners "can understand and apply the fundamental principles and concepts of computer science" and "can evaluate and apply information technology, including new or unfamiliar technologies, analytically to solve problems".

In examining curriculum statements across all key stages, it is possible to draw out many themes that relate to computing systems and networks. Depending on the desired level of detail, these themes can number into double figures, with some being present only at one stage of education. We analysed the national curriculum in England and identified four simple themes across computing systems and networks.

The first theme is **hardware**. This covers the physical devices and components that work together to form a computer system. The deeper learners explore this theme, the more they focus on how different components work, as well as the logical concepts and physical processes on which the system is built. In key stage 1, learners concentrate on being able to "recognise common uses of information technology". In key stage 2, learners are introduced to a "range of digital devices" and begin to understand the hardware that is needed for computer networks and the internet to function. They are also expected to apply this understanding as they learn to "control physical systems". In key stage 3, learners investigate the behaviour of physical systems and their underlying components to deepen their understanding of how computing devices work. They begin to explore the logical structures and circuits that all computing systems are built on. At key stage 4, all learners have the opportunity to develop this knowledge, with particular emphasis on the systems and networks they are likely to use in the world of work. Learners who work towards a computing GCSE qualification delve deeper to understand how a computer system works,

[4] Department of Education. (2013) *National curriculum in England: computing programmes of study*. [online] GOV.UK. Available at: https://www.gov.uk/government/publications/national-curriculum-in-england-computing-programmes-of-study/national-curriculum-in-england-computing-programmes-of-study. [Accessed 21 Jun. 2021].

with a particular focus on how the processor, memory, and storage work together to execute program instructions. Going even further, A level students at key stage 5 explore different computer architectures and develop their understanding of the logical constructs that all hardware is built upon.

The second theme is **software**. This encompasses internet services (including cloud computing), operating systems, applications, utilities (such as drivers), and assembly/machine-code language. The term software is introduced in the key stage 2 section of the curriculum, but only in terms of its use. In key stage 3, learners are introduced to the idea that hardware and software work together in computer systems and networks. By the time they reach key stage 4, learners should have an awareness of many different types of software and their practical application. Learners who work towards a computing GCSE qualification study the different categories of software and how these ultimately become instructions processed by the CPU. They begin to write their own programs, a process continued in greater depth at A level, along with a detailed knowledge of how programs are translated from high- to low-level languages.

The third theme is **network architecture**. This includes an understanding of the different types of computer networks, how they are physically connected, and the components needed to build them. This theme is first introduced in the national curriculum at key stage 2 in relation to the internet — it's many services and "the opportunities they offer for communication and collaboration". In key stages 3 and 4, learners explore the different types of networks and methods of connection. Learners who study for their computing GCSE and A level explore how multiple networks are connected, bridged, and organised.

The fourth theme is **data transmission**. This theme focuses on how data moves around networks. Data transmission is not explicitly covered in key stage 1, although learners are expected to use the internet, which is reliant on data transmission. In key stage 2, learners are introduced to how the internet enables communication and collaboration, and the importance of addressing and agreeing methods of communication. In key stage 3, the focus moves to how computer systems communicate with each other: learners explore IP addressing and some common protocols. Learners who study for their computing GCSE and A level use standard layered models to discover different protocols at each layer in computer networks.

While the above themes help describe the content within computer systems and networks, they are still very broad concepts. At different points in a learner's journey in computer systems and networks, they may explore the same or similar concepts, albeit from a different perspective or level of abstraction. For example:

- Within the **data transmission** theme, learners may first find out that devices within a network can communicate with each other. Later they explore the reasons why protocols are needed, and after they delve deeper, they become aware of a range of protocols and their uses. Eventually they become familiar with how those protocols are implemented.

- Similarly, within the **hardware** theme, learners begin by considering the inputs, processes, and outputs within an information system. They then consider the computing devices within the system and their inputs and outputs. After they look inside the device, they explore the common components within a computing device (memory, storage, CPU, etc.) to understand how the CPU and other components work.

In these examples, learners examine the system and wider networks from a range of perspectives, which we refer to as tiers:

- **System/network tier** – a highly abstract view in which learners focus on how systems and networks are used to solve problems
- **Device tier** – learners are concerned with familiar computing devices, including computers, phones, tablets, and embedded systems
- **Component tier** – learners look inside the device and understand the purpose of common constituent parts that make up every computing device
- **Implementation tier** – learners focus on the specific details of how the smallest components are built, how they work, and how they are controlled

From the analysis that follows, we can observe a general trend in progression from the highest, most abstract tier, to the lowest tier. (Figure 2)

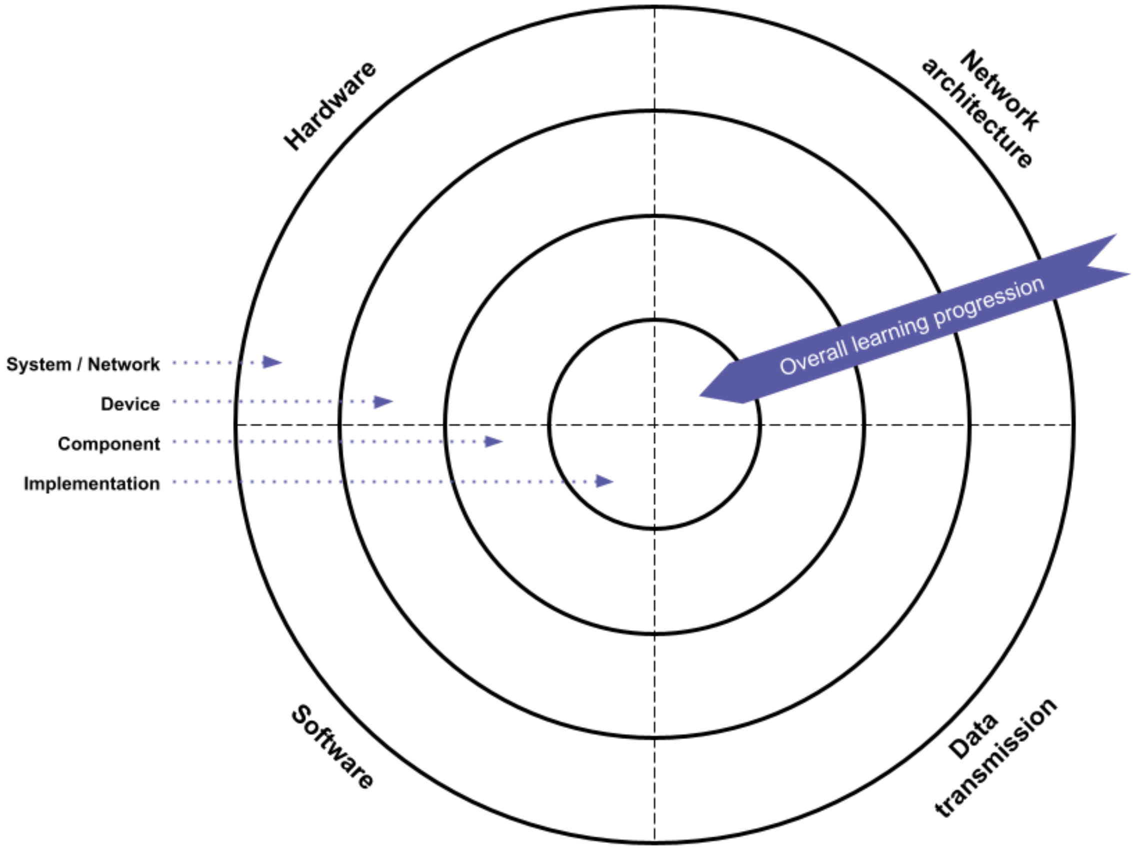

Figure 2: Our theme and tiered model of computer systems and networks

This report therefore proposes that the knowledge and skills within the computer systems and networking strands of the curriculum can be organised into four broad themes. These themes can then in turn be divided into four tiers of detail/abstraction, and the relevant concepts mapped to each of the 16 areas (see Table 1).

In this report, we look at each of the key stages for computing, examine how each key stage relates to the themes and tiers model, and show how learners can progress within each stage. The next section includes an overview of the Teach Computing Curriculum and its structure, followed by a summary of each key stage, and a presentation of learners' progression through the curriculum as a learning graph.

	Computer systems		Networks	
	Hardware	**Software**	**Network architecture**	**Data transmission**
System / Network	• Purposes of systems • Benefits of computer systems • Monitoring and controlling systems • Remote storage	• Web services • Cloud computing • Software as a service • Control systems software • Virtual machines	• What is a network • Purpose, uses, and risks • The internet • WAN, LAN, and PAN • Network topologies	• How and why we use networks • Data can be routed across a network • Network performance, bandwidth, and latency
Device	• General purpose vs purpose built embedded devices • Common device features • Peripherals	• Operating systems • Application software • Open/closed source • Human computing interaction • User interfaces	• Devices nodes within the network • Client-server and peer-to-peer networking • Thin and thick clients • Portable devices	• Connectivity • Role of protocols • Application layer protocols • DNS and IP addressing • Servers, email, web, etc.
Component	• Role of common components: ○ Storage ○ CPU ○ RAM, ROM, and cache ○ Sensors	• Utility software • Controlling IO devices • Hierarchy of programming languages	• Wired and wireless connections • Switches, access points, routers, gateways, bridges, etc. • NICs and WNICs	• Transport layer protocols • DHCP • Network address translation • Websockets
Implementation	• CPU architecture(s) • Fetch-decode-execute cycle • Logic gates/circuits • Buses • Interrupts • Storage media	• Device drivers and bios • Program translators • Assembly and machine-code language • Libraries, linkers, and loaders	• MAC addressing • Characteristics of transmission media, including copper, fibre optic, radio waves, etc.	• Network layered model(s) • Collision detection and avoidance • Multiplexing • Circuit and packet switching

Table 1: Computer systems and networks content organised by theme and tier

3. Computer systems and networks within the Teach Computing Curriculum

3.1 NCCE curriculum structure

As already mentioned, the NCCE exists to help teachers deliver the entire computing curriculum. The Teach Computing Curriculum, as well as other content from the NCCE, is built upon a 'taxonomy' system used to classify and catagorise content. This categorisation consists of ten strands that span the current national curriculum for computing in England. Each strand has a combination of skills and knowledge that feature throughout the national curriculum. Strands that are rich in knowledge form the basis of the units within the Teach Computing Curriculum. Other strands focus on skills across all units. Below is a summary table of the ten strands.

Strand	**Description**
Algorithms	Comprehend, design, create, and evaluate algorithms
Creating media	Select and create a range of media including text, images, sounds, and video
Computing systems	What a computer is and how its constituent parts function together
Design and development	The activities involved in planning, creating, and evaluating computing artefacts
Data and information	How data is stored, organised, and used to represent real world artefacts and scenarios
Effective use of tools	Use hardware and software tools to support computing work
Impact of technology	How individuals, systems, and society as a whole interact with computer systems
Networks	How networks can be used to retrieve and share information and come with associated risks
Programming	Create software to allow computers to solve problems
Safety and security	Understand risks when using technology and how to protect individuals and systems

Table 2: A summary of the ten strands in the NCCE content taxonomy

As shown in Table 2, the taxonomy contains two relevant strands: ‘Computing systems’ and ‘Networks’. These strands are the subject of this report, and focus on how the machine or system works. Therefore, these strands predominantly contain concepts to learn rather than skills to acquire.

In practice, these concepts are taught alongside other strands that focus on the application or evaluation of the concepts. Within the NCCE content taxonomy, there are three strands that closely relate to the ‘Computing systems’ and ‘Networks’ strands:

- Safety and security
- Impact of technology
- Effective use of tools

These three strands also intersect with other themes, such as ‘Programming’, ‘Algorithms’, and ‘Creating media’. For the purposes of this report, we have considered these themes to be out of scope, and instead focus solely on the progression of concepts within the ‘Computer systems’ and ‘Networks’ strands.

For more information on these themes and how they are addressed in the Teach Computing Curriculum, please refer to the teacher guides for your key stages[2] and the NCCE Digital Literacy Within the Computing Curriculum[5] report.

[5] National Centre for Computing Education. (2021) *Digital literacy within the computing curriculum*. [online] Available at: https://blog.teachcomputing.org/digital-literacy-within-the-computing-curriculum/ [Accessed 21 Jun. 2021].

3.2 Progression of computer systems and networking

To explore progression within these crucial strands, we consulted the objectives from the Teach Computing Curriculum for key stage 1 to 4 (age 5- to 16-year-olds), including the GCSE content. In order to examine progression further, we considered the curriculum objectives covered by the Isaac Computer Science website, which represents all A level exam boards in the UK. Each objective is already catagorised to the NCCE taxonomy, which makes it easier to show those objectives that sit within the 'Computer systems' and 'Networks' strands.

Each of these 385 objectives were then collaboratively catagorised by the tiers and themes identified in Section 3.1 that best described them. This allows for a broad representation of the progression from each key stage to the next. The following sections present a breakdown of this progression by key stage, along with the concepts explored by learners at each point in their journey.

Each stage of the curriculum is different; some are longer stages than others, some are statutory while others are elective, and those resulting in a qualification generally involve many more teaching hours. As might be expected, each key stage includes a recap of concepts that have been encountered at an earlier stage. This is particularly noticeable at points of transition, such as when learners move from primary to secondary education or when learners choose to study a GCSE or A level qualification in computing. Due to varying provision in different schools, prior knowledge cannot be assumed. This creates a degree of overlap between key stages. These differences and intersections make direct comparisons between key stages challenging. Instead, this report attempts to describe the focus and progression within each educational stage.

Key stage 1

In key stage 1, there are 11 learning objectives that focus on computer systems and networking. These are found in the Year 1and 2 units[6]:

- Year 1 – Technology around us
- Year 2 – Information technology around us

When these are mapped against the theme and tier model (Table 2), it is clear that there is a fairly narrow focus at this stage and from a relatively abstract perspective.

	Computer Systems		Networks	
	Hardware	Software	Network Architecture	Data Transmission
System / Network	2	0	0	0
Device	9	0	0	0
Component	0	0	0	0
Implementation	0	0	0	0

Table 3: Computing systems and networks content at key stage 1

While learners use a range of hardware and software during this phase, as well as accessing networks and web services, their conceptual focus is on the hardware itself and the role it plays in their everyday lives. In line with the computing programme of study at this stage, learners have no explicit objectives relating to software, networks, or the internet. Instead, the focus is on answering the following questions:

- What is a computing device?
- What is an information system?
- Where can we find examples of such devices and systems?
- What do these devices and systems have in common?
- How do we interact (input and output) with these devices?
- How do we benefit from the everyday use of computing systems and information technology?

As acknowledged in a previous report[5] about digital literacy, learners begin their schooling with a range of experiences of computing devices. One important goal at this stage is to

[6] Teach Computing. (n.d.) *Key stage 1*. [online] Available at: https://teachcomputing.org/curriculum/key-stage-1 [Accessed 21 Jun. 2021].

pool and build on that broad experience to help learners develop a shared understanding of computer systems and their uses.

At this stage of education, there is some significant crossover between the computer systems concepts and skills that are the focus of this report and the digital literacy competencies previously identified. As well as learning what is considered to be a computer system, learners investigate the purpose of many input and output devices. Learners can immediately apply this knowledge by regularly using a range of devices and their peripherals.

Below is a list of key concepts and skills that learners encounter at this phase of the Teach Computing Curriculum. (**Note:** This is not an exhaustive list, but a summary of the most important ideas at stage.)

Skill/Concept	Tier	Theme	Curriculum units
Identify examples of common computing devices (laptops, desktops, tablets, phones, etc.)	Hardware	Device	• Y1 Technology around us
Identify the common features of a computing system (screen, keyboard, mouse, camera, etc.)	Hardware	Device	• Y1 Technology around us
Understand the purpose of common input and output devices (keyboard, mouse, screen)	Hardware	Device	• Y1 Technology around us
Recognise technology and computing systems in their everyday lives	Hardware	Device	• Y2 Information technology around us
Compare the features and uses of a range of computing systems	Hardware	System	• Y2 Information technology around us
Understand the benefits and limitations of computing systems	Hardware	System	• Y2 Information technology around us

Table 4: Key skills and concepts at key stage 1

Key stage 2

As learners progress into, and through, key stage 2, they continue to develop their knowledge of hardware, as well as broadening their understanding to cover aspects of software, network architecture, and data transmission. The majority of their understanding is explored either from the fairly abstract 'System/Network' perspective or delving a little deeper exploring the 'Device' tier of our model.

Over their four years in this stage, and while following the Teach Computing Curriculum, learners encounter computer system and networking concepts in 21 objectives across six different units[7]. There is a unit dedicated to this topic for each year group, as well as **additional units** where computer systems and networking concepts are explicitly encountered (italicised below):

- Year 3 – Connecting computers
- Year 4 – The internet
- Year 5 – Sharing information
- *Year 5 – Selection in physical computing*
- Year 6 – Communication
- *Year 6 – Sensing*

	Computer Systems		Networks	
	Hardware	Software	Network Architecture	Data Transmission
System / Network	2	5	3	3
Device	3	1	1	1
Component	0	1	1	0
Implementation	0	0	0	0

Table 5: Computing systems and networks content at key stage 2

Within the **hardware** theme, learners encounter and apply the input, process, output model (IPO) to their use, classification, and application of hardware. Additionally, learners combine hardware and programming software to control electronic components in the real world. This connection of hardware and software through physical computing supports learners to

[7] Teach Computing. (n.d.) *Key stage 2*. [online] Available at: https://teachcomputing.org/curriculum/key-stage-2 [Accessed 21 Jun. 2021].

develop their own understanding of the notional machine while testing their understanding through programming.

Within the **software** theme, along with developing their skills in using a wide range of locally installed application software, learners begin to explore online services, including search engines, online office applications, and productivity tools. They begin to appreciate some of the benefits of using online tools as well as some potential drawbacks.

It is during this stage that learners first begin to explore the concept and workings of a network. They should view a network as a group of connected computing devices and have some appreciation of how those devices physically connect. Learners should be able to identify common components within the network and their purpose.

They learn about the need for addressing and common communication methods, as well as how data can flow along multiple paths to reach its destination.

Below is a list of key concepts and skills that learners encounter at this phase of the Teach Computing Curriculum. (**Note:** This is not an exhaustive list, but a summary of the most important ideas at stage.)

Skill/Concept	Tier	Theme	Curriculum units
Understand and apply the IPO model to devices and their components	Device	Hardware	Y3 Connecting computers
Compare life with and without technology, and identify the benefits and limitations	Device	Hardware	
Recognise common components of a computer network	Components	Network architecture	
Identify practical uses of networks, including sharing and collaboration	System/Network	Data transmission	
The internet is a network of networks that enable many services	System/Network	Network architecture	Y4 The internet

Understand how content is created, hosted, and shared on the internet	System/Network	Software	
Computing devices can be connected together to form systems	System/Network	Hardware	Y5 Sharing information
Data is broken into small ‘chunks’, called packets	Device	Data transmission	
Packets are delivered to the correct devices using unique IP addresses	Device	Data transmission	
Network devices need agreed methods of communication called protocols	Device	Data transmission	
Understand and evaluate different means of communicating online	System/Network	Software	
Write programs that control simple circuits and components	Component	Software	*Y5 Selection in physical computing*
Understand how search engines index, select, and rank results	System/Network	Software	Y6 Communication
Understand and evaluate a range of communication and collaboration methods online	System/Network	Data transmission	
Write programs that run on a controllable device	Device	Software	*Y6 Sensing*

Table 6: Key skills and concepts at key stage 2

Key stage 3

As learners embark on their secondary computing education, they should arrive with a firm foundation in their understanding of computer systems and networks. There will of course be some variation in their experience during key stages 1 and 2, but learners who have followed the Teach Computing Curriculum can be expected to:

- Identify examples of computing systems and networks around them
- Understand and apply the input, process, output (IPO) model to a variety of computing systems and their associated peripherals
- Understand the role of software in controlling hardware and providing a smooth user experience
- Differentiate between the internet and the many services that rely upon it, including the World Wide Web
- Have a basic understanding of how data is transmitted across networks, the need for IP addresses, and agreed communication methods (protocols)
- Create programs that use input and output devices, including screen, keyboard, sensors, LEDs, etc.

Across the seven units in key stage 3, learners focus on how the computing systems they use and the networks they access work[8]. There is a unit dedicated to this topic for each year group, as well as **additional units** where computer systems and networking concepts are explicitly encountered (italicised below):

- Year 7 – Networks: from semaphores to the internet
- *Year 8 – Developing for the web*
- Year 8 – Computing systems
- *Year 8 – Introduction to Python programming*
- *Year 9 – Representations: going audiovisual*
- *Year 9 – Physical computing programming*
- Year 9 – Cybersecurity

[8] Teach Computing. (n.d.) *Key stage 3*. [online] Available at: https://teachcomputing.org/curriculum/key-stage-3 [Accessed 21 Jun. 2021].

	Computer Systems		Networks	
	Hardware	Software	Network Architecture	Data Transmission
System / Network	0	6	1	7
Device	5	3	0	4
Component	5	2	2	0
Implementation	3	1	0	0

Table 7: Computing systems and networks content at key stage 3

In the **hardware** theme, learners identify the key internal components or architecture of computer systems and the roles the components perform. In doing so, they expand the familiar IPO model to incorporate the role of storage devices (IPOS). Learners examine how some devices, such as cameras and microphones, convert between analogue and digital data. They should understand that each hardware device is a complex collection of logic circuits and begin to construct simple logic circuits using AND, OR, and NOT gates.

Within the **software** theme, learners begin to categorise different types of software and, in particular, appreciate the role of the operating system. They should understand that all software instructions need to be translated to a simple machine language that controls the hardware. Learners also build on their understanding of internet services and cloud based software, including search engines and productivity tools.

From a **network architecture** perspective, learners largely review their understanding of the key components of modern computer networks. They consider the specific components needed for devices to successfully connect via wired and wireless technologies. Finally, learners apply their understanding to identify some vulnerabilities within a network.

A greater focus on networks at this stage occurs in the theme of **data transmission**, where learners explore in some detail the role of protocols, addressing, and data packets. They explore the idea of bandwidth as a measure of network capacity and the increased connectivity of devices all around us. There is also the opportunity to program simple wireless devices to send and receive messages and even create their own simple protocol.

Below we provide a list of key concepts and skills that learners encounter at this phase of the Teach Computing Curriculum. (**Note:** This is not an exhaustive list, but a summary of the most important ideas at stage.)

Skill/Concept	Tier	Theme	Curriculum units
Protocols as a means of standardised communication	Device	Data transmission	Y7 Networks: from semaphores to the internet
Using IP address to uniquely refer to devices on a network	Device	Data transmission	
Data is broken into packets for transmission and reassembled upon receipt	System/Network	Data transmission	
Bandwidth is a measure of the rate of data transmission	System/Network	Data transmission	
Understand the role of key components involved in serving webpages	System/Network	Data transmission	
Awareness of a range of internet services and their uses	System/Network	Data transmission	
Understand how search engines find, select, and rank results	System/Network	Software	*Y8 Developing for the web*
Distinguish between general purpose and purpose built or embedded devices	Device	Hardware	Y8 Computing systems
Computing devices all have a similar architecture	Component	Hardware	
Understand to function of key hardware components, including the CPU, memory, and storage	Component	Hardware	
Understand how the main hardware components work together	Component	Hardware	
Hardware is build from complex logic circuits	Implementation	Hardware	

Logic circuits consist of combinations of logic gates	Implementation	Hardware	
Understand the function of AND, OR, and NOT gates and combine to make simple circuits	Implementation	Hardware	
Understand the role of the operating system	Device	Software	
Understand that programs need to translated to be run by the machine	Implementation	Software	*Y8 Introduction to Python programming*
Apply the input, process, output, storage model to a physical computing device	Component	Hardware	*Y9 Physical computing programming*
Write programs that control input and output devices	Component	Software	
Write programs that can send and receive messages with other devices	Device	Data transmission	
Identify some vulnerabilities of a network and some common mean of attack	System/Network	Data transmission	Y9 Cybersecurity

Table 8: Key skills and concepts at key stage 3

Key stage 4

At key stage 4, although some learners may not choose to study a computing qualification, the national curriculum outlines in broad terms the learning and experiences that all learners are entitled to. It is up to schools how they provide this experience, but the Teach Computing Curriculum covers these experiences through five units of work[9]:

- Online safety
- IT and the world of work
- Media
- Spreadsheet modelling
- Project management

Given the requirements of the national curriculum at this stage of education largely focus on the use and application of computing, there are a limited number of objectives related to computer systems and networks.

The relevant objectives mostly appear in the '**IT and the world of work**' unit, with an additional minor mention in the '**Online safety**' content. The focus is on learners' understanding and application of various types of productivity software (typically cloud based) and their application for collaboration and communication. At the same time, learners explore the tools and technologies required for remote or mobile working.

	Computer Systems		Networks	
	Hardware	Software	Network Architecture	Data Transmission
System / Network	0	3	2	0
Device	0	0	1	0
Component	0	0	0	0
Implementation	0	0	0	0

Table 9: Computing systems and networks content at key stage 4

[9] Teach Computing. (n.d.) *Key stage 4*. [online] Available at: https://teachcomputing.org/curriculum/key-stage-4 [Accessed 21 Jun. 2021].

Skill/Concept	Tier	Theme	Curriculum units
Examples, functions, and features of online communication tools	System/Network	Software	IT and the world of work online safety
Evaluation of a range of cloud computing services	System/Network	Software	
Benefits, drawbacks, and security of ad hoc networks	System/Network	Network architecture	

Table 10: Key skills and concepts at key stage 4

GCSE computer science

Learners who choose to study computer science at GCSE explore the areas of computer systems and networks in greater depth. While the available exam specifications have some variation, they all require learners to have a detailed understanding of computer systems and networks.

Over their two years of study, learners encounter the 'Computer systems' and 'Networks' topics across several units of work[10]:

Year 10 – Computer systems
Year 10 – Programming (sequence)
Year 10 – Programming (strings and lists)
Year 11 – Networks
Year 11 – Security

	Computer Systems		Networks	
	Hardware	Software	Network Architecture	Data Transmission
System / Network	1	2	8	1
Device	2	4	0	4
Component	10	0	3	1
Implementation	14	5	1	2

Table 11: Computing systems and networks content within GCSE computer science

Within the **hardware** theme, learners predominantly focus at the component tier or lower, examining the components of a computer system and how they work together. Expanding on their prior learning at key stage 3, they explore the role of cache, the different types of secondary storage, and the internal components of the CPU. They also refresh their understanding of logic gates and circuits, using truth tables and Boolean expressions to represent them.

In parallel to their study of the CPU, within the **software** theme, learners find out about basic assembly language commands and use them to write simple programs. They explore the

[10] Teach Computing. (n.d.) *Key stage 4*. [online] Available at: https://teachcomputing.org/curriculum/key-stage-4 [Accessed 21 Jun. 2021].

relationships between high- and low-level programming languages, and how high-level programs are translated for execution. Learners also categorise different types of software and understand the role of the operating system and utility software.

Building on their prior understanding of **network architecture**, learners classify networks based on their size and scale, and the way in which nodes are organised (topology). They also define different communication models, including traditional client-server, as well as peer-to-peer networking. Learners compare the role of physical (MAC) and logical (IP) addresses on a network.

Learners explore **data transmission** within the network in greater depth, examining how data packets are routed through the network and factors that affect the network's performance. They explore a range of protocols, giving particular focus to the Internet Protocol and its associated four layer model, as well as the OSI seven layer model.

Skill/Concept	Tier	Theme	Curriculum units
Understand the roles of the CPU, main memory, cache, secondary storage, and the motherboard	Component	Hardware	Y10 Computer systems
Distinguish between RAM and ROM	Component	Hardware	
Understand how data is read and/or written using magnetic, optical, and solid-state storage	Implementation	Hardware	
Understand the roles of the internal components of a CPU	Implementation	Hardware	
Describe each stage of the fetch-decode-execute cycle	Implementation	Hardware	
Understand the factors that affect CPU performance	Implementation	Hardware	
Build logic gates to solve problems and mathematical operations	Implementation	Hardware	
Represent logic gates and circuits using truth tables and Boolean expressions	Implementation	Hardware	

Write simple programs using assembly language	Implementation	Software	
Understand the difference between high- and low-level programming languages and how translations occur between them	Implementation	Software	Y10 Programming (sequence)
Compare different types of networks, including PAN, LAN, and WAN	System/Network	Network architecture	Y11 Networks
Understand different communication models, including client-server and peer-to-peer	System/Network	Data transmission	
Understand the four layer model associated with the IP protocol	Implementation	Data transmission	
Describe the role of a range of protocols in the application layer	Device	Data transmission	
Describe the function of the UDP and TCP protocols in the transport layer	Component	Data transmission	
Understand the role that MAC addresses play in identifying network hardware	Component	Network architecture	
Measuring the performance of networks	System/Network	Data transmission	
Network vulnerabilities	System/Network	Network architecture	Y11 Security
Software to protect systems from cyberattacks	System/Network	Software	

Table 12: Key skills and concepts within GCSE computer science

A level computer science

Learners who go on to study A level computer science explore how the computer system works in great detail. While some time is dedicated to recapping prior understanding within the more abstract tiers, learners predominantly focus on the component and implementation tiers across the following topic areas[11]:

Boolean logic
Systems architecture
Memory and storage
Hardware
Software
Operating systems
Translators
High- and low-level languages
Network fundamentals
The internet
Network hardware
Communication

	Computer Systems		*Networks*	
	Hardware	Software	Network Architecture	Data Transmission
System / Network	3	9	14	8
Device	5	4	8	21
Component	10	10	19	6
Implementation	57	46	4	26

Table 13: Computing systems and networks content within A level computer science

Learners build upon prior learning of computer systems by considering a broader range of **hardware**, including control systems, embedded systems, virtual and networked storage, and more. They deepen their understanding of computation, explore the roles of registers within the CPU, as well as buses, buffers, and interrupts. They compare different CPU architectures, explore how multicore systems operate, and find out about the role of specialised hardware

[11] Isaac Computer Science. (n.d.) *All topics*. [online] Available at: https://isaaccomputerscience.org/topics [Accessed 21 Jun. 2021].

(such as GPUs). Finally, learners significantly expand their understanding of logic gates, circuits, notation, and new approaches to manipulating Boolean algebra.

As well as a broad understanding of common **software** categories, learners spend time exploring new types of software, including open and closed source software, bespoke software, control and safety software, and virtual machines. The role of the operating system is broken down to examine different types of OS, as well as the many functions it performs. They examine the different processes by which programs are translated into machine code, and the steps involved in these processes. Any past experience with assembly language is built on as learners address memory and perform bit manipulation.

Within **network architecture**, learners deepen their understanding of how networks are connected and combined, and discover new topologies, including mesh and hybrid. A broader range of network hardware is studied, including bridges and repeaters alongside alternative application of networks, such as the use of thin clients. They further categorise and explore the physical properties of different transmission media.

At this stage of education, learners cover the details of **data transmission**, using the four layer IP network as well as the seven layer OSI model. They learn about other processes carried out by network devices, including collision detection and avoidance, network address translation, and port forwarding. Finally, learners are also able to compare different transmission methods, as well as how systems verify accurate transmissions.

Table 14 provides a summary of the concepts that learners need to understand at this stage. (**Note:** For brevity, some concepts that appear at the GCSE stage have been omitted).

Skill/Concept	Tier	Theme	Curriculum units
Understand the purpose and suitability of a wide range of input and output devices to different situations	Device	Hardware	Hardware
Understand the workings and application of virtual storage devices	System/Network	Hardware	Memory and storage
Compare different types of primary and secondary storage	Component	Hardware	

Distinguish between general purpose and purpose built systems	System/Network	Hardware	Systems architecture
Compare CPU architectures (von Neumann and Harvard)	Implementation	Hardware	
Role of register, buses, and interrupts in the CPU	Implementation	Hardware	
RISC vs CISC architecture	Implementation	Hardware	
Multicore and parallel systems	Implementation	Hardware	
Logic gates, including NOT, AND, OR, NAND, NOR, and XOR	Implementation	Hardware	Boolean logic
Common logic circuits used in computation	Implementation	Hardware	
Boolean algebra and simplification	Implementation	Hardware	
De Morgan's laws	Implementation	Hardware	
Karnaugh maps	Implementation	Hardware	
Open vs closed source software	System/Network	Software	Software
Virtual machines and their limitations and benefits	Device	Software	
Distinguish between types of utility software, including encryption, compression, backup, and disk utilities	Component	Software	
Role of user interface in hiding hardware complexity from the user	Component	Software	
Distinguish between different types of software systems	System/Network	Software	Operating systems

Role of the operating systems in resource management, multitasking, and scheduling	Component	Software	
Role of the BIOS	Component	Software	
Role of device drivers	Component	Software	
Examples of high- and low-level languages and their uses	Component	Software	High- and low-level languages
Machine-code and assembly language	Implementation	Software	
Modes of addressing within assembly language	Implementation	Software	
Binary shifts and bit manipulation within assembly language	Implementation	Software	
Grouping instructions within assembly language	Implementation	Software	
Types of translators, including compilers, assemblers, and interpreters	Implementation	Software	Translators
Bytecode	Implementation	Software	
The role of linkers and loaders	Implementation	Software	
The stages of compilation and possible errors (translation and execution)	Implementation	Software	
Representing expressions and syntax using diagrams and notation	Implementation	Software	
Describe the structure of the internet	System/Network	Network architecture	The internet
Distinguish between a range of network topologies	System/Network	Network architecture	Network fundamentals

Subnetwork model	System/Network	Network architecture	
Thin- vs thick-client networking	Device	Network architecture	
Distinguish between a range of transmission media	Implementation	Network architecture	
Distinguish between a range of network hardware and components	Component	Network architecture	Network hardware
Layered network models (four layer IP and seven layer OSI)	Implementation	Data transmission	
Distinguish between a range of network protocols at each layer	Implementation	Data transmission	
Address allocation (DHCP) and name resolution (DNS)	Implementation	Data transmission	
Network collision detection (CSMA/CA)	Implementation	Data transmission	
Circuit switching vs packet switching	Implementation	Data transmission	
Static, dynamic, public, and private IP addresses	Implementation	Data transmission	Communication
Subnet masking, network address translation, and port forwarding	Implementation	Data transmission	
Performance indicators, including bit rate, baud rate, bandwidth, and latency	Implementation	Data transmission	
Transmission methods, including full and half duplex, and simplex	Implementation	Data transmission	
Transmission verification	Implementation	Data transmission	

Table 14: Key skills and concepts with A level computer science

3.3 Progression across key stages

The stage by stage progression presented in the previous section provides a detailed review of the concepts and skills taught at each stage and where they fit within our theme and tier model. While this provides a useful overview of the key concepts taught at each key stage, it also helps provide a view of what learners have studied and will go on to study. Taking this one step further, it is possible to distill these concepts and skills into a collection of ‘big ideas’. These big ideas have been presented as a high level learning graph (Figure 3), organised first by their tier and theme, and then by the key stage in which they **first** appear.

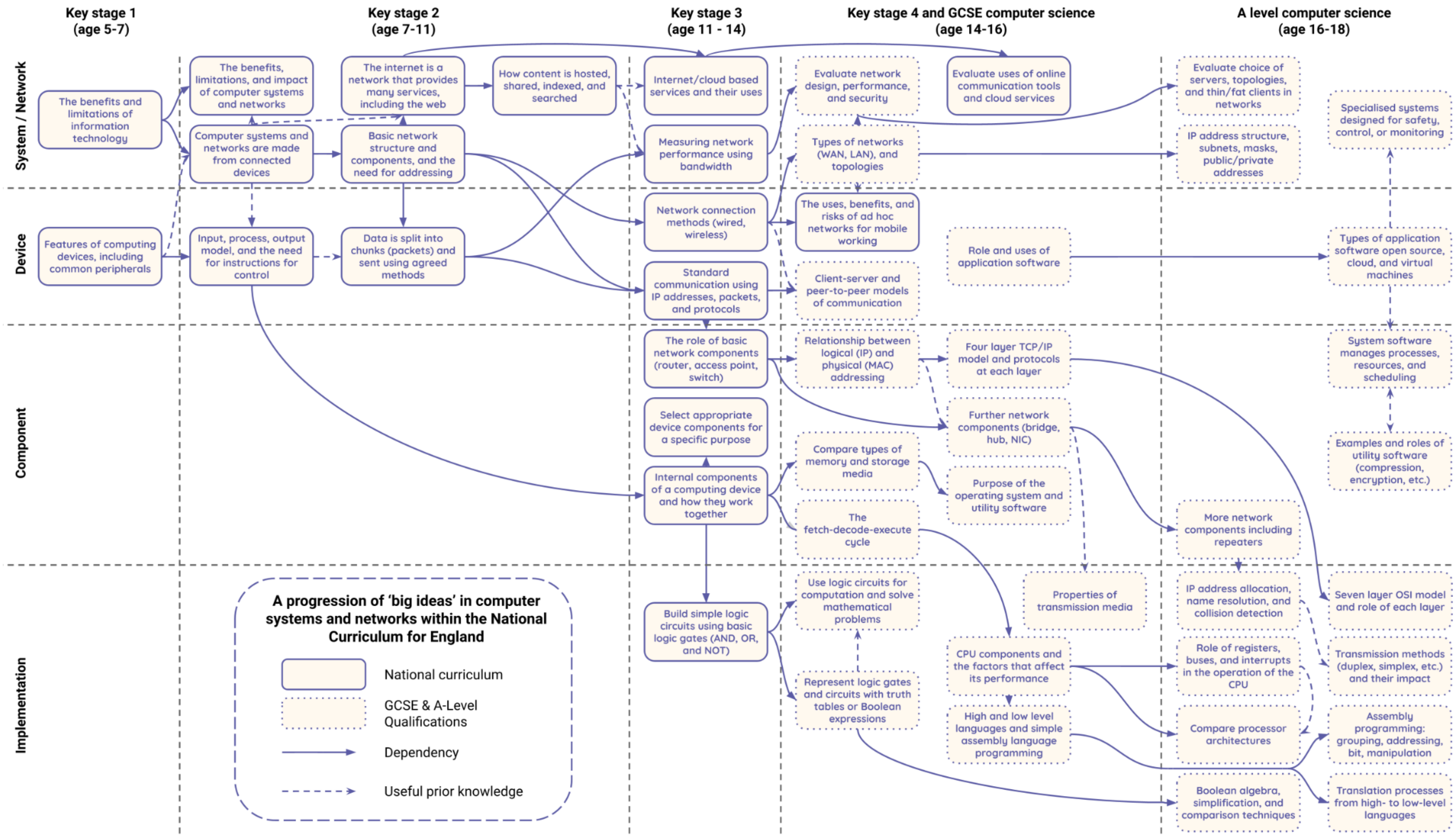

Figure 3 : A progression of 'big ideas' in computer systems and networks within the national curriculum for England

4. Pedagogical strategies for computer systems and networks

Pedagogy principles

The work of the National Centre for Computing Education is underpinned by 12 pedagogical principles that can be exemplified by a range of evidence, informed practices, and strategies. These principles apply across the teaching of computing; however, some are more applicable than others in each strand of the curriculum.

Figure 4 : NCCE 12 pedagogy principles

The area of computer systems and networking is predominantly focused on learners' understanding, retention, and ability to apply fairly abstract concepts. There are also a few skills and processes that learners need to acquire. With this in mind, teachers should focus on the following principles and related practices to tackle the 'Computer systems' and 'Networks' topics.

Lead with concepts

Support learners in the acquisition of knowledge through the use of key concepts, terms, and vocabulary, and provide opportunities to build a shared and consistent understanding.

- Carefully consider the key concepts your learners need to understand, the best order for them to experience them, as well as any dependencies or prior knowledge requirements.
- Use appropriate and consistent vocabulary to 'label' and describe each new concept. You could refer to existing glossaries or construct your own with learners to help them build confidence in their vocabulary.
- As learners develop their conceptual understanding and the accompanying vocabulary, they may need help to connect these concepts together. Constructing or referring to a concept map[12] can model the relationships between concepts and provide a blueprint for learners' internal schema.
- Help your learners to develop a shared understanding by using displays, discussions, and diagnostic questions to reinforce the key terminology and the concepts that it represents.

Unplug, unpack, repack

To teach new concepts, first unpack complex terms and ideas, then explore these ideas in unplugged and familiar contexts, before you repack this new understanding into the original concept.

As already acknowledged, there are many concepts within computer systems and networking that are abstract and highly complex. In many cases, taking a constructivist approach to learning these concepts is helpful and involves using a learner's existing knowledge from a more familiar context to explain (unpack) a new concept. This could be achieved in a number of ways, including analogy, storytelling, or an unplugged activity. Many educators are already using such approaches, but be careful to avoid potential learner misconceptions.

[12] National Centre for Computing Education. (2020). *Quick Read: Using concept maps to capture, communicate, construct, and assess knowledge*. [online] Available at: https://blog.teachcomputing.org/using-concept-maps-to-capture-communicate-construct-and-assess-knowledge/ [Accessed 21 Jun. 2021].

To help avoid these issues, teachers can apply a semantic wave[13] approach. In simple terms, this encourages educators to:

- Present learners with an abstract concept: "The internet is the physical network on which many services run, of which the World Wide Web is only one."
- Unpack the meanings within the concept and relate it to a familiar concept: "The internet is similar to a road network, it connects towns and cities together and each type of road user represents a different service."
- Explore the concept within this familiar analogous context: "How can traffic get from A to B? Which routes are fastest? Do wider roads have more capacity for traffic?"
- Repack the meanings of the original concept and draw similarities and differences between the analogy and the original computing context: "The roads are like individual connections in our network and the junctions like nodes. Network bandwidth is similar to the width of our roads, allowing more or less traffic through. However, because different road users all travel at different speeds, they can overtake each other and pull over at the side of the road, which isn't the same for network services."
- Finally, return the original concept in its own context: "There are many services, including the World Wide Web, that all use a global network called the internet."

Challenge misconceptions

Regardless of how well a concept is taught, there is always space for misconceptions to develop. In fact, sometimes we may knowingly introduce a misconception in order to simplify a concept or make it accessible. Recognising those misconceptions and knowing how to mitigate them is important, especially in an area of the curriculum that focuses on concepts.

- Teachers should make a conscious effort to seek out misconceptions and challenge them. Using regular formative assessment can help uncover misconceptions.
- Carefully written multiple choice questions can be used diagnostically[14] with distractors (wrong answers) that each result from a specific misconception.
- Concept mapping is again another useful tool. If learners create their own maps, these should be a reflection of their internal understanding and can help identify the root of a misconception.
- Peer Instruction[15] is a particular effective technique based on a flipped learning approach. Learners complete a task before the lesson, in which they 'learn' new concepts. The lesson time is then used to answer diagnostic questions collaboratively and relies on peer discussion to build consensus around a concept. It not only helps identify misconceptions, but also helps address and correct them.

---

[13] National Centre for Computing Education. (2020). *Quick Read: Using semantic waves to improve explanations and learning activities in computing*. [online] Available at: https://blog.teachcomputing.org/quick-read-6-semantic-waves/ [Accessed 21 Jun. 2021].
[14] Eedi (n.d.). *Teach Computing NCCE*. [online] Available at: https://eedi.com/projects/teach-computing [Accessed 21 Jun. 2021].
[15] National Centre for Computing Education. (2019) *Quick Read: Using peer instruction to discuss computing concepts*. [online] Available at: https://blog.teachcomputing.org/quick-read-4-peer-instruction/ [Accessed 21 Jun. 2021].

Model everything

As well as being a concept rich part of the computing curriculum, computer systems and networking has plenty of processes and practices that can be modelled.

- Use worked examples[16] to model skills that learners need to master, everything from simplifying Boolean expressions to the process of ranking search results.
- Use the same live coding approaches to model the process of programming in assembly as you would with higher-level languages.
- Remember, modelling is particularly beneficial to novices because it provides scaffolding that can be gradually taken away.

Make concrete

Bring abstract concepts to life with real world, contextual examples and a focus on interdependencies with other curriculum subjects. Our learners are surrounded by computer systems and networks. Try to use the resources around them to make this area of computing less abstract and more concrete.

- Talk about the systems and networks around them, particularly at home and school, but also present in their everyday lives.
- Help learners to connect this area of computing with everyday life and careers: go on a technology walk, invite a system admin to share their experience, or find your nearest computing museum.
- Use unused (or even 'historical') IT equipment as a resource to examine, touch, and even dismantle to help learners to connect the concept with the physical objects.
- Where possible, provide opportunities to experience a range of software, including different operating systems as well as open source software.
- Use old equipment or low cost devices like a Raspberry Pi to build and configure your own network with learners. You could demonstrate protocols, send messages, or build simple programs that communicate across it.

[16] National Centre for Computing Education. (2019) *Quick Read: Using worked examples to support novice learners*. [online] Available at: https://blog.teachcomputing.org/using-worked-examples-to-support-novice-learners/ [Accessed 21 Jun. 2021].

5. Professional development for computing teachers

A core part of the NCCE's role is to help teachers develop their subject knowledge through continued professional development (CPD).There are a number of routes for teachers to participate in CPD to support their understanding of computer systems and networking.

Table 15 shows the courses[17] that are available within the NCCE and are designed to support teachers' development of computer systems and networking subject knowledge. Teachers of A level computer science can find an additional range of bespoke courses[18] organised by Isaac Computer Science on a range of topics, including 'Computer systems' and 'Computer networks'.

Beyond accessing formal courses, there are many opportunities for computing teachers to learn through networks, such as Computing at Schools (CAS). These local communities continue to meet regularly and share best practices and skills and are therefore a great source of inspiration and development for teachers. As well as local support and meetups, teachers can find many CPD focused events[19].

[17] Teach Computing. (n.d.) *Computing courses for teachers*. [online] Available at: https://teachcomputing.org/courses [Accessed 21 Jun. 2021].
[18] Isaac Computer Science. (n.d.) *Events*. [online] Available at: https://isaaccomputerscience.org/events [Accessed 21 Jun. 2021].
[19] Computing at School. (n.d.) *Upcoming events*. [online] Available at: https://community.computingatschool.org.uk/events [Accessed 21 Jun. 2021].

Course	Format	Key stage	Description	Themes
Introduction to primary computing	Face to face or remote	Key stage 1 Key stage 2	This course helps teachers to understand the nature of computing in the curriculum — the breadth and depth of the subject. It provides practical experience of teaching different aspects of the computing curriculum.	• Hardware • Software • Network architecture • Data transmission
Teaching and leading key stage 1 computing	Face to face or remote	Key stage 1	This course aims to give you confidence in teaching the whole of the key stage 1 computing curriculum. It unpicks the curriculum, providing a clear vision of the expectations for this key stage.	• Hardware • Software
Teaching and leading key stage 2 computing	Face to face or remote	Key stage 2	This course addresses the key stage 2 computing curriculum. It explores how to support learners in becoming skilled and critical users of technology, and how to choose tools to help them achieve their goals while developing safe and acceptable online behaviours.	• Hardware • Software • Network architecture • Data transmission
Physical computing kit – KS2 Crumble	Face to face or remote	Key stage 2	This course is aimed at classroom teachers and subject leaders of computing who are looking to develop physical computing approaches in the KS2 classroom. It provides practical experiential learning, led by an experienced facilitator. The course will model teaching approaches that can be taken back to the classroom.	• Hardware • Software
An introduction to computer	Face to face or remote	Key stage 3 Key stage 4	During this course, teachers establish a foundational knowledge of the concepts, terminology, and classroom	• Hardware • Software

systems, networking, and security in GCSE computer science			practice related to the components of computer systems and how these can then be connected together to form a network.	• Network architecture • Data transmission
Computer processors and instruction sets	Face to face or remote	Key stage 3 Key stage 4	In this course, teachers explore the core of a modern computer. They learn how the instructions that humans write in computer programs are translated into machine code that the computer can process.	• Hardware • Software
Computer systems: input, output, and storage	Face to face or remote	Key stage 3 Key stage 4	During this course teachers take their first steps to opening up the computer; teachers explore the main components, their roles, and how they work together in computer systems.	• Hardware
KS3 computing for the non-specialist teacher	Face to face or remote	Key stage 3 Key stage 4	This course is designed to develop subject knowledge around key topics such as algorithms, data representation, hardware, and programming. It also explores useful and engaging strategies for delivering this content in the classroom.	• Hardware
Maths in computer science	Face to face or remote	Key stage 3 Key stage 4	Teachers explore how mathematics is firmly embedded at the core of computer science. They explore the foundational topics at the heart of mathematics and computer science.	• Hardware

Physical computing kit – KS3 micro:bit	Face to face or remote	Key stage 3	Teachers explore physical computing using the micro:bit device and how the Teach Computing Curriculum can be used to engage learners.	• Hardware • Software
Understanding computer systems	Online	Key stage 3 Key stage 4	Understanding how a computer works is crucial to working with technology effectively. This course explores what happens inside the machine and how computers turn inputs into outputs.	• Hardware • Software
Introduction to cybersecurity for teachers	Online	Key stage 3 Key stage 4	Teaching cybersecurity in schools is an important way to help learners stay safe while using technology. In this course, teachers are introduced to the core ideas of cybersecurity that should be taught in the classroom.	• Network architecture • Data transmission
An introduction to computer networking for teachers	Online	Key stage 3 Key stage 4	In this course, teachers learn to describe the different types of computer networks and understand how data can be transmitted securely. They cover real world usage of networking technology and become more familiar with how the internet works, including routing, DNS, and the World Wide Web.	• Network architecture • Data transmission
Design and prototype embedded computer systems	Online	Key stage 3 Key stage 4	This course explores how embedded systems are used in the world around us. It covers what makes an embedded system different from a general purpose system, and how embedded systems are specialised for a particular use.	• Hardware • Software

How computers work: demystifying computation	Online	Key stage 3 Key stage 4	In this course, teachers gain an understanding of how computers work at a fundamental level. They explore system architecture, along with how computers use binary and logic, and the fetch-decode-execute cycle. They also learn to build a range of simple circuits for maths, and then simulate various logic gates.	• Hardware • Software
Teach computing in schools: creating a curriculum for ages 11 to 16	Online	Key stage 3 Key stage 4	In this course, teachers can discover how to create a computing curriculum for learners aged 11 to 16, using freely available online resources to complement their existing materials.	• Hardware • Software • Network architecture • Data transmission
Teaching physical computing with Raspberry Pi and Python	Online	Key stage 3 Key stage 4	During this course, teachers create simple systems that respond to and control the physical world using a Raspberry Pi and Python. They gain knowledge of simple electronics and computing, as well as some Python programming experience.	• Hardware • Software
Understanding maths and logic in computer science	Online	Key stage 3 Key stage 4	In this course, teachers are introduced to maths and logic in computing in an engaging way. Using the concept of an escape room, they learn activities and games to help improve their knowledge and skills in this subject.	• Hardware

Fundamentals of computer networks	Face to face or remote	Key stage 3 Key stage 4	During the course, teachers demystify the hardware and network topologies used for data transfer between computers. They learn how computers of all shapes and sizes are connected physically, or wirelessly, and how this arrangement affects how data is shared.	• Network architecture • Data transmission
The internet and cybersecurity	Face to face or remote	Key stage 3 Key stage 4	Participants explore how the internet works and how its expansion has led to the rapidly growing cybersecurity industry. The course covers the inner working of local networks to global systems, and develops knowledge of computer security issues and practices.	• Network architecture • Data transmission
Physical computing kit – KS4 Raspberry Pi Pico	Face to face or remote	Key stage 4	During this course, teachers learn how to use physical computing to engage learners in programming projects and subject knowledge development.	• Hardware

Table 15: Courses to support teachers’ development of computer systems and networks

6. Conclusion

In this report, we've illustrated the importance of computer systems and networks for learners, and the role these topics play in developing learners' foundational knowledge. A firm understanding in the early stages of education can support future knowledge acquisition across the breadth of computing, and potentially prevent misconceptions.

We have reviewed and summarised this area of the curriculum and identified the key themes and the different perspectives (tiers) from which learners might view them. We have presented these themes and tiers as a model that can be used to categorise concepts and skills, which we've exemplified in Table 1. We hope this model and mapping provides teachers with a useful tool to reflect the concepts they teach and the depth (tier) to which they teach them. Within this model, we have chosen to focus on concepts that specifically relate to the structure or operation of computer systems and networks. The use, impact, and questions of safety and security have been considered to be beyond the scope of this report.

Using our themes and tiers model, we have been able to analyse the content of each key stage of computing, and track the progression and shift in focus over time. Overall, we have observed a general trend from the higher tiers of the model to the lower tiers as learners progress. We have used this analysis to synthesise a learning graph of 'big ideas' across these two strands of the curriculum, which spans from key stage 1 to key stage 5.

Being a highly concept rich area of the computing curriculum, we have been able to emphasise the pedagogical principles that best suit these topics. We have also provided practical examples of activities and approaches that demonstrate these principles and can be applied to the teaching of computer systems and networks.

Teachers looking to develop their own subject knowledge, as well as their practice in computer systems and networks, can take advantage of the course recommendations presented in Section 5 of this report.

We welcome feedback on this report, and we plan to publish more reports on other topic areas within the computing curriculum.