6

WHAT ARE INTERFACES FOR, REALLY?

Gabriel Egan

In almost all modern computers, everything is represented by zeroes and ones that exist as physical states of matter. This can be as electricity present or absent in circuits, as magnetized poles on the surfaces of disks, and as electrical charges pumped into or drained from capacitors. The meaning of these zeroes and ones is given by their context rather than being inherent. The same binary sequence 01000011 might in one context represent the age of a person in years (being equivalent to 67 in decimal counting), yet in another context represent the upper-case letter "C" (as it does in the ASCII encoding scheme), and in another context again be an instruction to the processor to move data from one memory cell to another (as it is in the Intel 8080 processor instruction set). In computing, context is everything.

To show what this means for how we conceive of and use human-computer interfaces, this chapter will sketch the technical innovation that gave rise to the context-dependency of digital representations and chart how, over time, accreted layers of meaning-making contextualization have obscured it. At the most fundamental level, the zeroes and ones in a computer are merely representations (when created) and interpretations (when read), since physical states of themselves bear no meaning and matter does not, in any case, exist in the sharply distinguished states that match the sharply distinguished mathematical concepts of zero and one. In computer design, any electrical pressure above an arbitrary voltage is treated as a one and any pressure below another arbitrary voltage is treated as a zero. The permitted limits of variation—known as tolerances in engineering—map the realities of the physical components to the abstraction of the binary system.

The software programs that we call interfaces are the same kinds of abstractions as the content they appear to contain and mediate to us: they are '"made" (in the abstract sense) of exactly the same kinds of zeroes and ones. The interface/content distinction is, to that extent, illusory. Tracing the history of this illusion will take us to a consideration of four digital datasets widely used by Shakespearians showing the best and worst of what is possible with this technology. What we find is while at the lowest level everything we are discussing exists merely as streams of zeroes and ones, the illusory interface/content distinction is nonetheless strictly policed, for reasons that must be understood in terms of the power relations that modern technologies are made to serve.

DOI: 10.4324/9780367821722-9

Context as meaning

The context-sensitivity of digital representations did not feature in the earliest digital computers. The Harvard Mark I computer completed in 1944 was the culmination of earlier work on mechanical calculating machines and it embodied in physical form (as a strip of paper tape punched with holes) a series of commands about the automated movement between memory cells called registers of the numbers representing the intermediate results of a series of calculations, and it also embodied in physical form (as paper cards punched with holes) the data upon which the calculations would start (Priestley 102–7). The two physical embodiments—of instructions, on one hand, and data, on the other—were kept separate and used different media: paper tape and punched card. What effect does this separation have in a computing machine? Crucially it made it impossible for the result of a calculation to form the basis for a new instruction, as happens when instructions are treated as if they are numbers.

In 1945, John von Neumann, responding in part to his experiences using the Harvard Mark I computer, came up with a revolutionary innovation in the design of computing machines (see Von Neumann). Virtually every digital computer since then has been a "Von Neumann" machine and nothing essential has changed: they have simply become faster each year. Von Neumann's innovation was to put the data being worked upon and the instructions for what to do with the data into the same storage medium, the same memory space, rather than keeping them separate. In this view, instructions are just data—specifically, data about what the machine should do next—and, conversely, data can be expressed as instructions, so, for example, instead of storing the number pi as a constant, it can be algorithmically calculated afresh each time it is needed.

The fact that the meaning of binary strings in computers is thoroughly contextual should alert us that the content/form distinction made in the Text Encoding Initiative (TEI) application of eXtensible Markup Language (XML), the distinction between a text and the DTD or schema that describes its parts and their inter-relations, is only the outermost of a series of nested contextualizations. A string of binary digits stands for a particular letter of the alphabet only in the context of an encoding scheme, such as the ASCII encoding scheme (now part of the UTF-8 scheme) first approved by the American Standards Association in 1965 (MacKenzie 211–97), in which the binary number 01000001 is designated as "A," 01000010 is designated as "B," 01000011 is "C," and so on. The ASCII context can be understood as providing metadata: data about the data, or a comment made about it. The contextualization may end at this single level of metadata. The world's oldest collection of free online digital texts, Project Gutenberg begun in 1971, standardized on ASCII encoding on the principle that this would give its texts the greatest possible longevity and widest reusability, since all computers understand this first level of contextualization and will correctly display as letters and punctuation a text encoded this way. ASCII encoding relies upon a context that every computer manufacturer has agreed to implement so that the processing of the string 01000001 is bound to result in the letter "A" being displayed, printed, or transmitted.

Project Gutenberg has been criticized for the decision to use this minimal encoding, most influentially by the co-creator of XML and TEI, C. M. Sperberg-McQueen, who objected that without additional metadata providing additional layers of context—textual apparatuses and descriptions of sources and principles of transcription—such impoverished texts cannot be the basis for serious scholarly work (see Sperberg-McQueen). The XML/TEI approach applies such an additional layer of contextualization, taking the writing beyond the mere imperative to interpret 01000001 as an "A" that arises from the ASCII context. XML's additional metadata assert that one part of the string of binary digits be understood as a verse line, another as a prose paragraph, and so on. This metadata can be embedded in the same file as the character data (as "tags" around the content words) but need not be: the XML metadata may reside in a separate file and merely point to the parts of the character data that it comments upon, in the technique called standoff markup.

As merely layers of additional commentary upon the text, all such metadata are in a sense various kinds of literary criticism. Even the choices to normalize to a single space the variety of spaces found in manuscript and printed sources, or to preserve the line-breaks in verse and to reflow paragraphs of prose, are kinds of micro-criticism. Digital texts allow us to see more easily than we can with printed texts that without the lowest level of metadata—in computers, the ASCII context—the content we call writing is inherently meaningless. In printed texts the letters and words are mere ink marks that acquire meaning only when we apply a context by choosing to understand them as standing for letters within a particular alphabet, and then apply a further layer of context to understand the resulting collection of letters as words within a particular language. In digital texts there are rather more layers of contextualization separating the underlying physical substrate—the active and inactive transistor circuits representing zeroes and ones—and the final image that human eyes and brains make sense of as writing.

It is common to refer to the last stage of this process as the human-computer interface but we should recognize that everything we see on a computer screen—the pixels we switched on by typing our text and the pixels switched on by the text-editing software to create frames that surround and contain our text—is all, at origin, merely the expression of strings of zeroes and ones. A text encoded in the internal format used by the Microsoft Word software is, at the lowest level of matter, indistinguishable from the software that created it: each is just a long binary number. Bearing this in mind disables the false interface/content distinction that clouds our habits of thinking about digital text. This interface/content distinction is especially pernicious when exploited in the power relations between creators and consumers of digital texts and other creative works, as I hope to show.

Habituation to particular practices of, and tools for, reading and writing—that is, habituation to our interfaces—makes those practices feel natural and effortless and makes the tools seem to disappear. When our interfaces to reading and writing are changed, our interactions with them feel at first unnatural and onerous. As the philosopher Martin Heidegger observed, our tools tend not to appear to us as they really are so long as we are usefully employing them, or as Terry Eagleton summarized him "when the hammer breaks, when we cease to take it for granted, its familiarity is stripped from it and it yields up to us its authentic being" so that a "broken hammer is more of a hammer than an unbroken one" (Heidegger 68–81, 149–52, 342–4; Eagleton 64). A consideration of the history of our interfaces with computers will help illustrate the technology's progress toward effortless utility and invisibility.

A brief history of human-computer interfaces

The earliest Von Neumann computers scarcely had a human interface to speak of. For those directly operating the computer the interactions were mediated through paper tape and punched cards, modified teleprinters, and large banks of switches and lamps by which individual binary numbers could be entered into the machine or read from it. In some cases the state of the hardware of an early Von Neumann computer could even be directly read by human operators without the need for intermediary lamps and switches (Lavington 13, 17–19, 65–6; Bashe 104–8).

The dominance of the punch card as the primary human interface in the early history of literary and linguistic computer projects can largely be explained by IBM's popularization of this medium in business data processing before the invention of the digital computer, and in that light the punched card might seem something of an obstacle to progress. But to individual users in Humanities departments, the punched card had distinct advantages over newer media that called for a closer relationship with the computer and those who looked after it.

Until the microcomputer revolution of the late 1970s, the long-term storage of a digital text on a magnetic medium such as disk or tape required ongoing interaction with a computing centre, which typically for an academic user was the computing service of her university. Such magnetic storage was relatively costly. By contrast, a digital text stored on a paper medium (tape or cards) could be kept in the user's office and its maintenance required only the minimal protection from water and extremes of temperature and humidity that books require.

Although it is technically possible to punch a new paper tape by hand without access to a computer, the task is time-consuming and intricate, and the resulting tape cannot easily be altered once it is punched. But because each card contains just one line of characters, a deck of punched cards holding a digital text could, by contrast, be easily extended by interlarding additional cards and could be edited by replacing existing cards with new ones. These tasks could be performed by the user without recourse to a computer, using desk-size electromechanical punching machines. A batch of punched cards offered a digital surrogate for the literary-historical text that gave the Humanities investigator an autonomy over the creation and editing of a text that came close to the autonomy that handwritten and typewritten files provided.

Working on the collation for the New Variorum Shakespeare edition of *A Midsummer Night's Dream*, R. L. Widmann reported such autonomy as a prime consideration:

> We use an IBM 029 key-punch. I have rented one of these, at $62 a month, for use by my three part-time student assistants, who are paid $2 an hour ... Punched cards were chosen as in input medium since I anticipated difficulties in correcting paper tape or magnetic tape without the help of professional staff.
>
> *(59n1)*

This model of local curation of texts still holds in some projects of computational stylistics in the sense that much of the investigator's time is spent on the relatively mundane tasks of creating and refining the large textual corpus on which the work is founded and relatively little time is spent actually processing it. Such curation of a text is non-computational in the sense of "computation" used by the early, mathematically oriented, pioneers of the new technologies. But these days this work of curation is itself largely undertaken using computers to run the various software aids provided by tools such as XML editors. Notice that Widmann's concern was to manage his text without needing the help of professional staff, and that the medium of punch cards empowered him in his relationship with the institutional providers of computing services. The historical account of technological development presented here is intended to show that particular technologies and ways of working shift the balance of power toward the user and, as we shall see, that others shift it away.

Histories of the development of academic computing services usually contrast the inconvenience of batch-mode data processing involving forms and punched cards with the interactive, conversational, interfaces that succeeded them. For example, Joy Lisi Rankin began her account of the liberating spread of teleprocessing—the ability to control a computer from afar over telephone lines—by contrasting the six-hour round trip that a mathematics professor at Dartmouth College had to take in 1958 to run a program that he hand-delivered to the Harvard University computer centre on punched cards with the ease of remote, conversational access available to a Dartmouth undergraduate student ten years later (12–13).

In a conversational interaction, the user operates what is known as a terminal: originally an adapted teleprinter, later a visual display unit comprising a keyboard and a video screen. The computer presents on the terminal's printer or screen a prompt, meaning an invitation to enter an instruction, and in reply the user types a short sentence that commands the computer to perform a particular action, for example, an imperative verb such as "RUN" followed by an accusative noun specifying the object to be acted upon such as "PROGRAM1." The results of executing a program might be the creation of new data within the computer and/or the production of a report or data presented on the printer or screen. After completing the commanded task, the computer presents a fresh prompt to invite the user to enter a fresh command. This interaction is sometimes also called command-line processing.

Such conversational interaction was always a possibility for the operators present in the same rooms as the earliest computers, but it later became viable as a mode of interaction for the wider user community only after the development of timesharing computer systems powerful enough to serve the needs of multiple simultaneous users. Timesharing computers give each user the illusion of dedicated access to the computer by the method of time-slicing: attending to each user for only a fraction of a second in a rotation rather as a busy waiter serves multiple tables in a restaurant. Timesharing computers could be accessed via terminals which might be in the same building as the computer or else connected remotely over conventional telephone lines. The first timesharing computers were large and expensive mainframe machines, but by the late 1960s cheaper minicomputers—most notably Digital Equipment Corporation's PDP-10 (see Bell et al.)—also supported such multiuser operation, albeit with fewer simultaneous users.

The conversational model dominated the teleprocessing revolution described in Rankin's history and formed the earliest computing experiences of the pioneers of the microprocessor computing revolution of the late 1970s, including Bill Gates, who later co-founded the Microsoft corporation (Manes and Andrews 23–36). Bringing the computer even closer to the user—putting it physically on her desk—the new microprocessor-based personal computers adopted the conversational model: the software used to operate the earliest personal computers, their primitive operating systems, expected the user to type commands onto the command line to execute programs and create and move files of data.

Most influential of the new microcomputers was the IBM Personal Computer (PC) introduced in 1981. Its operating system—called PC-DOS and effectively equivalent to the MS-DOS sold separately by its creators Microsoft—used a command-line interface. Programs running on a PC could themselves provide a Graphical User Interface for further interaction with the user, but the native operating systems of all microcomputers used a command-line conversational model until Apple introduced the Macintosh computer in 1984. The Macintosh's operating system was designed to be wholly graphical and used the metaphor of a desktop instead of a conversation and required users to manipulate iconic representations of objects (documents, storage devices, a wastebin, and so on) using a pointing device (the mouse) instead of typing their names at a command line (see Hertzfeld). The new visual metaphor rapidly replaced the conversational metaphor when Microsoft copied Macintosh's design to create its Windows operating system for the PC (Manes and Andrews 214–28).

The desktop metaphor transformed the user's visual tracking of what happens on a computer screen. With a command-line interface the user focusses intently on the single instruction sentence as she carefully types it, since even small errors of spelling or punctuation cause a command to be rejected or, worse, initiate an unintended operation. In this mode of close reading and writing, the screen space above the line currently being typed contains the sentences of previous commands and can be ignored other than as a reminder of what has just happened and a guide to what to type next.

The desktop-metaphor interface, by contrast, requires a visual scanning of the entire screen since the interaction could in theory transform any part of it. Close reading was replaced by visual scanning. For those charged with providing help to computer users this meant that the instructions to make something happen could no longer be conveyed as a series of commands to be carefully typed and instead required accurately describing in words a moving picture and a set of gestures the user should make with the mouse.

With the rise of Graphical User Interfaces, users came to expect that interactions with computers would involve visual scanning of a screen comprising relatively fixed furniture such as menus and persistent borders within which changeable contents appear. This kind of interaction intensifies a false content/form distinction as the fixed parts of the interface seem to give shape to the malleable data—words, numbers, pictures—that appear within them. For instance, the bounding boxes topped with menus and so-called ribbons that are the screen furniture provided by Microsoft Word appear to give shape to the user-chosen words typed within them, and to provide the means to reshape those words by changing the typeface, margins, and so on.

In truth, the apparently fixed screen furniture of an interface is as much the consequence of patterns of zeroes and ones inside the machine as are the apparently more malleable data held within the furniture. This becomes clear when the user inadvertently selects to alter or hide a part of the furniture. The anxiety displayed by computer users when their interfaces change, either by their own accidental instructions or because the manufacturer changes the design, is witness that the illusory permanence of the screen furniture has become deeply embedded in the cognitive expectations of computer users. If the change was initiated by the user, the knowledge of how to restore the lost furniture goes some way toward assuaging the anxiety, although it is not uncommon to hear users complain that it ought to be impossible to so easily harm such fundamental components of their interaction with a computer. If the change was initiated by a manufacturer and the users are powerless to reverse it, the anxiety amongst users may become intense and be collectively expressed.

The fundamental relations giving rise to anxiety about human-computer interfaces are ones of power and knowledge. At the lowest level inside the computer all representations are equal and the zeroes and ones are moved around by mindless processes that invoke no implicit meanings. Whatever meanings attach to the binary digits—assigning this stream the status of an instruction and that stream the status of data—arise solely from context. Such meaning-bearing contexts are nested one within another, so that the instructions that comprise a complex software package will designate some streams of binary digits as representations of the screen furniture and others as representations of the contents to be displayed within that furniture.

The worst kind of computer software reifies such high-level distinctions in the meaning of binary digits so that it feels as if some parts of the interaction really are fixed, the parts forming the interface, and that only the remaining parts are within the user's control. The choice to make software behave in this way reflects the power relations that inhere in the use of computers and resistance to it requires the assertion of power by the users of computers and better knowledge of how their machines actually operate. To illustrate this point, and show that bad choices are not exclusive to either the commercial or the publicly funded realm, we will now consider four illustrative examples of the publication of digital resources of special interest to Shakespeare scholars.

Four cases: The Oxford Complete Works of Shakespeare Electronic Edition (1989), Literature Online (LION, 1997–), the Henslowe-Alleyn Digitization Project (2005–), and the Database of Early English Playbooks (DEEP, 2007–)

Two of our illustrative examples are commercial products, the Oxford Complete Works of Shakespeare Electronic Edition and Literature Online, and require a purchase or subscription, and the other two are publicly funded and are free at the point of use. Our first example, the Oxford Complete Works of Shakespeare Electronic Edition (1989), predates the widespread adoption of Graphical User Interfaces in personal computing, coming as it does from the MS-DOS era of 1981 to about 1995.

The dominant transportable storage medium in the MS-DOS era was the floppy disk, available in two physical sizes—5¼ and 3½ inches wide—using a number of partially compatible file formats. This variety of standards gave each disk a capacity from 160 kilobytes (in the first IBM PC) to 1,440 kilobytes in the last format commonly in use before floppy disks became obsolete in the late 1990s. Floppy disks were used to distribute computer software but could also contain significant quantities of text, and Michael Best has documented the commercial projects to sell Shakespeare's works in this format. A practical constraint was the inherently small overall data capacity of the floppy disk: even in the most capacious format it could hold only four or five plays so that a large authorial canon might require a set of disks.

As detailed by Best, almost all the editions of Shakespeare made available on floppy disk, and later on CD-ROM, were based on out-of-copyright Victorian editions, most commonly the Globe Shakespeare (see the Shakespeare 1864 edition). An important exception was the Oxford Complete Works of Shakespeare Electronic Edition, based on the printed edition of 1986–87, which appeared in 1989 as a set of floppy disks for the IBM PC and compatible computers (see the Shakespeare 1989 edition). There were 20 disks in the 5¼-inch set and 10 in the 3½-inch set and each work, such as a play, occupied one text file on a disk, encoded in the ubiquitous ASCII file format that made them usable by every program for text display, processing, and editing.

The manual accompanying the Oxford Complete Works of Shakespeare Electronic Edition explained the markup conventions used within the files, which employed the system of COCOA tags that was first developed specifically for use with the COCOA concordance software and later the Oxford Concordance Program (see Russell; Hockey and Martin). The manual illustrated how this COCOA tagging provided information not available in the printed edition. For example, the printed works' type-layout convention for distinguishing prose from verse put the first line of speech on the same line as the speech prefix, if it was prose, and on the line below the speech prefix, if it was verse. While marking the start of a speech unambiguously, this convention cannot show transitions from verse to prose or vice versa occurring within a speech, which can—depending on vagaries of sentence and line length in relation to the width of the printed book's measure—be impossible to detect by sight. The electronic edition eliminated this ambiguity by providing explicit markup tags for all transitions from verse to prose and vice versa.

Likewise, there is no indication in the printed edition (beyond the lines being perhaps somewhat short) of occurrences of a run of three verse lines being amphibious in the sense that either the first and second or the second and third could together form a complete metrical unit. In the digital edition this was also explicitly marked up. Most usefully of all, since the 1986–87 Oxford Complete Works of Shakespeare was ground-breaking in its theorizing and practice regarding Shakespeare's collaborations with other writers, the digital text explicitly marked changes of author within the body of a work. This pioneering digital edition gave users what must have been for most their first sight of textual markup used to convey literary-critical assertions, in this case about changes of author and versification. The COCOA's system's relatively transparent and unobtrusive nature—Shakespeare's text is readily readable between the tags—and the edition's encoding in the universally understood ASCII format enabled anyone to make use of the extra information. This digital edition is the high watermark of openness in the commercial publication in electronic form of Shakespeare's works edited to the highest modern standards, and in that aspect at least, it has not yet been surpassed.

A standard floppy-disk drive is capable of writing a disk as well as reading it, so that turning a blank disk into a copy of one purchased from a publisher was cheap and easy for users to do. Indeed, the single command needed to do this was built into all personal computers' operating systems. There was nothing inherently suspicious about taking such a copy, and the Oxford Complete Works of Shakespeare Electronic Edition advised doing so for backup purposes, and its manual explained how (see Shakespeare 1989, ["Manual"] 1). From the publishers' point of view, the new CD-ROM physical format that became standard on PCs in the 1990s had one special advantage over the floppy disk as a distribution medium: it was read-only. Until the early 2000s, the CD-ROM drives in most computers could not write to disks, so copying a publisher's disks was beyond the ability of most users. A second advantage for publishers was that because each CD-ROM could hold as much data as about 400 floppy disks there was room not only for a copious text—plus images, and sound, and short video streams—but also for software.

Including software with the texts on a CD-ROM enabled publishers to disguise or even encrypt the raw texts so that instead of viewing them from the supplied disk with an interface of her own choosing, the user could reach them only via a publisher-supplied software application that had to be installed on her computer. With such a CD-ROM the user was paying not just for the raw data, the texts of Shakespeare, but also the means to inspect—to read or process—that data. Indeed, with most such CD-ROMs there was no other way to get at the Shakespeare works. They could not be simply extracted from the disk because, disguised and/or encrypted, they were invisible even to the user's operating system other than as inscrutably encoded files. The only way to see the texts within was to run software provided by the publisher that undisguised and/or unencrypted them for display.

The shift from floppy disks that simply carried texts that the user could manipulate with any software she already had to CD-ROM disks whose contents could be examined only with the software provided on the disk itself was a substantial transfer of power from the user to the publisher. The appropriate analogy with old technology would be the invention of printed books that produced blank or garbled pages when photocopied and could be read only using spectacles supplied by the publisher. To pursue this analogy a little further, it was as if each publisher's spectacles worked only with one publication so that the user had to acquire as many different spectacles as she had books. The balance of power shifted slightly toward the user again when CD-ROM drives capable of writing to blank disks became cheap enough to be installed in most new computers from the early 2000s, since this at least allowed the user to make multiple copies of an expensive CD-ROM to use in different locations, such as the home and office.

Although CD-ROMs gave publishers more control over what users did with their publications than they had with floppy-disks, the very fact that these CD-ROMs, like floppies, placed all the data in the user's hands at one time and in one physical object made it relatively trivial for advanced users to release the data from the digital enclosures the publishers put them in. The transition to predominantly online delivery of published materials in the early 2000s marked a much greater shift of power in favour of the publishers, since the user's computer need never contain all the data at one time. If the data are sent over the network in piecemeal fashion, so that at any one time the user's computer is given only part of the object under examination (say, one scene of a Shakespeare play), then it is technically more challenging for the user to create on her machine a complete copy of the entire object (the whole play).

This principle is used in video streaming over the Internet, for example from Google's YouTube service, in which the user's computer receives only a few frames of the moving image at any one time—just enough to display it while the next few frames are being sent—and hence never possesses the entire recording all at once. To reconstruct the whole of the original recording, a streaming-video user has to capture the frames as they are sent and locally recombine them to replicate the complete recording they were drawn from. As we will see, two of the four digital resources examined here use exactly this piecemeal approach to restrict access to the raw materials they present, allowing the user to examine only a small part of them at any one time.

The advantage, from the publisher's point of view, of online content delivery was readily apparent to Charles Chadwyck-Healey whose company (later bought by ProQuest) sold as standalone CD-ROM products the datasets it called *English Poetry, English Verse Drama, English Prose Drama, Early English Prose Fiction*, and *Editions and Adaptations of Shakespeare.* "My concern," Chadwyck-Healey wrote, "was that an English professor would borrow a set of *English Poetry* from the library and spend the weekend making copies to give to his/her students" (Chapter 20, "Literature Online [LION]"). As well as duplicating disks, users could transfer the contents to their local hard disks (by a process called virtualization), which, because they have shorter access times, make the process of retrieving information from the disks faster.

The Chadwyck-Healey company consolidated their literature CD-ROM collections into a single online service called Literature Online (LION) in 1997, accessible only to those with an institutional subscription. In place of physical disks that generated one-off sales and could be copied by the users to reproduce entire datasets, Chadwyck-Healey would now sell a service that generated an annual fee for letting users see only a part of the data at any one time. Charles Chadwyck-Healey was explicit about the benefits and the power relations:

> even if a user downloaded some texts, they would only be a small part of a much larger whole ... We would also know if there was unusual activity on the website, as we were able to monitor the usage of the data on our servers.
>
> *(Chadwyck-Healey Chapter 20, 'Literature Online [LION]")*

With the online service, the time taken for a result to appear on the user's screen is determined not only by the power and speed of the user's computer but also, and to a larger extent, by the power of Chadwyck-Healey's servers and the speed of the network connections between those servers and the user. Although LION does not block users from downloading the whole of a literary work, one work is the most her computer can possess at any one time. The user never possesses a full set of works as she did with the CD-ROM versions, so she cannot repurpose that full dataset for her own ends. This might seem a trivial consideration to many users, but such a transfer of the power to search the dataset from the user to the provider has severe consequences if the provider decides to reduce the range of searching options or unintentionally disables features in its own searching software.

Such unintentional disabling of features is not merely a hypothetical concern. On 28 June 2014, Chadwyck-Healey's parent company ProQuest changed the software that delivers LION to its users, inadvertently breaking LION's proximity-searching and variant-spelling features, and in the worst way possible for investigators who rely on them. After the change, the search results returned from the website are wrong, so that, for example, the counts of hits are untrue, but nothing visible on the screen indicates this fault and no error message is produced. This silent disabling of LION's advanced search options brought a halt to the work of researchers who rely on these features, and at the time of writing (November 2020), the fault has not been fixed. Such things cannot happen when users rely solely on their own computers and locally attached sources of data, and keep them unchanged.

LION is not the worst example of how online delivery gives the providers of datasets far-reaching power over their users. The Henslowe-Alleyn Digitization Project took digital photographs of the collection of papers belonging to the early modern theater impresario Philip Henslowe and his son-in-law, the actor Edward Alleyn, which are kept at Dulwich College in South London, and placed them online for free viewing. Copyright law exists to protect acts of originality and creativity, which for these documents means the originality and creativity of Henslowe, Alleyn, and the other early modern persons who contributed to the documents. Being about 400 years old, any copyrights subsisting in these documents have long since expired, but of course what the Henslowe-Alleyn Digitization Project gives its users are digital photographs of the documents, and the application of copyright laws to new media requires interpretation.

The landmark case of Bridgeman Art Library versus Corel Corporation established in 1999 that under American and British law the photographing of a flat surface containing an image or writing in order to provide the most faithful reproduction of it for viewers or readers constitutes an act of slavish copying, not originality. The United Kingdom Government's Intellectual Property Office published a notice in 2015 confirming this interpretation, remarking that "copyright can only subsist in subject matter that is original in the sense that it is the author's own 'intellectual creation'" and observing that it would be hard to see how anyone could claim copyright "if their aim is simply to make a faithful reproduction of an existing work" (Intellectual Property Office 3).

The Henslowe-Alleyn Digitization Project was funded by a number of private charities and directly by the people of the United Kingdom via the British Academy, which is itself funded by the British government. Despite being made with public money, the Project's website asserts that all the materials it provides are "copyrighted and cannot be downloaded, reproduced, copied, circulated or otherwise used" without the Project's permission and that "The copyright of all manuscripts in the Henslowe-Alleyn Papers belongs to the Governors of Dulwich College" (Ioppolo, "Copyrights, Reproductions, and Permissions"). Neither claim appears to be true under British law. The habit of treating the possession of a document as if this conferred copyright—which as the Berne Convention makes clear arises from originality and creativity not ownership—is deeply and harmfully ingrained in the culture of museums, libraries and archives.

In the case of the Henslowe-Alleyn Digitization Project, this culture of institutional irredentism has practical ramifications because from its inception in 2005 until a technical refurbishment released on 5 March 2020 the project provided access to the digital photographs using proprietary software (Zoomify and Adobe Flash) that prevented the user's computer from receiving the whole of a picture at once. Instead, the user was given a small movable window that revealed only part of the photograph at a time. The Henslowe-Alleyn Digitization Project chose a window 630 pixels wide and 450 pixels deep, which by 2020 was about one-sixth of the typical computer's screen size. The specious assertion of copyright in this case went hand-in-hand with a practical, intentional impediment of the user's freedom to exploit the materials as she would wish.

The Henslowe-Alleyn Digitization Project's reliance on the proprietary Adobe Flash format was the main reason it had to be refurbished. The Adobe Flash software is so poorly written that it provides an easy route for malicious software to infiltrate and take over its user's computer, and by 2020 all the major web-browser manufacturers had announced that for this reason they would stop supporting Flash later that year. Without remedial work, the Henslowe-Alleyn Digitization Project materials would simply disappear from view. The refurbished website uses the Open Source image viewing software called Open SeaDragon (Caton, personal correspondence, November 16, 2020) and the viewing window is now as wide as the user's screen rather than being artificially constrained to a small portion of it. The user is now able to download a single image in the resolution at which she is viewing it—so the better her computer screen, the larger the image she can download—but not to download the image at the full resolution at which it exists on the project's web-server. Nor is the user able to download more than one image at a time. Thus this publicly funded project continues to limit what users can do with its materials and make specious claims about copyrights.

The Database of Early English Playbooks (DEEP) contrasts with the Henslowe-Alleyn Digitization Project in almost every particular except that it too was built with public money by academic subject specialists and is free to use (Farmer and Lesser 2007–). Where the Henslowe-Alleyn Digitization Project asserts its creators' copyrights, DEEP makes its contents available under a Creative Commons Attribution Non-Commercial Share-Alike licence. Where the Henslowe-Alleyn Digitization Project explicitly forbids downloading the project's underlying data, DEEP explicitly encourages it by putting a "Download DEEP Data" link on its homepage, which leads to a page that offers the project's entire contents in HTML, Comma-Separated Values (CSV), and XML form.

Nothing in the design of the DEEP website is intended to limit the user's ability to work with the data, as the Henslowe-Alleyn Digitization Project does. Nothing in the DEEP website relies upon proprietary software, the main search functionality being provided by the language JavaScript, which conforms to an International Organization for Standardization (ISO) standard for scripting languages. Built in this way on open standards, DEEP has an excellent chance of remaining in good working order with minimal maintenance for many years to come.

Conclusion: Open Access, Open Source, Open Standards, and interfaces

Across academia, the Open Access movement is a response to the privatization of public goods and the limitation of users' freedom to do what they wish with these goods. In the publishing of academic journals and books the movement encourages writers to make their materials free at the point of reading, either by paying publishers to publish it (the so-called Gold option, favoured by most publishers) or by putting it on unfettered personal, or subject-centered, or institution-centered websites (the Green option, favoured by most writers). The Open Access movement took much of its inspiration from the Open Source movement in computer software, which promoted the sharing of computer programs in their full-text form, called source code. This is the form in which computer programs are originally written and in which anyone can make sense of them if she knows the particular computer programming language used.

Much of the infrastructure of the World-Wide Web runs on Open Source software, most notably the Apache webserver used by about half the world's websites and maintained by a team of volunteers. With proprietary software the purchaser is not allowed to see the source code and has to trust the supplier's assertions about what it does and how it works, whereas use of Open Source software ensures that many pairs of eyes examine it and can confirm that it does what it is supposed to do. The Open Source model has proved itself to be the best approach for creating robust and maintainable computer systems that last for a long period of time, whereas proprietary software is notoriously prone to rapid obsolescence, as with the Adobe Flash software used by the Henslowe-Alleyn Digitization Project.

Open Standards are the third desideratum of durable digital projects. All digital files are merely strings of zeroes and ones and, to that extent, all are merely long binary numbers. This is equally true of the binary files representing the works that belong to everyone—the texts of Shakespeare's plays, the images of early theatrical documents—and those over which particular individuals legitimately (albeit unhelpfully) claim ownership, most especially the interfaces to that content. This distinction between numbers that belong to everyone and numbers that are private property is invidious, giving rise to the bizarre notion of illegal numbers such as the large prime number used to decode the encrypted video signal on DVD films. The owners of such privatized numbers claim that merely communicating one of them to another person is a crime, and indeed arrests have been made for this supposed offence (see González).

When Shakespeare scholars plan a project to create new content, such as the digitization of a collection of records, they typically design a new interface to present that content to users, and most commonly these days it is a website. It is clear why they do this, since they want to make it as easy as possible for users to access and exploit the new resources. There is, however, a powerful argument that creators should not make new interfaces but should instead merely make the raw materials available on the web and let others pull these together howsoever they wish, as Peter Robinson argued. Such an approach gives users the kinds of freedom they have with the Oxford Complete Works of Shakespeare Electronic Edition.

A compromise alternative to Robinson's position would be to do both: construct interfaces for those who want them but also enable direct downloading of the raw materials by those who want to use them in their own ways. This is what DEEP does, because it is typical of the best Digital Humanities projects and all the best ones do this. Ideally, the raw materials should be encoded using Open Standards so that the user is not confined to any one set of software for making sense of them. But there exists a compromise alternative to this ideal too, which takes into account the reality that not all Open Standards are widely used. The open video format standard called Ogg, for example, is much less widely used than ones based on proprietary formats such as MPEG-4. If a proprietary format is widely used and there are many different tools for consuming and editing materials in this format, its lack of openness need not be especially problematic.

The standards for any digital encoding are merely the rules for making sense of a string of zeroes and ones; they are merely a statement of the contexts within which the various parts of a long binary number have particular meanings. The rules for how to make sense of long binary numbers differ from format to format. The rules of ASCII and PDF encoding (and others) tell us how to turn them back into simple digital texts, the rules of WAV and MP3 encoding (and others) tell us how to turn them back into sounds, the rules of JPEG and TIFF encoding (and others) tell us how to turn them back into pictures, and the rules of MPEG-4 and AVI encoding (and others) tell us how to turn them back into video streams.

We already have many digital tools for decoding texts, sounds, and still and moving pictures that work well for our scholarly and entertainment purposes. These tools are our interfaces. Oftentimes we have so many different tools for turning the binary digits of a particular format into readable and listenable words and pictures that these tools, even the proprietary ones, effectively cease to function as interfaces. If we dislike the way that one tool decodes or allows us to edit a TIFF-encoded file, we simply discard it and use another. With so many tools to choose from, we are tied to none. The only really dangerous encoding formats are those for which there is no rich variety of tools but rather just one proprietary tool for decoding it, as is the case with Adobe Flash.

When we have a wealth of different interfaces for each of our digital file formats, we are liberated from reliance on any particular one. This is the situation we enjoy with files encoded in the formats ASCII, PDF, WAV, MP3, JPEG, TIFF, MPEG-4, and AVI files, which between them represent a large proportion of the total means by which human culture is nowadays created and disseminated. But the files used for digital artefacts in the field of Literary Studies are still not always provided to us in such unfettered formats. The history of digital Shakespeare is littered with failed and broken interfaces that frustrated the scholarly endeavour. There is no reason this has to continue.

References

Bashe, Charles J., et al. *IBM's Early Computers*. 3, MIT Press, 1986.

Bell, C. G., A. Kotok, T. N. Hastings, and R. Hill. "Evolution of the DECsystem 10." *Communications of the Association for Computing Machinery (ACM)*, Special Issue on Computer Architecture, vol. 21, no. 1, 1978, pp. 44–63.

Best, Michael. "Shakespeare and the Electronic Age." *Shakespeare and the Text*, edited by Andrew Murphy, Blackwell, 2007, pp. 145–161.

Chadwyck-Healey, Charles. *Publishing for Libraries: At the Dawn of the Digital Age*. Bloomsbury, 2020.

Eagleton, Terry. *Literary Theory: An Introduction*. Basil Blackwell, 1983.

Farmer, Alan B. and Zachary Lesser. "Database of Early English Playbooks (DEEP): A Database of Bibliographical Information Delivered Over the Internet." 2007–. Available at: http//deep.sas.upenn.edu.

González, Andrés Guadamuz. "Trouble with Prime Numbers: DeCSS, DVD and the Protection of Proprietary Encryption Tools." *Journal of Information, Law and Technology*, vol. 3, 2002, pp. 1–31. Available at: https://era.ed.ac.uk/bitstream/handle/1842/2273/troublewithprimenumbers.pdf?sequence=1&isAllowed=y.

Heidegger, Martin. *Being and Time*. Translated by Joan Stambaugh, revised and with a foreword by Dennis J. Schmidt, State U of New York P, 2010.

Hertzfeld, Andy. *Revolution in the Valley: The Insanely Great Story of How the Mac Was Made*. Foreword by Steve Wozniak, O'Reilly, 2005.

Hockey, Susan, and J. Martin. "The Oxford Concordance Program Version 2." *Literary and Linguistic Computing*, vol. 2, 1987, pp. 125–131.

Intellectual Property Office. "Digital Images, Photographs and the Internet: Copyright Notice Number 1/2014." United Kingdom Patent Office, 2015.

Ioppolo, Grace. "Henslowe-Alleyn Digitization Project." 2005–. Available at: https://henslowe-alleyn.org.uk/.

Lavington, Simon. *Early British Computers: The Story of Vintage Computers and the People Who Built Them*. Manchester UP, 1980.

MacKenzie, Charles E. *Coded Character Sets: History and Development*. Addison-Wesley, 1980.

Manes, Stephen, and Paul Andrews. *Gates: How Microsoft's Mogul Reinvented an Industry and Made Himself the Richest Man in America*. Simon & Schuster, 1994.

Priestley, Mark. *A Science of Operations: Machines, Logic and the Invention of Programming*. Springer, 2011.

Rankin, Joy Lisi. *A People's History of Computing in the United States*. Harvard UP, 2018.

Robinson, Peter. "How We Have Been Publishing the Wrong Way, and How We Might Publish a Better Way." *Electronic Publishing: Politics and Pragmatics*, edited by Gabriel Egan. ITER Press, 2010, pp. 139–155.

Russell, D. B. "'COCOA' Manual: A Word Count and Concordance Generator for Atlas." Atlas Computer Laboratory, 1967.

Shakespeare, William. *Works [The Globe Edition]*. Edited by William George Clark and William Aldis Wright, London, 1864.

Shakespeare, William. *The Complete Works*. Edited by Stanley Wells, et al., electronic edition prepared by William Montgomery and Lou Burnard, Oxford UP, 1989.

Sperberg-McQueen, C. M. "Textual Criticism and the Text Encoding Initiative." *The Literary Text in the Digital Age*, edited by Richard J. Finneran, U of Michigan P, 1996, pp. 37–62.