Brief History Of Character Sets

Leo Ferres

October 2016

The smallest units we will work with in this book are the *character* and, for convenience, the *abstract glyph*. These concepts have endured a fair bit of abuse in the literature and it is therefore important to discuss them at some length here. The words come from the Greek χαρακτήρ and γλυφή, respectively, both meaning roughly "an engraved mark", a sign or symbol. In modern times, *The American Heritage Dictionary* (*AHD*) lists the word "character" as referring to "the symbols of a writing system". This sense of the word has been attributed to William Caxton (ca. 1422–1491), as he used it in his *Eneydos*, an English translation (of a French translation) of Virgil's Aeneid, printed by Caxton himself in 1490[1]. In turn, the *AHD* defines the glyph as a "symbolic figure, […] engraved or incised". Thus, in the case of "glyph", the word seems to carry a more figure-like connotation; compare for example the Phoenician character 𐤉 to an Egyptian *hieroglyph* (ἱερογλύφος, "sacred carving"), such as the "vulture" hieroglyph  , G1 (Birds) in the Gardiner code [see A. H. Gardiner, *Egyptian Grammar*, 1927]. Typography clearly differentiates characters (known as *graphemes*) from glyphs. While the former are abstract units of text as they occur in the writings of natural languages, the latter are concrete graphical presentations of those abstract units, bound to change with typeface and style[2]. The Roman A, the italic *A* and the Serif 𝖠, for instance, are different glyphs of the character A, as are a, *a* and 𝖺 of a.

In computer science, the word "character" has been used since the beginnings of the discipline, inherited in part from earlier work in *telegraphy* [See Eric Fischer's *The Evolution of Character Codes, 1874–1968*]. The history of character encodings is an interesting and relevant one for our purposes, worth retelling in at least in some small detail.

Charles Wheatstone (1802–1875) and William Fothergill Cook (1806–1879) designed the first machine to transmit encoded (not to be confused with encrypted) messages electrically, patented in May of 1837 in London, England. This machine had five needles, deflected in pairs by electro-magnets, thus se-

[1]The word "carecteris" appears in ch.VI, p.24, l.14 of the edition of Caxton's book published in London by Kegan Paul, Trench, Trübner & Co. and by Humphrey Milford, Oxford University Press in 1890, reprinted in 1913. There is a scanned version of the book at `http://www.archive.org/download/caxtonseneydos00virguoft/caxtonseneydos00virguoft.pdf`.

[2]Robert Bringhurst in *The Elements of Typographic Style* defines the glyph as "a version of a character" (p.292) and thinks of it as the "sort" of the digital age (p.331).

lecting one of twenty positions (i.e. $2\binom{5}{2}$ combinations). This encoding obviously left out six of the twenty-six letters of the English script, so they had that $\mathcal{S}'_{(en)} = \mathcal{S}_{(en)} \backslash \{\text{C, J, Q, V, X, Z}\}$. It is hypothesized that those words that included the missing characters were spelled with other similar sounding ones, for example substituting K or S for K, depending on the context, as "seremony" or "karriage".

Meanwhile, in the United States, Samuel Finley Breese Morse (1791–1872) was working on a different, simpler design. He thought of a way to transmit information using just one wire, instead of the five needed for the Wheatstone–Fothergill machine to work. He simply used the presence or absence of electricity as an encoding mechanism. Although in the early days of Morse code codes stood for words in a dictionary, by 1844 and with the help of Alfred Lewis Vail (1807–1857), the famous "dits" (dots, short signals) and "dahs" (dashes, long signals) encoding alphanumeric characters and punctation were born. It may appear as if Morse and Vail were actually working with binary representations. After all, the encodings relied on the presence or absence of an electrical pulse. However, although an early form of digital code, Morse code is a variable-length encoding, meaning some characters are encoded as one or more dots and/or dashes, the string "one" would be encoded (▬ ▬ ▬ ▬ ▪ ●), with subtle pauses (varying from one transmitter person to another) between the signals composing single characters, between characters, between words, etc.

It was not until thirty years later, in 1874, that the real bit-wise encoding of information began, due to Jean-Maurice-Émile Baudot (1845–1903). Baudot had invented a 5-bit code (2^5, 32 codes) plus a shifting mechanism that transformed it into a 6-bit encoding system (although 2 bits were used for encoding the shift), allowing, effectively, 60 characters. Baudot's code was an improvement over Morse Code, since all symbols were exactly the same length, 5 bits plus shifting, making mechanical decoding much simpler. For the first time, there were also control codes: shift and "note". The latter would print an asterisk * at the receiving end, which meant that an error had been made. This would later become the encoding for DELETE or DEL. Another peculiarity of the Baudot code is that zero was left undefined, meaning nothing, not very different from the contemporary zero byte. Arround 1899, Donald Murray (1866–1945) improved Baudot's piano-like system (where several keys had to be pressed at once to "compose" the "nickle", the 5 bits) by using a typewriter keyboard that hid the bit twiddling from the user. For the typewriter method, Murray needed what would later become format characters, and he introduced the now famous CR (Carriage Return) and LF (Line Feed) characters, which would return the drum to the initial position, and advance the paper one row.

In the 1930s, the bit-encoding methodology had already taken strong footing, and the need to standardize character encodings started to emerge. Standardization is not an easy task, many companies, governments and individuals have already functioning machinery that they would need to change, usually at enormous costs. Standardizing character encodings was not the exception. The "Comité Consultatif International Téléphonique et Télégraphique" or CCITT for short, was the first to attemtp a standardization, to relative success, by

devising the International Telegraphy Alphabet No. 2 (ITA-2), which was also adopted, with minor changes, as the American Teletypewriter code (USTTY). The ITA-2 was a 5-bit character encoding, much like Baudot code, but added several format "effectors" that scrambled characters. New control characters were added (BELL, NUL), although other writing systems, because of the number of characters in their scripts, omitted them and added their own, as was the case with MTK-2, the Russian version of ITA-2, and the Cyrillic alphabet.

By 1955 there were already 29 different encodings available, and another breakthrough had happened four years before. The UNIVAC (**UNIV**ersal **A**utomatic **C**omputer I) had been delivered to the United States Census Bureau on March 31, 1951, and is now widely heralded as the first computer that could work with numeric as well as alphabetic characters [See Eckert, Weiner, Welsh and Mitchell, *The UNIVAC System*, Joint AIEE-IRE Computer Conference, ACM, 1951, 6-16], and then the standardization efforts began.

Particularly, with the proliferation of and reliance on computers, the military were quite interested in making all their system communicate with each other. In 1957, the United States Army Signal Corps introduced FIELDATA, a 6-bit encoding with a "tag" bit, thus, essentially, a 6-bit + shift, totalling 128 possible characters. FIELDATA had a strong impact on what was to become ASCII a few years later, not surprising given the "proponents" of the encoding. FIELDATA, as the rest of the encodings since the time of Wheatstone and Fothergill Cook, were hardware dependent. FIELDATA, for instance, was designed for the computers and the software the military were using at the time (the UNIVAC 1100 and Cobol, the **CO**mmon **B**usiness **O**riented **L**anguage).

It was not until 1963 that the encodings stopped being machine-dependent (with one exception) and became standards in the true sense of the word, that is, proposed and analyzed by associations gathering people from different walks of life, including the military, the government and industry. This was the case of the American Standards Association (ASA), and the X3.2 Subcommittee, which in 1963 proposed the now famous and widespread ASCII (**A**merican **S**tandard **C**ode for **I**nformation **I**nterchange), a full-fledged 7-bit encoding with no particular machine in mind. ASCII was the standard, except, that is, for the IBM 360, which in 1963 shipped with IBM's EBCDIC, an 8-bit, non-continuous alphabet encoding. EBCDIC became very influential as well given the success of the IBM 360, and this hampered ASCII somewhat; although history has it that the latter clearly won. EBCDIC, however, is still found in legacy code.

The ASA and ECMA International, an international private standards organization, made a few modifications to the 1963 version of ASCII. In this version of 1967, it was still a 7-bit encoding, but there was consensus in adding the lowercase letters in the "control characters" section (from code 108 or `0x6c` onwards), and some considerations were made to languages other than English, which helped in the internationalization of ASCII (although in 1972, The ISO (International Organization for Standardization, Organisation Internationale de Normalisation, and also ratified by ECMA-6, produced a new version of ASCII that replaced the $ for ¤ (a4), the universal currency sign). The 1967 standard was a highly successful one, lasting for over 30 years.

With the exponential growth of computer sales around the world, the need for dealing with characters of different languages was becoming a problem. To help with this, in 1987, ISO devised the 8-bit "multi-part" character sets, where codes after `0x7f` could be reassigned depending on the language. These exchangeable character sets were called *parts*, and named ISO-8859-d, where $1 \leq d \leq 16$. For instance, **Part 1** encodes (almost) all characters from Western European scripts: Danish, Dutch, English, Faeroese, Finnish, French, German, Icelandic, Irish, Italian, Norwegian, Portuguese, Rhaeto-Romanic, Scottish Gaelic, Spanish, and Swedish. Of these, Dutch, Finnish and French are not complete. This was revised and missing characters were added in ISO-8859-15.

At long last, in 1991, the efforts were directed towards unifying ISO-8859 parts into a single database: a 16-bit encoding called the Unicode Standard version 1.0/ISO 10646 (or **U**niversal **C**haracter **S**et, or UCS). Unicode now encodes all known characters of all scripts, plus historical characters of all kinds, including Byzantine musical symbols, and characters of extinct languages. Unicode is the chosen character database for our algorithms. For the purposes of this book, you may think of Unicode characters as the mathematician's number. Everything will be done over this list of integers.