NWG/RFC #373 14 July 1972
NIC 11058 SU-AI

ARBITRARY CHARACTER SETS

by John McCarthy

It would be nice to be able to have documents stored in computers that could include arbitrary characters and to be able to display them on any CRT screen, edit them using any keyboard, and print them on any printer. The object of this memorandum is to suggest how to get there from here with special reference to the ARPA network.

Where are we now?

(1) At present, there is 96 character ASCII, and everyone agrees that it should be included in any larger set.

(2) Many installations are dependent on 64 character sets which do not even include the lower case latin alphabet.

(3) At the Stanford Artificial Intelligence Laboratory, we have a 114 character set that includes 96 character ASCII and which is implemented in our keyboards, displays, and line printer

(4) Printers are becoming available that get their character designs out of memory, for example, the Xerox XGP printer, one of which we are getting.

(5) The IMLAC type display has the character designs in main memory so that changing the displayed set is just a matter of reloading the memory.

(6) Many display systems share the character generator among many display units. In some of these, e.g. the Datadisc, arbitrary sets are probably feasible (using kludgery to be described later), but in other systems, e.g. our III's arbitrary sets are not feasible.

One possible approach to communication in expanded character sets is to produce an expanded standard set of characters, perhaps using 8 or 9 bits and expect new equipment to implement this set. This approach has the disadvantage that it will be very hard to get agreement on what the next step should be, and even if formal agreement is realized, many groups will find it in their interest to ignore the standard.

Therefore, I would like to suggest that the next step be to arbitrary character sets. I suggest implementing this in the following way:

(1) There be established a registry of characters. Anyone can register a new character. Each character has a unique number, 17 bits should be enough even to include Chinese. Besides this, each character has a name in ASCII usually mnemonic. Finally, the character has a design which is a picture on a 50 by 50 dot matrix.

(2) Besides the registry of characters, there is a registry of characters sets, which different groups are using for different classes of documents. A registered character set has a registry number and a table giving the correspondence between the character codes as bit sequences and the registered character numbers.

(3) Associated with a document is a statement of the character code used therein. This may be one of the registered codes or it may contain in addition modifications described by an auxiliary table giving the code correspondence with registered character numbers. A character code may have an escape character that says that the next character is described by its registry number. The statement of the character code may be a header on the document or the receiver may have to learn it by some other means, e.g. because its library catalog entry contains this information.

(4) Devices such as printers and displays draw characters in different ways and standardization doesn’t seem feasible at present. Therefore, it is necessary to provide a way of going from the standard description of a character using a 50 by 50 dot matrix to whatever method the device uses. This is up to the programmers who are supporting the device. Some may choose to manually create files describing how registered characters are implemented. They may find it too much work to provide for all the characters and to update their files when new characters are registered. Others will provide programs for going from the registered descriptions to descriptions compatible with their implementations. Perhaps most will hand tailor the characters most used and provide a program for the others.

(5) The easiest device to handle is the line printer because it is slow. At the beginning of the print job, the SPOOL program will look up the character set and load the printer's memory with the character designs used in the particular document. Sometimes, it may have to go through the network to one of the computers that stores the registry in order to find out what to do.

(6) Display systems that have a character memory for each display unit can be handled in about the same way. Users will occasionally experience delays when the display programs are surprised by unfamiliar characters.

(7) Display systems that share character memories require more complicated treatment. The object is to keep the memory large enough to keep all the characters that the current set of users is using and to handle the required table lookups from the different character codes in a nice way. There will be limitations on the diversity of character sets that can be in use simultaneously. Systems like the Datadisc that only look up the character when it is first written can be extended to work with large sets. Systems that have to look up each character code 30 times per second in order to maintain the display won't work so well.

I have no special ideas about how to make keyboards adaptable to arbitrary sets. Each user may have to fend for himself.

In this memorandum so far, I have ignored typography, i.e. the fact that in printed documents the same letter may be printed in many fonts. Perhaps, each character in each font will require a separate registered description, but with a constant difference between the numbers of the same character in different fonts. Installations will again have to decide what font distinctions they will implement.

Some other issues that might be considered are whether means can be provided to adapt texts automatically to the line and page lengths of the different devices.

It seems to me most likely that the typographical problems cannot be solved at this time, and it would be best to adopt conventions for registering character designs at this time, and leave typography for later.

In my opinion, there is no real obstacle to establishing the registry in the ARPA network now, getting the standards organization to work, and being able to exchange documents in extended character sets as soon as the various installations can acquire the printers and display devices.

It is the present policy of the Stanford Artificial Intelligence Laboratory to acquire no more devices that are wedded to fixed character sets.