Format Recommendations For Prestel Telesoftware

June 1986

CET Council for Educational Technology

This document supercedes the two booklets, Format Recommendations for Prestel Telesoftware, published February 1982 and Prestel Telesoftware Guidance Notes for Terminal Designers, published April 1982.

Contents

Published by the Council for Educational Technology,
3 Devonshire Street, London W1N 2BA

Printed by Direct Design (Bournemouth) Ltd, Printers,
Butts Pond Industrial Estate, Sturminster Newton, Dorset DT10 1AZ

Publisher's Note

Telesoftware is one of a number of vaguely defined terms which have been developed as part of the growth of information technology. The term was first coined in 1977 to describe the broadcasting of computer programs over television channels to intelligent teletext terminals.

Now, it is more widely used to describe the transmission of computer programs and data from one computer to another, either by broadcast radio or television, or via telephone lines.

This document describes the method by which computer programs and data are transmitted using the United Kingdom Public Viewdata Service *Prestel.*

The first part is a short introductory section describing the origin of the Prestel Telesoftware Format Recommendations, their history and their relationship to other work in this important area; the main body of the document consists of the format recommendations themselves.

The recommendations are generally framed from the point of view of a designer writing software for a downloader. However, where particular points are of importance for Information Providers who would wish to load telesoftware on to Prestel, these are included at the relevant place in the text.

Your comments on this document are always welcome, and should be sent to:

Council for Educational Technology for the United Kingdom
3 Devonshire Street
London
W1N 2BA

marked for the attention of the IT programme manager.

Introduction

In September 1980, the Council for Educational Technology (CET) invited computer manufacturers, software agencies and representatives of Prestel to discuss the formulation of a set of recommendations for the format of telesoftware on Prestel. As a result of those discussions, the Council for Educational Technology published a document in January 1981, entitled *Prestel Telesoftware Format Recommendations*.

The recommendations described a method of loading a single program source file on to a viewdata system, such as Prestel, and which was particularly suited to the transfer of a program written in BASIC from Prestel to the disc or tape backing store of a microcomputer.

At the end of 1981, as a result of work done on the Commodore PET, an extension to the recommendations to allow the transmission of files containing any character from an 8–bit character set was developed and subsequently published in February 1982.

Since that date, the experience gained from the increasing use of the format recommendations by CET during the CET Telesoftware Project, and also by other information providers on Prestel who have used the format recommendations, has resulted in the inclusion in the recommendations of some new facilities and also some clarification of the existing format.

Other countries, particularly those in Europe with viewdata systems which are either based on or are similar to Prestel, are using these recommendations as a starting point for a telesoftware service. In the longer term, however, it is desirable that, at least within Europe, a common telesoftware standard for viewdata systems is adopted. Work is continuing to define a telesoftware standard that will eventually supercede the recommendations described in this document.

Format Recommendations for Prestel Telesoftware

This document describes a method of storing computer programs and data on Prestel in such a way that they can be retrieved automatically by a microcomputer. The recommendations provide a means of detecting and recovering from errors in transmission that are encountered when using telephone connections to the Prestel computers. They also describe how the data should be organised on the Prestel system.

The recommendations are designed to work on viewdata systems which have the same general characteristics as the current Prestel system. Particular features which are required by the recommendations are summarised in a later paragraph.

The intention of the recommendations is to provide a means whereby a microcomputer can retrieve a file from the Prestel computer. This file may be a computer program or a data file, or some other information. The original assumption was that the program or data would be stored immediately on disc or cassette, either as it was being retrieved or immediately after the whole file had been retrieved. Since the first publication of these recommendations, various Prestel Information Providers have used the recommendations together with some rules of their own to extend the range of applications which can be accommodated using the format recommendations. Such extensions include retrieving a program, placing it directly into the memory of the microcomputer and executing the program immediately; it is also possible to retrieve a program which, when executed, will retrieve either another program or a file of data which can then be stored or used immediately by the first program. It must be emphasised that the format recommendations described here do not include the above facilities. However, using the format recommendations together with some additional rules can allow the sort of applications described above.

Files

The basic entity in telesoftware is the file. This is simply a quantity of data of a particular length and consists of one or more characters, where a character is an octet (8 bit byte) which can have a value between 0 and 255. The file is usually given a name, called its filename.

File Conversion

The basic entity on the Prestel system is the 'page'. A page can consist of up to 26 'frames' and each page is identified by a unique number (shown on the top row of the page). Each frame or page on Prestel consists of 24 rows, each of 40 characters. The top row and the bottom row are reserved for system messages, for example, the name of the information provider, the number of the page and the price of the page.

The remaining 22 rows are available to the information provider. Each of the 40 character positions on each row can contain either an alphanumeric symbol or one of 64 graphic symbols. It is not possible, however, to store characters whose ASCII value is less than 32 or greater than 127.

This means that to store a file of data consisting of a sequence of eight bit characters on the Prestel system, the file must first be converted from its original form to one in which only the allowed characters (ASCII 32 – 127) are present. In addition, in order to provide a means of identifying the start and end of each block of telesoftware characters on a Prestel frame, and to provide a means of indicating 'End of file', 'End of line' and other items of information, some form of coding scheme is required.

Error Checking

Because it is possible for pages of information retrieved from Prestel to be corrupted by noise on the telephone line, a method is required for detecting when corruption of the retrieved data has occurred. The method used is to start each block of data with a unique sequence of characters (|A) and put another unique sequence (|Z), together with a checksum, after the end of the block of data.

It is also possible for a series of pages to be retrieved in the wrong order. This occurs when the commands sent to the Prestel computer by the microcomputer are affected by noise on the telephone line. The result can be either that the command is lost altogether, or that it causes the wrong page to be sent. In order to allow the microcomputer to detect, and also possibly to correct this, each block has a short section at the beginning which distinguishes it from the preceding and following blocks.

Prestel transmits and expects to receive characters as 7 data bits plus one even parity bit. The downloading software should check the parity bit on each character received and, if a parity error is detected, should rerequest the page using the '*00' command. Once a page has been received without parity errors, the parity bit is no longer used and each received character should have the parity bit set to zero before being processed further.

After requesting Prestel to send a page, the downloading software should employ a timeout facility and should assume that, after a finite period during which no more characters are received, then a complete page or frame has been received.

Organisation on Prestel

In order to give the microcomputer some information about the program or data which the file that is about to be retrieved contains, the first frame always contains a filename and usually a count of the number of Prestel frames which the file occupies. This first frame is known as the *header frame*.

The second and subsequent frames contain the file itself.

When a file will not fit on one Prestel page (consisting of 26 frames), a second, continuation, page is used. This page is reached by using the '0' route from the 'z' frame of the page. If further pages are required, then the '0' route from the 'z' frame is used to reach further continuation frames.

The final frame of a telesoftware file contains a special marker indicating that the end of the file has been reached.

Coding

As has already been mentioned, the set of characters which an Information Provider can use on a Prestel frame is restricted. This means that, in order to load a file of data on to Prestel frames, the file must first be converted so that only the allowable characters are used.

The main way of doing this is to give one character a special meaning. The character chosen is the | character (coded 7/12) and this is known as the *sequence introducer.*

The figures 7/12 refer to the position of the character in the table of Prestel transmission codes included in the *Prestel Terminal Specification*, table 2A. The first digit is the column number and the numbers following the / character are the row number, ie 7/12 means column 7, row 12. In addition, since each column contains 16 entries, the ASCII value of the character can be calculated by multiplying the column number by 16 and then adding the row number, eg the | character is 7/12, which corresponds to the ASCII value 124.

Whenever the | character occurs on a Prestel frame containing telesoftware, it is followed by another alphabetic or numeric character which determines its meaning. For example, the two character sequence |A indicates the start of a telesoftware block. These sequences are known as escape sequences.

The full list of such character sequences is as follows:

Sequence	*Coded*	*Meaning*
|A	7/12,4/1	Marks the start of a telesoftware block and is followed by:
|G	7/12,4/7	This sequence is followed by the frame letter of the frame on which this telesoftware block occurs. The frame letter is a lower case alphabetic character. Where there is more than one telesoftware block on a frame, the frame letter is followed by two numeric characters (see Sequencing of Telesoftware Blocks and the section on Multiple Blocks per Frame below).
|Z	7/12,5/10	Marks the end of a telesoftware block. It is followed by three numeric characters which are the checksum for the block (see section on Checksum Calculation below).
|I	7/12,4/9	This sequence acts as a terminator for those escape sequences which were added to the original format recommendations. It is a very powerful feature and is described fully in the section Escape Sequence Terminator below.
|F	7/12,4/6	This escape sequence is inserted after the last character in the file and signifies End Of File. How a downloader handles the end of file condition is up to the designer of the downloader. The following observations may be of assistance: – On a CP/M based system, the handling of end of file depends on whether the file is a text file or not. A text file is written to disc with the final 128 byte block padded to the end of the block with ↑Z (1/10) characters. If the file is not a text file, then the final block of the file is usually padded with

		nulls (0/0) until the last 128 byte block is filled. Other operating systems adopt different conventions.
|L	7/12,4/12	End of line. This sequence can be used by an Information Provider to signify the end of a line in a source file. It is usually used to replace the characters CR (carriage return) and LF (line feed). Note, however, that different microcomputers use different characters for end of line; some use just CR or LF, others use the more usual CR LF, others may use LF CR and it is possible that some microcomputers may have a completely different system for signifying end of line. When you encounter this escape sequence, you should write to the output file whatever character or sequence of characters your particular microcomputer expects at the end of each source line. This escape sequence is also used on the header frame to separate the filename from the length field (see The Header Frame below).
|¾	7/12,7/13	This sequence is used to represent the three–quarters character ¾ which is used for another purpose. Whenever this escape sequence is encountered, you should simply write the character ¾ to the output file.
|E	7/12,4/5	This is the way in which an Information Provider can insert the character | in the output file. Remember that the character | has been given a special meaning. As with |¾ above, if you encounter this escape sequence then you should simply write the character | out to the output file.
|0	7/12,3/0	This group of escape sequences is provided to allow characters outside the range 32–127 to be included in the file. See the section on Character Code Extension below.
|1	7/12,3/1	
|2	7/12,3/2	
|3	7/12,3/3	
|4	7/12,3/4	
|5	7/12,3/5	
|T	7/12,5/4	This escape sequence is not in common usage but it has been included to allow an Information Provider to mark the start of the header section of the file, ie the filename and length (see below).
|D	7/12,4/4	This escape sequence is also not commonly used – in contrast to |T described above, it is used to allow an Information Provider to mark the start of the data section of the file, as opposed to the header section which gives the filename and number of frames that the file occupies (see above).

All other escape sequences are at present undefined. Further, all escape sequences not listed above are reserved for future use with the exception of all the lower case letters. These are intended for Information Providers and others who are experimenting with extensions to this set of recommendations. During the development phase of a new feature, lower case letters should be used for a new escape sequence. When the feature has been tried and tested and is generally accepted, then one of the reserved characters will be allocated to that function and the feature will be incorporated into this recommendation.

Escape Sequence Terminator

When the format recommendations were originally framed in 1980, a total of six escape sequences were defined. In 1981/82 a further seven escape sequences were defined. |I (one of the new escape sequences defined) was intended to make it easier for other, as yet unforeseen, sequences to be added easily, and with as little disruption as possible to users of downloaders written to comply with earlier publications of the recommendations.

Supposing we wanted to add a new escape sequence to the current list which would have the effect of reserving a certain number of blocks of disc space for our use. The escape sequence would be followed by one or more characters which would specify the amount of space to reserve.

The main problem we would encounter in introducing this facility is that any users with telesoftware programs written before this escape sequence was announced would no longer be able to download the particular file containing it. The escape sequence which was intended to help the telesoftware program to reserve sufficient space for the file would instead be treated as part of the file and would mean that, when users downloaded that file, they would get several seemingly spurious characters (the escape sequence and the following parameter) in the middle of their copy of the file.

To get around this problem, the following rules were devised. Escape sequences are divided into three categories:–

1 The set of escape sequences defined in the format recommendations published February 1982. These are

|A, |Z, |L, |I, |0, |1, |2, |3, |4, |5, |F, |E and |½

2 Those escape sequences to which generally accepted meanings have subsequently been attached. At present these are

|G, |T and |D

3 Those remaining escape sequences which have not yet been allocated a specific meaning.

When a telesoftware downloading program encounters an escape sequence from categories 2 and 3, it stops taking characters from the frame, converting them and storing them in the output file. Instead, if it is an escape sequence it recognises, then it processes the escape sequence, otherwise it simply ignores all the following characters until it reaches the escape sequence |I. The escape sequence |I instructs the downloading program to continue taking characters from the frame, convert them and store them in the output file.

One further point for Information Providers concerning escape sequences is very important. If for some reason an escape sequence will not fit completely within the current frame (leaving room for the final |Zxxx), then the complete escape sequence and any following parameters up to the occurrence of |I MUST be moved to the next frame. In no circumstances can an escape sequence be split across two frames.

Other Special Characters

Apart from the escape sequences described above, there is one other character which has a special meaning in the telesoftware file. This is the character ¾ (7/13). Whenever this character is encountered in a telesoftware file and it is NOT part of the escape sequence (7/12,7/13), it should be converted to the space character (2/0). The reason for this is a little complicated but is related to the way in which Prestel stores frames of information. If the last, say, six characters on a line of a Prestel frame are all spaces, then Prestel can reduce the number of characters transmitted to the microcomputer (or terminal) by stripping off the trailing spaces and transmitting a CR LF sequence instead. Since we do not want this feature for telesoftware, the way to ensure that Prestel does not strip trailing spaces is to convert the last character on each line when the file is loaded. If it is a space, then it is converted to the ¾ character.

Over the life of the format recommendations, the instructions for this situation have varied. Originally EVERY occurrence of the space character was replaced by the ¾ character. Then the decision was made to allow the conversion of spaces to ¾ to be purely optional. The current position for uploading files is that conversion of a final space on a line is mandatory; conversion of spaces elsewhere within a line is optional.

The above only affects those people loading files on to Prestel. For designers of software for downloading programs, the rule is always that if you encounter the ¾ on its own, then convert it to a space.

Checksum Calculation

For each telesoftware block, a checksum is calculated on all the characters between the escape sequences |A and |Z. The checksum is calculated as follows:

> When the escape sequence |A is encountered, the checksum is set to zero. As each character is received, its value is exclusive-OR'ed with the current value of the checksum. At the end of the block, when the escape sequence |Z is encountered, the checksum calculated above should correspond to the value of the three digits following the |Z sequence. The eighth, parity bit of each character MUST be set to zero before this calculation and the resulting value will always be between 000 and 127.

For example:-

```
|AThis is a checksum test|Zxxx
```

The value left in the checksum after all the characters between |A and |Z have been XOR'ed together is xxx in decimal. The three digits following |Z in the example above should be x x and x. If a terminal finds that its calculated version of the checksum does not equal the value transmitted, then it can assume that there has been corruption of the page and can ask Prestel to resend the page. Note that the checksum following the |Z is always three characters with leading zeroes if these are needed.

The viewdata command which causes Prestel to retransmit a page is the sequence

'*00'. It is strongly recommended that designers of downloading software should keep a check on the number of times that a particular page is rerequested and, if some particular number of retries is exceeded, should abandon the attempt to fetch the program and inform the user of the problem. It is always possible that an Information Provider has inadvertently put up a telesoftware file with an incorrect checksum on the page.

It is also recommended that you check for parity errors and, if a parity error occurs on a frame, you should rerequest the frame in the same manner as if a checksum error had occurred. This helps to improve the error checking.

Sequencing of Telesoftware Blocks

In the normal situation where the Information Provider is only putting one telesoftware block on each frame, then the following sequence will be at the start of each frame, immediately after the |A sequence indicating start of block

|Gc|I   where c is the frame letter of the current frame.

Since Prestel frames will be consecutively labelled 'a', 'b', 'c', and so on up to 'z' and, if a continuation page is used, will start again at 'a', it is possible for the downloading software to detect when a frame is received out of order, or when a request for the next frame has been lost because of noise on the telephone line. A limited amount of corrective action can be taken and the user can be informed of what is occurring.

The Header Frame

As mentioned earlier, the first frame of a telesoftware file will be the header frame containing the name of the file and the number of Prestel frames (excluding the header frame itself) that the file occupies. For example:

|A|Ga|ISORT.BAS|LOO3|Zxxx

It is recommended that the header frame should always be on the 'a' frame of a page, as in the example. However, several Information Providers do also put the header frame on the 'd' frame of the first page.

Note that the end of the line escape sequence |L is used in the header frame to separate the filename from the length information. As with the checksum, the length information is always three decimal digits with leading zeroes if these are needed. If for some reason the Information Provider is unable to provide the number of frames which the program occupies, then the number 999 should be used instead. This simply tells the downloading software that the number of frames is not known.

Character Code Extension

In order to represent characters in the range 0 – 255 by combinations of characters from the range 32 – 127, it is necessary to use some sort of shift technique. This is similar to the use of the SHIFT key on a typewriter to give you upper case (capital) letters instead of lower case letters. The same keys are used with the shift key to produce twice as many different symbols as there are keys on the typewriter.

At the beginning of a telesoftware file (ie the header frame) there is no shift in use. The characters in the range 32 – 127 (subject only to the rules for ¾ and the escape sequences given above) represent themselves; that is they are written to the output file exactly as they appear on the frame. In fact, at the start of every new telesoftware file the shift offset, as it is known, is always reset to zero.

The escape sequences which change this situation are the six escape sequences |0, |1, |2, ... |5. Each of these six escape sequences causes the downloader to alter the way in which it deals with the characters until another one of these escape sequences is encountered.

If one of the above sequences is encountered in a telesoftware file, then the downloader should change its shift offset (initially zero) according to the following table:–

Escape sequence	Shift offset set to
|0	0
|1	–64
|2	+64
|3	+96
|4	+128
|5	+160

When a character is read from a Prestel frame by the downloader, the ASCII value of that character should be added to the current value of the shift offset. The resulting character should then be written to the output file.

Note that the shifts are what are known as locking shifts, ie each change to the shift offset variable is permanent until another control sequence which changes the shift offset is encountered.

One further point for Information Providers; there is only *one* means to represent any particular character (0 – 255) on a telesoftware file. The following table shows which shift offset *must* be selected in order to represent a particular character in the file:

Character value	Shift offset which *must* be selected
0 – 31	|1
32 – 127	|0
128 – 159	|2
160 – 191	|3
192 – 223	|4
224 – 255	|5

Multiple Blocks per Frame

In some fairly specialised applications, it may be necessary to put several small telesoftware blocks on one Prestel frame. The use of this facility is not encouraged unless it is absolutely necessary because it means that a downloader must be even more complex and it is likely that only a few implementations of the telesoftware downloader will have this capability.

In order to try and prevent the downloader from missing some of the blocks on a frame, the escape sequence used for identifying telesoftware blocks (|G) is extended in this situation. Now, in addition to the frame letter which follows the |G, there are also two extra characters.

The first is a number in the range 0 – 9 which identifies which block this is on the frame. The first block will be numbered block 0, the second block will be numbered block 1, and so on with a maximum of ten blocks on the frame.

The second number is the number of the last block on the frame, using the same numbering as described above. For example, the third block on a frame which has 8 blocks in all would have at the start of the block

|A|Gc27|I ..... data .... |Zxxx

Characteristics of the Prestel System on which these Recommendations Rely

These recommendations assume that the Prestel database consists of a number of pages, each page being made up of a maximum of 26 frames. To step from one frame to another, the viewdata command '#' is used. In order to reach a continuation page from the 'z' frame of the current page, the viewdata routing command '0' is then sent. If any error is encountered in the reception of any frame, then the viewdata command '*00' will result in the frame being retransmitted.