

CHARACTER STREAM PROCESSING

- Character set
- Character escape conventions
- Canonical form
- Line editing conventions

CHARACTER SET CONSIDERATIONS

minimum: graphic range equal to a
typical office typewriter

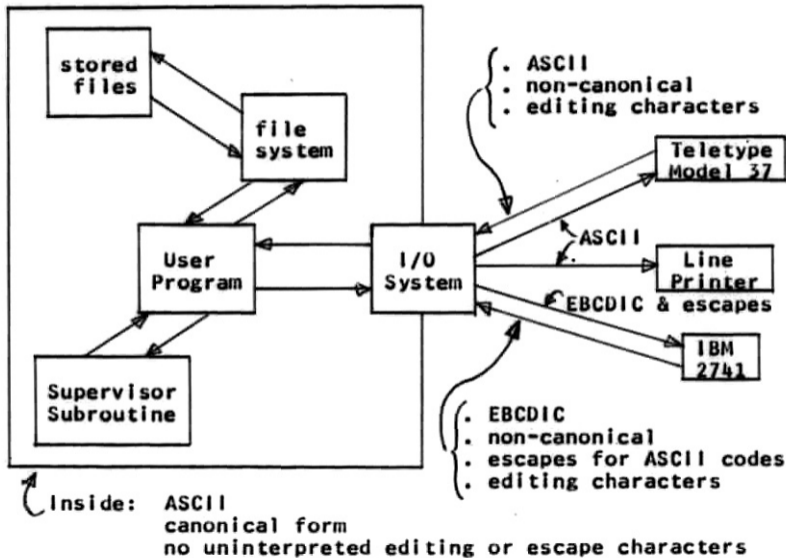
support: throughout system

- programs and data files
- program language literals
- symbolic file names
- symbolic subroutine linkage
- punched cards
- typewriters and displays
- printers

CHARACTER STREAM PROCESSING CONSIDERATIONS

1. Device Independence--any program with any terminal.
2. Unambiguous relation between printed image and stored string.

EXAMPLE OF DEVICE INDEPENDENT, UNAMBIGUOUS CHARACTER SET SUPPORT



ESCAPE CONVENTIONS FOR DEVICE INDEPENDENCE

Typed at 2741 terminal:

```
y = board_position <i,j >;
```

Character string passed to program:

```
y = board_position i,j ;
```

CANONICAL REDUCTION EXAMPLE

1. (user) locate "order"
2. (editor) the order has been filed.
3. (user) change "l" to "ll"
4. (editor) the order has been filled.

CANONICAL REDUCTION METHOD

typed: for that

stored:

_	BS	f	_	BS	o	_	BS
r			t	h	a	t	

1. Graphics typed in the same column are in ASCII collating sequence, separated by backspaces.
2. White space is represented by ASCII "blank" characters regardless of how typed.

LINE AND PRINT POSITION EDITING

1. typed: y = lenz#gth(string4);
 stored: y = length(string4);
2. typed: Siincer@Sincerly## yours,
 stored: Sincerely yours,
3. typed: we have five####two left.
 stored: we have two left.

OBSERVATIONS FROM USAGE

- Economically implementable
- Effective human interface
- User can ignore special features until needed
- Interaction between canonical form and horizontal tabulation