

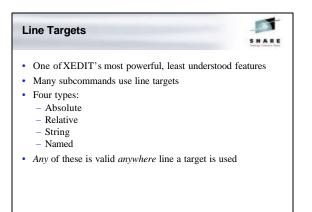
XEDIT Power Tools

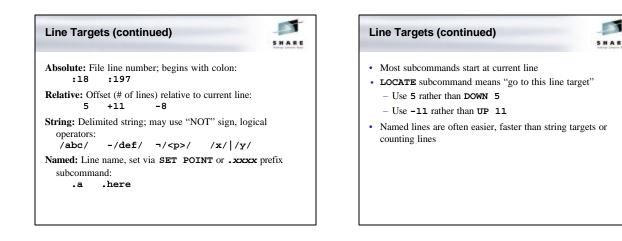
- Line targets
- Chained LOCATEs
- Little known subcommands
- Column commands
- SOS commands
- Invocation options
- SET and QUERY
- 327*x* keys

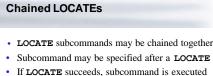
Programmable keys

.....

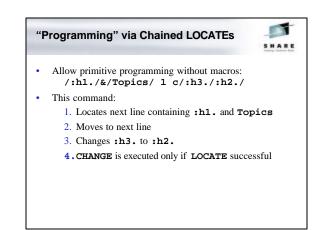
- (PF, PA, Enter)
- Macros
- APIs
- Selective line editing
- Prefix macros
- Display management
- Text processing







- If LOCATE fails, subcommand is not executed
- Reduces terminal I/O (useful on slow lines!)





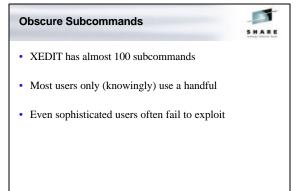
SHARE

Use chained LOCATES with REPEAT for complex operations:

/:h1./&/Topics/ 1 c/:h3./:h2./

```
repeat *
```

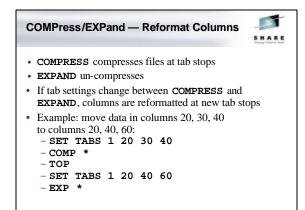
Same as previous, but repeated through rest of file

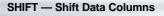


LPrefix — Logical Prefix

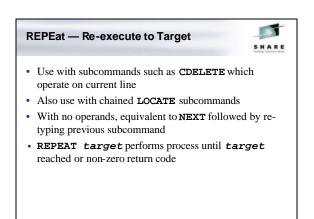


- Executes prefix subcommand on current line
 Available even in line-mode XEDIT!
- Faster than moving cursor to current line, then back to prefix area, typing prefix subcommand, pressing Enter
- Example: delete a section from a file:
 - LOCATE start of section
 - Set pending DD prefix subcommand with LP DD
 - LOCATE end of section
 - Use another LP DD to delete section



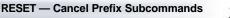


- · Moves data left or right
- Data moved, not file view like LEFT, RIGHT, RGTLEFT
- Deletes/spills data if necessary
- Respects ZONE columns



Merge — Merge Lines

- · Overlays groups of lines at specific column position
- Two line targets specify what to merge, column position operand specifies where
- Complex rules govern merging blank/non-blank data
- Complicated usage, but more efficient than other techniques
- · Typically used in macros





- · Prefix subcommands not yet executed are pending
- Use QUERY PENDING to locate
- **RESET** avoids having to locate to cancel

UPPercase/LOWercase — Convert Text



- · Convert one or more lines to upper/lower case
- Text between **ZONE** columns is changed
- Useful after text uppercased by SET CASE UPPER

CANCEL — QUIT Unmodified Files

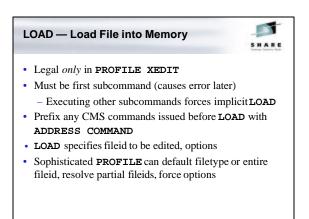


- XEDIT can edit many files at once
- CANCEL quits all unmodified files
- Changed files remain, QQUIT or FILE individually
- · Easier than pressing PF3 many times
- Safer than typing QQ many times!

COUnt — Count String Occurrences



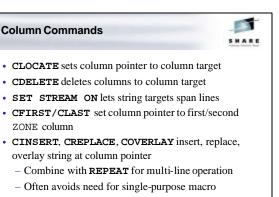
- · Counts string occurrences
- Use to:
 - Check quote or parenthesis nesting
 - Determine scope of change before starting
- String to be counted must match exactly
 - Actually uses CHANGE code under the covers
 - CASE IGNORE not honored (like CHANGE)
 - KEDIT adds second IGNORE for CHANGE, which KEDIT COUNT honors



Column Commands

-

- Column pointer points to a column position
- Setting shown at top of screen, on SCALE line
- Query via QUERY/TRANSFER/EXTRACT COLUMN
- Column commands set/use column pointer
 - Some use column target operand:
- Subset of line target varieties: absolute, relative, string (including logical NOT)
 - No logical AND, logical OR, or names

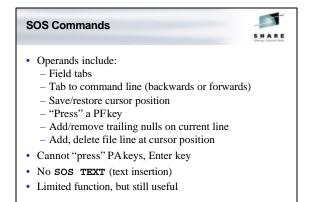


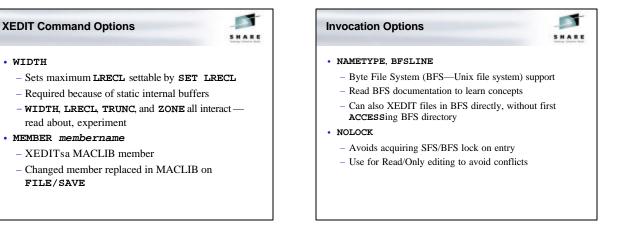
· CAPPEND macro (sort of) does "column append"

SOS Commands



- SOS: Screen Operation Simulation
- · Legacy of EDGAR, early full-screen CMS editor
- ${\tt SOS}$ functions use screen cursor position
- Simulates terminal actions (keystrokes)
- · Designed for use from macros or PF keys



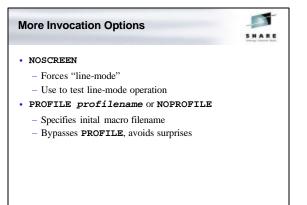


Options Useful for Applications

-

• NOMSG

- Suppresses XEDIT messages
- Avoids SET MSGMODE OFF in applications
- NOCLEAR
 - From display-management applications, avoids screen flash on XEDIT entry
 - Causes MORE... if in line-mode



Update Mode: Create Source Updates



- · Allows creating discrete updates to source files
- Changes made while editing are automatically saved as updates
- Supports source files up to LRECL 255
- **CTL** *ctlfn* applies updates using CMS multi-level update scheme (CNTRL and AUX files)
- The only way to do product maintenance!

Update Mode: Create Source Updates



- UNTIL updatename applies updates only through updatename, showing code image before a specific update
- **SIDcode** *string* creates Service IDentifier codes in columns 64-71 of updates
- Merge merges all updates into single update
- Seq8/NOSeq8 control whether 5- or 8-character sequence numbers are expected on source
- Incr *n* controls minimum update line increment

SET and QUERY



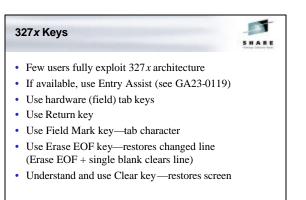
- 80+ **SET** options provide extensive tailorability
- Some poor defaults (SCALE, STAY, MSGLINE)
- Things to learn about:
 - QUERY options which have no matching SET (RING, NBFile, LENGTH, TARGET, LASTLORC)
 - CASE IGNORE

 - Complex synonyms, LINEND option
 - Uses for SET NULLS ON, SET FULLREAD ON
- STATUS macro displays/saves current settings

Customizing the Screen Many SET commands change screen appearance (CMDLINE, PREFIX, SCALE, etc.) Customize screen to your taste Experiment with SET PREFIX NULLS, SET NUMBER ON, SET SHADOW OFF Consider: SET MSGLINE ON 2 23 OVERLAY or SET MSGLINE ON -1 24 OVERLAY with SET CMDLINE TOP

Customizing the Screen

- Some screen features (title line, TOF/EOF lines, etc.) cannot be changed directly
- XEDIT builds many of these from message repository
- · Customize with user override for the repository!
 - Add WIDTH, last subcommand to top line of screen
 - Add userid to bottom right
 - Add end/top of range info to top/end of range lines

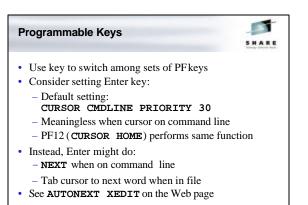


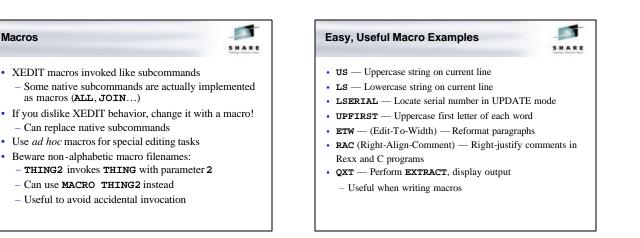
Programmable Keys



- PF keys, PA keys, Enter key are all programmable
- Most users don't set keys
- · Consider whether defaults are effective for you: - Do you use all PFkeys regularly?

 - Do you have 24 PFkeys?
 - Are there commands you type frequently?
 - Do you have **SET NULLS ON** in your **PROFILE**? (If so, default PA2 is wasted)
- Understand BEFORE/AFTER/ONLY/IGNORE operands





Macro Tips

Macros

- EXTract Put information into macro variables - Often returns information unavailable by other means
- **PREServe** /**Restore** Save/restore most settings
- CURSOR Position cursor on screen
- **COMMAND** Force subcommand execution
- MACRO Force macro execution
- MSG/EMSG/CMSG Display messages

An XEDIT API: DMSXFL xx • DMSXFL xx manipulate in -memory files • Call **DMSXFLXX** from assembler programs to: - Check existence of a file

- Read from a file
- Write to a file
- Locate a specific line
- Used extensively by **FILELIST**, **RECEIVE**, others
- Similar to CMS file I/O interface
- Documented in CMS Application Development Guide for Assembler, and in DMSXFL comments

Another API: DMSXMS



- DMSXMS MODULE performs SORT function
 - SORT actually only front-end macro for DMSXMS
- Use similar macro**MODULE** combination for other highperformance XEDIT extensions
 - Uses CMS SUBCOM interface
 - Examples: DMSXDB and DMSXUQ to delete blank/duplicate lines

Selective Line Editing

- Most users know of the ALL macro

 Uses SET SELECT/DISPLAY/SCOPE

 More macros improve selective line editing:

 TALL After ALL, toggle between selective/full display
 TN, TU Go to next/previous selected line after TALL
 - ALSO After ALL, add to selection (e.g., after changing ZONE)
 - **EXCLUDE** After **ALL**, remove lines from selection

ALLSIZE — Displays count of lines in current selection

Prefix Macros



- Most users use prefix subcommands: C, F/B, etc.
 Many are unaware of prefix macros
- SI, >, < prefix subcommands are prefix macros
- SET PREFIX SYNONYM maps prefix subcommands to macros
- · Unrecognized prefix subcommands invoke macros
- Add prefix subcommands using prefix macros
- Prefix macros receive special parameter list
 - Can support both command line and prefix invocation

Useful Prefix Macro Examples G — Embed (get) lines from another file R — Recover deleted lines U — Convert lines to uppercase L — Convert lines to lowercase ETW — (Edit-To-Width)—Reformat paragraphs ? — Set current line and move cursor to command line Any frequent task that requires moving cursor to command line is a candidate for a prefix macro!

Display Management

- XEDIT can be used for display management
 - Not a true screen manager
 - Best for small, simple applications
 - Advantages: portability, price
- SET RESERVED defines static lines on screen
- SET CTLCHAR allows user-defined input fields, highlighting, etc. in **RESERVED** lines

Display Management: READ

- **READ** subcommand in macros traps (stacks) user input: keys, screen changes, etc.
 - Useful for prompting from macros
- **READ** ALL traps input to fields in **RESERVED** lines
- **READ NOCHANGE** reflects screen changes to macro without changing file
- READ ALL NOCHANGE interprets CTLCHARs in file lines, not just RESERVED lines
 - Allows use of file as display template

XEDIT Text Processing



- Typical example: alter file under program control
 - **PROFILE** option passes control to application macro
 - NOMSG option suppresses XEDIT messages
 - Pass macro arguments after closing parenthesis:
 'XEDIT A FILE (NOMSG PROFILE XYZ)' args
- Perhaps use "skeleton" file as starting point:
 - Alter default values with CHANGE
 - Insert, delete lines as appropriate

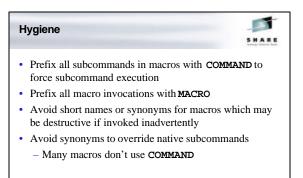
XEDIT Text Processing

- Use READ to prompt user
- Indicate result via SAVE followed by QUIT rc
- Exploit in-memory sort—faster than CMS SORT
- Clean up after errors—avoid leaving user in unexpected XEDIT session!

Example: XEDIT-based File Search Utility



- Common problem: search CMS files for a string
- · Many public-domain tools, vendor products exist
- Free solution: XEDIT, using macro as **PROFILE** XFIND takes search string, file specification
 Searches file(s), lists occurrences
- · Crude but usable—and less than 50 lines of code!
- Handles PACKed files, unlike most such tools
- Can be extended with controls for column, case, **ARBCHAR**, output, etc...



Hygiene

- Always specify SET never rely on implicit SET
- Use **SET LINEND OFF** in macros that execute user input or file data to avoid surprises
- Consider SET AUTOSAVE if system unstable
- Save file when you stop to think even stable systems can fail!

XEDIT on Other Platforms KEDIT — Mansfield Software Group XEDIT for Windows, MS-DOS, OS/2 Very similar to CMS XEDIT Extensions exploit workstation capabilities Mature, popular product Macros use Rexx or KEXX (built-in REXX subset)

- Windows version beautifully merges Windows and 3270 paradigms: intuitive, configurable — usable!
- www.kedit.com offers many powerful macros

XEDIT on Other Platforms



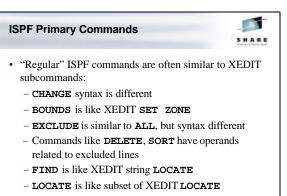
- THE The Hessling Editor
 - Freeware UNIX editor by Mark Hessling
 - Modeled on XEDIT, with KEDIT DOS influences
 - Uses Rexx macros
 - Includes ISPF compatibility features
 - Check out www.lightlink.com/hessling/
- uni-Xedit The Workstation Group
 - XEDIT for UNIX
 - uni-Rexx, uni-SPF also available
 - Check out www.wrkgrp.com

XEDIT for ISPF Users ISPF users have a particular challenge: XEDIT is too *close*: they're constantly confused Some of the confusion can be destructive Customization through macros can help a lot! Example: ISPF Line commands XEDIT calls them Prefix subcommands Use synonyms to change behavior ISPF users can make XEDIT tolerable!

ISPF "Line" Commands



- B undefined in XEDIT, means Before in ISPF
 Synonym to P in XEDIT
- **R** undefined in XEDIT, means Repeat in ISPF - Synonym to " (duplicate) in XEDIT
- A means Add in XEDIT, After in ISPF
 - Synonym to ${\bf F}$ in XEDIT
- Some others much harder to simulate, but can at least come close



- XEDIT does not support UNDO at all

Conclusions

-

- XEDIT is powerful, rich in function
- · Inexperienced users can add skills easily
- Learning more about it increases productivity
- · Clones enable skills transfer to other platforms
- · Read the manual!
- · Note unfamiliar facilities, try them
- Experimenting is fun and easy!

> What XEDIT tips can you share?

