# *Dbedit*

## Editing TurboIMAGE Datasets


| | Page |
|---|---|
| How can Dbedit help me with my work? | 3 |
| Accessing Dbedit | 4 |
| Finding entries with a known key | 6 |
| What if I don't know the key value? | 9 |
| Modifying data and global changes | 11 |
| Dbedit subcommands | 14 |
| Inserting and deleting data | 16 |
| MPE/iX Critical Item Update (CIU) | 18 |
| General guidelines | 19 |

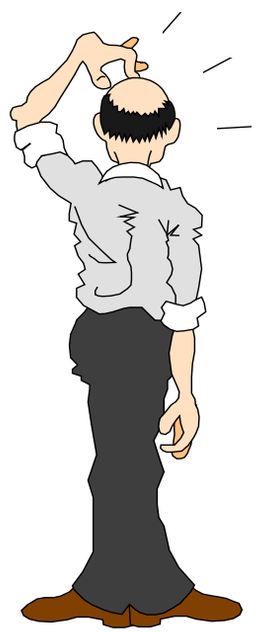
# *Editing a database with Dbedit*

- Dbedit uses simple commands to perform these editing operations:
  - listing entries
  - adding an entry
  - modifying an entry
  - deleting an entry
  - applying global changes to entries

- It can work on chains of entries or related entries
- It can modify key items

# How can Dbedit help me with my work?

- Dbedit is useful in many ways
  - Debugging programs
  - Fixing bad data
  - Building prototype databases



# *Accessing Dbedit*

- Step 1: Run Suprtool
- Step 2: Use the BASE command to open a database
- Step 3: Use the EDIT command to start Dbedit

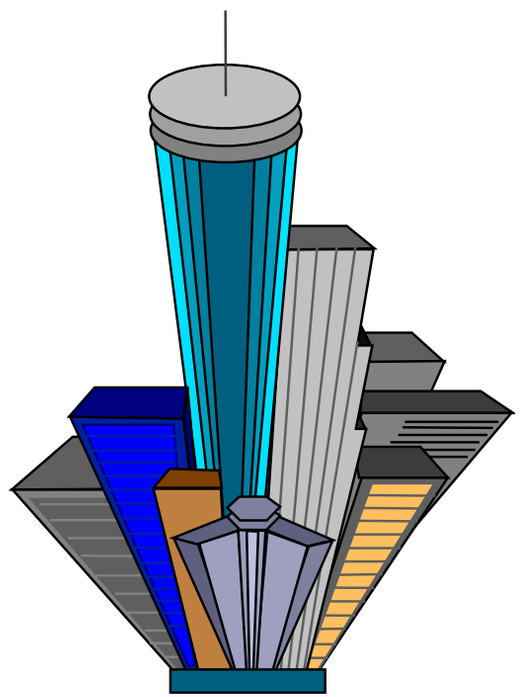
```
:run suprtool.pub.robelle
>base store.pub
>edit
#                    {Dbedit prompt}
```

# *Dbedit is built into Suprtool*

- Dbedit is a Suprtool component that functions independently

- Dbedit commands:

```
#form sets
#list m-customer
#modify d-sales;updatekey
#add d-inventory
#delete
#change m-product
#exit
```



# *Finding an entry with a known key*

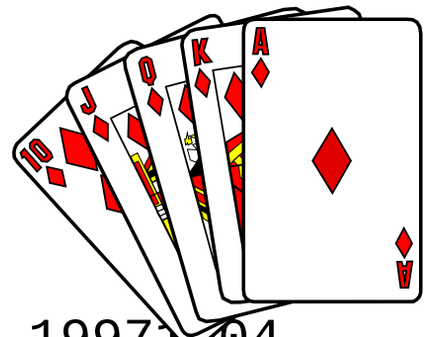
- Use LIST *setname* and specify a key value at the prompt

```
  #list m-customer
List in File: M-CUSTOMER
CUST-ACCOUNT       >10020_______
CITY          = Surrey          CREDIT-RATING   = 200000
CUST-ACCOUNT = 10020            CUST-STATUS     = 20
NAME-FIRST    = Walley          NAME-LAST       = Nisbet
STATE-CODE    = BC
STREET-ADDRESS  = 8877-149th Street
              (2)
POSTAL-CODE       = V3T4W2

List in File: M-CUSTOMER
    CUST-ACCOUNT         >_____Prompts for next value_____
```

# *Finding a chain of entries*

- Use LIST *setname* to specify a chain of entries

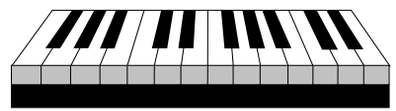


```
   #list d-sales
   List in File: D-SALES
     CUST-ACCOUNT          >10020____
       PRODUCT-NO          >_____{press return to omit}____
CUST-ACCOUNT     = 10020              DELIV-DATE       = 19971004
PRODUCT-NO       = 50511501           PRODUCT-PRICE    = 9831
PURCH-DATE       = 19971000           SALES-QTY        = 2
SALES-TAX        = 2753               SALES-TOTAL      = 22415

CUST-ACCOUNT     = 10020              DELIV-DATE       = 19971028
PRODUCT-NO       = 50512501           PRODUCT-PRICE    = 14660
PURCH-DATE       = 19971028           SALES-QTY        = 1
SALES-TAX        = 2052               SALES-TOTAL      = 16712
List in File: D-SALES
     CUST-ACCOUNT          >______{press return to end}___
```

# How can I change the search key?

- Use the KEY option to specify a different key and alter the search path



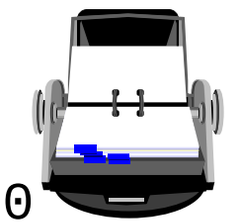
```
#list d-sales;key = product-no
List in File: D-SALES
    PRODUCT-NO        >50512501_
    CUST-ACCOUNT      >_________{enter value or press return}_
CUST-ACCOUNT    = 10003          DELIV-DATE      = 19971016
PRODUCT-NO      = 50512501       PRODUCT-PRICE   = 14562
PURCH-DATE      = 19971016       SALES-QTY       = 1
SALES-TAX       = 2039           SALES-TOTAL     = 16601

CUST-ACCOUNT    = 10020          DELIV-DATE      = 19971028
PRODUCT-NO      = 50512501       PRODUCT-PRICE   = 14660
PURCH-DATE      = 19971028       SALES-QTY       = 1
SALES-TAX       = 2052           SALES-TOTAL     = 16712

List in File: D-SALES
      PRODUCT-NO        >_________
```

# What if I don't know the key value?

- Use the ALL option to sequentially display all the entries in a dataset



```
  #list m-customer;all
List ALL records in File: M-CUSTOMER
CITY              = Vancouver       CREDIT-RATING    = 200000
CUST-ACCOUNT      = 10010           CUST-STATUS      = 20
NAME-FIRST        = Wayne           NAME-LAST        = Humphreys
STATE-CODE        = BC
STREET-ADDRESS    = #403-1075 Comox
                (2)
POSTAL-CODE       = V5T1H6

CITY              = Coquitlam       CREDIT-RATING    = 200000
CUST-ACCOUNT      = 10014           CUST-STATUS      = 20
NAME-FIRST        = Elizabeth       NAME-LAST        = Welton
STATE-CODE        = BC
STREET-ADDRESS    = 2788 Oxtoby Place
.....etc....
```

# *Listing related entries from other datasets*

The RELATED option with the LIST command searches for entries in the selected dataset and in related datasets

- If a master dataset is specified, Dbedit retrieves a master entry and then goes through the paths to detail sets

  #**list m-customer;related**

- If a detail dataset is specified, Dbedit retrieves a detail chain, then goes through the paths from master sets

  #**list d-sales;related**

# *Changing a noncritical field*

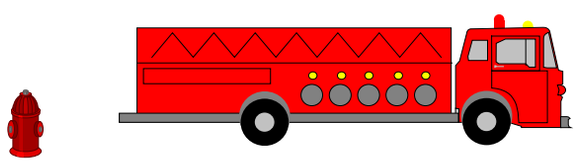
- Use the MODIFY command to change the values of noncritical fields in a record

```
#modify d-inventory : unit-cost
Modify within File: D-INVENTORY
  SUPPLIER-NAME  >STD Ribbons
  PRODUCT-NO     >105391
Enter new values(or <Return> to leave as is):
  SUPPLIER-NAME  = STD Ribbons
  PRODUCT-NO     = 105391
  UNIT-COST      = 500
                   _____  {enter new unit cost}
```

# How can I modify a critical field?

- Use the UPDATEKEY option to modify critical items

```
#modify d-inventory;updatekey
Modify within File: d-inventory
  SUPPLIER-NAME >*_______{no new value}
  PRODUCT-NO    >_______{press Return to omit}
Enter new values (or <Return> to leave as is):
  SUPPLIER-NAME  = STD Ribbons
                   STD Ribbon  {new key value}
```



# *Can I make a global change to a field?*

- If you need to change a field value in the entire dataset, use the CHANGE command

```
#change m-supplier
Enter existing key value to find:
  SUPPLIER-NAME >ACME
Enter new key value to replace with:
  SUPPLIER-NAME >ACME SUPPLY

SUPPLIER-NAME = ACME  CITY       = Los Angeles
STATE-CODE    = CA    STREET(1) = 100 Main
STREET(2)     =       ZIP-CODE  = 91201

  OK to change this entry[no]:Y
```

# *Subcommands in Dbedit*

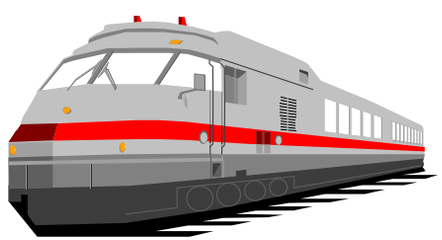
- In response to the Dbedit prompt for a field value, you can use the following subcommands:

| | |
|---|---|
| * | No new value |
| ? | Display the TurboIMAGE format or field |
| // | Quit the command |
| \\ | Quit the command |
| Ctrl-Y | Quit the command |
| [ | Treat rest of line as data, not as subcommand |
| ' | Set this field to all blanks (batch use) |
| = | Execute a calculator command |

# *Moving around in a field list*

- Try these subcommands to move to other entries in a field list:

| | |
|---|---|
| >> | Go to the end of the field list |
| << | Go to the beginning of the list |
| >3 | Go three fields forward in the list |
| <3 | Go three fields back in the list |
| *@fieldname* | Go to the fieldname |



# *Adding new entries to a dataset*

- Use the ADD command to insert a new record into a dataset

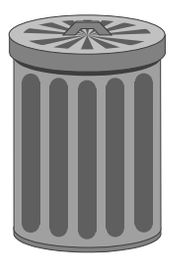
```
#add m-supplier
Add to File: M-SUPPLIER
  SUPPLIER-NAME >ACME
  CITY          >Los Angeles
  STATE-CODE    >CA
  STREET(1)     >100 Main
  STREET(2)     >_________  {press Return to omit}
  ZIP-CODE      >91201
```

# How can I delete an entry?

- It's easy to remove an entry using the DELETE command

```
#delete m-supplier
Delete from File: M-SUPPLIER
SUPPLIER-NAME >ACME
SUPPLIER-NAME = ACME  CITY       = Los Angeles
STATE-CODE    = CA    STREET(1)= 100 Main
STREET(2)     =       ZIP-CODE  = 91201

Is this the entry to delete[no]:Y
```

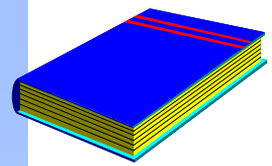


# *MPE/iX Critical Item Update (CIU)*

- CIU allows programs to modify critical search and sort fields in detail datasets using DBUPDATE
- By default, IMAGE databases have CIU disabled
- Dbedit requires CIU for the CHANGE command and the UPDATEKEY option with the MODIFY command
- Two ways to enable CIU
  1. **set *basename* ciupdate = on**
  2. **set *basename* ciupdate = allowed**

# General guidelines

- Dbedit works best on single entries or chains of entries
- Dbedit uses keyed access, but serial access can be specified with the LIST ALL command
- All Dbedit commands support the asterisk (*) subcommand
- All commands support a restrictive field list
- A semicolon (;) separates a command from its options



# Summary

- Like a text editor for dataset entries
- ADD, CHANGE, DELETE, LIST, MODIFY
- Updating key values