# *HowMessy is Your Database?*

- **A Robelle Tutorial**

  **Interex 1995**

  **Toronto, Canada**

  **August 15 - 18, 1995**

# *What's Inside*

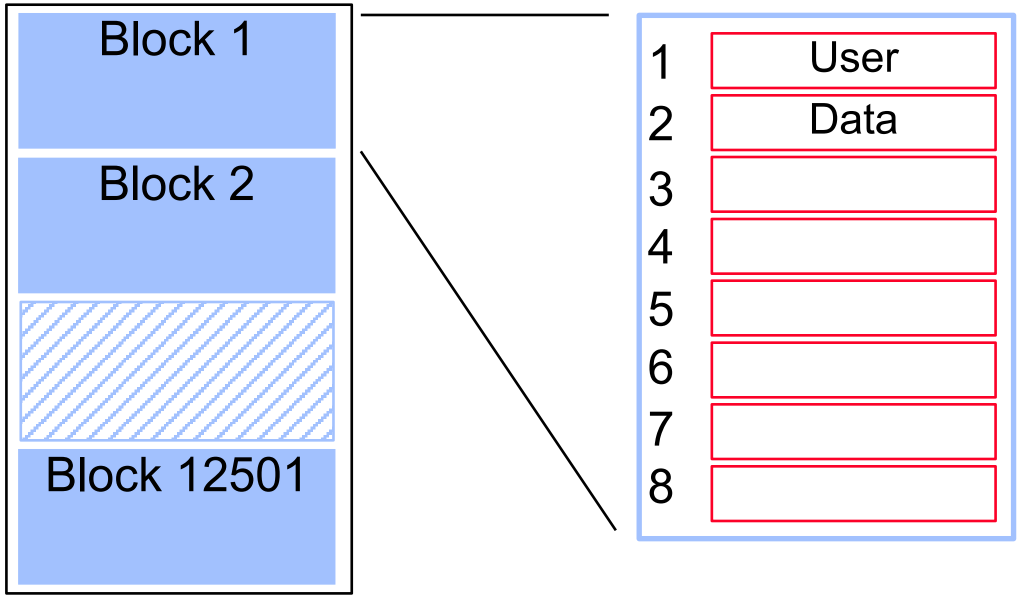
| | Page |
|---|---|
| How messy is your database? | 3 |
| Hashing algorithm | 6 |
| Interpreting master dataset lines | 13 |
| Master dataset solutions | 16 |
| HowMessy sample report (detail dataset) | 17 |
| Repacking a detail dataset | 22 |
| Detail dataset solutions | 26 |
| Estimating response time | 28 |
| Automating HowMessy analysis | 29 |
| Summary | 31 |

# How messy is your database?

- A database is messy if it takes more I/O than it should
- Unnecessary I/O is still a major limiting factor even on MPE/iX machines
- Databases are messy by nature
- Run HowMessy or DBLOADNG against your database
  - HowMessy is a bonus program for Robelle customers
  - DBLOADNG is a contributed library program

# Blocks

- TurboIMAGE does all I/O operations in blocks
- A block may contain many user records
- More entries per block means fewer I/Os
- Fewer I/Os means better performance



Block 1

Block 2

Block 12501

1 User

2 Data

3

4

5

6

7

8

Capacity: 100001

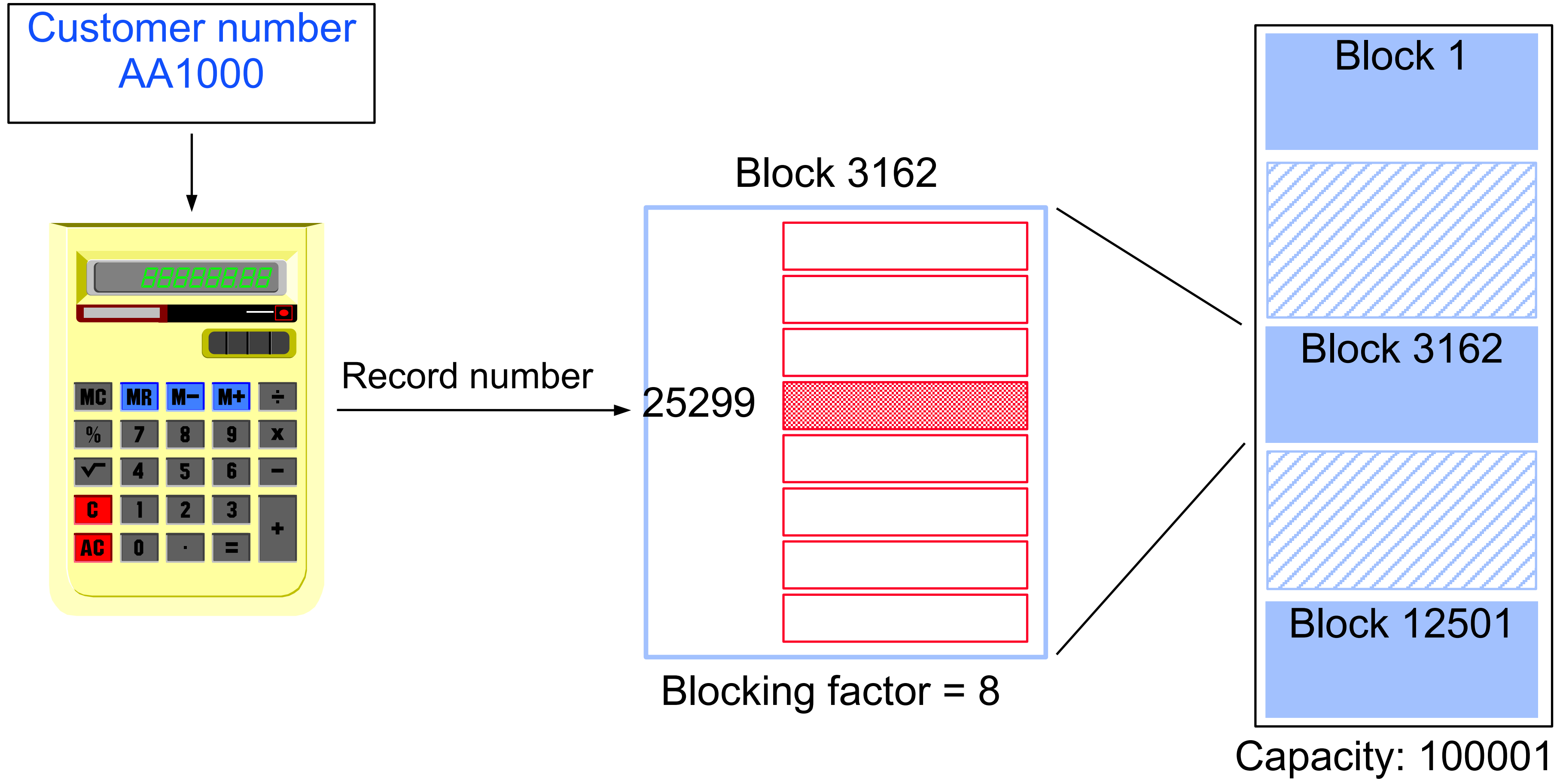
Blocking
factor = 8

# Record location in masters

- Search item values must be unique
- Location of entries is determined by a hashing algorithm or a primary address calculation
- Calculation is done on search item value to transform it into a record number between one and the capacity
- Different calculation depending on the search item type
  - X, U, Z, and P give random results
  - I, J, K, R, and E give predictable results

# *Hashing algorithm*

- Customer number AA1000 is transformed into a record number



Customer number AA1000

Record number

25299

Block 3162

Blocking factor = 8

Block 1

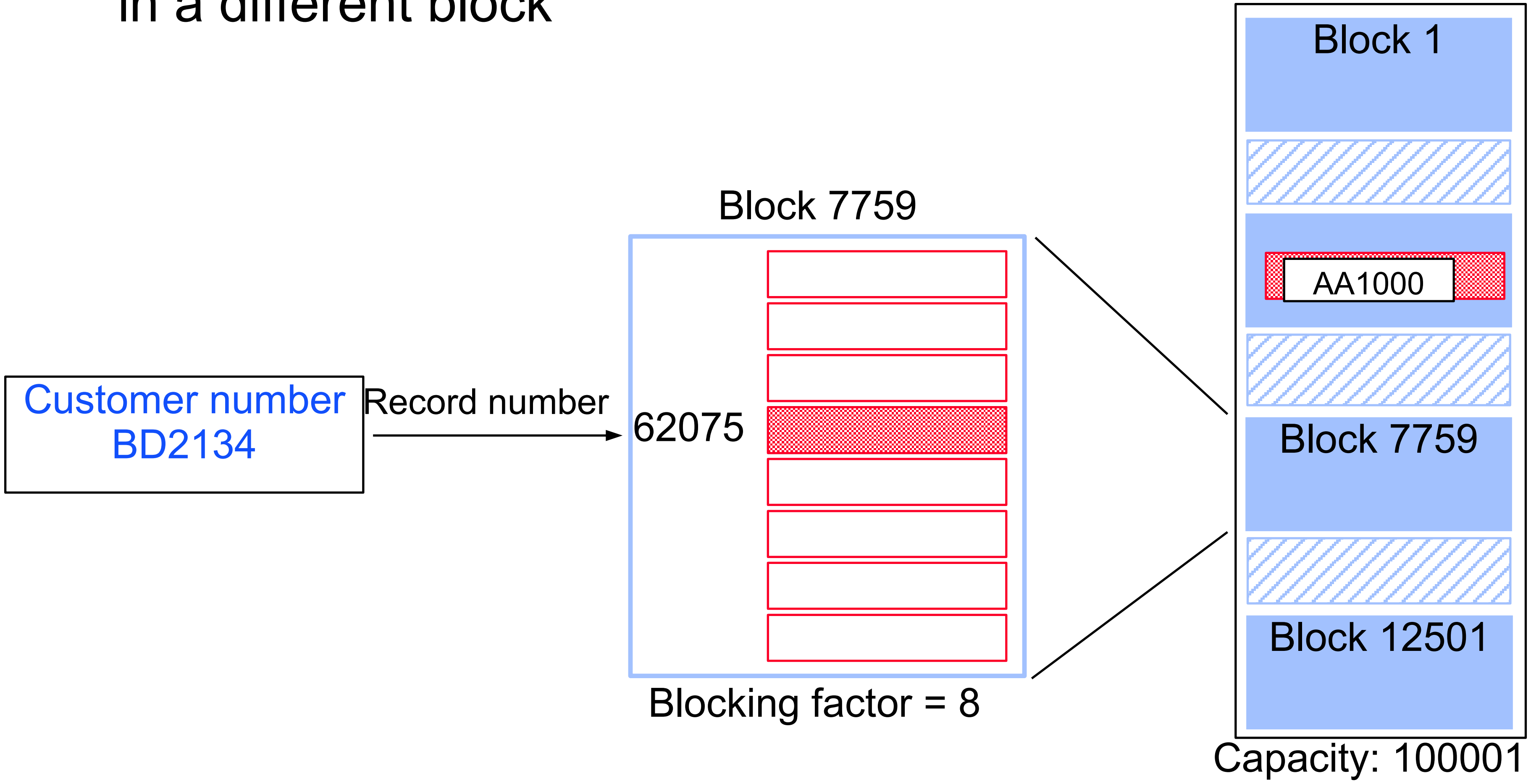
Block 3162

Block 12501

Capacity: 100001

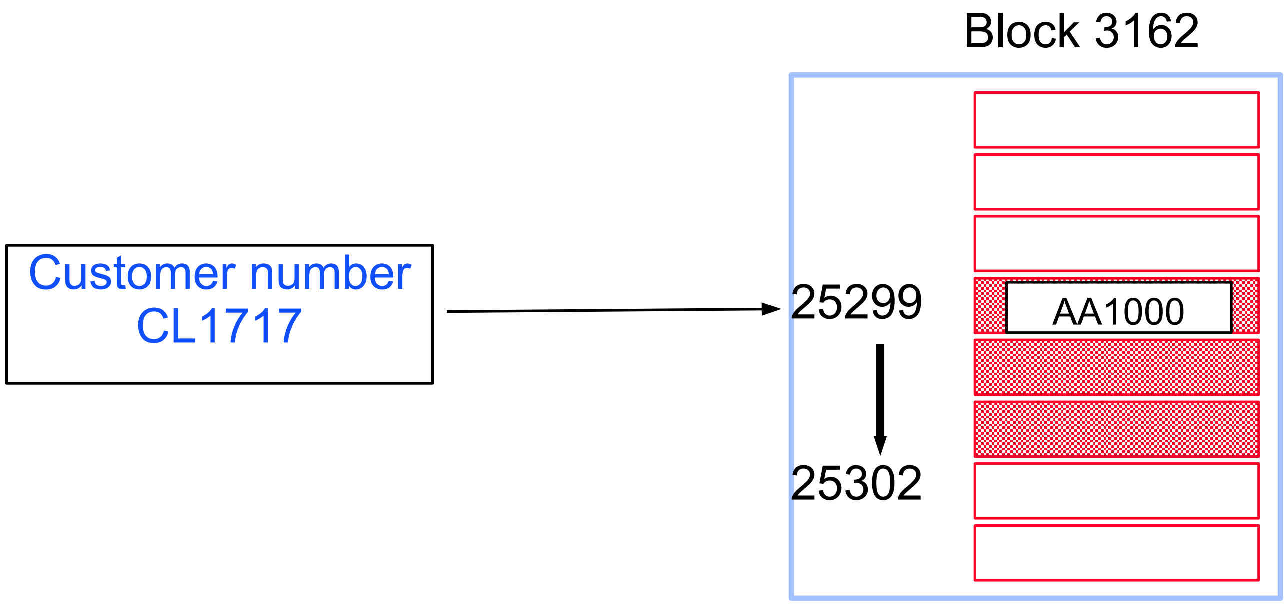
# *Hashing algorithm (no collision)*

- Customer number BD2134 gives a different record number in a different block



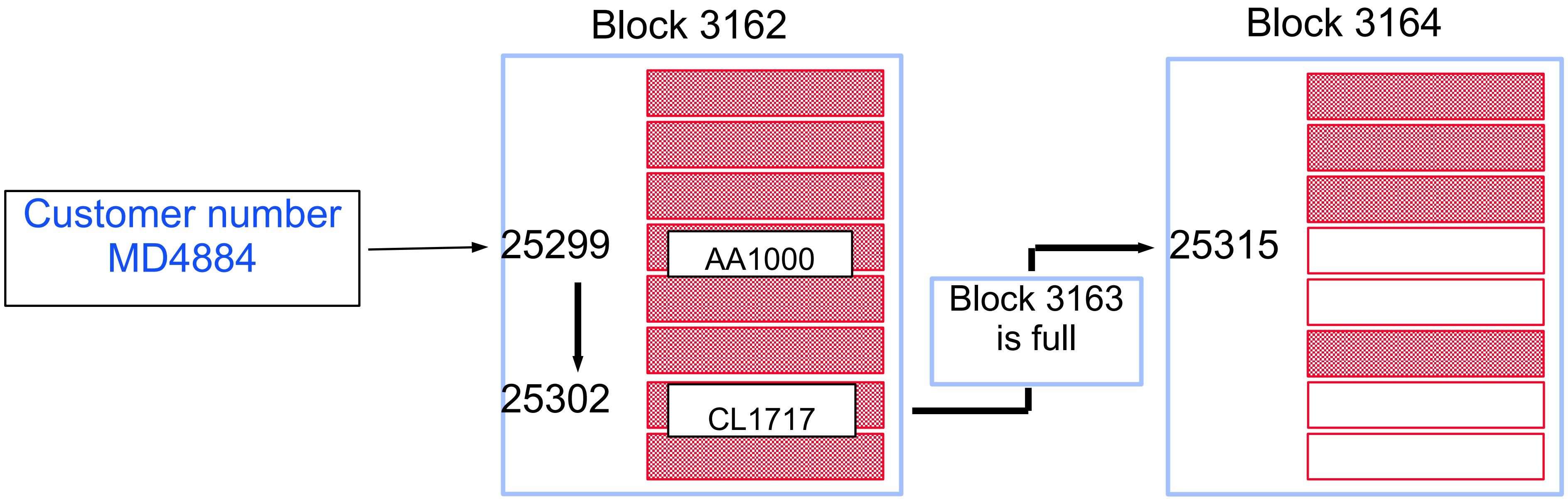
# Hashing algorithm (collision - same block)

- Customer number CL1717 hashes to the same record number as AA1000 location
- TurboIMAGE tries to find an empty location in the same block. If it finds one, no additional I/O is required.
- CL1717 becomes a secondary entry. Primary and secondary entries are linked using pointers that form a chain.

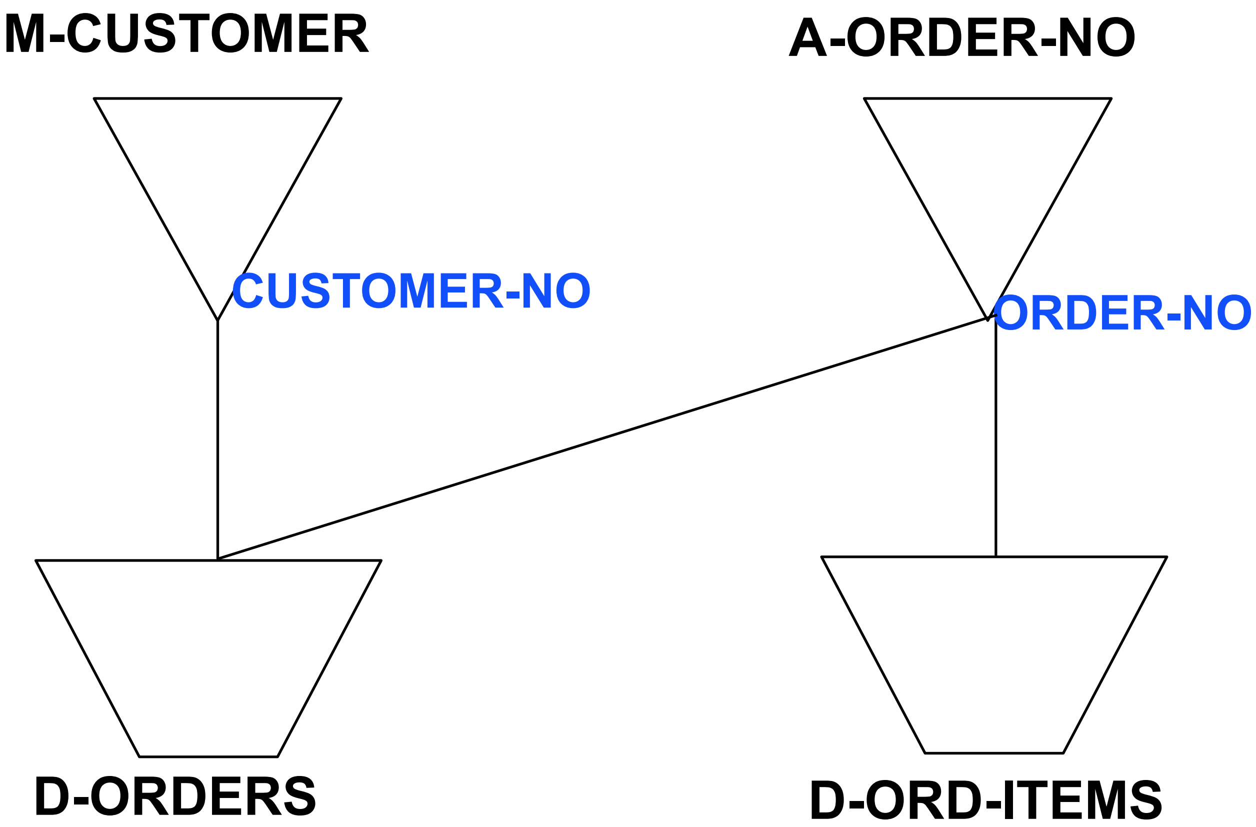


# Hashing algorithm (collision - different block)

- Customer number MD4884 collides with AA1000
- No more room in this block. TurboIMAGE reads the following blocks until it finds a free record location.
- In this case, MD4884 will be placed two blocks down. Now it requires two additional I/Os.



# *An example TurboIMAGE database*



# HowMessy sample report

```
HowMessy/XL (Version 2.2.1)        Data Base: STORE.DATA.INVENT         Run on: MON, JAN 9, 1995, 11:48 AM
  TurboIMAGE/3000 databases         By Robelle Solutions Technology Inc.                             Page: 1

                                                                    Secon-  Max
                          Type                                 Load daries  Blks  Blk
             Data Set            Capacity  Entries          Factor (Highwater)  Fact
 M-Customer       Man     248113     178018    71.7%     30.5% 1496         11
 A-Order-No       Ato    1266783     768556    60.7%     25.7%       1      70
 D-Orders         Det    1000000     768558    76.9%     (    851445)       32
 D-Ord-Items      Det    4000000    3458511    86.5%     (  3470097)        23


                          Max    Ave     Std    Expd     Avg   Ineff   Elong-
          Search Field   Chain  Chain    Dev   Blocks  Blocks   Ptrs    ation
          Customer-No     32    1.92    0.32    1.00    1.90   90.5%    1.90
          Order-No        10    1.35    0.62    1.00    1.00    0.0%    1.00
          !Order-No        1    1.00       0    1.00    1.00    0.0%    1.00
        S Customer-No     80   14.34   17.76    1.75    9.20   57.2%    5.25
        S !Order-No     1604    8.06   35.75    1.36   11.32   72.5%    8.34
```

# HowMessy sample report (master dataset)

```
HowMessy/XL (Version 2.2.1)      Data Base: STORE.DATA.INVENT          Run on: MON, JAN 9, 1995, 11:48 AM
 TurboIMAGE/3000 databases       By Robelle Solutions Technology Inc.                             Page: 1

                                                              Secon- Max
                           Type                          Load daries  Blks  Blk
            Data Set              Capacity  Entries    Factor (Highwater)  Fact
 M-Customer      Man     248113     178018    71.7%    30.5% 1496       11
 A-Order-No      Ato    1266783     768556    60.7%    25.7%      1     70
 D-Orders        Det    1000000     768558    76.9%    (    851445)     32
 D-Ord-Items     Det    4000000    3458511    86.5%    (   3470097)     23


                         Max    Ave      Std    Expd     Avg   Ineff  Elong-
           Search Field Chain  Chain     Dev  Blocks  Blocks    Ptrs    ation
           Customer-No     32   1.92    0.32    1.00    1.90   90.5%    1.90
           Order-No        10   1.35    0.62    1.00    1.00    0.0%    1.00
           !Order-No        1   1.00       0    1.00    1.00    0.0%    1.00
          S Customer-No    80  14.34   17.76    1.75    9.20   57.2%    5.25
          S !Order-No    1604   8.06   35.75    1.36   11.32   72.5%    8.34
```

# *Interpreting master datasets lines*

- Pay attention to the following statistics:
  - High percentage of secondaries (inefficient hashing)
  - High maximum blocks (clustering)
  - High maximum and average chains (inefficient hashing)
  - High inefficient pointers (when secondaries exist)
  - High elongation (when secondaries exist)

# Report on m-customer

- The number of secondaries is not unusually high
- However, there may be problems
  - Records are clustering (high Max Blks)
  - Long synonym chain
  - High percentage of inefficient pointers

| | | | | | Secon- | Max | |
|---|---|---|---|---|---|---|---|
| | Type | | | Load | daries | Blks | Blk |
| Data Set | | Capacity | Entries | Factor | (Highwater) | | Fact |
| M-CUSTOMER | Man | 248113 | 178018 | 71.7% | 30.5% | 1496 | 11 |

| | Max | Ave | Std | Expd | Avg | Ineff | Elong- |
|---|---|---|---|---|---|---|---|
| Search Field | Chain | Chain | Dev | Blocks | Blocks | Ptrs | ation |
| CUSTOMER-NO | 22 | 1.92 | 0.32 | 1.00 | 1.90 | 90.5% | 1.90 |

# Report on a-order-no

- Very tidy dataset
  - Number of secondaries is acceptable
  - Max Blks, Ineff Ptrs and elongation are at the minimum values, even if the maximum chain length is a bit high

| Data Set | Type | Capacity | Entries | Load Factor | Secondaries | Max Blks (Highwater) | Blk Fact |
|---|---|---|---|---|---|---|---|
| A-ORDER-NO | Ato | 1266783 | 768556 | 60.7% | 25.7% | 1 | 70 |

| Search Field | Max Chain | Ave Chain | Std Dev | Expd Blocks | Avg Blocks | Ineff Ptrs | Elongation |
|---|---|---|---|---|---|---|---|
| ORDER-NO | 10 | 1.35 | 0.62 | 1.00 | 1.00 | 0.0% | 1.00 |

# *Master dataset solutions*

- Increase capacity to a higher odd number
- Increase the blocking factor
  - Increase block size
  - Reduce record size
- Change binary keys to type X, U, Z, or P
- Check your database early in the design
- Use HowMessy on test databases

# HowMessy sample report (detail dataset)

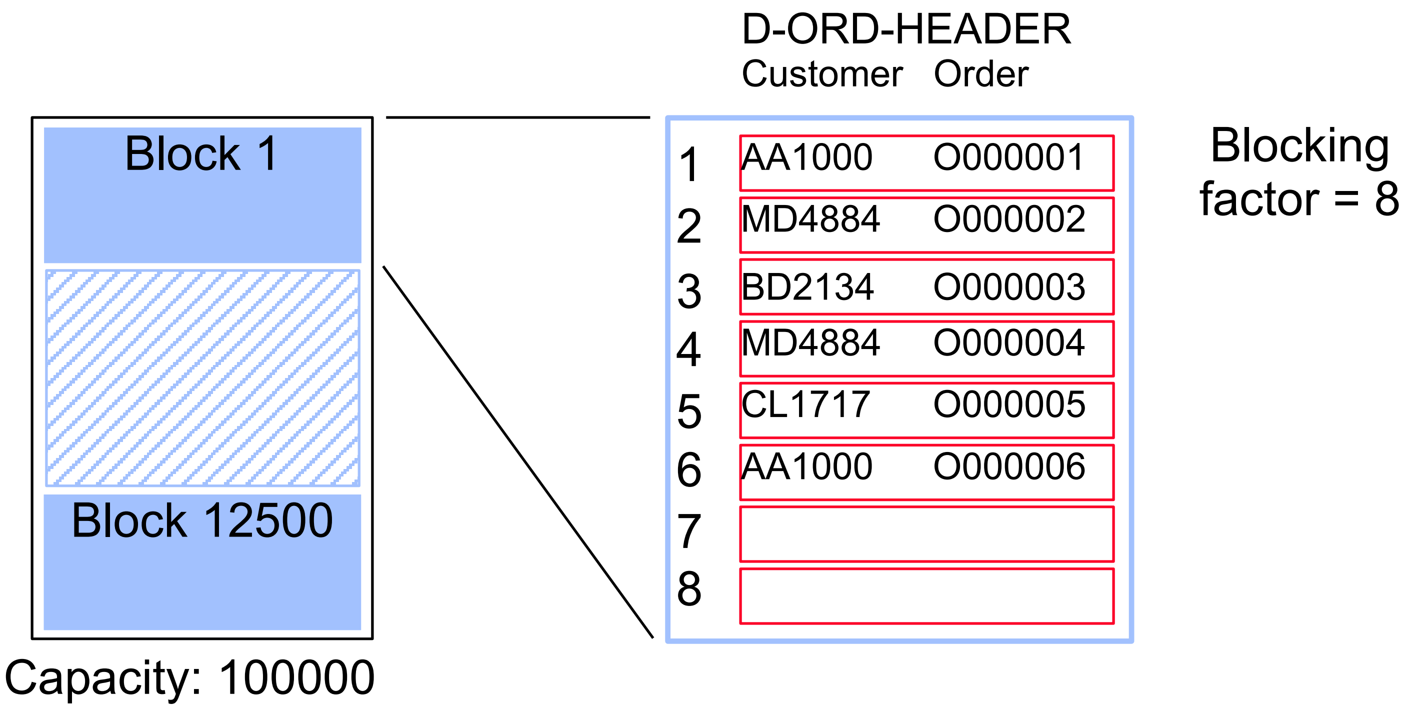
```
HowMessy/XL (Version 2.2.1)        Data Base: STORE.DATA.INVENT         Run on: MON, JAN 9, 1995, 11:48 AM
for TurboIMAGE/3000 databases      By Robelle Solutions Technology Inc.                             Page: 1

                                                          Secon- Max
               Type                              Load     daries Blks    Blk
 Data Set            Capacity    Entries      Factor     (Highwater)     Fact
M-CUSTOMER     Man     248113     178018       71.7%     30.5% 1496         1
A-ORDER-NO     Ato     126673     768556       60.7%     25.7%      1      70
D-ORDERS       Det    1000000     768556       76.9%     (   851445)       12
D-ORD-ITEMS    Det    4000000    3458511       86.5%     (  3470097)       23

                    Max      Ave      Std     Expd      Avg    Ineff   Elong-
 Search Field     Chain    Chain      Dev   Blocks   Blocks     Ptrs    ation
 Customer-No         22     1.92     0.32     1.00     1.90    90.5%     1.90
 Order-No            10     1.35     0.62     1.00     1.00     0.0%     1.00
 !Order-No            1     1.00        0     1.00     1.00     0.0%     1.00
S Customer-No        80    14.34    17.76     1.75     9.20    57.2%     5.25
S !Order-No        1604     8.06    35.75     1.36    11.32    72.5%     8.34
```

# Empty detail dataset

- Records are stored in the order they are created starting from record 1
- Records for the same customer are linked together using pointers to form a chain
- Chains are linked to the corresponding master entry

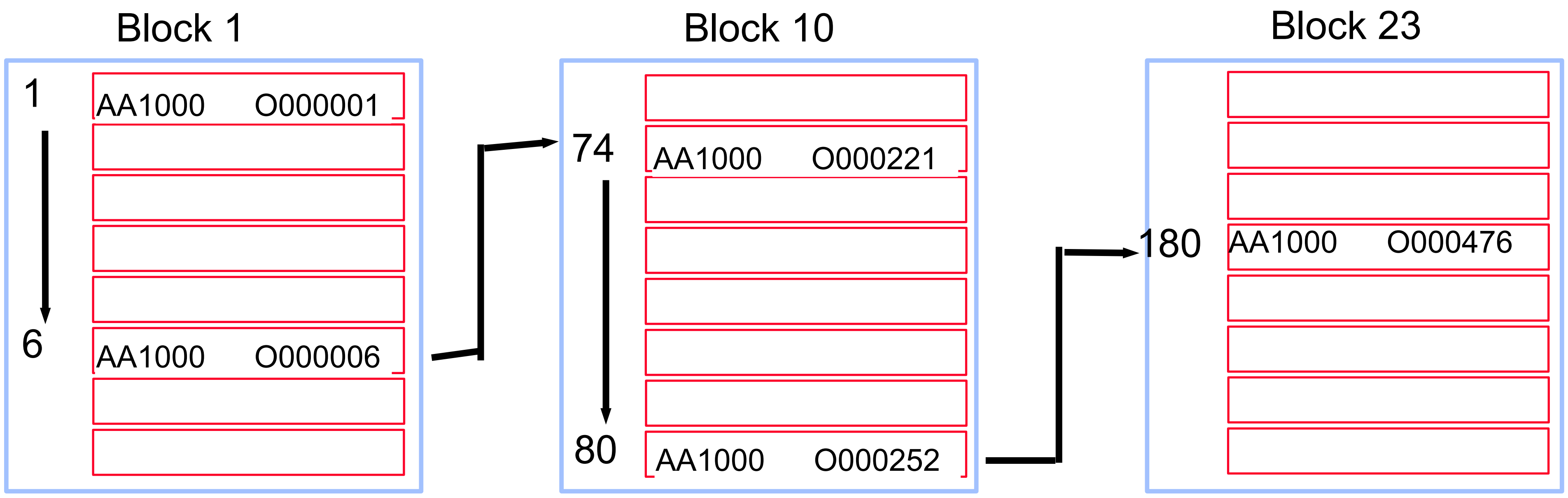


D-ORD-HEADER

| | Customer | Order |
|---|---|---|
| 1 | AA1000 | O000001 |
| 2 | MD4884 | O000002 |
| 3 | BD2134 | O000003 |
| 4 | MD4884 | O000004 |
| 5 | CL1717 | O000005 |
| 6 | AA1000 | O000006 |
| 7 | | |
| 8 | | |

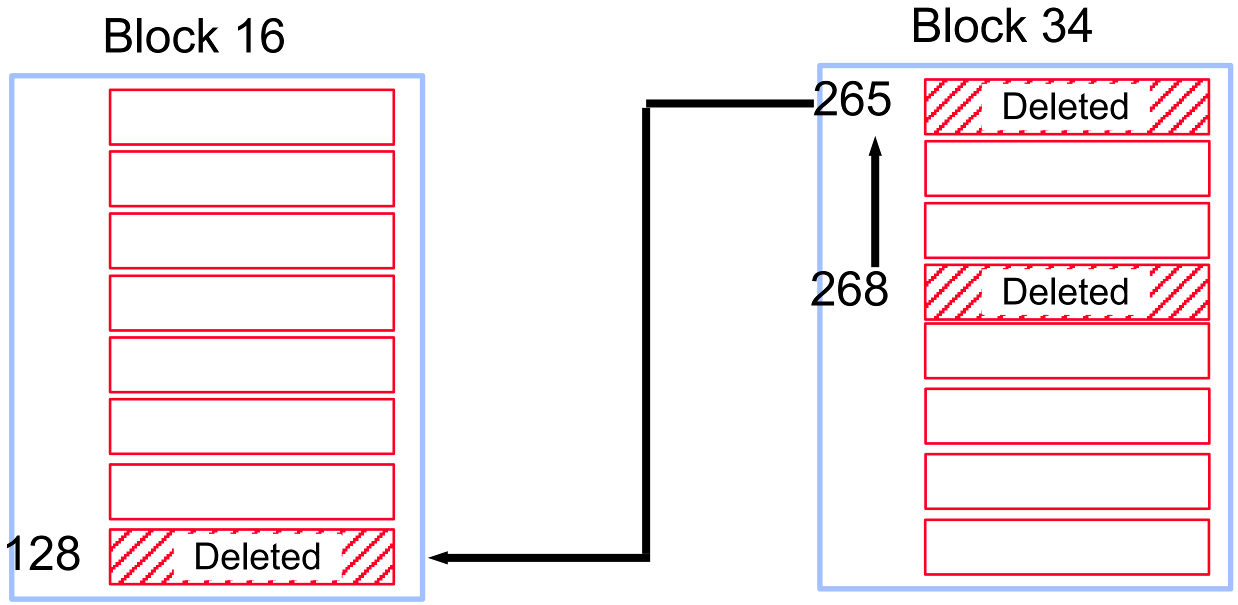
# Detail chains get scattered

- Over time, records for the same customer are scattered over multiple blocks



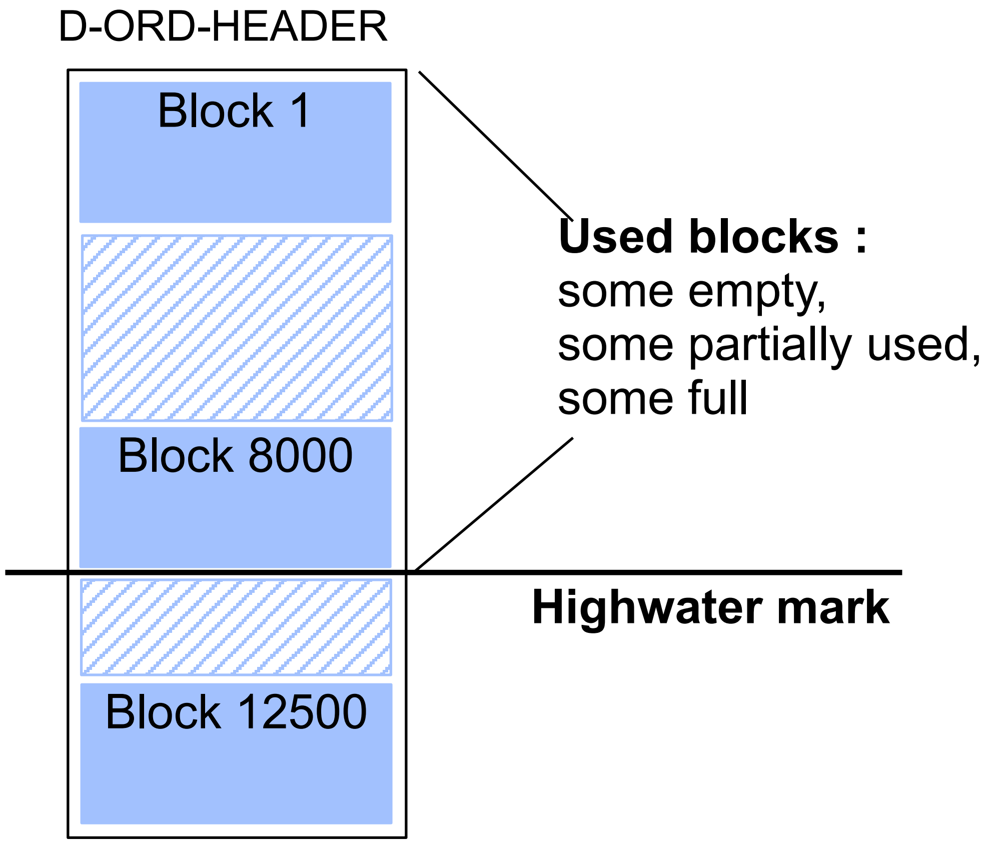
# Delete chain

- Deleted records are linked together
- TurboIMAGE reuses the records in the Delete chain, if there are any



# Highwater mark

- Indicates highest record location used so far
- Serial reads will scan the dataset up to the highwater mark

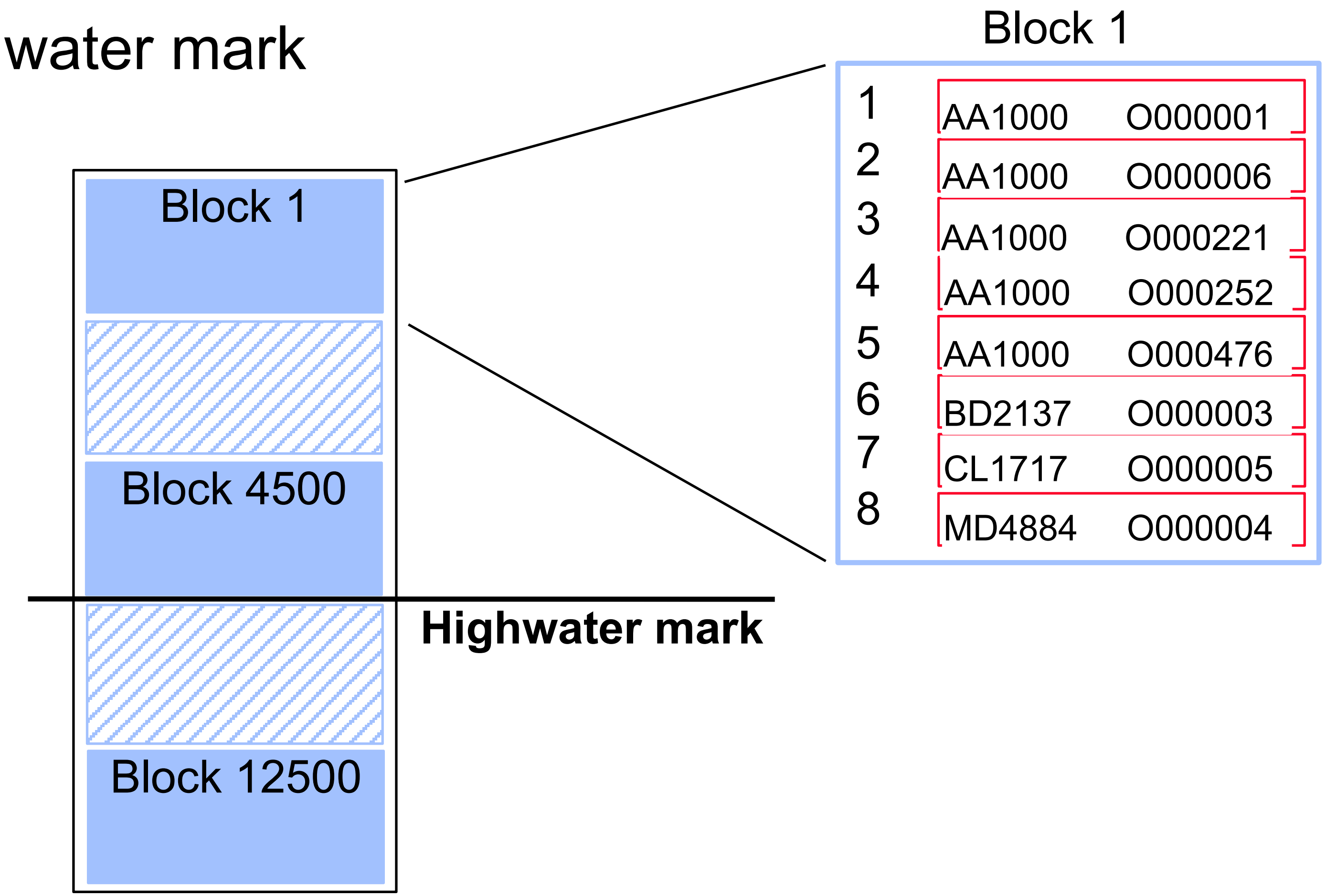


**Used blocks :**
some empty,
some partially used,
some full

**Highwater mark**

# Repacking a detail dataset

- Groups records along primary path
- Removes Delete chain (no holes)
- Resets highwater mark



Block 1

| | | |
|---|---|---|
| 1 | AA1000 | O000001 |
| 2 | AA1000 | O000006 |
| 3 | AA1000 | O000221 |
| 4 | AA1000 | O000252 |
| 5 | AA1000 | O000476 |
| 6 | BD2137 | O000003 |
| 7 | CL1717 | O000005 |
| 8 | MD4884 | O000004 |

# *Interpreting detail dataset lines*

- Pay attention to the following statistics:
    - Load factor approaching 100% (dataset full)
    - Primary path (large average chain and often accessed)
    - High average chain and low standard deviation, especially with a sorted path (Is path really needed ?)
    - High inefficient pointers (entries in chain not consecutive)
    - High elongation (entries in chain not consecutive)

# Report on d-orders

- Primary path should be on customer-no, not on order-no
- Highwater mark is high
- Repack along new primary path regularly

|  |  |  |  |  | Secon-Max |  |
|---|---|---|---|---|---|---|
|  | Type |  |  | Load | daries Blks | Blk |
| Data Set |  | Capacity | Entries | Factor | (Highwater) | Fact |
| D-ORDERS | Det | 1000000 | 768556 | 76.9% | (   851445) | 12 |

|  | Max | Ave | Std | Expd | Avg | Ineff | Elong- |
|---|---|---|---|---|---|---|---|
| Search Field | Chain | Chain | Dev | Blocks | Blocks | Ptrs | ation |
| !ORDER-NO | 1 | 1.00 | 0 | 1.00 | 1.00 | 0.0% | 1.00 |
| S CUSTOMER-NO |  | 80 | 14.34 | 17.76 | 1.75 | 9.20 | 57.2% |
| 5.25 |  |  |  |  |  |  |  |

# Report on d-ord-items

- Inefficient pointers and elongation are high
- Highwater mark is fairly high
- Repack the dataset regularly
- Is the sorted path really needed?

| | | | | | Secon- Max | |
|---|---|---|---|---|---|---|
| | Type | | | Load | daries Blks | Blk |
| Data Set | | Capacity | Entries | Factor | (Highwater) | Fact |
| D-ORD-ITEMS | Det | 4000000 | 3458511 | 86.5% | ( 3470097) | 23 |

| | Max | Ave | Std | Expd | Avg | Ineff | Elong- |
|---|---|---|---|---|---|---|---|
| Search Field | Chain | Chain | Dev | Blocks | Blocks | Ptrs | ation |
| S !ORDER-NO | 1604 | 8.06 | 35.75 | 1.36 | 11.32 | 72.5 | 8.34 |

# *Detail dataset solutions*

- Assign the primary path correctly
  - Search item with average chain length > 1 that is accessed most often
- Repack datasets along the primary path regularly
- Increase the blocking factor
  - Increase block size
  - Reduce record size
- Understand sorted paths
- Check your databases early in the design; use HowMessy on test databases

# *Minimum number of disc I/Os*

| Intrinsic | Disc I/O |
|---|---|
| DBGET | 1 |
| DBFIND | 1 |
| DBBEGIN | 1 |
| DBEND | 1 |
| DBUPDATE | 1 (non-critical item) |
| DBUPDATE | 13 (critical item) |
| DBPUT | 3 [+ (4 x #paths, if detail)] |
| DBDELETE | 2 [+ (4 x #paths, if detail)] |
| Serial reads: | |
| Master | Capacity / Blocking factor |
| Detail | # entries / Blocking factor |

# *Estimating response time*

- Deleting 100,000 records from a detail dataset with two paths would take:
  - 2 + (4 x 2 paths) = 10 I/Os per record
  - 100,000 records x 10 = 1,000,000 I/Os
- Classic: around  25 I/Os per second
  - 1,000,000 I/Os / 25 = 40,000 seconds
  - 40,000 seconds / 3600 = 11.1 hours
- iX: around 40 I/Os per second
  - 1,000,000 I/Os / 40 = 25,000 seconds
  - 25,000 seconds / 3600 = 6.9 hours

# *Automating HowMessy analysis*

- Recent version of HowMessy creates a self-describing file with these statistics

- Process the file with generic tools (Suprtool, AskPlus) or custom programs (COBOL, 4GL), and produce custom reports

- Send messages to database administrators

- Write “smart” job to fix databases without user intervention

# Processing Loadfile with Suprtool

- Datasets more than 80% full

```
        >input  loadfile
        >if loadfactor > 80
    >ext database, dataset, datasettype, loadfactor
    >list standard
```

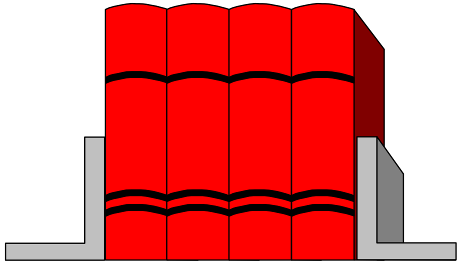
- Only one address per customer

```
        >input  loadfile
    >if  dataset = "D-ADDRESSES" and &
       maxchain > 1
```

# References

- The TurboIMAGE/3000 Handbook (Chapter 23)
- Available for $ 49.95 from:

WORDWARE<br>
P.O. Box 14300<br>
Seattle, WA 98114



# Summary

- TurboIMAGE databases become messy over time, especially if they are active
- HowMessy and DBLOADNG let you analyze the database’s efficiency
- You should have some knowledge of the internal workings of TurboIMAGE
- Monitor your databases regularly

# Exercise #1

| Data Set | Type | Capacity | Entries | Load Factor | Secondaries (Highwater) | Max Blks | Blk Fact |
|---|---|---|---|---|---|---|---|
| A-MASTER | Ato | 14505679 | 9709758 | 66.9% | 36.8% | 2395 | 29 |

| Search Field | Max Chain | Ave Chain | Std Dev | Expd Blocks | Avg Blocks | Ineff Ptrs | Elongation |
|---|---|---|---|---|---|---|---|
| MASTER-KEY | 37 | 1.58 | 1.26 | 1.00 | 1.88 | 48.5% | 1.88 |

# Exercise #2

```
                                                          Secon-Max
                   Type                           Load    daries Blks   Blk
Data Set                Capacity    Entries     Factor    (Highwater)   Fact
D-ITEMS             Det   620571     119213      19.2%    (    242025)     7

                          Max      Ave      Std      Expd      Avg      Ineff    Elong-
    Search Field         Chain    Chain     Dev     Blocks    Blocks    Ptrs     ation
   S !               ITEM-NO        3      1.00      0.02      1.00      1.00     0.0%   1.00
   S            SUPPLIER-NO        23      8.07      3.25      1.77      3.30    28.4%   1.86
    LOCATION              5938    11.62    63.64      2.24      2.53    13.2%     1.13
    BO-STATUS        9999999999.99          0.00 17031.0017047.00     14.3%     1.00
    DISCOUNT             99999   120.181337.15       3.73     39.37    31.9%    10.55
```