

## Range Checking in R-FETCH Routines

The GX version of the Window Manager has been extended to allow ranges of records to be displayed. In previous versions it was only possible to restrict records using the Dependant clause. While this generally works well for character fields, it does not allow a range of records to be selected when the discriminating key is computational (i.e. when required range boundaries do not happen to fall on a character boundary).

This enhancement overcomes the limitation by allowing you to specify a key range using the R-FETCH routine.

The R-FETCH routine previously only allowed records to be rejected from the displayed record set. An EXIT with 1 caused (and continues to cause) the current record to be ignored for window processing purposes. While it is possible to use this technique to ignore large numbers of records (thus displaying only a given range) this technique can be extremely inefficient. The R-FETCH routine has therefore been extended with 3 new EXIT conditions to allow the appropriate position to be reached during window processing.

To process a given range of keys, the R-FETCH routine tests whether the supplied record (i.e. the record currently in the I/O channel) is within the required range. If in range, all is well and no further action is required. If not, the program repositions the I/O channel at the first record or last record within the permitted range.

Where the current key value precedes the required range, the program reads the first record in the required range, and returns Exception 100. This indicates to Window Manager that the key value has been ADVANCED (in terms of ASCII Index Value) to a further point in the index.

Where the key value exceeds the required range, the program reads the last record in the required range and this time returns Exception 101. This indicates to Window Manager that the ASCII key value has been SET BACK to a prior position in the index.

When there are no records stored on the database within the required range, the program returns Exception 102 in the R-FETCH routine.

The following example demonstrates the coding technique. Using the DEMO database TR record, we want a Window which displays TR records in the range TR04 and TR08.

```
R-FETCH.
*
  IF TRTRNO < "TR04"
    FETCH FIRST TRTRN KEY "TR04"
    IGNORE EXCEPTION
    IF TRTRNO > "TR08" EXIT WITH 102
    EXIT WITH 100
  END
*
  IF TRTRNO > "TR08"
    FETCH LAST TRTRN KEY "TR08"
    IGNORE EXCEPTION
    IF TRTRNO < "TR04" EXIT WITH 102
    EXIT WITH 101
  END
EXIT
```

\* BEFORE START OF RANGE  
\* RETURN RECORD > = TR04 OR EOF  
\* SOF CONDITION CAUGHT BY NEXT LINE  
\* NO RECORDS IN RANGE  
\* KEY VALUE ADVANCED EXCEPTION

\* AFTER END OF RANGE  
\* GET THE LAST IN RANGE OR SOF  
\* SOF CONDITION CAUGHT BY NEXT LINE  
\* NO RECORDS IN RANGE  
\* KEY VALUE RETARDED EXCEPTION

This technique is generally vastly more efficient than simply rejecting out-of-range records on a one-by-one basis. However, it can only be used when the enquiry index is in the same order as the range you want to select. If it is not, the required value range is spread all over the index, meaning that there will be no start or end point you can position on.

A window written using this technique must therefore be restricted to allow enquiry only using the relevant index(es). In the above example, it is necessary to restrict the window to the TRTRN index only, or alternatively to disable the selection code when that index is not the currently selected enquiry index.

**Important Note:** This feature operates only on GX, and will return unpredictable results when run in non-GX mode.

## **Version Requirements**

GSM V8.1 SP-14