

B\$GTIO – Return Sanitized I/O Control Block

The B\$GTIO routine is available to return a sanitized version of the internal Speedbase I/O control block for a specified record.

1. Invocation

To obtain a sanitized version of the internal Speedbase I/O control block code:

```
CALL B$GTIO USING $record r1
```

where *\$record* is a valid Speedbase record-id prefixed with the \$ symbol. For example \$AD, \$SRC etc.; and *r1* is a control block of the following format:

```
01      R1
03      R1DBID      PIC X(8)          * DATABASE ID
03      R1DGEN      PIC 9(4) COMP     * DATABASE GENERATION #
03      R1RCID      PIC X(2)          * RECORD ID.
03      R1FRID      PIC X(20)         * LONG RECORD NAME (ONLY 6 SUPPL
03      R1RCNO      PIC 9(2) COMP     * RECORD NUMBER IN DICTIONARY
                                           * OR -1 FOR A DBX DATABASE
03      R1NOPR      PIC 9(2) COMP     * NO PRIMARY KEY IF = 1
03      R1DRLN      PIC 9(4) COMP     * DATA RECORD LENGTH (TOTAL)
03      R1XIXE      PIC 9(2) COMP     * NUMBER OF INDEX ENTRIES (RX)
03      R1XMNO      PIC 9(2) COMP     * NUMBER OF MASTR ENTRIES (RM)
03      R1XGFN      PIC 9(2) COMP     * NUMBER OF GVF ENTRIES (RG)
03      R1DLRN      PIC 9(9) COMP     * LAST DATA RECORD # ACCESSED
03      R1IXRQ      PIC 9(2) COMP     * LAST INDEX NUMBER ACCESSED
03      R1DSTR      PIC 9(9) COMP     * LOCK ADDRESS CODE.
03      R1LOCK      PIC S9 COMP       * REC LOCKED? 0=NO, 1=PROT, 2=EX
03      R1P-RA      PIC PTR           * PTR TO RECORD AREA
03      R1XRCN      PIC 9(4) COMP     * DBX RECORD NUMBER (IF R1RCNO = -1)
03      R1FILL      PIC X(24)         * FILLER TO 80 BYTES TOTAL
                                           VALUE LOW-VALUES
```

For GSM SP-38, or later, R1FILL is redefined as follows:

```
01      FILLER REDEFINES R1FILL
02      R1FLAG      PIC 9 COMP        * 0 = NO EXTRA FIELDS FOLLOW
                                           * 1 = EXTRA FIELDS FOLLOW
02      R1DULN      PIC 9(4) COMP     * USER RECORD LENGTH (EXCLUDES GVAS)
02      R1PKEY      PIC 9(2) COMP     * LENGTH OF PRIMARY KEY, OR 0 IF NONE
02      R1XSEQ      PIC 9(2) COMP     * INDEX NUMBER OF $SEQ KEY (OR 0 IF NONE)
02      R1SQLLEN    PIC 9(2) COMP     * KEY LENGTH OF THE $SEQ KEY (0 IF NONE)
```

2. STOP Codes and Exception Conditions

No STOP codes are generated by B\$GTIO.

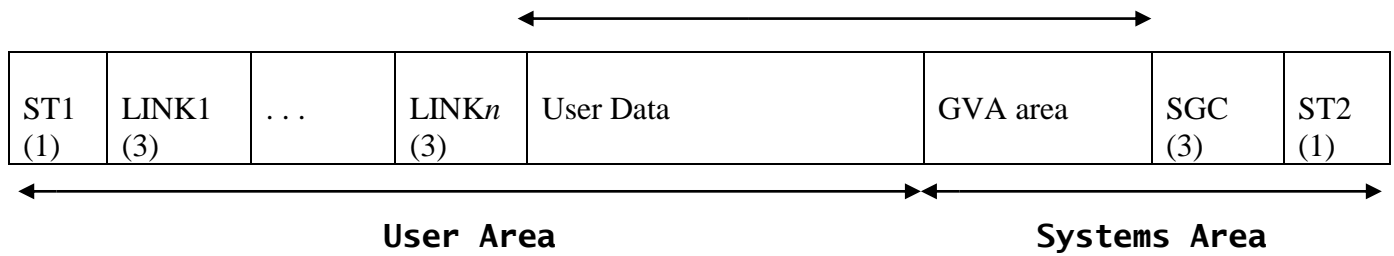
No exceptions are returned by B\$GTIO.

3. Programming Notes

The most common cause of a problem with the routine is passing the record name without the \$ symbol (e.g. OH instead of \$OH).

One of the uses of the B\$GTIO routine is to determine the total length of a record (as returned in R1DRLN); and thus the length of the data portion of the record. For non-DBX databases the format of the record is as follows:

Specified fields



ST1: Status Code 1 (\$rtST1)

All data records begin with a PIC 9(2) COMP status code. This is either positive to indicate the record is in use, zero to indicate the record has never been used, or negative to indicate the record was once in use, but has since been deleted and returned to the list of free records. The status code holds the number of the minor backup cycle during which the record was last modified, and is used to control incremental back-ups. The data-name of this field is \$rtST1, where *rt* is the two-character record ID specified in the ACCESS statement. This field may be examined by application frames but **must not be modified**.

Link n : Link to Masters (\$rtLNK(n))

The Relative Record Number (RRN) of each linked master record is stored on the servant data record. One 9(6) COMP (3 byte) RRN exists on the data record for each master record. For example, a record associated with three masters will therefore contain three 9(6) COMP RRN links, meaning a storage overhead of nine bytes per record.

Specified Fields

User-specified fields, as defined using the Speedbase dictionary maintenance utility, follow any master links. The size of this part of the record area is determined directly from the picture clauses assigned to each field. An 01 group level field is automatically generated by the compiler which includes all these data items, including GVA's. The name of this group field is the same as the record ID. This field allows redefinitions of the specified fields from within the Data Division, should this be necessary.

SGC: Sub Group Count (\$rtSGC)

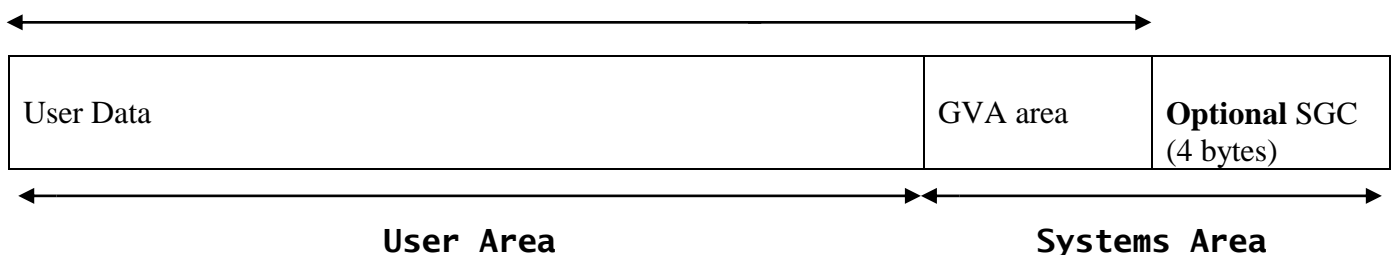
This 9(6) COMP field stores a count of the attached servant records (i.e. the number of records to which it acts as master). When this count is non-zero, the record may not be deleted. Any attempt to do so will cause an exception condition to be returned. This count cannot be modified by an application frame.

ST2: Status Code 2 (\$rtST2)

This PIC 9(2) COMP status code indicates the status of the system part of the data record. The code is either zero to indicate no changes have ever taken place to the system area, or positive to indicate the back-up cycle during which the system area was last updated. Status code value is undefined if the record is deleted (i.e. ST1 is negative).

For DBX databases the format of the record is as follows:

Specified fields



Note that both of the status fields (\$rtST1 and \$rtST2), and the Uplink Fields (\$rtULK()) are absent on DBX databases. **It would appear that the optional Sub-Group Count field is only present on those records that are linked to Servant records. Note also the Sub-Group Count field, \$rtSGC, has been expanded from a 3-byte PIC 9(6) COMP field to a 4-byte PIC 9(9) COMP field for DBX.**

Calculating the start offset and length of the User Data for non-DBX databases

The offset of the User Data/GVA Area of a non-DBX Speedbase record can be calculated by multiplying the value of R1XMNO (i.e. the number of master records) by 3 (i.e. the size of a PIC 9(6) COMP field) and adding 1 to allow for the PIC 9(2) COMP ST1 status field.

The length of the User Data/GVA Area of a non-DBX Speedbase can be derived from the value of R1DRLN returned by B\$GTIO by firstly subtracting the offset length (as explained above); then adjusting further by subtracting 4 (i.e. to allow for the 3 byte SGC Sub-Group Count field and the 1 byte ST2 status field).

Calculating the start offset and length of the User Data for DBX databases

There are no Speedbase status fields before start of the User Data/GVA Area of a DBX Speedbase record so no offset calculation is required.

For a record with no linked Servant records the length of the User Data/GVA Area of a DBX Speedbase is the value of R1DRLN returned by the B\$GTIO sub-routine.

For a record with one, or more, linked Servants records the length of the User Data/GVA Area of a DBX Speedbase can be derived from the value of R1DRLN returned by the B\$GTIO sub-routine by simply subtracting 4 (i.e. to allow for the 4 byte SGC Sub-Group Count field).

4. Examples

[EXAMPLE REQUIRED]

5. Copy-Books

See copy-book "R\$" in copy-library S.SYS32. Note that this copy-book **MUST** be expanded using a SUBSTITUTING clause. For example:

```
COPY "R$" USING "R1"
```

6. See Also

None.