**SYNTAX ELEMENTS**

**Element Description**

1. Verb - One of the verbs like Commands detailed later. Most Commands will accept any or all of the optional clauses
2. File modifier file modifiers DICT, ONLY=, WITHIN and TAPE modify the use of the file, and how it is accessed
3. File specifier Identifies the main data file to be processed. Usually the data section of a file, but could be a dictionary or a secondary data area.
4. Record-list - Defines which records will be eligible for processing. Comprises an explicit list of record keys or record selection clauses. An explicit list comprises one or more record keys enclosed in single or double quotes. A selection clause uses value strings enclosed in single or double quotes and has at least one relational operator. If no record list is supplied, all records in the file will be eligible for processing unless an “implicit”

record list is provided by preceding the Command with a selection Command such as GET-LIST or SELECT.

1. FROM list# - A number from 0 through 10 of an active select list that contains record IDs. The query operates on records whose IDs are in the select list.
2. Selection - Criteria Qualify the records to be processed. Comprises a selection connective (WITH or IF) followed by a field name. Field names can be followed by relational operators and value strings enclosed in double quotes. Logical Connectives AND/OR ca also be used. Expressions can be

parenthesized to specify precedence.

1. Sort-criteria - Specify the order in which the data is returned. Comprises a sort modifier, such as BY or BY-DSND, followed by a field name. Used

also to “explode” a report by sorting lines corresponding to multivalued

fields by value, and to limit the output of values (see output specification).

1. USING file specifier - Defines an alternate file for use as the dictionary.
2. Macro call - JQL allows the use of macros to predefine parts of a sentence. The macro definition contains one or more optional sentence elements. You

invoke the macro by including its name in a sentence. The jQL processor looks for the macro in the currently active dictionary and includes all of its text parts in the sentence.

1. Output-specification - Comprises the names of the fields to be included in the report, optionally preceded by a BREAK-ON connective or ‘TOTAL’ connective. Print limiters (Values strings enclosed in double quotes after the field name, optionally preceded by relational operators) can be used to restrict multivalue output
2. Format specification - Comprise modifiers, such as HEADING, ID-SUPP, and DBL-SPC, which define the overall format of the report.
3. Output-limiter - The WHEN clause, used to limit the output of multivalued fields.
4. Report-qualifiers - Special keywords used in formatting reports.

SORT - Lists selected records in sorted order

SORT-ITEM - Displays full listings of selected records in sorted order

SORT-LABEL - Displays items in a format suitable or mailing labels and other block listings

SREFORMAT - Redirects jQL output to a file or to a tape with records sorted by sort expression

SSELECT - Creates a sorted list of records that meet specified selection criteria

STAT - Displays numeric statistics for fields in a file

SUM - Adds numeric values in fields of records that meet specified selection criteria

T-DUMP - Copies records from disk to tape

T-LOAD - Copies records from tape to disk

jQL Verbs

**Verb Description**

BSELECT - Retrieves selected records and generates a list composed of data fields from those records as specified by any explicit or default output specifications. Each subvalue within a field becomes a separate entry within the list.

COUNT - Counts the records in a file

ESEARCH - Similar to SEARCH

LIST - Displays data from records in a file

LIST-ITEM - Displays full listing of selected records

LIST-LABEL - Displays records in a format suitable for mailing labels and other block listings

REFORMAT - Redirects jQL output to a file or tape.

SEARCH - Creates a select list of records that contain an occurrence of one or more specified strings

SELECT - Creates a list of records that meet specified selection criteria

SORT - Lists selected records in sorted order

SORT-ITEM - Displays full listings of selected records in sorted order

SORT-LABEL - Displays items in a format suitable or mailing labels and other block listings

SREFORMAT - Redirects jQL output to a file or to a tape with records sorted by sort expression

SSELECT - Creates a sorted list of records that meet specified selection criteria

STAT - Displays numeric statistics for fields in a file

SUM - Adds numeric values in fields of records that meet specified selection criteria

T-DUMP - Copies records from disk to tape

T-LOAD - Copies records from tape to disk

File Modifiers

As described below file modifiers DICT, ONLY=, WITHIN and TAPE modifies the use of the

file, and how it is accessed

**SYNTAX ELEMENTS**

{DICT} {ONLY} {WITHIN} {TAPE} filename{,data-section-name}

DICT - Specifies the dictionary section of the file and contains the

data for referencing. You must type the modifier DICT

before the filename. When modifying a filename by the

DICT the processor looks in the MD for attribute and macro

definition items.

ONLY Specifies that only item-ids are to be output and suppress

any default output contents. You can type the modifier

ONLY before filename or following all clauses, which

contain attribute names.

WITHIN Specifies a sublist such as bill of material items. Use

WITHIN only with the LIST and COUNT verbs and must

precede filename. Specify one item-id only; if you enter

more than one item-id, it displays an error message.

TAPE Tells the processor to retrieve data from a magnetic tape,

which written only in a T-DUMP format. This modifier

cannot be used with the sorting verbs such as SORT and

ST-DUMP, nor with tape output verbs, such as T-DUMP,

nor with the updating verb EDELETE

filename Specifies a dictionary section and a data section

data-sectionname

Specifies a data section other than the data section called

filename. It must follow filename and use a comma with no

spaces for separation.

**Implicit Item-List**

**EXAMPLE**

The following sentences will not list anything because the value strings cannot match any item-id

in the implicit list.

SELECT ORDER GT ‘200’

42 RECORDS LISTED

>LIST CUSTOMER = ‘40823’ ‘40825’

**Selection Criteria**

**SYNTAX**

WITH | IF { NOT } { EACH } field {value-selection clause} {{AND | OR}

{ WITH |IF } {NOT} {EACH} field {value-election clause}...}

Value selection clause has the form:

{relational-operator} “value string” {{logical-connective}

{relational operator} “value string”}. . .

BREAK-ON

**EXAMPLE 1**

SORT SALES P.CODE S.CODE =”ABCORDER ORD.ID ORD.QTY = “5””

Selects all the records in the ORDER file and outputs the ORD.ID data. The ORD.QTY data will

only be included if it matched 5 - any other value will be shown as blank.

**EXAMPLE 2**

SORT ORDER BY ORD.QTY BREAK-ON ORD.QTY ORD.ID

Selects all the records in the SALESORDER file in ORD.QTY order and outputs a line for each

record until the ORD.QTY changes. At this point, a control break triggers and outputs the

running total of ORD.QTY. At the end of the report, it displays a cumulative total for ORD.ID.

Options can be one or more of the following:

B Suppress initial line-feed.

C{n} Display running counters of the number of records selected and records

processed. Unless modified by n, the counter increments after every 500 records

processed or the total number of records if less than 500. The n specifies a number

other than 500 by which to increment. For Example, (C25) increments the

counter after every 25 records processed.

P Display running counters of the number of records selected and records

processed. Unless modified by n, the counter increments after every 500 records

processed or the total number of records if less than 500. The n specifies a number

other than 500 by which to increment. For Example, (C25) increments the counter

after every 25 records processed.Send the report to the printer.