Program Information

ENQWATCH

(Version 2.2.01)

Disclaimer

This computer program and associated materials has been developed by:

Kevin E. Ferguson,
Lead System Programmer
American National Insurance Company.

This material has been used successfully by **American National Insurance Company** and to the best of our knowledge this material and any system(s) which it is a part of are operational as of the service level or date stated in the body of this material (if so stated). However, ***no warranty*** is given or implied as to the accuracy of this material or any related material or systems, and ***no responsibility*** is assumed for any effect or modification directly or indirectly caused by the use of this material.

It is the responsibility of any user of this material to evaluate its usefulness to the user's environment.

American National Insurance Company does not guarantee to keep this or any related material current, nor does it guarantee to provide any corrections or extensions described by any users of this material or any corrections or extensions made in the future by **American National Insurance Company** itself.

Acknowledgements

This document refers to number of software products that are produced by other companies. In most cases the names of these products are trademarks and/or copyright of those companies. It is not our intention to claim either the name of the trademark, nor the product itself, these remain solely the right of the owning companies.

Table of Contents

1 Program Overview

The program, **ENQWATCH**, is based around the IBM supplied program ISGECMON with the features of the David Alcock's ISGECMOM added and massively re-written and some 'bells and whistles' added.

ENQWATCH will only work for dataset conflicts between batch jobs and TSO users. Conflicts between two or more batch jobs are not handled by this program.

It is designed to be used as a long running program, in fact it won't end at all until you issue a stop for, or cancel, it.

The purpose of the program is the monitor the system for any dataset contention and if the holder of the resource is a TSO user, it will send them a message requesting that they free the resource.

The added features of **ENQWATCH** include:

- The ability to force an iteration of the loop thereby allowing automation to be written to immediately send a message to a TSO user if they are holding up an important batch run.
- More statistics are available including how many times the iteration was forced by the above feature.
- The program has been modified to allow commands to take effect immediately rather than wait until the loop time limit is met.
- Dynamic adjustment of the wait time limit is now supported.
- The ability to cut SMF records. These will be cut even if the maximum number of notifies has been sent.

1.1 Library Information

The source code for **ENQWATCH** can be found in SPD2.PROD.SOURCE

The non-reentrant, APF authorized load module resides in SPD3.LINKLIB

ENQWATCH runs authorized because for some reason (better known to IBM) when a STIMER exit is used within a program, the program has to be authorized. It sure beats the heck out of me as to why, but there you go!

Passed Parameters: an optional wait time can be passed to the program. It must be between one and four numbers long. If omitted a default value will be used (60 seconds).

Called Programs: None **(Indirectly calls)** None

IBM macros used:

CVT	IEZCOM	LTORG	STORAGE
EXTRACT	IHAASCB	MODESET	TIME
GQSCAN	IRAOUCB	QEDIT	TPUT
IEZCIB	ISGRIB	SMFWTM	WAIT
	LOCASCB	STIMER	WTO

User macros used: None

User data mapping macros used: None

1.2 Assembled Program Values

There are four values that currently are only set when the program is assembled. If you need to change any of the the first three values, then the program will only pick them up after a restart. Only WAITTIME can be changed dynamically. These values are located at the top part of the program and they are:

1.2.1 MAXMSGS

This is the maximum number of 'Get the heck out of the dataset' messages that will be sent to single individual TSO users for any given ENQ conflict. The current value for this is 5. This value will not effect the SMF record being produced.

1.2.2 MAXOWNER

This is the value for the maximum number of ENQ conflicts that can be tracked. This value is currently set at 50. Note: only conflicts between TSO users and batch jobs are included.

1.2.3 AREASIZE

This value specifies the size of the area to be used to hold the RIBS and RIBES and should only be changed by a system programmer. This value is multiplied by 1000 in the program. The current value for AREASIZE is 10

1.2.4 SMFON

This is a flag that denotes whether SMF records are cut when an ENQ notify is issued. 1 means cut SMF records, any other value and no SMF records are cut.

The value can be changed dynamically using the SMF command.

1.2.5 SMFREC

This is the numeric value of the SMF record type to be cut if SMF recording is switched on. The value must be between 128 and 255.

Please be sure to check with your site SMF administrators before setting this value to ascertain which SMF record number to use. Some third party vendors produce SMF records in the range 128-255 notably Syncsort which cuts SMF type 208 records by default.

The default value for the SMF record number is 223.

1.2.6 WAITTIME

This is the value, in seconds, which the program will wait for between normal iterations of the loop cycle. The current value for WAITTIME is 60 seconds.

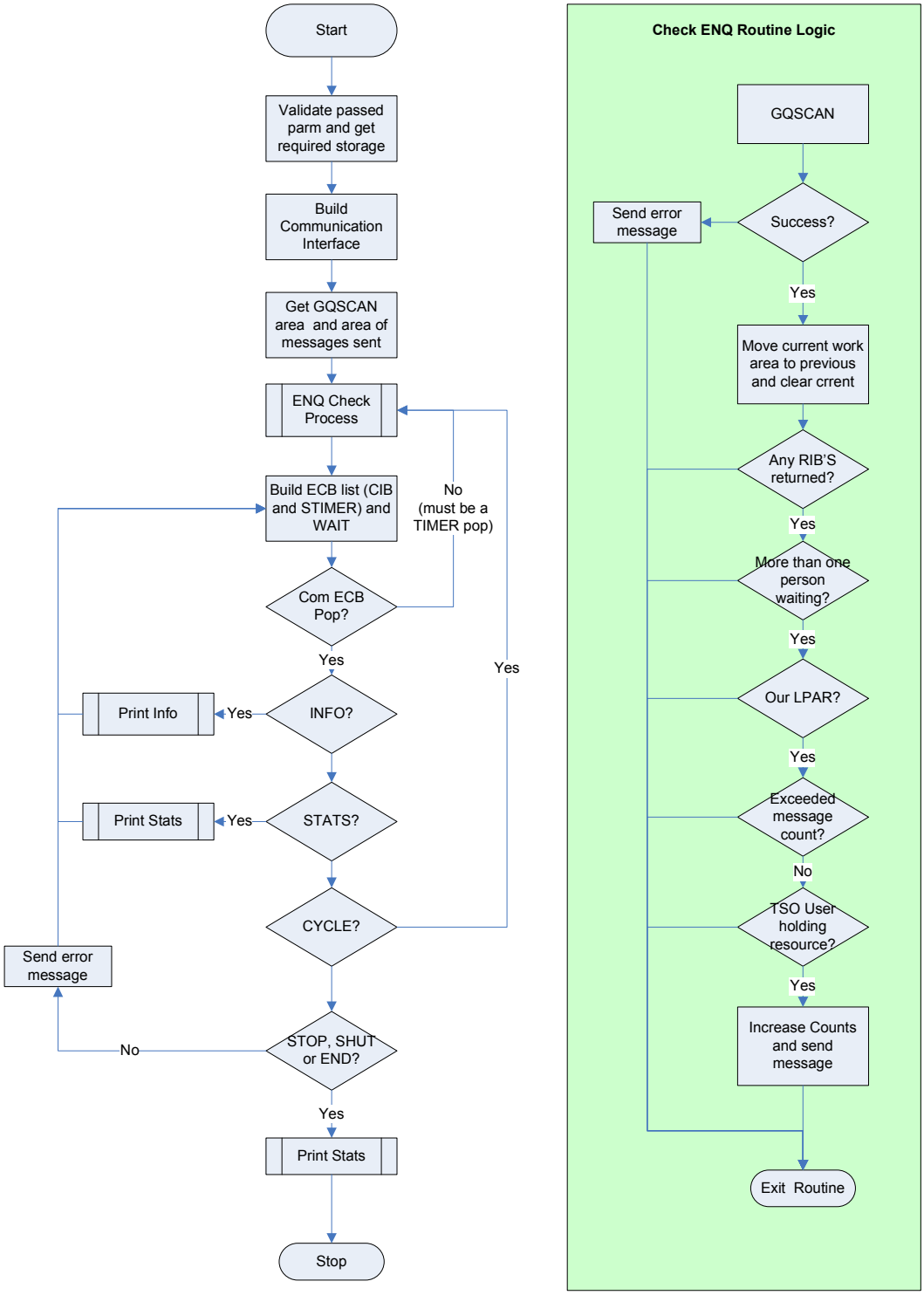
Do not exceed 9999 seconds or the program may fail.

The value for WAITTIME can be changed dynamically while the program is running, but obviously if a permanent change is desired this value needs to be changed, the program re-assembled and then restarted.

The current settings for these values can be displayed using the INFO command (F sysname,INFO)

1.3 Program Logic Flowchart

ENQWATCH Program Logic Flowchart



Linkage Requirements

ENQWATCH should be linked as AMODE 31, RMODE 24 (astute system programmers will know why ☺). It should also be authorized.

1.4 Required JCL:

```
1     //Jobname JOB ……..
2     //stepname      EXEC PGM=ENQWATCH
3     //STEPLIB DD    DISP=SHR,DSN=your.loadlib
```

JCL card	Required	Use
1	No	JOBCARD
2	Yes	EXEC card
3	Yes	This is the library where **ENQWATCH** resides The STEPLIB concatenation can only be removed if all the modules reside in LNKLST libraries.

1.5 Controlling ENQWATCH

ENQWATCH needs has an optional parameter that can be passed to change the WAIT time between loops at startup. By default the wait time is set to 60 seconds in the program.

The value of the wait time can be changed dynamically by using the WAIT command.

There are no control cards needed by **ENQWATCH**

1.6 Supported Operator Commands

Operator commands can be issued to **ENQWATCH** via the standard operating system modify command format i.e. (F stcname,command).

1.6.1 CYCLE

The CYCLE command will force **ENQWATCH** to do a GQSCAN immediately. The main use for this is to allow automation to issue the CYCLE command when a IEF099I JOB xxxxxxxx WAITING FOR DATA SETS message is issued.

This facility can be used, in conjunction with a long wait time, to limit the use of **ENQWATCH** to only loop when a production job needs a dataset that a TSO user is holding. It should be noted though that under these conditions **ENQWATCH** will still perform a normal GQSCAN and this will report on all datasets enqueues not just those of the production jobs.

1.6.2 INFO

The INFO command will result in **ENQWATCH** issuing the ENQW007I messages which will display information about the values that are currently in effect for the program.

The values returned are:

Maxmsgs=	This gives you the value of the maximum number TPUTS that will be sent to any single TSO user per dataset contention detected. This value can only be changed by changing it in the program itself and re-assembling **ENQWATCH**. The default is 5.
Maxowner=	This gives the current value for the maximum number of dataset contentions that can be handled at one time. This value can only be changed by changing it

in the program itself and re-assembling **ENQWATCH**. The default is 50.

Areasize= This gives the current value of the area available to GQSCAN to use to hold the returned RIBs and RIBE's. This value can only be changed by changing it in the program itself and re-assembling **ENQWATCH**. The default is 10,000 bytes.

Wait time= This gives the current value for the wait time being used by **ENQWATCH**. This value can be changed using the WAIT operator command. The message will also indicate how the value in use was set (parameter, program default or operator command). The default is 60 seconds.

SMF Info A status will be given for SMF recording (OFF or ON). Additionally it will tell you what the SMF record type number is.

1.6.3 SMF ON | SMF OFF

This command can be used to switch SMF record production off or on as required. The default for SMF recording can be set inside the program. The default for the SMF record number is 233 and this too can be set inside the program.

There must be a space or an equal sign following the word 'SMF'.

The only values that are accepted by the program are ON or OFF.

1.6.4 STATS (or STATUS)

The STATS (or STATUS) command will result in **ENQWATCH** issuing the ENQW008I messages which will display statistics about what the current execution of the program has done

The values returned are:

Notify= This gives a count of how many times **ENQWATCH** sent a message about a dataset contention to a TSO user. This a combined, accumulative count for all TSO users and all datasets contentions.

Cycles= This gives a count of how many CYCLE operator commands were issued to **ENQWATCH**.

Loops= This give a count of how many times the wait interval time limit was reached. In other words, how many GQSCANs were performed.

1.6.5 STOP (or SHUT or END)

Issuing STOP, or SHUT or END will cause **ENQWATCH** to enter shutdown processing. The operator can also issue a P stcname to shut **ENQWATCH** down.

1.6.6 WAIT=nnnn

The WAIT command is used to set a different value for the WAIT time within the program. Changing the value via a WAIT command is not a permanent change. The default (or the value passed on the EXEC card) will be used when **ENQWATCH** is recycled.

There must be a space or an equal sign following the word 'WAIT' and then the numeric for the wait time in seconds. The value can be 1-9999.

The ability to dynamically change the wait time allows the program's resource usage to be better controlled. By setting the value too low the program will loop more often, by setting the value too high the program will not detect dataset contention in a timely manner.

It would be possible to set the value high and use the CYCLE facility to limit the times that **ENQWATCH** goes to check for dataset contention to just when a production job is being held out (or after 9999 seconds).

1.7 Return Codes

The program will always set condition code 0…unless it abends with a S0C3

1.8 SMF Record Layout

The format of the SMF records produced by **ENQWATCH** is as follows:

Field Name	Offset		Length	Description
	Dec	Hex		
SMFLEN	0	0	4	Length of the SMF record.
SMFSEG	4	4	4	Segment indicator.
SMFSIND	8	8	1	System indicator.
SMFRECTY	9	9	1	Record Type (default is 223).
SMFTIME	10	A	4	Time that record was written to SMF.
SMFDATE	14	E	4	Date that record was written to SMF.
SMFSID	18	12	4	SMF id of the system that the record was produced on.
SMF_WHO	22	16	8	TSO id who got the notification message.
SMF_JOB	30	1E	8	Job waiting for the dataset.
SMF_DSNLEN	38	26	4	Length of dataset name.
SMF_DSName	42	2A	44	Dataset name being waited on.

2 Messages

ENQW001I : Starting - ENQWATCH last assembled on mm/dd/yy at hh.mm

Meaning This message is issued when **ENQWATCH** is started. The date and time values will be filled in with the actual date and time that ENQWATCH was last assembled.

Corrective Action This is an informational message and no action need be taken

---

ENQW002I - Enqueue monitor active

Meaning This message is an indication that ENQWATCH has obtained all of the storage it needs and successfully established the operator interface and is now waiting on the ECB list. In other words…it is up and running.

Corrective Action This is an informational message and no action need be taken

---

ENQW003I - Free of start CIB unsuccessful

Meaning This message is issued to indicate that **ENQWATCH** had trouble freeing the control interface block for a start command. This is no big deal as we don't use a subsystem start command!.

You should never see this message, but it is included for completeness.

Corrective Action This is an informational message and no action need be taken

---

ENQW004I - Enqueue monitor shutdown in progress.

Meaning This message is issued to indicate that and stop command was issued for **ENQWATCH** and it is being closed down.

Corrective Action This is an informational message and no action need be taken

---

ENQW005E - GQSCAN Failed. Retrying.

Meaning This message is issued to indicate that EQNWATCH had a problem issuing the GQSCAN macro to list dataset contentions. The program will attempt a retry at the next wait time expiration.

If the message continues to keep on coming out frequently then notify the System Programmer.

Corrective Action This message is an indication that the GQSCAN area is too small to hold all the contentions. If this message keeps on coming out consider increasing the value for ARAEASIZE and re-assembling the program.

ENQW006E - Invalid command.

Meaning This message is issued to indicate that the operator command entered for **ENQWATCH** was invalid or not supported.

Corrective Action Correct the command and retry.

ENQW007I - ENQWATCH was assembled using these values:

Meaning

The ENQW007I messages are issued to display the current value of the MAXMSGS, MAXOWNER, AREASIZE and WAIT time.

In the case of WAIT time there is also an indication of how the wait time limit was set (operator command, default or passed parameter).

These messages are issued and startup and as the result of the INFO operator command being issued.

An example of the ENQW007I messages follows:

```
ENQW007I - ENQWATCH was assembled using these values:
ENQW007I - Maxmsgs=5, Maxowner=50, Areasize=10,000
ENQW007I - Wait time=232 Seconds (Set via Parm).
ENQW007I - SMF Recording is ON. Record Number=233
```

Corrective Action

These are informational messages and no action need be taken

ENQW008I - Status: Notify=n, Cycles=c, Loops=l

Meaning

The ENQW008I message is issued either as a response to a STATS command or at **ENQWATCH** termination time

The message is a count of :

- How many times a message was sent to TSO users (The notify value)
- How many CYCLE commands were issued to the program (The cycle value)
- How many times the wait interval triggered a GQSCAN (The loops value)

Corrective Action

This is an informational messages and no action need be taken

ENQW009E - Invalid WAIT command. No action taken.

Meaning

The ENQW009Emessage is issued as the result of an invalid wait value being issued by an operator command.

The prior wait value will still be in effect until a successful WAIT command is processed.

Corrective Action

Correct the WAIT command and retry.

ENQW010E - Invalid PARM passed. Default wait taken

Meaning The ENQW010E message is issued when the parameter being passed to ENQWATCH on the EXEC JCL card is detected to be invalid.

The parameter will be ignored and the default wait time value used (currently 60 seconds).

Corrective Action The wait value can be changed dynamically via the WAIT command, but the JCL will need to be corrected in order to make this a permanent change

ENQW011I - SMF recording has been switched OFF | ON

Meaning The ENQW011I message is issued in response to a F ENQWATCH,SMF OFF or SMF ON command.

The status of SMF recording can be displayed by using the INFO command.

Corrective Action This is an informational messages and no action need be taken

ENQW012E - Invalid SMF command. No action taken.

Meaning This message is issued when the program detects that an invalid format of the F ENQWATCH,SMF command has been issued.

The only options for the SMF command are SMF OFF or SMF ON

Corrective Action Retry the command using the correct format.

Summary of Amendments

Date	By whom	Change id	Changes
3rd February 2011	K Ferguson	n/a	Initial program (Version 2.1.00) released
8th February 2011	K Ferguson	KF001	Moved waitime to a variable at the top of the program
24th February 2011	K Ferguson	KF002	Added SMF support
28th February 2011	K Ferguson	KF003	Altered the meaning of the AREASIZE value. It now is a number that gets multiplied by 1000.