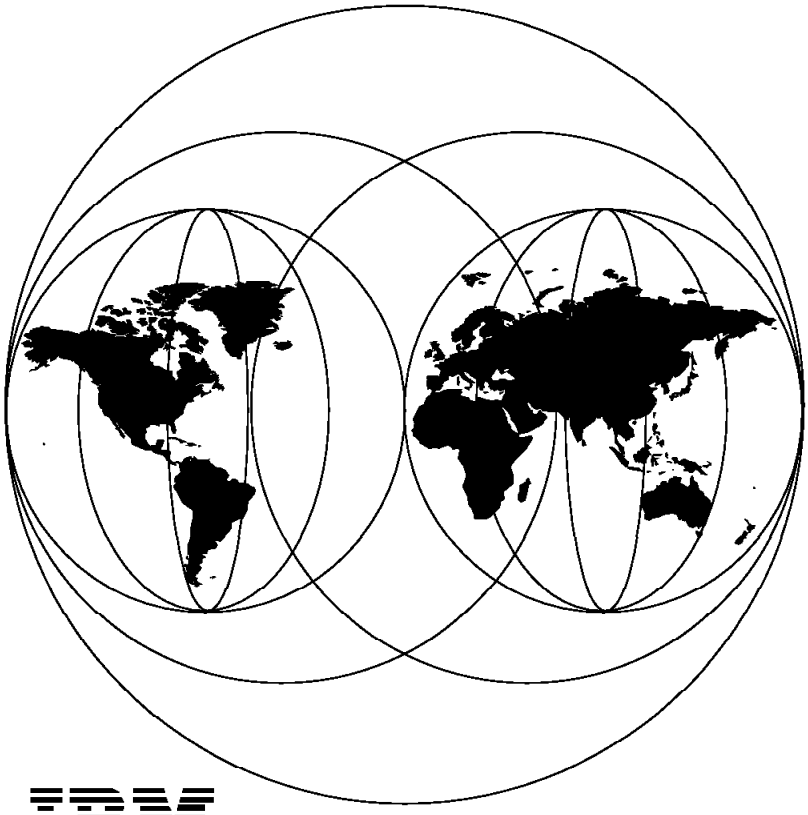
SG24-2130-00

Maximizing Your OPC/ESA Throughput

December 1997



IBM

International Technical Support Organization
Poughkeepsie Center

International Technical Support Organization

SG24-2130-00

Maximizing Your OPC/ESA Throughput

December 1997

Take Note!

Before using this information and the product it supports, be sure to read the general information in Appendix D, “Special Notices” on page 49.

First Edition (December 1997)

This edition applies to Version 1.3.1 of Operations Planning and Control (OPC/ESA), Program Number 5695-007 for use with the OS/390 Release 3 Operating System

Comments may be addressed to:
IBM Corporation, International Technical Support Organization
Dept. HYJ Mail Station P099
522 South Road
Poughkeepsie, New York 12601-5400

When you send information to IBM, you grant IBM a non-exclusive right to use or distribute the information in any way it believes appropriate without incurring any obligation to you.

Contents

Preface

Do you have questions on OPC or TME 10 OPC? This redbook may have your answers. It examines throughput issues for the Operations Planning and Control/Enterprise Systems Architecture (OPC/ESA) version 1.3.1 batch scheduling product (for use with the OS/390 release 3 operating system), explains how the various subtasks that update the current plan inter-relate and tells how you can maximize throughput and avoid bottlenecks.

This book was written for OPC/ESA specialists, administrators and system programmers. We assume that the reader is familiar with the basic processes of OPC/ESA and has a good understanding of MVS, JES and VSAM.

Note that, although this redbook specifically refers to the product known as OPC/ESA, the recommendations are equally relevant to the latest version, known as TME 10 OPC.

The Authors

This redbook was produced by a team of specialists from around the world working at the OPC development lab in Rome, Italy, in conjunction with the International Technical Support Organization Poughkeepsie Center.

Bob Watling is a Systems Management Technical Consultant based in the United Kingdom, with more than 30 years of Systems Management experience. He has been the IBM representative at the UK-based OPC user group for more than 15 years and has accumulated a vast amount of knowledge and experience over this period of time in the area of batch management.

Anna Dawson is also a UK-based Systems Management Technical Consultant, working in IBM Sheffield. Before joining IBM, she worked at a very large customer site where she was the primary person responsible for the day-to-day customization, implementation and exploitation of their batch scheduling environment. She has many years of experience with the OPC product and has focused most recently on the area of performance.

Thanks to the following people for their invaluable contributions to this project:

Maria Pia Cagnetta
OPC/ESA support manager, Tivoli laboratory, Rome

Rossella Donadeo
OPC/ESA support, Tivoli laboratory, Rome

Colette Kay
Karam Duhra
Software Support Specialists, PSS Central Services, Warwick

Bob Haimowitz
Paul Rogers
Mike Ebbers
ITSO, Poughkeepsie

Cy Atkinson
OPC Level 2 Software Support, Raleigh

Dick Orr
OPC Technical Support, Washington Systems Center

Doug Specht
OPC Software Migration Specialist, Pennsylvania

Comments Welcome

Your comments are important to us!

We want our redbooks to be as helpful as possible. Please send us your comments about this or other redbooks in one of the following ways:

- Fax the evaluation form found in “ITSO Redbook Evaluation” on page 61 to the fax number shown on the form.
- Use the electronic evaluation form found on the Redbooks Web sites:

  For Internet users `http://www.redbooks.ibm.com`
  For IBM Intranet users `http://w3.itso.ibm.com`

- Send us a note at the following address:

  `redbook@vnet.ibm.com`

Chapter 1. Introduction

With increases in outsourcing, many data centers are being merged. The scheduling requirements of each are being combined to produce higher and higher numbers of jobs that need to complete in ever-shrinking batch windows.

The drive for businesses to remain not only competitive but to lead their market increases the number of services they provide and the interfaces between them. This again increases the number of batch processes and tightens the batch window.

Shortening the delay between one process completing and its successors starting becomes more and more important.

Understanding the current workload and OPC/ESA will enable the most appropriate throughput enhancements to be made.

1.1 Recognizing a Throughput Problem

When the batch processes fail to make their deadline times without some easily identifiable reason, investigation into more obscure causes should be undertaken.

Pointers to a throughput issue might include:

- Dialog users complaining about very poor responses
- Many operations remaining in ready status for no apparent reason
- One or more images running no, or very few, jobs at what was a fairly busy time
- Delays between related batch jobs
- Many operations remaining in ″SU″ status

1.1.1 A Typical Scenario

You have been experiencing what appear to be performance problems with OPC/ESA. The start of the problems is traced back to a change that caused several input files to become available within a few seconds of each other, where they had previously been spread over several hours.

This change to the batch profile means more than 100 jobs become ready for submission at the same time. Submission takes over 15 minutes. Not only are these jobs affected, but also others are delayed, at times by up to half an hour. Also the operators are claiming it takes five minutes to access the ″operations in error″ dialogs and another five minutes to resubmit the repaired jobs.

This is unacceptable. What can you do?

Reversing the change is probably not a viable option and it does not deal with the underlying throughput issues. Instead, an understanding of how OPC/ESA controls the current plan changes and the processes involved will enable you to decide the best course of action from the options discussed later in this redbook.

1.1.2 Basic OPC/ESA Overview

OPC/ESA generally runs as two system tasks, the tracker and the controller.

The tracker passes events created by OPC/ESA interface programs and SMF and JES exits to the controller through a communications path.

The controller acts on those events, either updating the current plan or discarding the information if irrelevant.

The controller also handles requests from dialog users. Depending on the request, it will update the current plan or one of its other VSAM databases.

The controller is the ″scheduler″, determining which process is to be scheduled next and passing it to the appropriate executor task. For a job, it would pass the JCL to the destination defined in its work station, which would probably cause the JCL to be routed to a tracker for submission; thus we come full circle.

Today trackers may run on many different platforms, but historically OPC/ESA has controlled the batch scheduling on IBM (or compatible) mainframes running MVS/JES images.

Chapter 2. How OPC/ESA Processes Its Current Plan

Understanding how OPC/ESA handles the Current Plan (CP) is fundamental to understanding how to deal with any throughput problem.

An understanding of your own batch profile is also required to help determine a course of action that best suits your installation.

The four subtasks that update the current plan are:

- Normal Mode Manager (NMM)
- Event Manager (EM)
- Work Station Analyzer (WSA)
- General Service Task (GS)

The following sections provide a brief description of each of these subtasks.

2.1 The Event Manager, Reader, and Writer

The Event Manager (EM), Event Reader (ER) and Event Writer (EW) tasks together handle events. EM runs in the controller address space. Its purpose is to handle the events sent to the controller by local and remote trackers. The EM updates the Current Plan (CP) which can start other controller tasks (for example, Work Station Analyzer).

Events are created in two ways:

- Automatically by SMF/JES exits
- By user actions (via EQQUSINx routines, EQQEVPGM or TSO commands)

EW stores the events in the Event data set and passes them to the communication task (for example XCF, NCF, TCP/IP) to be transmitted to the controller if needed. ER or a communication task queues the events to the EM in the controller address space via the Data Router task. EM will process the events in its queue.

2.1.1 Initialization Statements Affecting Event Handling

The following statements are applicable to the Event Manager in the tracker task and are associated with removing irrelevant events and any delay between a job/task completing and the controller receiving that information. They do not influence the performance of the Event Manager in the controller task.

2.1.1.1 EWTROPTS

This is an option used by the EW task.

PRINTEVENTS Using a value of NO prevents print events (type 4) being passed to the controller. This is only valid if you do not want to track printing.

STEPEVENTS Using a value of NZERO or ABEND will cut down the number of step events passed onto the controller.

EWSEQNO Use this parameter when the tracker is connected by XCF or NCF (SNA), but not when trackers communicate by shared DASD. It causes the tracker to write events to both the event data set and the communications task at the same time, rather than writing them to the data set and then reading them back to the communications task. This speeds up the delivery time of the event from the tracker to the controller (it cuts out two I/O operations per event).

2.1.1.2 EXITS

CALL04 Use this parameter to load EQQUX004 (if you have written one). The exit can be used to filter out events of no interest to the controller before they are sent. An example would be to exclude testing jobs that could be identified by jobname prefix.

2.1.1.3 OPCOPTS

JCCTASK Do not use the JCC task unnecessarily. The tracker cannot pass job termination events to the controller until after the JCC has processed the joblog. Where JCC must be used, consider its usage carefully. JCC processes jobs singularly, so try to have a particular output class for JCC processing and create job-specific table entries, rather than using the general table.

If this task is activated, consideration should be given to all the parameters coded for the JCCOPTS statement.

CATMGT Do not use catalog management unnecessarily. Switching on this facility increases the traffic between the tracker and the controller significantly.

The function must be switched on if you wish to use the joblog retrieval facility

If this task is activated, consideration should be given to all the parameters coded for the JOBOPTS statement.

Consider allocating the event dataset with a larger LRECL value than the default to reduce the number of continuation events and therefore the number of I/O operations. See *OPC/ESA Installation Guide*, SH19-4010, for details of EQQEVDS and EQQEVDnn allocation.

2.2 General Service Subtask

The General Service subtask (GS) services a queue of requests from the dialogs, batch loader and program interface (PIF) to the OPC/ESA subsystem. The first request on the queue is selected and passed to a general-service executor. This process continues while there are requests on the queue. The GS subtask then waits for more work to arrive or for a stop-processing request to be posted. A stop processing request can be issued from other tasks within OPC, such as the Event Manager or Work Station Analyzer tasks. The General Service executors process the requests that are on the GS queue.

The GS task can attach up to five GS executor tasks to prevent service requests from being queued too long.

2.2.1 Initialization Statements Affecting the General Service Task

2.2.1.1 OPCOPTS

GSTASK This parameter allows you to specify how many dialog requests can be handled simultaneously, up to a maximum of five. The default is 5.

2.3 Normal Mode Manager

The Normal Mode Manager (NMM) subtask is responsible for the integrity of the controller data. NMM updates the current plan during CP and JS backups and when a new current plan (NCP) is being created.

2.3.1 Initialization Statements Affecting the Normal Mode Manager

Let us look at two of the parameters having to do with the backup function.

2.3.1.1 JTOPTS

BACKUP The value used here determines how often the NMM instigates an internal backup of the CP file. A value of NO allows the user to control when this backup takes place by executing the BACKUP command.

MAXJSFILE The value used here determines how often the NMM instigates an internal backup of the JS file. A value of NO allows the user to control when this backup takes place by executing the BACKUP command.

It should be noted that stopping OPC from performing an internal backup automatically, without having a regularly triggered process to replace it, will affect performance and may even bring down the controller.

The CP backup should be done no less than four times in any 24-hour period, at times that suit your disaster/contingency plans. The timing of the JS backups will depend on the frequency and timing of manual updates to the run-time JCL, but will probably be less frequent than the CP backups.

2.4 Work Station Analyzer

The Work Station Analyzer (WSA) subtask analyzes operations (jobs, started tasks and WTO messages) that are ready to start at:

- Automatically reporting computer work stations
- Non-reporting work stations
- General work stations with the WTO option set

For each active work station that is defined in the CP, the WSA sends a synchronization request to the related submit task.

The Work Station Analyzer identifies the most urgent operation to be submitted from a list of ready operations.

When a computer operation is selected, the JCL for the operation must be retrieved. Variable substitution, if required, is performed and the job input is

imaged to the JCL repository (JS file). A submit request is queued to the data router task, and the WSA updates the current plan status and records the event in the job tracking file.

The WSA subtask also processes any work station operation that has a deadline WTO message specified and issues any late or deadline messages.

2.4.1 Initialization Statements Affecting the Work Station Analyzer

Let us look at a parameter which affects performance of the Work Station Analyzer.

2.4.1.1 JTOPTS

QUEUELEN Defines the maximum number of ready operations that the WSA starts when it gets control of the current plan. The lowest (and default) value is 5.

2.5 Controlling Access to the Current Plan

To prevent simultaneous updates to the current plan, each subtask enqueues[1] on a system resource (SYSZDRK.CP) representing the current plan, when it has an update to do. When each task gets control, it will do as much as it can until it comes to the end of its queue, reaches a specified cut-off point or is signalled to stop by another task.

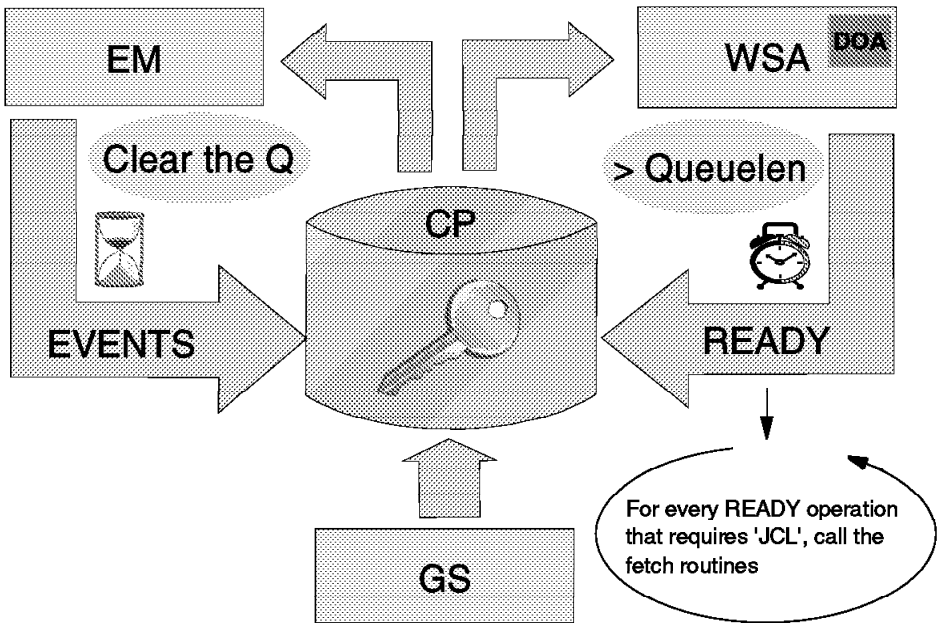


Figure 1. The CP Lock. Only the Work Station Analyzer, the Event Manager or the General Service task, shown above, may have access to the current plan at any one time. Each task must wait for access, while their queue of work may be increasing. The NMM task is not included in this figure, since its impact on performance is not great.

[1] The ENQ is issued with scope STEP inside the controller address space. It handles events sent to the controller by local and remote trackers. The synchronization with DP batch is done via ENQ with scope SYSTEMS and resource name SYSZDRK.ssnameXXXX, where XXXX can be TURN1, TURN2 and TURN3.

The amount of time each task holds the lock on the current plan and how much time is spent waiting is maintained by OPC and displayed to the started task message log if CPLOCK is included in the STATMSG keyword of the JTOPTS initialization parameter.

The issue frequency of these stats is governed by the value used on the BACKUP keyword of the OPCOPTS initialization parameter. The stats are displayed when more than 50% of that figure have been received since the last display.

Note: When the value NO is used for the BACKUP value, the figure used to display stats is the BACKUP default value of 400, which is every 200 events. In very busy periods, this can cause the stats to be displayed very frequently. It is possible to request a ZAP (fix for compiled code) from the change team to alter this defaulted value.

A request has been accepted by the change team to provide a permanent solution for this (request number REQ00046531). Your IBM representative can to advise you on the status of this request and you may wish to add your company to the list of those that concur with it.

2.5.1 An Analogy

To show how the Event Manager, the Work Station Analyzer and the General Service task interrelate, assume the Current Plan is an island and the different tasks are routes to it from the mainland. Using the following text, refer to Figure 2.

The Event Manager is a train that goes to the island. It travels very fast and can carry many passengers (events). However big or small the queue, the train takes everyone across. It takes a little longer to let them all off as it has only one exit. While the train is at the island station, the other routes to the island are closed. When all the passengers have left the train, it can return for more passengers and one of the other access routes is opened.

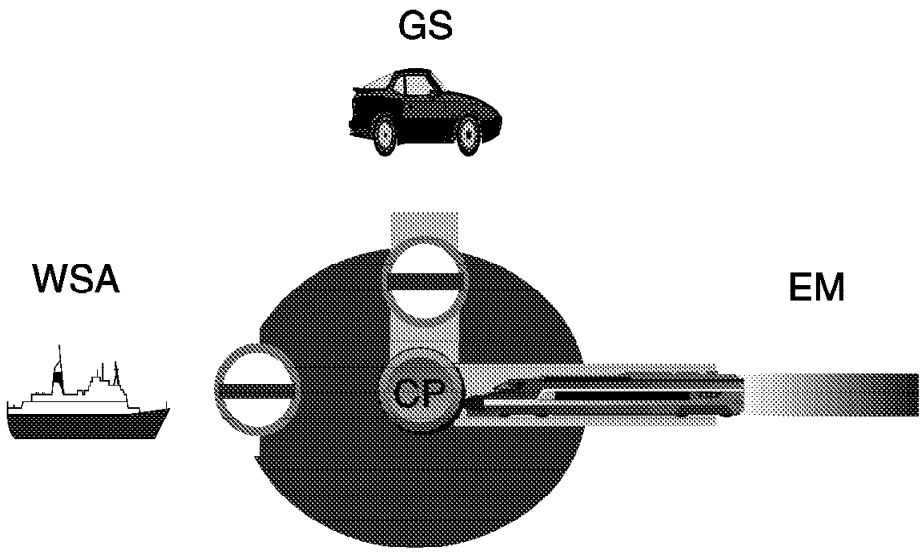


Figure 2. The EM Train. While the EM train is dropping off its passengers, all other routes are closed.

Now refer to Figure 3 as you read the following text. The GS cars on the road carry up to 5 people. The ferry and the train can request that the barrier be closed if they have passengers, so sometimes only one car gets across.

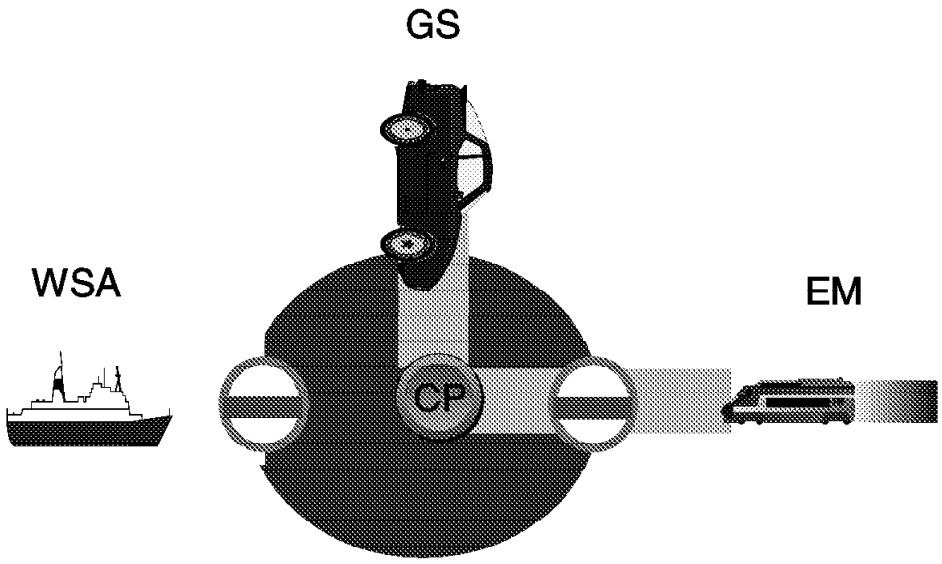


Figure 3. The GS Cars. When the GS car is on the island, all other routes are closed.

The cars represent the General Service task and the passengers are the subtasks. As some of the requests made through the General Service task can share access to the current plan, all the car doors can be opened simultaneously. Each passenger quickly returns to the car and drives off the island again. This allows either the train or the ferry passengers to disembark.

Refer to Figure 4 as you read the following text. The ferryman symbolizes the Work Station Analyzer. The size of his ferry is governed by the QUEUELEN parameter. When it is his turn to cross, he takes as many in his queue across as he can.

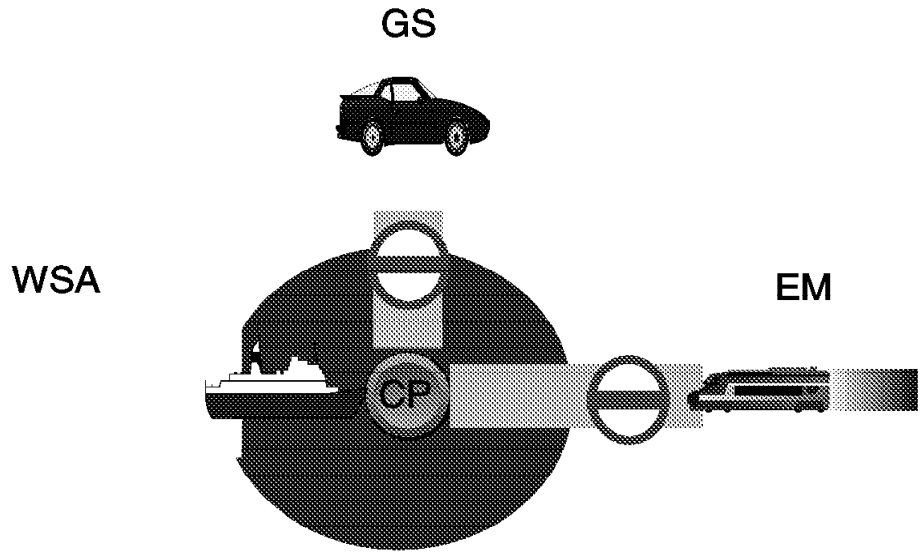


Figure 4. The WSA Ferry. While the WSA ferry is on the island, waiting for its passengers to return, all other routes to the CP island are closed.

The ferryman is a bit snobbish; he picks and chooses his passengers based on their relative importance. Sometimes a passenger bribes him (the EX command) so they can jump the queue. The passengers go shopping on the island for JCL and variables before returning to the ferry. The ferry then leaves the island to make room for the train or the car.

2.5.2 Striking a Balance

The CPLOCK statistics show how the WSA, GS and EM coexist.

It is highly unlikely that there will be no wait time showing against any of the tasks, as they generally all have something else to do when they relinquish the CP lock and will enqueue again.

The least busy is probably the General Service task; however, that is the task used by the external users of OPC, so its responsiveness colors their view of the performance of OPC.

Operators have every right to expect speedy responses. The last thing they want are responses so poor that, when they attempt to stop an operation from running, they find it has already started. Similarly, they need to correct jobs that have failed (and will elongate the batch window) as quickly as possible, not wait an extra five minutes just to re-submit them.

The obvious way to achieve this is to keep to a minimum the time that the Event Manager and the Work Station Analyzer need to update the current plan.

Even when the events have been filtered by the tracker tasks, the Event Manager nearly always has events to pass on. The time the EM takes is related to the number of events queued. The length of the queue depends on how long the EM must wait for the WSA to release the CP lock. The Event Manager processes events very fast, but in order to let the General Service task in quickly, the number of events on the queue should be kept to a minimum. You control this through the Work Station Analyzer.

With the WSA, you have a task that can be influenced. You can determine how long it holds the lock and consequently how long the other two tasks wait. The QUEUELEN parameter determines how many operations are scheduled by the WSA before the CP lock is released.

But....there is always a but! If the number of ready operations on the combined work stations ready list (the dynamic operation area (DOA) queue) rises rapidly, some jobs could continually be relegated to the back of the queue, delaying their submission for a considerable time. How much time depends on how long it takes the WSA to select an operation for submission and actually place it in started status, multiplied by the number of jobs there are in the queue.

The number of jobs in the queue will continually change. This is because new operations are reaching ready status as the Event Manager completes their predecessors and the resources that operations require are freed up or allocated elsewhere.

Striking the correct balance between job submission rate (QUEUELEN) and dialog responses is difficult without a thorough knowledge of the batch cycle and an OPC environment tuned to provide the best WSA throughput possible.

Chapter 3. Analyzing Your Throughput

Do you have an OPC throughput problem? Unless you have already done something about it, the answer is “probably.”

The question should be: Is your OPC throughput sufficient for your needs? Without analyzing your system, there is no reliable answer.

3.1 Gathering Statistics

The initialization statement JTOPTS has a STATMSG parameter. This has the following three keywords associated with it:

CPLOCK The Event Manager subtask issues messages EQQE004 and EQQE005, which describe how often different tasks have referenced the current plan data set.

EVENTS The Event Manager subtask issues messages EQQE000, EQQE006 and EQQE007, which describe how many events were processed and provides statistics for the different event types.

The messages for CPLOCK and EVENTS are issued when the number of events that have been processed by OPC since the previous message is greater than half the value of the BACKUP keyword, or 200 if the BACKUP value is NO.[2]

Both sets of statistics are generated at the same time.

GENSERV The General Service subtask issues messages EQQG010 to EQQG013, which describe how often different tasks have been processed and how long the General Service queue has been. OPC issues these messages every 30 minutes if any requests have been processed.

Deciphering these messages will tell you the length of the event manager queue between CP locks, roughly how many jobs were set to ″started″ and how many locks the Work Station Analyzer took to achieve this total.

See Figure 5 on page 12 for a sample of these statistics. Collating this information over time should enable you to decide whether throughput is an issue.

[2] You can request a ZAP (fix for compiled code) from the change team to alter this defaulted value. A request has been accepted by the change team to provide a permanent solution for this (REQ00046531).

```
EQQE000I TOTAL NUMBER OF EVENTS PROCESSED BY THE EVENT MANAGER TASK IS: 1149550
EQQE000I NUMBER OF EVENTS SINCE THE PREVIOUS MESSAGE IS:                  4001
EQQE000I EVENT MANAGER QUEUE LENGTH STATISTICS FOLLOW:
EQQE000I  TOTAL     Q1     Q2     Q5    Q10    Q20    Q50   Q100   >100
EQQE000I   3806   3678     88     27     11      2      0      0      0
EQQE006I EVENT MANAGER EVENT TYPE STATISTICS FOLLOW:
EQQE006I EVTYP   TOTAL  NEWEVS  TOTTIME  NEWTIME   TOTAVG  NEWAVG  DELAYED
EQQE007I ALL   1149550    4001   7589.4     80.6     0.02    0.02        0
EQQE007I 1      172678     656   4999.8     21.7     0.02    0.03        0
EQQE007I 2      172229     655   4093.9     20.3     0.02    0.03        0
EQQE007I 3S      46297     128    333.3      0.0     0.00    0.00        0
EQQE007I 3J     147908     615    572.2      0.0     0.00    0.00        0
EQQE007I 3P     221613     787   1273.1     27.2     0.05    0.03        0
EQQE007I 4       35195     154     30.8      0.1     0.00    0.00        0
EQQE007I 5      227641     837    122.4      0.2     0.00    0.00        0
EQQE007I USER    19960       2    723.6      0.0     0.03    0.00        0
EQQE007I CATM        0       0      0.0      0.0     0.00    0.00        0
EQQE007I OTHR   100277     132   5439.9     10.8     0.05    0.08        0
EQQE004I CP ENQ LOCK STATISTICS SINCE PREVIOUS MESSAGE FOLLOW:
EQQE004I TASK               NEXCL  NSHRD   HELD   WAIT  AVGHELD AVGWAIT
EQQE005I NORMAL MODE MGR       63      0    0.7    0.4     0.01    0.00
EQQE005I WS ANALYZER          137      0   13.1    5.0     0.09    0.03
EQQE005I EVENT MANAGER       3806      0   81.8    8.3     0.02    0.00
EQQE005I GENERAL SERVICE       15    122    3.5    0.6     0.02    0.00
```

Figure 5. Message Log CPLOCK Statistics. This shows output by the Event Manager after a number of events have been processed. They show the CPLOCK relationships between the NMM, WSA, EM and GS tasks, and a breakdown of the event types received since the last stats were output. Cumulative data since the controller was started is also shown.

Refer to *OPC/ESA Messages and Codes*, SH19-6719 for a full explanation of the messages issued by the STATMSG parameter.

In a perfect world, the TOTAL and Q1 values on the EQQE000I message line would be the same, with zero for all the other queue lengths. That would mean every time the Event Manager was passed an event, it was able to process it immediately.

Instead it is likely that your figures will resemble those in Figure 5, or reflect even longer queues. The Event Manager queue grows because it is enqueuing on the current plan. This is because the CP is being held by other tasks, most frequently by the Work Station Analyzer. The more the WSA has to do, the longer it holds the CP lock.

In this example, the most jobs that the WSA could have scheduled is 656; this equates to the number of type 1 events received. Type 1 events are produced when a job is read onto the JES queue using an internal reader, although not necessarily submitted by OPC. The lowest is 137, the number of CP locks held by the WSA, assuming at least one job was scheduled for each lock. This is quite a difference.

Understanding the workload at the time of day the statistics cover will help, but it would still be fairly inaccurate, unless no tasks start on any controlled image other than through OPC. To really know how many operations were scheduled, you need to review the job tracking logs.

3.2 Using the Job Tracking Log Data

It is possible to run analytical code against the job tracking archive and log files, or against the TRKLOG file produced by the daily planning job.

The format of the files is documented in the *OPC/ESA Diagnosis Guide and Reference*, LY19-6350 (available to IBM-licensed customers only).

To determine how many operations were set to ″scheduled″, you need to count the number of type 25 records generated. When analyzing these against the stats messages, you will need to limit the time frame covered to match the stats.

The audit program supplied as a sample with OPC deciphers the job tracking files or the TRKLOG data into a readable format. On reading the output produced, you can determine which task had control at any one time.

The time stamp in the audit report is limited to seconds. To determine how long the WSA takes to select and schedule an operation, it is more accurate if you can use the raw data, which is recorded in hundredths of seconds.

When the WSA gets control, the first message generated is a type 25, issued after the first operation has been selected and scheduled. Assuming there is more than one ready operation, the time between this type 25 and the next will show how long the WSA took to select and start the next operation.

A lot will depend on your operations requirements, so this data cannot be used alone to determine the amount of time being taken to schedule every operation. The average over time will provide a better guide to current throughput speed.

Chapter 4. Tuning OPC Throughput

In chapter 3, we described ways to analyze your throughput. But what is good throughput? It depends on your requirements. As long as OPC is scheduling the workload and there are no perceived delays, there is no need to do anything.

However, if you are experiencing delays, (or think you are), it is time to examine your system. Throughput issues generally occur during peak points in the batch profile.

Things to look for in the STATMSG messages are:

1. Both WSA and EM have the same number of CP locks
2. You have very few (if any) EM queues of only one or two events
3. You have long wait times on the EM and GS tasks
4. You see long hold times on the WSA

You can lessen the load on the EM by using EQQUX004 in the tracker tasks to filter out unnecessary events. But where the delays are due to the EM waiting for the WSA, it is the WSA task that needs attention.

4.1 Breakdown of Work Station Analyzer Task

To understand where improvements can be made, it is necessary to understand what the Work Station Analyzer does when it gets the CP lock.

1. Issues any duration or late alerts.
2. Searches the DOA[3] queue for the most suitable candidate for submission based on the following priorities:

   The operation has been EX′d
   The operation is priority 9
   The operation with the most urgent latest out time
   The operation′s priority 8 through 1, with 8 having the highest priority
   The operation′s duration
3. Searches the JS file for the JCL. If not found, the user exit EQQUX002 (if loaded) is searched, followed by the EQQJBLIB concatenation, if needed.
4. Performs the JCL variable substitution.
5. Updates the operations status to S.
6. Passes the JCL to the external router for transmission to the appropriate tracker.
7. Checks if the QUEUELEN value has been exceeded (maximum number of loops is QUEUELEN value plus 1, unless the fix for APAR PQ06842 has been applied when check is that QUEUELEN has been reached) and either starts searching the DOA again, or releases the lock.

[3] DOA stands for dynamic operation area, the control block used by OPC to represent an operation. It is used to build the *ready queue.* and by WSA to schedule the next piece of work.

As this list shows, the WSA has quite a lot to do. If the ready queue is large, it has to search it at least six times and fetch the JCL for six operations.

This “find-a-winner” routine is done very fast, as the DOA queue will almost certainly be in storage. At least 50% of the Current Plan should be in storage. Due to the frequency with which the DOA queue is referenced, it is unlikely to have been paged out.

Note: For additional information on the “find-a-winner” routine, see section B.3, “Find-A-Winner Algorithm” on page 39.

Fetching the JCL from the JS file should also be quite fast. Although it uses an I/O call, it is to a keyed record in a VSAM file (which should be on a cached unit).

It is the fetch of the JCL from the PDS library that takes the time.

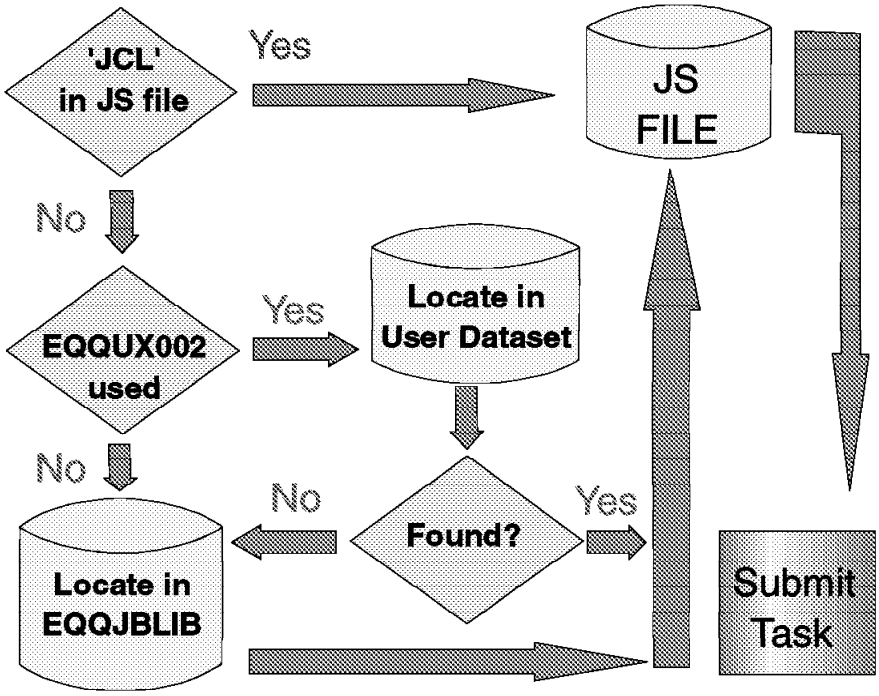


Figure 6. JCL Fetch Routine. The moving of JCL from the data set (for example, a PDS) to the VSAM file (JS) for further processing can be a major bottleneck, depending on the size of the directories OPC needs to search.

Even though their JCL libraries were on cached volumes, we know of a site that experienced delays of up to 7 seconds per job when fetching JCL. Multiply this by their QUEUELEN of 10 and the WSA had the potential to hold the CP lock for over a minute.

A lot of events can queue to the EM in this length of time and a lot of dialog users can be made very unhappy!

4.1.1 Improving Work Station Analyzer Throughput

Small gains in throughput can be made by ensuring:

- The VSAM files used for JCL imaging (JS files) are sized correctly
- The VSAM primary allocation is large enough and not split over extents
- CI and CA splits are kept to a minimum by allocating adequate freespace

The best improvements that can be made are in the area of JCL retrieval from PDS into VSAM.

The methods used to speed the access times are installation dependent. The hardware and software available will dictate the best options from those listed later in this chapter.

Regardless of which method is used to improve the speed of promotion of the JCL from the PDS library to the VSAM file, the best improvement that can be made to WSA throughput is to ensure that the JCL for the job is already in the VSAM file when OPC wants to submit it.

The most appropriate method of pre-staging JCL into the JS file is by using the OPC Program Interface (PIF). There are several samples of PIF usage in the SAMPLIB shipped with OPC.

However efficient the staging program is, it will still be bound by the performance problems that affect the WSA fetch times. The main problem is the fact that OPC searches the EQQJBLIB concatenation for the JCL. To find a member in the last data set of the concatenation, OPC must read the directories of all preceding PDS libraries. When they contain thousands of members, this can be very time consuming.

In order to circumvent this directory search and go directly to a small operation-specific library, use EQQUX002. A sample EQQUX002 is provided in the SAMPLIB shipped with OPC.

For improved performance of EQQUX002, it is recommended that EQQUX000 (the OPC stop/start exit) be used to do the open/close routines needed for the JCL PDS libraries.

During benchmark tests, different EQQUX002 code was tested. When doing open/close processing within EQQUX002, performance was degraded as compared to the EQQUX002 code assuming the data sets were already open and opening them with EQQUX000.

4.2 Software Solutions

Your installation may have software products installed that allow you to place library directories (or the most frequently used portion of them) in storage. For instance, you can use LLA for this purpose.

Either of these actions will improve retrieval times. We know of one installation where they defined their main PDS library (over 50,000 members) to a PDS management tool, keeping only a directory of the most recent and most frequently requested 1000 elements in storage. They reduced the time taken to pre-stage 14,000 operations from 12 hours to 3 hours.

It should be noted that, for each JCL fetch, OPC makes three calls to the EQQJBLIB concatenation, so placing the directory in storage after the first find greatly improves the find time for the subsequent calls.

The benchmark results in Appendix B, “Benchmark Details” on page 35 give details of savings using LLA when pre-staging JCL.

4.3 Example Specifications

The following specifications are examples of solutions that could be used when writing EQQUX002 code or a staging program. Whether they meet the needs of your installation will depend on what customization has already been done to OPC (for example, your naming standards).

4.3.1 EQQUX002

For more information about EQQUX002, refer to the *OPC/ESA Customization and Tuning* manual, SH19-4011.

To use EQQUX002 to retrieve JCL directly from a small PDS, rather than searching the EQQJBLIB concatenation, it is necessary to pass information to the exit to identify that library.

The data could be used to build a data set name for programs using dynamic allocation, or to build a DDname for those that used libraries defined to DD statements in the controller started task.

The data would need to be defined for all those operations whose JCL could be specifically addressed.

The following fields could provide the data:

Jobname A substring of the jobname information could be used to determine the data set name or the DDname.

Work station The work station name could be used to determine the data set name or the DDname.

Application Name A substring of the application name could be used to determine the data set name or the DDname.

Owner ID A substring of the owner ID name could be used to determine the data set name or the DDname.

Authority Group The authority group or a substring of it could be used to determine the data set name or the DDname.

Class The single character class field of each operation could identify 1-36 different data sets or DDnames. The contents of the class field has no purpose in normal OPC operations.

Forms The 8-character forms field has no purpose within OPC when defined for a computer operation. Any number of data sets or DDnames could be constructed using between 1 to 8 of the positions.

Depending on the installations standards, it may be possible to use information already coded within each operation or application. There may be some other

installation-specific processes that dictate the need to use the class or forms fields.

The EQQUX002 code would need to revert to using the EQQJBLIB concatenation if the job was not found in the specified library. It could also check on the value of a field, such as the class field, to check whether to search a specific library at all.

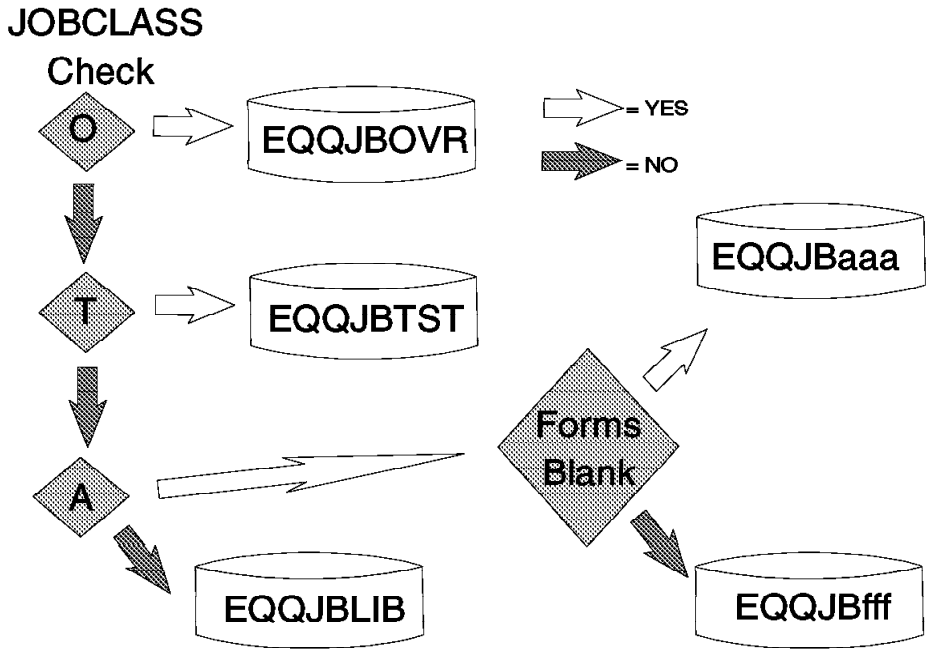


Figure 7. An EXIT02 Schematic. It shows a possible flow for the JCL fetch exit. The operations class field is checked to determine if a special DDname in the controller started task is to be used. This is determined by the class value, the first three characters of the application name, or the first three characters of the forms field.

4.3.2 Pre-Staging JCL

An OPC PIF program, for execution in batch, could be used to identify operations to be pre-loaded into the VSAM (JS) file.

The program would need to ensure that the job could be staged; otherwise control cards would be needed to directly identify eligible jobs.

A simple method to achieve this is to run a batch job to execute a PIF program that processed a list of applications (contained in control cards), specifically or generically. All the jobs in the applications would contain JCL that could be pre-staged into the JS file, well in advance of their execution.

It may be necessary to include an exclusion list for operations that could not be staged (for example, JCL that is generated just prior to scheduling). The details would be installation-specific.

Once all the batch cycle JCL was staged, a BACKUP JS command could be used to swap JS files. This would effect a repro and tidy up any CI and CA splits to further improve performance.

4.4 Tools and Tips

IBM Services can help you analyze your OPC system and assist you in implementing tuning solutions and exits.

You can contact your IBM local representative for more information.

Chapter 5. QUEUELEN Evaluation

The following 13 tests, shown in Table 1, were done during the OPC benchmark throughput tests. All options were exactly the same except for the following: the QUEUELEN parameter was incremented, the Long Term Plan was new, the Current Plan was new and the Job Submit file was empty. Tests for QUEUELEN 5 - 30 were run on a different day from those performed for QUEUELEN 40 - 500.

Note: The other activities on the processor at the time, although minimal, may not have been the same.

5.1 Test Description

The test consisted of one master application controlling the release of 10 successor applications. Each test lasted for 15 minutes and the results, shown in Table 1, were calculated as follows:

Jobs/Second Number of jobs processed in interval calculated by second

Avg GS Response The average of all GS WAIT times as shown by the CPLOCK statistics during the interval

Avg jobs per lock Calculated using the total locks taken calculated by the CPLOCK statistics during the interval

During these tests, the first job in each application added its own application back into the schedule using the program interface thus simulating dialog requests. However, since the newly added application was dependent on the completion of the initial application, the number of ready jobs in the schedule at any one time never exceeded 1000.

Table 1. Effect of QUEUELEN Parameter

Q-Len	Jobs/Second	Avg GS Response	Avg Job/Lock
05	3.23	0.33	1.15
10	3.24	0.76	1.32
15	3.23	0.82	1.63
20	3.05	1.15	1.66
30	3.16	1.41	1.41
40	4.01	0.57	1.41
50	4.17	0.62	1.26
100	4.47	0.48	1.39
150	4.41	0.47	1.41
200	4.16	1.35	1.73
300	4.13	1.11	1.64
400	4.41	0.47	1.35
500	4.84	0.87	1.34

These figures are average representations over the 15-minute test. However, by analyzing the statistics produced during the test, it is possible to see the real effect that the QUEUELEN parameter has.

The test machine used had a total of 100 initiators active. This delayed the WSA after the initial 500 operations had been queued, as it had to wait for more jobs to be added. These were added in groups of 50.

At the start of each benchmark, 500 jobs (10 applications with 50 jobs in each) were made ready at the same time. This would simulate a peak General Service task response time. However, once the benchmark had started, there were no controls on the order in which the jobs were processed and therefore no controls on the timings of the PIF additions to the schedule.

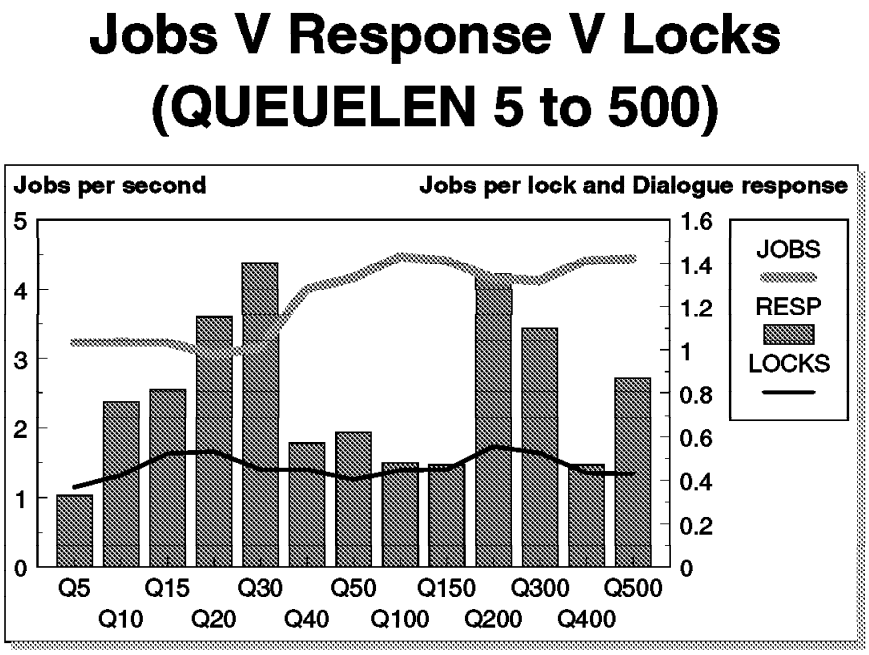


Figure 8. QUEUELEN Effect on Job Schedule Rate. This graph represents the number of jobs scheduled, on average, for each WSA lock measured against the wait time of the General Service task.

We can conclude from Figure 8 that the ideal QUEUELEN parameter for this particular workload mix would be 100.

This would give the best dialogue response and the highest job throughput, but at the same time a minimal number of locks would be taken per job.

In order to set the optimum QUEUELEN value on a well-tuned system, you have to understand your batch profile. You need to know where the peaks occur and what kind of peaks they are.

A *peak* is defined as:

1. Completion of one operation causing many operations to become ready
2. Several operations completing in a very short period of time, each causing several operations to become ready

In either case, a high QUEUELEN value is preferable. This will cause most or all of the operations to be submitted without delay, and will affect dialog responses for a only short period of time.

5.1.1 Conclusion

On a well-tuned system, a peak submission rate of 25 jobs per second is achievable.

Chapter 6. Recommendations

In order to ensure OPC/ESA performs well, both in terms of dialog response times and job submission rates, the following recommendations should be implemented. However, it should be noted that although these enhancements can improve the overall throughput of the base product, the amount of work that OPC/ESA has to process in any given time frame will always be the overriding factor. The recommendations are listed in the sequence that will provide the most immediate benefit.

6.1 Pre-Stage JCL

Move as much JCL as possible to the JS file before it reaches a ready state, using a program interface staging program. If severe problems are encountered it may be necessary to do this manually. The preferred method is to use an OPC PIF program.

A BACKUP JS command should be used once JCL has been staged to tidy up all the CI and CA splits caused by so many inserts. This will also improve performance on the JS VSAM files.

6.2 Implement EQQUX004

Implement EQQUX004 to reduce the number of events the Event Manager has to process.

Running non-production (or non-OPC controlled) jobs on a processor that has an OPC tracker started will generate events of no consequence that have to be written to the event data set and passed on to the controller. These will need to be checked by the controller's Event Manager against the current plan and discarded.

Removing these events can improve the overall performance of the OPC controller by lessening its overhead.

6.3 Place JCL Libraries in LLA or Equivalent

Place the job library(s), defined by the Data Definition statement EQQJBLIB in the OPC started task, in LLA or a PDS management product. For LLA, this is achieved by using the LIBRARY and FREEZE options.

Updates to libraries in LLA will not be accessible to OPC until an LLA REFRESH command has been issued for the library in question. A simple technique to cover this is to have a small UPDATE library concatenated ahead of the Production job library that would not be placed in LLA. On a regular basis, say weekly, move all the updated JCL from the update to the production library at the same time as a refresh LLA.

Note: The use of LLA for PDS libraries is dependent on the level of MVS installed.

6.4 JCL Library Placement

Ensure the JCL libraries are placed on the fastest possible storage device. The access times chart in Figure 9 shows the benefits that can be realized. Cache and DASD fastwrite are the combination of hardware and software tools you can use to place data in the storage controller.

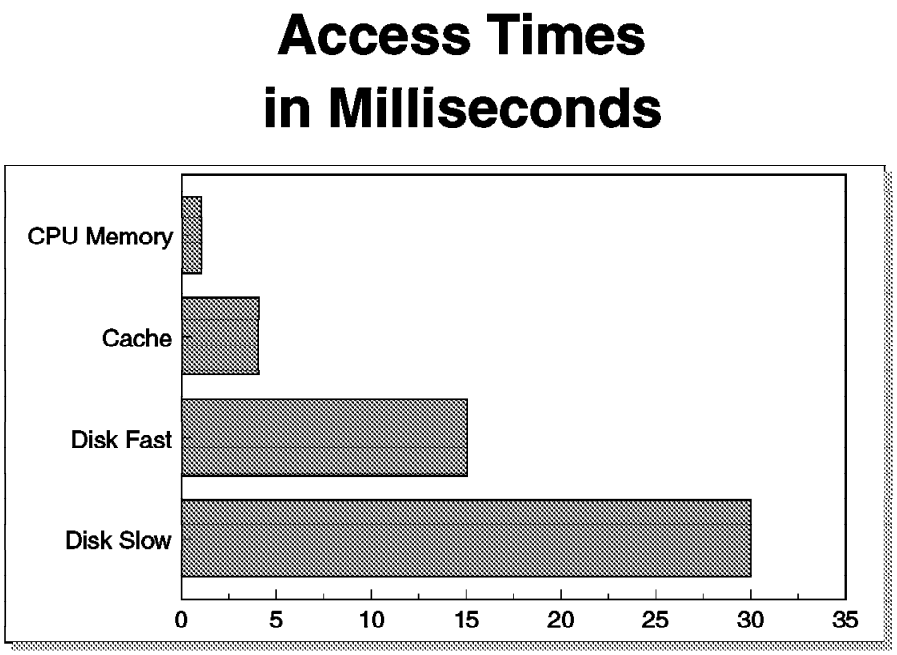


Figure 9. Access Times. Access time to the storage controller (cache) is faster than access to a disk. Access time to the former is four milliseconds while a disk can vary from 15 to 30 milliseconds. Memory is the fastest medium, although more expensive. Disk is the least expensive, but costly in terms of access times.

6.4.1 Use EQQUX002

Implement exit EQQUX002 to reduce JCL location time by reading the JCL from a specific Data Definition statement based upon a value or values available to the exit. Examples would be Application Name, Jobname, Jobclass or Forms type.

Note: Libraries defined specifically for use by the EQQUX002 exit should also be placed within LLA or an equivalent, if possible.

6.4.2 Use EQQUX000 with EQQUX002

Implement exit EQQUX000 to improve EQQUX002 performance by moving all open/close routines to OPC/ESA startup instead of each time the EQQUX002 is called.

The moving of JCL libraries under LLA and the introduction of EQQUX002 and EQQUX000 provided the increased performance that you see in Figure 10. Introducing the use of LLA provided the greatest single improvement. The exits alone caused significant improvement, and using LLA with exits was most beneficial.

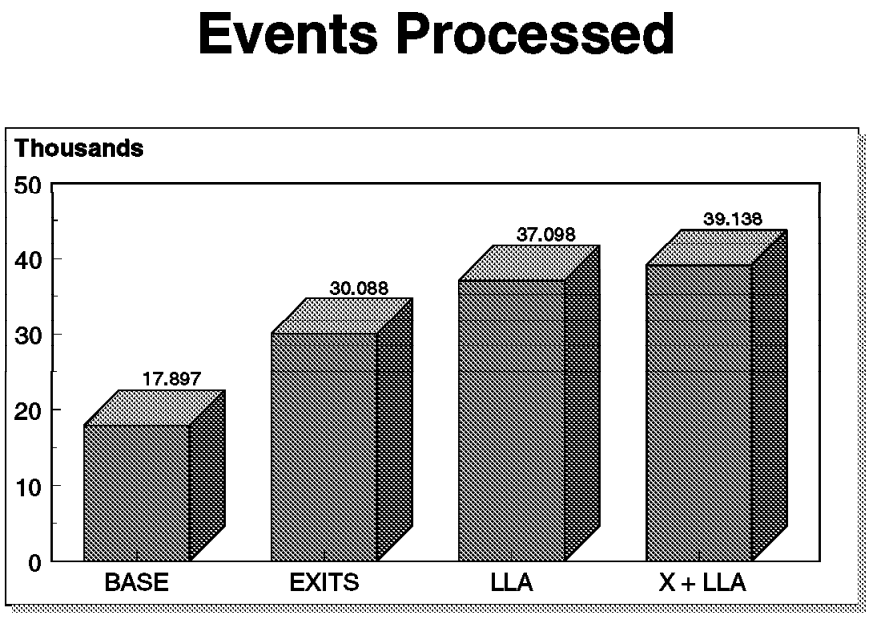


Figure 10. Events Processed Before and After Improvements. X refers to the use of EQQUX000 and EQQUX002 to retrieve JCL.

6.4.3 Library Maintenance

Ensure the JS file does not go into extents and that CI and CA splits are kept to a minimum. This will ensure that the JCL repository does not become fragmented, which leads to delays in job submission.

Ensure the JS file is backed up periodically, at times that are useful to your installation (see section 6.1, “Pre-Stage JCL” on page 27).

Enter a value of NO to the MAXJSFILE initialization parameter to avoid OPC initiating the JS backups. Run a batch job or TSO command regularly to execute the BACKUP (JS) command instead.

Clear out the JS file at regular intervals. It has a tendency to grow, since jobs that are only run once are never removed. There is a sample in SAMPLIB that can be used to delete items that are older than required.

6.4.4 Monitor Ready Queue

Monitor the maximum number of operations that are ready at any one time to assist in the setting of the QUEUELEN parameter and to eliminate any ready queue peaks during the planning period.

Monitoring the ready queue (DOA in Work Station Analyzer terms) is not a simple task using the standard OPC dialogs and statistic messages. However, it can be achieved by either writing a program interface module to regularly[4] query the work station activity or manually browsing each work station ready list, on a regular basis, and counting the number of operations that have a status of ″ready blank″ (R).

The IBM laboratory did enable a SNAP dump facility for the duration of the benchmark, which assisted with the testing and the monitoring of the DOA. This facility has now been formally requested to be included as a standard part of the product as an option on the STATMSG parameter. In addition, a program modification (ZAP) can also be supplied by the laboratory in Rome. This enables you to run with BACKUP(NO), yet you can control when any Event Manager statistics are produced.

Using BACKUP(NO), which many customers will want to do, will cause statistics messages to be generated by default every 200 events. Using this default value on a system that is already potentially stressed is far from ideal. The ZAP allows the user to amend the default number of events to a much larger figure, say 5000, which will greatly increase the time between the statistics.

6.4.5 QUEUELEN Balancing

Set the QUEUELEN parameter to achieve the desired balance between dialog response times and submission rates. This should only be done after all of the previous recommendations have been implemented.

Setting the QUEUELEN parameter to the maximum value will ensure that, once the CP lock has been taken, all ready operations will be processed. As long as all of the other recommendations have been implemented, the processing of the ready queue should be very fast. If the processing of the ready queue is still taking longer than expected, then dialog response times will suffer and, in the worst case, events may be lost due to Event Manager suspension routines.

When the Event Manager receives certain events out of the expected order, those events are suspended. Events are held in a suspended status for five minutes waiting for the ″missing″ events. If the missing events are not received, then the suspended events are processed and their associated operations placed into an error status.

Note: Based on the previous paragraph, your QUEUELEN parameter should never be greater than the number of operations that can be scheduled in any 5-minute interval. Identifying the maximum number of scheduled operations in any 5-minute interval can be achieved by analyzing the type 25 records within the job tracking data set.

---

[4] If you have no problems, monitor your system weekly to become familiar with it. When problems are being experienced, run the monitor every few seconds to gauge the extent of the problem.

6.4.6 Reduce the Size and Number of Members in EQQJBLIB

Consider all methods of reducing the number of members, and their size, within production JCL libraries.

Regularly housekeep the libraries and remove all redundant members.

Whenever possible, call procedures rather than maintaining large JCL streams in OPC libraries. Use JCL variables to pass specific details to the procedures, where procedural differences are based on data known to OPC, such as work station.

Allow the OPC/ESA exit EQQUX002 to create RDR JCL from a model. This idea is useful, for example, when several (especially if you have hundreds or thousands) of the members within the job library execute a procedure name that is the same as the jobname (or can be derived from it). Replacing the several members with just a few model members (held in storage) and having the exit modify the EXEC card, would reduce the size of the job library and therefore Work Station Analyzer overhead during JCL fetch times.

6.4.7 Exercise Good Scheduling Practices

- Specify priority 9 only in exceptional circumstances and ensure other priorities are used correctly.
- Ensure operation durations are as accurate as possible.
- Set deadlines only in appropriate places.

  These actions will ensure that the decisions made by OPC when finding the next ″most urgent″ job are as required by your installation.

  Inaccurate scheduling can cause many jobs to have the same internal priority to the scheduler. Preventing this will ensure the most critical jobs are scheduled first.
- Reduce unnecessary complexity in the schedules.

  Build dependencies only where they really exist. Each operation that completes has to notify all its successors; keeping the dependencies direct will shorten this processing.
- Keep ″schedule on time″ jobs to a minimum.

  Any operations that are scheduled purely on a time-release basis will be added to the ready list as soon as the current plan is cut. The WSA will need to check that their time has not been reached each time it gets the CP lock. Where it is necessary to schedule large numbers of jobs by time, reduce this overhead by batching jobs together behind ″dummy″ operations so that they only go ″ready″ shortly before their release time.

6.4.8 Review Your JES and MVS Tuning

Ensure your system is tuned to cope with the numbers of jobs being scheduled by OPC. It does no good to be able to schedule 20 jobs a second if the JES parameters are throttling back the systems and only allowing five jobs per second onto the JES queues.

Specifically review the manual *JES2 Multi-Access Spool in a SYSPLEX Environment*, GG66-3263, paying special attention to the values coded for HOLD and DORMANCY.

Remember, it is important to take steps to ensure that your MVS/JES system is properly tuned, in addition to taking the preceding specific steps.

6.4.9 Review Your Tracker and Work Station Setup

Where possible, work stations should direct their work to trackers for submission, especially where more than one system image is being controlled. This saves the controller the overhead of passing all the jobs to a single internal reader, which may itself prove to be a submission bottleneck. Delays would also be introduced in using some other router on the system (NJE) to pass the job to the appropriate execution system.

Consideration should be given to the method of communication used between the controller and the trackers. Of the three methods, XCF gives the best performance; however, its use is possible only in installations with the right hardware and software configurations. Using VTAM (the NCF task) is second in the performance stakes, with shared DASD being the slowest due to its being I/O intensive.

6.4.10 Review OPC Customization and Tuning Manuals

Especially review the chapters for basic tuning of the trackers and controller in relation to your installation. Implement any recommendations found that apply to your environment. Ensuring these recommendations are followed can make a big difference.

6.4.11 Review OPC Initialization Parameters

Ensure no unnecessary overhead has been caused by parameters that are not required by your installation, such as:

- Set PRINTEVENTS(NO) if printing is not tracked.
- Do not use STATMSG except when needing to analyze your system or when collecting historical data.

Appendix A. OPC Performance Parameters

The following statements and parameters can affect the performance of OPC.

For a detailed explanation on each of these parameters, refer to the *OPC/ESA Customization Guide,* SH19-4011.

- AUDIT
  - DATA or KEY. Specify only KEY where full DATA information is not required. This will reduce the amount of data written to the job tracking files.
- ALERTS
  - LATEOPER. Use only when deadlines are accurate. If there is a delay to batch, many jobs may produce alerts. Some automated trapping should be done and some means of preventing their production should be devised.
  - DURATIONS. Use only when durations are accurate and the default used for the ″limit of feedback″ figure in the initialization parameters will provide significant alert information.
- AUTHDEF
  - SUBRESOURCES. Specify only subresources that have rules defined. This will prevent unnecessary security calls being made.
- EWTROPTS
  - STEPEVENTS. Specifies when OPC creates events for ending job-steps. Preventing the creation of unnecessary events will reduce the controller overhead.
  - PRINTEVENTS. Specifies if OPC creates events for print tasks (type 4). Preventing the creation of unnecessary events will reduce the controller overhead.
  - HOLDJOB. Specifies if OPC holds and releases jobs on the JES queue.
  - EWWAIT. Specifies the time between reads of a submit/release data set.
- JOBOPTS
  - CATMCLAS. SYSOUT classes that catalog management checks. These should be kept to a minimum, preferably one, to minimize the amount of delay time between a job′s termination and its events being passed to the controller.
  - JOBLOGRETRIEVAL. Specifies how and when a job log is retrieved. Delayed retrieval will cause joblog data to be fetched only when requested. This will reduce the amount of traffic on the link between the tracker and controller.
  - MAXNUMUSYS. Amount of user SYSOUT sent with retrieved job logs. User SYSOUT can be very large.
- JTOPTS
  - BACKUP. Specifies if a CP backup is performed automatically and how many records are written to the JT log between backups. The frequency of the backup is important, as other users are locked out of the current

plan while it takes place. During busy periods, a backup may be performed every few minutes if the value assigned to this field is too low.

  - MAXJSFILE. Specifies if a JS backup is performed automatically and how large the JCL repository file grows before OPC performs a backup. As with the CP backup, a JS backup will lock the current plan.
  - QUEUELEN. Specifies the number of operations OPC schedules when the Work Station Analyzer subtask gets the CP lock.
  - STATMSG. Determines if OPC issues performance statistics for the current plan, Event Manager, and General Service. Producing these statistics is an overhead and should be avoided unless they are being analyzed.

- OPCOPTS
  - STORELOG. Specifies which job logs that were retrieved immediately are stored in the JS file.
  - VARSUB. Determines which jobs are scanned for JCL variables and directives. Unless nearly all your JCL contains JCL variables, it is more efficient to use the SCAN option and place //*%OPC SCAN cards in the appropriate JCL.
- ROUTOPTS
  - DASD. Lists the destinations (subrel, data sets, DDnames) for trackers connected via DASD (the slowest connection method).
  - SNA. Lists the VTAM LU names of the tracker systems connected via VTAM.
  - XCF. Lists the XCF member name of any trackers connected to the controller via XCF (the fastest connection method).

Note: You can get performance improvements by increasing the record length of your event data set. For tips, see the *OPC/ESA Installation Guide*, SH19-4010 and the *OPC/ESA Customization and Tuning* manual, SH19-4011.

Appendix B. Benchmark Details

In May 1997, before producing this redbook, benchmark tests were done against the OPC/ESA product using a dedicated IBM MVS processor based in Poughkeepsie, New York, USA.

The tests performed during the benchmark were split into three separate categories:

1. Job retrieval rates
2. Job submission rates
3. VSAM allocations

Job retrieval rates were monitored using a Program Interface (PIF) staging program that performed exactly the same process as the Work Station Analyzer function when it is requested to locate an operation's JCL.

By using this program in a controlled environment, the number of jobs that OPC/ESA could locate and store in the job submission VSAM file in a given time frame could be calculated based on different options in effect at the time. The tests performed in this way related to the location of the job libraries, exits (if any) used at the time, and the exploitation of any software- and hardware-specific facilities. Job retrieval rates ranged from between 23 and 666 jobs per minute, depending on the options in effect at the time.

Job submission rates were monitored by running several benchmark schedules over set periods of time. The program interface module EQQADD was utilized to add in additional jobs at specific points in the schedule to simulate user dialogs. By using this technique the General Service task (GS) could be monitored along with the Event Manager (EM) and Work Station Analyzer (WSA). During this process, some 500 operations would be set to ready at the same time, thus simulating a peak in the overnight batch. During the testing, the perception of dialog "lockouts" because of the amount of activity at peak times was around 8-10 seconds. During some of the tests, job submission rates reached a peak of 15,320 per hour (an average of 255 per minute) using 100 initiators, and this equated to 92,350 events.

VSAM allocations were modified for both Current Plan (CP) and Job Submission (JS) files in both the OPC/ESA started task and the batch planning jobs. There were no apparent benefits from changing any of these files allocations or buffering beyond that already recommended in the *OPC/ESA Customization and Tuning Guide*, SH19-4011.

Poor dialog response and job throughput can usually be attributed to the imbalance of the Work Station Analyzer, General Service and Event Manager tasks, caused by the excessive amount of time taken to locate the production Job Control Language. This is usually caused by the size of the directory and number of concatenated JCL libraries.

All the tests carried out in the benchmark were aimed at identifying the correct balance between these tasks.

The first set of tests were designed to prove that by using the latest technology in both hardware and software and by implementing OPC/ESA user exits, job throughput and user response times could be drastically improved.

The second set of tests were designed to prove what effect the QUEUELEN parameter (the key element in the equation of throughput versus dialog performance) has on the system.

The benchmark results proved conclusively that a well-tuned system, possible after implementing the recommendations in this redbook, is the ideal for any OPC/ESA installation.

B.1 Benchmark Tests - JCL Retrieval

The following list identifies the tests that have been performed using an OPC PIF staging program. This staging program, using the OPC programming interface, caused the JCL to be located and stored in the Job Submission (JS) file, just as it would be during normal OPC/ESA processing.

The comments show the options in effect at the time (for example, ALL joblibs in LLA, no exits in use).

You should also note that two versions of EQQUX002 were tested. EQQUX002 Version 1 (X2V1 below) opened and closed the data sets when the JCL was stored. EQQUX002 Version 2 (X2V2) used EQQUX000 (X0 below) to open/close the data sets at controller task start and end.

The staging program used the operations class field to determine if the JCL should be fetched from a specific library. If a value of ″A″ was found (″CLASS=A in use″ appears in the list), it then checked the forms field (first 3 characters) to determine the suffix of the DD name to use. Some tests had the exits loaded, but did not use them. This ensured there was no degradation involved in calling the exit, falling through it, and defaulting to search EQQJBLIB as normal.

TEST A ALL joblibs in LLA, X0 + X2V2 loaded, CLASS=A in use

TEST B ALL joblibs in LLA, no exits loaded

TEST C ALL joblibs in LLA, X0 + X2V2 loaded

TEST D Main joblib in LLA, no exits loaded

TEST E Main joblib in LLA, X0 + X2V2 loaded

TEST F ALL joblibs in LLA, X2V1 loaded

TEST G Main joblib in LLA, X2V1 loaded

TEST H No joblibs in LLA, X0 + X2V2 loaded, CLASS=A in use

TEST I ALL joblibs in LLA, no exits loaded, no cache, no DASD fast read/write

TEST J No joblibs in LLA, X0 + X2V2 loaded

TEST K No joblibs in LLA, no exits loaded, no cache, no DASD fast read/write

Note: The main joblib contained 47,000 members and was the third from the top of the EQQJBLIB concatenation for all tests except A and H, when it was third from the bottom (21 libraries concatenated).

Tests A and H processed more jobs since a new Current Plan was created to pick up the CLASS=A changes that were required to activate EQQUX002.

As you can see from Table 2, the staging process can stage anywhere between 23 jobs per minute and (after performance enhancements) 666 jobs per minute. If your ready queue was 100 jobs, then it could take anywhere between 9 and 260 seconds in elapsed time, a difference of 251 seconds. Therefore processing 10,000 jobs without tuning would mean more than 7 hours is spent fetching JCL.

Table 2. Staging Summary Table

TEST #	Elapsed Mins	Secs/job	Tot CPU	Jobs/Min
A	19	0.09	2.07	666
B	20	0.10	1.81	600
C	19	0.10	1.85	600
D	27	0.13	1.83	462
E	28	0.14	1.85	426
F	47	0.24	2.14	252
G	53	0.26	2.17	222
H	59	0.28	2.11	216
I	70	0.34	1.84	176
J	150	0.76	1.86	78
K	529	2.60	1.84	23

Note: While OPC is fetching JCL, the CP lock is held and other tasks (including users) have to queue on the current plan. With a QUEUELEN of 10 set, users could be held for 0.09 of a second up to a worst case of 3.8 seconds.

Setting the QUEUELEN high means the lock is held longer but more ready operations are started. Setting the QUEUELEN low means the lock is held for less time. This frees the current plan, giving dialog users good response, but fewer jobs are started.

In a busy system, many events are handled by OPC and many operations become ready in the plan. While the Work Station Analyzer (WSA) is processing ready events by finding the best candidate to submit, fetching the JCL and passing the request to the submit task, the current plan (CP) is locked. While the CP is locked, dialog users wishing to update the plan are queued and, more important, new events that have been received by the Event Manager are also queued.

B.2 Benchmark Tests - Job Submission Rates

To test the job submission rates in a controlled environment, the databases within OPC/ESA on the benchmark system were set up to simulate a larger-than-average OPC/ESA environment; the JCL libraries were set up in a similar fashion.

The following list shows the options in effect at the time of the benchmark.

Primary joblib	47,000 members
Other joblibs	17,000 members
Average # of records per job	20
EQQJBLIB concatenations	21
JCL libraries for EQQUX002	154
Applications in AD	25,180 (24,458) active
Work stations	69, (Computer 10, Automatic 37, Other 22)
Calendars	8
Special Resources	3,560
Ett	3,400
JCL Variable Tables	116
Current Plan Occurrences	4,600
Current Plan Operations	25,000 (CPU 19,000, Auto 4,000, Other 3000)

Many tests were performed to monitor the Work Station Analyzer, General Service and Event Manager subtasks. Tasks were performed with and without exits, with and without LLA, and using varying values for the QUEUELEN parameter.

Besides these tests, the effects of ″parallel servers″, ″deadlines″ and ″priorities″ were also evaluated. When parallel servers are in use by OPC/ESA, it has the effect of throttling back the number of ready operations that are submitted. This not only delays jobs but also causes the internal ready queue of OPC/ESA (DOA) to grow. Customers having more than 99 production initiators available to OPC/ESA production jobs on a work station have no option but to switch off control on parallel servers if they wish to utilize all of the production initiators.

Other factors that can cause the DOA queue to grow are special resources and time dependencies. Operations that are ″ready″ but cannot be submitted (because of ″waiting for time″, ″all parallel servers in use″ or ″waiting for special resource″) still have to be handled by the WSA find-a-winner algorithm. This algorithm is defined in section B.3, “Find-A-Winner Algorithm” on page 39.

In all the tests, the maximum number of initiators available to OPC/ESA for running the stress test workload was 100 maximum. The jobs that were actually processed were executing the IBM IEFBR14 module.

B.3 Find-A-Winner Algorithm

When the Work Station Analyzer gets control, it scans the internal ready queue (known as the DOA) to locate the next job to submit. This process is known as the “find-a-winner” algorithm. As mentioned earlier, not everything on the ready queue can be submitted. The following checks are made by the WSA before an operation is considered for submission.

- The work station used by this operation is active and has servers available
- Operation Status is R and has not been manually held (MH)
- Operation′s duration means that it can start and finish before its WS closure
- Any special resources the operation requires are available
- Job submission is activated
- No time dependency exists or time has been reached
- No catalog management action is running or pending
- All required work station resources are available

or

- The operation has been forced through the EX command

Once the operation has passed all of these tests and has been flagged as eligible for submission, the WSA scans the eligible list. Scanning or submission terminate when either:

- The number of operations submitted exceeds the value of the QUEUELEN parameter as specified within the OPC/ESA started task.
- There are no more ready operations eligible for submission.

To determine the sequence that the WSA should submit the ready operations, the following list highlights the sequence that is used to find the winner:

1. Priority 9
2. Earliest ″latest start time″

   The latest start time is calculated internally by OPC and is the latest time (based on the operation′s duration) that the operation must start by if it is to be completed before its deadline. If no deadline is specified for the operation, OPC uses the occurrence deadline and subtracts the duration for each successor operation on the critical path.
3. Priority 8 through 1, with 8 having the highest priority
4. Shortest estimated duration
5. First-in first-out basis

B.4 30-Minute Tests

Four nearly identical tests were set running, then stopped after 30 minutes. The tests were used to determine the effectiveness of using LLA and EQQUX000/2. Table 3, which follows, is also displayed as an events graph in Figure 10 on page 29.

Table 3. Effect of Using LLA and Exits

EXITS/LLA	BASE	X not LLA	LLA not X	LLA and X
WSA avg. held	0.27	0.08	0.07	0.04
WSA avg. wait	0.09	0.05	0.05	0.03
EM avg. held	0.02	0.01	0.01	0.01
EM avg. wait	0.25	0.01	0.01	0.00
GS avg. held	0.05	0.03	0.03	0.03
GS avg. wait	0.56	0.44	0.95	0.54
Total Events	17897	30088	37098	39138
Jobs Submitted	2955	4980	6133	6484
Servers	99	99	99	99
Dline/Priority	NO	NO	NO	NO
QUEUELEN	5	5	5	5
Duration	30	30	30	30

Note: In this table, X refers to the use of EQQUX000 and EQQUX002.

B.5 OPC Controller Started Task

The following JCL sample was used during the benchmark to run the OPC controller task using the controller options as defined in B.6, “OPC Controller Parameters” on page 43.

```
//OPC1      PROC INIT=CONTROL,PARMLIB=OPCESA.PARMLIB
//*************************************************************
//* OPC/ESA CONTROLLER
//*************************************************************
//OPCESA    EXEC PGM=EQQMAJOR,REGION=0M,PARM='&INIT',TIME=1440
//*************************************************************
//EQQJBLIB  DD  DISP=SHR,DSN=OPCESA.OVRLIB
//          DD  DISP=SHR,DSN=OPCESA.TEMP.OVERRIDE
//          DD  DISP=SHR,DSN=OPCESA.LV1ALIB
//          DD  DISP=SHR,DSN=OPCESA.LV1ALIB2
//          DD  DISP=SHR,DSN=OPCESA.IT1ALIB
//          DD  DISP=SHR,DSN=OPCESA.DB2.JOBLIB1
//          DD  DISP=SHR,DSN=OPCESA.LRDU.JOBLIB1
//          DD  DISP=SHR,DSN=OPCESA.ST9ALIB
//          DD  DISP=SHR,DSN=OPCESA.BD3ALIB
//          DD  DISP=SHR,DSN=OPCESA.MB2ALIB
//          DD  DISP=SHR,DSN=OPCESA.GWBLIB
//          DD  DISP=SHR,DSN=OPCESA.CLGLIB
//          DD  DISP=SHR,DSN=OPCESA.ENPLIB
//          DD  DISP=SHR,DSN=OPCESA.GAPLIB
//          DD  DISP=SHR,DSN=OPCESA.GENLIB
//          DD  DISP=SHR,DSN=OPCESA.ENDEVOR
//          DD  DISP=SHR,DSN=OPCESA.FASLIB
//          DD  DISP=SHR,DSN=OPCESA.INFOLIB
//          DD  DISP=SHR,DSN=OPCESA.NEWLIB1
//          DD  DISP=SHR,DSN=OPCESA.JOBLIBX
//          DD  DISP=SHR,DSN=OPCESA.JOBLIBT
//*************************************************************
//* DD CARDS FOR EQQUX002
//*************************************************************
//EQQJBLA0  DD  DISP=SHR,DSN=OPCESA.EQQJBLA0
//EQQJBLA1  DD  DISP=SHR,DSN=OPCESA.EQQJBLA1
//EQQJBLA2  DD  DISP=SHR,DSN=OPCESA.EQQJBLA2
//EQQJBLA3  DD  DISP=SHR,DSN=OPCESA.EQQJBLA3
//EQQJBLA4  DD  DISP=SHR,DSN=OPCESA.EQQJBLA4
//*
//EQQMLOG   DD  SYSOUT=S
//EQQMLIB   DD  DISP=SHR,DSN=OPCESA.INST.SEQQMSG0
//EQQPARM   DD  DISP=SHR,DSN=&PARMLIB
//EQQDUMP   DD  DISP=SHR,DSN=OPCESA.EQQDUMP
//EQQCKPT   DD  DISP=SHR,DSN=OPCESA.CKPT
//EQQBRDS   DD  SYSOUT=(A,INTRDR)
//EQQEVDS   DD  DISP=SHR,DSN=OPCESA.EVENT1
//EQQWSDS   DD  DISP=SHR,DSN=OPCESA.WS
//EQQADDS   DD  DISP=SHR,DSN=OPCESA.AD
//EQQRDDS   DD  DISP=SHR,DSN=OPCESA.RD
//EQQLTDS   DD  DISP=SHR,DSN=OPCESA.LT
//EQQJS1DS  DD  DISP=SHR,DSN=OPCESA.JS1,
//      AMP=('BUFND=10,BUFNI=10')
//EQQJS2DS  DD  DISP=SHR,DSN=OPCESA.JS2,
//      AMP=('BUFND=10,BUFNI=10')
//EQQOIDS   DD  DISP=SHR,DSN=OPCESA.OI
//EQQSIDS   DD  DISP=SHR,DSN=OPCESA.SI
```

```
//EQQCP1DS  DD  DISP=SHR,DSN=OPCESA.CP1,
//     AMP=('BUFND=10,BUFNI=10')
//EQQCP2DS  DD  DISP=SHR,DSN=OPCESA.CP2,
//     AMP=('BUFND=10,BUFNI=10')
//EQQCXDS   DD  DISP=SHR,DSN=OPCESA.CX
//EQQNCPDS  DD  DISP=SHR,DSN=OPCESA.NCP
//EQQNCXDS  DD  DISP=SHR,DSN=OPCESA.NCX
//EQQJTARC  DD  DISP=SHR,DSN=OPCESA.JTARC
//EQQJT01   DD  DISP=SHR,DSN=OPCESA.JT1
//EQQJT02   DD  DISP=SHR,DSN=OPCESA.JT2
//EQQJT03   DD  DISP=SHR,DSN=OPCESA.JT3
//EQQJT04   DD  DISP=SHR,DSN=OPCESA.JT4
//EQQJT05   DD  DISP=SHR,DSN=OPCESA.JT5
//EQQJT06   DD  DISP=SHR,DSN=OPCESA.JT6
//EQQPRLIB  DD  DISP=SHR,DSN=OPCESA.PRLIB
//EQQSTC    DD  DISP=SHR,DSN=OPCESA.STC
```

B.6 OPC Controller Parameters

The following parameters are a sample set to match the OPC controlling task defined in section B.5, “OPC Controller Started Task” on page 41.

```
JTOPTS   BACKUP(200000)
         JOBSUBMIT(YES)
         QUEUELEN(20)
         CURRPLAN(CURRENT)
         DUAL(NO)
         ETT(YES)
         HIGHRC(4095)
         JOBCHECK(SAME)
         JTLOGS(6)
         LIMFDBK(800)
         MAXJSFILE(NO)
         NEWOILIMIT(21)
         OFFDELAY(5)
         OUTPUTNODE(ANY)
         PLANSTART(8)
         PRTCOMPLETE(YES)
         SHUTDOWNPOLICY(110)
         SMOOTHING(50)
         STATMSG(CPLOCK,EVENTS,GENSERV)
         SUBFAILACTION(E)
         SUPPRESSACTION(C)
         SUPPRESSPOLICY(100)
         TRACK(JOBOPT)
         WSFAILURE(LEAVE,LEAVE,IMMED)
         WSOFFLINE(LEAVE,LEAVE,IMMED)
         NOERROR(TDBHKP*.X42HKPDP.DBU2D.0004,
                 TESTADD.*.*.0008,
                 X25*.*.*.0*,
                 IBMLEVEL.*.*.U25D,
                 TCPLOCAT.*.*.S222,
                 OJJJC%XT.*.*.*)

EXITS    CALL01(NO)
         CALL02(YES)    LOAD02(EQQUX0V2)
         CALL03(NO)
         CALL07(NO)
         CALL08(NO)
         CALL09(NO)
         CALL00(YES)


ALERTS   GENALERT()
         MLOG    (ERROROPER
                  OPCERROR
                  QLIMEXCEED)
         WTO     (ERROROPER
                  OPCERROR)

AUDIT    FILE(ALL)
         ACCESS(UPDATE)
         AMOUNT(DATA)

AUTHDEF  CLASSNAME(OPCCLASS)
         LISTLOGGING(FIRST)
```

```
         SUBRESOURCES(AD.OWNER
                      CP.OWNER
                      JV.OWNER
                      JS.OWNER
                      JS.WSNAME
                      RL.OWNER
                      RL.WSNAME)

DBCSOPTS APPLID(EBCDIC)
         OWNERID(EBCDIC)
         SORTORDER(KS)

INTFOPTS PIFCWB(OO)
         PIFHD(991231)

RESOPTS  ONERROR(FREESR)
         CONTENTIONTIME(0)
         DYNAMICADD(YES)
         LOOKAHEAD(0)

OPCOPTS  APPCTASK(YES)
         ERDRTASK(0)
         EWTRTASK(NO)
         NCFTASK(YES)
         NCFAPPL(OPCCONT1)
         GSTASK(5)
         GTABLE(DEFAULT)
         OPCHOST(YES)
         RECOVERY(YES)
         ARPARM(RECOVERY)
         VARSUB(SCAN)
         BUILDSSX(REBUILD)
         SSCMNAME(EQQSSCM4,TEMPORARY)

ROUTOPTS SNA(OPC1GBI1)
         XCF(EQQESC47,
             EQQESC48,
             EQQESC49,
             EQQESC50,
             EQQESC51,
             EQQESC52,
             EQQESC53,
             EQQESC54,
             EQQESC55,
             EQQESC56,
             EQQESC57,
             EQQESC58,
             EQQESC59,
             EQQESC60,
             EQQESC61,
             EQQESC62,
             EQQESC42,
             EQQESC43)

XCFOPTS  GROUP(OPC131) MEMBER(OPC1SC47)
```

B.7 OPC Tracker Started Task

The following JCL sample started an OPC MVS tracker during the benchmark.

```
//EQQE      PROC INIT=EQQE,SYS=OPCESA,
//          PARMLIB=OPCESA.PARMLIB
//*************************************************************
//* STARTED TASK PROCEDURE FOR OPC/ESA TRACKER * VERSION 1.2 *
//*************************************************************
//TRACKER   EXEC PGM=EQQMAJOR,REGION=4096K,
//          PARM='&INIT&SYSNAME',TIME=1440
//EQQMLOG   DD  SYSOUT=A
//EQQBRDS   DD  SYSOUT=(A,INTRDR)
//EQQPARM   DD  DISP=SHR,DSN=&PARMLIB
//EQQSTC    DD  DISP=(,PASS),DSN=&&T&SYSNAME,UNIT=SYSDA,
//          DCB=(RECFM=FB,LRECL=80,BLKSIZE=3280),
//          SPACE=(TRK,(1,1,1))
//EQQDUMP   DD  DUMMY
//EQQMLIB   DD  DISP=SHR,DSN=OPCESA.INST.SEQQMSG0
//EQQEVDS   DD  DISP=SHR,DSN=OPCESA.&SYS..&SYSNAME..EVENTE
//EQQEVD01  DD  DISP=SHR,DSN=OPCESA.&SYS..&SYSNAME..EVENTE
```

Appendix C. IBM Services

IBM Services can help you analyze your OPC system and assist you in implementing tuning solutions and exits.

You can contact your IBM local representative for more information.

Appendix D. Special Notices

This publication is intended to help OPC/ESA users in understanding the performance-related elements of the OPC/ESA product. It will enable the user to recognize any performance related problems and identify what can be changed to ensure optimum batch throughput. The information in this publication is not intended as the specification of any programming interfaces that are provided by OPC/ESA. See the PUBLICATIONS section of the IBM Programming Announcement for OPC/ESA for more information about what publications are considered to be product documentation.

References in this publication to IBM products, programs or services do not imply that IBM intends to make these available in all countries in which IBM operates. Any reference to an IBM product, program, or service is not intended to state or imply that only IBM′s product, program, or service may be used. Any functionally equivalent program that does not infringe any of IBM′s intellectual property rights may be used instead of the IBM product, program or service.

Information in this book was developed in conjunction with use of the equipment specified, and is limited in application to those specific hardware and software products and levels.

IBM may have patents or pending patent applications covering subject matter in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to the IBM Director of Licensing, IBM Corporation, 500 Columbus Avenue, Thornwood, NY 10594 USA.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact IBM Corporation, Dept. 600A, Mail Drop 1329, Somers, NY 10589 USA.

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The information contained in this document has not been submitted to any formal IBM test and is distributed AS IS. The information about non-IBM (″vendor″) products in this manual has been supplied by the vendor and IBM assumes no responsibility for its accuracy or completeness. The use of this information or the implementation of any of these techniques is a customer responsibility and depends on the customer′s ability to evaluate and integrate them into the customer′s operational environment. While each item may have been reviewed by IBM for accuracy in a specific situation, there is no guarantee that the same or similar results will be obtained elsewhere. Customers attempting to adapt these techniques to their own environments do so at their own risk.

Any performance data contained in this document was determined in a controlled environment, and therefore, the results that may be obtained in other operating environments may vary significantly. Users of this document should verify the applicable data for their specific environment.

Reference to PTF numbers that have not been released through the normal distribution process does not imply general availability. The purpose of including these reference numbers is to alert IBM customers to specific information relative to the implementation of the PTF when it becomes available to each customer according to the normal IBM PTF distribution process.

The following terms are trademarks of the International Business Machines Corporation in the United States and/or other countries:

IBM®
OPC
XT
MVS® (logo)
VTAM®

The following terms are trademarks of other companies:

C-bus is a trademark of Corollary, Inc.

Java and HotJava are trademarks of Sun Microsystems, Incorporated.

Microsoft, Windows, Windows NT, and the Windows 95 logo are trademarks or registered trademarks of Microsoft Corporation.

PC Direct is a trademark of Ziff Communications Company and is used by IBM Corporation under license.

Pentium, MMX, ProShare, LANDesk, and ActionMedia are trademarks or registered trademarks of Intel Corporation in the U.S. and other countries.

UNIX is a registered trademark in the United States and other countries licensed exclusively through X/Open Company Limited.

Other company, product, and service names may be trademarks or service marks of others.

Appendix E. Related Publications

The publications listed in this section are considered particularly suitable for a more detailed discussion of the topics covered in this redbook.

- *OPC/ESA Installation Guide*, SH19-4010
- *OPC/ESA Customization and Tuning*, SH19-4011
- *OPC/ESA Planning and Scheduling the Workload*, SH19-4012
- *OPC/ESA Controlling and Monitoring the Workload*, SH19-4013
- *OPC/ESA Programming Interfaces*, SH19-4014
- *OPC/ESA Diagnosis Guide and Reference*, LY19-6350 (available to IBM-licensed customers only)
- *OPC/ESA Messages and Codes*, SH19-6719
- *TME 10 OPC Installation Guide*, SH19-4379
- *TME 10 OPC Customization and Tuning*, SH19-4380
- *TME 10 OPC Planning and Scheduling the Workload* , SH19-4376
- *TME 10 OPC Controlling and Monitoring the Workload* , SH19-4377
- *TME 10 OPC Programming Interfaces*, SH19-4378
- *TME 10 OPC Diagnosis Guide and Reference*, LY19-6405 (available to IBM-licensed customers only)
- *TME 10 OPC Customization and Tuning*, SH19-4480
- *TME 10 OPC Messages and Codes*, SH19-6719

E.1 Redbooks on CD-ROMs

Redbooks are also available on CD-ROMs. **Order a subscription** and receive updates 2-4 times a year at significant savings.

CD-ROM Title	Subscription Number	Collection Kit Number
System/390 Redbooks Collection	SBOF-7201	SK2T-2177
Networking and Systems Management Redbooks Collection	SBOF-7370	SK2T-6022
Transaction Processing and Data Management Redbook	SBOF-7240	SK2T-8038
AS/400 Redbooks Collection	SBOF-7270	SK2T-2849
RS/6000 Redbooks Collection (HTML, BkMgr)	SBOF-7230	SK2T-8040
RS/6000 Redbooks Collection (PostScript)	SBOF-7205	SK2T-8041
Application Development Redbooks Collection	SBOF-7290	SK2T-8037
Personal Systems Redbooks Collection	SBOF-7250	SK2T-8042

How to Get ITSO Redbooks

This section explains how both customers and IBM employees can find out about ITSO redbooks, CD-ROMs, workshops, and residencies. A form for ordering books and CD-ROMs is also provided.

This information was current at the time of publication, but is continually subject to change. The latest information may be found at http://www.redbooks.ibm.com.

How IBM Employees Can Get ITSO Redbooks

Employees may request ITSO deliverables (redbooks, BookManager BOOKs, and CD-ROMs) and information about redbooks, workshops, and residencies in the following ways:

- **PUBORDER** — to order hardcopies in United States
- **GOPHER link to the Internet** - type GOPHER.WTSCPOK.ITSO.IBM.COM
- **Tools disks**

  To get LIST3820s of redbooks, type one of the following commands:

  ```
  TOOLS SENDTO EHONE4 TOOLS2 REDPRINT GET SG24xxxx PACKAGE
  TOOLS SENDTO CANVM2 TOOLS REDPRINT GET SG24xxxx PACKAGE (Canadian users only)
  ```

  To get BookManager BOOKs of redbooks, type the following command:

  ```
  TOOLCAT REDBOOKS
  ```

  To get lists of redbooks, type one of the following commands:

  ```
  TOOLS SENDTO USDIST MKTTOOLS MKTTOOLS GET ITSOCAT TXT
  TOOLS SENDTO USDIST MKTTOOLS MKTTOOLS GET LISTSERV PACKAGE
  ```

  To register for information on workshops, residencies, and redbooks, type the following command:

  ```
  TOOLS SENDTO WTSCPOK TOOLS ZDISK GET ITSOREGI 1998
  ```

  For a list of product area specialists in the ITSO: type the following command:

  ```
  TOOLS SENDTO WTSCPOK TOOLS ZDISK GET ORGCARD PACKAGE
  ```

- **Redbooks Web Site on the World Wide Web**

  http://w3.itso.ibm.com/redbooks

- **IBM Direct Publications Catalog on the World Wide Web**

  http://www.elink.ibmlink.ibm.com/pbl/pbl

  IBM employees may obtain LIST3820s of redbooks from this page.

- **REDBOOKS category on INEWS**
- **Online** — send orders to: USIB6FPL at IBMMAIL or DKIBMBSH at IBMMAIL
- **Internet Listserver**

  With an Internet e-mail address, anyone can subscribe to an IBM Announcement Listserver. To initiate the service, send an e-mail note to announce@webster.ibmlink.ibm.com with the keyword subscribe in the body of the note (leave the subject line blank). A category form and detailed instructions will be sent to you.

Redpieces

For information so current it is still in the process of being written, look at "Redpieces" on the Redbooks Web Site (http://www.redbooks.ibm.com/redpieces.htm). Redpieces are redbooks in progress; not all redbooks become redpieces, and sometimes just a few chapters will be published this way. The intent is to get the information out much quicker than the formal publishing process allows.

How Customers Can Get ITSO Redbooks

Customers may request ITSO deliverables (redbooks, BookManager BOOKs, and CD-ROMs) and information about redbooks, workshops, and residencies in the following ways:

- **Online Orders** — send orders to:

	IBMMAIL	**Internet**
In United States:	usib6fpl at ibmmail	usib6fpl@ibmmail.com
In Canada:	caibmbkz at ibmmail	lmannix@vnet.ibm.com
Outside North America:	dkibmbsh at ibmmail	bookshop@dk.ibm.com

- **Telephone orders**

United States (toll free)	1-800-879-2755
Canada (toll free)	1-800-IBM-4YOU
Outside North America	(long distance charges apply)
(+45) 4810-1320 - Danish	(+45) 4810-1020 - German
(+45) 4810-1420 - Dutch	(+45) 4810-1620 - Italian
(+45) 4810-1540 - English	(+45) 4810-1270 - Norwegian
(+45) 4810-1670 - Finnish	(+45) 4810-1120 - Spanish
(+45) 4810-1220 - French	(+45) 4810-1170 - Swedish

- **Mail Orders** — send orders to:

IBM Publications	IBM Publications	IBM Direct Services
Publications Customer Support	144-4th Avenue, S.W.	Sortemosevej 21
P.O. Box 29570	Calgary, Alberta T2P 3N5	DK-3450 Allerød
Raleigh, NC 27626-0570	Canada	Denmark
USA		

- **Fax** — send orders to:

United States (toll free)	1-800-445-9269
Canada	1-403-267-4455
Outside North America	(+45) 48 14 2207 (long distance charge)

- **1-800-IBM-4FAX (United States)** or **(+1)001-408-256-5422 (Outside USA)** — ask for:

  Index # 4421 Abstracts of new redbooks
  Index # 4422 IBM redbooks
  Index # 4420 Redbooks for last six months

- **Direct Services** - send note to softwareshop@vnet.ibm.com
- **On the World Wide Web**

Redbooks Web Site	http://www.redbooks.ibm.com
IBM Direct Publications Catalog	http://www.elink.ibmlink.ibm.com/pbl/pbl

- **Internet Listserver**

  With an Internet e-mail address, anyone can subscribe to an IBM Announcement Listserver. To initiate the service, send an e-mail note to announce@webster.ibmlink.ibm.com with the keyword subscribe in the body of the note (leave the subject line blank).

Redpieces

For information so current it is still in the process of being written, look at "Redpieces" on the Redbooks Web Site (http://www.redbooks.ibm.com/redpieces.htm). Redpieces are redbooks in progress; not all redbooks become redpieces, and sometimes just a few chapters will be published this way. The intent is to get the information out much quicker than the formal publishing process allows.

IBM Redbook Order Form

Please send me the following:

Title	Order Number	Quantity

First name Last name

Company

Address

City Postal code Country

Telephone number Telefax number VAT number

- Invoice to customer number
- Credit card number

Credit card expiration date Card issued to Signature

We accept American Express, Diners, Eurocard, Master Card, and Visa. Payment by credit card not available in all countries. Signature mandatory for credit card payment.

List of Abbreviations

CA	control area
CI	control interval
CP	current plan
DD	data definition
DOA	dynamic operation area
EM	Event Manager
EX	exception
GS	General Service task
IBM	International Business Machines Corporation
I/O	input/output
ITSO	International Technical Support Organization
JCC	job completion checker
JCL	job control language (MVS and VSE)
JES	job entry subsystem (MVS counterpart to VM′s RSCS)
LLA	library lookaside (MVS/ESA)
LLA	link-list lookaside (MVS/XA only)
MLOG	DFHSM log or journal record header
MVS	Multiple Virtual Storage (IBM System 370 & 390)
NCP	new current plan
NJE	network job entry
OPC	Operations Planning & Control (IBM program product)
OPC/ESA	Operations Planning & Control/Enterprise Systems Architecture (IBM)
PDS	partitioned data set
PIF	program interface
PROC	command procedure
SYSOUT	system output stream
SYSPLEX	systems complex
UNIX	an operating system developed at Bell Laboratories (trademark of UNIX System Laboratories, licensed exclusively by X/Open Company, Ltd.)
VSAM	Virtual Storage Access Method (IBM)
WS	work station
WSA	work station analyzer
WTO	write to operator

Index

F

G

H

I

J

K

L

M

U

V

W

X

Z

ITSO Redbook Evaluation

Maximizing Your OPC/ESA Throughput
SG24-2130-00

Your feedback is very important to help us maintain the quality of ITSO redbooks. **Please complete this questionnaire and return it using one of the following methods:**

- Use the online evaluation form found at http://www.redbooks.com
- Fax this form to: USA International Access Code + 1 914 432 8264
- Send your comments in an Internet note to redbook@vnet.ibm.com

Please rate your overall satisfaction with this book using the scale:
(1 = very good, 2 = good, 3 = average, 4 = poor, 5 = very poor)

Overall Satisfaction _____

Please answer the following questions:

Was this redbook published in time for your needs? Yes____ No____

If no, please explain:

_____

_____

_____

_____

What other redbooks would you like to see published?

_____

_____

_____

Comments/Suggestions: **(THANK YOU FOR YOUR FEEDBACK!)**

_____

_____

_____

_____

_____

Printed in U.S.A.