FORTRA

Robot Autotune 9

User Guide

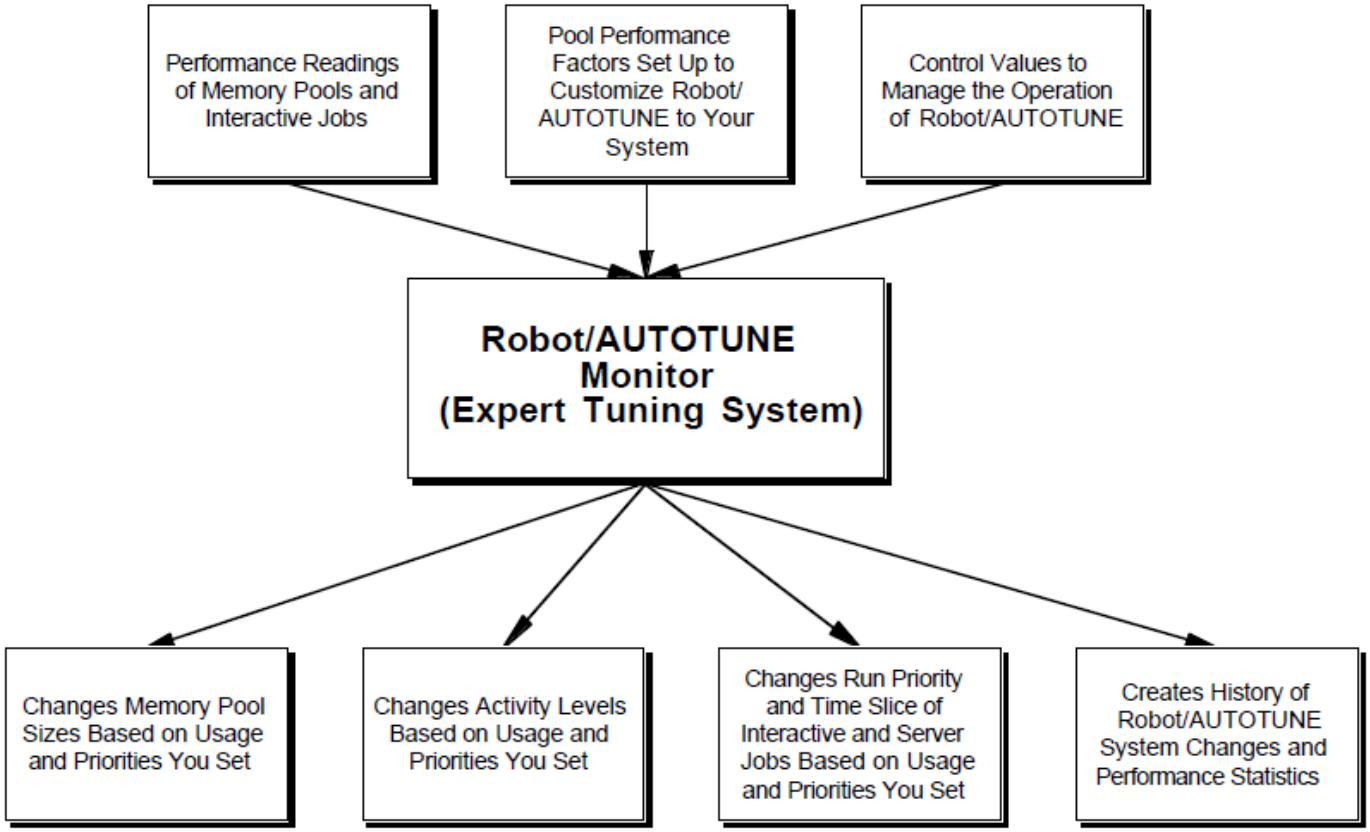
202308300207

Table of Contents

Table of Contents

Introduction

Robot Autotune System Flow Chart



Welcome to Robot Autotune, a software package for your IBM® eServer™ iSeries™ that automates performance tuning. When implemented properly, following this User Guide, Robot Autotune helps users get the system resources they need, when they need them. Users are happy because they enjoy consistently good response time.

■ Performance Tuning the Old-Fashioned Way

Why should you automate performance tuning? Because performance tuning the old-fashioned way is too labor-intensive.

If you have tried to do performance tuning on your own, you are already familiar with the following steps:

1. You read information about performance tuning.
2. You attend seminars about performance tuning.
3. You print boxes of paper to capture iSeries performance statistics so you can set pool sizes and activity levels.
4. You enter the commands to set pool sizes and activity levels.

RESULT: Your iSeries works great for about 45 minutes of the day. The rest of the day you still have dissatisfied users. Two months later, you go back to step one and start over.

■ Performance Tuning the Robot Autotune Way

To get out of this cycle, you need Robot Autotune.

1. Install Robot Autotune using the Installation Instructions.
2. Enter your performance factors using this User Guide.

RESULT: Your iSeries works great all day, every day! Your users think you are a genius. You and your staff never worry, think, or talk about performance tuning ever again.

How Does Robot Autotune Work?

Robot Autotune functions at the operating system level, making it fast and efficient. Our tests show that Robot Autotune uses only 15 seconds of CPU time per hour to tune the iSeries automatically. Robot Autotune takes performance readings at any interval you choose, and adjusts memory pool sizes and activity levels *automatically* to improve performance based on the system load and the priorities you set. If a pool is not using memory, the memory is moved to a pool where it's needed.

Special Performance Enhancement Options

Robot Autotune invented a breakthrough technique of isolating batch jobs in their own dynamic pools as they are submitted. When the job is finished, the pool is removed. These dynamic pools reduce the impact that batch jobs have on interactive users, while maximizing iSeries memory use.

As part of its dynamic pool process, Robot Autotune lets you manage job queues. Robot Autotune assigns job queues to a rotation group and allocates dynamic pools in rotation, so that each job queue gets an equal chance to execute. As jobs are started, ended, or placed on a job queue, Robot Autotune determines the number of available dynamic pools and allocates them to the job queues. As each job completes, a job in the next job queue in the rotation is eligible for processing, so jobs in lower-priority job queues get a chance to process.

Robot Autotune performs interactive job tuning by monitoring interactive jobs to see if a user is running a batch job interactively. When Robot Autotune finds such a job, it converts it to a "batch job" for the duration of the batch work by reducing the job's run priority and modifying its time slice. This decreases the impact of interactive batch jobs on other users' work.

Robot Autotune also manages batch (or server) jobs. It monitors their CPU usage and adjusts their run priority and time slice so they don't interfere with other users.

Jobs and Threads

Robot Autotune works with both jobs and threads on the iSeries to tune memory and maximize performance.

- An iSeries job is a process containing one or more threads, including an initial thread that is started when the job starts. Each thread has an identifier that is unique in the job to which the thread belongs. Job information is used for both the work entries and the maximum active counts associated with a subsystem.

- A thread, shorthand for thread of control, is an independent unit of execution within a program (all programs have at least one thread). A thread shares many of the resources assigned to the job, and can work in parallel with other threads to handle multiple requests simultaneously. Storage pool activity applies to threads rather than jobs, so Robot Autotune uses thread information to determine pool activity levels.

■ Dynamically Tunes LPAR Systems

In a partitioned (LPAR) iSeries system, Robot Autotune tunes main storage (memory) based on the amount allocated to the partition and your specifications. After each dynamic memory adjustment to the base pool (*BASE), Robot Autotune allocates the available memory throughout the partition for maximum performance and efficiency.

If you don't want to worry, think, or talk about performance tuning ever again, put Robot Autotune to work on your iSeries today. Your users will be grateful.

Notes:

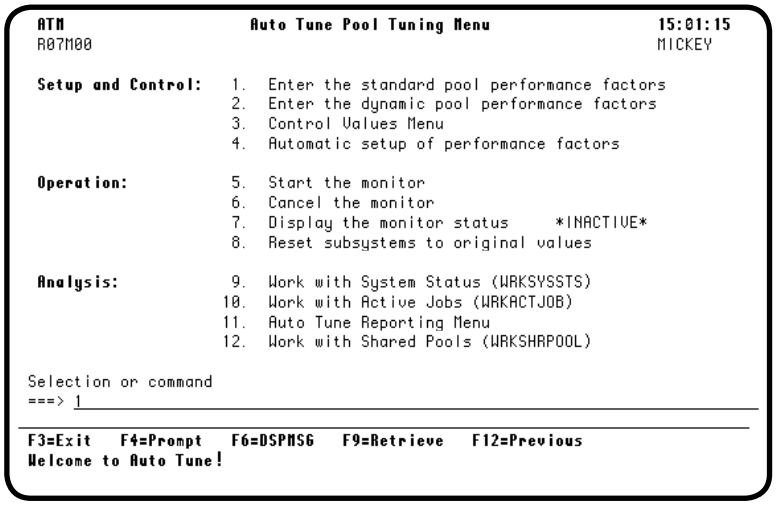
Getting Started

Starting Automatic Setup

After you install Robot Autotune, it determines automatically the number of workstations, printers, and batch jobs, if any, in each subsystem or shared pool. You will need to check this information and possibly make some changes.

Important

After you install Robot Autotune, you must customize your system's performance factors (explained on the following pages) to provide the system information Robot Autotune needs to function well. At a minimum, you must have (or set up) one pool for your workstations and another pool for your printers. If you are new to the iSeries, or aren't sure how to perform these steps, read the section "Getting Your iSeries Into Tunable Condition," later in this User Guide, before continuing.

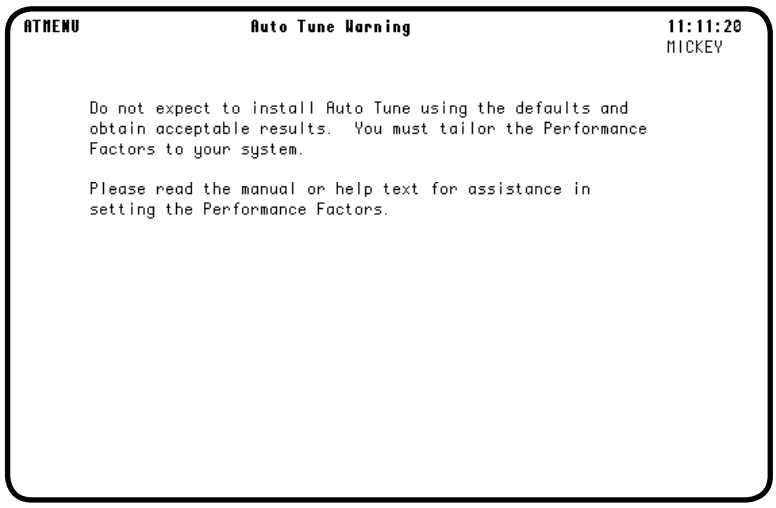


1. After installing Robot Autotune, enter the following command to display the Auto Tune Menu:

   ATLIB/ATM

   When the Auto Tune Menu first appears, the monitor status shown next to option 7 reads *INACTIVE*.

2. Enter a **1** on the command line to access automatic setup. The first time you select option 1, it takes you to the Auto Tune Automatic Setup panel. To return to this panel later, select option 4.
   Caution:
   If you use option 4, it overrides any of the standard pool performance factors you set using option 1.



3. Before the Automatic Setup panel displays for the first time, Robot Autotune displays a warning panel. This panel notifies you that Robot Autotune's automatic setup will probably **not** produce optimum performance for your system—you will need to fine tune the settings.
   Press Enter to continue.

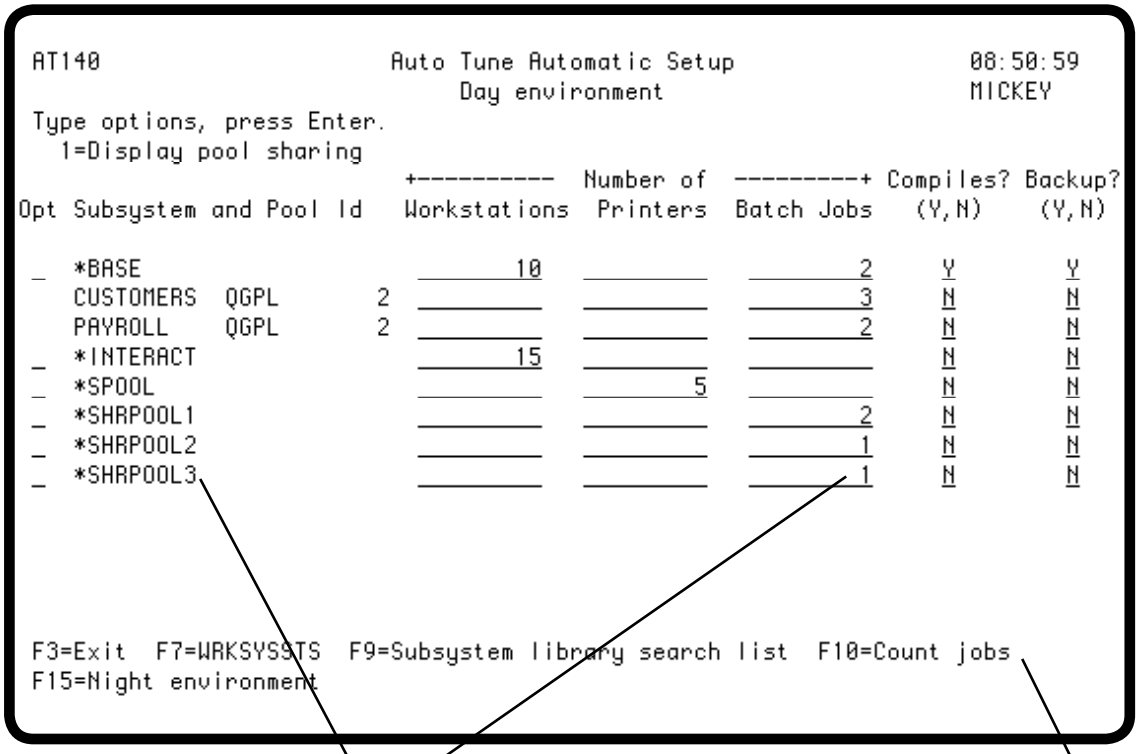
Reviewing Automatic Setup Values

When you initially started Automatic Setup, Robot Autotune located all active subsystem descriptions with defined pool IDs (both private and shared) on your iSeries. They are all listed on the Auto Tune Automatic Setup panel, displayed below.

In addition, Robot Autotune determined the number of threads within each active subsystem for each type of job: interactive (INT), printer (WTR), and batch (BCH, BCJ, ASJ, PJ, and all others). During Automatic Setup, Robot Autotune filled in the Automatic Setup panel using these values (see below).

Note: If you executed Automatic Setup for the first time when some of your subsystems were inactive and/or your system activity wasn't at a typical processing level, the number of threads counted may have given Robot Autotune an inaccurate representation of your system's tuning needs. Access Automatic Setup using option 4 and press **F10** with all subsystems active, during a period of normal system processing activity, if you want Robot Autotune to recalculate the number of threads used by your system.

Note: There must be one value for each subsystem or pool ID or you cannot exit from the panel. The value must be in the Workstations, Printers, or Batch Jobs field. If the fields are blank for a subsystem because the pool was not active, enter a value in one of the fields. You can adjust the value later, if necessary.

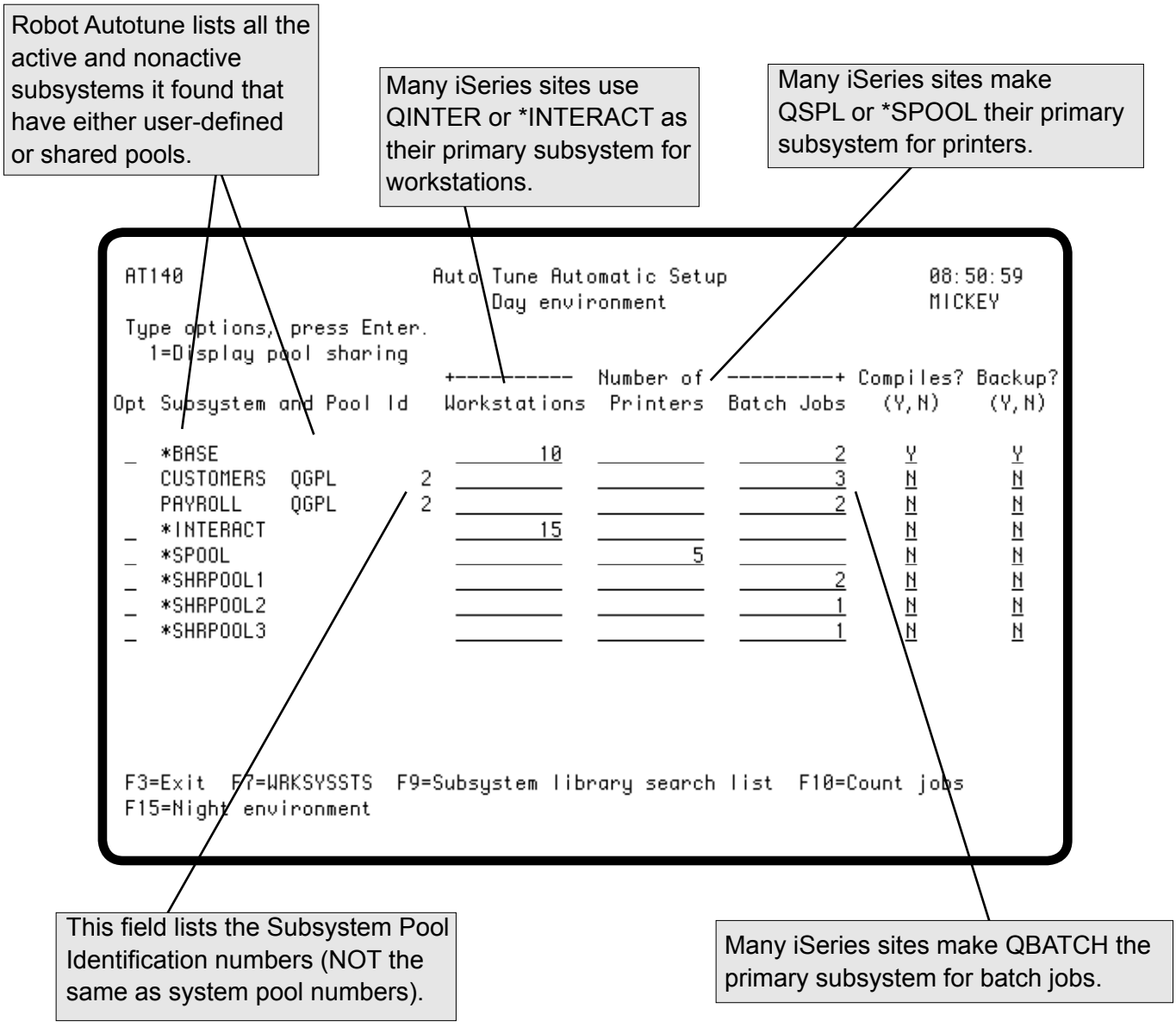


Each subsystem or pool ID must have a value on the Workstations, Printers, or Batch Jobs field. Enter a value if these fields are blank.

Press **F10** to recalculate the number of threads (not jobs) being used by your system. **Note:** If you press Enter, previously calculated standard pool performance values are overwritten with newly calculated values.

Reviewing Automatic Setup Values

The information below describes a typical iSeries setup of subsystem pools. Use this information to gain a better understanding of how to review the automatic setup values on the Auto Tune Automatic Setup panel.



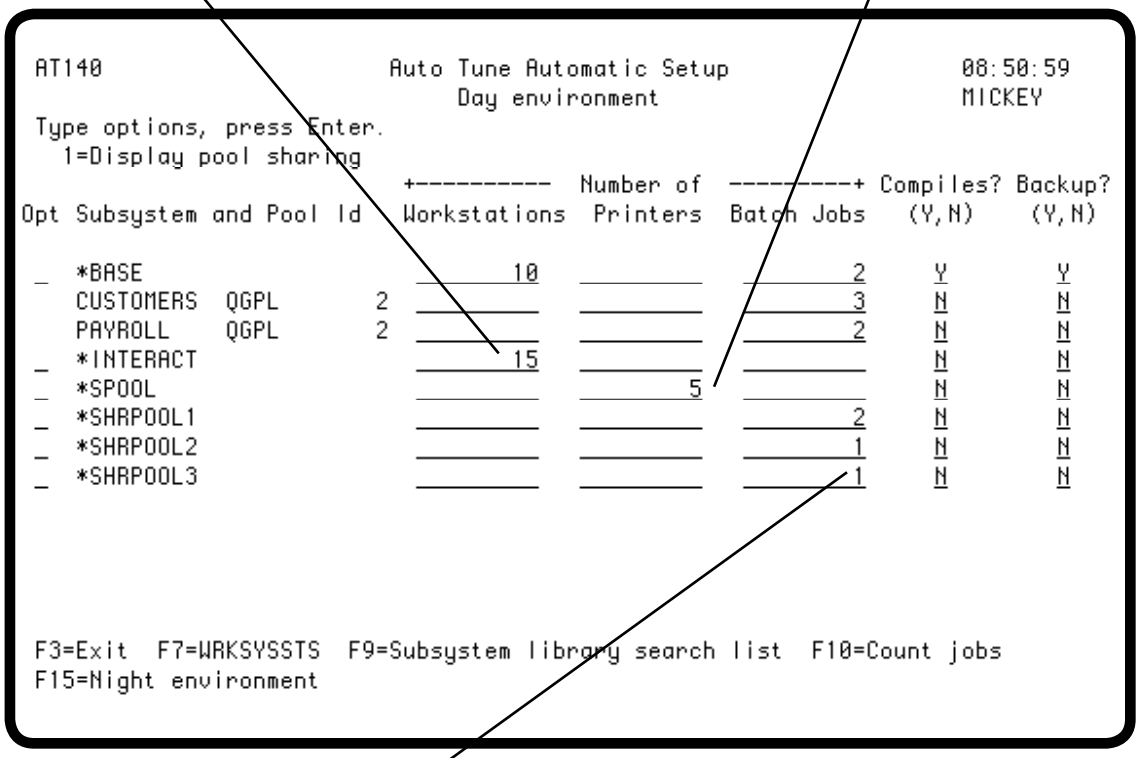
Reviewing Automatic Setup Values

With a better understanding of the Auto Tune Automatic Setup panel, you are ready to review the automatic setup values.

1. Review the number of job threads in the subsystem or pool for workstations, printers, and batch jobs.
2. Check the results of the Automatic Setup by entering the following commands: **WRKACTJOB** and **WRKSYSSTS**. These commands will give you an idea of the accuracy of the automatic setup values. If the values are not accurate, press F10 to recalculate the number of threads used by your system.

Review the total number of workstation job threads (type INT) operating in each subsystem. This entry indicates there are 15 interactive (INT) threads active in the *INTERACT pool.

Review the total number of printer threads (type WTR) operating in each system pool. This entry indicates there are 5 printer (WTR) threads active in the *SPOOL pool.

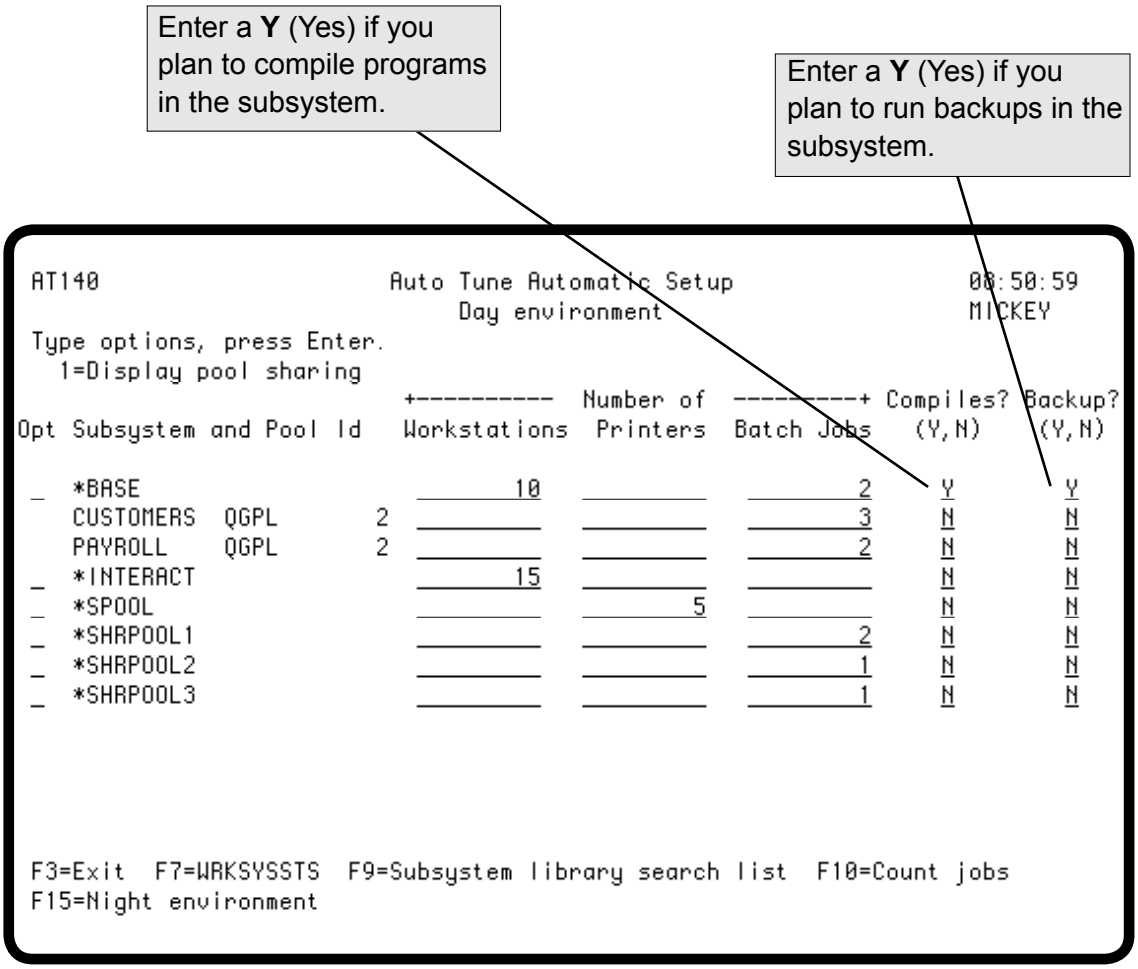


Review the total number of batch job threads that are active in each system pool (this covers all other types—BCH, BCJ, ASJ, PJ, and so forth.) This entry indicates there is 1 batch thread active in the *SHRPOOL3 pool.

Reviewing Automatic Setup Values

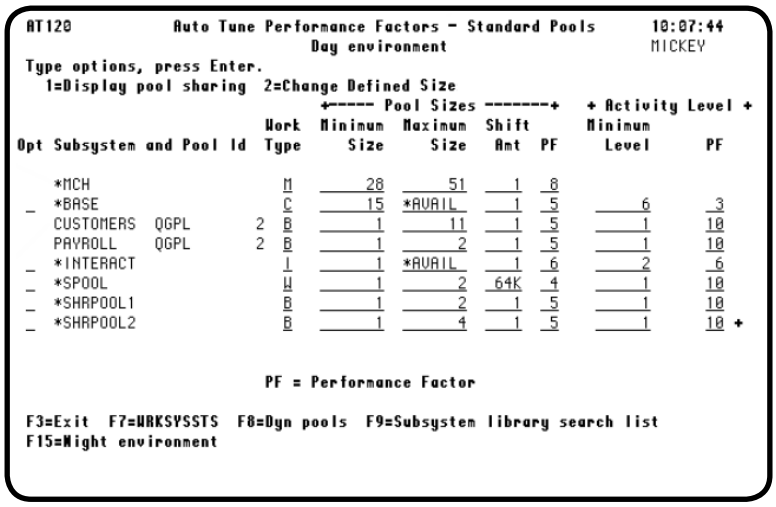
3. Indicate whether programs will be compiled, or disk backup jobs will be run, in each system pool.

4. If you're confident the automatic setup values are accurate and there is one value for each subsystem or shared pool, press Enter. Robot Autotune records the information and calculates performance factors and control values for each system pool based on this information. If you have a different night environment, press **F15** to display the data and repeat these steps.



Starting Robot Autotune

When it is finished calculating your system values, Robot Autotune displays these values on the Auto Tune Performance Factors - Standard Pools panel. The system values default to static values. You can enter percentage values, which may be more appropriate for LPAR systems, if you choose. You can start using these values to see how they work for your system. At any time you can return to this panel and change the performance factors and activity levels, even while Robot Autotune is running. Any changes you make are implemented with each Robot Autotune cycle, you do not need to stop and start Robot Autotune, or your system.

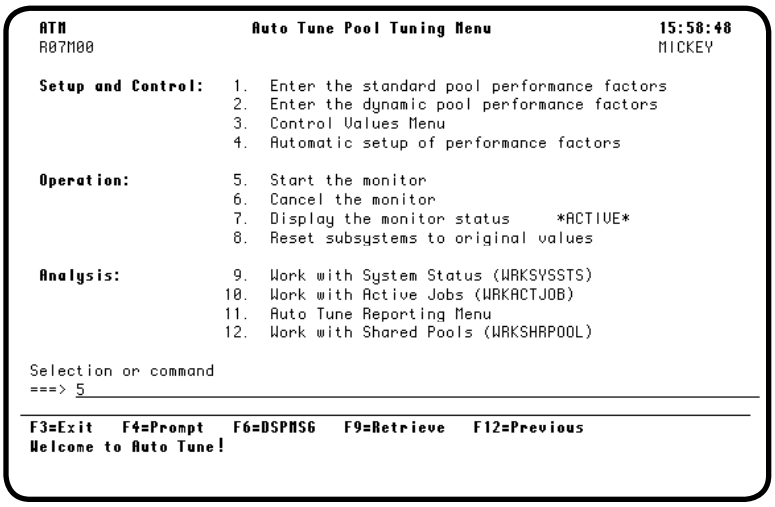


1. Press **F3** to return to the Auto Tune Menu.

   Important:

   Remember that you can fine tune the values you specify here by specifying either static or percentage adjustments (see the note) of memory for pool size factors. See the Setup and Control section for more information.

   Note: Percentage values are more appropriate for LPAR systems where memory changes dynamically.



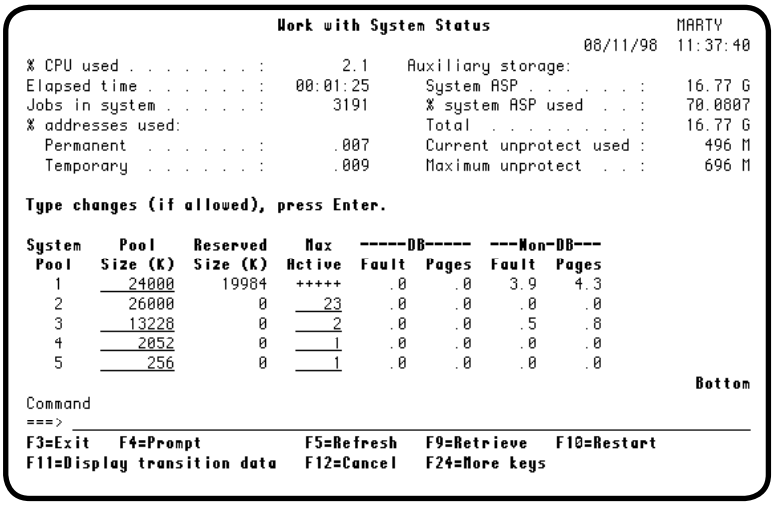
2. Enter a **5** on the command line to start Robot Autotune.

   As Robot Autotune starts, the status shown next to option 7 changes from *INACTIVE to *ACTIVE*.

Running Robot Autotune

Let Robot Autotune run for a day to determine whether you and your users are satisfied with your performance results. If not, adjust the performance factors and control values as described later in this User Guide. Soon Robot Autotune will be fully customized to your system and you won't have to worry about performance tuning.

Note: If you add memory to your system, you may need to change your performance factors, control values, or system values. See the Performance Refinement Guide, later in this User Guide, for more information.



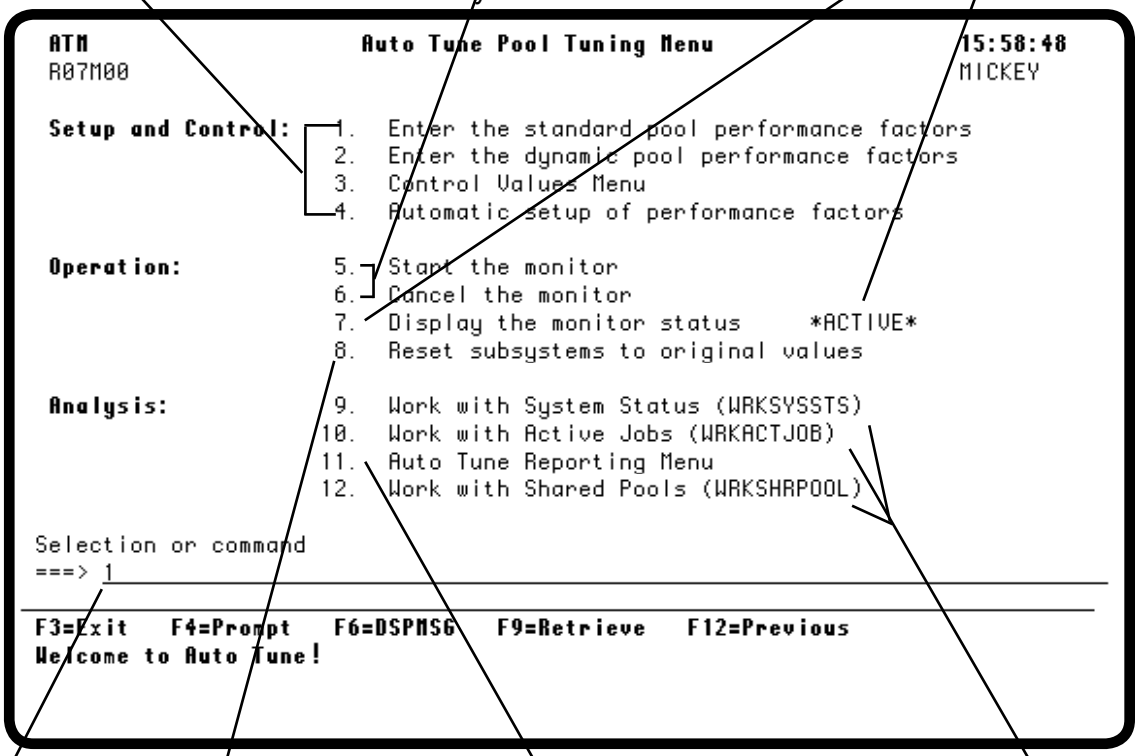
1. To watch Robot Autotune work, enter the command **WRKSYSSTS** on a command line. Or, if you are at the Auto Tune Menu, enter option **9**.

2. Press **F5** repeatedly to refresh the display and observe changes in pool sizes. These can be dramatic when the system load is changing. For example, lunch hour is a good time to see memory shift from interactive pools to batch pools.

Setup and Control

Displaying the Auto Tune Menu

After you have installed Robot Autotune and gone through the steps in the Getting Started section, you are ready to learn more detail about Robot Autotune. Enter the command **ATLIB/ATM** to display the Auto Tune Menu.



Use options **1** through **4** to set performance factors and control values to customize Robot Autotune to your system.

Use options **5** and **6** to start or stop the ATMONITOR and ATJOBQMGMT jobs in the ATMONITOR subsystem. Turn to the Operations section of this User Guide for more information about Robot Autotune operations, including how to perform this task automatically.

Use option **7** to display the ATMONITOR subsystem's jobs. The status of the ATMONITOR job in the ATMONITOR subsystem is always listed next to this option.

Enter your menu selection or a command on the command line.

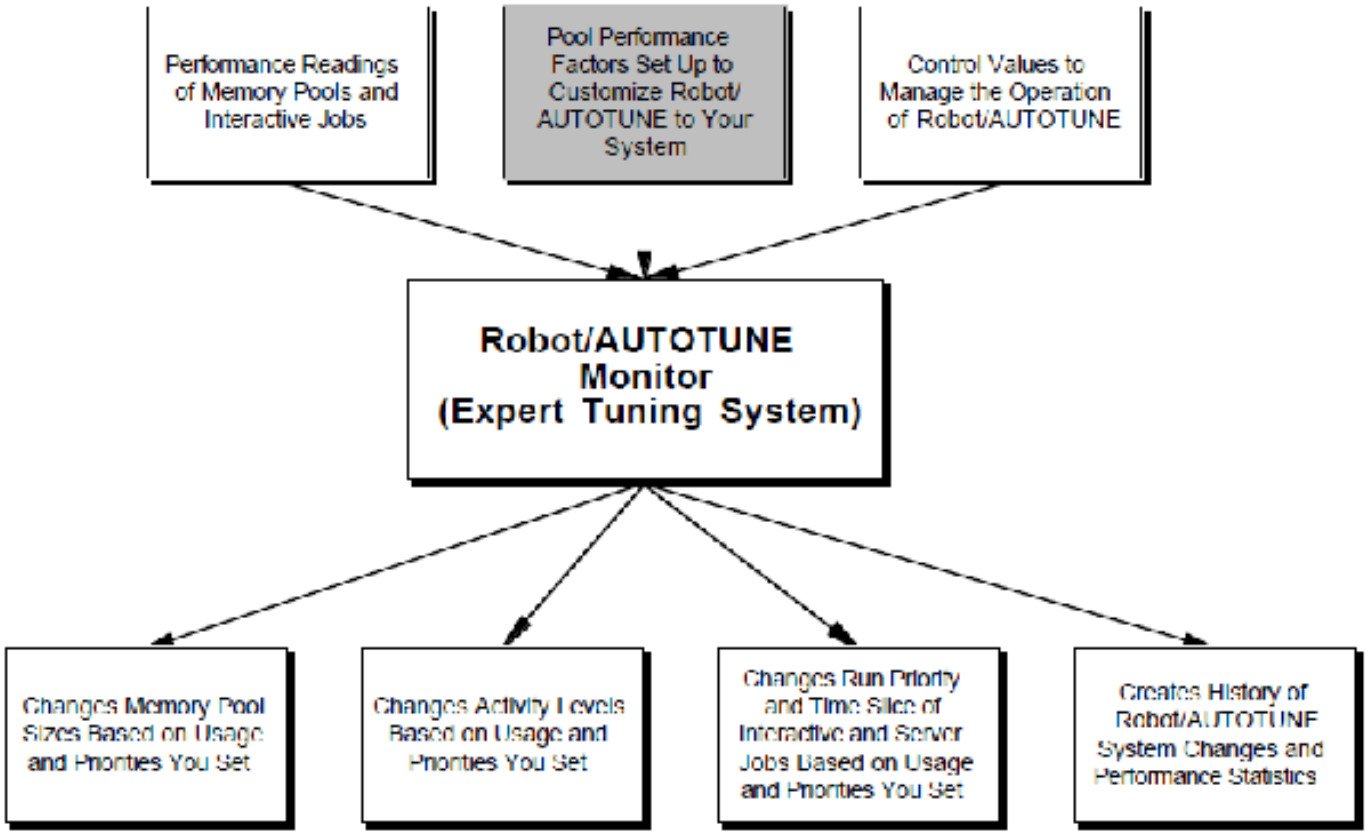
Use option **8** to immediately set pool sizes and activity levels back to the values in your subsystem descriptions.

Use option **11** to display the Auto Tune Reporting Menu. Use the menu to display or print statistics Robot Autotune has captured.

Use options **9**, **10**, and **12** to conveniently access these commonly used OS/400 commands.

Entering Performance Factors

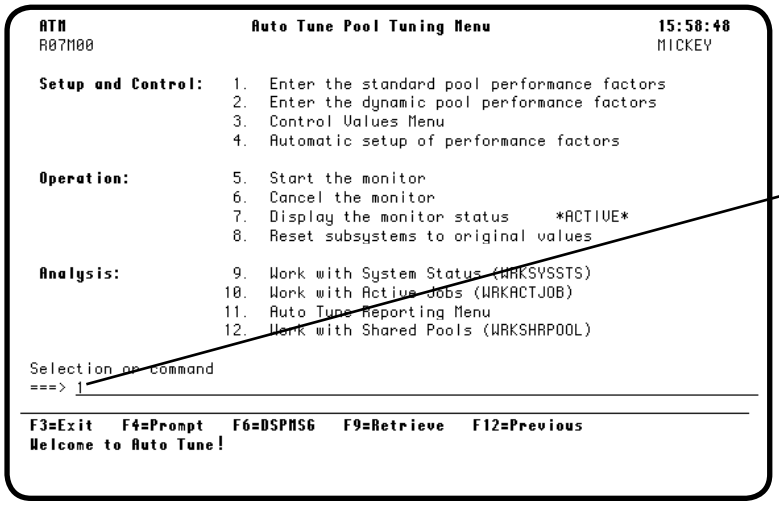
The first major section of the Auto Tune Menu allows you to set up your performance factors and control values. We will cover performance factors first.



Accessing the Standard Pools Performance Factors

The first time you select option **1** from the Auto Tune Menu, the Automatic Setup panel displays, as shown in the Getting Started section. After Robot Autotune is up and running, selecting option **1** displays the Auto Tune Performance Factors - Standard Pools panel. If you followed the steps in the Getting Started section of this User Guide, you have already seen this panel. Now you will use it in more detail and learn how to adjust the performance factors to customize Robot Autotune to your system.

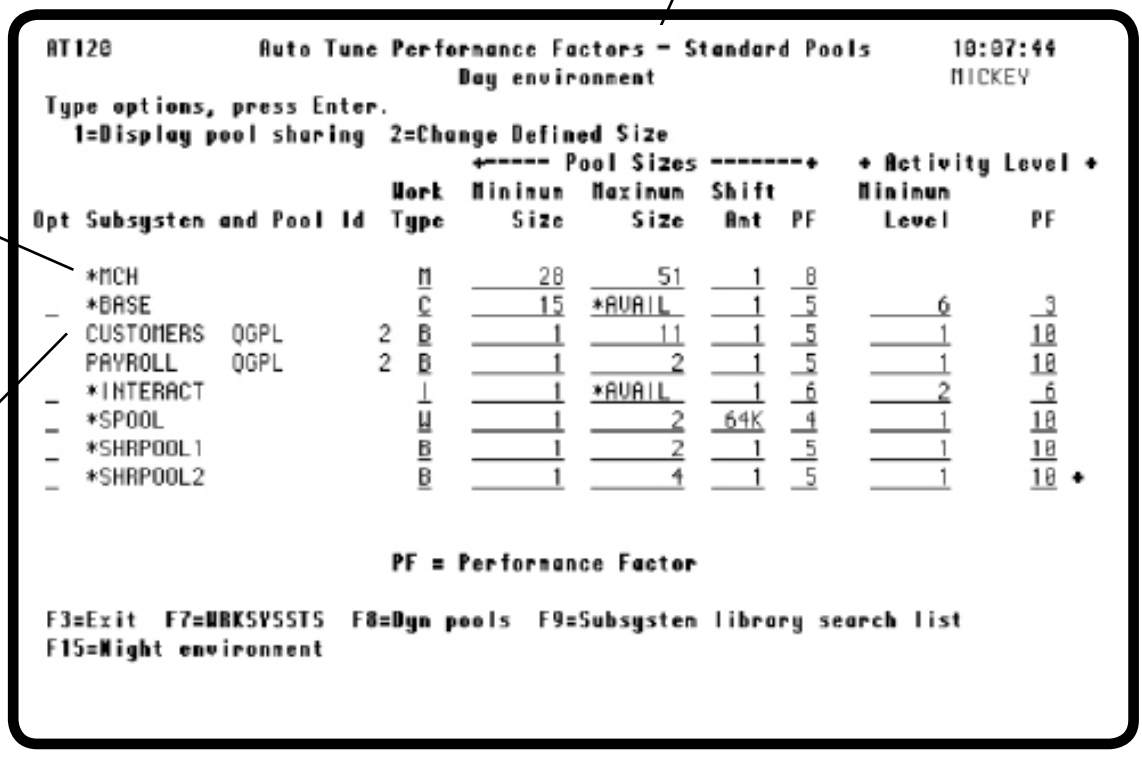
A Standard Pool is either a private pool or a shared pool. Pool names prefixed with an asterisk (*) are shared pools; pool names with no asterisk prefix are private pools. Robot Autotune treats private pools and shared pools the same way. For more information about pools, refer to the appropriate IBM documentation.



Enter a **1** on the command line to display the Auto Tune Performance Factors - Standard Pools panel.

An asterisk (*) before a pool name indicates that it is a shared

No asterisk in front of a pool name indicates it is a private



Setting Pool Size Factors

Robot Autotune looks at four factors related to pool size: minimum size, maximum size, shift amount, and performance factor.

When activity in a pool is low, Robot Autotune reduces the size of the pool. Conversely, when there is a lot of work to be done in a pool, Robot Autotune increases the size of the pool. It knows how large or small to make a pool because of the entries on the Auto Tune Performance Factors - Standard Pools panel. The minimum value shows Robot Autotune how small it can go; the maximum value, how large.

Static Versus Dynamic Pool Size Values

Robot Autotune allows you to specify memory pool size values—minimums, maximums, and shift amounts—in two different ways. You can specify a static amount of memory, in either kilobytes or megabytes. You can also specify minimum, maximum, and shift amount values as a percentage of the total memory available to the system, using any values from 1 to 80 (1% to 80%).

In a partitioned (LPAR) system, the amount of memory available to a partition can change dynamically based on the partition's workload and performance needs. As memory resources are added or removed dynamically, Robot Autotune adjusts the pool size values accordingly, based on your specifications. Therefore, you should use percentage performance factors on an LPAR system, where the amount of memory available changes dynamically.

Minimum Pool Size

Robot Autotune reduces the pool to its minimum size when activity is low. However, you don't want to set the minimum too low or users may experience delays in starting their jobs. It is especially important to set an adequate minimum pool size for the base pool (*BASE). If the base pool minimum is too low, system performance will be degraded no matter how much memory is in other pools. In addition, other batch subsystems that run save and restore functions should have a minimum size of at least 3 MB (or the equivalent percentage of system memory).

Robot Autotune will not allow the base pool to have a minimum size that is less than the OS/400 system value—QBASPOOL.

On the other hand, you don't want to set your minimum values too high or Robot Autotune will not have excess memory available to shift from pool to pool to meet changing work loads. The values suggested in the charts on the following pages are based on the experience of a cross-section of Robot Autotune users. They are recommended starting points only—you may adjust them to suit your configuration.

■ Maximum Pool Size

The only pools that need a maximum size are the machine pool and the pools that run batch jobs.

- The machine pool (*MCH) should have a maximum size of 40% of total CPU memory on small systems, and somewhat less on larger systems.

- In general, batch jobs will take as much memory as they can get, so you need to set a maximum to prevent them from monopolizing system resources. Because you can set the maximum to virtually any size (see the note), and the batch jobs will use it, you should choose the maximum based on the priority you want to give to these jobs. Batch jobs usually run more quickly in larger pools, so you may notice a significant improvement in job speed by setting a maximum size of more than 3 MB (or the equivalent percentage of system memory).

  Note: You cannot enter *AVAIL for the maximum pool size if the work type is B (batch).

■ Pool Size Shift Amount

How quickly Robot Autotune shrinks or expands a pool is determined by two related factors: the Pool Size Shift Amount and the tuning interval you select on the Auto Tune Control Values panel (control values and this panel are discussed later in this User Guide). For example, suppose you have a busy pool that could use more memory. If the shift amount for the pool is 10 MB (or the equivalent percentage of system memory) and your tuning interval is 10 seconds, Robot Autotune will shift 10 MB into the pool every 10 seconds until it has enough memory for the work, it reaches the maximum size for that pool, or the job completes.

If the pool size shift amount is too low, Robot Autotune will not be able to respond quickly to changes in workload. If the shift amount is too high, your system's performance may seem inconsistent.

■ Pool Size Performance Factor

The Pool Size Performance Factor (PF) and the type of work being performed in a pool determine the pool's priority when there is contention for memory. For example, a batch pool with a pool size performance factor of 5 has higher priority than an interactive pool with a PF of 5. The batch pool is assigned memory more quickly than the interactive pool because the Robot Autotune algorithm weighs *both* the PF and the work type in determining priority. If there is no contention for memory, these factors are ignored and the memory is moved where it is needed most.

Note: The machine pool (*MCH) should always have the highest Pool Size Performance Factor.

Start by assigning the values suggested in the Pool Size Factors Calculation Table that follows. If memory is not flowing where you want it to go, change the performance factor one step at a time until you achieve the desired results. The highest priority is 10; the lowest priority is 1.

When your interactive users are most active (usually during the day), it is generally wise to give *BASE and QINTER (or *INTERACT) the highest priority by assigning them a higher PF. Give batch jobs and printers a lower priority by assigning them a lower PF.

If you set your pool size performance factors this way and you still find that batch and printer pools are not being allocated enough memory, it indicates that workstations are putting an unusually large load on your system. This may mean that your interactive users are running batch jobs interactively. You can tune interactive jobs using the Control Values Menu (option 3 on the Auto Tune Menu). Tuning interactive jobs is discussed in detail later in this User Guide.

It is also possible that your server and batch jobs are consuming large amounts of system resources. Robot Autotune allows you to tune server jobs through the Control Values Menu. Tuning server jobs is also discussed later in this User Guide.

■ Setting Day and Night Priorities

You may want to set different priorities at night, giving batch jobs and printer pools a higher priority and/or larger pool sizes for faster processing. Make sure you set a maximum size for these pools for the reasons discussed earlier in Maximum Pool Size. You may also want to reduce the priority of the *BASE and QINTER (or *INTERACT) pools at night. As a result, Robot Autotune will give memory to the batch and printer pools up to the maximum pool size, regardless of the load in the interactive pool. When jobs are completed, memory is released and reassigned where it is needed. You can set up a daily priority change using the Control Menu by defining day and night values and setting a time to switch between them.

Pool Size Factors Calculation Table

Robot Autotune uses the values shown in the following table to calculate the initial pool settings from the information you enter during automatic setup. In addition, you can use this table as a guide to adjusting and refining pool values. It can be especially useful when you make changes to your system.

Subsystem(s)	Work Type	Minimum Pool Size	Maximum Pool Size	Shift Amount [1]	Perfomance Factor
*MCH (machine pool)	M	System value QMCHPOOL or 10% of total CPU memory[5]	40% of total CPU memory	1% of total CPU memory	8
*BASE (base pool with no batch work)	C	5-8% of total CPU memory[2]	Default (no maximum) *AVAIL	1% of total CPU memory	6
*BASE (base pool with batch work)	C	11-14% of total CPU memory [2]	Default (no maximum) *AVAIL	1% of total CPU memory	4
*INTERACT or QINTER (interactive workstation pool)	I	Minimum number of workstations in pool multiplied by 16K. The minimum is 660 K; the default is 1 MB.	Default (no maximum) *AVAIL	1% of total CPU memory	6
*SPOOL or QSPL (printer work)	W	Number of AFP printers in pool multiplied by 200 K (the minimum is 1500 K) plus the number of non-AFP printers in pool multiplied by 80 K. The minimum is 256 K; the default is 1 MB.	3 MB	100K	4
QBATCH (batch work or dynamic pools) [3]	B	The minimum is 300 K (2 MB if batch saves are used). The default is 1 MB.	The number of threads in the pool multiplied by 2 to 3 MB [4]	1% of total CPU memory	5
QCMN (communications work), QSERVER, QSYSWRK, QUSRWRK	C	The minimum number of threads in the pool multiplied by 16 K. The minimum is 300 K; the default is 1 MB.	Either the number of threads multiplied by 120 K, or 20% of the total CPU memory, whichever is smaller.	1% of total CPU memory	6

The following legend describes the job Work Types listed in the table:

Type	**Description**
B	Batch
C	Combined (both batch and interactive)
I	Interactive
M	Machine pool
W	Writer (printer, or spool)

Pool Size Factors Calculation Table—Notes

The following notes apply to the superscript entries in the Pool Size Factors Calculation Table.

Notes:

[1] The shift amount must be a minimum of 8K.

[2] The minimum pool size you set for *BASE should be greater than the QBASPOOL system value. This provides a buffer that is always available when allocating new memory pools. Without this buffer, if the amount of *BASE is adjusted downward to the QBASPOOL amount and a subsystem is started, the pool allocation may fail causing all of the subsystem work to be performed in *BASE. In an LPAR environment, having this buffer also helps when storage is shifted out of this partition.

[3] See the Batch job Storage Guidelines for more detailed information on specifying pool sizes for batch jobs.

[4] If only one job is processed in this pool at a time, set this maximum to 5 MB. However, you may want a separate pool for big save/restore jobs with a maximum of 25 MB.

[5] A warning message will be generated if you maintain a percentage less than ten percent. Large systems may not require 10%.

Batch Job Storage Guidelines

Use the following values to help you specify the pool size for various types of batch jobs. By setting your pool sizes to the proper levels, your jobs will process more efficiently and you reduce the chance that they will end abnormally.

Batch and Job Type	Initial Storage	Comments
Short-running Production	1 MB	Can require up to 3 MB
Long-running Production	3 MB	Run better with 25 MB
Compiles	15 MB	Run better with 25 MB
Reformat/Sort	2 MB	Smaller sorts may run with 2 MB; larger sorts may require more memory
Queries	2 MB	Larger queries may require more memory
Save/Restore	2 MB	Some save operations run with 1 MB; others may require 3 MB

Setting Activity Levels

Robot Autotune looks at two factors related to activity level on the Auto Tune Performance Factors - Standard Pools panel—the *minimum activity level* and the *activity level performance factor*.

Minimum Activity Level

The *minimum activity level* specifies the number of threads in a pool that can use CPU resources at any one time. Robot Autotune adjusts activity levels in batch and interactive pools differently.

- In batch pools, the activity level reflects a direct relationship between the number of threads in the pool and the activity level performance factor.
- In interactive pools, the activity level reflects a floating relationship among the active-to-wait transition rate, the wait-to-ineligible transition rate, and the activity level performance factor.

Activity Level Performance Factor

The *activity level performance factor* determines how quickly the activity level will change when workload in a pool changes. The activity level performance factor is an absolute number (rather than a relative priority) and has no relation to the activity level performance factor in other pools. A higher performance factor gives a pool better performance. The lowest activity level performance factor you can assign is 0 (zero), which causes no changes to be allowed; the highest is 10. If the activity level is too high, you will see excessive faulting and paging for the pool. When you lower the activity level, the ineligible count will rise, which is preferable to too high a level of faulting and paging.

Use the following table as a starting point for recommendations for minimum activity level and performance factor.

Subsystem and Type of Work	Minimum Activity Level	Performance Factor
*BASE (no batch work)	15% of threads	3
*INTERACT or QINTER (interactive workstation pool)	25% of the number of workstations in the pool	3
*SPOOL or QSPL (printer work)	1 (up to 5 printers) or 5 (5 or more printers)	5^
QBATCH (batch jobs or dynamic pools)	1	10
QCMN, QSERVER, QSYSWRK, QUSRWRK (communications work)	10-15% of the number of workstations in the pool	3

^ *The more printers you have, the lower the Performance Factor number should be.*

■ Printing Pool Size and Activity Level Factors

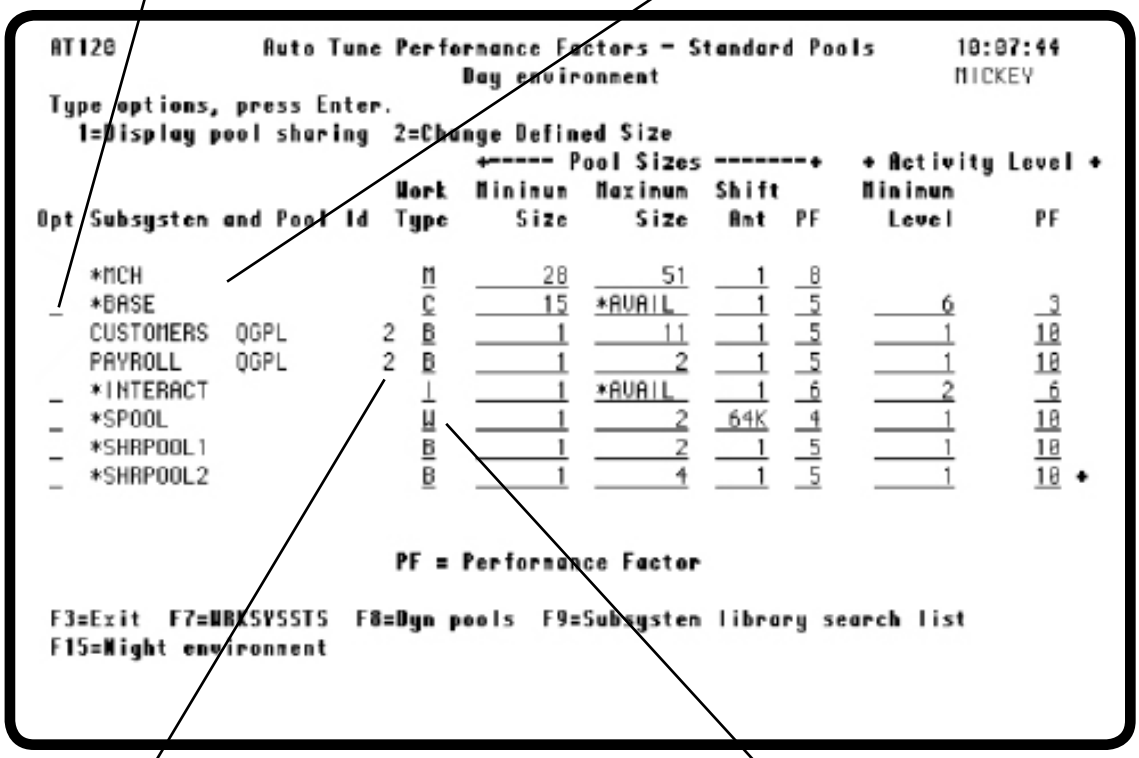
You can print a hard copy of your pool size and activity level factors. First, select option **11** on the Auto Tune Menu to display the Reporting Menu. Then, select the Performance factors/control values report from this menu. Or, make sure ATLIB is in your library list and enter **Call AT122** on a command line.

Entering Robot Autotune Performance Factors for Standard Pools

If you followed the steps in the Getting Started section of this User Guide, Robot Autotune performed an automatic setup for your system and filled in the values on the Performance Factors - Standard Pools panel. Now that you've read the discussions on the preceding pages, you can start adjusting these values intelligently. Just type over any value you want to change and press Enter to record your changes. Robot Autotune implements your changes at the next tuning interval after you exit the panel.

If there is an **Opt** field next to a pool name it is a shared pool. If there is no **Opt** field, it is not a shared pool. You can enter a **1** to display the subsystems that share the pool. Enter a **2** to define a specific Defined Size and Defined Activity Level for the shared pool.

A list of all the subsystem pool IDs that Robot Autotune found on your system. This includes both active and nonactive subsystems that have private or shared pools and a size greater than zero.



The Pool ID number shows the subsystem pool ID (this is not the same as the system pool ID).

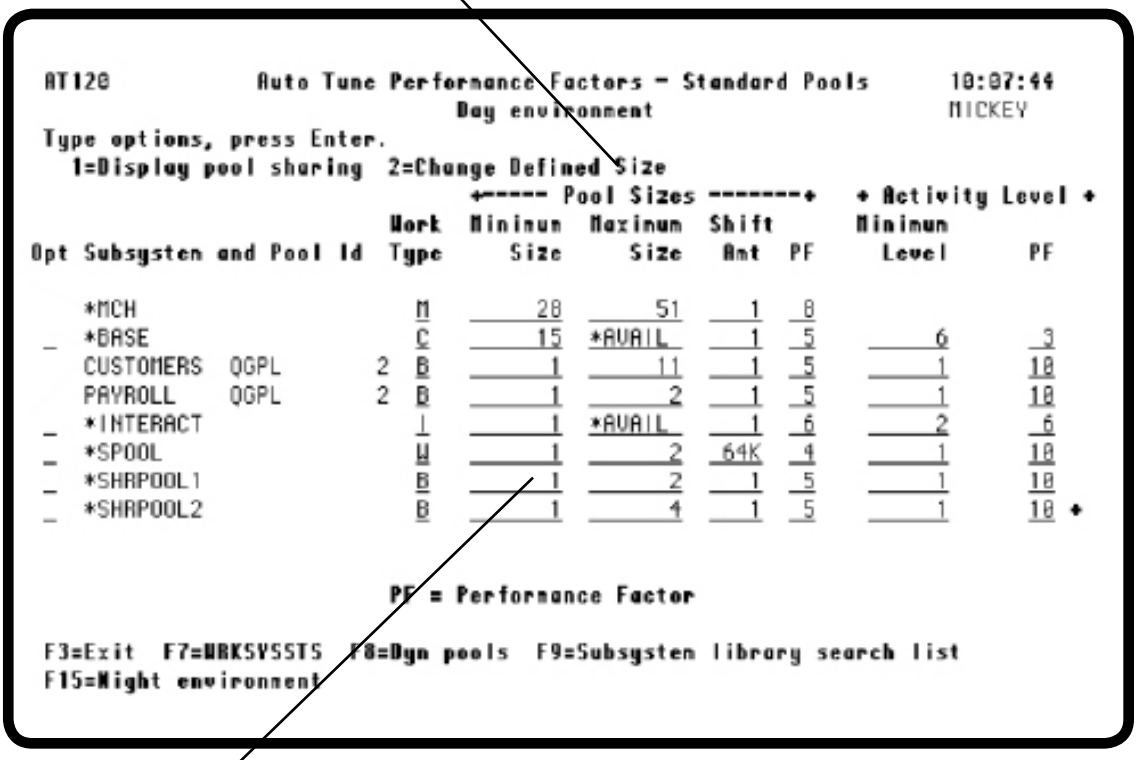
The work type indicates the type of jobs in the pool. The possible work types are:

- **C Combined** (both batch and interactive jobs)
- **B Batch**
- **I Interactive**
- **M Machine**
- **R Readers**
- **W Writers**

Entering Robot Autotune Performance Factors for Standard Pools

You can enter values for all fields that specify memory amounts—pool size minimum, pool size maximum, and pool shift amount—in megabytes (MB), kilobytes (KB), or as a percentage (%) of the total amount of memory defined for this partition. In addition, you can use different kinds of values in different fields. For example, you could specify the minimum pool size as a fixed amount of memory in MB or KB, and the maximum size as a percentage value. See the Setting Activity Levels topic for more information.

- To specify an amount in megabytes (MB), do not type anything following the value. If nothing follows the value specified, Robot Autotune assumes the value is in megabytes.
- To specify an amount in kilobytes (KB), enter a K following the value. If you specify a kilobyte number greater than 1023 K, Robot Autotune converts it to megabytes (MB), rounding down to the next lower value (for example, 1500 K is converted to 1 MB).
- To specify a percentage amount, enter a percentage sign (%) following the value. To check the amount available to the pool, use the WRKSHRPOOL command.



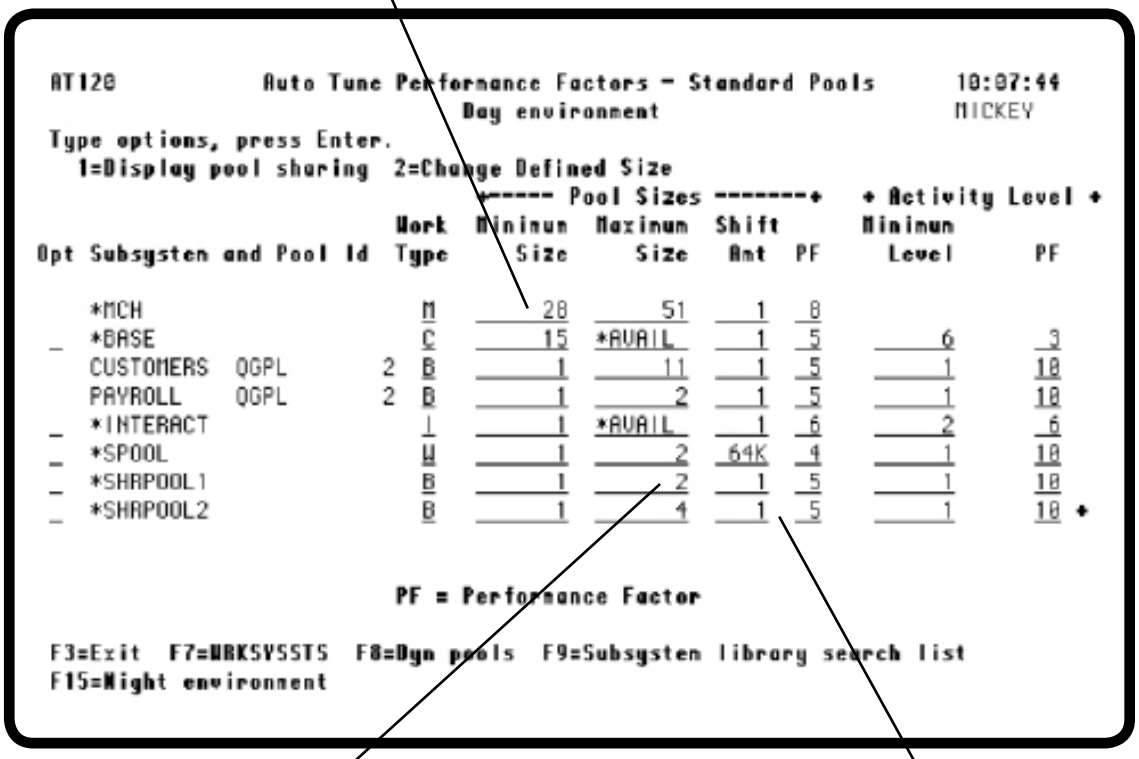
```
AT120          Auto Tune Performance Factors - Standard Pools      10:07:44
                          Day environment                          MICKEY
Type options, press Enter.
  1=Display pool sharing  2=Change Defined Size
                              +----- Pool Sizes --------+   + Activity Level +
                             Work  Minimum  Maximum  Shift       Minimum
Opt Subsystem and Pool Id    Type     Size     Size    Amt  PF     Level      PF

    *MCH                      M       28       51      1   8
_   *BASE                     C       15    *AVAIL     1   5         6       3
    CUSTOMERS  QGPL        2  B        1       11      1   5         1      10
    PAYROLL    QGPL        2  B        1        2      1   5         1      10
_   *INTERACT                 I        1    *AVAIL     1   6         2       6
_   *SPOOL                    W        1        2    64K   4         1      10
_   *SHRPOOL1                 B        1        2      1   5         1      10
_   *SHRPOOL2                 B        1        4      1   5         1      10 +

                          PF = Performance Factor

F3=Exit  F7=WRKSYSSTS  F8=Dyn pools  F9=Subsystem library search list
F15=Night environment
```

Because nothing follows the value specified (such as a K for kilobytes, or a percentage sign [%] for percentage), Robot Autotune assumes the values are in MB (the default).

Entering Robot Autotune Performance Factors for Standard Pools

The minimum size to which Robot Autotune will reduce the pool.

Note: You cannot set the minimum size for the base pool (*BASE) lower than the system value QBASPOOL. If you need to decrease the minimum size of this pool, first change the system value by using the command CHGSYSVAL QBASPOOL and then change the minimum size to match it in Robot Autotune.



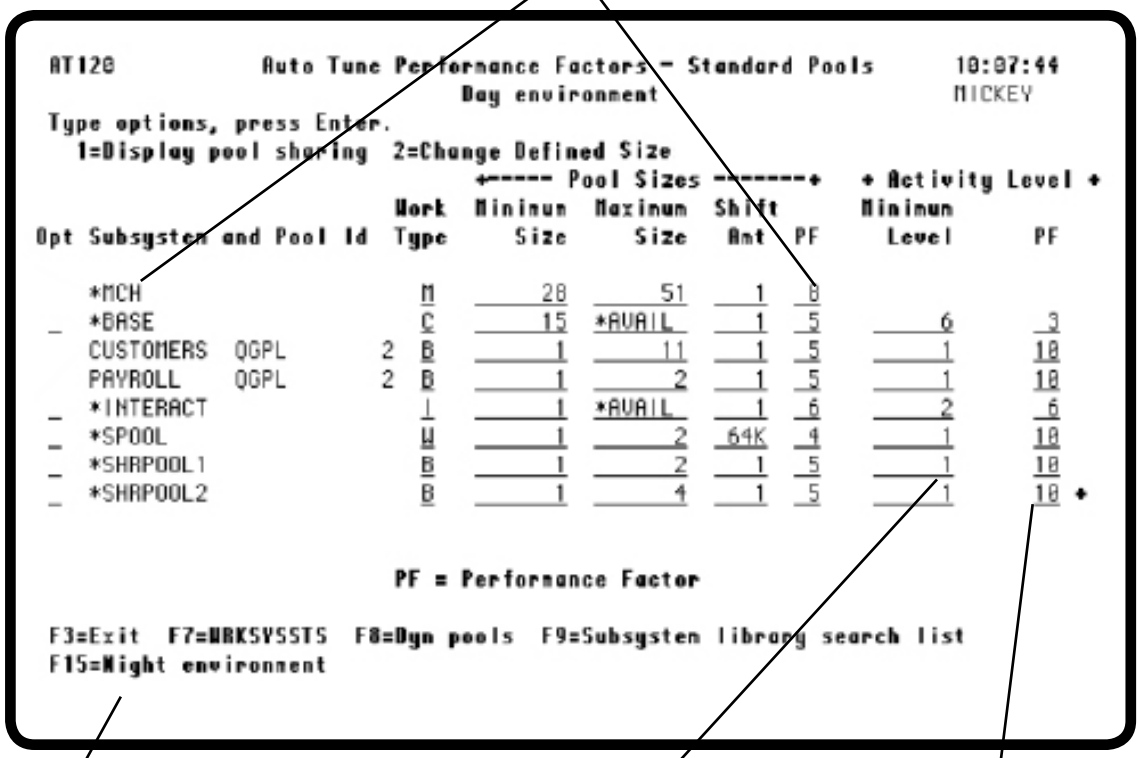
The maximum size to which Robot Autotune will increase the pool, specified in MB (blank following the value). See the discussion and table earlier for more information on setting this value.

The amount of memory Robot Autotune will shift when an adjustment is needed. See the discussion and table earlier for more information on setting this value.

Note: Since this is a 4-character field, if you specify kilobytes, you cannot specify a number greater than 999K because the K uses one digit.

Entering Robot Autotune Performance Factors for Standard Pools

Robot Autotune uses the Pool Size Performance Factor with the Work Type to determine which pool should have priority when there is contention for memory. The lowest value is 1; the highest is 10. If you want to give pools of the same work type the same priority, give them the same performance factor (see the discussion and table earlier for more information on setting this value). **Note:** The machine pool (*MCH) should always have the largest performance factor number on the system. If the machine pool does not have enough memory, system performance will be seriously degraded.



Press **F15** to display and edit Standard Pools Performance Factors for the night environment.

The minimum Activity Level specifies the number of threads in a pool that can use CPU resources at any one time. See the Pool Size topics and Pool Size Factors Calculation table earlier for more information on setting this value.

The Activity Level Performance Factor values range from zero (0) to 10. This factor is absolute—the higher the value, the higher the activity level. Enter a zero (0) if you want Robot Autotune to keep the activity level at the minimum level. See the Activity Level topics and Activity Level Performance table earlier for more information on setting this value.

Displaying the Subsystems that Share a Pool

Enter a **1** in the **Opt** field to display a list showing the names of the subsystems that share a pool. The example below shows the subsystems that share the *BASE pool.

Enter a **1** in the **Opt** field to display a list of subsystems that share the pool. If there is no Opt field next to a pool, the pool is not shared.

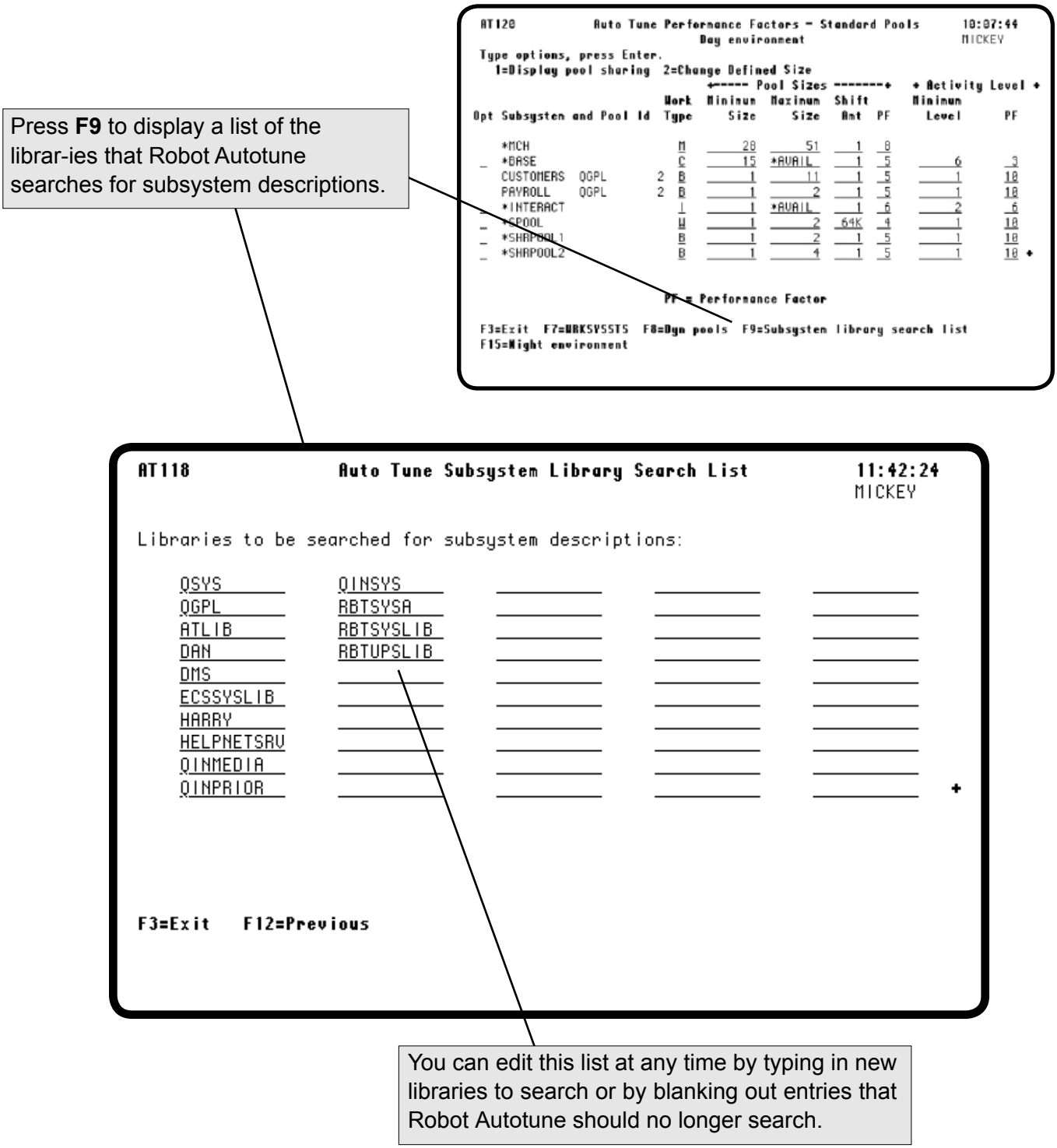
```
AT120          Auto Tune Performance Factors - Standard Pools      10:07:44
                          Day environment                          MICKEY
Type options, press Enter.
  1=Display pool sharing  2=Change Defined Size
                                 +----- Pool Sizes --------+  + Activity Level +
                                 Work  Minimum  Maximum  Shift      Minimum
Opt Subsystem and Pool Id        Type     Size     Size    Amt  PF    Level      PF
    *MCH                          M       20       51     1   8
1   *BASE                         C       15   *AVAIL     1   5        6       3
    CUSTOMERS  QGPL        2      B        1       11     1   5        1      10
    PAYROLL    QGPL        2      B        1        2     1   5        1      10
_   *INTERACT                     I        1   *AVAIL     1   6        2       6
_   *SPOOL                        W        1        2   64K   4        1      10
_   *SHRPOOL1                     B        1        2     1   5        1      10
_   *SHRPOOL2                     B        1        4     1   5        1      10 +

                          PF = Performance Factor

F3=Exit  F7=WRKSYSSTS  F8=Dyn pools  F9=Subsystem library search list
F15=Night environment
```

```
AT120                    Auto Tune Pool Sharing                  10:08:34
                                                                 MICKEY

        Shared Pool     Subsystem

         *BASE          CUSTOMERS   QGPL        1
                        PAYROLL     QGPL        1
                        QBASE       QINMEDIA    1
                        QBASE       QINSYS      1
                        QBASE       QSYS        1
                        QBATCH      QINMEDIA    1
                        QBATCH      QINSYS      1
                        QBATCH      QSYS        1                    +


F3=Exit    F12=Previous
```

The subsystem names, the libraries where the subsystems are found, and subsystem pool IDs.

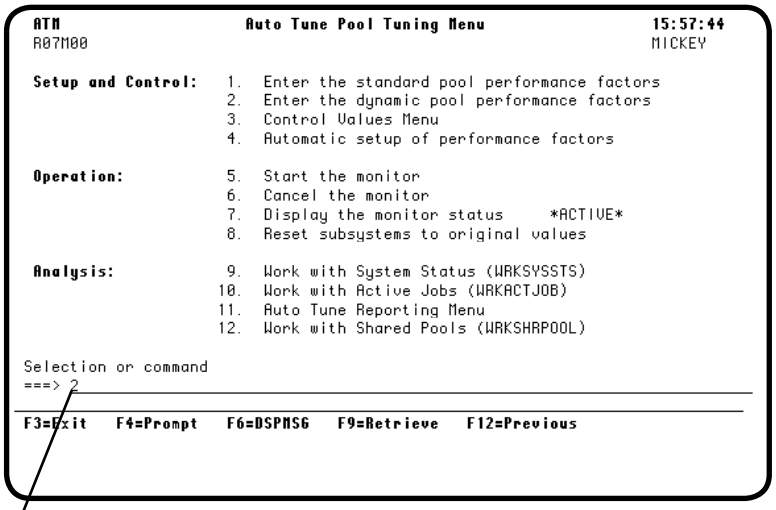
Displaying the Libraries that Robot Autotune Searches

Press **F9** to display a list of the libraries that Robot Autotune searches for subsystem descriptions. You can modify entries on this list at any time. The Auto Tune Subsystem Library Search List panel has two additional purposes. It allows you to add new subsystems/libraries after Robot Autotune is installed and, it lets you remove subsystems/libraries that Robot Autotune picked up automatically during first-time processing that you do not want Robot Autotune to search.

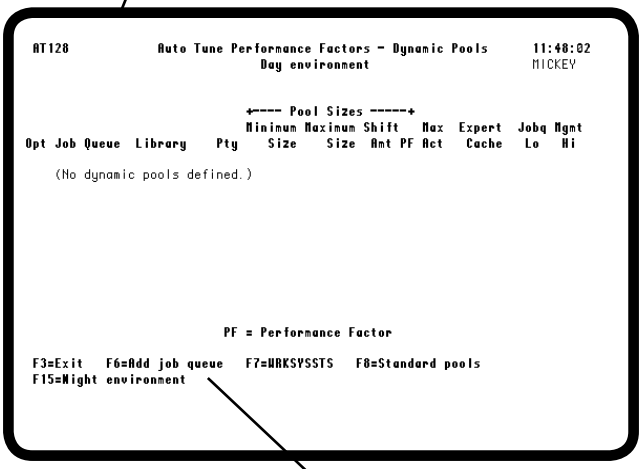


Accessing the Dynamic Pools Performance Factors

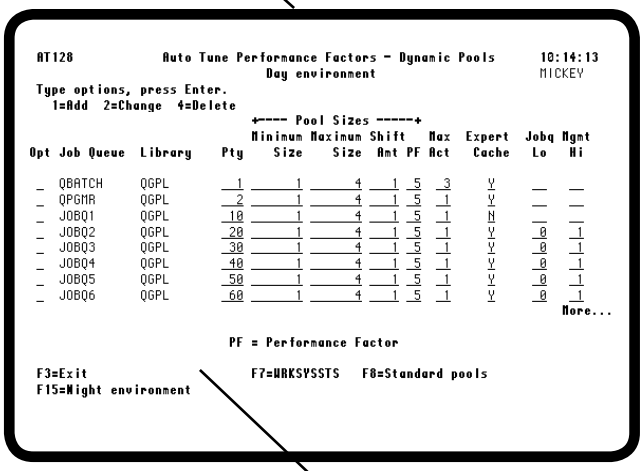
Enter option **2** on the Auto Tune Menu to access the Auto Tune Performance Factors - Dynamic Pools panel. Dynamic pools are temporary private pools that Robot Autotune creates for batch jobs. This feature can greatly reduce the impact that batch jobs have on the performance of your iSeries. Dynamic pools, an advanced feature of Robot Autotune, are discussed in detail in the Advanced Performance Tuning section of this User Guide.



Enter option **2** to display the Auto Tune Performance Factors - Dynamic Pools panel. The first time you access this panel, Robot Autotune should be inactive. In this exam-ple no dynamic pools are defined.



In this example, a group of job queues were already defined and Robot Autotune was active. See the Advanced Performance Tuning section of this User Guide for more information about dynamic pools.



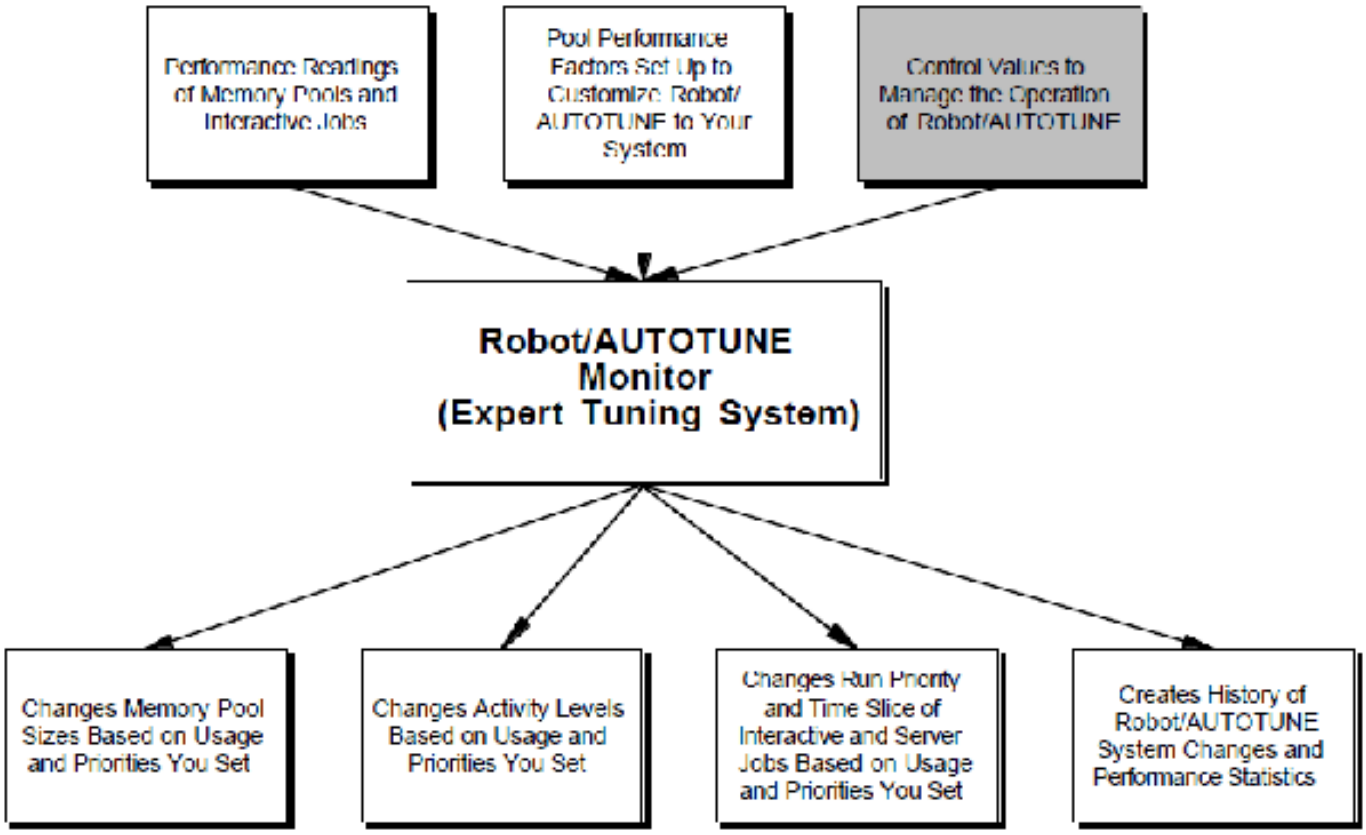
Press **F6** to add a dynamic pool job queue.

Note: Jobs that run constantly, such as monitor jobs, are poor choices for dynamic pool job queues. These jobs never end, so a dynamic pool associated with a monitor job is never removed.

F6 does not display if Robot Autotune is active.

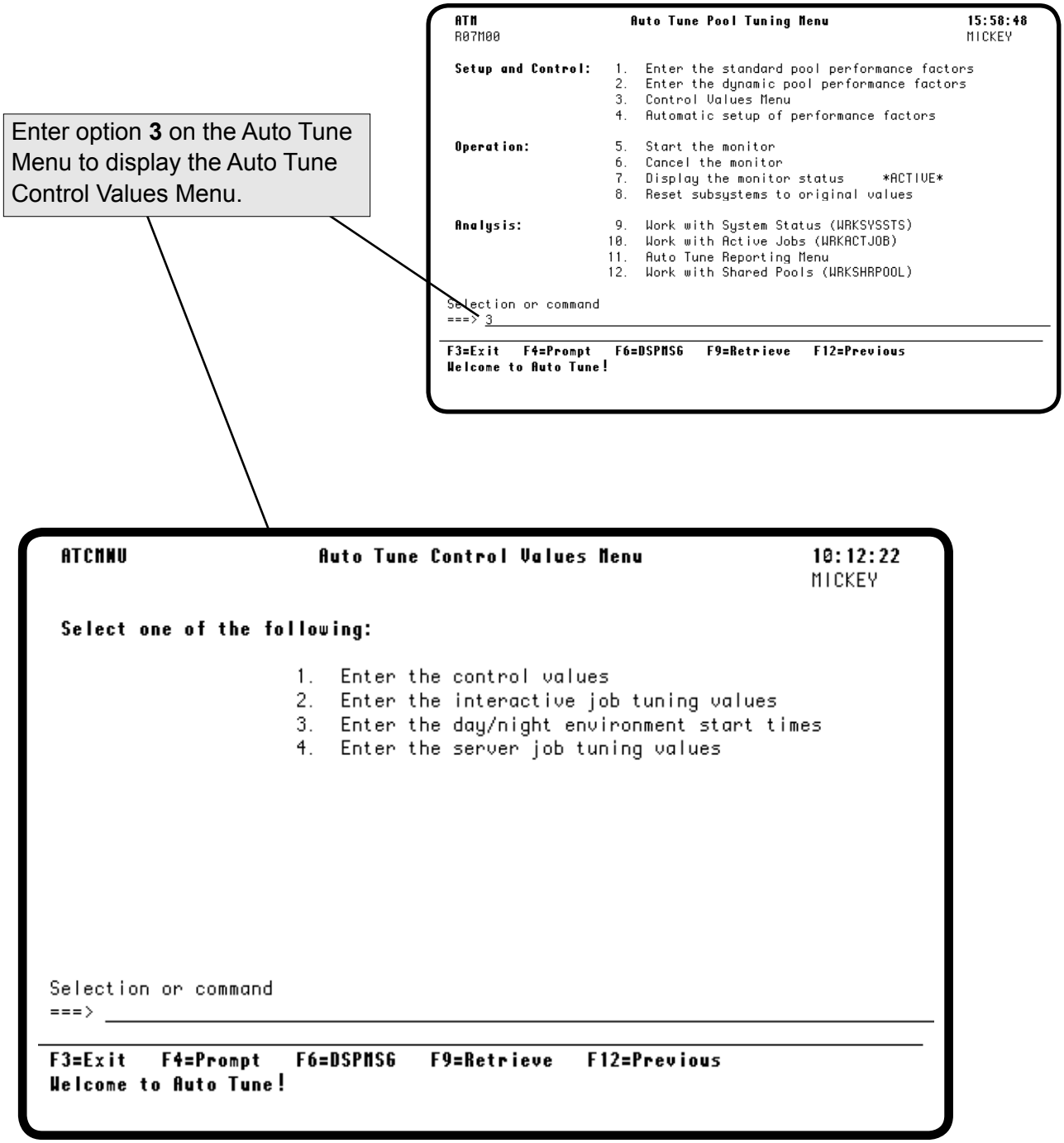
Control Values

In addition to performance factors, Robot Autotune uses control values to manage its operations.



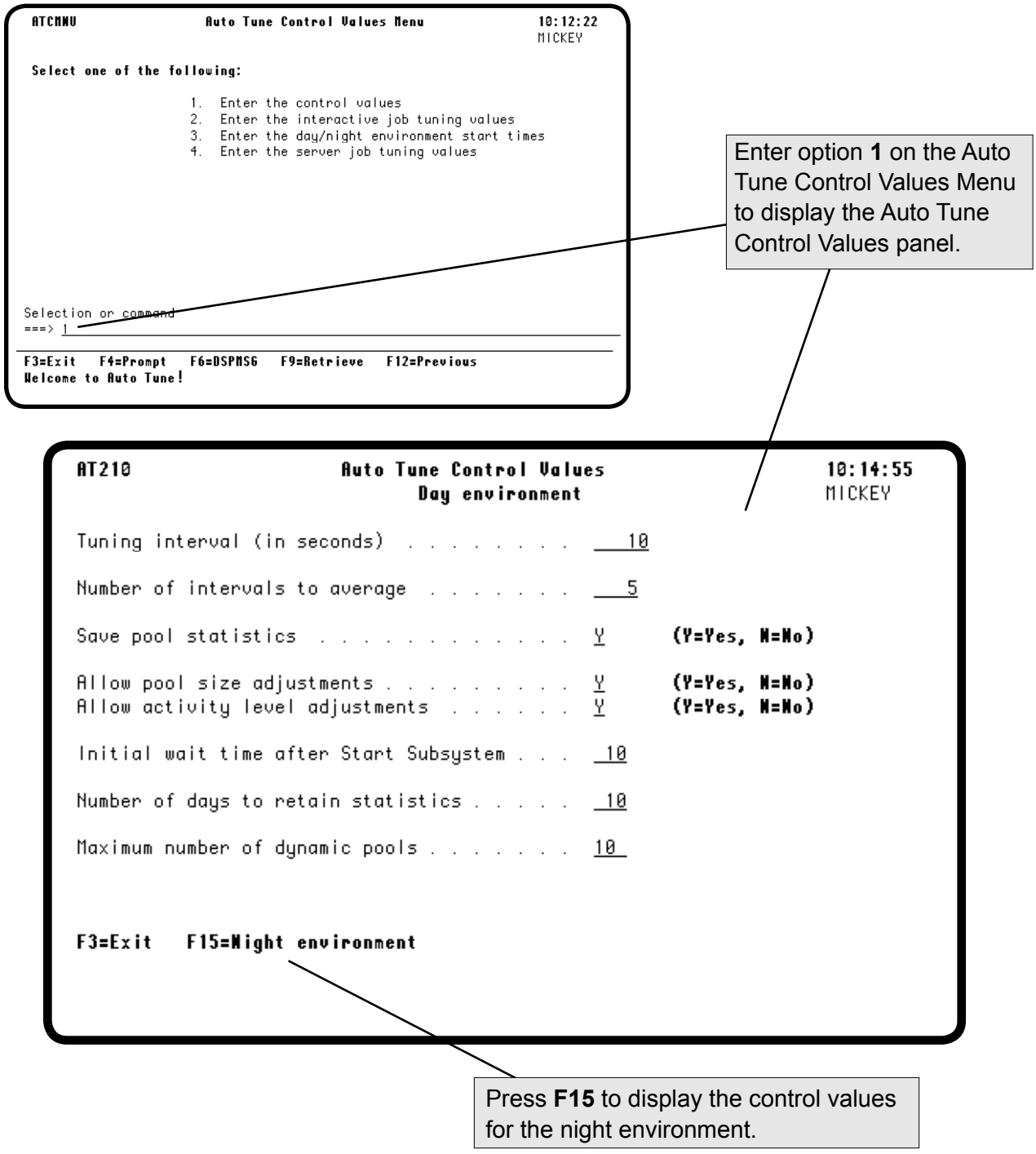
Accessing the Control Values Menu

You can adjust the control values through the Auto Tune Control Values Menu. Enter option **3** on the Auto Tune Menu to display the Control Values Menu.



Entering Control Values

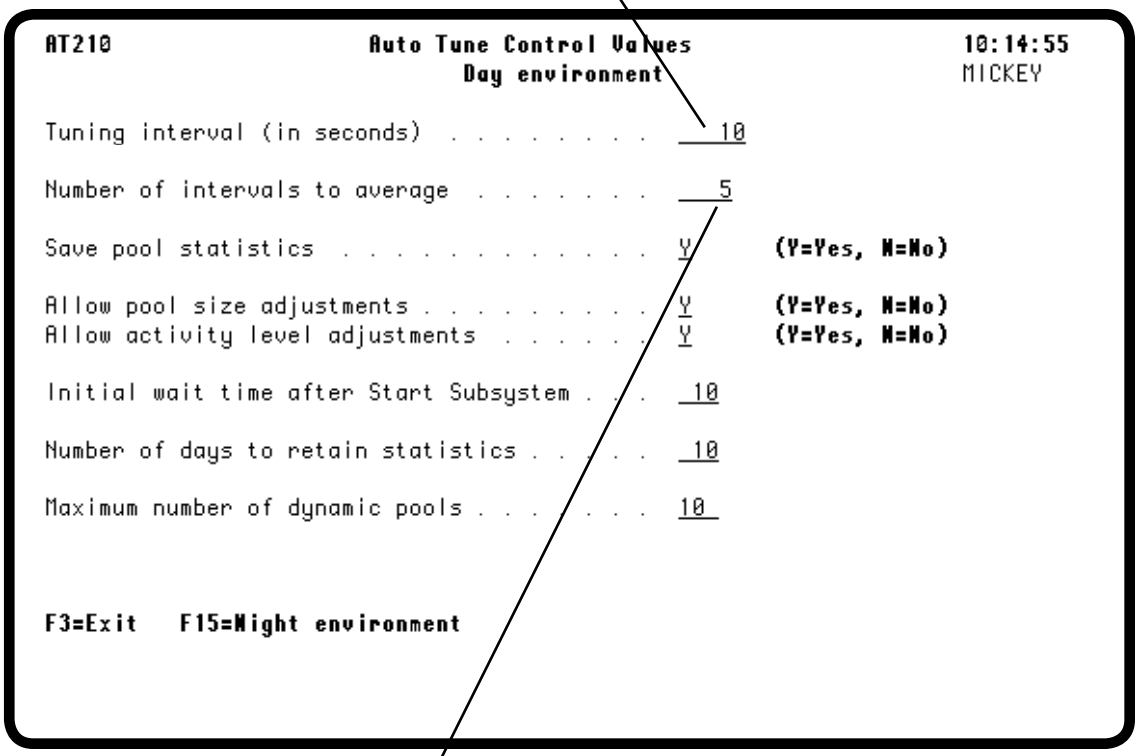
Enter option **1** on the Auto Tune Control Values Menu to display the Auto Tune Control Values panel. From this panel, you can adjust a number of factors important to the operation of Robot Autotune. These factors include the tuning interval, the number of intervals to average, the initial wait time after the subsystem is started, and other factors involved with operations and statistics. The control value defaults provided with Robot Autotune are appropriate for most environments.



Entering Control Values

You can adjust the time period that Robot Autotune uses to take performance readings on the Auto Tune Control Values panel. This time period is called the *tuning interval*. Because Robot Autotune operates at the operating system level, you can use even the shortest interval of five seconds without fear of putting a load on your system. You also can adjust the number of intervals that Robot Autotune averages in evaluating system performance from this panel.

Enter the number of seconds Robot Autotune should wait between system performance readings. This time period is called the tuning interval. Robot Autotune is designed to work at intervals of less than 20 seconds (10 is the recommended setting). However, you may want to specify a longer interval if you want your system to react only to longer-term shifts in workload. The tuning interval range is from 5 to 3600 seconds.



Enter the number of performance measurements (intervals) you want Robot Autotune to use to compute each weighted average it uses for evaluating system performance.

- The range is 1 to 10 (default is 5).
- If you specify an interval of 1, the system adjustment is based on only the current performance measurement.
- Using bigger numbers produces more stable activity levels.

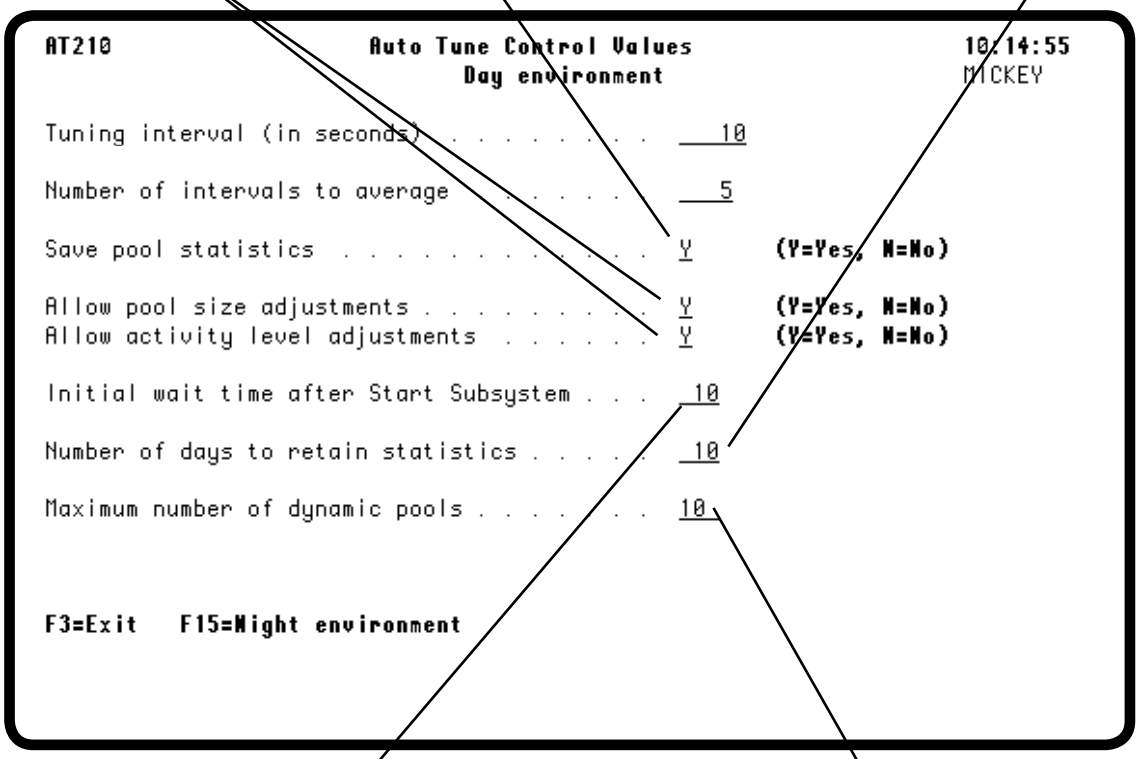
Entering Control Values

Robot Autotune lets you save memory pool performance statistics and specify the number of days these statistics are saved.

Enter a **Y** to have Robot Autotune save pool statistics.

Enter a **Y** to allow Robot Autotune to adjust pool sizes and activity levels.

Specify the number of days that statistics objects should remain in the library ATLIB before Robot Autotune automatically deletes them. You can enter 0 to 900 days.



The amount of time (in seconds) that Robot Autotune should wait before it begins its work. We recommend to wait at least 10 seconds so that the system can return to normal operation after an IPL.
Note: Robot Autotune also waits this amount of time after the ATMONITOR subsystem is started before beginning its work.

Enter the number of dynamic pools that Robot Autotune can create and use at one time.

- You can specify up to 62 dynamic pools.
- The total number of standard and dynamic pools cannot exceed 64.

Tip: You may want to allow Robot Autotune to create and use more dynamic pools at night when you are doing mainly batch work. Press **F15** to access the night environment.

Saving Pool Performance Statistics

You can have Robot Autotune save pool performance statistics. These are similar to the statistics obtained from the **WRKSYSSTS** command and are for your use.

You can display or print pool statistics by selecting an option from the Auto Tune Reporting Menu. The accumulated statistics also can be put into a database file and downloaded to a PC. These files can be used by popular spreadsheet programs or you can write programs to process these readings. You also can use Robot Monitor, the performance monitoring package from Fortra, to display and print graphs of some Robot Autotune statistics. See the Robot Monitor User Guide for more information.

Performance statistics are saved in user space objects. When you do a **DSPLIB ATLIB**, you see them as *USRSPC objects that start with the letters ATPS and end with the date they were created.

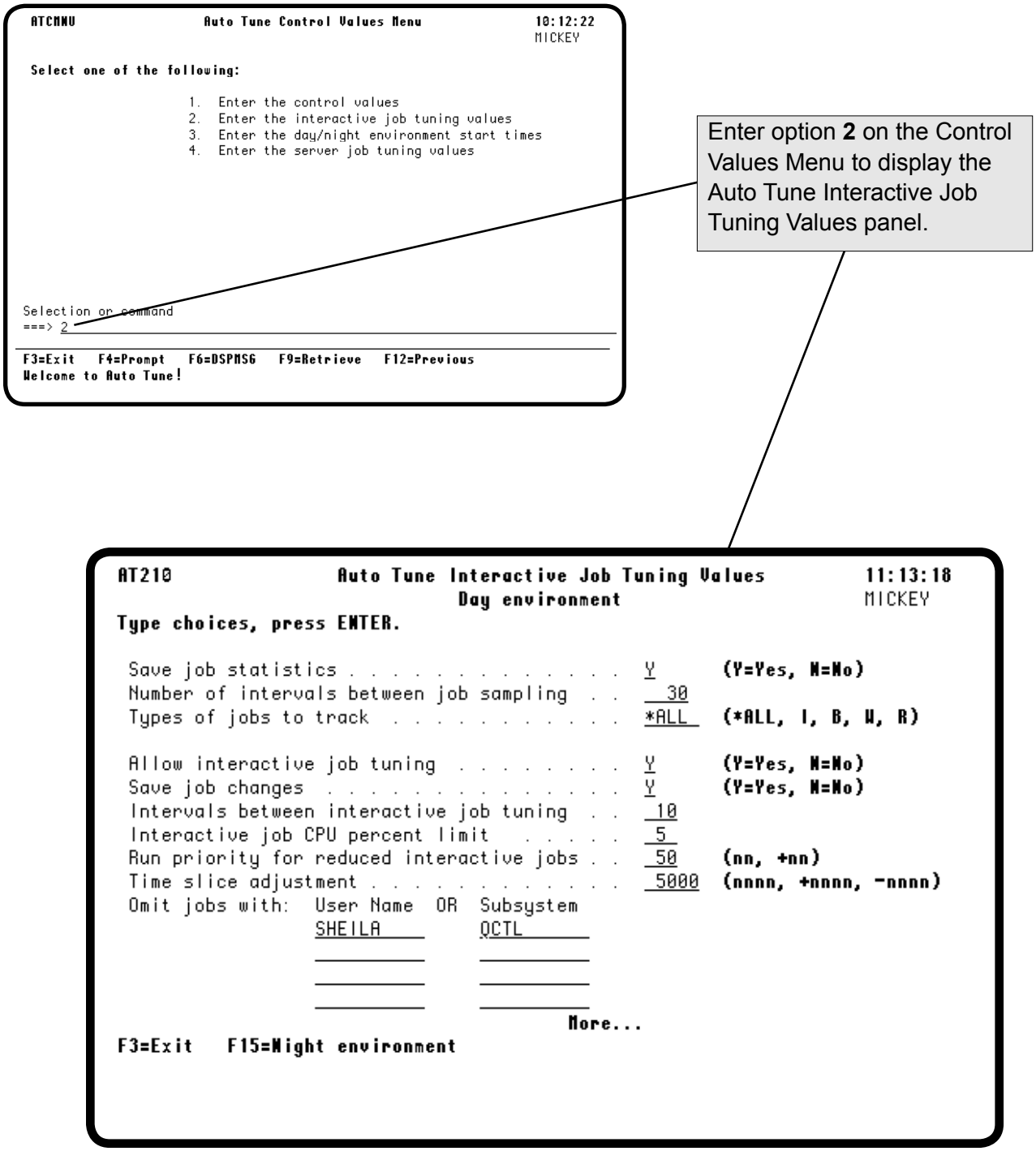
- A record is created for each active subsystem, dynamic pool, and shared pool at each tuning interval.
- The size of the user space depends on the number of pools, tuning intervals, and so forth. For example, if you ran Robot Autotune 24 hours a day with a tuning interval of 10 seconds, Robot Autotune would create an 800K user space every day.
- The maximum size of the user space is 16 MB. If Robot Autotune fills this space for a given day, no further statistics are saved.
- A new user space for statistics is created automatically the next day.

We recommend that you save performance statistics as soon as you have customized Robot Autotune to your system. Saving statistics uses disk space, but you can reduce the amount of system resources used by having Robot Autotune automatically delete statistics that reach a certain age. You enter the number of days to retain statistics on the Auto Tune Control Values panel. If you keep performance statistics, we recommend that you keep at least two days worth.

Note: Robot Autotune deletes statistics automatically at midnight of the last day specified by the Number of days to retain statistics field on the Auto Tune Control Values panel.

Tuning Interactive Jobs

Use the Auto Tune Interactive Job Tuning Values panel to specify whether you want to allow Robot Autotune to tune *interactive* jobs. You also use this panel to specify whether to save statistics for *all* jobs.



Tuning Interactive Jobs

Interactive job tuning is a way of finding users who are running batch work interactively and assigning their job a lower run priority for the duration of the batch work.

You want to tune interactive jobs to prevent high-utilization jobs from taking over system resources and making the system perform poorly for other users. The kinds of jobs that can cause problems include looping jobs, compilations, complex queries, poorly written database applications that access too many records, and various system tasks like displaying the library QSYS or displaying and searching large spool files.

To have Robot Autotune perform this function, you need to give it permission to monitor for interactive jobs that need tuning and to supply a few values. Robot Autotune needs to know how often to tune interactive jobs, what percentage of CPU usage indicates jobs that need to be tuned, and what run priority to assign these jobs. You can also specify adjustments to the time slice allowed for the job, exclude jobs from specified subsystems, and exclude user profile names from interactive job tuning.

Setting the Interactive Job Tuning Interval

You set the interactive job tuning interval as a multiple of the tuning interval. The smallest interactive job tuning interval is equal to the tuning interval. The smaller the interval the more system resources are needed to accomplish interactive job tuning. A larger interval may allow high utilization, but short duration jobs to pass through the system unnoticed (which may be desirable).

Setting the CPU Usage Threshold

You set the CPU utilization rate that triggers interactive job tuning. We recommend using 20% as a starting point.

To tune your interactive jobs more precisely, start by saving interactive job statistics. Look at the range of CPU utilization caused by interactive jobs. For example, if you find that your interactive jobs typically use 1–4% of the CPU, you could set your CPU utilization threshold to 5% to detect jobs out of the normal range. (If you have a lot of programmers, you might need to set a higher level of CPU utilization before job tuning is invoked, maybe 10%.) The closer you set your threshold to the top of the normal range for your system, the more sensitive the interactive job tuning becomes.

Setting the Run Priority

You also set the run priority for interactive jobs that are being tuned. You can specify either a fixed value or a positive amount by which to change the current run priority. Generally, you should set the run priority so it is *below* the priority of all other interactive work and *equal to or slightly above* the priority of batch work.

Note: A lower number reflects a higher run priority for a job.

Robot Autotune monitors for interactive jobs that need tuning. When it finds an interactive job with a CPU usage above the threshold that you specified and a very low transaction rate, Robot Autotune recognizes that batch work is being done interactively. Robot Autotune automatically changes the run priority for the job to the priority you specified or by the adjustment specified. The job returns to its original run priority automatically when Robot Autotune detects that the CPU usage percent has gone below the limit and the transaction rate has increased.

Adjusting the Time Slice

In addition to changing the job's run priority, you can specify an adjustment to the time slice allowed for the job. If you enter a time slice adjustment, the time slice is changed when the job's run priority changes. You can enter a specific value, or a positive or negative amount by which to adjust the time slice. If you do not specify a time slice adjustment, the current time slice remains in effect.

Omitting Jobs from Interactive Tuning

You also may have jobs that you want to exclude from interactive tuning. Just enter the subsystem names in which these jobs run, or the user profile names whose jobs should not be tuned, and Robot Autotune omits them from interactive tuning.

Saving Job Changes

You can tell Robot Autotune to save job change statistics. Robot Autotune collects job change information for any interactive or server jobs that are tuned. The statistics can be copied to a database file using an option on the Auto Tune Reporting Menu. You can process these statistics with your own programs.

Note: Job change statistics are saved in user space objects. When you do a **DSPLIB ATLIB**, you will see them as *USRSPC objects that start with the letters ATJT and end with the date they were created. The size of the user space depends on the number of jobs and the sampling interval. The maximum size is 16 MB. If Robot Autotune fills this space for a given day, no further statistics are saved. A new user space for statistics is created automatically the next day.

Saving Job Statistics

Robot Autotune job statistics are similar to those obtained from the **WRKACTJOB** command and are for your use. You may want to save them for a while when you first install Robot Autotune to troubleshoot problem jobs on your system.

You can display the job statistics by selecting an option from the Auto Tune Reporting Menu. You can copy the accumulated statistics to a database file and download them to a PC, or process them with your own programs.

Job statistics are saved in user space objects. When you do a **DSPLIB ATLIB**, you will see them as *USRSPC objects that start with the letters ATJS and end with the date they were created.

- The size of the user space depends on the number of jobs and the sampling interval. The maximum size is 16 MB.
- If Robot Autotune fills this space for a given day, no further statistics are saved.
- A new user space for statistics is created automatically the next day.

Saving System Resources

Unlike performance statistics, we recommend that you stop saving job statistics as soon as you have identified problem jobs. Robot Autotune uses system resources to save these statistics. If you want to continue looking at job statistics, you can reduce the amount of system resources used by having Robot Autotune automatically delete job statistics that reach a certain age. Enter the number of days to retain statistics on the Auto Tune Control Values panel.

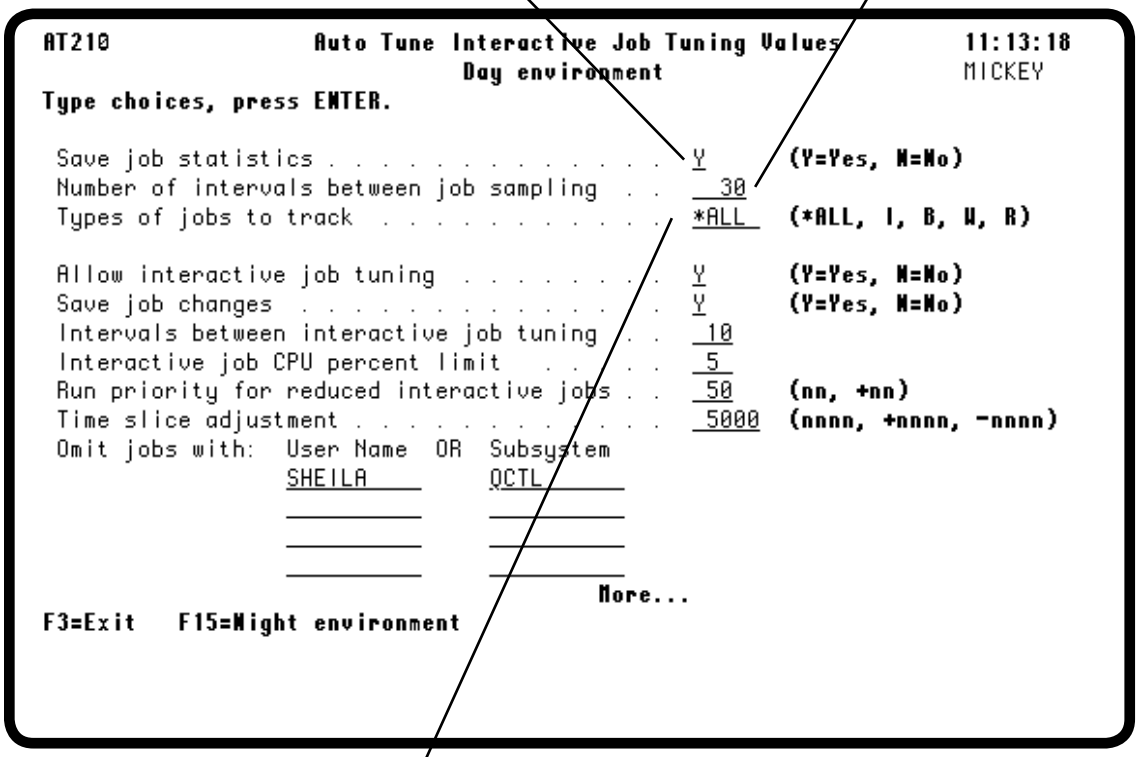
Note: If you keep job statistics, we recommend that you keep at least two days worth.

Saving Job Statistics

Several fields on the Interactive Job Tuning Values panel allow you to specify if you want to save job statistics and how to save them. Robot Autotune can track the statistics for the type of job you specify or for all jobs, including system jobs.

Enter a **Y** to save job statistics.

Enter the number of tuning intervals between job statistics sampling. For example, if you want to save job statistics every 300 seconds and your tuning interval is 10 seconds, enter a value of 30 in this field.



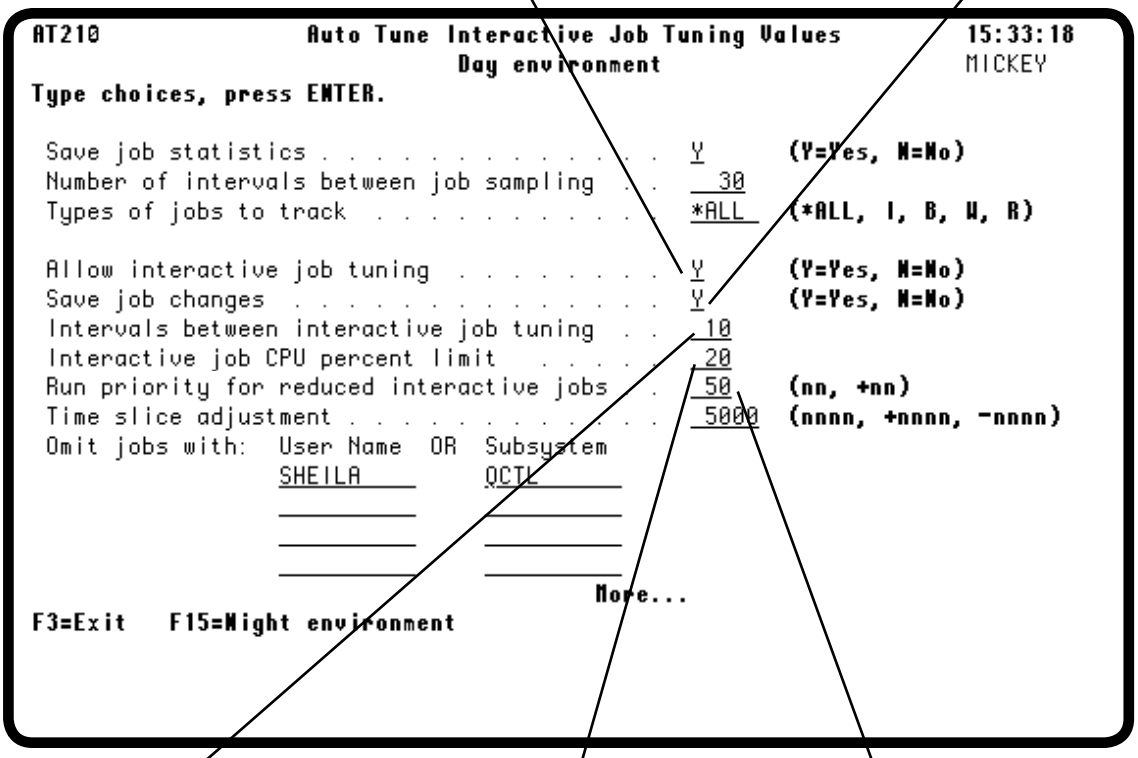
Enter ***ALL** to have Robot Autotune keep track of statistics for all jobs on the system, including system jobs. Or, enter one or more of the following options to track specific types of jobs:

I	Interactive	**R**	Readers
B	Batch	**W**	Writers

Entering Interactive Job Tuning Values

Enter a **Y** to allow interactive job tuning. Enter an **N** to turn off interactive job tuning. In general, this field should be set to **Y** unless you are recording job statistics and you want to evaluate the effect of not having Robot Autotune tune interactive jobs.

Enter a **Y** to save any changes made to an interactive or server job. You can copy the information into a database file using an option on the Auto Tune Reporting Menu.



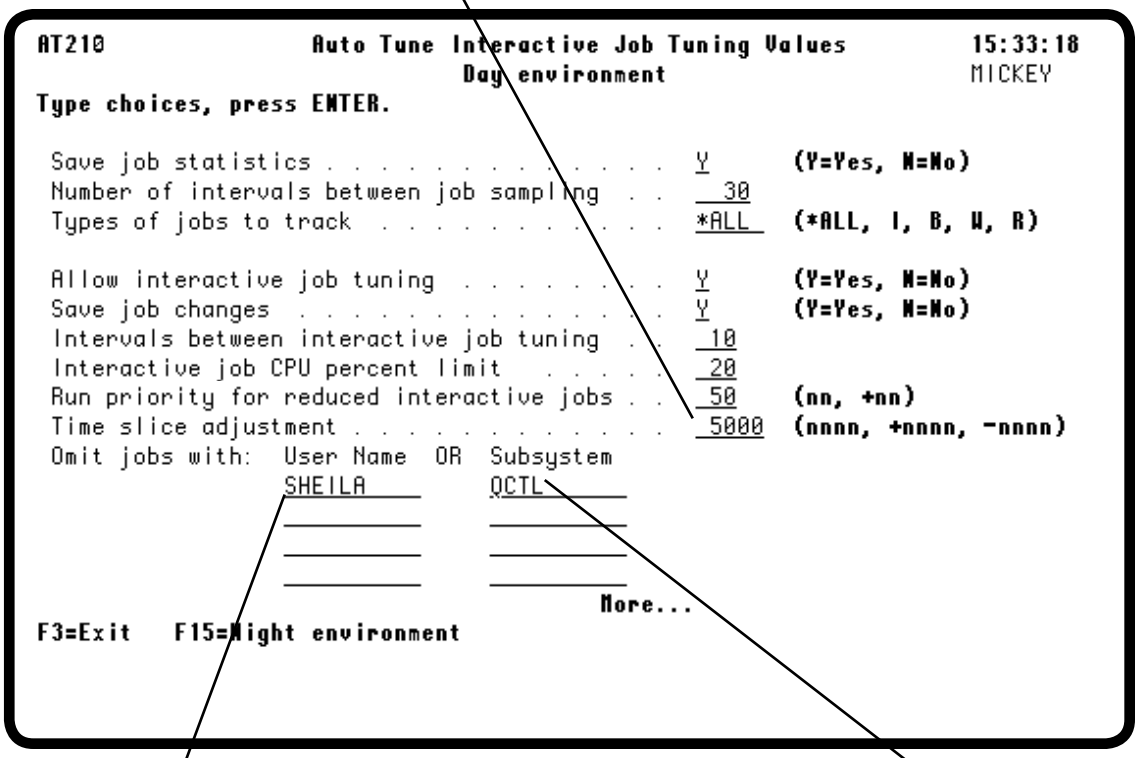
Enter the number of tuning intervals between interactive job tuning. For example, if you want to tune interactive jobs every 100 seconds and your tuning interval is 10 seconds, enter a value of 10 in this field. We recommend that you use 10 as a starting point.

Enter the reduced run priority for interactive jobs that are tuned. You can enter either a specific value or a positive amount by which to change the current run priority. The default is 50, which should work for most systems.

Enter the CPU usage level that should trigger interactive job tuning. You can use 20% as a starting point. Valid entries in this field are 1 to 99.

Entering Interactive Job Tuning Values

You can specify that the time slice allowed for a job be adjusted when the job's run priority changes. Enter either a specific value, or a positive or negative value by which to change the time slice. If you do not specify a time slice adjustment, the current time slice remains in effect.



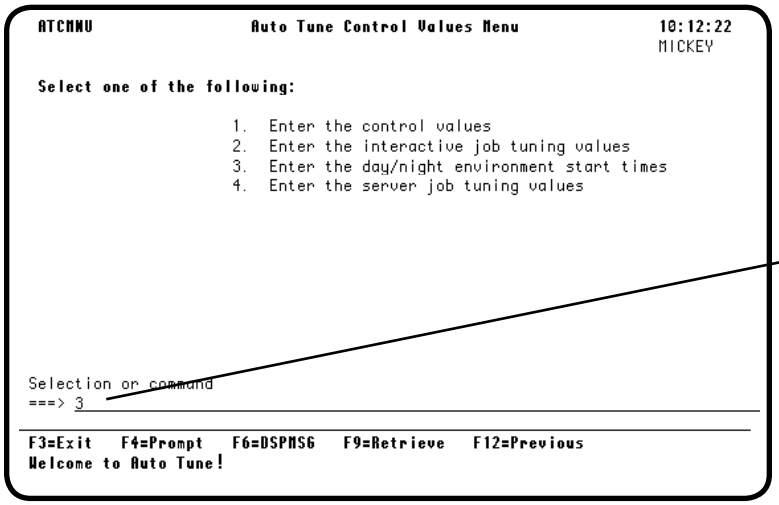
Jobs can be excluded from interactive job tuning. Enter either the subsystems in which these jobs run, or the user profile names whose jobs should not be tuned, and Robot Autotune omits them from interactive tuning. You can enter multiple user profile names or subsystem names.

We recommend you omit your controlling subsystem (QCTL in this example) from tuning. In general, because of performance considerations, you do not want to tune jobs in your controlling subsystem.

Entering the Day/Night Environment Start Times

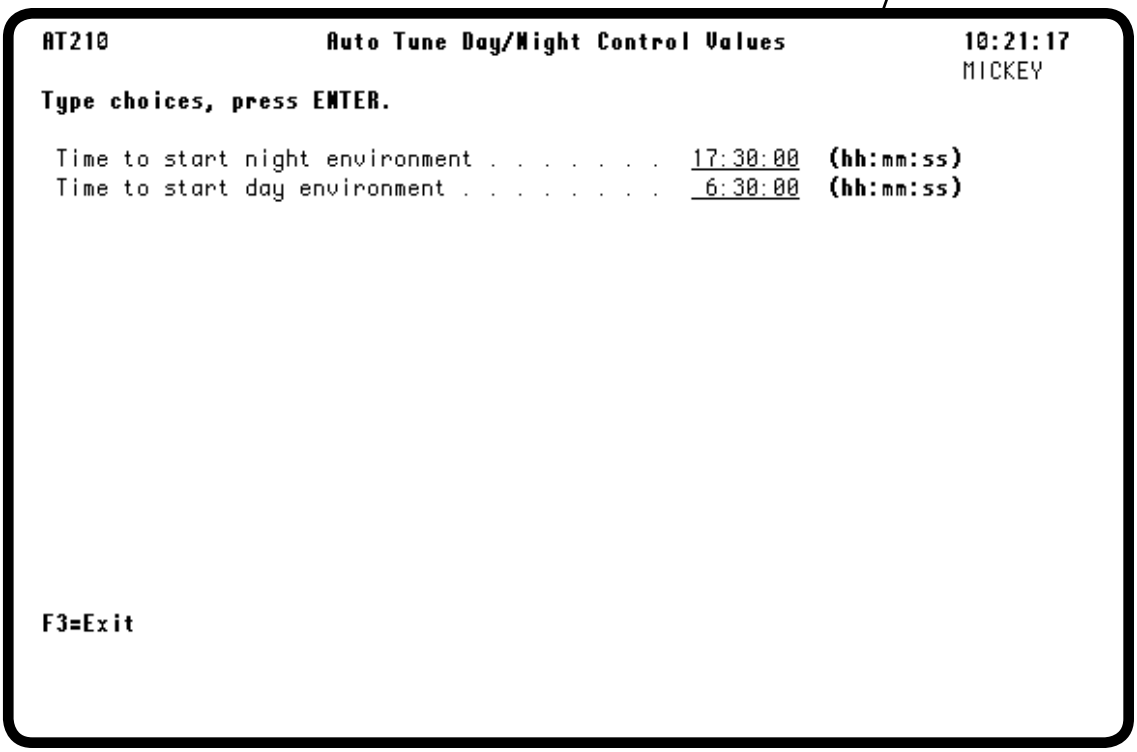
Robot Autotune lets you enter two different sets of performance factors and control values. These sets of factors are designated the day and night environments. Robot Autotune automatically switches from one set of factors to the other at the time you specify on the Auto Tune Day/Night Environment Control Values panel.

If you are a Robot Schedule user, you also can switch between environments by scheduling the **ATLIB/CHGATSCH** (Change Auto Tune Schedule) command with the appropriate time parameters.



Enter **option 3** on the Auto Tune Control Values Menu to display the Auto Tune Day/Night Control Values panel.

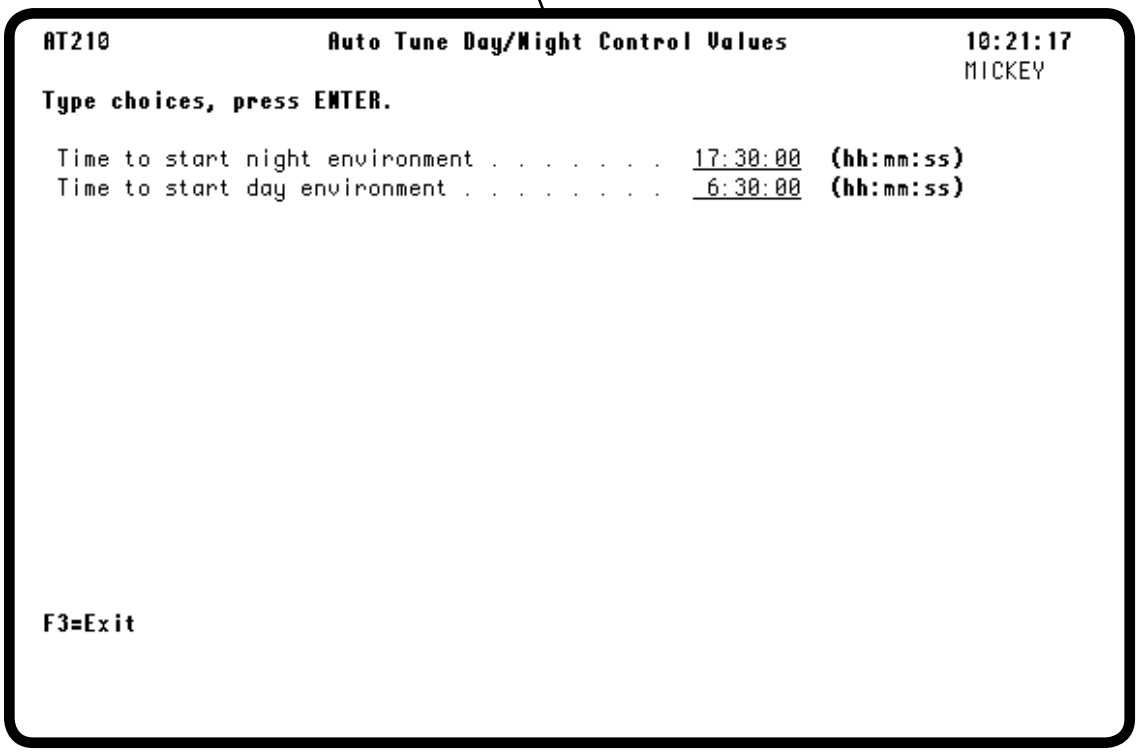
Or, type the command **ATLIB/CHGATSCH** and press **F4** to access the prompt panel from any command line.



Entering the Day/Night Environment Start Times

Use this panel to specify the time to switch to your night performance factors and back to your day factors. This switch affects all standard and dynamic pool performance factors and control values except the day/ night start times.

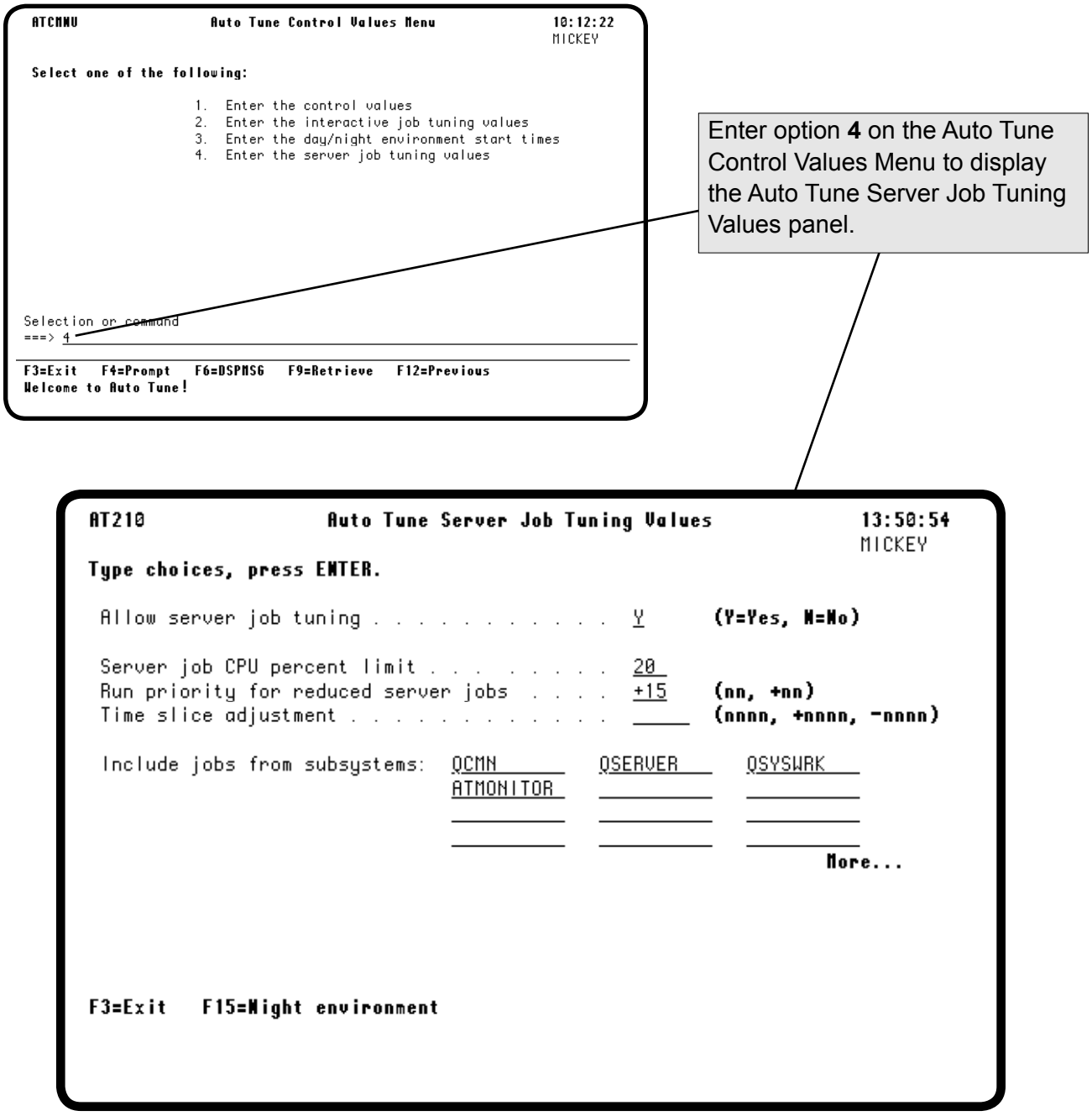
Enter the times to start the specified environment in hour, minute, second format using a 24-hour clock. If you don't want Robot Autotune to use different sets of performance factors, enter the same time in each field and Robot Autotune will use the day environment factors all the time.



Entering Server Job Tuning Values

Robot Autotune allows you to adjust iSeries server (communication and batch) jobs through the Auto Tune Server Job Tuning Values panel. Consider having Robot Autotune manage those jobs that tend to use large amounts of CPU. These can include jobs that run in QSYSWRK, QUSRWRK, QCMN, and even ATMONITOR.

Use this panel to turn server job tuning on or off. If you turn it on, you then can specify changes to the run priority of a job, the time slice, and the subsystems that contain the jobs to be tuned.

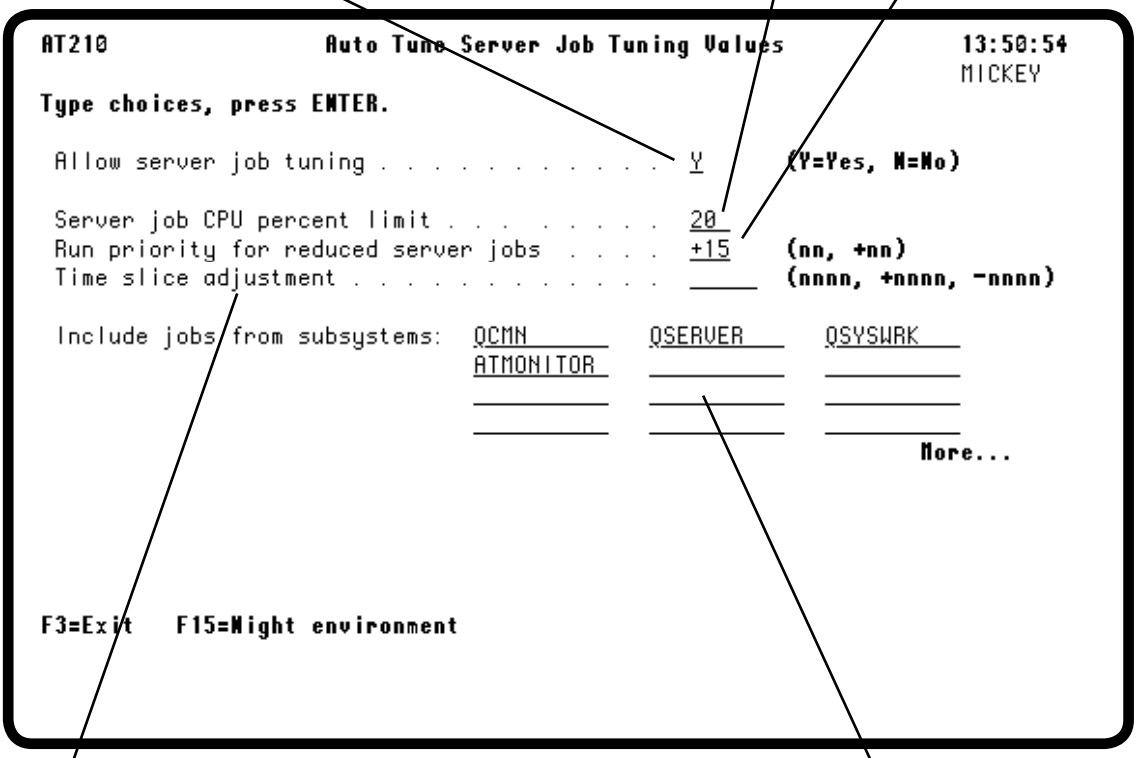


Entering Server Job Tuning Values

Enter the run priority for server jobs being tuned. You can enter either a specific value or a positive amount by which the run priority will be changed. The job is adjusted by changing the run priority to the specified value or adding the value to the job's current run priority. The job's original run priority is restored when the CPU usage falls below the specified percent value.

Specify if you want to allow Robot Autotune to tune server jobs. Enter a **Y** to allow tuning; enter an **N** if you don't want Robot Autotune to tune those jobs. Robot Autotune uses the same tuning interval for server jobs as it does for interactive jobs.

Specify the percent of CPU usage that should trigger server job tuning. When the CPU usage reaches the percent you've specified, the server jobs' run priority is reduced.



You can specify that the time slice allowed for a job be adjusted when the job's run priority changes. Enter either a specific value, or a positive or negative value by which to change the time slice. If you do not specify a time slice adjustment, the current time slice remains in effect.

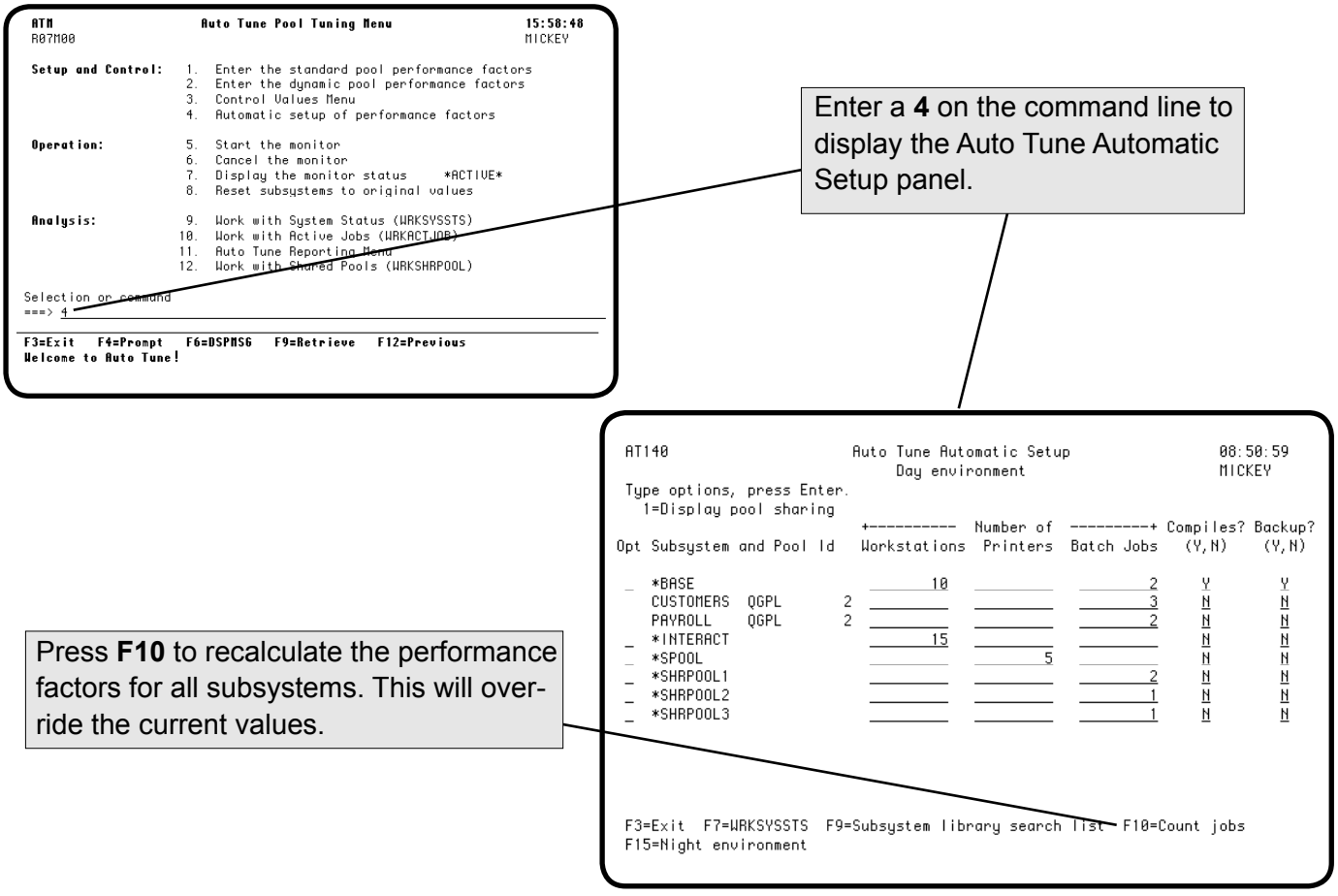
Enter the subsystem names of the jobs that should be tuned. Include any subsystems that contain batch or communication jobs. All the jobs in the specified subsystems are eligible for tuning.

Accessing Automatic Setup of Performance Factors

You can access the Auto Tune Setup panel two ways. When you first install Robot Autotune, select **option 1** on the Auto Tune Menu to display the Automatic Setup panel. This panel is designed primarily for the initial setup of Robot Autotune. If you are setting up the product for the first time, follow the instructions in the Getting Started section of this User Guide.

In general, if your system setup changes, you can make all the adjustments you need by referring to the Performance Factors and Activity tables and entering the information on the Auto Tune Performance Factors - Standard Pools panels. For example, if you added three new printers to the QSPL subsystem, you would increase the Maximum Pool Size field value by three times 80 K, or 240 K (according to the table), and press Enter to record your changes.

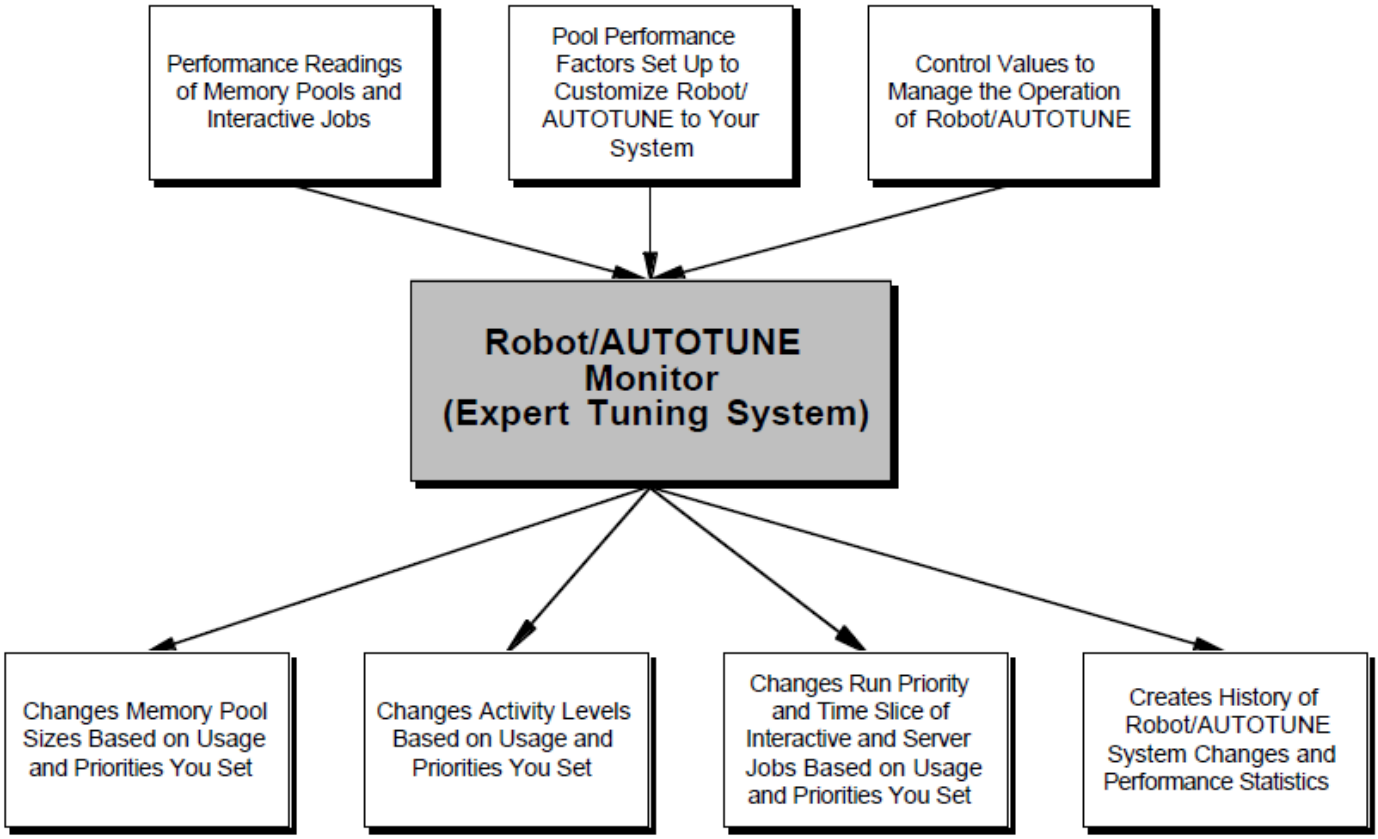
However, if you want Robot Autotune to recalculate performance factors for *all* subsystems, select **option 4** on the Auto Tune Menu to return to the Auto Tune Automatic Setup panel, then press **F10**. Robot Autotune recalculates pool sizes and performance factors for all subsystems (even if you haven't made any changes). Any adjustments you have made to Robot Autotune's automatically calculated values will be overwritten and must be reentered. It is a good idea to have a printout of your performance factors before making changes using this panel. You can exit the panel without causing any change by pressing **F3**.



Notes:

Operations

Operating Robot Autotune



Operating Robot Autotune

Robot Autotune is designed for easy operation by automating some functions and entering commands for others. Once you have Robot Autotune set up and running, you can perform many functions without accessing the Auto Tune Menu. If you prefer, you can use the menu to perform the functions. For example, we recommend that you start and stop Robot Autotune automatically each time you start up or power down your system.

Security and Authority

Anyone who has QPGMR authority can run Robot Autotune. If you want to restrict certain parts of Robot Autotune, refer to the Diagnostics section of this User Guide.

Starting the Monitor

To start Robot Autotune automatically, put the following command in your startup program:

```
ATLIB/STRAT
MONMSG MSGID(ATI0035)
```

The first command starts Robot Autotune; the second command monitors for a specific Robot Autotune message—ATI0035. This message ("Not authorized to ATMONITOR subsystem description.") indicates the person who tried to start Robot Autotune is not authorized to do so. By monitoring for this message, you can find out if Robot Autotune failed to start as part of your startup procedure. Robot Autotune runs in its own subsystem (ATMONITOR) so it will not interfere with the rest of your operations.

You can start Robot Autotune manually using its menu. Look at the status of the Robot Autotune subsystem next to option 7 on the Auto Tune Menu. If Robot Autotune is inactive, start it by selecting **option 5**, Start the monitor. The status changes to *Started*, and then *Active*. The iSeries command, **STRSBS ATMONITOR**, *will not* start Robot Autotune and the ATMONITOR job correctly.

Stopping the Monitor

You can automate the process of cancelling the monitor by including the command **ATLIB/ENDAT** or **ATLIB/CNLAT** in your power down procedures or program.

Note: The ENDAT command is the same as the CNLAT command used in previous versions of Robot Autotune. If you have the CNLAT command in your procedures or programs, you can continue to use it.

To end the monitor manually, enter the command **ATLIB/ENDAT** or **ATLIB/CNLAT** from a command line, or select option **6** from the Auto Tune Menu. Whenever the Robot Autotune monitor is cancelled, Robot Autotune automatically restores pool sizes and activity levels to those contained in your subsystem descriptions, if possible. You should always end the monitor before you power down the system.

Note: We recommend that you stop and restart the ATMONITOR job on a regular basis. This clears entries in the job log and prevents it from becoming too large.

■ Starting and Stopping Subsystems

You can still start and stop subsystems just as you always have as long as those subsystems are described to Robot Autotune. However, with Robot Autotune, you have a better and easier method: keep the subsystems active all the time. Reduce the minimum pool size in Robot Autotune according to the minimum values recommended in this User Guide. Then, when the subsystems are used, Robot Autotune increases or decreases the pool sizes as needed.

■ Robot Autotune Makes No Permanent Changes

Even though Robot Autotune continually changes pool sizes and activity levels, it makes no changes to subsystem descriptions. When you start up your subsystems, the initial pool sizes and activity levels are set as indicated on the subsystem description. If you notice that Robot Autotune drastically changes the initial values after startup every day, you should change the subsystem descriptions so they are close to what Robot Autotune averages.

■ Resetting Subsystems to Their Original Values

Occasionally, you may want to overrule the settings made by Robot Autotune and reset all your subsystems to their subsystem descriptions. Robot Autotune provides three ways to do this:

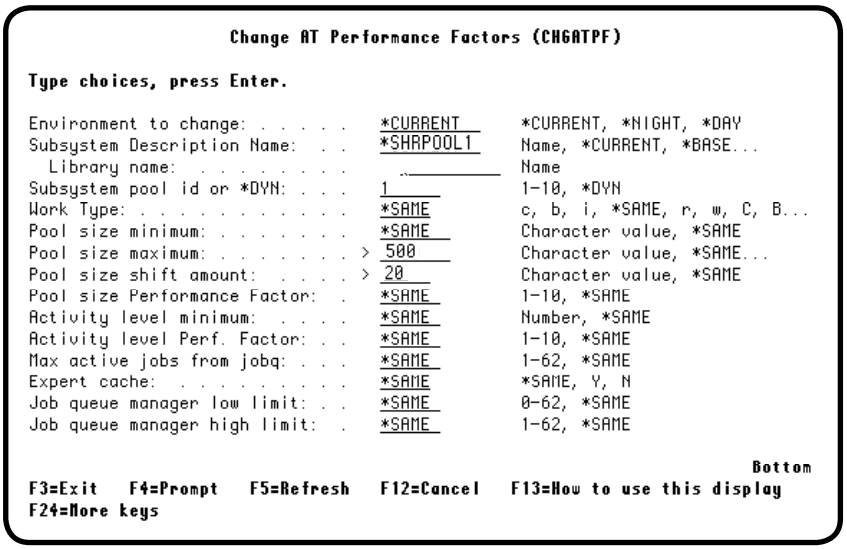
1. Cancel the Robot Autotune monitor by entering the command **ATLIB/ENDAT** or **ATLIB/CNLAT**, or by selecting **option 6** on the Auto Tune Menu. All pool sizes and activity levels will be set back to your subsystem descriptions, if possible, and *tuning will stop*.
   ...or
2. Execute the command **ATLIB/RESETSBS** from the command line. This sets all pool sizes and activity levels back to your subsystem descriptions (tuning continues).
   ...or
3. Select option **8** on the Auto Tune Menu to reset subsystems to their original values (tuning continues).

Changing Performance Factors with a Command

Robot Autotune provides a command, **CHGATPF**, to change performance factors from outside Robot Autotune. This command is especially useful if you have a high priority job that uses a lot of memory. By placing this command as the first step in a program, you increase the maximum batch pool size and the pool priority for the job. Robot Autotune still reduces the actual pool size automatically if memory is unused (no matter how high the maximum is set). Use the command with your normal settings as the last step in the program to return the settings to normal.

The default for the **CHGATPF** command parameters is ***SAME**, which will not change the particular setting. The library defaults to QSYS. You can specify the special value ***CURRENT** for the SBSD parameter. This allows Robot Autotune to change the pool that the job is in.

For example, you might change a few values, as follows:

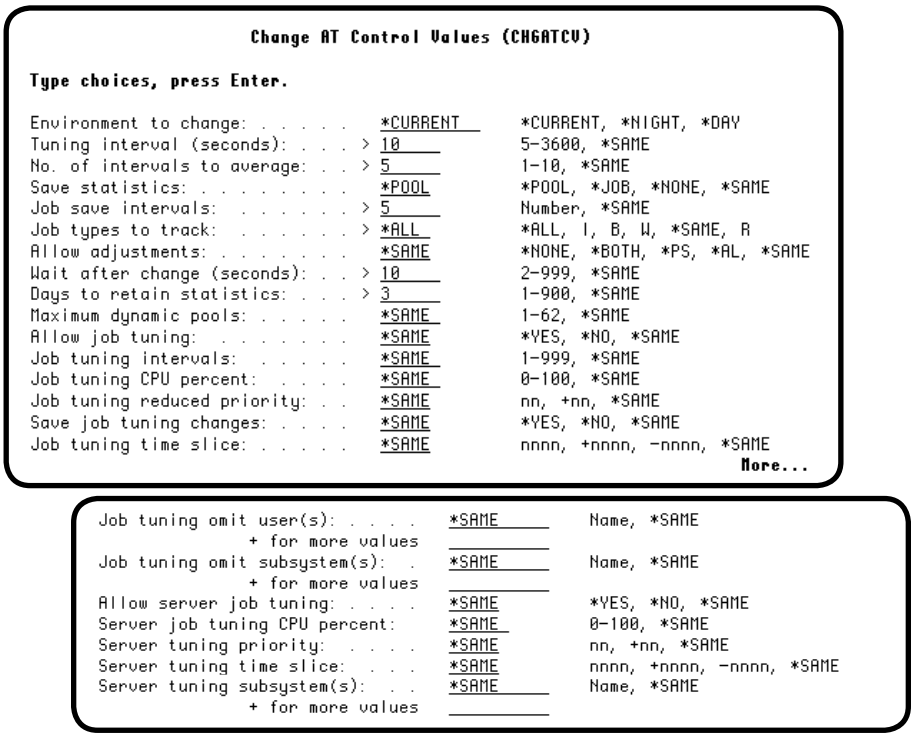


```
                    Change AT Performance Factors (CHGATPF)

Type choices, press Enter.

Environment to change: . . . . .   *CURRENT     *CURRENT, *NIGHT, *DAY
Subsystem Description Name:  . .   *SHRPOOL1    Name, *CURRENT, *BASE...
  Library name:  . . . . . . . .     __________  Name
Subsystem pool id or *DYN: . . .   1           1-10, *DYN
Work Type: . . . . . . . . . . .   *SAME        c, b, i, *SAME, r, w, C, B...
Pool size minimum: . . . . . . .   *SAME        Character value, *SAME
Pool size maximum: . . . . . . > 500           Character value, *SAME...
Pool size shift amount:  . . . > 20            Character value, *SAME
Pool size Performance Factor:  .   *SAME        1-10, *SAME
Activity level minimum:  . . . .   *SAME        Number, *SAME
Activity level Perf. Factor: . .   *SAME        1-10, *SAME
Max active jobs from jobq: . . .   *SAME        1-62, *SAME
Expert cache:  . . . . . . . . .   *SAME        *SAME, Y, N
Job queue manager low limit: . .   *SAME        0-62, *SAME
Job queue manager high limit:  .   *SAME        1-62, *SAME

                                                                      Bottom
F3=Exit   F4=Prompt   F5=Refresh   F12=Cancel   F13=How to use this display
F24=More keys
```

Changing Control Values with a Command

Robot Autotune also provides the **CHGATCV** command to change the control values. You might schedule this command in Robot Schedule to save statistics the second Tuesday of every month so that you can have a record of the load on your machine. The default for the **CHGATCV** command parameters is ***SAME**, which will not change the particular setting.

For example, you could use the **CHGATCV** command to change some control values, as follows:



```
                    Change AT Control Values (CHGATCV)

Type choices, press Enter.

Environment to change: . . . . .   *CURRENT     *CURRENT, *NIGHT, *DAY
Tuning interval (seconds): . . . > 10          5-3600, *SAME
No. of intervals to average: . . > 5           1-10, *SAME
Save statistics: . . . . . .       *POOL       *POOL, *JOB, *NONE, *SAME
Job save intervals: . . . . . . > 5            Number, *SAME
Job types to track: . . . . . . > *ALL         *ALL, I, B, W, *SAME, R
Allow adjustments: . . . . . .     *SAME       *NONE, *BOTH, *PS, *AL, *SAME
Wait after change (seconds): . . > 10          2-999, *SAME
Days to retain statistics: . . . > 3           1-900, *SAME
Maximum dynamic pools: . . . . .   *SAME       1-62, *SAME
Allow job tuning: . . . . . .      *SAME       *YES, *NO, *SAME
Job tuning intervals: . . . . .    *SAME       1-999, *SAME
Job tuning CPU percent: . . . .    *SAME       0-100, *SAME
Job tuning reduced priority: . .   *SAME       nn, +nn, *SAME
Save job tuning changes: . . . .   *SAME       *YES, *NO, *SAME
Job tuning time slice: . . . . .   *SAME       nnnn, +nnnn, -nnnn, *SAME
                                                                  More...
```

```
Job tuning omit user(s): . . . .   *SAME       Name, *SAME
             + for more values
Job tuning omit subsystem(s): .    *SAME       Name, *SAME
             + for more values
Allow server job tuning: . . . .   *SAME       *YES, *NO, *SAME
Server job tuning CPU percent:     *SAME       0-100, *SAME
Server tuning priority: . . . .    *SAME       nn, +nn, *SAME
Server tuning time slice: . . .    *SAME       nnnn, +nnnn, -nnnn, *SAME
Server tuning subsystem(s): . .    *SAME       Name, *SAME
             + for more values
```

Backing Up Robot Autotune

After you have installed and configured Robot Autotune, you should cancel the monitor and save the ATLIB library. There is no need to back up Robot Autotune on an ongoing basis, but you should back up ATLIB whenever you change your setup.

Recovery Site/Hot Site Restoration

If you are conducting recovery site or “hot site” testing, install and configure Robot Autotune as a new installation—do *not* restore ATLIB at a recovery site.

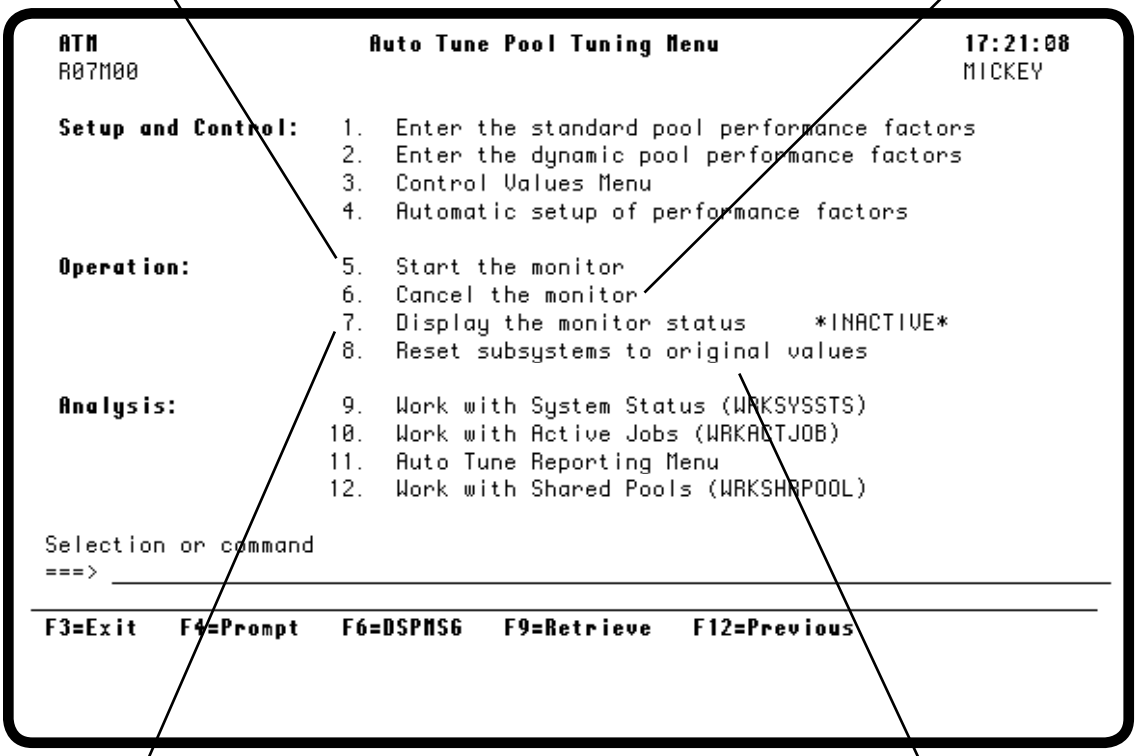
Using the Menu to Operate Robot Autotune

You can use the Auto Tune Menu to perform the major operational functions of Robot Autotune. From the Auto Tune Menu, you can:

- Start the Robot Autotune monitor
- Cancel the monitor
- Display the monitor status
- Reset subsystems to their original values

Select option **5** to start the monitor.

Select option **6** to cancel the monitor.



Select option **8** to have Robot Autotune attempt to reset all subsystems to match their subsystem descriptions and continue tuning.

Use **option 7** to display the ATMONITOR subsystem's jobs. The status of the ATMONITOR job in the ATMONITOR subsystem is always listed next to this option. Possible statuses are:

Inactive The monitor is not active. You can start it by selecting option **5.**

Started The monitor subsystem has been started but has not yet completed its startup procedures.

Active The monitor is active and tuning.

When you select option 7, the IBM OS/400 Work with Subsystem Jobs panel displays showing you more detail about the subsystem and allowing you to perform various operations.

Notes:

Performance Refinement Guide

Prerequisites:
Before you use this Performance Refinement Guide, you should do the following:

- Read the previous sections of this User Guide.
- Verify that the following subsystems—QINTER (interactive), QSPL (printers), and QBATCH (batch jobs)—are in separate pools.
- Install Robot Autotune and complete the Automatic Setup of Performance Factors.
- Run Robot Autotune for at least 20 minutes.

If you have this problem:	First, do this:	Then, try this:
Batch jobs do not get enough memory to run the jobs. The batch pool does not reach its maximum pool size.	Give the batch pool higher priority than the interactive pool (QINTER or *INTERACT).	
Batch jobs run too slowly.	• Check to see whether batch pools are reaching their maximum size. If they aren't, see the above solution. • If you are using dynamic pools, the maximum size of the batch pool is too low. Raise it 2-3 MB, or more. • If you are not using dynamic pools, use them. Set the maximum size of the batch pool to 5 MB, or more.	Use Robot Schedule run your batch work when system activity is lower.
Batch jobs drag down the performance of the interactive jobs.	• If you are not using dynamic pools, use them. Turn to the Advanced Performance Tuning section of this Guide for more information. • If you are using dynamic pools, the maximum size of the batch pool is too high. Reduce the size in increments of 3 MB until you get the interactive performance you want.	Use Robot Schedule to run you batch work when system activity is lower.

If you have this problem:	First, do this:	Then, try this:
Batch jobs terminate abnormally with a message that there is not enough memory to do a system function.	Increase the minimum pool size for the batch pool (QBATCH) by 3 MB. If saves are done in batch, the minimum pool size should be 5 MB or more.	
Batch, interactive, and printer jobs run too slowly.	• Make sure both the ATMONITOR subsystem and the ATMONITOR job within the subsystem are active. If they aren't, start them. • Increase the minimum pool size of *BASE by 20%.	• If you are using IBM iSeries Access for Windows, increase the activity level PF of the *BASE pool by 1. • Let Robot Autotune manage your server jobs by turning on server job tuning.
It takes too long for users to sign on.	Raise the minimum activity level for the interactive pool by 1 or 2.	
Remote users need better response time.	Have Robot Autotune manage your server and communication jobs.	
Interactive jobs have bad response time even when no batch jobs are running.	Check the minimum activity level listed for your interactive pool on the Standard Pools Performance Factors panel. If it is substantially above the minimum listed on the Pool Size Factors Calculation Chart, reduce it by entering a lower number.	• Make sure the machine pool size is correct. • Make sure the interactive maximum pool size is set at ***AVAIL**. • If your disk utilization is over 88%, consider purchasing another disk drive. If not, add more memory.
Printers stop in the middle of printing a report.	Raise the minimum activity level for the printer pool (QSPL or *SPOOL) by 1 or 2.	Raise the printer pool maximum by 1 MB, or more.

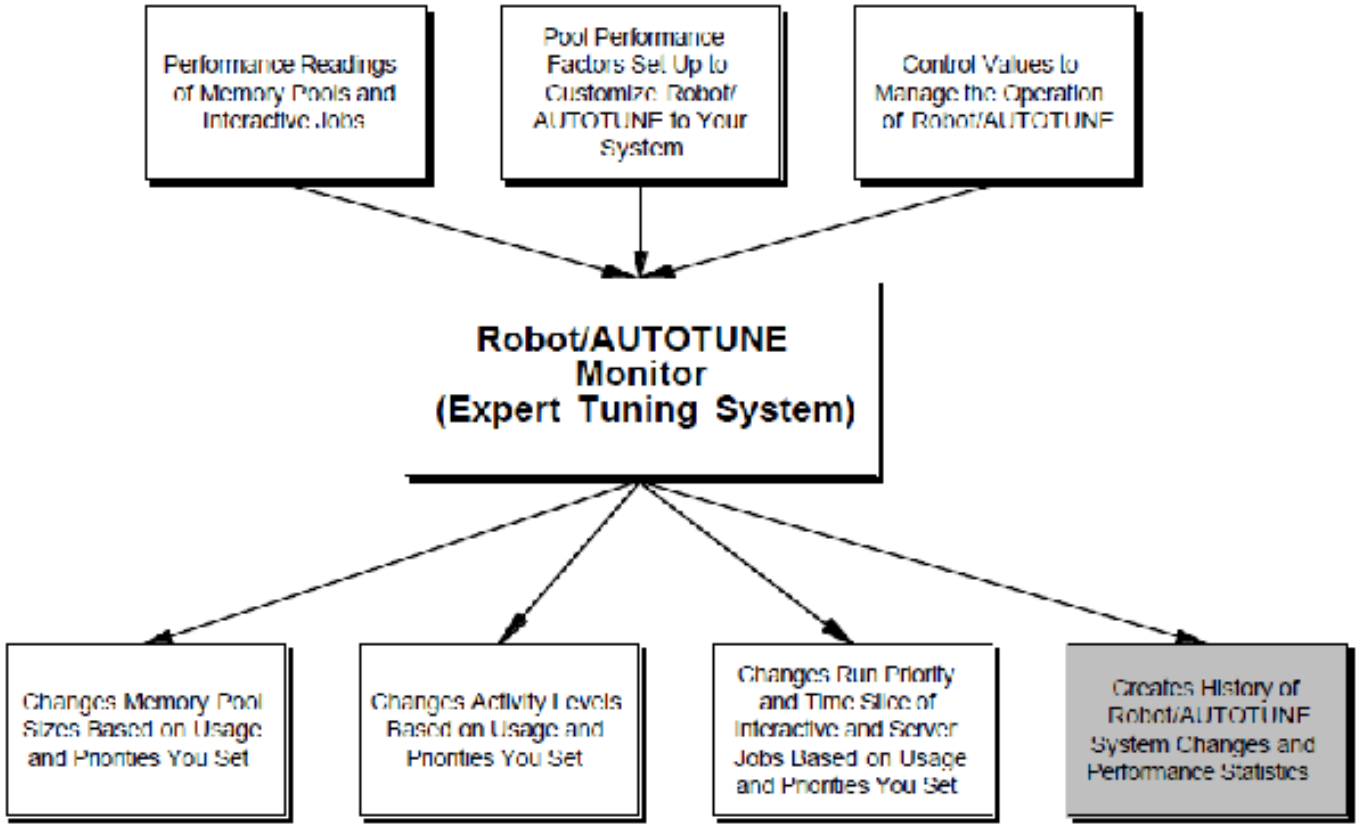
If you have this problem:	First, do this:	Then, try this:
Dynamic pools are running in System Pool 2 (*BASE).	Check to see if both the ATMONITOR subsystem and ATMONITOR job within the subsystem are active. If not, start them.	Check to see if all the system pools are being used. When all 64 system pools are assigned to subsystems or dynamic pools, additional dynamic pool jobs run in System Pool 2 (*BASE).
Job queues are assigned to run in dynamic pools, but they are still running under the batch subsystems.	Start the ATMONITOR subsystem before you start the batch subsystem. Dynamic pool job queue entries are defined to both the ATMONITOR subsystem and the batch subsystem. The subsystem that is started first is assigned the job queue by the operating system. If you assign all the job queues in the batch subsystem to dynamic pools, you don't need to start the batch subsystem.	
Some subsystems are not showing up on the Auto Tune Standard Pools Performance Factors panel.	Enter a **1** in the **Opt** field next to any subsystem with an Opt field to display the list of pools that share the pool. You may find the subsystem that didn't show up is sharing a pool like *BASE, INTERACT, or *SPOOL. Refer to "Getting your iSeries in Tunable Condition" if you need instructions about putting an existing subsystem in its own pool.	
System Console is receiving messages saying subsystem ineligible.	Raise minimum activity level by 1 - 2 in *BASE pool.	
Pool sizes and activity levels are not changing.	Check to see that both the ATMONITOR subsystem and the ATMONITOR job are active. If not, start them.	

If you have this problem:	First, do this:	Then, try this:
Installed a new iSeries (or upgraded one) and Robot Autotune doesn't work.	Changing the serial number or the model number of the system requires a new security code. Contact Robot Technical Support for assistance.	
Robot Autotune ended with the error MCH1002.	Check to see if the following objects are owned by QSECOFR: AT200S and AT400S. If not, change ownership to QSECOFR.	You may have a security code problem. Check your Security code and call Robot Technical Support for assistance.
MCH0603 occurs when starting Robot Autotune.	Go to the Auto Tune Menu. Select Option 1 and press Enter. Then select Option 2 and press Enter.	You may have a security code problem. Check your Security code and call Robot Technical Support for assistance.
Robot Autotune went down and ATCTLQ was damaged.	Take screen prints of options 1, 2, and 4 from the Robot Autotune main menu, and take screen prints of all three control value panels under option 3. Call ATRESET to delete and re-create the data queue. Then, reenter your performance factors and control values from the screen prints.	
Upgraded OS/400 and Robot Autotune doesn't work.	Call Robot Technical Support. Since Robot Autotune works so closely with the operating system, a different release is required when you upgrade.	
We try to capture statistics at the busiest time of day on a large, busy system. The user space for storing statistics f lls up before we get to the part of day I am interested in.	Robot Autotune creates the user spaces for statistics at the time the day environment begins. Make your control values for the day and night environments the same temporarily. Begin the day environment close to the time of day for which you want to capture statistics.	

Notes:

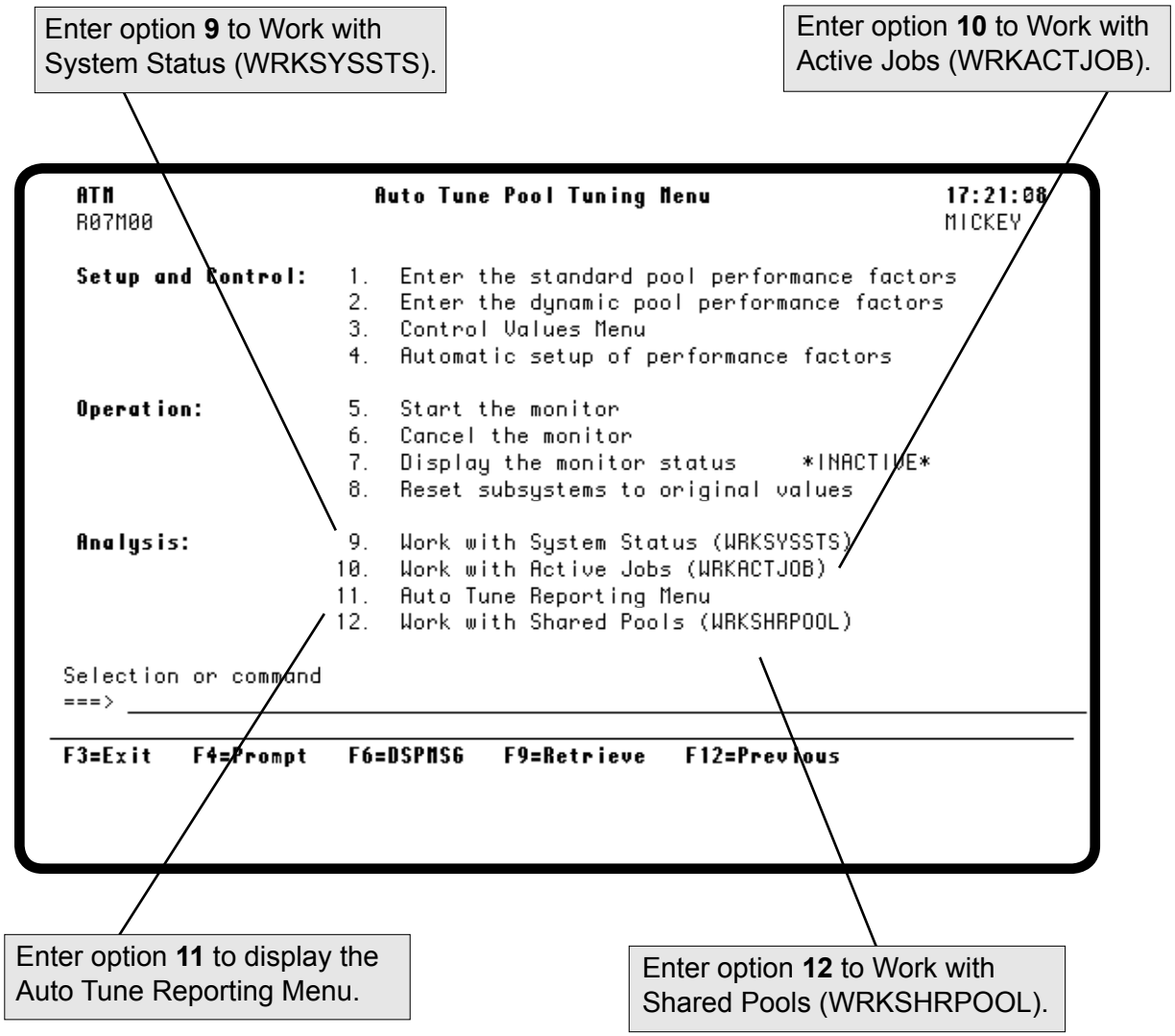
Statistics

Printing and Displaying Robot Autotune Statistics



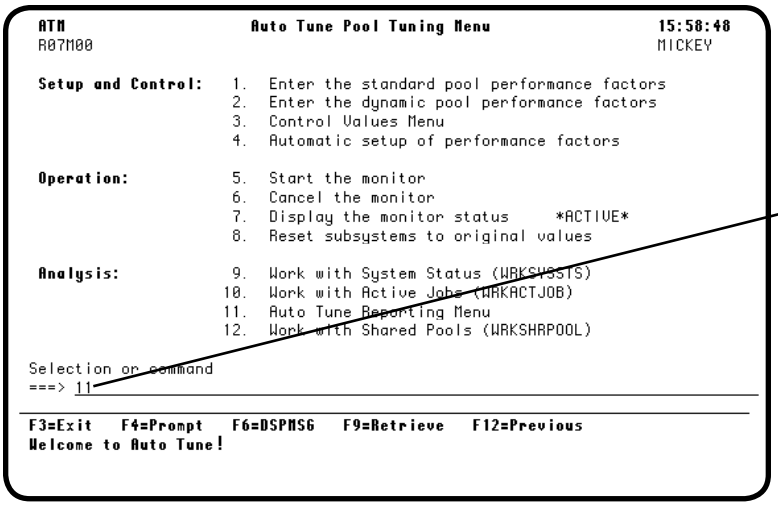
Working with Robot Autotune Statistics

The third major section of the Auto Tune Menu offers easy access to three IBM commands—WRKSYSSTS, WRKACTJOB, and WRKSHRPOOL, as well as access to a separate reporting menu. This reporting menu allows you to display Robot Autotune statistics at your workstation, print them, or place them in an iSeries database file for use by customer-written applications or queries. In addition, you can use Robot Monitor, the performance monitoring software from Fortra, to access and graph some of these statistics.



Accessing the Auto Tune Reporting Menu

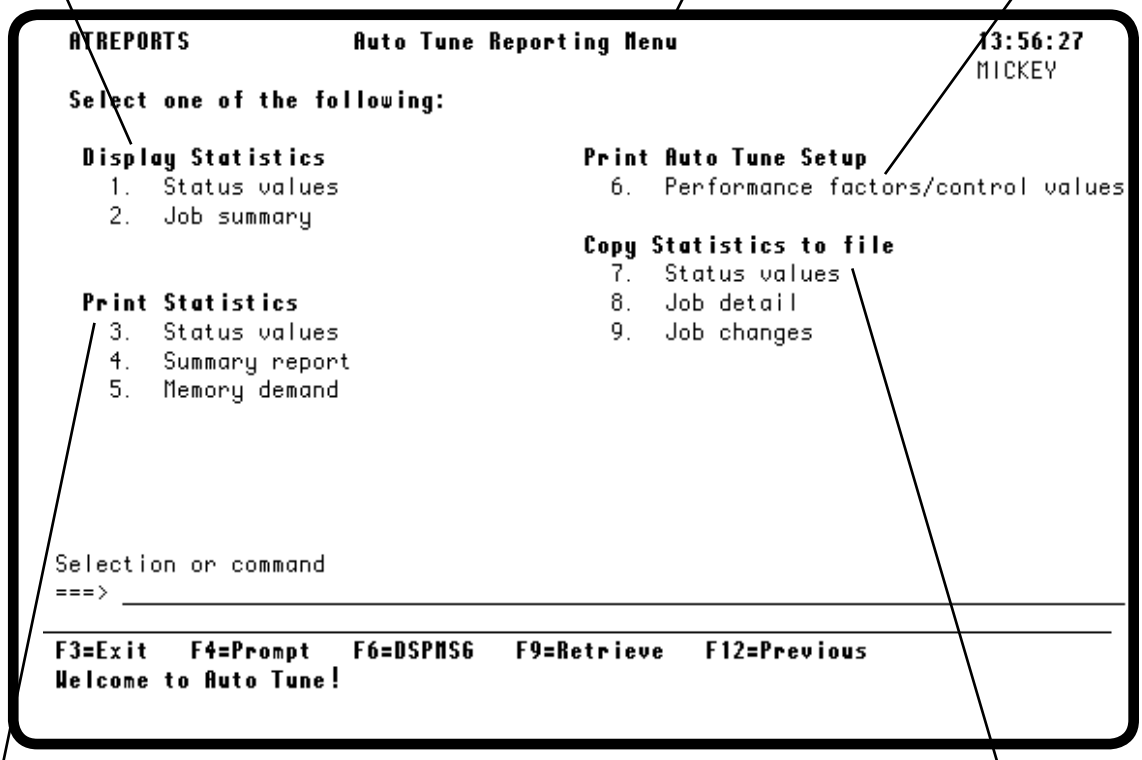
If you told Robot Autotune to save statistics on one of the control value panels, the Auto Tune Reporting Menu allows you to display or print these statistics or to copy the statistics into a database file. You can also print setup information from this menu.



Enter option **11** to display the Auto Tune Reporting Menu.

Select option **6** to print Auto Tune Performance Factors and Control Values that you have set up. You can print these values and save the print out in case setup values change.

Select option **1** or option **2** to display Robot Autotune statistics on your workstation.

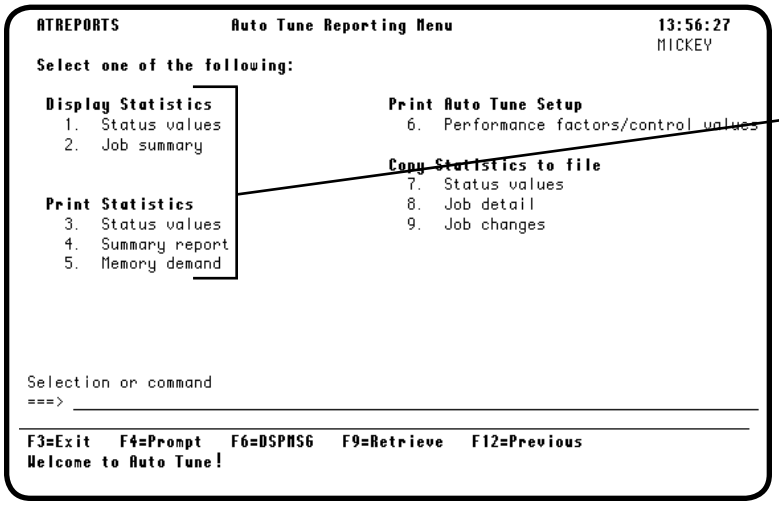


Select options **3**, **4**, or **5** to print statistics.

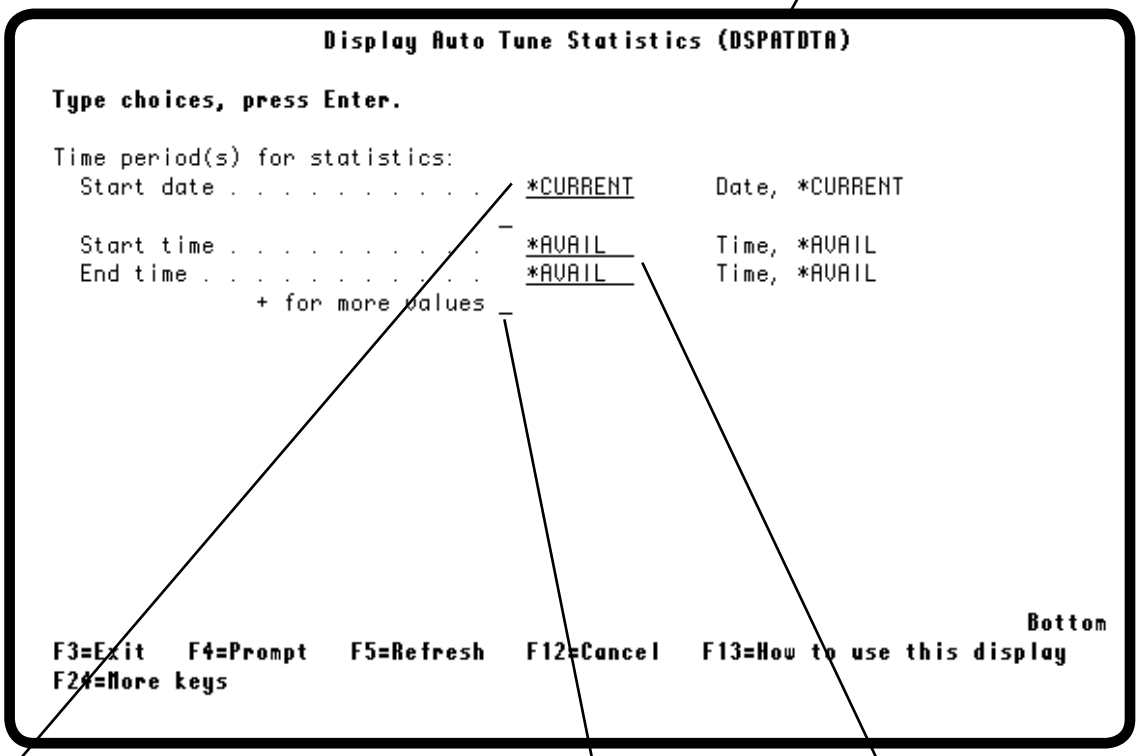
Select options **7, 8,** or **9** to copy statistics to an iSeries database file for use by customer-written applications or queries.

Display Auto Tune Statistics Prompt (DSPATDTA)

If you want to display or print Robot Autotune statistics, you can choose options 1 through 5 on the Auto Tune Reporting Menu. After you select the menu option, the Display Auto Tune Statistics (DSPATDTA) prompt panel displays. Use this panel to enter the date and time period for the statistics to be displayed on your workstation or printed in the report.



Select options **1** to **5** to display or print Robot Autotune statistics. Each time, the DSPATDTA prompt panel displays and you enter the date and time periods for the statistics (as shown below). Press Enter to continue.



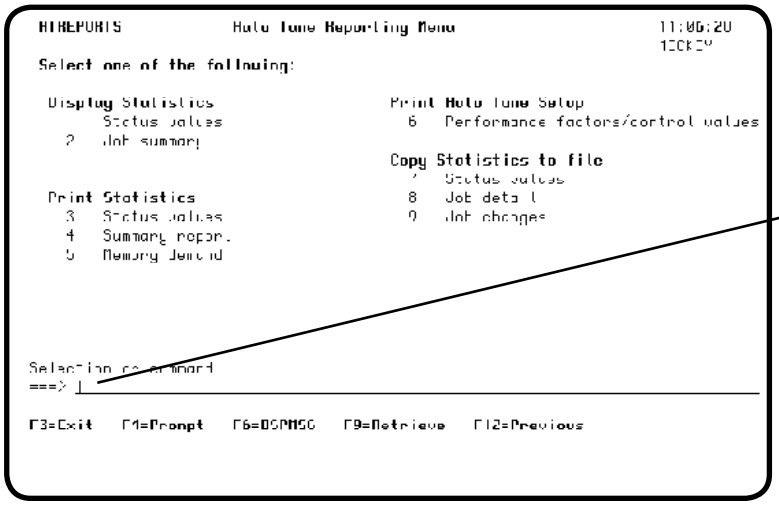
Enter a start date in system date format. The default, *CURRENT, gives you the current date.

Enter a plus sign (+) in this field to specify multiple time periods.

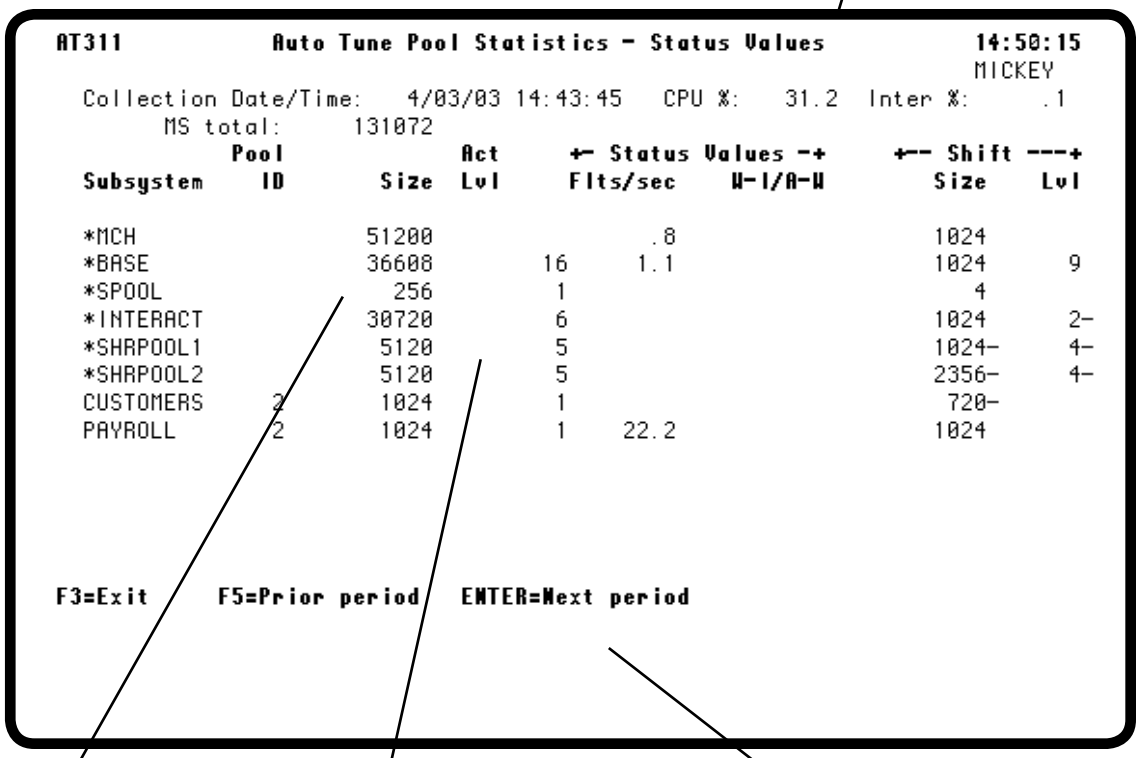
Enter a start and end time for the statistics (for example, enter 3:30 p.m. as 153000). *AVAIL lists all available statistics.

Displaying Status Values

Select option **1** on the Auto Tune Reporting Menu to display status values.



Select option **1** to display Status Values statistics. Fill in the date and time range on the DSPATDTA panel and press Enter to display the Auto Tune Pool Statistics - Status Values panel.



The pool size (in KB) at time of reading.

The activity level at time of reading.

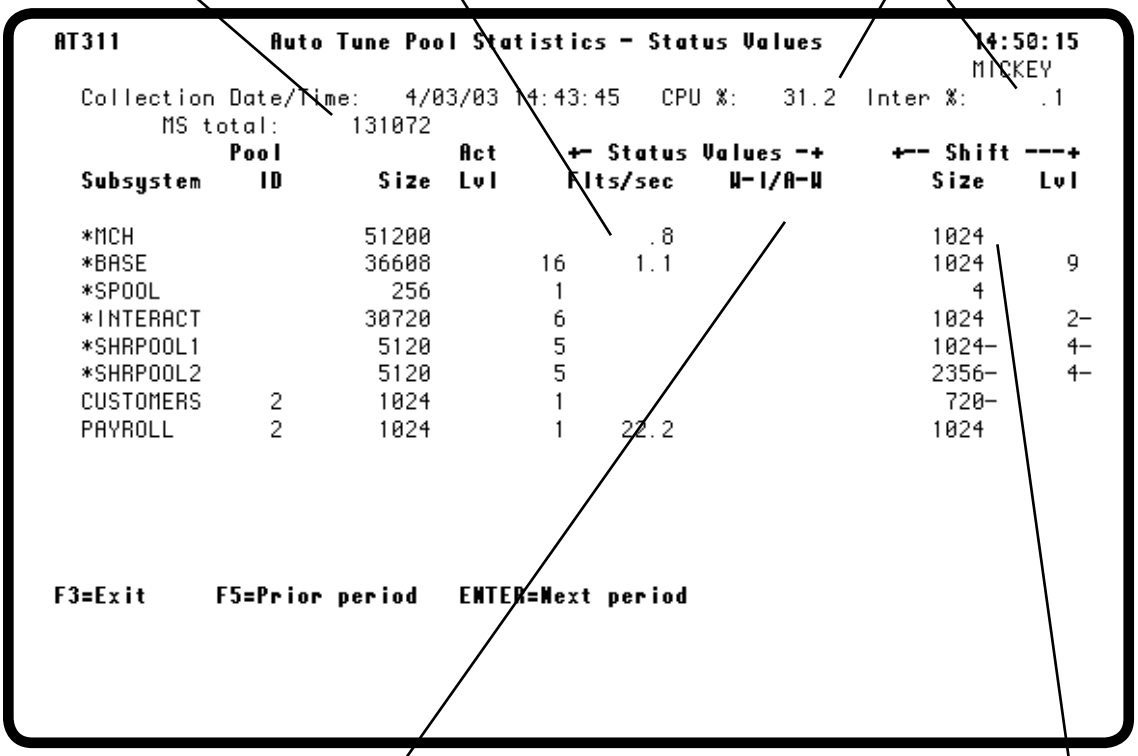
Press Enter to display the statistics for the next period.

Displaying Status Values

The database and non-database faults per second rate. See the appropriate IBM documentation for information about faulting guidelines and acceptable fault rates for various systems.

Indicates the amount of the total main storage (memory) available to the system in KB.

Indicates the percentage of total CPU and the percent of CPU used by interactive jobs in the system.

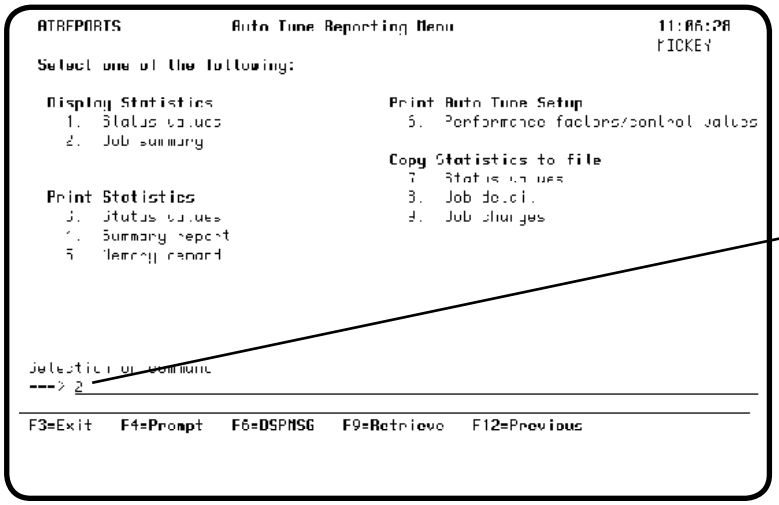


Shows the value of the wait-to-ineligible transition rate divided by the active-to-wait transition rate. See the appropriate IBM documentation for a complete discussion of acceptable values for various systems.

Robot Autotune adjustments made to pool size and activity level based on this reading.

Displaying Job Summary Statistics

Select **option 2** on the Auto Tune Reporting Menu to display job summary statistics.



Select option **2** to display Job Summary statistics. Fill in the date and time range on the DSPATDTA panel and press Enter to display the Auto Tune Pool Statistics with Job Summary panel.

Faults at time of reading.

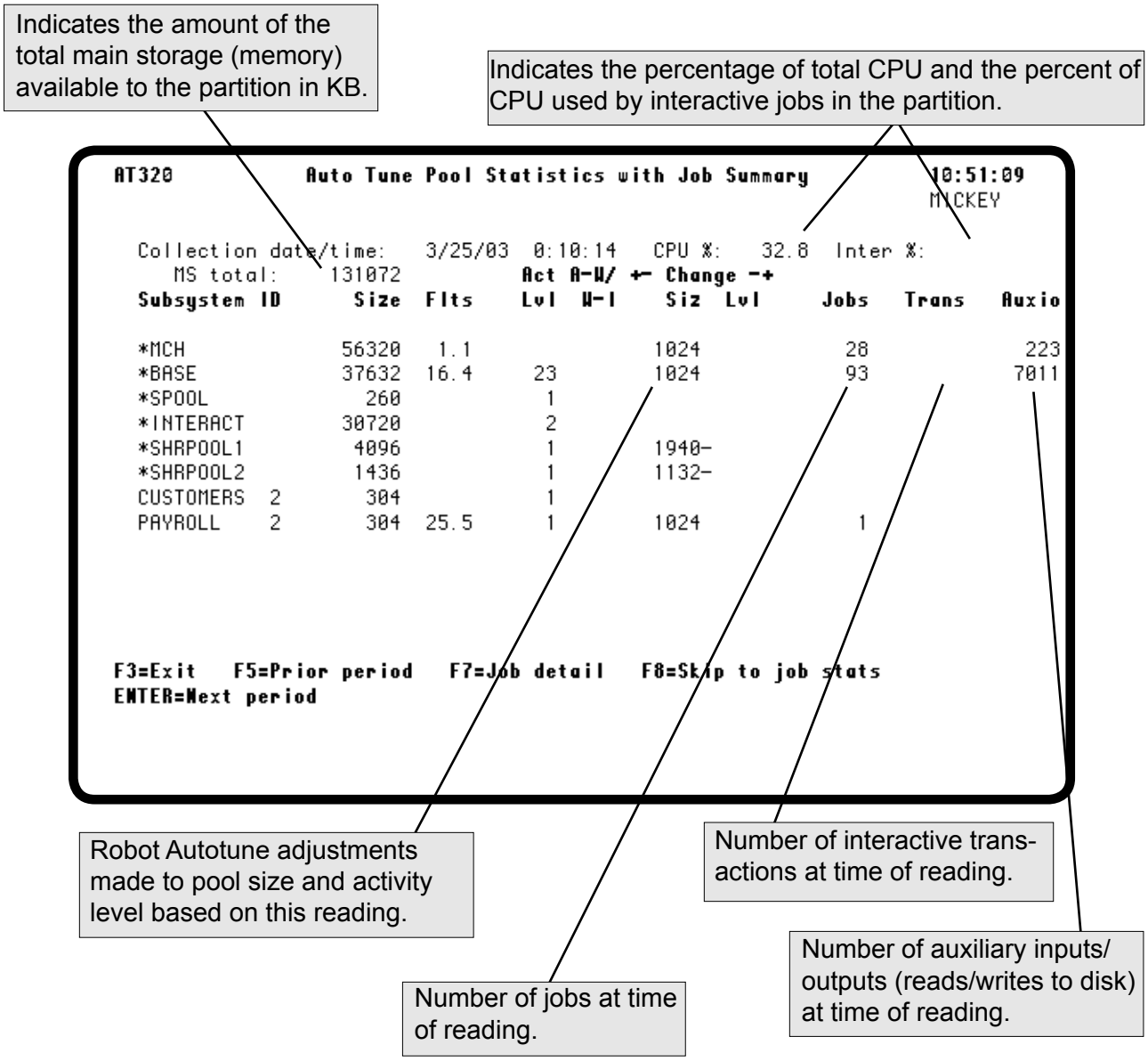
Activity level at time of reading.

```
AT320          Auto Tune Pool Statistics with Job Summary          10:51:09
                                                                   MICKEY

  Collection date/time:   3/25/03  0:10:14   CPU %:   32.8  Inter %:
     MS total:    131072          Act A-W/ +- Change -+
  Subsystem ID      Size  Flts    Lvl  W-I    Siz  Lvl     Jobs   Trans   Auxio

  *MCH             56320   1.1                1024            28             223
  *BASE            37632  16.4     23         1024            93            7011
  *SPOOL             260             1
  *INTERACT        30720             2
  *SHRPOOL1         4096             1         1940-
  *SHRPOOL2         1436             1         1132-
  CUSTOMERS  2       304             1
  PAYROLL    2       304  25.5       1         1024             1



F3=Exit   F5=Prior period   F7=Job detail   F8=Skip to job stats
ENTER=Next period
```

Press Enter to display the statistics for the next period.

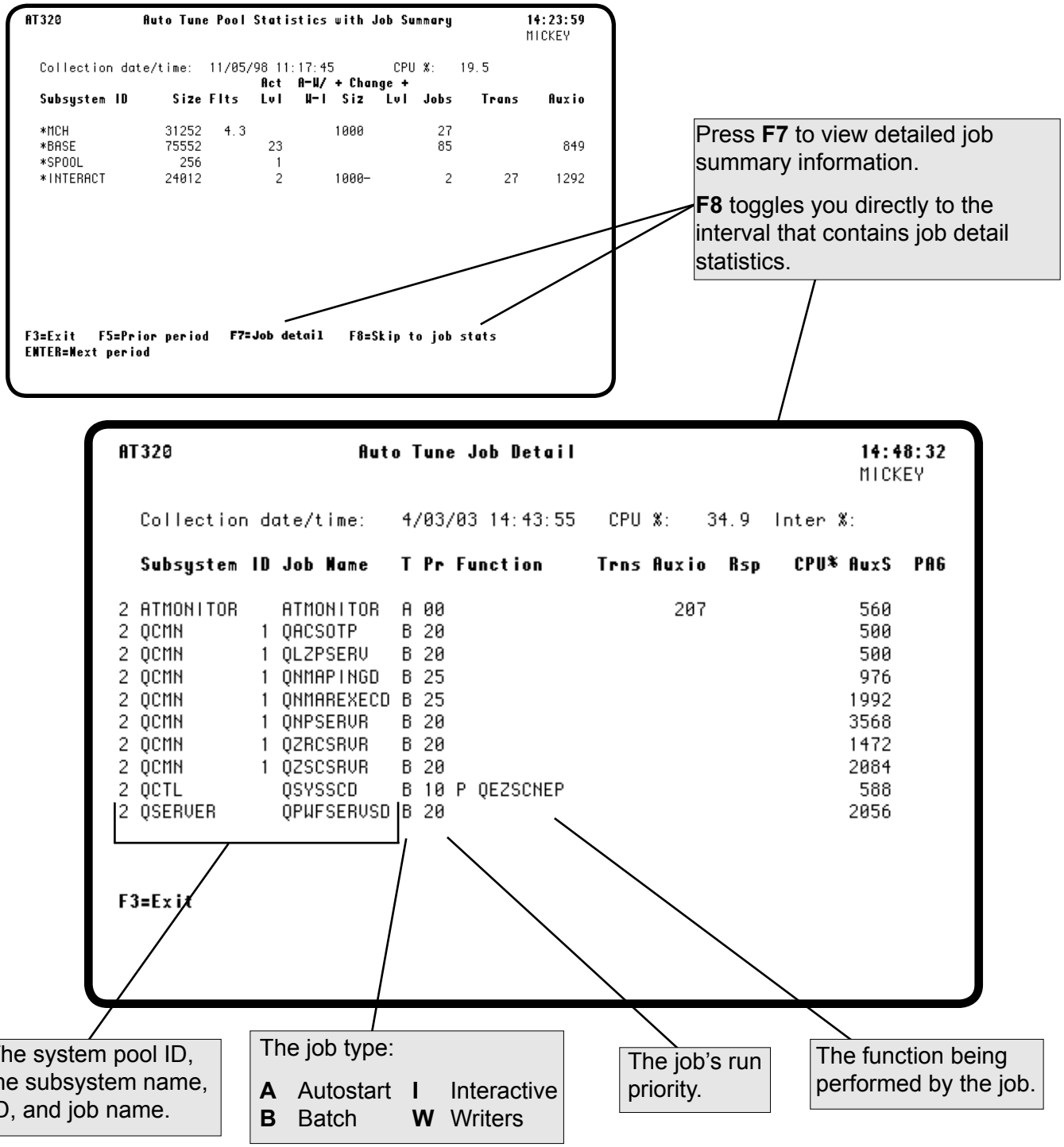
Pool size (in KB) at time of reading

Displaying Job Summary Statistics

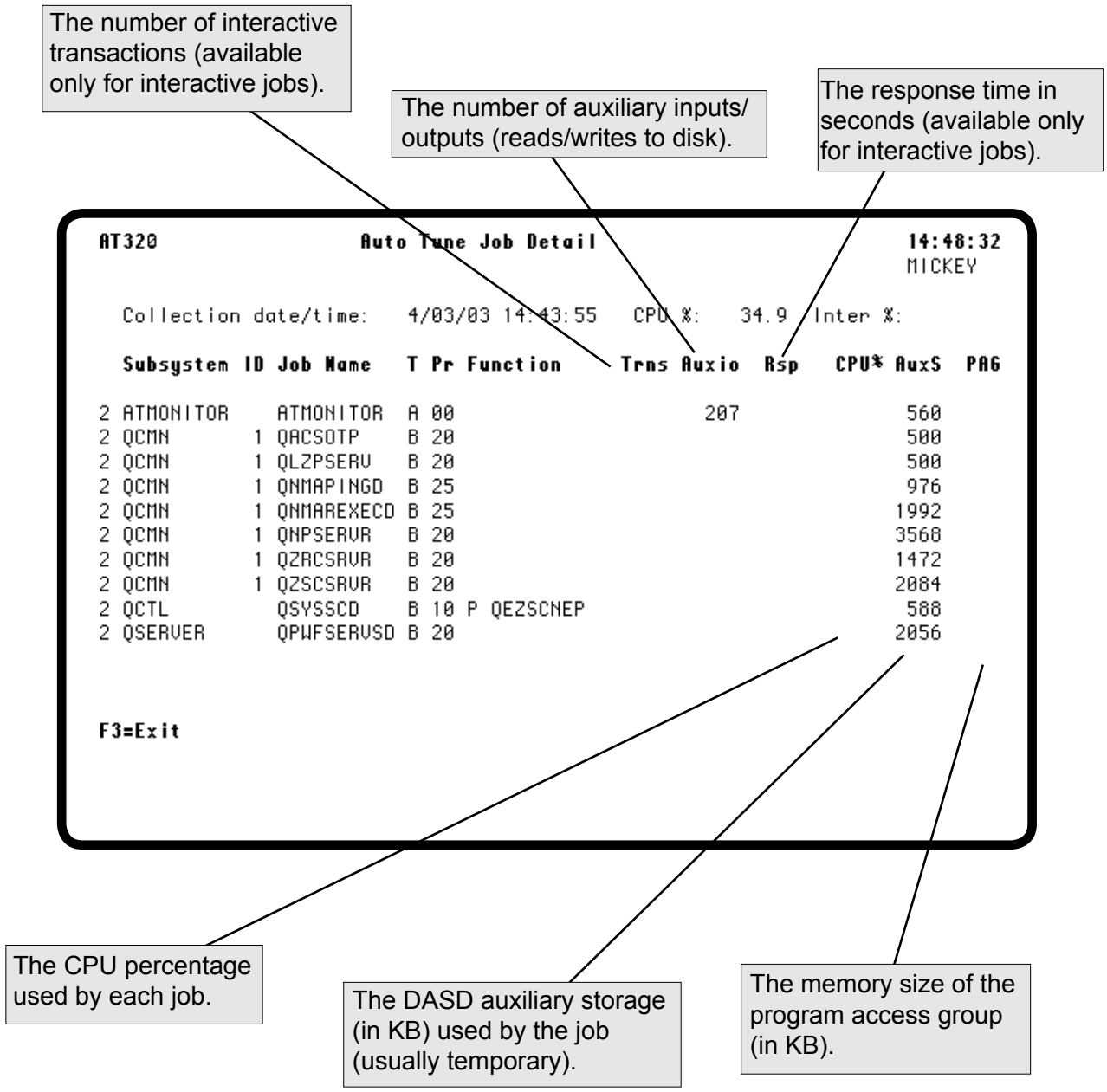


Displaying Job Detail

You can view detailed job summary statistics for jobs at certain intervals if you have selected to save job statistics on the Auto Tune Interactive Job Tuning Values panel. **F7**, Job Detail, will appear on the Auto Tune Pool Statistics with Job Summary panel according to the "number of intervals between job sampling" value you have entered on the Control Values panel. The values that appear are collected between job summary intervals. For instance, if you enter 30 as the number of intervals, the Job Detail function key option is available for every 30th collection interval.

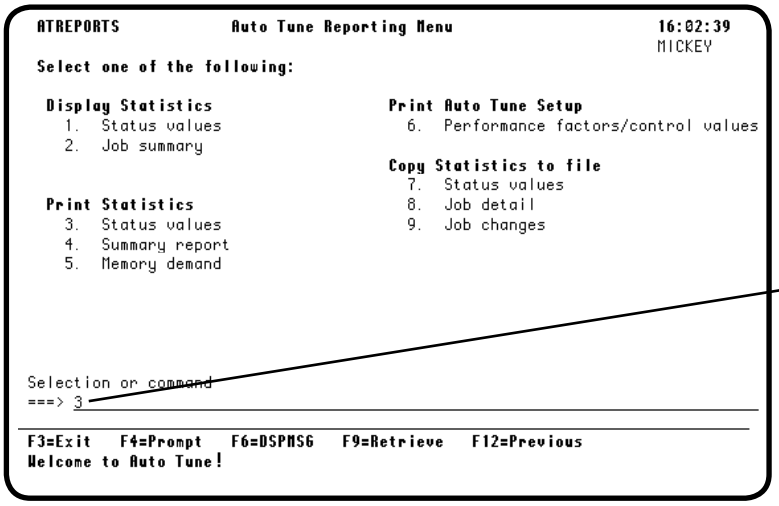


Displaying Job Detail

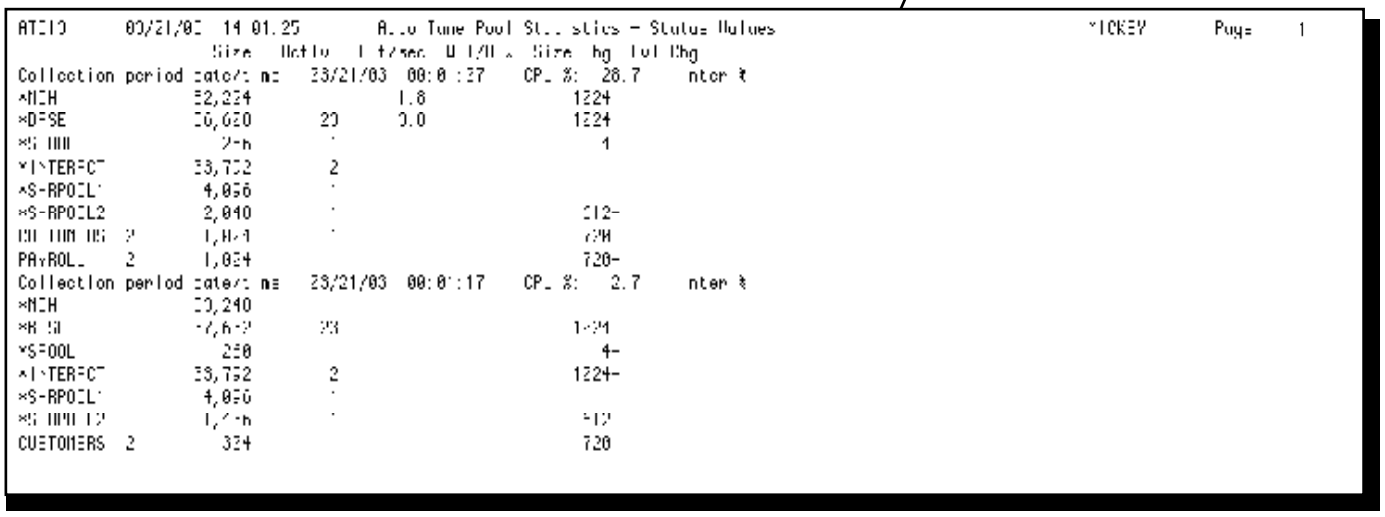


Printing Status Values

The Status Values Report includes the same information found on the Status Values display. For a description of the fields found on this report, see the discussion on the Auto Tune Pool Statistics - Status Values panel.



Select option **3** to print the Status Values statistics. Fill in the date and time range on the DSPATDTA panel and press Enter to submit the report.

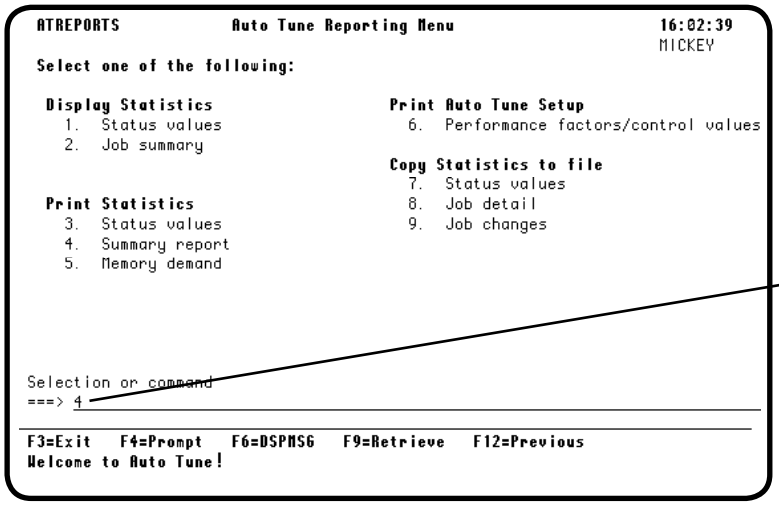


Printing the Summary Report

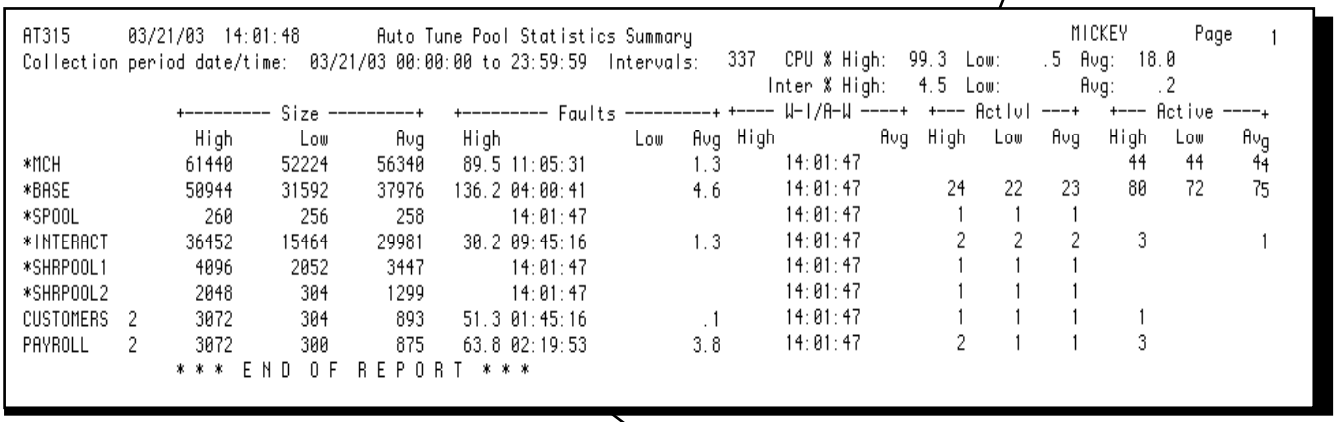
Robot Autotune creates a very useful summary report that shows the following for each period requested:

- Pool size minimum and maximum
- Activity level minimum and maximum
- Active jobs minimum and maximum
- Total faults minimum and maximum, with times they occurred
- Transition rate ratio minimum and maximum, with times they occurred
- Average of each of the above

You should print a report for one day of every week and save it. A series of declining performance reports can provide good proof when equipment upgrades are needed.



Select option **4** to print Summary Report statistics. Fill in the date and time range on the DSPATDTA panel and press Enter to submit the report.



AT315 03/21/03 14:01:48 Auto Tune Pool Statistics Summary MICKEY Page 1

Collection period date/time: 03/21/03 00:00:00 to 23:59:59 Intervals: 337 CPU % High: 99.3 Low: .5 Avg: 18.0
Inter % High: 4.5 Low: Avg: .2

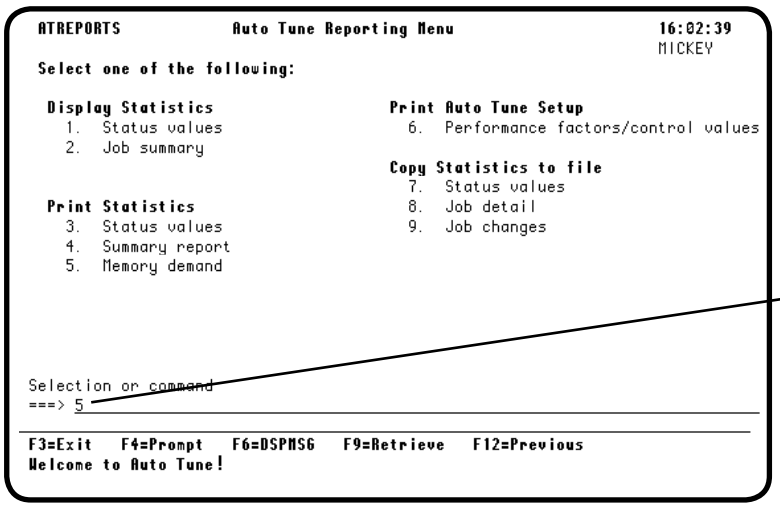
		Size High	Size Low	Size Avg	Faults High	Faults Low	Faults Avg	W-I/A-W High	W-I/A-W Avg	ActIvl High	ActIvl Low	ActIvl Avg	Active High	Active Low	Active Avg
*MCH		61440	52224	56340	89.5 11:05:31		1.3	14:01:47					44	44	44
*BASE		50944	31592	37976	136.2 04:00:41		4.6	14:01:47		24	22	23	80	72	75
*SPOOL		260	256	258	14:01:47			14:01:47		1	1	1			
*INTERACT		36452	15464	29981	30.2 09:45:16		1.3	14:01:47		2	2	2	3		1
*SHRPOOL1		4096	2052	3447	14:01:47			14:01:47		1	1	1			
*SHRPOOL2		2048	304	1299	14:01:47			14:01:47		1	1	1			
CUSTOMERS	2	3072	304	893	51.3 01:45:16		.1	14:01:47		1	1	1	1		
PAYROLL	2	3072	300	875	63.8 02:19:53		3.8	14:01:47		2	1	1	3		

*** END OF REPORT ***

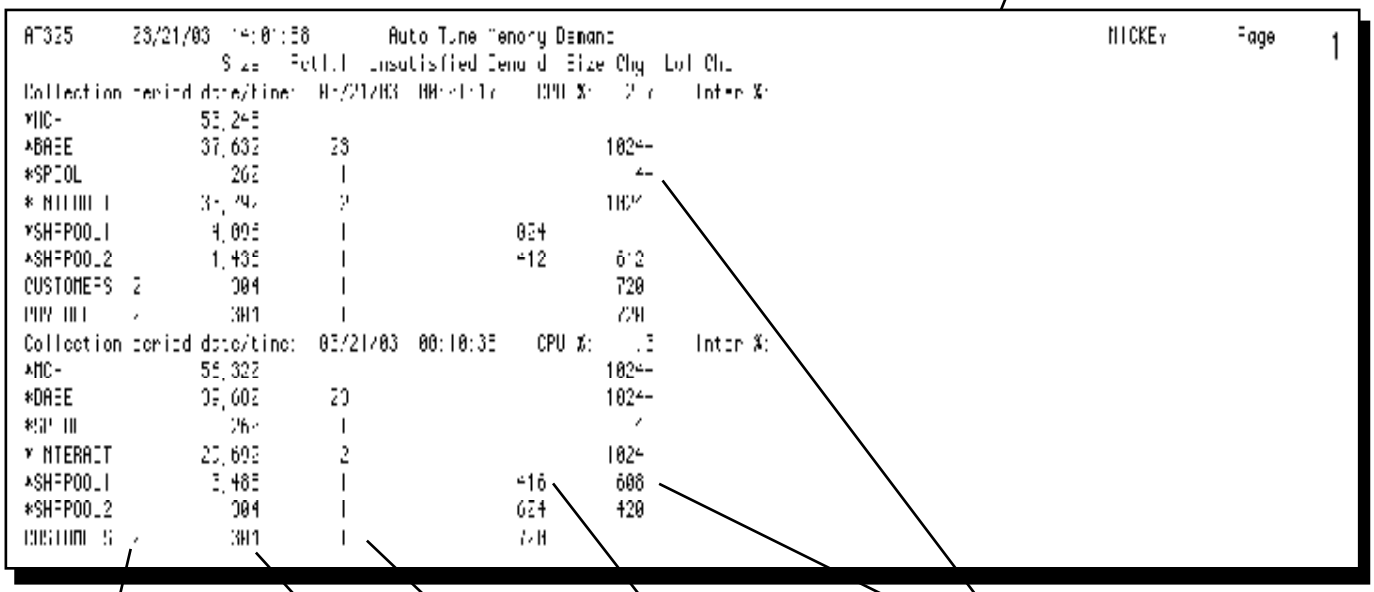
For more information about acceptable fault rates and Wait-to-Ineligible/ Active-to-Wait ratios, refer to the appropriate IBM documentation.

Printing Memory Demand Statistics

The Auto Tune Memory Demand report lists the instances when Robot Autotune needed more memory to properly tune a pool, but there wasn't enough available. This report can help you determine whether you need a memory upgrade, or whether you need to make adjustments to your performance factors.



Select option **5** to print Memory Demand statistics. Fill in the date and time range on the DSPATDTA panel and press Enter to submit the report.



Pool size (in KB) at time of reading.

Activity level at time of reading.

Robot Autotune adjustments made to pool size and activity level based on this reading.

The pool ID appears next to the pool name. If no ID is shown, the pool is shared. A "D" indicates a dynamic pool.

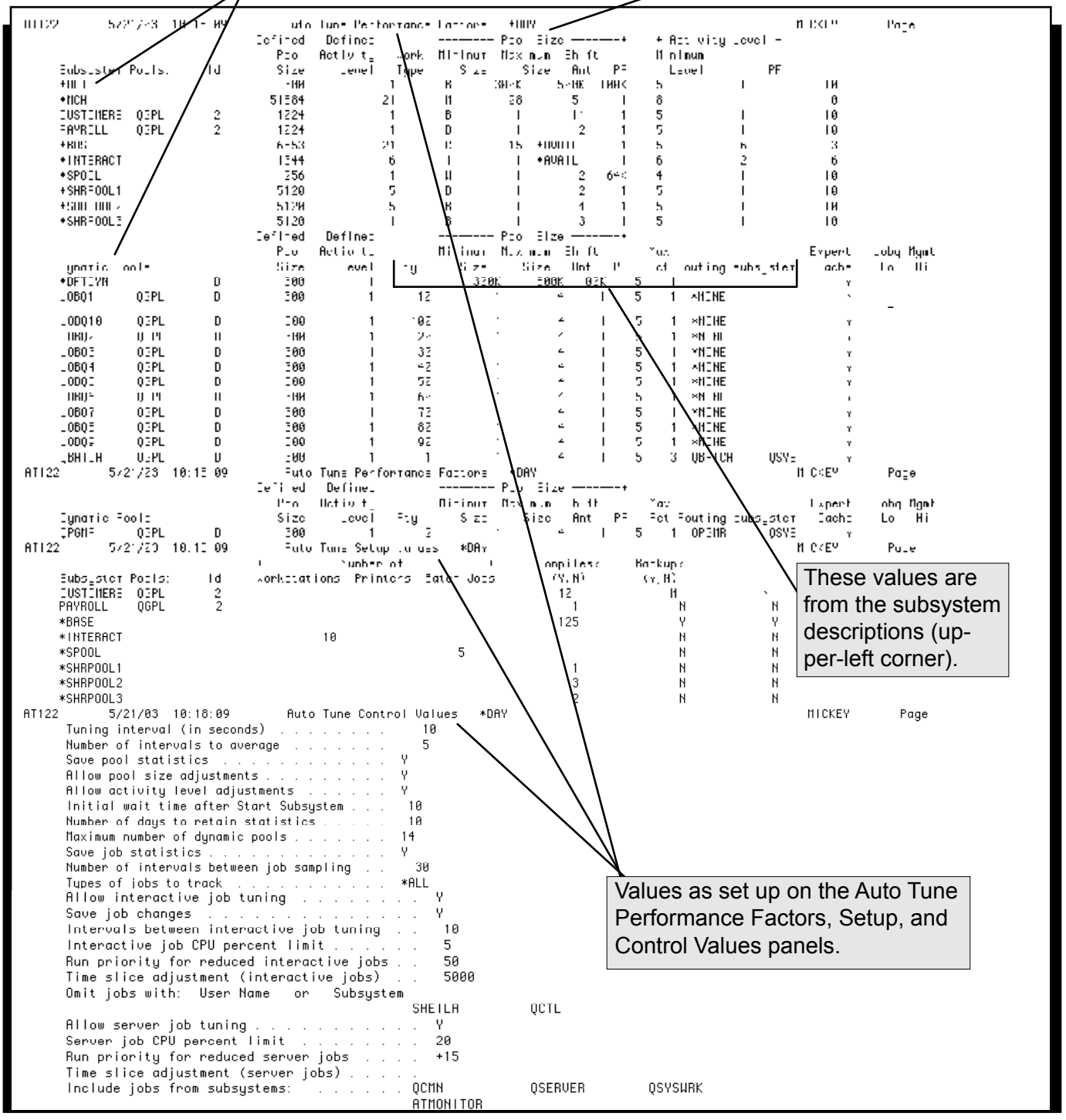
Amount of memory that was needed to properly tune the pool, but was unavailable at time of demand.

Printing Auto Tune Performance Factors and Control Values

You can print a report showing Robot Autotune performance factors and control values as you have defined them. Select option **6** on the Auto Tune Reporting Menu to submit the report (or issue the following command: **CALL ATLIB/AT122**.) You can save the printout to have a record of your setup values.

Robot Autotune uses the default values *DFT (standard pools) and *DFTDYN (dynamic pools) for pools that have not been defined to it.

The report lists day and night values on separate pages.

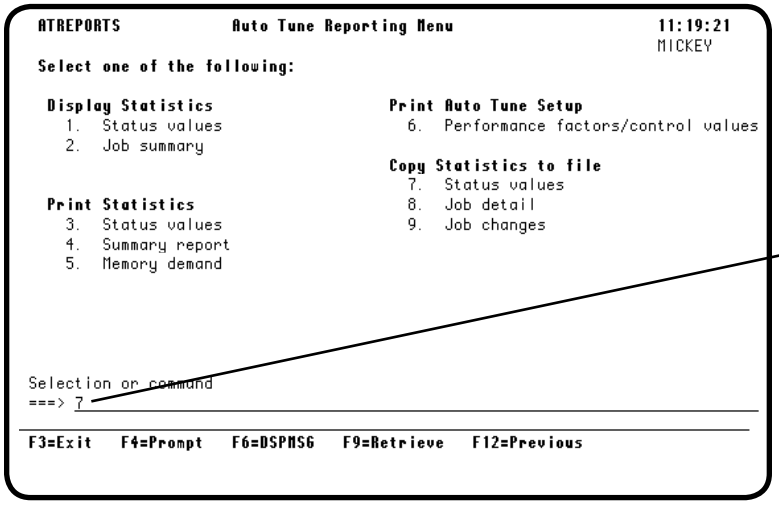


These values are from the subsystem descriptions (upper-left corner).

Values as set up on the Auto Tune Performance Factors, Setup, and Control Values panels.

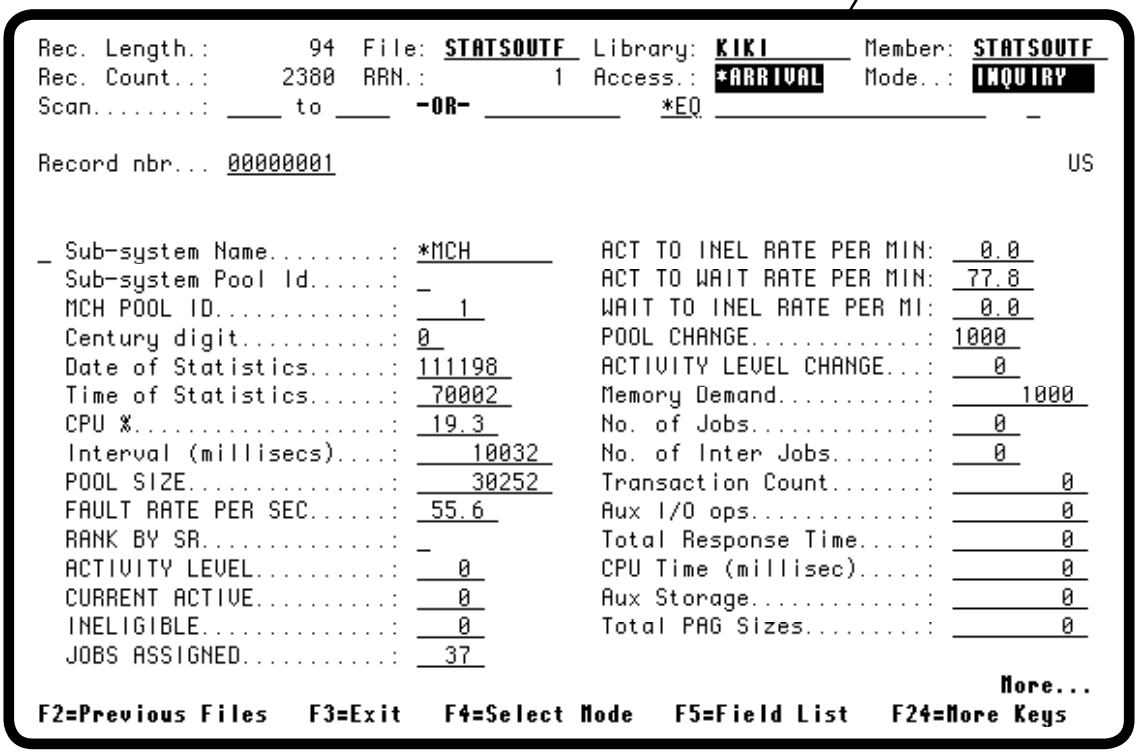
Copying Statistics to a Database File

Robot Autotune can create a database file of the statistics it generates, including job statistics, if they are being collected. You can process the statistics with your own programs. To copy Robot Autotune statistics to a database file, fill in the Outfile member name parameter on the DSPATDTA command. If the file does not exist, Robot Autotune creates it. If the file exists, Robot Autotune clears it and creates a record of each pool for each reading. The file has space for job statistics, but these fields are valid only for readings that have saved job statistics.



Select option **7** to copy Robot Autotune statistics to a database file. On the DSPATDTA panel, fill in the date and time range, Output File name, Library name, and Outfile Member Name.

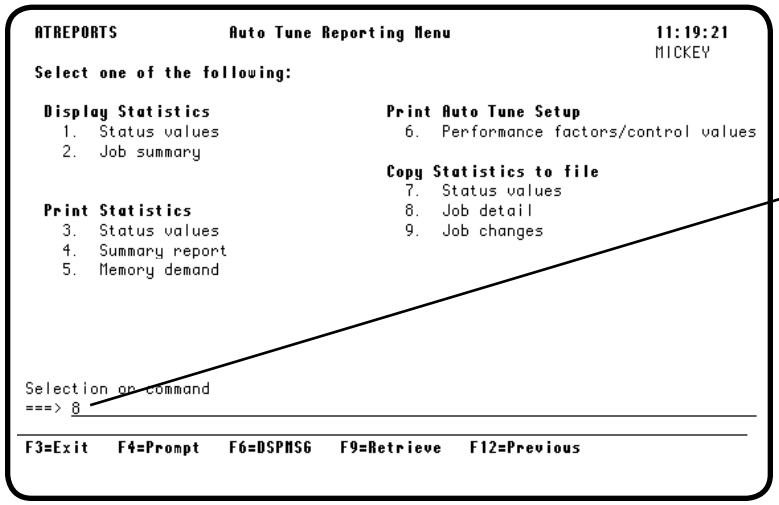
Easy View** display of Robot Autotune statistics copied to a database file.



***Easy View is a separate software package sold by Fortra.*

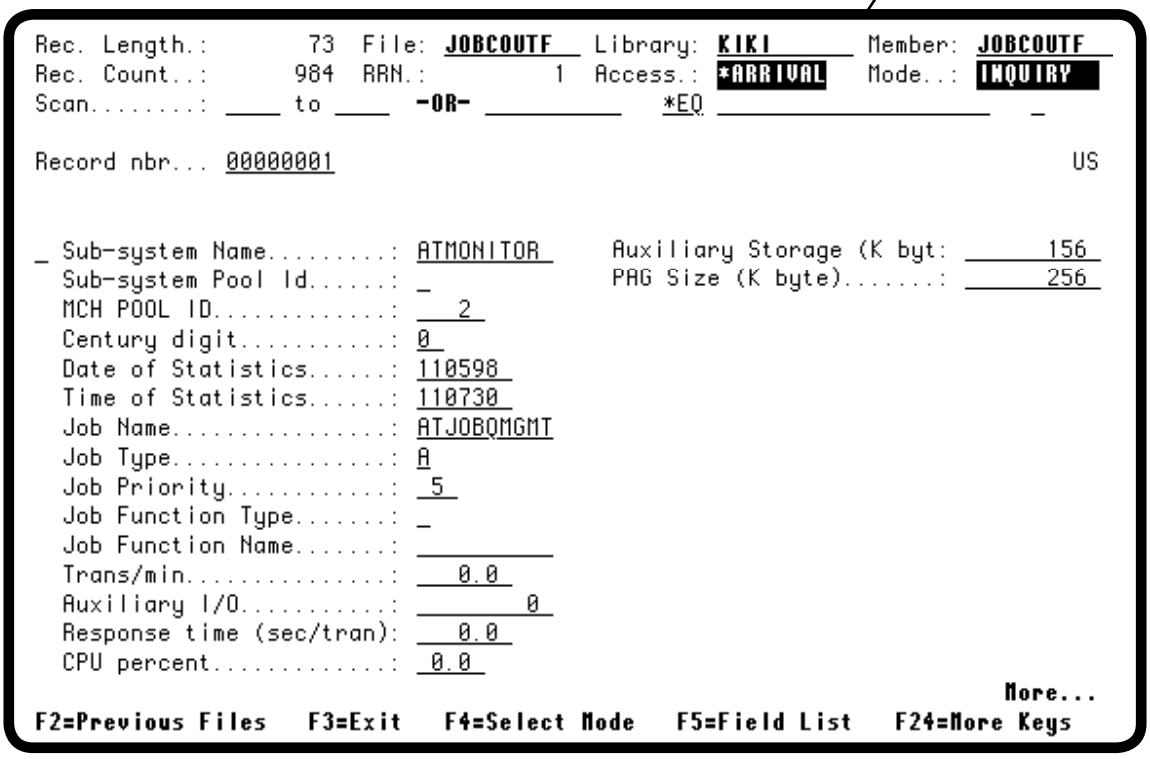
Copying Job Details to a Database File

If you are collecting job statistics, Robot Autotune can copy the information to a database file. You can then process the statistics with your own programs. Select option **8** on the Auto Tune Reporting Menu, or use the *JOB option on the DSPATDTA command. Fill in the Outfile member name parameter on the DSPATDTA prompt panel. If the file does not exist, Robot Autotune creates it. If the file already exists, Robot Autotune clears it. Then, Robot Autotune creates a record of each job for each reading when there were active jobs.



Select option **8** to copy Robot Autotune job statistics to a database file. On the DSPATDTA panel, fill in the date and time range, Output File name, Library name, and Outfile Member Name.

Easy View** display of the job detail statistics copied to a database file.

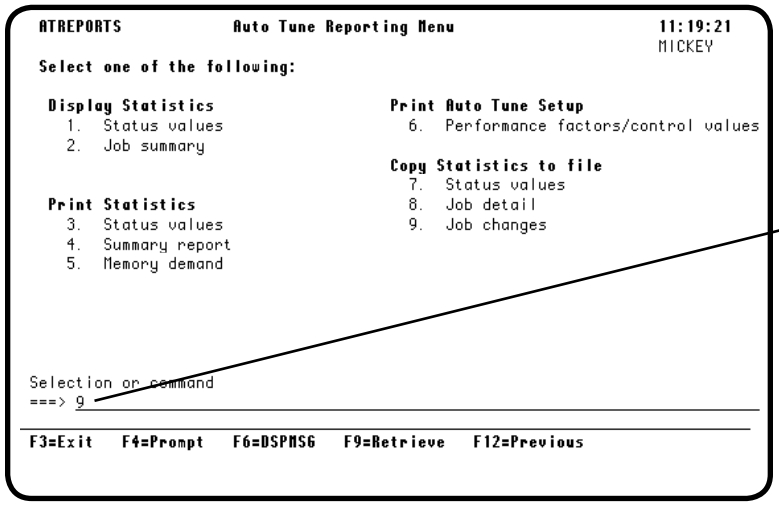


***Easy View is a separate software package sold by Fortra.*

Copying Job Changes to a Database File

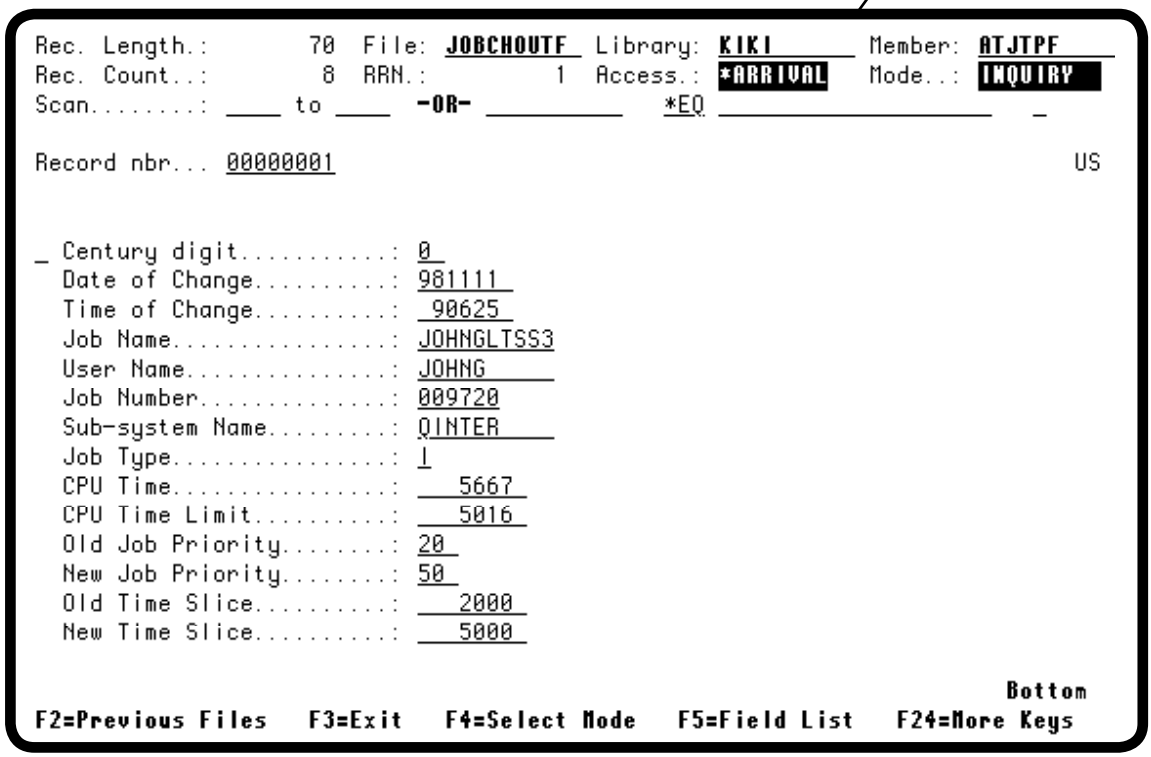
If you are saving job changes to interactive and server jobs, select option **9** on the Auto Tune Reporting Menu to copy the information to a database file. You can then process the statistics with your own programs. Fill in the Outfile member name parameter on the DSPATDTA prompt panel.

- If the file does not exist, Robot Autotune creates it.
- If the file exists, Robot Autotune clears it and creates a record of each interactive or server job tuned by Robot Autotune.



Select option **9** to copy Robot Autotune job changes to a database file. On the DSPATDTA panel, fill in the date and time range, Output File name, Library name, and Outfile Member Name.

Easy View** display of the job change statistics copied to a database file.



***Easy View is a separate software package sold by Fortra.*

Advanced Performance Tuning

Accessing the Auto Tune Performance Factors - Dynamic Pools Panel

Robot Autotune has a unique, advanced performance tuning feature—called Dynamic Pools—that can greatly reduce the impact of batch jobs on the performance of your iSeries. Robot Autotune doesn't require you to use dynamic pools, but they do provide some significant advantages.

```
ATM                      Auto Tune Pool Tuning Menu                  15:57:44
R07M00                                                               MICKEY

Setup and Control:   1.  Enter the standard pool performance factors
                     2.  Enter the dynamic pool performance factors
                     3.  Control Values Menu
                     4.  Automatic setup of performance factors

Operation:           5.  Start the monitor
                     6.  Cancel the monitor
                     7.  Display the monitor status     *ACTIVE*
                     8.  Reset subsystems to original values

Analysis:            9.  Work with System Status (WRKSYSSTS)
                    10.  Work with Active Jobs (WRKACTJOB)
                    11.  Auto Tune Reporting Menu
                    12.  Work with Shared Pools (WRKSHRPOOL)

Selection or command
===> 2

F3=Exit    F4=Prompt    F6=DSPMSG    F9=Retrieve    F12=Previous
```

Select option **2** on the Auto Tune Pool Tuning Menu to display the Auto Tune Performance Factors - Dynamic Pools panel.

Note: If you have not defined any dynamic pools, a message on the display indicates that no pools are defined.

```
AT128              Auto Tune Performance Factors - Dynamic Pools        10:14:13
                                Day environment                         MICKEY
Type options, press Enter.
  1=Add  2=Change  4=Delete
                                +---- Pool Sizes -----+
                                Minimum Maximum Shift     Max  Expert  Jobq Mgmt
Opt Job Queue  Library    Pty     Size    Size  Amt PF    Act   Cache    Lo   Hi
 _  QBATCH     QGPL       1        1       4    1  5     3     Y
 _  QPGMR      QGPL       2        1       4    1  5     1     Y
 _  JOBQ1      QGPL      10        1       4    1  5     1     N
 _  JOBQ2      QGPL      20        1       4    1  5     1     Y       0    1
 _  JOBQ3      QGPL      30        1       4    1  5     1     Y       0    1
 _  JOBQ4      QGPL      40        1       4    1  5     1     Y       0    1
 _  JOBQ5      QGPL      50        1       4    1  5     1     Y       0    1
 _  JOBQ6      QGPL      60        1       4    1  5     1     Y       0    1
                                                                    More...

                          PF = Performance Factor

F3=Exit                      F7=WRKSYSSTS   F8=Standard pools
F15=Night environment
```

Batch Jobs Need a Separate Pool

If batch jobs are allowed to run only in *BASE or in pools with interactive jobs, they can gobble up available CPU time and not let other jobs run properly. Similarly, two batch jobs running in the same pool will waste time fighting each other for system resources.

Robot Autotune's Dynamic Pools to the Rescue

If you let Robot Autotune manage your batch job queues, it takes care of these situations. Robot Autotune encloses each batch job in its own dynamic pool (running in the ATMONITOR subsystem) when the job starts. No other jobs can execute in this pool for the duration of the job. By giving each job its own pool, Robot Autotune reduces page faulting and maximizes performance.

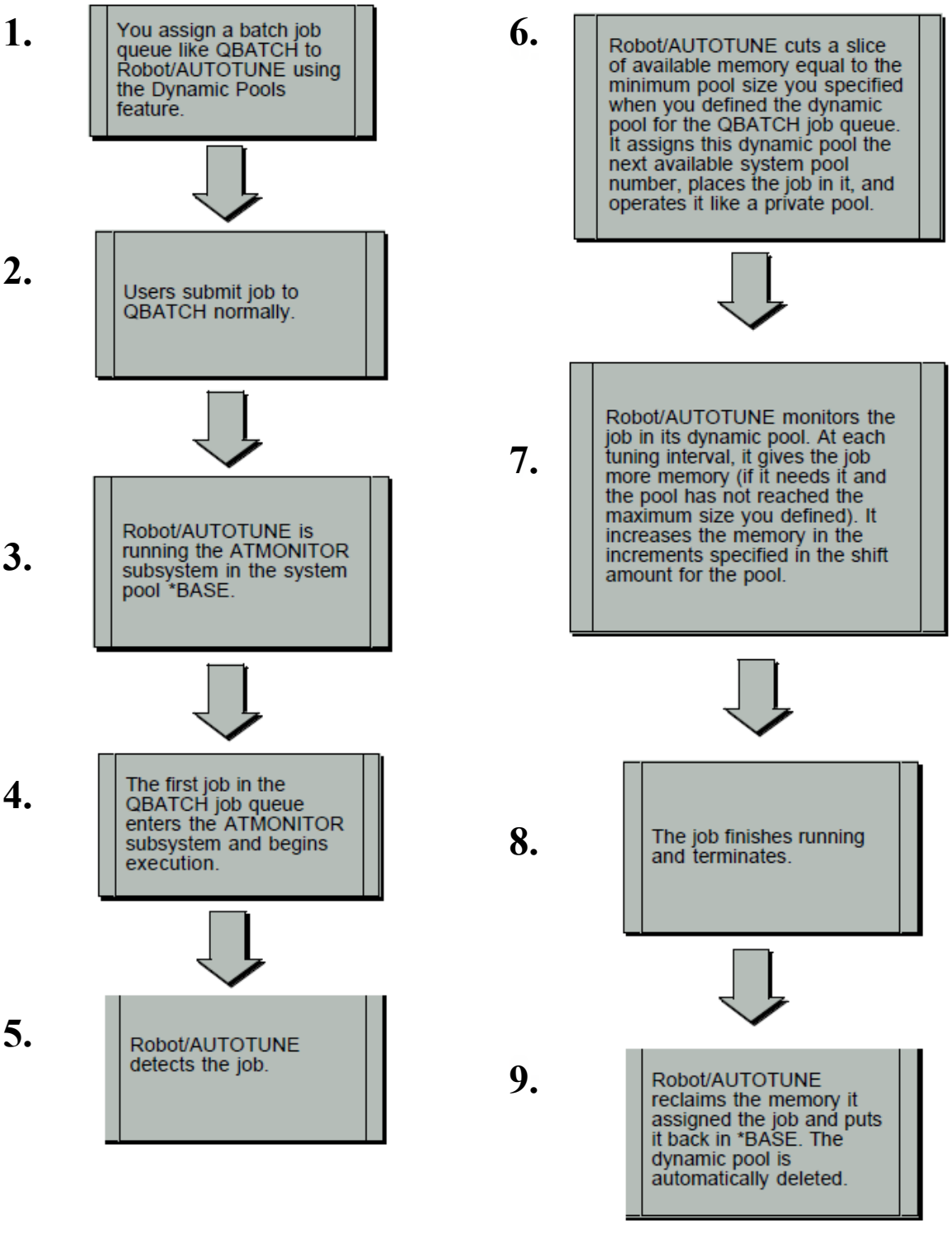
When the job finishes and no others are waiting in the queue to run, a dynamic pool is deleted after 12 Robot Autotune tuning cycles. Otherwise, dynamic pools remain on the system until another pool needs memory, or they are reused by a different job.

Thus, Robot Autotune automatically minimizes the impact of batch jobs on your system. For each job queue, you can enter minimum (at least 3 MB is recommended) and maximum pool sizes, shift amounts, and performance factors. You also have the option of placing job queues in a rotation group. Robot Autotune's job queue manager rotates the available dynamic pools among these job queues to allow them the opportunity to process. Jobs no longer wait on a low-priority job queue because they can’t process.

OS/400 systems can have a total of 64 system pools and the system pools *MCH and *BASE use up two of these pools. If you have also defined subsystems to use other system pools (such as *INTERACT and *SPOOL), the number of available pools is reduced further. The total number of pools should not exceed the OS/400 limit because if this pool limit is exceeded, Robot Autotune runs the job in *BASE.

The alternative to dynamic pools is to place each job queue in its own subsystem and have Robot Autotune manage each subsystem pool. Using dynamic pools is much easier than setting up multiple batch subsystems.

How Dynamic Pools Work



Converting Batch Subsystems to Dynamic Pools

To convert your existing batch subsystems to dynamic pools, follow these steps:

1. Enter the names of active batch job queues on the Auto Tune Performance Factors - Dynamic Pools panel. If you do not know the names of the job queues your system uses, enter the following command to display them:
   WRKJOBQ JOBQ(*ALL)
   Make sure the queue is a batch job queue before you enter it on your dynamic pools job queue list.
   The execution environment for dynamic pool jobs is determined by the subsystem name you enter in the routing entries field for the job queue. If you don't make an entry in that field, the class of the ATMONITOR subsystem routing entries is used. The normal batch values for execution priority and time slice are used. You can change them if you want.

2. If you change *all* the job queues in a batch subsystem to use dynamic pools, you no longer need to start the batch subsystem. Therefore, you should change your IPL procedures or program so that the changed batch subsystems are not started.
   If you want some job queues to run in dynamic pools and other job queues to execute in your batch subsystems, change your IPL procedures or program so that Robot Autotune starts before any other subsystem. Robot Autotune will allocate the job queues it manages and the batch subsystems will manage the remaining unallocated job queues.

3. The unused batch subsystems are still listed on the Auto Tune Performance Factors - Standard Pools panel. This is not a problem. We recommend you leave them there because you might want to use them in the future.

Note: If all job queues in a batch subsystem have been changed to use dynamic pools, you no longer need to start the batch subsystem. However, you may still want to start the subsystem after ATMONITOR so that if the ATMONITOR subsystem stops unexpectedly, your jobs will be able to run under the batch subsystem. If you do select this option, make sure the batch subsystems are defined using minimum values.

Changing Dynamic Pool Performance Factors

You can change performance factors on the Auto Tune Performance Factors - Dynamic Pools panel. Or, you can change any dynamic pool performance factor with the command **CHGATPF**. Instead of the subsystem name in the SBSD parameter, enter the job queue name and library. Then, enter ***DYN** in the POOL parameter.

For example, to set up the QBATCH job queue for nighttime operation, you could have ROBOT execute the following command at 17:30:

```
CHGATPF  SBSD(QGPL/QBATCH)  POOL(*DYN)  PSMAX(5200)
```

See the Operations section of this User Guide for a complete description of the CHGATPF command.

Dynamic Pools Run in the ATMONITOR Subsystem

Robot Autotune and dynamic pools run in the ATMONITOR subsystem. When you cancel the Robot Autotune monitor, the batch jobs currently running will finish normally. Any jobs remaining on the job queue will not run until the monitor is started again.

The dynamic pool definition uses the assigned subsystem description routing entry to determine the run priority and time slice for jobs. It will use the time slice and run priority defined for the class maintained for the routing entry. Other class values are ignored. If no subsystem is specified, the default is *NONE and the class definition for the ATMONITOR subsystem is used.

The job queue entries created for each job queue assigned to dynamic pools are normally single-threaded, allowing only one job to run at a time. If you change the Max Act field on the Dynamic Pools panel, you can allow multi-threaded job queues. Each thread runs in a separate dynamic pool.

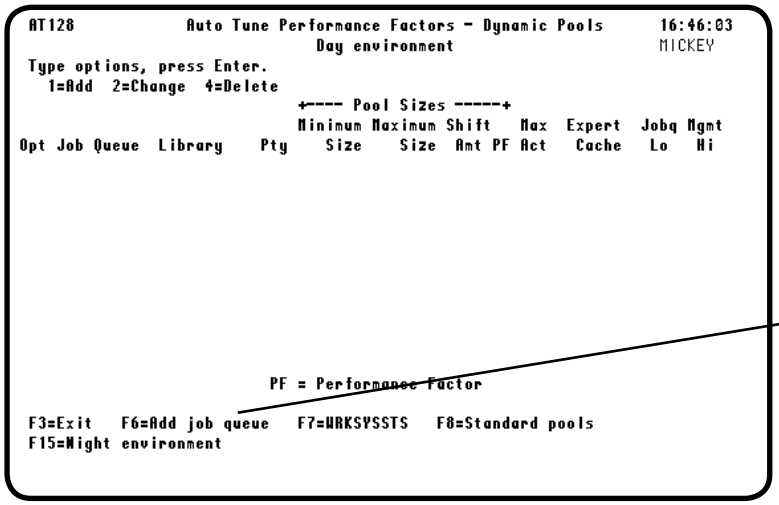
Note: If the job queue is assigned to a rotation group, the maximum active value is ignored. See the discussion on Job Queue Management, later in this section, for complete information on how Robot Autotune can manage your lower-priority job queues.

Using Dynamic Pools for Saves

If you are going to perform system saves in dynamic pools, the minimum pool size for the dynamic pools must be set at 3 MB or higher. If the backup command is executed in a pool size less than 3 MB, it may terminate abnormally.

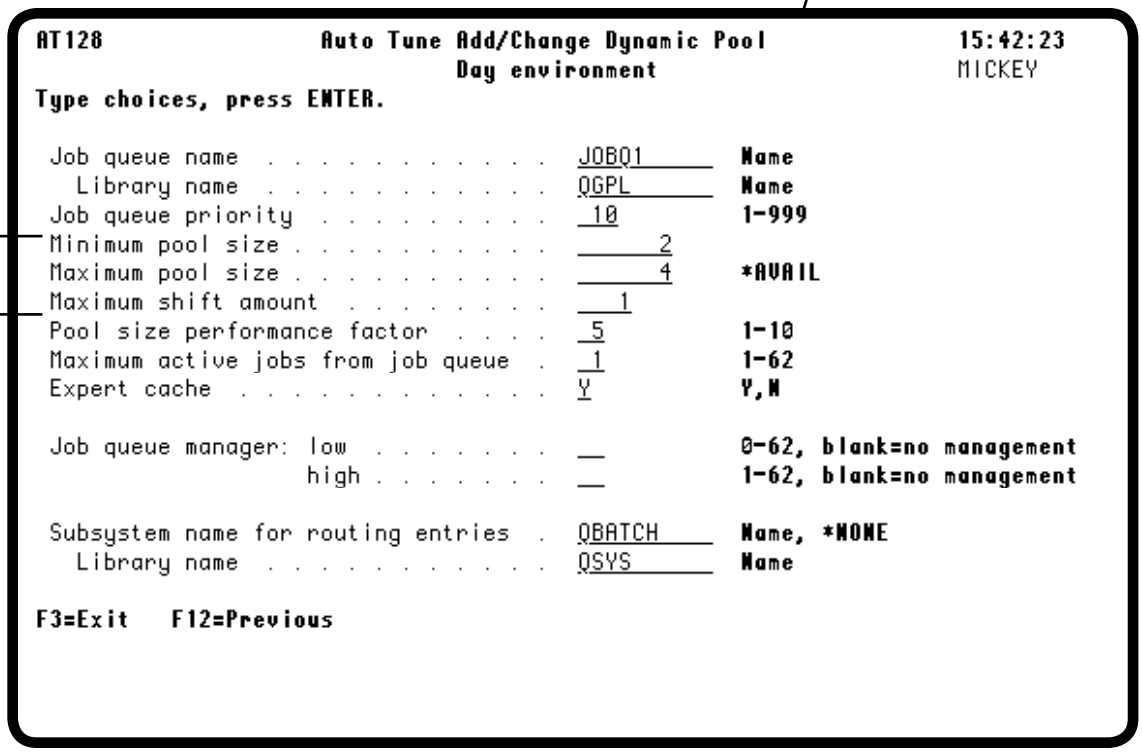
Adding Robot Autotune Dynamic Pools

To begin adding job queues to the Dynamic Pools list, Robot Autotune must be inactive. Select option **6** on the Auto Tune Menu to end Robot Autotune. When Robot Autotune is inactive, select option **2**, to display the Auto Tune Performance Factors - Dynamic Pools panel. **F6** is now enabled; press **F6** to display the Auto Tune Add/Change Dynamic Pool panel.



With Robot Autotune inactive, press **F6** to display the Auto Tune Add/Change Dynamic Pool panel.

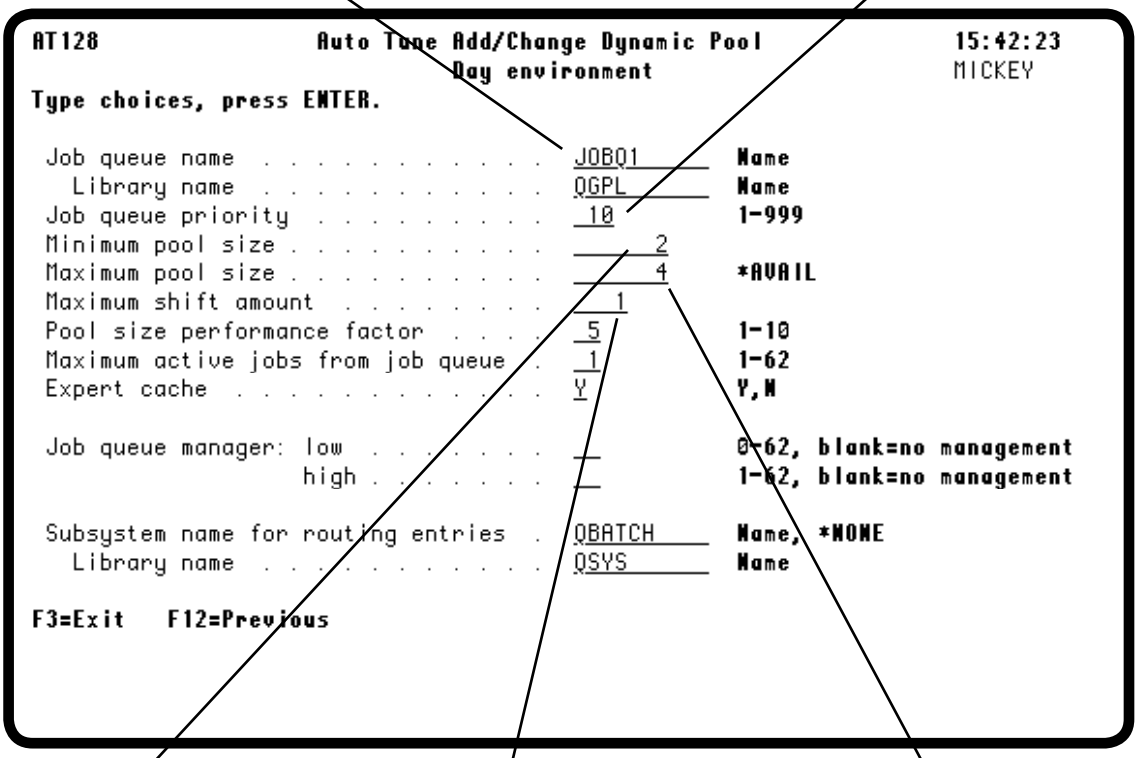
You can specify these values as either a fixed amount (in KB or MB), or as a percentage of the total amount available to the partition.



Entering Dynamic Pools Performance Factors

Enter the name of the job queue (and its library) to be added to the list of dynamic pool job queues.

Enter the priority for this job queue. Jobs on higher priority queues become active before jobs on lower priority queues. The range of priorities is 1 to 999 (with 1 being the highest priority). The priority must be a unique value; different job queues cannot have the same priority.



Enter a minimum pool size. Robot Autotune will start the dynamic pool with this minimum amount.

- This value can be in MB (default), KB (enter a K afterwards), or as a percentage of the total amount available to the system (enter a percent sign [%] after the value).
- The recommended minimum pool size is not less than 1 MB.

Enter a maximum pool size. Robot Autotune will not let the dynamic pool expand beyond this size.

- The recommended starting point for a maximum size for batch jobs is 3 MB. The larger the size of the pool, the faster the throughput. Certain jobs, such as compiles and saves, need a larger amount.
- This value can be in MB (default), KB (enter a K afterwards), or as a percentage of the total amount available to the system (enter a percent sign [%] after the value).

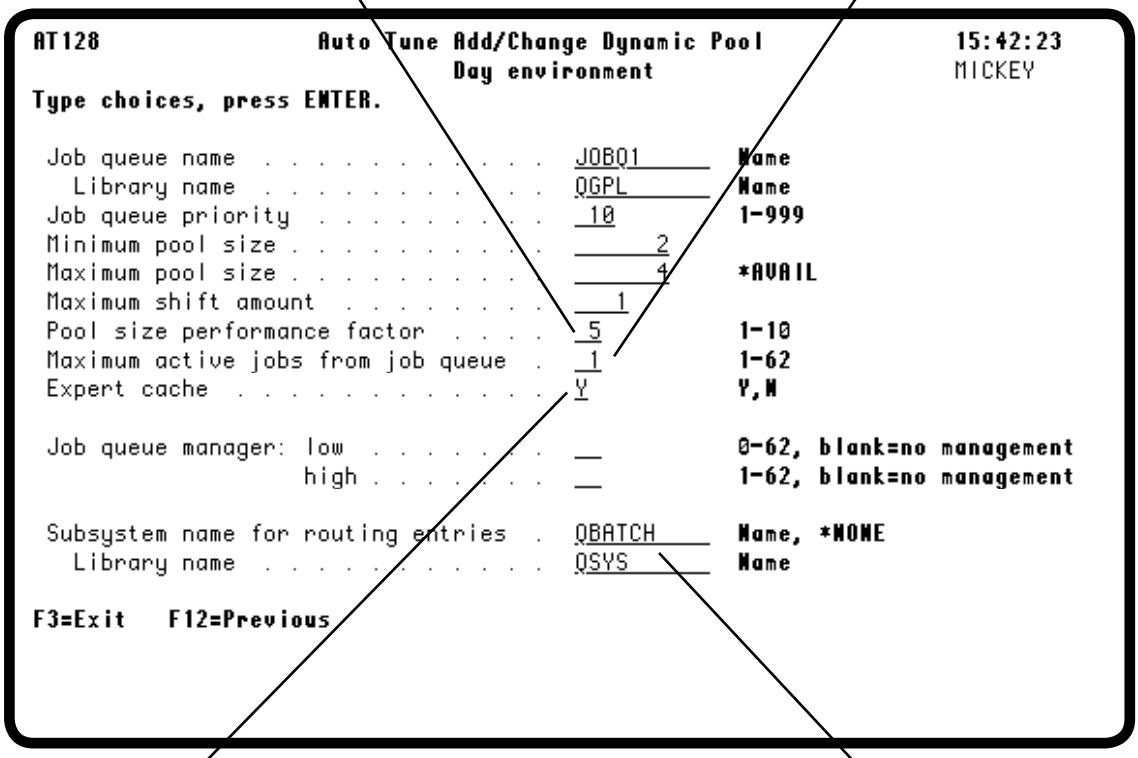
Enter a maximum shift amount. This is the maximum amount of memory Robot Autotune will shift when it makes an adjustment. This value can be in MB (default), KB (enter a K afterwards), or as a percentage of the total amount available to the system (enter a percent sign [%] after the value).

Entering Dynamic Pools Performance Factors

Enter a performance factor to be used to allocate memory. The highest factor is 10. If you want all jobs to have the same priority, give each dynamic pool the same performance factor. These factors are set the same way as the pool size performance factors for standard pools. Turn to the Entering Performance Factors section for more information.

Enter the maximum number of jobs that can be active concurrently from the job queue. This feature allows you to have multiple active jobs from the same queue, each contained in its own dynamic pool.

Note: If you are assigning the job queue to a rotation group (by entering values in the Job queue manager high/low fields), Robot Autotune does not use the value entered in the Maximum active jobs field. See the discussion on Job Queue Management for more information.



Use this field to turn off the OS/400 expert cache feature. The default value is **Y**, which makes expert cache available for the dynamic pool (Paging option=*CALC). However, there may be situations when you don't want to use expert cache. Enter an **N** to turn it off (Paging option=*FIXED).

Enter the name of the subsystem and library that Robot Autotune should refer to for routing entries. When jobs execute in dynamic pools, Robot Autotune uses the routing entry attributes from these subsystems, including time slice and run priority for the class. Therefore, even jobs that require special routing data programs can run in dynamic pools. The default entry is *NONE. Job queues with the routing entry of *NONE use the routing information defined for the ATMONITOR subsystem.

Job Queue Management

Robot Autotune offers an advanced function—job queue management—that works with your dynamic pools. It makes sure that even lightly used job queues have the opportunity to process, which helps all of your jobs to finish processing and your system to run efficiently.

You're probably familiar with the following scenario: jobs in your high-priority job queues are handled easily by the operating system and process quickly. Because jobs are always being added to these job queues, these high-priority jobs continue to process. But, what about your lower-priority job queues? If your system is typical, jobs in those queues just sit and wait with no chance to process. Sometimes, they wait because higher-priority job queues are full.

Robot Autotune's job queue management feature solves that problem. When you look at your job queues, you'll probably find a few that always have jobs in them—these are your high-priority queues. Robot Autotune has always handled those job queues by allocating dynamic pools and processing the jobs quickly and efficiently. These are your job queues that are not managed.

Now look at the rest of your job queues—there may be one for every application running on your iSeries. These job queues have a lower priority and jobs are sent to these queues less often. Take low priority, lightly used job queues and place them in a *rotation group*, and they become your managed job queues. As jobs are placed on a job queue, Robot Autotune allocates dynamic pools to the job queues in the rotation group by determining the number of available dynamic pools not in use. It allocates these dynamic pools to the jobs queues up to the maximum number of dynamic pools defined on the Control Values panel. As each job completes, a job from the next job queue in the rotation group becomes eligible for processing. Robot Autotune cycles through the job queues in the rotation group so that all job queues have the chance to process.

When Should You Use a Rotation Group?

When and how you use a rotation group depends on how busy your job queues are. Two common situations where rotation groups can improve your job processing are:

- You have a few highly used (high priority) job queues and many lightly used (low priority) job queues. Place the lightly used job queues in a rotation group. This is the situation described above.

- You have a number of job queues, all of which have approximately the same activity level. To gain more control over the processing sequence, place all the job queues in a rotation group.

Advantages of Using a Rotation Group

Placing job queues in a rotation group provides several advantages to your job processing. Most importantly, all your jobs will process—no more waiting for hours while higher-priority jobs take all your dynamic pools.

In addition, it can be a real time saver for your operator. Instead of having to watch job queues to see which ones have jobs that are waiting, your operator can just trust Robot Autotune to handle everything. No more placing some job queues on hold so jobs in other job queues can process. And, no more having to move jobs from one job queue to another as you try and juggle your processing priorities.

Assigning Job Queues to a Rotation Group

You assign job queues to a rotation group using the same Dynamic Pools panel you use for defining all your dynamic pools. To add job queues to a rotation group, you specify both a minimum (low) and a maximum (high) number of jobs that can be processed from each queue, up to the total number of available dynamic pools. A low of 0 tells Robot Autotune that when the current job finishes processing, a job in the next job queue in the rotation group becomes eligible for processing.

The high value determines the number of jobs from the job queue that can start processing before Robot Autotune goes on to the next job queue in the rotation group. Specifying a high value of 1 makes the job queue single-threaded; only one job at a time is taken from the job queue. If you want more than one job at a time to be run from a job queue, set the high value to the number of jobs that can run.

Robot Autotune also looks at the priority of each job queue in the rotation group to determine which job will process next. Jobs arriving in job queues with a higher priority are eligible to process before jobs in lower-priority queues. However, when the job in the higher-priority job queue completes, Robot Autotune continues to cycle through the rest of the job queues in the rotation group (in order of priority) until all jobs have the opportunity to process.

Note: A lower number reflects a higher run priority for a job queue.

We recommend that you set all lower-priority job queues to have a low of 0 and a high of 1. This allows one job from each job queue to process and then, when the job completes, the dynamic pool is available to the next job queue in the rotation group. You can set a higher-priority job queue to a low of 1 (and a high of 2 or 3) so that a job in the job queue is eligible to process next, no matter where in the rotation cycle Robot Autotune is. Although the low and high values can be any number up to the maximum number of available pools, we recommend that you limit them to the suggested values.

Implementation Tips

Use the following information to help you set up your rotation group.

- The number of dynamic pools must be greater than the maximum possible number of jobs that are allowed in the job queues that are not in the rotation group (that is, the job queues that have only a maximum active value specified). Otherwise, the job queues that are not in the rotation group will allocate all the dynamic pools and the jobs in the rotation group will never have the opportunity to process.
  Total the number of jobs specified in the Maximum active jobs field from each job queue not in the rotation group. Then, make sure the Maximum number of dynamic pools specified on the Control Values panel is greater than that total. The more dynamic pools that are allowed, the more execution slots will be available to the jobs queues in the rotation group.

- If you specify both a maximum active value and job queue manager high and low values, Robot Autotune normally ignores the maximum active value and treats the job queue as being assigned to the rotation group
  However, if the low and high values in the job queue manager fields are both set to 1, Robot Autotune removes the job queue from the rotation group and treats it as a single-threaded job queue with maximum active value of 1.

- When you set up a rotation group, we recommend that you assign all job queues with a selected priority and lower to the rotation group. This places all job queues from that point to the end of the list of job queues in the rotation group, allowing Robot Autotune to manage them.

Starting to Use Job Queue Management

Robot Autotune's job queue management function is controlled by the ATJOBQMGMT job running in the ATMONITOR subsystem. Before you can start using job queue management, enter the following command on a command line:

```
ATLIB/INZJQM
```

The Initialize Job Queue Management (**INZJQM**) command adds an autostart job entry for the ATJOBQMGMT job to the subsystem description for ATMONITOR. Then, whenever Robot Autotune starts, the ATJOBQMGMT job starts automatically.

Note: The ATMONITOR subsystem must be inactive when you run the INZJQM command because it makes changes to the ATMONITOR subsystem description.

If you later choose to stop using Robot Autotune's job queue manager, enter the Remove Job Queue Management (**RMVJQM**) command on a command line. This reverses the changes made by the INZJQM command to the ATMONITOR subsystem description. Again, ATMONITOR must be inactive to run the RMVJQM command.

If the ATJOBQMGMT job ends while Robot Autotune is active, you can use the Start Job Queue Manager (**STRJQM**) command to restart the job. To end the ATJOBQMGMT job, use the End Job Queue Manager (**ENDJQM**) command. If you don’t use the RMVJQM command to change the ATMONITOR subsystem description, the ATJOBQMGMT job will restart automatically the next time Robot Autotune starts.

Adding a Job Queue to a Rotation Group

When you are defining your dynamic pools, you can add a job queue to a rotation group by entering values in the Job queue management fields. Assigning a job queue to a rotation group allows Robot Autotune to manage lower-priority queues so that jobs in those queues have the opportunity to process.

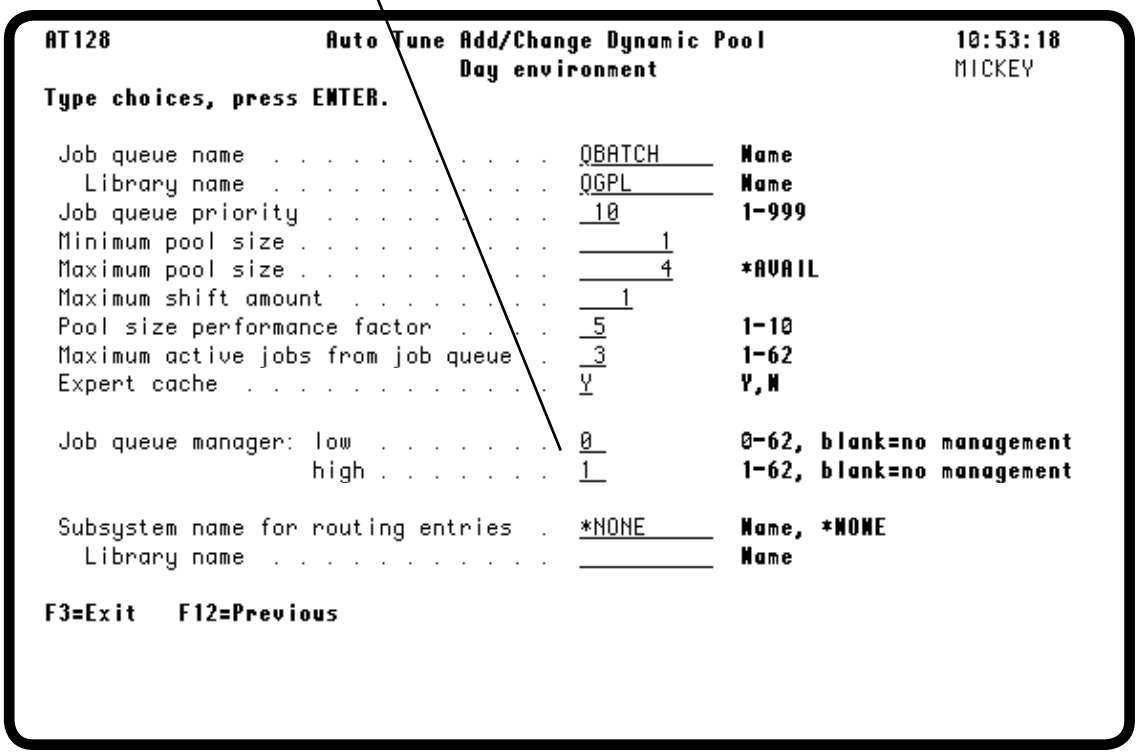
Enter values to assign the job queue to a rotation group. All job queues that are assigned to a rotation group are allocated dynamic pools on a rotating basis so that they have a chance to execute.

- You can enter any value between 0 and 62 in the low field. We recommend that you set the low value to 0 or 1.

  A value of 0 tells Robot Autotune that when the current job finishes processing, a job in the next job queue in the rotation group becomes eligible for processing.

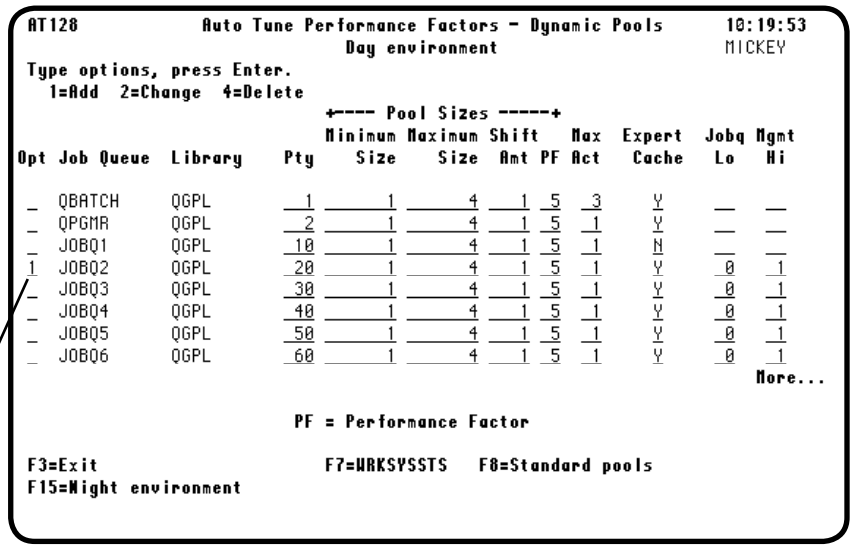
  Specify a value of 1 to make sure that a job in the job queue will always be eligible to process next no matter where Robot Autotune is in the rotation cycle.
- Although you can enter any value between 1 and 62 in the high field, this value should not be more than 2 or 3. The value in the high field tells Robot Autotune how many jobs can start processing from the job queue before it goes on to the next job queue in the rotation group. We recommend using 1 in the high field because a 1 in the high field allows only one job at a time to process from the job queue.

Note: If you enter values in both the Maximum active jobs from job queue field and the Job queue manager field, Robot Autotune ignores the value in the Maximum active jobs from job queue field.

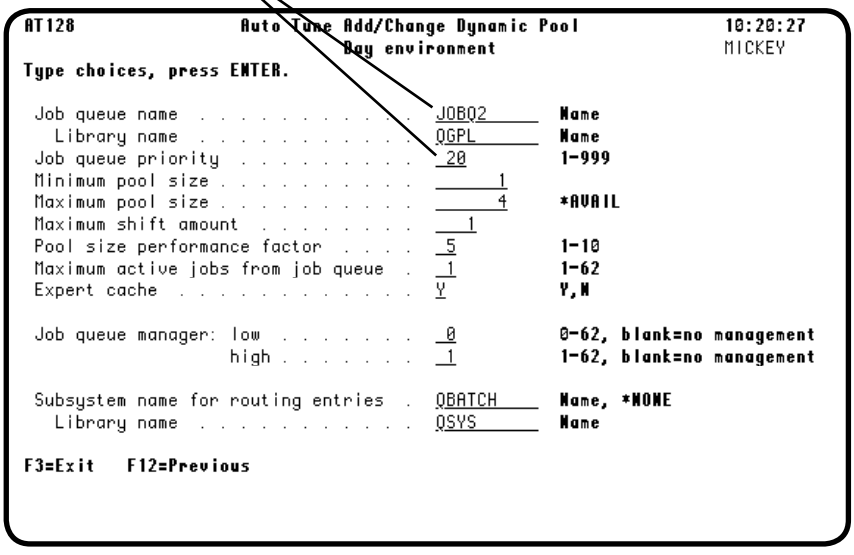


Adding a Job Queue to the Dynamic Pools List

The Auto Tune Performance Factors - Dynamic Pools panel lists all the job queues that currently use dynamic pools. If you have a job queue that is not allocated to another subsystem, you can add it to the list even while Robot Autotune is active. Robot Autotune handles this by copying the information for an existing job queue. Enter a **1** next to an existing job queue that is defined the way you want the new job queue to be defined. Then, change the job queue's name and run priority. Robot Autotune will allocate a dynamic pool to the job queue as soon as a job arrives on the queue.

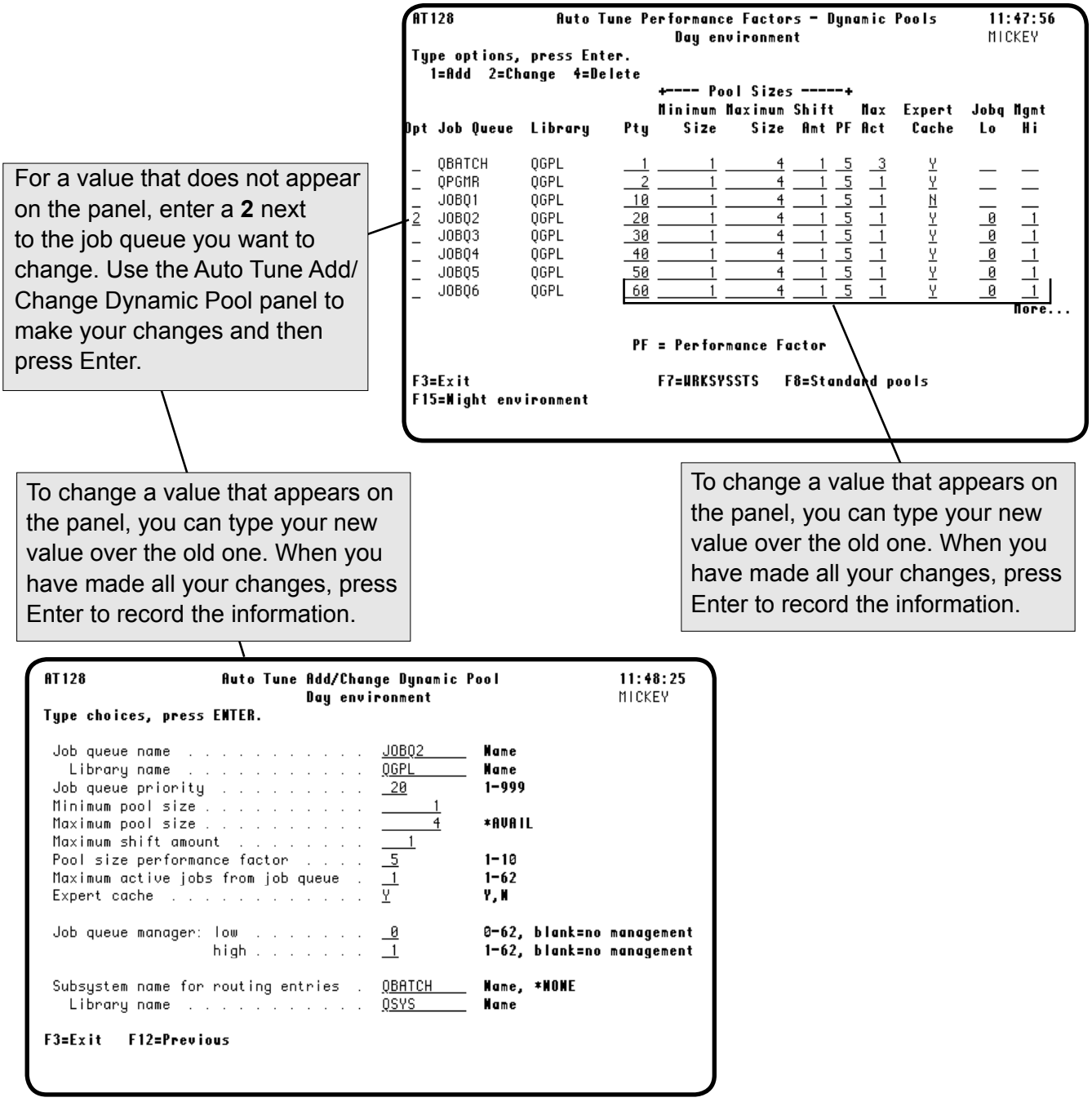


To add a new job queue to Robot Autotune's dynamic pools list, enter a **1** next to an existing job queue. The Auto Tune Add/Change Dynamic Pool panel displays with a copy of the existing job queue information. You must change the name and run priority for the new job queue. You also can make additional changes if you want. Then, press Enter to add the job queue.



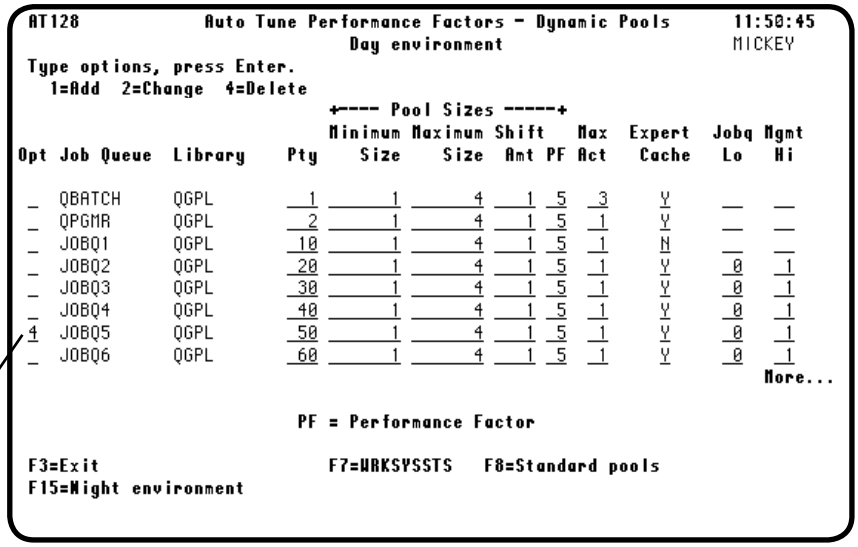
Changing Dynamic Pools Performance Factors

You can change the performance factors for a job queue while Robot Autotune is active. If the values you want to change appear on the Dynamic Pools panel, you can just type in your new values and press Enter to record your changes. If a value does not display on the panel, enter a **2** next to the job queue to display the Auto Tune Add/Change Dynamic Pool panel. Make your changes and press Enter.



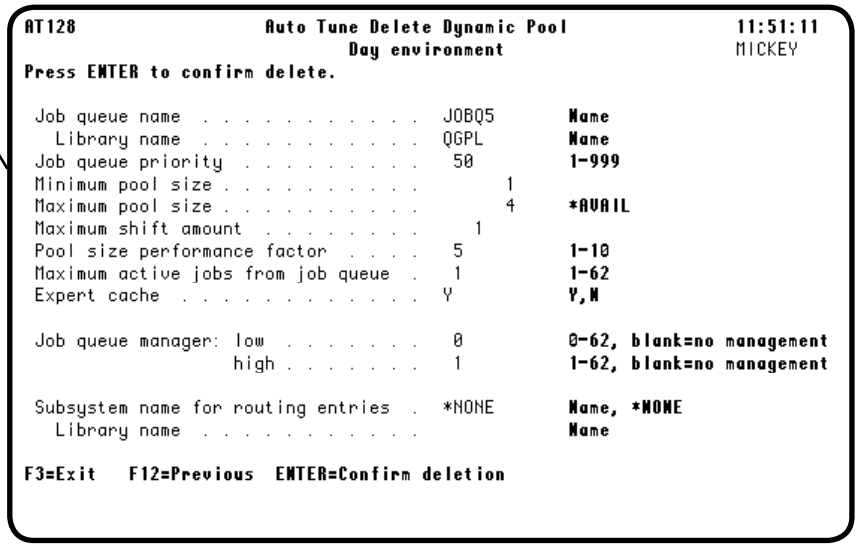
Deleting Dynamic Pools

You can delete a job queue from the list while Robot Autotune is active. If you delete a job queue from Robot Autotune, jobs on the queue still will process; however, they will not be placed in dynamic pools. On the Auto Tune Performance Factors - Dynamic Pools panel, enter a **4** next to the job queue you want to delete. The Auto Tune Delete Dynamic Pool panel displays so you can confirm you want to delete the job queue.



```
AT128          Auto Tune Performance Factors - Dynamic Pools      11:50:45
                          Day environment                         MICKEY
Type options, press Enter.
  1=Add  2=Change  4=Delete
                              +---- Pool Sizes -----+
                              Minimum Maximum Shift    Max  Expert  Jobq Mgmt
Opt Job Queue  Library    Pty    Size    Size  Amt PF Act   Cache   Lo   Hi

 _  QBATCH     QGPL       1       1       4    1  5   3      Y
 _  QPGMR      QGPL       2       1       4    1  5   1      Y
 _  JOBQ1      QGPL      10       1       4    1  5   1      N
 _  JOBQ2      QGPL      20       1       4    1  5   1      Y      0    1
 _  JOBQ3      QGPL      30       1       4    1  5   1      Y      0    1
 _  JOBQ4      QGPL      40       1       4    1  5   1      Y      0    1
 4  JOBQ5      QGPL      50       1       4    1  5   1      Y      0    1
 _  JOBQ6      QGPL      60       1       4    1  5   1      Y      0    1
                                                                   More...

                          PF = Performance Factor

F3=Exit                      F7=WRKSYSSTS   F8=Standard pools
F15=Night environment
```

To delete a job queue from Robot Autotune's dynamic pools, enter a **4** next to the job queue you want to delete. When the Auto Tune Delete Dynamic Pool panel displays, verify you have selected the right job queue and press Enter to confirm the deletion.



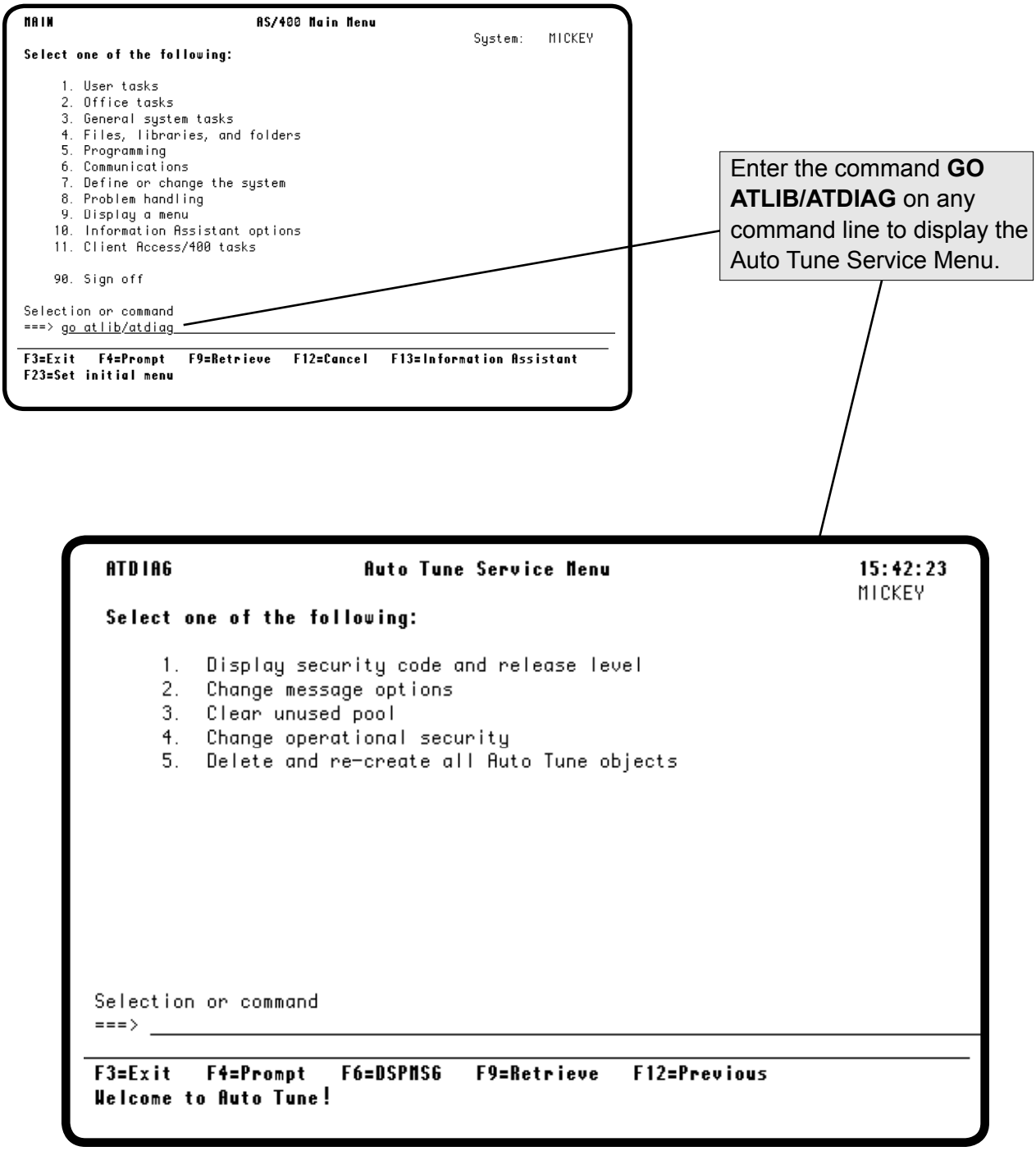
```
AT128                  Auto Tune Delete Dynamic Pool                11:51:11
                          Day environment                         MICKEY
Press ENTER to confirm delete.

 Job queue name . . . . . .  JOBQ5      Name
   Library name . . . . . .  QGPL       Name
 Job queue priority . . . .   50        1-999
 Minimum pool size . . . . .       1
 Maximum pool size . . . . .       4    *AVAIL
 Maximum shift amount . . .     1
 Pool size performance factor .   5      1-10
 Maximum active jobs from job queue .  1  1-62
 Expert cache . . . . . .  Y            Y,N

 Job queue manager: low . . . .   0      0-62, blank=no management
                    high . . . .  1      1-62, blank=no management

 Subsystem name for routing entries .  *NONE     Name, *NONE
   Library name . . . . . .                      Name

F3=Exit   F12=Previous  ENTER=Confirm deletion
```

Notes:

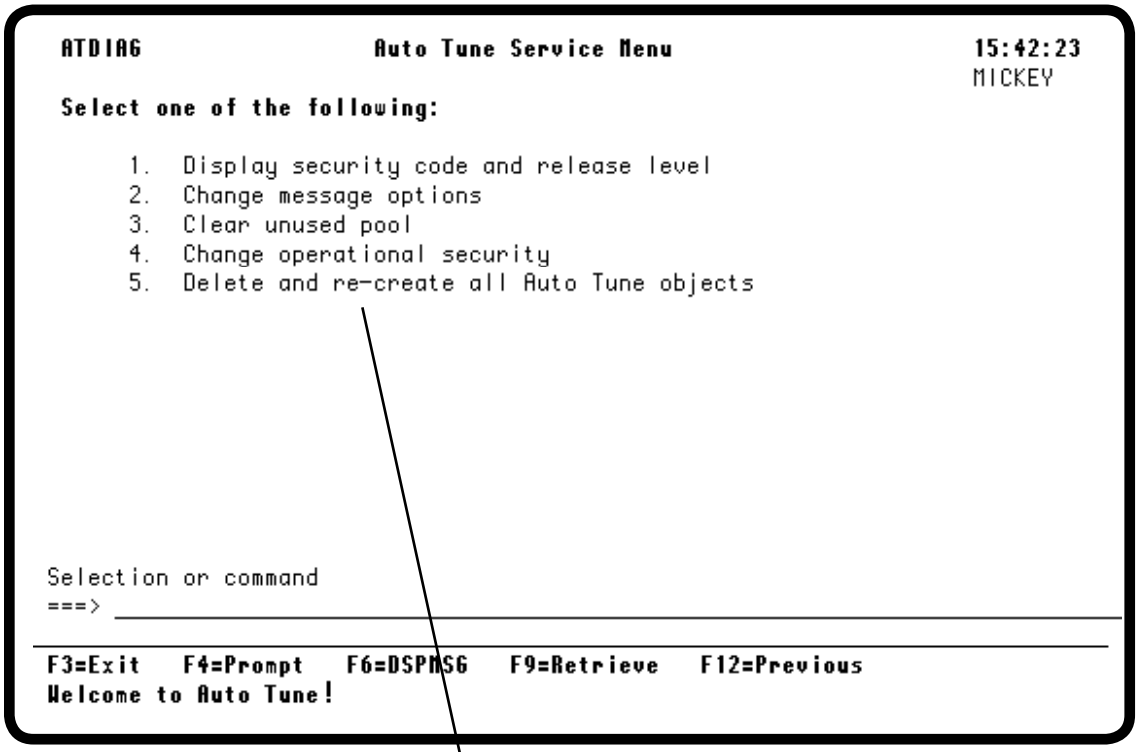
Diagnostics

Accessing the Auto Tune Service Menu

Robot Autotune provides a service menu that offers several diagnostic options. In general, you will not need to use this menu. However, you may be asked to go to the service menu when you call Robot Technical Support. To display the Auto Tune Service Menu, enter the command **GO ATLIB/ATDIAG** on any command line.



Auto Tune Service Menu Options



You can select from five Auto Tune Service Menu options:

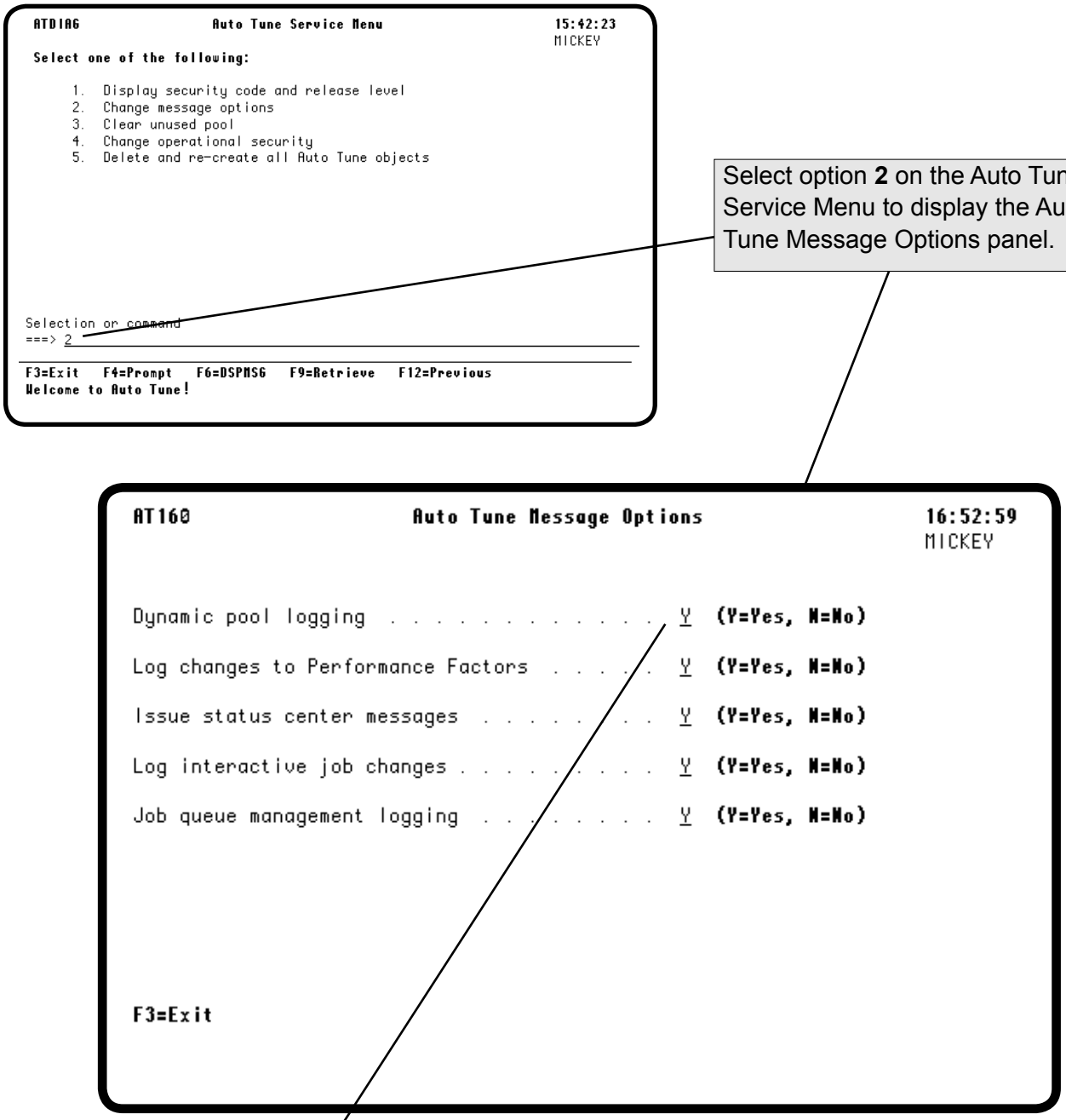
1. **Display security code and release level—**Displays the Auto Tune Security Code panel showing the Robot Autotune Version/Release level you are running, your system serial number, your system model number, and your Robot Autotune security code.
2. **Change message options—**Displays the Auto Tune Message Options panel so you can change logging and status message options. See the discussion on the next page.
3. **Clear unused pool—**Displays the Clear Orphan Pool panel. Select this option to clear pools not being used by the system. You can enter the pool ID when the panel displays.
4. **Change operational security—**If you are signed on as the security officer, you can use this option to change Robot Autotune security levels:

   ***EXCLUDE** Users cannot use Auto Tune Menu options 1-4.

   ***USE** Users can use Auto Tune Menu options 1-4, but cannot change values.

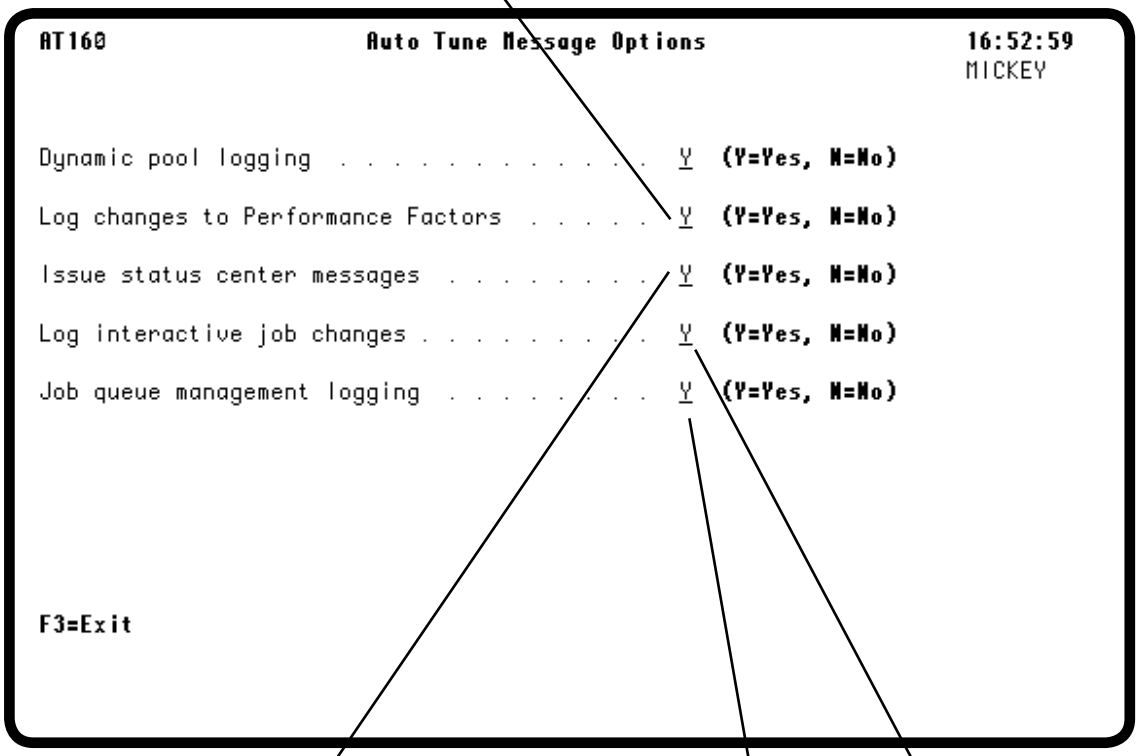
   ***CHANGE** Users can use Auto Tune Menu options 1-4, and can change any values.
5. **Delete and re-create all Auto Tune objects—**Select this option to reset Robot Autotune to its original settings as shipped from Fortra.

Changing Auto Tune Message Options



Changing Auto Tune Message Options

Enter **Y** to log changes in performance factors and control values to the system audit journal (if it is used on your system), or to the system history log. An entry is made each time a performance factor is changed, showing the pool affected. In addition, overrides that are added or removed are listed. Each control value update is noted.



Enter **Y** to have Robot Autotune send messages to the Robot NETWORK Status Center. Robot Autotune sends a message to the Status Center when the following events occur:

Event	**Message Type**
ATMONITOR starts	Information [IN] message
Interactive or server job is tuned	Information [IN] message
ATMONITOR ends normally	Warning [WR] message
An extra or missing pool is found	Attention [AT] message
Your security code expires	Attention [AT] message
ATMONITOR ends abnormally resulting in a function check	Attention [AT] message

Enter **Y** to place messages in the ATMONITOR job log about interactive and server jobs that had their run priority or time slice adjusted.

Enter **Y** to place messages in the ATJOBQMGMT job log about changes made to job queues that are assigned to a rotation group.

Notes:

Getting Your iSeries Into Tunable Condition

■ Introduction

If you are running your iSeries as shipped by IBM, QBASE is the controlling subsystem. The *SPOOL shared pool is used for your printer jobs, the shared pool *INTERACT is used for your interactive jobs, and the batch subsystem runs its jobs in the *BASE pool.

You can retain the shared pools for the printer and interactive jobs, and Robot Autotune handles them like any other pool. You should change the QBATCH subsystem to run in a pool other than *BASE. Follow the steps below to set up Robot Autotune and get your system into a tunable condition.

1. Modify Your Existing IPL Startup Program

Enter the following sample IPL startup program in QCLSRC and compile it in QGPL.

```
PGM              /* PUT THIS PROGRAM SOURCE IN QCLSRC */
   MONMSG (CPF0000 ATI0035)
  ATLIB/STRAT                          /* START AUTO TUNE */
  /* IF YOU HAVE ROBOT */
  STRSBS RBTSYSLIB/RBTSLEEPER          /* START ROBOT */
  STRSBS QINTER        /* START LOCAL INTERACTIVE SBS */
  STRSBS QBATCH                        /* START BATCH SBS */
  STRSBS QSPL                         /* START WRITERS SBS */
  STRSBS QSNADS          /* FOR INTEROFFICE MESSAGES */
  STRSBS QCMN                    /* NEEDED FOR ECS LINE */
  /* ENTER ANY OTHER SUBSYSTEM STARTUPS */
  STRPRTWTR PRT01                       /* START PRINTER */

ENDPGM
```

This program is used to start the individual subsystems that you will modify in the following steps.

2. Terminate QBATCH

At the system console, enter the following command:

```
ENDSBS SBS (QBATCH) OPTION (*IMMED)
```

3. Set Up a Pool for Batch Jobs

a. Add a pool to the QBATCH subsystem description using the Change Subsystem Description command. Use the following command to define *SHRPOOL1 to the subsystem description:

```
CHGSBSD SBSD(QGPL/QBATCH) POOLS ((2 *SHRPOOL1))
MAXJOBS(1)
```

If you want to run more than one job at a time in QBATCH, increase the maximum active jobs parameter (MAXJOBS). If you need to change the pool definition, use the CHGSHRPOOL command.

b. Change the routing entries within the subsystem so that all jobs using this routing entry execute in the new pool. To find all the routing entries for a subsystem, enter the command **DSPSBSD** (*subsystem name*). Then, take option **7** to display routing entries. Write down the sequence number for each routing entry.

   To see the pool number that the routing entry uses, enter a **5** in front of the routing entry and press Enter. Enter the following command for each routing entry sequence number in this subsystem:

   CHGRTGE SBSD(QGPL/QBATCH) SEQNBR(9999) POOLID(2)

   Note: Subsystem routing entries direct your jobs to the proper pool number. We used 9999 as our sample routing sequence number in the commands in the following pages. You must change all routing entries for each subsystem to use the new pool set up for that subsystem.

c. Use the CHGSYSVAL command to change the system value QTSEPOOL (time slice and pool) to *NONE.

4. Start the Subsystem

Enter the following command to start the batch subsystem:

STRSBS SBS(QBATCH)

5. Reset System Values (if necessary)

The system values QDYNPTYSCD (dynamic priority scheduler) and QDYNPTYADJ (dynamic printer adjustment) default to 1 (on). Leave these values as shipped. If you have changed either value, use the CHGSYSVAL command to reset them to their original (default) value.

6. Reset Robot Autotune

Change the Auto Tune Performance Factors. This causes Robot Autotune to recognize the new pools. Enter the following command to display the Auto Tune Menu:

ATLIB/ATM

If this is the first time you are entering performance factors, select option **1** to display the Auto Setup panel. To use Auto Setup at other times, select option **4**.

You are ready to performance tune your iSeries.

Setting Up Subsystems and Pools for Maximum Performance

After you are satisfied with the way Robot Autotune manages your present setup, you may want to try changing your subsystems for maximum performance and efficiency. The changes described below are the result of running Robot Autotune on many different iSeries systems.

Because Robot Autotune reduces an unused pool to its minimum size, you are not penalized for setting up multiple pools and subsystems on your system. Multiple pools are practical only if you are using Robot Autotune. Robot Autotune will work without these separate pools, but it works best if you set them up.

Note: The maximum number of pools allowed is 64.

*BASE

There should be no subsystem jobs running in *BASE. This pool should be reserved for OS functions. The only exception is if you have a subsystem containing sleeper jobs (jobs that wake up periodically to do their work, such as QCMN, QSNADS, or Robot Schedule).

Shared Pools

If there is a requirement that workstations be assigned to separate subsystems, but are using the same files and programs, consider assigning a shared pool to the subsystems. The pool ,*INTERACT, can be assigned a memory size and activity level and placed in the subsystems pool definition.

Workstation Pools

Workstations that do dissimilar work should be in different pools. You can use Robot Autotune to give different classes of workstations different priorities.

Remote Workstations and Pools

Remote workstations are a good example of when you might need a separate pool. Users have had surprisingly good results when these workstations are in their own subsystem. The remote workstations tend to follow the same pattern of usage and function better in their own separate pool. Robot Autotune will change the pool sizes during the day to match their usage pattern.

Printer Pools

All printer jobs should be in one pool.

Communication Pools

Depending on how much you use communication subsystems (QSNADS, QSERVER, QSYSWRK, QUSRWRK, or QCMN), you may want to set these up in separate pools. We recommend that you do so if these subsystems are used heavily.

Giving an Existing Subsystem Its Own Pool

If you have the typical setup with the subsystem QBATCH running in *BASE pool, the following commands will put the subsystem in its own pool. Better yet, use Robot Autotune's dynamic pools to execute your batch jobs.

1. End the Robot Autotune monitor.
2. Enter the following commands:

   CHGSBSD SBSD(QBATCH) POOLS((*n* *SHRPOOL1 1))

   where *n* is the first unused subsystem pool number.

   CHGRTGE SBSD(QBATCH) SEQNBR(9999) POOLID(*n*)

   Check your routing entries for the correct sequence number. Use the pool ID (*n*) established above.
3. Enter the **CHGSHRPOOL** command and specify a storage size for the shared pool.
4. Start the subsystem with the command: **STRSBS QBATCH**
5. Go to the Performance Factors panel and enter the performance factors for the new pool.
6. Start Robot Autotune.

Domino, WebSphere, and WebAccess Applications

If you are using Domino, WebSphere, or WebAccess applications in any of your subsystems, we recommend that you configure these applications to run in their own shared memory pool.

Creating Another Batch Subsystem With Its Own Pool

It's a good idea to set up a separate batch pool for programmers to use for their compilations (compiles). Compiling programs uses large amounts of memory, so by increasing the maximum pool size to 25 MB or larger for the compilation pool, compilations will complete quickly, letting you go home earlier.

Normally, you would use dynamic pools to execute your batch jobs. But, as an example, let's say you wanted to set up a separate batch subsystem. These are the steps you'd follow to create a new batch subsystem that runs in its own pool. We used the subsystem name COMPILE, but you can use any name you want.

1. End the Robot Autotune monitor.

2. Enter the following commands:

   CRTSBSD SBSD(COMPILE) POOLS((*n* *SHRPOOL2 1) (1 *BASE)) +

   MAXJOBS(1)

   where *n* is the first unused subsystem pool number.

   CRTCLS CLS(COMPILE)

   CRTJOBQ JOBQ(COMPILE)

   ADDJOBQE SBSD(COMPILE) JOBQ(COMPILE)

   ADDRTGE SBSD(COMPILE) SEQNBR(9999) CMPVAL(*ANY) +

   PGM(QSYS/QCMD) POOLID(*n*)

   Use the pool ID (*n*) established above.

3. Change the job descriptions that your programmers use to submit compilations so they use the new job queue.

   CHGJOBD JOBD(QPGMR) JOBQ(COMPILE)

4. Enter the **CHGSHRPOOL** command and specify a storage size for the shared pool.

5. Start the subsystem with the command, **STRSBS COMPILE**.

6. Put the startup command in your IPL procedures.

7. Go to the Performance Factors panel. Enter the performance factors for the new pool.

8. Start the Robot Autotune monitor.

Appendix

ATD0201

Message: Subsystem &1 pool &2 not identified to Auto Tune.

Cause: The subsystem or pool was not identified on the Performance Factors panel either because the pool is a new pool or the library containing the subsystem description was not in the Robot Autotune library search list.

Recovery: Go to the Performance Factors panel. Press **F9**. Add the library to the search list and press Enter. Press Enter on the Performance Factors panel to record the change, then restart the Robot Autotune monitor.

ATD0203

Message: Unidentified pools or Performance Factors not completely entered.

Cause: Subsystem descriptions were changed without resetting the Performance Factors, or the premature use of **F3** on the Performance Factors panel. The monitor quits running after unidentified pools are found.

Recovery: Go to the Performance Factors panels (option 1 on the Auto Tune Menu for standard pools and option 2 for Dynamic Pools). If all the information is listed, press Enter to make Robot Autotune recognize it. If additional information or changes are needed, type them in and press Enter.

ATD0209

Message: AT pools do not match OS pools, cause &1.

Cause 1: AT &3 pool exists that OS is not aware of at &2.

Cause 2: OS pool &3 exists that AT is not aware of at &2.

Action Taken: Pools reset to OS sizes, Robot Autotune monitor realigned with existing pools and restarted.

ATD0210

Message: Pool &3 changed by non-AT function. Current: &1 Old: &2

Cause: This message is usually caused when a start or end subsystem command is issued while ATMONITOR is active.

Recovery: No action is required. If this message occurs frequently, you should consider managing your subsystems differently. For example, you might want to use a shared pool for allocating and removing memory, rather than starting and ending subsystems.

ATD0231

Message: Dynamic pool not created.

Cause: Occurs when a dynamic pool is requested but could not be allocated. This can happen for one of the three reasons shown in the second level message text:

1.) All 64 system pools are allocated, so no pools are available

2.) Insufficient storage

3.) Job queue not on dynamic pool panel.

Recovery 1: Change the maximum number of dynamic pools permitted to a lower number.

Recovery 2: Lower the minimum pool sizes for other pools.

Recovery 3: Add the job queue to the dynamic pools panel so Robot Autotune recogizes it.

ATD0248

Message: One or more pools reset to system definition.

Cause: You asked Robot Autotune to reset pool sizes to the sizes defined through OS/400 and it was unable to do so. This can happen for one of two reasons:

1. One or more dynamic pools existed at the time so it was impossible to set pools to the defined sizes

2. A prior error occurred that indicates a subsystem description for a pool being reset was not changed.

Recovery: Both of these conditions may be normal and require no action on your part. If you suspect the second reason listed, look at your prior messages to pinpoint the cause of this message and determine whether additional action is needed.

ATD0431

Message: Resource Request Received
Cause: A normal message indicating that a non-Robot Autotune request to change a memory pool allocation or size (such as an STRSBS or ENDSBS command) has occurred.

ATD0437

Message: Resource Request Completion Received
Cause: A normal message indicating that Robot Autotune acknowledges that a pool's size or allocation status has changed.

ATD0438

Message: Resource Request Completion Not Received

Cause: An error message indicating that a pool's size or allocation status did not change in the time allowed and Robot Autotune has stopped.

Recovery: None. The ATMONITOR job ended and created a dump. You should collect this dump and contact Robot Technical Support.

ATI0030

Message: AT Monitor shutdown has been requested but did not respond within time limit.

Cause: The ATMONITOR job may be in an abnormal state or has taken longer than 8 seconds to respond due to heavy system load. The ATMONITOR subsystem was terminated *CNTRLD.

ATI0036

Message: ATMONITOR not started with STRAT command.

Cause: Someone attempted to start the monitor using a command other than STRAT.

Recovery: Use the STRAT command to start the monitor or select the start option from the Auto Tune Menu.

CPF3372

Message: Cannot allocate job queue &4 in library &5.

Note: This message can be found in two places:
1. If starting QBATCH, or in the QBATCH job log, it indicates Robot Autotune has allocated the queue to a control job for dynamic pools and is a normal message.

2. If starting ATMONITOR, or in the ATMONITOR job log, it indicates Robot Autotunecould not control the jobs for dynamic pools (see Recovery).

Recovery: Start Robot Autotune before starting any other subsystem that has the job queue defined to it.

Symbols

A

B

C

D

E

H

I

J

M

O

P

Q

R

S

T

U

W